

THESIS SUBMITTED TO THE
UNIVERSITY OF MOHAMED BOUDIAF – MSILA



FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE

MASTER THESIS

In partial fulfillment of the requirements for the degree of

Master in : Computer science

Specialty : System d'information Et Genie Logiciel (SIGL)

By :

Amira Lamrour

khadidja Denidni

Title of the thesis

Skin Diseases Detection From Images

Under the supervision of

Mahdjoubi Roussafi

Composition of the jury

M. **Hamani Said**

Lecturer A

University of Msila

President

M. **Ghemougui Abdessatter**

Lecturer A

University of Msila

Examiner

2024/2025

إهداء

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ قَالَ تَعَالَى : یَرْفَعُ اللّٰهُ الَّذِیْنَ آمَنُوا مِنْكُمْ وَالَّذِیْنَ أُوتُوا
الْعِلْمَ دَرَجَاتٍ

أهدی هذا النجّاح إلی من أُحمِل اسمہ بِكُلِّ فخرٍ إلی من حصد الأشواک عن دربی
لیمهد لی طریق العلم الی من کلل اللّٰه بالهیبة والوقار .. الی من علمنی العطاء
بدون انتظار إلی أبی الغالی إلی من کاذ دعاؤها سر نجّاحی ، إلی من حاکت
سعادتی بخيوط منسوجة من قلبها .. إلی بسمّة الحیاة وسر الوجود... إلی أغلی
الحبایب أُمی

إلی أهلی واخوتی

لمرور امیرة

إهداء

وآخر دعواهم أن الحمد لله رب العالمين الحمد لله على التمام، الحمد لله على الكمال.

قال تعالى: وأما بنعمة ربك فحدث، سأحدثكم عن إحدى أعظم النعم، ألا وهي نعمة البصر. بفضل هذا البحث، تمكنت من إدراك هذه النعمة وفضلها. الحمد لله حمداً كثيراً طيباً مباركاً فيه.

قال تعالى: لئن شكرتم لأزيدنكم، وإني أشكر نعمتك التي أنعمت بها علي يا الله.

إلى نفسي

قال ابن تيمية: ما لا يكون بالله لا يكون، وما لا يكون لله لا ينفذ ولا يدوم. وقال الإمام مالك: ما كان لله دام واتصل، وما كان لغيره انقطع وانفصل.

إلى عائلتي

أهدي ثمرة جهدي المتواضع إلى من وهبوني الحياة والأمل، والنشأة على شغف الاطلاع والمعرفة، ومن علموني أن أرتقي سلم الحياة بحكمة وصبر؛ برا، وإحساناً، ووفاءً لهما: والدي العزيز، والوالدي العزيزة.

إلى من وهبني الله نعمة وجودهم في حياتي، إلى العقد المتين الذين كانوا عوناً لي في رحلة بحثي: إخواني وأخواتي.

إلى من كاتفني ونحن نشق الطريق معا نحو النجاح في مسيرتنا العلمية، إلى رفيق دربي: طيب جمال عزيز.

وأخيراً إلى كل من ساعدني، وكان له دور من قريب أو بعيد في إتمام هذه الدراسة، سائلاً المولى أن يجزي الجميع خير الجزاء في الدنيا والآخرة.

ثم إلى كل طالب علم سعى بعلمه ليفيد الإسلام والمسلمين بكل ما أعطاه الله من علم ومعرفة.

خديجة دنيدي

Acknowledgments

We would like to express our deepest gratitude and sincere appreciation to all the individuals and institutions that have contributed to the successful completion of this thesis.

First and foremost, we extend our heartfelt thanks to our thesis advisors **Dr. Mahdjoubi RoUssafi** for their invaluable guidance and unwavering support throughout this research journey. Their expertise, constructive feedback, and dedication to our academic growth have been instrumental in shaping this work. We are truly grateful for their mentorship and encouragement.

We also wish to express our appreciation to the faculty members of the Informatics Department at Msila University, whose knowledge and dedication to teaching and research have greatly influenced our intellectual development. Their passion for education has been a constant source of inspiration. Furthermore, we sincerely thank the participants of this study for their willingness to take part in our research. Their cooperation and valuable insights have enriched the findings of this thesis.

Finally, we extend our gratitude to everyone who has contributed, directly or indirectly, to this work—whether through discussions, feedback, or encouragement. Your support has played a vital role in bringing this research to fruition.

To all those mentioned above, and to anyone whose name may have been inadvertently omitted, please accept our heartfelt thanks. Your contributions have been invaluable in making this thesis a reality.

ملخص

تُمثل أمراض الجلد تحدياً كبيراً للصحة العامة، إذ يصعب تشخيص بعضها بدقة في مرحلة مبكرة. وقد تكون تقنيات التشخيص التقليدية، مثل الفحص البصري، عرضة للأخطاء وتستغرق وقتاً طويلاً. في السنوات الأخيرة، شهدنا تطورات ملحوظة في مجال الذكاء الاصطناعي، وخاصةً التعلم العميق، مما فتح آفاقاً جديدة في المجال الطبي. في هذه الدراسة، نقترح نظاماً قائماً على التعلم العميق للكشف المبكر عن أمراض الجلد من خلال تحليل صور الجلد.

الكلمات المفتاحية: الأمراض الجلدية ، التعلم الآلي ، التعلم العميق، الرعاية الصحية، المعلوماتية الصحي.

Abstract

Many skin diseases are a big problem for public health since it can be hard to identify them early on. Some conventional diagnosis methods such as inspecting a patient, are not always perfect and may take a long time. Recent years have seen impressive developments in artificial intelligence, especially in deep learning, helping to bring about new possibilities in medicine. In our study, a deep learning system is suggested to assist in spotting skin diseases early by studying skin images.

Keywords: Skin diseases, Machine learning, Deep learning, Healthcare, Health informatics.

Contents

List of Figures	i
General Introduction	2
1 Skin Diseases SD	8
1.1 Introduction	8
1.2 Skin Diseases	8
1.3 Types of Skin Diseases	9
1.3.1 Inflammatory Skin Diseases	9
1.3.2 Infectious Skin Diseases	9
1.3.3 Chronic Skin Diseases	10
1.3.4 Genetic Skin Diseases	11
1.4 Skin Structure	11
1.5 Common Skin Diseases	12
1.6 Chronic Skin Diseases	13
1.7 Skin Diagnosis Techniques	14
1.7.1 Clinic Examination	14
1.7.2 Skin Biopsy	15
1.7.3 Light Microscopy Examination	15
1.7.4 UV imaging	18
1.8 Classification of Skin Diseases	19
1.9 Conclusion	21
2 Machine Learning (ML) and Deep Learning (DL)	23
2.1 Introduction	23
2.2 Artificial Intelligence (AI)	23
2.3 Machine Learning (ML)	24
2.3.1 Types of Machine Learning	25
2.4 Deep Learning (DL)	28

2.4.1	Artificial Neural Networks (ANN)	28
2.4.2	Deep Neural Network architectures	34
2.5	Conclusion	37
3	Case Analysis	39
3.1	Introduction	39
3.2	Related Works	39
3.2.1	A Method of Skin Disease Detection Using Image Processing and Machine Learning	39
3.2.2	Leaf and Skin Disease Detection Using Image Processing	40
3.2.3	A Image Analysis System to Detect Skin Diseases (SD)	41
3.3	Case Study	42
3.4	System Design	43
3.4.1	CNN Architecture	43
3.4.2	Transfer Learning	45
3.5	Conclusion	47
4	Implementation and Results	49
4.1	Introduction	49
4.2	Implementation frameworks and tools	49
4.2.1	Python	49
4.2.2	Matplotlib	50
4.2.3	Tensorflow	50
4.2.4	PyTorch	51
4.2.5	Keras	51
4.2.6	Kaggle	51
4.2.7	Google Colab	51
4.2.8	Cross Validation	52
4.2.9	Model evaluation metrics	52
4.2.10	Second dataset	58
4.3	Application	65
4.3.1	Tools Used	65
4.4	Conclusion	65
	General Conclusion	67
	Bibliography	69

List of Figures

1.1	Images of inflammatory skin diseases	9
1.2	Images of Rosacea and Vitiligo and Hidradenitis Suppurativa	11
1.3	skin stucture	12
1.4	Images of eczema and Psoriasis and acne Rosacea and Hidradenitis Suppurativa Vitiligo	14
2.1	AI Vs ML Vs DS	24
2.2	Machine learning design	25
2.3	Supervised learning example	26
2.4	Unsupervised learning example	27
2.5	Reinforcement learning in dog training	28
2.6	Biological neuron structure	29
2.7	Biological and artificial neuron design	31
2.8	Artificial neural networks architecture	33
2.9	CNN architecture for visual recognition	35
2.10	ImageNet Top-1 accuracy CNN models	35
2.11	RNN architecture	36
3.1	Flow chart of Disease Detection System	41
3.2	Architecture of proposed method	42
3.3	Handheld Remidio On Phone and iExaminer Allyn	43
3.4	CNN Architecture	44
3.5	ResNet 50	45
3.6	Dense-Net	46
3.7	Efficient-Net	46
4.1	python version used in colab.	50
4.2	Examples of Matplotlib works.	50
4.3	Data Loader class	54

4.4	Simple CNN Architecture	54
4.5	Training Loop	55
4.6	Validation Loop	55
4.7	Result	56
4.8	Model Architecture	56
4.9	Transfer Learning Approach	57
4.10	Transfer Learning Approach	57
4.11	Data Visualisation	58
4.12	Changing Weights	59
4.13	Result Of Training with Simple CNN	60
4.14	Result Transfer Learning	60
4.15	Combine Model	61
4.16	Left Model	62
4.17	Right Model	62
4.18	Combined Model	62
4.19	Confusion Matrix	63

Introduction

General Introduction

Many people around the world are impacted by skin diseases (SD) and these conditions greatly affect their quality of life. Some of these conditions are acne which is common and easy to treat, while others are psoriasis and various types of skin cancer which are more severe and hard to manage. It is important to get the right diagnosis on time to prevent problems and make the right treatment decisions, but getting this sort of dermatological care is not possible for many in poorly resourced regions.

Because of quick developments in AI and DL, there have been important shifts in methods and approaches used to address this problem. Deep Learning DL models known as Convolutional Neural Networks (CNNs) have demonstrated impressive results on different imaging tasks, especially within medical imaging. CNNs are suggested for extracting different layers of features from pictures of the eye and have high accuracy in spotting the major reasons for DR in them.

Creating a strong DL model for categorizing skin diseases generally requires lots of annotated data which is hard to find in medicine. At this point, transfer learning turns out to be very beneficial. Such pre-trained models are known as transfer learning models, where we update them using a specific task's dataset (often less extensive compared to what they were trained on) to improve performance on that task. Adjusting these complex models enables them to accurately find and identify different types of skin diseases from images.

Image processing helps greatly in creating computerized systems for spotting skin diseases. We can use computer vision to help physicians give better diagnoses and offer effective treatments sooner which support better patient results.

For our research, we examine and use a deep learning process to classify skin conditions seen in photos. Beginning with the important skin diseases and their clinical role, the authors also tell how deep learning techniques can identify various images. In the next step, a solution is offered by training a tool to classify skin diseases from the pictures it sees. The steps during implementation include preprocessing data, creating a model, running training and looking at the outcomes.

Eventually, the model's results are assessed to find out how effective it is, determining its strengths, weaknesses and future direction.

Context

Artificial intelligence has been growing quickly and deep learning is one of the central tools used now in healthcare. A major area where these technologies make a difference is medical image analysis which greatly assists with making diagnoses and decisions. It is difficult to accurately spot skin diseases with images because there are many varieties, some appear alike and human errors or delays can arise, particularly where medical facilities are not well equipped or close by.

Considering the skin is the largest part of the body, paying attention to it may point out potential health problems warning signs. Because smartphones with advanced cameras are widely accessible, photos of skin can now be collected in large numbers which forms a strong dataset for training deep models. In other words, this study is focused on using artificial intelligence to make a system that can automatically diagnose skin diseases, speed up and improve how diagnoses are done and enable access to health services for more people.

Problem

Skin diseases are very common and they lead to a high number of visits to healthcare providers everywhere. They can be simple and disappear quickly such as acne or can develop into more difficult and persistent conditions such as psoriasis or eczema. There are situations where these become severe systemic-related disorders. It is very important to find and treat skin diseases early to avoid complications and make the patient's life better.

But many times, diagnosing is done with hand-held medical tools or visual examinations which requires skilled medical team members. Since there are not many dermatologists, especially in rural or poor regions, many health problems go without adequate diagnosis for too long which worsens their management. Sometimes, factors such as differences in skin shade, lighting and the quality of images can still lower the accuracy of diagnoses, no matter what the technology is.

With great improvements in artificial intelligence and deep learning, computer vision technology, especially Convolutional Neural Networks (CNNs), can now play a bigger role in examining medical images. They have been able to perform well in the areas of detecting tumors, diagnosing eyes and imaging which makes them useful in dermatology

as well.

Even so, there are still some difficulties in making an effective automatic system for diagnosing skin diseases, including:

- The importance of datasets that are large, high quality and include people of many ages, ethnicities and skin tones.
- Needing to preprocess and improve an image so that lighting and noise from the environment have less of an effect.
- The difficulty in distinguishing between visually similar skin conditions.
- To make sure models are useful in clinical practice, models need to be checked against dependable performance metrics.

These difficulties form the basis for our research question: What are the steps to make an effective and accurate model for detecting skin diseases by deep learning and image processing techniques, handling data that is both diverse and must be preprocessed, so it can assist in diagnostic procedures and enhance patient care?

Scope and Objectives

It looks at making a system that automatically identifies skin conditions through image processing, using Convolutional Neural Networks (CNNs) for deep learning. Only selected skin conditions called acne, psoriasis, eczema and dermatitis are identified and classified using images of the skin. This field involves data collection, organizing images, model design and training and checking model performance with general metrics. The study wants to guarantee that healthcare professionals can work with the model smoothly due to a user-friendly interface.

It is not equipped to diagnose patients in live settings and it does not intend to replace a dermatologist's job. Rather, it supports early identification and gives physicians a tool for making decisions.

The main objects of the exploration are as follows

- Accumulate and appropriate a large range of high-quality pictures that represent several skin diseases, paying attention to factors such as age, skin tone and ethnicity.
- Use brightness correction, normalize the image colors and remove noise to improve clarity and the same level of brightness.

- Design and implement a Convolutional Neural Network (CNN) tailored for the classification of skin diseases.
- Examine the results by looking at accuracy, recall, precision and F1-score to judge how dependable the model's diagnosis is.
- Check the model's performance and note down what worked well and what could be better and then suggest ways to improve it.
- Provide a useful interface that is easy for doctors and healthcare providers to operate.

Motivation

Applications of both machine literacy (ML) and Deep Literacy (DL) in each field of health and medical image analysis have determined some positive trends in identity and opinion for the presence of colorful disease.

Thus, we are convinced that by exploiting these available technical devices, one can design a skin diseases that are effective, accurate and universally available. Thus, using our proposed system to detect SD and obtain proper care at the long phase, using our proposed system, the risk arising from this complaint can be significantly reduced and the situation has improved. Life standard for various cases.

In addition to refining healthcare issues, our result has many lower sides. Through simplifying the first stage of the case and simplifying the first stage of treatment, one should facilitate both cases and health care doctors. Similarly, our A-rooted devices can increase the expansion of SD Buddha, which allows relevant examinations in multilateral health environment without technical sophistication.

It can save a lot of cost by avoiding expensive medical complications in turn, which are holding anywhere in this process. Similarly, obtaining an initial diagnosis can help reduce cases of aggressive treatment requirement and, perhaps, reevaluate surgery, which will help lighten the load on both cases and overall health management industry. As a result, our result can not only improve existing conditions in the health sector, but can also help people with skin diseases to give better quality of life.

Proposed System

With recent advances in AI, this investigation relies mainly on deep learning approaches such as CNNs to design a system that spot skin diseases from images. A large number

of skin images with different dermatological problems like acne, psoriasis, eczema and inflammatory issues are used to train the model. Because of the detailed training, the model is able to spot unique markers of skin diseases, among them are spots, redness, scaling and visible inflammation.

The main advantage of this system is that it can notice skin changes and identify the beginning of diseases, so treatment can start early and the condition is less likely to progress. They can check the system's suggestions and reference them before making clinical decisions.

As a result, the model for skin disease screening offers accurate and reliable diagnosis, can be scaled up and is designed to be easy to use, so it is appropriate for many healthcare institutions, even in poor or remote areas. Implementing this method can detect skin diseases early, so doctors have to use fewer strong drugs and patients enjoy a higher quality of life.

Dissertation Structure

The thesis dissertation is structured into the following chapters:

Chapter 1: Skin diseases (SD).

Chapter 2 : Machine Learning (ML) and Deep Learning (DL).

Chapter 3 : Proposal.

Chapter 4 : Implementation and Results

Finally, Conclusions are drawn, and future works are presented.

Chapter 1

Skin Diseases SD

Chapter 1

Skin Diseases SD

1.1 Introduction

Since skin is the body's biggest organ, it is responsible for defense against different risks, control of temperature and sensation. Several things can cause these conditions, like genes, the environment, contracting infections or losing tolerance to one's body, so these conditions usually result in many symptoms that affect a person's daily living. Skin diseases are primarily grouped as inflammatory disorders (including eczema and psoriasis), infectious diseases (such as impetigo and herpes simplex), autoimmune diseases (like lupus erythematosus and scleroderma) and neoplastic disorders (which cover both harmless growths and malignant tumors). Contact dermatitis happens when the skin reacts to irritants or allergens, usually making it red, itchy and uncomfortable. When someone has a possible skin disease, a medical history is taken, it is examined and sometimes tests or a sample from the skin is done to find the cause. Treatments may be customized, involving topical creams, corticosteroids, different medications, phototherapy or modifications to diet and daily habits that help control symptoms and stop upcoming attacks. Skin diseases are important to study and treat because, being chronic and challenging, they require people to take ongoing care and researchers to work to create better treatments.

1.2 Skin Diseases

There are numerous skin diseases that can impact the skin which is the largest organ in the body. Irritations, inflammations and infections are some ways these diseases may show and their causes might include allergies, certain irritants, genes or defects in the immune system [1].

Skin diseases cover all types of conditions that can block, irritate or cause inflammation

in the skin which result in rashes, lesions or skin color changes. Severity can differ in all types of skin diseases which may involve usual problems such as dermatitis and eczema, as well as serious diseases such as skin cancer.

On the whole, skin diseases play an important role in causing illnesses for people of various ages and backgrounds.

1.3 Types of Skin Diseases

1.3.1 Inflammatory Skin Diseases

These illnesses are marked by inflamed skin that leads to symptoms. Examples are redness, inflammation and itching. Some of the more typical cases are:

1. Eczema (often referred to as atopic dermatitis) is an ongoing problem that leads to dry and itchy skin. inflamed skin. Most of the time, it begins because of outside factors and it can differ from ranging from being mild to very severe [2].

2. Psoriasis is an autoimmune condition where skin grows much faster than usual. cells accumulate which causes scaled and inflamed areas. Cases of psoriasis show up as red patches. The scales on a zebrafish are thick and silvery.

3. Seborrheic Dermatitis results in reddish and greasy skin with flakes, commonly. hair everywhere; from your scalp and face. Going through stress and hormonal changes can worsen the problem [2].

4. Urticaria (Hives): The skin condition causes itchy and raised welts to appear. often happens because of allergic reactions.



Figure 1.1: Images of inflammatory skin diseases(Eczema,Psoriasis,Seborrheic Dermatitis,Urticaria) [3].

1.3.2 Infectious Skin Diseases

Pathogens such as bacteria, viruses, fungi ,and parasites can all result in skin infections. Cuts and abrasions create points where the infection could enter. Hair follicle ailments can also arise, leading to many problems. Common Kinds of Skin Infections

1. Bacterial Infections:

Skin infections by Staphylococcus and Streptococcus species are very common. If these bacteria get inside the skin, mainly through open wounds, they might cause infections, as well as current skin problems [4].

You can get bacterial skin infections such as impetigo (which often affects children) and severe diseases like necrotizing fasciitis.

2. Viral Infections:

Viral skin infections are possible if the varicella-zoster virus is involved. Which results in shingles and the herpes simplex virus which may result in they can cause conditions like genital herpes [5].

Usually, these infections are demonstrated by rashes or blisters that appear on the skin.

3. Fungal Infections:

Dermatomycoses or fungal infections, may harm the skin and hair. nails. Fungal diseases regularly found in people are often ringworm and athlete's foot [6].

Superficial or deep infections can happen, depending on what kind of fungus is involved. involved.

4. Parasitic Infections:

Skin problems can also be caused by diseases such as scabies (from mites) and those caused by hookworms. infections.

They are usually managed by the use of topical creams or lotions.

1.3.3 Chronic Skin Diseases

Chronic skin problems long-term can have a major effect on someone's everyday living. Care for these diseases is continuous and the symptoms and severity can differ a lot among people.

Open the door to typical long-term skin disorders

1. Psoriasis causes red, itchy and scaly patches to develop on someone's skin. The rash often causes itchiness and pain and it tends to get worse occasionally [7].

2. An example is eczema (Atopic Dermatitis) which is a long-lasting skin disorder that causes skin to become dry, itchy and swollen. Although it mainly happens to children, night terrors can last beyond childhood.

3. Than 16 million people in the US have rosacea which results in facial redness. redness, the presence of blood vessels and sometimes a few pus-filled bumps. Symptoms return, especially if stress is high. It can go on for weeks or months [8].

4. Vitiligo results in skin patches becoming lighter because the cells that cause color are badly affected.

melanocytes which produce pigment in your body.

5. A painful disease known as Hidradenitis Suppurativa forms lumps under the skin in places where skin rubs together, for instance, the armpits and groin.



Figure 1.2: Images of Rosacea and Vitiligo and Hidradenitis Suppurativa [3].

1.3.4 Genetic Skin Diseases

They produce different molecules and have needed abilities for promoting connections and communication between cells to maintain how well the skin is put together. Particularly, one change in one of these the fate of organs and their functions can be ended abruptly by these molecules. whenever networks are involved, it can create cell separation, blistering and many other notable symptoms seen in inherited skin diseases. Many monogenic skin diseases were studied in the past decades using the classical approaches of genetics. Mainly, these studies have highlighted the many substances and genetic and molecular processes that allow for skin to be both strong and soft. Because of the human genome project and new advances in genetics, a lot of progress has been made in revealing the causes of complex skin diseases [9].

1.4 Skin Structure

The skin is by far the biggest part of our bodies and covers our entire surface. they are vital to our health by shielding our cells and tissues. It consists It has three main layers: epidermis, dermis and hypodermis (also referred to as lies just under the skin (subcutaneous tissue) [1] [10] .

1. Epidermis: The epidermis which forms the topmost layer, is only thin but is also tough. It is mainly includes keratinocytes, these cells make keratin, a protein that acts to protect the skin [11]. Melanocytes are found in the epidermis and they make melanin which determines skin color [10].

2. Dermis: Beneath the epidermis is the dermis which is thicker and filled with blood vessels, nerves and connective tissue are also included in the body systems. That area includes hair follicles and sweat glands. All of these glands, known as sebaceous (oil)

glands, are vital for keeping the skin in good condition [10]. 3. Hypodermis: - The hypodermis which is the shallowest layer, connects the dermis to tissues below. For example, muscles and bones. Adipose tissue which offers protection, is the main component of fat. protecting and supporting the body for good posture [12].

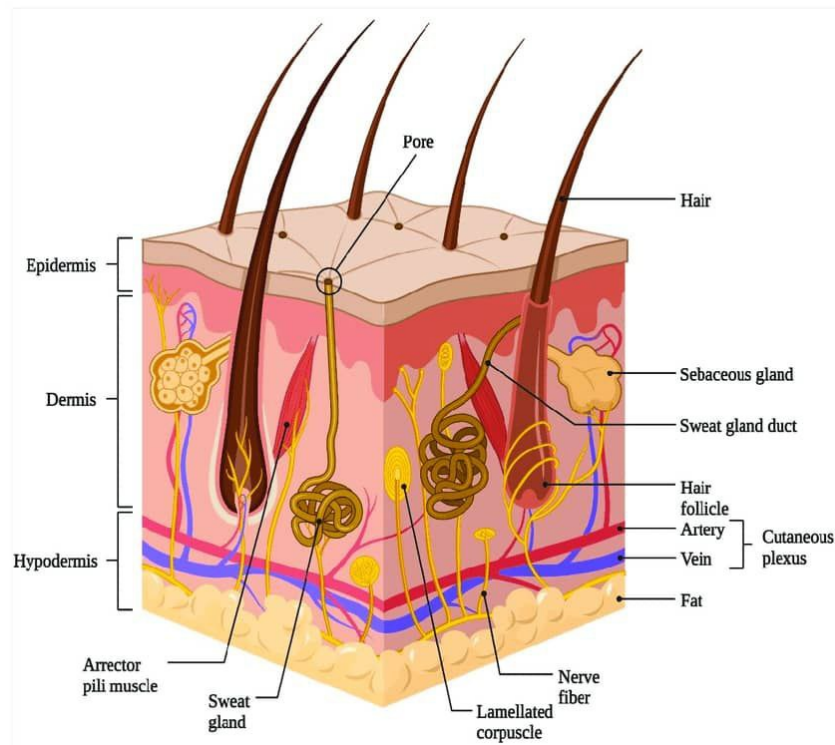


Figure 1.3: skin stucture [3]

1.5 Common Skin Diseases

Skin diseases are found in many forms and what causes them, how they present and how they are treated is also different. Below are some skin conditions that are common these days.

1. Acne: Acne is a common problem for people, especially during the teenage years. People with this skin type usually have pimples, blackheads and cysts forming mostly on their face, back and shoulders. Acne can be made worse by different hormones, stressful situations and some prescribed medicines [13].

2. Eczema (Atopic Dermatitis) is a common skin problem. Eczema makes the skin become very dry, itchy and red. In children, it is common and allergens, irritants and stress are frequent triggers.

3. Psoriasis: Psoriasis speeds up the growth of skin cells which produces red, swollen and scaly patches on the skin. Some infections, high stress and certain medicines can

cause it.

4. Rosacea: A key symptom of rosacea is a noticeable redness on the face, together with visible veins. It may result in minor, red bumps that are filled with pus. Rosacea is sometimes caused by spicy foods, drinking alcohol, being in the sun or stress.

5. Hives (Urticaria) : When an allergic reaction, stress or something else happens, it can cause hives which appear as raised, itchy bumps on the skin. Sometimes they come on very fast and differ in their size.

6. Cold sores (Herpes Simplex Virus) are another example. The herpes simplex virus causes cold sores and these usually appear as blisters near the mouth. Stress, being ill or sun exposure can cause melanoma.

7. Vitiligo: Vitiligo occurs when the cells that provide pigment to the skin are damaged which leads to areas of depigmentation. It is sometimes related to various autoimmune diseases [1].

8. Skin Infections: The skin can be affected by bacterial, viral or fungal infections and these can cause impetigo, warts and athlete's foot. Depending on the organism causing the illness, patients may need a particular treatment.

Recognizing these frequent skin diseases can make it easier to find and treat the associated symptoms promptly. If you have skin problems that don't go away or that are very severe, consulting a doctor is a good choice.

1.6 Chronic Skin Diseases

Common Types of Chronic Skin Conditions

1. Another name for eczema is atopic dermatitis and it causes dry and itchy skin.“ and appear to clog the pores. Parents and adults may both experience it and it usually needs longterm therapy to prevent flare-ups [14].

2. Psoriasis occurs when the immune system leads to overgrowth of skin The body is affected by patches that become thick, are red in color and are surrounded by silvery scales. People may develop psoriasis. Back pain is often accompanied by other health problems and can be very uncomfortable.

3. Although most associate acne with teenagers, it can continue into adulthood. If acne persists over time, it may result in marks on the skin and cause emotional problems, thus a different approach is needed. treatment options.

4. Rosacea leads to a red face, obvious blood vessels on the face and sometimes other problems. acne-like bumps. Various triggers may cause heartburn, for example, foods that are spicy or hot. and alcohol.

5. Hidradenitis Suppurativa causes chronic pain and swelling in affected people. Bumps found under the skin usually, where people do a lot of rubbing, for example, in the armpits groin [15].

6. Vitiligo creates white patches on the skin because it causes cells to lose color. This situation is emotionally stressful for people who are impacted. Melanocytes exist when the cells making melanin are killed or become dormant. functioning.

For chronic skin conditions, a number of things can play a role in their development such as

- Genetics: People's chances of getting certain skin issues can be increased by things they've inherited from their family.

Weather changes, allergens and various irritants can cause symptoms to get worse.

- Some Health Problems: Diabetes and inflammatory bowel disease can be connected to skin issues.



Figure 1.4: Images of eczema and Psoriasis and acne Rosacea and Hidradenitis Suppurativa Vitiligo [16].

1.7 Skin Diagnosis Techniques

1.7.1 Clinic Examination

Cutaneous lesions are usually diagnosed by gathering a history and doing a physical examination. It is important in history taking to mention the exact circumstances under which the symptoms began. Details about the progress of the disease, treatments used and the ways these treatments affect patients should be noted. The disorder should be described by including its date of onset and its starting location, the paths the tumors follow and the way they evolve. The main skin functions involve experiences of pain and itch. Where and how the speech is given Finding additional abnormalities is often critical: associated signs in other parts of the body medications, coexisting diseases and problems with the immune system are examples. Therefore and total history taking It is often very

important to collect all the previous information. Then, because there is such a large number In skin diseases caused by the environment, taking a proper medical history is important. about lifestyle habits, professional experience and home environment. items that might have been developed, adjusted or tested, so as not to omit them, In some cases such as that of a contact dermatitis. Dermatological physical Looking for suspicious chemicals needs training. One must be able to spot the how the lesions look, where they are and their design in the skin distribution [17].

1.7.2 Skin Biopsy

Typically, the skin punch biopsy is done using 3-mm-wide biopsy skin samples. every 50 μm inch, fifty-five thin sections result from one sample. Of the examination of 3 to 4 sections is done as the skin biopsy is conducted. a group that includes people from the entire population. Pathologists stain sections with immunohistochemistry. As a panaxonal marker, antibodies show against ubiquitin carboxy terminal hydrolase (ie, protein) Article also points out that a protein called gene product [PGP] 9.5 antibody is both highly set and present in high levels in neurons. to show the presence of nerve fibers beneath the epidermis. A standard process is used to measure nerve fibers. approach with reliable sensitivity, specificity and positive predictive value. This the analysis of the skin punch biopsy will be called “analysis of nerve fiber biopsy.” This test is called a “skin biopsy.” Skin biopsy examinations can show if there are nerves in the epidermis. decreased nerve tissue which may indicate neuropathy and neuropathic pain [18].

1.7.3 Light Microscopy Examination

Basic Principles and Methods

Fundamental Components

Light microscopy is a crucial diagnostic tool in dermatology that allows visualization of skin structures up to 1,000 times their actual size [19]. The examination process involves:

- Transmission of visible light through specimens.
- Enhancement of contrast through various techniques.
- Non-destructive examination preserving sample integrity [20].

Key Microscopy Methods:

1. Brightfield Microscopy : Most commonly used method for examining stained skin sections [21].

2. Polarized Light Microscopy : Enhances visibility of structures like collagen fibers [22].
3. Phase Contrast Microscopy : Enables examination of unstained samples
4. Fluorescence Microscopy : Allows visualization of specific proteins using fluorescent markers

Specimen Collection and Preparation:

Types of Skin Biopsies:

Several biopsy methods are available depending on diagnostic needs:

- Excision Biopsy : Complete lesion removal [23].
- Incisional Biopsy : Partial lesion removal [23].
- Shave/Tangential Biopsy : Horizontal section removal [23].
- Punch Biopsy : Full-thickness circular sample
- Curettings : Tissue fragment removal [23].
- Fine Needle Aspiration : Cell aspiration [23].

Specimen Processing:

1. Fixation :
 - Immediate placement in 10% neutral buffered formalin
 - Formalin volume should be 10x the tissue volume [23].
 - Proper labeling with patient information and biopsy details [23].
2. Processing and Embedding :
 - Dehydration using alcohol
 - Clearing with xylene
 - Paraffin embedding
 - Sectioning into 3-5 micron slices,22]]

Staining Techniques :

Primary Staining Methods :

1. Hematoxylin and Eosin (H&E):

- Standard staining method
- Stains nucleus blue (hematoxylin) and cytoplasm pink (eosin)
- Provides detailed tissue morphology visualization

2. Periodic Acid-Schiff (PAS) :

- Detects polysaccharides, glycoproteins, and mucins
- Particularly useful for identifying fungal organisms
- Produces magenta coloration [24]

3. Special Stains :

- Trichrome stains for collagen and muscle fibers
- Silver stains for reticular fibers
- Immunohistochemistry for specific antigen detection [25].

Quality Control and Troubleshooting :

Common Challenges and Solutions :

1. Specimen Orientation:

- Proper orientation is crucial for dermal-epidermal junction visualization [26].
- Use of orientation markers or dyes.

2. Small Specimen Handling:

- Use of fine instruments.
- Employment of biopsy pads or mesh for support [26].

3. Quality Control Measures:

- Regular equipment maintenance and calibration [26].
- Consistent staining procedures [26].
- Multiple pathologist review for accuracy [26].
- Comprehensive record keeping [26].

Standard Operating Procedures :

1. Equipment Maintenance:

- Regular calibration of microscopes.
- Cleaning of lenses and components [26].

2. Documentation:

- Detailed recording of findings.
- Generation of formal pathology reports [26].

3. Waste Management:

- Proper disposal of samples and materials.
- Compliance with laboratory regulations [26].

Applications in Dermatology

Light microscopy serves multiple purposes in dermatology:

- Diagnosis of various skin conditions.
- Research into disease pathophysiology.
- Educational tool for training dermatologists.
- Monitoring treatment effectiveness.

1.7.4 UV imaging

UV reflectance photography happens when UV light hits and is reflected from the subject. the camera records information from the light reflected off your skin to create the image - which is like UV light doing the same thing. photography done with normal visible light. In the past, it was mostly used to examine the skin. increases changes in image contrast, especially with melanin-rich structures and is being explored more today. sun care area and the imaging of sunscreens. The waves in UV light have shorter lengths. Points out the skin surface, so that marks such as lines and wrinkles are drawn to attention. The more light of a short wavelength gets absorbed, the more the melanin thickens. it appears darker under UV light than under natural light and the difference between the two is noticeable As wavelength decreases, light will behave this way as seen in Figure 15.1 that shows the hand The image below shows a Fitzpatrick skin type 2 subject, illuminated by visible light (not ultraviolet). filters that block UV wavelengths between 390 and 310 nm using 10-nm band-pass filter sets and a xenon lamp the use of a light source and a modified camera was necessary. Photographs of the different racial

groups quit being hidden. The ranges of wavelength in Figure 1.6 are adjusted for camera variations. In response, the filter and change the way light passes and is transmitted by the diffuse reflectance standard. Cross-polarization has not been done, so what can be seen are specific features. bounce from the muscles just below the skin. The change in camera, lens or image affects your footage. calibration, filters and the way light is produced will be covered later on in the chapter [27].

1.8 Classification of Skin Diseases

Class I: Inflammations

Group I: Simple Inflammations (Allied to Simple Dermatitis)

1. **Erythema** ([a] simplex , [b] multiforme , [c] chronicum – comprehending papulatum, nodosum, strophulus, squamosum, pityriasis furfuracea, membranacea, rubra)
2. **Herpes** (simplex and zoster)
3. **Urticaria** (idiopathic, from ingestion of particular kind of food; from uterine affections; or persistent)
4. **Dermatitis** (idiopathic, as from burns or frostbite; or symptomatic, as of erysipelas; or phlegmonous, as furunculus, anthrax, Aleppo tubercle)
5. **Pemphigus** (benign, persistent, and foliaceous)

Group II: Eczematous Inflammations (Allied to Eczema)

1. **Eczema** (erythematodes ; E. papulosum, comprising lichen simplex and prurigo ; E. vesiculare ; E. rubrum; E. pustulosum, comprising impetigo sparsa, figurata, and pilaris ; E. lichen ; E. squamosum ; E. pityriasis).
2. **Acne** (comprising A. simplex; A. pilaris ; A. rosacea) .
3. **Ecthyma**([a] simplex ; [b] chronicum = rupia; [c] gangrenosum).
4. **Psoriasis** (punctata, guttata, nummularis, circinata [lepra], gyrata, confluens)

Group III:

1. Ulcers.
2. Idiopathic.

3. Sympathetic.
4. Constitutional.

Class II: New Formations

Group I: Homologous New Formations

1. **Epidermic** (epithelial growths, comprising callositus, clavus ichthyosis, cornu cutaneum).
2. **Pigmentary** (lentigo, ephelis, moles, melanosis, chloasma, silver stain, leucopathia).
3. **Dermic** (cicatrix normal, or cheloid, cutaneous tumor [wens], multiple tumors [mycosis], molluscum simplex, condylomata, verruca vulgaris, verruca mollis).

Group II: Heterologous New Formations

1. **Pseudoplasms** (lupus, comprehending maculosus, tuberculosus, hypertrophicus, exedens, serpiginosus; and lepra, comprehending maculosa, tuberculosa anaesthetica, exulcerans).
2. **Neoplasms** (epithelioma, carcinoma).

Class III: -Hæmorrhages-e. g., petechia, vibices, ecchymosis, purpura

Class IV: -Diseases of Accessory Organs-e. g., hair, nails, sweat glands

Class V: Diseases Defined by Uniform Causes

Class V: Diseases Defined by Uniform Causes

1. **Parasitic Diseases.**
2. **Syphilitic Eruptions.**
3. **Eruptions of Specific Fevers.**
4. **Scrofulodermata [28].**

1.9 Conclusion

Ultimately, dermatology cares for people who have a broad range of skin diseases these pose major difficulties for healthcare systems everywhere. With advancements among diagnostic methods, this includes medical checks, skin biopsy and light-based tools With microscopes and using UV light, healthcare professionals are able to see problems more clearly. They are responsible for assisting and managing people with these conditions well. The many different types of skin diseases included The evolution of medicine covers inflammatory, infectious, chronic and genetic conditions which reveals why we need a basic and detailed knowledge of the skin and what can go wrong with it. Going forward To work on skin diseases, you need to use the most innovative approaches. Many options in treatment and care that are adapted to each individual patient get the best possible care. Dermatology will move forward as we adapt to improving technology. new health-related science and inventions eventually enhance care for patients. better treatments leading to improved life quality for people with skin conditions.

Chapter 2

Machine Learning (ML) and Deep Learning (DL)

Chapter 2

Machine Learning (ML) and Deep Learning (DL)

2.1 Introduction

New methods for explore medical diagnostics, especially in dermatology, are being explored because of Machine Learning (ML) and Deep Learning (DL). More data and improved computer power are letting researchers use these technologies in various medical fields. Artificial Intelligence (AI) is gaining popularity in dermatology and we are seeing its helpful effects in recognizing and classifying skin problems at an early stage. Deep Learning has demonstrated the ability to spot acne, psoriasis, eczema and melanoma clearly in clinical pictures for dermatologists.

While others depend heavily on engineering features, deep learning can learn from raw images simply because of neural networks. Because of this, DL-based systems can handle a wide range of dermatological diseases and be scaled as needed. Also, these systems can lead to customized care for patients, a clearer picture of eye disease signs and the ability to catch severe cases ahead of time, all of which benefit patient outcomes.

Right now, deep learning is becoming popular in dermatology for being simple to use as well as for being very accurate. But, before appreciating how deep learning works in skin disease diagnosis, it helps to know what led to its advent.

2.2 Artificial Intelligence (AI)

Artificial intelligence (AI) is about designing machines that have certain abilities. to show behaviors that are human-like.

Machine Learning (ML) and Deep Learning (DL) belong to the area of Artificial

Intelligence. (AI) built to develop algorithms and models that can learn with data. While they Even though they are not the same, there are a few similarities and important differences They are used for various purposes A quick intro to Machine Learning (ML) and Deep Learning (DL) [29].

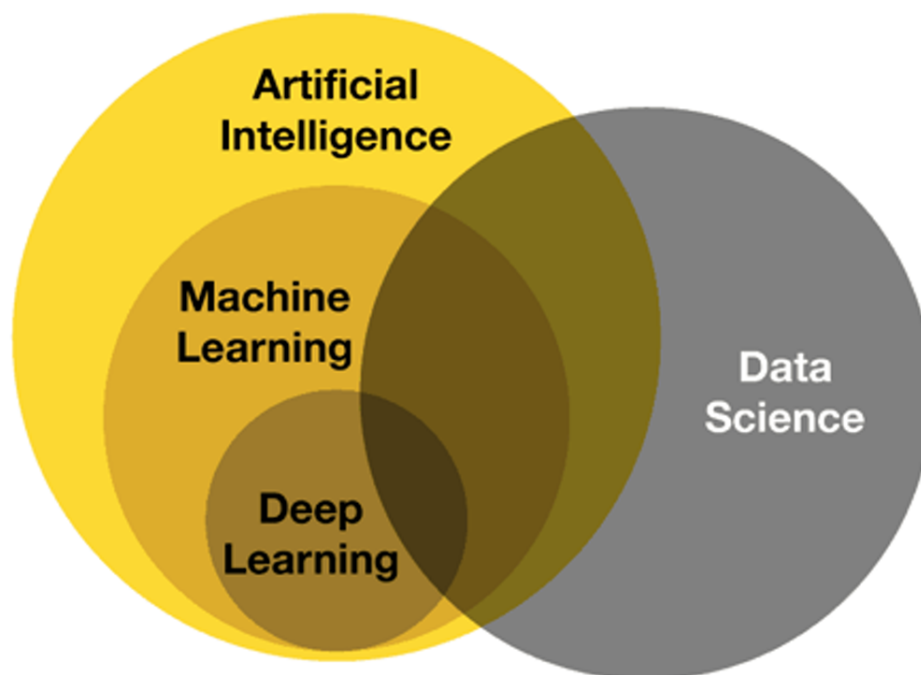


Figure 2.1: AI Vs ML Vs DS [30]

2.3 Machine Learning (ML)

Machine learning (ML) is a branch of Artificial Intelligence (AI) that systematically applies algorithms to synthesize the underlying relationships among data and information. For example, Machine learning (ML) systems can be trained on automatic speech recognition systems (such as iPhone’s Siri) to convert acoustic information in a sequence of speech data into a semantic structure expressed in the form of a string of words.

Machine learning (ML) is already finding widespread uses in web search, credit scoring, stock market prediction, gene sequence analysis, behavior analysis, smart coupons, drug development, weather forecasting, big data analytics, and many more applications.

Machine learning (ML) will play a decisive role in the development of a host of user-centric innovations [31]. In 1959, Arthur Samuel described machine learning (ML) as the field of study that gives computers the ability to learn without being explicitly programmed. He concluded that programming computers to learn from experience should

eventually eliminate the need for much of this detailed programming effort. According to Tom M. Mitchell's definition of Machine learning (ML): A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E [31].

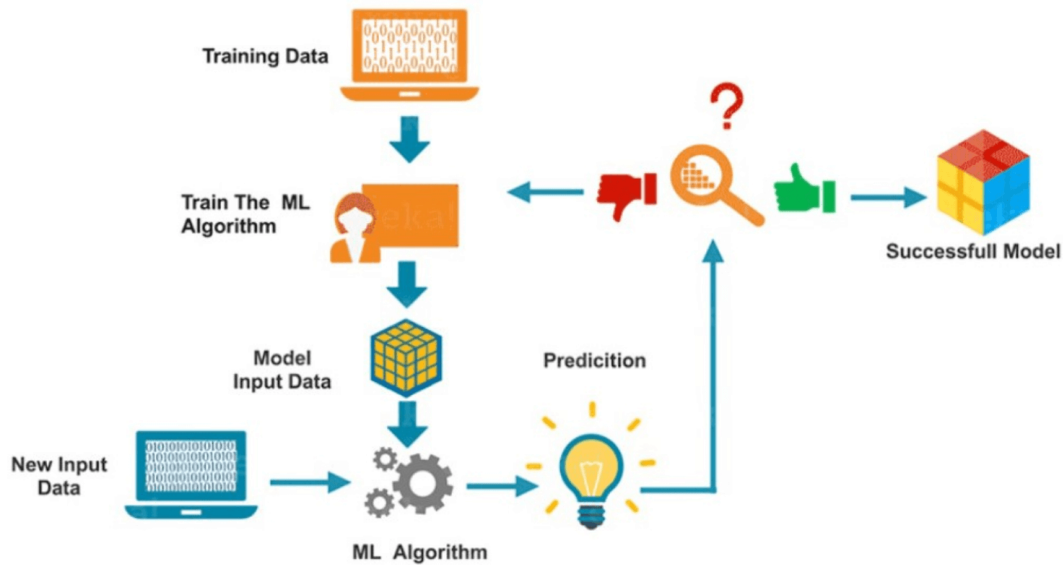


Figure 2.2: Machine learning design [32]

2.3.1 Types of Machine Learning

In the taxonomy, the machine learning (ML) algorithm is applied according to the outcome desired by the algorithm. Main types of algorithms are: The terms here are supervised learning, unheard learning and reinforcement learning.

2.3.1.1 Supervised Learning

Where the algorithm develops a function to turn input into the desired output. A common kind of supervised training work is the Formation classification problem. A function can map data in a way that defines how to learn (to approximate) a task. I encountered a vector as one of many statistics in trial applications. Celebration.

Some forms of supervised learning involved are linear regression, logistic region, cart and naive The algorithms used are Bayes, KNN and Deep Learning (DL). We can see from Figure 2.3 that it includes two classes: apples and Cupcake. A group of training

data exists for each of those tasks. The algorithm has learned the distinguishing aspects of each class and is now able to predict the class of something never seen before.

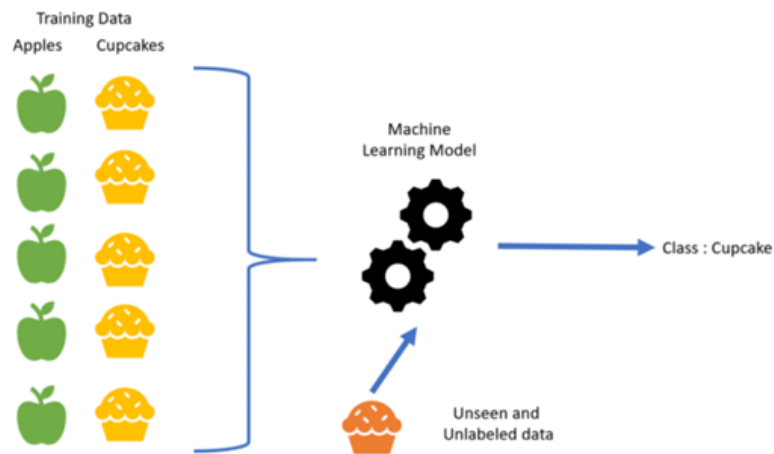


Figure 2.3: Supervised learning example [33]

2.3.1.2 Unsupervised Learning

By watching what happens, the model figures out the structures hidden in the data. After being fed a dataset, the model automatically looks for similarities and identifies patterns in the data by forming clusters. What looking at data this way won't do is give the clusters labels, so we can't call a cluster apples or mangoes, but we can tell them apart. [32].

These are examples of unsupervised learning techniques: Apriori, K-means, SVM and PCA. For example, observing Figure 2.4, we find that if we offer apple and carrot, then the possibility that the rabbit might be fed just the carrot is 0. As the main information, cheese goes through each model, although it is not able to create a separation. Look at the treelike structure without the label to decide if it is an Apple or something else, since there are no more new points added. T.

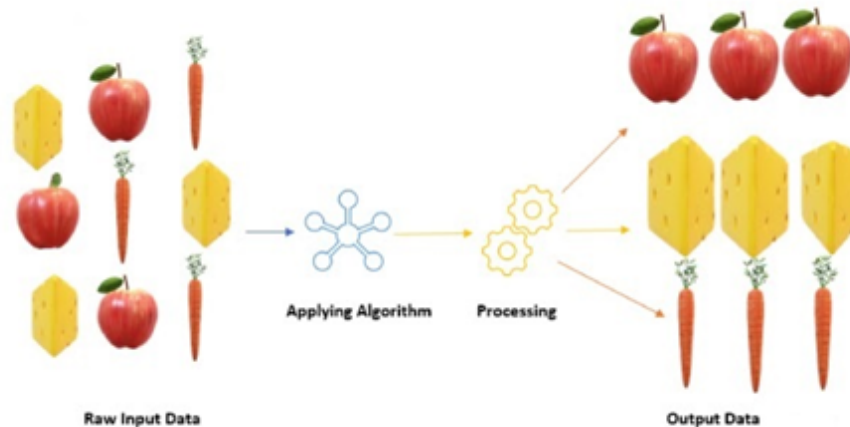


Figure 2.4: Unsupervised learning example [34]

2.3.1.3 Reinforcement Learning

Reinforcement Learning is about taking suitable actions to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path to take in a specific situation. Reinforcement learning differs from the supervised learning in a way that in supervised learning the training data has the answer key with it, so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer and the reinforcement agent decides what to do in order to perform the given task. In the absence of training data set, it is bound to learn from its experience [35].

The main purpose of reinforcement learning in this depiction is to guide the dog. An agent aims to fulfill a given task in a specific environment which involves the surroundings. the dog and the trainer are both included. At this point, the trainer says a command or gives a cue and the dog observes (observation). Following the reward or the cue, the dog makes a move. When the action is near to the behavior you hope to see, the trainer may offer a reward, for example, a food tidbit or a toy; if not, there is no reward given. In other cases, there will be a negative consequence provided. At first during training, the dog will do things like rolling around. While the command given is sitting, it tries to connect certain sights with consequences you receive for the things you do.

The best situation for the dog is responding the correct way through training. So that it receives treats for every right response. Basically, reinforcement learning training focuses on adjusting the dog's policy. that it gains abilities that produce the most reward. Following the training. The dog should be able to watch the owner and act accordingly

when necessary. [36].

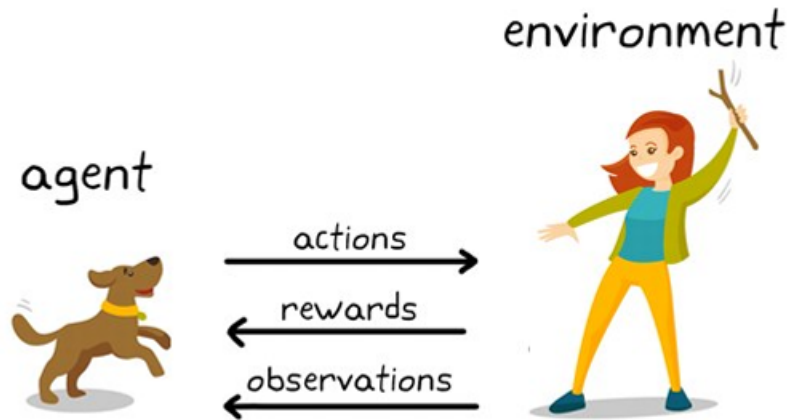


Figure 2.5: Reinforcement learning in dog training [36]

2.4 Deep Learning (DL)

Deep Learning (DL), as a recent branch of Machine Learning (ML), focuses on training computers to handle tasks that the brain does naturally such as recognizing images. At present, Deep Learning (DL) approaches have greatly influenced computing. Image applications often use programs like classifiers and methods for separating parts of a picture, called segmentation. certain parts of the work, periodic reports and ending the project.

Deep learning is concerned with a special type of machine learning. The term AI is also used for a technology known as Artificial Neural Networks (ANN) which emulates how the brain of a human functions. Deep learning methods developed in recent years are very strong for supporting supervised learning.

By adding extra layers and extra units in each layer, a deep network becomes capable of representing complex functions Organizations are become more complex. Most tasks which require mapping input vectors to output vectors and are easy through deep learning, it becomes possible for a person to do a task quickly if the dataset is extensive models and large numbers of labeled examples for training [37].

2.4.1 Artificial Neural Networks (ANN)

Artificial Neural networks (ANN) have become the subject of great interest in the past few years. when a bot can study sample maps on its own and understand them, this power is called universal approximation.

The word network is used because they typically consist of combining many sources. they have different functions called “neural” since they are inspired by neuroscience [37].

It is up to the teaching algorithm to use the layers in the best way, but the training data does not say what every layer should achieve. Though scientists started looking into the brain in the 1950s, the differences between artificial and biological neurons are more than just their contents, since artificial neurons were inspired by biological ones. For this reason, we must understand how neural networks start with origin units.

2.4.1.1 Biological Neuron

The neuron is the basic building block of the brain and central nervous system. Neurons are specialized cells that transmit chemical and electrical signals. The brain is made up entirely of neurons and glial cells, which are non-neuronal cells that provide structure and support for the neurons. Nearly 86 billion neurons work together within the nervous system to communicate with the rest of the body. They are responsible for everything from consciousness and thought to pain and hunger [38]. The above image shows the basic structural components of an average neuron, including the dendrite, cell body, axon, and axon terminal.

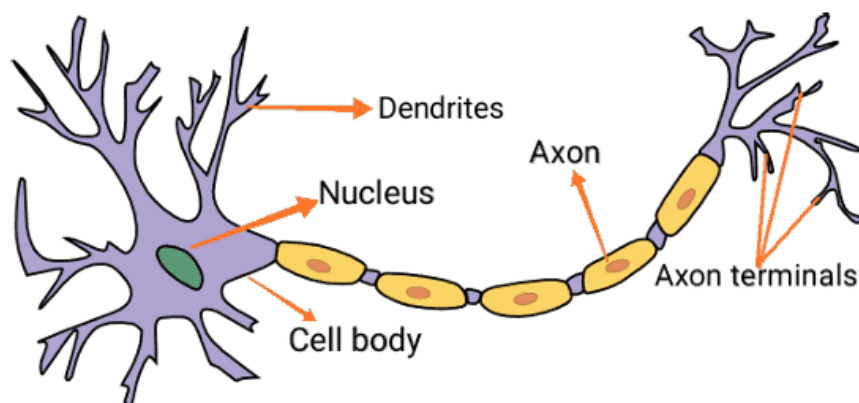


Figure 2.6: Biological neuron structure [39]

- **1. Dendrites** : are structures that stick out from the body of the cell as branches. function is to pick up messages from other neurons and let those messages travel on the part of the neuron called the cell body.
- **2. cell body (or soma)**: it has a nucleus and endoplasmic made of both rough and smooth internal structures small things like reticulum, Golgi apparatus, mitochondria and other parts of cells.

- **3. An axon:** Basically, the axon is a tube that carries an electrical message from the neuron's cell body (or the dendrites of another cell) to the ends of its branching terminals. Then, the neuron can send the impulse on to another neuron. This structure which appears in the cell body, provides a link between the axon and the rest of the nerve cell.
- **4. The synapse:** it is the connection formed when a dendrite from one neuron and an axon terminal from another meet. Unlike a real structure, it allows particular chemical reactions to happen.

Function of a Neuron: The specialized structure and organization of neurons allows them to transmit signals in the form of electric impulses from the brain to the body and back. Individually, neurons can pass a signal all the way from their own dendrites to their own axon terminals; but at a higher level neurons are organized in long chains, allowing them to pass signals very quickly from one to the other. One neuron axon will connect chemically to another neuron's dendrite at the synapse between them. Electrically charged chemicals flow from the first neuron's axon to the second neuron's dendrite, and that signal will then flow from the second neuron's dendrite, down its axon, across a synapse, into a third neuron's dendrites, and so on [40].

2.4.1.2 Artificial Neuron

Artificial neuron is a basic building block of every artificial neural network. Its design and functionalities are derived from the observation of a biological neuron that is the basic building block of biological neural networks (systems) which includes the brain, spinal cord, and peripheral ganglia. Biological neurons and artificial neurons are similar in design and functionalities. where the left side of Figure 2.7 represents a biological neuron with its soma, dendrites and axon and where the right side of Figure 2.6 represents an artificial neuron with its inputs, weights, transfer function, bias, and outputs [44].

The information sent to a biological neuron gets received by the dendrite and the soma. processes information and then uses an axon to send it to other parts of the body or brain. For an artificial neuron, An artificial neuron received inputs and each input is given a special weight. The input can be multiplied by a weight separately. The artificial neuron next adds up the inputs, adds in the bias and begins processing. the sum of functions in the transfer function. After everything, the artificial neuron sends out its final processing. cakes are formed using output(s). Artificial neuron models are simple which brings various benefits. explaining with mathematics below [41]:

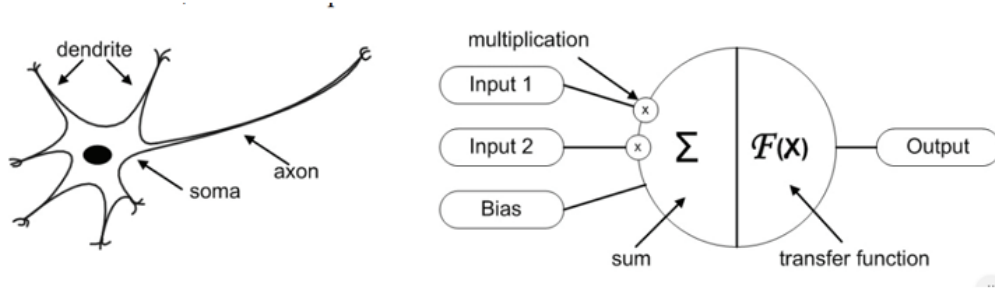


Figure 2.7: Biological and artificial neuron design [41]

$$y(k) = F\left(\sum_{i=0}^m w_i(k) \cdot x_i(k) + b\right) \quad (2.1)$$

Where:

- $x_i(k)$ is the input value in discrete time k where i ranges from 0 to m .
- $w_i(k)$ is the weight value in discrete time k where i ranges from 0 to m .
- b is the bias.
- f is the transfer function.
- $y(k)$ is the output value at discrete time k .

As seen from a model of an artificial neuron and its equation 2.1 the major unknown variable of our model is its transfer function. Transfer function defines the properties of artificial neuron and can be any mathematical function. We choose it on the basis of problem that artificial neuron (artificial neural network) needs to solve and in most cases we choose it from the following set of functions: binary step function, linear function and non-linear function [41].

Binary step function:

A binary step function turns on (activates) when an input reaches a certain level or threshold. If the input is placed at night. If the potential in the neuron drops beneath a particular value, the neuron fires and sends the exact same signal let's look at the next layer now.

$$y = \begin{cases} 0 & \text{if } w_i x_i \geq \text{threshold} \\ 1 & \text{if } w_i x_i < \text{threshold} \end{cases} \quad (2.2)$$

The problem with a step function is that it does not allow multi-value output for example, it cannot support classifying the inputs into one of several categories [42].

Linear function:

In linear activation, the formula is: $A = cx$ (2.3). The inputs are multiplied by the related weights which forms the output. the signal changes with the input. In a certain

way, a linear function is preferable to a step. this is convenient because it can return more answers besides yes and no. There are, however, serious issues with using a linear activation function:

- **Not possible to use back-propagation** (gradient descent) to train the model the derivative of the function is a constant, and has no relation to the input, X . So it's not possible to go back and understand which weights in the input neurons can provide a better prediction.
- **All layers of the neural network collapse into one** with linear activation functions, no matter how many layers in the neural network, the last layer will be a linear function of the first layer (because a linear combination of linear functions is still a linear function). So a linear activation function turns the neural network into just one layer.

A neural network with a linear activation function is simply a linear regression model. It has limited power and ability to handle complexity varying parameters of input data [42].

Non-linear function:

Modern neural network models depend on different kinds of non-linear activation functions. They help the model to work better. to design systems that map the inputs to the outputs which is crucial for dealing with and studying important data such as images, videos, audio and data sets They usually don't fall neatly into a simple, step-by-step model and can be very complex.

Virtually anything we can put in a computer can be modeled as a functional computation. can help to provide that the used activation function is not linear.

Non-linear functions address the problems of a linear activation function:

- They allow back-propagation because they have a derivative function which is related to the inputs.
- They allow the stacking of multiple layers of neurons to create a deep neural network. Multiple hidden layers of neurons are needed to learn complex data sets with high levels of accuracy [42].

The most famous non-linear activation functions are mentioned in the table 2.1 below.

Function	Formula	Advantages	Disadvantages
Sigmoid	$S(x) = \frac{1}{1+e^{-x}}$	Smooth gradient. Output values bound. Clear predictions.	Vanishing gradient. Outputs, not zero centered. Computationally expensive.
Hyberbolic Tangent	$\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$	Zero centered.	Like the sigmoid function.
Softmax	$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$	Able to handle multiple classes Useful for output neurons.	Does not support non linearly separable data. Does not support null rejection.
Relu	$y = \max(0.1x, x)$	Computationally efficient.	The dying Relu problem

Table 2.1: Some of non-linear activation functions

2.4.1.3 Layers

In this classic artificial neural network there are many types of layers used in the network, each type of layer is responsible for some computations.

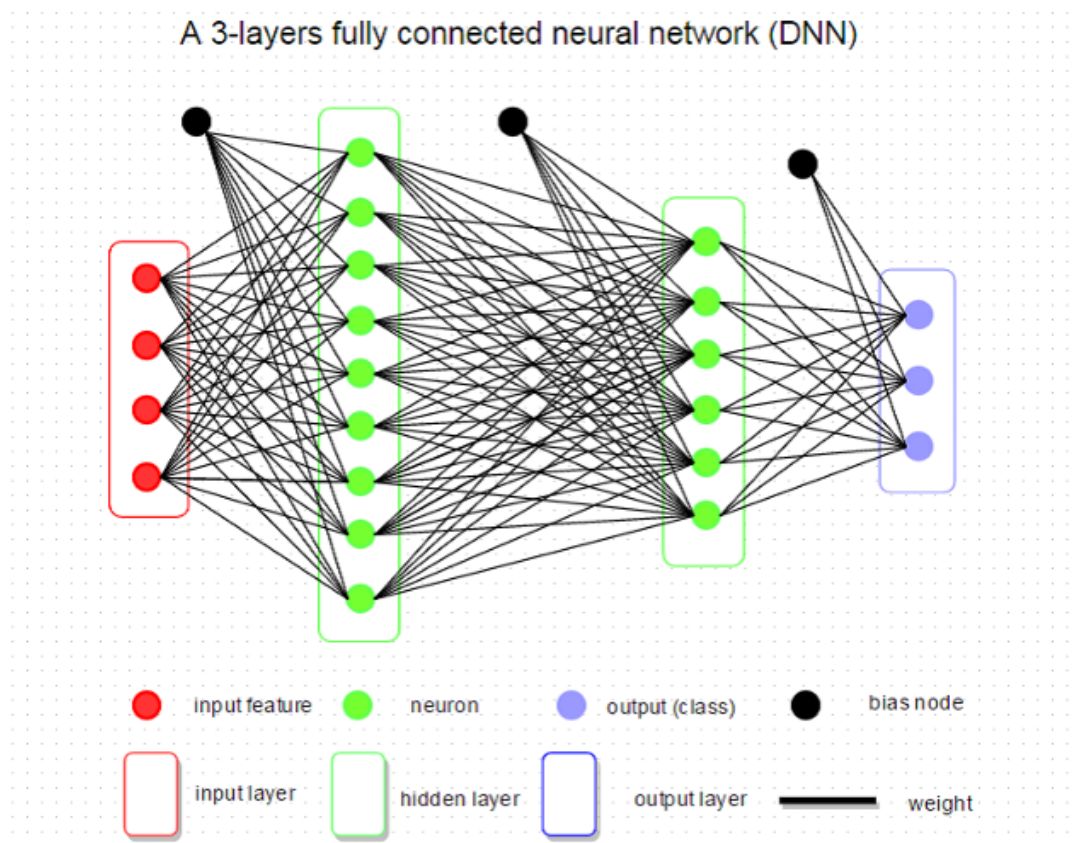


Figure 2.8: Artificial neural networks architecture [37]

- **Input layer:** Input layer is the first layer in the neural network, composed of input neurons, and brings initial data to the hidden layers for further processing.
- **Hidden layer:** Layers or a set of layers that are not the input or output layer. Deep learning seeks to find the very best way to solve an optimization problem. Most of these calculations are actually done in the hidden layer(s), not the final

layer. when data is not very complex, fewer hidden layers are needed. if the data is complex, it's better to limit the number of hidden layers addressing a simple issue with a complex structure may result in overfitting. the problem means that the model will not fit to the data well. Usually, each hidden layer produces a vector of numbers. The space in which the number of hidden layers determines how wide the model will be.

- **Output layer:** The output layer is the last layer in the neural network which produces the outputs of the program, in classification tasks, the size of the output layer is equal to number of classes [37].

2.4.2 Deep Neural Network architectures

There are many different neural network architectures and with time will be more architectures, so we choose some popular and widely used neural network architectures that you should know in order to advance your knowledge about neural network architectures.

2.4.2.1 Convolutional Neural Networks (CNN)

Convolutional neural networks, or CNNs in short, are the popular choice of neural networks for different Computer Vision tasks such as image recognition. The name convolution is derived from a mathematical operation involving the convolution of different functions.

There are 4 primary layers or stages in designing a CNN:

- **Convolution:**The input signal is received at this layer.
- **Subsampling or Pooling :**Inputs received from the convolution layer are smoothed to reduce the sensitivity of the filters to noise or any other variation.
- **Activation:**This layer controls how the signal flows from one layer to the other, similar to the neurons in our brain.
- **Fully connected:**In this stage, all the layers of the network are connected with every neuron from a preceding layer to the neurons from the subsequent layer [43].

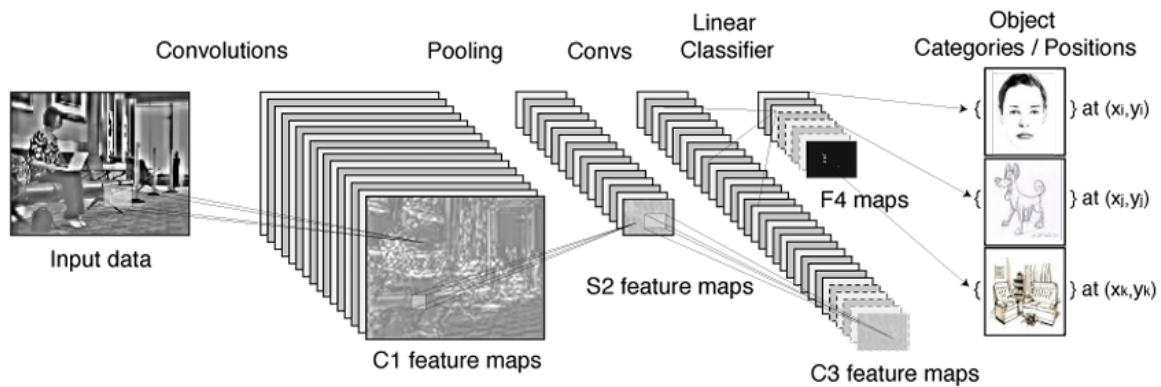


Figure 2.9: CNN architecture for visual recognition [44]

Advantages Of CNN

- Very good for visual recognition.
- Once a segment within a particular sector of an image is learned, the CNN can recognize that segment present anywhere else in the image.

Disadvantages of CNN

- CNN is highly dependent on the size and quality of the training data.
- Highly susceptible to noise [45] [46].

There are various architectures of CNNs available, which have been key in building algorithms which power and shall power AI as a whole in the foreseeable future. Some of them: LeNet [47], Alex Net [48], ResNet [49], EfficientNet [50] and Inception Net [51].

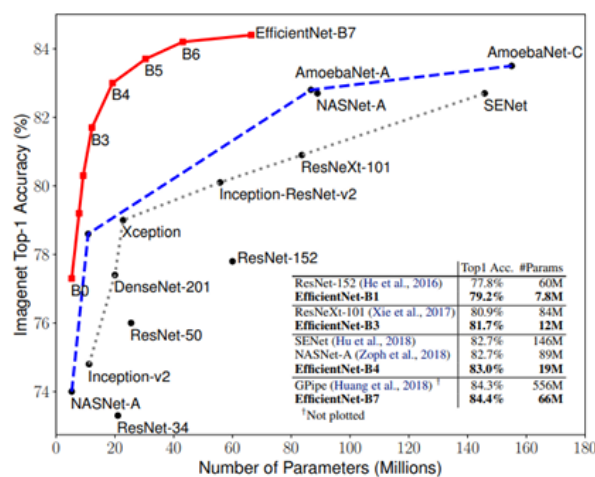


Figure 2.10: ImageNet Top-1 accuracy CNN models [50]

2.4.2.2 Recurrent Neural Networks (RNN)

Because information order plays a key role in some domains, Recurrent Neural Networks (RNNs) have grown very popular. Therefore, they discover many such as in real-world uses like natural language processing and generating computerized speech. In particular, language translation includes computer programs, human translation and machine translation. They get the name 'RNN' since a uniform task takes place repeatedly for every item of information. The element of a sequence depends on what has been done so far in the sequence as well as its input value. Look at these networks as having a place to store every piece of data they work with, using it to reach the final result [52]. There has been a lot of research and development of different types of RNNs over the last few years:

- **Bidirectional RNN** : The output in this type of RNN depends not only on the past but also the future outcomes.
- **Deep RNN** : Because there are several layers stacked for each step in this type, information can be processed faster and with more accuracy. To make the filters ignore such noises or vagaries in the data.

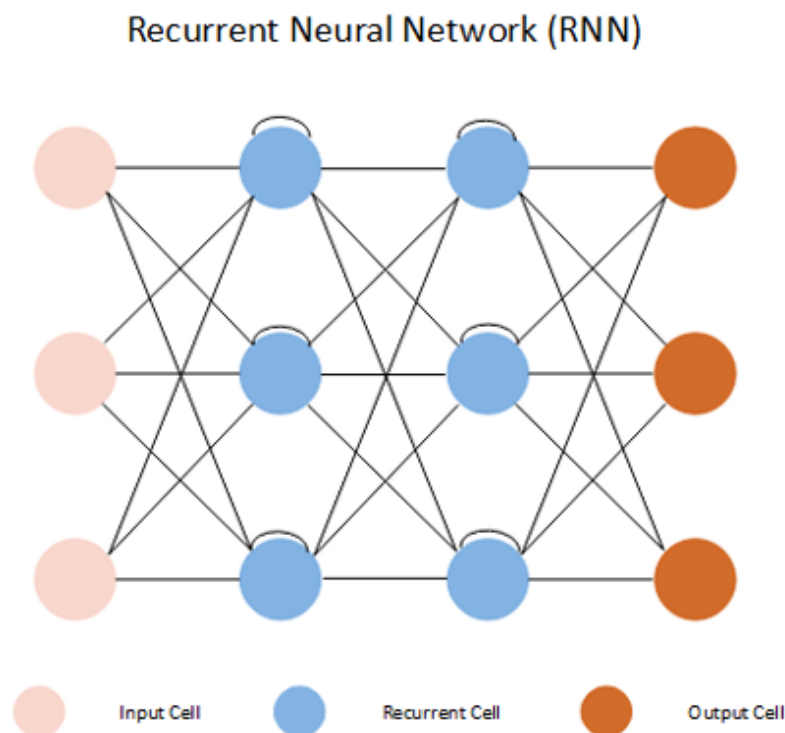


Figure 2.11: RNN architecture [53]

Advantages of RNN

- In contrast to a conventional neural network, an RNN uses only one set of parameters for all steps. It makes the task of learning fewer parameters much easier.
- You could employ RNNs with CNNs to produce trustworthy descriptions for new, unidentified items images.

Disadvantages Of RNN

- It is hard for RNNs to focus on things that occur a long time in advance. It stays true for the most part in When a noun is intervened by numerous words such as in long sentences or paragraphs. Also, the verb is important in a sentence.
- Restricted Boltzmann Machines cannot be designed with very extensive layers. The issue takes place because of the activation. It is used in RNNs to slow the amount of gradient at every extra layer [54].

Architectures of RNNs: Identity-RNN, np-RNN, LSTM and GRU.

2.5 Conclusion

We explained AI techniques in this chapter and people still use them today. Machine learning (ML) and deep learning (DL) are some of the main technologies involved. We listed the most recent information about AI for medical rules and definitions, as well as its devices. It was also mentioned that both K. R. Raymya and his colleagues had proposed a system using SVM which helps identify and classify aromatic spot edema with more than 85 % accuracy before. For Esfahan. A system designed to catch diabetic retinopathy (DR) with the help of carts. At 93% effectiveness.

Chapter Three will show you our proposal.

Chapter 3

Case Analysis

Chapter 3

Case Analysis

3.1 Introduction

Knowing the issue, we will now discover more in this chapter about related studies and how they resolved the problem. We will discuss the case of our study next before turning to system design (datasets and CNN).

3.2 Related Works

The earlier research on machine learning methods for retinal image classification is mentioned in this subsection.

3.2.1 A Method of Skin Disease Detection Using Image Processing and Machine Learning

This paper called "A Method of Skin Disease Detection Using Image Processing and Machine Learning" was presented at the 16th International Learning Technology Conference in 2019 and suggests an automated mechanism for finding skin diseases early with image processing and machine learning. Because skin diseases are common in hot countries like Saudi Arabia and current tests are costly and take too long, the research looks for an easier, faster and cheaper way to use a digital image and a computer. Using 80 images from online sites per category (Normal, Melanoma, Eczema, Psoriasis), a Convolutional Neural Network (AlexNet) is applied to extract features after properly resizing the images to 227×227 pixels. A Support Vector Machine (SVM) is used to sort these features and suggest the correct kind of skin disease. Twicemed implemented the system in MATLAB 2018b and it correctly identified all three targeted skin conditions and healthy skin. As

the study shows, these systems work well in places with limited resources and to improve them, efforts are planned for a mobile app, dermatological scanning and covering more conditions [55].

3.2.2 Leaf and Skin Disease Detection Using Image Processing

The study describes an automated method that detects both plant leaf diseases and human skin conditions using pictures and algorithms. Using MATLAB, the system goes through a series of steps such as preprocessing, using K-Means clustering for segmentation, extracting features and classification by means of a support vector machine (SVM). It separates leaf photos from skin images by analyzing their color and then it identifies melanoma, basal-cell carcinoma and other illnesses like leaf diseases. To do feature extraction, pictures are changed from RGB to LAB and HSI color formats and both GLCM and Otsu's thresholding algorithms are used to improve the accuracy of classification. Through experiments, the system was found to detect diseases with a very high accuracy of nearly 96 % which shows it can identify diseases early. They point out that the system can help in real-time health and farming situations with very little work required and excellent results [56].

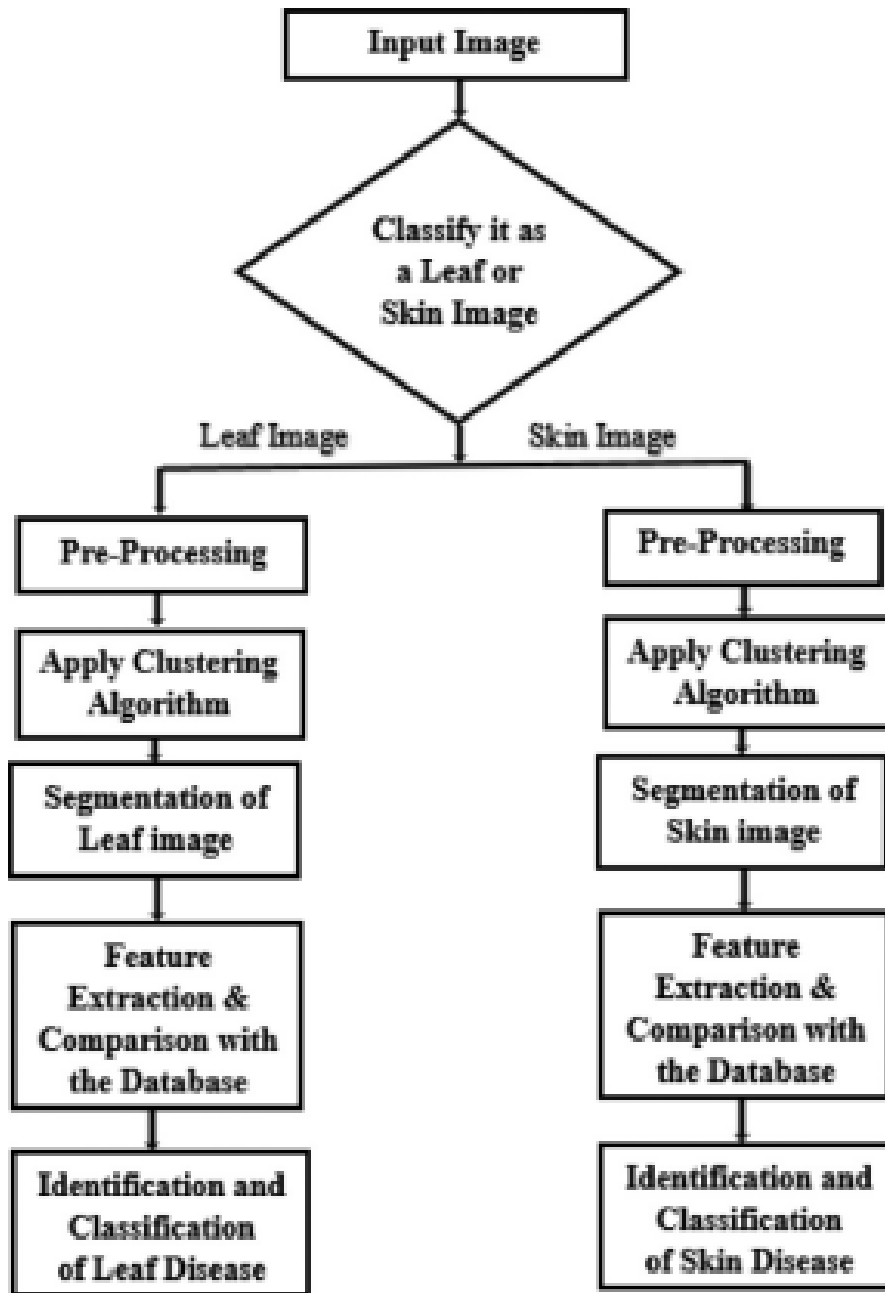


Figure 3.1: Flow chart of Disease Detection System [56]

3.2.3 A Image Analysis System to Detect Skin Diseases (SD)

The paper describes how an automated method for detecting skin diseases like skin cancer, psoriasis and dermatophytosis was built by using image processing and statistical techniques. Images may be captured or taken from a database and in the process, the system removes noise, divides objects from the background and finds statistical characteristics like variance, standard deviation, correlation factor and entropy. After that, the

image is given a classification: healthy, eczema, vitiligo or psoriasis, using a two-level Adaboost classifier that separates normal skin from diseases first and then identifies the particular disease among the three it can detect. Experiments shown over 90 % correct classifications, indicating the system's success in detecting conditions early and prompting users to get health advice. Researchers have also suggested future growth by making the process faster and including more data points [57].

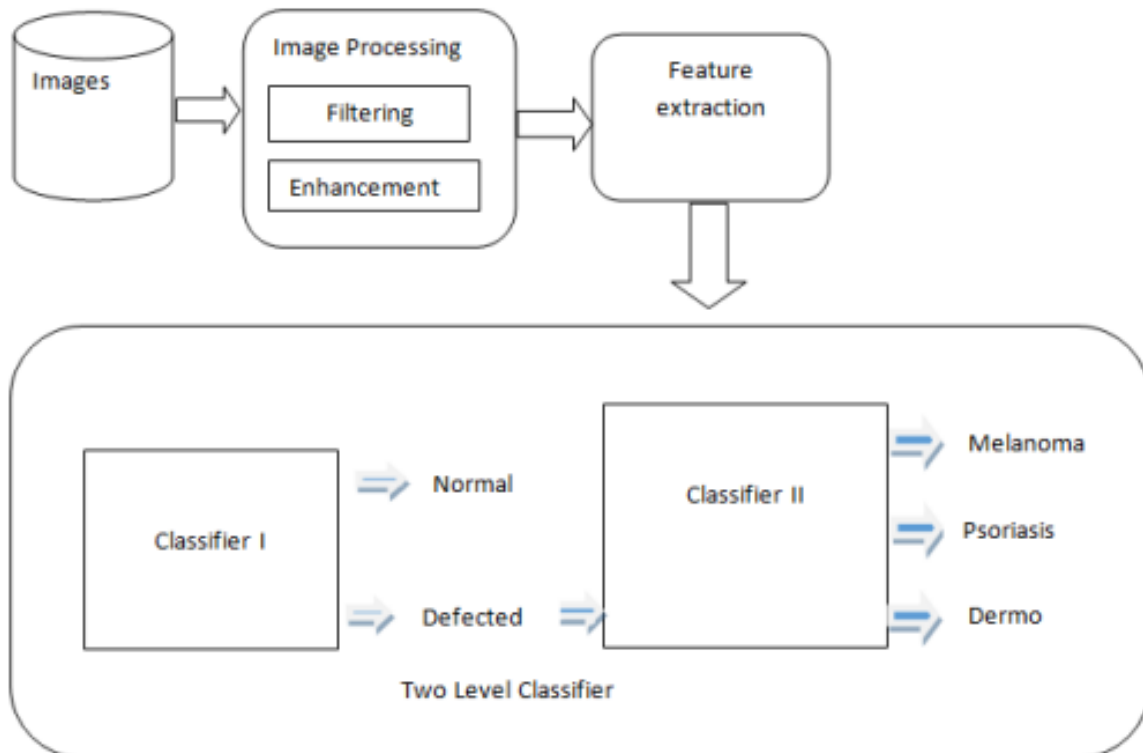


Figure 3.2: Architecture of proposed method [57]

3.3 Case Study

The aim of this study is to create a model using Deep Learning (DL) that can be combined with smartphone applications and to benefit from the radical improvement in the visual and sensory capabilities of phones over the past decade. Network imaging has evolved rapidly over the past 160 years and is now the mainstay of clinical care and management of patients with retinal and systemic diseases. Fundus photography is widely

used to widely detect Diabetic Retinopathy (DR), age-related glaucoma and macular degeneration. With the radical improvement in the visual and sensory capabilities of smartphones over the past decade, the use of smartphones in fundus imaging has become a more powerful clinical tool, especially in low-resource areas where advanced information infrastructure and remote imaging systems are not available.

Smartphone photography and video shooting have been used at increased frequency since 2010, as first reported by Lord and colleagues. Currently, there are at least five known apparatus/software package solutions on the market: DigiSight Paxos Scope (San Francisco, USA), Peek Retina (London, UK), D-EYE (Padova, Italy), Remidio Fundus on Phone (Bangalore, India), and Welch Allyn Panoptic with iExaminer (Skaneateles Falls, USA) [58].



(a) Handheld Remidio



(b) iExaminer

Figure 3.3: Handheld Remidio On Phone [59] and iExaminer Allyn [?]

3.4 System Design

3.4.1 CNN Architecture

Simple CNN Architecture

The model contains 3 blocks CONV and 2 Fully Connected layers :

first block we apply 16 filters of (3,3) kernel size, activation function Relu and Max Pooling of (2,2) that reduce size of the image

output size of first block (16,3,3) 16 :channels, 3 second bloc we use 32 filters, same activation function and pooling, last block we use 64 filters, then fully connected we

applied 3 Max Pooling of (2,2) the size of image (224,224) will be (28,28) and we have 64 filters so $(64 \times 28 \times 28) = 50,176$

512 it's just hyper-parameter that we chose finally last fully connected layer (512, 5) where the 5 is a number of classes

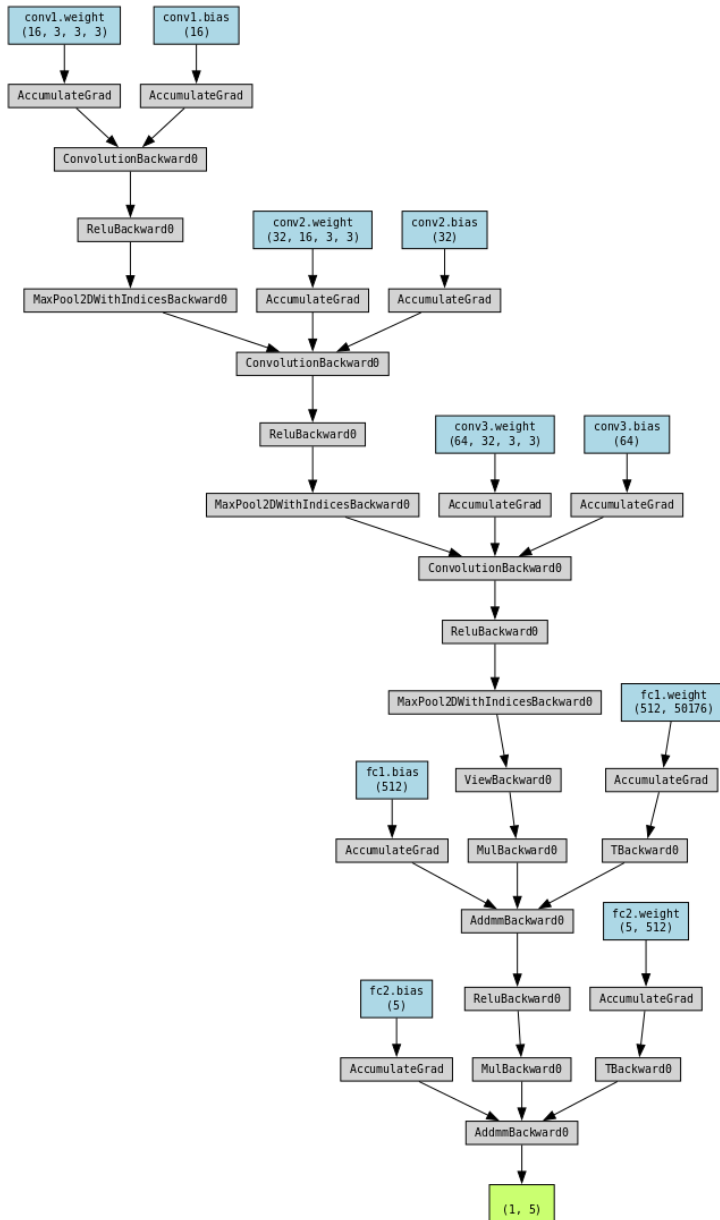


Figure 3.4: CNN Architecture

3.4.2 Transfer Learning

Transfer learning (TL) is a research problem in machine learning (ML) that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem [60]. transfer learning is the use of pretrained models in image classification tasks, which has increased lately, due to the speed while training and the efficiency these models provide, this is due to the pretrained weights. in this work, we will try some pretrained models to classify our images and see if these models will give better results.

Pretrained Model Used : (3.1)

ResNet 50 :

Many people consider the ResNet architecture to be the top choice from all Convolutional Neural Network models. Microsoft Research presented Residual Networks (ResNet) in 2015 and the network immediately set new records by performing better than any previous deep learning model [61].

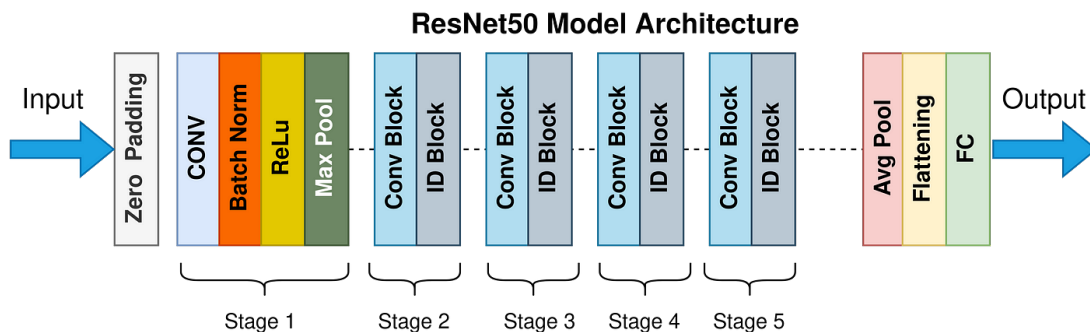


Figure 3.5: ResNet 50 [62].

Dense-Net :

Dense-Net uses connections between each layer to boost how well convolutional neural networks work. It has proven to be very efficient in carrying out tasks such as image classification, object detection and segmentation in computer vision. It is applied in several image-related areas, like face recognition, telling the type of an animal, searching for items in photos and detecting cancer cells [63].

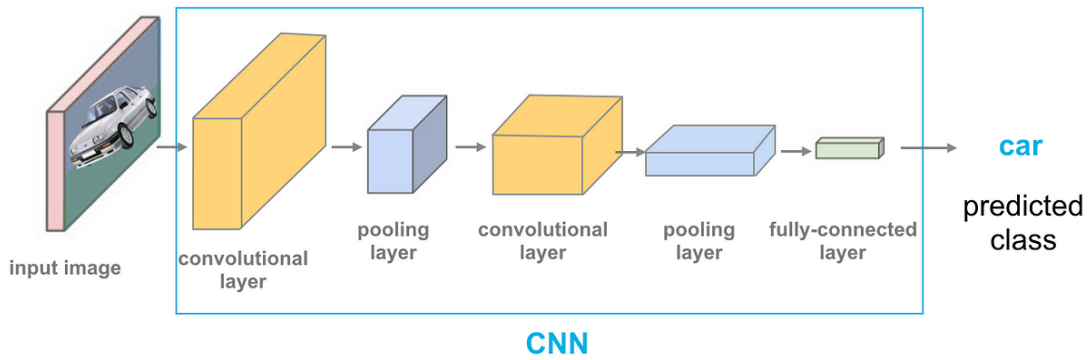


Figure 3.6: DenseNet [64].

Efficient-Net :

Efficient-Net was developed using the technique called "compound scaling." It deals with the longstanding issue of choosing model size, how accurate it is and how much it can be computed efficiently. The concept of compound scaling is to increase the network's width, depth and resolution [65].

EfficientNet Architecture

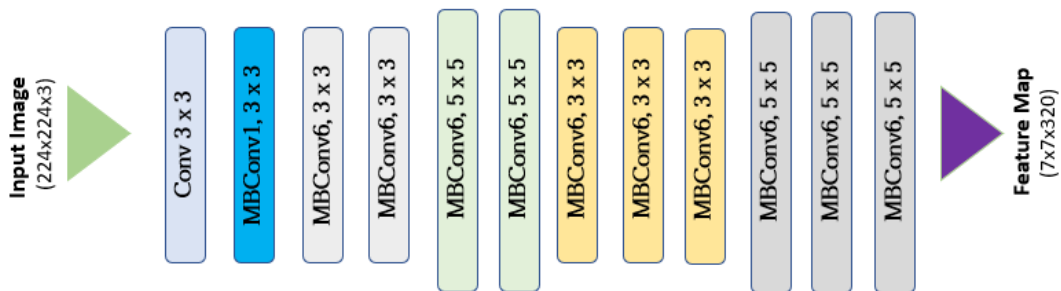


Figure 3.7: Efficient-Net [66].

3.5 Conclusion

In this chapter we talked about related work and our design and the two ways to solve problem like (related work) using (ML) SVM and our system where we chose (DL) CNN and Transfer Learning.

Now that we understood all this, let's dive into the implementation where we see more challenging things.

Chapter 4

Implementation and Results

Chapter 4

Implementation and Results

4.1 Introduction

after describing advanced work in the last chapter. This chapter covers how to apply the actions to create the suggested system. The first part focuses on picking the dataset, showing some of its photos and listing the needed development tools. Before looking at some lines of code, we will look at each experience and its results.

4.2 Implementation frameworks and tools

Top artificial intelligence progress depends on having strong deep learning frameworks that support preprocessing, grouping and evaluation with their in-built tools and prebuilt templates. Because of these frameworks, scientists can put more focus on the main parts of their work and not have to start over each time they start a new project.

The different development frameworks and tools discussed in the thesis are explained here.

4.2.1 Python

The most used and famous programming language in data science, is a high-level programming language, and its core design philosophy is all about code readability and syntax which allows programmers to express concepts in a few lines of code. Python is developed under an OSI-approved open-source license, making it freely usable and distributable, even for commercial use. It is used successfully in thousands of real-world business applications around the world, including many large and mission-critical systems. The python version used in this work is 3.10.12 [67].

```
✓ [1] import sys
0s print("python version"+sys.version)

python version3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0]
```

Figure 4.1: python version used in colab.

4.2.2 Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. Matplotlib was originally written by John D. Hunter. Everything in Matplotlib is organized in a hierarchy. At the top of the hierarchy is the Matplotlib state-machine environment, which is provided by the Matplotlib.pyplot module. At this level, simple functions are used to add plot elements (lines, images, text, etc.) to the axes in the Figure. Pyplot's state-machine environment behaves similarly to MATLAB and should be most familiar to users with MATLAB experience [68].



Figure 4.2: Examples of Matplotlib works. [68]

4.2.3 Tensorflow

An open-source machine learning library from Google made for use in both scientific research and applied projects, using Python and C++ as its programming languages. Considered to be the most frequently used techniques for deep learning and with a lot of activity in GitHub. Very generally, TensorFlow is a Python package designed to help people express arbitrary computation is shown as a network of data movement [69].

4.2.4 PyTorch

PyTorch combines the machine learning (ML) library of Torch with a Python-based top-level API to create neural networks. Because it is easy to use and adaptable, the leading ML framework is popular in academic and research circles [70].

4.2.5 Keras

Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear and actionable feedback upon user error. Keras is also a favorite among deep learning researchers, coming in 2nd in terms of mentions in scientific papers after tensorflow [71]. Keras uses as Backend tensorflow and Theano, we have used Keras with tensorflow back end in the implementation part of this work.

4.2.6 Kaggle

Kaggle provides a place online for data scientists and professionals in machine learning and it is owned by Google LLC. At the beginning, Kaggle organized machine learning competition and now has a public data platform, an online workbench for data science and short form credentials. AI education. Kaggle Kernels allows analysts to work together and repeat their analysis on the cloud. Kernels enables scripting in R and Python, Jupyter Markdown files and reports are part of the data. Kagglers can use Kaggle Kernels and get a free-of-charge access to Cloud GPU NVIDIA K80 with 16 GB of RAM, each session lasts for 6 hours [72].

Lately in 2019, a Kaggle challenge appeared and provided the motivation for our current work. Photo interpretation detecting blindness in 2019 [73].

4.2.7 Google Colab

Google Colab is a free cloud service with free GPU and TPU 2 with a limit of 12 hours per session and 12 GB of RAM limits. The execution of deep models with complicated architecture and huge data (as in our case) can lead to the use of extreme amounts of hardware materials, Google Colabs provides a free GPU, currently they provide GPU with specs Tesla K80, having 2496 CUDA cores, compute 3.7, 12 GB (11.439 GB Usable) GDDR5 VRAM. and a CPU of CPU: 1xsingle core hyper threaded i.e(1 core, 2 threads) Xeon Processors @2.3Ghz (No Turbo Boost), 45 MB Cache [74].

Why do we need GPU in deep learning ?

A graphics processing unit (GPU) is a kind of processor that is particularly useful for certain calculations. On the other hand, a central processing unit (CPU) is a type of processor. able to solve basic calculation tasks. Most of the common tasks our electronic devices do are powered by CPUs. Because of its design, GPUs can process tasks at a higher speed than CPUs. Even so, this does not happen in all cases.

The relative speeds of a GPU and a CPU change depending on the computations being done. The kind of computation ideal for a GPU is one that can be done at the same time on many processors. It describes a computation where problems are split into pieces and worked on independently. Computation is done by dividing it into many independent operations that can be carried out together.

The calculations are then put together or coordinated, to make up the final result. the initial computation that was broken down. The tasks that can be created from breaking a large task into parts the number of cores in the hardware decides this. It is the cores that perform computation in the processor and each CPU generally CPUs have four, eight or sixteen cores and GPUs can have many more [75].

4.2.8 Cross Validation

To find out how a model will work on unseen data, machine learning often uses cross validation. It is done by dividing the data into several groups or subsets, using one group for validation and training the model with the others. This is done many times, as the fold you use to validate changes with every round. Each validation method is tested and the results are combined to give a strong estimate of what the model can do. Cross validation makes the chosen model for deployment stronger and assures that it will work well on new data [76].

4.2.9 Model evaluation metrics

This subsection discusses how we can evaluate our model results, what makes a model better than another and how we can predict that our model will give good results in the testing .

Many evaluation metrics to evaluate classification output quality.

4.2.9.1 Accuracy

Accuracy is the number of correctly predicted data points out of all the data points. More formally, it is defined as the number of true positives and true negatives divided

by the number of true positives, true negatives, false positives and false negatives, as it summarized in the equation: 4.1.

$$\text{Accuracy} = \frac{TN + TP}{TN + TP + FN + FP} \quad (4.1)$$

A true positive or true negative falls into the category the algorithm was meant to identify. Is the statement true or is it false? A model can make errors and a false positive or false negative is an error made by the algorithm incorrectly classified.

4.2.9.2 Recall

Named also Sensitivity, Recall is the ratio of correctly predicted positive observations to the all observations in actual class, in other way, recall measures the proportion of actual positives that are correctly identified.

the formula:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4.2)$$

4.2.9.3 Precision

Out of all the classes, how much we predicted correctly. It should be high as possible.

the formula:

$$\text{Recall} = \frac{TP}{TP + FP} \quad (4.3)$$

4.2.9.4 Implementation

As we have mentioned before, this data set is preprocessed so we don't have to do the preprocessing step

Any model trained should pass through these three steps:

- 1: Data loading and Preprocessing
- 2: Define Model Architecture
- 3: Start training and validation

1: Data Loading and Preprocessing :

```

1: class DiabetMessidorEyeDataSet(Dataset):
    def __init__(self, data_dir, transform=None):
        self.data = ImageFolder(data_dir, transform=transform)

    def __len__(self):
        return len(self.data)

    def __getitem__(self, idx):
        return self.data[idx]

    @property
    def classes(self):
        return self.data.classes

```

Figure 4.3: Data Loader class

2: Define Model Architecture (code)

```

1: class DiabetModel(nn.Module):
    def __init__(self, num_classes=5):
        super(DiabetModel, self).__init__()
        # Define the layers of the model
        self.conv1 = nn.Conv2d(3, 16, kernel_size=3, padding=1)
        self.conv2 = nn.Conv2d(16, 32, kernel_size=3, padding=1)
        self.conv3 = nn.Conv2d(32, 64, kernel_size=3, padding=1)
        self.pool = nn.MaxPool2d(2, 2)
        self.fc1 = nn.Linear(64 * 28 * 28, 512)
        self.fc2 = nn.Linear(512, num_classes)
        self.dropout = nn.Dropout(0.5) # Increased dropout

    def forward(self, x):
        # Apply convolutions, followed by batch normalization, activation, and pooling
        x = self.pool(F.relu(self.conv1(x)))
        x = self.pool(F.relu(self.conv2(x)))
        x = self.pool(F.relu(self.conv3(x)))
        # Flatten the tensor for the fully connected layer
        x = x.view(-1, 64 * 28 * 28)
        x = self.dropout(x)
        x = F.relu(self.fc1(x))
        x = self.dropout(x)
        x = self.fc2(x)
        return x

```

Figure 4.4: Simple CNN Architecture

3 : Training and Validation

```

model1.to(device)
num_epochs = 3
scaler = GradScaler()
train_losses, val_losses = [], []
for epoch in range(num_epochs):
    model1.train()
    running_loss = 0.0
    correct_train = 0
    total_train = 0
    for images, labels in tqdm(train_loader, "Training Loop"):
        images, labels = images.to(device), labels.to(device)
        optimizer1.zero_grad()
        with autocast():
            output = model1(images)
            loss = criterion1(output, labels)
            _, predicted = torch.max(output, 1)
            correct_train += (predicted == labels).sum().item()
            total_train += labels.size(0)
        scaler.scale(loss).backward()
        scaler.step(optimizer1)
        scaler.update()
        running_loss += loss.item() * labels.size(0) # Use loss.item() to get the scalar value
    train_loss = running_loss / len(train_loader.dataset)
    train_accuracy = correct_train / total_train
    train_losses.append(train_loss)

```

Figure 4.5: Training Loop

```

model1.eval()
running_loss = 0.0
correct_val = 0
total_val = 0
true_labels = []
predicted_labels = []
with torch.no_grad():
    for images, labels in tqdm(val_loader, 'Validation Loop'):
        images, labels = images.to(device), labels.to(device)
        with autocast():
            output = model1(images)
            loss = criterion1(output, labels)
            _, predicted = torch.max(output, 1)
            correct_val += (predicted == labels).sum().item()
            total_val += labels.size(0)
            true_labels.extend(labels.cpu().numpy())
            predicted_labels.extend(predicted.cpu().numpy())
        running_loss += loss.item() * labels.size(0) # Use loss.item() to get the scalar value
    val_loss = running_loss / len(val_loader.dataset)
    val_accuracy = correct_val / total_val
    val_losses.append(val_loss)
    print(f"""
Epoch {epoch+1}/{num_epochs} :
Train loss: {train_loss}
Train accuracy: {train_accuracy}
Validation loss: {val_loss}
Validation accuracy: {val_accuracy}""")

```

Figure 4.6: Validation Loop

4 : Result Of Simple CNN

```

Epoch 1/3 :
Train loss: 1.4416285196940104
Train accuracy: 0.3472222222222222
Validation loss: 1.1210801866319444
Validation accuracy: 0.4555555555555555
Loading widget...
Loading widget...

Epoch 2/3 :
Train loss: 1.150192419687907
Train accuracy: 0.43194444444444446
Validation loss: 1.1229390462239583
Validation accuracy: 0.48333333333333334
Loading widget...
Loading widget...

Epoch 3/3 :
Train loss: 1.1215344746907552
Train accuracy: 0.43611111111111111
Validation loss: 1.0646538628472222
Validation accuracy: 0.46388888888888889

```

Figure 4.7: Result

As we see no validation loss improvement, so more epochs it's just an over fitting to the data.

What about transfer learning??

5: Transfer Learning Approach

```

class DiabetModel(nn.Module):
    def __init__(self , num_classes = 5):
        super(DiabetModel, self).__init__()
        self.base_model = timm.create_model('efficientnet_b4', pretrained=True)
        self.features = nn.Sequential(*list(self.base_model.children())[:-1])

        out_size = 1792
        self.classifier = nn.Sequential(
            nn.Flatten(),nn.Dropout(0.3),nn.Linear(out_size , num_classes)
        )

    def forward(self , x) :
        x= self.features(x)
        output = self.classifier(x)
        return output

```

Figure 4.8: Model Architecture

6 : Result of Transfer Learning

```

Epoch 1/3 :
Train loss: 0.5870658801661597
Train accuracy: 0.7673611111111112
Validation loss: 1.0642919699350992
Validation accuracy: 0.5722222222222222
loading widget...
loading widget...

Epoch 2/3 :
Train loss: 0.2623870308200518
Train accuracy: 0.9111111111111111
Validation loss: 1.0390099419487848
Validation accuracy: 0.6944444444444444
loading widget...
loading widget...

Epoch 3/3 :
Train loss: 0.16450613116224608
Train accuracy: 0.9423611111111111
Validation loss: 1.4902902709113226
Validation accuracy: 0.65

```

Figure 4.9: Transfer Learning Approach

Ok it's a little bit of batter but not what we expect, let's try cross-validation

7 : Cross Validation

```

Epoch 5: val_loss did not improve from 0.25131
21/21 ————— 6s 286ms/step - accuracy: 0.7805 - loss: 0.6461 - val_accuracy: 0.930
6 - val_loss: 0.2543 - learning_rate: 1.0000e-08
Epoch 5: early stopping
Restoring model weights from the end of the best epoch: 1.
Average training loss: 0.7301204562187195
Average training accuracy: 0.7387607395648956
Average validation loss: 0.7935424953699112
Average validation accuracy: 0.6694444477558136
Maximum training accuracy: 0.939393937587738
Maximum validation accuracy: 0.9583333134651184

```

Figure 4.10: Transfer Learning Approach

The result is amazing 95% validation accuracy

sorry to tell you it's an overfitting, as we know cross-validation changes samples, so the model memorize all samples of our dataset when we tested it was worse than the previous model

So what is the problem and how we can solve it ???

the problem is our dataset it's too small we can't find more data just in Google because it's a medical problem so we searched in Kaggle for more data, and finally, we found another one.

4.2.10 Second dataset

4.2.10.1 About Data

This is augmented data that contains the same classes with different images; it's 10 times bigger than the previous one found in Kaggle.

Link to Data : [\[77\]](#)

4.2.10.2 Implementation

Now that we know how it's work, let's dive in directly

1 : Data Preprocessing

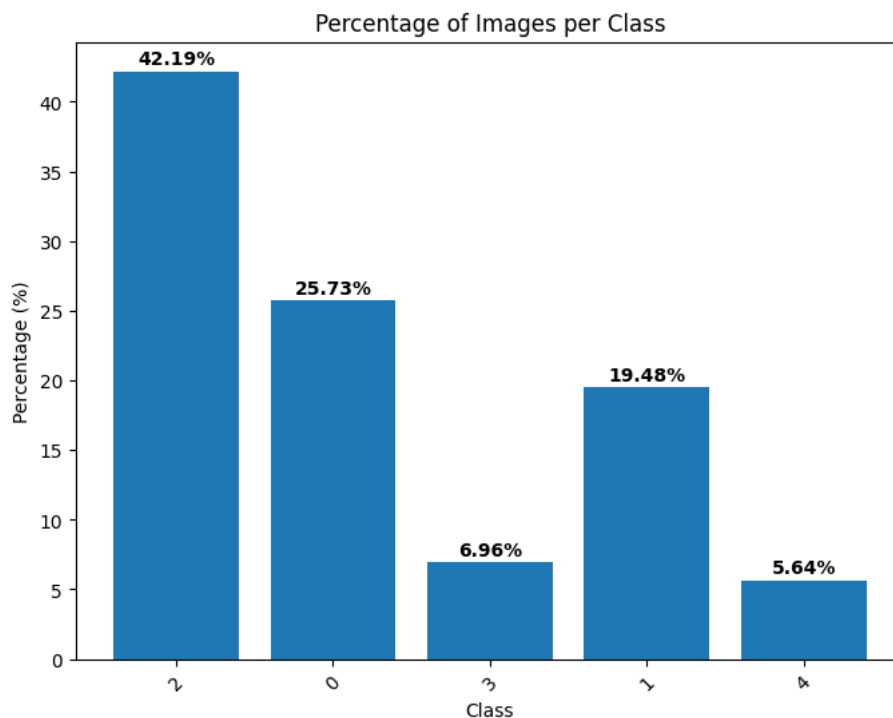


Figure 4.11: Data Visualisation

Data imbalance is present. This challenge affects our model in a major way. Imagine it in this way:

for example, if you teach a child basic math skills. Letting him practice only multiplication and less division means he'll learn multiplication well and division not as much. This problem also appears with models and datasets where one category is much larger than the other. So what options do we have to choose from ????

2 : Solution Proposed :

1 : SMOTE The name SMOTE comes from the phrase Synthetic Minority Over-sampling Technique. The technique is used before training to manage class imbalance. It achieves balance by producing more cases for the minority class [78].

We tried to apply this solution, but it consumes a lot of memory space, and we were unable to run it in our notebook it's stop

2 : Change classes weight in CrossEntropyLoss :

This will simply pay attention to minority classes more than others, we can modify it as we like

```
classes_weights = torch.tensor([25.441205631568508,  
                                19.551854055125915,  
                                42.692841562561966,  
                                6.702359706523894,  
                                5.611739044219711])  
  
criterion = torch.nn.CrossEntropyLoss(weight=classes_weights).to(device)
```

Figure 4.12: Changing Weights

3 : Result of Simple CNN (Used In Previous Data) :

```

Training Loop: 100% ██████████ 5016/5016 [21:59<00:00, 4.09it/s]
Validation Loop: 100% ██████████ 440/440 [02:21<00:00, 3.12it/s]
Epoch 1/5 - Train loss: 1.0315, Train accuracy: 0.6161, Validation loss: 0.8531, Validation accuracy: 0.7335
Training Loop: 100% ██████████ 5016/5016 [14:26<00:00, 5.71it/s]
Validation Loop: 100% ██████████ 440/440 [01:38<00:00, 4.52it/s]
Epoch 2/5 - Train loss: 0.9732, Train accuracy: 0.6272, Validation loss: 0.8513, Validation accuracy: 0.7330
Training Loop: 100% ██████████ 5016/5016 [14:24<00:00, 5.91it/s]
Validation Loop: 100% ██████████ 440/440 [01:39<00:00, 4.30it/s]
Epoch 3/5 - Train loss: 0.9391, Train accuracy: 0.6355, Validation loss: 0.8437, Validation accuracy: 0.7335
Training Loop: 100% ██████████ 5016/5016 [14:02<00:00, 5.98it/s]
Validation Loop: 100% ██████████ 440/440 [01:38<00:00, 4.60it/s]
Epoch 4/5 - Train loss: 0.8989, Train accuracy: 0.6462, Validation loss: 0.8302, Validation accuracy: 0.7327
Training Loop: 100% ██████████ 5016/5016 [14:04<00:00, 5.48it/s]
Validation Loop: 100% ██████████ 440/440 [01:38<00:00, 4.73it/s]
Epoch 5/5 - Train loss: 0.8512, Train accuracy: 0.6643, Validation loss: 0.8312, Validation accuracy: 0.7340
Training complete
    
```

Figure 4.13: Result Of Training with Simple CNN

4 : Result of Transfer Learning :

```

Training Loop: 100% ██████████ 5016/5016 [27:33<00:00, 3.03it/s]
Validation Loop: 100% ██████████ 440/440 [01:59<00:00, 3.69it/s]
Epoch 1/5 - Train loss: 0.7687, Train accuracy: 0.6936, Validation loss: 0.6366, Validation accuracy: 0.7872
Training Loop: 100% ██████████ 5016/5016 [28:08<00:00, 2.97it/s]
Validation Loop: 100% ██████████ 440/440 [01:57<00:00, 3.74it/s]
Epoch 2/5 - Train loss: 0.4095, Train accuracy: 0.8471, Validation loss: 0.6531, Validation accuracy: 0.7911
Training Loop: 100% ██████████ 5016/5016 [27:53<00:00, 3.00it/s]
Validation Loop: 100% ██████████ 440/440 [01:56<00:00, 3.78it/s]
Epoch 3/5 - Train loss: 0.2179, Train accuracy: 0.9209, Validation loss: 0.7475, Validation accuracy: 0.7777
    
```

Figure 4.14: Result Transfer Learning

Left and Right Eyes Models : (4.4)

Explanation :

Actually, when we looked at the data and the names of the images, we noticed that the eyes are split too (left and right). Ok, so what's the problem? The problem is that images are only matrices, and feature extraction works with numbers, so the features of the left eye aren't like those of the right eye, even if the two eyes are the same class, for example, 'No DR'. Same class, but different features.

That will cause bad predictions even if we get good accuracy, so what we can do is build two models and combine them in one model.

The output is an array of 5 values that contain the probability of each class, so he will loop item by item and get the probability of each class.

Example :

let tell we pass an image of (left eye), class : (4: Proliferative DR)

The output of the two models will be like this:

left_model : [0.1, 0.3, 0.3, 0.5, 0.8]

right_model : [0.2, 0.4, 0.4, 0.5, 0.4]

left model gets 0.8 (80%) for class 4 and the right model only 0.4, but the average is the max value

The value be always the max one, that's why it's better.

Implementation

We use for left_model and right_model The same architecture used for the previous model

Class Combined Model

```
class CombinedModel(nn.Module):
    def __init__(self, model1, model2):
        super(CombinedModel, self).__init__()
        self.model1 = model1
        self.model2 = model2

    def forward(self, x):
        output1 = self.model1(x)
        output2 = self.model2(x)

        combined_output = (output1 + output2) / 2 # Average the predictions

        return combined_output

full_model = CombinedModel(left_model , right_model)
```

Figure 4.15: Combine Model

Results : (4.5)

Left Model :

```

model.safetensors: 100% ██████████ 77.9M/77.9M [00:00<00:00, 116MB/s]
Training Loop: 100% ██████████ 2495/2495 [12:49<00:00, 3.33it/s]
Validation Loop: 100% ██████████ 218/218 [00:53<00:00, 4.21it/s]
Epoch 1/2 - Train loss: 0.8318, Train accuracy: 0.6686, Validation loss: 0.6734, Validation accuracy: 0.7810
Training Loop: 100% ██████████ 2495/2495 [12:48<00:00, 3.63it/s]
Validation Loop: 100% ██████████ 218/218 [00:52<00:00, 4.25it/s]
Epoch 2/2 - Train loss: 0.4919, Train accuracy: 0.8135, Validation loss: 0.6994, Validation accuracy: 0.7735
Training complete

```

Figure 4.16: Left Model

Right Model :

```

model.safetensors: 100% ██████████ 77.9M/77.9M [00:00<00:00, 116MB/s]
Training Loop: 100% ██████████ 2495/2495 [12:49<00:00, 3.33it/s]
Validation Loop: 100% ██████████ 218/218 [00:53<00:00, 4.21it/s]
Epoch 1/2 - Train loss: 0.8318, Train accuracy: 0.6686, Validation loss: 0.6734, Validation accuracy: 0.7810
Training Loop: 100% ██████████ 2495/2495 [12:48<00:00, 3.63it/s]
Validation Loop: 100% ██████████ 218/218 [00:52<00:00, 4.25it/s]
Epoch 2/2 - Train loss: 0.4919, Train accuracy: 0.8135, Validation loss: 0.6994, Validation accuracy: 0.7735
Training complete

```

Figure 4.17: Right Model

Combined Model :

```

model.safetensors: 100% ██████████ 77.9M/77.9M [00:00<00:00, 116MB/s]
Training Loop: 100% ██████████ 2495/2495 [12:49<00:00, 3.33it/s]
Validation Loop: 100% ██████████ 218/218 [00:53<00:00, 4.21it/s]
Epoch 1/2 - Train loss: 0.8318, Train accuracy: 0.6686, Validation loss: 0.6734, Validation accuracy: 0.7810
Training Loop: 100% ██████████ 2495/2495 [12:48<00:00, 3.63it/s]
Validation Loop: 100% ██████████ 218/218 [00:52<00:00, 4.25it/s]
Epoch 2/2 - Train loss: 0.4919, Train accuracy: 0.8135, Validation loss: 0.6994, Validation accuracy: 0.7735
Training complete

```

Figure 4.18: Combined Model

Actually, these results are good, but let's see the confusion metrics to understand why these are bad results not good

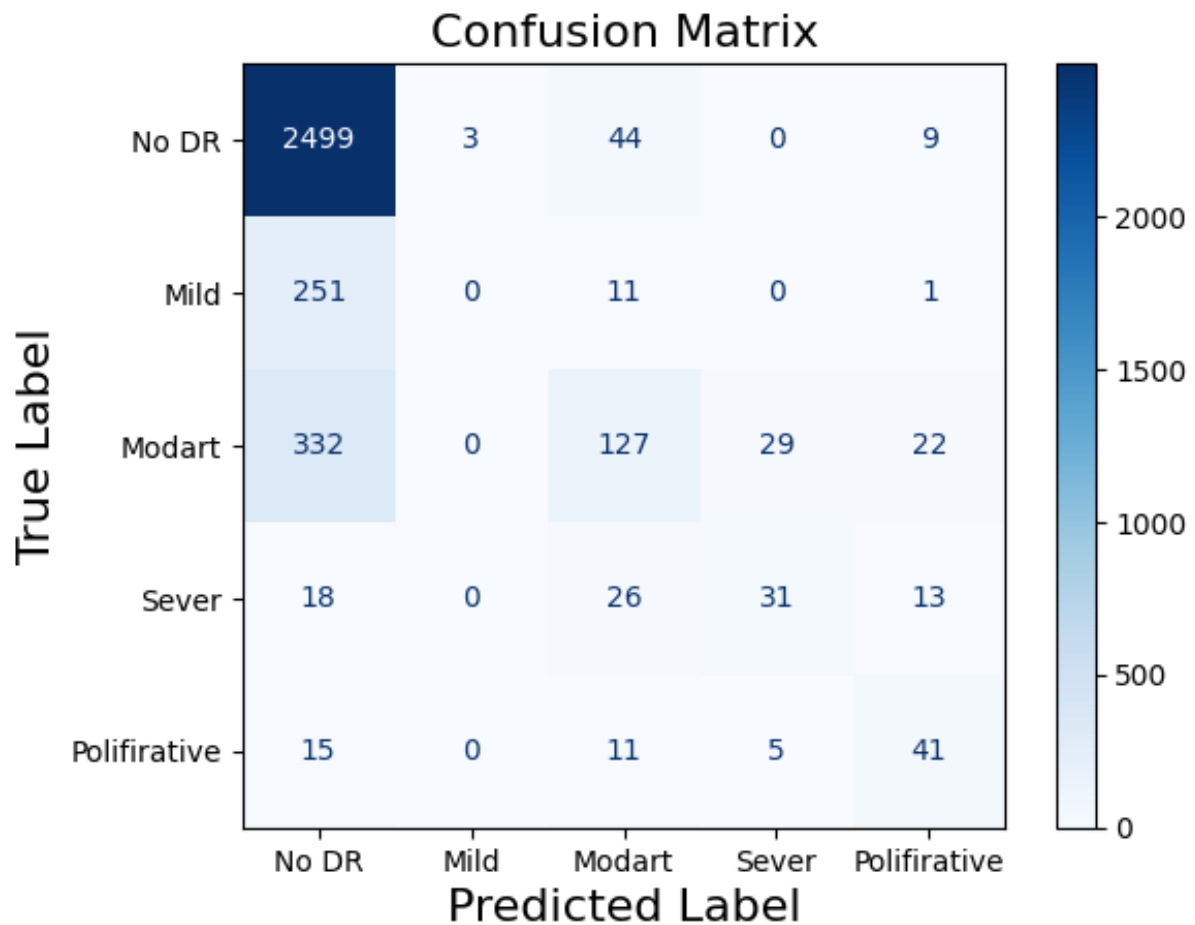


Figure 4.19: Confusion Matrix

Explanation :

to understand how the results are bad, see the predicted values are 2499 images in class (No DR) and are the true values to, so it's a good prediction

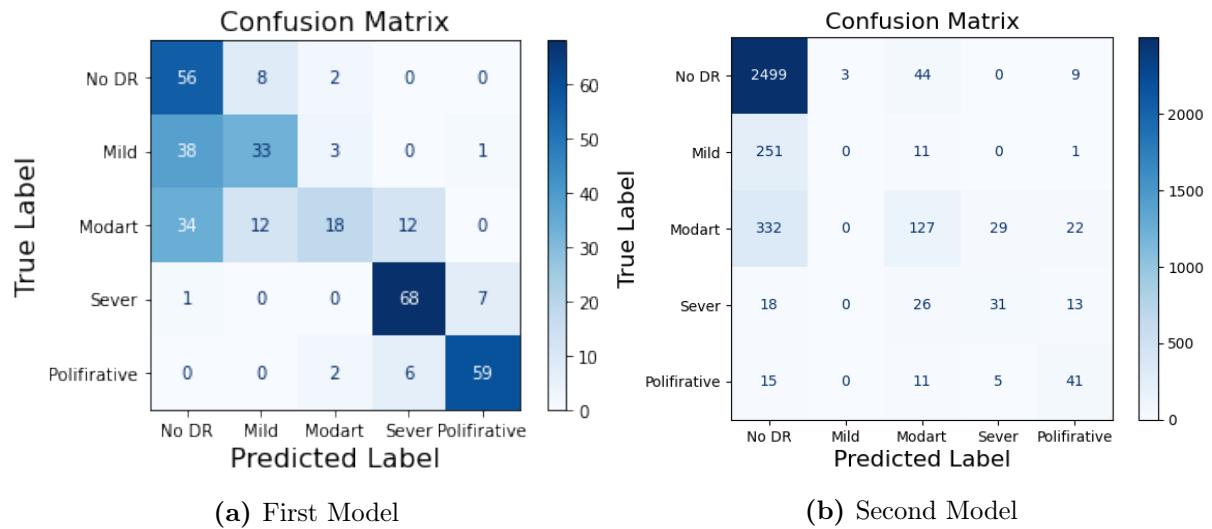
but let's see (Mild) and (Moderate) (215 and 332) are predicted as (No DR)

so how we get this accuracy is simple because data are imbalanced it's classified the other classes in (No DR) class

the real (No DR) class contains a lot of images, so it gives 77% accuracy

Comparison : (4.6)

let's compare two final models that we build using first data and second data



as we see first model is more stable and predicts more (Sever) and (Proliferative) values so it detects the disease

4.3 Application

4.3.1 Tools Used

4.3.1.1 Rect-Native :

Developing good and efficient mobile applications is made possible largely because of mobile development frameworks. Picking the correct framework is important for both the quality of development and the experience users have. Developers worldwide use React Native as a leading mobile development framework. The functionality and overall power of features are strong for iOS and Android users [79].

4.3.1.2 Expo :

Expo lets you build Android and iOS apps in a simpler way. It includes file-based routing, a basic set of native modules and several other features. The Expo community is active on both GitHub and Discord, since Expo is open source [80].

4.3.1.3 TensorFlow Lite :

The tools in TensorFlow Lite support implementing machine learning on phones, smart devices and edge computers [81].

4.4 Conclusion

In this chapter, we found out that, in deep learning, the dataset is what gives us the most power. To increase the accuracy, we have used several methods and have done a lot of other things (SMOTE, processing data by sight and so on). Even so, it's very difficult to deal with this problem if your data is small or not well-balanced.

Conclusion

General Conclusion

Our final paper explained how to diagnose skin diseases, including those that may lead to acne and psoriasis, using a deep learning (DL) system.

We started by studying skin diseases in general, then came the study of the human skin and two specific dermatological concerns such as acne and psoriasis, along with their symptoms and ways of diagnosis. This led us to highlight relevant ideas in machine learning (ML) and deep learning (DL) and we started thinking of possible answers to our problem. What other studies have explored this subject was also considered.

As part of our effort, we suggested using Convolutional Neural Networks (CNN). A classifier is used to assign one of five categories (0, 1, 2, 3 and 4) for determining a skin disease. We first set up the training data and focused on two main parts of the study: The first part was training the CNN model (training phase) and the second part was using it for predicting (inference phase).

The proposed system comprises two modules:

Module 1: transfer learning model with small data

Module 2: transfer learning model with imbalanced data

Even with these difficulties like small numbers of training samples, one group having more examples than others and the quality of images being less than ideal, research has continued. The most serious challenges were faced when getting the data ready and doing the testing and there was not enough powerful hardware for training which took quite a bit of time.

We intend to proceed by solving further issues and by developing new models using several different datasets to identify more skin diseases in the future. Also, we want to incorporate our model into dermatoscopic imaging machines to help dermatologists with their diagnoses. We hope to bring our model to smartphones with imaging devices, so patients can complete a first check of their skin directly at home.

Perspectives :

Working toward a large and varied database helps the model become more solid and accurate. Potential trends for the future are:

Collecting more-diverse and better-quality samples and adding these to the data set help fine-tune and improve the performance of a model.

Saving Computation: Refining the structure of the Convolutional Neural Network (CNN) to ensure the model works quickly and uses minimal resources, making it favorable for real-time tasks.

Using the system within telemedicine to offer remote diagnostic services to people who live in areas far from medical centers.

User-Friendly Design: Building both mobile and desktop apps that are simple for patients and healthcare workers to use with the diagnostic AI.

Extensive checks are performed by applying the model to many demographically diverse data sets to make sure the model can be trusted.

By adopting these perspectives, we hope to produce major improvements in automated skin disease diagnosis and thus help improve

Bibliography

- [1] Cleveland Clinic. Skin diseases. <https://my.clevelandclinic.org/health/diseases>. Accessed: 2025-04-07.
- [2] Harvard Health Publishing. Inflammatory skin conditions: eczema, seborrheic dermatitis, and psoriasis. <https://www.health.harvard.edu/diseases-and-conditions/inflammatory-skin-conditions-eczema-seborrheic-dermatitis-and-psoriasis>. Accessed: 2025-04-08.
- [3] Webteb. Webteb website. <https://www.webteb.com/>. Accessed: 2025-04-08.
- [4] Healthline. Skin infection. <https://www.healthline.com/health/skin-infection>. Accessed: 2025-04-08.
- [5] MedlinePlus. Skin infections. <https://medlineplus.gov/skininfections.html>. Accessed: 2025-04-08.
- [6] Wikipedia. Skin infection. https://en.m.wikipedia.org/wiki/Skin_infection. Accessed: 2025-04-08.
- [7] Healthline. Skin disorders. <https://www.healthline.com/health/skin-disorders>. Accessed: 2025-04-08.
- [8] Gainesville Dermatology & Skin Surgery. Chronic skin conditions. <https://www.gainesvilledermatologyskinsurgery.com/chronic-skin-conditions/>. Accessed: 2025-04-08.
- [9] Gina M. DeStefano and Angela M. Christiano. Cold spring harbor perspectives in medicine. *Cold Spring Harbor Perspectives in Medicine*, 4(10):a015172, 2014.
- [10] National Center for Biotechnology Information (NCBI). Anatomy, skin (integument), epidermis. <https://www.ncbi.nlm.nih.gov/books/NBK470464/>. Accessed: 2025-04-08.

- [11] Merck Manuals. Structure and function of the skin. <https://www.merckmanuals.com/home/skin-disorders/biology-of-the-skin/structure-and-function-of-the-skin>. Accessed: 2025-04-08.
- [12] Medscape. Skin disorders overview. <https://emedicine.medscape.com/article/1294744-overview>. Accessed: 2025-04-08.
- [13] Health.com. Skin conditions. <https://www.health.com/condition/skin-conditions/>. Accessed: 2025-04-08.
- [14] Yale Medicine. Skin problems. <https://www.yalemedicine.org/conditions/topics/skin-problems>. Accessed: 2025-04-08.
- [15] NIAMS. Skin diseases. <https://www.niams.nih.gov/health-topics/skin-diseases>. Accessed: 2025-04-08.
- [16] DermNetNZ. Rosacea images. <https://dermnetnz.org/images?query=Rosacea>. Accessed: 2025-04-08.
- [17] D. Lipsker. *Clinical Examination and Differential Diagnosis of Skin Lesions*. Springer Paris, London, 2013.
- [18] The journal of pain. *The Journal of Pain*, 18(7):757–757, 2017.
- [19] Technology Networks. An introduction to the light microscope, light microscopy techniques and applications. <https://www.technologynetworks.com/analysis/articles/an-introduction-to-thelight-microscope-light-microscopy-techniques-and-applications>. Accessed: 2025-04-08.
- [20] AZoM. Optical microscopy techniques. <https://www.azom.com/article.aspx?ArticleID=11631>. Accessed: 2025-04-08.
- [21] Lumen Learning. Instruments of microscopy. <https://courses.lumenlearning.com/suny-microbiology/chapter/instruments-of-microscopy>. Accessed: 2025-04-08.
- [22] Indian Journal of Dermatology, Venereology and Leprology (IJDVL). Polarized light microscopy. <https://ijdvl.com/polarized-light-microscopy>. Accessed: 2025-04-08.
- [23] DermNetNZ. Preparation of specimen. <https://dermnetnz.org/cme/dermatopathology/preparation-of-specimen>. Accessed: 2025-04-08.

- [24] University of Michigan Medicine. Introduction to histology stains: Periodic acid schiff (pas). <https://histology.medicine.umich.edu/resources/introduction-histologystains>. Accessed: 2025-04-08.
- [25] Wiley Online Library. Grocott or gomori methenamine silver stain techniques. *Cytopathology*, 2025. <https://onlinelibrary.wiley.com/doi/10.1111/cup.13769>.
- [26] Lab Tests Guide. Microscopic examination sops and quality control. <https://www.labtestsguide.com/microscopic-examinationsops>. Accessed: 2025-04-08.
- [27] *Handbook of Cosmetic Science and Technology*. CRC Press, United States, 2022.
- [28] W. Aitken. *The Science and Practice of Medicine, Volume 2*. C. Griffin, United Kingdom, 1864.
- [29] Flatiron School. Deep learning vs. machine learning, May 2021. Accessed on May 18, 2023.
- [30] Persado. Artificial intelligence vs. machine learning vs. data science, 2021. Accessed on May 18, 2023.
- [31] Mariette Awad and Rahul Khanna. *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. Apress, New York, NY, 2015.
- [32] Edureka. What is machine learning?: Types of machine learning. <https://www.edureka.co/blog/what-is-machine-learning/>, May 2019. Accessed: 2024-05-23.
- [33] Canburak Tumer. Machine learning basics with examples- part 2 supervised learning. <https://2u.pw/909RiIt6>, Aug 2018. Accessed: 2024-05-23.
- [34] Analytics Steps. Introduction to machine learning: Supervised and unsupervised learning. <https://2u.pw/HPBmNUhh>, 2020. Accessed: 2024-05-23.
- [35] Richard S Sutton and Andrew G Barto. *Introduction to Reinforcement Learning*, volume 135. MIT Press, Cambridge, MA, 1998.
- [36] MathWorks. Three things to know about reinforcement learning. <https://www.kdnuggets.com/2019/10/mathworks-reinforcement-learning.html>, Oct 2019. Accessed: 2024-05-24.
- [37] Jeff Heaton. Ian goodfellow, yoshua bengio, and aaron courville: Deep learning. *Genetic Programming and Evolvable Machines*, 19:305–307, 2018.

- [38] Harvey Lodish. Overview of neuron structure and function. <https://www.ncbi.nlm.nih.gov/books/NBK21535/>, Jan 1970. Accessed: 2024-05-24.
- [39] Kite. Introduction to artificial neural networks in python. <https://roioverflowload.com/biology-inspires-the-artificial-neural-network/>, Aug 2019. Accessed: 2024-05-24.
- [40] Boundless. Boundless psychology. <https://courses.lumenlearning.com/boundless-psychology/chapter/neurons/>. Accessed: 2024-05-24.
- [41] Andrej Krenker, Janez Bester, and Andrej Kos. Introduction to the artificial neural networks. In *Artificial Neural Networks: Methodological Advances and Biomedical Applications*, pages 1–18. InTech, 2011.
- [42] MissingLink. 7 types of activation functions in neural networks: How to choose? <https://missinglink.ai/guides/neural-network-concepts/7-types-neural-network-activation-functions-right/>. Accessed: 2024-05-24.
- [43] Amey Varangaonkar et al. Top deep learning architectures. <https://hub.packtpub.com/top-5-deep-learning-architectures/>, Sep 2018. Accessed: 2024-05-24.
- [44] Athanasios Voulodimos et al. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018:1–13, 2018.
- [45] Manjunath Jogin et al. Feature extraction using convolution neural networks (cnn) and deep learning. In *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 2319–2323. IEEE, 2018.
- [46] Ping-Rong Chen et al. Dsnet: An efficient cnn for road scene segmentation. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 424–432. IEEE, 2019.
- [47] Yann LeCun et al. Lenet-5, convolutional neural networks. <http://yann.lecun.com/exdb/lenet>, 2015. Accessed: 2024-05-26.
- [48] Deep convolutional neural networks (alexnet). https://d2l.ai/chapter_convolutional-modern/alexnet.html#architecture. Colab [mxnet] Open the notebook in Colab, Colab [pytorch] Open the notebook in Colab, Colab [tensorflow] Open the notebook in Colab. Accessed: 2024-05-26.

- [49] Kaiming He et al. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [50] Mingxing Tan and Quoc V Le. Efficientnet: Improving accuracy and efficiency through automl and model scaling. *arXiv preprint arXiv:1905.11946*, 2019.
- [51] Christian Szegedy et al. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.
- [52] Danilo Mandic and Jonathon Chambers. *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*. Wiley, Chichester, England, 2001.
- [53] Neural network examples. <https://www.edrawsoft.com/neural-network-examples.html>. Accessed: 2024-05-26.
- [54] Hongyang Xu et al. Sentiment analysis of chinese version using svm & rnn. In *Proceedings of the 6th International Conference on Information Engineering*, pages 1–5, 2017.
- [55] Nawal Soliman ALKolifi ALEnezi. A method of skin disease detection using image processing and machine learning, 2019.
- [56] Manjunatha Badiger, Varuna Kumara, Sachin CN Shetty, and Sudhir Poojary. Leaf and skin disease detection using image processing. *Global Transitions Proceedings*, 3(1):272–278, 2022.
- [57] Pravin S Ambad and AS Shirsat. A image analysis system to detect skin diseases. volume 6, pages 17–25, 2016.
- [58] Michael Abramo and Christine N. Kay. Chapter 6- image processing. In Stephen J. Ryan et al., editors, *Retina (Fifth Edition)*, pages 151–176. W.B. Saunders, London, 5th edition, 2013.
- [59] Remidio fundus , we blend technology and user centric design to craft beautiful digital solutions for your ophthalmic screening and diagnosis. <https://www.remidio.com/>. Accessed: 2024-05-26.
- [60] Jeremy West, Dan Ventura, and Sean Warnick. Spring research presentation: A theoretical foundation for inductive transfer. In *Brigham Young University, College of Physical and Mathematical Sciences*, volume 1.08, 2007.

- [61] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [62] Suvaditya Mukherjee. The annotated resnet-50. <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758>. Accessed : 07/06/2024.
- [63] <https://medium.com/nocoding-ai/densenet121-760df192f12d>. Densenet. <https://medium.com/nocoding-ai/densenet121-760df192f12d>.
- [64] NOCODING AI. Densenet. <https://medium.com/nocoding-ai/densenet121-760df192f12d>.
- [65] Petru Potrimba. What is efficientnet? <https://blog.roboflow.com/what-is-efficientnet/>. Accessed: 07-06-2024.
- [66] wisdomml. EfficientNet and its Performance Comparison with Other Transfer Learning Networks. <https://wisdomml.in/efficientnet-and-its-performance-comparison-with-other-transfer-learning-networks>
- [67] Python core team. Python. <https://www.python.org/>. Accessed: 2024-05-31.
- [68] matplotlib documentation. matplotlib documentation. https://matplotlib.org/faq/usage_faq.html#coding-styles. Accessed: 2024-05-31.
- [69] TensorFlow. Tensorflow. <https://www.tensorflow.org/>. Accessed: 2024-05-31.
- [70] IBM. Pytorch: An open source machine learning library for python, 2024. Accessed: 26-05-2024.
- [71] Keras Team. Keras. <https://keras.io/>. Accessed: 2024-05-31.
- [72] Your Machine Learning and Data Science Community. Kaggle. <https://www.kaggle.com/>. Accessed: 2024-05-31.
- [73] APTOS 2019 Blindness Detection competition. APTOS 2019 Blindness Detection competition.
- [74] Google Colab. Google colab. <https://colab.research.google.com/>. Accessed: 2024-05-31.
- [75] CUDA Explained - Why Deep Learning uses GPUs. Cuda explained - why deep learning uses gpus.

- [76] GeeksforGeeks. Cross validation in machine learning. <https://www.geeksforgeeks.org/cross-validation-machine-learning/>, 2021. Accessed: 30-05-2024.
- [77] Sam991. Diabetic retinopathy - augmented data. <https://www.kaggle.com/datasets/sam991/diabeticretinopathy-agumented-data>, 2021. Accessed: 30-05-2024.
- [78] GeeksforGeeks. Smote for imbalanced classification with python. <https://www.geeksforgeeks.org/smote-for-imbalanced-classification-with-python/>. Accessed: 30-05-2024.
- [79] Back4App Blog. React native framework. <https://blog.back4app.com/react-native-framework/>, 2024. Accessed:30-05-2024.
- [80] Expo Documentation. Get started - introduction. <https://docs.expo.dev/get-started/introduction/>, 2024. Accessed: 30-05-2024.
- [81] TensorFlow. Tensorflow lite guide. <https://www.tensorflow.org/lite/guide#:~:text=TensorFlow%20Lite%20is%20a%20set,%2C%20embedded%2C%20and%20edge%20devices.>, 2024. Accessed: 30-05-2024.