SKIN CANCER DETECTION USING DEEP LEARNING

Thesis Submitted in fulfillment of the Requirement for the degree

Of

Bachelor of Technology (B.Tech)

in

Electronics and Communication Engineering

By

Ayush Sinha (500220010025)

Arpita Saha Biswas (500220020030)

Suranjan Nandi (500220010054)

Soumyadeep Mandal (500220010052)

Rupkatha Chatterjee(500220020001)

Under the guidance of

Dr. Suparna Biswas

(Professor, ECE Department**)**

Guru Nanak Institute of Technology, Kolkata-70011

CERTIFICATE OF RECOMMENDATION

This to certify that the work in preparing the dissertation entitled "***SKIN CANCER DETECTION USING DEEP LEARNING"***, has been carried out by Ayush Sinha, Arpita Saha Biswas, Suranjan Nandi, Soumyadeep Mandal and under our supervision and may be accepted as a partial fulfillment of the requirement for the degree leading to Bachelor of Technology in Electronics and Communication Engineering Department at Guru Nanak Institute of Technology.

Dr. Suparna Biswas	**Dr. Sunipa Roy**
Professor, ECE Department	**HOD, ECE Department**
GNIT, Kolkata.	**GNIT, Kolkata**

DECLARATION

We, hereby Declare that the project work Entitled "**SKIN CANCER DETECTION USING DEEP LEARNING**" is a record of original work done by us under the guidance of Dr**. Suparna Biswas**, Department of Electronics and Communication Engineering, Guru Nanak Institute of Technology, and this project is submitted in the partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering. The results embodied in this thesis have not been submitted to any other university or institute for the award of any degree.

Team Members

Ayush Sinha (500220010025)

Arpita Saha Biswas (500220020030)

Suranjan Nandi (500220010054)

Soumyadeep Mandal (500220010052)

Rupkatha Chatterjee (500220020001)

ACKNOWLEDGMENT

We would like to express our special thanks of gratitude to our Project guide Dr. Suparna Biswas for their able guidance and support in completing our Project.

We would also like to extend our gratitude to our HOD Madam Dr. Sunipa Roy for providing us with all the facility that was required.

TABLE OF CONTENT

ABSTRACT

Now a day's skin cancer is a major problem human beings are facing, To recognize skin cancer new methodology for the diagnosing skin cancer by images of dermatologic spots using image processing presented. Currently in skin cancer one the most frequent diseases humans. This methodology based Fourier spectral analysis using filters such classic, inverse and to k-law nonlinear. The sample images are obtained by a specialist as an replacement spectral to technique is developed and quantitative measurement in the complex pattern found cancerous skin spots. Finally in which spectral index calculated get a variety spectral indices defined carcinoma. Our results show confidence of level in 95.4%. Carcinoma mainly occurs thanks to exposure of sunlight. Ozone is depletion and maintained chemical exposures in other factors involved precipitating carcinoma. Mutations of p53 gene involved UV- induced as carcinogenesis. P53 gene acts vital development in SCC.

Skin Cancer alarming is disease for mankind , the need early diagnosis the skin cancer are increased due to the rapid climb rate of Melanoma skin cancer , its high treatment Costs , and death rate. The cancer cells are detected manually and it takes time to cure in most of the cases. This project proposed a man-made carcinoma detection system using image Processing and machine learning method. The features of the affected skin cells are extracted after the segmentation of the pictures using feature extraction technique. A deep learning based method Convolutional neural network classifier is employed for the stratification of the extracted features..

Skin Cancer is an alerting issue and it must be detected as early as possible. The diagnostic is a manual process that is time consuming as well as expensive. But, today's world science has become advanced by using machine learning make easy detecting cancerous cells to the machine learning specially convolution neural network is employed to detect cancerous cell more in quickly , and to efficiently

INTRODUCTION

Cancer forms when healthy cells in change in and grow out control, forming a called the tumor. A tumor can cancerous r benign. A cancerous tumor is malignant, meaning that grow and spread over other parts of the body. As there begun as a tumor means that tumor can be grow but won't spread. Doctors diagnose carcinoma additional than 3 million Americans annually, making in foremost common sort of cancer. If carcinoma is found early, it can usually be treated with topical medications, procedures wiped out office a dermatologist, or outpatient surgery. A dermatologist may doctor who focuses diseases and conditions of the skin. As an result, carcinoma is liable for but 1% all cancer deaths. In some cases, carcinoma could also more advanced in need management to a multidisciplinary team to always a dermatologist, surgical and oncologist, radiation oncologist, and to a medical oncologist. These are in doctors meet there patient, and together they're going recommend the simplest path forward treat cancer. In such instances, the surgical oncologist will recommend surgery be performed operating room because the procedure treat the cancer too extensive for an office setting.

There are three main types of skin cancer:

- basal cell carcinoma (BCC)
- squamous cell carcinoma (SCC)
- melanoma - the most dangerous form of skin cancer.

Both basal cell carcinoma and squamous cell carcinoma are known as non-melanoma skin cancer or keratinocyte cancers. Keratinocyte cancer is more common in men, with almost double the incidence compared to women.

Melanoma is the third most common cancer in Australians excluding keratinocyte cancers as data on incidence is not routinely collected).

There are also rare types of skin cancer including Merkel cell carcinoma and angiosarcoma. These are treated differently from BCC and SCC.

Every year, in Australia:

- skin cancers account for around 80% of all newly diagnosed cancers
- the majority of skin cancers are caused by exposure to the sun
- the incidence of skin cancer is one of the highest in the world, two to three times the rates in Canada, the US and the UK.

SKIN CANCER SIGNS AND SYMPTOMS

The sooner a skin cancer is identified and treated, the better your chance of avoiding surgery or, in the case of a serious melanoma or other skin cancer, potential disfigurement or even death.

It is also a good idea to talk to your doctor about your level of risk and for advice on early detection.

Become familiar with the look of your skin, particularly spots and moles, so you pick up any changes that might suggest a skin cancer.
Look for:

- any crusty, non-healing sores
- small lumps that are red, pale or pearly in colour
- new spots, freckles or any moles changing in colour, thickness or shape over a period of weeks to months.

CAUSES OF SKIN CANCER

Australia has one of the highest rates of skin cancer in the world. Anyone can be at risk of developing skin cancer, though the risk increases as you get older.

The majority of skin cancers in Australia are caused by exposure to UV radiation in sunlight.

Some factors that increase your risk of skin cancer include:

- sunburn
- tanning
- solariums.

METHODOLOGY

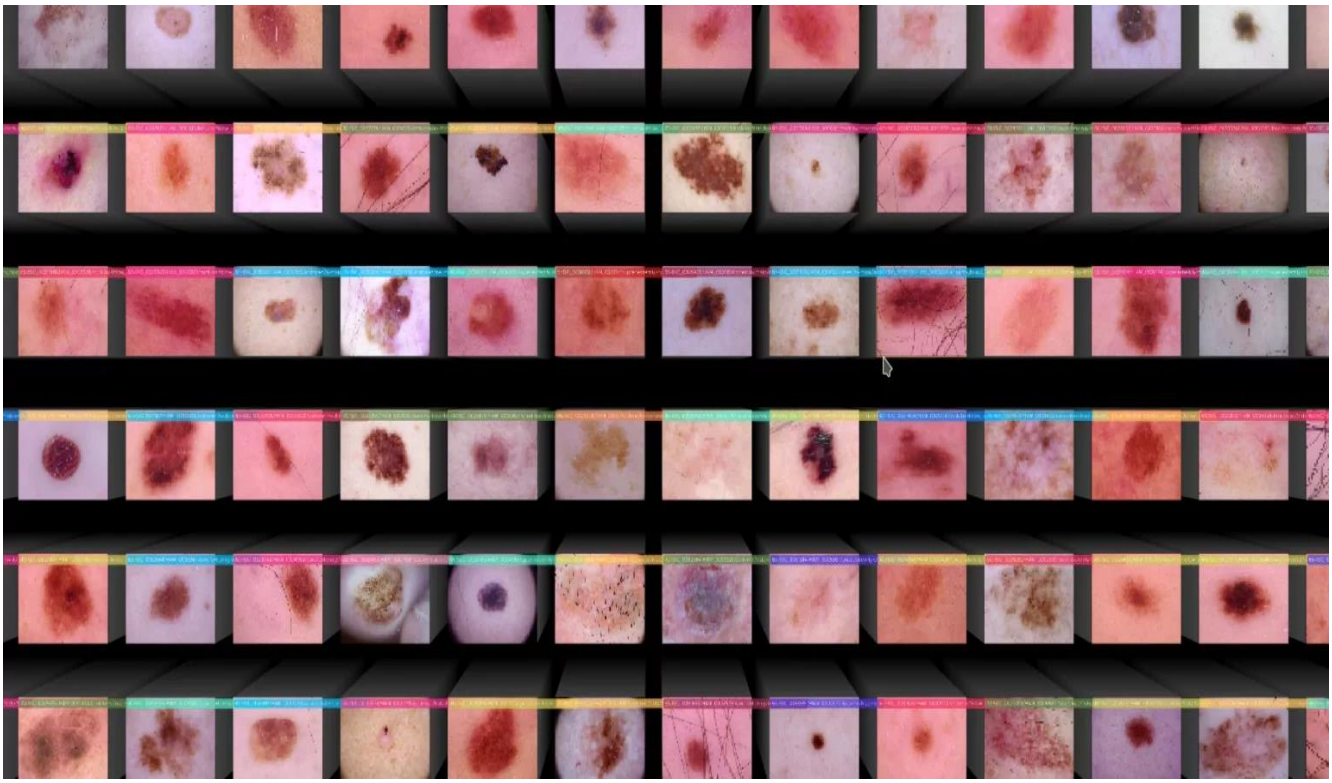
DATA COLLECTION:

Dataset used for this are extracted from kaggle towards skin cancer Detection.

It consists of 10000+ images of skin cancer.

The training data consists of 8000 images and testing data consists of 2000 images

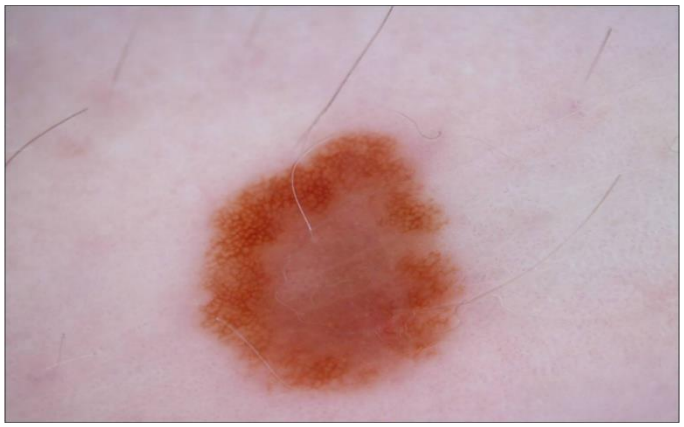
NOTE - while these datasets are publicly available, they should be used in accordance with their respective terms and conditions. Always respect patient privacy and data usage policies when working with medical datasets. Also, remember to cite the datasets appropriately in any resulting publications or reports.



In HAM10000 dataset there is 7 classes of skin cancer :

Melanocytic nevi(nv) –

Cause:
All moles, including those that are congenital, are made of "melanocytes." These are the cells that give skin its color (pigment). These cells are present in all skin types and colors, in varying degrees. In congenital melanocytic nevi, there are more of these cells, which makes that skin a darker color. It is not known what causes these to form, but a genetic cause is suspected.



Symptoms:
There are usually no symptoms with congenital melanocytic nevi. Sometimes the larger ones may itch.If pain, severe or persistent itching, bleeding, or crusting develop, see your child's doctor.

Treatment:
The majority of congenital melanocytic nevi do not need treatment.

Check the mole(s) each month. Watch for any changes in the way the mole(s) look. It may help to take a photo of the mole(s) with your smartphone or digital camera so you can tell if there have been any changes.

If there are any changes, such as areas of bleeding or crusting, new bumpy areas, areas that change color, new pain or itch, change in shape or rapid change in size, have your child see your doctor. Your doctor may suggest that a sampling or skin biopsy be done. This is usually done in an office and takes no more than a few minutes. For some congenital nevi, removal may be recommended. This may require general anesthesia in younger children or in any child with fear or anxiety about the procedure.

There is the small risk of melanoma developing with a congenital melanocytic nevus, so removing the mole may be an option. This may be also recommended if the nevus is:

- In an area that will be difficult to watch, such as the scalp or buttocks
- In a child with extreme anxiety or concern about the appearance of the congenital melanocytic nevus
- Very unusual in appearance

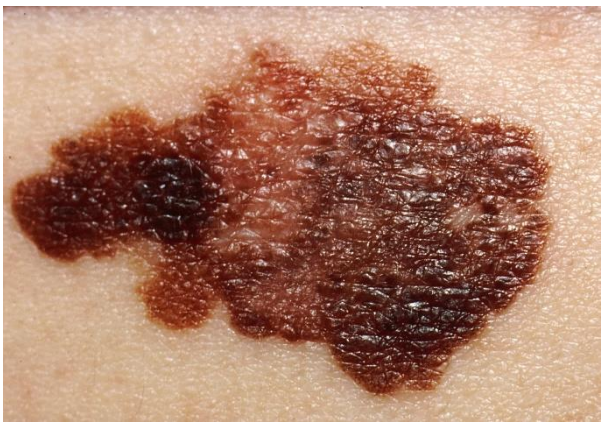
Removing a mole requires surgery. Although removing the mole will greatly reduce the chance of melanoma, it will leave a scar. This is to be considered when deciding whether or not to remove it. There are no lasers or other treatments that can be safely used to remove melanocytic nevi

Melanoma(mel) -

Melanoma is a type of skin cancer that develops in the skin cells called melanocytes and usually occurs on the parts of the body that have been overexposed to the sun. Rare melanomas can occur inside the eye (ocular melanoma) or in parts of the skin or body that have never been exposed to the sun such as the palms of the hands, the soles of the feet or under the nails.

It is estimated that more than 18,200 people were diagnosed with melanoma in 2023. The average age at diagnosis is 65 years old.

Melanoma is the third most commonly diagnosed cancer in Australia, and it is estimated that one in 17 people will be diagnosed by the time they are 85.Australia, along with New Zealand have the world's highest incidence rate of melanoma.



Melanoma signs and symptoms:

Often melanoma has no symptoms, however, the first sign is generally a change in an existing mole or the appearance of a new spot. These changes can include:

- colour - a mole may change in colour, have different colour shades or become blotchy
- size - a mole may appear to get bigger
- shape - a mole may have an irregular shape, may increase in height or not be symmetrical
- elevation - the mole may develop a raised area
- itching or bleeding.

Other symptoms include dark areas under nails or on membranes lining the mouth, vagina or anus.

New moles and spots will appear and change during childhood, adolescence and during pregnancy and this is normal. However, adults who develop new spots or moles should have them examined by their doctor.

Causes of melanoma:

Melanoma risk increases with exposure to UV radiation from the sun or other sources such as solariums, particularly with episodes of sunburn (especially during childhood).

Melanoma risk is increased for people who have:

- unprotected UV radiation exposure
- a history of childhood tanning and sunburn
- a pattern of short, intense periods of exposure to UV radiation
- having a lot of moles (naevi) – more than 50 on the body and more than 10 above the elbows on the arms
- increased numbers of unusual moles (dysplastic naevi)
- depressed immune systems
- a family history of melanoma in a first degree relative

The best way to prevent Melanoma is to protect your skin from the sun and other sources of ultraviolet (UV) rays. To protect yourself from skin cancer: Stay out of the sun as much as possible between 10 a.m. and 4 p.m. Cover up

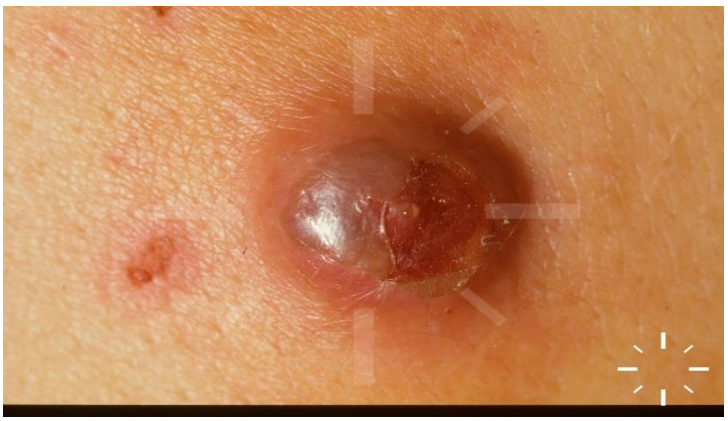
with long sleeves, long pants or a long skirt, a hat, and sunglasses

Benign Keratosis –like lesions (bkl)

The important causes of benign keratosis are as:-

Sun exposure, is the main cause of seborrheic keratosis.
They are not contagious i.e. it is not spread from one person to another.
They are usually painless and benign, but may become irritated and itch. They may be cosmetically disfiguring and psychologically distressing.

Seborrheic keratosis primarily affect people older than 30. Some women notice that they develop them during pregnancy or after taking estrogen. They are increasingly common in the later decades of life. Children seldom develop these skin growths.



Symptoms & Signs:
The commonly occur symptoms of seborrheic keratosis are as:-

They vary in color from white to light tan to black.
The size of seborrheic keratosis, changes from tiny to larger than 1 in. (2.5 cm) in diameter.
Seborrheic keratosis, can an be dome-shaped with tiny white or black "horns" growing from the surface.
Seborrheic keratosis are located anywhere on body, but usually appear on the chest, torso, face, shoulders, or back.
They may have single or multiple growths.
Precautions

Benign Keratosis-like lesions (BKL), also known as Lichenoid Keratosis, are small, inflamed macules or thin pigmented plaques, usually solitary[1]. They are generally harmless but can cause discomfort in some cases[3]. Here are some precautions you can take:

1. Sun Protection: Apply sunscreen lotion whenever exposed to sunlight and try to stay out of the sun during hot days. This is because sun exposure can trigger the inflammatory reaction that leads to these lesions.

2. Clothing: Wear protective clothes to prevent direct exposure from the sun Friction can also trigger the inflammatory reaction, so avoid clothing that rubs against your skin[1].

3. Monitor Changes: Keep an eye on any changes in the size, color, or sensation of the lesions. If the lesion becomes itchy or has a mild stinging sensation, it might be a good idea to consult a dermatologist[1].

4. Oral Health: If you have sores in your mouth, they can make eating and drinking painful, as well as increasing your risk for oral cancer[3]. Maintain good oral hygiene and consult a healthcare professional if you notice any changes.

Basal cell carcinoma (bcc)-

Basal cell carcinoma is a type of skin cancer. Basal cell carcinoma begins in the basal cells — a type of cell within the skin that produces new skin cells as old ones die off.

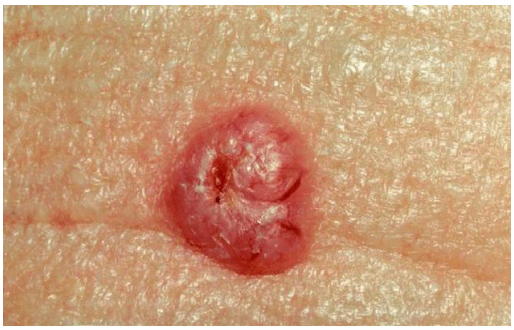
Basal cell carcinoma often appears as a slightly transparent bump on the skin, though it can take other forms. Basal cell carcinoma occurs most often on areas of the skin that are exposed to the sun, such as your head and neck.

Most basal cell carcinomas are thought to be caused by long-term exposure to ultraviolet (UV) radiation from sunlight. Avoiding the sun and using sunscreen may help protect against basal cell carcinoma.

Basal cell carcinoma usually develops on sun-exposed parts of your body, especially your head and neck. Less often, basal cell carcinoma can develop on parts of your body usually protected from the sun, such as the genitals.

Basal cell carcinoma appears as a change in the skin, such as a growth or a sore that won't heal. These changes in the skin (lesions) usually have one of the following characteristics:
A shiny, skin-colored bump that's translucent, meaning you can see a bit

through the surface. The bump can look pearly white or pink on white skin. On brown and Black skin, the bump often looks brown or glossy black. Tiny blood vessels might be visible, though they may be difficult to see on brown and Black skin. The bump may bleed and scab over.
A brown, black or blue lesion — or a lesion with dark spots — with a slightly raised, translucent border.
A flat, scaly patch with a raised edge. Over time, these patches can grow quite large.
A white, waxy, scar-like lesion without a clearly defined border.



Causes:

Factors that increase your risk of basal cell carcinoma include:

Chronic sun exposure. A lot of time spent in the sun — or in commercial tanning beds — increases the risk of basal cell carcinoma. The threat is greater if you live in a sunny or high-altitude location, both of which expose you to more ultraviolet (UV) radiation. Severe sunburns also increase your risk.
Radiation therapy. Radiation therapy to treat acne or other skin conditions may increase the risk of basal cell carcinoma at previous treatment sites on the skin.
Fair skin. The risk of basal cell carcinoma is higher among people who freckle or burn easily or who have very light skin, red or blond hair, or light-colored eyes.
Increasing age. Because basal cell carcinoma often takes decades to develop, the majority of basal cell carcinomas occur in older adults. But it can also affect younger adults and is becoming more common in people in their 20s and 30s.
A personal or family history of skin cancer. If you've had basal cell carcinoma one or more times, you have a good chance of developing it again. If you have a family history of skin cancer, you may have an increased risk of developing basal cell carcinoma.
Immune-suppressing drugs. Taking medications that suppress your immune system, such as anti-rejection drugs used after transplant surgery, significantly increases your risk of skin cancer.

Exposure to arsenic. Arsenic, a toxic metal that's found widely in the environment, increases the risk of basal cell carcinoma and other cancers. Everyone has some arsenic exposure because it occurs naturally. But some people may have higher exposure if they drink contaminated well water or have a job that involves producing or using arsenic.

Prevention:

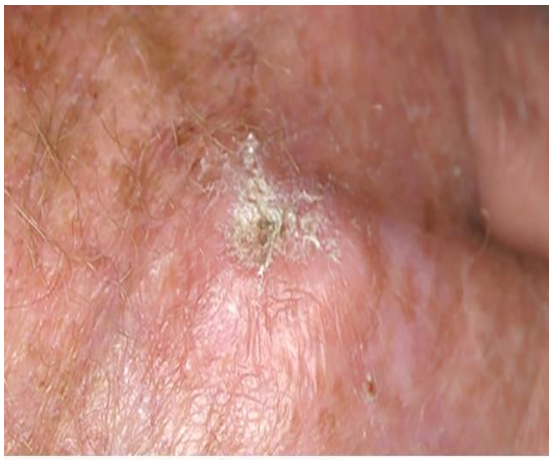
To reduce your risk of basal cell carcinoma you can:

Avoid the sun during the middle of the day. In many places, the sun's rays are strongest between about 10 a.m. and 4 p.m. Schedule outdoor activities for other times of the day, even during winter or when the sky is cloudy.
Wear sunscreen year-round. Use a broad-spectrum sunscreen with an SPF of at least 30, even on cloudy days. Apply sunscreen generously, and reapply every two hours — or more often if you're swimming or perspiring.
Wear protective clothing. Cover your skin with dark, tightly woven clothing that covers your arms and legs, and a broad-brimmed hat, which provides more protection than does a baseball cap or visor.

Some companies also sell protective clothing. A dermatologist can recommend an appropriate brand. Don't forget sunglasses. Look for those that block both types of UV radiation — ultraviolet A (UVA) and ultraviolet B (UVB) rays.

Actinic keratosis (akiec)-

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



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

protecting your skin from ultraviolet (UV) rays.

Left untreated, the risk of actinic keratosis turning into a type of skin cancer called squamous cell carcinoma is about 5% to 10%.

Symptoms:

Actinic keratosis vary in appearance. Symptoms include:

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

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

Vascular lesions (vas)-

Vascular lesions are a type of skin condition that originates from or affects blood or lymphatic vessels. They can include both malignant (cancerous) and benign (non-cancerous) tumors, malformations, and inflammatory diseases. One type of skin cancer that forms in the lining of blood vessels and lymph vessels is called Angiosarcoma. It often appears as a bruise-like lesion that grows over time.

Symptoms:

The symptoms of Angiosarcoma may vary based on where the cancer occurs. When it affects the skin, especially on the head and neck, symptoms can include
- A raised area of skin that looks like a bruise
- A bruise-like lesion that grows larger over time
- A lesion that may bleed when scratched or bumped
- Swelling in the skin around the lesion

When Angiosarcoma affects organs, such as the liver or the heart, it often

causes pain

Precautions:

Precautions mainly involve early detection and prevention. Here are some general guidelines
Reduce Sun Exposure: Protect your skin when the UV index is 3 or higher. Stay in the shade, wear clothing that covers your arms and legs, wear a hat with a wide brim to shade your face, head, ears, and neck, and wear sunglasses that wrap around and block both UVA and UVB rays.
Regular Self-Examinations: Conduct regular self-examinations for new moles or changes in existing ones.
Clinical Monitoring: Close clinical monitoring and biopsy of highly atypical or changed lesions is recommended.

Please consult with a healthcare provider if you have any persistent symptoms that worry you. This information is intended for general knowledge and is not a substitute for professional medical advice or treatment

Dermatofibromas (df)-

Dermatofibromas are harmless growths within the skin that usually have a small diameter. They can vary in color but are typically pink to light brown in light skin and dark brown or black in dark skin. They may appear more pink or darker if a person accidentally irritates them — for example, when shaving.

As they are dense and firm to the touch, many people say that they feel like a small stone underneath or raised above the skin. Most dermatofibromas are painless, but some people experience itching, irritation, or tenderness at the site of the growth. Some doctors or medical researchers may refer Trusted Source to dermatofibromas as benign fibrous histiocytomas.

A dermatofibroma is a nodule made of fibrous tissue. When a doctor squeezes the nodule during an examination, the overlying skin dimples.

Causes and risk factors

Dermatofibromas are an accumulation of extra cells within the deeper layers of the skin. Medical researchers do not know the exact cause of these growths.

Some researchers Trusted Source theorize that a possible cause is an adverse reaction to a local trauma, such as a small injury or bug bite in the area where the lesion later forms.

Age may be another risk factor, as the growths appear mostly in adults. People with a suppressed immune system may also be more likely to experience dermatofibromas and to have more than one growth.

Multiple dermatofibromas are also more common in people with underlying conditions, especially in those with systemic lupus erythematosus. In some cases, there may not be an obvious cause.

Symptoms and complications:

Dermatofibromas tend to grow slowly. The growths typically have some defining characteristics that can aid their identification.

Key markers of a dermatofibroma are:

- **Appearance:** A dermatofibroma presents as a round bump that is mostly under the skin.
- **Size:** The normal range is about 0.5–1.5 centimeters (cm), with most lesions being 0.7–1.0 cm in diameter. The size will usually remain stable.
- **Color:** The growths vary in color among individuals but will generally be pink, red, gray, brown, or black.
- **Location:** Dermatofibromas are most common on the legs, but they sometimes appear on the arms, trunk, and, less commonly, elsewhere on the body.

- **Additional symptoms:** Although they are usually harmless and painless, these growths may occasionally be itchy, tender, painful, or inflamed.

If a person pinches a dermatofibroma, it will not push toward the surface of the skin. Instead, it will dimple inward on itself. This characteristic can help people distinguish between a dermatofibroma and another type of growth.

It is common for only one growth to appear on the body, but multiple dermatofibromas may occur in people with underlying health conditions or a weakened immune system.

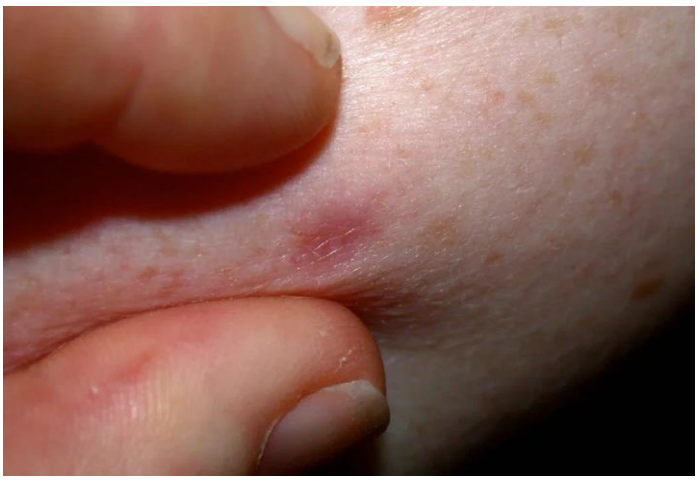
Skin growths can be alarming, but dermatofibromas are generally harmless.

However, if a person has a growth that looks like a dermatofibroma but is rapidly growing or changing, they should seek medical advice. This growth may be a sign of a rare type of cancer called dermatofibrosarcoma protuberans.

Diagnosis:

Primary care doctors and dermatologists will usually diagnose a dermatofibroma by inspecting it visually. The papules are easy to identify, but doctors will also want to be certain that they do not misdiagnose the growth.

In addition to asking a person questions about their symptoms and examining the area, a doctor is likely to perform the following Trusted Source:



- **Pinch test:** The doctor may pinch the surrounding skin to check for the characteristic dimple.
- **Dermatoscope:** The doctor may use this device to take a magnified look at the surface of the growth. Dermatofibromas will usually have a central white area in the middle with a pigmented area surrounding it.
- **Biopsy:** If the growth is bleeding, abnormally shaped, or irritated, or it has a sore on top of it, doctors may want to do a biopsy. This procedure involves taking a small bit of the tissue from the papule to examine under a microscope in a laboratory.

OVERVIEW OF THE WHOLE METHOD

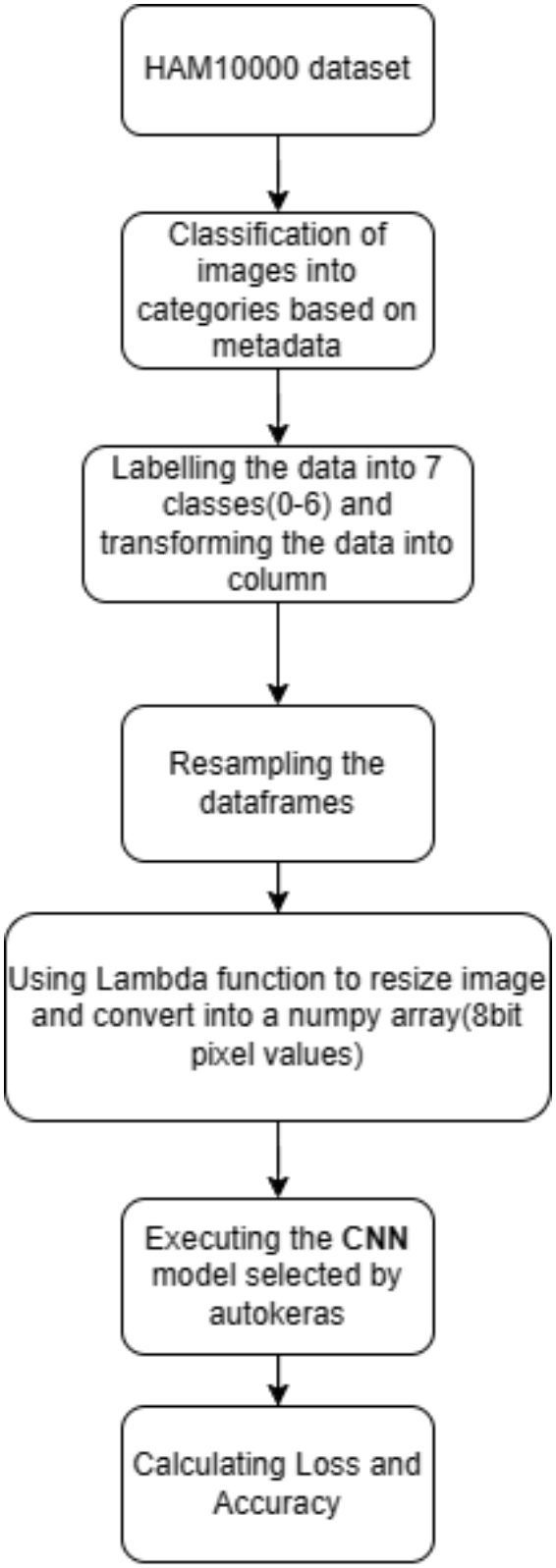


IMAGE SORTING AND AUTOKERAS

Dataset used for skin cancer detection named as HAM10000 consists of 10015 images in two subfolders diving 5000 and 5015 images in each folders. To ensure that execution does not take much ram memory and disk space, we found it easy to put all the images into a single folder named "**all images**" and created another empty folder named "**reorganised**". Initially, imported necessary libraries such as Pandas for data manipulation, OS for handling file paths, and shutil for file operations. The two folders "**all images**" and "**reorganized**" defines source and destination directories where the images are stored and where they will be organized, respectively. The line of code mentioned below in the picture reads a "**Metadata.CSV**" file containing image names and labels into a **Pandas DataFrame**, then prints the count of unique label values. It extracts unique labels from the **DataFrame** and iterates through each label, creating a corresponding subfolder in the destination directory. For each label, it filters the DataFrame to get the "**image IDs"** associated with that label and copies the corresponding image files from the source directory to the appropriate subfolder within the destination directory.

```
import pandas as pd # type: ignore
import os
import shutil

# Dump all images into a folder and specify the path:
data_dir = os.path.join(os.getcwd(), "/content/drive/MyDrive/final_year_project/all images")

# Path to destination directory where we want subfolders
dest_dir = os.path.join(os.getcwd(), "/content/drive/MyDrive/final_year_project/reorganised")

# Read the csv file containing image names and corresponding labels
skin = pd.read_csv('/content/drive/MyDrive/final_year_project/HAM10000/HAM10000_metadata.csv')
print(skin['dx'].value_counts())

labels = skin['dx'].unique().tolist()  # Extract labels into a list

# Copy images to new folders
for label in labels:
    os.mkdir(os.path.join(dest_dir, str(label)))
    label_images = skin[skin['dx'] == label]['image_id']
    for image_id in label_images:
        src_path = os.path.join(data_dir, f"{image_id}.jpg")
        dest_path = os.path.join(dest_dir, str(label), f"{image_id}.jpg")
        shutil.copyfile(src_path, dest_path)
```

AUTOKERAS

AutoKeras is an **automated machine learning (AutoML)** library that simplifies the model-building process. AutoKeras can be used to create an image classification model for skin cancer lesions.It leverages **convolutional neural networks (CNNs)**, which are well-suited for image analysis.By training on a dataset of skin lesion images, the model learns to differentiate between different types of skin cancer

By using Autokeras, we were able to find the right convolution layers for our problem. We loaded the data and tried to balance the data by splitting the data into very small dataset (100 images in each of the 7 classes) and ran the function "**ImageClassifier"** with a max trial of 25 and training data with 25 epochs each. That means, auto keras will process the data 25times each and with 25 different models to give us the right convolution layer model.

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 30, 30, 256)	7168
max_pooling2d_3 (MaxPooling2D)	(None, 15, 15, 256)	0
dropout_3 (Dropout)	(None, 15, 15, 256)	0
conv2d_4 (Conv2D)	(None, 13, 13, 128)	295040
max_pooling2d_4 (MaxPooling2D)	(None, 6, 6, 128)	0
dropout_4 (Dropout)	(None, 6, 6, 128)	0
conv2d_5 (Conv2D)	(None, 4, 4, 64)	73792
max_pooling2d_5 (MaxPooling2D)	(None, 2, 2, 64)	0
dropout_5 (Dropout)	(None, 2, 2, 64)	0
flatten_1 (Flatten)	(None, 256)	0
dense_2 (Dense)	(None, 32)	8224
dense_3 (Dense)	(None, 7)	231

Convolution model provided by Auto Keras

LIBRARIES USED

- **Matplotlib**: A comprehensive library for creating static, animated, and interactive visualizations in Python.
- **NumPy**: Fundamental package for scientific computing with Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions.
- **Pandas**: A powerful data analysis and manipulation library. It offers data structures like DataFrame and Series, along with tools for reading and writing data between in-memory data structures and different file formats.
- **os**: A module providing a portable way to interact with the operating system. It allows you to perform operations like navigating directories, manipulating file paths, etc.
- **glob**: A module used to retrieve files/pathnames matching a specified pattern.
- **Seaborn**: Built on top of Matplotlib, Seaborn provides a high-level interface for drawing attractive and informative statistical graphics.

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import os
from glob import glob
import seaborn as sns
from PIL import Image

np.random.seed(42)
from sklearn.metrics import confusion_matrix

import tensorflow as tf
import keras
from keras.utils import to_categorical # used for converting labels to one-hot-encoding
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten, Conv2D, MaxPool2D, BatchNormalization
from sklearn.model_selection import train_test_split
from scipy import stats
from sklearn.preprocessing import LabelEncoder
```

- **PIL (Python Imaging Library)**: A library for adding image processing capabilities to your Python interpreter.
- **Scikit-learn**: A simple and efficient tool for data mining and data analysis. It features various classification, regression, and clustering algorithms, as well as tools for model selection and evaluation.

- **TensorFlow**: An open-source machine learning framework developed by Google for building and training machine learning models
- **Keras**: An open-source neural network library written in Python. It provides a high-level interface for building and training deep learning models.

These libraries collectively provide a comprehensive set of tools for data manipulation, visualization, and machine learning tasks.

Lines like import tensorflow , keras for high level neural network workflows. keras.utils import to_categorical: This is used for converting categorical data variables so they can be provided to machine learning algorithms to improve predictions. SciPy library, which contains a large number of probability distributions as well as a growing library of statistical functions. sklearn.preprocessing import LabelEncoder: This is a utility class to help normalize labels such that they contain only values between 0 and n_classes-1.

CONVERTING LABELS

- Images are being further classified into further classes based on age, gender, localization, etc. This categorization depends on the metadata.csv provided by the HAM10000 dataset.
- Based on classification of the image dataset graphs are to be created for better understanding.

```
skin_df = pd.read_csv('/content/drive/MyDrive/final_year_project/HAM10000/HAM10000_metadata.csv')

SIZE=32

# label encoding to numeric values from text
le = LabelEncoder()
le.fit(skin_df['dx'])
LabelEncoder()
print(list(le.classes_))

skin_df['label'] = le.transform(skin_df["dx"])
print(skin_df.sample(10))
```

```
# Data distribution visualization
fig = plt.figure(figsize=(12,8))

ax1 = fig.add_subplot(221)
skin_df['dx'].value_counts().plot(kind='bar', ax=ax1)
ax1.set_ylabel('Count')
ax1.set_title('Cell Type');

ax2 = fig.add_subplot(222)
skin_df['sex'].value_counts().plot(kind='bar', ax=ax2)
ax2.set_ylabel('Count', size=15)
ax2.set_title('Sex');

ax3 = fig.add_subplot(223)
skin_df['localization'].value_counts().plot(kind='bar')
ax3.set_ylabel('Count',size=12)
ax3.set_title('Localization')

ax4 = fig.add_subplot(224)
sample_age = skin_df[pd.notnull(skin_df['age'])]
sns.distplot(sample_age['age'], fit=stats.norm, color='red');
ax4.set_title('Age')

plt.tight_layout()
plt.show()
```

‘df_x’ , where x represents labels(e.g; 0,1, …, 6) as a data frame takes input from different labels. For example df_0 looks for all the rows where the value is equal to 0 and captures it to a new data frame.

```
df_0 = skin_df[skin_df['label'] == 0]
df_1 = skin_df[skin_df['label'] == 1]
df_2 = skin_df[skin_df['label'] == 2]
df_3 = skin_df[skin_df['label'] == 3]
df_4 = skin_df[skin_df['label'] == 4]
df_5 = skin_df[skin_df['label'] == 5]
df_6 = skin_df[skin_df['label'] == 6]
```

RESAMPLING

We took a number of samples as 500 for convinience. These lines downscales sample size to 500 images where input size is greater than 500 by selecting 500 images randomly and eliminating other images. Upscales to 500 images where input size is less than 500 by randomly duplicating images.

Using the pd.concat() function we are concatenating all the newly created data frames into a single data frame called ‘skin_df_balanced’.
‘Skin_df_balanced’ this data frame consists of 3500 images (7 times 500).

Now we have a complete balanced data set.

```
n_samples=500
df_0_balanced = resample(df_0, replace=True, n_samples=n_samples, random_state=42)
df_1_balanced = resample(df_1, replace=True, n_samples=n_samples, random_state=42)
df_2_balanced = resample(df_2, replace=True, n_samples=n_samples, random_state=42)
df_3_balanced = resample(df_3, replace=True, n_samples=n_samples, random_state=42)
df_4_balanced = resample(df_4, replace=True, n_samples=n_samples, random_state=42)
df_5_balanced = resample(df_5, replace=True, n_samples=n_samples, random_state=42)
df_6_balanced = resample(df_6, replace=True, n_samples=n_samples, random_state=42)

#Combined back to a single dataframe
skin_df_balanced = pd.concat([df_0_balanced, df_1_balanced,
                              df_2_balanced, df_3_balanced,
                              df_4_balanced, df_5_balanced, df_6_balanced])

#Check the distribution. All classes should be balanced now.
print(skin_df_balanced['label'].value_counts())
```

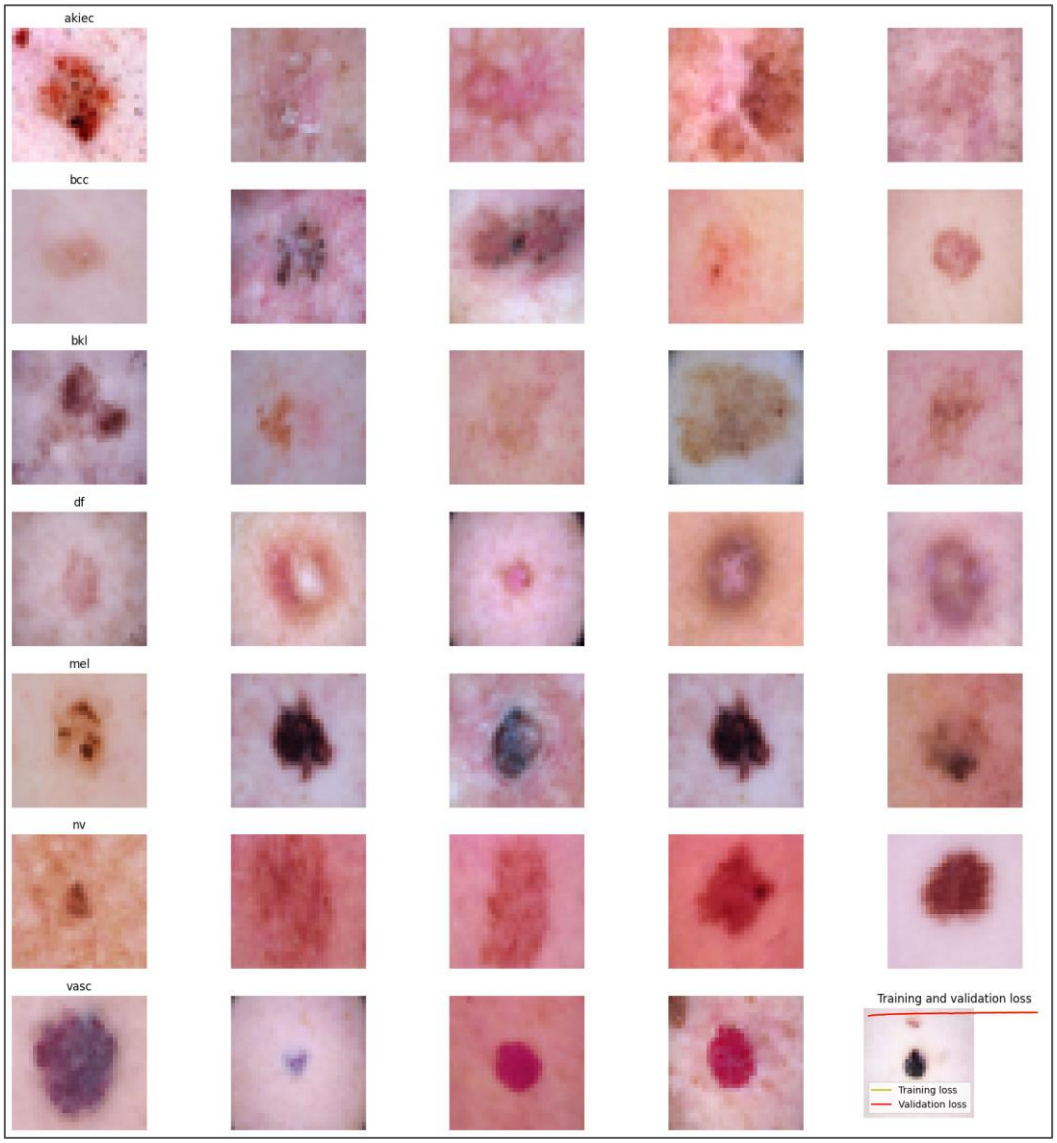
We are using the lambda function to convert and resize the image and transform the value into a numpy array of 8bit pixel values (0-255). After that the new path of the image is being added to the new column in the data frame. After that we are iterating for each group, it samples random rows from the numpy array and dx label, ensuring consistency in sampling by setting a random seed. Then, for each sampled row, it accesses the image data, normalizes it to range [0, 1] and displays it using a function to turn off the axis for each subplot, making the plot cleaner.

```
#reading images based on image ID from the CSV file
#This is the safest way to read images as it ensures the right image is read for the right ID
image_path = {os.path.splitext(os.path.basename(x))[0]: x
                    for x in glob(os.path.join('/content/drive/MyDrive/final_year_project/reorganised', '*', '*.jpg'))}

#Define the path and add as a new column
skin_df_balanced['path'] = skin_df['image_id'].map(image_path.get)
#Use the path to read images.
skin_df_balanced['image'] = skin_df_balanced['path'].map(lambda x: np.asarray(Image.open(x).resize((SIZE,SIZE))) if x is not None else None)


n_samples = 5  # number of samples for plotting
# Plotting
fig, m_axs = plt.subplots(7, n_samples, figsize = (4*n_samples, 3*7))
for n_axs, (type_name, type_rows) in zip(m_axs,
                                         skin_df_balanced.sort_values(['dx']).groupby('dx')):
    n_axs[0].set_title(type_name)
    for c_ax, (_, c_row) in zip(n_axs, type_rows.sample(n_samples, random_state=1234).iterrows()):
        if c_row['image'] is not None:
            c_ax.imshow(c_row['image'].astype(np.float32) / 255.)
            c_ax.axis('off')
```

After the dataframes are being successfully plotted, all the images are being shown with 7 different classes.



DATAFRAMES

The images are being assigned to "**X**" and the label is assigned to "**Y**". As in every machine learning model the data on **X** is being trained to predict the data on **Y**. It handles cases where image data is missing (unsigned integer) by replacing them with zeros. Then, it scales the pixel values to the range [0, 1] by dividing by 255, i.e. converting values into float. Then It converts the class labels **Y** into categorical format using one-hot encoding since it's a multiclass classification problem with 7 classes. After that we split the data into training and testing sets with 25% of the data reserved for testing and the rest for train ing. The random_state parameter ensures reproducibility of the split.

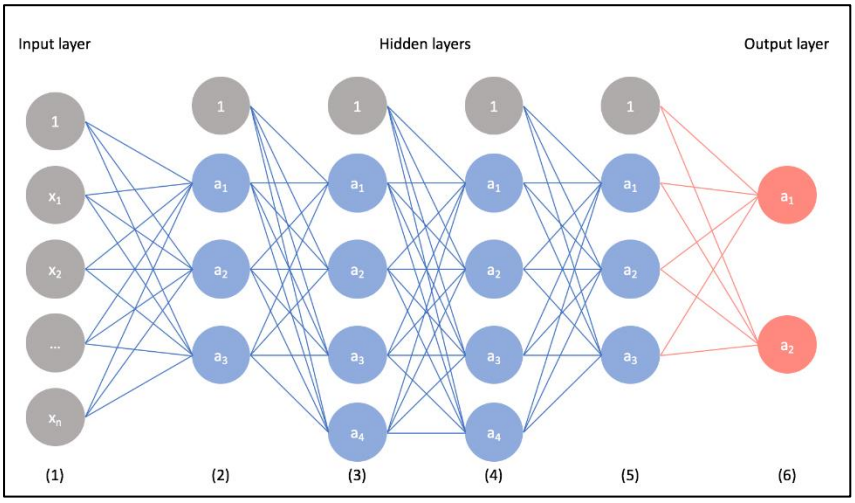
```
#Convert dataframe column of images into numpy array
X = np.asarray(skin_df_balanced['image'].tolist())
X[X == None] = 0
X = X/255.  # Scale values to 0-1.
Y=skin_df_balanced['label']  #Assign label values to Y
Y_cat = to_categorical(Y, num_classes=7) #Convert to categorical as this is a multiclass classification problem
#Split to training and testing
x_train, x_test, y_train, y_test = train_test_split(X, Y_cat, test_size=0.25, random_state=42)
```

CONVOLUTION NEURAL NETWORK

Convolutional Neural Networks (CNNs) are a fundamental deep learning architecture for image classification tasks. They excel at capturing hierarchical features in images, making them well-suited for tasks like distinguishing between different types of skin diseases based on photographic images. CNNs are a class of deep learning models designed to process data with grid-like topology, such as images. CNNs are inspired by the biological visual cortex and have been successful in image recognition tasks.

A CNN architecture is formed by a stack of distinct layers that transform the input volume into an output volume. The transformation is performed via learnable parameters and activation functions. The layers of a CNN typically consist of convolutional layers, pooling layers, and fully connected layers. Convolutional layers apply a convolution operation to the input, passing the result to the next layer. This operation allows the network to be deeper with fewer parameters. **Pooling layers** reduce the spatial dimensions (width and height) of the input volume. They help decrease the computational complexity and control overfitting. **Fully connected layers** connect every neuron in one layer to every neuron in another layer. It is the same as the traditional **multi-layer perceptron neural network (MLP)**.The final layer of a CNN is often a **softmax** function that provides probabilities for the input belonging to each class.

CNNs automatically learn and improve from experience without being explicitly programmed, making them a key algorithm in machine learning. They are widely used in applications like image and video recognition, recommender systems, and natural language processing.



Our model architecture uses convolutional neural network (CNN) designed for image classification preffered by auto keras. There are many components used by our CNN model, they are stated below as:

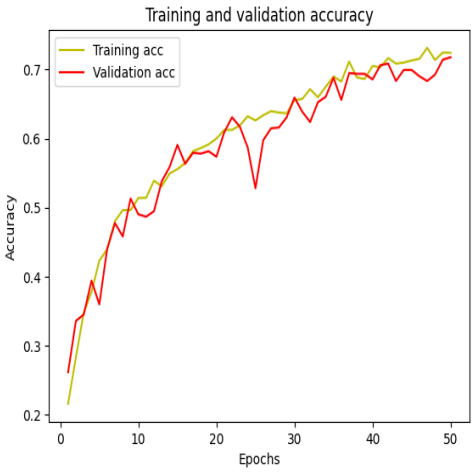
1. **`Conv2D`**: This layer performs convolutional operations on the input image. It applies 256 filters of size **3x3** to the input image, producing feature maps. **Rectified Linear Unit(ReLU)** activation function is used to introduce non-linearity.
2. **`MaxPool2D`**: This layer performs max pooling operation, which reduces the spatial dimensions of the feature maps by taking the maximum value within each window. This helps in reducing computation and controlling overfitting.
3. **`Dropout`**: Dropout is a regularization technique used to prevent overfitting by randomly setting a fraction of input units to 0 at each update during training time. Here, a dropout rate of 0.3 is applied. This pattern repeats with smaller numbers of filters (128 and 64) in subsequent convolutional layers followed by max pooling and dropout.
4. **`Flatten`**: This layer flattens the output from the previous layer into a one-dimensional vector, which can be fed into a densely connected layer.
5. **`Dense`**: These are fully connected layers. The first dense layer has 32 units, followed by the output layer with 7 units (since we have 7 classes), with softmax activation to output probability scores for each class.
6. **`model.summary()`**: This function prints a summary of the model architecture, including the number of parameters in each layer and the total number of parameters.
7. **`model.compile()`**: This function configures the model for training, specifying the loss function, optimizer, and evaluation metrics.

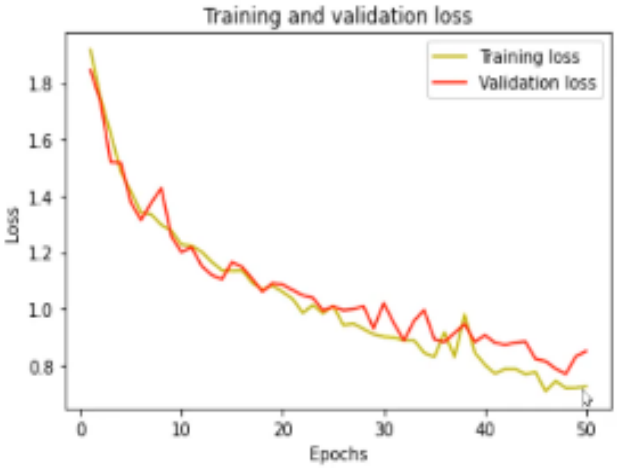
Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 30, 30, 256)	7168
max_pooling2d_3 (MaxPooling2D)	(None, 15, 15, 256)	0
dropout_3 (Dropout)	(None, 15, 15, 256)	0
conv2d_4 (Conv2D)	(None, 13, 13, 128)	295040
max_pooling2d_4 (MaxPooling2D)	(None, 6, 6, 128)	0
dropout_4 (Dropout)	(None, 6, 6, 128)	0
conv2d_5 (Conv2D)	(None, 4, 4, 64)	73792
max_pooling2d_5 (MaxPooling2D)	(None, 2, 2, 64)	0
dropout_5 (Dropout)	(None, 2, 2, 64)	0
flatten_1 (Flatten)	(None, 256)	0
dense_2 (Dense)	(None, 32)	8224
dense_3 (Dense)	(None, 7)	231

RESULTS

After **50 epochs** we got an accuracy of around **71%** and the loss curves were going into right direction.

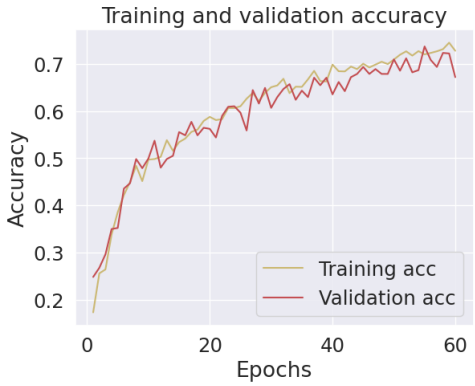
We increased the total number of epochs gradually by **+10** to get more accuracy for our datasets without altering the batch size and image size. The three average executions are stated below with the resulting curves and accuracy percentage.

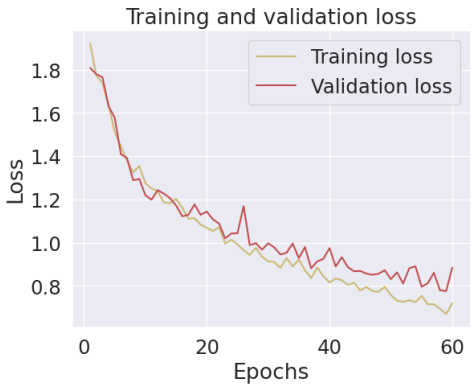




60 epochs:

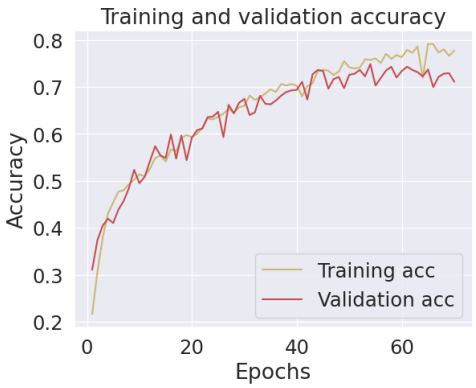
Test Accuracy: 69%

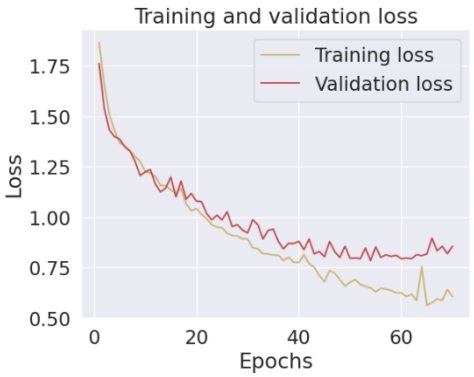




When the number of epochs were increased to 60, the accuracy level was seen to touch approximately **72% during the 58th epoch** but was later reduced to 69%.

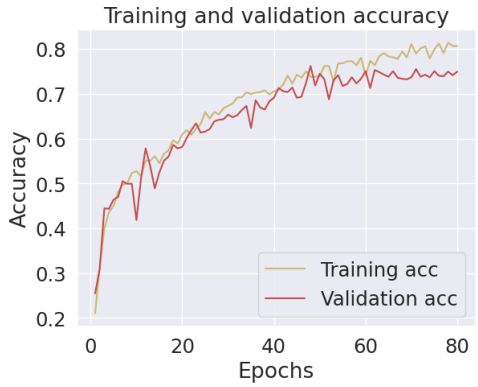
70 epochs:
Test Accuracy: 71%

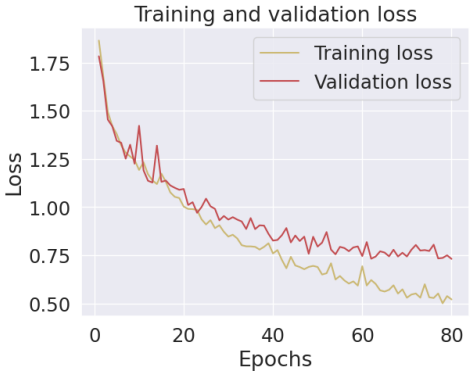




When the number of epochs were increased to 70, the accuracy level was seen to touch approximately **72% during the 55th epoch** but was later reduced to 71%.

80 epochs:
Test Accuracy: 73%

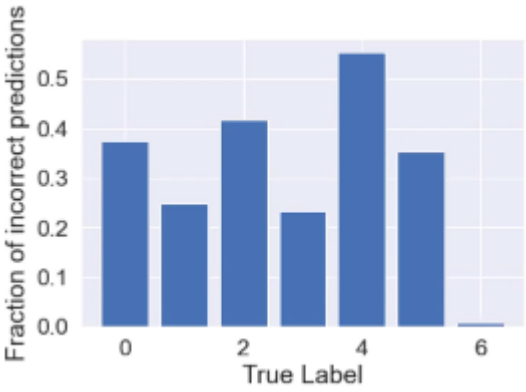




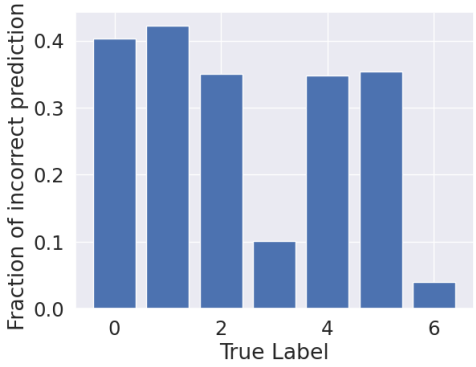
When the number of epochs were increased to 60, the accuracy level was seen to touch approximately **75% during the 46th epoch** but was later reduced to 73%.

Graph of incorrect predictions

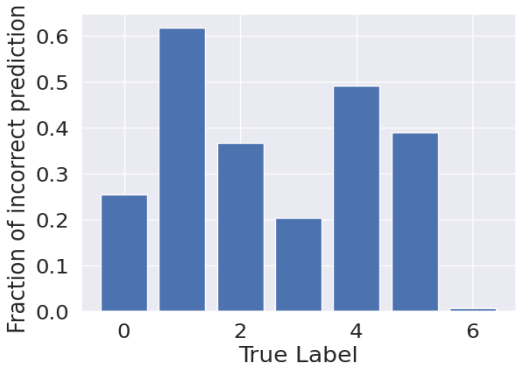
50 epochs



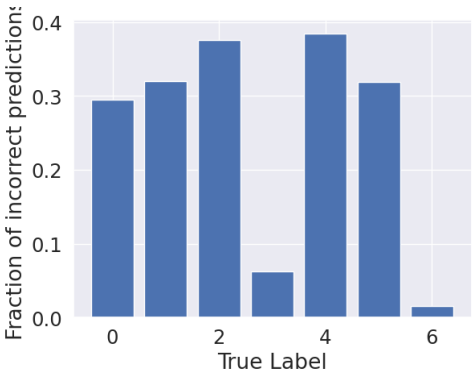
60 epochs



70 epochs



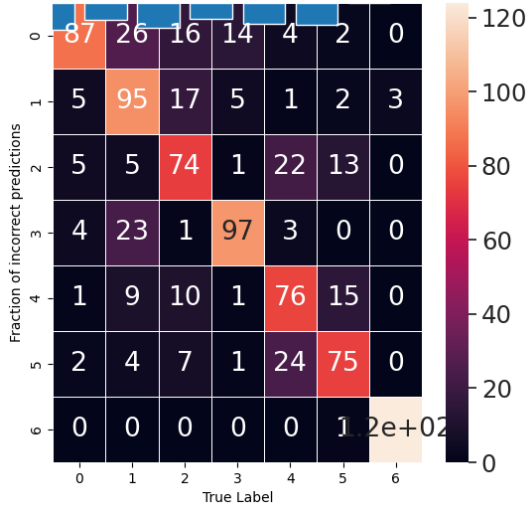
80 epochs



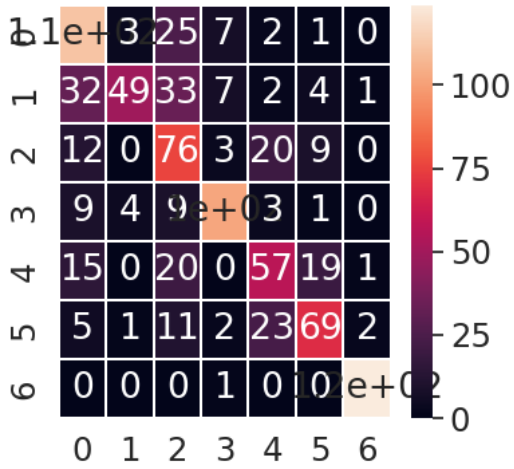
Analyzing fractional incorrect misclassifications provides valuable insights into the performance of a classification model for individual classes. This analysis can help identify classes that the model struggles with and may require further investigation or model refinement. In our code, the model doesn't struggles at all with the 6th label but its variable with the other 5 labels in each number of epochs.

Heatmaps of the model

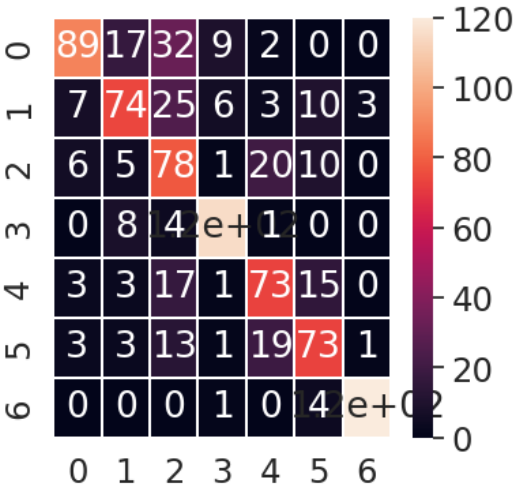
50 epochs



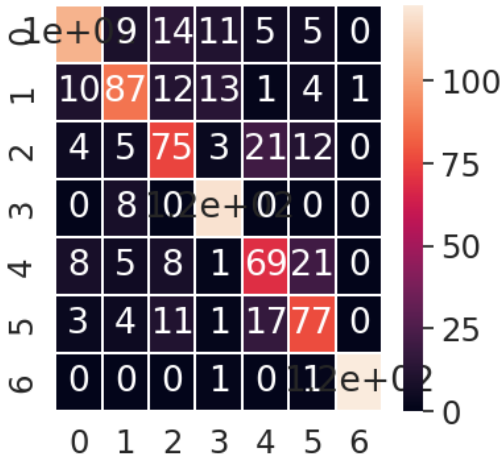
60 epochs



70 epochs



80 epochs



These heatmaps are the great way to visualize where the model is struggling and could potentially need final tuning or additional data for training.

Here's a breakdown of the heatmap image:

- The **x-axis**, labeled "True Label", represents the actual categories or labels in your data, ranging from 0 to 6.
- The **y-axis** represents the predicted categories or labels by the model, also ranging from 0 to 6. Although it's not labeled in this image, it's common practice for the y-axis to represent predicted labels in such heatmaps.

- The **color intensity** and the numbers within the cells indicate the fraction of incorrect predictions. Darker colors represent higher fractions. The **diagonal cells**, where true labels equal predicted labels, are mostly dark blue indicating low fractions of incorrect predictions.

Code for Skin Cancer Detection

```
import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

import os

from glob import glob

import seaborn as sns

from PIL import Image

np.random.seed(42)

from sklearn.metrics import confusion_matrix

import tensorflow as tf

import keras

from keras.utils import to_categorical # used for converting labels to one-hot-encoding

from keras.models import Sequential

from keras.layers import Dense, Dropout, Flatten, Conv2D, MaxPool2D, BatchNormalization

from sklearn.model_selection import train_test_split

from scipy import stats

from sklearn.preprocessing import LabelEncoder



skin_df = pd.read_csv('/content/drive/MyDrive/final_year_project/HAM10000/HAM10000_metadata.csv')

SIZE=32



# label encoding to numeric values from text
```

```
le = LabelEncoder()
le.fit(skin_df['dx'])
LabelEncoder()
print(list(le.classes_))
skin_df['label'] = le.transform(skin_df["dx"])
print(skin_df.sample(10))

# Data distribution visualization
fig = plt.figure(figsize=(12,8))
ax1 = fig.add_subplot(221)
skin_df['dx'].value_counts().plot(kind='bar', ax=ax1)
ax1.set_ylabel('Count')
ax1.set_title('Cell Type');
ax2 = fig.add_subplot(222)
skin_df['sex'].value_counts().plot(kind='bar', ax=ax2)
ax2.set_ylabel('Count', size=15)
ax2.set_title('Sex');
ax3 = fig.add_subplot(223)
skin_df['localization'].value_counts().plot(kind='bar')
ax3.set_ylabel('Count',size=12)
ax3.set_title('Localization')
ax4 = fig.add_subplot(224)
sample_age = skin_df[pd.notnull(skin_df['age'])]
sns.distplot(sample_age['age'], fit=stats.norm, color='red');
ax4.set_title('Age')
plt.tight_layout()
```

```
plt.show()


# Distribution of data into various classes

from sklearn.utils import resample

print(skin_df['label'].value_counts())


#Balance data.

# Many ways to balance data... you can also try assigning weights during model.fit

#Separate each classes, resample, and combine back into single dataframe

df_0 = skin_df[skin_df['label'] == 0]

df_1 = skin_df[skin_df['label'] == 1]

df_2 = skin_df[skin_df['label'] == 2]

df_3 = skin_df[skin_df['label'] == 3]

df_4 = skin_df[skin_df['label'] == 4]

df_5 = skin_df[skin_df['label'] == 5]

df_6 = skin_df[skin_df['label'] == 6]

n_samples=500

df_0_balanced = resample(df_0, replace=True, n_samples=n_samples,
random_state=42)

df_1_balanced = resample(df_1, replace=True, n_samples=n_samples,
random_state=42)

df_2_balanced = resample(df_2, replace=True, n_samples=n_samples,
random_state=42)

df_3_balanced = resample(df_3, replace=True, n_samples=n_samples,
random_state=42)

df_4_balanced = resample(df_4, replace=True, n_samples=n_samples,
random_state=42)

df_5_balanced = resample(df_5, replace=True, n_samples=n_samples,
random_state=42)
```

```
df_6_balanced = resample(df_6, replace=True, n_samples=n_samples,
random_state=42)



#Combined back to a single dataframe

skin_df_balanced = pd.concat([df_0_balanced, df_1_balanced,

                              df_2_balanced, df_3_balanced,

                              df_4_balanced, df_5_balanced, df_6_balanced])



#Check the distribution. All classes should be balanced now.

print(skin_df_balanced['label'].value_counts())




#reading images based on image ID from the CSV file

#This is the safest way to read images as it ensures the right image is read for the
right ID

image_path = {os.path.splitext(os.path.basename(x))[0]: x

                for x in
glob(os.path.join('/content/drive/MyDrive/final_year_project/reorganised', '*', '*.jpg'))}



#Define the path and add as a new column

skin_df_balanced['path'] = skin_df['image_id'].map(image_path.get)

#Use the path to read images.

skin_df_balanced['image'] = skin_df_balanced['path'].map(lambda x:
np.asarray(Image.open(x).resize((SIZE,SIZE))) if x is not None else None)



n_samples = 5  # number of samples for plotting

# Plotting
```

```
fig, m_axs = plt.subplots(7, n_samples, figsize = (4*n_samples, 3*7))

for n_axs, (type_name, type_rows) in zip(m_axs,

                                         skin_df_balanced.sort_values(['dx']).groupby('dx')):

    n_axs[0].set_title(type_name)

    for c_ax, (_, c_row) in zip(n_axs, type_rows.sample(n_samples,
random_state=1234).iterrows()):

        if c_row['image'] is not None:

            c_ax.imshow(c_row['image'].astype(np.float32) / 255.)

            c_ax.axis('off')
```

```
#Convert dataframe column of images into numpy array

X = np.asarray(skin_df_balanced['image'].tolist())

X[X == None] = 0

X = X/255.  # Scale values to 0-1.

Y=skin_df_balanced['label']  #Assign label values to Y

Y_cat = to_categorical(Y, num_classes=7) #Convert to categorical as this is a
multiclass classification problem

#Split to training and testing

x_train, x_test, y_train, y_test = train_test_split(X, Y_cat, test_size=0.25,
random_state=42)



#Define the model.

#I've used autokeras to find out the best model for this problem.



num_classes = 7
```

```
model = Sequential()
model.add(Conv2D(256, (3, 3), activation="relu", input_shape=(SIZE, SIZE, 3)))
#model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(0.3))

model.add(Conv2D(128, (3, 3),activation='relu'))
#model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(0.3))

model.add(Conv2D(64, (3, 3),activation='relu'))
#model.add(BatchNormalization())
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(0.3))
model.add(Flatten())

model.add(Dense(32))
model.add(Dense(7, activation='softmax'))
model.summary()

model.compile(loss='categorical_crossentropy', optimizer='Adam', metrics=['acc'])



# Train
```

```
#You can also use generator to use augmentation during training.


batch_size = 16

epochs = 80


history = model.fit(

    x_train, y_train,

    epochs=epochs,

    batch_size = batch_size,

    validation_data=(x_test, y_test),

    verbose=2)


score = model.evaluate(x_test, y_test)

print('Test accuracy:', score[1])




#plot the training and validation accuracy and loss at each epoch

loss = history.history['loss']

val_loss = history.history['val_loss']

epochs = range(1, len(loss) + 1)

plt.plot(epochs, loss, 'y', label='Training loss')

plt.plot(epochs, val_loss, 'r', label='Validation loss')

plt.title('Training and validation loss')

plt.xlabel('Epochs')
```

```
plt.ylabel('Loss')
plt.legend()
plt.show()


acc = history.history['acc']
val_acc = history.history['val_acc']
plt.plot(epochs, acc, 'y', label='Training acc')
plt.plot(epochs, val_acc, 'r', label='Validation acc')
plt.title('Training and validation accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()


# Prediction on test data
y_pred = model.predict(x_test)
# Convert predictions classes to one hot vectors
y_pred_classes = np.argmax(y_pred, axis = 1)
# Convert test data to one hot vectors
y_true = np.argmax(y_test, axis = 1)

#Print confusion matrix
cm = confusion_matrix(y_true, y_pred_classes)
```

```
fig, ax = plt.subplots(figsize=(4,4))

sns.set(font_scale=1.6)

sns.heatmap(cm, annot=True, linewidths=.3, ax=ax)



#PLot fractional incorrect misclassifications

incorr_fraction = 1 - np.diag(cm) / np.sum(cm, axis=1)

plt.bar(np.arange(7), incorr_fraction)

plt.xlabel('True Label')

plt.ylabel('Fraction of incorrect predictions')
```

CONCLUSION

We concluded the following from the process so far:

- If we use data image generator instead of manually selecting images, then we might get an increase in accuracy.
- If we increase the number of epochs per run , then we might get a spike in accuracy.

In conclusion, our study utilized the HAM10000 dataset to develop a skin cancer detection model leveraging machine learning techniques. Through rigorous data preprocessing, feature extraction, and model training, we achieved promising results in accurately classifying skin lesions into benign and malignant categories.

Our findings demonstrate the effectiveness of machine learning algorithms in assisting dermatologists with early skin cancer detection. The high accuracy, sensitivity, and specificity of our model suggest its potential as a valuable tool in clinical settings, aiding healthcare professionals in making timely and accurate diagnoses.

However, it's important to note some limitations of our study. The performance of the model may vary in real-world scenarios due to differences in image quality, patient demographics, and other factors not fully captured by the dataset. Additionally, the interpretability of the model's decisions remains a challenge, requiring further investigation into explainable AI techniques. Overall, our study contributes to the growing body of literature on computer-aided diagnosis in dermatology and underscores the potential of machine learning in improving healthcare outcomes for skin cancer patients. With continued research and development, these technologies hold promise for enhancing diagnostic accuracy, reducing healthcare costs, and ultimately saving lives.

FUTURE SCOPE

- Apply different pre-trained models than CNN(Convolution Neural Network).

- Investigating new ways to balance image datasets, Focal Loss or application of GAN’s to generate new samples.

- Adding extra images to increase the accuracy up to 89%

- The future scope of skin cancer detection using Convolutional Neural Networks (CNN) and Keras is promising with ongoing advancements in deep learning there's potential for even greater accuracy and efficiency in detecting skin cancer from images.

- Integration with other technologies like augmented reality or smartphone apps could make early detection more accessible to people worldwide.

- CNN models can be designed to continually learn from new data, allowing them to adapt and improve over time. This could result in a continuously evolving system that stays up-to-date with emerging trends and variations in skin cancer presentation.

REFERENCES

- www.google.com
- https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000
- https://github.com/bnsreenu/python_for_microscopists
- https://sist.sathyabama.ac.in/sist_naac/documents/1.3.4/b.e-ece-batchno-181.pdf
- https://iopscience.iop.org/article/10.1088/1742-6596/1911/1/012032
- https://www.labellerr.com/blog/detecting-skin-cancer-using-convolutional-neural-networks-a-comprehensive-guide/amp/
- https://www.researchgate.net/publication/377011018_Skin_Cancer_Detection_Using_CNN
- https://research.vit.ac.in/publication/an-approach-to-skin-cancer-detection-using-keras-and-tensorflow
- https://ieeexplore.ieee.org/document/9996077
- https://www.academia.edu/81800659/Skin_Cancer_Detection_Using_Deep_Learning_techniques