A Mini Project Report On

COMPARITIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR SKIN CANCER DETECTION

Submitted in partial fulfillment of the requirements for the award of the degree

Bachelor of Technology

In

Department of Computer Science and Engineering

By

N. Sairam 21241A05G9

Under the Esteemed guidance of

R. Soujanya

Assistant Professor

Department of Computer Science and Engineering

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Bachupalli, Kukatpally, Hyderabad, Telangana, India,500090

2023-2024

GOKARAJU RANGARAJU

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

CERTIFICATE

This is to certify that the minor project entitled "**Comparative Analysis of Machine Learning Algorithms for Skin Cancer Detection**" is submitted by **N. Sairam (21241A05G9)** in partial fulfillment of the award of a degree in BACHELOR OF TECHNOLOGY in Computer Science and Engineering during the academic year **2023-2024.**

INTERNAL GUIDE

R. SOUJANYA

Assistant Professor

HEAD OF THE DEPARTMENT

Dr. B. SANKARA BABU

Professor

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

Many people helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we wish to express our deep gratitude to our internal guide **R. Soujanya**, **Assistant Professor**, Department of CSE for his support in the completion of our project report. We wish to express our honest and sincere thanks to **Dr. B. Sankara Babu, HOD,** Department of CSE, and to our principal **Dr. J. Praveen** for providing the facilities to complete our mini project. We would like to thank all our faculty and friends for their help and constructive criticism during the project completion phase. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

N. Sairam **21241A05G9**

DECLARATION

We hereby declare that the Mini Project entitled "**Comparative Analysis of Machine Learning Algorithms for Skin Cancer Detection**" is the work done during the period from **2023-2024** and is submitted in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering from **Gokaraju Rangaraju Institute of Engineering and Technology.** The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

N. Sairam **21241A05G9**

ABSTRACT

Cancer is one of the most lethal illness that strikes in the hearts of people. There are almost 100 types of cancers each having their own characteristics and behaviour. One of them is skin cancer. Skin cancer can potentially lead to death if it is not detected in time and treated well. Skin cancer detection poses a challenge in dermatology, seeking efficient diagnosis methods for timely intervention. Conventional approach may lead to errors in diagnosing the disease or delay in detection of the disease. This project aims to provide efficient approach in detecting the disease at early stages by the user himself.

To solve this problem a comprehensive dataset of dermoscopic images is preprocessed and enhanced the image quality followed by feature extraction to extract characteristics for classification. Several machine algorithms like Convolutional neural networks (CNN) are used in detection and comparing the effectiveness of each approach. The results of the comparative analysis will provide a valuable information of the efficiency of different algorithms in detection of skin cancer. Overall, this project contributes for the automated skin cancer detection by identifying the optimal machine learning techniques and aims to enhance the early detection and improve patient outcomes.

CHAPTER 1

INTRODUCTION

1. Introduction

Cancer is one of the most deadliest disease, which cause death in millions over a year. The cancer cells will spread all over the body in no time. There are over 100 types of cancers like lung cancer, breast cancer, hair cancer, skin cancer and many more, each of it have their own symptoms and dangerous in their own ways. Detecting any type of cancer at early stage place crucial role in diagnosing it.

As cancer is one of the reason for causing death, skin cancer being the common cancer in causing deaths. Skin cancer is divided as Melanoma and Non-Melanoma, under Non-Melanoma there are various types like Basal Cell Carcinoma (BCC), Dermatofibroma, Vascular Lesions etc. According to World Health Organization (WHO), around 132,000 Melanoma cases and 2-3 million Non-Melanoma cases found globally each year.

Although there are several existing techniques in detecting skin cancer they have their own limitations. With traditional methods in detecting skin cancer there are disadvantages like, detection takes lots of time and some times results might not be accurate, considering these challenges we have integrated the deep learning techniques in detecting skin cancer at early stage. We have incorporated 7 different classes of skin cancer for detection. Because of this the model can predict vast classes of skin cancers.

In designing our deep learning model we have used several machine learning algorithms like Convolutional Neural Network (CNN) and Support Vector Machine (SVM) to train the model and the dataset we have used is HAM10000 dataset from Kaggle which consists of 10015 images of all 7 classes. Augumentation is done for achieving the balanced dataset so that model can train accurately.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 SOFTWARE REQUIREMENTS:

- **Operating System** **:** Windows 10 or higher
- **Coding language** **:** Python
- **Text Editor** **:** Spyder
- **Framework** **:** Flask

2.2 HARDWARE REQUIREMENTS:

- **Hard disk** **:** 50GB
- **RAM** **:** 8GB
- **Processor** **:** i5 or above
- **GPU** **:** IRISXe

2.3 DATA SET:

Skin lesions images of all 7 classes of skin cancer is collected from kaggle website as named HAM10000 dataset consisting of 10015 images.

CHAPTER 3

LITERATURE SURVEY

Sana Nazri, Rafael Garcia. **[1]** they performed Automatic Skin Cancer Detection Using Clinical Images in year 2023. They used Support Vector Machine (SVM) for training dataset, this datasets involves MedNode Dataset, DermaQuest Dataset. But the draw back that performed obtained they used Asymmetric Boarder Color Diameter (ABCD) for feature extraction but our Convolution Neural Network extracts features directly in an efficient manner.

Naga Srinivasu, Gnana sivasai, Fazal ijaz. **[2]** they performed Classification of Skin Disease Using Deep Learning Neural Networks in year 2021. They integrating LSTM with the MobileNet V2 on dataset HAM10000. But this failed due to this MobileNetV2 with the LSTM model needs a more significant number of parameters for better accuracy.

Mehwish Dildar, Shumalia Akram, Muhammed Irfan, Abdul Hakeem. **[3]** they performed Skin Cancer detection using Deep learning techniques in year 2021. They used Generative Adversal Network (GAN) for training datasets involves DermQuest Dataset, AtlasDerma Dataset. But the draw back that obtained is that this methodologies is not much efficient for training.

Andre Esteva, Brett Kuprel. **[4]** they made dermatologist-level classification of skin cancer with deep neural networks in year 2017. They used Convolution Neural Network (CNN) on dataset ISIC Dermoscopic Archive, the Edinburgh Dermofit Library and data from the Stanford Hospital. This not provide sufficient validate on unseen or external datasets to asses the model's generalization beyond the training datset.

John Mitchell. **[5]** they performed Skin Disease Recognition Method Based in image color in year 2018. They used Support Vector Machine (SVM) on PH2 Dataset. They used for disease recognization but this don't produce accurate results because this won't extract all features from the images this lead to inaccurate results for the end user.

Yeh-Chi-Lo, Ming Chao. **[6]** they overviewed on Automatic Skin Lesion Segmentation Using Deep Fully Convolution Networks. They use the methodologies like Jaccard Distance for skin cancer detection. They use ISBI 2016 Skin Lesion dataset, but their model won't perform accurate because their methodology is not sufficient to train the model.

CHAPTER 4

PROPOSED APPROACH, MODULES DESCRIPTION AND UML DIAGRAMS

4.1 Proposed Approach

This concept has many phases that are crucial. First, obtain the data from the datasets and then the data split into segments where we have the training data set followed by the testing data set. And then the preprocessing and the data augmentation which are very crucial in the boosting of the model are carried out and performed. Thus, additional approaches include increasing the zoom factor, angulation, and flipping to increase the level of diversification. And then we use Convolution Neural Network (CNN) Algorithm with that we can to find or can extract the features of the images on the augumented data-set. Following that, passes to refine of the initial trained model to meet the requirements of a stringent validation check learning for making the model more general and better. When done for performance, accuracy, sensitivity, and specificity are examples of some of the key values recovered.

4.2 Modules

1. Data Collection: All datasets used in our research are sourced from Kaggle site and particularly, HAM10000 dataset includes 10,015 images of skin lesions, which are distributed into 7 classes of skin cancer. Moreover, we have also received an image of a CSV file which show us that in which skin class which image belongs to.

2. Data Preprocessing: Once the data is gathered, preprocessing is performed on the data set by scaling the pixel intensity values and resizing the images to the input shape needed. The augmentation is then applied to balance the data so as to have the same number of images in each class. Last, the data is split into a training set and a test set.

3. Feature Extraction: The third and most sensitive step for the model to learn appropriately is feature extraction. Before deciding whether or not an image suggests cancer, it is critical to extract features. Hence, for extracting features, the Convolutional Neural Network (CNN) algorithm is employed, as it is able to extract features by using convolutional layers.

4. Model Development: CNNs are applied in the development of a model in skin cancer detection because they are efficient on images. The model is optimized on the training data set and then adjusted in order to create the most effective model on the validation set. Different fine tuning parameters are used to train the model efficiently.

5. Evaluation: After that, the model that is built has to be tested to check its performance using several evaluation parameters. The high accuracy means the better model we are going to have. The measures that are used to evaluate a model are accuracy, precision, and recall.

6. Deployment: It is made user friendly for the end users and it is made in such a way that even one can predict cancer all by him or herself. The model is implemented in the front end by the Flask application in which the user uploads an image and the application predicts if the cancer is of a certain type or the subject has no cancer.

4.3 UML Diagrams

4.3.1 USECASE DIAGRAM

The interaction between actors and system functionalities are represented graphically in a use case diagram the users can upload images of the affected skin area and prediction is done. The connections between actors and the use cases are shown in the diagram.

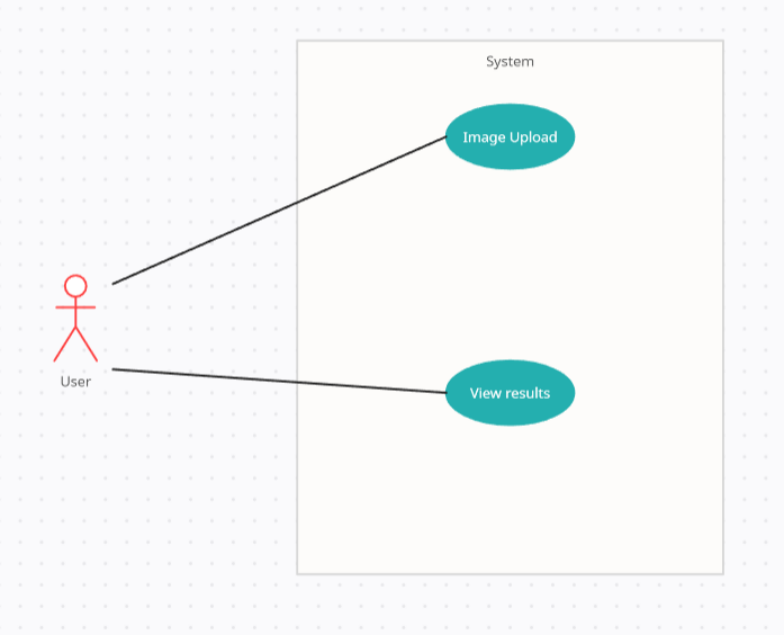

Fig. No:1 Use Case Diagram

4.3.2 CLASS DIAGRAM:

A class diagram shows how classes are related to one another and how system is organized It displaces the classes along with their methods, characteristics, and relationships it aids in system design and communication and gives a personal picture of the system structure.

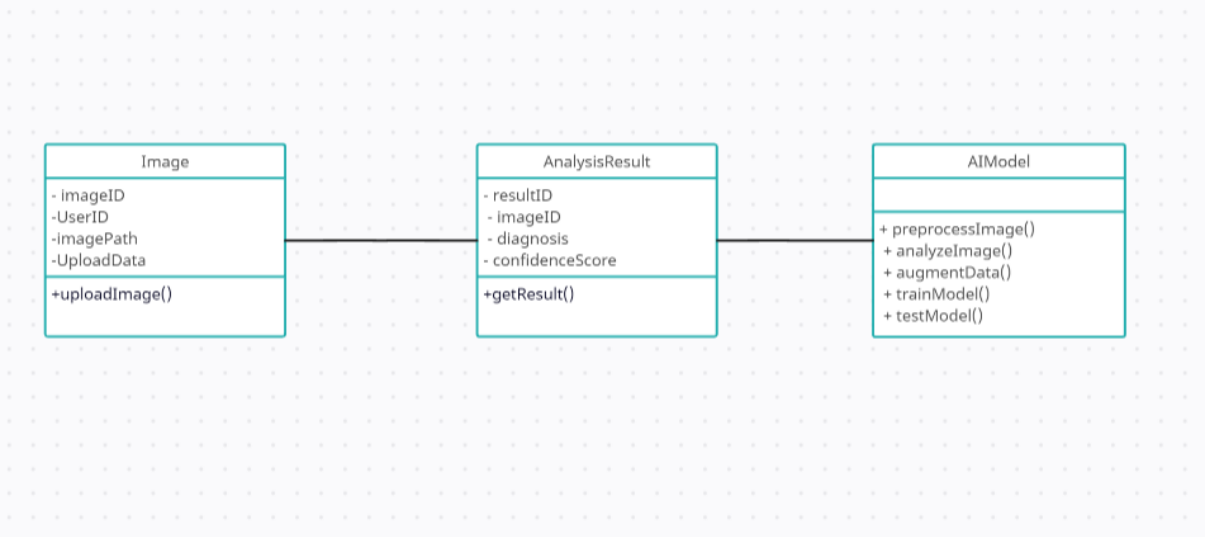

Fig. No:2 Class Diagram

4.3.3 ACTIVITY DIAGRAM

Activity Diagram visually represent the flow of activities, actions and decisions in a system or process. They display the sequence of actions, decision points and transitions between activities. Activity diagrams help understand workflow, identify bottlenecks, and communicate system behavior.

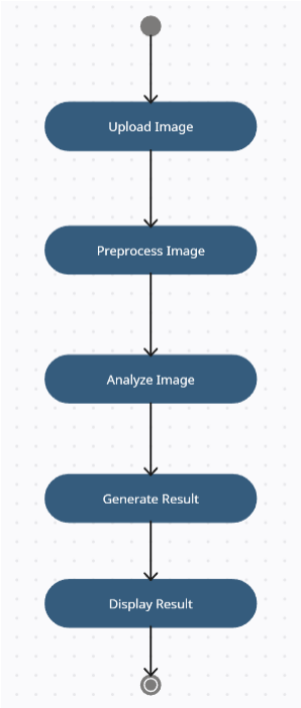

Fig.No:3 Activity Diagram

4.3.4 SEQUENCE DIAGRAM

One type of interaction diagram that illustrate how and in what order processes interact with one another is a sequence diagram. It is a Message Sequence Chart construct. An object's interactions are arranged chronologically in a sequence diagram. It shows the classes and objects that are a part of the scenario as well as the messages that are passed between the objects in order for the scenario to function. Sequence diagrams are commonly linked to the realizations of use cases in the Logical View of the system that is being developed. Event diagrams, event situations, event timing diagram are other names for sequence diagram

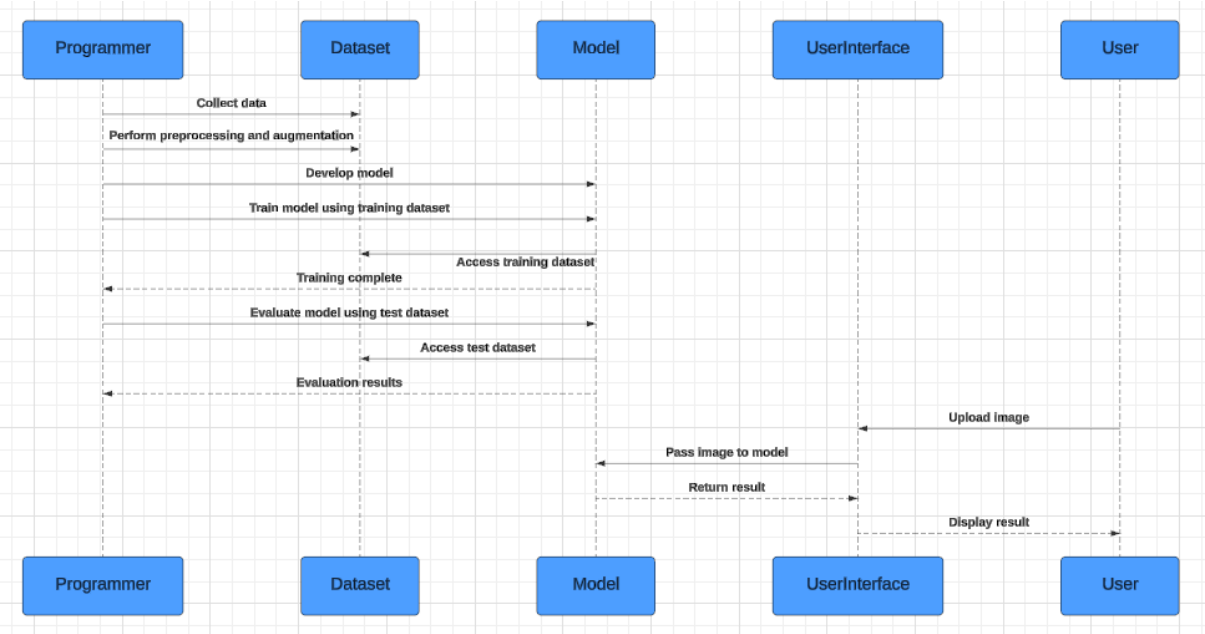

Fig.No:4 Sequence Diagram

4.3.5 DEPLOYMENT DIAGRAM:

The visual representation of software and hardware components distributed among network nodes is achieved by a deployment diagram. It displays the component relationships as well as their physical distribution. Deployment diagrams aid in stakeholder communication, the knowledge of system architecture, and the identification of possible problem.

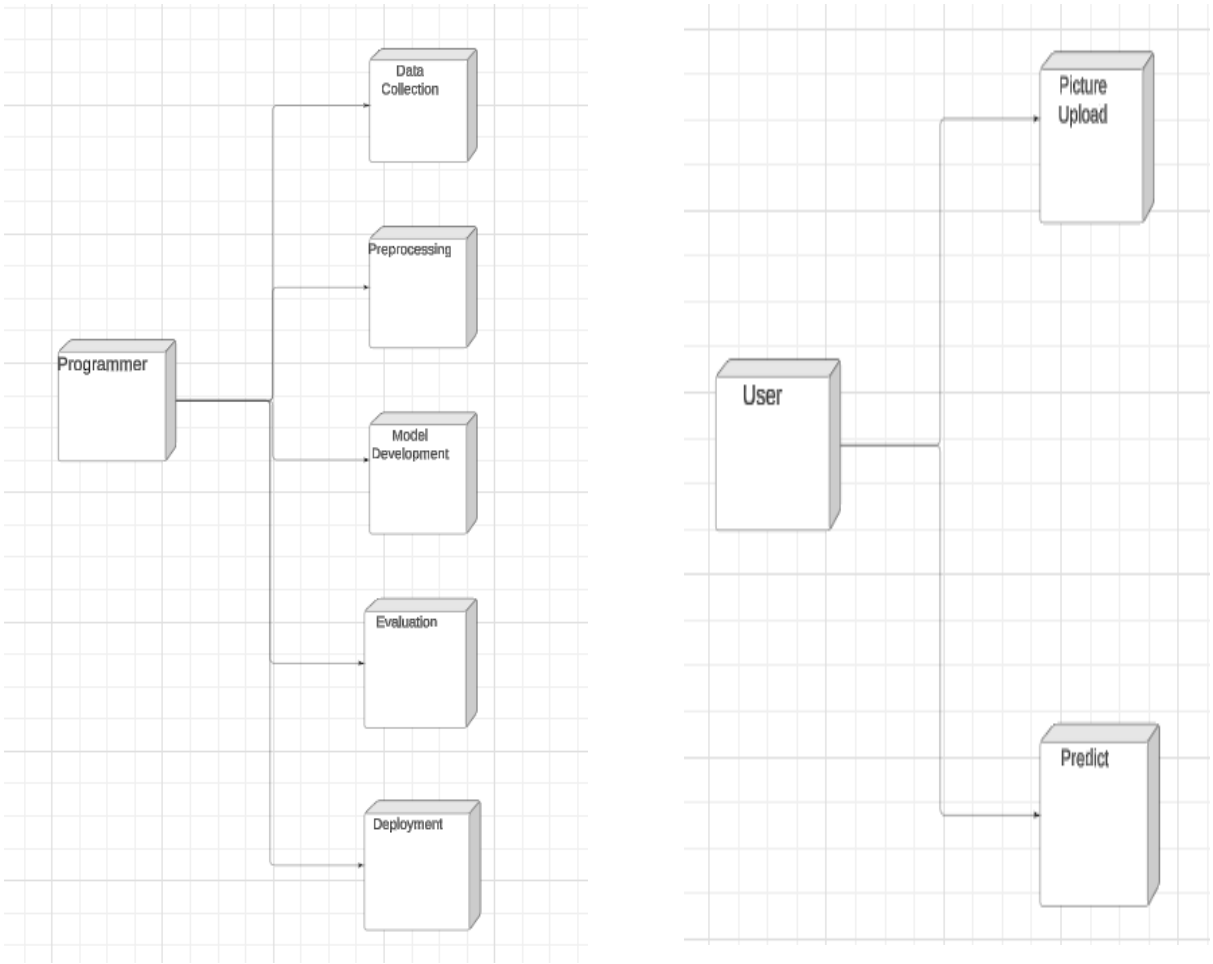

Fig.No:5 Deployment Diagram

CHAPTER 5

IMPLEMENTATION, EXPERIMENT RESULTS AND TEST CASES

Implementation

model.py

The below code model.py is the most important component of our project skin cancer detection. It manages the model that predicts the results of our project. To develop the model we used the algorithm Convolutional Neural networks.

```
import tensorflow as tf

import keras
import numpy as np
from tensorflow.keras.preprocessing.image import ImageDataGenerator

from keras.models import Sequential

from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense

train_datagen = ImageDataGenerator(rescale=1/255 ,shear_range = 0.2 , zoom_range = 0.2 )

training_set = train_datagen.flow_from_directory('D:/Mini Project 3-2/Final project/models/split_data_balanced/train',target_size=(64,64),batch_size=32,class_mode='categoric

training_set.class_indices

test_datagen = ImageDataGenerator(rescale=1/255)

test_set = test_datagen.flow_from_directory('D:/Mini Project 3-2/test_data',target_size=(64,64),batch_size=32,class_mode='categorical')

classifier = Sequential()
```

We used the components like Conv2D, MaxPolling and also added extra layers like Flatten layer and Dense Layer.

Saved the model as HDF5 file and loaded the model for testing.

```
# Add convolutional layers
classifier.add(Conv2D(filters=32, kernel_size=3, activation='relu', input_shape=(64, 64, 3)))
classifier.add(MaxPooling2D(pool_size=2))

# Add more convolutional layers
classifier.add(Conv2D(filters=64, kernel_size=3, activation='relu'))
classifier.add(MaxPooling2D(pool_size=2))

# Add more convolutional layers
classifier.add(Conv2D(filters=128, kernel_size=3, activation='relu'))
classifier.add(MaxPooling2D(pool_size=2))

# Flatten the output before passing to the fully connected layers
classifier.add(Flatten())

# Add dense layers
classifier.add(Dense(units=256, activation='relu'))
classifier.add(Dense(units=128, activation='relu'))

# Output layer
classifier.add(Dense(units=7, activation='softmax'))

# Compile the model
classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Display the model summary
classifier.summary()
```

```
# Assuming you have `training_set` and `test_set` defined
# Train the model
history = classifier.fit(x=training_set, validation_data=test_set, epochs=50)

# Save the model
classifier.save('last_model.h5')

# Load the model
model = tf.keras.models.load_model('last_model.h5')
test_set.reset()
predictions = classifier.predict(test_set, steps=test_set.n // test_set.batch_size + 1)
predicted_classes = np.argmax(predictions, axis=1)
true_classes = test_set.classes

from sklearn.metrics import precision_score, recall_score, f1_score, classification_report
class_labels = list(test_set.class_indices.keys())

# Generating the Classification Report
report = classification_report(true_classes, predicted_classes, target_names=class_labels)
print(report)
```

app.py

The below code app.py is mainly used in our project to run it in the localhost. This will load our saved model (lastmodel.h5) and render our main page template, upload page template, and finally the result page template.

```
from flask import Flask, render_template, request, redirect, url_for
import tensorflow as tf
from tensorflow.keras.preprocessing.image import img_to_array, load_img
import numpy as np
import os

app = Flask(__name__)
model = tf.keras.models.load_model('D:/Mini Project 3-2/Final project/models/last_model.h5')

UPLOAD_FOLDER = 'D:/Mini Project 3-2/Final project/uploads'
if not os.path.exists(UPLOAD_FOLDER):
    os.makedirs(UPLOAD_FOLDER)

app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

# Define the 7 classes of skin cancer
classes = ['Melanoma', 'Melanocytic nevi', 'Basal cell carcinoma', 'Actinic keratoses', 'Benign keratosis-like lesions', 'Dermato

@app.route('/')
def index():
    return render_template('index.html')
```

```
@app.route('/upload', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        if 'file' not in request.files:
            return redirect(request.url)
        file = request.files['file']
        if file.filename == '':
            return redirect(request.url)
        if file:
            filepath = os.path.join(app.config['UPLOAD_FOLDER'], file.filename)
            file.save(filepath)
            return redirect(url_for('result', filename=file.filename))
    return render_template('uplaod2.html')

@app.route('/result/<filename>')
def result(filename):
    filepath = os.path.join(app.config['UPLOAD_FOLDER'], filename)

    # Load and preprocess the image
    img = load_img(filepath, target_size=(64, 64))
    img = np.array(img)
    img = np.expand_dims(img, axis=0)
```

```
    predictions = model.predict(img)
    if(predictions[0][0]==1):
        x="Actinic keratoses"
    elif(predictions[0][1]==1):
        x="Basal cell carcinoma"

    elif(predictions[0][2]==1):
        x="Benign keratosis like lesions"
    elif(predictions[0][3]==1):
        x="Dermatofibroma"
    elif(predictions[0][4]==1):
        x="Melanoma"
    elif(predictions[0][5]==1):
        x="Melanocytic nevi"
    elif(predictions[0][6]==1):
        x="Vascular lesions"



    return render_template('result.html', predicted_class=x, filename=filename)

if __name__ == '__main__':
    app.run(debug=True)
```

EXPERIMENTAL RESULTS:

The interface design in our experiment incorporated a button labeled “Click to Proceed,” positioned on the main page, which led the subjects to an image upload option. In this stage, the users upload their skin lesion images and then press the “Predict” button for processing by the built-in skin cancer detection model. The outcomes are displayed on a separate results page, which enlightens whether the lesion exhibits skin cancer features or not and in the affirmative case, the type is highlighted. Being an interface design it has a focus on simplicity and ease of use to ensure high speed and accurate diagnostic reasoning from the system.

Fig. No: 6 Main Interface

This webpage allows users to upload an image of a skin lesion for analysis. Our model assesses the image and identifies the type of cancer present.

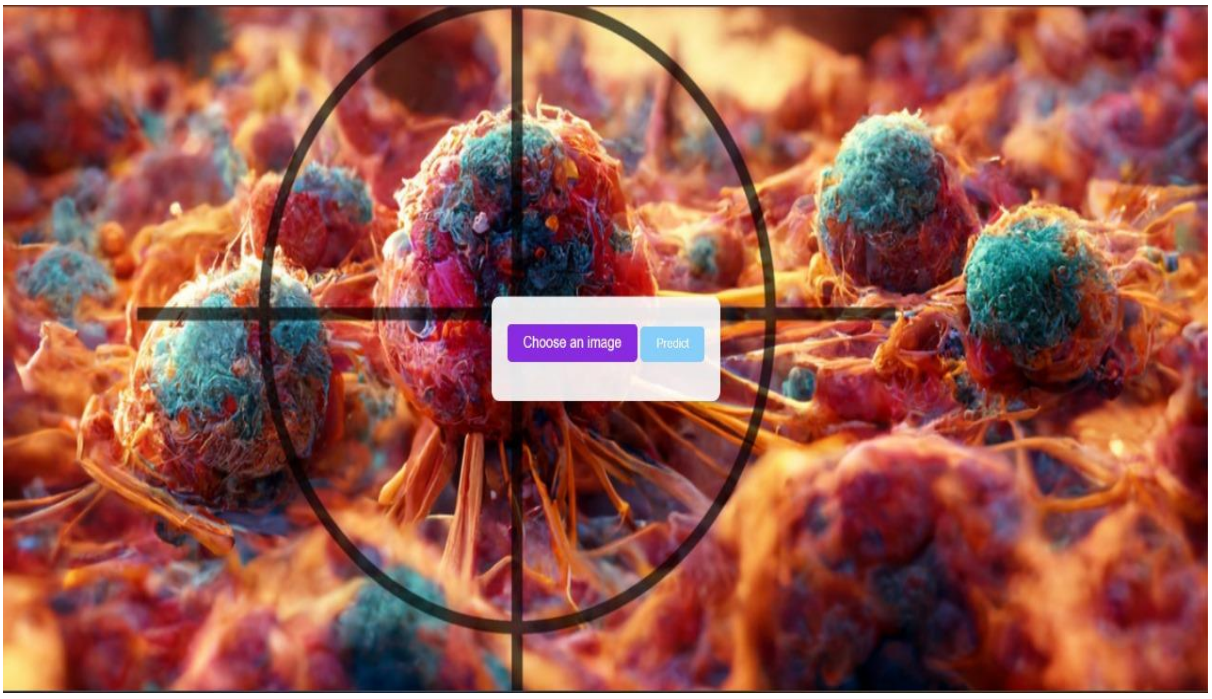

Fig. No: 7 Webpage where Image uploaded

When you uploaded the image you have depicted a skin lesion, our model has processed the image and the type of cancer, if there is, in the lesion has been identified. Here are the results of the analysis as follows.

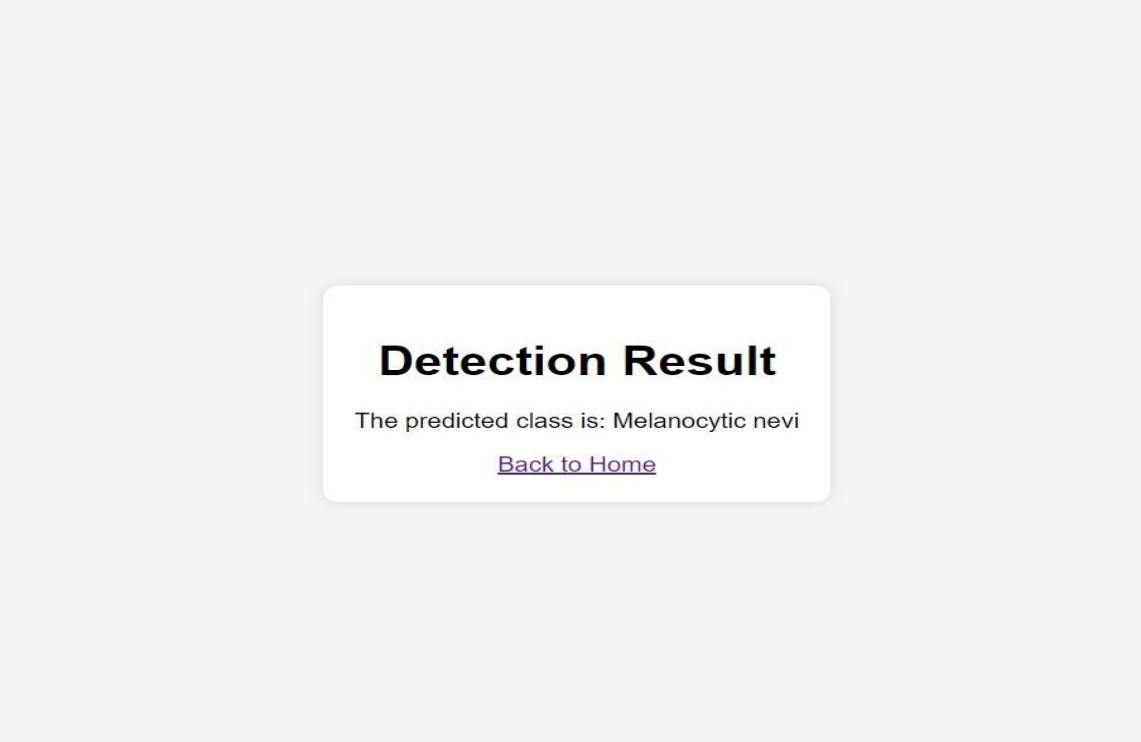

Fig. No: 8 Webpage where output get displayed

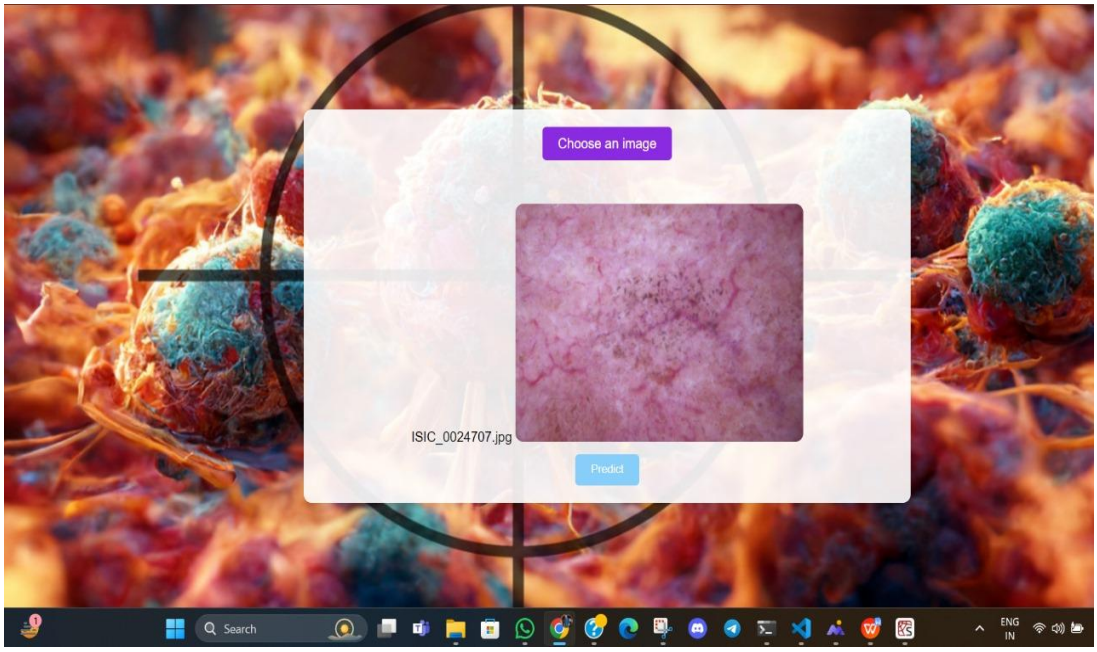

Fig. No: 9 Test Case-1 Actinic Keratosis

Fig. No: 10 Test case-2 Basal Cell Carcinoma

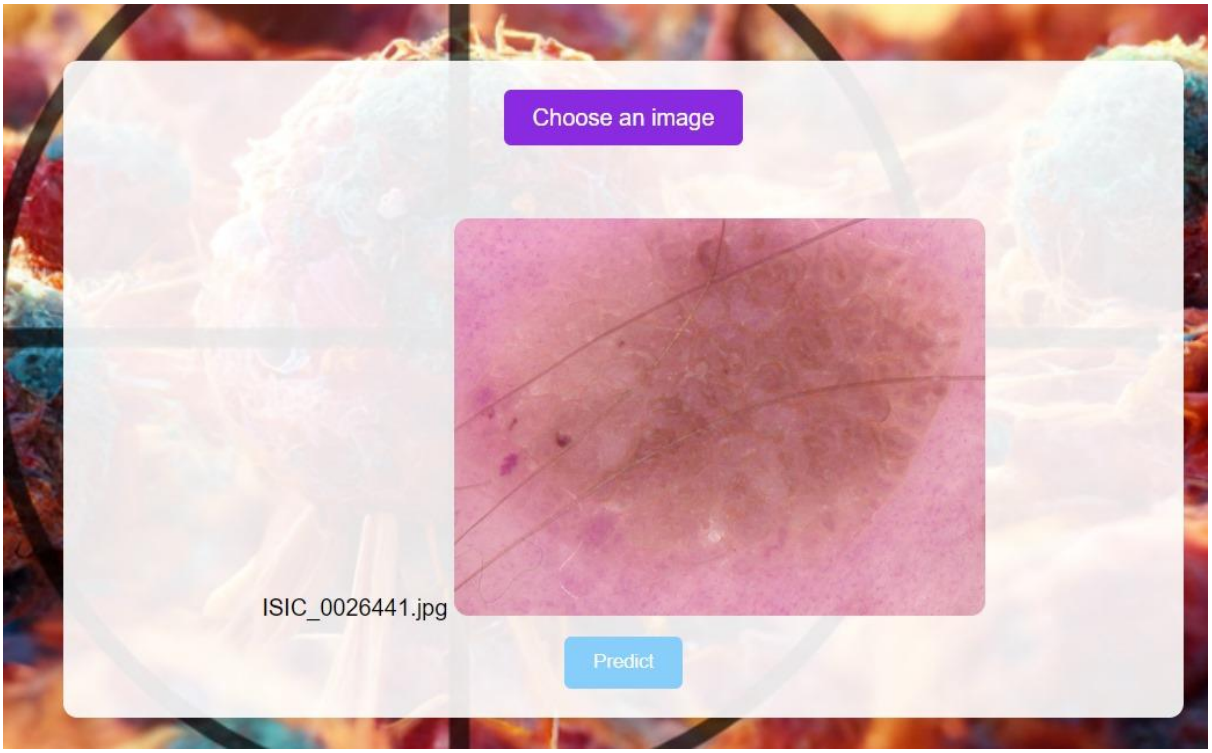

Fig. No: 11 Test Case-3 Benign Keratosis

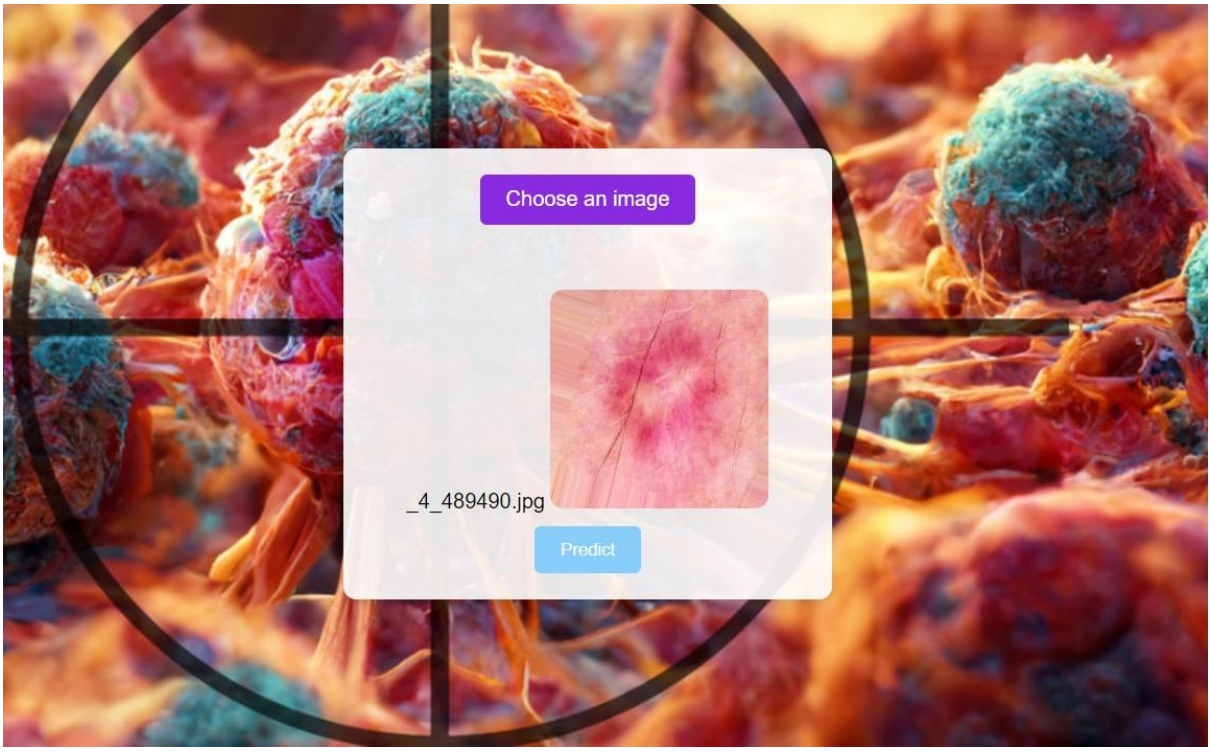

Fig. No: 12 Test Case-4 Dermatofibroma

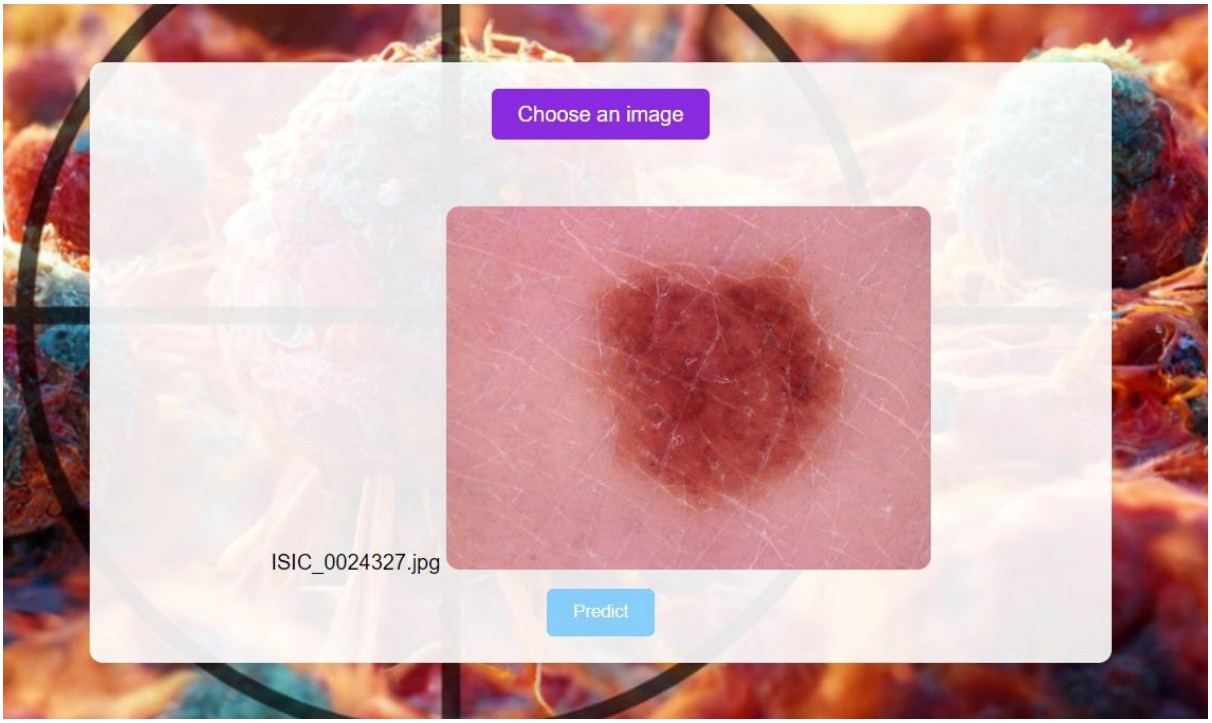

Fig. No: 13 Test Case-5 Melanocytic nevi

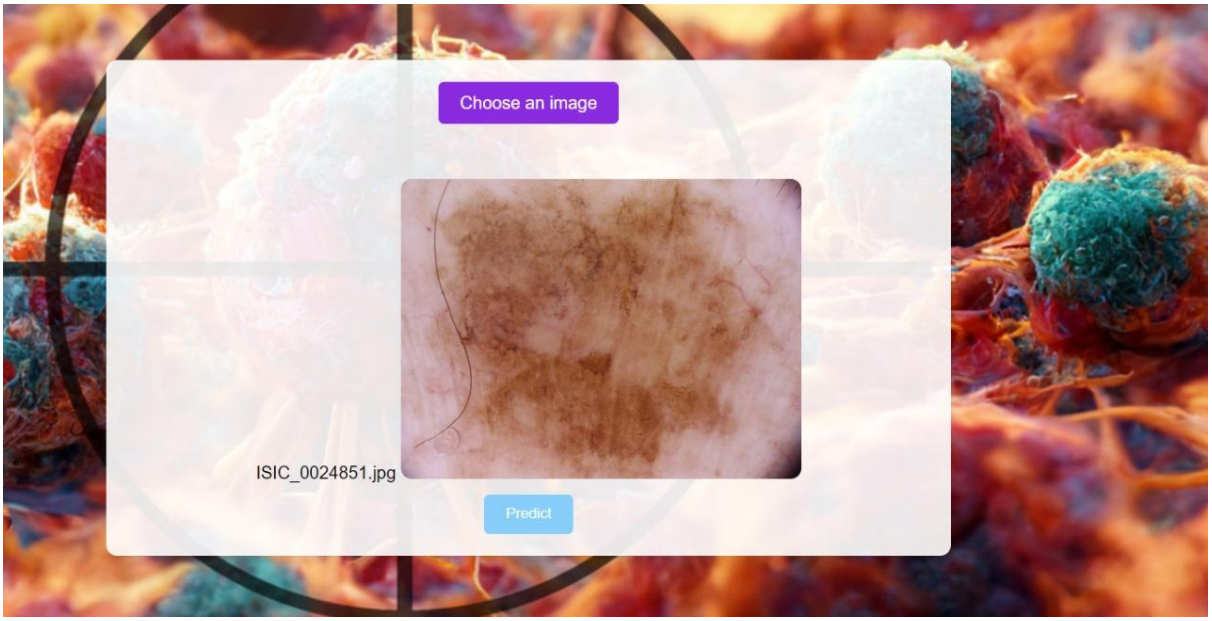

Fig. No: 14 Test Case-6 Melanoma

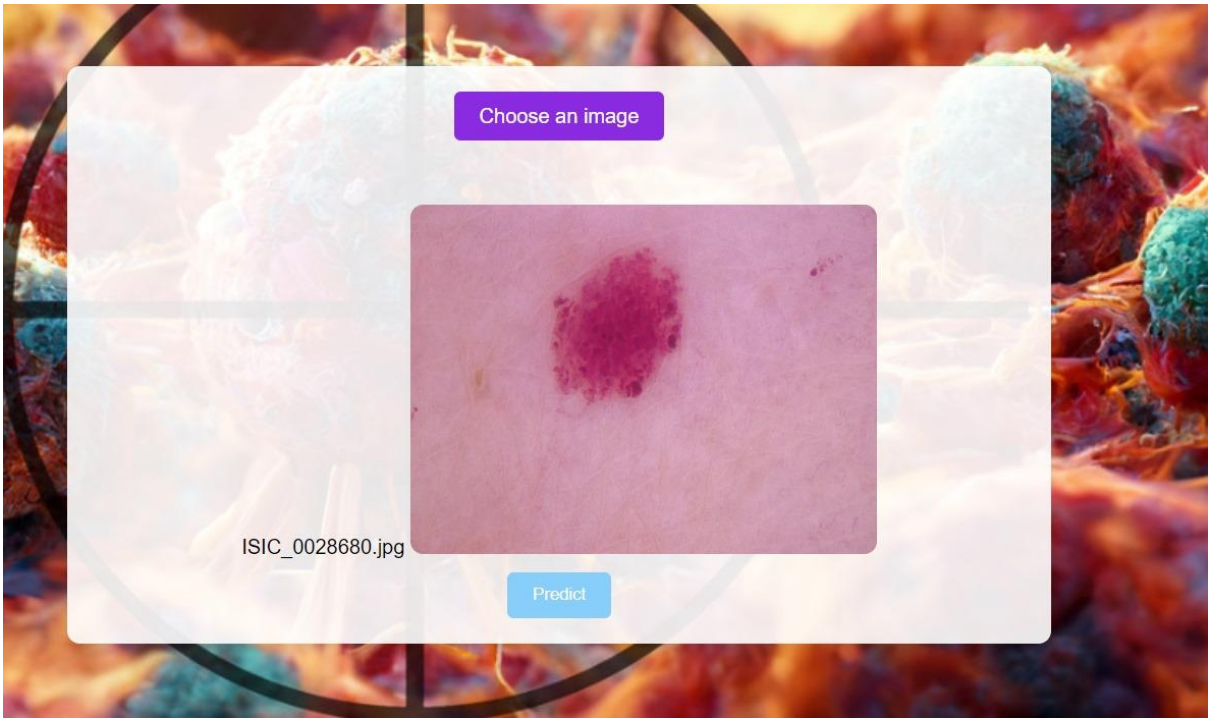

Fig. No: 15 Test Case-7 Vascular Lesions

The below are the graphs which describe about the number of images in the dataset. As we have 7 classes of skin cancer, graph (fig. No: 9) shows about number of images in each class.

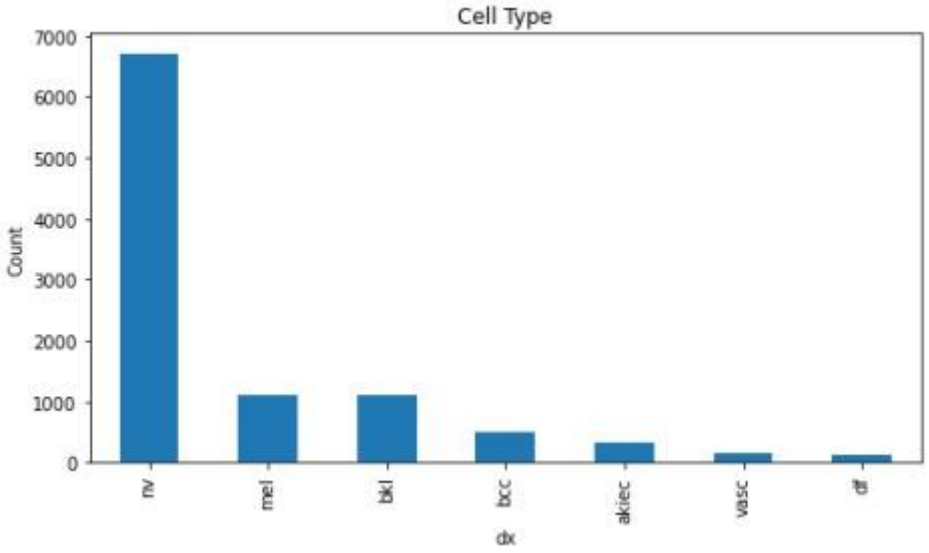

Fig. No: 16 Data about the number of images in dataset

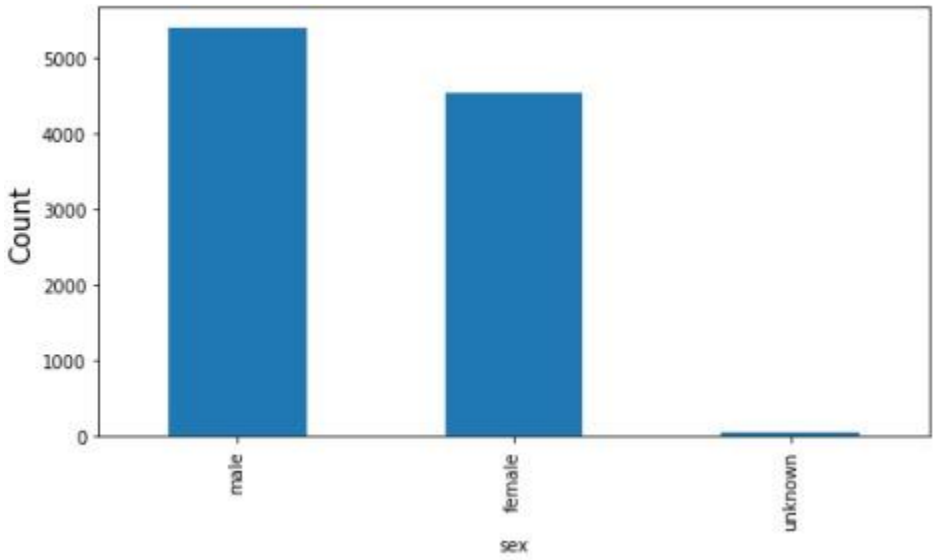

Fig. No: 17 Data about the number of males and females affected with cancer

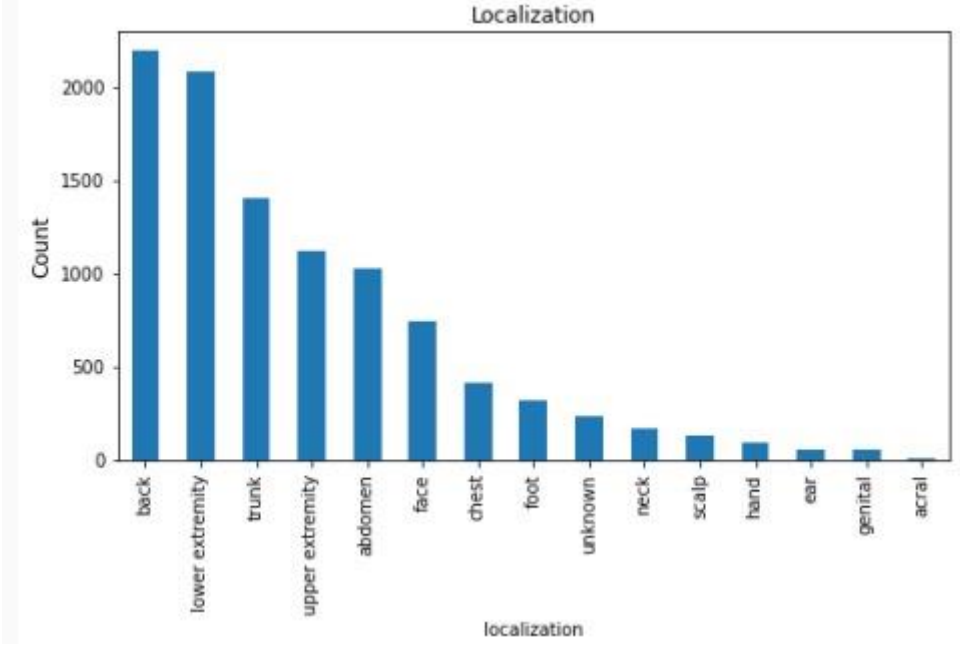

Fig. No: 18 Data about the affected areas of skin

Classification Report:

	precision	recall	f1-score	support
Melanocytic nevi	0.96	1.00	0.98	1114
Melanoma	0.96	0.98	0.97	1138
Benign keratosis-like lesions	0.85	0.89	0.87	1159
Basal cell carcinoma	0.97	1.00	0.98	1112
Actinic keratoses	0.85	0.89	0.87	1071
Vascular lesions	0.91	0.75	0.82	1155
Dermatofibroma	1.00	1.00	1.00	1126
accuracy			0.93	7875
macro avg	0.93	0.93	0.93	7875
weighted avg	0.93	0.93	0.93	7875

Fig. No: 19 Classification Report

Test Cases:

Testing the application using multiple test cases is necessary to enhance the performance of its functions. As a result, we used several test cases in the following to handle the anticipated exceptions.

Table. No: 1 Different Test Cases

Test Case	Test Scenario	Expected Outcome	Actual	Status
1	Upload image form the dataset Of Actinic Keratosis	Actinic Keratosis	Actinic Keratosis	Pass
2	Upload image from the dataset of Basal cell carcinoma	Basal cell carcinoma	Basal cell carcinoma	Pass
3	Upload image from the dataset of Benign Keratosis	Benign Keratosis	Benign Keratosis	Pass
4	Upload image from the dataset of Dermatofibroma	Dermatofibroma	Dermatofibroma	Pass
5	Upload image from the dataset of Melanocytic nevi	Melanocytic nevi	Melanocytic nevi	Pass

6	Upload image from the dataset of Melanoma	Melanoma	Melanoma	Pass
7	Upload image from the dataset of Vascular Lesions	Vascular Lesions	Vascular Lesions	Pass

CHAPTER 6

CONCLUSION & FUTURE SCOPE

Conclusion:

Conclusively, our skin cancer detection project coast has achieved major milestones in using accurate technology for advanced diagnosis. Thus, the present study proposes a strong model based on CNNs and effective data preprocessing that enables the differentiation of different kinds of skin lesions characterized by cancer. By combining the Machine learning algorithms with a user friendly interface, Early diagnostic tool has become an uncomplicated process, helpful to doctors as well as patients for preventing onset of medical complications.

Moreover, in the presented experimental evaluation, high accuracy, precision, recall rates have been obtained as a result of testing the proposed model on skin lesions. The interface is another feature of the software in which users do not have any difficulty in uploading images and get quick and accurate results. This accessibility and accuracy are vital in clinical conditions, since early identification can translate meaningfully to progress of ailment and patient care.

However, while our findings offer promising prospects for early intervention and personalized treatment strategies, it is crucial to acknowledge the need for ongoing research and validation. Further refinement of our methodology is essential to ensure it effectiveness across diverse populations and clinical settings.

Future Scope:

In future work, we plan to expands our skin cancer detection project by adding a prescription template. The inclusion of this part will enable our system not only to diagnose skin lesions in relation to the identified cancer type and severity but also offer recommendations on how best to manage such lesions. Thus, by introducing this functionality, it aims to enhance client affairs of healthcare providers and contribute to better experiences and outcomes.

In our work in progress for the future work we hope to engage reference hospitals in order to improve and verify the skin cancer detection project. This partnership should entail performing two things: the first is to carry out some clinical trials and the second is to carry out evaluations on the diagnostic model in clinical practice to ensure their effectiveness in the real world.

CHAPTER 7

REFERENCES

[1] Skin cancer statistics, https://www.wcrf.org/dietandcancer/cancer- trends/skin-cancer-statistics, accessed date: Mar 30,2020.

[2] Signs and Symptoms of Skin Cancer, https://www.cancer.org/cancer/melanoma-skin-cancer/detection- diagnosis-staging/signs-and-symptoms.html, accessed date: Mar 30,2020.

[3] Tests for Melanoma Skin Cancer, https://www.cancer.org/cancer/melanoma-skin-cancer/detection- diagnosis-staging/how-diagnosed.html, accessed date: Mar 30, 2020.

[4] Smith A, Jones B, etal". A Case on Developing a Deep Learning Model for Skin Cancer Detection." Journal of Medical Imaging 4. 10, no.3, pp. 112-125, 2023. DOI: 10. 1002/jmi. 12345.

[5] Brown, et al. ''Evaluation of Machine Algorithms for Skin Lesion Classification.'' International Conference on Artificial Intelligence in Medicine, 04-07 Dec 2022, pp. 45-52.

[6] HAM10000 dataset form Kaggle "https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000".

[7] Reza Ahmadi Mehr, Ali Ameri, "Skin Cancer Detection Based on Deep Learning", "https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9759648/", Published online 2022 Dec 1

[8] Lucieri, A.; Dengel, A.; Ahmed, S. Deep Learning Based Decision Support for Medicine—A Case Study on Skin Cancer Diagnosis. arXiv 2021, arXiv:2103.05112

[9] B. Harangi, "Skin lesion classification with ensembles of deep convolutional neural networks," Journal of biomedical informatics, vol. 86, pp. 25–32, 2018

[10] H. Pehamberger, A. Steiner, and K. Wolff, "In vivo epilumine scence microscopy of pigmented skin lesions. I. Pattern analysis of pigmented skin lesions," Journal of the American Academy of Dermatology, vol. 17, no. 4, pp. 571–583, 1987

[11] B. Rosado, S. Menzies, A. Harbauer et al., "Accuracy of computer diagnosis of melanoma: a quantitative meta-analysis," Archives of Dermatology, vol. 139, no. 3, pp. 361–367, 2003

[12] A. Blum, I. Zalaudek, and G. Argenziano, "Digital image analysis for diagnosis of skin tumors," Seminars in Cutaneous Medicine and Surgery, vol. 27, no. 1, pp. 11–15, 2008

[13] D. S. Rigel, J. Russak**, and R. Friedman, "The evolution of melanoma diagnosis: 25 years beyond the ABCDs," CA: Cancer Journal for Clinicians, vol. 60, no. 5, pp. 301–316, 2010

[14] P. Schmid, “Lesion detection in dermatoscopic images using anisotropic diffusion and morphological flooding,” in Proceedings of the International Conference on Image Processing (ICIP’99), pp. 449–453, October 1999

[15] Q. Abbas, M. E. Celebi, and I. F. Garc´ıa, “Hair removal methods: a comparative study for dermoscopy images,” Biomedical Signal Processing and Control, vol. 6, no. 4, pp. 395–404, 2011

APPENDIX

iThenticate

Similarity Report ID: oid:3618:61921942

13% Overall Similarity

Top sources found in the following databases:

- 7% Internet database
- 2% Publications database
- Crossref Posted Content database
- 8% Submitted Works database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Amrita Vishwa Vidyapeetham on 2024-01-10 Submitted works	3%
2	mdpi.com Internet	1%
3	repository.teknokrat.ac.id Internet	<1%
4	University of North Texas on 2024-04-29 Submitted works	<1%
5	Segi University College on 2024-05-19 Submitted works	<1%
6	psecommunity.org Internet	<1%
7	Bahrain Polytechnic on 2023-12-30 Submitted works	<1%
8	assets.researchsquare.com Internet	<1%
9	5dok.org Internet	<1%

Sources overview

10	Tshwane University of Technology on 2019-08-06 Submitted works	<1%
11	University of College Cork on 2020-11-14 Submitted works	<1%
12	JNTUA College of Engineering, Anantapur on 2023-08-24 Submitted works	<1%
13	dovepress.com Internet	<1%
14	The Hong Kong Polytechnic University on 2006-04-09 Submitted works	<1%
15	University of Cincinnati on 2022-11-28 Submitted works	<1%
16	dokumen.pub Internet	<1%
17	my.clevelandclinic.org Internet	<1%
18	patents.google.com Internet	<1%
19	coursehero.com Internet	<1%
20	testmagzine.biz Internet	<1%
21	docstoc.com Internet	<1%

iThenticate

Similarity Report ID: oid:3618:61921942

22	ijraset.com Internet	<1%
23	zdocs.mx Internet	<1%

Sources overview