Skin Lesion Classification Using Deep Learning

An Application of CNN for Multi-class Skin Disease Prediction

About Skin Lesions

- What is a Skin Lesion ?

A skin lesion is any abnormal change or growth on the skin, such as moles, sores, or discolorations. Lesions are categorized into benign (non-cancerous) or malignant (cancerous) types.

- Causes of Skin Lesions :

-UV Radiation: Sun or tanning bed exposure damages skin cells, increasing cancer risk.

-Genetics: Family history raises the risk of skin lesions, especially cancer.

-Inflammation/Infections: Conditions like acne or eczema cause lesions through ongoing irritation.

Project Overview

- The Problem :

Accurately diagnosing skin cancer is challenging due to the need to distinguish between various types of lesions. Misdiagnosis can lead to treatment delays and negatively impact patient outcomes.

- Why We Choose This Project :

We chose this project to assist doctors in quickly and accurately diagnosing skin lesions.

Dataset Overview: Skin Cancer MNIST: HAM10000

- HAM10000: Human Against Machine with 10,000 Dermatoscopic Images.
- The dataset consists of images and associated with csv text file . We use these modalities together in a multi-class classification model .
- The dataset contains the following columns:

  -lesion_id : A unique identifier for each lesion.

  -image_id : The corresponding ID for each image in the dataset.

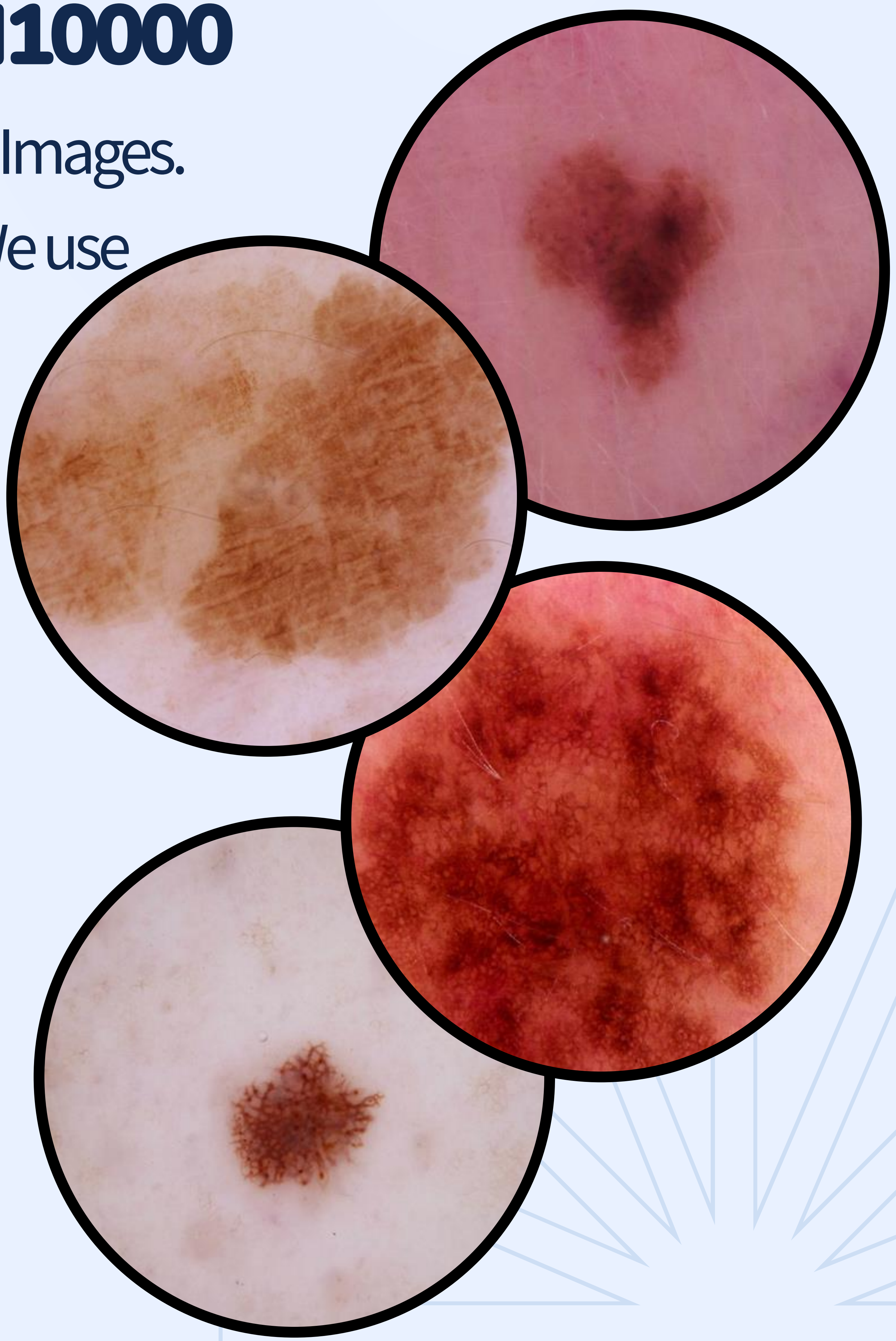
  -dx : The diagnosis label for the lesion (e.g., bkl, nv, mel, etc.).

  -dx_type : The method used to confirm the diagnosis (e.g.,histopathology, follow-up).

  -age : The age of the patient.

  -sex : The gender of the patient (male or female).

  -localization : The anatomical location of the lesion on the body (e.g., scalp, back, etc.).



Dataset Overview: Skin Cancer MNIST: HAM10000

- The dataset contains labeled skin lesion images for 7 different classes :

  -bkl (Benign Keratosis-like lesions)

  -nv (Melanocytic Nevi)

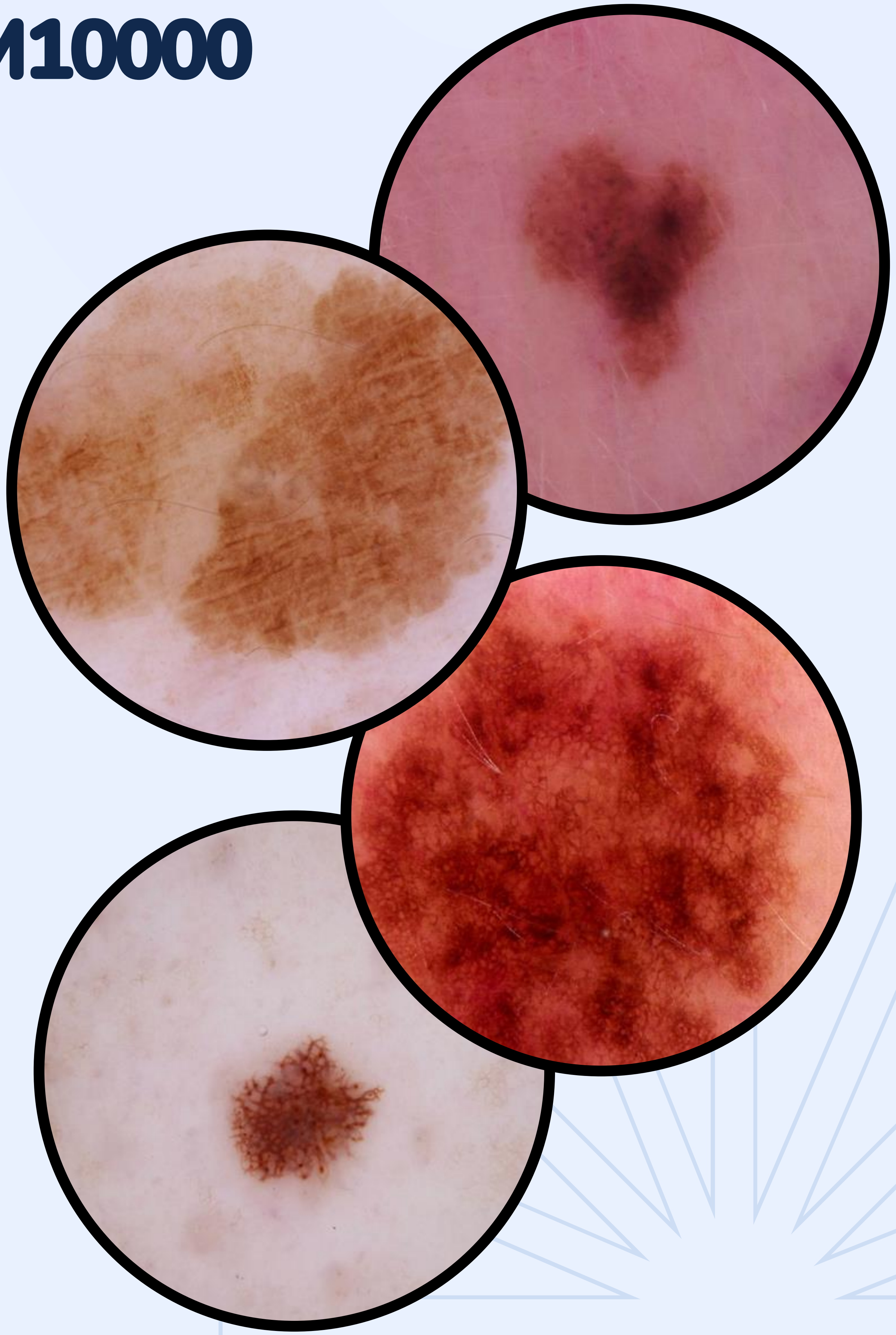
  -df (Dermatofibroma)

  -mel (Melanoma)

  -vasc (Vascular Lesions)

  -bcc (Basal Cell Carcinoma)

  -akiec (Actinic Keratoses and Intraepithelial Carcinoma)



Essential Phases of the Workflow

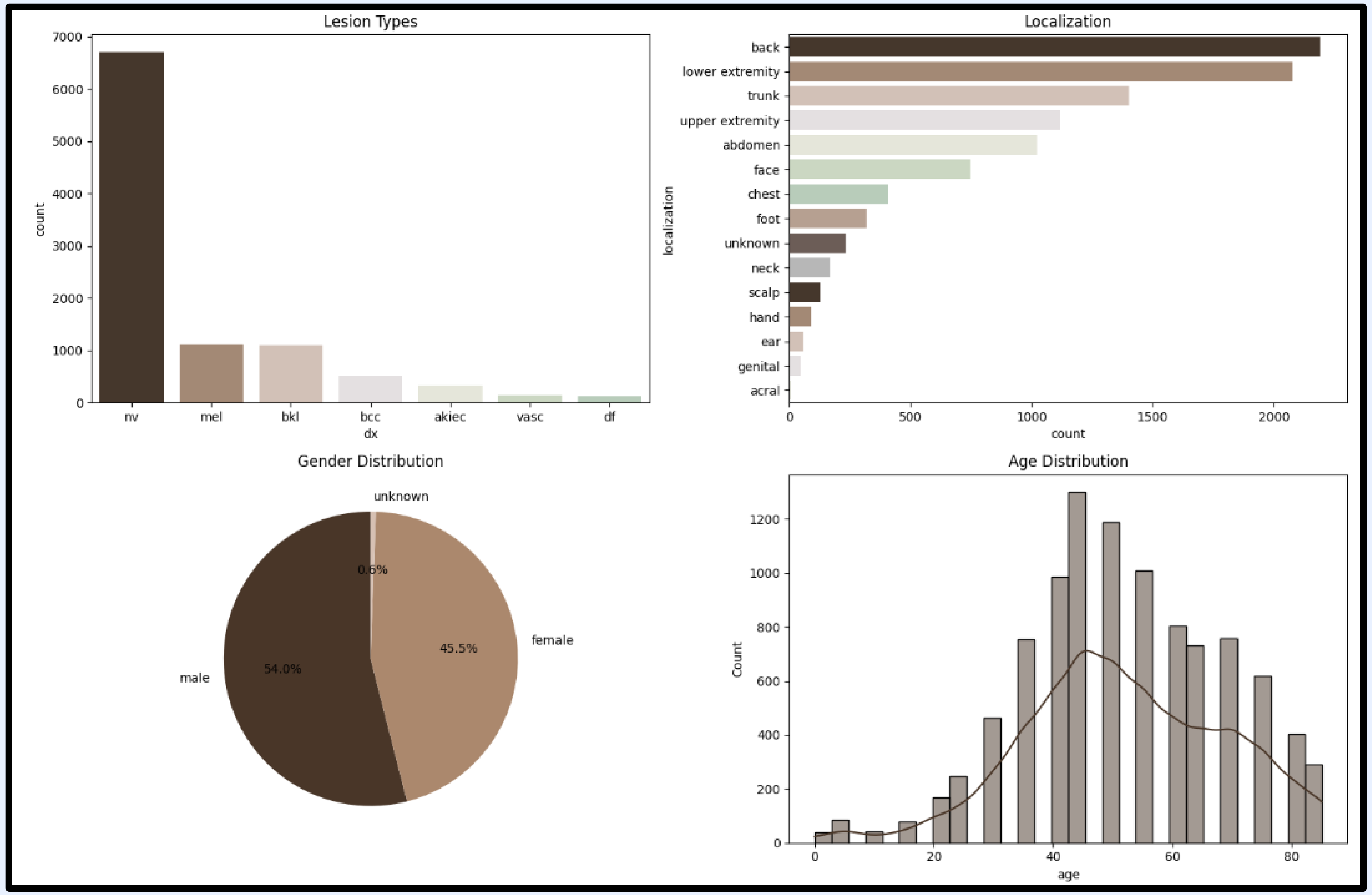
- Visualization
- PreProcessing
- Model Architecture
- Evaluation

Visualization

visualization of data

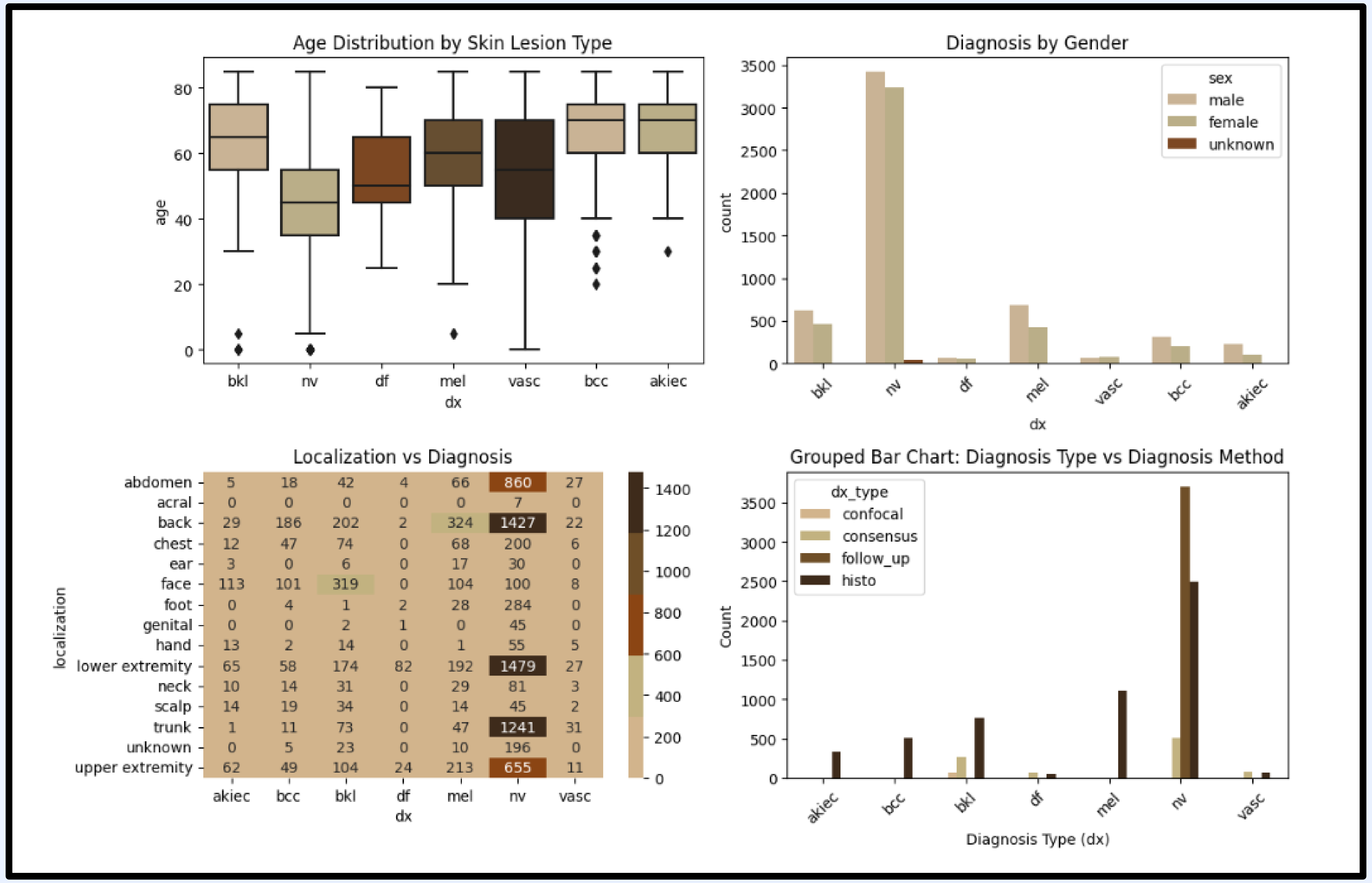
At first ,We visualized relationships representing the distribution of various dataset features, such as :

- lesion types count using bar chart .
- lesion locations count using horizontal bar chart .
- gender count using pie chart .
- age distribution using histogram and density curve .



Next, we explored relationships between various features to uncover patterns, such as:

- how age varies across lesion types using poxplot
- gender distribution for each diagnosis using countplot
- lesion localization trends using heatmap .
- the diagnostic methods used for each lesion type using grouped bar chart.



localization	akiec	bcc	bkl	df	mel	nv	vasc
abdomen	5	18	42	4	66	860	27
acral	0	0	0	0	0	7	0
back	29	186	202	2	324	1427	22
chest	12	47	74	0	68	200	6
ear	3	0	6	0	17	30	0
face	113	101	319	0	104	100	8
foot	0	4	1	2	28	284	0
genital	0	0	2	1	0	45	0
hand	13	2	14	0	1	55	5
lower extremity	65	58	174	82	192	1479	27
neck	10	14	31	0	29	81	3
scalp	14	19	34	0	14	45	2
trunk	1	11	73	0	47	1241	31
unknown	0	5	23	0	10	196	0
upper extremity	62	49	104	24	213	655	11

After visualization for image and text data

we know that

- **Data is imbalanced**
- **Find null value**
- **Find unnecessary column**
- **Find categorical columns**

these all will solved in preprocessing stage

Preprocessing

Preprocess on images data

Preprocess on text data

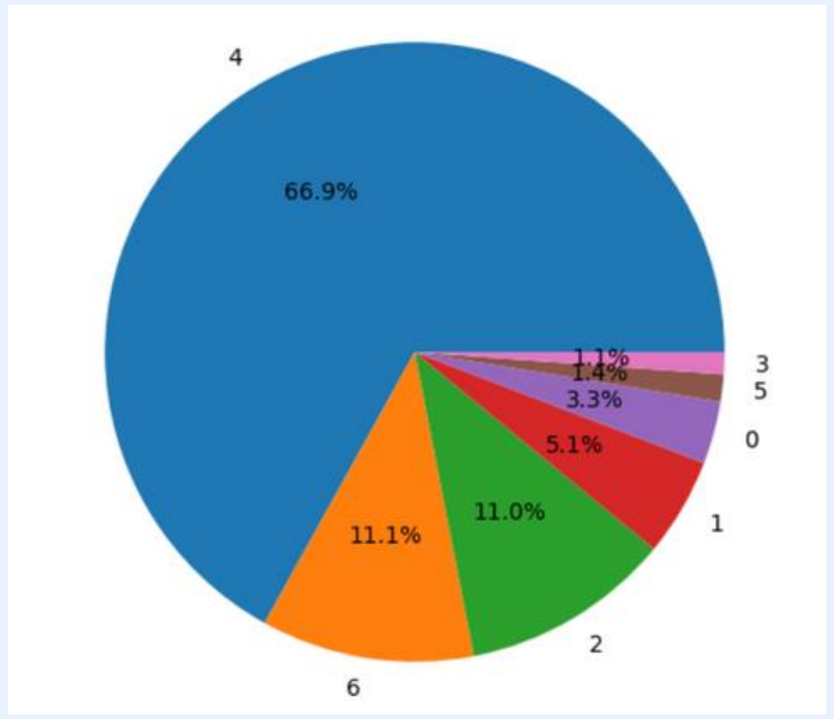
Preprocessing on images:

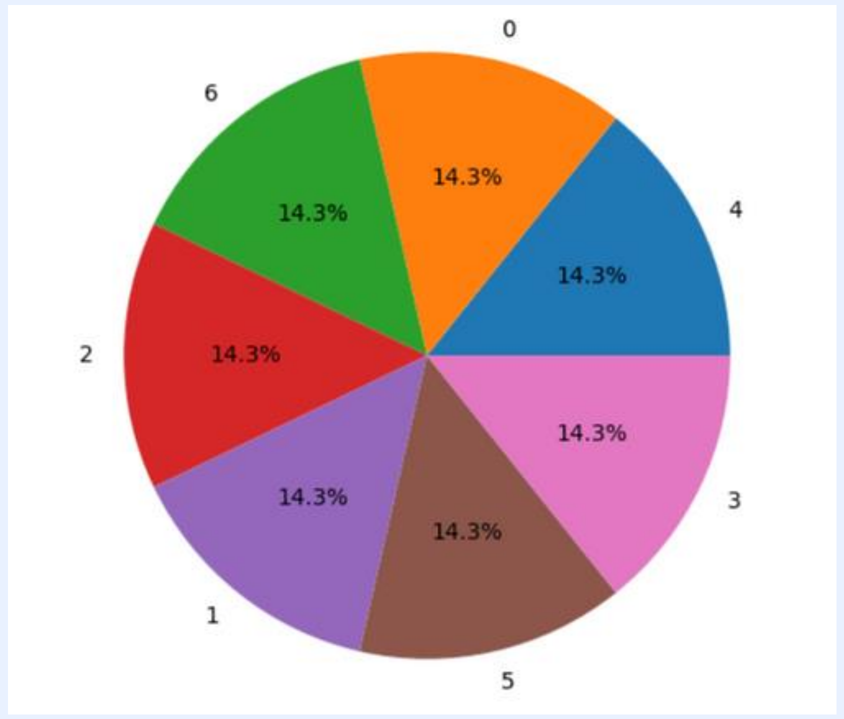
Normalization

scaling data to a specific range [0, 1], to improve model performance by ensuring consistent feature magnitudes.

Augmentation

to solve Imbalanced image data problem





Preprocessing on Table

- Drop Unnecessary Columns
- Handel category columns
- Handel Imbalance
- Handel Missing value
- scale

Model

Model Architecture

- Model on images:

A Convolutional Neural Network (CNN) is used to classify skin lesions into 7 categories. The model includes:

- Input Shape: (28, 28, 3).

- Convolutional Blocks: Three blocks with increasing filters (32, 64), ReLU activation, MaxPooling, and Dropout for regularization.

- Fully Connected Layers: Two dense layers (64, 128 units) and ReLU activation.

- Model on Table :

- input shape: (4,).

- Fully connected layer (16) with ReLU activatioin .

Model Architecture

- Combine two models:
  - Combine two models and enter them to another fully connected layers.
  - 5 Fully connected layers (128 , 64 ,32 , 16 ,7) with ReLU and SoftMax activation for classification output.

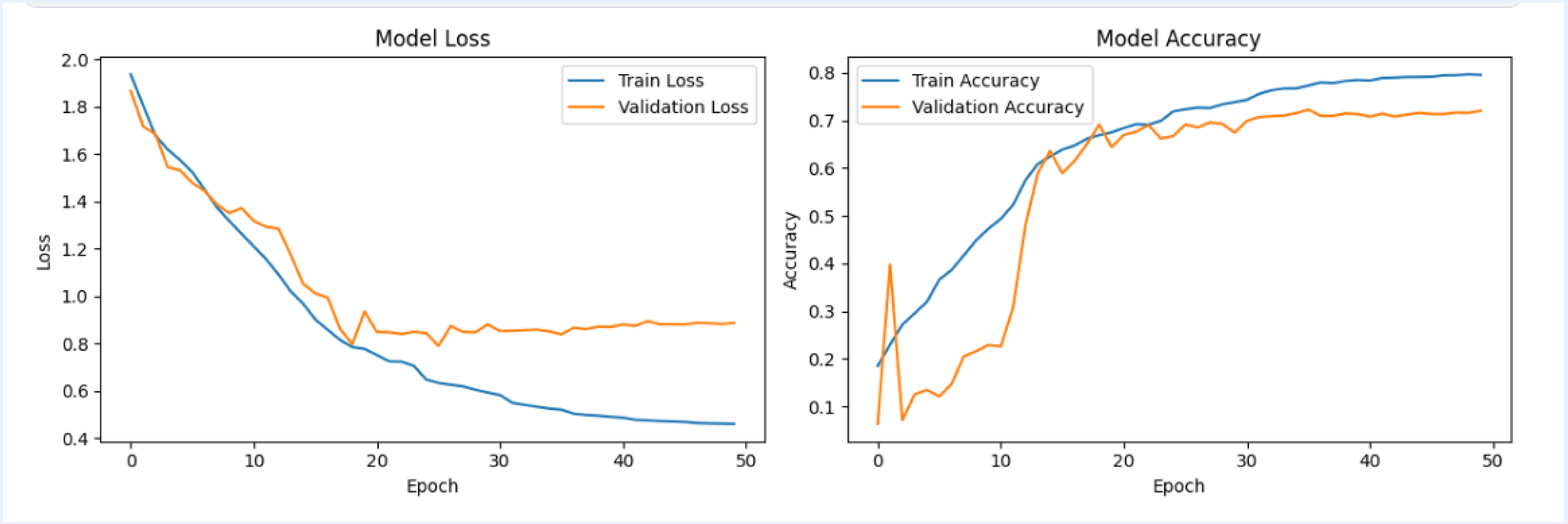
- Model training :
  - Loss Function: Sparse Categorical Crossentropy.
  - Optimizer: Adam
  - Callbacks: EarlyStopping, ModelCheckpoint, and ReduceLROnPlateau for efficient training.

Evaluation

Model Performance Analysis

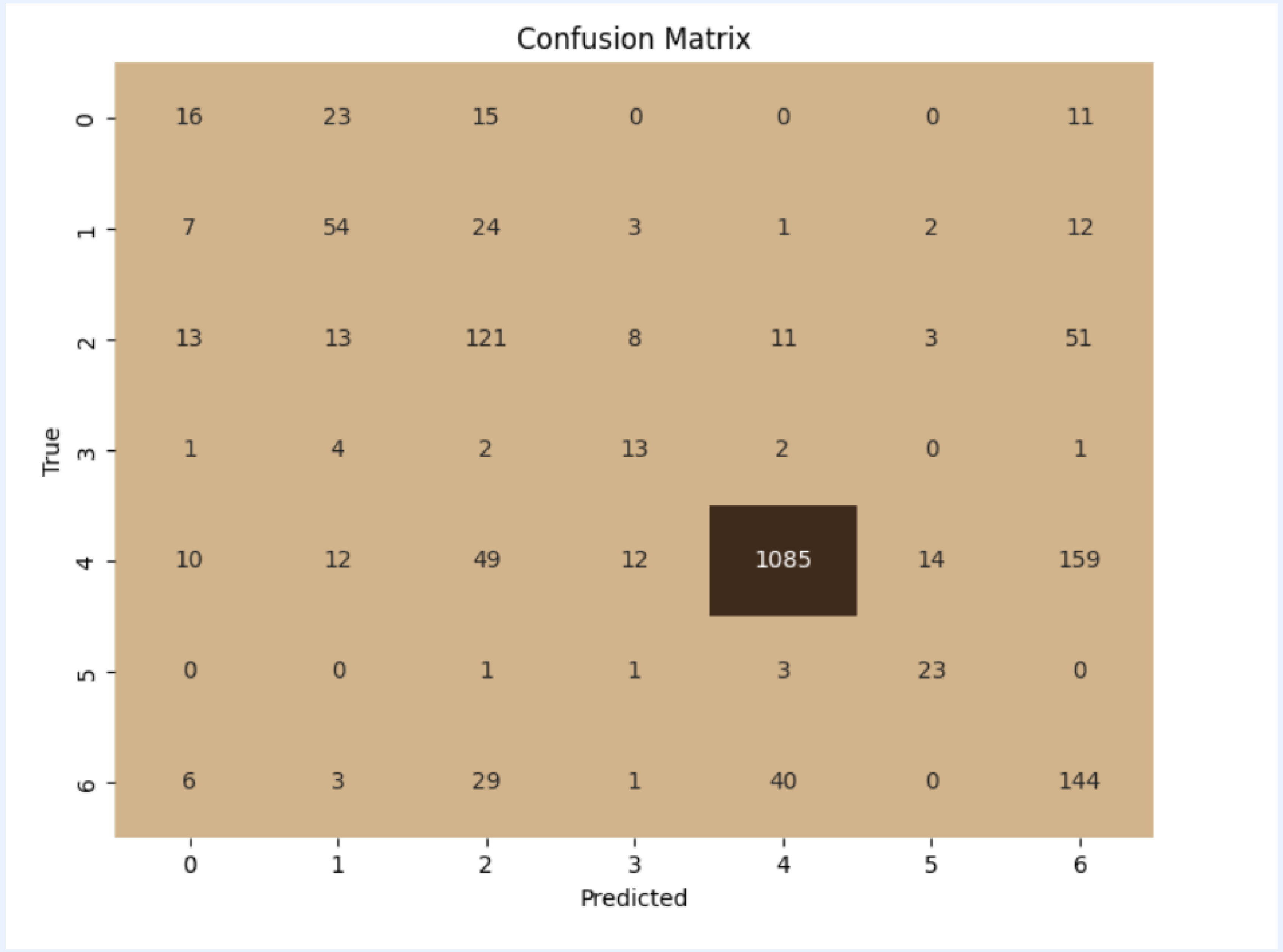
- Both Training Accuracy and Validation Accuracy improve over time, demonstrating that the model captures meaningful features from the data.

- Validation Accuracy stabilizes at a satisfactory level, showing the model's ability to generalize well to unseen data.



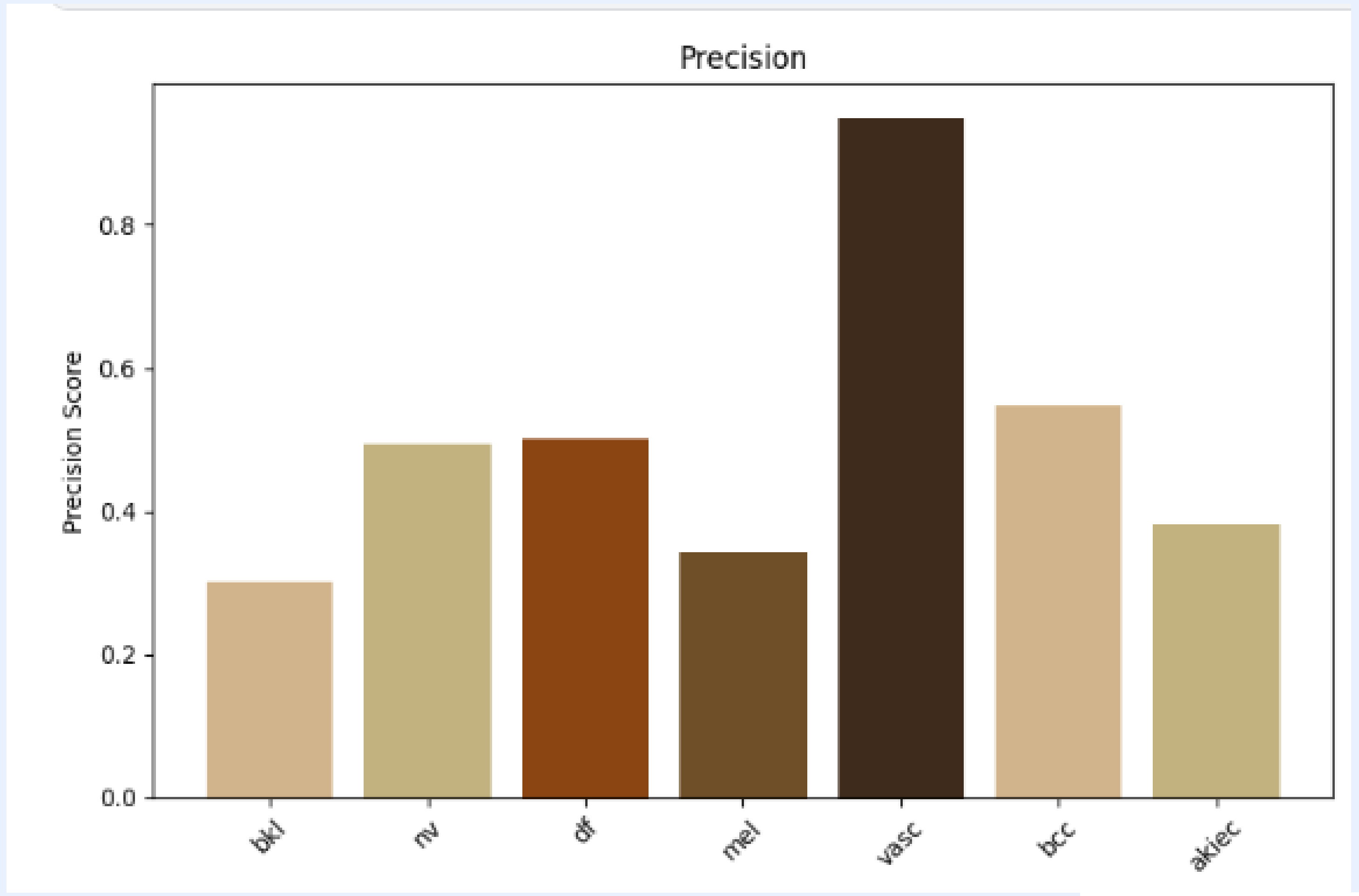
Model Performance Analysis : Confusion Matrix

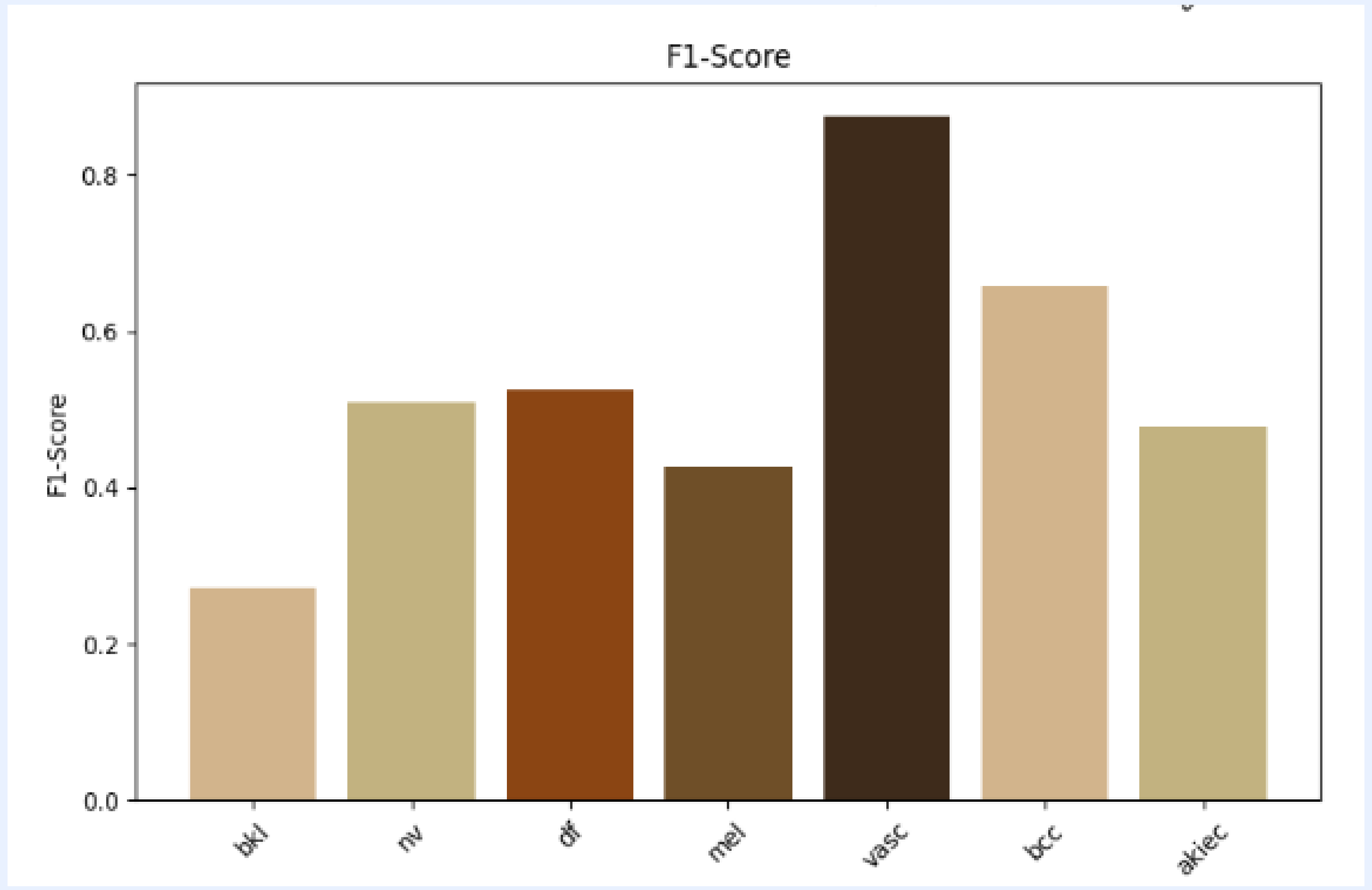
- The diagonal values in the confusion matrix represent the number of correct predictions for each class.
- A large proportion of predictions fall along the diagonal, showing that the model performs well in classifying most categories correctly.

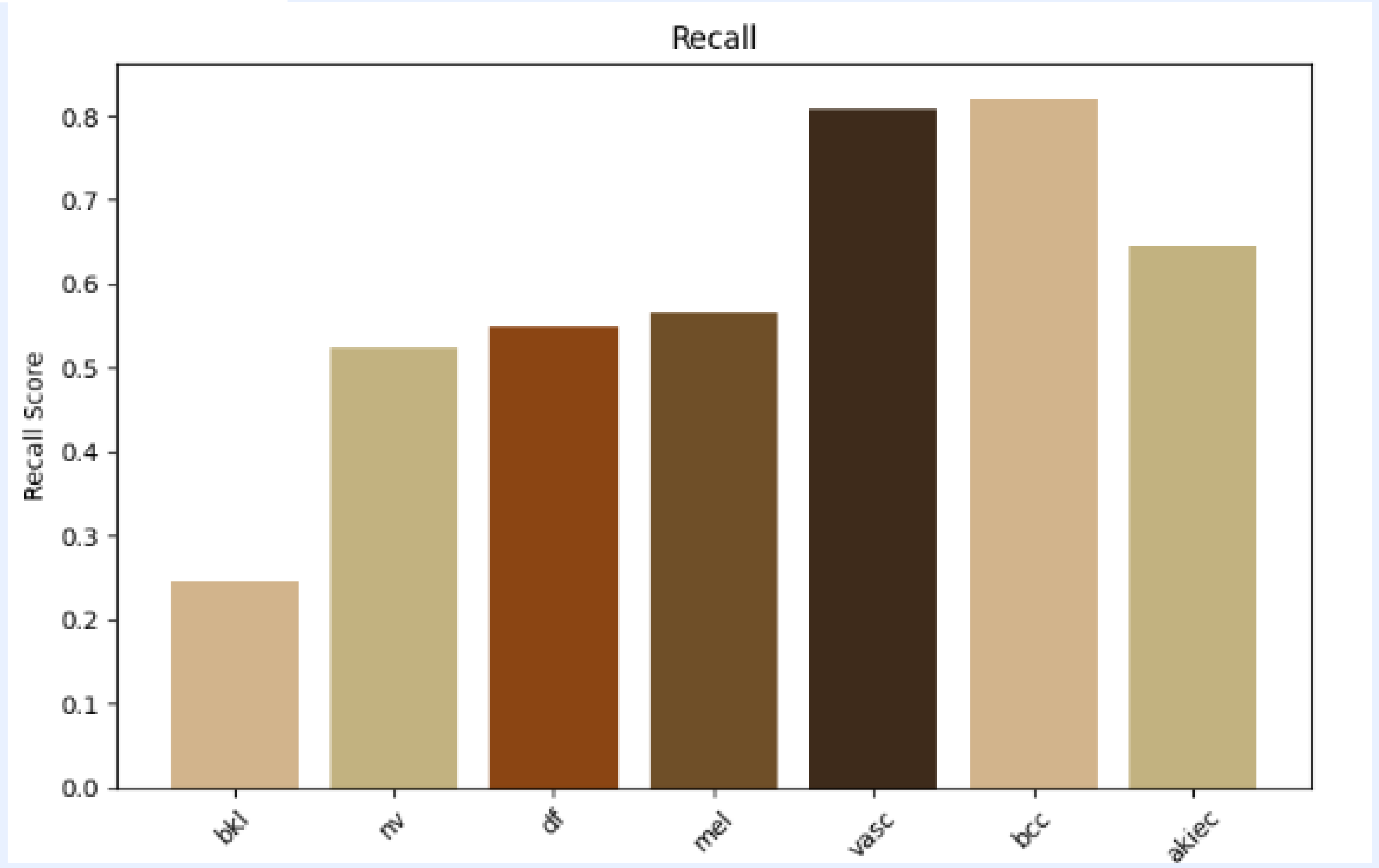


True \ Predicted	0	1	2	3	4	5	6
0	16	23	15	0	0	0	11
1	7	54	24	3	1	2	12
2	13	13	121	8	11	3	51
3	1	4	2	13	2	0	1
4	10	12	49	12	1085	14	159
5	0	0	1	1	3	23	0
6	6	3	29	1	40	0	144

Model Performance Analysis : another metrics





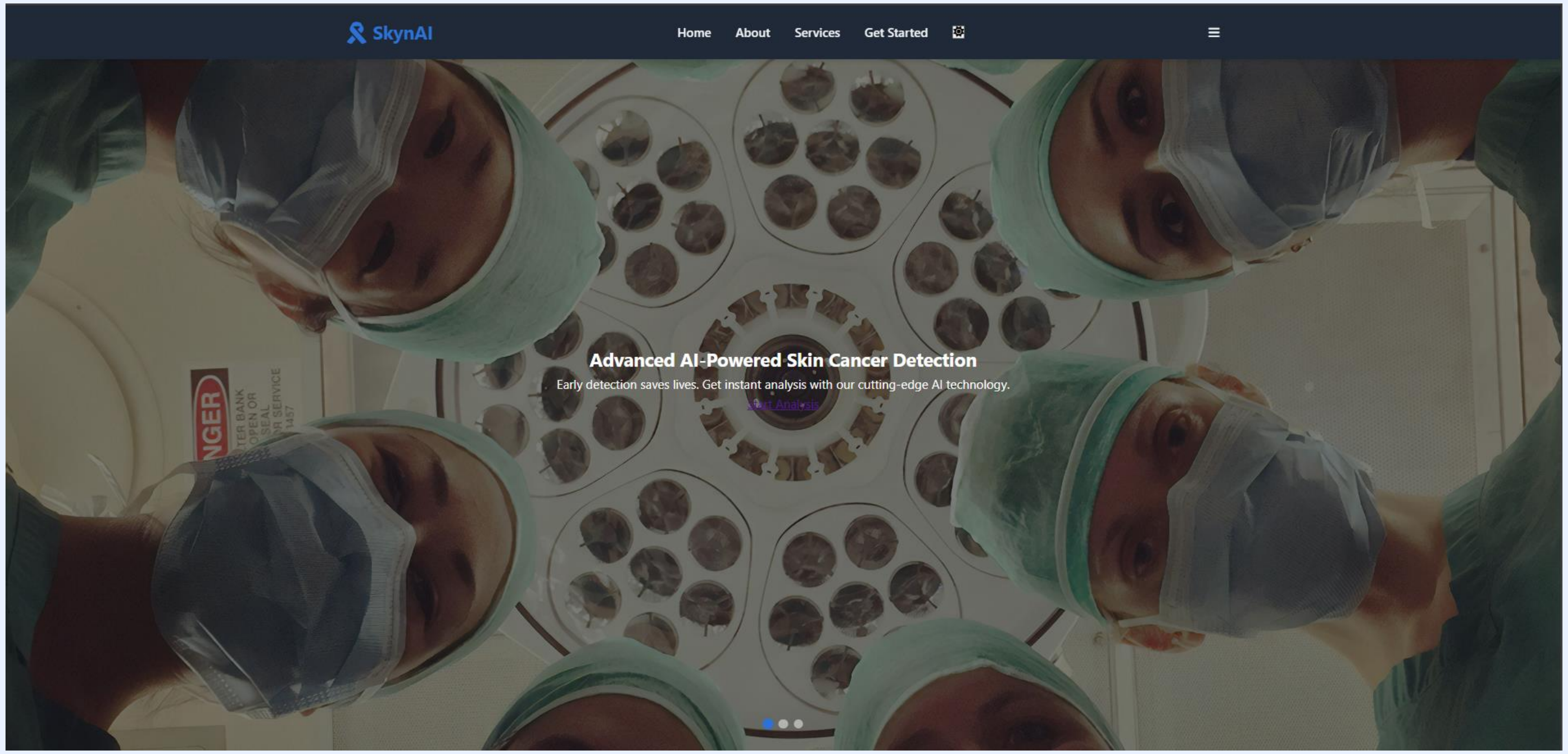


Deployment

SkynAI is a web application that helps detect potential skin cancer through image analysis

🏗️ Project Structure

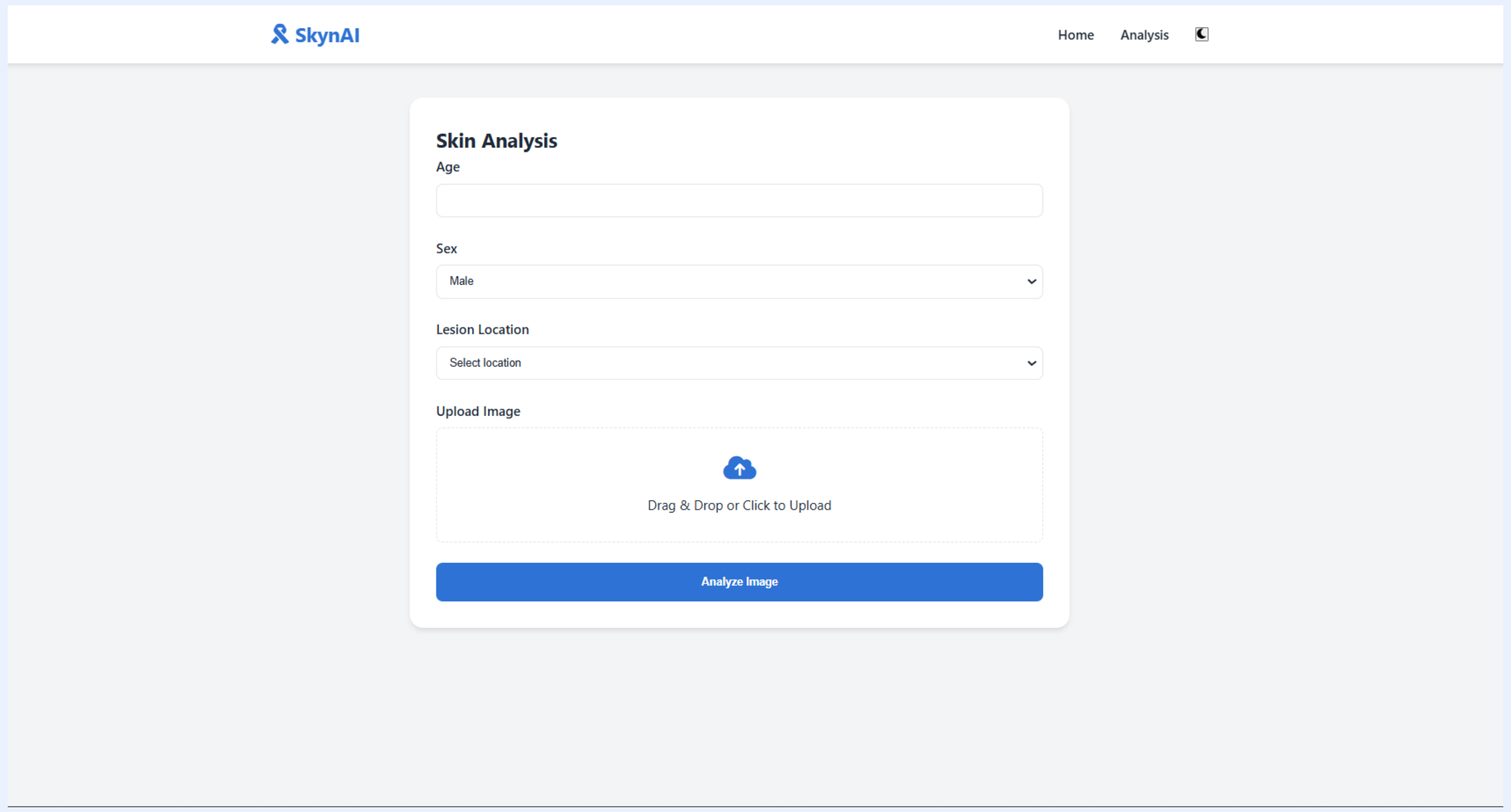
```
skynai/
│
├── app.py              # Main Flask application
├── static/
│   ├── js/
│   │   ├── prediction.js  # Handle predictions
│   │   └── theme.js       # Dark/light theme toggle
│   └── styles/            #css files
├── templates/
│   ├── index.html        # Landing page
│   └── predict.html      # Prediction interface
└── model.h5            # Trained ML model
```



Advanced AI-Powered Skin Cancer Detection

Early detection saves lives. Get instant analysis with our cutting-edge AI technology.

“Enter your information ,upload your skin image and get the diagonal."



Skin Analysis

Age

Sex

Male

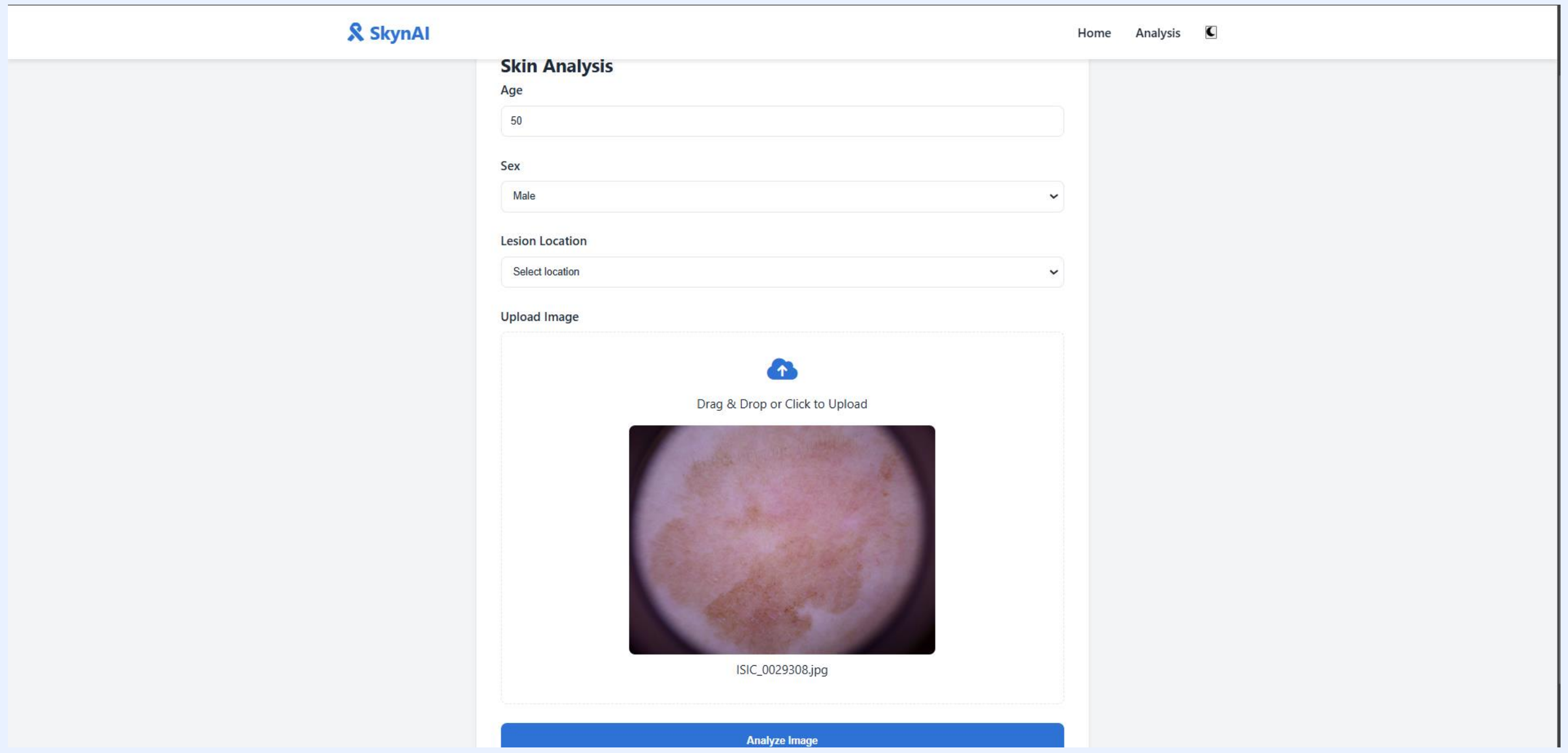
Lesion Location

Select location

Upload Image

Drag & Drop or Click to Upload

Analyze Image



Skin Analysis

Age

50

Sex

Male

Lesion Location

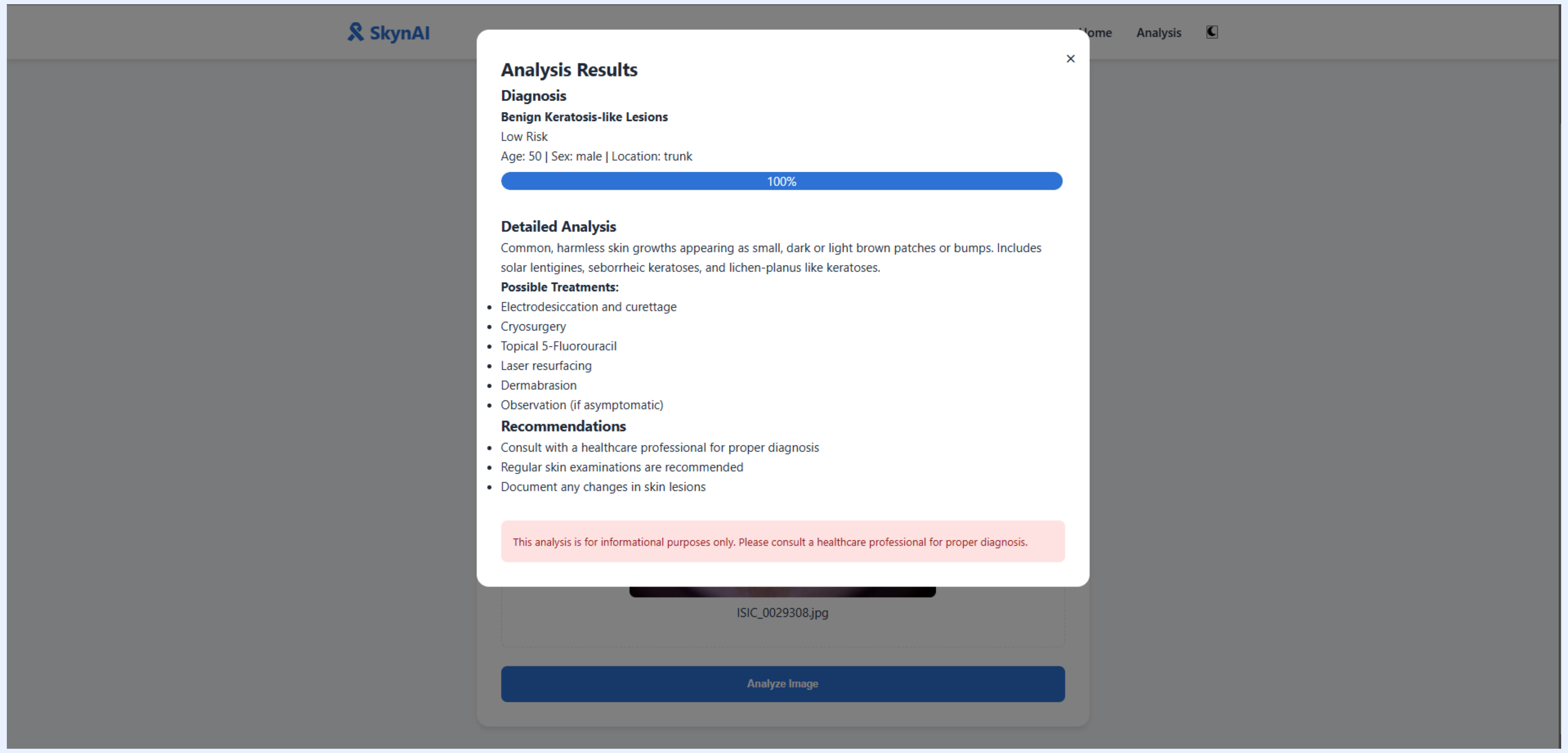
Select location

Upload Image

Drag & Drop or Click to Upload

ISIC_0029308.jpg

Analyze Image



Analysis Results

Diagnosis

Benign Keratosis-like Lesions

Low Risk

Age: 50 | Sex: male | Location: trunk

100%

Detailed Analysis

Common, harmless skin growths appearing as small, dark or light brown patches or bumps. Includes solar lentigines, seborrheic keratoses, and lichen-planus like keratoses.

Possible Treatments:

- Electrodesiccation and curettage
- Cryosurgery
- Topical 5-Fluorouracil
- Laser resurfacing
- Dermabrasion
- Observation (if asymptomatic)

Recommendations

- Consult with a healthcare professional for proper diagnosis
- Regular skin examinations are recommended
- Document any changes in skin lesions

This analysis is for informational purposes only. Please consult a healthcare professional for proper diagnosis.

ISIC_0029308.jpg

Analyze Image

Running

Go to the Repo via this link: https://github.com/SalmaElgezawy/Skin_Cancer_classification

01. Download ZIP

- From the Code button: press **Download ZIP.**
- Unzip the file.
- run the **requirement.txt** to ensure the required libraries is installed.
- Run the code

02. Using Git

- git clone https://github.com/SalmaElgezawy/Skin_Cancer_classification.git
- cd repository directory
- pip install -r requirements.txt python main.py
- Run the code
- Run the Falsk app with python app.py

THANK YOU