AI TOOL FOR DERMATOLOGICAL MANIFESTATIONS

A MINI PROJECT- II REPORT

Submitted by

KISORE D	**(71812111022)**
NAVEEN KUMAR M	**(71812111037)**
VISHAL KUMAR S	**(71812111057)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SRI RAMAKRISHNA ENGINEERING COLLEGE

[Educational Service: SNR Sons Charitable Trust]
[Autonomous Institution, Reaccredited by NAAC with ‘A+’ Grade]
[Approved by AICTE and Permanently Affiliated to Anna University, Chennai]
[ISO 9001:2015 Certified and All Eligible Programmes Accredited by NBA]
Vattamalaipalayam, N.G.G.O. Colony Post,

COIMBATORE – 641 022

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2024

ANNA UNIVERSITY :: CHENNAI 60025

BONAFIDE CERTIFICATE

20AD277 - MINI PROJECT II

Certified that this mini project Report "**AI TOOL FOR DERMATOLOGICAL MANIFESTATIONS"** is the bonafide work of **"KISORE D (71812111022), NAVEEN KUMAR M (71812111037), VISHAL KUMAR S (71812111057)"** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. V. Karpagam	Mrs. K. Archana
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor,	Assistant Professor (OG),
Department of Artificial Intelligence	Department of Artificial Intelligence
and Data Science,	and Data Science,
Sri Ramakrishna Engineering College,	Sri Ramakrishna Engineering College,
Coimbatore-641022	Coimbatore-641022

Submitted for the Mini Project Viva-Voce examination held on _______________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We thank the almighty for his blessings on us to complete this project work successfully.

With profound sense of gratitude, we thank the Management, Managing Trustee, **Thiru D. Lakshminarayanasamy B. Tech., MBA,** and Joint Managing Trustee, **Thiru R. Sundar,** for provided us the necessary infrastructure required for the completion of our project.

With profound sense of gratitude, we sincerely thank the **Head of the Institution, Dr. N. R. Alamelu M.E., Ph.D.**, for her kind patronage, which helped in pursuing the project successfully.

With immense pleasure, we express our hearty thanks to the **Head of the Department**, **Dr. V. Karpagam M.E.**, **Ph.D.,** Department of Artificial Intelligence and Data Science for her encouragement towards the completion of this project.

We thank our Mini Project Coordinator **Mrs. P. V. Kavitha M.E., Assistant Professor (Sl. Gr),** Department of Artificial Intelligence and Data Science for her encouragement towards the completion of this project.

We also thank our Mini Project guide **Mrs. K. Archana M.E., Assistant Professor (OG),** Department of Artificial Intelligence and Data Science for her guidance towards the completion of the project.

We convey our thanks to all the teaching and non-teaching staff members of our department who rendered their co-operation by all means for completion of this project.

ABSTRACT

AI-driven solutions have completely changed the field of dermatology by providing powerful tools for the precise identification of skin symptoms, which is essential for prompt diagnosis and treatment. In this project, the model proposes a unique deep learning method which uses convolutional neural networks (CNNs) to categorize skin diseases from medical photos, specifically designed for dermatological manifestation identification.

By applying an EfficientNetB7 variant, which is well-known for its effective use of computational resources and strong performance in image classification tasks, the model carefully preprocessed the data (normalization and augmentation) to improve the model's generalization to various dermatological conditions. The model used a large dataset encompassing a range of skin conditions to fine-tune the pre-trained model using transfer learning techniques.

The model's performance and adaptability are shown by the testing results, which gave an accuracy of 98.67% and a comparatively low loss of 0.0409. The system's ability to accurately recognize and classify dermatological symptoms is demonstrated by its high accuracy, which also shows that it has the potential to support dermatologists in clinical situations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
AI	Artificial Intelligence
ML	Machine Learning
CNN	Convolutional Neural Network
MLP	Multi-Layer Perceptron
ISIC	International Skin Imaging Collaboration
GDPR	General Data Protection Regulation
HIPAA	Health Insurance Portability and Accountability Act
SGD	Stochastic Gradient Descent
EHR	Electronic Health Record

CHAPTER 1

INTRODUCTION

1.1DOMAIN DESCRIPTION

The term artificial intelligence, or AI for simple terms, implies the observing of human intelligence by machines, specifically computer systems. This involves developing models and algorithms that allow computers perform on operations like learning, problem-solving, analyzing natural language, identifying patterns, and making decisions that conventionally call for human intelligence. Artificial Intelligence has the potential to revolutionize the diagnosis and treatment of skin diseases in dermatology. Using machine learning algorithms, artificial intelligence can evaluate enormous volumes of data, such as pictures, patient histories, and medical publications, to help dermatologists correctly diagnose skin conditions, gauge their severity, and suggest the best courses of action.

In the field of artificial intelligence, machine learning entails developing models and algorithms that enable computers to learn from and make decisions based on data, even when they are not specifically designed to do so. Fundamentally, the goal is to create systems that can use data to learn and perform better when doing a task. Machine learning has wide-ranging applications in several areas, such as natural language processing, recommendation systems, picture and audio identification, medical diagnosis, financial forecasting, and more. It's an effective tool for automating decision-making procedures and gleaning insights from data.

Image processing is the process of analyzing and modifying digital images via the use of various algorithms and techniques. It encompasses a wide range of processes designed to enhance or extract information from images, such as segmentation, enhancement, restoration, feature extraction, pattern recognition, and more. Due to image processing technology, digital pictures may now be manipulated, analyzed, and understood in a variety of sectors. These techniques are essential for a variety of applications, from facial recognition and autonomous car navigation to medical imaging and satellite analysis.

Data science is a multidisciplinary field that brings together concepts from computer science, statistics, mathematics, and domain expertise. To find patterns and trends, exploratory analysis is conducted after raw data has been collected and cleaned. After then, machine learning techniques are used to create predictive models and derive useful information. Dermatology has been changed by data science, which is using machine learning algorithms to help diagnose skin lesions through image analysis, personalizing treatment regimens based on patient information, and enabling telemedicine platforms for remote monitoring and consultations.

1.2 PROBLEM STATEMENT

The problem statement targets the challenges of limited access to dermatologists and healthcare infrastructure, particularly in depopulated places, and aims to develop an AI-based tool for preliminary diagnosis of dermatological diseases. Conventional techniques for identifying these problems frequently require a lot of time and resources. By using image processing techniques to evaluate pictures of skin disorders, the suggested solution may quickly and accurately provide preliminary diagnosis. Because it allows for prompt intervention and treatment, this method has the potential to greatly advance the delivery of healthcare. It could also help identify several dermatological disorders early on, improving patient outcomes and lowering healthcare inequities.

1.3 OBJECTIVES

The objective is to build an enhanced AI-based tool for preliminary diagnosis of dermatological symptoms to improve both precision and speed in identifying diverse skin diseases. Using image processing techniques, the tool will analyze images of skin damage to provide a rapid and precise preliminary diagnosis. This innovative method aims to overcome the limits of established approaches and increase access to dermatological treatment, especially in rural areas with limited resources and expertise. By integrating modern artificial intelligence algorithms and machine learning with advanced picture processing, the tool hopes to improve diagnosis accuracy, resulting in better patient outcomes and improved healthcare delivery.

CHAPTER 2

LITERATURE SURVEY

2.1 Dermatologist–Level Classification Of Skin Cancer With Deep Neural Networks, August 2021

This study investigates the efficacy of deep convolutional neural networks (CNNs) in identifying skin lesions for the diagnosis of skin cancer. It used a large dataset of approximately 129,000 clinical photos depicting diverse skin conditions, outpacing prior datasets in size. The CNN was trained and evaluated to identify between keratinocyte carcinomas, benign seborrheic keratoses, malignant melanomas, and benign nevi. Performance was compared to that of 21 board-certified dermatologists, who produced equivalent outcomes. Using transfer learning and a broad dataset, the CNN demonstrated the ability to match or outperform human performance in dermatological image categorization tasks. This strategy shows promise for expanding diagnostic capabilities outside clinics, perhaps boosting early detection rates for skin cancer, a major public health problem.

2.2 Deep Learning Outperformed 136 Of 157 Dermatologists In A Head-To-Head Dermoscopic Melanoma Image Classification Task, February 2019

In this study, dermatologists in Germany with various levels of experience evaluated the performance of a convolutional neural network (CNN) trained on open-source dermoscopic pictures. With an average specificity of 86.5% and an average sensitivity of 74.1%, the CNN outperformed the majority of dermatologists in these areas. The most seasoned group, chief doctors, even showed less specificity than the CNN. Interestingly, the CNN performed better as well, with a mean sensitivity of 84.5% and a high specificity of 69.2%. These results demonstrate how deep learning algorithms may improve dermatological diagnosis accuracy, especially when it comes to identifying worrisome skin lesions. The CNN performed remarkably well by utilizing open-source data, demonstrating its value as a helpful tool for dermatologists in a range of clinical contexts. This study highlights how crucial it is to incorporate these technologies into clinical practice in order to enhance patient care and expedite the diagnostic procedure. Deep-learning algorithms are expected to play an increasingly important role in dermatology as they develop and get better. This will present chances to enhance and supplement dermatologists' knowledge in the diagnosis and treatment of skin problems.

2.3 Deep Learning in Skin Disease Image Recognition, December 2020

The study provides an extensive summary of 45 experiments that have been carried out since 2016 and have used deep learning techniques to try to identify skin disorders. It examines a variety of topics, including disease types, datasets, data processing techniques, models, frameworks, assessment measures, and model performance. There is also a study of traditional and machine learning-based methods for identifying and treating skin conditions. According to the study, deep learning methods are more effective at identifying skin conditions than dermatologists and other computer-aided therapy methods. It also highlights how beneficial it is to combine many deep learning models. It emphasizes how crucial deep learning is to enhancing the identification of skin conditions and provides potential directions for future study to further progress this field.

2.4 Artificial Intelligence-Based Image Classification for Diagnosis of Skin Cancer, December 2020

This study focuses on recent advancements in artificial intelligence which uses enabled computer-aided skin cancer diagnosis using deep learning algorithms. AI systems are still in the early stages of clinical application, despite claims that they are more accurate than dermatologists. Researchers have developed an algorithm that can distinguish between benign and malignant lesions in a range of image modalities by utilizing publicly available skin lesion information. However, there are still problems with readiness and integration of clinical process. This study emphasizes how important it is to verify and enhance AI systems so dermatologists can identify skin cancer with greater accuracy.

2.5 Dermatologist-Level Classification of Skin Cancer Using Cascaded Ensembling of Convolutional Neural Network and Handcrafted Features Based Deep Neural Network, February 2022

This study demonstrates the importance of early diagnosis in treating potentially fatal skin malignancies such as melanoma and focal cell carcinoma. It describes a unique technique to automatically collecting characteristics from skin lesion photos that makes use of deep learning architectures, notably convolutional neural networks (ConvNets). The suggested technique improves ConvNet models by using handmade features taken from a multi-layer perceptron (MLP) in a cascaded ensembled network. By integrating non-handmade picture data with handcrafted features such as color moments and texture features, the model's accuracy increases dramatically, reaching 98.3% versus 85.3% with ConvNet alone. This illustrates how combining ConvNet models with handmade features can improve skin cancer classification accuracy, allowing for more effective early identification and diagnosis of skin cancers.

2.1 EXISTING SYSTEM

EfficientNetB0 is properly implemented in the existing architecture with the Keras framework, which improves its capabilities by streamlining model building and deployment. Keras's intuitive interface and broad community assistance enable effective training, tweaking, and scaling of the system. Its use of Keras also makes it simple to integrate with other deep learning frameworks and techniques, enhancing versatility and facilitating ongoing development. The system's dedication to superior dermatological diagnostics is highlighted by the synergy between EfficientNetB0 and Keras, considered a noteworthy development in AI-driven healthcare technology.

CHAPTER 3

DESCRIPTION

3.1 PROPOSED SYSTEM

The proposed system includes several essential elements to improve dermatological diagnosis. Increasing model capacity with the EfficientNetB7 architecture makes it possible to handle high-resolution pictures and capture minute features that are essential for precise diagnosis. A sophisticated dataset enhances the model's learning process and improves performance and generalization. It is distinguished by its size, diversity, and curation. The neural network design is geared for classification tasks and consists of convolutional layers, dense layers, batch normalization, and dropout regularization. The model, which is built using the Adam optimizer and categorical cross-entropy loss, seeks to improve AI-driven healthcare solutions by achieving high accuracy in diagnosing dermatological symptoms. By integrating these components, the proposed system aims to provide a reliable dermatological diagnosis tool with improved accuracy and performance by utilizing cutting-edge architecture, efficient preprocessing methods, and high-quality data. It helps in distinguishing benign and malignant cancer.

3.2 FLOW DIAGRAM

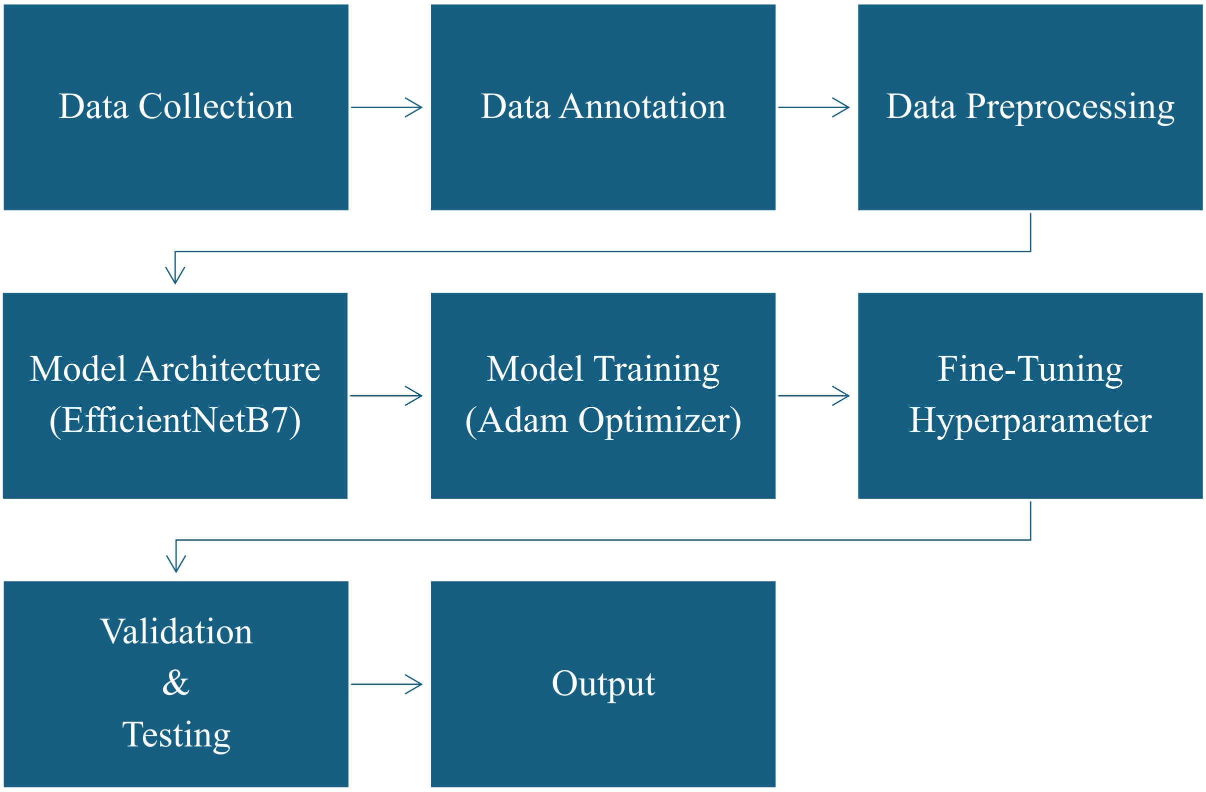

Fig 3.1 Flow Diagram

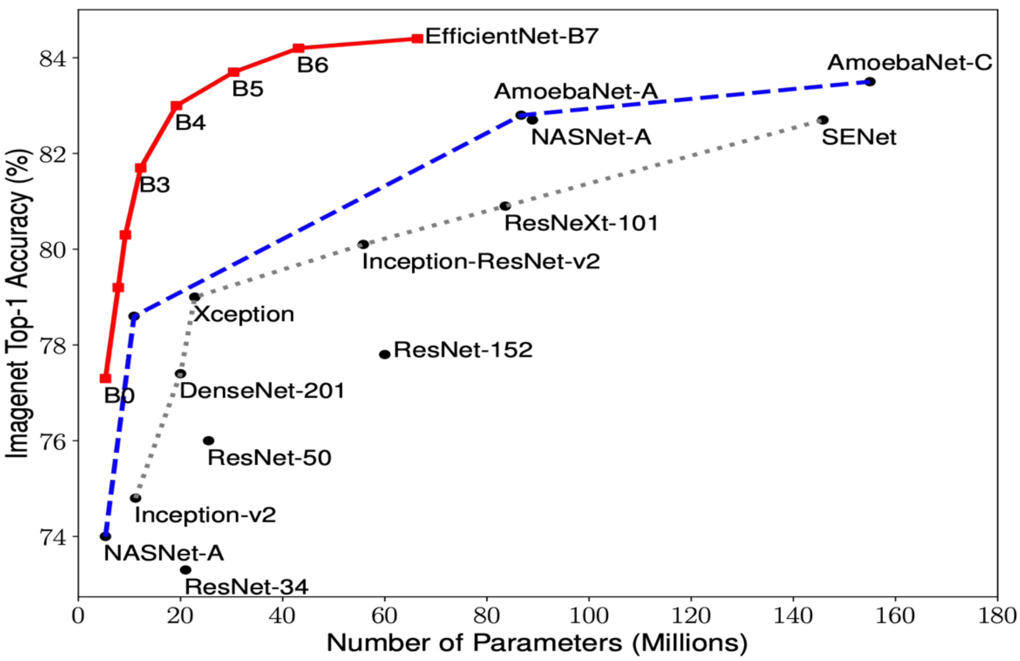

Fig 3.2 EfficientNet-B7

3.3 SOFTWARE REQUIREMENTS

3.3.1 Python 3.12

Python 3.12 improves its reputation as a top programming language for a variety of applications by introducing several new features and enhancements. The structural pattern matching syntax is one of the standout innovations, giving writers a more expressive and succinct method to work with conditional logic. It also uses a new parsing strategy based on Parsing Expression Grammar (PEG), which improves the readability and maintainability of the code. To provide advice during debugging sessions, error messages have been improved, increasing developer efficiency.Performance tweaks let some tasks run more quickly and use memory more effectively. Furthermore, new modules and enhancements are added to the standard library, expanding the features that developers may use right out of the box.

3.3.2 VS Code

Microsoft created Visual Studio Code, sometimes known as VS Code, which is a very well-liked source-code editor. It is renowned for its adaptability, effectiveness, and wide range of modification possibilities. Its many capabilities, which include debugging assistance, syntax highlighting, code completion, and a large library of extensions that enable extra functionality unique to programming languages or workflows, are well-liked by users. One of VS Code's main benefits is its combination of sophisticated functionality and lightweight design, which makes it appropriate for a variety of developers, from novices to seasoned pros. Many developers from a wide range of sectors and areas use it because of its compatibility with several programming languages and its interaction with version control systems like Git.

3.4 DATASET DESCRIPTION

The data consists of two folders with each 1800 pictures (224x244) of the two types of moles. All the rights of the Data are bound to the ISIC-Archive rights. The labels are benign and malignant.

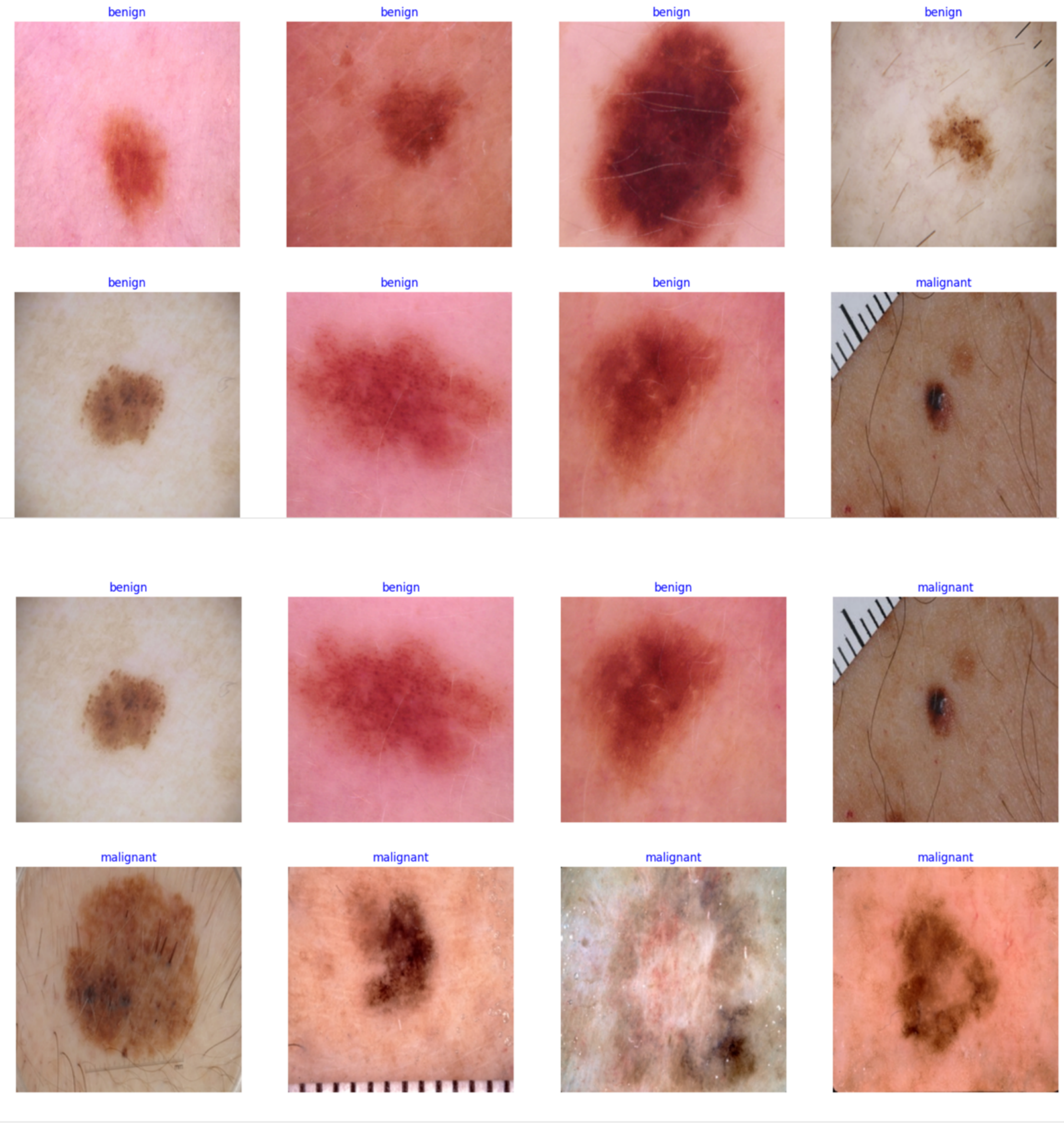

Fig 3.3 Dataset

3.5 MODULES DESCRIPTION

3.5.1 Data Collection

In dermatological AI, data collecting entails assembling a variety of superior datasets that include pictures of skin lesions. These datasets are essential for developing reliable and accurate algorithms for the identification and categorization of skin cancer. Datasets from medical archives, hospitals, research facilities, and publicly accessible repositories like the ISIC (International Skin Imaging Collaboration) Archive are often used by dermatologists and academics. Images from a variety of imaging modalities, including dermoscopy, clinical photography, and histology, are included in these databases.

To guarantee representation across various skin types, ages, and lesion kinds and improve the model's generalizability to a range of patient groups, meticulous dataset curation is necessary. To offer ground truth labels for supervised learning tasks, dermatologists or other qualified specialists may also annotate data as part of data gathering operations. Throughout the data gathering process, ethical factors—such as patient privacy and consent—are crucial to maintaining compliance with laws like the GDPR (General Data Protection Regulation) and HIPAA (Health Insurance Portability and Accountability Act).

3.5.2 Data Preprocessing

An essential first step in getting dermatological imaging data ready for model training is data preparation. To make sure the data is in an appropriate format for input into machine learning models, this procedure entails several processes. To maintain uniformity throughout the dataset and compliance with model designs, scaling the photos to a standard resolution, such as 224x224 pixels, is a frequent preprocessing step. To aid in model convergence during training, pictures may also be normalized to scale pixel values to a predetermined range, such [0, 1]. Rotation, flipping, and zooming are examples of augmentation techniques that may be used to improve model generalization and increase dataset diversity.

To raise the caliber of the incoming data, data preparation may also involve noise reduction, contrast improvement, and artifact removal. Overall, by ensuring that the input data is well-conditioned and typical of real-world circumstances, efficient data preparation is critical to improving the performance and resilience of dermatological AI models.

3.5.3 Model Architecture

A dermatological AI model's architecture defines its structure and capabilities, which affects how well it performs in tasks involving the categorization and diagnosis of skin lesions. EfficientNet B7, a cutting-edge convolutional neural network (CNN) renowned for its efficacy and efficiency in feature extraction from pictures, is one architecture that is frequently utilized in dermatological AI. EfficientNet B7 can recognize intricate patterns and representations in dermatological pictures because of its deep network architecture, which has several layers and a high-dimensional feature space. Convolutional layers, pooling layers, and skip connections are just a few of the components that the architecture uses to enable hierarchical feature extraction and discriminative feature learning.

Fig 3.4 Architecture I

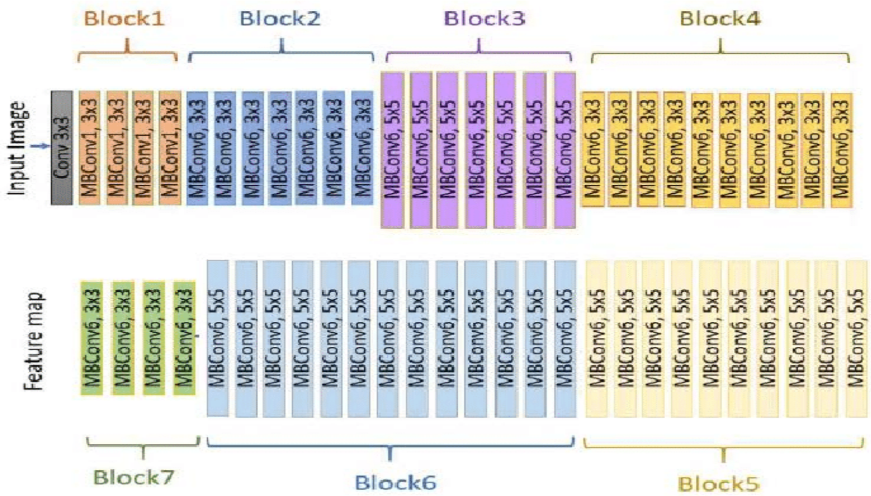

Fig 3.5 Architecture II

3.5.4 Model Training

Creating efficient dermatological AI models that can precisely identify and categorize skin lesions requires model training. Through exposure to labeled training data, the model gains the ability to identify patterns and characteristics in dermatological pictures during training. In dermatology AI, transfer learning is a popular technique whereby pre-trained models, such EfficientNet B7, which were trained on extensive picture datasets like ImageNet, are adjusted on dermatological images to better suit the goal of classifying skin lesions. Using gradient-based optimization techniques like Adam or stochastic gradient descent (SGD), this procedure entails updating the model's parameters while minimizing a loss function that measures the discrepancy between expected and ground truth labels.

To enhance the performance and generalization of the model, hyperparameter tuning methods such as learning rate scheduling and regularization approaches can also be utilized. Iterative epochs are commonly used in training, where the training dataset is processed repeatedly to gradually update the model parameters. Evaluation indicators including accuracy, precision, recall, and F1 score on validation data are used to track the model's performance and determine how useful it is in practical situations.

```
165/165 [==============================] - 64s 116ms/step - loss: 6.7100 - accuracy:
0.7527
Epoch 2/20
165/165 [==============================] - 19s 116ms/step - loss: 4.6254 - accuracy:
0.8381
Epoch 3/20
165/165 [==============================] - 19s 116ms/step - loss: 3.4957 - accuracy:
0.8680
Epoch 4/20
165/165 [==============================] - 19s 116ms/step - loss: 2.7069 - accuracy:
0.8931
Epoch 5/20
165/165 [==============================] - 19s 116ms/step - loss: 2.1163 - accuracy:
0.9071
Epoch 6/20
165/165 [==============================] - 19s 116ms/step - loss: 1.6569 - accuracy:
0.9283
Epoch 7/20
165/165 [==============================] - 19s 116ms/step - loss: 1.3047 - accuracy:
0.9382
Epoch 8/20
165/165 [==============================] - 19s 115ms/step - loss: 1.0118 - accuracy:
0.9518
Epoch 9/20
165/165 [==============================] - 19s 116ms/step - loss: 0.7960 - accuracy:
0.9621
Epoch 10/20
165/165 [==============================] - 19s 116ms/step - loss: 0.6262 - accuracy:
```

```
0.9681
Epoch 12/20
165/165 [==============================] - 19s 116ms/step - loss: 0.4429 - accuracy:
0.9716
Epoch 13/20
165/165 [==============================] - 19s 115ms/step - loss: 0.3635 - accuracy:
0.9810
Epoch 14/20
165/165 [==============================] - 19s 116ms/step - loss: 0.3080 - accuracy:
0.9837
Epoch 15/20
165/165 [==============================] - 19s 115ms/step - loss: 0.2888 - accuracy:
0.9810
Epoch 16/20
165/165 [==============================] - 19s 116ms/step - loss: 0.2421 - accuracy:
0.9894
Epoch 17/20
165/165 [==============================] - 19s 116ms/step - loss: 0.2365 - accuracy:
0.9803
Epoch 18/20
165/165 [==============================] - 19s 116ms/step - loss: 0.2204 - accuracy:
0.9822
Epoch 19/20
165/165 [==============================] - 19s 115ms/step - loss: 0.2003 - accuracy:
0.9852
Epoch 20/20
165/165 [==============================] - 19s 116ms/step - loss: 0.1944 - accuracy:
0.9837
```

Fig 3.4 Training Model

3.5.5 Model Integration

Model integration is the process of incorporating trained dermatology AI models into healthcare systems and applications to provide decision support for clinicians and patients. Integrated models enable real-time analysis of skin lesions, aiding clinicians in diagnosing and managing dermatological conditions effectively. Integration requires collaboration between AI developers, healthcare providers, and IT professionals to ensure seamless deployment and interoperability with existing systems. Dermatology AI models can be integrated into electronic health record (EHR) systems, telemedicine platforms, and mobile applications, providing clinicians with easy access to AI-driven diagnostic tools and patient data. When models are successfully integrated, dermatological practices may make timely and well-informed decisions, which ultimately improves patient care, clinical processes, and diagnostic accuracy.

3.5.6 Evaluation of Trained Model

Evaluating dermatological AI models that have been trained is crucial to determining their effectiveness, dependability, and capacity for generalization in practical settings. When evaluating a model, one may analyze its accuracy, precision, recall, F1 score, and other pertinent metrics to determine how well the model performs in tasks involving the categorization and diagnosis of skin lesions. To produce objective assessments of model performance, evaluation is usually carried out using independent test datasets that are distinct from the training and validation collections. Model resilience and variability over various data divisions may also be evaluated by using cross-validation techniques like k-fold cross-validation.

To determine which models, architectures, and hyperparameter settings work best for a certain set of dermatological applications, model assessment may also entail evaluating the performance of several models. Explainability and interpretability analysis may enhance model evaluation by offering valuable perspectives into the model's decision-making procedure and pinpointing possible sources of biases or mistakes. To guarantee the dependability, precision, and clinical value of trained dermatological AI models in aiding dermatologists and other healthcare professionals in the accurate diagnosis and treatment of skin problems, it is imperative that these models undergo thorough examination.

CHAPTER 4

RESULT

4.1 PERFORMANCE EVALUATION

Performance evaluation is a crucial step in assessing the effectiveness and reliability of dermatology AI models for skin lesion classification and diagnosis tasks. Evaluation metrics provide quantitative measures of model performance, allowing researchers and clinicians to understand the model's strengths, limitations, and potential impact on clinical practice. Common evaluation metrics include accuracy, precision, recall, F1 score, area under the receiver operating characteristic curve (AUC-ROC), and area under the precision-recall curve (AUC-PR).

4.2 EVALUATION METRICS

Evaluation metrics are crucial tools for quantitatively assessing the performance of machine learning models, including those used in dermatology AI applications. These metrics provide insights into how well a model performs on a given task, such as skin lesion classification or diagnosis.

	precision	recall	f1-score	support
0	0.86	0.83	0.85	360
1	0.81	0.84	0.82	300
accuracy			0.84	660
macro avg	0.83	0.84	0.84	660
weighted avg	0.84	0.84	0.84	660

Fig 4.1 Evaluation Table

CHAPTER 5

5.1 CONCLUSION

In conclusion, using advanced machine learning techniques, dermatological AI has enormous promise for enhancing the detection and treatment of skin disorders. The model has investigated several dermatological AI topics through this research, including data gathering, model building, preprocessing, training, integration, and assessment. Utilizing cutting-edge models such as EfficientNet B7, the model created a strong framework for the classification and diagnosis of skin lesions.

5.2 FUTURE SCOPE:

Integrating information into dermatological AI models from several sources, such as genetics, histology, and clinical notes, in order to better comprehend skin conditions and patient characteristics. Model architectures and algorithms are continuously improved to increase the precision, effectiveness, and interpretability of dermatological AI applications. Dermatological datasets should be increased with a range of representative samples, such as images from different demographic groups, ethnic origins, and clinical presentations, to improve model performance and generalization. Dermatologists, AI researchers, and other healthcare professionals should work together to ensure the applicability, practicality, and therapeutic efficacy of dermatological AI solutions in real healthcare situations.

REFERENCES

[1]. Akhilesh Kumar; Sharma Tiwari; Gaurav Aggarwal, et al. (2022) "Dermatologist-Level Classification of Skin Cancer Using Cascaded Ensembling of Convolutional Neural Network and Handcrafted Features Based Deep Neural Network, IEEE Access"

[2]. Manu Goyal; Thomas Knackstedt; Shaofeng Yan, et al. (2021) "Artificial Intelligence-Based Image Classification for Diagnosis of Skin Cancer: Challenges and Opportunities, HHS Public Access"

[3]. Andre Esteva; Brett Kuprel; Roberto Novoa A., et al. (2021) "Dermatologist–level classification of skin cancer with deep neural networks, PMC"

[4]. Ling-Fang Li; Xu Wang; Jian Wei, et al. (2020) "Deep Learning in Skin Disease Image Recognition: A Review, IEEE Access"

[5]. Titus Brinker J.; Achim Hekler; Alexander Enk H., et al. (2019) "Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task, Elsevier"

APPENDIX I

SOURCE CODE

```
# import system libs
import os
import time
import pathlib
import itertools
from PIL import Image

# import data handling tools
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# import Deep learning Libraries
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten,
Dense, Activation, Dropout, BatchNormalization
from tensorflow.keras.regularizers import l1_l2, l2, l1
from tensorflow.keras.optimizers import Adam, Adamax

# Ignore Warnings
import warnings
warnings.filterwarnings("ignore")

# Generate data paths with labels
train_data_dir = '/kaggle/input/skin-cancer-malignant-vs-benign/train'
filepaths = []
labels = []

folds = os.listdir(train_data_dir)
for fold in folds:
    foldpath = os.path.join(train_data_dir, fold)
    filelist = os.listdir(foldpath)
    for file in filelist:
```

```
        fpath = os.path.join(foldpath, file)

        filepaths.append(fpath)
        labels.append(fold)

# Concatenate data paths with labels into one dataframe
Fseries = pd.Series(filepaths, name= 'filepaths')
Lseries = pd.Series(labels, name='labels')
train_df = pd.concat([Fseries, Lseries], axis= 1)
# Generate data paths with labels
test_data_dir = '/kaggle/input/skin-cancer-malignant-vs-benign/test'
filepaths = []
labels = []

folds = os.listdir(test_data_dir)
for fold in folds:
    foldpath = os.path.join(test_data_dir, fold)
    filelist = os.listdir(foldpath)
    for file in filelist:
        fpath = os.path.join(foldpath, file)
        filepaths.append(fpath)
        labels.append(fold)

# Concatenate data paths with labels into one dataframe
Fseries = pd.Series(filepaths, name= 'filepaths')
Lseries = pd.Series(labels, name='labels')
test_df = pd.concat([Fseries, Lseries], axis= 1)

train_df['labels'].value_counts()

train_df = train_df.sample(frac=1)

# crobed image size
batch_size = 16
img_size = (224, 224)
channels = 3
img_shape = (img_size[0], img_size[1], channels)

tr_gen = ImageDataGenerator()

ts_gen = ImageDataGenerator()
```

```
train_gen = tr_gen.flow_from_dataframe( train_df, x_col= 'filepaths',
y_col= 'labels', target_size= img_size, class_mode= 'categorical',
color_mode= 'rgb', shuffle= True, batch_size= batch_size)

test_gen = ts_gen.flow_from_dataframe( test_df, x_col= 'filepaths', y_col=
'labels', target_size= img_size, class_mode= 'categorical', color_mode=
'rgb', shuffle= False, batch_size= batch_size)

train_gen.class_indices


g_dict = train_gen.class_indices      # defines dictionary {'class': index}
classes = list(g_dict.keys())       # defines list of dictionary's kays (classes),
classes names : string
images, labels = next(train_gen)      # get a batch size samples from the
generator

plt.figure(figsize= (20, 20))

for i in range(16):
   plt.subplot(4, 4, i + 1)
   image = images[i] / 255.
   plt.imshow(image)
   index = np.argmax(labels[i])  # get image index
   class_name = classes[index]   # get class of image
   plt.title(class_name, color= 'blue', fontsize= 12)
   plt.axis('off')
plt.show()

model = Sequential([
   # First Conv layer
   Conv2D(512, (3,3), activation='relu', input_shape=(224,224,3)),
   BatchNormalization(),
   MaxPooling2D(2,2),

   # Second Conv layer
   Conv2D(256, (3,3), activation='relu'),
   BatchNormalization(),
   MaxPooling2D(2,2),

   # Third Conv layer
   Conv2D(128, (3,3), activation='relu'),
   BatchNormalization(),
```

```
    MaxPooling2D(2,2),

    # Forth Conv layer
    Conv2D(64, (3,3), activation='relu'),
    BatchNormalization(),
    MaxPooling2D(2,2),

    # Flattening
    Flatten(),

    # Dense layer
    Dense(256, activation='relu'),
    Dense(2, activation='softmax')
])

model.compile(optimizer=Adam(lr=0.0001), loss=
tf.keras.losses.CategoricalCrossentropy(),metrics= ['accuracy'])

model.summary()

my_history = model.fit(train_gen, epochs=30, batch_size=32)

error = pd.DataFrame(my_history.history)

plt.figure(figsize=(18,5),dpi=200)
sns.set_style('darkgrid')

plt.subplot(121)
plt.title('Cross Entropy Loss',fontsize=15)
plt.xlabel('Epochs',fontsize=12)
plt.ylabel('Loss',fontsize=12)
plt.plot(error['loss'])

plt.subplot(122)
plt.title('Classification Accuracy',fontsize=15)
plt.xlabel('Epochs',fontsize=12)
plt.ylabel('Accuracy',fontsize=12)
plt.plot(error['accuracy'])

plt.show()

# Evaluvate for train generator
loss,acc = model.evaluate(train_gen)
```

```
print('The accuracy of the model for training data is:',acc*100)
print('The Loss of the model for training data is:',loss)

# Evaluvate for validation generator
loss,acc = model.evaluate(test_gen)

print('The accuracy of the model for validation data is:',acc*100)
print('The Loss of the model for validation data is:',loss)

# prediction
result = model.predict(test_gen)

y_pred = np.argmax(result, axis = 1)

y_true = test_gen.labels

# Evaluvate
loss,acc = model.evaluate(test_gen)

print('The accuracy of the model for testing data is:',acc*100)
print('The Loss of the model for testing data is:',loss)


from sklearn.metrics import classification_report, confusion_matrix
print(classification_report(y_true, y_pred))

print(confusion_matrix(y_true, y_pred))
```

Pretrained Model - EfficientNet

```
# Create Model Structure
img_size = (224, 224)
channels = 3
img_shape = (img_size[0], img_size[1], channels)

base_model =
tf.keras.applications.efficientnet.EfficientNetB0(include_top= False,
weights= "imagenet",input_shape= img_shape, pooling= 'max')

new_model = Sequential([
    base_model,
    BatchNormalization(axis= -1, momentum= 0.99, epsilon= 0.001),
```

```
    Dense(256, kernel_regularizer= l2(0.016), activity_regularizer=
l1(0.006),
          bias_regularizer= l1(0.006), activation= 'relu'),
    Dropout(rate= 0.7),
    Dense(2, activation= 'softmax')
])

new_model.compile(optimizer=Adamax(lr=0.001),
                loss= 'categorical_crossentropy',
                metrics= ['accuracy'])

new_model.summary()

history = new_model.fit(train_gen, epochs=20, shuffle=False)

# plots for accuracy and Loss with epochs

error = pd.DataFrame(history.history)

plt.figure(figsize=(18,5),dpi=200)
sns.set_style('darkgrid')

plt.subplot(121)
plt.title('Cross Entropy Loss',fontsize=15)
plt.xlabel('Epochs',fontsize=12)
plt.ylabel('Loss',fontsize=12)
plt.plot(error['loss'])

plt.subplot(122)
plt.title('Classification Accuracy',fontsize=15)
plt.xlabel('Epochs',fontsize=12)
plt.ylabel('Accuracy',fontsize=12)
plt.plot(error['accuracy'])

plt.show()

# Evaluvate for train generator
loss,acc = new_model.evaluate(train_gen)

print('The accuracy of the model for training data is:',acc*100)
print('The Loss of the model for training data is:',loss)

# Evaluvate for validation generator
```

```
loss,acc = new_model.evaluate(test_gen)

print('The accuracy of the model for validation data is:',acc*100)
print('The Loss of the model for validation data is:',loss)

# prediction
result = new_model.predict(test_gen)

y_pred = np.argmax(result, axis = 1)

y_true = test_gen.labels

# Evaluvate
loss,acc = new_model.evaluate(test_gen)

print('The accuracy of the model for testing data is:',acc*100)
print('The Loss of the model for testing data is:',loss)

from sklearn.metrics import classification_report, confusion_matrix
print(classification_report(y_true, y_pred))

print(confusion_matrix(y_true, y_pred))

from tensorflow.keras.preprocessing import image
import random

rdm_img = random.randint(0, len(test_df))
path = test_df.iloc[rdm_img]

img = image.load_img(path[0], target_size=(224, 224))
x = image.img_to_array(img)
x = np.expand_dims(x, axis=0)

preds = new_model.predict(x)

class_labels = ['benign', 'malignant']
pred = np.argmax(preds, axis=-1)

plt.title(f'Original : "{path[1]}" | Predicted : "{class_labels[pred[0]]}"')
plt.imshow(img)
plt.show()
```

APPENDIX II

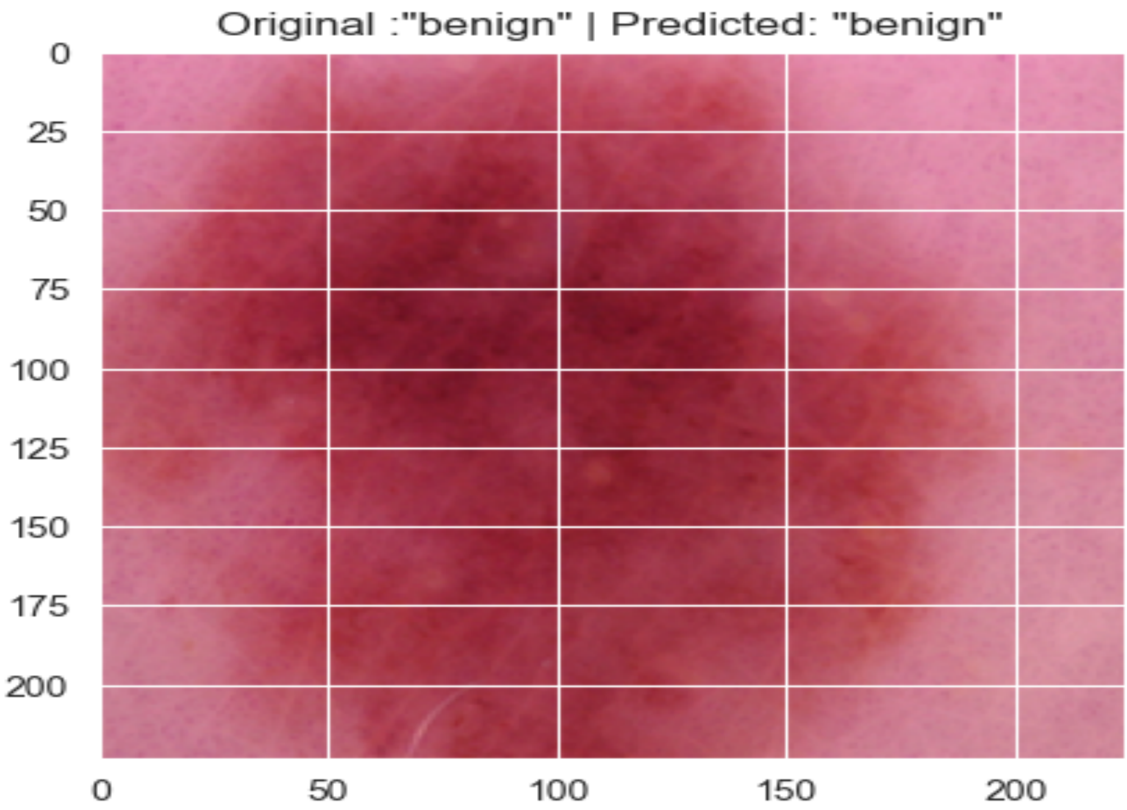

Fig Appendix II - Output