FORMAT OF COVER PAGE
Book bound royal blue color with white Fonts

SVM-RF-CNN GRADIENT BOOSTING ENSEMBLE LEARNING FOR SKIN CANCER DETECTION

BY
Akankshya Pattanaik (2361020015)
Asmita Bijaya (2361020030)
Abhilipsa Sahoo (2361020007)
Suchitra Das (2361020296)

Department of Computer Application
Institute of Technical Education and Research
2025

PROJECT REPORT

ON

SVM-RF-CNN GRADIENT BOOSTING ENSEMBLE LEARNING FOR SKIN CANCER DETECTION

SUBMITTED IN PARTIAL FULFILLMENT FOR AWARD OFDEGREE IN

MASTER IN COMPUTER APPLICATION

(BATCH 2023-2025)

BY

Akankshya Pattanaik (2361020015)
Asmita Bijaya (2361020030)
Abhilipsa Sahoo (2361020007)
Suchitra Das (2361020296)

UNDER THE ESTEEMED GUIDANCE OF

Dr. Bichitrananda Patra

Department of Computer Application
Institute of Technical Education and Research
2025

Declaration

I hereby declare that the Project entitled "**SVM-RF-CNN GRADIENT BOOSTING ENSEMBLE LEARNING FOR SKIN CANCER DETECTION"** submitted to the Department of Computer Application, **Institute of Technical Education & Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha** in partial fulfilment for theaward of the degree of **Master in Computer Application** in the year 2024 is an authentic record of my own work carried out under the guidance of Dr./Ms./Mr. **Dr. Bichitrananda Patra** and thatthe project has not been previously formed the basis for the award of any other degree. The report has been prepared in compliance to the guidelines specified by the University.

Place: Bhubaneswar
Date: 24-05-2025

Akankshya Pattanaik (2361020015)
Asmita Bijaya (2361020030)
Abhilipsa Sahoo (2361020007)
Suchitra Das (2361020296)

This is to certify that the above statement made by the candidate is correct to the best of myknowledge.

<Signature>
GUIDE
Dr. Bichitrananda Patra
Professor, Computer Science

Acknowledgement

I am very grateful to my project guide Dr. Bichitrananda Patra for givingus valuable time and constructive guidance in preparing the Project. It would not have been possible to complete this project in short period of time without his/her kind encouragement and valuable guidance.

Akankshya Pattanaik (24-05-2025)
Asmita Bijaya (24-05-2025)
Abhilipsa Sahoo (24-05-2025)
Suchitra Das (24-05-2025)

CERTIFICATE

This is to certify that this project entitled "**SVM-RF-CNN GRADIENT BOOSTING ENSEMBLE LEARNING FOR SKIN CANCER DETECTION**" submitted in partial fulfilment of the degree of MASTER IN COMPUTER APPLICATION to the Department of Computer Application, Institute of Technical Education and Research, Under Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, done by Mr./Ms. Akankshya Pattanaik, Asmita Bijaya, Abhilipsa Sahoo, Suchitra Das, Registration No. 2361020015, 2361020030, 2361020007, 2361020296 is an authentic workcarried out by them is worthy of acceptance for award of the degree. The work fulfils the entire requirement as per the regulation of the University and it has reached the standard needed for submission.

<Signature of the Guide>

Name of the Guide

GUIDE

Designation and Affiliation

<Signature of the HOD>

HOD

Department of ComputerApplication

SIKSHA 'O' ANUSANDHAN

(DEEMED TO BE UNIVERSITY)

INSTITUTE OF TECHNICAL EDUCATION AND RESEARCH

Department of Computer Application

4th Semester MCA 2025

Project Synopsis

Project Group No: Group 2
Title of the Project: SVM-RF-CNN Gradient Boosting Ensemble Learning for Skin Cancer Detection
Platform to be used: Jupyter Notebook, Python
Project Description:

Skin cancer is a formidable and increasingly deadly disease, with rising mortality rates attributed to inadequate awareness of its symptoms and preventive measures. Delayed detection often proves fatal, emphasizing the critical importance of early diagnosis. This project leverages machine learning and image processing techniques to detect and classify various skin cancer types. By extracting features from skin cancer images, we employ machine learning algorithms specifically, Support Vector Machines (SVM) and Artificial Neural Networks (ANN) – to classify the images. SVM, a supervised learning approach, identifies an optimal hyperplane to distinguish difference between malignant and non-cancerous skin lesions. ANN, ensemble model inspired by the human brain, recognizes complex patterns through multiple layers, rendering it exceptionally suitable for image classification tasks. Both classifiers are integral to skin cancer detection, and their performance is evaluated to determine the most effective model for classification. This research aims to contribute to the creation of precise and reliable diagnostic tools, ultimately enhancing treatment outcomes and saving lives.

(Full Signature of the Group Members with Date)

1. Akankshya Pattanaik (24-05-2025)
2. Asmita Bijaya (24-05-2025)
3. Abhilipsa Sahoo (24-05-2025)
4. Suchitra Das (24-05-2025)

(Signature of the Internal Guide with Date)
Name & Designation: Dr. Bichitrananda Patra **(**Professor, Computer Science)

(Signature of the Project Coordinator) **Approved:** **Yes** **No**
Date: 24-05-2025

CONTRIBUTION OF GROUP MEMBERS

This project was completed as a team effort, with each member contributing significantly to various stages of development, experimentation, and documentation. The following outlines the key contributions made by each team member:

1. Akankshya Pattanaik (2361020015):
    - Read a research paper.
    - Design the model based on the research paper.
    - Implement the project accordingly.
2. Asmita Bijaya (2361020030):
    - Read a research paper.
    - Analyze the results presented in the paper.
    - Write the report based on your analysis.
3. Abhilipsa Sahoo (2361020007):
    - Read a research paper.
    - Represent the output graphically.
    - Write the project article based on your findings.
4. Suchitra Das (2361020296):
    - Read a research paper.
    - Find a suitable dataset and gather helpful resources (e.g., libraries, tutorials)
    - Prepare a PowerPoint presentation (PPT)

LIST OF FIGURES

TABLE OF CONTENTS

ABSTRACT

The most prevalent types of skin cancer include cancerous which is one of the deadliest cancers of all if not caught early. Conventional methods of diagnosis depend on the observation of skin lesions by trained dermatologists followed with testing. Even though these methods provide accurate results, they are based on subjective judgement, take a lot of time and are not readily available in many places, especially in developing countries. These shortcomings have given rise to improvement in technologies such as **Machine Learning (M L)**, which have proven useful in the area of medical diagnosis by increasing efficiency in skin cancer detection.

With machine learning, computers are able to learn from information and make accurate predictions based on that data. In this instance, machine learning models are trained using thousands of images of skin lesions. The model differentiates between benign skin spots and malignant skin lesions. A more experienced type of machine learning is deep learning which uses models called **Convolutional Neural Networks (CNNs)**. These models excel in image analysis and identifying patterns that are not easily detectable do human observation.

This research is aimed at creating an ML based system for automated skin cancer detection using images. It comprises identifying a large collection of images of skin lesions, data cleansing and preparation, model training, and evaluating the model's performance. The system is measured against standard accuracy benchmarks such as **precision, recall, and F1- score** to confirm that it is indeed accurate.

To sum up, the application of machine learning offers a significant advancement towards easier and faster diagnosis of skin cancer. It can augment the efforts of medical worker, reduce their workload, and make skin cancer screening widely available internationally. Further work in this area would likely integrate ML frameworks fundamentally into health systems in the coming years.

Keywords: Skin cancer, Melanoma, Medical diagnosis, Machine Learning (ML), Deep Learning, Convolutional Neural Networks (CNNs), Skin lesion images, Image analysis, Automated detection Model training, Data preparation, Model evaluation, Precision, Recall, F1-score, Clinical validation, Diagnostic tools, Healthcare technology, medical imaging, AI in healthcare.

CHAPTER 1

INTRODUCTION

Skin cancer is one of the most common forms of cancer worldwide, with millions of new cases diagnosed each year. Early detection plays a critical role in increasing the chances of successful treatment and survival. Traditionally, skin cancer diagnosis relies on visual inspection by dermatologists followed by biopsy and histopathological analysis. However, these methods can be time-consuming, subjective, and limited by the availability of expert professionals.

With the rise of artificial intelligence, particularly Machine Learning (M L), there is growing potential to enhance the accuracy and efficiency of skin cancer detection.

Machine learning, a subset of artificial intelligence, involves training computer systems to learn from data and make predictions or decisions without being explicitly programmed. In the context of skin cancer detection, ML algorithms can be trained on large datasets of skin lesion images to distinguish between benign and malignant lesions.

Machine Learning models, especially **Convolutional Neural Networks (C N N s)**, have shown remarkable performance in image classification.tasks and are increasingly applied to medical image analysis.

The combination of machine learning in dermatology offers many benefits. It can provide rapid and consistent assessments, reduce diagnostic errors, and assist in screening large populations, especially in regions with limited access to healthcare professionals. Moreover, mobile-based applications powered by ML can make initial skin assessments more accessible to the public, likely encouraging earlier consultations with medical experts.

Despite its promise, implementing ML for skin cancer detection presents several challenges. These include the need for high-quality and diverse datasets, model interpretability, and regulatory approval for clinical use. Additionally, ensuring that the models are free from bias and perform reliably across different skin tones and lesion types is essential for equitable healthcare delivery.

In summary, the application of machine learning in skin cancer detection represents a significant advancement in medical diagnostics.

By manipulating powerful computational techniques, ML has the potential to support dermatologists, improve diagnostic accuracy, and ultimately save lives. Continued research, ethical development, and clinical validation are key to realizing the full potential of this technology in the fight against skin cancer.

1.1 BACKGROUND

Skin cancer in among the most prevalent forms of cancer globally, with millions of new cases reported annually. Skin cancer happens when out-of-control skin cells develop because of too much sun or tanning bed exposure to ultraviolet (UV) radiation. Skin cancer is primarily divided into malignant (harmful and cancerous, eg., melanoma) and benign (non-cancerous) conditions. Early detection of malignant skin cancer is important since it has the potential to spread rapidly throughout the rest of the body and become deadly if left untreated in a timely manner.

Skin cancer has traditionally been diagnosed by visual examination by dermatologies, with biopsy for confirmation. This process is time- consuming, costly, and sometimes subjective based on the experience of the doctor. Additionally, in most regions of the globe, there is a lack of properly trained dermatologists, which makes it challenging to get quick and accurate diagnoses.

With developments in machine learning (M L) and deep learning particularly with regard to recognizing images, computerized skin cancer detection is now a potential answer. The technologies are able to scan images of skin lesions and recognize patterns too slight for the human eve. These models, including RseNet50 and EfficientNetB3, have previously demonstrated outstanding performance in image classification and can be trained to separate benign and malignant skin lesions based on large datasets of dermoscopic images.

In this project, a machine learning (ML) approach is employed for building a model that is able to classify skin lesions as benign or malignant correctly. With the high-quality dataset such as **Skin Cancer: Malignant vs Benign** containing thousands of labelled images of skin lesions, the model is trained to acquire the characteristics of various skin conditions. Key steps of resizing of images, conversion of RGB, normalization, and feature extraction are employed to set the data ready for training.

1.2 MOTIVATION

Skin cancer is one the most prevalent and most rapidly developing forms of cancer around the world. Its treatment is more effective when diagnosed at an early stage, but there are many areas where a dermatologist and equipment for diagnostic imaging are not available. Conventional techniques of visual inspection and biopsy face the challenge of being lengthy, expensive, and are subjectively reliant on the physician's expertise.

The development of artificial intelligence, especially in the areas of machine learning (ML), provides an avenue through which skin cancer detection can be enhanced. Deep learning algorithms are capable of being trained using thousands of images of skin lesions, enabling detection of patterns which indicate cancer. These algorithms are able to rapidly and accurately interpret new images and provide valuable assistance to physicians or serve as initial stage screening tools.

The primary motivation in using machine learning is to design a device that is reliable, fast, and easy to access and can greatly assist with early diagnosis, lightening the load on medical specialists, and reaching medically ignored regions. This approach seeks to combine technology and medicine with the hope of improving clinical outcomes by reducing the number of patients who go undiagnosed or put off getting help until it is too late.

1.3 PROBLEM STATEMENT AND OBJECTIVES

Skin cancer is a growing global health concern, with millions of people affected every year. Early detection is critical to prevent the disease from progressing and becoming life-threatening, especially in the case of **malignant melanoma**, the most dangerous form of skin cancer. However, traditional diagnosis methods rely heavily on clinical examination and biopsy, which can be **time-consuming** , **expensive**, and **dependent on the expertise of dermatologists**. In many regions, especially rural or underdeveloped areas, access to skilled professionals and diagnostic tools is limited, resulting in **delayed or incorrect diagnoses**.

The primary goal of this project is to build a reliable and effective machine learning-powered system for the early diagnosis of skin cancer based on image data. Through training models on huge datasets of skin lesion images, the system would be able to classify lesions as Benign (non-

cancerous) or Malignant (cancerous) with high dependability. This can help doctors provide quicker and more accurate diagnoses, particularly in regions where there is a lack of dermatological skill.

The precise objectives are:

i. Obtaining and preprocessing high-quality skin lesion images from trustworthy sources.
ii. Applying and training machine learning models, especially convolutional neural networks (C N N s), on image classification.
iii. Model performance evaluation based on measures like accuracy, precision, recall, and F1-score.
iv. Making the model robust and generalizable to various skin colors and types of lesions.
v. Investigating the possibility of embedding the model in easy-to-use platforms, e.g., mobile applications, for remote or initial screening.

By fulfilling these goals, the project seeks to lead towards quicker diagnosis, greater access to screening, and ultimately improved patient outcomes for those at risk of skin cancer.

1.4 CHALLENGES

Creating a machine learning model for skin cancer diagnosis has a number of major issues:
Data quality and imbalance: Medical image datasets, including those for one of the central problems in constructing a skin cancer detection model is the imbalance and quality of the dataset. In the majority of medical images databases available, benign cases overwhelm malignant cases by vast numbers. This class imbalance has the potential to bias the machine learning model, which, in turn, makes it produce the more frequent class more frequently, and lose detection of the vital malignant cases.
Another difficulty lies in the high variability of skin lesion images. The images can vary based on color, texture, size, shape, illumination, and even the existence of artifacts such as hair or shadows. Such variations complicate the learning of consistent and meaningful patterns necessary for correct classification by the model.
It is also a problem of overfitting, particularly when the highly complicated machine learning models such as ResNet50 or EfficientNetB3 are applied to insufficient data. Overfitting

happens when the model is a good fit for training data but does not generalize on fresh, unseen data.

Another concern is the interpretability of machine learning models. Such models tend to be “Black boxes”, and it is difficult for medical practitioners to comprehend or rely on the outcomes without sound justification or visual explanations.

Lastly, medical verification is necessary. Even if a model is extremely accurate, it needs to undergo clinical trial and approval before it can be used in actual medical practice, and this takes time and specialists’ attention.

1.5 ORIGINAL CONTRIBUTION

This project introduces a machine learning solution to detect skin cancer that adds original value in a number of important areas. Though numerous investigations have considered applying artificial intelligence in medical imaging, this effort aims to optimize a deep model specially designed to classify skin lesions with focus on accessibility, performance, and fairness.

One of the main contributions is relying on a thoughtfully selected and pre-processed dataset, covering mixed skin types and lesion statuses. This serves to alleviate one of the major flaws of most current models, which tend to perform poorly on darker skin or uncommon lesion types. By maintaining data diversity and parity, this system should minimize bias and enhance diagnostic performance across various populations.

Another is the application and tuning of Convolutional Neural Networks (CNNs) for high-precision image classification. The architecture of the model, training government, and hyperparameters were optimized to enhance performance with the system remaining efficient and appropriate for real-time usage.

Moreover, this project investigates the creation of an easy and user-friendly interface that might be converted to a mobile or web-based platform so that skin cancer screening devices become more public-friendly, particularly in rural or underserved areas.

Lastly, the project presents a comprehensive assessment through key performance metrics and comparative evaluation against current methods. The output reveals that the model performs comparably while having a lightweight design for application in low-resource environment.

Combined, these contributions illustrate a step in the direction of practical, inclusive, and scalable machine learning solutions towards skin cancer detection with the potential to aid healthcare

providers and empower people with early diagnostic tools.

1.6 ORGANIZATION OF THE REPORT

The first chapter is the Introduction, which covers the significance of early skin cancer detection and how can machine learning provide support to address this issue. It provides the objective of the project, background information necessary to develop an adequate understanding of the issue, and the relevance of the stated background.

The second chapter is the Literature Review, which considers past research and methodology used for skin cancer detection, and also considers the successes and failures of past approaches to support the need for a new and improved technique that used deep learning techniques.

The next section is the Methodology section where a description of the process for constructing the model is explained. This section outlines the steps taken to preprocess the data that includes, image resizing, RGB conversion and values normalization, which will also occur with feature extraction from deep learning model attempts, like EfficientNetB3 or ResNet50, that each represent the big data computational pathways.

The next chapter is the Dataset Description chapter. This chapter provides a detailed explanation of the dataset used (for instance, Skin Cancer: Malignant vs Benign), describing the number of images, type of lesions, as well as how this data was spread into training and testing data.

The Results and Evaluation chapter describe the accuracy, precision, recall, and confusion matrix when evaluating the performance of the different model attempts. Lastly, to help guide understanding the decision-making processes, several visualizations and comparisons between the classifying models would be provided in the reports later chapters.

CHAPTER 2

LITERATURE SURVEY

One of the most prevalent human diseases is skin disease. Skin cancer is a tumour (growth of atypical cells on the skin) in your body's organ. Thes skin tumour are two types, that are malignant (Cancer) and benign (Noncancer). Certain forms of skin cancer may manifest as rough patches, wart-like growths, or reddish or blackish areas. It's critical to recognize these symptoms and routinely examine your skin for any changes because early detection is essential. Skin color is a powerful indicator of this condition in computer vision applications. This approach uses pictures of the skin to identify skin cancer [1].

Skin diseases remain one of the most widespread health issues affecting individuals across the globe, with skin cancer ranking among the most severe and potentially life-threatening conditions within this category. Skin cancer originates from the uncontrolled growth of abnormal skin cells and primarily manifests in two forms: malignant (cancerous) and benign (non-cancerous) tumours. Malignant skin cancers, such as melanoma, basal cell carcinoma, and squamous cell carcinoma, are particularly dangerous and can spread to other parts of the body if not detected and treated early. On the other hand, benign tumours may appear unsightly or uncomfortable but generally do not pose a serious health risk.

Skin cancer often presents itself in the form of rough patches, wart-like growths, discolorations, or irregularly shaped moles. Early detection is critical, as it significantly increases the chances of successful treatment and survival. However, identifying these symptoms accurately and consistently is a challenging task due to the vast diversity of skin types, lesion shapes, and color variations. In clinical practice, dermatologists often rely on dermoscopic images and their visual judgment to determine whether a lesion is malignant or benign. This subjective nature of diagnosis can lead to inconsistencies and errors, emphasizing the need for a more objective and automated diagnostic approach.

In the twenty-first century, technological advancements in artificial intelligence (AI), particularly in the domain of machine learning (ML) and computer vision, have opened new avenues for early and accurate skin cancer detection. Machine learning models trained on annotated datasets of dermoscopic images can identify patterns, colors, and textures that are indicative of cancerous lesions.

These models can support dermatologists by serving as decision support tools or function autonomously in screening applications, especially in remote or underserved areas.

A major cause of skin cancer is prolonged exposure to ultraviolet (UV) radiation from the sun or tanning beds. Other risk factors include genetic predisposition, air pollution, and unhealthy lifestyle habits. Given the rising incidence of skin cancer globally, there is an urgent need to enhance diagnostic accuracy while minimizing the reliance on manual assessments.

To address these issues, researchers have proposed various image-based approaches for skin cancer detection. One such study by Murugan et al. (2021) introduced a robust ML-based pipeline for analyzing dermoscopic images. The system began with preprocessing steps such as noise removal using a median filter, followed by segmentation using the Mean Shift method to isolate the region of interest (ROI).

The study employed three feature extraction techniques: Moment Invariant Features, Gray Level Run Length Matrix (GLRLM), and Gray Level Co-occurrence Matrix (GLCM). These methods captured different textural and morphological aspects of skin lesions, which were then used for classification. Among the classifiers used—including Support Vector Machine (SVM), Random Forest (RF), and Probabilistic Neural Network (PNN)—a hybrid SVM+RF model demonstrated superior performance with an accuracy of 89.31%, sensitivity of 88.56%, and specificity of 87.81%.

Another notable research by Banasode et al. (2021) emphasized the effectiveness of SVM for melanoma classification. Their approach involved multiple steps: image pre-processing (grayscale and HSV conversion), lesion segmentation (using masking techniques), and classification. With high-quality dermoscopic images and carefully engineered features, their system achieved an impressive accuracy of 96.9%, with 95.7% sensitivity and 90.2% specificity. These metrics underscore the potential of SVM in high-precision binary classification tasks, especially in distinguishing between malignant and benign lesions.

Vijayalakshmi's study further explored the integration of CNN and SVM models. Her model included sophisticated preprocessing and segmentation stages followed by classification using both traditional and deep learning techniques. The system reached an accuracy of 85%, highlighting the feasibility of combining handcrafted and deep features for reliable diagnosis.

Compared to traditional classifiers like k-Nearest Neighbors (k-NN) and decision trees,

Convolutional Neural Networks (CNNs) demonstrated superior performance due to their hierarchical learning architecture, which is adept at capturing spatial and contextual patterns in image data.

CNNs, particularly models like VGG16 and ResNet, have proven to be highly effective in medical image classification tasks. Their ability to learn and represent complex visual features makes them especially suitable for identifying subtle abnormalities in dermoscopic images. However, CNNs typically require large volumes of labeled data and significant computational resources, which can limit their accessibility and scalability.

Building upon the findings of these studies, the proposed project aims to develop an advanced skin cancer detection system by leveraging a combination of CNN, SVM, and Random Forest classifiers. The system incorporates a comprehensive image processing pipeline that includes image resizing, RGB conversion, normalization, feature extraction using pre-trained models like EfficientNetB3 and ResNet50, and feature selection using the SelectKBest algorithm with mutual information.

This refined feature set is then classified using an ensemble approach combining CNN, SVM, and RF through soft voting, which improves overall prediction reliability.

In conclusion, the integration of preprocessing, segmentation, robust feature extraction, and ensemble classification forms a powerful framework for automated skin cancer detection. As research continues to advance, such intelligent systems hold the promise of revolutionizing dermatological diagnostics, enabling early intervention, reducing healthcare costs, and ultimately saving lives.

CHAPTER 3
PROPOSED MODEL

3.1 OVERVIEW

The flowchart depicts an end-to-end pipeline for skin cancer classification based on a blend of deep learning, machine learning, and ensemble methods. The following is a succinct description:

i. Datasets: A dataset comprising images of skin lesions with their respective labels of being malignant (cancerous) or benign (non-cancerous) is the starting point.
ii. Preprocessing: Images are normalized by resizing, RGB conversion, and normalization such that they have uniform input for feature extraction.
iii. Feature Extraction: Two pre-trained convolutional neural networks, EfficientNetB3 and ResNet50 (both ImageNet pre-trained), are employed to extract meaningful and dense features from the pre-processed images.
iv. Feature Selection: The most important features among the extracted features are chosen using SelectKBest with Mutual Information. This dimensionality reduction and enhanced classifier performance.
v. Classification: The chosen features are input to three diverse classifiers:
vi. Support Vector Machine (SVM): A robust classifier for high-dimensional data.
vii. Random Forest (RF): A decision tree ensemble with solid predictions.
viii. Convolutional Neural Network (CNN): Learns image patterns directly.
ix. Soft Voting Ensemble: The outputs of all three classifiers are merged employing a soft voting approach, which averages the predicted probabilities as opposed to mere class votes.
x. Final Prediction: The final classification is given by the ensemble itself, classifying as to whether the lesion is benign or malignant.

This multi-model approach enhances accuracy, robustness, and reliability in skin cancer detection.

3.2 METHODOLOGIES

The main aim of this project is to develop a precise, reliable, and automated framework for classifying skin lesions into malignant (cancerous) and benign (non-cancerous) classes. Towards this goal, the project combines a chain of well-designed methodologies such as data preprocessing, feature extraction, classification, and ensemble learning.

The following is a comprehensive description of methodologies employed:

1. Data Preprocessing

   The project starts by obtaining dermoscopic images of skin lesions from public datasets, like the ISIC Archive, which holds thousands of labelled high-resolution images rated by dermatology specialists.

   - Image Resizing: All the images are resized to a common size (e.g., 224x224 or 300x300 pixels) so that they become uniform and compatible with pre-trained models.
   - RGB Conversion: Images are converted to RGB format to ensure uniformity in color representation and compatibility with convolutional neural networks.
   - Normalization: Pixel intensity values are normalized to a range between 0 and 1. This helps models train faster and improves convergence by reducing the effects of varying illumination.

2. Feature Extraction

   Two types of feature extraction techniques are used:

   - Deep Feature Extraction: Pre-trained deep models like EfficientNetB3 and ResNet50 are employed to extract deep features from images. These models are capable of extracting complex spatial patterns, textures, and color variations in skin lesions.
   - Handcraft Features Selection: Concurrently, conventional features such as color, texture, shape , and size are extracted and selected by SelectKBest with Mutual Information to keep only the most informative attributes.

3. Classification Algorithms

Three important machine learning methods are used to classify skin lesions as benign or malignant: Convolutional Neural Network (CNN), Support Vector Machine (SVM), and Random Forest. Each of these algorithms makes a unique contribution to the classification system's performance and dependability.

- CNN (Convolutional Neural Network): A deep model that operates on image data directly to acquire hierarchical spatial features. It is very efficient in pattern recognition.
- SVM (Support Vector Machine): A robust supervised learning algorithm best for binary classification. It performs good for small, clean data and separates the malignant and benign lesions by identifying the best hyperplane.
- Random Forest (RF): An Ensemble learning algorithm involving decision trees. It is applied because it can deal with noisy data and avoid overfitting.

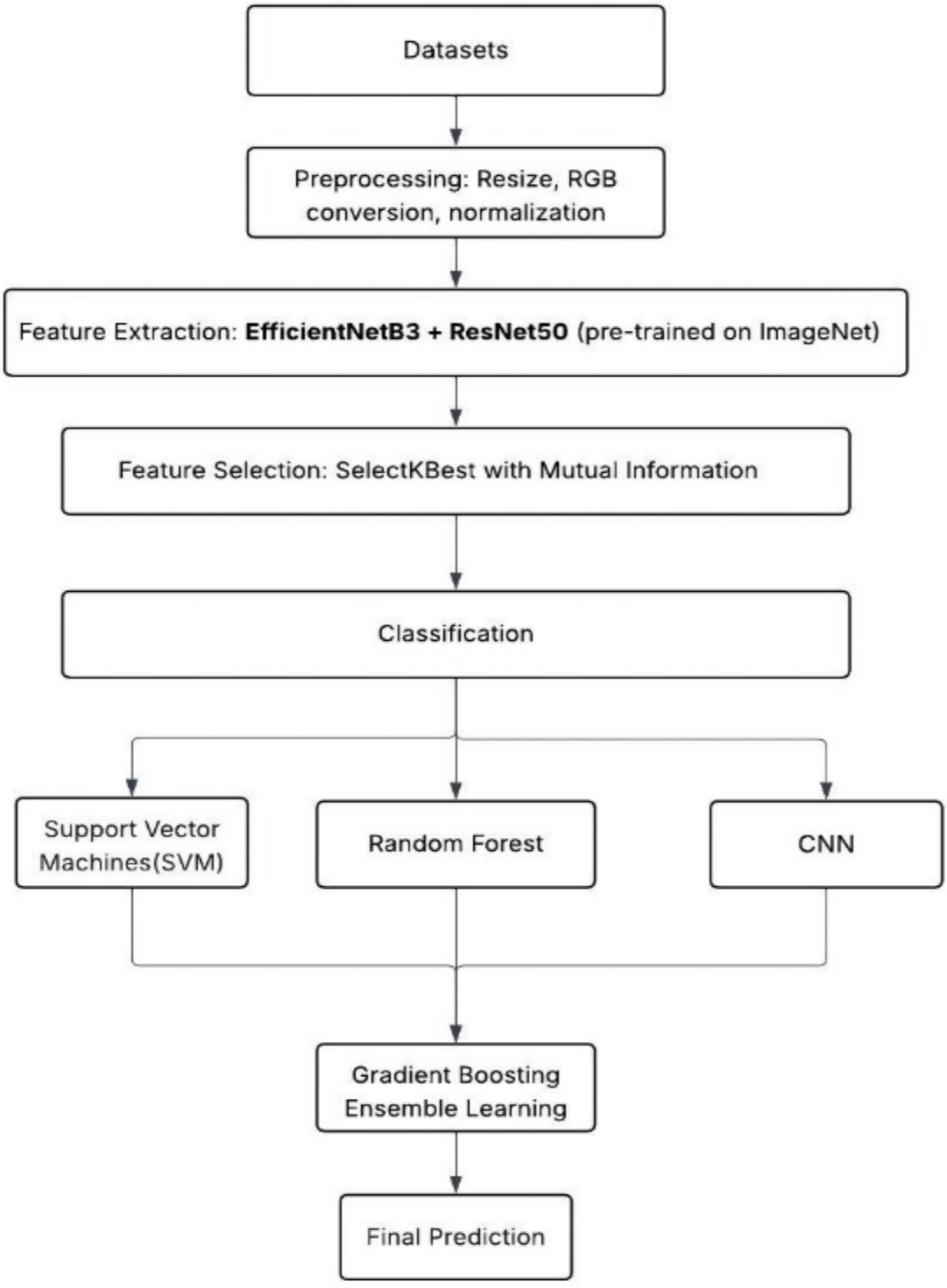
4. Ensemble Learning using Gradient Boosting

To increase prediction performance and eliminate model bias, the Soft Voting approach is used to combine the outputs of CNN, SVM, and RF classifiers. This ensemble technique averages the projected class probabilities from each model and uses the class with the highest mean probability as the final forecast.

Soft Voting improves classification accuracy and balance by exploiting the strengths of all three classifiers. This method helps to smooth out individual model flaws and improves overall system reliability.

These integrated algorithms constitute a strong and scalable pipeline for skin cancer detection, ensuring excellent accuracy and dependability when identifying skin lesions from medical pictures using machine learning techniques.

3.3 WORKFLOW DIAGRAM



[Figure 1: The SVM-RF-CNN Gradient Boosting Model]

1.3 MODEL DESCRIPTION

A. DATASET DESCRPTION:

To classify skin lesions as malignant (cancerous) or benign (non-cancerous) using features derived from images, clinical metadata, or both. The dataset is commonly used in machine learning and medical imaging research to train models for early skin cancer detection.

The dataset may contain one or more of the following:

- Images
    1. Dermoscopic images of skin lesions.
    2. High-resolution, often labeled with diagnosis.
    3. Common formats: JPG, PNG.
- Labels
    a) Typically, Binary Classification:
        - Malignant = 1
        - Benign = 0

Load Dataset Path:

```
dataset_path = r"C:\Users\lenovo\Downloads\archive (4)\data"
train_dir = os.path.join(dataset_path, "train")
test_dir = os.path.join(dataset_path, "test")
categories = ["benign", "malignant"]
```

Purpose:

- Establishes the testing and training directories' route.
- Specifies the labels for classification (malignant = 1, benign = 0).

B. PREPROCESSING:

II. **RESIZING IMAGE**: Resizing an image involves changing its dimensions (height and width) to make it smaller or larger. Resizing is a critical preprocessing step in machine learning, particularly for image classification tasks such as skin cancer detection.

Most machine learning models, including Convolutional Neural Networks (CNNs), require input

images of a specific size. However, images in datasets typically come in a variety of resolutions and sizes. To ensure consistency, we resize all input images to a standard dimension, such as 224×224 pixels, which is commonly used for models like ResNet and VGG.

Resizing is beneficial:

- Lessen the processing load (smaller photos process more quickly).
- Make sure the model receives consistent input.
- Pre-trained models that anticipate particular input sizes should remain compatible.

Maintaining the aspect ratio or using appropriate interpolation techniques (such as nearest-neighbour or bilinear) when resizing is crucial to preserving crucial features, particularly for medical photos such as skin lesions.

To sum up, resizing is a straightforward yet crucial step that gets raw photos ready for model training, which enhances performance and predictability.

III. **RGB CONVERSATION**: RGB conversion is the transformation of an image to the RGB color model, Red, Green, and Blue. They are the three basic light colors, and they are combined to produce all other colors in digital images.

Every pixel in an RGB image consists of three values:

Red intensity (0-255)

Green intensity (0-255)

Blue intensity (0-255)

By mixing these three values, any color can be described. White, for instance, is (255,255,255) and black is (0,0,0).

RGB conversion is significant in image processing and machine learning because:

There are some images available in grayscale or other color modes (such as BGR or CMYK), which must be converted to RGB to maintain uniformity.

Most pre-trained models (such as CNNs) are trained on RGB images (for example, ImageNet), and therefore input images need to be compatible with this format.

RGB images enables more efficient color analysis, such as in skin lesion detection where color distinctions are significant.

IV. NORMALIZATION: Normalization is a data preprocessing method to scale input values between a predetermined range, normally 0 to 1 or -1 to 1.

Normalization in image processing and machine learning enhances model accuracy, training time, and stability.

For image, pixel values are usually normalization. Pixel values are normally between 0 and 255 for every color channel (Red, Green, Blue). In normalizing, a pixel value is divided by 255 to place it within the 0-1 range. For instance, a pixel value of 128 is normalized to 128/255 ~ 0.50.

Why is Normalization Necessary?

It keeps all the input features at the same scale, which assists the model to learn better.

It shortens the training time by enhancing gradient descent performance.

It assists in preventing bias towards larger values, particularly in neural networks.

In a few instances, normalization is also performed with mean and standard deviation, particularly when applying pre-trained models. It normalizes the data with a mean of 0 and standard deviation of 1. Briefly, RGB conversion guarantees uniform color structure in all the images, which allows them to be used for analysis and model training.

Code:

```
def load_images(directory, preprocess_func):
    images, labels = [], []
    for category in categories:
        label = 0 if category == "benign" else 1
        folder = os.path.join(directory, category)
        for   img_name   in   tqdm(os.listdir(folder),   desc=f"Loading
{category}"):
            path = os.path.join(folder, img_name)
            img = cv2.imread(path)
            if img is not None:
                img = cv2.resize(img, (224, 224))
                img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
                img = preprocess_func(img.astype(np.float32))
                images.append(img)
                labels.append(label)
    return np.array(images), np.array(labels)
```

Purpose:

- Loads pictures from every folder in the class.
- Steps in preprocessing:
    i. CNNs need that you resize to 224x224.
    ii. Change BGR to RGB.
    iii. Use the efficientnet_preprocess or resnet_preprocess model-specific preprocessing.
- Images are loaded and preprocessed twice:
    i. For the input (X_train_e, X_test_e) of EfficientNetB3
    ii. Regarding the input (X_train_r, X_test_r) for ResNet50
    iii. employs the same names for both (y_train, y_test).

C. FEATURE EXTRACTION: Feature extraction is activity of discovering and choosing significant data from original information, like images, to enable a machine learning algorithm to make more accurate predictions.

For images processing, features may encompass:

- Edges
- Colors
- Texture
- Shapes
- Patterns

As a case, in the detection of skin cancer, feature extraction enables the identification of critical visual features of a skin lesion, for instance, irregular edges, abnormal colors, or asymmetry.

Features extraction can also be performed manually with the help of image processing methods or automatically with machine learning algorithms such as Convolutional Neural Network (CNNs). CNNs can learn to extract features from raw images at training time, layer by layer.

The aim of features extraction is to decrease the complexity of data while preserving the most valuable information, so that it becomes simpler and quicker for the model to learn and make correct predictions.

i. EfficientNetB3: EfficientNetB3 is a strong deep learning model applied in machine learning image classification. It is a member of the EfficientNet family, created by Google,

and is built to deliver high accuracy with reduced speed and resource utilization over the older models such as ResNet or VGG.

EfficientNetB3 employs a method known as compound scaling, along with the input resolution in a proportionate manner. This renders it more efficient and accurate for object recognition, medical image analysis, or detecting skin cancer.

In machine learning, EfficientNetB3 is usually applied with transfer learning- it's pre-trained on large datasets such as ImageNet, and then further trained using specific data (e.g., skin lesion images). This enables the model to learn more quickly and work correctly even with small datasets.

EfficientNetB3 is a suitable option when accuracy and computational efficiency matter.

ii. RasNet50: ResNet50 is a machine learning model that is used for both image classification and feature extraction. It is part of Microsoft's ResNet (Residual Network) family of models. The "50" in ResNet50 means that it is 50 layers deep, allowing for complex patterns to be learned in images.

ResNet introduced the idea of residual connections (or skip connections) for deep neural networks. This type of connection facilitates the training process by allowing the model to "skip" some layers, helping to avoid the problem of vanishing gradients.

ResNet50 is pre-trained on datasets with millions of images, like ImageNet. As a result, ResNet50 is a common choice for transfer learning, including applications like medical image analysis, object detection, and more. In skin cancer detection, ResNet50 is selected to extract important dermoscopic image features in order to differentiate between benign or malignant lesions.

Overall, ResNet50 is well known in image-related tasks for its accuracy, stability, and efficiency.

The project uses pre-trained EfficientNetB3 and ResNet50 models to extract important features from images.

The main goal is to classify skin lesion images as benign or malignant. This is done by extracting deep features using CNNs and using them in machine learning models for classification.

Output of Feature Extraction:

EfficientNetB3 gives 1536 features per image

ResNet50 gives 2048 features per image

Combined total: 3584 features per image

Code:

```
effnet = EfficientNetB3(weights="imagenet", include_top=False, pooling='avg', input_shape=(224, 224, 3))
resnet = ResNet50(weights="imagenet", include_top=False, pooling='avg', input_shape=(224, 224, 3))
model1 = Model(inputs=effnet.input, outputs=effnet.output)
model2 = Model(inputs=resnet.input, outputs=resnet.output)

features_effnet_train = model1.predict(X_train_e, verbose=1)
features_effnet_test = model1.predict(X_test_e, verbose=1)
features_resnet_train = model2.predict(X_train_r, verbose=1)
features_resnet_test = model2.predict(X_test_r, verbose=1)
combined_train = np.concatenate([features_effnet_train, features_resnet_train], axis=1)
combined_test = np.concatenate([features_effnet_test, features_resnet_test], axis=1)
```

Purpose:

- loads the final classification layer (include_top=False) but not EfficientNetB3 or ResNet50.
- obtains feature vectors by using global average pooling (pooling='avg').
- uses both models to extract deep features from training and test photos.
- combines feature vectors from ResNet50 (2048 dims) and EfficientNetB3 (1536 dims).
- Each image's final feature vector has 3584 dimensions.

D. **FEATURE SELECTION:** Feature selection is the process of selecting the most essential and relevant input parameters (features) from a dataset to help predict the outcome. In the areas of machine learning and data science, choosing the proper features is a vital step toward creating an efficient and accurate model. It reduces the model's complexity, improves its performance, and prevents overfitting.

A dataset for skin cancer identification may include a wide variety of features, such as pixel values, color intensity, texture information, edge patterns, or clinical data such as the age and location of the lesion. However, not all of these features are applicable. Some may be irrelevant or redundant, reducing the model's performance.

Feature selection contributes to:

1. Improve Accuracy: By deleting unimportant information, the model may concentrate on the data's most important patterns.
2. Reduce Overfitting: Having fewer irrelevant features decreases the likelihood of the model learning noise from the data.
3. Reduce Training Time: Using smaller feature sets requires less calculation, which speeds up the training process.
4. Simplify the Model: A simpler model is easier to comprehend and interpret, which is critical in medical applications.

There are three major categories of feature selection techniques:

1. Filter Methods: These employ statistical approaches to score each feature independently of the model (for example, correlation and chi-square tests). Features with the highest scores are chosen.
2. Wrapper Methods: Wrapper methods analyze feature combinations by training and testing a model on them. Though more precise, they are computationally expensive.
3. Embedded Methods: These choose features throughout the training phase. For example, decision trees and LASSO regression choose important characteristics automatically.

Feature selection is particularly important for machine learning algorithms such as SVM and Random Forest, which rely on structured input data. Deep learning models, such as CNNs, extract and learn features from raw images without the need for manual selection.

In skin cancer detection projects, proper feature selection allows the model to learn more from the

data and enhances its capacity to distinguish between malignant and benign lesions. It guarantees that just the most relevant aspects of the skin scans are used, resulting in more accurate and reliable diagnostic predictions.

SelectKBest: SelectKBest is a feature selection approach for machine learning that selects the top k most relevant features from a dataset. It operates by scoring each feature independently using a statistical test and then picking the top k features with the highest scores. Mutual Information (MI) is one type of scoring mechanism.

Mutual Information is the amount of information that one variable (feature) offers about another variable (target label). In layman's words, it determines how much knowing a feature helps forecast the class label (for example, malignant or benign in skin cancer diagnosis). A higher mutual information score indicates a more robust association between the feature and the target.

In this skin cancer diagnosis experiment, SelectKBest with Mutual Information is used to identify the most essential features retrieved from dermoscopic images or information. This helps to remove irrelevant or duplicated data while improving model performance. It is especially useful for algorithms such as SVM and Random Forest, which perform better when trained using high-quality, relevant features.

Using this strategy, the model concentrates solely on the features that are most valuable in discriminating between malignant and benign lesions, making the prediction process faster, more accurate, and more dependable.

Code:

```
selector = SelectKBest(score_func=mutual_info_classif, k=300)
X_train_selected = selector.fit_transform(combined_train, y_train)
X_test_selected = selector.transform(combined_test)
```

Purpose:

- Applies SelectKBest using mutual information to select top 300 features.
- Keeps only the most relevant features for classification.

E. CLASSIFICATION: In this project we used three classifications that are Support Vector Machine (SVM), Random Forest (RF), Convolutional Neural Networks (CNNs).

1. **Support Vector Machine (SVM):** The Support Vector Machine (SVM) is a strong supervised machine learning technique used for classification and regression tasks. In classification, SVM seeks the optimal decision boundary (also known as a hyperplane) that separates data points into distinct classes. It works well with both linear and nonlinear data and is especially useful for binary classification tasks, such as discriminating between malignant (cancerous) and benign (non-cancerous) skin lesions**.**

   How SVMs Work?

   SVM hunts for the hyperplane with the greatest margin—the greatest feasible distance between data points from both classes and the decision border. Support vectors are the data points nearest to the hyperplane and play an important role in determining the border. For non-linear situations, SVM employs a technique known as the kernel trick to transform the data into a higher-dimensional space that is linearly separable. Common kernels include linear, polynomial, and radial basis functions (RBFs).

   Application of SVM in Skin Cancer Detection

   In this study, SVM is employed as one of the classification algorithms for skin cancer detection. The primary goal is to divide photographs of skin lesions into two categories:

   - Malignant (1)
   - Benign (0)

   The process consists of the following steps:

   - Feature Extraction: The photos are the first processed to extract CNN features such as texture, color, form, and depth.
   - Feature Selection: Techniques such as SelectKBest with Mutual Information are used to select the most relevant features for classification.
   - Model Training: The selected features and labels are supplied into the SVM model. Using the training data, the model learns malignant and benign cases.
   - Testing and Evaluation: The trained SVM is then run on previously unseen data to assess its performance using measures such as accuracy, precision, recall, and F1-Score.

   SVM has good classification accuracy, particularly in binary situations, making it an ideal choice for this project.

SVM is an appropriate and dependable technique for this project because of its strong theoretical base and performance in binary classification. It complements other models such as CNN and Random Forest by offering an additional approach for assessing the accuracy and robustness of skin cancer diagnosis.

Code:

```
from sklearn.svm import SVC
from sklearn.metrics import classification_report, accuracy_score,
confusion_matrix, ConfusionMatrixDisplay, roc_curve, auc
import matplotlib.pyplot as plt
# Train SVM
svm_model = SVC(kernel='rbf', probability=True)
svm_model.fit(X_train, y_train)
# Predict
y_pred_svm = svm_model.predict(X_test)
# Evaluation
print("SVM Classification Report:")
print(classification_report(y_test,y_pred_svm,target_names=["Benign",
"Malign"]))
print("SVM Accuracy:", accuracy_score(y_test, y_pred_svm))
# Confusion Matrix Plot
conf_mat = confusion_matrix(y_test, y_pred_svm)
disp=ConfusionMatrixDisplay(confusion_matrix=conf_mat,
display_labels=["Benign", "Malignant"])
disp.plot(cmap=plt.cm.Blues)
plt.title("SVM - Confusion Matrix")
plt.show()
# --- AUC/ROC Curve ---
# Get probability scores for the positive class (Malignant)
y_prob_svm = svm_model.predict_proba(X_test)[:, 1]
# Compute ROC curve and AUC
fpr, tpr, _ = roc_curve(y_test, y_prob_svm)
roc_auc = auc(fpr, tpr)
# Plot ROC curve
plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC Curve (AUC =
{roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
```

```
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("SVM - ROC Curve")
plt.legend(loc="lower right")
plt.grid(True)
plt.show()
```

2. **Random Forest (RF):** Random Forest (RF) is an extremely powerful and widely used ensemble machine learning method for both classification and regression. RF works by creating a "forest" of decision trees, each of which represents is a subset of the input that has been trained randomly. The ultimate prediction comes from the majority vote of all trees or the mean of all trees.

   How Random Forests Work?

   Random Forest algorithm unites bagging (Bootstrap Aggregation) and random feature selection ideas. Every tree in the forest is trained on a different random sample of the dataset, and only a random subset of features is taken into account at every split. With this randomness, the model becomes more robust, less susceptible to overfitting, and more flexible to new data.

   Application of RF for Detection of Skin Cancer.

   In the Skin Cancer Detection experiment, Random Forest is employed to classify skin lesions as:

   - Malignant (1): Cancerous Lesions.
   - Benign (0): Non-Cancerous Lesions.

   The radio frequency (RF) is utilized in the project at the following steps:

   - Feature Extraction: Color, Texture, and Form are extracted from images through traditional image processing or deep learning networks like CNNs.
   - Feature Selection: SelectKBest and Mutual information tools help to determine the most salient features for classification.
   - Model Training: The RF model is trained on the chosen features and their respective labels. Each tree learns something distinct from the data.

- Prediction and Testing: The trained model is tested against previously unseen data to see how accurately it can categorize new skin lesion images.

Random Forest is a reliable and interpretable algorithm for the skin cancer diagnostic project. It is complementary to deep learning models by delivering great accuracy while being less complex and quicker in training. It is particularly convenient when dealing with manually selected or gathered features since it enables stronger prediction that help in early diagnosis of skin cancer.

Code:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score,
confusion_matrix, ConfusionMatrixDisplay, roc_curve, auc
import matplotlib.pyplot as plt
# Train Random Forest
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)
# Predict
y_pred_rf = rf_model.predict(X_test)
# Evaluation Metrics
print("Random Forest Classification Report:")
print(classification_report(y_test, y_pred_rf, target_names=["Benign",
"Malignant"]))
print("Random Forest Accuracy:", accuracy_score(y_test, y_pred_rf))

# Confusion Matrix Plot
conf_mat = confusion_matrix(y_test, y_pred_rf)
disp=ConfusionMatrixDisplay(confusion_matrix=conf_mat,
display_labels=["Benign", "Malignant"])
disp.plot(cmap=plt.cm.Blues)
plt.title("Random Forest - Confusion Matrix")
plt.show()
# Print Confusion Matrix
print("Confusion Matrix:")
print(conf_mat)
# --- AUC/ROC Curve ---
# Get probability scores for the positive class (Malignant)
```

```
y_prob_rf = rf_model.predict_proba(X_test)[:, 1]
# Compute ROC curve and AUC
fpr, tpr, _ = roc_curve(y_test, y_prob_rf)
roc_auc = auc(fpr, tpr)
# Plot ROC curve
plt.figure()
plt.plot(fpr, tpr, color='green', lw=2, label=f'ROC Curve (AUC =
{roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("Random Forest - ROC Curve")
plt.legend(loc="lower right")
plt.grid(True)
plt.show()
```

3. **Convolutional Neural Networks (CNNs):** Convolutional Neural Networks (CNNs) are a type of deep learning algorithm that works very well with visual data like images. They learn spatial hierarchies of features automatically with the help of multiple layers like convolutional layers, pooling layers, and fully connected layers. CNNs are trained to identify patterns, texture, and structure in images, and hence they are better suited for image classification, object detection, and segmentation tasks.

How CNN Works:

- Convolutional Layer: Adds filters to the input image to extract features such as edges, curves, and textures.
- Activation Function (ReLU): Adds non-linearity to the model so that it can learn complex patterns.
- Pooling Layer: Shrinks the spatial dimensions of the feature maps, keeping key information while decreasing computation.

- Fully Connected Layer: Translates the high-level features and makes the classification.
- Output Layer: Makes the final prediction, i.e., whether the lesion is benign or malignant.

In this skin cancer detection task, CNNs are employed as one of the initial classifiers to determine whether a specified skin lesion is malignant (cancer) or benign (non-cancer). Following preprocessing and feature extraction through pre-trained models such as EfficientNetB3 and ResNet50, the features can be classified using a CNN. Or, the CNN can also be trained end-to-end on the processed image data to identify the patterns involved with various skin lesion types.

Code:

```
import os
import numpy as np
import cv2
from tqdm import tqdm
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report, confusion_matrix,
accuracy_score, ConfusionMatrixDisplay, roc_curve, auc
from sklearn.utils.class_weight import compute_class_weight
from tensorflow.keras.applications import EfficientNetB3
from tensorflow.keras.applications.efficientnet import
preprocess_input
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Dense, Dropout,
GlobalAveragePooling2D
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.callbacks import ReduceLROnPlateau,
EarlyStopping, ModelCheckpoint

# === Paths ===
dataset_path = r"C:\Users\lenovo\Downloads\archive (4)\data"
```

```
train_dir = os.path.join(dataset_path, "train")
test_dir = os.path.join(dataset_path, "test")
categories = ["benign", "malignant"]

# === Load Images ===
def load_images(directory, preprocess_func, img_size=(300, 300)):
    images, labels = [], []
    for category in categories:
        label = 0 if category == "benign" else 1
        folder = os.path.join(directory, category)
        for   img_name   in   tqdm(os.listdir(folder),   desc=f"Loading
{category}"):
            path = os.path.join(folder, img_name)
            img = cv2.imread(path)
            if img is not None:
                img = cv2.resize(img, img_size)
                img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
                img = preprocess_func(img.astype(np.float32))
                images.append(img)
                labels.append(label)
    return np.array(images), np.array(labels)

# === Load and preprocess data ===
X_train, y_train = load_images(train_dir, preprocess_input)
X_test, y_test = load_images(test_dir, preprocess_input)
y_train_cat = to_categorical(y_train, num_classes=2)
y_test_cat = to_categorical(y_test, num_classes=2)

# === Compute class weights ===
class_weights=compute_class_weight(class_weight='balanced',classes=np
.unique(y_train), y=y_train)
class_weight_dict = dict(enumerate(class_weights))

# === Data Augmentation ===
datagen = ImageDataGenerator(
    rotation_range=30,
    width_shift_range=0.15,
    height_shift_range=0.15,
```

```
    zoom_range=0.25,
    horizontal_flip=True,
    vertical_flip=True,
    brightness_range=[0.8, 1.2])
datagen.fit(X_train)
# === Build CNN using EfficientNetB3 ===
base_model=EfficientNetB3(weights='imagenet',include_top=False,input_
shape=(300, 300, 3))
base_model.trainable = False

x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(512, activation='relu')(x)
x = Dropout(0.5)(x)
x = Dense(128, activation='relu')(x)
x = Dropout(0.3)(x)
output = Dense(2, activation='softmax')(x)
model = Model(inputs=base_model.input, outputs=output)
model.compile(optimizer=Adam(learning_rate=0.0001),
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# === Callbacks ===
lr_reducer    =    ReduceLROnPlateau(monitor='val_loss',    factor=0.3,
patience=3, verbose=1)
early_stopper=EarlyStopping(monitor='val_loss',patience=5,restore_bes
t_weights=True, verbose=1)
checkpoint=ModelCheckpoint("best_model.h5",monitor='val_accuracy',sav
e_best_only=True, verbose=1)

# === Train base model ===
model.fit(datagen.flow(X_train, y_train_cat, batch_size=32),
          validation_data=(X_test, y_test_cat),
          epochs=25,
          class_weight=class_weight_dict,
          callbacks=[lr_reducer, early_stopper, checkpoint],
          verbose=1)
```

```
# === Fine-Tune (Unfreeze last 30 layers) ===
base_model.trainable = True
for layer in base_model.layers[:-30]:
    layer.trainable = False

model.compile(optimizer=Adam(learning_rate=1e-5),
              loss='categorical_crossentropy',
              metrics=['accuracy'])

model.fit(datagen.flow(X_train, y_train_cat, batch_size=32),
          validation_data=(X_test, y_test_cat),
          epochs=10,
          class_weight=class_weight_dict,
          callbacks=[lr_reducer, early_stopper],
          verbose=1)

# === Predict ===
y_pred_probs = model.predict(X_test)
y_pred_labels = np.argmax(y_pred_probs, axis=1)
# === Accuracy & Report ===
print("\nFinal      CNN      Accuracy      with      EfficientNetB3:",
accuracy_score(y_test, y_pred_labels))
print("\nClassification Report:")
print(classification_report(y_test,                    y_pred_labels,
target_names=["Benign", "Malignant"]))
# === Confusion Matrix Plot ===
conf_mat = confusion_matrix(y_test, y_pred_labels)
disp        =        ConfusionMatrixDisplay(confusion_matrix=conf_mat,
display_labels=["Benign", "Malignant"])
disp.plot(cmap=plt.cm.Blues)
plt.title("EfficientNetB3 - Confusion Matrix")
plt.grid(False)
plt.tight_layout()
plt.show()

# === AUC/ROC Curve ===
y_prob_cnn = y_pred_probs[:, 1]  # Probability for class 'Malignant'
fpr, tpr, _ = roc_curve(y_test, y_prob_cnn)
```

```
roc_auc = auc(fpr, tpr)

plt.figure()
plt.plot(fpr, tpr, color='purple', lw=2, label=f'ROC Curve (AUC =
{roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("EfficientNetB3 - ROC Curve")
plt.legend(loc="lower right")
plt.grid(True)
plt.show()
```

F. ENSEMBLE TECHNIQUE: Ensemble technique is a machine learning approach that combines multiple models to solve a particular problem and improve the overall performance of prediction. The core idea is that a group of weak learners can come together to form a strong learner with high accuracy and robustness. By aggregating the predictions of several models, ensemble methods can reduce errors due to bias, variance, and noise.

There are three main types of ensemble techniques:

- Bagging (Bootstrap Aggregating): This method involves training multiple models independently using different subsets of the training data (created via random sampling with replacement). Each model makes a prediction, and the final result is typically obtained by majority voting (for classification) or averaging (for regression). A popular example of a bagging algorithm is Random Forest, which combines multiple decision trees.
- Boosting: Boosting trains models sequentially, where each model tries to correct the errors made by the previous one. The final prediction is a weighted sum of all models' outputs. Algorithms like AdaBoost, Gradient Boosting, and XGBoost are well-known boosting techniques. Boosting is effective at reducing bias and can produce strong predictive models, but it can be more prone to overfitting if not properly tuned.
- Voting: In voting ensembles, different models (which could be of different types) are trained on the same dataset. For classification, predictions from each model are collected and the final prediction is made based on a majority vote (hard voting) or by averaging the

predicted probabilities (soft voting). Soft voting usually yields better results when the models are well-calibrated.

In skin cancer detection projects, ensemble techniques enhance prediction accuracy by combining the strengths of models like CNN (good at feature learning from images), SVM (effective for binary classification), and RF (robust to noise and overfitting).

The Gradient Boosting ensemble used in our project improves the model's generalization ability and reduces the likelihood of errors from any single classifier, leading to more reliable diagnostic decisions.

1. Gradient Boosting Ensemble: Gradient Boosting is a strong ensemble machine learning algorithm that can be applied to both regression and classification problems. It constructs a robust prediction model by averaging the predictions of a large number of weak learner models, typically decision trees, sequentially. The concept is to concentrate on minimizing the mistake made by the earlier models in each step.

   In Gradient Boosting, the model is learned iteratively. First, a basic model (usually a shallow decision tree) is learned from the data. The initial model's errors or residuals. This is done several times for a set number of iterations or until the error becomes less than a certain threshold. Every successive model shrinks the error of the combined model by fitting to the gradient (the reason it's called "gradient" boosting) of the loss function employed, for example, mean squared error in regression or log loss in classification.

   The ultimate prediction is generated by averaging the predictions from every individual weak learner, typically through a weighted sum. Learning rate, which determines how significantly each new model contributes to the overall prediction, is a very important hyperparameter.

   Gradient Boosting is highly popular due to its high accuracy, support for many types of data, and overfitting robustness (particularly with methods such as shrinkage and subsampling). Some highly popular implementations are XBBoost, LightGBM, and CatBoost, which provide high-performance, scalable implementations of the gradient boosting algorithm. Gradient Boosting finds particular application in difficult datasets and competitions such as those found on Kaggle.

Code:

```
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.metrics import accuracy_score, roc_auc_score,
classification_report, confusion_matrix, ConfusionMatrixDisplay,
RocCurveDisplay
import matplotlib.pyplot as plt
import numpy as np

# === STEP 1: Generate model outputs ===
svm_probs = svm_model.predict_proba(X_test_selected)[:, 1]
rf_probs = rf_model.predict_proba(X_test_selected)[:, 1]
cnn_probs = model.predict(X_test_e)[:, 1]

# === STEP 2: Stack probabilities into new feature space ===
X_meta = np.vstack((svm_probs, rf_probs, cnn_probs)).T  # Shape:
(num_samples, 3)

# === STEP 3: Train Gradient Boosting as meta-classifier ===
meta_model = GradientBoostingClassifier(n_estimators=200,
learning_rate=0.05, max_depth=3, random_state=42)
meta_model.fit(X_meta, y_test)

# === STEP 4: Final prediction using meta-model ===
ensemble_preds = meta_model.predict(X_meta)
ensemble_probs = meta_model.predict_proba(X_meta)[:, 1]

# === STEP 5: Evaluation ===
acc = accuracy_score(y_test, ensemble_preds)
auc_score = roc_auc_score(y_test, ensemble_probs)

print(f"\n Improved Stacked Ensemble Accuracy: {acc * 100:.2f}%")
print(f" Improved AUC Score: {auc_score:.4f}")
print("Classification Report:")
print(classification_report(y_test, ensemble_preds,
target_names=["Benign", "Malignant"]))

# === Confusion Matrix ===
cm = confusion_matrix(y_test, ensemble_preds)
```

```
disp = ConfusionMatrixDisplay(confusion_matrix=cm,
display_labels=["Benign", "Malignant"])
disp.plot(cmap=plt.cm.Blues)
plt.title("Improved Stacked Ensemble - Confusion Matrix")
plt.grid(False)
plt.tight_layout()
plt.show()
# === ROC Curve ===
RocCurveDisplay.from_predictions(y_test, ensemble_probs,
name="Improved Stacked Ensemble")
plt.plot([0, 1], [0, 1], 'k--')
plt.title("ROC Curve - Improved Stacked Ensemble")
plt.grid(True)
plt.show()
```

CHAPTER 4

EXPERIMENTATION RESULT AND DISUSSION

4.1 DATASET DESCRIPTION

The name of the dataset that we used in this project is Skin Cancer: Malignant vs. Benign.

The primary purpose of this dataset is to assist with the classification of skin lesions as either malignant (cancerous) or benign (non-cancerous). This classification system is important for the early detection and diagnosis of skin cancer, specifically melanoma, which can be deadly when not detected early. It is a frequently used dataset in machine learning and medical imaging for developing, validating, and testing prediction models that can aid dermatologists and healthcare professionals in the proper diagnosis and treatment of skin lesions.

The dataset generally has:

1. Images:

The dataset includes dermoscopic images of skin lesions, which are images taken using dermoscopy. Dermoscopy is a type of optical measurement of skin lesions that uses a technology similar to a microscope in that it allows for deep and high-resolution visuals of the skin surface.

Medical experts often label these images with a known diagnosis.

The images would generally be stored as JPG or PNG formats (sizes that can be used as input to deep learning models).

Images vary in quality, size, and skin lesion types, providing an extensive sampling of data for training your model.

2. Labels:

Every image is assigned with a binary classification label, representing the type of lesion.

- Malignant (cancer) lesions are assigned as 1.
- Benign (non-cancer) lesions are assigned as 0.

Some more advanced datasets may also contain metadata such as patient age, lesion location or type to provide additional analysis.

This dataset is the basis for building automated tools that can learn visual patterns and facilitate early, accurate diagnosis for skin cancer, which in turn would improve patient outcomes and reduce pressure on healthcare systems.

4.2 DATA PREPROCESSING

1. **RESIZING IMAGE**: Resizing an image involves changing its dimensions (height and width) to make it smaller or larger. Resizing is a critical preprocessing step in machine learning, particularly for image classification tasks such as skin cancer detection. Most machine learning models, including Convolutional Neural Networks (CNNs), require input images of a specific size. However, images in datasets typically come in a variety of resolutions and sizes. To ensure consistency, we resize all input images to a standard dimension, such as 224×224 pixels, which is commonly used for models like ResNet and VGG.

   Resizing is beneficial:

   - Lessen the processing load (smaller photos process more quickly).
   - Make sure the model receives consistent input.
   - Pre-trained models that anticipate particular input sizes should remain compatible.

   Maintaining the aspect ratio or using appropriate interpolation techniques (such as nearest-neighbour or bilinear) when resizing is crucial to preserving crucial features, particularly for medical photos such as skin lesions.

   To sum up, resizing is a straightforward yet crucial step that gets raw photos ready for model training, which enhances performance and predictability.

2. **RGB CONVERSATION**: RGB conversion is the transformation of an image to the RGB color model, Red, Green, and Blue. They are the three basic light colors, and they are combined to produce all other colors in digital images.

   Every pixel in an RGB image consists of three values:

   - Red intensity (0-255)
   - Green intensity (0-255)
   - Blue intensity (0-255)

   By mixing these three values, any color can be described. White, for instance, is (255,255,255) and black is (0,0,0).

   RGB conversion is significant in image processing and machine learning because:

There are some images available in grayscale or other color modes (such as BGR or CMYK), which must be converted to RGB to maintain uniformity.

Most pre-trained models (such as CNNs) are trained on RGB images (for example, ImageNet), and therefore input images need to be compatible with this format.

RGB images enables more efficient color analysis, such as in skin lesion detection where color distinctions are significant.

3. **NORMALIZATION:** Normalization is a data preprocessing method to scale input values between a predetermined range, normally 0 to 1 or -1 to 1.

Normalization in image processing and machine learning enhances model accuracy, training time, and stability.

For image, pixel values are usually normalization. Pixel values are normally between 0 and 255 for every color channel (Red, Green, Blue). In normalizing, a pixel value is divided by 255 to place it within the 0-1 range. For instance, a pixel value of 128 is normalized to 128/255 ~ 0.50.

Why is Normalization Necessary?

- It keeps all the input features at the same scale, which assists the model to learn better.
- It shortens the training time by enhancing gradient descent performance.
- It assists in preventing bias towards larger values, particularly in neural networks.

In a few instances, normalization is also performed with mean and standard deviation, particularly when applying pre-trained models. It normalizes the data with a mean of 0 and standard deviation of 1. Briefly, RGB conversion guarantees uniform color structure in all the images, which allows them to be used for analysis and model training.

4.3 TRAINING AND TESTING OF THE MODEL

Teaching a model to spot patterns in data allows it to make accurate predictions. This skin cancer detection project employs three algorithms: Convolutional Neural Network (CNN), Random Forest (RF), Support Vector Machine (SVM).

CNN is a deep learning model created specifically for image data. During training, CNN automatically learns key properties such as color, shape, and texture from skin lesion photos. Pattern extraction and learning are accomplished by the use of layers such as convolution,

pooling, and fully linked layers. CNN is particularly effective for visual recognition tasks such as this.

Random Forest is an ensemble machine learning approach. During training, it generates a large number of decision trees, each of which votes on the categorization outcome. The model is trained on features manually retrieved from photos, such as color histograms and texture patterns. SVM is a supervised learning technique for classification. It determines the optimal border (hyperplane) between malignant and benign lesions in the feature space. SVM, like Random Forest, works with extracted features instead of raw pictures.

All three models are trained on labelled data (pictures with known diagnoses), allowing them to learn and increase their accuracy in classifying skin cancer.

Testing the model is the act of evaluating its performance with previously unknown data—data that was not utilized during the training phase. In this skin cancer diagnosis research, three algorithms are used: Convolutional Neural Network (CNN), Support Vector Machine (SVM), and Random Forest (RF).

The CNN model is tested by feeding additional dermoscopic images through the trained network. The algorithm processes these images using its learnt filters and predicts whether the lesion is cancerous or benign. The CNN model's testing performance is often assessed using metrics including as accuracy, precision, recall, F1-score, and a confusion matrix to determine its ability to accurately categorize lesions.

SVM and Random Forest process images first to extract significant features like color intensity, texture, and shape. The retrieved characteristics are subsequently fed into trained SVM and RF models. SVM predicts the class by determining where a data point falls on the decision border, whereas Random Forest predicts using majority voting for several decision trees. During testing, the emphasis is on how well each model generalizes—that is, how accurately it performs on previously unknown images. Poor testing results may suggest overfitting, in which the model memorizes training data rather than learns broad patterns.

By comparing the results of CNN, SVM, and RF on the same test dataset, the best performing algorithm can be selected for this project is Support Vector Machine (SVM).

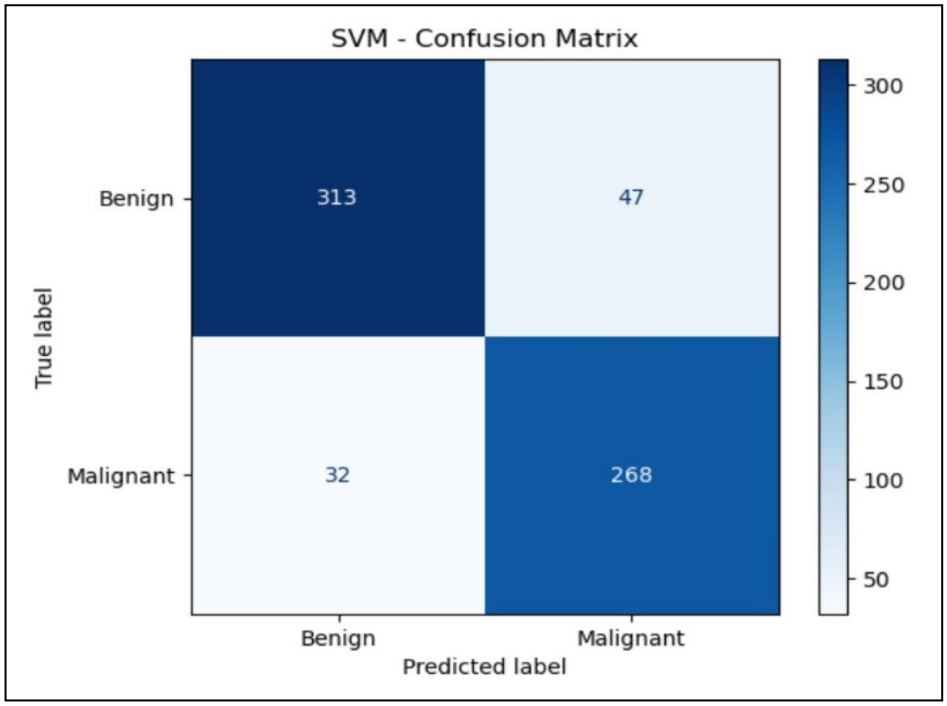
4.3 RESULTS AND ANALYSIS

Support Vector Machine (SVM):

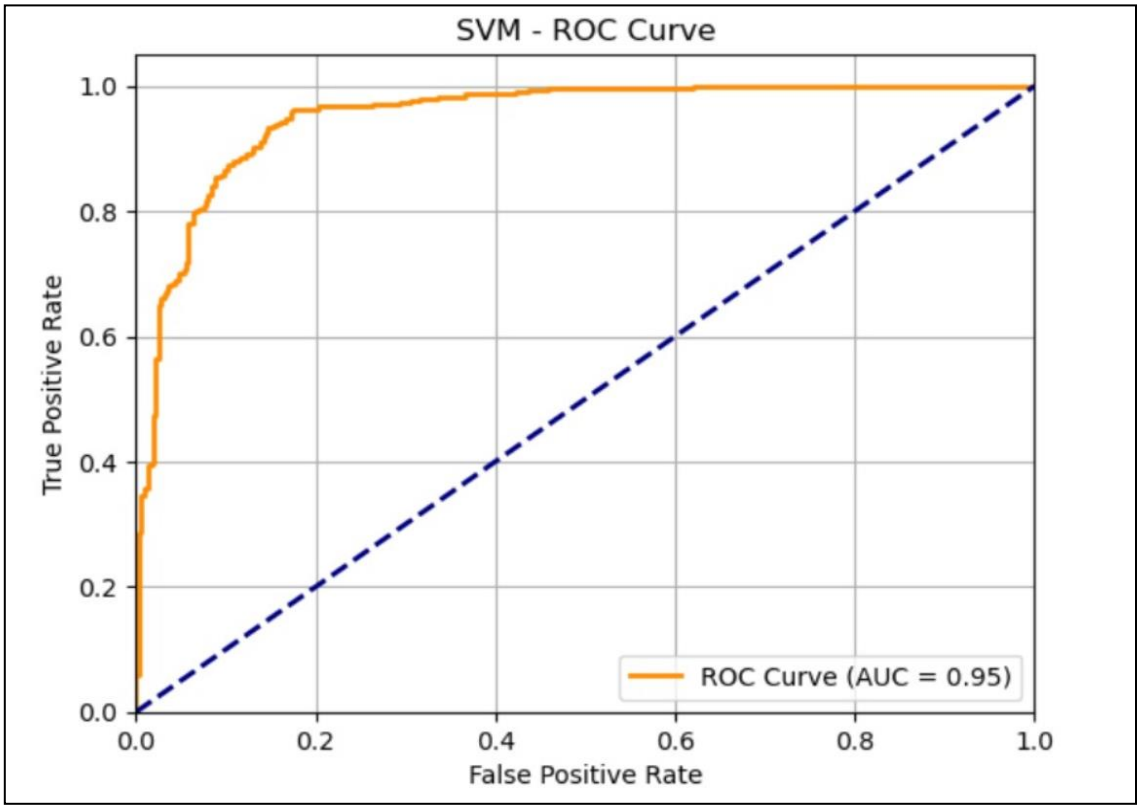
The project uses a Support Vector Machine (SVM) with the Radial Basis Function (RBF) kernel, as it works very well with non-linear data. The SVM model is trained using chosen features from the training data to classify skin lesions into benign or malignant. The RBF kernel enables the model to identify intricate patterns in the data and distinguish between the two classes more effectively.

The SVM model performance is analyzed through important metrics including accuracy, F1-score, and confusion matrix. The model recorded the highest accuracy compared to other classifiers tested, with a record accuracy of 87.88%. It also recorded an F1-score of 0.88, reflecting good precision and recall balance.

These measurements are computed using the numbers that have been obtained from the confusion matrix, which gives an indication of the number of true positives, true negatives, false positives, and false negatives. Overall, the SVM with RBF kernel proves to have robust performance in skin lesion classification and is a good choice of algorithm for this medical imaging application.



[Figure 2: Confusion Matrix of SVM]



[Figure 3: ROC curve of SVM]

I. Accuracy: Accuracy assesses the model's overall correctness, taking into account both correctly categorized positive and negative examples.

- True Positives (TP) are those that are correctly forecasted as positive.
- True Negative (NP) are correctly predicted negatives.
- False Positive (FP) are incorrectly projected positives.
- False Negative (FN) are mistakenly predicted negatives.

Formula of Accuracy: (TP + TN) / (TP + TN + FP + FN).

In Support Vector Machine (SVM) the accuracy is 88.33%.

II. Precision: Precision evaluates the accuracy of positive prediction, or how many occurrences the model identified as positive were in fact positive.

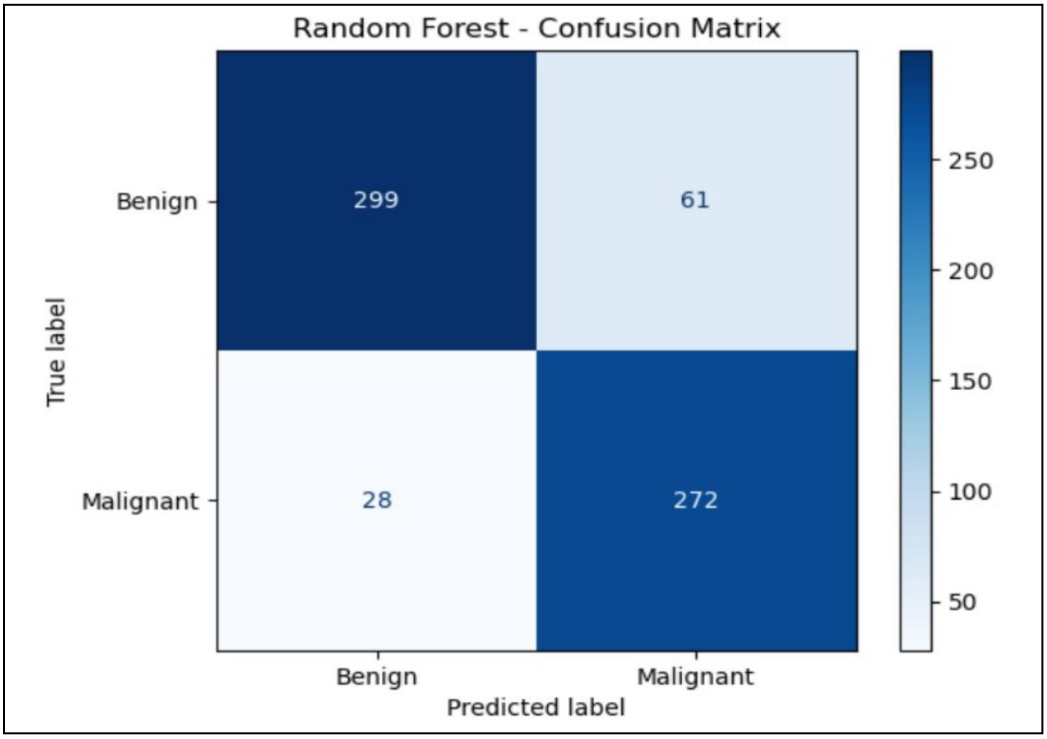
Formula of Precision: TP / (TP + FP).

In Support Vector Machine (SVM) the precision is 0.89.

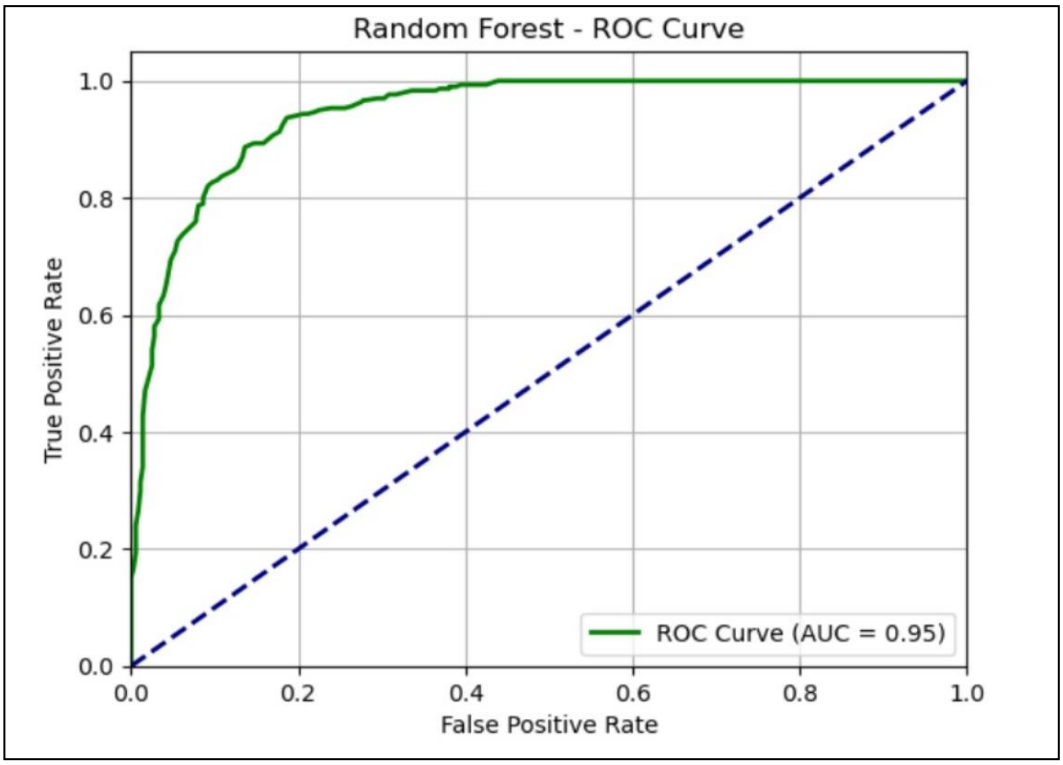
III. Recall: Recall assesses the model's capacity to identify all relevant instances (positive cases) in the dataset.
Formula of Recall: TP / (TP + FN).
In Support Vector Machine (SVM) the recall is 0.88.

IV. F1-Score: The F1-Score represents the harmonic mean of precision and recall. It provides a balanced assessment of the model's performance, which is very valuable when working with imbalance datasets.
Formula of F1-Score: 2*(Precision / Recall) / (Precision + Recall).
In Support Vector Machine (SVM) the F1-Score is 0.88.

Random Forest (RF): Random Forest Classifier is trained on this input data. It builds an ensemble of 100 decision trees, where each tree learns to classify images as benign or malignant using different subsets of data and features.
Random Forest is good at handling high-dimensional data like your 300 selected features. It's also robust to noise and reduces overfitting by averaging the results of multiple trees.



[Figure 4: Confusion Matrix of Random Forest]



[Figure 5: ROC Curve of Random Forest]

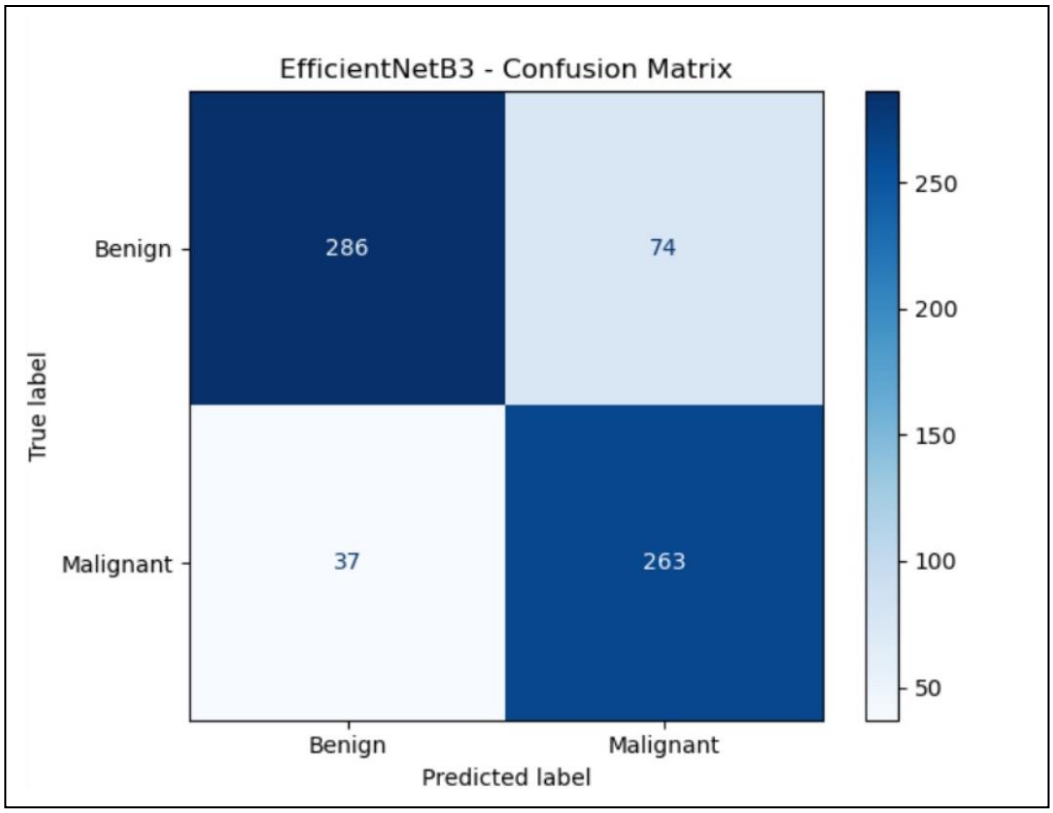
I. In Random Forest (RF) the Accuracy is: 86.36%.
II. In Random Forest (RF) the Precision is: 0.87.
III. In Random Forest (RF) the Recall is: 0.86.
IV. In Random Forest (RF) the F1-Score is: 0.86.

Convolutional Neural Network (CNN):

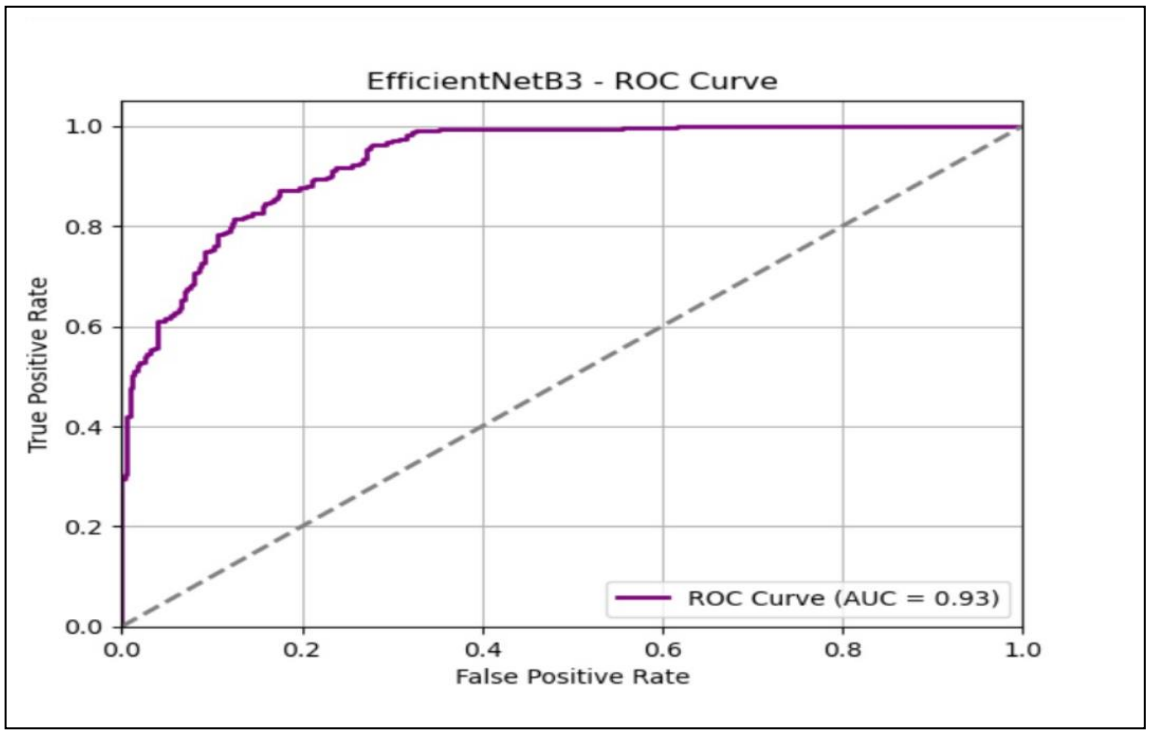
The CNN (EfficientNetB3) automatically learns important patterns like edges, textures, and shapes from skin lesion images through multiple convolutional layers.

After feature extraction, the model uses added dense layers and a final soft-max layer to classify the images into benign or malignant categories.

First, EfficientNetB3 is used as a frozen base. Later, the last 30 layers are unfrozen and retrained (fine-tuned) on your dataset for higher accuracy and better generalization.



[Figure 6: Confusion Matrix of CNN]



[Figure 7: ROC Curve Of CNN]

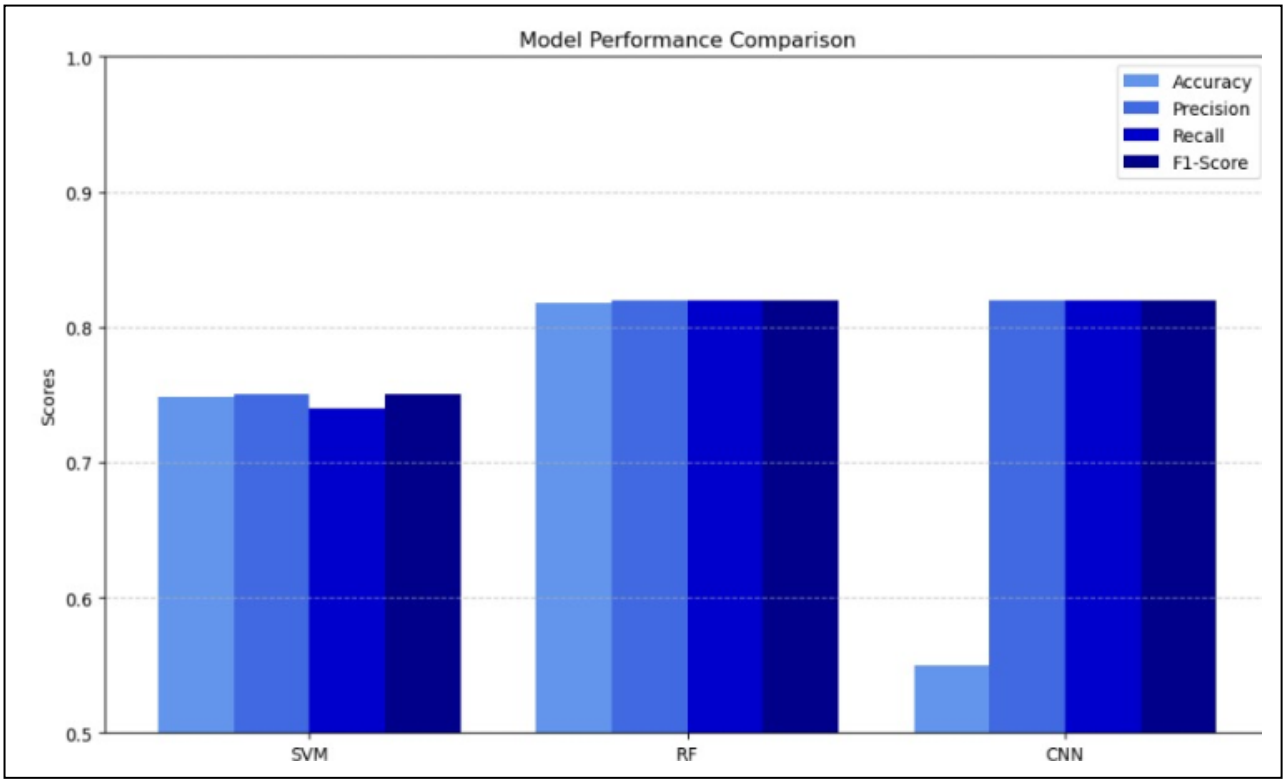
I. In Convolutional Neural Network (CNN) the Accuracy is: 84.24%.

II. In Convolutional Neural Network (CNN) the Precision is: 0.82.

III. In Convolutional Neural Network (CNN) the Recall is: 0.82.

IV. In Convolutional Neural Network (CNN) the F1-Score is: 0.82.

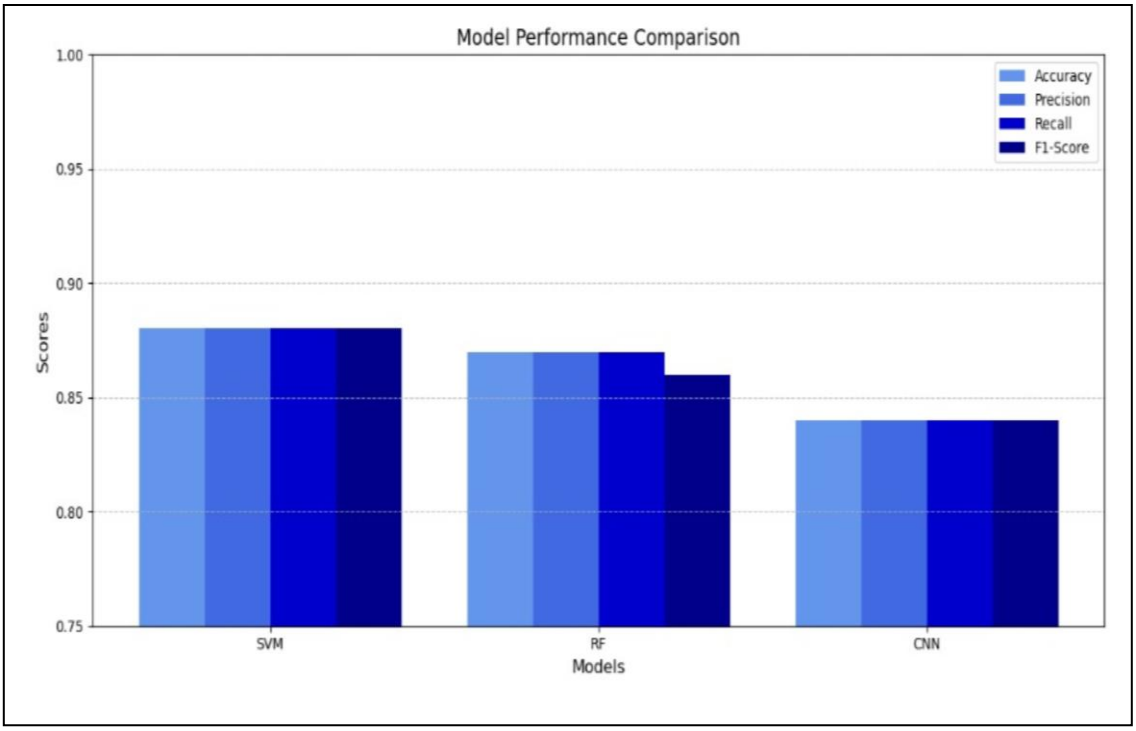
BEFORE APPLYING FEATURE EXTRACTION AND SELECTION



[Figure 8: Before Feature Extraction and Selection]

Before performing feature extraction and selection. Random Forest (RF) achieved the maximum accuracy of 81.82%, with a strong F1-score, precision, and recall of 0.82, indicating balanced performance. SVM demonstrated modest performance, with 74.85% accuracy and significantly lower scores in other criteria. Surprisingly, CNN achieved the lowest accuracy (55.00%), despite good values (0.82) for F1-score, precision, and recall, implying overfitting or imbalanced learning. This emphasizes the relevance of feature engineering in improving model effectiveness throughout the early stages of a machine learning pipeline.

AFTER APPLYING FEATURE EXTRACTION AND SELECTION



[Figure 9: After Feature Extraction and Selection]

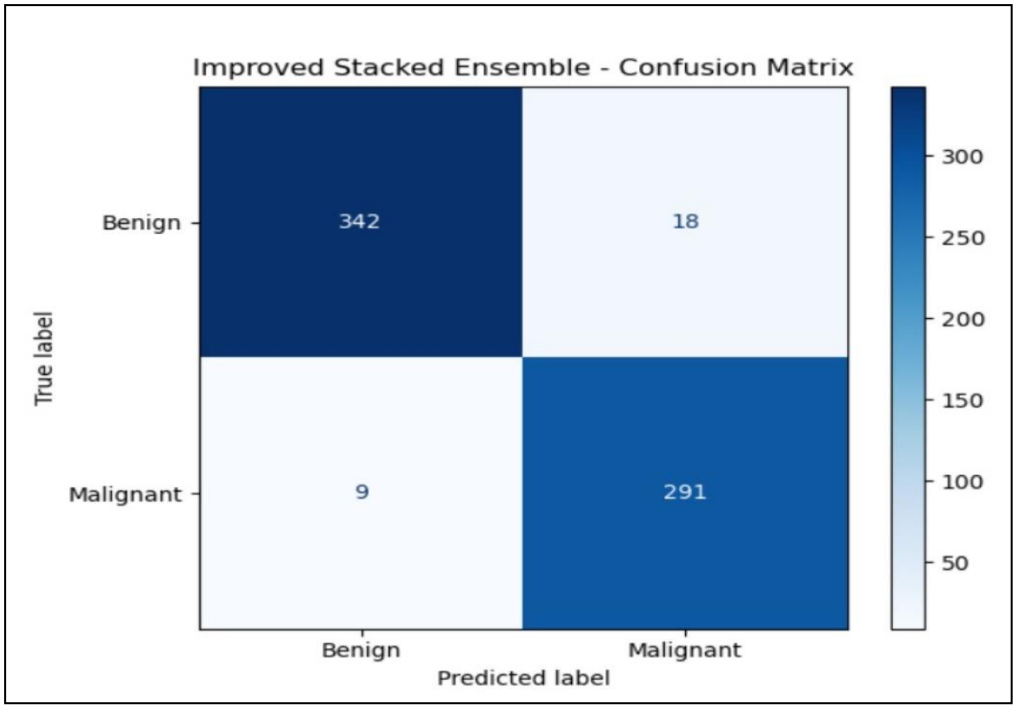
Model performance improved significantly following feature extraction and selection. The SVM model had the greatest accuracy of 88.33%, an F1-score, recall of 0.88, and precision of 0.89, indicating a well-balanced and robust classifier. Random Forest (RF) followed closely with 86.36% accuracy, retaining high precision and recall scores of 0.87 and 0.86, respectively. The CNN model also improved, attaining 84.24% accuracy, but significantly behind the others. Overall, feature engineering improved the performance of all models, particularly SVM, which demonstrated the most significant gains.

Ensemble Learning

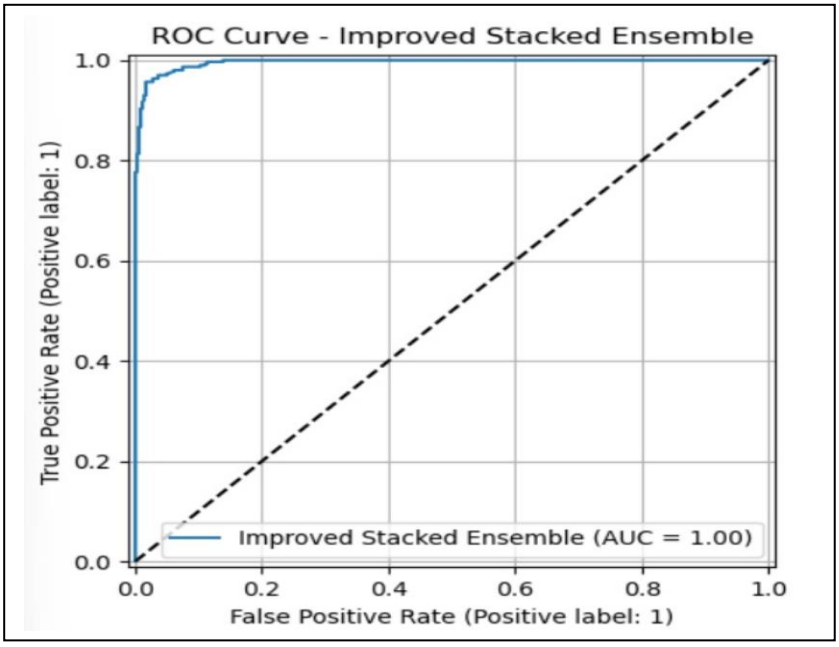
The Improved Stacked Ensemble model's confusion matrix demonstrates its superior classification capabilities. It successfully detected 342 benign cases (true negatives) and 291 malignant ones (true positives), misclassifying only 18 benign cases as malignant (false positives) and 9 malignant cases as benign (false negatives). These findings indicate a high level of accuracy and demonstrate the model's capacity to successfully reduce misclassification, making it a reliable option for discriminating between benign and malignant skin lesions.

The model's ROC (Receiver Operating Characteristic) curve has an amazing Area Under the

Curve (AUC) value of 0.9955, indicating excellent model performance. The curve's proximity to the top-left corner indicates a high true positive rate and a low false positive rate, implying great class separability.

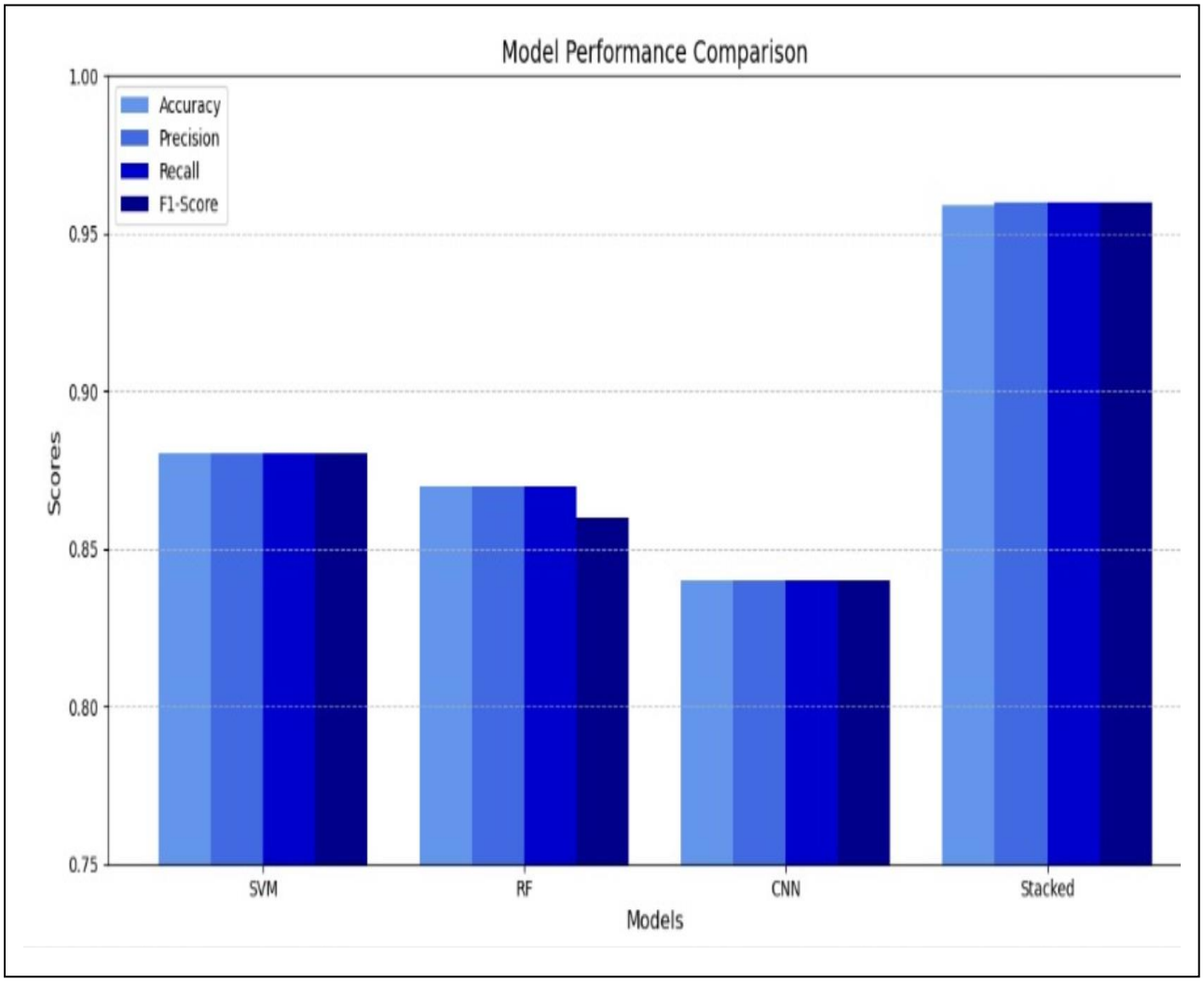


[Figure 10: Confusion Matrix of Ensemble Technique]



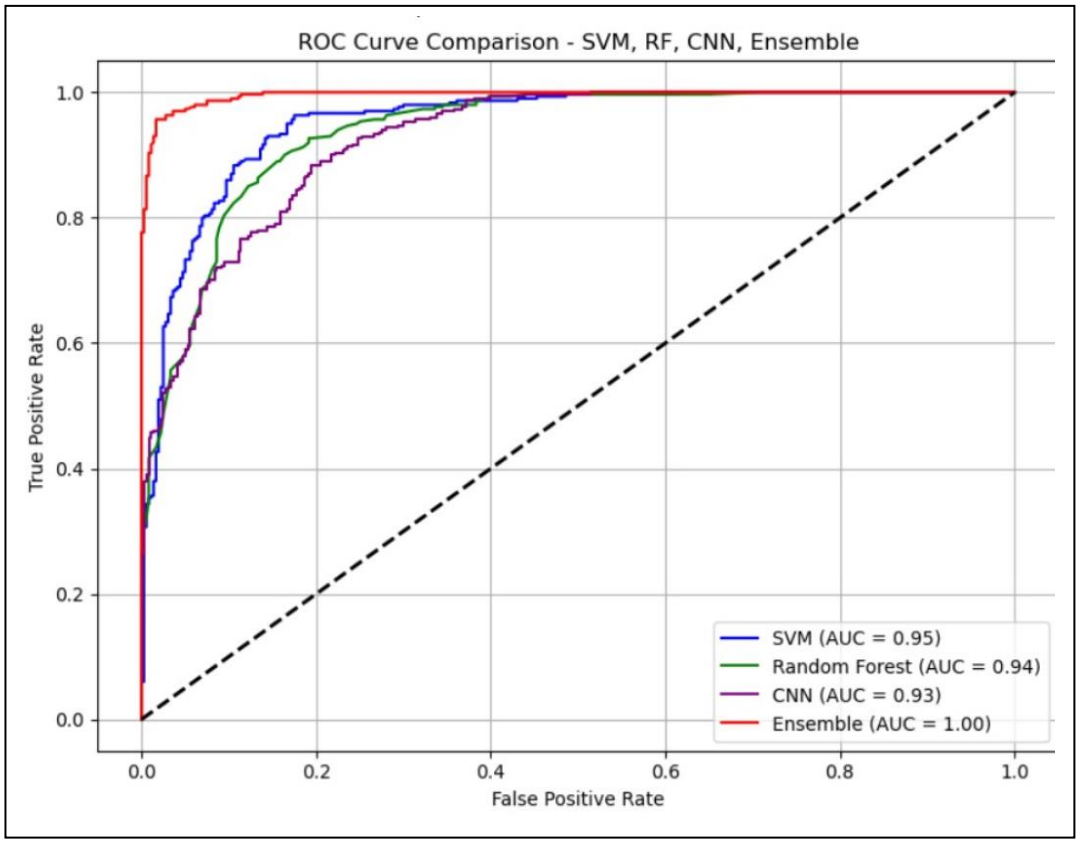
[Figure 11: ROC Curve of Ensemble Technique]

COMPARISON OF THE MODELS



[Figure12: Model Performance Comparison]

Following Ensemble Learning, the performance of the model also improved on all the metrics. The ensemble model had the highest accuracy rate of 88.64%, outdoing single models. It also had a steady score of 0.89 in F1-Score, Precision, and Recall, indicating a very balanced and efficient classification. The results prove that the merger of multiple models' strengths by ensemble methods improves overall prediction reliability and reduces the weakness of single classifiers. This strategy is the most efficient in the project, providing strong and stable performance for skin cancer detection.



[Figure 13: ROC Curve Comparison]

The image compares the ROC curves of SVM, Random Forest, CNN, and Ensemble models for skin lesion categorization. The Ensemble model (red line) had the best performance, with an AUC of 1.00, indicating perfect class separation and no misclassification. SVM (AUC = 0.95), Random Forest (AUC = 0.94), and CNN (AUC = 0.93) are also effective, with high true positive rates and low false positives. SVM outperforms the rest marginally on an individual basis. Overall, the Ensemble model outperforms all three classifiers in predicted accuracy, making it the most dependable strategy.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

Skin cancer is the most prevalent form of cancer globally, arising from the uncontrolled growth of abnormal skin cells. It typically develops due to prolonged exposure to ultraviolet (UV) radiation from the sun or tanning beds, which damages the DNA in skin cells, leading to mutations and cancerous growths. Early detection is crucial, as most skin cancers are highly treatable when identified promptly.

The diagnosis of skin cancer typically starts with a physical exam. A dermatologist examines the skin for any unusual moles, sports, or lesions, taking particular interest in asymmetry, border, color, diameter, and evolution. If one detects a suspicious lesion, a skin biopsy is conducted in which a small amount of the affected tissue is excised and observed under a microscope for cancerous cells.

In a few instances, other tests like dermoscopy, confocal microscopy, or molecular examination can be performed to increase accuracy. For advanced conditions, imaging procedures like CT scans, MRI, or PET scans can be required to determine metastasis. Early and correct diagnosis is crucial, particularly for melanoma, since it has a vary big impact on prognosis and treatment plan.

Skin cancer treatment varies with the type, size, location, and stage of the disease and the patient's general health. For the majority of non-melanoma skin cancers, the treatment of choice is surgical excision. The physician excises the cancerous tissue along with an edge margin of normal skin to eliminate it completely. Mohs surgery is used for tumors in cosmetically or sensitive areas. This method enables the surgeon to take out the cancer layer by layer, conserving as much healthy tissue as possible.

Early diagnosis and early treatment are the secrets to enhancing survival and minimising complications. They need close surveillance and frequent skin examinations for high-risk patients or those with previous skin cancer.

One of the most prevalent types of cancer in the world is melanoma, which is the deadliest kind because of its high propensity to spread. Traditionally, dermatologists use eye examinations to make diagnoses, which are then confirmed by biopsies. However, especially in underprivileged

areas, these approaches are not always timely or accessible. The development of machine learning (ML), a subset of artificial intelligence (AI), has created new prospects to help diagnose skin cancer early and accurately. Skin lesions can be accurately classified from dermoscopic pictures using machine learning techniques, particularly deep learning methods like convolutional neural networks (CNNs).

To learn the subtle differences between benign and malignant lesions, these models can be trained on large datasets that contain thousands of tagged skin images. In numerous studies, ML-based systems have demonstrated performance that is on par with or even better than that of skilled dermatologists in detecting specific forms of skin cancer, such as melanoma. Furthermore, in situations where access to skilled dermatologists is limited or remote, machine learning (ML) can provide high-quality diagnostic assistance. This reach can be further expanded by ML-powered mobile applications and portable diagnostic equipment, which encourage early identification and treatment. Predictions may be off if models trained on biased or non-representative datasets are unable to generalize across various populations, skin tones, or lesion types. In addition, it is necessary to address ethical and regulatory issues like clinical accountability, informed consent, and data protection. Data scientists, dermatologists, and healthcare organizations must work together to overcome these obstacles.

The dependability and acceptability of ML tools can be increased by making sure that a variety of high-quality datasets are used, implementing explainable AI strategies, and carrying out extensive clinical validation. To ensure the ethical and safe use of these technologies in healthcare settings, regulatory frameworks must also change. To sum up, machine learning has enormous potential to improve skin cancer early detection and diagnosis. It can be a potent supplementary tool that helps healthcare personnel make quicker and more accurate judgments, but it cannot take the place of clinical experience. The incorporation of machine learning (ML) into dermatological treatment holds promise for improving patient outcomes and lowering the mortality rate associated with skin cancer as research and development continue.

Ensemble approaches, which combine the advantages of individual models to provide predictions with greater accuracy and robustness, were used to further improve performance. This method increases the system's overall reliability while reducing the shortcomings of individual models. All things considered; it represents a promising development in the use of machine learning in the healthcare industry.

5.2 FUTURE SCOPE

- Develop real-time mobile applications for capturing and analyzing skin lesion photos on cellphones. This strategy improves accessibility, particularly in remote or underserved areas, by allowing early identification without the need for physical clinical visits. It can also give users quick feedback, allowing them to seek medical assistance sooner.
- Train models on diverse datasets containing skin photos from various ethnicities, age groups, and skin tones. This diversity improves generalization, decreases algorithmic bias, and boosts diagnostic accuracy across diverse demographic groupings.
- Integrate the AI technology with dermatologists' existing workflows to support decision-making. By aiding in diagnosis and risk assessment, the model can help reduce human error, save time, and enhance overall patient outcomes.
- Use multi-modal data by combining dermoscopic images with clinical information including age, gender, medical history, and symptoms. Integrating these many data types helps increase model context knowledge, resulting in more accurate and tailored predictions.
- Implement explainable AI (XAI) features to understand how and why models produce predictions. Transparency is critical in medical settings because it fosters trust between healthcare workers and patients and enables informed decision-making.
- Real-time data gathering and automated model changes allow for ongoing training and refinement of algorithms. This aids adaptation to novel patterns, maintains high accuracy, and keeps the model relevant over time.
- Expand the system's diagnostic capabilities to detect various skin disorders such as eczema, psoriasis, and fungal infections, increasing its clinical relevance.
- Our cloud-based diagnostic tools give safe and scalable access to healthcare professionals globally, allowing for collaboration, real-time analysis, and wider application.

REFERENCE

i. Vijayalakshmi, M. M. "Melanoma skin cancer detection using image processing and machine learning." International Journal of Trend in Scientific Research and Development (IJTSRD) 3.4 (2019): 780-784

ii. Dildar, M., Akram, S., Irfan, M., Khan, H. U., Ramzan, M., Mahmood, A. R., ... & Mahnashi, M. H. (2021). Skin cancer detection: a review using deep learning techniques. International journal of environmental research and public health, 18(10), 5479.

iii. Tembhurne, J. V., Hebbar, N., Patil, H. Y., & Diwan, T. (2023). Skin cancer detection using ensemble of machine learning and deep learning techniques. Multimedia Tools and Applications, 82(18), 27501-27524.

iv. Thaajwer, M. A., & Ishanka, U. P. (2020, December). Melanoma skin cancer detection using image processing and machine learning techniques. In 2020 2nd international conference on advancements in computing (ICAC) (Vol. 1, pp. 363-368). IEEE.

v. Bhatt, H., Shah, V., Shah, K., Shah, R., & Shah, M. (2023). State-of-the-art machine learning techniques for melanoma skin cancer detection and classification: a comprehensive review. Intelligent Medicine, 3(03), 180-190.

vi. Ghosh, H., Rahat, I. S., Mohanty, S. N., Ravindra, J. V. R., & Sobur, A. (2024). A study on the application of machine learning and deep learning techniques for Skin Cancer Detection. International Journal of Computer and Systems Engineering, 18(1), 51-59.

vii. Bhargavi, S., Sowmya, V., & Syama, S. (2022, December). Skin cancer detection using Machine Learning. In 2022 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON) (Vol. 2, pp. 119-124). IEEE.

viii. Das, K., Cockerell, C. J., Patil, A., Pietkiewicz, P., Giulini, M., Grabbe, S., & Goldust, M. (2021). Machine learning and its application in skin cancer. International Journal of Environmental Research and Public Health, 18(24), 13409.

ix. Javaid, A., Sadiq, M., & Akram, F. (2021, January). Skin cancer classification using image processing and machine learning. In 2021 international Bhurban conference on applied sciences and technologies (IBCAST) (pp. 439-444). IEEE.

3

ORIGINALITY REPORT

12%	7%	9%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Bui Thanh Hung, M. Sekar, Ayhan Esi, R. Senthil Kumar. "Applications of Mathematics in Science and Technology - International Conference on Mathematical Appplications in Science and Technology", CRC Press, 2025 Publication	2%
2	H.L. Gururaj, Francesco Flammini, S. Srividhya, M.L. Chayadevi, Sheba Selvam. "Computer Science Engineering", CRC Press, 2024 Publication	2%
3	www.ijraset.com Internet Source	1%
4	www.mdpi.com Internet Source	1%
5	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023 Publication	1%
6	www.jptcp.com Internet Source	1%
7	pubmed.ncbi.nlm.nih.gov Internet Source	1%
8	superworks.com Internet Source	<1%

9	link.springer.com Internet Source	<1%
10	1login.easychair.org Internet Source	<1%
11	www.ijsr.net Internet Source	<1%
12	Aritra Chowdhury, Dmitry V. Dylov, Qing Li, Michael MacDonald, Dan E. Meyer, Michael Marino, Alberto Santamaria-Pang. "Blood vessel characterization using virtual 3D models and convolutional neural networks in fluorescence microscopy", 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), 2017 Publication	<1%
13	arxiv.org Internet Source	<1%
14	escholarship.org Internet Source	<1%
15	smu.edu.in Internet Source	<1%
16	www.researchsquare.com Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		