Melanoma Detection Using Classification

Muqadas, Maryam, Khadija, Nighat
University of Punjab
mscsf24m007@pucit.com

January 14, 2025

Abstract

Melanoma is one of the deadliest types of skin cancer, with early detection being critical to improving survival rates. This study focuses on the application of machine learning classification methods for melanoma detection, emphasizing the use of manual feature extraction techniques. Segments and extracts the region of interest for further analysis, reducing the image size and focusing on critical areas.Key features such as texture, color, shape, and border irregularity are manually extracted from dermoscopic images and used to train traditional machine learning algorithms like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Decision Trees. Unlike deep learning models, which automatically learn features from raw data, manual feature extraction allows for domain-specific insights and reduces computational complexity. The results demonstrate that carefully selected handcrafted features can achieve competitive accuracy, making these techniques suitable for scenarios with limited data or computational resources. Challenges such as feature selection, noise reduction,hair removal and the need for robust preprocessing are also discussed, along with strategies to address these issues. The paper also presents the current challenges and future directions for skin cancer classification using machine learning approaches.

Keywords: Skin Cancer Classification, Machine Learning, Image Processing, Manual Feature Extraction

1 Introduction

Melanoma, a malignant form of skin cancer, is responsible for the majority of skin cancer-related deaths worldwide despite accounting for a smaller proportion of overall skin cancer cases. Early detection is crucial as it significantly increases the chances of successful treatment and improves patient survival rates. Traditionally, dermatologists rely on clinical and dermoscopic examination for diagnosing melanoma, but these methods often require substantial expertise and are prone to variability between practitioners.

In recent years, the integration of machine learning (ML) techniques into melanoma detection has shown promising results. ML algorithms can analyze dermoscopic images to classify lesions as malignant or benign, potentially matching or surpassing the diagnostic accuracy of dermatologists. However, the effectiveness of these models depends heavily on the quality of the features used to train them.

This study focuses on the application of machine learning classifiers in melanoma detection using manual feature extraction techniques. Unlike deep learning models, which automatically learn features, manual feature extraction involves identifying and computing domain-specific attributes such as texture, color, shape, and border irregularities. These features, derived from expert knowledge, are then used to train traditional ML algorithms like Support Vector Machines (SVM), Decision Trees, and K-Nearest Neighbors (KNN).

The advantages of this approach include reduced computational complexity, the ability to work effectively with smaller datasets, and better interpretability of the results. However, challenges such as selecting optimal features, dealing with noise in dermoscopic images, and ensuring robustness across different datasets persist.

2 Related Work

- **Ali et al. (2021):** This paper presents an enhanced DCNN model for skin cancer classification, evaluated against popular transfer learning models (e.g., ResNet, VGG-16). Using the HAM10000 dataset, the authors employed preprocessing steps such as noise removal, normalization, and data augmentation to improve model performance, achieving a high classification accuracy of 91.43%. Their use of data augmentation and specialized feature extraction techniques in DCNN provides a reliable method for early skin cancer detection[5].

- **X et al. (2023):** This study proposes a multi-level ensemble approach for skin cancer classification using transfer learning with an innovative triple attention mechanism. Using the ISIC 2019 dataset, the study employs EfficientNet, ResNet, and DenseNet as base models, combining them through weighted averaging to form a robust ensemble model. The introduction of triple attention—channel, spatial, and depth-wise—enhances feature extraction, making the model more sensitive to critical features in skin lesion images[4].

- **Kassem et al. (2021):** This systematic review explores both ML and DL approaches for skin lesion classification, highlighting advancements, challenges, and the effectiveness of different methodologies. Using datasets like ISIC and PH2, the study finds that DL models, particularly CNNs and transfer learning methods, significantly improve diagnostic accuracy over traditional ML methods[1].

- **Viknesh, Chandran Kaushik and Kumar (2023)**, they proposed two methods for detecting skin cancer and focus specifically on melanoma cancerous cells using image data. The first method employs convolutional neural networks, including AlexNet, LeNet, and VGG-16 models, and we integrate the model with the highest accuracy into web and mobile applications. We also investigate the relationship between model depth and performance with varying dataset sizes. The second method uses support vector machines with a default RBF kernel, using feature parameters to categorize images as benign, malignant, or normal after image processing. The SVM classifier achieved an 86.6 percent classification accuracy, while the CNN maintained a 91 percent accuracy rate after 100 compute epochs. The CNN model is deployed as a web and mobile application with the assistance of Django and Android Studio[3].

- **Gautam, Diwakar and Ahmed (2015)**,In paper color images of melanoma are imparted to classify them among malignant and benign classes using Support Vector Machine (SVM) optimized by Sequential Minimal Optimization(SMO). As a part of the preprocessing step, an illumination compensation based segmentation algorithm is deployed. The segmentation process is followed by the proposed iterative dilation method to remove noise from a lesion. Some prominent features calculated from the segmented image based on asymmetric lesion-behavior, border irregularity, color variations and spanned diameter. Finally, these feature vector applied as an input to SVM classifier, which is used to distinguish malignant from benign samples of skin lesions. The dataset is divided into training and testing data to account and validate the system performance.[2].

3 Dataset

The HAM10000 dataset was used, which is designed for skin lesion classification tasks. It contains 5000 high-resolution dermoscopic images, with a balanced distribution of 2500 malignant (melanoma) and 2500 benign lesions. Images were preprocessed by resizing to 256×256, grayscale conversion, and normalization.

4 Methodology

4.1 Data Preprocessing

The preprocessing steps include:

1. Resizing images to a uniform resolution of 256×256.
2. Grayscale conversion to reduce computational complexity.
3. Applying a median filter to remove noise.
4. Hair removal using morphological operations.

Gray Scale Image: Grayscale image, where each pixel value is a single intensity value representing the brightness of the pixel. The process typically involves a weighted sum of the three color channels (BGR) to calculate the intensity of the grayscale pixel. The formula is as follows. Gray = 0.299 * Blue + 0.587 * Green + 0.114 * Red

Noise removal: Median filtering is a highly effective method for removing salt-and-pepper noise, as it replaces each pixel with the median of its neighbors, preserving edges and details better than Gaussian Blur Median=middle value in the sorted list

4.2 Feature Engineering

Feature Extraction: Extracting features from segmented skin lesions reduces the amount of initial data by measuring specific features that distinguish one type of input from another. Feature extraction provides important skin information, including math, images, data, and evidence. This process converts the amount of data into a record that is easy to analyze with high performance. In our study, we extracted specific and unique features that distinguish malignant melanoma from benign melanoma. The data collected are used to identify skin lesions.

First-order statistical features: First-order labels are widely used and are simple indicators that describe the distribution of pixels used in the image area defined by the cover.

1. Mean:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Mean is the average of all pixel values in the image.

2. Variance:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

Variance measures the dispersion or spread of the pixel values from the mean.

3. Standard Deviation:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Standard deviation is a measure of the spread of the pixel values, indicating how much the pixel values deviate from the mean.

4. Skewness:

$$\gamma_1 = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i - \mu}{\sigma}\right)^3$$

Skewness measures the asymmetry of the image's pixel intensity distribution. Positive skewness indicates a right-skewed distribution, while negative skewness indicates a left-skewed distribution.

5. Kurtosis:

$$\gamma_2 = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i - \mu}{\sigma}\right)^4 - 3$$

Kurtosis measures the "tailedness" of the pixel intensity distribution. High kurtosis means the distribution has heavy tails.

6. Median: The median is the middle value of the pixel intensities in the image when they are sorted in ascending order.

7. Entropy:

$$H(X) = -\sum_{i=1}^{N} p(x_i)\log(p(x_i))$$

Entropy measures the randomness or unpredictability of the image's pixel intensity distribution.

Gray Level Co-occurrence Matrix: The GLCM is a robust technique for extracting image features by mapping the gray level co-occurrence probabilities based on the spatial relationships of the pixels in various angular directions . GLCM is essential in texture analysis, where fine details from an object are captured. It involves two pixels: a neighboring pixel and a reference pixel. The statistical approach of texture analysis takes into account the spatial arrangement of the pixels. The GLCM of the entire lesion describes the texture by assessing how frequently pairs of pixels with specific brightness values and orientation appear in an image.

1. Energy:

$$\text{Energy} = \sum_{i,j} P(i,j)^2$$

Energy measures the uniformity or textural uniformity in the image. Higher energy values indicate a more uniform texture.

2. Contrast:

$$\text{Contrast} = \sum_{i,j} (i-j)^2 P(i,j)$$

Contrast measures the intensity contrast between a pixel and its neighbor over the entire image. Higher contrast indicates greater intensity variations in the texture.

3. Homogeneity:

$$\text{Homogeneity} = \sum_{i,j} \frac{P(i,j)}{1+(i-j)^2}$$

Homogeneity measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. Higher values indicate more uniform and smoother textures.

4. Correlation:

$$\text{Correlation} = \frac{\sum_{i,j} (i-\mu_x)(j-\mu_y)P(i,j)}{\sigma_x \sigma_y}$$

Correlation measures the correlation between pixel values in the image. High correlation means the pixels are more linearly related.

Local Binary Pattern (LBP) computes texture information by comparing each pixel's intensity with its neighboring pixels within a defined radius. For each neighbor, a binary value of 1 is assigned if the neighbor's intensity is greater than or equal to the center pixel, otherwise 0. The resulting binary values form an LBP code, representing local texture patterns. The LBP feature extraction involves generating these codes for the entire image, computing their histogram, and optionally normalizing it for further statistical analysis.Histogram is computed from LBP code and generate features from histogram like lbp$_mean, lbp_std, lbp_skewness, kurtosis, entropy$.

Shape features from an input image, specifically for an object or lesion (e.g., a segmented region). Here's a step-by-step explanation of how to compute these features :

- **Area:** Number of pixels in the region, representing its size.
- **Perimeter:** Length of the boundary of the region.
- **Eccentricity:** Describes how elongated the region is. Values range from 0 (perfect circle) to 1 (line-like).
- **Solidity:** Ratio of the object's area to the area of its convex hull.
  - High solidity (≈ 1): Compact shapes with no gaps.
  - Low solidity: Irregular or concave shapes.
- **Bounding Box:**
  - *bbox_minrow, bbox_mincol, bbox_maxrow, bbox_maxcol:* Minimum and maximum coordinates of the bounding box around the object.
- **Circularity:** How circular the object is:

  $$\text{Circularity} = \frac{4\pi \times \text{Area}}{\text{Perimeter}^2}$$

  A value of 1 indicates a perfect circle.
- **Roughness:** Boundary irregularities, comparing the actual perimeter to the convex perimeter.
- **Fractal Dimension:** Describes the complexity of the shape:

  $$\text{Fractal Dimension} = \frac{\text{Perimeter}^2}{\text{Area}}$$

Extracts edge-related features from an image using the Canny Edge Detection algorithm. It calculates the edge density, which is the percentage of edge pixels in the image, and returns it as a feature. The edge density is computed by dividing the total number of edge pixels by the total number of pixels in the image.

The Hu Moments, which are a set of seven values that capture the shape and distribution of color or intensity in an image. These moments are invariant to translation, scale, and rotation, making them useful for shape analysis. The function calculates the Hu Moments of the given image using cv.moments() and cv.HuMoments(), then returns the mean of these moments as a feature.

Extracting Region of interest: The ROI extraction process focuses on isolating the significant regions of an image for analysis, using binary masks as guides. First, the images undergo preprocessing, such as hair removal, to create binary masks that highlight areas of interest. These masks are then refined using a filtering function, which removes small or irrelevant objects that are either too small or far from the central region of the image, ensuring the focus remains on meaningful features. Once the masks are filtered, the ROI extraction function is applied to retain the pixel intensities of the original image corresponding to the marked areas in the mask, while setting all other regions to black. This process is repeated for each image and its respective mask, resulting in a collection of ROIs that isolate critical features, removing irrelevant background information. The extracted ROIs are ready for further analysis or use in machine learning models, ensuring the focus remains on the most relevant parts of the images.

A correlation heatmap is a graphical representation that visualizes the correlation matrix of a set of features in a dataset. In the context of image processing or machine learning, it is used to identify relationships between different features or variables. The values in the correlation matrix range from -1 to 1.

Feature Transformation and scaling: Feature transformation is an important step in feature engineering of numeric data and it is used to handle skewed data. Machine Learning Deep Learning algorithms are highly dependent on the input data quality. If the data quality is not good then even the high-performance algorithms are of no use.

Principal Component Analysis (PCA) : By applying PCA, the code reduces the dimensionality of the data while retaining the most critical information, which can improve model efficiency and reduce over fitting in subsequent machine learning steps. Here, pca = PCA(20) initializes PCA to reduce the dataset to 20 principal components. The number 20 specifies the number of principal components (new features) to retain after the transformation.

Choosing the Optimal Number of Components: By looking at the plot, you can decide how many components to retain based on the cumulative explained variance. Typically, you might choose to retain enough components to explain, for example, 95 percent of the total variance.

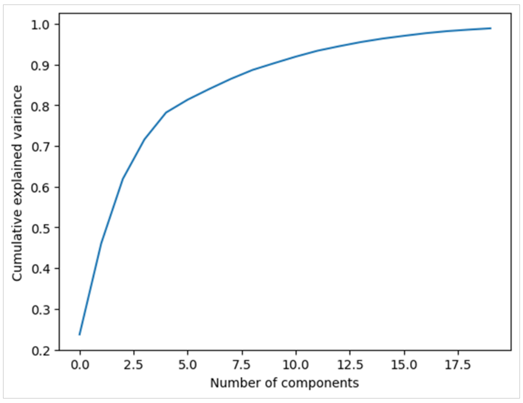


Figure 1: Visualizing Variance Retention

Handling Skewness: Techniques and Transformations To mitigate the effects of skewness and improve the performance of machine learning models, data preprocessing techniques and transformations can be applied.

4.3 Classification

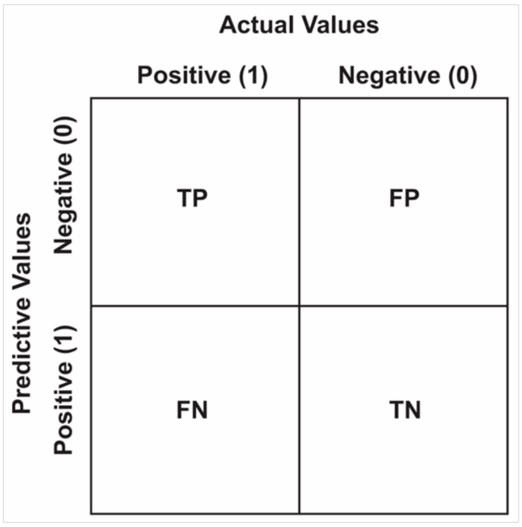
The extracted features were used to train various ML classifiers, including:

- **Logistic Regression:** Effective for binary classification.
- **Support Vector Machines (SVM):** Finds an optimal hyperplane for separation.
- **K-Nearest Neighbors (KNN):** Instance-based learning.
- **Random Forest:** Ensemble of decision trees.

5 Model Evaluation

The confusion matrix (Figure 2) is a table often used to represent the performance of the classification model on known test data. From here, various metrics, such as accuracy, recall, precision, and F1 score, can be calculated to evaluate the effectiveness of the classification model.

- **True Positive (TP):** A positive sample is correctly predicted.
- **True Negative (TN):** Negative class instances are predicted as negative instances.



		Actual Values	
		Positive (1)	Negative (0)
Predictive Values	Negative (0)	TP	FP
	Positive (1)	FN	TN

Figure 2: Confusion matrix

- **False Positive (FP):** A negative class is incorrectly predicted as positive (also known as a "Type I error").
- **False Negative (FN):** A positive sample is incorrectly predicted as negative (also known as a "Type II error").

Classifier accuracy is defined as the percentage of test groups classified by the model, as described in Equation 1.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{1}$$

Sensitivity, also known as recall, is defined by Equation 2:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \times 100\% \tag{2}$$

Specificity measures the true negative rate and is defined by Equation 3:

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100\% \tag{3}$$

Precision is defined by Equation 4:

$$\text{Precision} = \frac{TP}{TP + FP} \times 100\% \tag{4}$$

The F-measure, which balances precision and sensitivity, is defined by Equation 5:

$$F_{\text{measure}} = 2 \cdot \frac{\text{Precision} \cdot \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \times 100\% \tag{5}$$

6 Results and Discussion

The dermoscopic images obtained from the dataset were initially processed, segmented using various image processing techniques, and classified using ML methods. The dermoscopic images collected from the dataset are shown in Fig 2. (a) malignant, (b) benign.

Figure 3: Dermoscopic images (a) malignant, (b) benign.

The image collection was initially converted to 256 × 256. Grayscale images are easier and faster to process than color images. This conversion converts the RGB image into a grayscale image. Figure 3 show grayscale images.

For Hair Removal Steps and Their Results:

Blackhat Transformation: Result: Emphasizes dark regions that stand out from the background.

- **Dilation: Result**: Thicker dark areas and edges that are easier to analyze.

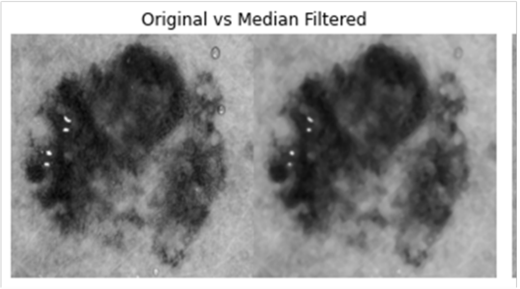


Figure 4: Original gray scale VS median gray scale

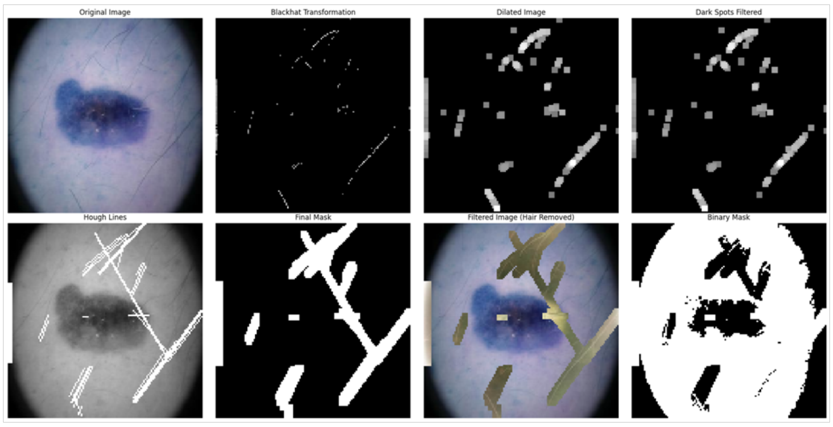


Figure 5: Results of each step for removing unwanted details

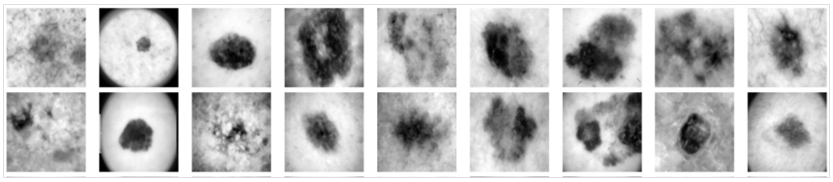


Figure 6: Hair removal operation on median filtered image

- **Dark Spots Filtered:** Result: Dark spots are clearly highlighted, which could represent features like tumors.
- **Hough Line Transform:Result:** Lines are drawn, helping in defining boundaries or structures in the image.
- **Final Mask:Result:** Masked image highlighting potential areas of concern.
- **Hair Removed Image**:Result: Unnecessary features are removed, allowing better visualization of regions of interest.
- **Otsu Binary Mask:Result:** The foreground (e.g., tumor) is separated from the background using a binary mask.

Extracting Region of Interest (ROI)

The process of extracting the Region of Interest (ROI) involves isolating specific regions in an image using a binary mask. The following steps summarize the methodology:

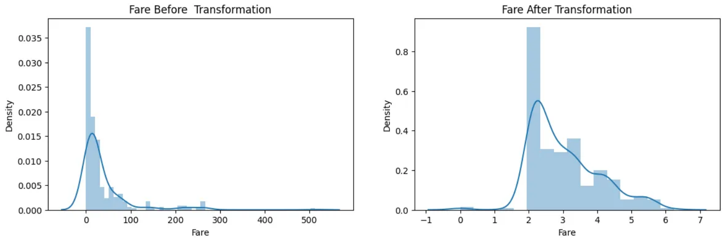


Figure 7: handling skewness

1. **Convert Mask to Binary:** The binary mask is converted to a format where all non-zero values are replaced with 1, ensuring a proper binary representation(Figure 8).

2. **Extract the ROI:** Multiply the original grayscale image with the binary mask. This retains the pixel intensities within the ROI (where the mask is 1) and sets all other areas to zero(Figure 9).

The resulting image contains only the desired region of interest, with other regions blacked out. The ROI extraction is particularly useful in applications where specific features or areas of an image need to be analyzed while ignoring irrelevant parts.

Features transformation results: Yeo-Johnson Transformation Similar to the Box-Cox transformation, the Yeo-Johnson transformation is a power transformation method that can handle skewed data. Unlike Box-Cox, Yeo-Johnson allows for transformation of both positive and negative values, making it more flexible in practice(Figure 7).

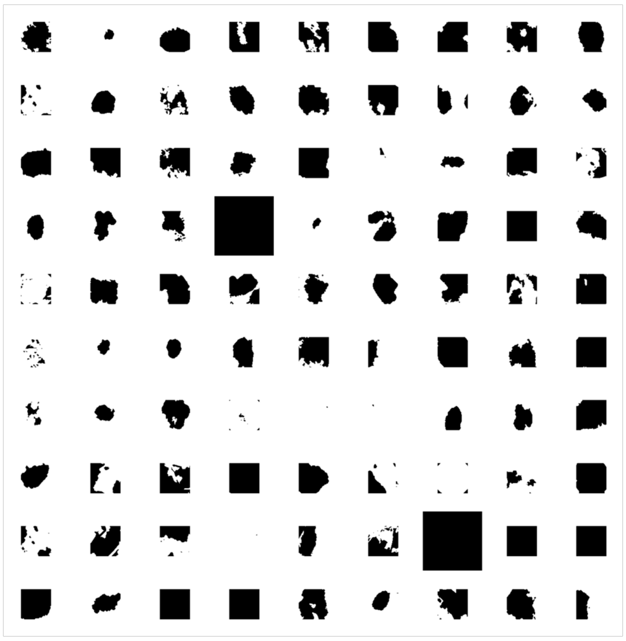


Figure 8: Binary Mask

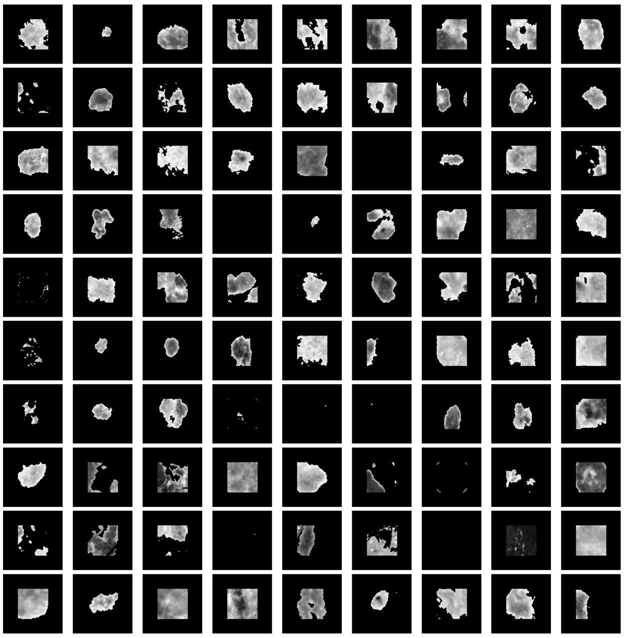


Figure 9: Extraction of ROI

6.1 Model Performance Comparison

The performance of five different machine learning models is summarized in Table 1.

Table 1: Performance Metrics of Different Models Without Segmentation

Model	Accuracy	Precision	Recall	F1-Score
Support Vector Machine (SVM)	0.87	0.84	0.87	0.85
Decision Tree	0.82	0.77	0.80	0.79
Random Forest	0.88	0.85	0.87	0.86
K-Nearest Neighbors (KNN)	0.85	0.82	0.87	0.84
Logistic Regression	0.87	0.84	0.86	0.85

Table 2: Performance Metrics of Different Models With Segmentation

Model	Accuracy	Precision	Recall	F1-Score
Support Vector Machine (SVM)	0.87	0.83	0.87	0.87
Decision Tree	0.82	0.77	0.81	0.79
Random Forest	0.88	0.84	0.87	0.85
K-Nearest Neighbors (KNN)	0.85	0.84	0.84	0.82
Logistic Regression	0.86	0.82	0.84	0.83

7 Findings:

The comparison of model performance metrics reveals notable differences between models applied with and without segmentation. Random Forest consistently achieves the highest accuracy (0.88) and balanced performance across precision, recall, and F1-score in both cases, making it the most robust model. Support Vector Machine (SVM) also performs well, with stable accuracy (0.87) and high recall, indicating its reliability in detecting positive cases. K-Nearest Neighbors (KNN) and Logistic Regression show moderate performance, while the Decision Tree model records the lowest scores, particularly in precision and F1-score. Incorporating segmentation improves the precision and F1-score of SVM and Random Forest slightly, while other models see marginal improvements or stability.

8 Future Work:

To further enhance model performance, several avenues for future work can be explored. Hyperparameter tuning should be prioritized, as optimizing parameters such as the number of estimators in Random Forest, kernel selection in SVM, or the number of neighbors in KNN could lead to significant performance improvements. Additionally, experimenting with advanced feature selection techniques may help identify the most relevant features, reducing noise and improving classification accuracy. Integrating ensemble methods or deep learning models, such as Convolutional Neural Networks (CNNs), could also be explored to better capture complex patterns in the data. Lastly, increasing the dataset size and ensuring balanced class distributions will provide more robust training and evaluation, potentially improving model generalization and reducing overfitting.

9 Conclusion

Segmentation has a positive effect on model performance, especially in models like Random Forest and SVM, which show slight increases in precision and F1-score after segmentation. Among all models, Random Forest emerges as the best-performing classifier, achieving consistently high accuracy and balanced metrics, both with and without segmentation. This indicates its robustness and suitability for the given task. Meanwhile, Decision Tree shows room for improvement, as its metrics are comparatively lower across both scenarios. These results suggest that segmentation enhances the model's ability to focus on relevant features, improving classification performance in certain cases.

References

[1] Machine learning and deep learning methods for skin lesion classification and diagnosis: a systematic review. *Diagnostics*, 11(8):1390, 2021.

[2] Diwakar Gautam and Mushtaq Ahmed. Melanoma detection and classification using svm based decision support system. In *2015 annual IEEE India conference (INDICON)*, pages 1–6. IEEE, 2015.

[3] Chandran Kaushik Viknesh, Palanisamy Nirmal Kumar, Ramasamy Seetharaman, and Devasahayam Anitha. Detection and classification of melanoma skin cancer using image processing technique. *Diagnostics*, 13(21):3313, 2023.

[4] Wenjing Yang, Jing Gao, Jingjing Chen, Maolin Yang, Junchi Ma, Bing Chen, and Haonan Sun. Deep learning for skin lesion segmentation: A review and outlook. In *International Conference on Artificial Intelligence in China*, pages 175–185. Springer, 2023.

[5] Muhammad Zia Ur Rehman, Fawad Ahmed, Suliman A Alsuhibany, Sajjad Shaukat Jamal, Muhammad Zulfiqar Ali, and Jawad Ahmad. Classification of skin cancer lesions using explainable deep learning. *Sensors*, 22(18):6915, 2022.

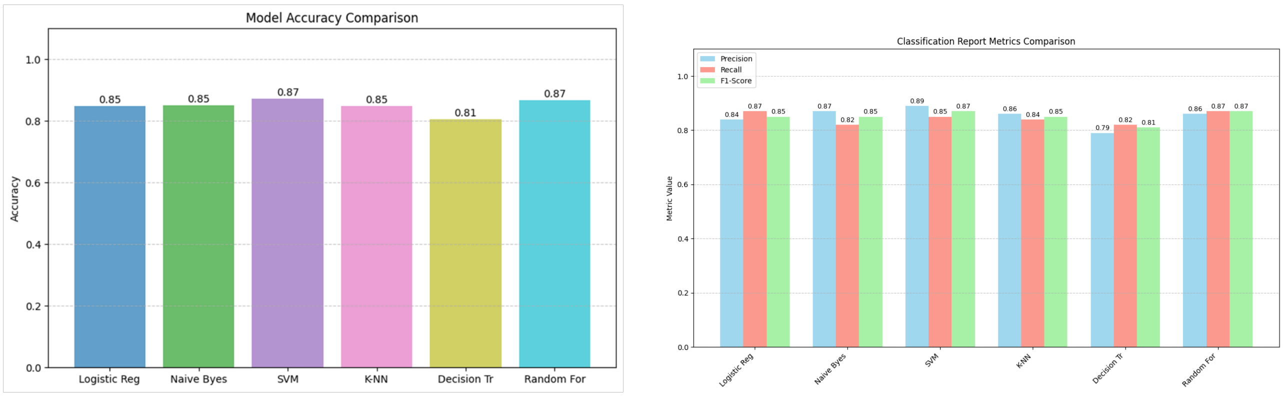


Table 3: Comparison Without Segmentation

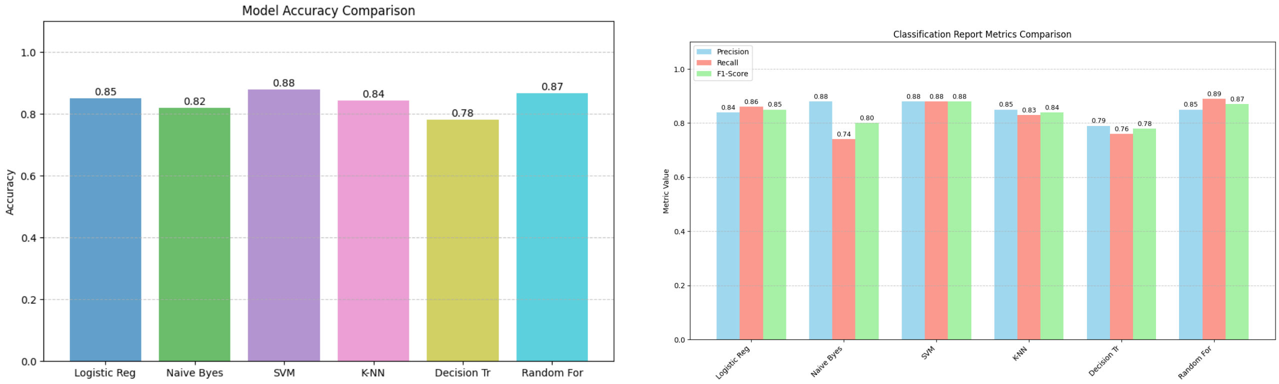


Table 4: Comparison With Segmentation