Journal of Theoretical and Applied Information Technology
30th April 2026. Vol.104. No.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

COMPREHENSIVE SKIN DISEASE CLASSIFICATION: ENHANCING EARLY DETECTION THROUGH ADVANCED IMAGE PROCESSING AND MACHINE LEARNING MODELS

JESWIN PAUL PANDIAN[1], SAMSON ISAAC[2], P. SUBHA HENCY JOSE[3], P. ANANTHA CHRISTU RAJ[4]

[1] Research Scholar, Division of Biomedical Engineering, Karunya Institute of Technology and Science, Coimbatore
[2] Assistant Professor, Division of Biomedical Engineering, Karunya Institute of Technology and Science, Coimbatore
[3] Associate Professor, Division of Biomedical Engineering, Karunya Institute of Technology and Science, Coimbatore
[4] Assistant Professor, Division of Robotics and Automation Engineering, Karunya Institute of Technology and Science, Coimbatore

Email: Jeswinpaul14@gmail.com

ABSTRACT

Skin disease varies from benign blemishes to cancerous malignancies. This study uses large datasets to classify skin diseases as malignant or non-cancerous to improve early diagnosis and treatment. Our study examines basal cell carcinoma, squamous cell carcinoma, actinic keratosis, seborrheic keratosis, melanoma, granulomas, angiomas, and angiokeratomas, as well as benign conditions like melanocytic nevus, dermatofibroma, hemangioma, pilar cysts, keratinous cysts, moles, skin tags, and wart for comparison, healthy skin photos are supplied. Two datasets—10,000 melanoma photos and 5,000 images of various skin conditions—represent varied skin types and lesion appearances. Preprocessing included scaling all photos to uniform dimensions, image augmentation (flipping, rotating, scaling), and pixel value normalization to aid machine learning algorithm convergence. Dermatologists cleaned data and tagged photos with condition-specific information for supervised learning. We oversampled minority classes and created synthetic data using SMOTE to correct class imbalance. Our study focuses on segmentation and feature extraction to properly identify skin cancer, non-cancerous lesions, and healthy skin. For exact lesion border delineation, MorphACWE, MorphGAC, thresholding, edge detection, clustering, and semantic segmentation were used. Both standard and sophisticated approaches were used to extract features, including HOG, LBP, GLCM, Gabor filters, and shape descriptors like Hu and Zernike moments. Skin disease classification was tested using five machine learning models: ResNet/DenseNet, GAN, Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbor (KNN). Our dataset contained 10,000 melanoma photos, 10,000 healthy skin images, and images of actinic keratosis, melanocytic nevus, dermatofibroma, hemangioma, keratinous cysts, pilar cysts, lipoma, moles, skin tags, and warts. We choose the best skin disease classification model by comparing F1-score, recall, accuracy, and precision. This study emphasizes the need to combine image processing technologies to improve skin lesion identification. We want to improve dermatology by allowing quick and exact detection of skin problems, which will improve clinical choices and treatment results. The report also recommends dermatological diagnostics research and examines model selection in practical problems healthcare settings.

Keywords: *Skin Diseases, Melanoma, Image Segmentation, Feature Extraction, Machine Learning, Classification, Dermatology.*

1. INTRODUCTION

Skin diseases, which range from benign blemishes to malignant malignancies, represent substantial issues in dermatology. Early and correct diagnosis of these disorders is critical since it may significantly improve patient outcomes [1], particularly in instances of skin cancer. The rising worldwide frequency of skin cancer highlights the need for better diagnostic techniques that can discriminate among malignant and non-cancerous skin disorders with high accuracy. Historically [2], dermatological diagnoses were mainly reliant on

clinical examination and histological investigation, which, although useful, are time-consuming and opening up new opportunities for automated and more accessible skin disease categorization. Using these technologies, researchers want to create systems that may help physicians make quicker and more accurate diagnoses, resulting in timely therapeutic interventions.

This work aims [3] to improve skin illness categorization by using a large dataset that contains photos of a variety of skin disorders, both malignant and benign, as well as normal skin. The dataset is an important resource since it not only offers a diverse range of skin types and lesion appearances, but it also enables for the development of machine learning models that can generalize effectively across populations [4]. To guarantee the classification models' robustness, many critical procedures were taken during dataset preparation. These included picture scaling to preserve consistency, augmentation methods including flipping, rotating, and scaling to increase model robustness to changes in image orientation, and pixel value normalization to boost machine learning algorithm convergence rates [5][6]. Additionally, dermatological professionals thoroughly cleaned the data to remove duplicates and low-quality photos, confirming the training data's dependability. Given the dataset's intrinsic class imbalance, notably the overrepresentation of melanoma photos, synthetic data creation techniques such as SMOTE and oversampling of minority classes were used. These strategies were critical in reducing model bias toward more common illnesses, resulting in balanced performance across all skin disease categories. The study [7] also looks at advanced segmentation and feature extraction methods, which are critical for correctly detecting lesion borders and extracting significant patterns from pictures. The work [8] intends to capture the complex morphological and textural properties of skin lesions by combining classic techniques such as Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP) with more advanced approaches such as Gabor filters and shape descriptors.

Five machine learning models—ResNet/DenseNet, GAN, Random Forest, Support Vector Machine [9], and K-Nearest Neighbor—were thoroughly tested to verify their efficacy in identifying different skin conditions. The performance of these models was evaluated using important measures such as F1-score, recall, accuracy, and precision, with the objective of determining the most dependable model for clinical use. This study aims to make a substantial need specialist knowledge. Machine learning and image processing tools have recently emerged, contribution to dermatology by integrating rigorous data preparation, cutting-edge image processing methods, and strong machine learning models. The results [10] have the potential to increase skin disease categorization accuracy, allow for earlier diagnosis, and improve treatment outcomes for people throughout the globe.

2. LITERATURE SURVEY

A number of points have been made in favor of using AI to aid in the medical field. The post-pandemic situation fuels the excitement around telemedicine technological help for rural regions. In the context of dermatological healthcare, the imbalanced distribution of dermatologists makes general practitioners the primary medical experts who provide dermatology treatment in rural regions. This research compares [11] several approaches to segmentation methods for localizing skin lesions on a given picture. The study is based on skin pigmentation level, pigmentation contrast between healthy skin and the lesion, and lesion distribution. Overall, Grab-cut is the most resilient for any given circumstances. Given that [12] there are around 3000 skin illnesses, detecting and treating them is a difficult task. This complicates the diagnosis of skin illnesses, emphasizing the need of precise identification in order to treat the problem successfully. Current deep learning-based skin detection methods are often limited to a restricted number of skin disorders, operate on tiny datasets, and seldom reach Top-5 accuracy of more than 70%.

Misdiagnosis [13] is common because some illnesses seem so similar that they might be mistaken for another. As a result, there is a need for a computer-based system for skin disease detection and categorization using photographs to increase diagnosis accuracy while also dealing with the lack of human specialists. The study's objective [14] is to improve diagnostic accuracy while overcoming the obstacles associated with distinguishing between skin conditions that seem anatomically similar. This strategy seeks to enhance patient care by reducing diagnostic time and promoting more practical and accurate treatment alternatives. The skin [15] is the biggest organ in the body and serves an important protective function. Skin illnesses may be inherited or environmental, affecting individuals of various ages, races, and genders. A dermatologist is the ideal medical specialist to consult for skin disorders since they can check the whole individual and consider their medical history. A multitude of [16] structures

and systems comprise the skin, the dermis, the epidermis, the subcutaneous tissues, the lymphatic vessels, the nerves, the blood vessels, and the muscles. Various germs may alter the skin's texture or generate color; allergic reactions; hidden bacteria; fungal growth on the skin; and microbes can also cause a skin disease.

In this research [17], we will further investigate how to leverage Convolutional neural networks are faster and more exact than traditional approaches for detecting and identifying LSDV. We also discussed potential applications for this new technology, including the advantages of using C NNs for this purpose. Using deep learning target detection approaches that use cloud and edge computing, this study [18] focuses on common skin disorders including acne and dermatitis. The methods chosen are You Only Look Once (YOLOv5), EfficientDet, M2det, and single shot multi-box detector (SSD). In the field of popular science dermatology, a skin disease detection system has been developed to facilitate the rapid diagnosis of skin ailments. We have [19] used excitation-emission matrices to analyze the fluorescence of tissue samples from degenerative skin disorders. These observations are part of a proof-of-concept research for a novel noninvasive optical diagnostic method for degenerative skin conditions. Collecting several images of healthy and LSD-infected cattle is an important part of our method [20]. In order to extract features from images, advanced learning algorithms like CNN are used for preprocessing. Following this, we will classify cow images as either healthy or infected with LSD using Machine Learning models like Support Vector Machines (SVM) and Random Forest Neural Networks.

3. METHODOLOGY

The technique for this work is intended to address the limitations of utilizing machine learning models to categorize skin disorders. It starts with the collecting and preparation of various datasets, followed by stringent data cleaning and augmentation to improve model performance. We then use sophisticated segmentation and feature extraction methods to correctly capture skin lesions' morphological and textural properties. Multiple machine learning models are developed and tested to determine the best illness categorization technique. Finally, the approach is evaluated to verify that the results are consistent and generalizable.

Dataset Collection: For this investigation, we used two primary datasets. The first dataset included 10,000 photos of melanoma, whereas the second featured 5,000 photographs of various different skin diseases, both malignant and benign. These datasets were collected from public sources and dermatological clinics to provide a diverse range of skin types and lesion presentations.

Data Pre-processing: The pre-processing step was critical in preparing the datasets for analysis. To standardize the collection, all photos were first scaled to uniform dimensions. Image augmentation methods such as flipping, rotating, and scaling were used to improve the model's resilience to changes in image orientation and size. Pixel normalization was then carried out to improve the convergence rate of machine learning systems.

Data Cleaning and Quality Assurance: To verify the dataset's quality and dependability, data cleaning processes were used. Dermatologists manually inspected pictures to eliminate duplicates and low-quality photographs. The photos were additionally annotated with condition-specific information to aid with supervised learning in later phases.

Class Imbalance Handling: Given the excessive amount of melanoma photos relative to other conditions, methods for addressing class imbalance were required. To produce synthetic data, we employed the Synthetic Minority Over-sampling Technique (SMOTE) as well as oversampling minority classes. This strategy helped to balance the dataset, preventing the model from being biased toward the more common classes. The minority data is replicated from the minority set of data in a traditional oversampling process. Although it increases the quantity of data available, it does not provide the classification model with any additional knowledge or variation. To address this problem, The author introduced the SMOTE which is the most effective and commonly used technique. For balancing the dataset, it generates fresh synthetic samples. SMOTE generates synthetic data using the KNN technique. It calculates the distance between the two sample points as well as the distance between the feature vector and its nearest neighbour. It then multiplies the distance by a value between 0 and 1 at random. At the computed distance, a new point is picked on the line segment. This procedure is repeated for each of the feature vectors that have been discovered. In terms of preventing over-fitting and under-fitting, SMOTE performs better than basic sampling approaches.

Segmentation: The segmentation process was crucial for precisely defining lesion borders. To accurately segment skin lesions, we used a variety of approaches such as MorphACWE, MorphGAC,

thresholding, edge detection, clustering, and semantic segmentation. This phase was critical for separating the regions of interest in the photos for further analysis.

3.1. Thresholding

Otsu multi-threshold image segmentation is a statistical technique that maximizes the variance between classes. Suppose we want to divide an image into K categories, so we need to find $K-1$ thresholds $T_1, T_2, \ldots, T_{K-1}$. The total between-class variance is defined as follows:

$$\sigma_b^2 = \sum_{k=0}^{k-1} \omega_k (\mu_k - \mu_T)^2 \quad (1)$$

Where, μ_T is the overall mean gray level, and μ_k is the mean gray level of a class/region. It is depicted in Eqs. (2) and (3).

$$\mu_T = \sum_{i=0}^{L-1} i.P(\mathrm{i}) \quad (2)$$

$$\mu_k = \frac{\sum_{i=T_{k-1}+1}^{T_k} i.P(\mathrm{i})}{\omega_k} \quad (3)$$

Where, ω_k represents the weight of class k, L is the number of gray levels, and (i) is the probability of gray level i.

$$\omega_k = \sum_{i=T_{k-1}+1}^{T_k} P(\mathrm{i}) \quad (4)$$

Where, $T_0 = -1$ and $T_K = L - 1$.

$$P(\mathrm{i}) = \frac{n_i}{N} \quad (5)$$

$$N = \sum_{i=0}^{L-1} n_i \quad (6)$$

Where, n_i is the number of pixels at gray level i.

3.2. Edge detection

The efficient prior edge detection methods assume that one deals with well-defined edges in an image and try to detect them. However, sometimes we deal with images that are highly focused. In such images, we are still interested in the detection of both the foreground object and the background smooth edges, as the focus was made artificially by a photographer, usually, for artistic reasons. According to our knowledge, there are no efficient methods that can deal with sharp and smooth edges at the same time.

3.3. K-means clustering

k-means is one of the famous unsupervised learning techniques for cluster analysis. Cluster analysis is used to aggregate or divide a dataset into several clusters according to the similarity value. For k-means, it needs to decide the number of clusters (k) in advance. It starts with randomly generated centroids and iteratively calculates new centroids to converge to the final clusters. There are four steps in k-means.

Step 1: The positions of k centroids are generated randomly.
Step 2: Each data point in the dataset will be assigned to its nearest centroid, and then new centroids are generated.
Step 3: To recalculate the new clusters by assigning all data points to their nearest centroids, and then new clusters are created.
Step 4: The process will be iterated between step 2 and step 3 until the stop criteria has been met.
Ward method uses the minimum within-cluster variance to generate clusters. The method uses the error sum of squares (ESS) to calculate the distances from data points to the mth cluster. Here, X_m is the centroid of the m^{th} cluster, and X_{lm} is the l^{th} data point in the m^{th} cluster. Fuzzy c-means are one of the famous clustering techniques that assigns every data point (n) in a dataset belonging to every cluster (m) with a certain fuzzy value. The fuzzy membership value is calculated as follows:

$$\mu_{nm} = X_n + m \quad (7)$$

Where, μ_{nm} is the fuzzy membership value for the data point X_n belonging to the m cluster.

3.4. Feature Extraction

After segmentation, feature extraction was carried out using both classic and sophisticated approaches. This work extracted color features using Histogram of Oriented Gradients (HOG) and analyzed texture using Local Binary Patterns (LBP). In addition, we used Gray-Level Co-occurrence Matrix (GLCM) and Gabor filters to capture texture information. Shape descriptors such as Hu moments and Zernike moments were used to describe the intricate patterns and geometries of lesions.

3.5. HOG

HOG is a feature extraction technique that extracts features from every position of the image by constructing logic histograms of the object from the images. In this paper, the images are first passed through the HOG descriptor for data size reduction and searching for an object to detect. Thereafter, the histograms are created and computed over the whole images that are retrieved from several video frames. These histograms are then appended into a single feature vector using the exponential equation 2^{ℓ}, representing the grid level (ℓ) for all cells along the dimensions. However, the correspondence on the whole input images between the vectors and histograms bins is ensured by limiting the level $(\ell)\ to \leq 3$ and computed using the following equation.

$$v = \mathcal{K} \sum_{i=1}^{\ell} 4^{l}; i \leq 3 \quad (7)$$

where v, denotes vector dimensions, $\mathcal{K}$ denotes bins, ℓ defines grid level. This equation ensures that all images that are extremely large and rich in texture are weighted the same as low texture images within the set parameters. It is also used to guard and control our algorithm against over-fitting.

3.6. LBP

In this study, a method was developed for extracting features from an image to identify emotions. We depend not only on the shadow effect of the grayscale images but also on using a new kernel-based method to enhance the shadow effect to extract the features that are flexible and classifier friendly. We have proposed two kernels on the LBP of an image to be more precise about the shadow and light effect of the face parts, which mainly decides the face's emotional states. In this step, the pre-processed image was taken and applied to the serial process is shown:

$$G(x,y) = \sum_{i=1}^{1} \sum_{j=-1}^{1} K_1\,(\mathrm{i,j}) * \mathrm{S}_1(\mathrm{x-i,y-j}) \quad (8)$$

Generally, $LBP_{(P,R)}$ is used in one radius on eight directional coordinates of the matrix value where P is the number of pixels to be considered and R is the radius from the central pixel. However, we used two LBP ($LBP_{(8,1)}\ and\ LBP_{(8,2)}$) and applied two kernel matrix to calculate the central pixel of that cell. Considering the first stage of the image, we have divided it into sub-cells where 3×3 for $LBP\,(8,1)$ and 5×5 for $LBP\,(8,2)$ with two proposed kernels. A sample 3×3 image segment has been shown and the model is shown for the first Kernel, where each matrix is a 45° rotation, and the central matrix is the 3×3 cell of the pre-processed image. Considering that S_1 denotes the grey estimation of the pixel point in the 3×3 neighborhood of the pre-processed image, and the kernel value of pixel points in the area is K_1, the central pixel can be obtained by applying the first rotation kernel with Eq. (8). Here, K_1 is eight rotational kernels with 45° rotations each. Therefore, Eq. (8) was applied eight times to obtain the value $q_0\ to\ q_7$, where (x,y) is the central pixel value, which will make the pixel matrix for 1^{st} Kernel. By converting the positive value as one and the negative value as 0, we obtain the central decimal pixel value. By using the sample image segment, we used to show the calculation to find the central pixel matrix values $q_0\ to\ q_7$.

$$H(x,y) = \sum_{i=-2}^{2} \sum_{j=-2}^{2} K_2\,(\mathrm{i,j}) * \mathrm{S}_1(\mathrm{x-i,y-j}) \quad (9)$$

The model for the second Kernel is shown each matrix is a 45° rotation, and the central matrix is 5 × 5 cells of the pre-processed image. Again, accepting that S_2 denotes the grey estimation of the pixel point in the 5×5 neighborhood of the pre-processed image and the kernel value of pixel points in the area is K_2, the value of the central pixel can be obtained by applying the second Kernel with Eq. (10).

$$BM(x,y) = \left(\sum_{i=-1}^{1} \sum_{j=-1}^{1} K_1(i,j) * S_1\,(x-i, y-j)\ AND \left(\sum_{i=-2}^{2} \sum_{j=-2}^{2} K_2(i,j) * S_2(x-i, y-j) \right) \right) \quad (9)$$

Similarly, kernel K_2 will have eight rotations with 45° each for obtaining $q_0\ to\ q_7$ values where $H\,(x,\ y)$ is the central pixel which will make the

pixel matrix for 2^{nd} Kernel. Once again, converting the positive value as one and negative value as 0, we acquire the central decimal pixel value. In the final stage, we have applied bitwise AND of $G(x, y)$, $H(x, y)$. The binary output value of a model is determined which tells to the nearby change data between the center point and the 8-neighborhood pixels. It counts the number of spatial transitions from 0 to 1 or 1 to 0. In this stage, the equation will be as follows:

$$BM(x,y) = G(x,y)\ AND\ H(x,y) \quad (10)$$

where BM (x, y) is the binary matrix, the values of which are defined as 1 if G (x, y) = H (x, y) = 1 or 0 if any of $G(x, y)$ or $H(x, y)$ is 0. We have used an assessment by applying a condition to find the output cell's central pixel in decimal in Eq. (11).

$$MSLBP(x_c, y_c) = \sum_{n=0}^{7} BM(w_n)2^n \quad (11)$$

Where, w_n corresponds to the neighboring binary value of the eight surrounding pixels of the binary matrix BM and $MSLBP(x_c, y_c)$ is the final central decimal pixel value. For one image, neighbor pixels are generally related; thus, the binary sequences of $MSLBP(p, r)$ of the various radius can be seen as described. After ascertaining all values from left to right, we have obtained a binary pattern for every cell of an image. Taking all weighted values into account, we have found a decimal number in symmetric neighbor sets for various coordinates (x, y). The grey values of neighbors that are not the focal region for matrices can be evaluated by commitment. After that, we discovered one histogram for each cell, then we have concatenated all those histograms from each cell into a one-linear histogram. There will be a two-dimensional matrix for each image of seven classes where rows represent the image index, and the column represents the features. This long concatenated histogram is the initially featured vector with many noises and mismatched values within a class. We have normalized the histogram data to solve this kind of problem, which shows good accuracy in validation test cases compared with the original feature vectors.

$$FV = uLBP\left(MSLBP_{(x,y)}\right) \quad (12)$$

3.7. GLCM

The GLCM has been proved to be a powerful approach for image texture analysis. It describes how often a pixel of gray level i appears in a specific spatial relationship to a pixel of gray level j. The GLCM defines a square matrix whose size is equal to the largest gray level N_g appearing in the image. The element P_{ij} in the (i, j) position of the matrix represents the co-occurrence probability for co-occurring pixels with gray levels $i\ and\ j$ with an inter-pixel distance δ and orientation θ. The author proposed 14 original statistics (e.g., contrast, correlation, energy) to be applied to the co-occurrence matrix to measure the texture features. The most widely used textural measures are considered in this study: energy (ENE), contrast (CON), entropy (ENT), and inverse difference (INV). Energy is a measure of the local uniformity. Entropy is inversely related to the energy, and it reflects the degree of disorder in an image. Contrast measures the degree of texture smoothness, which is low when the image has constant gray levels. The inverse difference describes the local homogeneity, which is high when a limited range of gray levels is distributed over the local image.

3.8. Gabor filter

The Gabor features have been used commonly in image processing and analysis. The Gabor filter, originally developed by Dennis Gabor, is a linear filter mostly employed edge detection, surface evaluation, feature extraction, object recognition, and many other applications. These filters possess optimum locality in the frequency and spatial domain effective for texture/surface mapping applications. These are types of bandpass filters that allow a certain frequency range while rejecting frequencies outside of it. Mathematically, certain parameters influence how the Gaussian filter will operate and how it will respond to different feature elements. A 2-D Gabor filter can be regarded as a sinusoidal signal with a Gaussian wave modulating specific frequency and orientation. The filter has two orthogonal components representing real and complex imaginary components. The two elements can be used individually or in a complex number. The equation is written as:

$$F(x, y;\ \sigma, \vartheta) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{u^2+v^2}{2\sigma^2}\right)\exp\left(i2\pi\frac{u}{\lambda}\right) \quad (13)$$

where, $u = xcos\vartheta + ysin\vartheta, and\ v = -xsin\vartheta + ycos\vartheta$. λ is the wavelength of the complex exponential signal, ϑ is the alignment of the normal to parallel lines of the Gabor filter, and σ is the scale parameter or standard deviation of Gaussian envelope. These parameters control the size and

shape of the Gabor function. The Gabor filter can be used for testing the building edges in test samples. The response of the Gabor filter for the test image is given as below:

$$G(x, y;\ \sigma, \vartheta) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{u^2+v^2}{2\sigma^2}\right)\cos 2\pi\left(\frac{u}{\lambda}\right) \quad (14)$$

Where, $G(x, y; \sigma, \vartheta)$ represents the maximum regions having similar characteristics with the filter. By using this information, the local feature points could be extracted. To do so, first, there is a need to search for the local maxima in $G(x, y; \sigma, \vartheta)\ for\ x \in \{1,2, \dots,\ N\},\ y \in \{1,2, \dots,\ M\}$.

3.9. Model Training

We used the pre-processed datasets to train five different machine learning models: ResNet/DenseNet, Generative Adversarial Networks (GAN), Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbor (KNN). The models were chosen for their capacity to handle the intricacies of skin disease categorization, and they were trained using segmented and feature-extracted datasets.

3.10. ResNet

Among the various models capable of classifying satellite imagery, we used the ResNet 18 (Residual Network) neural network, which is a convolutional neural network (CNN), renowned for its ability to train very deep networks using residual blocks. A residual block is equipped with a jump connection that bypasses one or more layers, effectively addressing the problem of evanescent gradients during training and improving the flow of information. In a residual block, the original input is added to the output of the layer block, creating a "skip connection". This allows the model to learn residual functions from the original input. A residual block can be mathematically represented as:

$$y = F(x,\ \ \{W_i\}) + x \quad (15)$$

Here, x is the input to the residual block. $F(x, W_i)$ is the transformation learned by the block (with weights W_i). The output y is the sum of the input and the learned residual. This formulation is based on the principle that instead of learning the full mapping H(x), the network learns the residual function:

$$F(x) = H(x) - x = H(x) = F(x) + x \quad (16)$$

This residual learning strategy simplifies optimisation, as learning the residual is often easier than learning the full transformation. More importantly, the identity (skip) connection allows the gradient to bypass one or more layers during back-propagation, enabling it to flow directly through the network. This mechanism effectively mitigates the vanishing gradient problem, which is common in very deep networks. As a result, ResNet architectures can be scaled to considerable depths (e.g., 50, 101, or 152 layers) without degradation in training performance or convergence issues. The jump connection adds the original input (x) to the output $(F(x, \{W_i\}))$, making it easier to learn residual functions. Jump connections allow the gradient to propagate more easily through the network during the back-propagation process, reducing the problem of the gradient disappearing and improving training stability. ResNet can be trained with hundreds or even thousands of layers without compromising performance. This has enabled ResNet to achieve outstanding results in computer vision tasks such as image recognition.

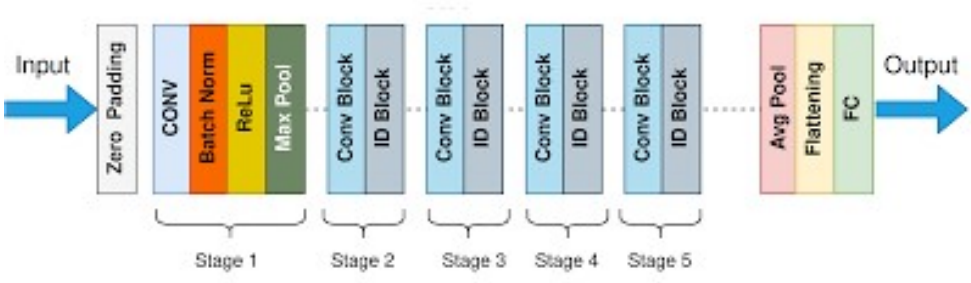

Fig 1: ResNet architecture

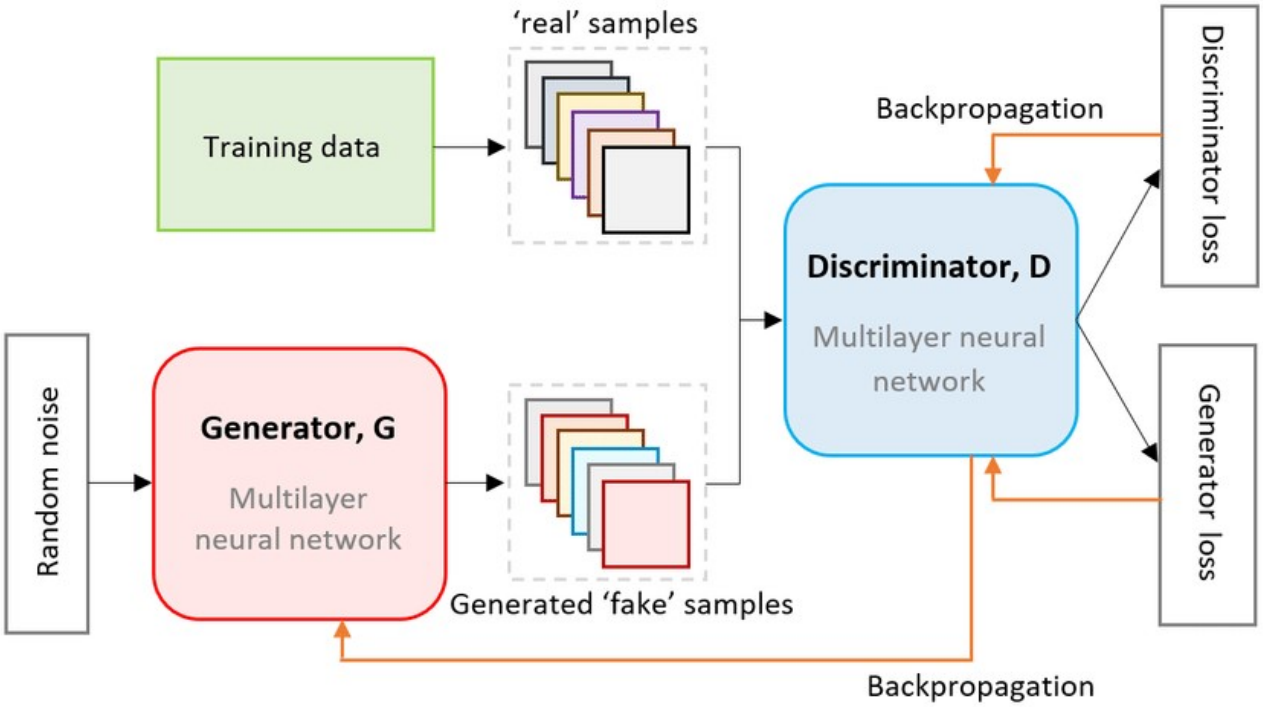

Fig 2: GAN architecture

3.11. GAN

GAN defines two neural networks: generator G and discriminator D. In a traditional adversarial network, the data distribution of the generator is defined as p_g over data x. A prior input $p(z)$ is used as an input noise variable. The mapping of the input noise to the data space is represented as $(z;\theta_g)$. The generator G is a differential function represented by an MLP with parameter θ_g. The second MLP used in this model, called the discriminator is represented as $(x;\theta_g)$ which outputs a scalar. (x) represents the probability that x came from data rather than from noise p_g. The variable $p_{dat}(x)$ refers to the original data distribution. $E_x \sim p_{dat}(x)$ is the expectation value function: it means that the expected value of x is assumed to be distributed over $p_{dat}(x)$. The value function (G, D) represents a min–max game between the generator and the discriminator. The discriminator is trained to maximize $(\max_D)$ the probability of assigning the correct label to the training example $log(x)$. Simultaneously, the generator is trained to minimize $(minG)$ the function $log(1 - D(G(z))$. In summary, a min–max game with value function (G, D) is defined as:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data(x)}}[\log D(x) + E_{z\sim p_z(z)}[log\left(1 - D\big(G(z)\big)\right) \quad (17)$$

The initial samples generated by G are not optimal enough to bypass the detection criterion of D, and are rejected by the discriminator. The generator keeps generating the adversarial samples and updating the parameters for the subsequent samples, and the generator learns a better evasion technique to fool the discriminator.

3.12. Support Vector Machine (SVM)

The SVM is a collection of theoretically powerful machine learning algorithms. The fundamental principle of SVM lies in creating an optimum hyper-plane also referred to as a decision boundary or optimal boundary that maximizes the distance between the nearest samples (support vectors) to the plane and effectively separates classes. The model seeks to find the optimal separating hyper-plane between classes by focusing on the training cases that occur at the edge of the class distributions, the support vectors, with the other training cases

effectively discarded. As a result, the approach can yield high accuracy with small training datasets that cut the costs of training data acquisition, this is considered as one of the pros of employing the algorithm. The basis of the SVM approach to classification is, therefore, the notion that only the training samples that lie on the class boundaries are necessary to separate classes. The construction/mathematical definition of optimum hyper-plane is significantly determined by the nature of the distribution of the training samples, i.e., whether the datasets are discernible effectively or separable with certain inevitable errors. When two classes are completely separable the decision boundary between them is represented by two equations $w.x_i + b \geq +1\ (for\ y_i = +1)$, and $w.x_i + b \leq -1\ (for\ y_i = -1)$; otherwise, it is defined by $w.x \geq 1 - \xi_i\ (for\ y_i = +1)$ and $w.x_i + b \leq -1 + \xi_i (for\ y_i = -1)$.

Where w is the norm to the optimal plane, x is training data (points) on hyperplane, b represents the bias, (ξ_i) is the slack variable that is the offset of the misclassified data from the optimal plane, and "y_i" is labeled data/classes. Moreover, in cases of indiscernible samples, the slack variable (ξ_i) and penalty value also known as cost value (C) are introduced to penalize the outliers, i.e., to regularize/compensate for misclassification/errors. Finally, the best hyper-plane that classifies the data with the largest gap between the support vector and the plane is obtained by minimizing the norm (w) function $F(w)$, for purely separable samples; and $F(w, \xi)$, for non-differentiable data, which are the most common data type in remote sensing. Accordingly, they are expressed mathematically:

$$F(w) = (w'w)^{\frac{1}{2}} \quad (18)$$

$$F(w, \varepsilon) = (w'w)^{\frac{1}{2}} + C\left(\sum_{i=1}^{k} \varepsilon_i\right)^1 \quad (19)$$

In the second function, the variable C is used to control the extent of the penalty to regularize misclassified training datasets, i.e., data lie on the wrong side of the optimal hyper-plane, and hence they play a significant role in influencing the accuracy and/or the capacity of the algorithm to generalize. If the value of C is set to be high, the penalty factor is high, which leads to over-fitting and diminishing the power of the model to generalize unseen data. On the other hand, if it is small, it can result in a smoothing, i.e., a biased model or under-fitting. Therefore, setting an appropriate value of this parameter is crucial. In this regard, the author suggests a moderate value to overcome the trade-off. The constant C is not the only parameter that affects the performance of the SVM algorithm. There exist other important parameters that profoundly control its effectiveness; these include the type of kernel functions, functions that are used to project/map linearly inseparable samples to higher dimensional space so that they can be classified linearly, and their parameters. Concerning kernel functions, despite the existence of numerous different types of kernels, such as linear, polynomial, radial basis function, and sigmoid, the radial basis function (RBF) kernel is the most effective and commonly used parameter in remote sensing image classifications and hence we adopted it for this work. For the RBF kernel to perform well, two parameters, namely, penalty value (C) and gamma (γ), should be meticulously chosen. The effect of gamma (γ), a parameter that controls the width of the kernel, on the SVM model using the RBF kernel is similar to C, in that if a high value is assigned to it, the model will overfit and does not generalize well. In addition to being the most widely used and robust classifier for remote sensing image classification, SVM has several advantages. As it is a non-parametric ML algorithm it efficaciously handles multi-modal datasets with hundreds of bands/channels, i.e., it is insensitive to problems associated with the dimensionality of data. It also works well with a smaller amount of training samples, given that good representative data that lie at the boundary of the class distributions are fed in and allow us to define the optimum hyper-planes. Moreover, unlike other advanced ML algorithms such as the neural network, it does not fall in local minimum at times, it always results in the global maximum.

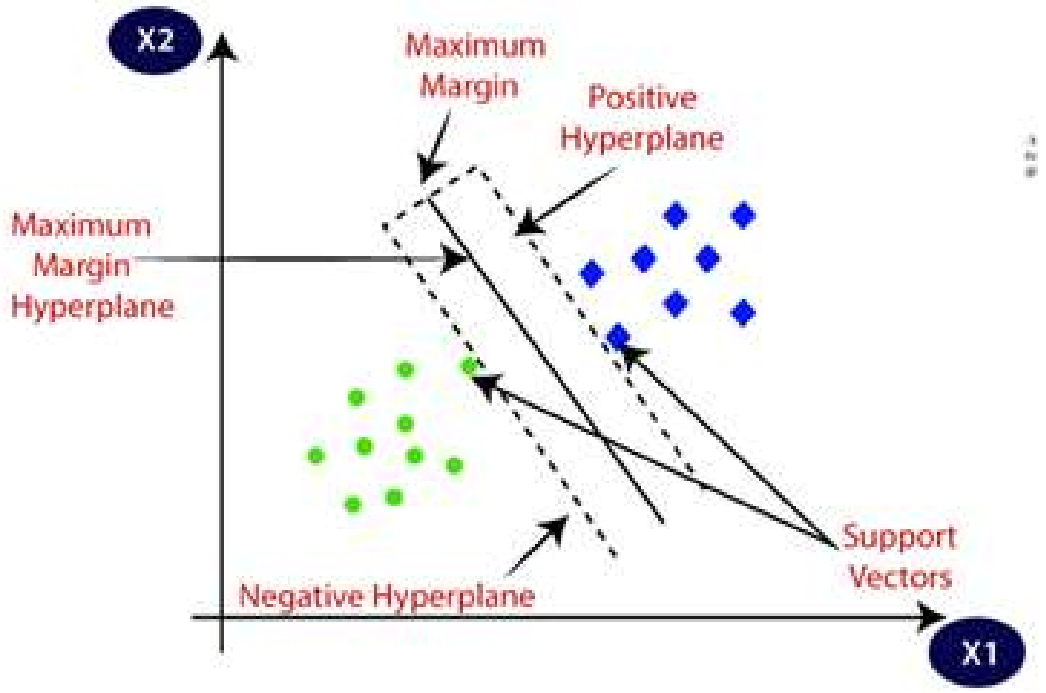

Fig 3. SVM architecture

3.13. Random Forest

The RF algorithm is a decision tree-based ensemble learning algorithm. RF utilizes the advantages of the decision tree algorithm's high speed and accuracy when dealing with classification problems by creating several decision tree models. Multiple decision trees have no association, and errors are mutually minimized, resulting in more accurate and robust classification findings. The first step in model construction is to choose a sampling method for generating a sub-dataset. Whether to use bootstrap sampling is the first hyper-parameter of RF, and the randomness of RF data is reflected here. The second step in model construction is to construct a decision tree, with the number of decision trees being the second hyper-parameter. There are four hyper-parameters in the decision tree: the maximum depth of the decision tree, the splitting standard, the minimum number of samples for internal node splitting, and the minimum number of samples for leaf nodes. The characteristics of each decision tree are picked at random, and the random forest's seventh hyper-parameter is the maximum number of selections. Because seven hyper-parameters impact the performance of the RF classifier, this study optimizes the combination of these seven hyper-parameters. Following the construction of the RF model, the test set samples are input, and each decision tree in the forest judged separately, with the classification results of the samples output. Finally, the results of all decision trees are combined using the voting mechanism, and the class with the most votes is the class to which the sample belongs.

Journal of Theoretical and Applied Information Technology
30th April 2026. Vol.104. No.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

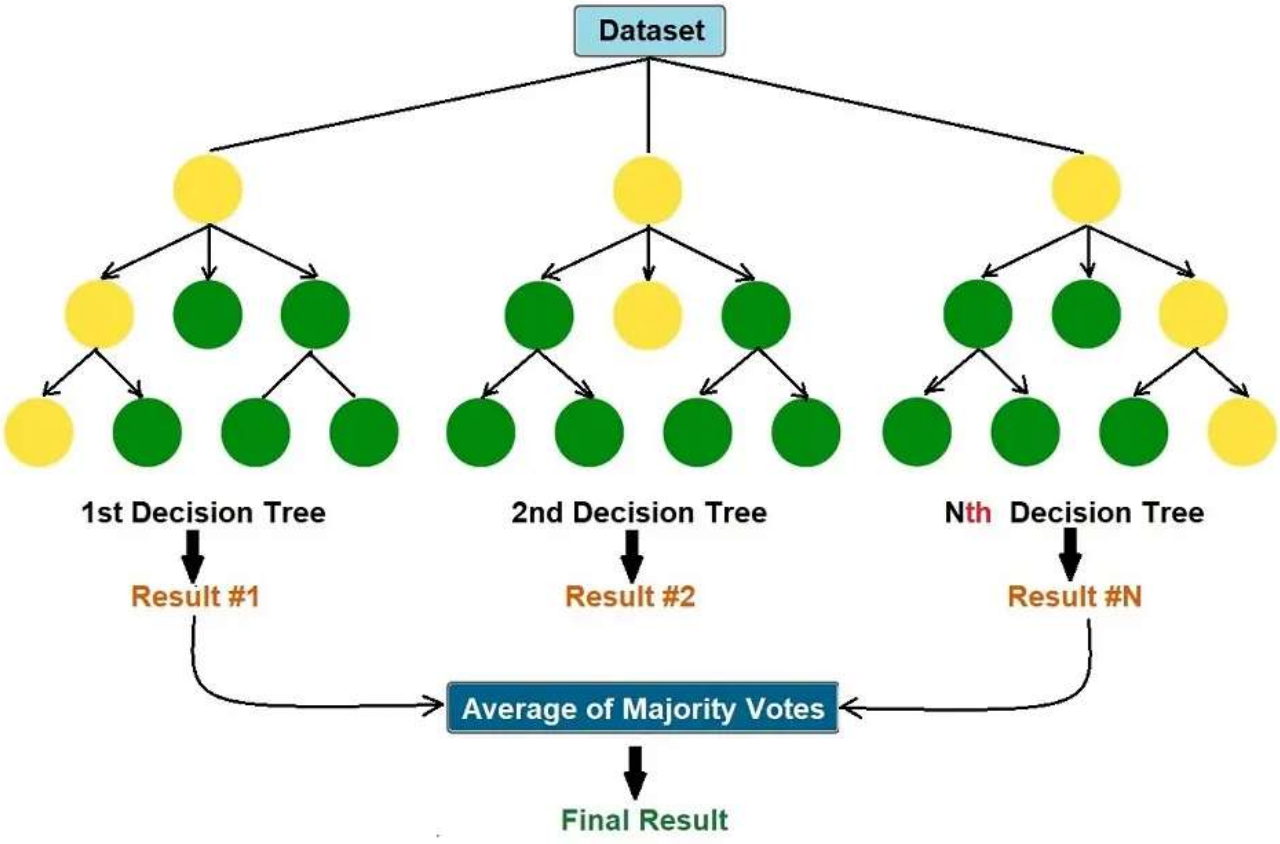

Fig 4. RF model

4. RESULT AND ANALYSIS

Our findings show that combining sophisticated pre-processing techniques, segmentation approaches, and machine learning models may accurately classify skin disorders. The preparation methods, which included picture augmentation, normalization, and data cleaning, were critical in improving the dataset's quality and consistency. These processes guaranteed that the models were trained on high-quality data, devoid of biases caused by class imbalance or low picture quality. SMOTE and oversampling significantly resolved class imbalance, enabling the models to function consistently across both common and unusual skin diseases. During the segmentation stage, approaches such as MorphACWE and MorphGAC, as well as thresholding and edge detection, were effective in properly identifying skin lesion borders. This exact segmentation was necessary for distinguishing the lesions from the healthy skin, allowing for more accurate feature extraction and categorization. The feature extraction procedure, which included Histograms of Oriented Gradients (HOG), Local Binary Patterns (LBP), GLCM, Gabor filters, and shape descriptors such as Hu and Zernike moments, offered a thorough depiction of the lesions' morphological and textural properties. This multifaceted approach to feature extraction enabled the models to capture the fine features required to discriminate between various skin diseases. To measure the performance of the models in categorizing diverse skin diseases, their F1-score, recall, accuracy, and precision were all calculated. This assessment was necessary to determine the best-performing model for skin disease categorization. To verify the models' resilience, validation procedures were used. Cross-validation was performed to evaluate model performance on various subsets of the data, and the results were examined to confirm the model's applicability to fresh data.

Table 1: Performance Metrics for iteration 1

Metric (%)	GAN	ResNet/ DenseNet	Random Forest	SVM	KNN
Accuracy	0.991	0.992	0.994	0.995	0.998
Precision	0.992	0.993	0.994	0.996	0.999
Recall	0.991	0.995	0.996	0.997	0.998
F1-Score	0.993	0.994	0.995	0.996	0.998
Sensitivity	0.989	0.991	0.993	0.994	0.997
Specificity	0.994	0.995	0.997	0.998	0.999

Table 2: Performance Metrics for iteration 2

Metric (%)	GAN	ResNet/ DenseNet	Random Forest	SVM	KNN
Accuracy	0.98	0.982	0.984	0.985	0.988
Precision	0.982	0.983	0.984	0.986	0.989
Recall	0.981	0.985	0.986	0.987	0.988
F1-Score	0.983	0.984	0.985	0.986	0.988
Sensitivity	0.989	0.981	0.983	0.984	0.987
Specificity	0.984	0.985	0.987	0.988	0.989

Table 3: Performance Metrics for iteration 3

Metric (%)	GAN	ResNet/ DenseNet	Random Forest	SVM	KNN
Accuracy	0.97	0.972	0.974	0.975	0.978
Precision	0.972	0.973	0.974	0.976	0.979
Recall	0.971	0.975	0.976	0.977	0.978
F1-Score	0.973	0.974	0.975	0.976	0.978
Sensitivity	0.979	0.971	0.973	0.974	0.977
Specificity	0.974	0.975	0.977	0.978	0.979

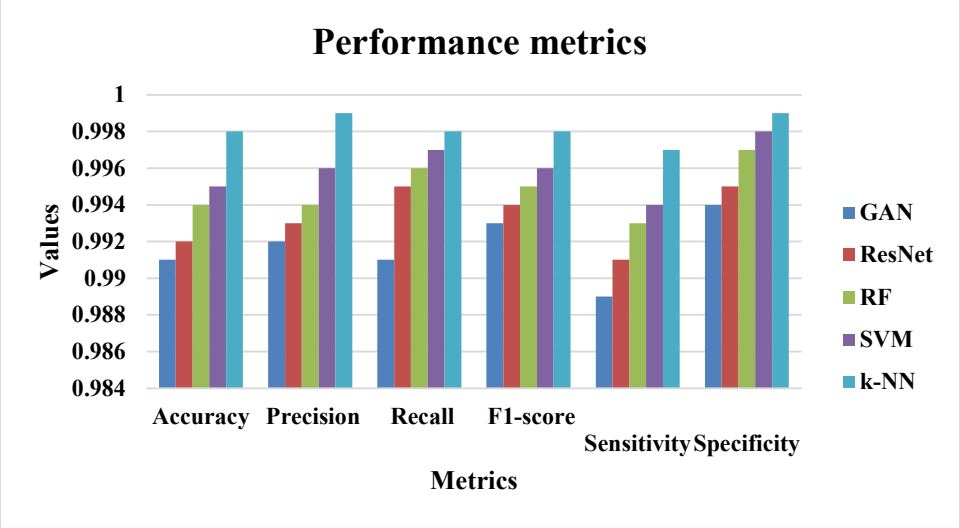

Fig 5. Performance Evaluation For Iteration 1

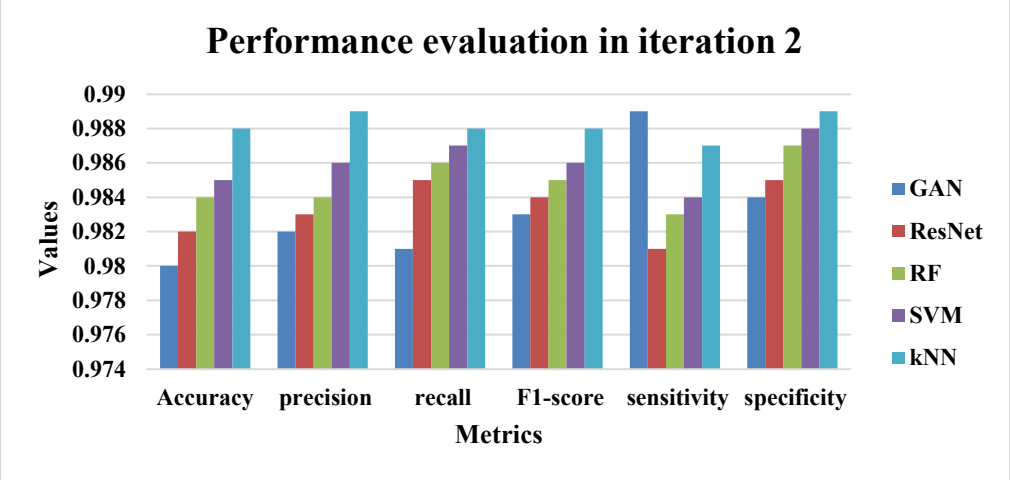

Fig 6. Performance Evaluation For Iteration 2

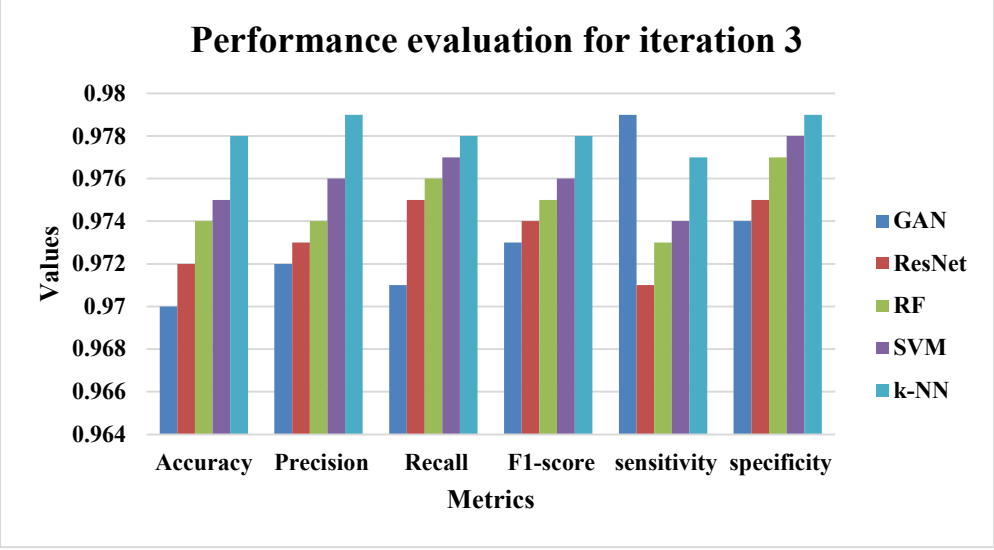

Fig 7. Performance Evaluation For Iteration 3

The efficacy of the machine learning models differed, with ResNet/DenseNet and GAN models outperforming more conventional models like RF, SVM, and K-NN. ResNet/DenseNet, with their deep learning architectures were very good at collecting the pictures' complex characteristics, resulting in greater classification accuracy as in Fig 5 to Fig 7 for both malignant and benign diseases. GANs also shown potential, notably in producing synthetic data other cancers, which is crucial for early identification and treatment. However, the models exhibited some heterogeneity in performance across various kinds of benign diseases, indicating that additional refining is required to discriminate between closely related benign lesions. In the framework of real-world medical applications, the results show that deep learning models may help dermatologists make more accurate diagnoses. The research also emphasizes the need of a well-pre-processed dataset since the quality of the input data has a direct influence on the model's performance. The findings imply that combining various image processing and machine learning approaches may considerably increase the accuracy and reliability of skin disease detection systems. Future research should concentrate on extending the dataset to encompass a broader range of skin disorders, further refining the models, and testing their practical value in clinical settings.

5. CONCLUSION

The study's conclusion emphasizes the major advances in dermatological diagnoses made possible by combining advanced image processing methods and machine learning models. We were able to establish a strong framework for the categorization of skin illnesses, spanning from benign conditions to malignant malignancies, by using large datasets and stringent pre-processing methods. The pre-processing step, which included picture scaling, augmentation, normalization, and data cleaning, was crucial in standardizing the data and improving the model's performance by ensuring that the input data was high quality and devoid of biases that may affect the results. The segmentation techniques used like MorphACWE, MorphGAC, and edge detection, were helpful in properly outlining lesion borders, which is required for exact categorization. These strategies enabled for the separation of lesions from healthy tissue, resulting in a clear focus for feature extraction. HOG, LBP, GLCM, Gabor filters, and shape descriptors were used to extract detailed morphological and textural information from the lesions. This multimodal strategy guaranteed that the that closely matched actual photos, increasing the model's capacity to generalize to previously unknown data. The assessment metrics showed that, although all models performed well on the skin disease classification test, deep learning models such as ResNet/DenseNet outperformed others in terms of overall accuracy and dependability. The accuracy and recall rates were especially high for melanoma and
models could efficiently distinguish between diverse skin states, which improved their overall accuracy and dependability. Deep learning architectures such as ResNet/DenseNet and GAN outperformed other machine learning models examined, especially in identifying malignant illnesses such as melanoma. These models outperformed standard approaches such as Random Forest, SVM, and KNN, demonstrating the ability of deep learning to handle the intricacies of skin disease categorization. The capacity of GANs to produce high-quality synthetic data also helped to alleviate class imbalance, improving the models' generalizability and performance. The study's results are significant for the future of dermatological diagnosis. The deep learning models' excellent accuracy and dependability indicate that they might be useful tools in clinical settings, assisting dermatologists in making better informed and faster diagnoses. The study also underlines the significance of high-quality, well-preprocessed datasets, since the effectiveness of these models is strongly reliant on the quality of the input data. Furthermore, the work lays a solid platform for future research, notably the investigation of more complicated skin disorders and the improvement of machine learning models' diagnostic capacities. To summarize, this study represents a big step forward in the use of machine learning to dermatology. It illustrates that with the correct mix of data preprocessing, segmentation, feature extraction, and sophisticated modelling approaches, extremely accurate and reliable skin disease classification systems may be developed. These tools have the potential to change dermatology by enabling earlier detection, more accurate diagnosis, and, eventually, improved patient outcomes. As the area evolves, more research and development will be required to realize the full potential of these technologies in real-world clinical settings.

REFERENCES

[1]. S. A. R. Naqvi, A. T. Mobashsher, B. Mohammed, D. Foong, and A. Abbosh, "Benign and Malignant Skin Lesions: Dielectric Characterization, Modelling and Analysis in Frequency Band 1 to 14 GHz," IEEE Transactions on Biomedical Engineering, vol. 70, no. 2, pp. 628-639, Feb. 2023, doi: 10.1109/TBME.2022.3199094.

[2]. G. S, N. S, and N. S, "Skin Disease Classification using Convolutional Neural Network," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2022, pp. 177-182, doi: 10.1109/ICAAIC53929.2022.9792785.

[3]. E. O. Oludipe et al., "Research Implication for Infectious Skin Disease and Phytotherapy in Developing Countries Based on 21st Century Bibliometric Trends," 2023 International Conference on Science, Engineering and Business for Sustainable Development Goals (SEB-SDG), Omu-Aran, Nigeria, 2023, pp. 1-11, doi: 10.1109/SEB-SDG57117.2023.10124639.

[4]. P. Ghadekar, A. Bongulwar, A. Jadhav, R. Ahire, A. Dumbre, and S. Ali, "Ensemble Approach to Solve Multiple Skin Disease Classification Using Deep Learning," 2023 International Conference on Device Intelligence, Computing and Communication Technologies (DICCT), Dehradun, India, 2023, pp. 296-301, doi: 10.1109/DICCT56244.2023.10110278.

[5]. V. Sharma and K. Kanwar, "Lumpy Skin Disease Detector," 2023 Seventh International Conference on Image Information Processing (ICIIP), Solan, India, 2023, pp. 806-810, doi: 10.1109/ICIIP61524.2023.10537770.

[6]. S. S. Vellela, R. D, C. Sowjanya, K. B. S K, L. Dalavai, and K. K. Kumar, "Multi-Class Skin Diseases Classification with Color and Texture Features Using Convolution Neural Network," 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), Gautam Buddha Nagar, India, 2023, pp. 1682-1687, doi: 10.1109/IC3I59117.2023.10398028.

[7]. R. G. Tiwari, H. Maheshwari, V. Gautam, A. K. Agarwal, and N. K. Trivedi, "Enhancing Skin Disease Classification and Privacy Preservation through Federated Learning-Based Deep Learning," 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2023, pp. 1-7, doi: 10.1109/ICAIIHI57871.2023.10489335.

[8]. P. Nagaraj, V. Muneeswaran, K. J. Krishna, K. Y. Reddy, J. R. Morries, and G. P. Kumar, "Identification of Skin Diseases using a Novel Deep CNN," 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2022, pp. 992-997, doi: 10.1109/ICESC54411.2022.9885330.

[9]. R. C. Suganthe, N. Shanthi, M. Geetha, R. Manjunath, S. Mithilesh Krishna, and P. Muthu Balaji, "Performance Evaluation of Transfer Learning Based Models On Skin Disease Classification," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-7, doi: 10.1109/ICCCI56745.2023.10128213.

[10]. E. Geetha Rani, M. Afeef Hussain, M. Azeezulla, M. Shandilya, and P. Susan Varughese, "Skin Disease Diagnosis Using VGG19 Algorithm and Treatment Recommendation System," 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India, 2023, pp. 1-8, doi: 10.1109/I2CT57861.2023.10126212.

[11]. N. A. Catur Andryani et al., "Analysis of Segmentation Method for Skin's Lesions Identification," 2023 International Conference on Information Technology and Computing (ICITCOM), Yogyakarta, Indonesia, 2023, pp. 369-374, doi: 10.1109/ICITCOM60176.2023.10442460.

A. Balasundaram, A. Shaik, B. R. Alroy, A. Singh, and S. J. Shivaprakash, "Genetic Algorithm Optimized Stacking Approach to Skin Disease Detection," IEEE Access, vol. 12, pp. 88950-88962, 2024, doi: 10.1109/ACCESS.2024.3412791.

[12]. K. A. Olatunji, A. Oguntimilehin, O. A. Adeyemo, O. M. Aweh, A. I. Abiodun, and O. A. Bello, "Skin Disease Classification using Deep Learning Methods," 2022 5th Information Technology for Education and Development (ITED), Abuja, Nigeria, 2022, pp. 1-8, doi: 10.1109/ITED56637.2022.10051236.

[13]. V. Arora, A. Johnson, L. Aggarwal, P. Goswami, and S. Alam, "Dermcare - Skin Disease Classification from Image," 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2023, pp. 933-

938, doi: 10.1109/ICCCIS60361.2023.10425226.
[14]. V. Anand and P. Shourie, "Ensemble Model for Seven-class Categorization of Skin Disease using Dermoscopy Images," 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT), Kollam, India, 2023, pp. 1463-1467, doi: 10.1109/ICCPCT58313.2023.10244957.
[15]. M. Vidhyalakshmi and M. Kanchana, "AMLGB-: Efficient Model for Skin Disease Detection and Classification using Adaptive Machine for Light Gradient Boosting," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2023, pp. 985-992, doi: 10.1109/ICSSIT55814.2023.10060982.
[16]. K. Tikarya, Y. V. Jain, and D. Bhise, "A review: Cattle Breed and Skin Disease Identification Using Deep Learning," 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2023, pp. 835-842, doi: 10.1109/ICCCIS60361.2023.10425776.
[17]. Z. Wu, G. Lai, X. Lin, and K. L. Chung, "A Real-Time Skin Disease Recognition System Based on YOLOv5 Algorithm," 2023 IEEE 6th International Conference on Electronic Information and Communication Technology (ICEICT), Qingdao, China, 2023, pp. 290-293, doi: 10.1109/ICEICT57916.2023.10245878.
[18]. T. I. Genova, E. G. Borisova, I. P. Angelov, P. P. Troyanova, and I. N. Terziev, "Fluorescence evaluation of tissue samples from skin collagen-related diseases," 2022 International Conference Laser Optics (ICLO), Saint Petersburg, Russian Federation, 2022, pp. 1-1, doi: 10.1109/ICLO54117.2022.9839958.
[19]. D. H. Patil, M. Pawar, M. Jaiswal, P. Rane, and S. Jagtap, "Lumpy Skin Disease Prediction Using Machine Learning," 2023 4th IEEE Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2023, pp. 1-5, doi: 10.1109/GCAT59970.2023.10353350.