International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 15, No.1 (Mar-24)

http://dx.doi.org/10.12785/ijcds/150188

Implementing Image Processing and Deep Learning Techniques to Analyze Skin Cancer Images

Snowber Mushtaq[1] **and Omkar Singh**[2]

[1,2]*National Institute of Technology, Srinagar, India*

Received 9 Jun. 2023, Revised 17 Nov. 2023, Accepted 24 Jan. 2024, Published 1 Mar. 2024

Abstract: Skin cancer is one of the few malignancies that seem to be completely incurable. If not diagnosed and maneuvered at the inauguration, it spreads to several human body components. It emerges while the skin cell is in contact with sunshine and develops primarily due to the rattling progression of skin cells. To facilitate simpler and more rapid rescue lives, a responsible automatic system for identifying skin lesions is essential for earlier detection. The strategy for successful skin cancer detection employs Image Processing and Deep Learning techniques. In this article, we have experimented with an amalgamation of various Image Processing Techniques including hair removal, median filter, Gaussian blur, and techniques of Deep Learning for the image of skin cancer study. Examining histopathology images requires a professional physician to categorize the images precisely. The Convolution Neural Network is the preliminary classifier utilized here. It is a cutting-edge learner-operated image classification since it can categorize images without depending on automatic feature extraction from an individual image. The main objective of this investigation is to enhance the heftiness of the classifier by approximating a distinguishable combination of different Image Processing and Deep Learning Techniques.

Keywords: Image Processing Technique,Convolutional Neural Networks(CNN), Hair Removal Technique

1. INTRODUCTION

According to the World Health Organization (WHO), cancer of the skin can be seen in one out of every three cases of cancer sickness, and in consonance with statistics from the Skin Cancer Foundation, one of every three people will suffer from cancer of skin later in life. A million people have been infected with skin cancer in places such as Australia, Canada, and the United States [1], and the number has been steadily increasing over the years. It has stood as the most common overall disease. And seem to have considerable negative effects on the health of the inhabitants globally. The skin is believed as one of the considerable all-encompassing organs in the mortal body, as it regulates our body temperature and rescues us from extreme heat and sunlight. It is a fat and moisture storage organ. When skin cells are overexposed to ultraviolet (UV) light, skin cancer can develop [2]. According to a 2017 study, skin cancer accounts for 1.79 of the international illness hurdle estimated in disability-adjusted vitality generation [3]. Skin cancer accounts for around 7% of current cancer patients globally [4].

Convolution Neural Networks (CNNs) are considered an expansion to artificial networks of neurons. And are presenting noticeable outcomes even in complicated assignments like processing of images, classification, and detection of objects [5]. They [6] are used in a lot of applications due to their increased and adequate enactment in diverse aspects of the pharmaceutical imaging approach, which include lesion classification, MR image fusion, panoptic analysis, tumor diagnosis, and breast cancer. CNN is a widely employed deep learning architecture for the image or optically corresponding tasks, such as object distinction and image classification, established on the Multi-Layer Perceptron (MLP) algorithm. The disparity separating these two techniques is that the CNN nodes are connected to some nodes in the previous layers, while MLP nodes are completely linked with all other nodes in the previous layers. This distinction allows CNN to achieve more usefulness in an image-related workload. They process better than one-dimensional data at a time. The convolution Layer extracts applicable knowledge from the input image [7]. A Pooling Layer is a subsequent layer behind the Convolution Layer. Its role is to reduce the overall complexity of the extracted element. A fully connected is the last layer, where all of the information from the prior neuron or layer ends at this layer.

Over the past years, computer analysis-based skin cancer revelation research has advanced enormously. By executing a competition event in 2018, "The International Skin Imaging Collaboration (ISIC)" occasion has evolved into a mandated standard in skin cancer detection. A diverse approach has been evaluated [8]. Experimenters have operated

E-mail address: snowbernitsri@gmail.com, omkarparihar@nitsri.com

diverse algorithms and techniques to augment the accuracy of diagnosis. When Fukushima and Le-Cunn presented the convolution neural network (CNN) image analysis smacked record ecstasy. CNNs are the most outstanding cutting-edge procedures for the image category since they mirror the human visual processing system. We bound our research to the Convolution Neural Network algorithm for skin cancer images, despite the availability of literature obtainable on image classification.

2. Dermoscopic skin cancer image categorization using CNN

Dascalu et al. [9] conducted research to ascertain how sonification using a crude skin magnifier and polarised light would affect the accuracy of diagnosis (SMP). The accuracy of the clinical output for SMP that emerges from Deep Learning processing of dermoscopic images after sonification suggests that the dermoscopy's quality isn't the primary factor impacting Deep Learning detection of skin cancer.

Kadampur et al. [10] developed a model that helps improve the accuracy of predicting skin cancer using a model-driven cloud architecture integrated with Deep Learning techniques. Tested on standard datasets, the proposed models offered a metric area under the curve of 99.7%.

Ameri et al. [11] introduced a deep-learning method for identifying skin cancer using the HAM10000 dermoscopic image database of skin lesions. The region under the receiver operating characteristic (ROC) arc for the presented instance stood at 0.91. Performance of classification of 84%, a sensitivity of 81%, and a specificity of 88% was found using a confidence score cutoff of 0.5.

Goyal et al. [12] addressed the improvements in digital image-based AI solutions for the detection of skin cancer, in addition to some issues and potential future advancements to all of this AI technology to assist skin specialists and increase their ability to identify cancer of the skin.

Brinker et al. [13] provided the first detailed analysis of recent research on using CNNs to categorize skin lesions. And highlights the difficulties that must be overcome in the future and why it is so difficult to compare the processes that are currently being used.

Dermoscopic images were used in comparative research by Kassani et al. [14] to classify skin lesion malignancy. To enhance image quality by increasing generalization capacity, pre-processing techniques like lighting emendation, contrast augmentation, and artifact removal have been applied. The tests reveal that ResNet50 performs better than its counterpart architectures VGGNet19, VGGNet16, Xception, and AlexNet, with classification results as strong as 92.08% and an F1-score of 92.74%.

Hekler et al [15] studied the advantages of integrating Artificial and human intelligence for diagnosing cancer related to the skin. For the multiclass assignment, the human-machine team had an accuracy of 82.95%. This was 1.36% better than the 81.59% accuracy of the CNN, which was the better of the two separate classifiers. Results show that combining human and artificial intelligence produces better outcomes.

Saba et al. [16] studied breast, brain, lung, liver, and skin cancer leukemia are among the human body cancers that can be detected utilizing machine learning approaches. The purpose of the study was to review, assess, categorize, and address these present advancements. They came to the conclusion that precision for each type of cancer is far from mature.

Haggenmüller et al. [17] researches have highlighted the three primary categories of images (dermoscopic, clinical, and histopathological WSIs). They found, that there is a need for rigorously controlled investigations that contrast the conclusions of AI-based classification models with the assessments made by doctors following in-person patient evaluations.

Adegun et al. [18] seek to provide a contemporary survey that would help researchers create effective algorithms that automatically and effectively identify melanoma from images of skin lesions. The classification performance of the skin lesion images is boosted by combining deep learning models with well-preprocessed and segmented images.

An algorithm centered on a binary categorization of lesions of the ulcerated skin (MM vs. melanocytic nevus) was designed by Yu et al. [19]. The outcomes of two professional dermatologists and two untrained general physicians were contrasted with the authors' CNN. It thus demonstrates the promise of computerized melanoma diagnosis using CNN for particular subtypes like acral MM on the feet and hands. The CNN attained mean sensitivity, specificity, and accuracy levels that were similar to those of qualified skin specialists (92.6%, 71.8%, and 81.9% vs. 96.6%, 67.0%, and 81.4%).

For the classification of skin lesions, an efficient data augmentation method and a pre-trained deep learning methodology are proposed by Bozkurt et al. [20]. To categorize skin cancer images, the Inception-Resnet-v2 hybrid network model is developed. The Inception-Resnet-v2 model obtained 95.09% efficiency in his investigation with an augmented dataset, comparable to 83.59% performance with the given dataset.

Vijayalakshmi et al.[21] constructed the model into three stages, which include data gathering and augmentation, model construction, and prediction. For a superior structure and accuracy of 85%, they combined several artificial intelligence techniques, including Convolutional Neural Network and Support Vector Machine, with image processing technology.

Alam et al. [22] employed AlexNet, InceptionV3, and RegNetY-320, Deep learning-based models, to categorize skin cancer. Additionally, various blends of hyperparameters were used to fine-tune the suggested framework. For both the imbalanced and balanced datasets, outcomes demonstrate that RegNetY-320 outscored InceptionV3 and AlexNet in the areas of accuracy, F1-score, and receiver operating characteristic (ROC) curve. The outcomes of the suggested architecture outperformed that of traditional approaches. The suggested framework might aid in the early detection of diseases [23], which might prevent deaths, lessen the need for unneeded biopsies, and lower expenses for patients, dermatologists, and healthcare professionals.

3. Deep Learning Methods for Predicting Skin Cancer

Skin cancer detection employs Deep Neural networks significantly. Regarding linkages between neurons, their architecture resembles the human brain. To resolve specific issues, the nodes collaborate. After being taught on specific assignments, artificial neural networks become authorities in the areas in which they were trained.

A. Skin cancer detection methods based on Artificial Neural Networks

An artificial neural network is a mathematical, nonlinear predicting technique. Its design was influenced by the natural framework of the human brain. Three layers of neurons constitute an ANN. The neurons that input data transmit data to the second or intermediate layer of neurons. The hidden layers comprise the connecting layers. The third layer(output layer) receives signals from intermediate neurons. Backpropagation learns the interactions between the input and output layers as computations at each layer. Skin cancer detection systems classify the collected characteristics using it. To learn the weights present at each link, a neural network employs a backpropagation or feed-forward architecture. Neural networks based on feed-forward design transfer data in a single direction.

B. Skin cancer detection methods based on convolutional neural networks (CNNs)

Convolution neural networks are a crucial subset of deep neural networks that have become prevalent in computer vision. It is utilized for image classification, collecting input images, and recognition of images. CNN is an excellent tool for collecting and learning local and global data because it uses simpler characteristics like curves and borders to generate deeper characteristics [24] like shapes and corners. Convolutional, pooling, and fully connected comprise CNN's hidden layers. In the detection, segmentation, and classification processes of medical imaging, CNN-based algorithms have achieved outstanding results.

C. Skin cancer detection methods based on Kohonen Self-Organizing Neural Networks (KNN)

One particularly well-known type of deep neural network is the Kohonen self-organizing map. Typically, a KNN has two layers. The initial layer in the 2-D plane is known as the input layer, whereas the second layer is known as the competitive layer. Both of these layers are completely interconnected, and each link is made between the first and second layer dimensions. Without being aware of the relationships between each input data component, a KNN can be employed for data clustering. KNNs lack an output layer as every node in the competitive layer also serves as the output node.

4. About Dataset

The dataset we are using is from kaggle.com. It consists of 3,600 images of benign and malignant. The data consists of two folders, each image having a size 224x244. The "International Skin Imaging Collaboration (ISIC)" database provided a dataset. ISIC is a recognized academia-industry collaboration forum earmarked to promote digital skin imaging to support decreasing the occurrence of delinquent diagnosis-related death. ISIC delivers a database of images of skin conditions that may be screened by illness class utilizing the built-in filter. Fig 1 shows certain of the dataset's images.

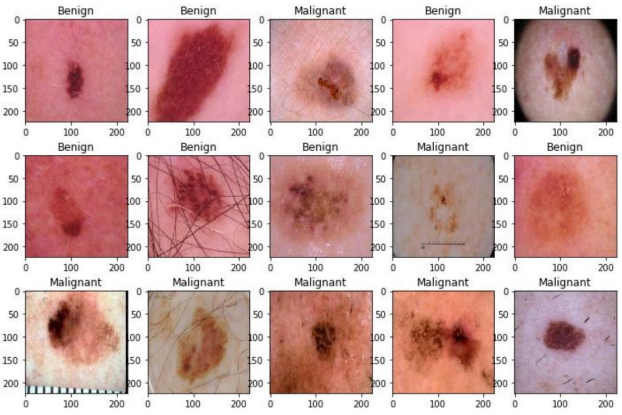


Figure 1. some of the images from the dataset.

5. Activation Function for Deep Learning

Deep Neural Networks can play multiple roles, such as voice investigation, voice or pattern identification, and object classification. It has proven valuable in a multitude of sectors. Structures with hidden layers are also included. This indicates that there are more levels than one [25],[26]. Because of their remarkable characteristics, Deep Learning architectures can acclimate in phases of real-world operation with real images. Only a few layers exist for the classification function in, for instance, the first Deep Learning model, In the LeNet5 model, there were around five layers [27]. Moreover, due to the necessity for exceptional network models, applicability, and possible computing potential inflation, there has been an expanded depth [28]. There are twelve layers in AlexNet [29], sixteen layers in VGGNet [30], 22 layers comprise GoogleNet [31], and one hundred and fifty-two layers in ResNet structure [32]. As a result, Stochastic Depth architectures now have around one thousand two hundred layers, something which

can still be taught. Going deeper into neural networks will offer you a more reasonable interpretation of the hidden layers, which will enable improving the enactment and training of a model. The activation unit evaluates the result of the network of neurons. It takes the output from the previous layer and transforms it into data that may be used as input for the next layer.

Activation Function Features:

1) Vanishing Gradient Problem:

Because of the gravitation of the network, slopes tend to evaporate, pushing the weight to zero, which is known as the vanishing gradient concern. As an outcome, don't desire its activation functions to drive the gradient to zero (0).

2) Centered-Zero:

The activation function's output should be regulated toward zero to avoid gradients from transferring in one direction.

3) Computational Expense:

Because they are used in each layer and, therefore, must be estimated millions of times in deep networks, they should be simple to compute algorithmically.

4) Differentiable:

Because neural networks are guided by gradient descent, the layers in the instance must be distinguishable or slightly some extent, comprehended. That is a requirement of the activation layer.

Some generally employed non-linear activation operations are:

1) Sigmoid Function:

The sigmoid function is a particular mode of the logistic activation function and is denoted by $\alpha(t)$ or sig(t). It is mathematically computed as:

$$\alpha(t) = \frac{1}{1 + e^{-t}}$$

It is also described as a squashing function, the range of the function is $(0, 1)$ and its domain is the collection of all real numbers. It is differentiable, continuous, and monotonic throughout, united with the characteristic that its derivative can be represented in terms of itself. As a result, while utilizing a backpropagation technique, it is simple to construct the updated equations for learning the weights in a neural network. When the neuron's activation saturates at either tail of 0 or 1, the gradient at the regions is about zero, which is a genuinely troublesome feature of the sigmoid function. The function predicts the likelihood as an output. The sigmoid is the best choice when considering the possibility of anything being only amid the range of 0 and 1. Fig 2 shows a graph of the sigmoid and its derivative.

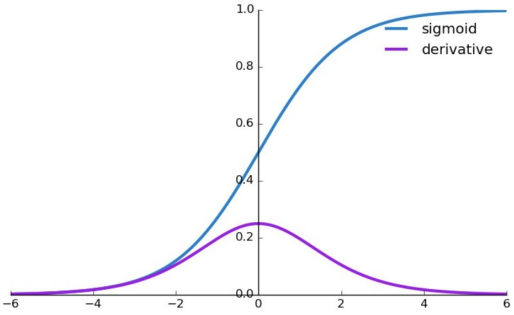


Figure 2. Graph of sigmoid function and its derivative.

2) Softmax function:

The Softmax is a type of logistic regression that normalizes a data point inside a vector of values that arrives with a probability distribution with cumulative aggregates of 1 or more. It generates values between the range [0,1], eluding binary classification and accommodating many classes or dimensions in a neural network model. So softmax is additionally seldom introduced as a multinomial logistic regression. It is represented as:

$$\sigma(z)_i = \frac{(e^z)^i}{\sum_{y=1}^{k} (e^z)^j}$$

3) softplus function:

The softplus function is expressed mathematically as:

$$f(x) = \ln(1 + \exp(x))$$

And

$$f'(x) = \frac{1}{1 + \exp(-x)}$$

It is frequently used for analytical applications. It intensifies the stability and execution of a Deep Neural Network because of its regularity and non-zero gradient. It avoids saturation and induces regularization by unboundedness in the uppermost boundary and boundedness in the below boundary. It is discouraged because of more computations than ReLU and its variants. Fig 3 shows a graph of softplus function.

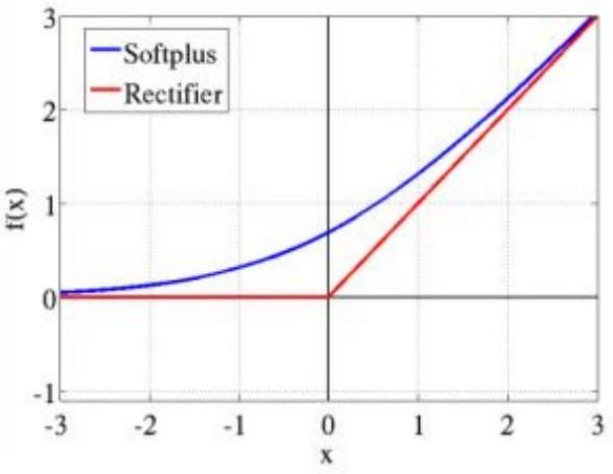


Figure 3. Graph of Softplus function.

4) Tanh function:

Tanh (hyperbolic tangent) is analogous to the logistic sigmoid, but it went a step further. The bounds of the tanh function are from (−1 to 1). It has a sigmoidal shape. (s-shaped). In the tanh graph, negative input will be charted as aggressively negative, while zero values will be plotted near zero. It is differentiable and monotonic. However, its derivation is not constant, and it is mostly used to classify groups. It suffers from a vanishing gradient problem and is always preferred over the sigmoid function. Fig 4 shows a graph of tanh and its derivative. It is mathematically defined as:

$$\tanh(x) = \frac{e^{x} - e^{(} - x\Big)}{e^{x} + e^{(} - x)}$$

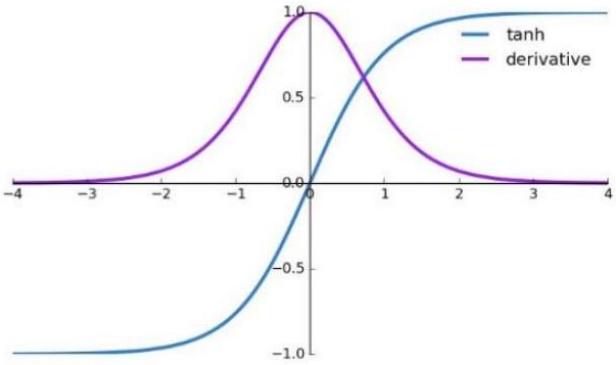


Figure 4. Graph of tanh and its derivative.

4) ReLU function:

The Linear activation function called rectified linear activation function, or ReLU, is a piece-wise linear function that thus directly outputs if it is real; otherwise, nothing if the output is less. It has evolved as the preferred activation function for many neural networks. It is easier to prepare and achieve better results with such a model that incorporates it. It is an activation function proposed by Fukushima in 1969 [17], with strong biological and mathematical reinforcement. It has been conditioned to improve the training of deep neural networks even more. It sets its threshold at 0, i.e. $f(x) = \max(0, x)$. When x is 0, it outputs 0.

It employs a simplistic formula:

$$f(x) = \max(0, x)$$

The disappearing gradient problem was created by the activation functions that existed before ReLU, such as the sigmoid or tanh activation functions, which were saturated and produced the vanishing gradient problem. This implies that high values of tanh and sigmoid will become 1.0, while low values will become −1 or 0. Likewise, the functions are extremely responsive to fluctuations in their input around the midpoint, considering such 0.5 for sigmoid and 0.0 for tanh. Whenever the value is 0 we have a dying activation function known as dying Relu.

5) Leaky ReLU:

The ReLU Activation functions function is strengthened by the Leaky ReLU function. Because the gradient is 0 for all input values smaller than zero, the ReLU activation function may generate a dying ReLU problem. As a result, the neurons within this region will be deactivated. Instead of setting the ReLU activation function to 0 for negative inputs(x), a small linear component of x is used here. The formula for this activation function is:

$$f(x) = \max(\alpha * x, x)$$

6) ELU:

ELU, also known as Exponential Linear Unit fixes some of the problems with ReLU's. For this activation function, an alpha α positive value is selected. It is a non-saturating activation function; As a result, the problem of erupting and disappearing slopes is avoided. Also, it does not suffer from problems like dying neurons. It has been confirmed to be more reliable than ReLU and its alternatives like Leaky-ReLU(LReLU) and Parameterized-ReLU(PReLU). It commences higher accuracy and lowering training times in a Network of neurons as compared to ReLU, and its variants. Its mathematical equation is:

$$ELU(x) = \begin{cases} x & \text{if } x > 0 \\ (e^{x} - 1) & \text{if } x < 0 \end{cases}$$

The ELU activation is slower to compute than the ReLU and its variants (due to exponential function). But during training, it convergences faster. However, at test time, an ELU network will be slower than a ReLU interface.

6. Loss Functions in Deep Learning:

The function of loss is essential to evaluate the loss of the given model. As a result of this weights can be corrected to lessen the loss on the next revaluation. The optimization approach requires that the error rate for this specific instance of the model be frequently simulated. Loss functions are characterized into two major categories - the Regression losses and the Classification losses.

The Regression Loss Functions.

The Predictive modeling of Regression includes predicting a real-valued quantity.

1) Mean Square Error:

The MSE attempts to measure how close a regression line would be to a set of data points. This is accomplished by multiplying the square of the distances between both the point and the regression line. Thus the squaring eliminates any contradictory signs. Larger variations are weighted more heavily. The average of a set of errors is observed here. The forecast is reliable when MSE is lower.

$$MSE = \frac{\sum_{i=1}^{k} \left(\left(t_i - \widehat{t_i} \right)^2 \right.}{k}$$

2) Mean Absolute Error:

The mean of the absolute differences between predictions and actual observations is determined by calculating as mean absolute error. It computes the measurement of error without exposing its direction. MAE is extra sturdy to outliers compared to MSE.

$$MAE = \frac{1}{2} \sum_{i=1}^{n} |x_i - x|$$

3) Huber Loss (Smooth Mean Absolute Error)

The Huber loss blends both MSE and MAE. On the condition that loss is higher, it changes the quadratic equation to linear. If the error is less than the cutoff then MSE will be used otherwise MAE can be used. The loss function is defined computationally as:

$$L = \begin{cases} \frac{1}{2}(y - \widehat{y})^2 & \text{if } |y - \widehat{y}| < \varepsilon^2 \\ \varepsilon |y - \widehat{y}| - \frac{1}{2}\varepsilon^2 & \text{otherwise} \end{cases}$$

Classification Loss Functions.

1) Binary Cross Entropy:

Cross-entropy comprises an ordinarily used loss function to use for classification difficulties. It calculates the difference in probability distributions between two sets of data. If the Binary cross-entropy is small, it intimates that the two partitions are similar. It computes a score that affords the negative average difference between the actual and predicted probabilities for predicting class 1. That score penalizes the probability based on the objective from the expected cost. The loss function is calculated mathematically as:

$$\text{Loss} = -1m \sum_{i=1}^{n} y_i \ln(\widehat{y}) + (1 - y_i) \ln(1 - \widehat{y_i})$$

$$\begin{cases} -y * \log(\widehat{y}) & \text{if } y = 1 \\ -(1 - y) * \log(1 - \widehat{y}) & \text{ify } = 0 \end{cases}$$

3) Hinge Loss:

One form of the loss function is hinge loss which is used to construct classifiers. It is practiced for maximum-margin classification, common prominently for support vector machines (SVMs) [33]. Although not differentiable, it's a curved function that makes it simple to operate with usual convex optimizers used in the Deep learning domain.

$$\text{Loss} = \sum_{j \neq y_i} \max\left(0, s_j - s_{y_i}\right) + 1$$

7. Optimizers in Deep Learning.

Optimizers are algorithms employed to depreciate loss function or hype production's efficacious. They are unsustainable mathematical functions on the model's trainable parameters, such as Weights and Biases. It reduces the losses by adjusting the weights and learning rate of the neural network.

Types of optimizers:

1) Gradient Decent:

Gradient Descent is the commonly used optimization algorithm. In linear regression and classification algorithms, it is extensively used. The gradient descent algorithm is often used in backpropagation in neural networks. It's a loss function's first-order derivative-based first-order optimization algorithm. It dictates how the weight values should be redesigned so that the function reaches a predetermined value. The loss is communicated from one layer to the next, and the weights are adjusted based on the losses, allowing the loss to be depreciated.

$$w_{\text{new}} = w_{\text{old}} - \alpha * \frac{\partial loss}{\partial w_{\text{old}}}$$

This formula computes the gradient for the dataset pro-

vided in a single update, making it a very time-consuming process. Furthermore, it necessitates memory resources and is computationally expensive.

2) Stochastic Gradient Descent:

Instead of exerting the entire dataset for each iteration in stochastic gradient descent, here, the batches of data are randomly selected. The Algorithm first selects the beginning parameters w and n. Then, randomly scrape the data at each repetition to arrive at an estimated minimum. The cost of constant updates is high in terms of computation, and they can also result in noisy gradients, which can cause the error to rise rather than fall.

$$\theta = \theta - \alpha \cdot \Delta j(\theta, x(i), y(i))$$

3) Mini-Batch Gradient Descent:

Instead of using all of the training data, only a subset of the dataset is used to compute the loss function in this case. Thus less iteration is required. As a result, it is more flexible than both stochastic and batch gradient descent algorithms. The technique seems to be more economical and enduring than the preceding variants. The cost function in mini-batch gradient descent is harsher than in batch gradient descent, but it is still smoother than in stochastic gradient descent. As a result, mini-batch gradient descent is typical and achieves a good mix of speed and precision.

$$\theta = \theta - \alpha * \Delta j(\theta; B(i))$$

Where B(i) is batch size.

4) Adagrad (Adaptive Gradient Decent):

Individual iterations of the adaptive gradient descent algorithm use different learning rates. The variation in the learning rate throughout training is determined by the variation in the parameters. Because real-world datasets comprise both sparse and dense features, this adjustment is hugely advantageous. The Adagrad algorithm updates the weights using the formula below:

$$w_{\text{new}} = w_{\text{old}} + \frac{\eta}{\sqrt{\alpha_t} + \epsilon} * \frac{\delta \text{ loss}}{\delta w_{\text{old}}}$$

Where α is the learning rate and η is constant.

It has a greater impact than gradient descent methodologies and their variants, and it converges faster. The AdaGrad optimizer has a flaw in that it aggressively and monotonically reduces the learning rate.

5) RMS Prop(Root Mean Square):

RMS prop proposed by Geoffrey Hinton. It is a gradient-based optimization technique used in training neural networks. It is a stochastic strategy for mini-batch learning. It uses an adaptive learning rate rather than ministering the learning rate as a hyperparameter. Hence the learning rate changes over time.

$$V_{db} = \beta \cdot V_{db} + (1 - \beta) \cdot dw^2$$
$$V_{dw} = \beta \cdot V_{dw} + (1 - \beta) \cdot db^2$$

$$W = W - \alpha \cdot \frac{dw}{\sqrt{V_{dw}} + \epsilon}$$
$$B = B - \alpha \cdot \frac{db}{\sqrt{V_{db}} + \epsilon}$$

6) Adam(Adaptive Moment Estimation):

Adam is a gradient-based first-order algorithm. Its stochastic objective function is based on adaptive lower-order moment estimations. The updated directive is ascertained by the first moment and normalized by the second moment. The process is too fast and converges too quickly. It corrects high variance and vanishing learning rates. However, it is computationally costly. It has the following first and second-order momentum:

$$\widehat{y_t} = \frac{m_t}{1 - \beta_1^t}$$
$$\widehat{v_t} = \frac{v_t}{1 - \beta_2^t}$$

Where M(t) is the first moment which is the Mean and V(t) is the second moment which is the uncentered variance of the gradients.

Update the parameters:

$$O_{t+1} = O_t - \frac{\eta}{\sqrt{\widehat{v_t} + \epsilon}} \cdot \widehat{m}_t$$

8. Proposed System

The technique of integrating Convolutional Neural Networks and Image Processing is illustrated in Figure 5. The given data is split into training, and testing data with each utilizing 70% and 30%, of the data respectively. Here the input images are acquired from the data set, then after applying different image processing techniques. The results are received by Convolutional Neural Networks in order to get the results. Our procedure is thoroughly explained in the following steps:

a) Input Image:

The system's input images are acquired from the available datasets ISIC containing 3,600 images of benign and

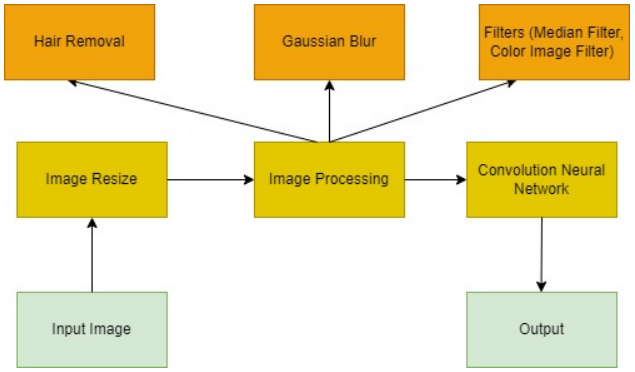


Figure 5. Proposed system.

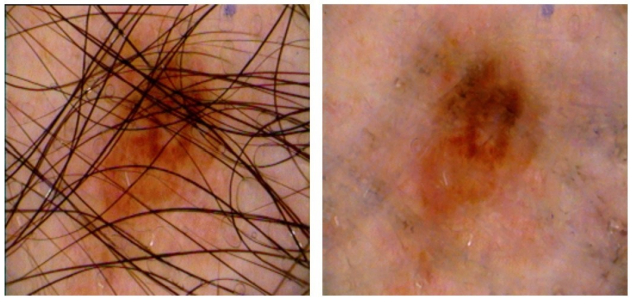


Figure 6. a)Shows Dermoscopic image before applying Dullrazor hair removal Technique. b) Shows Dermoscopic image after applying Dullrazor hair removal Technique.

malignant images.

b) Image Resize:

Images need to be pre-processed as they are obtained in any lighting conditions. Image scaling is part of the pre-processing in this case. A crucial stage in image pre-processing is image resizing. Algorithms that use Deep Learning learn more quickly from tiny images. The neural network learns four times as many pixels in a larger input image, which extends the architecture's learning period.

c) Image Preprocessing:

We used a combination of various techniques for Image Processing in order to get better outcomes. These consist of:

1) Hair Removal Technique

Utilizing pre-processing software, this technique removes hair from images. Here we have used the DullRazor Technique in order to perform the same. The DullRazor carries out the subsequent actions:

i) By using a universal gray-scale morphological closing procedure, the dark hair positions are identified.

ii)It confirms that the pixels of hair have a long structure and are thin, and it substitutes a bilinear interpolation for the confirmed pixels.

iii) With an adaptive median filter, it smooths the hair pixels that were changed.

The results of the hair removal approaches are shown in Figure 6.

2) Gaussian Blur

In order to minimize noise, an image creates the Gaussian blur feature by blurring (smoothing) the image with a function called a Gaussian blur. It could be compared to a non-uniform low-pass filter that keeps low spatial frequency whilst lowering picture noise and unimportant details. Typically, a Gaussian kernel is convolved with an image to accomplish it. In 2-D form, this Gaussian kernel is written as:

$$G_{2D}xy\sigma = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Here x and y are the indices of position and σ is the standard deviation. The Gaussian distribution's dispersion around the median is controlled by the value of σ, It controls how much a pixel's surroundings are blurred.

2) Filters

a)Median Filter

A median filter is a nonlinear filter where each result item is determined as the median of the number of input examples below the window of results. The outcome is the central weight behind the input values that had been ordered. Typically, odd numbers of strikes are utilized. A horizontal window with three taps is frequently used for median filtering; rarely, five or even seven taps are used. Noise reduction can benefit from median filtration.

a)Color Image Filter

The performance of the imaging system is hampered by noise that affects the visual data and degrades the visual appeal of an image. A wide range of application fields place a high priority on the creation of beautiful, good-quality color images. That implies image filtering as noise frequently taints images taken with sensing equipment and relayed through networks of communication. Thus, Color picture filtering becomes a crucial component of any image processing technique when the final image is used for manual or automated analysis.

3) Convolution Neural Network

The Deep Neural Network known as Convolution Neural Network (CNN) or ConvNet, can generalize more effectively than networks with fully connected layers as a result of its profound feed-forward framework [34]. CNN is the idea of biologically influenced sensors with a hierarchy for features [35]. It can effectively recognize features from a

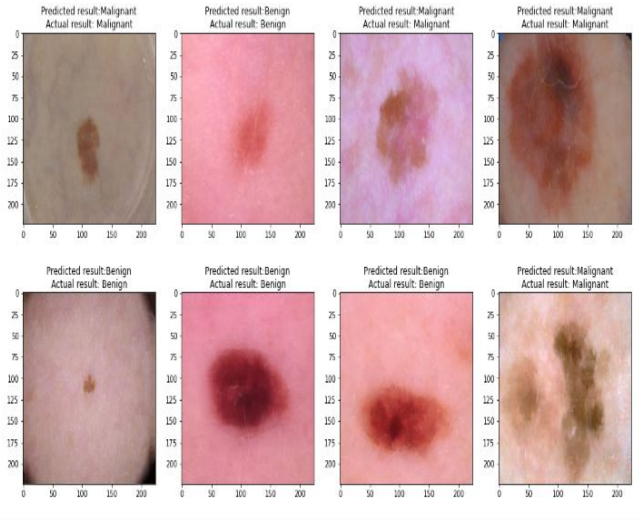
Network in order to obtain results. Implementing objects and acquiring very abstract features [36]. The notion of implementing weight sharing, which drastically reduces the number of parameters that need training and improves generalization, is the main motivation to utilize CNN. Convolution, pooling, and fully connected layers are the three kinds of layers (or building elements) that make up a typical CNN [37]. In the first two layers, the feature extraction process is carried out by convolution and pooling layers, and in the third place a completely interconnected layer is called fully connected, the extracted features are transformed into the output, for classification. [38]. A convolution layer is essential in CNN, which comprises several different mathematical processes, including the specialized linear operation known as convolution.

9. Results and Discussion

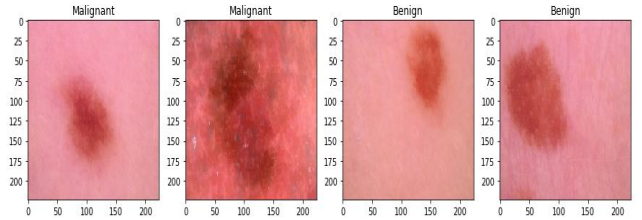
We have conducted the experiment using different approaches:

a) First we experimented without using image processing. Here first we resized the image, then after passing through the CNN (Convolutional Neural Network). The results are illustrated in Figure 7.
b) Figure 8 illustrates the results obtained after resizing hair removal and then Gaussian blur. The images after processing were then passed to the Convolution Neural Network to obtain results.
c) Figure 9 illustrates the results obtained after resizing hair removal and then color image filtering. The images after processing were then passed to the Convolution Neural Network to obtain results.
d) Figure 10 illustrates the results obtained after resizing hair removal and then Median filtering. The images after processing were then passed to the Convolution Neural Network to obtain results.
e) Figure 11 illustrates the results obtained after resizing hair removal, Median filtering, and then Gaussian filtering. The images after processing were then passed to the Convolution Neural Network to obtain results.
After observing the results, we conclude that integrating neural Deep Learning and the processing of image techniques improves the results.

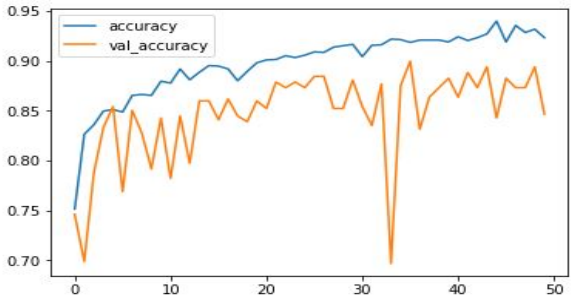
Table 1 Shows a comparison of various techniques applied. After observation, it is seen that image processing techniques with CNN improve the overall accuracy compared to CNN without image processing techniques.



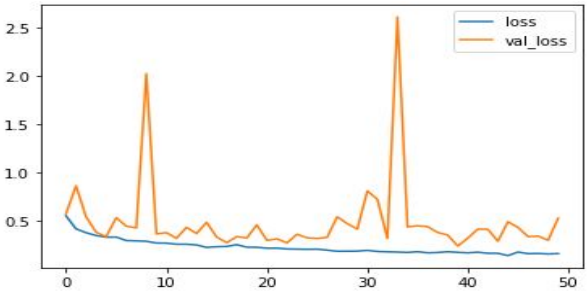
(a) Some of the images from the Dataset



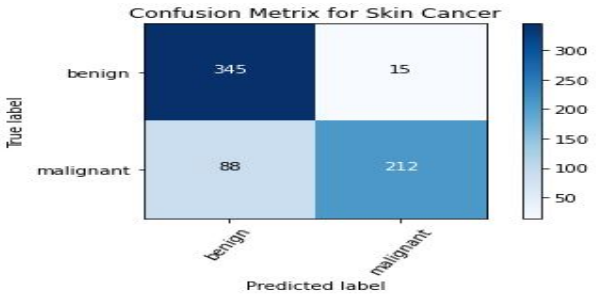
(b) Some of the images after resizing



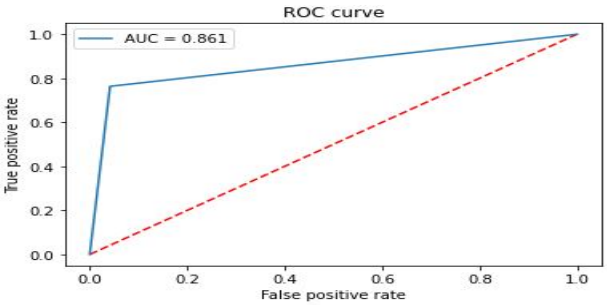
(c) Accuracy Graph



(d) loss graph

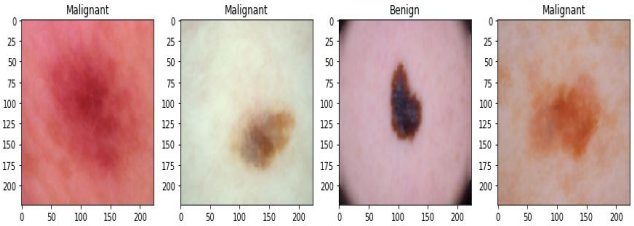


(e) Confusion matrix before normalization

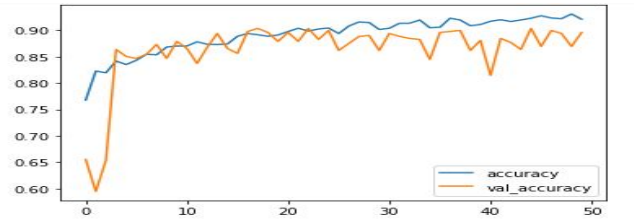


(f) ROC Graph

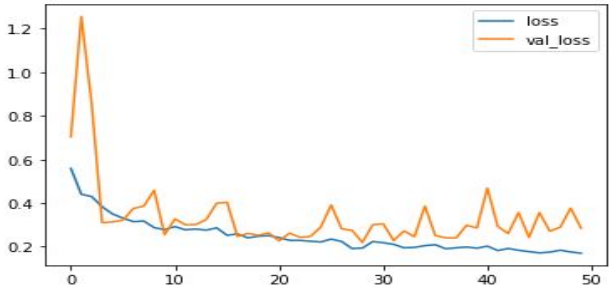
Figure 7. Results above are achieved after Resizing and applying CNN Technique



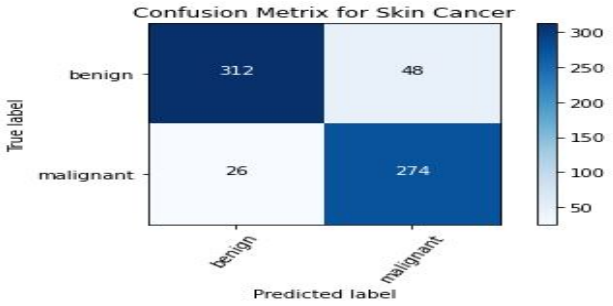
(a) Sample images after resizing, applying hair removal and Gaussian Blur Technique.



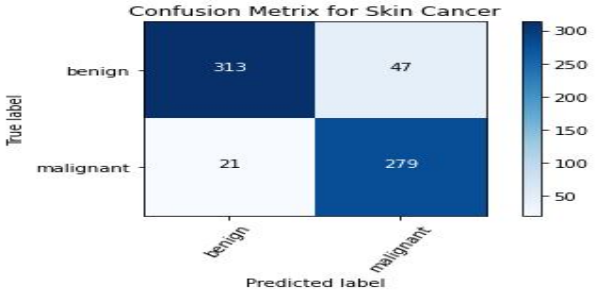
(b) Accuracy Graph



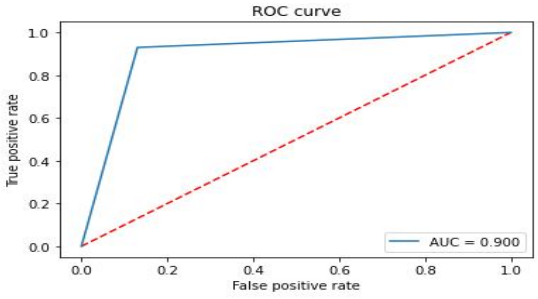
(c) Loss Graph



(d) Confusion Matrix Before applying Normalization

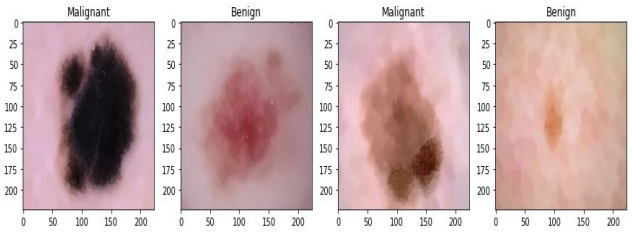


(e) Confusion Matrix After applying Normalization

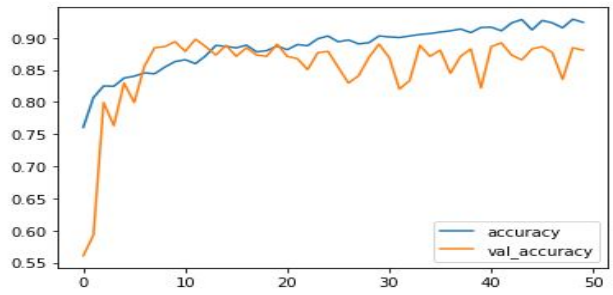


(f) ROC curve.

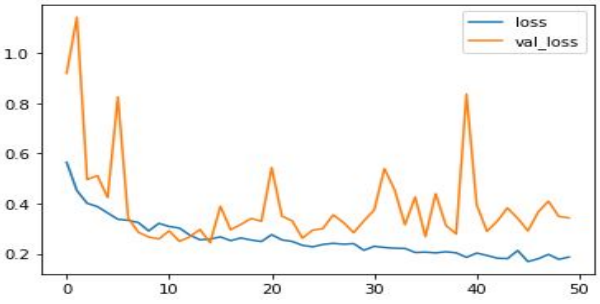
Figure 8. Results (a to f) achieved using CNN after applying Gaussian Blur and Hair removal Image Processing Technique



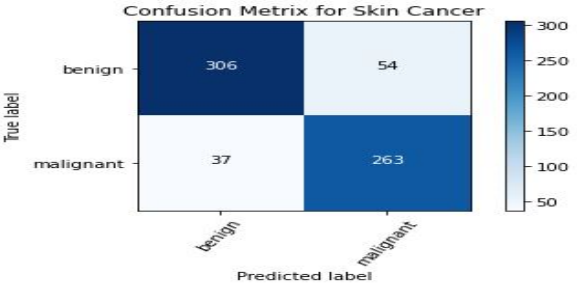
(a) Some of the images from the Dataset after resizing, color image filtering, and hair removal.



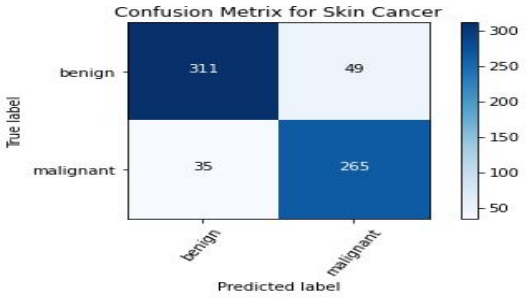
(b) Accuracy Graph



(c) Loss Graph

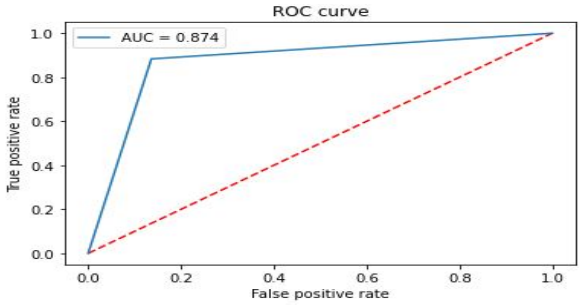


(d) Confusion Matrix Before applying Normalization

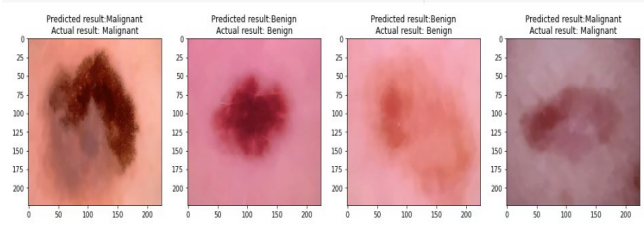


(e) Confusion Matrix After applying Normalization

(f) Precision Recall F1 score.

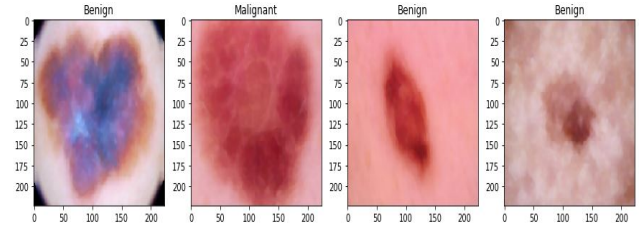


(g) ROC curve.

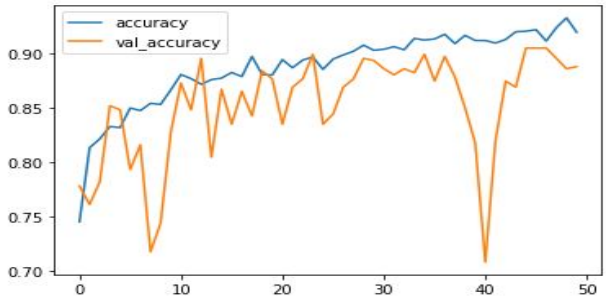


(h) Some of the predicted images.

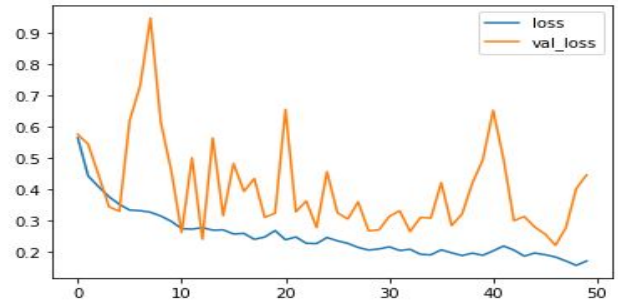
Figure 9. Results (a to h) achieved using CNN after applying hair removal and color image filtering process of Image Processing Technique



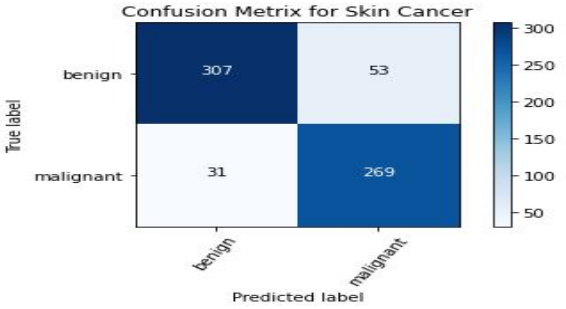
(a) Some of the images from the dataset after resizing, color image filtering, and hair removal.



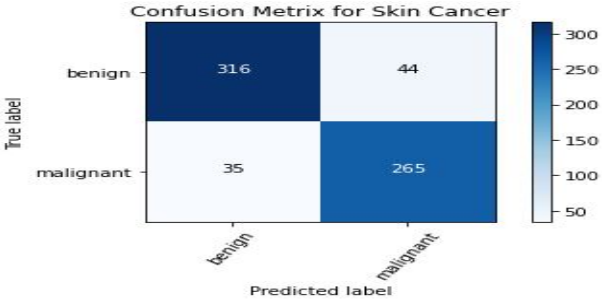
(b) Accuracy Graph



(c) Loss Graph



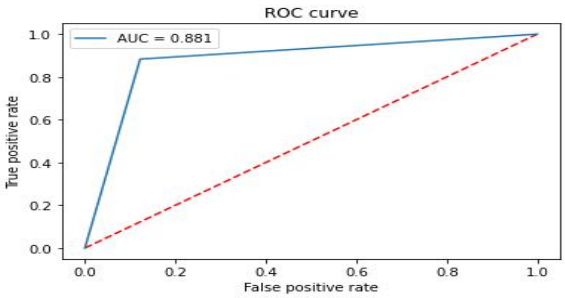
(d) Confusion Matrix Before applying Normalization



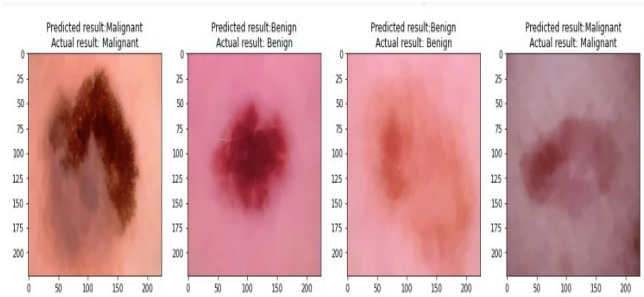
(e) Confusion Matrix After applying Normalization

```
'              precision    recall  f1-score   support\n\n           0       0.90      0.88      0.89       360\n
1       0.86      0.88      0.87       300\n\n    accuracy                           0.88       660\n   macro avg       0.
88      0.88      0.88       660\nweighted avg       0.88      0.88      0.88       660\n'
```

(f) Precision Recall F1 Score.

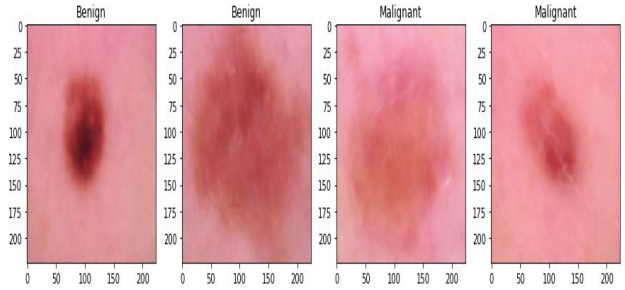


(g) ROC curve.

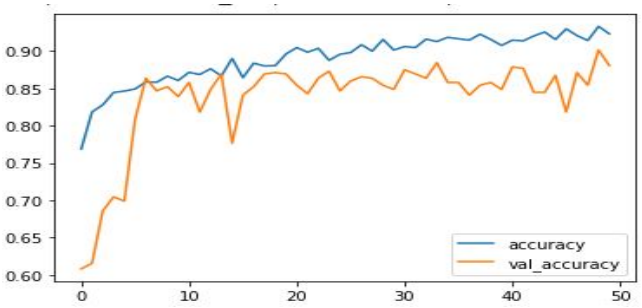


(h) Some of the predicted images.

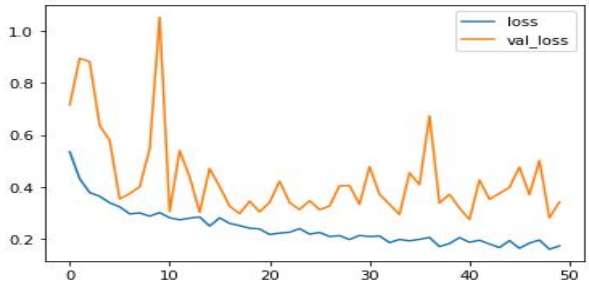
Figure 10. Results (a to h) achieved using CNN after applying hair removal and median filtering Image Processing Technique



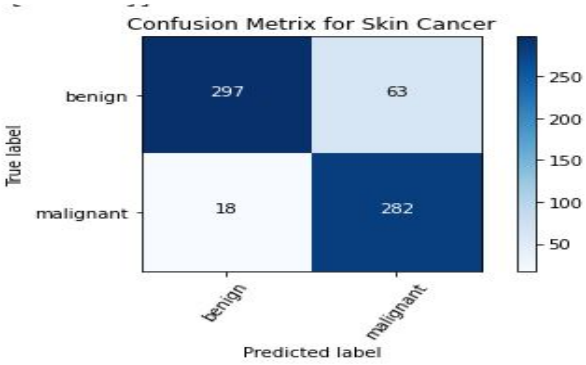
(a) Some of the images from the dataset after resizing, color image filtering, and hair removal.



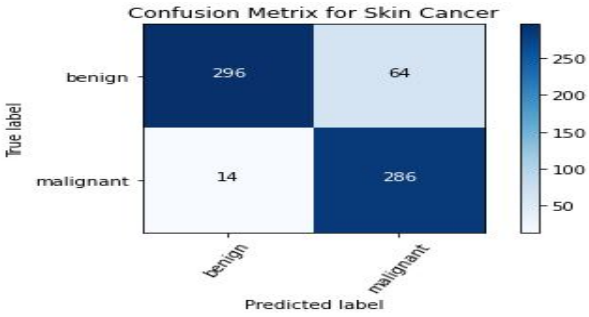
(b) Accuracy Graph



(c) Loss Graph



(d) Confusion Matrix Before applying Normalization

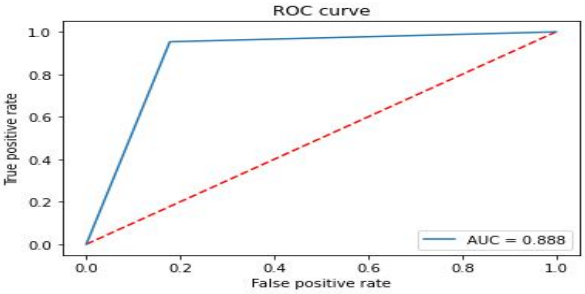



```
from sklearn.metrics import classification_
```

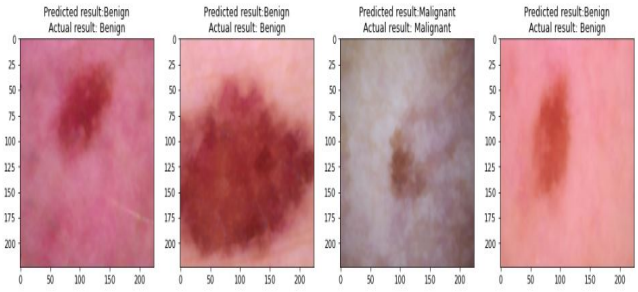
(e) Confusion Matrix After applying Normalization

```
'              precision    recall  f1-score   support\n\n           0       0.95      0.82      0.88       360\n
1       0.82      0.95      0.88       300\n\n    accuracy                           0.88       660\n   macro avg       0.
89      0.89      0.88       660\nweighted avg       0.89      0.88      0.88       660\n'
```

(f) Precision Recall F1 Score



(g) ROC curve.



(h) Some of the predicted images.

Figure 11. Results (a to h) achieved using CNN after applying hair removal and median filtering then Gaussian filter Image Processing Technique

TABLE I. Comparison table

Technique	AUC
CNN	.861
Gaussian Blur + Hair Removal Technique + CNN	.900
Hair Removal Technique + Color Image Filtering + CNN	.874
Hair Removal + Median Filtering + CNN	.881
Hair Removal + Median Filtering + Gaussian Filtering + CNN	.888

10. Conclusion and Future Work

Compared to current methods, the methodology suggested in this work is more suitable for identifying and diagnosing skin issues. In this paper, we explore Image processing and Deep Learning Methods for skin cancer detection. Our model's preliminary basis serves well because of the combination of Image processing and Deep Learning.

Our future goal is to increase testing precision while working on real-time skin lesion diagnosis. If larger datasets are accessible, we also intend to use our suggested model to classify skin cancer images on those larger datasets. The planned work is anticipated to assist the dermatologist in more accurately and quickly examining and categorizing the type of skin cancer. It will also help lower the overall expenses of a skin cancer diagnosis.

References

[1] K. M. R.L. Siegel and A. Jemal, "statistics cancer statistics 2019 companion, ca cancer," *Cancer. statistics*, pp. 7–34, 2019.

[2] UnderstandIng Skin Cancer - A Guide for People with Cancer their Families and Friends, 2018,[online] Available: https://www.cancer.org.au/about-cancer/typesof-cancer/skincancer.html.

[3] C. Karimkhani, R. P. Dellavalle, L. E. Coffeng, C. Flohr, R. J. Hay, S. M. Langan, E. O. Nsoesie, A. J. Ferrari, H. E. Erskine, J. I. Silverberg *et al.*, "Global skin disease morbidity and mortality: an update from the global burden of disease study 2013," *JAMA dermatology*, vol. 153, no. 5, pp. 406–412, 2017.

[4] T. Gansler, P. A. Ganz, M. Grant, F. L. Greene, P. Johnstone, M. Mahoney, L. A. Newman, W. K. Oh, C. R. Thomas Jr, M. J. Thun *et al.*, "Sixty years of ca: a cancer journal for clinicians," *CA: a cancer journal for clinicians*, vol. 60, no. 6, pp. 345–350, 2010.

[5] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[6] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[7] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis *et al.*, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[8] A. Masood, A. Ali Al-Jumaily *et al.*, "Computer aided diagnostic support system for skin cancer: a review of techniques and algorithms," *International journal of biomedical imaging*, vol. 2013, 2013.

[9] A. Dascalu and E. David, "Skin cancer detection by deep learning and sound analysis algorithms: A prospective clinical study of an elementary dermoscope," *EBioMedicine*, vol. 43, pp. 107–113, 2019.

[10] M. A. Kadampur and S. Al Riyaee, "Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images," *Informatics in Medicine Unlocked*, vol. 18, p. 100282, 2020.

[11] A. Ameri, "A deep learning approach to skin cancer detection in dermoscopy images," *Journal of biomedical physics & engineering*, vol. 10, no. 6, p. 801, 2020.

[12] M. Goyal, T. Knackstedt, S. Yan, and S. Hassanpour, "Artificial intelligence-based image classification methods for diagnosis of skin cancer: Challenges and opportunities," *Computers in biology and medicine*, vol. 127, p. 104065, 2020.

[13] T. J. Brinker, A. Hekler, J. S. Utikal, N. Grabe, D. Schadendorf, J. Klode, C. Berking, T. Steeb, A. H. Enk, and C. Von Kalle, "Skin cancer classification using convolutional neural networks: systematic review," *Journal of medical Internet research*, vol. 20, no. 10, p. e11936, 2018.

[14] S. H. Kassani and P. H. Kassani, "A comparative study of deep learning architectures on melanoma detection," *Tissue and Cell*, vol. 58, pp. 76–83, 2019.

[15] A. Hekler, J. S. Utikal, A. H. Enk, A. Hauschild, M. Weichenthal, R. C. Maron, C. Berking, S. Haferkamp, J. Klode, D. Schadendorf *et al.*, "Superior skin cancer classification by the combination of human and artificial intelligence," *European Journal of Cancer*, vol. 120, pp. 114–121, 2019.

[16] T. Saba, "Recent advancement in cancer detection using machine learning: Systematic survey of decades, comparisons and challenges," *Journal of Infection and Public Health*, vol. 13, no. 9, pp. 1274–1289, 2020.

[17] S. Haggenmüller, R. C. Maron, A. Hekler, J. S. Utikal, C. Barata, R. L. Barnhill, H. Beltraminelli, C. Berking, B. Betz-Stablein, A. Blum *et al.*, "Skin cancer classification via convolutional neural networks: systematic review of studies involving human experts," *European Journal of Cancer*, vol. 156, pp. 202–216, 2021.

[18] A. Adegun and S. Viriri, "Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of state-of-the-art," *Artificial Intelligence Review*, vol. 54, pp. 811–841, 2021.

[19] C. Yu, S. Yang, W. Kim, J. Jung, K.-Y. Chung, S. W. Lee, and B. Oh, "Acral melanoma detection using a convolutional neural network for dermoscopy images," *PloS one*, vol. 13, no. 3, p. e0193321, 2018.

[20] F. Bozkurt, "Skin lesion classification on dermatoscopic images using effective data augmentation and pre-trained deep learning approach," *Multimedia Tools and Applications*, vol. 82, no. 12, pp. 18 985–19 003, 2023.

[21] L. P. Vanka and S. Chakravarty, "Melanoma detection from skin lesions using convolution neural network," in *2022 IEEE India Council International Subsections Conference (INDISCON)*. IEEE, 2022, pp. 1–5.

[22] T. M. Alam, K. Shaukat, W. A. Khan, I. A. Hameed, L. A. Almuqren, M. A. Raza, M. Aslam, and S. Luo, "An efficient deep learning-based skin cancer classifier for an imbalanced dataset," *Diagnostics*, vol. 12, no. 9, p. 2115, 2022.

[23] H. Y. Mir and O. Singh, "Power-line interference and baseline wander elimination in ecg using vmd and ewt," *Computer Methods in Biomechanics and Biomedical Engineering*, pp. 1–20, 2023.

[24] M. ur Rehman, S. H. Khan, S. D. Rizvi, Z. Abbas, and A. Zafar, "Classification of skin lesion by interference of segmentation and convolotion neural network," in *2018 2nd International Conference on Engineering Innovation (ICEI)*. IEEE, 2018, pp. 81–85.

[25] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA transactions on Signal and Information Processing*, vol. 3, p. e2, 2014.

[26] J. A. Hertz, *Introduction to the theory of neural computation*. Crc Press, 2018.

[27] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[32] K. J. Piczak, "Recognizing bird species in audio recordings using deep convolutional neural networks." in *CLEF (working notes)*, 2016, pp. 534–543.

[33] L. Rosasco, E. De Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?" *Neural computation*, vol. 16, no. 5, pp. 1063–1076, 2004.

[34] C. Nebauer, "Evaluation of convolutional neural networks for visual recognition," *IEEE transactions on neural networks*, vol. 9, no. 4, pp. 685–696, 1998.

[35] J. Fieres, J. Schemmel, and K. Meier, "Training convolutional networks of threshold neurons suited for low-power hardware implementation," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. IEEE, 2006, pp. 21–28.

[36] Z. Zhang, "Derivation of backpropagation in convolutional neural network (cnn)," *University of Tennessee, Knoxville, TN*, vol. 22, p. 23, 2016.

[37] I. Arel, D. C. Rose, and T. P. Karnowski, "Deep machine learning-a new frontier in artificial intelligence research [research frontier]," *IEEE computational intelligence magazine*, vol. 5, no. 4, pp. 13–18, 2010.

[38] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, pp. 611–629, 2018.

Snowber Mushtaq Snowber Mushtaq earned her M.Tech. degree in Computer Science and Engineering from the Sri Mata Vaishnu Devi University, Katra, in 2016. Currently, She is pursuing her Ph.D. at the National Institute of Technology, Srinagar, with a research focus on Deep Learning.

Omkar Singh Dr. Omkar Singh holds M.Tech. and Ph.D. degrees from the National Institute of Technology, Jalandhar. He currently serves as an Assistant Professor at the NIT, Srinagar. His research specializes in Digital Signal Processing, Wavelets, Filter Banks, and Adaptive Filters. Biomedical Signal Processing.