Skin Cancer Detection and Classification using Deep Learning Approach

Abhishek Waghchaure   
Dr. Vishwanath Karad MIT World Peace University, Pune, India  
Email:[abhisw28@gmail.com](mailto:abhisw28@gmail.com)

Deepali Pankaj Javale  
Dr. Vishwanath Karad MIT World Peace University, Pune, India  
Email:[deepali.javale@mitwpu.edu.in](mailto:deepali.javale@mitwpu.edu.in)

line 1: 3rd Given Name Surname  
Line 2: *dept. Name of organization   
(of Affiliation)*  
line 3: *name of organization   
(of Affiliation)*line 4: City, Country  
line 5: email address or ORCID

*Abstract*—Skin cancer is one of the most prevalent and potentially fatal forms of cancer worldwide, emphasizing the critical need for accurate and efficient detection methods. In this paper, we propose a novel approach for skin cancer detection utilizing an encoding-decoding algorithm in conjunction with convolutional neural networks (CNNs). The encoding-decoding algorithm facilitates robust feature extraction from skin lesion images, while CNN architectures including DenseNet201, VGG16, and Xception are employed for multi-class classification of skin lesions into seven distinct categories. The methodology is rigorously evaluated on a comprehensive dataset, and experimental results demonstrate the effectiveness of the proposed approach in accurately identifying various types of skin cancer. Furthermore, comparative analysis across different CNN architectures provides valuable insights into their respective strengths and limitations in the context of skin cancer detection. Overall, this research contributes to advancing the field of computer-aided diagnosis for skin cancer, offering promising prospects for early detection and improved patient outcomes.

Keywords—Railway Tracking, PIC Microcontroller, Sensors

# Introduction

Skin cancer is a significant public health concern worldwide, with its incidence rising steadily over the past few decades. According to the World Health Organization (WHO), skin cancer accounts for approximately 1 in every 3 cancers diagnosed globally, making it the most common form of cancer among humans [1]. While various factors contribute to the development of skin cancer, including genetic predisposition and environmental exposure to ultraviolet (UV) radiation, early detection remains paramount for effective treatment and improved patient outcomes [2]. In this context, advancements in artificial intelligence (AI) and machine learning (ML) techniques offer promising avenues for enhancing the accuracy and efficiency of skin cancer detection.

The detection and classification of skin lesions present unique challenges due to the diverse range of lesion types and their subtle visual characteristics. Traditional methods of skin cancer diagnosis primarily rely on visual inspection by dermatologists, which can be subjective and prone to errors, particularly in cases where lesions exhibit ambiguous features [3]. Computer-aided diagnosis (CAD) systems leveraging ML algorithms have emerged as valuable tools to assist dermatologists in the early detection and classification of skin cancer [4]. These systems analyze digital images of skin lesions to extract relevant features and classify them into benign or malignant categories, thereby aiding clinicians in making informed diagnostic decisions.

In recent years, deep learning algorithms, particularly convolutional neural networks (CNNs), have demonstrated remarkable success in various image recognition tasks, including medical image analysis [5]. CNNs are ideally suited for skin cancer detection due to their ability to automatically learn discriminative features from raw image data, without the need for handcrafted feature extraction [6]. Moreover, the availability of large-scale annotated datasets, such as the International Skin Imaging Collaboration (ISIC) dataset [7], has facilitated the development and evaluation of CNN-based CAD systems for skin cancer diagnosis.

Despite the progress made in the field of automated skin cancer detection, several challenges persist, including the classification of skin lesions into multiple subtypes and the interpretation of model predictions in clinical settings [8]. To address these challenges, researchers have explored novel approaches combining feature extraction techniques with CNN architectures for improved classification performance. In this paper, we propose a comprehensive methodology for skin cancer detection that integrates an encoding-decoding algorithm with state-of-the-art CNN architectures, namely DenseNet201, VGG16, and Xception.

The encoding-decoding algorithm serves as a feature extraction mechanism, capturing hierarchical representations of skin lesion images that are subsequently fed into the CNNs for classification. This hybrid approach leverages the complementary strengths of both feature-based and data-driven methods, thereby enhancing the discriminative power of the CAD system. Additionally, we extend the classification task beyond the binary distinction of benign versus malignant lesions to encompass seven distinct classes of skin cancer, enabling more granular diagnostic capabilities.

# Literature Reviews

Dorj et al. used an online dataset consisting of 3753 photos from four classes. [1]. Employing AlexNet for feature extraction and an ECOC SVM for classification, they achieved an amazing 94.2 percent accuracy. It is worth noting that when the online dataset was gathered, the usual benchmark rules were not followed. Rezvantalab et al. [2] employed an alternative methodology using the 120 photos from eight classes in the HAM10000 dataset. The authors used a variety of pre-trained models to report their results, including DenseNet 201, ResNet 152, InceptionV3, and InceptionResNetV2. The most accurate model, DenseNet 201, achieved an accuracy of 86.59 percent. For presenting the results, AUC values were calculated and reported for each model and each class. Each class also had multiple test specimens associated with it. The AUC values for the three classes in the PH2 dataset ranged from 93.80 percent to 99.30 percent. Hosny et al. [3] reached a remarkable 98.61 percent accuracy while using a modified version of AlexNet. They applied this tailored model to boost the original dataset and succeeded in acquiring a total of 4400 images. An AUC of 81.40 percent was obtained by Dascalu and David's [4] study of the ISIC 2017 dataset, which divided 5161 images into two classes. Their unique approach involves sonification and K-means clustering to assess how image quality affects diagnostic accuracy.

In a different investigation, Pham et al. [5] employed the ISIC 2016 dataset (172 photos) and the HAM10000 dataset (1113 images), both of which were from a single class. Their method yielded an accuracy of 74.75 percent using LBP balanced random forest, HSV, and linear normalization. Comparing the color, texture, and form of melanoma skin cancer cells was the primary goal of this investigation. In order to attain an accuracy of 82.95 percent, Hekler and co-authors [6] combined physician judgments with Convolutional Neural Network (CNN) predictions utilizing the HAM10000 and ISIC datasets (a total of 11,444 pictures) with five classes. Interestingly, this study obtained results for binary and multiclass classifications using the XGBoost algorithm.

The HAM10000 dataset has been studied by several groups, with the best accuracy results coming from Emara et al. who used a modified InceptionV4 model. Their approach yielded performance of approximately 94.7% accuracy. The InceptionV4 modifications were mainly oriented to handle the unbalanced class ratios found in that dataset. The study by Chaturvedi et al. was not quite as successful with a result of 83.1% on the same dataset. It used a pretrained MobileNet architecture (and was able to use some Transfer Learning techniques because the original melanoma dataset was quite large, containing 38,569 photographs).

Mohapatra et al.'s study made advantage of the seven different classes found in the HAM10000 dataset. [9]. Using an unaltered, pre-trained MobileNet model, they achieved an accuracy of 80%. In contrast, Chen et al.'s [10] N/A dataset included nine distinct kinds of skin lesions. Chen et al. utilizing a pre-trained ResNet50 model on the N/A dataset, achieved an accuracy of 83.74%. Additionally, they showed how to effectively classify nine distinct types of skin lesions using the ResNet50 model.

The National Cancer Center, Tokyo, provided Jinnai et al. [11] with a dataset of 5,846 photographs divided into six categories. The accuracy of their FRCNN, BCD, and TRN techniques was 86.2 percent, 79.5 percent, and 75.1 percent, in that order. In order to compare classifiers, they also used a bespoke dataset that consisted of the two main groups, benign and malignant. Using ResNetXt101, Chaturvedi et al. [12] achieved an astonishing accuracy of 92.83 percent by analyzing the seven-class HAM10000 dataset. An in-depth analysis by Chaturvedi and colleagues revealed the best hyperparameter settings for identification of histopathology images, and their results showed that the ResNetXt101 model was the top-performing model for this task.

Skin cancer is among the deadliest cancers. It results from damage to skin cells' DNA that is not repaired. The damage brings on skin alterations. Several causes exist, but prolonged, unprotected sun exposure is the most frequent and harmful. In addition, exposure to dangerous substances or unhealing wounds are risk factors for skin cancer. However, all age groups are seeing an increase in occurrences. The age group between 15 and 29 is the fastest-growing in terms of skin cancer cases. Because these problems are so serious, many researchers have worked on developing a number of different techniques for the early detection of skin cancer. In order to identify skin cancer, these methods examine how lesions appear how two elements compare (symmetry, for example) or how color, size, or form contrast with the norm. Some experts would even go so far as to predict with certainty who would prevail in a boxing fight between benign and melanoma. However, these attempts rarely provide coherent programming for early skin cancer detection—most programs rely on human visual specialists to evaluate a histological section's septal (sandwich) imaging [13].

Skin cancer is one of the most widespread and deadly kinds of cancer found around the globe. Reducing the death rate from skin cancer requires early diagnosis of the disease. The conventional approaches to diagnosing skin cancer are costly, time-consuming, and prone to spreading the illness. They are also uncomfortable. Dermoscopy can be used to diagnose skin cancer noninvasively, although it still has some drawbacks. Artificial intelligence (AI) has advanced dramatically in the last several years and is crucial to diagnosing many diseases. Automated detection systems based on artificial intelligence (AI) hold the promise of enhancing the accuracy of skin cancer diagnoses in biomedical engineering. They do this by attempting to overcome some of the more egregious faults of traditional diagnosis methods. In this study, an automated skin cancer early detection system is created and described. It interacts with dermoscopic images of skin lesions through artificial intelligence. The system's segmentation phase employs snake and region-expanding algorithms adapted to current conditions. The outcomes demonstrate that adaptive snake outperforms region expanding in accuracy and efficiency. Support vector machines and artificial neural networks are the main methods used in the categorization phase, which comes to the conclusion that ANLs are better than SVMs in this instance. The technique utilizing artificial neural networks (ANNs) attains 94% accuracy, 96% precision, 95.83% specificity, 92.30% sensitivity (sometimes called recall), and an F1 score of 0.94. The device is user-friendly, takes time, and is efficient enough to quickly provide patients with the "skin cancer or not" decision. [14]

Undoubtedly among the deadliest forms of cancer, skin cancer holds the dubious distinction of being one of the primary causes of mortality globally. Skin cancer has a far better prognosis when detected early. Most current methods rely on human inspectors who have received training and work in well-lit environments. Nevertheless, these approaches have the potential to be lethal when they fail. The current environment permits and requires a deep learning-assisted visual inspection technique to diagnose skin cancer. The state of the art for these kinds of techniques is surveyed in this study. [15]

To lower the number of parameters, we replaced the excitation and squeezing components of the model with the realistically advantageous channel attention component. In order to efficiently utilize synthetic features, we suggested employing cross-layer connections between Mobile modules. We applied dilated convolutions to improve the receptive field. We also focused on optimizing the model's performance by fine-tuning the hyperparameters, a crucial component of any optimization work. For the pre-trained MobileNet-V3, we employ advanced optimization techniques like Bayesian optimization to determine the ideal hyperparameters. We evaluated our improved MobileNet-V3 against the following melanoma detection and segmentation techniques: ResNet-152v2, VGG-19, MobileNet, VGG-16, and MobileNet-V2 (training and testing on the HAM-10000 dataset). The metrics used to report how successfully these techniques located and correctly diagnosed the melanomas (compared to the findings from human pathologists) are precision, sensitivity, accuracy, and specificity. Our research shows that the MobileNet-V3 model operates with 97.84% precision, 96.35% sensitivity, 98.86% accuracy, and 97.32% specificity when optimized hyperparameters. Not only did this research yield results, but it also paid off. For the patients who stood to gain the most, the returns came in the shape of even better medical care—possibly lifesaving and affordable. [16]

An independent, threshold-based approach is recommended for segmenting, classifying, and detecting skin cancers. The proposed method within this approach uses a meta-heuristic optimizer called the sparrow search algorithm (SpaSA). For the segmentation phase of the process, five different configurations of the U-Net model (U-Net, U-Net++, Attention U-Net, V-net, and Swin U-Net) are used. The pre-trained models that the authors use in this study include VGG16, VGG19, MobileNet, MobileNetV2, MobileNetV3Large, MobileNetV3Small, NASNetMobile, and NASNetLarge. The authors used the meta-heuristic SpaSA to optimize the hyperparameters of these eight CNN models. Five public sources provided the dataset. Two datasets were created from the segmented photos: two-classes and ten-class. The best results reported to date for the "skin cancer segmentation and classification" dataset were obtained with U-Net++, which has DenseNet201 as its backbone architecture. It employed a variant of the cosine loss function, producing a loss of 0.104 on the test set and achieving impressive scores across several other metrics: 94.16% on accuracy, 91.39% on the F1-score, 99.03% on AUC, and 96.08% and 96.41% on IoU for the two classes defined in the dataset. More surprisingly, the authors also reported that U-Net++ was able to achieve 77.19% and 75.47% on two different instances of a weakly-supervised training test set. The Attention U-Net with DenseNet201 performed the best on the "PH2" dataset, reporting a loss of 0.137 along with precision, accuracy, AUC, and other numbers that peaked at 92.74% with a precision of 94.75% and numbers that dropped as low as 68.04% with "squared hinge" and "hinge" loss configurations at the end of its scoring list. The overall accuracy of our convolutional neural network (CNN) experiments achieved a high of 98.27% when we applied them to the "ISIC 2019 and 2020 Melanoma" dataset. A MobileNet pre-trained model provided the basis for our best model. The pre-trained MobileNet model, operating on a different dataset, achieved a second-place accuracy of 98.83% among our skin cancer classification models. Our lowest accuracy (85.87%) came from using a MobileNetV2 pre-trained model on a different skin cancer dataset. Our accuracy rates for each dataset were competitive when we compared our approach's results with those of 13 similar studies. [17]

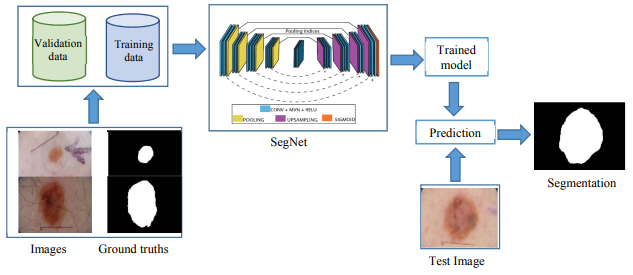
Humans experience a high incidence of skin cancer, with the most common types being nonmelanoma cancers such as SCC and BCC; the number of these cancer types is rising. Skin cancer is not homogeneous, however, and the most dangerous and deadly skin cancer is melanoma. Melanomas can arise in normal skin or in moles. They can appear and change fast in their looks. Cancers of the skin, particularly melanoma, call for accurate and rapid detection. Otherwise, the appearance of melanoma as with cancer in general leads to excess morbidity and death that is unnecessary and avoidable. Convolutional neural networks, or CNNs, are receiving more attention as practical methods for automating procedures that may classify lesions based on their malignant status and visual identification. This work has created a new strategy for early skin cancer detection. Processing dermoscopic pictures forms its foundation. The architecture of the model is based on the VGG-16 network, which is a widely recognized convolutional neural network (CNN) framework. However, instead of using the usual configuration of the VGG-16 network, we chose to use an improved version of the network as the main architecture in our model. As we will explain, most of the improvements take the shape of adjustments to the model's image data processing pipeline. Naturally, we want to increase skin cancer detection accuracy to the point where it is deemed operationally viable in a real-world setting. According to the results, the model we suggested is more accurate than the other tested approaches [18].

Consequently, interest in machine learning has surged recently.Machine learning is mostly associated with deep neural networks (DNNs). DNNs are the greatest ML technology for solving practical problems like speech recognition, computer vision, or even health-related issues. DNNs are creatures of determinism. They function. However, without some confidence metric, it is impossible to know how confident they are in their work truly. Either guesstimate it using a prior distribution (which is Bayesian) or use your DNN to perform multiple "forward passes" and use the output to produce a confidence level. This research presents a novel use of the MCD method that enables the construction a deep neural network (DNN) that incorporates a knowledge of uncertainty. We demonstrate that this novel kind of DNN, an uncertainty-aware DNN, can forecast the problem's output and its "calibration," or the degree to which the result is certain or uncertain. Crucially, we set up our DNN to identify when it is guessing by assigning a high predictive entropy to every scenario in which it has made a mistaken prediction [19].

In this paper, the author present the technique we developed for enhancing skin cancer classification and for effecting a more accurate segmentation of skin lesions. We used a dynamic graph cut algorithm to accomplish this.The suggested methodology addresses the common over- and under-segmentation observed in cut algorithms by accurately segmenting skin lesions, even tiny ones. We further demonstrate the usefulness of data augmentation. In a recent skin cancer contest, our training achieved an excellent performance measure of 97.986% across six classes, mostly because our model significantly reduced false positives compared to the next best competitor. Ultimately, the outcomes of numerous tests employing two distinct transferring models show that our model's success is mostly attributable to the errors it avoided, not the fact that it uses new training photos. [20]

# Skin cancer Detection Using Deep Leartning Algorithm

Fig. 1 depicts the system's block diagram of skin cancer detection using SegNet.



1. Block diagram of the Skin lesion segmentation using SegNet

## Skin lesion Image Database

The 200 dermoscopic images in the IPH2 dermoscopic collection and their label masks. Each image is 572 x 765 pixels in fixed dimensions and is an RGB image. [14] The dataset is available to the general public for research and experimentation. Each image was first reduced in size to 192 × 256 for training purposes before being sent into the network. It lessens the network's training parameters, duration, and complexity without appreciably changing the outcomes.

## SegNet

A convolutional neural network (CNN) architecture called SegNet was created specifically for pixel-by-pixel image segmentation tasks. In this case, it is used for segmenting features in medical images, such as cell nuclei in dermatological images from the PH2 dataset.

SegNet is a deep semantic pixel-wise segmentation architecture. It is proposed by computer vision and robotics group members of Cambridge University. SegNet comprises a pixel-wise classification layer, a matching decoder network, and an encoder network. Batch normalization, a ReLU, and one or more non-linear convolutional layers precede the non-overlapping max pooling and subsampling layers in each encoder network. Decoders are generally similar to encoders; the main difference is that they are linear. The decoders utilize the max-pooling to carry out non-linear upsampling. These conserve high-frequency values and reduce the number of trainable parameters in segmented images. The result from the last decoder is given to the softmax layer, and the final output is obtained [9].

Here is an overview of the key components and steps:

* The SegNet architecture is implemented using Keras. It consists of an encoding and a decoding stage.
* The encoding stage includes convolutional layers with batch normalization and activation functions, followed by max-pooling layers for down-sampling.
* The decoding stage involves up-sampling followed by transposed convolutional layers with batch normalization and activation functions.
* Skip connections combine low-level and high-level features, aiding in the precise localization of objects.
* The final layer outputs a binary segmentation mask.

Here, the binary cross-entropy is employed as the loss function. The cross-entropy function calculates the deviation of each class's prediction from the true value. The ultimate loss is calculated by averaging the classwise errors. This problem has just two classes: black or white (0 or 1), depending on the mask. Thus, instead of using the categorical cross-entropy that was first suggested, binary cross-entropy is utilized as the loss function in this instance. The binary cross-entropy is in the below form:

(1)

The network's SGD (Stochastic Gradient Descent) is the optimizer. One of the numerous values commonly used for the learning rate parameter, the learning rate, is set to 0.001, making it a crucial hyperparameter in the optimization.

An updated rule inspired by physical optimization is also provided by momentum. The benefits of utilizing momentum with SGD include significantly accelerating the learning process from tiny changes. Similarly, all parameters' velocities are saved and utilized in the update process. For optimization, a momentum value of 0.9 is utilized.

## Performance Evaluation

Metrics such as Intersection over Union (IoU), Dice Coefficient (DI), Precision, Recall (Sensitivity), and Accuracy were used to assess the performance of the proposed system.

* Intersection over Union (IoU): This metric, which gauges the overlap between expected and ground truth segmentation, is also called the Jaccard Index. The computation involves dividing the union of the anticipated and ground truth regions by the intersection of these regions.

(2)

IoU values range from 0 to 1, with higher values indicating better overlap.

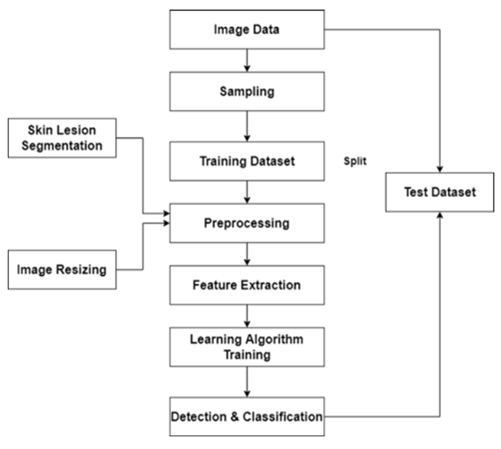
* Dice Coefficient (DI): The similarity between the predicted and ground truth segmentations is also measured by the Dice Coefficient. It is derived by dividing the total of the areas of the anticipated and ground truth regions' intersections twice.

(3)

Similar to IoU, the Dice Coefficient has a range of values from 0 to 1, where larger values correspond to more accurate segmentation.

# Skin cancer Recogntion using Deep Learning Algorithm

This approach was created to categorize various forms of skin cancer. Fig. 2 displays the proposed system's block diagram.



1. Block diagram of Skin cancer recognition

## Dataset Preparation

The following source provides the dataset that the suggested system is built upon. Segmentation and classification of ham1000 available at https://www.kaggle.com/datasets/surajghuwalewala

The comprehensive HAM10000 dataset makes a substantial contribution to the field of computer-aided skin cancer diagnosis research. It includes 10,000 excellent pictures of skin lesions. A dermatoscope was used to capture each image, which had a resolution of 3000 × 2000 pixels. The photographs are similar to histological sections because the dermatoscopes used to take them were calibrated to 100x, the same magnification as a light microscope. Thus, studies on deep learning for skin cancer can be trusted with diagnostic data. The skin lesions within the dataset are classified into multiple classifications, including basal cell carcinoma (BCC), nevus, and melanoma. Images of benign and malignant lesions are included in the examples, providing the dataset with the depth and breadth that are essential for accurately training AI.

## Dataset preprocessing

A popular method for image processing that reduces noise in skin lesion photos and improves the sharpness of the features is the median filter. In order to use it, first choose a pixel to work on, and then calculate the neighborhood's median pixel values. That neighborhood is selected based on its size and specific shape, which is typically square or circular. The pixel value is changed to the median value once it has been determined. The neighborhood's size and shape are the only factors that can be changed for the median filter. Nonetheless, it works well on photos with edges and for eliminating impulsive noise.

Noise reduction using a median filter in the context of skin lesion photos can help to increase the precision of later image analysis activities, such feature extraction or machine learning-based categorization. During the image capture process, noise of various kinds, such as speckle or random fluctuations in pixel intensity, can appear in skin lesion images.

Although median filtering is effective in some circumstances, the features of the picture noise and the objectives of the image processing must be taken into consideration when selecting a filtering method. Furthermore, any image processing method should be applied carefully since it could affect how medical images are interpreted.

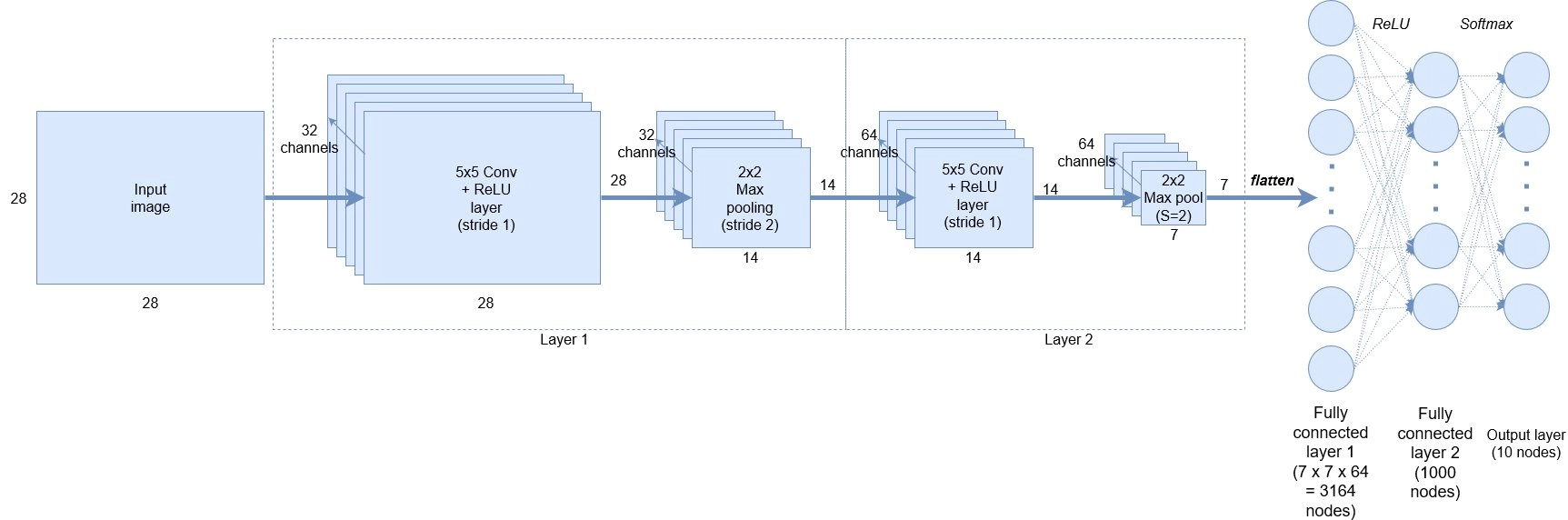
## Dataset Splitting

The splitting of the dataset is a critical step in the creation of a machine learning model. The dataset is split into two subsets in this method: training (80%) and validation (20%). The training set is a collection of examples with known labels that are used to build a model. While being trained, the model discovers the patterns and connections among the labels in the training set. By evaluating the model on untested data, the validation set helps with hyperparameter adjustment and model evaluation during training, avoiding overfitting and guaranteeing optimal performance. Lastly, the testing set serves as an objective gauge of the model's performance in real-world situations by evaluating the model's generalization abilities on entirely new data. This thorough dataset separation technique makes it easier to build reliable machine learning models that can make precise predictions and perform well in generalization.

## Classification of the skin lesion using deep learning algorithm

CNN, Densenet201, Xception, and hybrid CNN-SVM algorithms were employed in this method. This section presents the detailed architecture of all the algorithms.

### CNN: Convolutional neural networks (CNNs) are specialized in image recognition and classification. They are multi-layered feed-forward neural networks composed of filters and filter banks that extract features from images. By adjusting filter weights, CNNs can detect specific features like edges or curves. A typical CNN architecture alternates between convolutional and pooling layers, followed by one or more fully connected layers. The architecture of CNN is depicted in Fig.3.



1. Architecture Diagram of CNN algorithm

#### Convolutional Layer: The convolutional layer in a CNN is essential for feature extraction from images, identifying key features like edges and textures. It reduces the number of parameters by using a sparse connection method, where filters with fixed weights move over the image to detect specific features. This process, known as "feature mapping," uses multiple filters (channels) to produce distinct outputs. Each filter learns different features, enhancing the network's ability to understand the image. For grayscale images, the convolution layer's output is three-dimensional, representing the number of channels and their individual outputs.

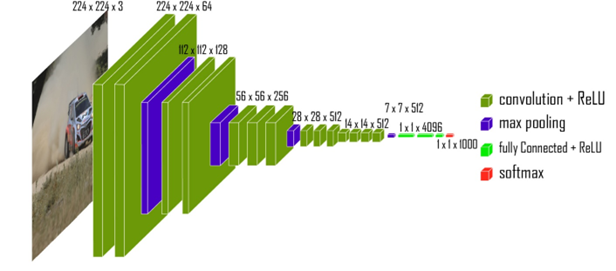
#### ReLU Layer: ReLU (Rectified Linear Unit) is a non-linear activation function that operates element-wise on a feature map. It transforms values less than or equal to zero to zero, effectively highlighting only the positive values. This can be mathematically expressed as F(x) = max(0, x) ), where values below zero are replaced with zero, emphasizing the positive aspects of the feature map.

#### Pooling Layer: The dimensionality of each activation map is decreased. Still, the most critical data are kept in the pooling layer. The images given serve to produce a number of non-overlapping rectangles. And now, what is pooling? Pooling is a sliding window approach, like many others, but instead of using tunable weights, it applies some statistical function to the contents of its window. Max pooling is the most commonly used form of pooling; it uses the max() function on the contents of its window. Some other variations, including mean pooling (which takes the statistical mean of the contents), are also used sometimes. We’ll be focusing on max pooling in this chapter. The following diagram illustrates the max pooling process.

#### Flattening Layer:Convolutional neural networks (CNNs) effectively create object representations from high-resolution data. However, to achieve a final classification, a traditional classifier needs to be added to interpret the network's rich output. This involves flattening the CNN output into a one-dimensional vector before applying the classifier. The pooled feature map, derived from pooling operations, is essential for understanding and processing the CNN output, providing more than just a human-readable summary.

#### Fully Connected Layer: The fully connected layer (FCL) is used to classify images based on features extracted from the CNN. A Softmax activation function classifier processes the signals from the FCL, producing a list of probabilities for each class label. These probabilities indicate the network's confidence in each class, essentially rating the class labels. By adding the FCL on top of a CNN, images can be effectively classified into various categories.

### Vgg16: Each year, ImageNet organizes the Large Scale Visual Recognition Challenge (ILSVRC), a major computer vision competition with numerous participating teams. Competitors first tackle the challenge of localizing objects in images, then progress to the more complex task of categorizing elements within those images. One notable entry came from researchers at the University of Oxford in the UK, which generated significant excitement in the field. The architecture of Vgg16 model is presented in Fig.4.



1. VGG-16 model architecture

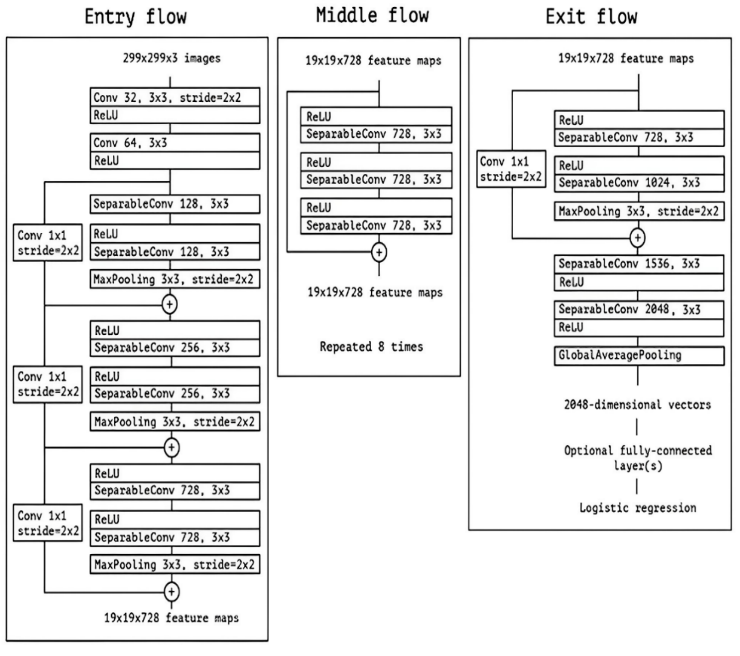
The ImageNet dataset, featuring 14 million images across 1,000 classes, allows the model to achieve a remarkable top-5 test accuracy of 92.7%. To reach this level of accuracy, the model processes input images with dimensions of 224 × 224 × 3. The network's first two layers each have 64 channels and use a 3 × 3 filter size, with "SAME" padding in TensorFlow. This padding technique helps maintain the image's spatial dimensions throughout the layers.

The model uses 256 filters of size 3 × 3 in each of the next two convolutional layers. It then includes two sets of three convolutional layers, each followed by a max-pooling layer. Each set consists of 512 filters of size 3 × 3, with padding values of (3, 3). Finally, after the last max-pooling layer, the image is processed by a stack of two convolutional layers, employing 3 × 3 filters instead of the larger 7 × 7, ZF-11 × 11, or AlexNet filters.

After adding a convolutional and max-pooling layer to the stack, a (7, 7, 512) feature map is obtained, which is then flattened into a feature vector of 25,088 values. This vector is processed through three fully connected layers: the first produces a vector of length (1, 4096), the second also generates a vector of length (1, 4096) (though dropout between these layers is not used here), and the third creates a vector of length (1, 1000) representing the 1,000 classes from the ILSVRC challenge. The output of the third layer is sent to a softmax layer to normalize the classification vector. All hidden layer’s use ReLU as their activation function.

The goal of the research project ImageNet is to build a large database of photographs with annotations, such labels. A variety of computer vision tasks have previously shown the efficacy of models like InceptionV1, InceptionV2, VGG-16, and VGG-19, which have been pretrained using ImageNet. They were created from the ground up and trained on an enormous dataset (over 14 million photos) with an enormous number of categories (about 20,000). The models are vast and deep due to the volume of picture data, which makes them highly efficient in extracting features from images. The image annotation project's pretrained models can be used to "fine-tune" computer vision tasks that have been assigned to various images (of various categories).

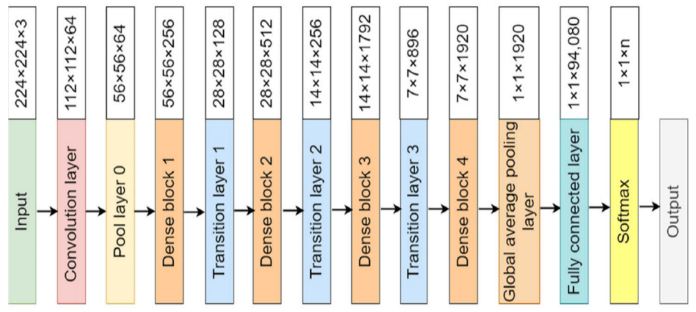
### Xception: The Xception architecture, introduced by François Chollet in 2016, is an advanced version of the Inception network designed to address limitations of traditional CNNs and enhance their capabilities. The name "Xception" stands for "Extreme Inception," reflecting its deepened version of the Inception module. Xception replaces standard convolutional layers with depthwise separable convolutions. This approach divides convolution into two distinct phases: depthwise convolution, which applies a single filter per input channel, and pointwise convolution, which performs a 1x1 convolution to combine features across channels. The architecture of Xception algorithm is shown in Fig. 5.



1. Architecture of Xception model

The Xception module, as depicted in Figure 5.12, consists of three main components: the entering flow, the middle flow, and the exit flow. The entering flow starts with two blocks of depthwise separable convolutions, followed by ReLU activations and residual connections. Max pooling layers and various types of separable convolutions are also included, with specific strides documented for each layer. Instead of concatenating tensors, skip connections use 'ADD' operations. The entering flow processes an image from 299 × 299 × 3 to 19 × 19 × 728. The diagram also details image dimensions, layer configurations, filter numbers and shapes, pooling operations, and the optional fully connected layer in the middle and exit flows.

### DenseNet201: Transfer learning is highly effective for classification tasks with small datasets, and Deep Transfer Learning (DTL) can further enhance results. This paper introduces a DTL model based on DenseNet201, which uses a convolutional neural network architecture with ImageNet weights to extract features. DenseNet connects each layer to all preceding layers in a feed-forward manner, addressing the vanishing gradient problem by ensuring that each layer receives input from all prior layers. Although this creates a larger input and output space for each layer, it avoids making the model impractically large by concatenating outputs from all previous layers at each step. The architecture of Densenet algorithm is shown in Fig. 6.



1. Architecture of Densenet201

DenseNet201 is a convolutional neural network within the DenseNet model family, known for its dense connection topology. In DenseNet, each layer receives inputs directly from all preceding layers, which mitigates the vanishing gradient problem, promotes feature reuse, and facilitates feature propagation. This design enhances the network’s efficiency and performance. For a comprehensive overview of the DenseNet201 architecture, detailed information can be reviewed.

* DenseNet201: This version of the DenseNet architecture features 201 layers, encompassing activation, batch normalization, convolutional, and other layers. The "201" signifies the network's depth, making it suitable for complex tasks and broader datasets.
* Pre-trained Weights: DenseNet201 is commonly pre-trained on large datasets such as ImageNet. This pre-training enables the model to learn hierarchical features that can be fine-tuned for specific tasks, enhancing its performance in various applications.
* Architecture: DenseNet201 employs a deep convolutional network with dense connection patterns, bottleneck layers, and transition blocks. This design allows it to capture intricate patterns in data, making it effective for image classification and other computer vision tasks. The use of pre-training and transfer learning further improves its capabilities across different applications.

## Performance Evaluation

A number of performance criteria are used to evaluate the proposed system, including accuracy, recall, F1-score, and precision. These are common metrics used to evaluate classifiers. What's good about them is that they offer contrasting perspectives on how well a classifier works. Additionally, they are simple to calculate from the confusion matrix, which is another useful feature. Let's examine each metric in turn:

* Accuracy: By computing the ratio of correct predictions to total predictions, accuracy assesses the overall correctness of the classifier's predictions. It has the following definition:

(4)

* where the numbers represent the number of true positive predictions (TP), true negative predictions (TN), false positive predictions (FP), and false negative predictions (FN). Although accuracy gives a broad picture of the classifier's performance, imbalanced datasets might not be a good fit for it.
* Precision: The percentage of accurately anticipated positive cases among all positively predicted instances is the subject of precision. It is computed as follows:

(5)

A classifier's precision tells us how well it avoids producing false positive results. A reduced rate of misclassifying negative cases as positive is shown by a better precision.

* Recall, also known as True Positive Rate or Sensitivity, quantifies the percentage of accurately anticipated positive cases among all positive cases that actually occur. It is computed as follows:

(6)

Recall highlights the classifier's ability to identify positive instances correctly, and it is particularly useful when the goal is to minimize false negatives.

* F1 score: The F1 score is a metric that balances recall and precision by combining both into one. It is computed as follows and is the harmonic mean of recall and precision:

(7)

Accuracy and recall are balanced by taking into account both erroneous positives and false negatives in the F1 score. When there is an imbalance in the distribution of classes or when recall and precision are equally crucial, it is helpful.

When it comes to binary classification tasks, these metrics are particularly important because there are two classes: positive and negative. By computing them independently for each class and then averaging them (e.g., micro-averaging, macro-averaging), they can also be applied to multi-class classification issues.

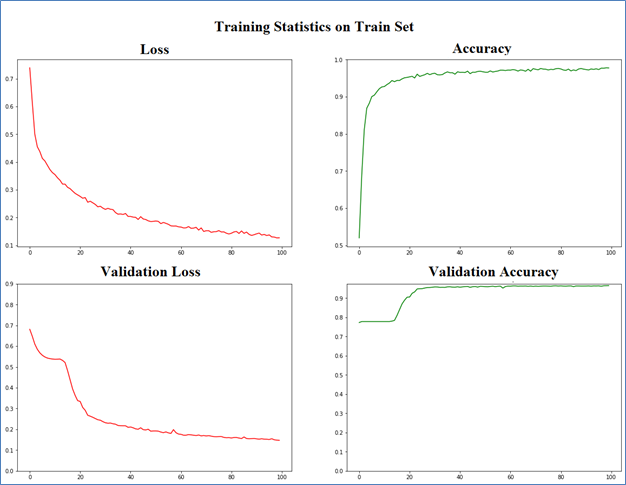
Prioritizing a set of metrics requires careful consideration of the unique demands and features of your classification task. For instance, memory may be more significant in medical diagnosis to reduce false negatives, whereas precision may be more critical in spam email classification to prevent false positives.

# Result

This section presents the results of skin cancer detection and recognition.

## Results of Skin canceer segmentation

The training loss and accuracy obtained for the SegNet for 100 epochs are presented in Fig. 7.



1. Training progress graph of SegNet for skin cancer segmentation

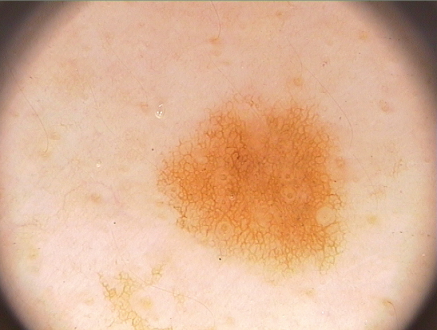
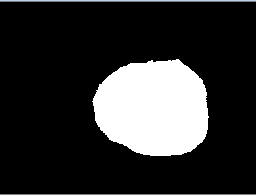
From Fig.7, it is observed that the training loss of the algorithm decreases as the epoch increases while the accuracy of detection increases. The proposed system achieved good detection accuracy with lower loss. The performance of the SegNet algorithm on different testing images is tabulated in Table I.

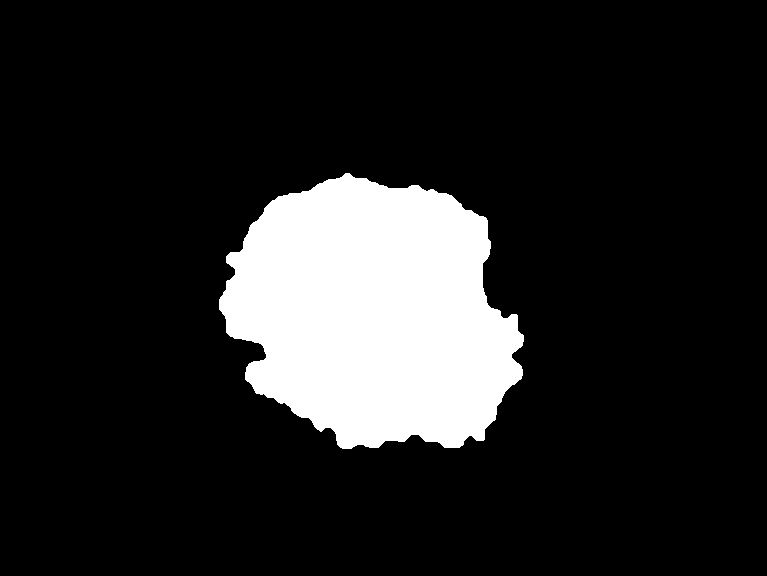
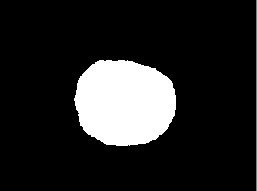
1. Performance of the Segnet for skin cancer segmentation

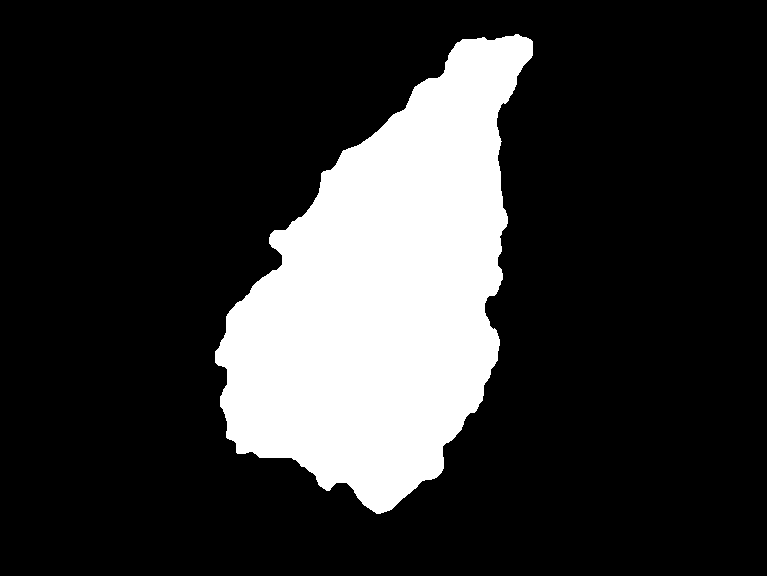
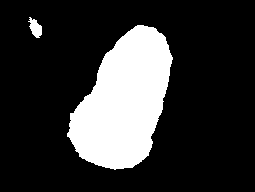
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Image Name** | **IOU** | **DI** | **Precision** | **Recall** | **Accuracy** |
| IMD390 | 88.6487 | 93.6959 | 94.1563 | 93.3030 | 97.5891 |
| IMD392 | 91.3809 | 93.10577 | 91.7550 | 94.4533 | 97.9594 |
| IMD393 | 82.5091 | 91.7462 | 96.8662 | 87.0751 | 91.6137 |
| IMD394 | 81.4715 | 88.3025 | 99.0446 | 79.4143 | 95.4935 |
| IMD395 | 84.5796 | 93.0471 | 88.3356 | 98.1681 | 95.2107 |
| IMD396 | 78.6634 | 89.8264 | 88.3665 | 91.2851 | 95.7885 |
| IMD397 | 77.4382 | 90.6989 | 95.4223 | 86.3694 | 93.7947 |
| IMD398 | 73.4582 | 85.7910 | 97.364 | 76.6427 | 80.8492 |
| IMD399 | 83.2597 | 90.0542 | 83.1518 | 98.0891 | 95.8943 |
| IMD400 | 70.5491 | 88.2608 | 99.5692 | 79.2122 | 90.3401 |
| IMD402 | 86.0697 | 93.4688 | 96.6370 | 90.4492 | 96.8892 |
| IMD403 | 86.0531 | 93.5515 | 98.5809 | 88.9933 | 91.0624 |
| IMD404 | 72.8725 | 88.1839 | 80.7587 | 97.0557 | 87.9130 |
| IMD405 | 81.5398 | 88.6892 | 83.671 | 94.1985 | 96.0306 |
| IMD406 | 79.9282 | 91.7546 | 94.6498 | 88.9359 | 92.2932 |
| IMD407 | 79.8268 | 90.9848 | 96.39 | 86.0291 | 93.4773 |
| IMD408 | 92.1096 | 95.8868 | 97.3147 | 94.4909 | 92.8751 |
| IMD409 | 85.8480 | 92.1197 | 96.4400 | 88.1463 | 90.1570 |
| IMD410 | 74.6085 | 86.7594 | 97.7357 | 77.9552 | 86.9954 |
| IMD411 | 96.4960 | 98.3061 | 99.3681 | 97.2624 | 96.7285 |
| IMD413 | 80.7056 | 89.5501 | 100.0 | 81.0572 | 82.0800 |
| IMD417 | 89.1924 | 94.4246 | 97.3593 | 91.6499 | 89.8010 |
| IMD418 | 86.7094 | 93.0006 | 92.5390 | 93.4416 | 92.0043 |
| IMD419 | 91.7258 | 96.3433 | 93.1384 | 99.7670 | 93.9066 |
| IMD420 | 79.8968 | 88.3287 | 88.1336 | 88.4920 | 82.8938 |
| IMD421 | 85.4635 | 92.8264 | 91.8733 | 93.7829 | 87.0442 |
| IMD423 | 74.4015 | 89.6570 | 82.1417 | 98.6506 | 85.0484 |
| IMD424 | 76.5431 | 86.9446 | 100.0 | 76.8805 | 77.2705 |
| IMD425 | 50.2752 | 67.0301 | 100.0 | 50.3504 | 57.9203 |
| IMD426 | 50.7592 | 70.3199 | 89.8994 | 57.6574 | 71.9563 |
| IMD427 | 89.4674 | 95.4487 | 92.1210 | 98.9965 | 96.8892 |
| IMD429 | 86.8115 | 91.9125 | 87.0902 | 97.2628 | 96.8404 |
| IMD430 | 90.1406 | 94.4177 | 91.0153 | 98.2227 | 97.6155 |
| IMD431 | 85.4778 | 94.7782 | 96.5293 | 93.0510 | 95.3369 |
| IMD432 | 84.3751 | 91.3896 | 91.7192 | 91.0015 | 96.3277 |
| IMD433 | 86.1530 | 84.3162 | 76.5241 | 93.6214 | 95.7946 |
| IMD434 | 84.3383 | 89.3746 | 86.8090 | 91.9834 | 95.6685 |
| IMD435 | 82.9337 | 91.1022 | 89.9630 | 92.2472 | 85.9456 |
| IMD436 | 88.5409 | 94.9120 | 95.7019 | 94.0830 | 94.3074 |
| IMD437 | 83.0516 | 94.5458 | 92.9558 | 96.1681 | 94.5271 |

Table I presents the performance metrics of a SegNet model for skin cancer segmentation over multiple testing images. The metrics include Intersection over Union (IoU), Dice Coefficient (DI), Precision, Recall, and Accuracy. From Table I, it is observed that the detection accuracy is obtained promisingly.

The qualitative analysis of the proposed system is presented in Fig. 8.

(a) (b) (c)

1. Qualitative analysis of the proposed system (a) Input image, (b) Groundtruth, and (c) Output of the proposed system

The proposed system is developed using Python. The qualitative analysis of the developed system focuses on assessing the correctness of skin lesion segmentation, and the results indicate a notable achievement in minimizing false positives. Qualitative analysis involves a thorough visual examination of the segmentation outcomes, comparing them to ground truth or reference images to ensure accuracy. The emphasis on lower false positives is crucial in medical image analysis, as it signifies a reduced likelihood of incorrectly identifying non-lesion areas as lesions.

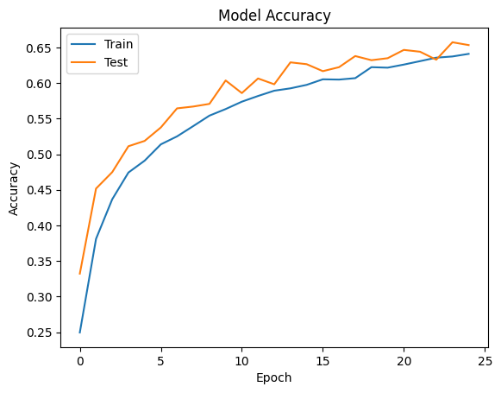
## Skin cancer Recogntion

The proposed system is developed to classify skin cancer into different type. The results of different deep learning algorithms for classification of skin lesion into seven different categories are presented in this section.

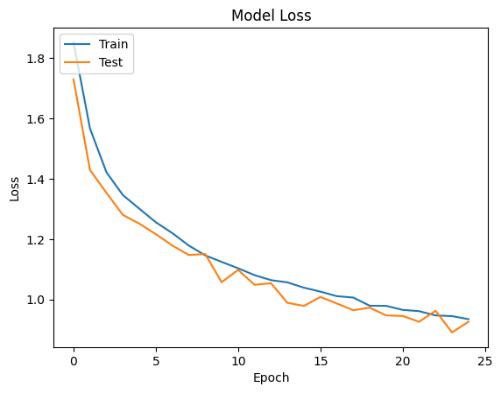
### Results of Skin Cancer Recognition Using Deep Learning Algorithm

The proposed system is developed to classify skin cancer into different type.

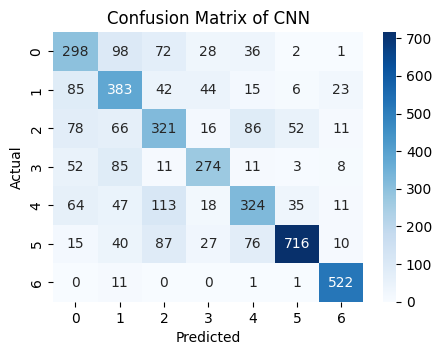
#### CNN: The CNN algorithm's results for classifying skin cancer into seven distinct types are shown in Fig. 9 below.

****

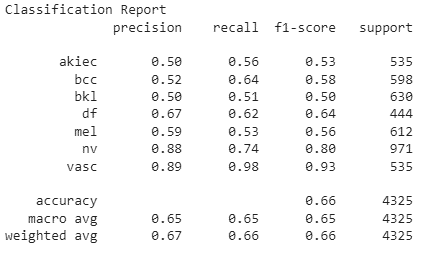
(a)

****

(b)



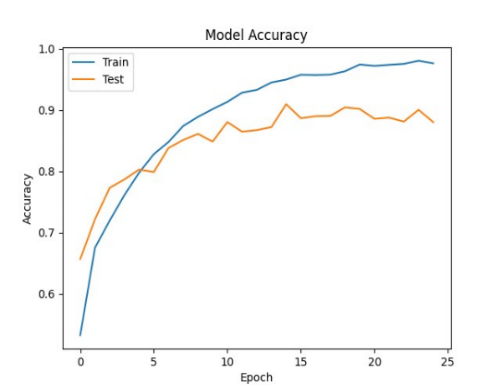
(c)



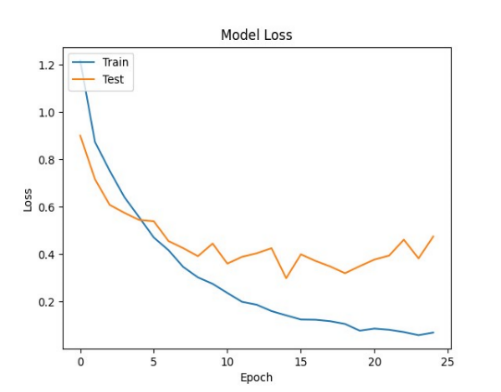
(d)

1. Training performance of CNN on HAM10000 Dataset (a) Accuracy (b) Loss (c) Confusion Matrix (d) Classification report

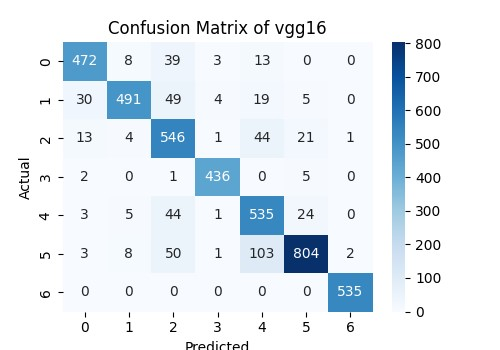
#### Vgg16: The results of Vgg16 algorithm for classification of skin cancer into 7 different type is presented below in Fig.10.

****

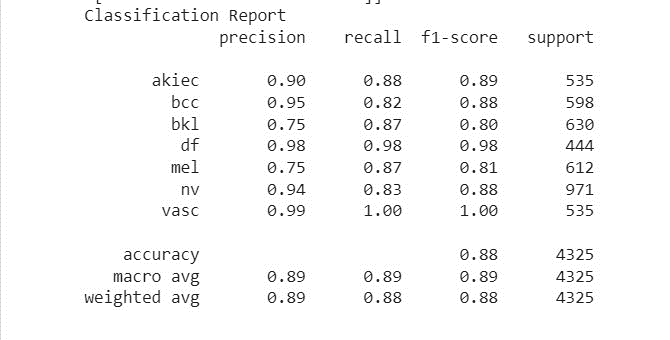
(a)

****

(b)

****

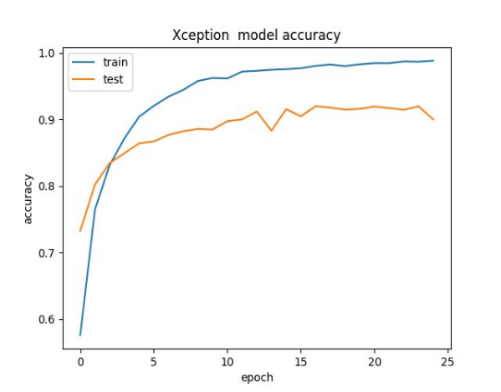
(c)



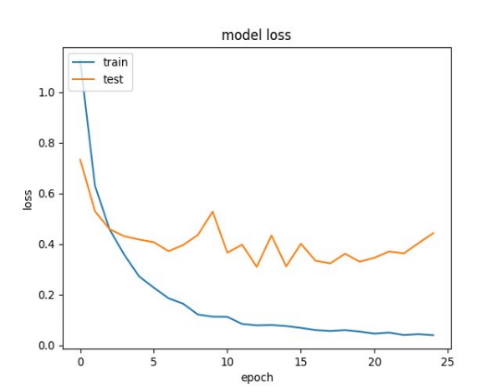
(d)

1. Training performance of Vgg16 on HAM10000 Dataset (a) Accuracy (b) Loss (c) Confusion Matrix (d) Classification report

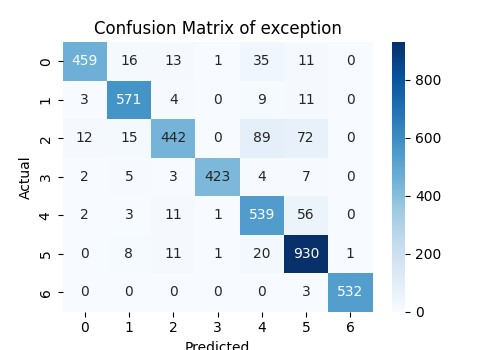
#### Xception: The results of Xception algorithm for classification of skin cancer into 7 different type is presented below in Fig.11.

****

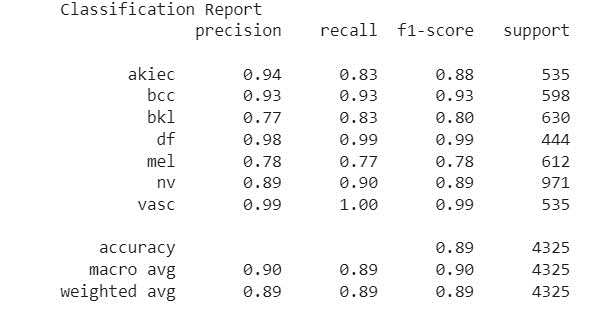
(a)

****

(b)

****

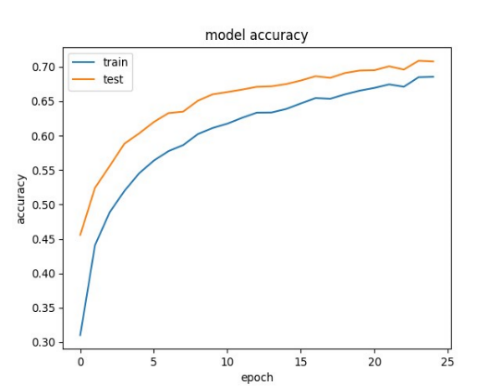
(c)



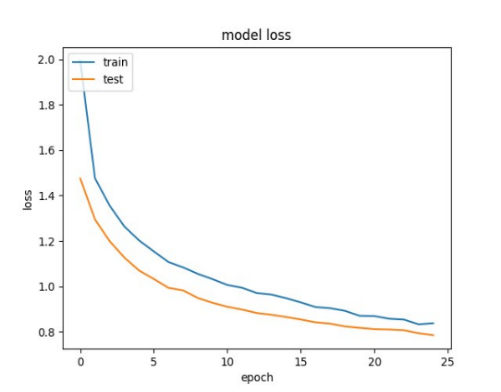
(d)

1. Fig. 6.5 Training performance of Xception on HAM10000 Dataset (a) Accuracy (b) Loss (c) Confusion Matrix (d) Classification report

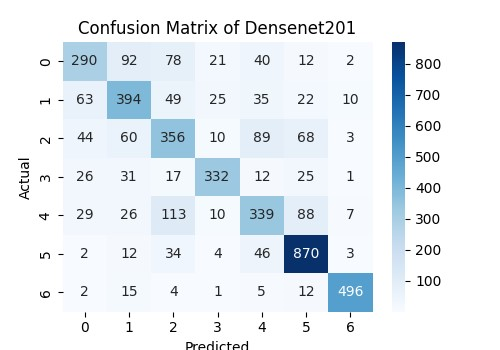
#### Densenet201: The results of Densenet201 algorithm for classification of skin cancer into 7 different type is presented below in Fig.12.

****

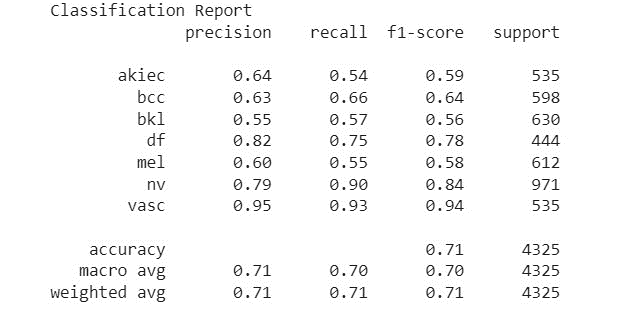
(a)

****

(b)

****

(c)



(d)

1. Training performance of Densenet201 on HAM10000 Dataset (a) Accuracy (b) Loss (c) Confusion Matrix (d) Classification report

The comparative analysis of the three algorithms for classification of skin cancer into seven different type is present in Table II.

1. Comparative analysis of different classifiers on the HAM10000 dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Algorithm** | **Precision** | **Recall** | **F1-Score** | **Accuracy** |
| CNN | 0.65 | 0.66 | 0.66 | 0.66 |
| Vgg16 | 0.89 | 0.88 | 0.88 | 0.88 |
| Xception | 0.89 | 0.89 | 0.89 | 0.89 |
| Densenet | 0.71 | 0.71 | 0.71 | 0.71 |

A comparison of different classifiers applied to the HAM10000 dataset is shown in Table 6.1, with performance measured in terms of precision, recall, F1-score, and total accuracy. Three algorithms Xception, DenseNet, and CNN are evaluated. CNN's accuracy is 0.66, recall is 0.66, F1-score is 0.66, and precision is 0.65. This suggests that although CNN does quite well overall across various parameters, it is not particularly strong in any one area. Xception, on the other hand, performs better, with accuracy, recall, and F1-score all at 0.90 and precision at 0.90. This shows that Xception maintains a balanced performance in terms of precision and recall while achieving high levels of accuracy. Densenet201's precision, recall, F1-score, and accuracy are all 0.72, placing it in the middle of the other two classifiers. This suggests a performance that is consistent throughout the assessed measures, however below Xception. With respect to precision, recall, and F1-score metrics, Xception seems to be the most successful classifier for the HAM10000 dataset, according to this analysis, which also shows a high degree of accuracy.

# Conclusion

The proposed system effectively enhances skin cancer detection and recognition using the PH2 dataset, applying augmentation techniques like rotation and flipping for better generalization. The SegNet architecture carefully balances feature extraction and spatial information, while optimization methods such as Stochastic Gradient Descent (SGD), batch normalization, and activation functions boost model efficiency. Detailed explanations of the training process ensure transparency and reproducibility. The study demonstrates how CNNs, particularly VGG16, Xception, and Densenet201, improve skin cancer detection. It highlights Xception's superior performance in accuracy, precision, recall, and F1-score. Densenet201’s connectivity and VGG16’s simplicity are key considerations, with a thorough evaluation suggesting areas for future improvement. The research advances diagnostic tools for skin cancer, emphasizing the importance of ongoing innovation in medical image processing to improve patient outcomes and public health.

The study provides a strong foundation for advancing automated skin cancer detection and categorization. Future improvements could focus on developing customized models tailored to skin cancer classification, potentially surpassing the accuracy of existing architectures like Xception and Densenet201. Expanding datasets to include diverse populations and clinical scenarios could enhance the generalizability and robustness of these models. Additionally, integrating patient demographics, histopathology, and clinical data could significantly improve diagnostic accuracy beyond visual inspection, further advancing the field of dermatological image analysis.

##### References

1. U.-O. Dorj, K.-K. Lee, J.-Y. Choi, and M. Lee, "The skin cancer classification using deep convolutional neural network," Multimedia Tools and Applications, vol. 77, no. 8, pp. 9909–9924, 2018.
2. A. Rezvantalab, H. Safigholi, and S. Karimijeshni, "Dermatologist level dermoscopy skin cancer classification using different deep learning convolutional neural networks algorithms," 2018.
3. K. M. Hosny, M. A. Kassem, and M. M. Foaud, "Skin Cancer Classification using Deep Learning and Transfer Learning," in 2018 9th Cairo International Biomedical Engineering Conference (CIBEC), pp. 90–93, Cairo, Egypt, 2018. [Online].
4. A. Dascalu and E. O. David, "Skin cancer detection by deep learning and sound analysis algorithms: a prospective clinical study of an elementary dermoscopy," eBioMedicine, vol. 43, no. May, pp. 107–113, 2019.
5. T. C. Pham, G. S. Tran, T. P. Nghiem, A. Doucet, C. M. Luong, and V.-D. Hoang, "A Comparative Study for Classification of Skin Cancer," in 2019 International Conference on System Science and Engineering (ICSSE), pp. 267–272, Dong Hoi, Vietnam, 2019.
6. A. Heller, J. S. Utikal, A. H. Enk, et al., "Superior skin cancer classification by the combination of human and artificial intelligence," European Journal of Cancer, vol. 120, pp. 114–121, 2019.
7. T. Emara, H. M. Afify, F. H. Ismail, and A. E. Hassanien, "A Modified Inception-v4 for Imbalanced Skin Cancer Classification Dataset," in 2019 14th International Conference on Computer Engineering and Systems (ICCES), pp. 28–33, Cairo, Egypt, 2019.
8. S. S. Chaturvedi, K. Gupta, and P. S. Prasad, "Skin Lesion Analyser: An Efficient Seven-Way Multiclass Skin Cancer Classification Using MobileNet," in Advanced Machine Learning Technologies and Applications. AMLTA 2020. Advances in Intelligent Systems and Computing, A. Hassanien, R. Bhatnagar, and A. Darwish, Eds., vol. 1141, Springer, Singapore, 2021.
9. S. Mohapatra, N. V. S. Abhishek, D. Bardhan, A. A. Ghosh, and S. Mohanty, "Skin Cancer Classification Using Convolution Neural Networks," in Advances in Distributed Computing and Machine Learning, A. Tripathy, M. Sarkar, J. Sahoo, K. C. Li, and S. Chinara, Eds., vol. 127 of Lecture Notes in Networks and Systems, Springer, Singapore, 2021.
10. M. Chen, W. Chen, W. Chen, L. Cai, and G. Chai, "Skin cancer classification with deep convolutional neural networks," Journal of Medical Imaging and Health Informatics, vol. 10, no. 7, pp. 1707–1713, 2020.
11. S. Jinnai, N. Yamazaki, Y. Hirano, Y. Sugawara, Y. Ohe, and R. Hamamoto, "The development of a skin cancer classification system for pigmented skin lesions using deep learning," Biomolecules, vol. 10, no. 8, p. 1123, 2020.
12. S. S. Chaturvedi, J. V. Tembhurne, and T. Diwan, "A multiclass skin cancer classification using deep convolutional neural networks," Multimedia Tools and Applications, vol. 79, no. 39-40, pp. 28477–28498, 2020.
13. Mehwish Dildar, Shumaila Akram, Muhammad Irfan, Hikmat Ullah Khan, Muhammad Ramzan, “Skin Cancer Detection: A Review Using Deep Learning Techniques,” Int J Environ Res Public Health. 2021
14. Nourelhoda M. Mahmoud & Ahmed M. Soliman, “Early automated detection system for skin cancer diagnosis using artificial intelligent techniques,” Scientific Reports, 2024
15. Maryam Naqvi, Syed Qasim Gilani, Tehreem Syed, Oge Marques, and Hee-Cheol Kim, “Skin Cancer Detection Using Deep Learning A Review,” Diagnostics (Basel). 2023
16. Umesh Kumar Lilhore, Sarita Simaiya, Yogesh Kumar Sharma, “A precise model for skin cancer diagnosis using hybrid U-Net and improved MobileNet-V3 with hyperparameters optimization”, Scientific Reports, 2024
17. Balaha, H. M. & Hassan, A.E.-S. Skin cancer diagnosis is based on deep transfer learning and a sparrow search algorithm. Neural Comput. Appl, 2023
18. Tabrizchi, H., Parvizpour, S. & Razmara, J. An improved VGG model for skin cancer detection. Neural Process. Lett. 2023.
19. Shamsi, A. et al. A novel uncertainty-aware deep learning technique applied to skin cancer diagnosis. Neural Comput. Appl., 2023.
20. Adla, D. et al. A full-resolution convolutional network with a dynamic graph cut algorithm for skin cancer classification and detection. Healthcare Analytics (2023).