RESEARCH PAPER

Performance Analysis of Modified EfficientNetB3 for Automated Skin Cancer Classification

Soujenya.Voggu[1], Shadab Siddiqui[2]

[1]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad-500075, Telangana, India,*
Geethanjali College of Engineering and Technology Hyderabad, Telangana, India
[2]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad-500075, Telangana, India*

ABSTRACT
Despite skin cancer being the most prevalent form of cancer in the world, it remains one of the most challenging aspects to diagnose due to the various forms it may take. Commonly, diagnosis revolves around visual examination and histopathology, which have their own disadvantages related to subjectivity and high expertise. This study proposed an alternative approach by utilizing the effective and efficient deep learning model EfficientNetB3. In the current study, deep learning techniques are studied in the context of their use in skin cancer spot prediction and classification. Deep networks based on the advancements in CNNs, notably the EfficientNetB3 architecture, are applied and proved to enhance the generalization capabilities of machine-based systems in cancer recognition. For the EfficientNetB3 model to learn complex patterns, it is pre-trained on extensive datasets and then fine-tuned and applied with transfer learning techniques to the dermatological images' specific characteristics. To increase the competition between features and reduce overfitting, dense and dropout layers are incorporated. This experiment's proposed setup, which involved a Kaggle dataset containing images of malignant and benign skin moles, shows the high achievement of the system in predicting and classifying skin cancer. The performance statistics, specifically the training and validation losses and accuracies, indicated the system's reliability and strength. The quantitative results presented here suggest that the proposed model is superior to the previous ones and thus indicates the possible level of accuracy and efficacy in skin cancer diagnosis by the proposed method..
Keywords: Deep Learning Skin Cancer Detection and Classification, Convolutional Neural Networks, EfficientNetB3, Transfer Learning, Dermatological Image Analysis.
How to cite this article: Voggu S, Siddiqui S, Performance Analysis of Modified EfficientNetB3 for Automated Skin Cancer Classification .Int J Drug Deliv Technol. 2026;16(2s): 327-241; DOI: 10.25258/ijddt.16. 327-241
Source of support: Nil.
Conflict of interest: None

INTRODUCTION

Recent evaluations through quantitative measures in the realm of dermatology spotlight not only the increasing prevalence of skin cancer but also the big thirst for better diagnostic tools. The data analysis indeed showed a worrying pattern in the increase of reported cases. Therefore, leading researchers and practitioners have been working on different methods for the early detection of the disease. To start with, the tests for the computational methods and machine learning algorithms have delivered promising results in terms of accuracy. Medical science and technology are thus integrating their efforts in the area of skin cancer detection -quick and accurate ones. The problems arise from the skin lesion appearance, the non-specific clinical signs, and the need for a high-level professional with specific knowledge and skills to make the diagnosis. It seems that only a well-thought-out and automated system could overcome the mentioned obstacles and, thus, offer a reliable and consistent way of predicting and diagnosing skin cancer

Nevertheless, skin cancer forecasting and classification with high accuracy still pose a great challenge notwithstanding the medical science breakthrough. Skin cancer is a general term for various cancers, all caused by the abnormal growth of skin cells. The most common types of skin cancer are basal cell carcinoma, squamous cell carcinoma, and melanoma, which differ in their specific characteristics and aggressiveness [5]. Classification is an essential step in the development and treatment of the disease as it enables the selection of the appropriate therapeutic measures that yield good outcomes for patients. The skin cancer classification process is complicated; it requires the assessment and differentiation of histological features and clinical appearance, and more recently, the use of advanced technologies such as machine learning algorithms. The latter process employs several types of information such as dermatoscopic images and patient characteristics to categorize the lesions as either benign or malignant [6]. This distinguishes the cancerous lesions at a very early stage and provides the necessary treatment

**Author for Correspondence: Soujenya.Voggu*

approaches. The present methods that rely on visual inspections and manual classifications are not dependable because they are influenced by human factors and are thus very subjective. As a result, the growing incidence of skin cancer puts more strain on the healthcare system, and this also makes the need for automatic and robust categorization methods more urgent. Because the current situation entails the availability of a solution that can produce a safe and accurate result regardless of the level of expertise, there is a growing urgency in terms of possible innovations in skin cancer classification techniques.

The requirement that the gap between detection and classification be filled perfectly delineates the necessity of employing the state-of-the-art technology such as deep learning, which can, in turn, help a great deal in enhancing the skin cancer detection and classification models. Deep learning, machine learning's leading edge has been widely recognized overseas in medical image analysis. For skin cancer prediction [7] and classification [8] its learning of rich and hierarchical patterns from complex data sets automatically is the ideal scenario. One way of dealing with the described problems is by making use of deep learning techniques, especially CNNs. This method holds much promise in obtaining accurate and rapid diagnoses because of its ability to efficiently handle and analyze a large number of dermatological images. Additionally, it could revolutionize the process of skin cancer classification by overcome the challenges outlined above. The distinctive potential used in deep learning and deep learning's ability to identify complex patterns and features from complex datasets all by itself offer an opportunity to increase diagnosis quality. The study examines the potential of deep learning as an innovative capability, while it is considered valuable in overcoming the challenges of accurate and immediate skin cancer prediction and classification.

LITERATURE

Hardik Nahata et al [9] developed a Convolutional Neural Network model, which allows an accurate identification of various forms of skin cancer, as well as to assist in an early diagnosis. In the Python programming language, the CNN classification model is programmed. Keras and Tensorflow implement the model as the backend framework. The model developed and tested with several network topologies, which is changing the types of layers used to train the network. These layers are Convolutional layers, Dropout layers, and Pooling layers, as well as Dense layers, and others. In addition, the model uses Transfer Learning to accelerate an early convergence. The model is tested and trained using data from the dataset taken from the archives of the "International Skin Imaging Collaboration" (ISIC) competition.

Mehwish Dildar et al [10] provided a comprehensive and methodical analysis of deep learning techniques for the prompt detection of skin cancer. A comprehensive examination of research publications, published in respectable academic journals and relevant to the field of skin cancer diagnostics, has been carried out. The results of this study are clarified by presenting various tools, graphs, tables, approaches, and frameworks, which help to improve understanding of the subject matter.

Mohammad Ali Kadampur et al [11] aims to develop deep learning models to classify dermal cell pictures and detect skin cancer. The use of cloud-based architecture, powered by models, relies on deep learning algorithms to develop highly accurate skin cancer prediction models. This study illuminates procedural aspects behind the making of models and their employment in the classification of photomicrographs of skin cells.

Michał Strzelecki and his team [12] studied and evaluated different applications of particular types of artificial intelligence algorithms for skin melanoma detection. The topic includes classic systems based on dermoscopic image analysis and full-body systems aimed specifically at the whole body to detect both moles and disease-related changes. Furthermore, the article explores the increasing prevalence of smartphone apps that enable the examination of lesion images. The study offers a quantitative assessment of the presented systems, with a specific focus on the validation methods for the developed algorithms. Moreover, it assesses the benefits and constraints of artificial intelligence in the processing of lesion images, pinpointing obstacles that need solutions for the enhanced use of AI in dermatology.

Md Shahin Ali et al [13] described a deep convolutional neural network (DCNN) model that uses a deep learning approach to accurately distinguish between benign and malignant skin lesions. In the preprocessing step, filters or kernels are first used to remove noise and artifacts. Then, the input images are normalized and features are extracted to develop accurate categorization. Eventually, data augmentation is done to be able to increase the number of images and improve categorization Maad M. Mijwil et al. [14]. Additionally, in this research, the proposed DCNN model performance is validated by assessing different transfer learning schemes such as AlexNet, ResNet, VGG-16, DenseNet, MobileNet, and their many combinations.

Dibaloke Chanda et al [15] selected and trained the deep learning network to analyze more than 24,000 photos of skin cancer. The chosen model in this work is the convolutional neural network for classifying photos. Three separate architectures were investigated for the title work, each with the same series classification task three architecture InceptionV3, ResNet, and VGG19, but with different settings. Thus, it is possible to establish the most efficient architecture for classifying the photos. The acquired findings were quite good, reflecting a high degree of accuracy in differentiation between a benign and malignant cancer kinds as proposed by Jaisakthi S M et al. [16]. The dataset used inthe work consisted of high-quality resolution photos of the ISIC repository from2019 to 2020. Based on the extensive testing, it was found that the InceptionV3 architecture was the most appropriate choice for the given job.

Walaa et al [17] presented a novel ensemble strategy that encompasses three Deep Convolutional Neural Networks (DCNNs) equipped with distinctive dropout layers specifically tailored to foster learning at the feature-level,

enabling DCENSnet the ability to achieve an optimized balance between bias and variance [18]. The experimental investigation on the well-recognized HAM10000 skin lesion dataset indicated that the proposed model surpasses the capability of the state-of-the art networks as of today.

Vipin et al [19] proposed a Deep neural model that can classify different forms of skin cancer binary, whether the lesion is melanoma or nonmelanoma, accurately. Image datasets in this study are publicly available datasets from the ISIC-2019 and ISIC-2020 challenges, which are characterized by various resolutions and imbalanced classification problems. Researchers utilized the EfficientNet design for getting more accurate classification, applying transfer learning for the model. The architecture has a remarkable ability to regulate the network's depth, breadth, and resolution in order to learn multiclass lesion photos beneficially. Having further extended the dataset due to the imbalanced nature of the classes and increased the amount of metadata presence for better results of classification, as well as having optimized the work of the EfficientNet model with optimizer ranger that made a huge impact on the reduction of necessary hyperparameters tuning due to which the current study has been able to approach state-of-the-art outcomes [20]. Multiple studies examined the use of different transfer models, noting the prevalence of EfficientNet versions in the topic of processing images of skin lesions.

PROPOSED METHOD

3.1 Skin Cancer Prediction

Early detection is essential if the treatment of skin cancer, a common and potentially lethal disease, is to be successful. Even more remarkable have been deep learning techniques in medical science, proving outstanding performance in picture analysis and template recognition. This study investigates deep-learning models for skin cancer identification on the large volume of photos in dermatological databases to boost diagnostic accuracy. The purpose of the investigation is to demonstrate that it is possible to detect skin cancer at an early stage using artificial intelligence.

3.1.1 EfficientNetB3

EfficientNetB3 is part of the efficientNet family, a convolutional neural network architecture that boosts accuracy and performance while using far fewer parameters than traditional CNNs. The primary aim of EfficientNet is to strike a balance between the model's size and its computational effectiveness, thus enabling it to run on a wide range of devices and applications.EfficientNetB3 is one of the variants of the EfficientNet architecture. The term "B3" in its name describes the scale or depth of the model within the EfficientNet structure. In the context of EfficientNet architecture, scaling is a strategy in which one looks at expanding the depth and breadth of the network simultaneously to find the best trade-offs between the model's capacity and computing efficiency. As a standalone figure, B3 is larger and more powerful than the previously mentioned B0 while also maintaining the optimizations that made EfficientNet successful.

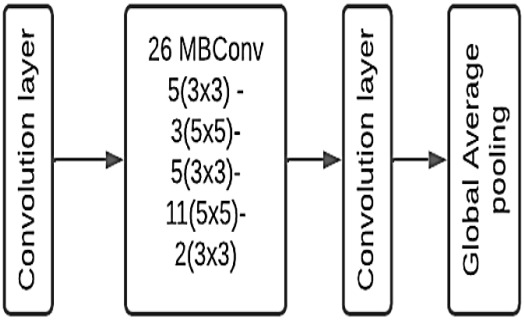


Fig. 1 EfficientNetB3 Architecture

The design aspects of EfficientNetB3 are based n compound scaling where depth, breadth, and resolution of the network are increased concurrently to improve performance. Figure 1 shows the architecture of EfficientNetB3. The underlying structure is manufactured with efficient building blocks inspired by inverted residual blocks with linear bottlenecks, akin to those found in the latest CNN models. The blocks are designed to be good at capturing complex data patterns efficiently, reducing the workload on processing.

EfficientNetB3 is pretrained on huge datasets such as Imagenet in order to grasp all common characteristics available in different types of images. Subsequently, the model can be either fine-tuned or transfer-learned in such a way that it goes through another domain pre-training that is more suitable for traditional use of scarce labeled data. Conversely, EfficientNetB3 is the termination of the process where we come to terms with model complexity and efficacy that are good enough in situations with limited computing power. That is the reason it can be utilized in a wide range of computer vision applications.

EfficientNetB3 has many benefits:

Enhanced Precision and Broad Applicability: EfficientNetB3, one of many models in the EfficientNet family is leading the way of high performance accuracy on a series of image classification benchmarks. The method utilizes a compound scaling technique to the utmost by changing each of the network's dimensions (depth, width, resolution) corresponding to its ability to perform better—which of course without any extra computational requirements.

Optimal Model Size: EfficientNetB3 has achieved a remarkable accuracy while still maintaining a good trade-off between the size of the model and its performance. At the same time, it doesn't require so much memory like the larger networks, thus it is easier to implement in less powerful areas of application.

Computational efficiency: EfficientNetB3 applies the method of compound scaling to swing the pendulum towards the best usage of computation resources. Accuracy per parameter and computational overhead of the model are comparison with other standard models performance in a new way, thus the model has been placed at the best performance of the two extremes.

Transfer Learning Capabilities: Transfer learning is the concept that a machine learning (ML) model can take

knowledge it has gained from one task and apply it to another task that is related. EfficientNetB3 is one of the models that used transfer learning by initially training on a large dataset and subsequently allowing it to be fine-tuned for the specific task. It's especially useful when not much data is available as per the task.

Adaptability in Various Tasks: The architecture of EfficientNetB3 is primarily for image classification, however, it can be utilized in other areas of computer vision such as object detection and segmentation. The very fact that it is this flexible is what makes it applicable to numerous kinds of projects practically.

Scalability: Scalability refers to the property of a system or process that can be enlarged without compromising its performance, which also includes withstand increased load. The compound scaling that is done in EfficientNetB3 allows a smooth transition of the model size from say EfficientNet-B0, B1, B2 and so on. Which enables you to choose a model that fits the computer resources needed and specific work requirements.

Cutting-edge performance: EfficientNetB3 achieved competitive results on standard benchmark datasets. And hence it is a very good contender for several image classification tasks.

Noise tolerance: EfficientNetB3 incorporates techniques such as batch normalization and dropout in the model architecture, which makes the trained model more robust to noise or diverse set of training data.

Suitable for implementation in real-world scenarios: EfficientNetB3 is already a highly accurate and efficient model that can be used in real-world applications. This is really beneficial when working on limited computational resources ie; edge device, mobile app etc.

Assistance and available materials provided by the community: EfficientNetB3 is one of such critically-acclaimed models that have received great community support. Consequently, there are frequent tools, tutorials and pre-trained models available to help make it easier to use in different applications.

3.1.2 Proposed Method

The framework of the proposed method using EfficientNetB3 is shown in figure 2

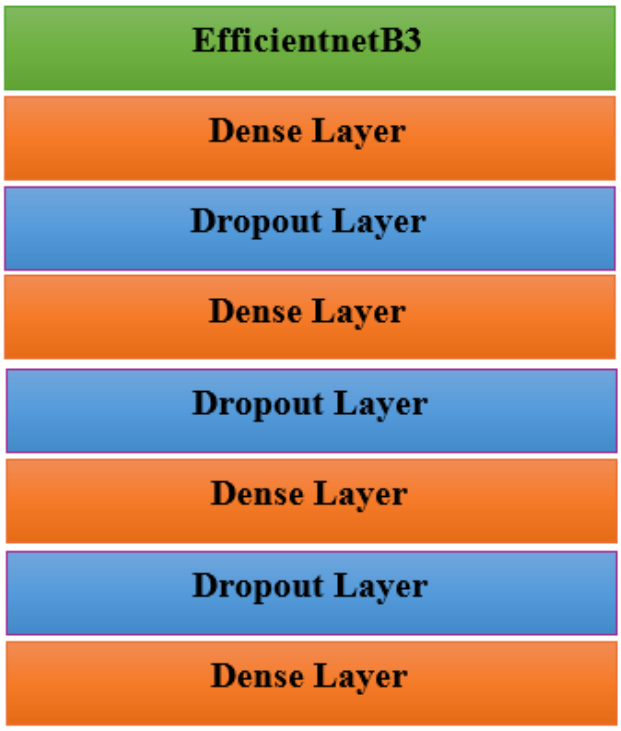


Fig. 2 Proposed method Architecture

Dense Layer

Dense Layer It is a crucial element of neural networks, and it plays a vital role even more in deep learning models. The key of this system is that it connects every single neuron to each and every neuron in the previous layer, which results in an enormous network of connections. Therefore, this means that every neuron in current layer is informed from all neurons of previous layer which are represented by statistical weights.Dense Layer is the most basic form of such layers who are responsible for capturing complex patterns / correlations in data by adjusting weights during training. The weights are essential for controlling the connection strength of neurons and are required to influence predictive or classificational power on a network. The model adjusts these weights from learned data, so the projected vectors are as close to the desired output.

The Dense Layer does simply perform a weighted summation of the input values for each neuron; but it also applies an activation function to introduce non-linearity. This activation function plays an important role in allowing the neural network to learn complex patterns present in the input which are inherently non-linear.Further, the Dense Layer also enhances neural networks capabilities to a great extent in learning various characteristics and correlations simply because it offers such high level of flexibility. This layer shows generality and ability to adjust to any kinds of input making it helpful for image detection, natural language processing, etc.

Dropout Layer

Dropout was an important and useful ingredient in many neural network architectures (especially deep learning models). It is a function used to regularize or govern the overfitting of your model, which happens when the model

learns not just from the real data distribution but also, unintentionally, from noise present in the training data. Dropout is a popular regularization method that reduces overfitting by randomly setting neurons in one layer to zero during the training process.The Dropout Layer causes a particular portion of neurons to be randomly deactivated during the forward pass of training, momentarily setting their outputs to zero. By employing this strategy, the network is obliged to create extra features, which is assumed to result in a more comprehensive and strong product. Because of dropout, a different grouping of neurons connection is produced every time, hence, the reasoning behind this is similar to training through ensemble methods.

Inspired by this learning process, the idea of Dropout is to make it such that no single neuron can dominate the training and prediction processes. Model can better handle new inputs yielding in that way better results for future unknown cases. Dropout is a way to do generalization, as it keeps the neural network from relying too heavily on any one set of attributes. It creates a more stable and agile learning process.Dropout is usually used only in the training process, and hence it should be disabled during the testing or inference process. The complete network is used during testing, and the weights obtained are adjusted for the purpose of not applying a dropout. In this way, it guarantees that predictions from the model on new data can be trusted and they are accurate.

ReLU

Rectified Linear Unit aka ReLU is the most commonly used activation function in deep learning models and artificial neural networks. Since this method is so simple and effective, it is used today in a wide range of different applications such as image identification or natural language processing. This special feature of Rectified Linear Unit will be even more explored in the following section.

ReLU helps in introducing non-linearity to the network. In a neural network, each neuron receives input signals and then calculates this with weights to convey an outcome using some activation function. ReLU is a functional form which virtually turns off any of the negative numbers in that weighted sum.

Mathematically the function can be written as:

$$f(x) = \max(0, x) \qquad (1)$$

where x represents the input.

Advantages of the ReLU are its simplicity and computational efficiency. It is a traditional activation function that produces the simplest neural network and due to very simple math, it really computations fast so this makes ReLU ideal for training deep neural networks with many layers. Also, the ReLU is non-saturating which makes it less susceptible to vanishing gradients than some other activation functions. This capability helps in training deep neural networks effectively.

However, the Rectified Linear Unit (ReLU) poses a problem. The downside of these improvements is that if the “dying ReLU” phenomenon takes place when weights are updated such that no value in a layer can be positive, therefore some neurons will never activate during training and always output 0, then back-propagation learns to zero out just those nodes. This is known as the gradient saturation, a large range value passes through a ReLU neuron. Consequently, the neuron's weights are adjusted in a manner that consistently produces an output of zero. In order to tackle this problem, different variations such as Leaky ReLU and Parametric ReLU have been proposed. These differences enable the existence of minor, non-zero alterations in negative inputs, thereby preventing neurons from becoming completely inactive.

Softmax

The softmax function is a popular mathematical function that you will find typically used in different chapters of mathematics fields, like statistics and machine learning. It is a widely used in the final layer of the neural network to solve problems of multiple class classifications. Softmax is short for soft maximum It is used to convert a vector of raw scores or logits into a probability distribution.

Softmax function: Its input is a vector of real-valued integers and transforms them to a probability distribution over different classes. Which is calculated by the exponential function on every vector element (to ensure they are positive) and then, normalize. Mathematical form of the softmax function

$$softmax(x)_i = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}} \qquad (2)$$

Here, *x* represents the input vector, *N* denotes the number of classes, *e* symbolizes Euler's number (about 2.71828), and i signifies each element in the vector. The numerator takes the exponential of a given class and the raw score, while the denominator sums over all classes. This is to make sure sum of final figures adds up to 1. The softmax function is integral to transforming the original outputs of a model into probabilities, allowing for the easily understandable interpretation of those predictions made by your model. The projected class is usually the one with the highest probability. When training a neural network using optimization methods such as gradient descent, differentiability is a fundamental requirement.

3.1.3 Adamx Optimizer

Adamax is a frequently used optimization technique in the training of artificial neural networks that is based on gradients. The name "Adamax" is formed by merging the terms "Adam" with "infinity norm" (max). The integration of important ideas from the Adam optimizer and the infinity norm efficiently and reliably increases the training of neural networks.The scale of adjustments to the model's parameters by training learning_rate in Adamax. This example sets the learning rate to 0.001, so that only a small step is taken. In optimization algorithms learning rate matters a lot, and it can significantly determine your neural network performance in terms of how fast it converges to the minimum (convergence) during training.

Combining components from the Adagrad and RMSprop algorithms, the Adamax optimizer is a modified variant of the Adam optimizer. Adam adjusts each parameter's learning rates by considering its squared gradient and

previous gradient. However, Adamax incorporates the idea of the infinite norm. A mathematical notion known as the infinity norm denotes the highest absolute value included within a range of values. An optimization approach called Adamax is used on the exponential moving average of the gradient's infinity norm.Similar to Adam, but can converge faster because it uses the infinity norm that provides better stability and predictability for sparse gradients or noisy training data. Therefore, it seems like a decent choice when you are trying to train your neural network and there might be some noise or ambiguity in the data. Using a low learning rate in this case 0.001 is the common way to tradeoff between stability and convergence speed. This would prevent the model to oscillate or diverge during training.

The paper introduces a novel approach to early-stage skin cancer detection, using advanced deep learning methods, with a specific emphasis on the EfficientNetB3 architecture. EfficientNetB3 merges compound scaling in such a manner that the model's complexity and the computation's efficiency come to a consensus. This way, the model ranks high as one of the best performers in the area of image processing with both accuracy and efficacy. Besides, the model has been pre-trained over massive datasets such as ImageNet, which has added new features, i.e., high accuracy, model size optimization, fast computation, transfer learning, easy adaptiveness to various tasks, scalability, best in the field performance, relaxation of the noise requirement, and all that needed for real-world applications. The proposed method consists of basic components like Dense Layers for pattern recognition, Dropout Layers to prevent overfitting, ReLU activation function for linearization, and Softmax for multi-class classification. To optimize training of the neural network, Adamax optimizer - which merges attributes of Adam and the infinite norm - is utilized for the whole training period to ensure robust optimization. Sometimes, the learning rate parameter becomes very important to maintain stability and facilitate convergence. The purpose of the design is to create a reliable and efficient system for fast skin cancer detection using artificial intelligence.

3.2 Skin Cancer Classification

This study aims to improve the accuracy of skin cancer classification using CNNs efficiently. The major challenge in the automated diagnostic process is to extract complex patterns and information from images, for which CNNs are used.

3.2.1 CNN

Convolutional Neural Networks (CNNs), these are the standard deep learning models created for computer vision tasks such as object detection, image classification and recognition. Because of their unique design, inspired by the functioning of visual cortex in animals, CNNs have shown to work really well on complex visual data.A Convolutional layer is the fundamental building block of any Convolution Neural Network (CNN). Simply put, convolution is sliding a small filter or kernel over an input image that captures the localized properties of edges, textures and patterns. This could be a method to teach the network how to represent the input data correctly in some form of hierarchy. At the top layers, basic elements are learned and gradually combined to achieve even more complex and abstract features in deeper lower level.

CNNs require pooling layers to reduce the computational load on input volume along with its spatial dimensions. Two common pooling techniques are the mean pool, which computes its average and max pool, which picks up the largest value from a group of values. Pooling is capable of noticing the most significant properties extracted by your network, which stabilizes it and makes it more robust to changes on input.CNN makes use of weight sharing, where the same set weights and biases are used in different spatial regions on input data. When using large datasets, CNNs provide good results because of the parameter sharing which reduce the total number of parameters and able to find similar features from input sections that allows for learning with the pattern observed.

CNN is a collection of convolutional and pooling layers, followed by eventually concluded fully connected layer to achieve high-level feature integration efforts as well as classification. The final stages usually be using softmax activation and hence, probability distributions over types of classes which will help us to make predictions. Convolutional neural networks (CNNs) are trained and their parameters tuned using labeled datasets and methods like gradient descent with backpropagation.

3.2.2 Proposed CNN

Figure 3 shows the proposed CNN model designed to classify the categories of skin cancer.

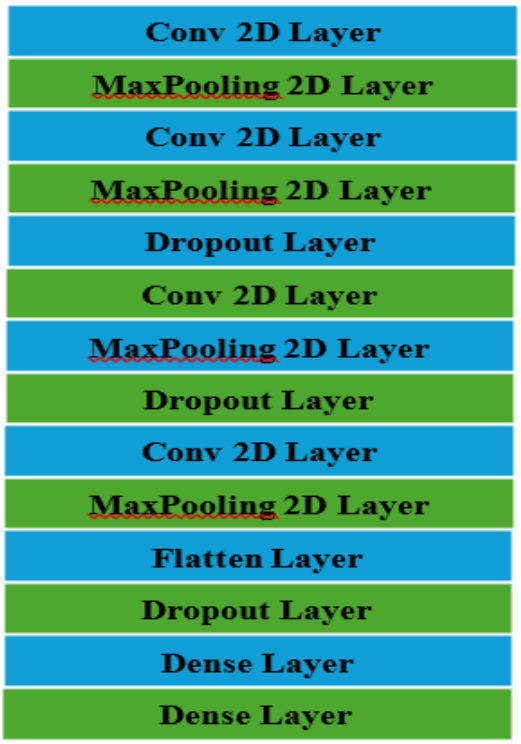


Fig.3 Proposed method Architecture

Convolutional 2D Layer

The Convolutional 2D Layer is significant part of convolutional neural networks (CNNs) used for solving

image related tasks such as segmentation, object detection and picture classification. Since this layer removes any hierarchy information about input images it is very important to help the network learn and detect complicated features.The Convolutional 2D Layer on the input data performs convolution. In convolution, there is a small filter called the kernel that slides through an input image and computes dot product between this kernel and area of the image it is currently on. The above process is repeated all over the input image to get a stack of filtered output values. To catch important patterns and features, the filter's weights are updated variable parameters by a network during training.

One of the biggest advantages of a Convolutional 2D Layer is its ability to preserve spatial hierarchies. Initially the layer identify local patterns like edges and textures using standard parameters along with basic filters. Higher layers are devoted to more abstract and complex features such as parts of objects, layouts etc. Hierarchical feature extraction specifies how well the network can understand a picture in terms of its content from top to bottom.In addition to this, there are few more stuffs given as input for Convolutional 2D Layer which enhances the performance. The stride parameter determines how far the filter is translated at each step of traversing the input and consequently affects how detailed will be the feature maps. Padding is a parameter which specifies whether or not to use the border, it add some extra pixels around input in order to maintain spatial information on edges. These components heighten the flexibility of the layer and make it a more efficient way to handle various image sizes while still preserving spatial correlations.

MaxPooling 2D Layer

Convolutional neural networks (CNNs) rely substantially on the MaxPooling 2D layer, and they are widely used image recognition applications. The main purpose of this layer is to reduce the spatial dimensions of the input feature maps whilst retaining its essential information. This is very successful mechanism to remove computational complexity as well as extract informative features via MaxPooling.In the pooling phase, a input feature map is split into non overlapping rectangular regions. These areas are usually recognized as pooling windows or kernels. The maximum value is selected in each window, which is to be used as representative of that corresponding location. MaxPooling Layer uses the maximum activation value in a given local region, can capture the most important features. The pooling method is individually applied to each channel in the input feature map.

MaxPooling is one of the simplest ways to achieve translation invariance, which means reducing the sensitivity of the network on how precisely features got placed at input. This property makes it more invariant to photo inputs, so locations of items in sample photos can change. MaxPooling further helps in reducing the spatial dimensions of feature maps helping to decrease the number of parameters and improving efficiency.In MaxPooling 2D, the default is a pooling window of size 2×2 with a stride value two which will move the pooling window by two pixels at each movement. This will downsample the data by dividing spatial dimensions of input feature map into half. The size and stride of the pooling window will be determined by how much downsampling, as well individual assignment requirements.

Dropout Layer

Dropout is a regularization technique commonly used in neural network topologies to minimize overfitting. The Dropout layer makes this different by selectively disabling values of input units by setting them zero during training. This essentially renders those units inactive in the network for that iteration. This reduces the over dependence of these neurons and features during training, making it harder for them to specialize in a single task or example but leaves room for more general representations.

There is a stochastic function, dropout damages the dependence between neurons and increases the generalization capability of network reducing overfitting. Dropout rate, often specified in the range 0.2–0.5, is a proportion of units to randomly drop out while training or creating model ensembles. Dropout has been demonstrated to increase the capacity of neural networks in generalization, hence improving the ability for a model with dropout to perform better at new examples.

Flatten Layer

The flatten is crucial to neural networks in the design and use of deep learning frameworks such as TensorFlow or PyTorch. This function simply flattens the input data that is passing as multi dimensional array, matrix or tensor to one-dimensional. It is necessary to transform from convolutional or pooling layers, which traditionally operate with multidimensional input, i.e. it can be a picture, to a fully connected layer, to which each neuron is connected to all neurons of the previous layer.

Flattened layer compresses the spatial dimensions of the input while retaining the necessary information to advance to fully linked layers. Consequently, this layer connects convolutional characteristics learning layers with dense classification layers, enabling the network to study and develop hierarchical features from complex input information.

Dense Layer

A Dense Layer, which is also referred to as a fully connected layer, is an essential layer in neural networks from which other machine-learning tasks, especially deep learning, are derived. In this layer, every neuron or node is connected to all other neurons of the previous and next layers, hence dense founding a matrix of connections. The layer is referred to as “dense” due to its connecting the other neuron.

Neurons of this layers receive input from the previous layer, the input is then multiplied by matrix of weights and sum the results in order to get the output of each neuron. In addition, a bias term is often added to the sum. The activation function is then applied and introduces non-linearity to the layer. This dense layer is essential to capture data’s complex patterns and relationships, form which the neural network learns hierarchical representations and enables predictions across a variety of tasks, including

image classification, natural language processing, and regression, among others.

Adam Optimizer

Adam method is highly popular as a machine learning model optimization technique and is often implemented in neural networks. Adaptive Moment Estimation (Adam) is created by integrating two other algorithms: RMSprop and Momentum.The learning rate, which is denoted as lr, is a vital hyperparameter in optimization algorithms like Adam. Fundamentally, it determines the size of the steps that are taken in the course of the optimization process. A higher learning rate results in fast convergence, but it increases the risks of moving beyond the optimal values; in a converse sense, a low-level leads to slow convergence in a stepwise fashion with improved accuracy.

In the phrase "Adam(lr=0.001)," the value of lr=0.001 specifies the learning rate for the Adam optimizer as 0.001. This particular figure of the learning rate is often used as a standard initial setting and is commonly suggested or utilized as a default. Nevertheless, the ideal learning rate might differ based on the particular dataset and issue being addressed. A hyperparameter is an adjustable parameter that practitioners may need to optimize throughout the model training process in order to obtain optimal performance.When setting up the Adam optimizer with a learning rate of 0.001, it implies that in each training iteration, the model's parameters will be adjusted by multiplying the gradient of the loss function with respect to those parameters by 0.001. This procedure is carried out repeatedly, aiming to minimize the loss function and enhance the model's performance on the assigned job.

The provided sequence of layers depicts a standard architecture of a convolutional neural network (CNN) used for the classification of skin cancer. The Convolutional 2D layers have the task of acquiring spatial hierarchies of information in the input pictures, detecting patterns that are indicative of various skin diseases. MaxPooling 2D layers decrease the size of the feature maps, emphasizing the most significant information. Dropout layers are applied to reduce the issue of overfitting, which implies the stochastic deactivation of a portion of neurons during the training process. The Flatten layer reshapes the 2D feature maps into a 1D vector, making data suitable for the fully connected layers. The fully connected Dense layers at the end of the network make the final class decision based on the learned features. The Adam Optimizer is a computational technique applied to modify the model’s weights during the training process, which enhances the speed at which how quickly the model converges and how effectively it performs in a Convolutional Neural Network. In essence, the architecture is intended to generate sequentially organized features from skin cancer images and differentiate the disease with high precision. Dropout layers and the Adam optimizer are employed to regularize and fine-tune the training procedure.

EXPERIMENTAL RESULTS

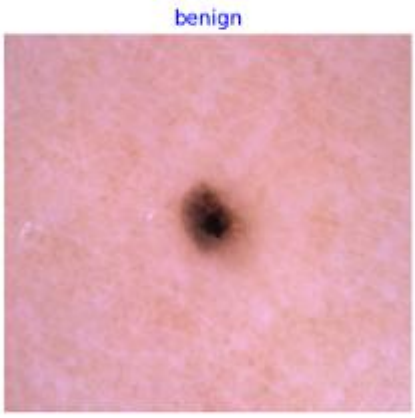
This section provides a detailed analysis of the results obtained through the recommended methodology for the current simulations. In the current simulation, the dataset was available in the open-source Kaggle site. Implementing recommended methodologies of processing the datasets used in the present study.

4.1 Skin Cancer Prediction

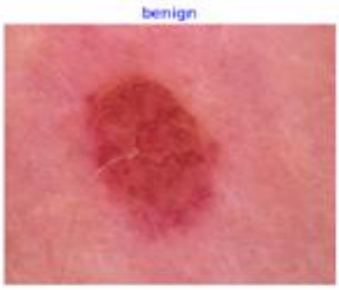
This dataset consists of an equal number of photos that belong to one-of-two types: 1) benign and (2) malignant. It includes two directories. There are 1,800 moles images 224*244 pixels of different categories of moles in total; each of them belongs to a different category. Figure 4 shows the sample images in the dataset.

A model is able to learn from the training data, there are two key metrics used in this manner and these are Training accuracy and Validation accuracy. The training accuracy is a measure of what percent of the instances in the training set have been correctly predicted, meaning how much has actually learned from the information given to it. The metric quantifies how spot on the model was with its predictions for the labels of these instances it did see in your training data. A high training accuracy suggests the model has been able to recognize patterns and relationships within our data

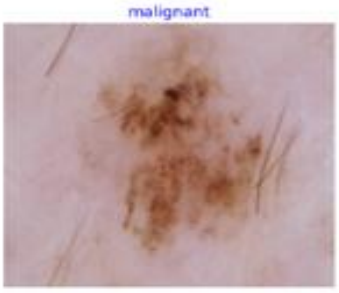
(a)



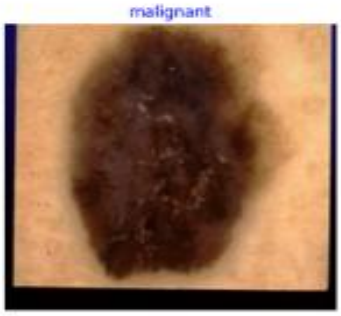
(b)


benign

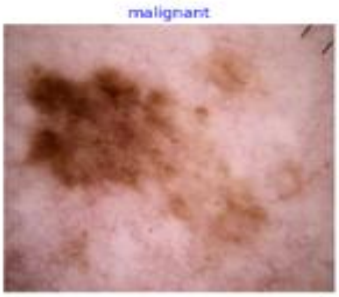
(c)


malignant

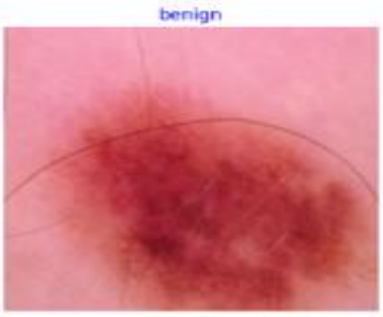
(d)


malignant

(e)


malignant

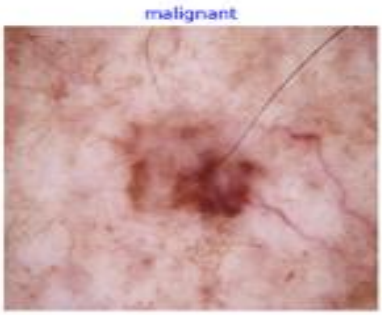
(f)


benign

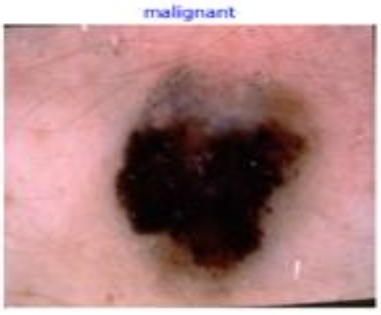
(g)


malignant

(h)


malignant

(i)


malignant

(j)

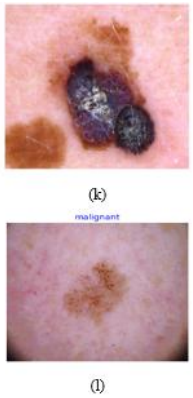


Fig.4 Sample images in the dataset

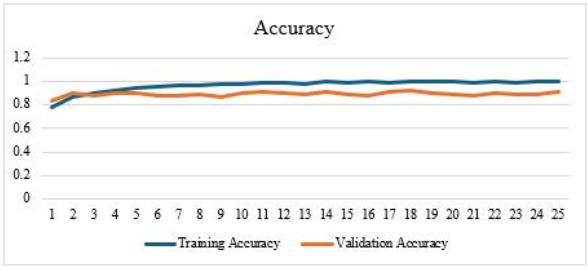


Fig. 5 Training and Validation Accuracy

Validation accuracy, shown in figure 5, is a further statistic that is used to verify the performance of a model. The validation accuracy refers to the performance of the model on data it has never seen before. The validation set (dataset), on the other hand, is used to confirm whether the model can generalize well or not. More importantly, a high validation accuracy indicates that the model has learned to generalize and predict fresh data at an acceptable level as well.

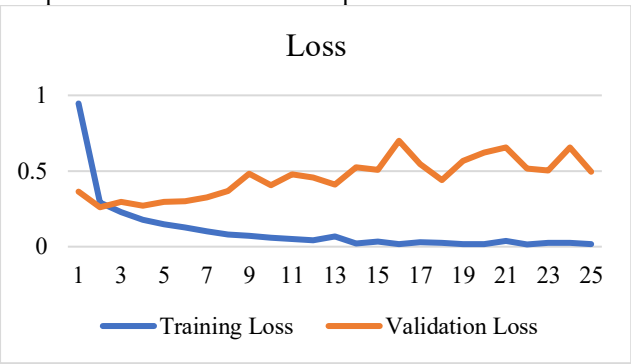


Fig. 6 Training and Validation Loss

Training loss and validation loss should, shown in figure 6, enhance the accuracy while becoming powerful to keep track of learning capabilities. Training loss is used to determine the difference between the predictions of the model and the actual labels in the training data. The focus of the training process is to minimize this metric, meaning that the algorithm is taught to produce more accurate predictions. However, the lowest training loss cannot be used as a guarantee for the model's performance on the new, unseen data.

The validation loss is how the model performs on the validation set, as well as a measure of its generalization ability. To prevent overfitting from occurring, the latter is when the model becomes so focused on trying to memorize the training data that it performs poorly on new or unseen data, one must carefully optimize their loss on training while also paying close attention to loss on validation. Thus, one ensures that the model not only generalizes well in relation to the training data but also in relation to novel samples, as it has successfully transferred its learning

Table 1. Classification Report

	Precision	Recall	F1-Score
benign	0.90	0.95	0.92
malignant	0.94	0.88	0.90
Total Accuracy	0.92		

Table 1 below is a classification report, which is a short summary of the performance of a classification model, usually utilized in the machine learning domain. It summarizes the metrics such as Precision, Recall, F1-Score of different classes available in the dataset. In this specific case, the classes were "benign" and "malignant," implying a binary classification problem that is generally related to medical or security settings.

The statistic referred to as "Precision" in the table measures the accuracy of the model in predicting positive cases. The precision of the "benign" class equals 0.90, which means that the model predicted 90% of the cases as accurate who were actually true benign. The precision of the "malignant" class equals 0.94, which means the model predicted 94% of the occurrence related to be malignant as accurate. Thus, precision is necessary in the situations whenever the positive false results are costly.

The second statistic, recall, measures the fraction of the positive in the dataset that the model can successfully capture. A value of 0.95 for the benign category implies that the model captured 95% of the actual benign cases. It also suggests that the value of 0.88 for the malignant category implies that it was able to capture only 88 % of the real malignant. This metric becomes vital when failing to catch the positive events possesses a huge risk.The third measure, sometimes referred to as the F1-Score, is given by the harmonic mean of the accuracy and recall as shown below. It is a more balanced measure to fairly asses how well our model is performing. The F1-Score for class "benign" is 0.92, while for "malignant" is 0.90. A larger F1-Scores means greater overall superiority by accounting for both accuracy and recall.

The lower row of the table shows the Total Accuracy, which reflects the model's integrated accuracy across all groups. In this example, Total Accuracy is 0.92, which means that the model classified 92% of all cases in the dataset correctly

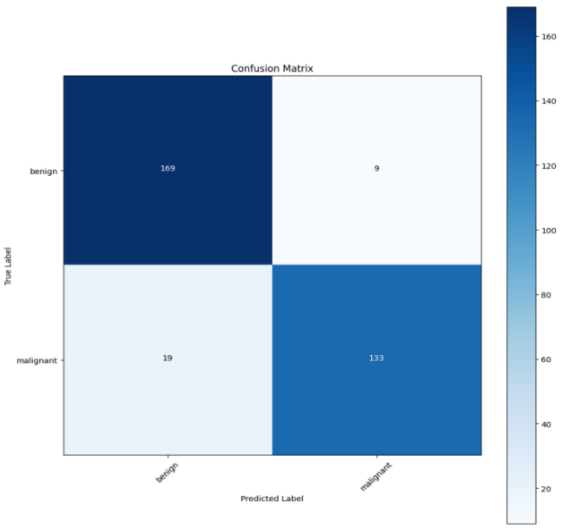


Fig.7 Confusion Matrix

Confusion matrix, shown in figure 7, is a tabulated summary used mostly to determine the effectiveness of a classification scheme. The criteria thoroughly assess the effiency of the model by tabulating the total amounts of true positives (TP), true negative (TN), false positive (FP), and false negative (FN) measures regarding specific problem. The confusion matrix could also be beneficial to medical diagnosis or any other problem when it comes to binary classification.

The confusion matrix provided, referred to as Figure 7, is associated with a binary classifying problem where the two classes are labeled as “benign” or “malignant.” The matrix is divided into four areas with the rows presenting the actual class and the columns reflecting the predicted classes.

Within this particular matrix:

The cell located in the top-left corner (169) corresponds to the number of true negatives (TN), which indicates the accurate prediction of cases as benign.

The cell located in the top-right corner (9) corresponds to false positives (FP), indicating the cases that were classified as malignant but are really benign.

The cell located in the bottom-left position (19) represents false negatives (FN), indicating cases that were classified as benign but are in fact malignant.

Then, the cell in the bottom-right position (133) are true positives (TP), meaning cases that were correctly predicted as malignant

Table 2. Comparative Analysis

Methods	**Accuracy**
CNN [17]	0.86
Resnet-50 [18]	0.87
EfficientNetB0 [19]	0.91
Proposed EfficientNetB3	**0.92**

The data in table 2 present a detailed analysis of various protocols applied to the same job, with accuracy serving as evaluation scheme. The techniques employed were, Convolutional Neural Network (CNN), Resnet-50, EfficientNetB0 and a new model called EfficientNetB3. The accuracy values of the two methods evaluated here show how well each model was able to perform its desired task. In terms of the best performance, EfficientNetB3 is proved to have outstanding performance among all other methods with its highest accuracy rate 0.92. This indicates that this model is able to detection and understanding various complex patterns in data.

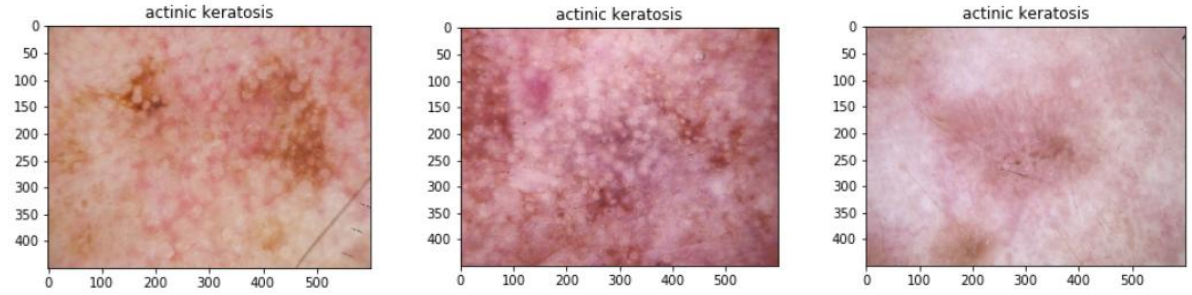
4.2 Skin Cancer Classification

The dataset contains 2357 photos of oncological diseases (malignated + benign) from the International Skin Imaging Collaboration (ISIC). The images within the competition are labeled according to an ISIC classification scheme and had equally divided numbers but a small number of classes showed high accuracy with melanoma and benign moles as majority.

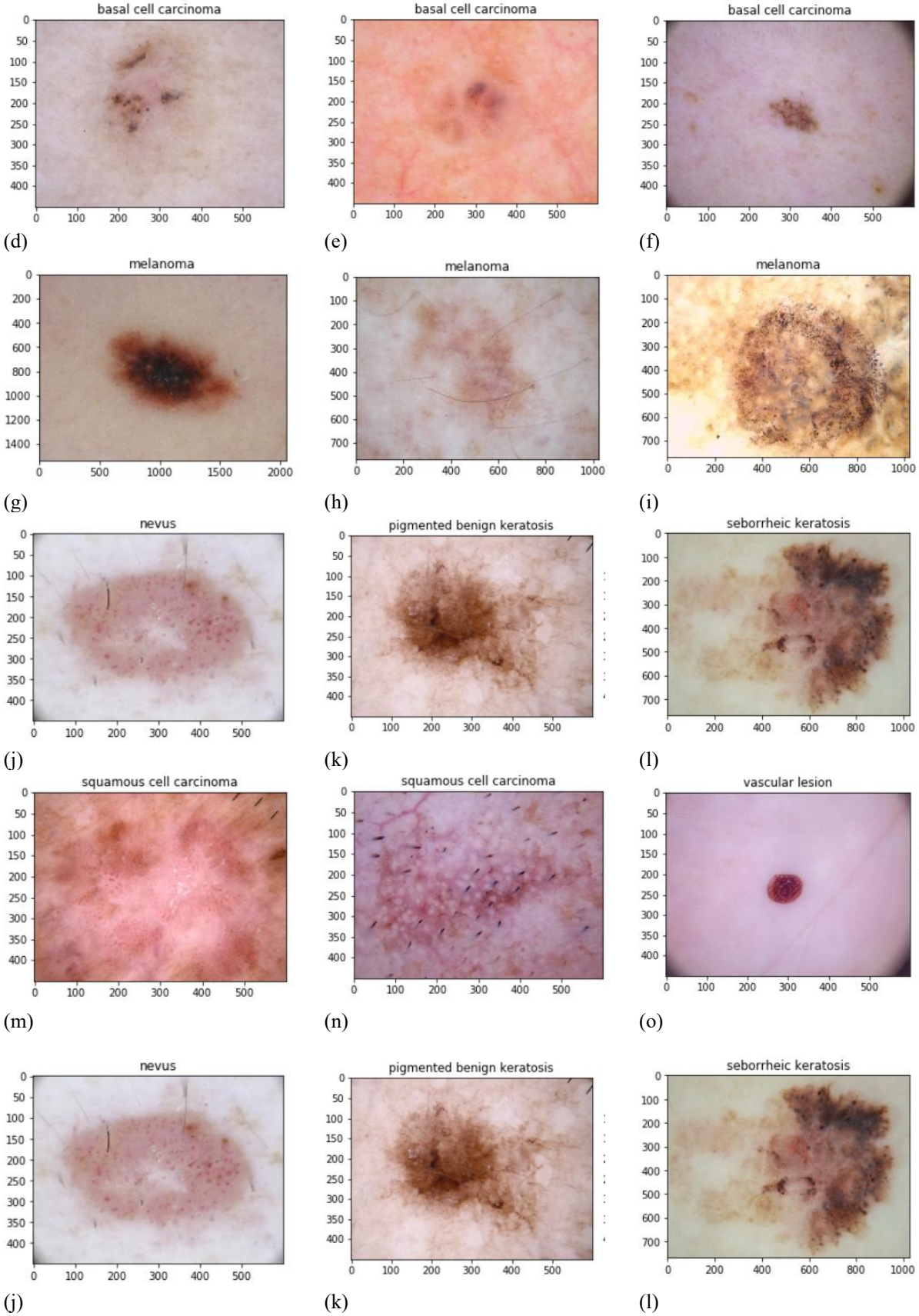
The data set contains the following diseases:

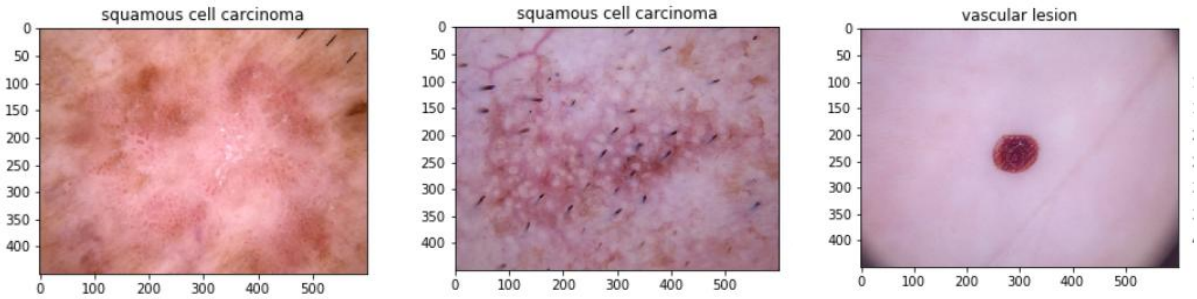
actinic keratosis
basal cell carcinoma
dermatofibroma
melanoma
nevus
pigmented benign keratosis
seborrheic keratosis
squamous cell carcinoma
vascular lesion

The sample images of the dataset are shown in figure 8.



(a) (b) (c)


basal cell carcinoma
(d)
basal cell carcinoma
(e)
basal cell carcinoma
(f)
melanoma
(g)
melanoma
(h)
melanoma
(i)
nevus
(j)
pigmented benign keratosis
(k)
seborrheic keratosis
(l)
squamous cell carcinoma
(m)
squamous cell carcinoma
(n)
vascular lesion
(o)
nevus
(j)
pigmented benign keratosis
(k)
seborrheic keratosis
(l)



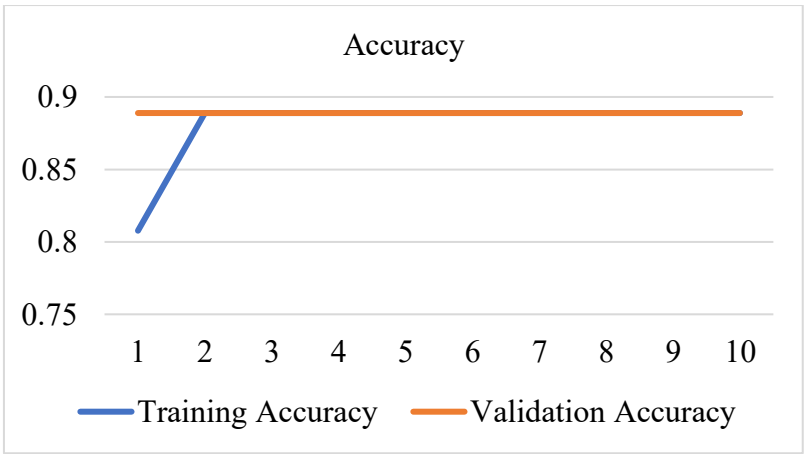


Fig.9 Training and Validation Accuracy

The training accuracy, shown in figure 9, is a measure of how the model is performing on the training data. The measure is obtained by comparing the actual labels of records in the dataset with those predicted by the model.
The validation set is a separate subset of the dataset that has not been used by the model in training. After every iteration, the model's performance is tested on the validation set and the validation accuracy recorded

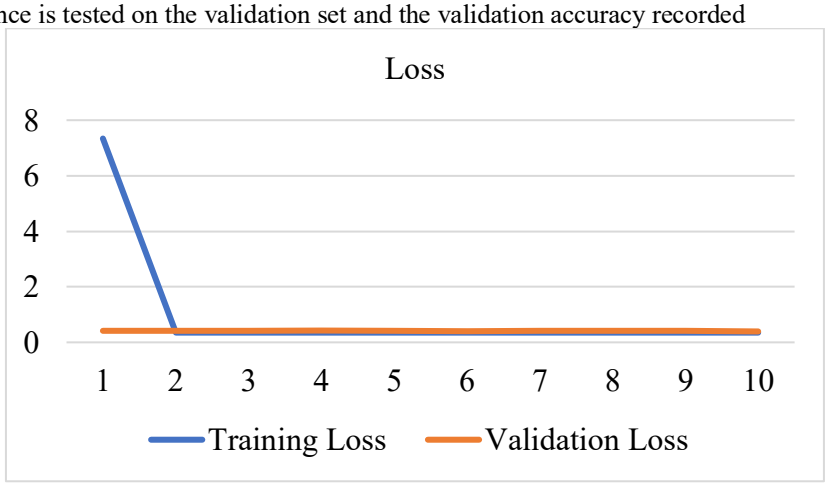


Fig. 10 Training and Validation Loss

While the loss function measures the difference between the model's predictions and the ground-truth labels, the validation loss is the result of an external dataset that the model has not seen before. It is an independent measure of how well the model is likely to perform on new data, which is similar to how well it would be able to make sense of new, unseen examples. The training and validation loss is shown in figure 10.

Table 3. Comparative Analysis

Method	**Accuracy**
AlexNet [20]	43.73%
Xception [21]	48.56%
ResNet50 [22]	70.90%
CNN (Proposed)	88.89%

The comparative analysis table 3 summarizes the comparison of several picture categorization algorithms using their accuracies. These approaches are AlexNet, Xception, ResNet50, and the recommended Convolutional Neural Network. AlexNet achieves the lowest accuracy of 43.73%; Xception achieves 48.56%, and ResNet50 is superior to both is due to its better accuracy, 70.90%. The recommended Convolutional Neural Network is the best among the mentioned approaches since it reaches 88.89%.

DISCUSSION

The experimental results shows that skin cancer detection is strongly achieved using the proposed model Efficient NetB3.The proposed model shows minimal over fitting as depicted by training and validation accuracy and loss curves. As per the classification report, 0.90 precision, 0.95 recall is achieved for benign cases whereas 0.94 precision, 0.88 recall is achieved for malignant cases thereby improving overall accuracy to 92%. A clear distinction between benign and malignant cases is shown in confusion matrix. The experimental results highlighted the benefit of deeper feature extraction method and shows that proposed EfficientNetB3 model outperform than other models (CNN, ResNet-50, EfficientNetB0). As a whole, the results are validating the effectiveness of proposed approach for accurate skin cancer detection.

CONCLUSION

The current study contributed a new approach to skin cancer diagnosis in the field of deep learning by employing the EfficientNetB3 network. This advanced model distinguished by exceptional scaling and extraordinary performance in the analysis of dermatological images signaled a substantial development in medical imaging and diagnostics. Moreover, the outcomes indicated an outstanding dispersion performance as the model effectively differentiated malignant skin lesions from benign ones with a 92% classification rate. The study pointed out the positive impacts of the deep learning algorithms in medical diagnosis and the necessity of automated and reliable skin cancer classification systems in the near future. The approach adopted minimizes the subjective assessments, improves the accuracy of the diagnostics process, and encourages the early detection and treatment of skin cancer, which is evidenced by a higher percentage of patient recovery. Future work will include augmenting the dataset, further enhancing the model through its optimization, and integrating additional diagnostic criteria to boost its predictive power.

Conflict of Interest Statement

The authors state that they are not aware of any potential conflicts of interest

REFERENCE

1. N. Hasan, A. Nadaf, M. Imran, U. Jiba, A. Sheikh, W. H. Almalki, S. S. Almujri, Y. H. Mohammed, P. Kesharwani, and F. J. Ahmad, "Skin cancer: understanding the journey of transformation from conventional to advanced treatment approaches," Molecular Cancer, vol. 22, no. 1, p. 168, 2023.

2. Verma, A. Zanoletti, K. Y. Kareem, B. Adelodun, P. Kumar, F. O. Ajibade, L. F. O. Silva, et al., "Skin protection from solar ultraviolet radiation using natural compounds: a review," Environmental Chemistry Letters, pp. 1–23, 2023.

3. L. Alfredsson, B. K. Armstrong, D. A. Butterfield, R. Chowdhury, F. R. de Gruijl, M. Feelisch, C. F. Garland, et al., "Insufficient sun exposure has become a real public health problem," Int. J. Environ. Res. Public Health, vol. 17, no. 14, p. 5014, 2020.

4. N. Nataren, M. Yamada, and T. Prow, "Molecular skin cancer diagnosis: Promise and limitations," J. Mol. Diagnostics, vol. 25, no. 1, pp. 17–35, 2023.

5. Y. G. Xu, J. L. Aylward, A. M. Swanson, V. S. Spiegelman, E. R. Vanness, J. M. C. Teng, S. N. Snow, and G. S. Wood, "Nonmelanoma skin cancers: basal cell and squamous cell carcinomas," in Abeloff's Clinical Oncology, pp. 1052–1073, Elsevier, 2020.

6. F. Brutti, F. La Rosa, L. Lazzeri, C. Benvenuti, G. Bagnoni, D. Massi, and M. Laurino, "Artificial Intelligence Algorithms for Benign vs. Malignant Dermoscopic Skin Lesion Image Classification," Bioengineering, vol. 10, no. 11, p. 1322, 2023.

7. T. Mazhar, I. Haq, A. Ditta, S. A. H. Mohsan, F. Rehman, I. Zafar, J. A. Gansau, and L. P. W. Goh, "The role of machine learning and deep learning approaches for the detection of skin cancer," Healthcare, vol. 11, no. 3, p. 415, 2023.

8. S. Q. Gilani, T. Syed, M. Umair, and O. Marques, "Skin Cancer Classification Using Deep Spiking Neural Network," J. Digital Imaging, pp. 1–11, 2023.

9. H. Nahata and S. P. Singh, "Deep learning solutions for skin cancer detection and diagnosis," in Machine Learning with Health Care Perspective: Machine Learning and Healthcare, pp. 159–182, 2020.

10. M. Dildar, S. Akram, M. Irfan, H. U. Khan, M. Ramzan, A. R. Mahmood, S. A. Alsaiari, A. H. M. Saeed, M. O. Alraddadi, and M. H. Mahnashi, "Skin cancer detection: a review using deep learning techniques," Int. J. Environ. Res. Public Health, vol. 18, no. 10, p. 5479, 2021.

11. 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 Med. Unlocked, vol. 18, p. 100282, 2020.

12. M. Strzelecki, M. Kociołek, M. Strąkowska, M. Kozłowski, A. Grzybowski, and P. M. Szczypiński, "Artificial Intelligence in the detection of skin cancer: state of the art," Clin. Dermatol., 2024.

13. M. S. Ali, M. S. Miah, J. Haque, M. M. Rahman, and M. K. Islam, "An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models," Machine Learning with Applications, vol. 5, p. 100036, 2021.
14. M. M. Mijwil, "Skin cancer disease images classification using deep learning solutions," Multimedia Tools Appl., vol. 80, no. 17, pp. 26255–26271, 2021.
15. D. Chanda, M. S. H. Onim, H. Nyeem, T. B. Ovi, and S. S. Naba, "DCENSnet: A new deep convolutional ensemble network for skin cancer classification," Biomed. Signal Process. Control, vol. 89, p. 105757, 2024.
16. J. SM, M. P, C. Aravindan, and R. Appavu, "Classification of skin cancer from dermoscopic images using deep neural network architectures," Multimedia Tools Appl., vol. 82, no. 10, pp. 15763–15778, 2023.
17. W. Gouda, N. U. Sama, G. Al-Waakid, M. Humayun, and N. Z. Jhanjhi, "Detection of skin cancer based on skin lesion images using deep learning," Healthcare, vol. 10, no. 7, p. 1183, 2022.
18. N. Khasanah and M. N. Winnarto, "Application of Deep Learning with ResNet50 for Early Detection of Melanoma Skin Cancer," J. Med. Informatics Technol., pp. 16–20, 2024.
19. V. Venugopal, N. I. Raj, M. K. Nath, and N. Stephen, "A deep neural network using modified EfficientNet for skin cancer detection in dermoscopic images," Decision Analytics J., vol. 8, p. 100278, 2023.
20. D. A. Nurlitasari, R. Y. N. Fuadah, and R. Magdalena, "Skin Cancer Classification Systems Using Convolutional Neural Network with Alexnet Architecture," in Proc. 2nd Int. Conf. Electronics, Biomed. Eng. Health Informatics: ICEBEHI 2021, Surabaya, Indonesia, Nov. 3–4, 2021, pp. 227–236, Springer Nature, 2022..