Application of Deep Convolutional Neural Network in Multiclass Skin Cancer classification using custom CNN architecture

by

Nadia Shafique
18201138
Kaynat Bint Shaheen
19101408
Zarjis Husain Sikder
19101630
Utsho Dey
19301042
Sharforaz Rahman Swacha
23141040

A final thesis report submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2023

Declaration

We are hereby confirming the following claims:

There are no prior written or published works included in the thesis unless they are properly referenced with complete and exact citations. The thesis that is being presented was a unique project that we finished when we were undergraduate students at Brac University. All key sources of support have been appropriately addressed in the thesis. The thesis doesn't contain any content which has been accepted or presented for an Interview in the desire of earning another academic certificate or degree from another university or any other institution.

Student's Full Name & Signature:

Nadia Shafique
18201138

Kaynat Bint Shaheen
19101408

Zarjis Husain Sikder
19101630

Utsho Dey
19301042

Sharforaz Rahman Swacha
23141040

Approval

The final thesis report titled "Application of Deep Convolutional Neural Network in Multiclass Skin Cancer classification using custom CNN architecture" submitted by-

1. Nadia Shafique (18201138)
2. Kaynat Bint Shaheen (19101408)
3. Zarjis Husain Sikder (19101630)
4. Utsho Dey (19301042)
5. Sharforaz Rahman Swacha (23141040)

Summer 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of BSc. in Computer Science and Engineering on [25 May 2023].

Examining Committee:

Supervisor:
(Member)

Faisal Bin Ashraf
Lecturer
Dept. of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Dewan Ziaul Karim
Lecturer
Dept. of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam,PhD
Professor
Dept. of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Ms. Sadia Hamid Kazi,PhD
Chairperson and Associate Professor
Dept. of Computer Science and Engineering
Brac University

Abstract

Skin diseases represent a significant global health concern, and prompt and precise diagnosis is necessary for efficient treatment. Convolutional Neural Networks (CNNs), in particular, have shown tremendous promise in the diagnosis of skin diseases due to their capacity for processing and learning from complex patterns in visual data. Employing 28x28 RGB images taken from the HAM10000 dataset, the purpose of this work is to develop and assess a customized CNN model created exclusively to aid in the classification of different skin conditions. This method allows the model to efficiently learn the distinctive characteristics of each type. Our model is evaluated using a number of metrics, such as accuracy, precision, recall, and F1-score. We have also compared our results to well-known pre-trained models like ResNet50 and EfficientNetB0/B2. In comparison to existing pre-trained models, our own model performs better due to its increased test accuracy, reduced test loss, and computational parameters. Additionally, it has fewer trainable parameters as well as a shorter training time per epoch, which makes it appropriate for deployment in situations with constrained computational resources. In conclusion, Our model promises to improve diagnostic accuracy, perhaps enabling earlier and more effective methods for diseases of the skin because of its higher performance and computational advantages.

Keywords: Skin cancer, CNN model, BKL, convolution layer, accuracy, recall, AUC, MEL, BCC, AKIEC, Deep Learning, ML, Dermoscopic Data, convolutional neural network, NV, DF, precision, ROC curve, F1 score, VASC.

Dedication

The research is dedicated to all the people fighting skin cancer and to their families for their constant encouragement. We respect those in the medical field who work tirelessly to find new, creative solutions. Our commitment also includes overlooked and underserved communities, which are frequently disregarded in skin cancer research and treatment.

Finally, this piece serves as an indication of the supreme value of health to individuals overcome by the speed of life. May this discovery encourage development, engender optimism, and help us move toward a world without skin cancer.

Acknowledgement

First and foremost, we would want to convey our sincerest gratitude towards the Almighty for bestowing us with direction and good luck that allowed us to thoroughly and promptly conclude our research.

Furthermore, we as a group wish to convey our heartfelt gratitude to our supervisor, Mr. Faisal Bin Ashraf, and co-supervisor, Mr. Dewan Ziaul Karim, whose invaluable counsel and strong support have helped us throughout our research journey and enabled us to successfully complete it.

Last but not least, we want to thank our parents from the bottom of our hearts. We have been shaped into the people we are now by their unwavering faith and support, which has served as our pillar. We were able to get to this stage in our educational career because of their faith in us.

List of Acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Network
DF	Dermatofibroma
ML	Machine Learning
UV	Ultraviolet
NV	Melanocytic Nevi
ReLU	Rectified Linear Unit
SVM	Support Vector Machine
VASC	Vascular Lesions
ResNet	Residual Neural Network
AUC	Area Under Curve
MEL	Melanoma
ROC	Receiver Operating Characteristic
TN	True Negative
TP	True Positive
CNN	Convolutional Neural Networks
BCC	Basal Cell Carcinoma
WHO	World Health Organisation
BKL	Benign Keratosis-like Lesions
AKIEC	Actinic Keratoses and Intraepithelial Carcinoma

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 What Is Cancer?

A category of illnesses known as cancer is defined by the unchecked growth and division of aberrant cells. Any area of the body may be impacted, and that area can take many forms. The cause of cancer is complex and can include genetic mutations, exposure to harmful substances, lifestyle habits including cigarette use, poor food choices, as well as not exercising enough, and infections such as human papillomavirus (HPV)[48].

Through the circulation or lymphatic system, cancer cells may spread beyond their initial origin to various parts of the human body. This phenomenon is referred to as metastasis. Cancer symptoms depend on the kind and phase of this disease, but typical signs include exhaustion, unusual weight loss, as well as persistent pain.

Typically, an amalgam of medical imaging, biopsies, and various other testing is performed to identify cancer. An operation treatment with chemotherapy, therapy with radiation, and immunotherapy can be considered as treatment options. The kind and stage of cancer, as well as the patient's overall condition, determine the most suitable course of treatment.

Changes in lifestyle, which involve giving up smoking, adopting a healthy diet, and partaking in regular physical activity, may help prevent cancer, as well as getting vaccinated against HPV and getting screened for cancers with early detection tests. While cancer can be a serious and life-threatening disease, many people with cancer can live long and healthy lives through early detection and effective treatment.

1.1.1 Skin Cancer

The most prevalent type of cancer is skin cancer, which develops when skin cells that include melanoma and squamous cell carcinoma proliferate rapidly. Contact with and varieties of cancerous cells tend to be the primary causes of skin cancer.

Early detection has always been important for the successful treatment of skin cancer, so it is important to have any suspicious moles or spots checked by a doctor. Regular skin self-exams and dermatologist screenings can also help with early de-

tection [40].

Prevention of skin cancer includes protecting skin from UV radiation by seeking shade, wearing protective clothing and hats, and using sunscreen with a high SPF. Avoiding tanning beds and checking moles regularly can also help reduce the possibility of skin cancer.

1.1.2 Problems of Diagnosing Skin Cancer

According to WHO data, there is going to be an increase in the number of persons suffering from skin cancer to 13.1 million by the end of 2030 [12] [23]. This constant rise poses a threat to helping stage 4 metastatic melanoma patients survive since it is nearly untreatable. It also has the highest death rates of the year, which is a scary sign that could save the lives of many patients.

Dermatologists find out if a person has skin cancer by taking pictures of them and looking at the results to see if they have cancer cells. Dermatologists distinguish between malignant as well as benign melanoma, as each consists of cancerous cells. Such a framework's drawback lies in the fact that processing several patients requires an extended period of time. Besides that, increasing the speed of detection requires a lot of workforces, thus raising costs as well. As a result, the developing detection system could speed up the identification of skin cancer, assisting dermatologists by making their jobs easier and faster [29]. However, the biggest challenge lies in proper skin lesion analysis acquired through dermoscopic images, in which the image resolution is limited by noise, shadowy areas, hairs, and reflections, which could affect the accuracy of the skin lesion analysis, as well as image capture settings along with factors like varying distances, imperfect illumination, low resolution, etc. leading to misclassification [33][22].

1.1.3 ML and its Use in Skin Cancer Detection

Usually, dermoscopy examines the skin surface. It increases the possibility of detecting skin cancer by diagnosing the lesions. But skin malignancies frequently start out as noncancerous blemishes. Although not malignant, these spots are skin anomalies that could eventually turn carcinogenic. [16]. Though dermoscopy detects melanoma, a type of skin cancer, more accurately than blind observation, the correctness of the diagnosis is dependent upon the dermatologist's skill. Moreover, when the condition first appears, it is not always easy to tell the difference between melanoma and melanocytic nevi. As a result, physicians require an autonomous detection tool. Even when a skilled dermatologist uses dermoscopy to make a diagnosis, about 75% and 84% of melanoma diagnoses are considered to be reliable [5]. So, computer-aided diagnostics are needed to make the diagnosis more accurate and faster. Some things that humans and dermoscopy can't see, like asymmetry, texture features, and color differences, can be seen by computers. One such method is utilizing machine learning (ML) which creates an AI-based model that assists in recognizing such minor traits as well as using the huge knowledge gathered through training via a vast data set to find skin cancer cells efficiently.

The CNN network is a sort of artificial intelligence that has proven successful in addressing issues with visualization identification. Additionally, it takes details out of the images, using the assistance of numerous layers like pooling, convolution, and fully connected layers. It extracts information from the images fed into it as a dataset. A significant issue arises when attempting to train a deep CNN model using little training data; because of the small amount of data, it can not classify correctly, so this problem still needs to be solved. Transfer learning and parameter fine-tuning are two strategies that have come to light for this issue[14].

So A CNN model trained with different dermoscopy images should be able to find some hidden features and patterns that tell the skin lesion type. The proposed approach is anticipated to be more accurate than any other possible trending method. So that it can help detect early skin cancer, which will positively impact the medical field.

1.2 Problem Statement

The number of persons who are diagnosed with skin cancer on a yearly basis has been steadily climbing. UV exposure is the primary cause that originates from extremely high levels of ultraviolet light reaching the surface of the planet due to the ozone layer's diminishment [4]. In its early stages, skin cancer is very treatable and can be cured with ease by using simple procedures or methods. However, advanced skin cancer is resistant to treatment with any therapy. Therefore, it is essential to diagnose and treat diseases in their earliest stages.[11]

A physician will go through a number of procedures in order to make a diagnosis of cancer of the skin in the traditional way. The initial step is visually inspecting any lesions that have been found. Dermoscopy is performed for further examination to look more closely at the pattern of skin lesions that have been found. During this step of the process, a gel will be put to the visible skin lesions, and then they will be examined using a magnifying device for enhanced clarity[3].]According to the ABCDE technique, which is used by certain specialists to identify skin lesions, a number of the factors, include the lesion's border, color, diameter, asymmetry, and rate of growth. which are taken into consideration when making a diagnosis [38]. However, the achievement or inability of the examination entirely relies on the degree of ability possessed by the dermatologists as well as the clinical facilities. In addition, the manual process that is used to examine skin cancer takes a significant amount of time, and there is a possibility that an error caused by a person could occur during the process of diagnosis.

In this context, a deep learning architecture may be of use. A crucial part involves multi-feature extraction that uses convolutional neural networks of deep learning architectures. It is necessary for the CNN architecture to receive a dataset containing raw images, which is able to recognize a variety of skin-related characteristics, including color, texture, shape, and so on. It is recommended that the dataset be divided in half so that the majority of the information can be employed for training the model as well as the rest of the information can be utilized to predict the outcome since this will result in greater accuracy. The model should be able to

identify a cancer cell from the dermoscopic images that are provided without the assistance of a medical professional at any point in the process. Figure 1 mentioned "A summarized diagram of the Proposed Solution."

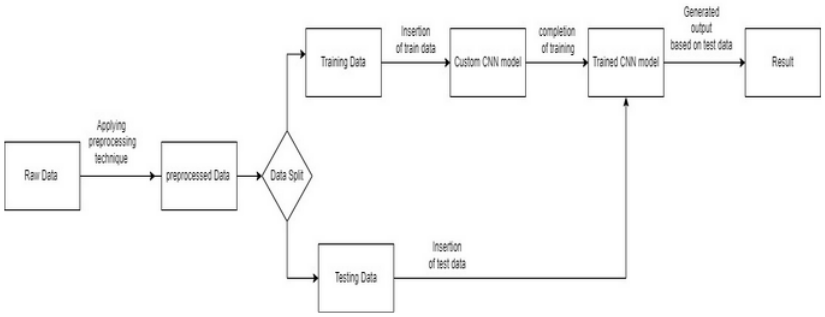

Figure 1: A summarized diagram of the Proposed Solution

1.3 Research Objectives

The goal of this study is to construct a convolutional neural network-based skin cancer spot identification structure as well as classify it into types. Basically, our objective is to explore the effectiveness of different CNN architectures and training procedures with regard to the correct identification of cancerous cells extracted from images of skin cells. In order to accomplish this, we will first collect a large dataset consisting of dermoscopic images of skin that have been labeled, and then we will use this dataset to both train and assess several CNN models. In addition, we will investigate whether or not data augmentation and the accuracy of the models' categorization can be enhanced using transfer learning. The final objective of this research is to create a reliable and efficient technique for the identification of skin cancer beforehand that can assist dermatologists in making accurate diagnoses and enhance the outcomes. This tool will be developed in order to fulfill the ultimate goal of this research.

1.4 Paper Orientation

The reader is given an overview of skin cancer in the 1st chapter. Skin cancer has been briefly discussed, along with its seriousness, likelihood of diagnosis, problem statement, and research purpose. An overview of CNN and its building blocks, including a summarized description of the Pooling Layer, Convolutional Layer, and Max pooling, as well as the last layer activation function along with fully Connected Layers, has been provided in Chapter 2. Gradient descent, overfitting, loss functions, and other concepts have all been covered. The literature review in Chapter 3 discusses the detection of skin cancer via deep learning using dermoscopic pictures and clinical data based on previously published papers. It also demonstrates why deep learning should be used by showing accuracy-based prior results. The reader is introduced to the data set used for the research in Chapter 4. This chapter has also focused on applied preprocessing techniques to the images and data analysis. Chapter 5 discusses the research model that has been proposed. The chapter also includes

a brief discussion of several models for comparison with the one being proposed. On the basis of the dataset from Chapter 4 and the models discussed in Chapter 5, chapter 6 illustrates the models' accuracy. Furthermore, the additional indicators of performance include the f1 score, precision score, and recall. The outcomes of each model have been eventually compared in Chapter 6, which, too, explains why our proposed custom CNN is better. In Chapter 7, articles and studies using the same dataset were gathered, and their models' accuracy was compared to the proposed model. This chapter highlights a contrast between the published papers and our research based on the model's accuracy. The conclusion portion of Chapter 8 is followed by Chapter 9 list of sources used in the research on this topic.

Chapter 2

Background

2.1 Convolutional Neural Network

Convolutional Neural Systems (CNNs) are a sort of deep learning architecture for artificial neural networks that directly learn from data, often used to identify objects, categories, or classes by recognizing specific patterns [1]. They resemble Multilayer Perceptrons (MLPs), with each neuron in the MLP having a single activation function that directs the weighted inputs to the output. Adding multiple hidden layers to an MLP transforms it into a deep MLP [13].

A Convolutional Neural Network (CNN) comprises three fundamental layers: the convolution layer, the pooling layer, and the fully connected layer [37]. The convolution and pooling layers generate deep features from raw data, with additional features extracted by repeated application. The extracted features are subsequently conveyed to the interconnected layers for the purpose of being mapped to the output, such as classification [37]. Figure 2 provides a visual representation of the CNN's basic structure [42]

.

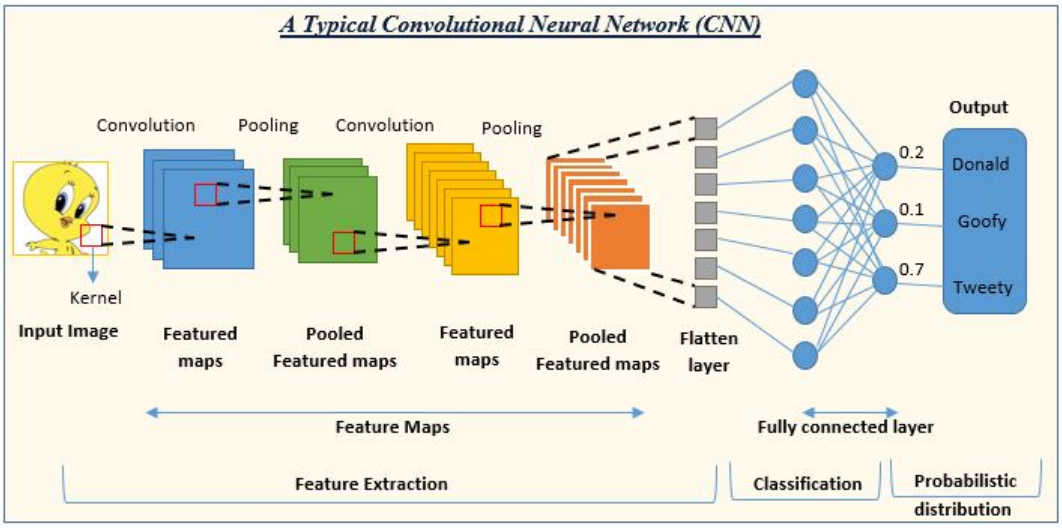

Figure 2: CNN's Fundamental Structure [42]

2.2 Building Blocks of CNN Architecture

A regular CNN has many layers for convolving and pooling, which connect to fully connected layers. A usual pattern has some connected layers and later some layers that repeat, called convolution and pooling. When input data is transformed into output, it's known as forward propagation.

2.2.1 Convolution Layer

The extraction of features is a crucial task in Convolutional Neural Networks (CNNs), which is performed by using mathematical processes such as convolution and activation functions. In CNNs, the convolution process combines two functions to create a third function by applying a kernel or filter, a small array of integers, to the tensor input data. The generation of a feature map, consisting of outputs located at dissimilar positions within the output tensor, emerges from the product of individual components of the kernel and tensor. In order to extract diverse features from the datasets, a variety of kernels are utilized on the input tensor.

The size and number of kernels used in the convolution process are essential factors. The most common kernel size is 3x3, but 5x5 and 7x7 are also frequently used. Conventionally, The operation of convolution prohibits the kernel center from coinciding with the outermost element of the input tensor, consequently leading to a loss of information within the produced feature map. To address this issue, a method known as zero padding is utilized [21]. Figure 3 illustrates the convolution process with no padding.

The stride, or the distance between two adjacent kernel locations, is another important factor in the convolution process. Sometimes, feature maps are downsampled using a larger stride. Weight sharing is a crucial aspect of the convolutional operation, which refers to the distribution of kernels across image locations. The proposed methodology obviates the necessity to frequently search for localized characteristics, as it enables other kernels to utilize the localized features identified by a given kernel as variables.

The comprehension of the advanced feature patterns facilitated by substantial feature maps may be achieved through the process of downsampling and the subsequent implementation of a pooling operation. Additionally, the efficacy of the model can potentially be augmented through the limitation of the number of parameters.

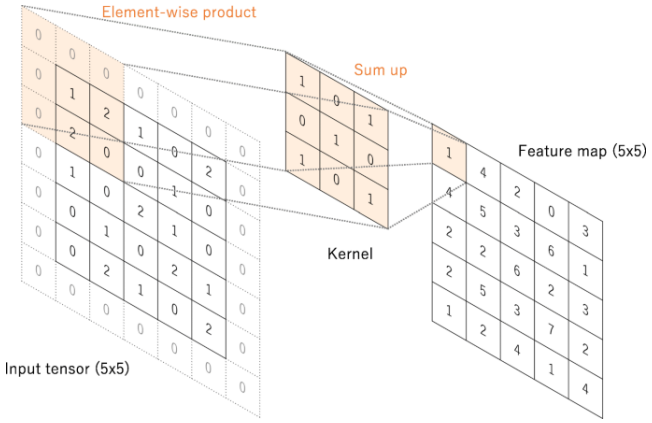

Figure 3: Convolutional Operation With Zero Padding [21]

2.2.2 Non-Linear Activation Function

After a certain process, a special kind of math is used to change the data. The ReLU is used a lot in CNNs because it's popular. Figure 4 depicts some of the common activation techniques employed in the inner layers of CNNs [21].

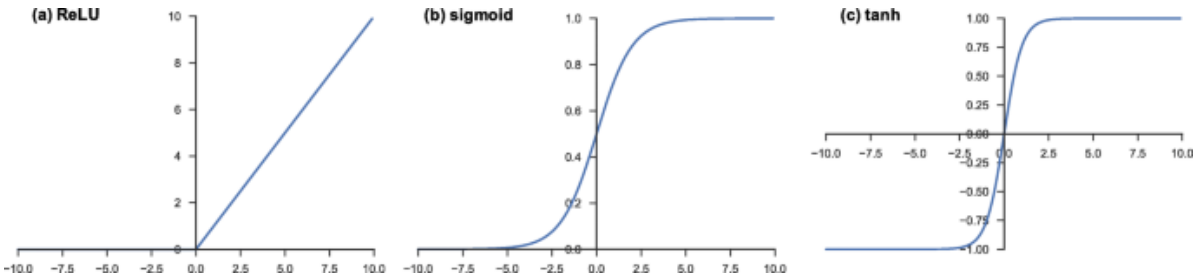

Figure 4: Frequently Used Activation Functions [21]

2.2.3 Pooling Layer and Max Pooling

In order to improve precision in the identification of slight variations and deformations, alongside minimizing the number of parameters involved in the training process, the pooling layer is employed as a means of downsampling the feature map and reducing its overall dimensions. The pooling operation most frequently employed in practice is referred to as "max pooling." This approach involves partitioning the feature map into subregions and identifying the maximum value within each subregion. The standard max-pooling filter exhibits a dimension of 2x2 and a stride of 2, resulting in the feature map experiencing downsampling of two-fold. The modus operandi of the Max Pooling Layer is depicted in Figure 5.

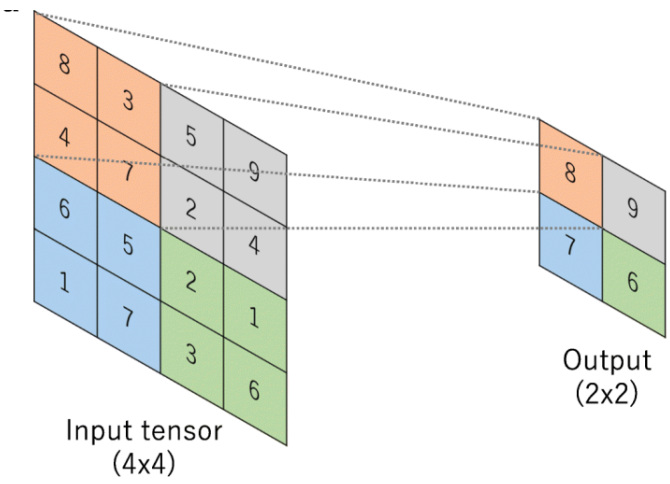

Figure 5: Operation of a Max Pooling Layer [21]

2.2.4 Fully Connected Layers

The results yielded by the convolutional and pooling layers are coupled to one or more fully connected layers, where each input is connected to each output through a trainable weight. After the convolution layer has conducted feature extraction and the pooling layer has decreased its dimensionality, the features become relayed to the fully connected layers. This further results in the generation of conclusive outputs consisting of probabilities for classification purposes. The determination of the final output nodes for the fully connected layer is contingent upon the number of classes. As previously stated, a nonlinear function, such as the rectified linear unit (ReLU), is employed subsequent to each layer.

2.2.5 Last Layer Activation Function

In the context of neural networks, the activation function employed in the final layer exhibits a considerable variation from the ones utilized in the preceding levels. This variation is determined by the specific task, as suggested by empirical evidence [21]. Table 1 in the source article provides a list of typical activation functions utilized in the final layer for various specific tasks [21].

Task	Last Layer Activation Function
Binary Classification	Sigmoid
Multiclass Single-class Classification	Sigmoid
Multiclass Multi-class Classification	Sigmoid
Regression to Continuous Values	Identity

Table 1: Common Last Layer Activation Functions for Particular Tasks [21]

2.3 Training a Network

The purpose of designing a neural network is to recognize a distinct set of convolutional kernels and fully connected layer weights that produce outputs with minimal

deviation from the labeled input dataset. The common strategy for training neural networks with hidden layers is through backpropagation, which employs the gradient descent optimization technique and a loss function to adjust the network's parameters and minimize the error.

2.3.1 Loss Function

In this study, the forward propagation system was employed to determine the cost by measuring the difference between the true affair and the affair handled by the network. The loss function utilized is determined by the type of problem being addressed. Cross-entropy is frequently utilized for multiclass bracket tasks, whereas mean square error is used for retrogression problems involving nonstop variables. In this study, double-cross entropy was used as a way to measure how well the system works.

2.3.2 Gradient Descent

The gradient descent algorithm is a widely recognized approach for optimizing neural networks by iteratively adjusting the network's modifiable parameters, comprising its weights and kernels. This process is executed with the objective of minimizing the loss function. The present research endeavor shall employ a more sophisticated approach known as "adaptive moment estimation" (Adam), which represents a significant advancement beyond traditional gradient descent techniques. Adaptive modulation of the learning rate for each parameter is achieved by utilizing measurements of how steep and how quickly the changes in the slope of something are called measurements of how steep and how quickly the changes in the slope of something are called the first and second moments of the gradients. This approach facilitates expeditious and efficacious optimization.

2.3.3 Adam Optimizer

The combination of Root Mean Square Propagation (RMSprop) and Stochastic Gradient Descent (SGD) with momentum is often employed to augment Adam's efficacy. Adam uses a moving average of the gradients rather than the SGD with momentum approach, The adaptive modification of the learning rate is contingent upon the initial and secondary moments of the gradients in a manner akin to the algorithm known as RMSprop. It is an extremely adjustable optimization strategy because the learning rates can be changed depending on the conditions selected. Adam utilizes adaptive moment estimation to approximate the first and second moments of the gradient and adjusts the learning rates of individual weights based on this estimation. Adam also keeps an average of previous gradients that decays exponentially, calculated by applying a certain formula.[21]

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

where m_t = first moment, v_t = second moment, $\beta 1$ and $\beta 2$ = hyper parameter, t=batch number

The typical values for $\beta 1$ and $\beta 2$ are 0.9 and 0.999, respectively. To update the weight, Adam employs the subsequent mathematical expression:

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

where θ = first moment, η = the learning rate, ϵ = zero-avoidance parameter = 10exp(-8)

2.4 Data and Ground Truth Labels

The efficacy of deep learning or machine learning algorithms is largely contingent upon the quality of the datasets and the accuracy of the ground truth labels. They play a crucial role in determining a model's usefulness in practical applications. It can take a long time and a lot of work to get high-quality datasets and ground truth labels. Hence, it is imperative to meticulously select appropriate datasets and ground truth labels for a specific application. There are many excellent sources of medical images, but in order to use a model for a particular subject or function, it is required to have datasets with well-defined ground truth labels. In order to ensure the model's success, additional thought must be paid to the selection of datasets and ground truth labels.[21]

Training, validation, and testing sets are the three categories into which datasets are commonly divided. The training set is utilized to educate the neural network via the process of forward propagation to determine the loss values and then using backpropagation to modify the network's parameters. The validation set is used during training to adjust the hyperparameters and choose the best model. The finished network or model is then tested on the test set, and its performance is assessed in light of modifications made using the training and validation datasets. Since the training model's hyperparameters are changed based on how well it performs with the evaluation set, it is crucial to handle the test and evaluation sets independently. The proper evaluation and optimization of deep learning and machine learning models depend on these distinctions in dataset management and evaluation. Figure 6 shows a division of the dataset for training a model.

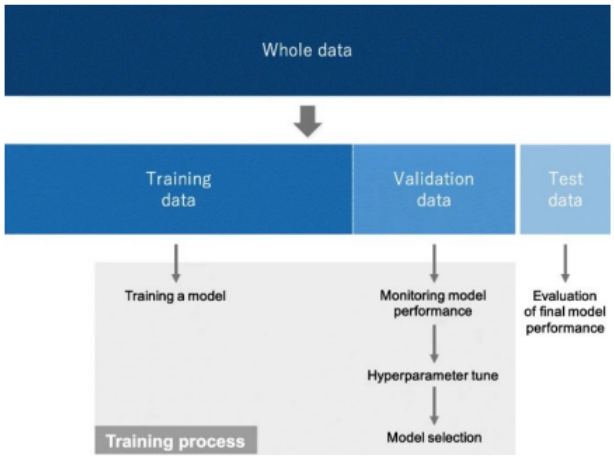

Figure 6: Division of Dataset for Training a Model [21]

2.5 Overfitting

Overfitting is when a demonstrator learns too much from the training data and struggles to perform effectively on fresh data because it hasn't accurately detected the underlying patterns or signals in the data. This indicates that it detects the insignificant information in the data rather than the crucial information. We utilize the test set to assess the model's performance since it can occasionally perform too well, which is a concern. In order to determine whether the training data is being used excessively, In empirical studies, it is common practice to assess the discrepancy between the losses and accuracies attained by the training and validation sets. A visual check is what is being done here.

Even if certain easy approaches may be used to prevent errors, having more examples to practice with is the most effective way to avoid overdoing it. When doing something a specific way is difficult, individuals employ several strategies to make it easier. Dropout, weight decay, and data augmentation are some of these strategies. Please give the text that should be simplified.

Dropout occurs when some portions of a model are switched off during training to improve the model's performance. Weight decay, also known as L2 regularization, is a penalty applied to the weights of the model. Batch normalization acts as an extra layer, ensuring that the numbers entering the following layer are all identical. This is done for a limited number of numbers at a time. This keeps overfitting from happening. It is imperative to have the capacity to regulate the process of learning in order to mitigate the number of training epochs requisite for constructing a profound neural network. Data augmentation techniques, which entail changing the training data by flipping, translating, cropping, rotating, and randomly deleting data, are one approach to accomplish this. Data augmentation allows the dataset to be increased and the network to be trained on a broader and more diversified set of data without the need for new data collection. As a consequence, the model can improve its robustness and generalization to previously encountered data so that when the model is being trained, it gets different inputs [21], [30]

2.6 Transfer Learning

Training deep learning models on huge datasets can be difficult owing to the high processing demands and storage requirements. To address this challenge, transfer learning is a commonly employed technique that involves the training of a neural network on a vast dataset, such as ImageNet, followed by the application of the pre-trained model to a particular task of concern. The key benefit of transfer learning is that characteristics learned by the network on a big dataset may be transferred and utilized for a new, related job, even if the datasets appear to be unrelated. Deep learning models are especially well-suited for transfer learning because they can detect minute changes in datasets, allowing them to be employed efficiently with small datasets for a wide range of tasks. Several pre-trained models have been built for this purpose, including AlexNet, VGG, and ResNet, and are widely utilized in a variety of applications.[21]

Fixed feature extraction is one method of using a pre-trained neural network, in which the fully connected layers of the pre-trained network are eliminated and just the convolutional and pooling layers, referred to as the convolutional base, Fixed feature extractor retained. The convolutional basis is then used to extract features from the dataset of interest before training a new classifier on top of these features. Because the pre-trained convolutional base does not need to be retrained, utilizing it as a fixed feature extractor reduces the computing cost and time necessary for training. Instead, just the new classifier must be trained on the relevant dataset, which is usually smaller than the pre-trained dataset used to train the convolutional basis. This strategy has been shown to be beneficial in a variety of applications and can increase deep learning model performance on fewer datasets, This method is depicted in Figure 7 [21].

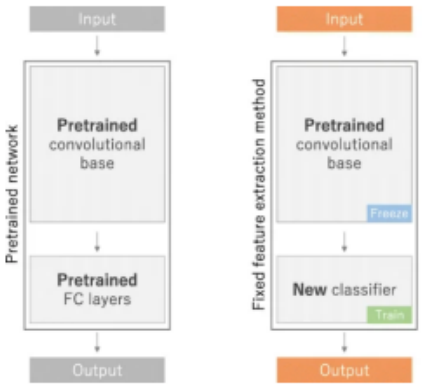

Figure 7: Fixed-Feature Extraction of Transfer Learning [21]

2.7 Why Use CNN for This Research?

Convolutional neural networks (CNNs) are a common deep learning technology used for image classification and other visual data-related tasks. Their specialty is in extracting subtle features from large amounts of visual data, and they have outperformed traditional machine learning algorithms in a variety of tasks.

There are various characteristics that make CNNs ideal for research purposes. Initially, In this research, they learned from large collections of photos, which would be difficult or time-consuming to obtain through traditional ways. Second, they have the capacity to recognize detailed features in photos that would be difficult or impossible to identify manually. Another benefit is that they can apply their expertise to fresh data successfully, which is very important in picture classification jobs. This is due to their use of convolution, a method that enables them to acquire non-linear features within the input space. SVMs, on the other hand, are limited in their ability to grasp features that are linear within the input area.

Finally, CNNs excel at generalizing to novel data, which makes them extremely adaptable to new jobs. This is because they can learn about features that stay consistent even when the input data changes little. SVMs, on the other hand, are overfitting and fail to apply their learned insights to fresh data. Overall, CNNs are the most sophisticated image classification technology, and recent research shows that they outperform traditional machine learning algorithms. CNNs are expected to become increasingly powerful and adaptable tools for image classification as they advance in development.

Chapter 3

Literature Review

3.1 Research Using Clinical Data

A new dataset containing medical views captured with smartphones and patient-specific Scientific proof. Based on a study by [28], they propose a straightforward method for combining image and clinical data features. Low-level handcrafted characteristics are used to differentiate melanomas and keratinocyte cancers. On paper[28], Shape, color, and texture are some aspects that can be extracted, these methods rely heavily on conventional computer vision algorithms. Using well-established deep learning models, researchers aggregate clinical images and data features. They present an analytical study, demonstrating the effect of victim medical info on skin cancer diagnosis. In order to determine how this information affects their predictions, they compare the performance of the models with and without this data. These features should help the model perform better at identifying both pigmented and non-pigmented lesions. Additionally, for the diagnosis of MEL, changes in lesion pattern and elevation are essential.SCC and BCC have clinical characteristics that are nearly identical. Moreover, Both bleed, hurt, itch, and affect the same age range, altitude, and geographical area. Consequently, These traits are not expected to help identify SCC from BCC. To get the total number of features T that will feed the classifier, take into account Nimg and Ncli, the number of features extracted from the pictures and clinical data, respectively, Those must combine both sets of features according to cf,

$N_{img} = \lceil \frac{N_{cli}}{1-cf-N_{cli}} \rceil$

And also To address the imbalanced labels, researchers employ a weighted loss function based on label frequency. The determined weight for a particular label I is as follows:

$w_i = \frac{N}{n_i}$

In figure 8 results of both clinical features and images are shown,

Model	ACC	BACC	P	R	F1	AUC
ResNet-50	0.788 ± 0.025	0.750 ± 0.033	0.800 ± 0.028	0.788 ± 0.025	0.790 ± 0.027	0.958 ± 0.007
ResNet101	0.757 ± 0.021	0.711 ± 0.019	0.784 ± 0.021	0.756 ± 0.021	0.766 ± 0.022	0.953 ± 0.003
GoogleNet	0.779 ± 0.011	0.714 ± 0.028	0.780 ± 0.011	0.780 ± 0.009	0.778 ± 0.007	0.948 ± 0.008
MobileNet	0.762 ± 0.040	0.717 ± 0.020	0.774 ± 0.031	0.762 ± 0.042	0.762 ± 0.037	0.948 ± 0.013
VGGNet-13	0.746 ± 0.027	0.704 ± 0.007	0.758 ± 0.013	0.748 ± 0.029	0.744 ± 0.023	0.937 ± 0.011
VGGNet-19	0.750 ± 0.013	0.709 ± 0.023	0.776 ± 0.005	0.748 ± 0.013	0.756 ± 0.010	0.946 ± 0.004
AVG	0.764 ± 0.023	0.718 ± 0.022	0.779 ± 0.018	0.764 ± 0.023	0.766 ± 0.021	0.948 ± 0.008

Figure 8: Result of both clinical features and images [28]

3.2 Research Using Dermoscopic Images

One paper[27] proposes a methodology for identifying and classifying skin lesion cell pictures of malignant and benign cells. It is CNN pre-trained. CNN architectures that have been pre-trained, such as Inception V3, VGG19, ResNet50, and SqueezeNet, extract low-level picture information. These characteristics are sent into a dense, well-connected softmax layer to classify skin cancer cell types. ISIC picture archive provided the dataset [50]. These photos showed benign and malignant skin lesions. It started with 23,000 photos. After augmentation, there are about 5,000 photos. 3.8 thousand images trained the architecture and 1.2 thousand tested it. Data augmentation has been used with several CNN architectures to improve dataset size and perform geometric changes on the picture dataset to reduce overfitting. The author experimented twice. 80% for training and 20% for testing. The figures also have been split into 70 percent and 30 percent and went through experiments. Inception V3 has 93.10% accuracy for 80-20 data split, VGG19 94.05%, SqueezeNet 95.85%, and ResNet50 97.00. Finally, his framework generated 99.20 accuracies. For the 70-30 data split, the suggested framework was the most accurate. The proposed framework is a Deep IoHT framework based on transfer learning for skin cancer detection and classification.

In another article[15], Mendes and da Silva used the pre-trained ResNet-152 deep learning architecture in 2017 to identify 12 different types of skin lesions. Additionally, data augmentation and transformation techniques have been applied. On the test set, the network had a melanoma accuracy of 96% and a BCC accuracy of 91%. The length of the training, nevertheless, was excessive.

Nasir M [19] (2018) developed a feature selection-based approach to categorize skin lesions in the PH2 dataset using Support Vector Machine. The usage of a hybrid segmentation method has taken place. All the components of color, form, and texture have been joined together. A classification accuracy of 97.5% was attained by using the proposed model.

In the paper [26], python, Keras, and Tensorflow will create the CNN classification model. The model is built and verified using several network architectures by varying the types of layers used to set the network. Convolutional layers, dropout layers, pooling layers, and dense layers are examples. The International Skin Imaging Collaboration archival dataset will test and train the model (ISIC). This project used two datasets. The first dataset of 10,015 dermoscopic 600×450 pixel pictures. 7 skin diseases are included. Another dataset has 25,333 pictures. Each is 1022×767

pixels, dermatoscopic. 8 skin disorders are in the dataset. The combined dataset had 35,348 raw pictures. Data augmentation has been increasing testing accuracy and model robustness to new combined data. Data augmentation and picture normalization have been applied which accelerates network convergence. Scaling all pixel values simplifies computer computations. Transfer learning procedures will help the model converge early. The dropout approach reduces overfitting and improves theoretical error in all DNNs. As accuracy cannot assist in evaluating the whole model's efficiency so for each skin lesion class, they evaluate Accuracy, Precision, Recall, F1 Score, and Support.

ResNet50 Accuracy 0.85, Precision 0.86, Recall 0.85, F1 score 0.85, Support 6944. MobileNet Accuracy 0.85, Precision 0.86, Recall 0.85, F1 score 0.85, Support 6944. VGG16 Accuracy, Precision, Recall, F1 score 0.87. 6944. Inception V3 total Accuracy, Precision, Recall, F1 score 0.90. 6944. For InceptionResnet V2, Accuracy is 0.91, Precision is 0.91, Recall is 0.91, F1 score is 0.91, and Support is 6944. InceptionResnet averaged 91% accuracy.

3.3 Research Using Multiclass Skin Cancer Classification

A study done by [35] showed an ensemble classification model using deep learning which was designed with the intention of classifying skin cancer into many categories. The study made use of the 25,331 dermoscopy photos from eight categories that are part of the ISIC 2019 repository collection. The authors randomly selected 1500 images from each of the NV, BCC, Melanoma, and BKL classes, and included all available images from the remaining four categories in the dataset. The final trained dataset had 5690 photos, whereas the test dataset was produced by taking 25% of the whole dataset, yielding 1797 images. In total, the dataset included 7487 dermoscopic images.

The study compared the accuracy of individual deep learning models such as ResNet, InceptionV3, DenseNet, InceptionResNetV2, and VGG-19, which achieved accuracies of 72%, 91%, 91.4%, 91.7%, and 91.8%, respectively, against the precision of suggested ensemble models based on majority and weighted majority voting, which achieved accuracies of 98% and 98.6%, respectively. While the accuracy rate was successful, the study did not perform extensive pre-processing on the images or carry out lesion segmentation, which could have improved the reliability and generalization of the results.

In their [34] paper, the categorization of multiclass skin lesions using deep learning which is a new framework is provided. Data augmentation, feature extraction using deep learning models, feature integration, component selection, and classification are all processes in the system. The HAM10000 dataset has been implemented in experimentation, and both augmented and non-augmented datasets are evaluated for accuracy. The accuracy achieved for the enhanced dataset was 95.0% for feature fusion and 91.7% for feature selection, compared to the non-augmented dataset accuracy of 64.36% for ResNet-50 and 49.98% for ResNet-101.

The paper [25] describes a multi-class multi-level (MCML) approach to categorizing skin diseases. Two approaches are compared: traditional machine learning and deep learning. Additionally, multi-class single-level versus multi-class multi-level approaches are evaluated. The dataset used in the study is obtained from multiple sources, including the ISIC and PH2 datasets, and consists of four image classes. Randomly selected 918 images are used for each class, with a 70:30 split between training and testing data, and 58 photos from each class set aside for cross-validation. Traditional machine learning involves preprocessing techniques such as image scaling, hair removal, and black frame removal to prepare the image for segmentation. ROI is enhanced with segmentation, and features are extracted from the images. As opposed to that, the deep learning approach involves combining a CNN architecture with AlexNet and transfer learning techniques for image classification.

Accuracy:

Approach	Traditional ML Approach	Deep Learning Approach
MCSL	61.64%	96.03%
MCML	63.79%	96.47%

Table 2: MCML classification algorithm performs better in both cases.

MCML classification algorithm performs better in both cases.

3.4 Major Findings and Scope of Research in skin cancer

Some are the major findings and scope for skin cancer research that we have found from the above literature review are,

1. A massive proportion of skin cancer is unavoidable, and if detected early, it is generally treatable. Yet, it's crucial to use caution when outdoors because the majority of skin malignancies are caused by sun exposure. The risk of developing skin cancer and early aging are both increased by frequent sunlight exposure.

2. Dermoscopic images often provide more precise and specific information than clinical images when it comes to identifying skin cancer cells. Dermoscopic photos are captured using a dermoscopy, a specialized magnifying instrument, while clinical images are often described as skin photographs acquired with a standard digital camera or smartphone. Dermoscopic images allow for a much higher level of magnification and give a closer look at the skin's surface. They may be able to reveal details and patterns that the naked eye or clinical imaging would miss. Hence, combining it with CNN dermoscopic images typically provides more precise and thorough information

3. Traditional machine learning (ML) and deep learning approaches give optimistic outcomes on skin cancer cell detection, but deep learning approaches

have been highly effective in image-based tasks such as skin cancer detection. Deep learning approaches use neural networks to disable feature extraction and acquire features directly from the input data provided. This enables more efficient and accurate feature learning, potentially leading to improved performance. Traditional ML approaches, on the other hand, typically require the choosing and retrieval of hand-crafted features from input data, such as hue, texture, and shape features. These characteristics are then fed into a classification model. The model's precision is heavily reliant on the quality of the hand-crafted features, which might be laborious and susceptible to mistakes to generate.

4. One of the most frequently used deep learning architectures for the diagnosis of skin cancer is CNN. It has been shown to perform classification results on several skin cancer datasets successfully, which has contributed to its popularity.

5. Binary classification of skin lesions has been investigated more thoroughly, Nevertheless, compared to binary classification, there are still fewer researchers that have looked at the multiclass classification of skin lesions. This is most likely due to the fact that binary classification is a simpler issue to address, as well as the fact that many datasets for skin lesion classification are binary in nature. On the other hand, despite the amount of research that has been done regarding the multiclass classification problem, there is still room for further development in this area.

Chapter 4

The Dataset

4.1 Data Collection

For the purpose of this research, the dataset that has been used is Skin Cancer MNIST: HAM10000, which was released in 2018 as part of ISIC. There are approximately 10,015 images of skin lesions with pigment in them. It is freely accessible via the Harvard Dataverse and can be accessed via the Kaggle API.[20]

This dataset includes seven categories of skin lesion types. which consist of representative instances from each of the primary pigmented lesion diagnosis classes. They are dermatofibroma (df), benign keratosis-like lesions (bkl), melanoma (mel), basal cell carcinoma (bcc), melanocytic nevi (nv), vascular lesions (vasc), Actinic keratoses and Bowen's disease (akiec). The Fig.9 here, several examples of the visualizations using this dataset are provided, which highlights the different forms of skin lesions.

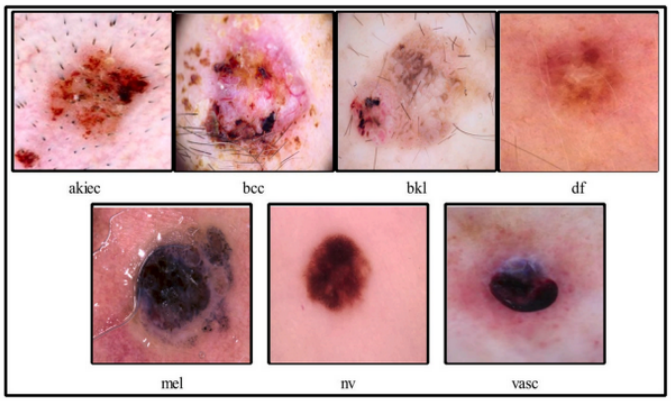

Figure 9: Visual representation of of the HAM10000 dataset's skin lesions [43]

We acquired the above-mentioned dataset from Kaggle.com using the Kaggle API to have access to it. The dataset was then cleaned and processed, with any damaged or missing photos being removed. Additionally, We created sets for training and validation from the dataset using an 80:20 split.
Original citation of the Skin Cancer MNIST: HAM10000 dataset can be found in [43] and [20].

4.2 Data Analysis

In this study, we performed multiclass skin cancer detection with the Skin Cancer MNIST: the HAM10000- dataset using a deep CNN. We began by loading the dataset into a Pandas DataFrame and mapping the integer class labels to their corresponding diagnostic categories.
There are a total of seven categories of skin cancer images. They are -

Melanocytic Nevi (nv): When melanocytic cells are seen in epidermal nests, the condition is referred to as a melanocytic neoplasm and just a minimum of 3 melanocytic cells that are in direct contact with each other can be considered such which is inside the dermis or other tissues. The kind of cancer known as melanocytic neoplasms may either be benign (nevi) or just malignant (melanoma). Nevi are considered to be harmless (benign) clonal proliferations involving melanocytic cells. The term comes from a Latin origin referring to "birthmark," signifying that nevi developed at birth. Despite the underlying meaning, however, a lot of nevi appear after birth. These acquired nevi, interestingly, have both environmental and genetic risk factors associated with malignant melanoma. [17]

Melanoma (mel): Melanoma is a kind of cancer that harms the skin's pigment-producing cells. Melanoma is usually visible as a dark brown or black, shapeless mark that differs from a person's other moles or freckles. It may be a combination of dark and light colors with sharp, uneven boundaries. The majority of skin cancer deaths are caused by melanoma [45]. If not treated right, it can become very dangerous. Among the other skin cancers, it is the type of malignancy that develops most frequently.

Vascular Lesions (vasc): Arguably, the most popular and frequently carried out cutaneous laser procedures is the treatment of developed vascular lesions. Telangiectases, pyogenic, cherry angiomas, spider angiomas, granuloma, venous lake, and leg vein abnormalities are examples of acquired vascular lesions.[6]

Benign Keratosis-like Lesions (bkl): This classification includes benign skin lesions that look like keratoses. a skin disease characterized by thicker, scaly patches. These lesions are usually non-cancerous and can comprise seborrheic keratosis or actinic keratosis.[47]

Actinic Keratoses and Intraepithelial Carcinoma (akiec): Actinic keratoses are skin neoplasms characterized by abnormalities in chromosomes that typically affect surfaces where the sun is exposed to the portion of the skin. Although artificial UV light exposure, x-irradiation, and long-term solar radiation may additionally result in these premalignant lesions, long-term solar radiation is frequently the root cause. Actinic keratoses can range in size from tiny papules of 1 to 2 mm to massive plaques. Typically flesh-colored, erythematous, or highly pigmented, they may have a hyperkeratotic surface.[8]

Basal Cell Carcinoma (bcc): In white people, the most widespread form of cancer is basal cell carcinoma. Its prevalence has been increasing by up to 10% per year

worldwide. Although death is low because basal cell carcinoma rarely metastases, this cancer produces significant morbidity and lays a huge impact on the global healthcare system. The main cause of basal cell carcinoma is UV light exposure.[1]

Dermatofibroma (df): Dermatofibromas are often thought to be normal lesions. Furthermore, they are virtually always harmless, asymptomatic, and are usually best left untreated. Reactive fibrous lesions with neural or smooth muscle features are believed to be dermatofibroma variations. They show uncommon clinical and, more typically, histologic features in a limited number of patients, which can create differential diagnosis challenges. [2]

In figure 10, there are a total of seven categories of skin lesion images from the HAM10000 dataset that are displayed below.

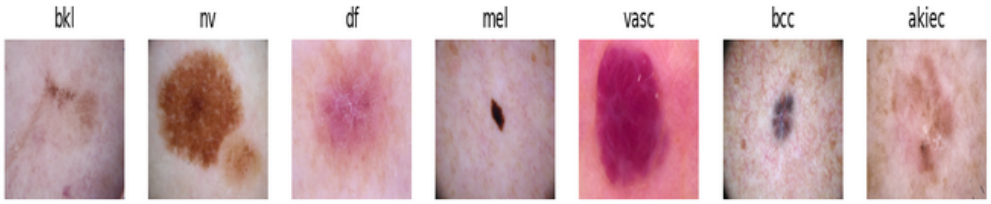

Figure 10: Seven types of skin lesion images of HAM10000 dataset [43]

4.2.1 Frequency Distribution of Data

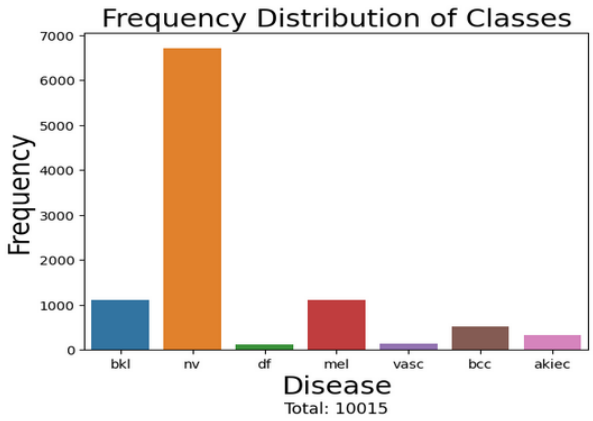

Figure 11: Graphical representation of frequency distribution of classes

Figure 11 shows the frequency distribution of different categories ('dx') in the dataset. It utilizes a count plot to display the number of occurrences of each disease class. The disease categories are represented by the x-axis, and the frequency for each category is displayed by the y-axis. After interpreting the graph out of the total 10015 images, we can see that "nv" is the most frequent category where the count is below 7000, followed by "mel" where the count is above 1000. Similarly, the "bkl" count is above 1000 but less than that of "mel." The other categories have relatively smaller counts, like "bcc" having images above 500, "akiec" having

images less than 500, "vasc" having images less than half of the "akiec" category, and lastly, "df" containing images of the least of the overall dataset which is less than 150. Overall, this frequency distribution provides insight into the prevalence of different categories of skin cancer within the dataset, with non-cancerous lesions "nv" being the most common and the least common being the "df." The title of the plot is "Frequency Distribution of Classes."

In conclusion, it is affirmative that there is an imbalance in the quantity of skin cancer samples in the dataset. The amount of samples that are more prevalent is 'nv,' and the rest categories contain samples that are below average.

4.2.2 Distribution of Disease over Gender

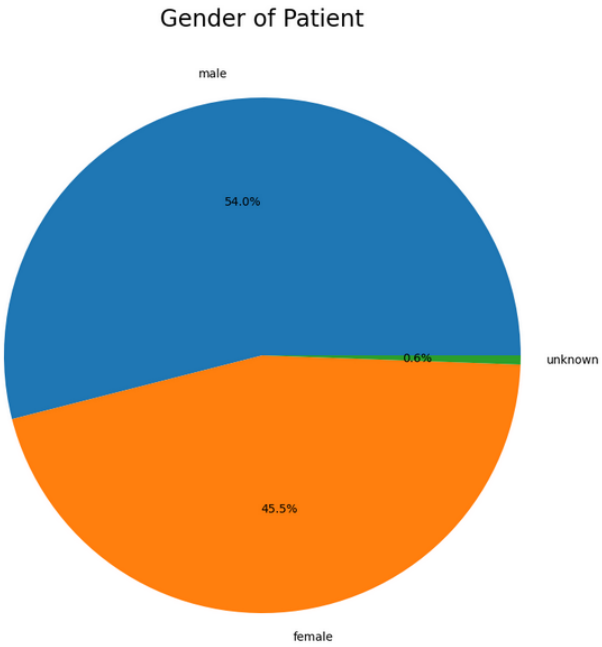

Figure 12: Pie Chart for the gender of patients

Figure 12 represents the distribution of disease based on gender. This pie chart illustrates the breakdown of skin cancer cases among patients of various genders (sex). The labels on the chart indicate the gender categories (e.g., male, female), and the matplotlib's autopct argument formats the percentages displayed on the chart. The largest portion, constituting more than 50% of the total cases, belongs to male patients. Close behind are the female patients, who account for less than 50% of the cases representing a significant portion of those who are affected by skin cancer. The remaining represents cases where the gender information is unknown which forms a relatively smaller segment compared to the other two. The title of the plot is "Gender of Patient."

Apparently from the above pie chart, skin cancer seems to occur more in the male gender than the female gender and the unknown gender. Overall, life risk is crucial for both genders.

4.2.3 Histogram of Age of Patients

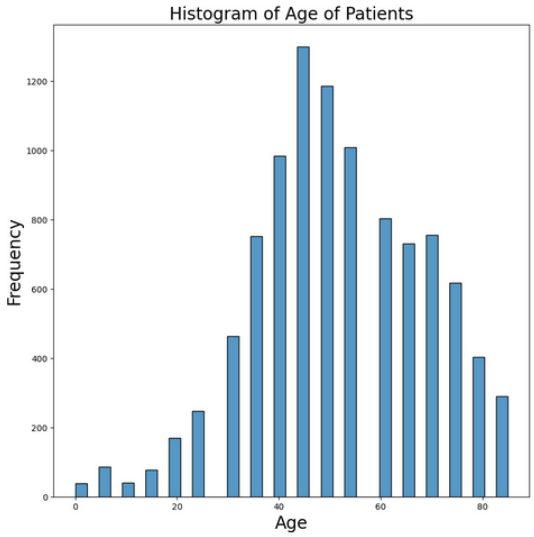

Figure 13: Histogram for the age of patients

Figure 13 displays a histogram of the age distribution of the number of patients in the dataset. It uses a histogram plot to represent the frequency of different age groups. Age values are represented on the x-axis, and the proportion of each age category is shown on the y-axis. After analyzing the histogram, we see, the number of patients aged between 0 to 20 years old are seen to be less likely to have cancer. For those aged from 20 to 40 years old, the count of patients becomes a higher gradient. Patients between the ages of 40 and 55 make up the majority of skin cancer patients overall. For individuals aged between 60 to 85 years old, the amount fluctuates, and there is a gradual downfall in the number of patients. The title of the plot is "Histogram of Age of Patients."

Overall, it can be said that the number of patients that are highly concentrated range between 40 to 55 years, and the least number of patients range between both the younger and the older ones.

4.3 Data Preprocessing for Use in Models

In order to prevent bias or differences in the models' predictions caused by the data format, images must be preprocessed before they can be implemented in the CNN models. In these datasets, we implemented data augmentation, data normalization, and data segmentation.

4.3.1 Data Augmentation

Before augmenting the data, we found that the data were highly imbalanced. Here in table 3, the aggregate amount of images across the dataset's seven distinct groups. It shows the representation of highly imbalanced data as each category fails to include the exact same subtotal number of images. This imbalanced feature will not

Name	Before Augmentation Image number
Melanoma (mel)	1113
Benign Keratosis-like Lesions (bkl)	1099
Vascular Lesions (vasc)	142
Dermatofibroma (df)	115
Basal Cell Carcinoma (bcc)	514
Melanocytic Nevi (nv)	6705
Actinic Keratoses (akiec)	327
Total	10015

Table 3: Count of images in the sample across the entire seven categories

provide us with the best accuracy as it will have biases in its prediction. So, we ran image augmentation to resolve this issue by creating a balanced dataset.

Here first, we implemented a data augmentation technique to preprocess the data. By applying different types of image transformation methods, such as flipping, rotation, and mirroring, to the existing images of the classes that are needed to be augmented, we performed the data augmentation.

The classes that needed to be augmented are bkl, mel, akiec, bcc, df, vasc. In nv class, it has the highest number of images. We wanted other classes to have the same number of images, which is 6705 images for each class. Python Imaging Library has been used to perform these transformations.

To implement data augmentation, at first, we got the subdirectories of the parent directory, then iterated over each subdirectory and got the full path to the subdirectory. After that, we checked the number of files in the subdirectory. If the number of files is less than 6705 in the subdirectory, We randomly choose an image from that subdirectory and randomly perform any of the transformations (flipping, rotating, mirroring) and store them in the same subdirectory. That is how the number of images increases for the other classes. The loop goes on until each subdirectory has 6705 images.

So after applying the augmentation, here is the visual representation of our dataset. In Figure 14, After augmentation, the total amount of images for every group is presented.

Now the collection is balanced, and there are 46935 total images.

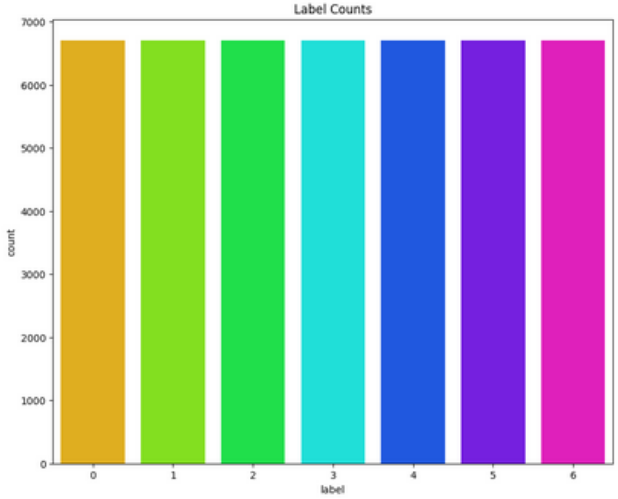

Figure 14: Number of images after having augmentation for each group

4.3.2 Data Normalization

An essential stage in multiclass picture classification tasks is data normalization. It involves converting picture pixel values to an average range. To ensure that the input data is uniform and comparable across various images, data normalization aims to scale down all the image attributes to a single scale. Here are several reasons why multiclass picture classification requires data normalization.:

1. **Reduces the impact of varying scales:** Different sizes and pixel value ranges are possible for images. For instance, one image can contain pixel values between 0 and 255, whereas another might have values between 0 and 1. Due to the model's tendency to give variables with larger scales a higher priority, these scale differences may provide problems throughout the training phase. By bringing all of the pixel values into a consistent range, which might be 0 to 1 or -1 to 1, normalizing the data solves this problem.

2. **Enhances model convergence:** In order to help optimization techniques during model training to converge, data normalization is used. It makes the model's parameter updating process more effective if the input data have similar scales. This is especially critical for deep learning models which depend on gradient-based optimization methods. Lack of normalization of the data can cause the model's convergence to be delayed or even unachievable.

3. **Prevents dominance of certain features:**Different picture attributes may have various ranges and distributions when used in image categorization. If features with wider ranges aren't normalized, they might take control of learning and mask the impact of other features. By normalizing the data, each feature contributes fairly to the model's learning process, avoiding biases in favor of particular features.

4. **Increases model generalization:** Normalization enhances the model's generalization capacity by lowering the impact of unimportant variations of the input data. As a result, the model can then concentrate on the crucial patterns and structures in the images instead of being distracted by non-essential pixel value variations. This enhances the model's capability to correctly categorize images on unseen data.

In our dataset, the pixel value of the images was not in the same range. The range difference is very high. As a result, the value with a higher range may dominate certain features. To tackle this problem, we implemented normalization on our dataset; as a result, the pixel values converged between 0 to 1.

Chapter 5

The Models

Upon executing the preprocessing procedures delineated in Chapter 4, in order to establish a connection between the dataset's features, the dataset has been prepared for ingestion via Convolutional Neural Network. In order to both train and test purposes, the dataset was partitioned at an 80:20 ratio. where 80% of the data was utilized for training as well as 20% of them for testing. Additionally, 20% of the whole training data was set aside for validation, with the remaining data being used to train the model.

Th`train_test_split()` technique from SciKit-Learn was used to create this splitting.[53], with parameter shuffle=True ensuring a diverse class distribution within each data subset. This stratification allows for the assessment of potential model overfitting. Figure 15 illustrates the data segmentation, implemented to conduct models' testing and training.

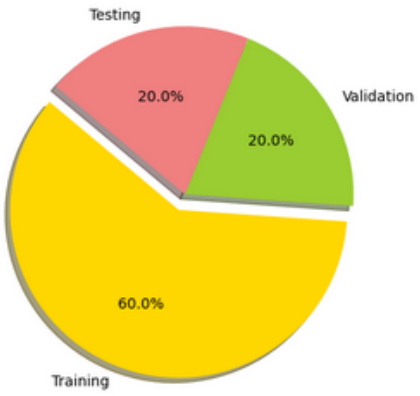

Figure 15: Splitting of the dataset for Model's Training and Testing

5.1 Transfer Learning Models

In a particular form of machine learning technique termed transfer learning, which is a particular model first pre-trained on a sizable dataset., then gets fine-tuned on a smaller and similar types dataset and performs that specific task. The perfect

scenario in machine learning entails that there would be plenty of labeled training samples with an identical distribution as the test sample. However, in many cases, gathering enough training data is prohibitively costly, time-consuming, or even unachievable. As a result, the resulting traditional models are frequently unsatisfactory. Transfer learning, which particularly concentrates on knowledge transfer among domains, is a favorable machine learning system for overcoming the aforementioned challenge.[41] In this study the pre-trained models we have employed, the ImageNet dataset was used to train those transfer learning models. The dataset is an image database arranged based on the WordNet hierarchy. [9] Several models were evaluated with the TensorFlow library, and they were discussed with their outcomes.

5.1.1 Resnet50 Model

In order to address the issue of vanishing gradients which may happen in quite deep neural networks, the ResNet50 model was developed. This problem refers to the situation where the gradients become extremely small, making it difficult to update the weights during training. This can lead to slow convergence and reduced effectiveness of the network. In deep neural networks, stacking more layers on top of an existing network can increase inaccuracy. However, ResNet mitigates this by implementing identity mappings through shortcut connections. The advantage of this shortcut identity mapping is that it does not add extra parameters to the model, reducing computational time.[51]

The architecture of ResNet50 consists of:

1. One layer is produced initially by a convolution layer with a kernel dimension of 7 x 7 and 64 different kernels, all of which have a stride size of 2.

2. Following that, a max pooling layer which contains stride size 2.

3. After that, the next convolution has three kernels: an initial 1 x 1,64 kernel, then an additional 3 x 3,64 kernel, with another 1 x 1,256 kernel. This stage includes nine layers, each of which is repeated three times.

4. Following that, a 1 x 1,128 and a 3 x 3,128 are the first two kernels that are visible. Third, we have another kernel of 1x1, 512. The whole thing went on four times. During this process, an aggregate of 12 layers is created.

5. A kernel of size 1 x 1,256 is the next kernel, which is followed by two more kernels of size 3 x 3,256 and size 1x1,1024, which are then repeated six more times making an aggregate of 18 layers.

6. The following layer consisted of a 1 x 1,512 followed by two further layers of 3 x 3,512 then 1 x 1,2048 kernels. This process was done repeatedly for an aggregate of three times making it a total of nine layers.

7. Next, we execution of an average pool, and last, create a fully connected layer with a thousand nodes using a softmax function, which produces one layer.[51]

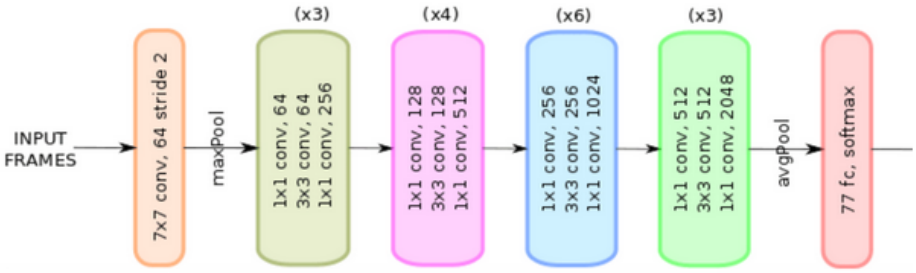

Figure 16: ResNet-50 neural network architecture [52]

Here Figure 16 shows the visual representation of the ResNet-50 neural network architecture.
With the exception of its top layers, we used the pre-trained ResNet50 model in this research. In addition, we added layers for flattening and pooling, plus an entirely connected layer with 128 hidden units activated by ReLU.The output of the model was a dense layer containing 7 units as well as a softmax activation function. The layers of the base model were frozen to preserve their learned information, and the compilation of the model was done through the Adam optimizer, where we utilized sparse categorical cross-entropy to be the loss function and accuracy as the metric. The training was done using a batch size of 128 with a validation split of 0.25 over 100 epochs. On the basis of validation accuracy, early stopping was used, and based on validation loss, the best model was saved. Further analysis was conducted using TensorBoard logs. A solid model for classifying 7 different classes was produced by this architecture and training method.

5.1.2 EfficientNetB2 Model

The EfficientNet architecture, including the EfficientNetB2 model, was designed to scale up convolutional neural networks in a more structured manner. Unlike traditional approaches that scale up network depth, width, or resolution separately, EfficientNet scales up these three dimensions together using a compound coefficient. This approach allows for a better balance between the different resources and can lead to improved performance.

The EfficientNetB2 model is a specific version of the EfficientNet model. It is larger than EfficientNetB1 but smaller than EfficientNetB3 according to the number of parameters and the computational expenses. The configuration of the model is built upon a mobile inverted bottleneck convolution (MBConv), similar to the one used in MobileNetV2, but with several improvements.[46]

An overview of EfficientNetB2's structure is provided below:

1. Initially, the model has a 3x3 convolution layer.
2. This is followed by several MBConv layers. Each MBConv layer contains an expansion phase (using 1x1 convolutions), a depthwise convolution phase (using 3x3 convolutions), and a projection phase (also using 1x1 convolutions).

Additionally, there is a skip connection that adds the input of the block to its output, similar to ResNet.

3. Each MBConv layer has a distinct variety of filters and input sizes, which are based on the scaling of the main EfficientNet model and are particular to that layer.

4. After the MBConv layers, there comes a 1x1 convolutional layer, succeeded by a global average pooling layer.

5. For classification, the last layer consists of a fully connected layer that utilizes a softmax activation function.[45]

In the study, the researchers utilized the EfficientNetB2 model and incorporated a subsequent layer of global spatial average pooling. Using ReLU activation, it was followed by a layer that was fully connected and contained 128 hidden units. The last layer was made up of 7 units with softmax activation, each catering to a different class. The foundation model layers were frozen during training, and the model included numerous trainable parameters.

The evaluation of the model was based on accuracy being the performance metric, while the loss function employed was sparse categorical cross-entropy. The Adam optimizer was employed. 100 epochs of training were performed using a batch size of 128 and a validation split of 25%. Seeking the best average outcome, the procedure was performed several times.

5.1.3 EfficientNetB0 Model

"EfficientNetB0 is the baseline model in the EfficientNet series, aimed at achieving the appropriate balance between the network's resolution, breadth, and depth-through a method known as compound scaling. Unlike traditional scaling methods that predominantly focus on one dimension at a time, given a constant ratio, compound scaling scales all dimensions evenly. However, achieving this balance requires thorough manual tuning, and if not correctly balanced, it can often lead to suboptimal accuracy and efficiency.
Key features of the EfficientNetB0 architecture include[49] :

1. It employs a Convolutional Neural Network (CNN) structure that takes into account resources like memory and FLOPs and uses compound scaling to boost its efficacy in an existent ConvNet.

2. Its architecture is built upon the mobile inverted bottleneck convolution (MBConv), similar to the one used in MobileNetV2, but with several improvements.

3. The specific number of layers and their configurations in EfficientNetB0 are determined by the outcome of the compound scaling process.

Figure 17above illustrates the structure of EfficientNet-B0.

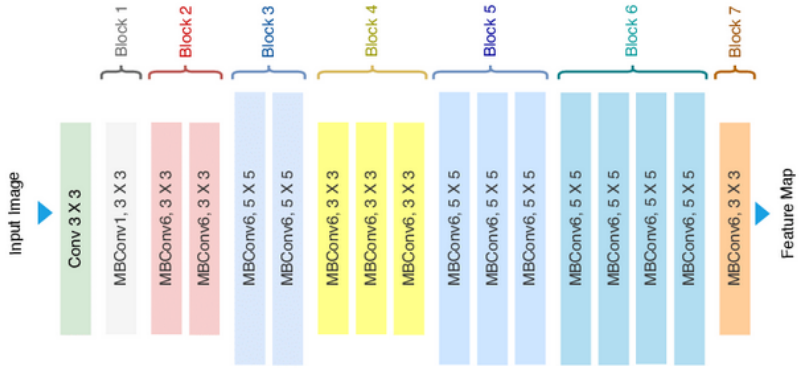

Figure 17: EfficientNet-B0 architecture [44]

The EfficientNetB0 model, which was pre-trained on the ImageNet dataset, served as the foundation for this project. In place of the original top layer, the following layers were employed: a GlobalAveragePooling2D layer, a fully connected layer where 128 units were hidden in ReLU activation, and a softmax output layer tailored for seven classes.

The layers of the base model were frozen, resulting in a mixture of trainable and non-trainable parameters. The model utilized the Adam optimizer, employed the sparse categorical cross-entropy loss function, and evaluated performance based on accuracy. With a batch size of 128 and a train-validation split of 75:25, training was conducted over 100 epochs.

5.2 Custom CNN Model

The primary objective of this study was to develop an exclusive CNN model. The objective was to develop quick, simple models that needed fewer training parameters but were more accurate than the pre-trained models that were already available to be utilized in this work.

The implemented Convolutional Neural Network (CNN) model adopts the following architecture:

In the developed Convolutional Neural Network (CNN) model, the following architecture is implemented:

1. An input convolution layer employing 16 kernels of size 3x3, supplemented with ReLU activation function, and padding set to 'same'. This layer is designed to take input images of dimensions 28x28x3.

2. Another convolution layer is set up with 32 kernels of size 3x3, featuring the ReLU activation function. Notably, this layer doesn't include padding.

3. , In order to reduce the spatial dimensions of the output from the preceding layer, a max pooling layer with a 2x2 kernel size, is then implemented.

4. The architecture then incorporates a convolution layer with 32 3x3 kernels, accompanied by ReLU activation and padding set to 'same'.

5. Following this, a convolution layer with 64 3x3 kernels is included, offering ReLU activation. This layer operates without padding.

6. A max pooling layer with a 2x2 kernel size is introduced, applying 'same' padding, which prevents the dimensions from being rounded down during the pooling process.

7. Another convolution layer with 128 3x3 kernels is added, which uses ReLU activation and padding set to 'same'.

8. The model includes an additional max pooling layer with a 2x2 kernel size, applying 'same' padding.

9. Subsequently, a flattened layer is introduced, which reshapes the 3D outputs of the preceding layer into a 1D vector.

10. Next, the model incorporates a fully connected (Dense) layer comprising 64 nodes, which applies the ReLU activation function.

11. The following layer has 32 nodes and is completely linked (dense), also features ReLU activation

12. Finally, the design is completed with a dense output layer with seven nodes as well as a softmax activation function hereby facilitating multi-class classification.

In figure 18, model architecture 1 is displayed. And in figure 19, the model architecture 2 is displayed:

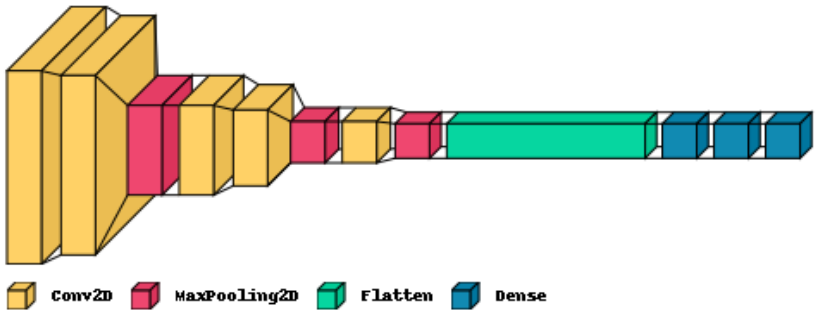

Figure 18: Model Architecture 1

With only 182,791 trainable parameters, the customized Convolutional Neural Network (CNN) model exhibits effectiveness and efficiency. Because of the decreased architectural and computational complexity, results can be obtained more quickly with fewer resources being used.

The model was built using the Adam optimizer, with accuracy serving as the performance metric and "sparse categorical cross-entropy," making it the loss function. The training of the model spanned across 100 epochs, with the training phase effectively completing at epoch 49 due to early stopping because of 10 successive epochs

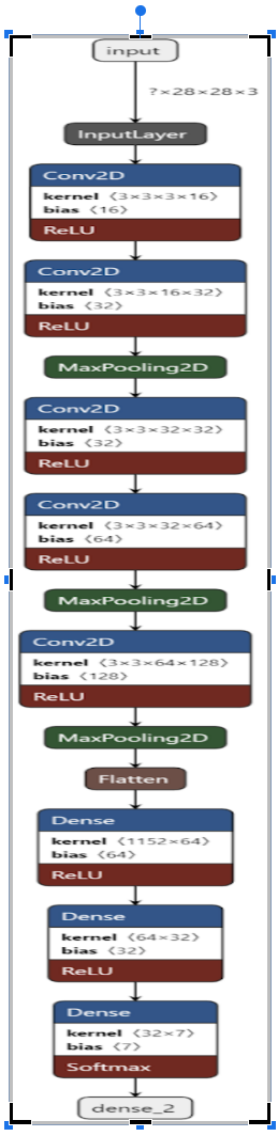

Figure 19: Model Architecture 2

without validation accuracy gain.

The model is expected to achieve exceptional accuracy on the test set and a low loss with a batch size of 128 and a validation split of 25%, demonstrating its effectiveness in categorizing various skin disorders. The quick average training duration of around one second per epoch highlights the effectiveness and efficiency of this unique CNN architecture.

5.3 Model Evaluation

The models were evaluated using a variety of parameters in the study. To properly comprehend and assess the research, it will be necessary to first understand these metrics before proceeding to analyze the model findings.

5.3.1 Confusion Matrix

The employment of a confusion matrix represents a valuable instrument in the assessment of the efficacy of a model utilized for a multiclass image categorization assignment. It gives a detailed breakdown of the predictions produced for all the classes in the dataset.

A square matrix with rows and columns corresponding to the different classes in the dataset is used to represent the confusion matrix for multiclass classification. Each cell in the matrix represents the number or frequency of samples projected to belong to a specific class.

There are several terms in the confusion matrix that need to be understood:

1. **True Positive (TP):** This is the number of samples accurately predicted as belonging to a specific class.
2. **True Negative (TN):**There are no actual negatives in multiclass classification since we are not actually considering binary results. As a result, in the confusion matrix for multiclass situations, this value is inapplicable.
3. **False Positive (FP):** When a sample is wrongfully identified as belonging to one class when it actually is related to another, this is sometimes referred to as a Type 1 error.
4. **False Negative (FN):** Type 2 error occurs when a sample is incorrectly calculated as not belonging to a specific class, but it, in fact, belongs to that class.

For each class, several performance indicators, including accuracy, precision, recall (sensitivity), and F1-score, can be determined by evaluating the confusion matrix. These metrics provide information about the model's performance in particular classes and also its overall performance.[18]

The confusion matrix also makes it easier to spot trends or patterns in the misclassification of different classes. It demonstrates which classes often get confused with each other, focusing on any areas in which the model may fall short or succeed.

To conclude, the confusion matrix is crucial in determining how well a multiclass image classification model performs. A model's accuracy and efficiency can be improved by understanding the matrix and using the performance measures that are generated.

5.3.2 Accuracy

The accuracy of a classification model, which is generally provided as a percentage, can be used to evaluate its performance. It is the percentage of forecasts when the anticipated value and the real value are exactly the same. This is calculated using the formula below :

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{5.1}$$

5.3.3 Loss

The loss function serves as a type of benchmark for the model's performance in classifying the appropriate groups of images. We have decided to employ the sparse categorical cross entropy loss function in these projects for multiclass image classification. Because it performs well once the labels are presented as numbers.

The loss function contrasts the actual labeling of the images with the probabilities that were predicted for each class. It computes a measure of how dissimilar things are. The goal is to reduce this disparity, which means getting the projected probability as close to the actual labels as possible.
The model learns to alter its parameters in a way that increases its accuracy by reducing the sparse categorical cross entropy loss during training. It improves its ability to assign greater probabilities to accurate classes and fewer probabilities to incorrect classes. This improves the model's ability to classify images into the appropriate categories.
Choosing the appropriate loss function is critical since it directs the model's learning process. Because it is developed for multiclass classification problems with integers as labels, the sparse categorical crossentropy loss function is an excellent fit for the project. It aids the model's prediction and performance when categorizing images with numerous classes.

5.3.4 Training Vs. Validation Accuracy and Loss

The crucial performance criteria during the training and evaluation of models are the metrics for accuracy and loss. Both the training and validation datasets have had these measures computed.

The training accuracy of a model represents how well it learns through the given training data. It computes the proportion of correctly identified data in the training set. Validation accuracy, on the other hand, evaluates the model's capacity for prediction and generates accurate generalization on new, previously unknown data. It assesses how the model performed on the validation set, which contains data that the model was not exposed to during training.

When the training and validation accuracy are both similar, it indicates that the model is operating well and is not overfitting. Overfitting happens when a model gets overly specialized in training data and fails to apply itself to new data. In many circumstances, the training accuracy exceeds the validation accuracy, showing that the model retains noise or unnecessary patterns from the training data.

In a similar way, training and validation losses present information about the model's quality and training. The training loss is the difference between the projected outputs of the model and the actual labels used for the training data. The validation loss computes the same mistake but on a different dataset. If the training and validation losses are similar, it suggests that the model is operating well. Deviations in the two losses may indicate overfitting or underfitting.

The phenomenon of overfitting is manifested when the training loss greatly exceeds the validation loss, indicating an excessive complexity of the model that encompasses noise or extraneous information specific to the training dataset. This trend is typically associated with machine learning algorithms that exhibit a high degree of specificity to the training data, resulting in a lack of generalizability to unseen data. However, underfitting is defined by large training and validation losses, indicating that the model is not adequately catching the patterns in the info.

As a result, it is preferable to achieve training and validation accuracies that are near to each other, as well as training and validation losses that are comparable, suggesting that the model is successfully learning and generalizing on the given data with no overfitting or underfitting.

5.3.5 Sensitivity and Specificity

Sensitivity is the percentage of actual positive outcomes that were anticipated to be positive, and it is computed using the formula :

$$\text{sensitivity} = \frac{TP}{TP + FN} \tag{5.2}$$

Specificity, on the contrary, is a measure of the proportion to actual negative outcomes that were anticipated as negative, and it is determined using the formula:

$$\text{specificity} = \frac{TN}{TN + FP} \tag{5.3}$$

The objective of a model is to get high sensitivity and high specificity.[18]

5.3.6 Precision, Recall and F1 Score

The determination of precision involves the calculation of the proportion of actual positive cases to the anticipated total of positive instances through the utilization of a specific mathematical formula:

$$\text{precision} = \frac{TP}{TP + FP} \tag{5.4}$$

In contrast, recall is a synonym for sensitivity and shares the exact identity F1 score, which is determined as the harmonic mean of precision and recall, and holds particular significance in cases where class distributions are not equally represented. A high F1 score (1) is indicative of low false positive (FP) and false negative (FN) rates, suggesting a precise identification of genuine threats and a reduced occurrence of false alarms. It is determined using the following formula:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.5}$$

5.3.7 ROC Curve and AUC

A probability curve called a ROC curve is used to assess how well a model performs for different categorization threshold values. The values of sensitivity and 1-specificity are used to plot it on the y- and x-axes. These are indicated in the graph as TPR and FPR, respectively.

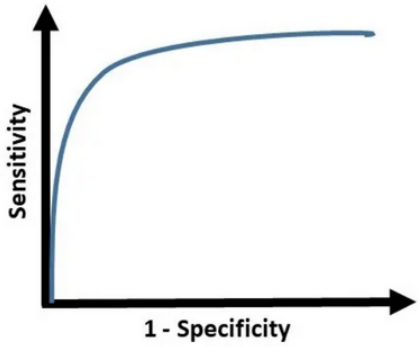

Figure 20: AUC ROC Curve [7]

The area under the receiver operating characteristic (ROC) curve, commonly referred to as AUC, is a statistically-based metric utilized for evaluating the efficacy of a classifier in effectively distinguishing between distinct classes. The capacity to accurately discern true positives from true negatives can be expressed by an AUC value nearing unity, whereas an AUC value nearing zero implies that the model is misclassifying most true negatives as positive and true positives as negative. This suggests that the model remains capable of discernment, albeit with reversed labels.[18]

Scale-invariant and classification-threshold-invariant characteristics are meritorious for AUC since they measure the quality of the predictions without taking into account the classification threshold that is used.

Chapter 6

Result Analysis

For analyzing our model we have used some of the parameters. They are confusion matrix analysis, precision, f1-score, and recall score. We have also proved, the training vs validation graph of all the other pre-trained models as well as for our custom model.
The graph represents the loss and accuracy differences between training and validation data. Finally, we have compared the accuracy, test loss, early stopping, average time per epoch, and total trainable parameter of all the models to show the best result.

6.1 Confusion Matrix Analysis

For our unique Convolutional Neural Network (CNN) model, we will give a confusion matrix in the next section. This matrix takes a thorough look at the model's predictions and provides light on both accurate as well as potential misclassifications. This can be helpful in determining where the model needs to be improved. The Confusion Metrics are shown in figure 21.

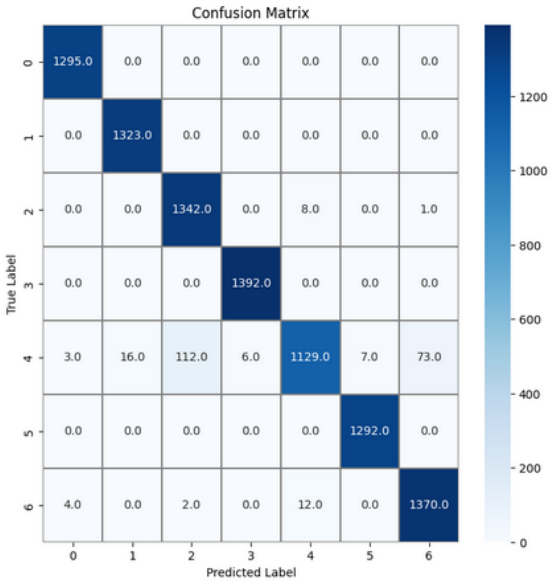

Figure 21: Confusion Matrix

6.2 Classification Report: Precision, Recall, and F1-Score

The performance of our unique Convolutional Neural Network (CNN) model is further examined in this section. We've produced a classification report that lists metrics like precision, recall, and F1-score for each of the seven skin conditions the model was trained to identify.

This table 4 provides a thorough analysis of the model's effectiveness for each class. While recall denotes the model's capacity to detect every positive instance, precision denotes its ability to accurately identify positive occurrences for each class. The F1-score, which provides a balanced assessment of the model's performance, is a harmonic mean of precision and recall.

Class	Precision	Recall	F1-Score	Support
AKIEC	0.99	1.00	1.00	1295
BCC	0.99	1.00	0.99	1323
BKL	0.92	0.99	0.96	1351
DF	1.00	1.00	1.00	1392
NV	0.98	0.84	0.91	1346
VASC	0.99	1.00	1.00	1292
MEL	0.95	0.99	0.97	1388
Overall/Average	0.98	0.97	0.97	9387

Table 4: Classification Report

6.3 Model Performance Evaluation: Training vs. Validation Graphs

An understanding of how effectively each model is generalizing and learning may be gained from the training vs. validation accuracy graph. It is preferable to have a model with high training and validation accuracy and a small difference between the two, as this shows that the model is improving and isn't overfitting.

Similar to this, the training vs. validation loss graph makes it very evident how the model's error falls off with time. The model is learning and generalizing successfully if both the training and validation loss reduce over time and converge to a low value.

We can compare the learning dynamics of the models and their tendency to overfit or underfit using these visualizations, which helps us understand their performance better. In figure 23 22, All model performance graphs are displayed.

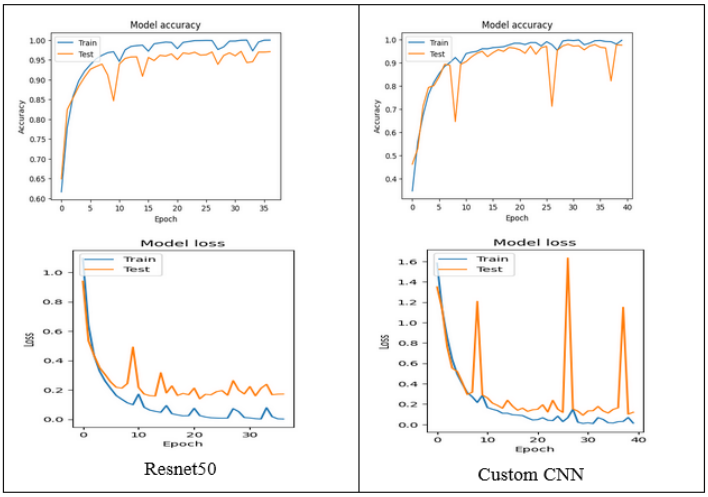

Figure 22: Resnet50 and Custom CNN graph

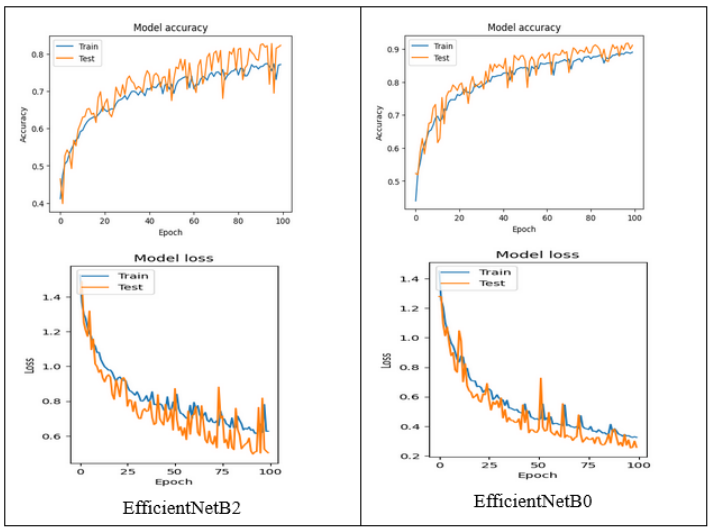

Figure 23: EfficientNetB2 and EfficientNetB0

6.4 Comparative Performance Analysis

The following Table 5 provides a comprehensive comparison of four different convolutional neural network models utilized in this study: ResNet50, EfficientNetB2, EfficientNetB0, and a Custom CNN. The comparison is based on several performance and efficiency metrics, including test accuracy, test loss, early stopping point, average training time per epoch, and the total number of trainable parameters. This comparative analysis aims to highlight the strengths and weaknesses of each model and provide insights into their suitability for different tasks.

Model	Accuracy	Loss	EarlyStopping	TestingTime	Trainableparameters
Resnet50	0.9726	0.1625	37	5s	263,175
EfficientNetB2	0.8253	0.4984	100	7s	181255
EfficientNetB0	0.9173	0.2489	100	6s	164871
Custom CNN	0.9740	0.1208	40	1s	182791

Table 5: Loss, Test Accuracy and Training duration of all the Models

1. **ResNet50:** This model had the highest test accuracy (0.9726) and test loss (0.1625), respectively. The average training duration per epoch was 5 seconds, and the model used early stopping at the 37th epoch. ResNet50 has the most trainable parameters (263,175), which may indicate that it has the ability to model complex patterns in the data.

2. **EfficientNetB2:** This model's test accuracy was 0.8253 and test loss was 0.4984. The model had an average training duration per epoch of 7 seconds and its training ended after the 100th epoch. The EfficientNetB2 model architecture appears less complex than ResNet50 with 181,255 trainable parameters, which may account for its lesser accuracy.

3. **EfficientNetB0:** This model received test results of 0.9173 accuracies and 0.2489 loss. This model's training was also terminated at the 100th epoch, just like EfficientNetB2, but it took only 6 seconds on average for each epoch. The model had the fewest number of trainable parameters among any of the others, with 164,871.

4. **Custom CNN:** The Custom CNN model performed better than ResNet50 in terms of test accuracy, achieving a score of 0.9740, and had the lowest test loss of all models, which is 0.1208. The model had the shortest average training time per epoch of just 1 second and used early stopping at the 49th epoch. Despite its outstanding performance, the Custom CNN only had 182,791 trainable parameters, which is a modest number and suggests that it learned effectively from the data.

Visualizations are used to further investigate the efficiency and performance of the four models (ResNet50, EfficientNetB2, EfficientNetB0, and Custom CNN). Line plots comparing each model's accuracy and loss for training and validation across epochs are shown in the following figures.

Chapter 7

Discussion

We have found some papers that used the same dataset, Skin Cancer MNIST: HAM10000 [20]. In their paper, they built a model and found accuracy after testing their model using the dataset. The image sizes were different on each paper. For our research, we have used 28x28x3 for the image size. In paper [10], for their model, while the model was tested using the HUM10000 dataset the highest accuracy that was achieved was 88.5%. In another paper [32], the accuracy that was achieved was 86.1% using the HUM10000 dataset. For the paper [36], the model had an accuracy of 83.1%. In paper [24] and [31] the accuracy were 85.50% and 91.5% respectively, In the paper, [39], with fusion the model's accuracy was 95%, and with feature selection it was 91.7%. The table 6 below shows all the paper's accuracy with their published dates:

Paper with reference	Year	Dataset	Accuracy(%)
[36]	2021		86.1%
[32]	2021		88.5%
[31]	2021	HAM10000	85.5%
[24]	2020		83.1%
[39]	2021		91.5%
[34](fusion)	2021		95%
[34](feature selection)	2021		91.7%
Proposed CNN			97.4%

Table 6: Different Paper's Accuracy Score

Despite our assumption that small images of 28x28x3-pixel won't provide us precise accuracy using our customized model. However, it exceeded our expectations and offered an excellent accuracy of 97.40%.

Chapter 8

Conclusion

Our unique Convolutional Neural Network (CNN) model for categorizing skin diseases has proven to be useful as a diagnostic tool. It demonstrated good precision and recall scores across all classes, with an overall accuracy of 0.974, highlighting its durability and dependability.

With a higher test accuracy of 0.974, our model was particularly better than previously trained models like ResNet50 and EfficientNetB0/B2. Additionally, it reported a test loss that was lower (0.1208) than other pre-trained models, indicating greater generalization to new data.

The model also demonstrated effectiveness with a reduced average training time per epoch (1s), showing faster learning without losing accuracy. This attribute allows it to be a better option in settings with restricted computational resources.

Our model decreases the possibility of overfitting by having only 182,791 trainable parameters, which makes it a feasible option for applications in the real world.

ResNet50 as well as the Custom CNN of our research achieved the best accuracy, according to the results, however, the Custom CNN might have been more efficient because of its lower test loss, shorter training time, and fewer parameters. However, the optimum model would depend on the specific requirements and constraints of a certain operation. For instance, if computational resources or training time are crucial considerations, then this Custom CNN would be the ideal choice.

In the future, we intend to improve our model by adding more datasets that span a larger range of skin disorders. We intend to research more advanced and refined model structures to improve interpretability. The long-term objective is to include the model we created in real-time diagnosing tools like a mobile app or web platform. We intend to work with healthcare organizations to conduct clinical trials to verify the efficacy of the model.

Bibliography

[1] M. A. Scott and W. C. Johnson, "Lichenoid benign keratosis," *Journal of Cutaneous Pathology*, vol. 3, no. 5, pp. 217–221, 1976.

[2] J. P. Callen, D. R. Bickers, and R. L. Moy, "Actinic keratoses," *Journal of the American Academy of Dermatology*, vol. 36, no. 4, pp. 650–653, 1997.

[3] B. Marie-Lise, A. Beauchet, P. Aegerter, and P. Saiag, "Is dermoscopy (epiluminescence microscopy) useful for the diagnosis of melanoma?: Results of a meta-analysis using techniques adapted to the evaluation of diagnostic tests," *Archives of Dermatology*, vol. 137, pp. 1343–1350, 2001.

[4] J. G. Einspahr, S. P. Stratton, G. T. Bowden, and D. S. Alberts, "Chemoprevention of human skin cancer," *Critical Reviews in Oncology/Hematology*, vol. 41, pp. 269–285, 2002. DOI: 10.1016/S1040-8428(01)00185-8.

[5] G. Argenziano, H. P. Soyer, S. Chimenti, *et al.*, "Dermoscopy of pigmented skin lesions: Results of consensus meeting via the internet," *Journal of the American Academy of Dermatology*, vol. 48, pp. 679–693, 2003.

[6] C. S. M. Wong, R. C. Strange, and J. T. Lear, "Basal cell carcinoma," *Bmj*, vol. 327, no. 7418, pp. 794–798, 2003.

[7] B. Zelger, B. G. Zelger, and W. H. Burgdorf, "Dermatofibroma—a critical evaluation," *International Journal of Surgical Pathology*, vol. 12, no. 4, pp. 333–344, 2004.

[8] S. Astner and R. R. Anderson, "Treating vascular lesions," *Dermatologic therapy*, vol. 18, no. 3, pp. 267–281, 2005.

[9] J. Deng and et al., "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, and et al., "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[11] R. E. Perrotta, M. Giordano, and M. Malaguarnera, "Non-melanoma skin cancers in elderly patients," *Critical Reviews in Oncology/Hematology*, vol. 80, pp. 474–480, 2011. DOI: 10.1016/j.critrevonc.2011.04.011.

[12] C. R. UK. "Cancer world wide - the global picture." Retrieved March 16, 2019. (2012), [Online]. Available: http://www.cancerresearchuk.org/cancer-info/cancerstats/world/the-global-picture/.

[13] U. R. Acharya, S. L. Oh, Y. Hagiwara, *et al.*, "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, 2017. DOI: 10.1016/j.compbiomed.2017.08.022.

[14] M. A. Hedjazi, I. Kourbane, and Y. Genc, "On identifying leaves: A comparison of cnn with classical ml methods," in *2017 25th Signal Processing and Communications Applications Conference (SIU)*, 2017, pp. 1–4. DOI: 10.1109/SIU.2017.7960257.

[15] D. B. Mendes, "Skin lesions classification using convolutional neural networks in clinical images," *arXiv*, vol. 1812.02316, 2017.

[16] S. Rai, "Benign skin lesions," *Medicine*, vol. 45, no. 7, pp. 435–437, 2017.

[17] N. Codella, V. Rotemberg, P. Tschandl, *et al.*, "Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic)," 2018. arXiv: 1902.03368 [cs.CV].

[18] J. D'Souza, *Let's learn about auc roc curve! [photograph]*, https://medium.com/greyatom/lets-learn-about-auc-roc-curve-4a94b4d88152, Accessed on May 17, 2023, 2018.

[19] M. Nasir, M. A. Khan, M. Sharif, I. Lali, T. Saba, and T. Iqbal, "An improved strategy for skin lesion detection and classification using uniform segmentation and feature selection based approach," *Microsc Res Tech*, vol. 81, no. 6, pp. 528–543, 2018.

[20] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific data*, vol. 5, p. 180 161, 2018.

[21] R. Yamashita, M. Nishio, R. K. G. Do, *et al.*, "Convolutional neural networks: An overview and application in radiology," *Insights Imaging*, vol. 9, pp. 611–629, 2018. DOI: 10.1007/s13244-018-0639-9.

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

[23] W. H. Organization. "Skin cancer." Retrieved March 16, 2019. (2019), [Online]. Available: http://www.who.int/en/.

[24] S. S. Chaturvedi, K. Gupta, and P. S. Prasad, "Skin lesion analyser: An efficient seven-way multi-class skin cancer classification using mobilenet," in *Proceedings of the International Conference on Advanced Machine Learning Technologies and Applications*, Cairo, Egypt, 2020, pp. 165–176.

[25] N. Hameed, A. M. Shabut, M. K. Ghosh, and M. Hossain, "Multi-class multi-level classification algorithm for skin lesions classification using machine learning techniques," *Expert Systems with Applications*, vol. 141, p. 112 961, 2020. DOI: 10.1016/j.eswa.2019.112961.

[26] V. Jain and J. M. Chatterjee, *Machine Learning with Health Care Perspective*, ser. Learning and Analytics in Intelligent Systems 13. Springer Nature Switzerland AG, 2020. DOI: 10.1007/978-3-030-40850-3_8.

[27] A. Khamparia, P. Singh, P. Rani, D. Samanta, A. Khanna, and B. Bhushan, "An internet of health things-driven deep learning framework for detection and classification of skin cancer using transfer learning," *Trans Emerging Tel Tech*, e3963, 2020. DOI: 10.1002/ett.3963.

[28] A. G. Pacheco and R. A. Krohling, "The impact of patient clinical information on automated skin cancer detection," *Computers in Biology and Medicine*, vol. 116, p. 103 545, 2020. DOI: 10.1016/j.compbiomed.2019.103545.

[29] M. Vidya and M. V. Karki, "Skin cancer detection using machine learning techniques," in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, IEEE, 2020, pp. 1–5.

[30] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 13 001–13 008. DOI: 10.1609/aaai.v34i07.7000.

[31] F. Afza, M. Sharif, M. Mittal, and D. J. Hemanth, "A hierarchical three-step superpixels and deep learning framework for skin lesion classification," *Methods*, 2021.

[32] T. Akram, Y.-D. Zhang, and M. Sharif, "Attributes based skin lesion detection and recognition: A mask rcnn and transfer learning-based deep learning framework," *Pattern Recognition Letters*, vol. 143, pp. 58–66, 2021.

[33] M. Al-Dabbagh, A. Al-Refai, A. Alfaraj, *et al.*, "Automated skin cancer detection using deep learning: A systematic review and meta-analysis," *JMIR medical informatics*, vol. 9, no. 11, 2021.

[34] e. a. Hindawi, "A computer-aided diagnosis system using deep learning for multiclass skin lesion classification," *Computational Intelligence and Neuroscience*, vol. 2021, p. 9 619 079, 2021. [Online]. Available: https://www.hindawi.com/journals/cin/2021/9619079.

[35] N. Kausar, A. Hameed, M. Sattar, *et al.*, "Multiclass skin cancer classification using ensemble of fine-tuned deep learning models," *Applied Sciences*, vol. 11, no. 22, p. 10 593, 2021.

[36] M. A. Khan, Y.-D. Zhang, M. Sharif, and T. Akram, "Pixels to classes: Intelligent learning framework for multiclass skin lesion localization and classification," *Computers & Electrical Engineering*, vol. 90, p. 106 956, 2021.

[37] A. Mamun, G. Rafin, A. Alam, and M. Sefat, "Application of deep convolutional neural network in breast cancer prediction using digital mammograms," in *2021 2nd International Informatics and Software Engineering Conference (IISEC)*, IEEE, 2021, pp. 1–7.

[38] A. Redha and H. Ragb, "Skin lesion segmentation and classification using deep learning and handcrafted features," *arXiv*, 2021. eprint: 2112.10307.

[39] O. Sevli, "A deep convolutional neural network-based pigmented skin lesion classification application and experts evaluation," *Neural Computing and Applications*, 2021.

[40] "Skin cancer," American Cancer Society. (2021), [Online]. Available: https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2021/skin-cancer.pdf.

[41] F. Zhuang and et al., "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021. DOI: 10.1109/JPROC.2020.3004555.

[42] S. Shah, "Convolutional neural network: An overview," Mar. 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/.

[43] T. Ahmed and N. H. N. Sabab. "Classification and understanding of cloud structures via satellite images with efficientunet." (2023), [Online]. Available: https://www.researchgate.net/figure/Architecture-of-EfficientNet-B0-with-MBConv-as-Basic-building-blocks_fig4_344410350.

[44] ——, *Classification and understanding of cloud structures via satellite images with efficientunet*, ResearchGate, 2023. [Online]. Available: https://www.researchgate.net/figure/Architecture-of-EfficientNet-B0-with-MBConv-as-Basic-building-blocks_fig4_344410350.

[45] PyTorch, *Torchvision.models.efficientnet$_b$2*, https://pytorch.org/vision/main/models/generated/torchvision.models.efficientnet_b2.html, Accessed on May 17, 2023, 2023.

[46] TensorFlow, *Tf.keras.applications.efficientnetb2*, https://www.tensorflow.org/api_docs/python/tf/keras/applications/efficientnet/EfficientNetB2, Accessed on May 17, 2023, 2023.

[47] J. Y. Wang, E. B. Wang, and S. M. Swetter, "What is melanoma?" *JAMA*, vol. 329, no. 11, pp. 948–948, 2023.

[48] "Cancer," World Health Organization. (), [Online]. Available: https://www.who.int/cancer/en/.

[49] C. Imane. "Efficientnet [model architecture]," OpenGenus. (), [Online]. Available: https://iq.opengenus.org/efficientnet/.

[50] "International skin imaging collaboration (isic) image archive." Accessed February 20, 2019. (), [Online]. Available: https://www.isic-archive.com/#!/topWithHeader/onlyHeaderTop/gallery.

[51] A. Kaushik. "Understanding resnet50 architecture," OpenGenus. (), [Online]. Available: https://iq.opengenus.org/resnet50-architecture/.

[52] *Privacy-constrained biometric system for non-cooperative users*, Scientific Figure on ResearchGate, Available from: https://www.researchgate.net/figure/ResNet-50-neural-network-architecture-56_fig4_336805103 [accessed 11 May, 2023].

[53] MedCalc, *ROC curve analysis*, https://www.medcalc.org/manual/roc-curves.php?fbclid=IwAR35mH6AJNWhql0hFK2rnVUoF7t2ZY60cvfdLmge0fJkPAvi_DptRUsAYEo, Accessed on May 17, 2023, n.d.