TC

TOKAT GAZİOSMANPAŞA UNIVERSITY

GRADUATE EDUCATION INSTITUTE

DEPARTMENT OF COMPUTER ENGINEERING

MASTER PROGRAM

USING DEEP LEARNING ARCHITECTURES FOR SKIN CANCER CLASSIFICATION

MASTER'S THESIS

Bafreen Abduljabar Mohammed

Supervisor :Assoc. Prof. Dr. Özkan İnik

TOKAT - 2024

AGREMENT

According to the thesis writing guide of Tokat Gaziosmanpaşa University Graduate Education Institute, Assoc. Prof. Dr. Özkan İnik I declare that "Using deep learning architectures for skin cancer classification", which I have prepared under the consultancy of member Assoc. Prof. Dr. Özkan inik, is an Orginal study in accordance with scientific ethical values and rules, and that I will accept all kinds of legal sanctions if the contrary is determined.

27/6/2024

Bafreen Abduljabar Mohammed

JURY ACCEPTANCE AND APPROVAL

The defense examination of the thesis titled **'Using Deep Learning Architectures for Skin Cancer Classification'** prepared by **Bafreen Abduljabar Mohamed**, was held on (27/06/2024). It was accepted by the jury members of that session as a Master's Thesis in the Graduate Education Institute of Tokat Gaziosmanpaşa University, Department of Computer Engineering, by Unanimous Vote of the Jury given below.

Jury Memebers(Title, Name Surname) **Signiture**

Chairman : Assoc. Prof. Dr. Özkan İNİK

Member : Assoc. Prof. Dr. Turgut ÖZSEVEN

Member : Asst. Prof. Dr. AHMET KARADOĞAN

APPROVAL

……./……./…….

Director of the Graduate Education Institute

ACKNOWLEDGMENTS

Above all, I want to thank God from the bottom of my heart for giving me the courage, discernment and persistence I needed to finish my thesis.

I would like to express my sincere gratitude to Assoc. Prof. Dr. Özkan İnik, my advisor, for all his help, advice, and consistent support. The completion of this research was possible thanks to your knowledge and support.

I owe my family a huge debt of gratitude for their unwavering patience and support throughout this journey. Your trust in me has always motivated me.

Finally, I would like to thank the broader scientific community for their contributions in the areas of deep learning and medical image analysis, as well as the creators and maintainers of the ISIC database, whose datasets were essential for the experimental validation of my work.

ABSTRACT

USING DEEP LEARNING ARCHITECTURES FOR SKIN CANCER CLASSIFICATION

Bafreen Mohammed

Master Degree, Computer Engineering

Thesis Advisor : Assoc. Prof. Dr. Özkan İnik

June 2024 , + 80 Pages

Skin cancer, one of the most common types of cancer, requires early and accurate diagnosis for successful treatment. For this reason, computer vision systems are used to assist specialist doctors. Especially with the development of deep learning architectures, this process has accelerated even more. In recent years, Convolutional Neural Networks (CNNs) have become a powerful deep learning method that has shown impressive results in image-based classification applications. For this reason, in this thesis, the original versions of the popular CNN models ResNet, InceptionV3, AlexNet and VGG16 and the adapted models obtained by modifying these models are used for skin cancer classification. In addition to these models, a new novel CNN model was developed. All these deep learning models were tested on the ISIC dataset, which is widely recognized for its comprehensive images of skin cancer. Before using this dataset, preprocessing was performed on the dataset as part of the thesis. With this preprocessing, the imbalance in the amount of data between classes was eliminated with data augmentation techniques. Deep learning models were trained with the reorganized dataset and the performance of the models was measured on the test data. The training of the models was performed by rearranging the image sizes in the dataset according to the size in the input layer of the deep learning models. The proposed CNN model achieved an accuracy of 83.22% for the dataset with an image size of 224x224x3 and a higher accuracy of 86% for images with a size of 128x128x3. These results demonstrate the versatility of the proposed CNN model and its potential for practical clinical applications by demonstrating its strong performance at various image resolutions. The results obtained with the proposed CNN model are extensively compared with other developed and pre-trained CNN models. Since the pre-trained models have been previously trained on high dimensional datasets, they have shown relatively high performance compared to the proposed method in this study. The VGG16 model, which is one of the pre-trained models and shows the lowest accuracy, performed relatively poorly with an accuracy of 86%. On the other hand, InceptionV3 outperformed the other models with a high accuracy of 90%, while AlexNet showed a remarkable accuracy of 89%. Finally, ResNet50 performed well with an accuracy of 88%. The results obtained show that the proposed model achieved an accuracy rate close to these models despite being trained with fewer datasets than the pre-trained models. This thesis study emphasized the importance of utilizing advanced deep learning models and comprehensive datasets to improve diagnostic accuracy and thus achieve better clinical outcomes in skin cancer treatment.

Key Words: Skin Cancer, Deep learning , Conventional neural network , Classification, ISIC dataset.

ÖZET

CİLT KANSERİNİN SINIFLANDIRILMASI İÇİN DERİN ÖĞRENME MODELLERİNİN KULLANILMASI

Bafreen Muhammed

Yüksek Lisans, Bilgisayar Mühendisliği

Tez Danışmanı : Doc. Dr. Özkan İnik

Haziran 2024, +80 Sayfa

En yaygın kanser türlerinden biri olan cilt kanserinin başarılı bir şekilde tedavi edilmesi için erken ve kesin teşhis konulması gerekmektedir. Bu sebeple uzman hekime yardımcı olması için bilgisayarlı görü sistemleri kullanılmaktadır. Özellikle derin öğrenme mimarilerinin geliştirilmesi ile bu süreç dahada hızlanmıştır. Evrişimsel Sinir Ağları (CNN'ler), son yıllarda görüntü tabanlı sınıflandırma uygulamalarında etkileyici sonuçlar gösteren güçlü bir derin öğrenme yöntemi haline gelmiştir. Bu sebeple yapılan bu tez çalışmasında cilt kanserinin sınıflandırılması için popüler CNN modelleri olan ResNet, InceptionV3, AlexNet ve VGG16 modellerinin orjinal sürümleri ile bu modellerin yeniden tasarlanması ile elde edilen uyarlanmış modeller kullanılmıştır. Bu modellere ilaveten özgün tasarlanmış yeni bir CNN modeli geliştirilmiştir. Bütün bu derin öğrenme modelleri cilt kanseriyle ilgili kapsamlı görüntüleriyle geniş çapta tanınan ISIC veri kümeleri üzerinde denenmiştir. Bu veri seti kullanılmadan önce, tez kapsamında veri seti üzerinde ön işlem gerçekleştirilmiştir. Yapılan ön işlem ile sınıflar arası veri adet dengesizliği veri arttırma teknikleri ile giderilmiştir. Yeniden düzenlenen veri seti ile derin öğrenme modelleri eğitilmiş test verisi üzerinde modellerin performansı ölçülmüştür. Veri setindeki görüntü boyutları derin öğrenme modellerinin giriş katmanındaki boyuta göre yeniden düzenlenerek eğitimler gerçekleştirilmiştir. Önerilen CNN modeli, görüntü boyutu 224x224x3 olan veri setinde %83,22 gibi bir değere ve 128x128x3 boyutundaki görüntülerde ise daha yüksek bir doğruluk oranı olan %86 ulaşmıştır. Bu sonuçlar, önerilen CNN modelinin çok yönlülüğünü ve çeşitli görüntü çözünürlüklerinde güçlü performansını göstererek pratik klinik uygulamalara yönelik potansiyelini ortaya koymuştur. Önerilen CNN modeli ile elde edilen sonuçlar diğer geliştirilmiş ve önceden eğitilmiş CNN modelleri ile kapsamlı bir kıyaslama yapılmıştır. Ön eğitilmiş modeller daha önce yüksek boyuttaki veri setleri üzerinde eğitildiği için bu çalışmada önerilen yönteme nispeten yüksek başarı göstermiştir. Ön eğitilmiş modellerden olan ve en düşük doğruluğu sergileyen VGG16 modeli %86 doğruluk oranıyla nispeten zayıf performans göstermiştir. Öte yandan, InceptionV3 %90'lık yüksek bir doğrulukla diğer modelleri geride bırakırken, AlexNet %89'luk kayda değer bir doğruluk göstermiştir. Son olarak ResNet50 modeli ise %88 doğrulukla önemli bir performans sergilemiştir. Elde edilen sonuçlar, önerilen modelin ön eğitilmiş modellere göre daha az veri seti ile eğitilmesine rağmen bu modellere yakın bir doğruluk oranı elde etmiştir. Bu tez çalışması, tanısal kesinliği artırmak ve dolayısıyla cilt kanseri tedavisinde daha iyi klinik sonuçlara ulaşmak için gelişmiş derin öğrenme modellerinden ve kapsamlı veri kümelerinden yararlanmanın önemini vurgulamıştır.

Anahtar Kelimeler: Cilt Kanseri, Derin öğrenme, Evrisimsel sinir ağı, Sınıflandırma, ISIC veriseti.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table No **Page**

LIST OF FIGURES

Figure No **Page**

LIST OF ABBREVIATIONS

ADAM : Adaptive Moment Estimation

AI: Artificial Intelligence

BN : Batch Normalization

CNN: Convolutional Neural Networks

DL: Deep Learning

DTL: Deep Transfer Learning

FC: Fully Connected

FN: False Negatives

FP: False Positives

ML: Machine Learning

ReLU: Rectified Linear Unit

ResNet: Residual Network

RMSprop: Root Mean Square Propagation

SGD: Stochastic Gradient Descent

TL: Transfer Learning

TN: True Negatives

TP: True Positives

UV: Ultraviolet Radiation

VGG: Visual Geometric Group

WHO: World Health Organization

1 INTRODUCTION

1.1 Purpose and Importance of the Thesis

Numerous reasons contribute to the relevance of skin cancer as a global health concern. It is, first and foremost, one of the most prevalent forms of cancer, affecting millions of individuals annually. Skin cancer has far-reaching consequences that go beyond numbers, posing serious health hazards to those who are affected. Melanoma, the deadliest kind, can metastasis quickly if not discovered early, resulting in a greater death rate. Even non-melanoma skin cancers, which are less fatal, can cause serious health consequences if left untreated, such as deformity and impairment.

Furthermore, skin cancer is strongly associated with UV exposure and lifestyle choices. Prolonged exposure to UV radiation from the sun raises the chance of acquiring skin cancer. As a result, promoting sun safety behaviors including sunscreen use and protective clothing is critical for preventative efforts. Despite awareness initiatives, skin cancer rates continue to climb, highlighting the importance of effective detection and treatment measures. This highlights the need of using computer vision systems to diagnose skin cancer. These systems use complex algorithms to analyze photos of skin lesions, providing various benefits over traditional diagnosis methods. For starters, they improve accuracy by detecting small changes in moles or tumors that would otherwise go undetected. This precision is critical for early detection, as detecting skin cancer in its early stages dramatically improves patient outcomes.

Additionally, computer vision systems allow for more efficient healthcare delivery. By automating the analysis process, these technologies speed up the diagnosis, allowing for timely intervention and treatment. This is especially important in areas with limited access to dermatologists, where such technology can bridge the gap and offer prompt care for patients.

Furthermore, computer vision systems' continual learning capabilities hold promise for long-term progress in skin cancer diagnosis. These systems develop their algorithms by analyzing large datasets, thereby improving diagnostic accuracy and performance.

In conclusion, skin cancer poses serious health risks, needing appropriate identification and treatment measures. Computer vision systems play an important part in this effort, providing improved accuracy, efficiency, and accessibility in skin cancer diagnosis. We can reduce the negative impacts of skin cancer and, eventually, save lives by using technology to improve early diagnosis.

1.2 Thesis Organization

In section 1 of this thesis study discusses the goal, significance, and structure of the thesis explained. The related work and research resource are studies in section 2. In 3 the Details regarding the materials utilized were provided. CNN classification and deep learning (DL) Details regarding the Proposed CNN Model and the pre-trained models that were employed are provided. Likewise, Simultaneously, a thorough explanation of the ISIC data sets used in this thesis is provided. In section 4 the architecture of the approach explained, the method's training and testing phase, and its parameter parameters Details regarding the technique, including optimization, were provided. This section also includes information on the data sets that were acquired for the thesis. contained the model evaluation tools that we used in out thesis. Result and discussion in section 5. shall first discuss performance requirements in this section. The required justifications have been provided. Additionally developed for the models classification architectures, parameter values and the resulting performance values are provided. Lastly, the pretrained model test results of the suggested approach were compared with those of alternative approaches, and any necessary discussions were had. The conclusion and next steps are outlined in 6. The reference was documented in 7.

2 LITERATURE REVIEW

A review of the literature reveals that numerous strategies have been used to develop computer-aided automatic diagnostics systems for the classification of skin cancer type, which use dermoscopic images as input and produce classification results as skin cancer type. developed a convolutional neural network for the classification of skin cancer.(Anas, Gupta et al. 2017) Four types of classification were used to create the melanoma classification (Krizhevsky, Sutskever et al. 2012). At the Neural Information Processing Systems Conference in 2012, Krizhevsky et al. (ali Bagheri, Montazer et al. 2012) presented a novel technique for classifying big (1.2 million high-resolution) photos in the ImageNet LSVRC-2010. In computer vision, the first-mover advantage belongs to a convolutional neural network. The AlexNet model has several advantages, including excellent implementation, a limited number of training parameters, and extraordinary validity. The network's pre-training on the ImageNet dataset is another big benefit. The input images may also be clearly separated from anything else in the ImageNet network. Because the image is being seen through an endoscope, it is mostly dark. Typically, the tool is the contrasting object against the background, although the true topic of the shot is tissue. Given that the majority of the images have a repetitive background with a specific type of surgical equipment that is typically the same color and shape, this provides a significant dilemma. Multiclass categorization is demonstrated in a variety of articles, namely, (Xiao-Feng, Xue-ying et al. 2010, Amornsamankul, Promrak et al. 2011, Yan and Yang 2014, Yan and Yang 2014, Wiharto, Kusnanto et al. 2015, Shah, Shah et al. 2023). They discovered that CNN and ANN were excellent in early skin cancer diagnosis using a variety of datasets and hybrid models, demonstrating the potential for these technologies to improve skin cancer detection accuracy. In order to reduce time and effort during the diagnosis process, the study underlines the uniqueness of using deep learning techniques for skin cancer detection, as well as the critical need for such a system. This study's possible application is the creation of more effective and precise skin cancer detection systems, which could lead to earlier diagnosis and improved treatment outcomes. Overall, this study stresses the necessity of using cutting-edge technologies, such ANN and CNN, in the fight against skin cancer and highlights the

potential benefits of these methods in terms of patient outcomes.(Vijayalakshmi, Uma et al. 2018). Using the PH2 dataset, we created a five-layer convolution neural network (CNN) classifier to determine if a skin lesion was a nevus or a melanoma. The model's performance was evaluated using the area under the curve (AUC), sensitivity, specificity, and classification accuracy metrics. It attained 100% AUC, 94% sensitivity, 97% specificity, and around 95% accuracy on the test set. (Mukherjee, Adhikari et al. 2019). In a different study (Shahsavari, Khatibi et al. 2023), Soumen Mukherjee et al. tested the performance of their Proposed CNN, dubbed CNN malignant lesion detection (CMLD), utilizing DERMOFIT and MEDNODE, which represent datasets of images of malignant lesions, both independently and combined. When these datasets were used individually, DERMOFIT and MEDNODE achieved 90.58% and 90.14% accuracy, respectively.Together, they achieved an accuracy of 83.07%. To identify skin lesions, (Marchetti, Codella et al. 2018) developed a novel computer-aided technique known as the Ensemble of Deep (SLDED) model. The researchers used a modified faster Regions with CNN networks (R-CNN) with deep learning model (VGGNet) feature extractor to achieve a mean average precision (mAP) of 0.96 utilizing the ISIC archive database, which contains 4668 skin lesion images for lesion localization. They use test data from ISIC (Mendonça, Celebi et al. 2015) and PH2 (Jiang, Cao et al. 2020) to evaluate the experimental classification results on 934 and 200 photos. They obtained average accuracy of 97.1% and 96%, precision of 87.1% and 90.2%, AUC of 98.6% and 98.1%, and recall of 86.7% and 85.4% for the ISIC and PH2 test data, respectively. (Kumar and Vatsa 2022) suggested the Channel & Spatial Attention Residual Module (CSARM-CNN) model, a complete architecture capable of effectively and automatically segmenting skin lesions. Spatial pyramid pooling was utilized to generate multiscale input images. A weighted cross-entropy loss function was applied to either side of the output layer to calculate the model's overall loss. The authors conducted their evaluations using two publicly accessible standard datasets, ISIC 2017 and PH2. The results demonstrated competitive results in terms of accuracy and specificity, with 94.96% and 95.23% accuracy and 99.03% and 99.45% specificity, respectively.(Harangi 2017) On the ISIC dataset, we evaluated and tested two deep neural-based classification algorithms: a recurrent neural network and a convolution

neural network, as well as an XG-Boost decision tree-based approach. The authors attempted to determine which one provides the most accurate metric for categorization performance. Benchmarks for algorithm performance include loss, precision, accuracy, recall, ROC, and F1 score. With an accuracy of 89.6%, they demonstrated that the VGG16 architecture performed best for CNN. The bidirectional RNN architecture outperforms the others in terms of accuracy (95.96%). The XG-Boost approach achieves an accuracy rate of 97.22%. studied, and a recommendation was made to use an ensemble of deep CNNs to increase the precision of individual models for the three skin cancer categorization classes. The InceptionV3, AlexNet, ResNet, and VGGNet models attain accuracy rates of 84.2%, 84.8%, 82.8%, and 81.3%, respectively. Furthermore, the combination of the InceptionV3, AlexNet, and VGGNet models achieved the highest accuracy of 83.8%. They also achieved recall rates of 59.2%, 51.8%, 52.0%, and 43.4% for each of their individual models. (Harangi 2017) demonstrated a linear classifier that can more effectively identify up to ten skin lesions by utilizing a feature extracted from a CNN pre-trained on a sample of 1300 real photographs. The recommended technique eliminates the requirement for lesion segmentation or complicated preprocessing. This approach achieved an accuracy of 85.8% for five classes and 81.9% for 10 classes. Nonetheless, there were insufficient pictures used in the work's training to extract relevant features from the dataset.(Harangi 2017). Nyiri and Kiss (Shahin, Kamal et al. 2018) . Investigated a variety of cutting-edge methods for assembling deep neural networks for the aim of identifying skin lesions, including hyper-parameters and pre-processed data sets. Ensembling can be used not just to mix different machine learning models, but also to combine different hyperparameter choices for these models. The Xception model, which was evaluated with seven classes from the ISIC2017 and ISIC2018 datasets, achieved an accuracy of 90.1%. On the other hand, (Nyíri and Kiss 2018) measures an accuracy of 80.1% for seven classes using the VGG16 model. A deep neural network-based framework (Mikołajczyk and Grochowski 2018) A classification system is suggested to categorize the seven types of skin cancers using a combination of the ResNet-50 and Inception V3 models. The system achieves an accuracy of 89.9%. Majtner et al.'s (Majtner, Yildirim-Yayilgan et al. 2019) novel strategy made use of CNN methods. Instead of using

neural networks to classify their samples, they used CNN techniques for pre-processing and feature extraction, which produced a better classifier. The method made use of the 379 test samples and 900 training samples that make up the International Skin Imaging Collaboration (ISIC) publically available dataset. The samples were divided into two groups: benign and malignant. Pre-processing involved making the image grayscale and lowering its size. After that, the system used AlexNet to apply binary masking and bounding box algorithms to extract features. Four distinct classifiers and a large number of metrics were employed after LDA techniques were used to minimise the number of features. KNN outperformed the other classifiers in terms of precision (86%) and specificity (99.9%).Gulati et al utilized pretrained networks, namely AlexNet and VGG16, in two distinct scenarios. (Ahmed, Zeng et al. 2023): is both a tool for extracting features and a framework for transfer learning. The PH2 dataset was acquired from the Dermatology Service at Hospital Pedro Hispano in Portugal. There are 200 images in all, 160 of which come under the benign group and 40 under the malignant category. Preprocessing was performed to accommodate the different input sizes of the aforementioned networks. VGG16 and AlexNet were individually utilized for feature extraction and transfer learning. The VGG16 strategy emerged as the most effective technique as a transfer learning model. The outcomes yielded a precision of 97.5% and a specificity of 96.87%. Vipin et al. implemented a two-stage technique consisting of segmentation and classification. (Vipin, Nath et al. 2021). The primary motivation for developing such a system was to eliminate human interaction from the melanoma diagnosis process, hence reducing errors and time spent performing the operation. The system used a 13,000-picture ISIC dataset, which was reduced to 7,353 photographs by removing unnecessary images. In the segmentation process, the symmetric U-Net was utilized, which consists of three main components: the expanding/upsampling path, the bottleneck, and the contracting/downsampling path. Each of these elements makes use of maximum pooling and convolutional layers. Nasr-Esfahani et al. invented the CNN. (Nasr-Esfahani, Samavi et al. 2016) For the purpose of preprocessing, feature extraction, and classification. The researchers utilized a dataset consisting of 170 nondermoscopic photographs sourced from the digital image collection of the department of dermatology at the University Medical

Center Groningen (UMCG). In order to enhance the training process, a total of 6,120 synthetic and Orginal photographs were included in the dataset. The techniques employed included Gaussian filtering, mask construction, illumination correction, and preprocessing. The CNN architecture consisted of an input layer, followed by two alternating convolutional and max-pooling layers, and concluded with a fully connected layer. The results were categorized into two classes: melanoma and nevus. The suggested model yielded results with a specificity of 80% and an accuracy of 81%. Attia et al. utilized a fully convolutional neural network (CNN) using autoencoder-decoder topologies, such as FCN and SegNet. (Attia, Hossny et al. 2017). The suggested framework utilizes 7 convolutional layers and 2 maximum-pooling layers to encode input images.In order to enhance the outcomes, this approach was augmented by incorporating a Long Short Term Memory framework alongside a recurrent neural network. The ISBI 2016 competition, titled "Skin lesion analysis Toward Melanoma Detection," supplied a dataset of 900 lesion photographs for model training and 375 test images for evaluation purposes. The Proposed model achieved an accuracy of 98% and a specificity of 94%. Mukherjee et al. developed an architecture for the detection of malignant lesions using Convolutional Neural Networks (CNN), referred to as CNN Malignant Lesion Detection (CMLD). (Mukherjee, Malu et al. 2012). The architecture of this model consists of an input layer, two max-pooling layers that divide the three two-dimensional (2D) convolutional layers, a fully connected layer at the end, and a soft-max layer for prediction. The datasets Dermofit and MEDNODE, consisting of 13,000 and 1,700 images respectively, were integrated. Accuracy values of 90.14% and 90.58% were achieved for each dataset. After combining the datasets, a precision of 83.07% was attained. When compared to previous studies in the same field and traditional classification models that rely on specific features, both models show superior individual results. Sanketh et al. suggested a CNN architecture consisting of two convolutional layers, two max pool layers, and a final fully connected layer. (Sanketh, Bala et al. 2020). The main objective of this model was to detect skin cancer at an early stage in order to initiate treatment promptly and reduce the mortality rate. The dataset consisted of 2,719 photographs obtained from ISIC, which were categorized into two groups: benign and malignant. A total of 816 photos were used for

testing purposes, while 1,906 images were used for training. Before obtaining the optimum outcome of 98%, a number of different parameters were defined and given variable values. A CNN model with convolutional layers of two separate strides was Proposed by Rahi et al. (Rahi, Khan et al. 2019). Between the input layer and the fully connected layer, they were interconnected in multiple ways using max-pooling layers, resulting in a total of nine layers. In order to enhance the results, dropout layers have been implemented. The dataset consists of 2,967 pictures that are categorized into two groups: benign and malignant. These were split into 360 test shots (360 benign and 300 malignant) and 1,440 training images (1,197 benign and 1,197 malignant). The suggested model yielded results with a specificity of 78.81% and an accuracy of 84.76%. Daghrir et al. introduced a hybrid strategy. (Daghrir, Tlig et al. 2020), The task involves combining a convolutional neural network with two conventional machine learning methods, namely CNN and SVM. The dataset consisted of 640 images of lesions obtained from the International Skin Imaging Collaboration (ISIC) collection. 128 of these images were utilized for testing purposes, while the remaining 512 photographs were allocated for training. The CNN architecture consists of a total of nine layers following the input layer. These layers include two dropout layers, three max pooling levels, three convolutional layers, and the final fully connected layer used for prediction. Several preprocessing techniques were employed, including the morphological snakes method, filters, and the Otsu method. Before the majority vote, the final classification was performed using all three methods. The CNN algorithm had the highest individual accuracy of 85.5%, followed by the SVM algorithm with 71.8% accuracy. Acosta and colleagues. (Jojoa Acosta, Caballero Tovar et al. 2021) Implemented a pre-trained ResNet152 architecture using a combination of mask-based and region-based convolutional neural network (CNN) techniques. The dataset for the International Symposium on Biomedical Imaging 2017 challenge was used to compare the results with the models used in the ISIC 2017 competition. The architecture's implementation consisted of two distinct phases. Initially, a bounding box was generated around the lesion with Mask RCNN. Subsequently, ResNet152 was employed to categorize the lesion as either benign or malignant. The eVida M6 model demonstrated superior performance compared to the top-performing ISIC 2017

models, with an accuracy of 90.4% and a specificity of 92.5%, surpassing them by 3.66%. Using the Keras Sequential API, Rahi et al. (Rahi, Khan et al. 2019) Proposed the utilization of a Convolutional Neural Network (CNN) and conducted a comparative analysis of the results with pre-trained models such as VGG16, RESNET50, and DENSENET121. The dataset, named "HAM10000," consists of 10,015 photographs and was acquired from the ISIC repository. A recommendation was made to employ a 6-layer Convolutional Neural Network (CNN) architecture, including of two max pooling layers, four convolutional layers, and a fully connected layer for making predictions. The model exhibited a precision rate of 79%. Subsequently, the dataset was subjected to VGG16, RESNET50, and Densenet50 models. Among them, Densenet50 exhibited the highest accuracy, achieving a remarkable 90%, Table 2.1 shows the dataset, methods, and accuracy of other work.

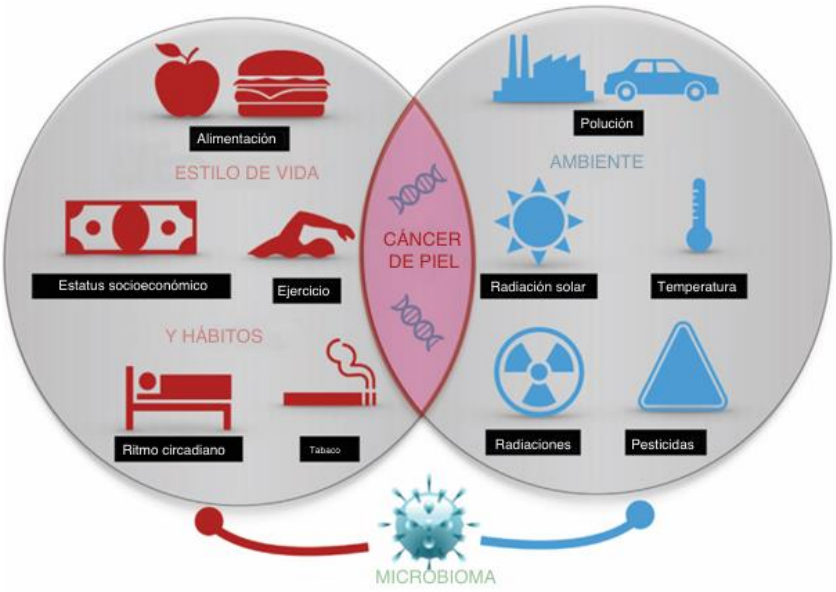

Figure 2.1 Exposure to the environment during a person's life, habits, and lifestyle, which may have an impact on the start of skin cancer (Gracia-Cazaña, González et al. 2020).

Table 2. 1 Techniques, datasets, and outcomes of classifying skin cancer using CNN

Reference	Dataset	Method	Accuracy
(Adegun and Viriri 2019)	PH2 ISIC 2017	DCNN	95%
(Ahmed, Zeng et al. 2023)	PH2	AlexNet,VGG16	97.5%
(Jayapriya and Jacob 2020)	ISBI 2016	FCNs based on VGG16 and InceptionV3	88.92%
(Nida, Irtaza et al. 2019)	ISBI 2016	Fuzzy C-Mean Clustering and Deep RCNN	94.20%
(Khan, Sharif et al. 2020)	ISBI 2017	DenseNet with IcNR	93.40%
(Hosny, Kassem et al. 2019)	DermsIS+Derm quest	Optimized CNN	93%
(Hosny, Kassem et al. 2019)	MED-NODE	Transfer Learning and pre-trained deep neural network	97.79%
(Rodrigues, Ivo et al. 2020)	ISBI 2018	Ensemble-ResNet and Inception V3	89%
(Majtner, Yildirim-Yayilgan et al. 2019)	ISBI 2017	LDA with CNN	85.80%
(Albahar 2019)	ISIC Archive	CNN and Novel Regularizer	97.49%
(Mukherjee, Malu et al. 2012)	Dermofit and MEDNODE	CNN Malignant Lesion Detection (CMLD)	83.07%
(Sanketh, Bala et al. 2020)	ISIC	A CNN created with two convolutional layers, two max pool layers, and a final fully linked layer	98%
(Rahi, Khan et al. 2019)	Benign and malignant	CNN model with convolutional layers of two distinct strides	84.76%
(Daghrir, Tlig et al. 2020)	ISIC Archive	hybrid approach	85.5%
(Jojoa Acosta, Caballero Tovar et al. 2021)	ISIC 2017	CNN combined with pretrained ResNet152	90.4%
(Rahi, Khan et al. 2019)	ISIC Archive	CNN network with CGG16,RESNET50,DENSNET50	90%
(Vipin, Nath et al. 2021)	ISIC	U-Net	88.7%
(Nasr-Esfahani, Samavi et al. 2016)	Dermoscopic	CNN	81%
(Attia, Hossny et al. 2017)	ISBI 2016	CNN with autoencoder-decoder architecture such as FCN,segNet	98%

This section examines nineteen studies that employed various methodologies to categorize skin cancer. Some studies employed CNN models, while others utilized pre-trained CNN models and attained notable levels of accuracy. However, this study aimed to assess the precision of our model and pretrained models by utilizing multiple pretrained models, in comparison to earlier research.

In terms of output, CNNs can produce better outcomes than conventional machine learning methods. This is because convolutional neural networks, or CNNs, have several layers, each of which improves the network's ability to learn. CNNs do not convert the image into a single column vector, in contrast to traditional methods or simple feedforward networks. But even if the sizes could fluctuate, the layout of the pixels stays the same. This methodology differs from other conventional methods in that it makes spatial information preservation easier.

Because even the smallest details matter when it comes to accurately categorising lesions and malignancies, this allows for enhanced performance and more sophisticated feature extraction. Generally speaking, conventional methods are limited and should only be applied to simple feature extraction or classification jobs. CNNs, on the other hand, have several layers and filters built into their architecture, which makes them extremely flexible. This makes it possible to generate a unique design that complements the current photos or dataset. As a result, Convolutional Neural Networks (CNNs) perform better and provide a more individualised and adaptable method that can be adjusted to meet specific needs.

3 MATERIAL AND METHOD

3.1 Deep Learning

A branch of machine learning called deep learning, has garnered considerable interest and widespread usage in recent times on account of its exceptional efficacy across a range of applications, including but not limited to speech recognition, natural language processing, and image recognition. Fundamentally, deep learning entails the instruction of neural networks comprising numerous layers (hence the nomenclature "deep") in order to autonomously acquire hierarchical data representations. These representations facilitate the network in deriving progressively more abstract and intricate interpretations of the input data, thereby enhancing the precision of the predictions and classifications in Figure 3.1 Diagram showing how machine learning, deep learning, and artificial intelligence are nested. The entire field of developing intelligent systems is referred to as artificial intelligence. Algorithms for machine learning, a branch of artificial intelligence, enable systems to learn from data.

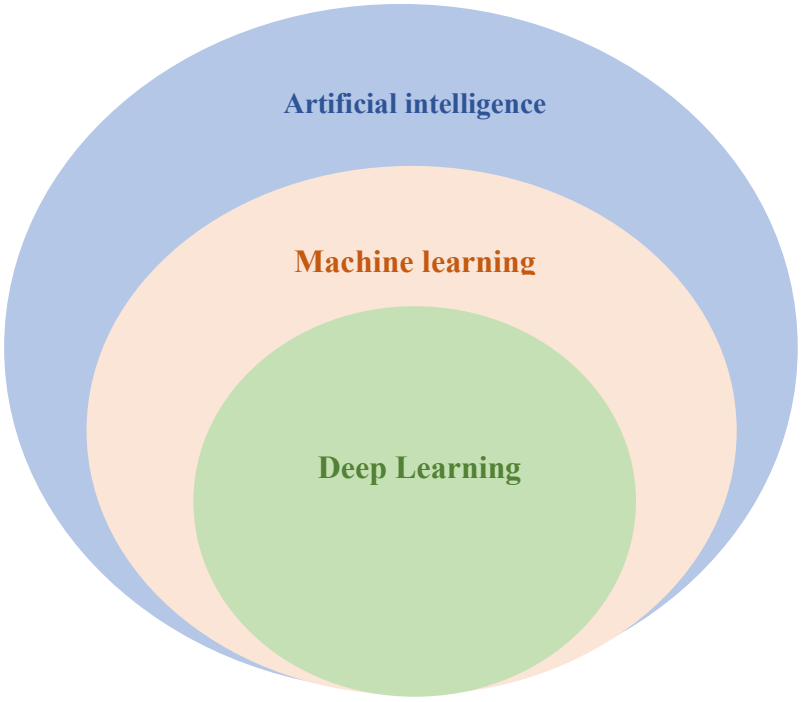

Figure 3.1 Artificial intelligence, machine learning, deep learning

The following is an exhaustive exposition of deep learning, accompanied by citations for additional research. Deep learning is predicated on artificial neural networks, which are computational models that draw inspiration from the architecture and operation of biological neural networks found within the human brain. A neural network is composed of layers of interconnected elements. Every individual neuron executes a basic computation and transmits the result to neurons in the subsequent stratum. The input layer is the initial stratum, followed by the output layer; intermediate strata are referred to as concealed layers{Heaton, 2018 #53}.

The training process for deep learning models consists of backpropagation, wherein the model's parameters (weights and biases) are adjusted iteratively in order to minimize the discrepancy between the predicted outputs and the actual objectives. Typically, a substantial quantity of annotated training data is necessary for this procedure, which is executed utilizing optimization algorithms like stochastic gradient descent {Heaton, 2018 #53}.

Diverse deep learning architectures exist, each of which is optimized for a particular category of data and set of tasks. Recurrent Neural Networks (RNNs) are more effective with sequential data, such as speech and text, whereas Convolutional Neural Networks (CNNs) are extensively employed in the domain of image processing{Shanmugamani, 2018 #54}.

Deep learning has been effectively implemented in numerous sectors, including autonomous vehicles, computer vision, natural language processing, healthcare, and finance. Self-driving vehicles, medical image analysis, financial forecasting, object detection, and machine translation are all instances of applications{Heaton, 2018 #55}.

Challenges and Future Directions: Deep learning, despite its achievements, encounters obstacles such as the requirement for substantial quantities of annotated data, the comprehensibility of models, and resilience against adversarial attacks. Promising areas for future research encompass the advancement of more streamlined architectures, the exploration of ethical and societal ramifications, and the fusion of deep learning with other artificial intelligence methodologies{Sejnowski, 2018 #56}.

3.2 Convolutional Neural Network CNN

CNN is a specific type of deep neural network (Nugroho, Nugraha et al. 2021) .influenced by biological studies (Liang, Deng et al. 2020). due to its unique topology. The introduction of it took place in Fukushima in 1998. It has broad applications in identifying activities, recognizing text, classifying images, detecting objects, and localizing them. (Pang, Cao et al. 2016), and scanning result analysis. Neurons constitute their composition, and individual neurons can be trained to prioritize particular activities according to their synaptic strength. A basic Convolutional Neural Network (CNN) is made up of numerous hidden layers, an output layer, and an input layer. Convolution (Weng, Zhou et al. 2019), pooling (Guo, He et al. 2019), and fully connected (FC) layers (Dhillon and Verma 2020) are some of the hidden layers. The layers consist of two main components: feature extraction and categorization. The layers of CNN are presented by Eq (3.1)

$$Z^{[i]} = \sum(w^{[i]}\, x^{[i]} + b^{[i]}) \tag{3.1}$$

Where $Z^{[i]}$ is the current layer , $x^{[i]}$ is the input layer, $w^{[i]}$ represents the weights for the i layer and $b^{[i]}$ is the bais (Panwar, Gupta et al. 2020).the simple design of CNN(Phung and Rhee 2018).

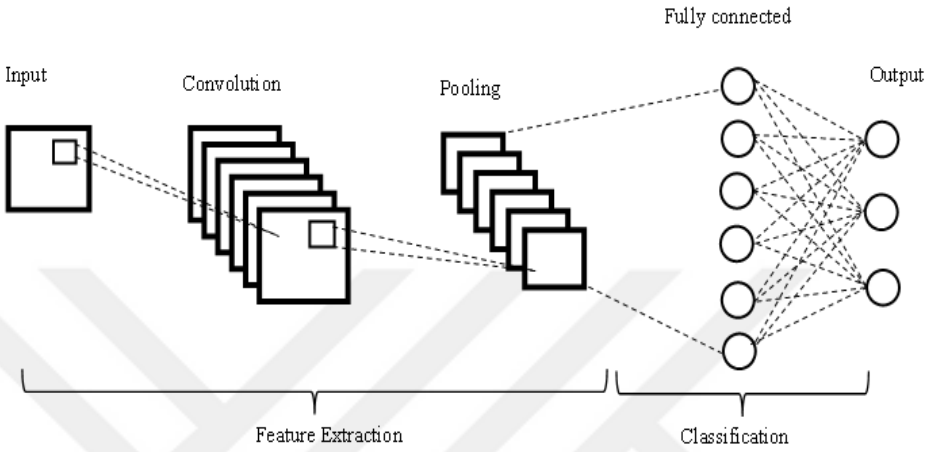

Figure 3.2 Using convolutional and pooling layers for feature extraction and pattern recognition, (Phung and Rhee 2018).

Machine learning algorithms are trained to autonomously classify malignant and benign skin patches. Convolutional Neural Networks (CNN) are utilized for the purpose of detecting skin cancer. Assess the accuracy of trained models by utilizing diverse test pictures.(Shah, Shah et al. 2023).

These methods have been utilized in several medical imaging applications, such as the classification of brain tumors and the detection of breast cancer. (Singh, Farrelly et al. 2023). Convolutional neural networks (CNNs) have been crucial in integrating, researching, and diagnosing medical imaging data. This advancement has facilitated medical professionals to conduct more comprehensive investigations on uncommon ailments, expedite patient diagnoses, and strategize medicinal procedures. Nevertheless, the accuracy and training of CNNs are hindered by issues related to the quantity, quality, and imbalance of data. Moreover, the expenses for training might significantly increase when a healthcare system requires multiple CNN models. By incorporating topology and geometry into the architecture of convolutional neural networks (CNNs), either in the convolution layers or data preprocessing stages, these approaches can effectively mitigate the problems faced by CNNs.(Yu, Yang et al. 2021). In the field of medical imaging, CNN models typically

undertake tasks that can be broadly classified into three fundamental categories: segmentation, detection, and classification. Popular models such as AlexNet, InceptionV3, VGG, and ResNet employ a combination of fully-connected layers and various classifiers to achieve categorization. (He, Zhang et al. 2016).

3.2.1 Convolutional layer

The convolution layer is an essential and pivotal component of a Convolutional Neural Network (CNN). To construct a feature map, the pixel matrix obtained from the given picture or object is subjected to multiplication or convolution. Feature maps possess the essential advantage of retaining all the unique characteristics of an image, hence decreasing the quantity of data that necessitates investigation. The input data is convolved using a matrix utilized as a feature detector or filter. At its fundamental level, a feature detector is a set of data that can be analyzed by a machine. Adjusting the threshold value of the extracted feature can lead to a diverse range of image alterations.(Chen, Chen et al. 2020).

3.2.2 Pooling layer

Pooling is an essential process for lowering the dimensionality of feature maps. It selectively retains only the necessary components while also diminishing spatial invariance. This diminishes the quantity of characteristics that can be learned in the model. This aids in mitigating the issue of overfitting. A Convolutional Neural Network (CNN) is capable of identifying an item, even if its shape is distorted or tilted, due to the pooling mechanism. This mechanism enables the CNN to assimilate various resolutions and orientations present in an image. (Shustanov and Yakimov 2019). There are other types of pooling, such as global average pooling (GAP), average pooling, and maximum pooling. Max pooling is a technique that involves breaking an image into smaller rectangular sub-regions and then selecting the highest value inside each sub-region. (Zhang, Xiang et al. 2020). One of the most popular sizes for max-pooling is 2×2 (Albawi, Mohammed et al. 2017). Figure 3.3 depicts how max pooling works.

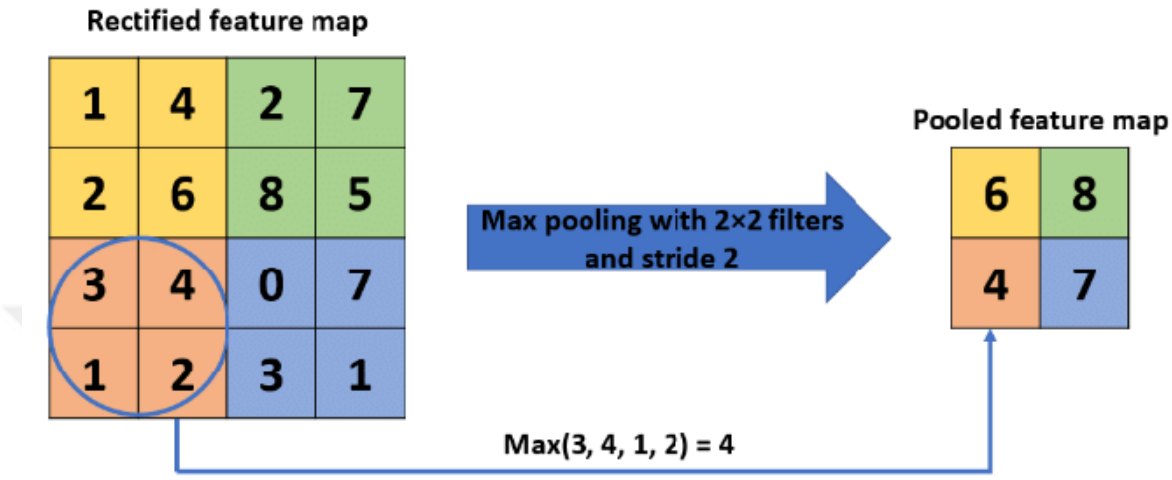

Figure 3.3 Max pooling operation (Albawi, Mohammed et al. 2017)

3.2.3 Fully connected layer

The dense layer, sometimes referred to as the fully connected layer (FC layer), serves as the ultimate layer in a neural network architecture. It has a pivotal function in generating the final result or determining choices in a categorization process. (Qian, Huang et al. 2020). The model undergoes training using the backpropagation technique to optimize its weights and acquire knowledge of intricate patterns and correlations among the features. The matrix data is often compressed before being transmitted to neurons. Dropout, a regularization technique, is sometimes used to mitigate overfitting in fully connected layers.

3.3 Activation Functions

Activation functions are employed in Convolutional Neural Networks (CNNs) to introduce non-linearities to the output of each neuron. Neurons become activated or discharged when the activation function receives input. The network possesses the capacity to obtain intricate patterns and alter incoming data in a non-linear fashion. The choice of the activation function should be determined by the particular architectural design and goals of the network. The activation functions frequently employed in Convolutional Neural Networks (CNNs) encompass.

3.3.1 Sigmoid

As seen in Figure 3.4 the sigmoid activation function compresses the input into the range of [0, 1]. It is frequently applied to binary classification issues in the output layer (Jaiswal, Gianchandani et al. 2021) . The 3.2 formula can be used to express the sigmoid function.

$$y = \frac{1}{1+e^{-(\sum w_I x_I)}} \tag{3.2}$$

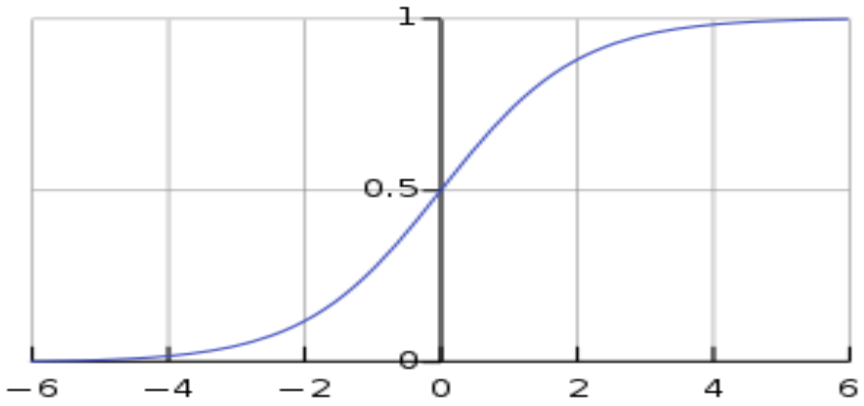

Figure 3.4 The Sigmoid curve (Jaiswal, Gianchandani et al. 2021)

3.3.2 Rectified linear unit (ReLU)

One of the activation functions in CNNs that is most frequently utilized is the Rectified Linear Unit (ReLU). As seen in Figure 3.5, it sets all negative values to zero while maintaining positive values. ReLU can be mathematically expressed in Eq(3.3) and is computationally efficient as well as helpful in resolving the vanishing gradient problem (Yu, Adu et al. 2020) .

$$\mathrm{ReLU}(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \tag{3.3}$$

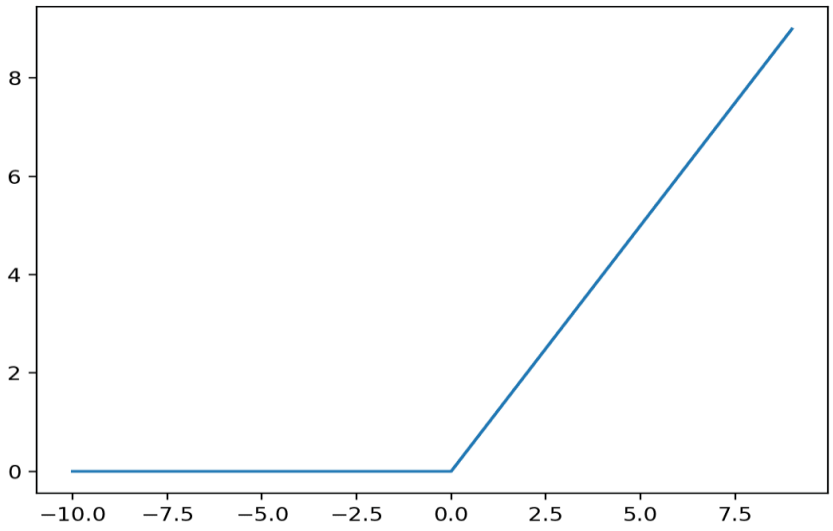

Figure 3.5 RelU activation function (Yu, Adu et al. 2020).

3.4 Optimizer

3.4.1 Adam

One well-liked optimization approach is called Adaptive Moment Estimation (Adam). This method is a variation of stochastic gradient descent that employs adaptive estimation for both the first and second moments (m_t and v_t). It modifies the learning rate (α) for each parameter by considering its past gradients and updates. Adam is renowned for its effectiveness and rapid convergence on a variety of assignments. The following explains how the Adam optimizer updates the model weight θ:

$$\theta_t = \theta_{t-1} - \alpha \frac{m_t}{\sqrt{v_t + \varepsilon}} \tag{3.4}$$

Where

$$m_t = \beta_1 \times m_{t-1} + (1 - \beta_1) \times g_t \tag{3.5}$$

$$v_t = \beta_2 \times v_{t-1} + (1 - \beta_2) \times g_t^2 \tag{3.6}$$

Here,g_t shows the gradient's weights in relation to the loss function, θ^{t-1} is the weight in real time $t-1$, θ_t indicates the weight at time t, while the first-moment estimate is given by m_t , and v_t is the estimated second raw moment,β1 and ββ2 are hyperparameter (Sarker, Islam et al. 2020).

3.4.2 RMSprop

This is an extra adaptive optimisation technique meant to reduce the Adagrad's learning rate that decreases monotonically. The learning rate is divided by the square root of the

exponentially decreasing average of squared gradients, or $R[d^2]_t$, in the RMSprop weight update procedure. The Eq (3.7) provides the weight update rule.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{R[d^2]_t + \epsilon}} \tag{3.7}$$

3.4.3 SGD

An integral part of CNN training is the optimizer, which updates the network's parameters (weights and biases) in accordance with the calculated gradients of the loss function. The optimizer chooses how to apply these changes in order to minimize loss and improve CNN performance. The following optimizers are frequently employed in CNNs, gradient descent stochastic (SGD) The optimization technique known as stochastic gradient descent (SGD) is extensively applied in the domains of machine learning and deep learning. Through the changing of the model's parameters, this iterative optimization technique tries to minimize the cost or loss function. Stochastic Gradient Descent (SGD) demonstrates exceptional efficiency when dealing with extensive datasets and intricate models, as it processes each training example individually. The gradient of the loss function with respect to the model parameters is the basis for parameter adjustment, which is the core principle of Stochastic Gradient Descent (SGD). Every iteration of Stochastic Gradient Descent (SGD) computes the gradient by employing a solitary randomly selected training instance. By including stochasticity into the optimization procedure, the technique may efficiently traverse intricate loss landscapes that lack convexity and prevent being trapped in local minima. (Singh, Kumar et al. 2021).

3.5 Popular CNN Models

The most advanced CNN models are well known for their exceptional abilities and creative approaches to challenging computer vision tasks. Every model has presented innovative architectural designs or approaches that have greatly enhanced the efficiency, scalability, or effectiveness of deep learning models for image analysis. Moreover, their robust ability to make broad conclusions allows them to excel in analyzing both familiar and unfamiliar information, becoming them extremely adaptable and useful in diverse fields. Moreover, the presence of pre-trained models and open-source implementations has greatly assisted their extensive acceptance, rendering them readily available to both researchers and practitioners. Their flexibility to scale up or down makes them even more attractive, since they may be customized to meet specific computational limitations or performance needs. The appeal of these cutting-edge CNN models has been solidified in both academic research and practical applications because to their exceptional performance, innovation, accessibility, scalability, and community support.

Four popular CNN models were used in the research ResNet50, VGG16, InceptionV3, and AlexNet. These models have received considerable attention and appreciation in the field of computer vision because to their superior performance and applicability across a wide range of image processing applications. By merging these well-known architectures into our research framework, The strength and efficiency of these models may be taken advantage of to address complex problems in image classification, object recognition, and other related fields. It is hoped that the distinct strengths and capacities of these models will be harnessed to achieve significant advances in our work and contribute to the larger landscape of computer vision research.

3.5.1 ResNet50

A well-known deep learning architecture, the ResNet (Krizhevsky, Sutskever et al. 2012) model uses residual mapping and identity shortcut links throughout its whole structure to get

better performance results. The ResNet50 model differs from typical CNN design by using skip connections to mitigate the problem of gradient vanishing that occurs during deep network training. Unlike the conventional approach of using input from all previous layers to capture condensed visual features, ResNet50 selectively skips connections to maintain gradient flow. Figure 3.6. CNN design. Remaining blocks (RBs), the building block of ResNet designs that can elide one or more layers, are the basis of this design.

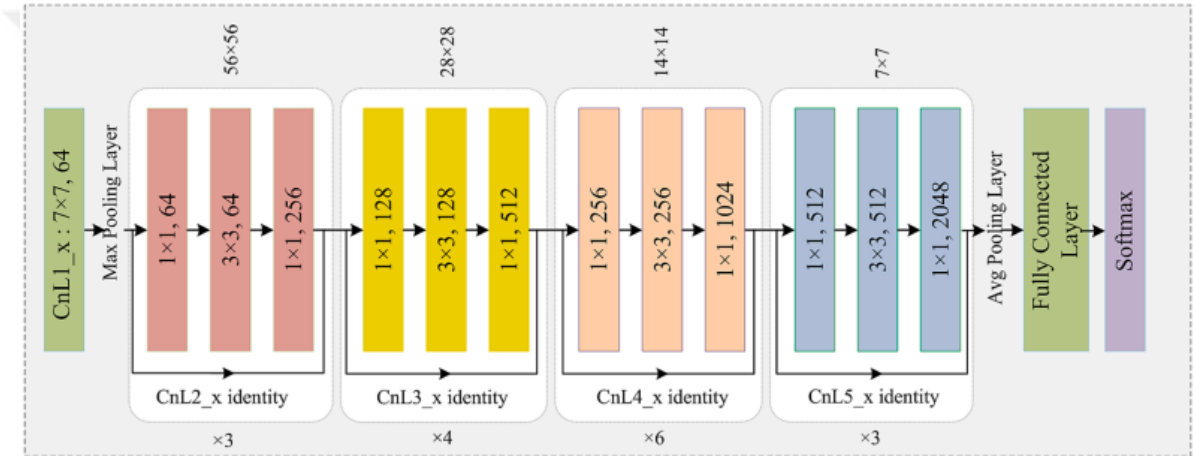

Figure 3.6 organizational view of ResNet50(Krizhevsky, Sutskever et al. 2012)

3.5.2 AlexNet

In 2012, Alex Krizhevsky, Geoffrey Hinton, and Ilya Sutskever suggested AlexNet, as seen in Figure 3.7 The convolutional layers of the deeper, more filter-rich network are made up of 11 × 11, 5 × 5, and 3 × 3 filters as well as 2 × 2 overlapping max pooling. All fully-connected and convolutional layers received ReLU activations. Additionally, the gradient was updated using stochastic gradient descent (SGD) with momentum ($\theta = \theta - \eta \cdot \nabla_0 J(\theta; X^{(i)}; y^{(i)})$) (Maqsood and Damaševičius 2023). To increase classification accuracy, LRN (Local Response Normalization) was added to each ReLU function. The definition of LRN is show in equation 3.8.

$$b^{i}_{xy} = a^{i}_{xy}/(k + \propto^{\min(N-1,i+n/2)} (a^{j}_{x.y})^{2})^{\beta} \quad (3.8)$$

where the input of the subsequent layer is indicated by $a^{i}_{x.y}$which regularized the output provided by kernel I through an activation function at position x,y,$b^{i}_{x.y}$ N is the total number of feature maps in this layer, and n is the size of the former and latter n/2 feature maps along the c+hannel that make up the normalization neighbor. Keep in mind that users initialize the hyper-parameters of k, n, ∝, and β, and that the sum is performed along the channels.

To increase the training pace with two GPUs, AlexNet was specifically split into two pipelines for parallel processin.

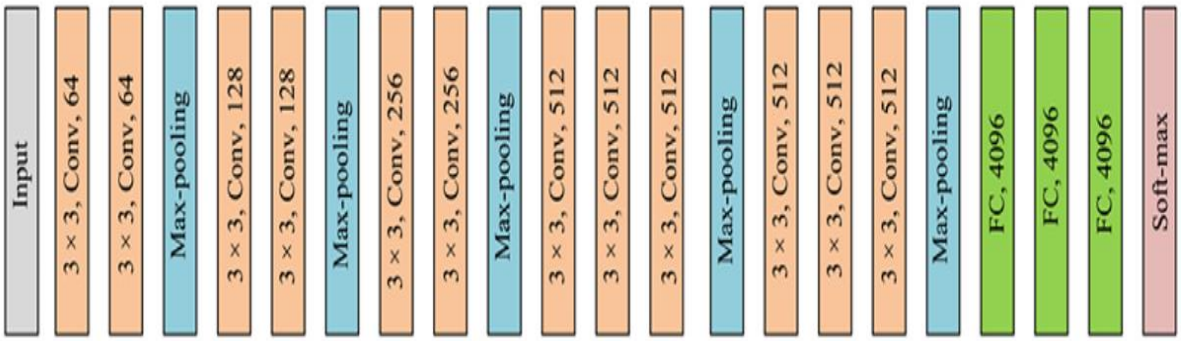

Figure 3.7 AlexNet structure (LeCun, Bengio et al. 2015)

3.5.3 VGG16

The CNN network VGG16 was created by Andrew Zisserman and Karen Simonyan(Simonyan and Zisserman 2014).VGG16 was used in its construction, and it has a CNN depth of 16 available. There are 138 million parameters in the VGG16 framework. As the name suggests, the VGG-16 architecture consists of sixteen layers total: thirteen convolutional layers, five maximumpooling layers, and three fully connected layers. Instead of using a huge stride 1 filter, VGG uses a 3 3 filter with a receptive field comparable to a 7 7 convolutional layer. The model was initially developed using the 224 x 224 × 3 ImageNet dataset. The FC layer architecture is changing as part of this work. The FC layer is removed

from the native model and replaced with a new FC layer comprised of only eight classes. This network has 224 × 224 × 3 inputs and 7 × 7 filters for the first layer. The previous layer's filter sizes are 1 x 1, 3 x 3, and 1 x 1, in that order. The modified network is trained with the TL. In the TL technique, the batch size, epoch, and learning rate are 64, 100, and 0.01, respectively. After training the network, the feature extraction operation is carried out on the average pooling layer. The size of the retrieved features is M × 2048. In Figure 3.8, Figure 3.9, the architecture of VGG16 CNN model is displayed (Sanghvi, Aralkar et al. 2020).

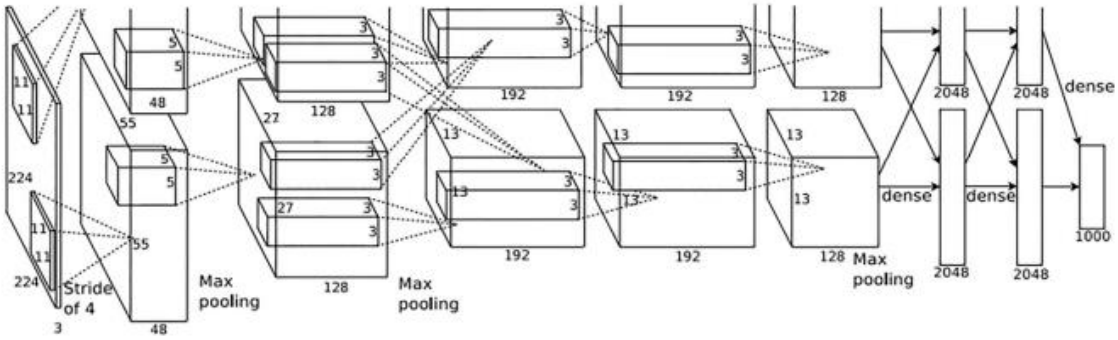

Figure 3.8 Architecture of VGG16 for feature extraction (Sanghvi, Aralkar et al. 2020)

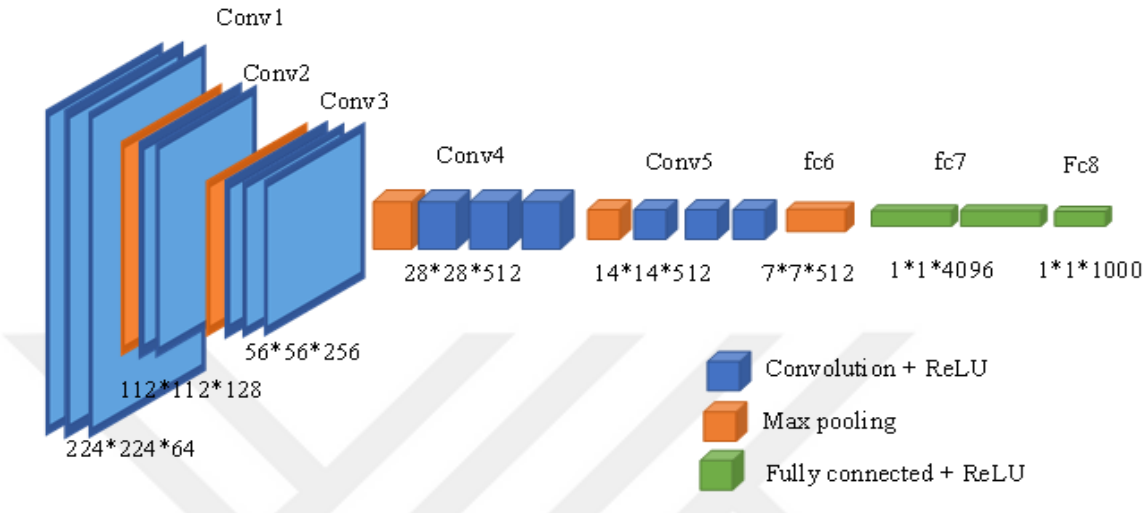

Figure 3.9 The basic structure of VGG (Sanghvi, Aralkar et al. 2020)

3.5.4 InceptionV3

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was won by InceptionV3 in 2014(Szegedy, Liu et al. 2015). The idea of network structure was first Proposed by Christian Szegedy (Lin, Li et al. 2020).The structure is based on the LeNet and AlexNet framework structures, with slight adjustments made to the network width and depth. There are 22 network layers in the structure. It uses a new kind of parallel structure

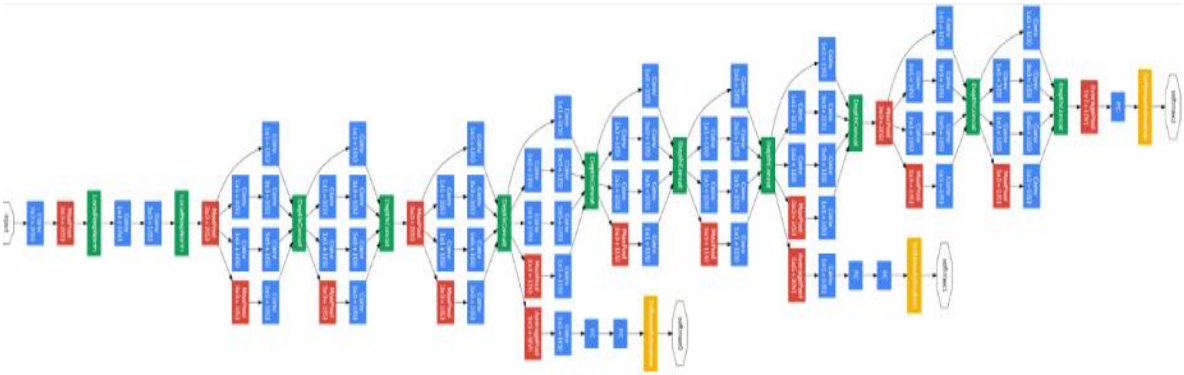

Figure 3.10 architecture of InceptionV3 (Yuesheng, Jian et al. 2021)

that significantly reduces the training cycle. The DL research explosion has peaked thanks to InceptionV3 and VGGNet (Yuesheng, Jian et al. 2021) Figure 3.10,Figure 3. 11 show the details of InceptionV3 architecture.

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 3. 11 Layer by Layer architectural details of InceptionV3(Yuesheng, Jian et al. 2021)

3.6 Model Performance and Evaluation Tools

By contrasting the true and predicted labels, A confusion matrix was created, allowing us to see how well the model performed. With the use of this matrix, which offers comprehensive insights into the kinds and frequency of misclassifications the model makes, we may pinpoint particular classes in which the model performs well or poorly in Figure 3. 12. Shown the confusion matrix.

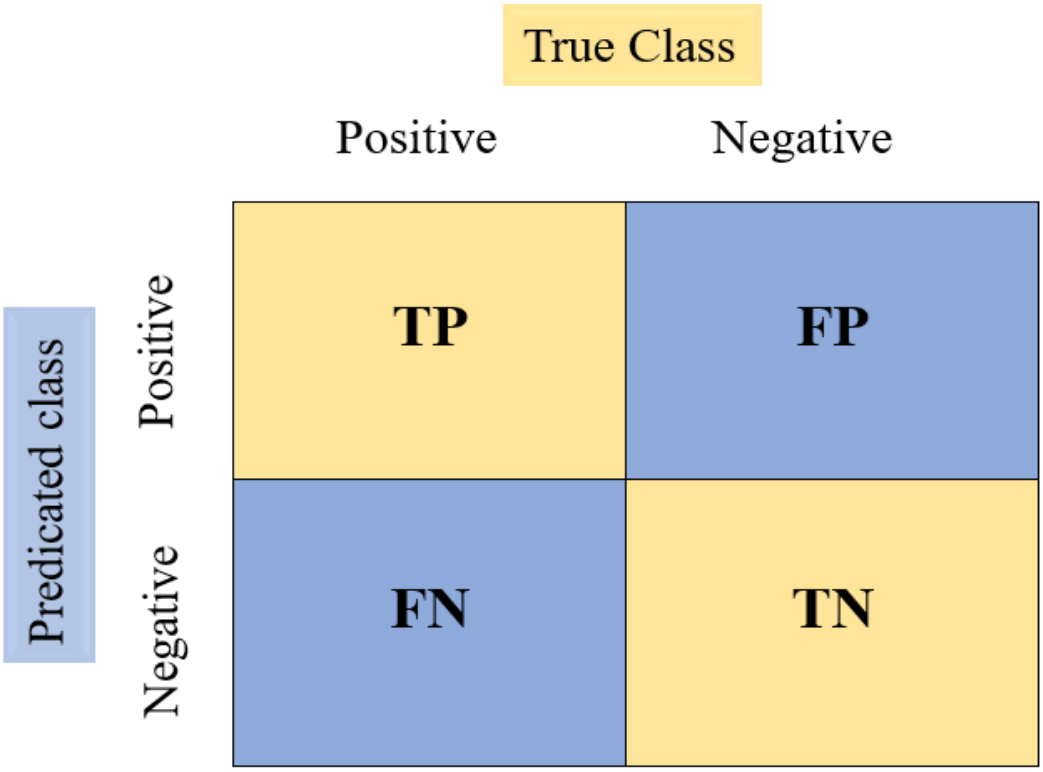

Figure 3. 12.This image is the confusion matrix, depicting the performance of the classification model by showing the counts of true positives, true negatives, false positives, and false negatives.

The confusion matrix will enlarge to accommodate more classes in the event of a multi-class classification problem, with each row denoting the actual class and each column the projected class.

True Positives (TP): Both the actual yield and the anticipated outcome occurred as expected. True Negatives (TN): The instance where the real yield was also false, despite our expectations being false. False Positives (FP): An instance where the genuine yield was not what we had expected to be true. False Negatives (FN): An instance where we expected a false result but the actual yield was also true (Ali, Miah et al. 2021).

True Positives (TP): the situations in which the actual outcome satisfied our expectations of reality.

True Negatives (TN): The instance where the real yield was also false, despite our expectations being false.

False Positives (FP): When an event occurs that we expected to be true but the actual yield is likewise untrue.

False Negatives (FN): An instance where the real yield was true despite our expectation that it would be false (Ali, Miah et al. 2021).

3.6.1 Accuracy

Accuracy is the proportion of accurate forecasts, which includes true positives and true negatives, to all projections, which includes false positives, false negatives, true positives, and true negatives(Ali, Miah et al. 2021).According to Eq. (3.9), this parameter measures the statistical validity of the forecast. In this case, FN = False Negative, FP = False Positive, and TP = True Positive. Because this measure depends on both the FP and FN, relying just on it might occasionally be deceptive when assessing a predictor's performance. This implies that two models that have the same accuracy can have different FP and FN values, with one having high FP and low FN and the other having the opposite values. Because the first model has a lower FN for the delicate medical scenario—which may not be determined only by the models' accuracy values—it can thus be chosen over the second.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.9}$$

3.6.2 Precision

Precision is calculated by dividing the total number of positive outcomes that the classifier predicted by the number of correct positive results.(Ali, Miah et al. 2021). Precision is calculated according to Equations (3.10). The precision measures the number of images that belong to a specific class among the anticipated images.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3.10}$$

3.6.3 Recall

refers to the percentage of relevant examples (true positives) that have been retrieved out of all the relevant instances in the dataset when used in the context of machine learning and information retrieval. Put another way, it assesses a model's capacity to identify every pertinent example in a dataset. (Ali, Miah et al. 2021),the equation 3.11 calculate the recall.

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3.11}$$

3.6.4 F1-score

As a single metric that balances accuracy (the ratio of true positives to all projected positives) and recall (the ratio of true positives to all real positives), the F1-score is, in fact, the harmonic mean of precision and recall. It is especially helpful in situations where you want to take into account both false positives and false negatives, like in datasets that are imbalanced and have a higher prevalence of one class than the others. (Ali, Miah et al. 2021).

Using the harmonic mean of precision (P) and recall (R), the F1-score is computed.the equation (3.12) calculate the F1-score.

$$F1 = \quad 2 \text{ x } \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.12}$$

It is noted that the F1 score, recall, and precision all have values between 0 and 1. Better performance for a particular classification job is indicated by higher precision, recall, and F1 score values (Chanda, Onim et al. 2024).

3.6.5 Plots of accuracy and loss

Throughout the training epochs, we charted the accuracy and loss for both training and validation. These plots aid in the comprehension of the model's learning behaviour by showing whether the model is overfitting or underfitting and how effectively it generalises to new data.

3.6.6 Classification report

The comprehensive report that was generated include the F1-score, recall, accuracy, and total accuracy for each class. A comprehensive evaluation of the model's performance in several classes is offered, with special attention to the harmonious coexistence of recall and precision.

3.7 Dataset Description

The International Skin Imaging Collaboration (ISIC) provided 2357 images of benign and malignant oncological illnesses, which made up the dataset that we used. Except for melanomas and moles, whose photos are somewhat predominate, all photographs were classified in accordance with the ISIC categorization, and each subset was split into the equal amount of images (Esteva, Kuprel et al. 2017). On the other hand, the datasets used have varying quantities of photos per class. Nine balanced datasets were produced for skin cancer categorization. Figure 3.13 Show the dataset samples and Table 3.1 Describe the number of images in each class.

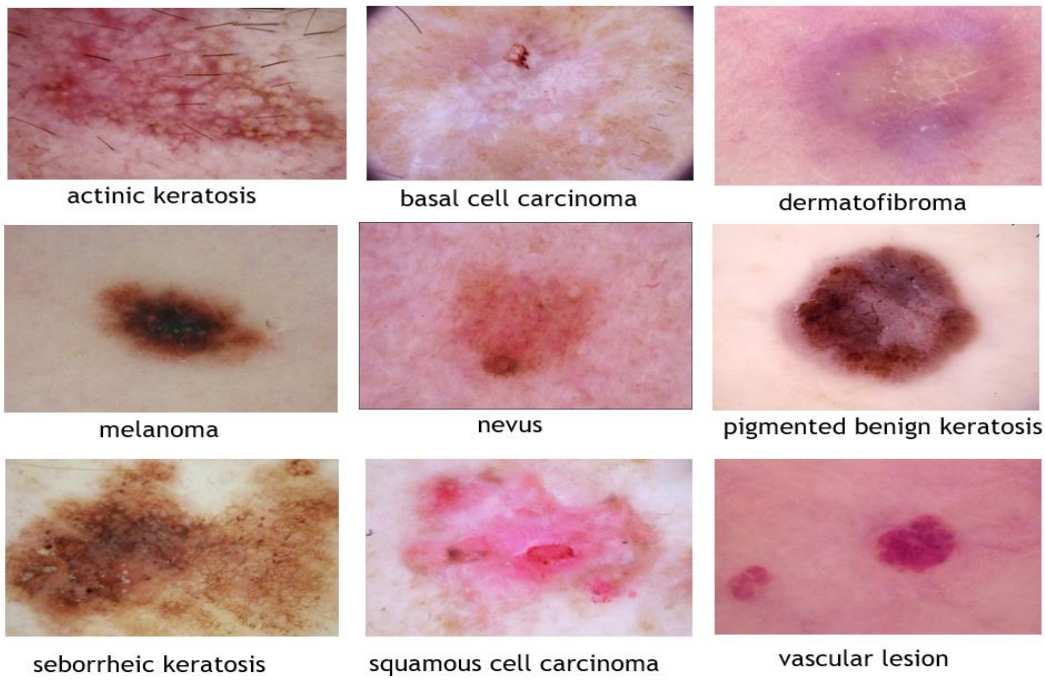

Figure 3.13 some random dataset images from ISIC dataset

Table 3. 1 ISIC dataset Before Augmentation

Class name	Number of images
actinic keratosis	130
basal cell carcinoma	392
Dermatofibroma	111
Melanoma	454
Nevus	373
pigmented benign keratosis	478
seborrheic keratosis	80
squamous cell carcinoma	197
vascular lesion	142
Total	2357

3.8 Data Preprocessing

The term "pre-processing" describes all of the modifications made to unprocessed data prior to supplying it to algorithms for deep learning or machine learning (ML). A function called (resize_image_array) was utilised. This function takes an image file path as input, opens it with the Python Imaging Library (PIL) or Pillow (Image.open), and resizes it to a fixed size.we resized it to 224x224 pixels using the resize method. The resized image is then converted to a NumPy array via np.asarray and returned.

Concurrent.Futures is used to parallelize the operation and handle the resizing of several photos efficiently.A thread pool executor with a given maximum number of worker threads is created using the ThreadPoolExecutor(max_workers=max_workers) function. Ideally, the max_workers variable should be set to the appropriate amount of parallelism or to the number of available CPU cores.

Executor.map(df['image_path'], resize_image_array).Using the thread pool executor for concurrent execution, the resize_image_array function is applied to each image path in the DataFrame (df['image_path']) by calling the tolist()) method call. The map method creates

an iterator over the results and divides the workload across the worker threads. Image_arrays = list(...) creates a list called image_arrays from the iterator of resized image arrays. The outcomes of concurrent execution are combined in this stage and presented in a comprehensible list manner.

Lastly, the resized image arrays are put into the original DataFrame (df) as a new column called "image." When working with a large number of photos, this ThreadPoolExecutor-based method optimises the image scaling process. Opening, resizing, and converting photos to arrays are handled by the resize_image_array function, and concurrent execution makes sure that these tasks are completed effectively in parallel.

In conclusion, the procedure entails manipulating images using the Pillow or PIL library's Image class and then converting them into NumPy arrays using np.asarray so that machine learning algorithms can process them further.

3.8.1 Resizing the images

A single-step image scaling process is applied in our own CNN model, precisely adjusting the image dimensions to 224x224 pixels. The architecture of our model relies heavily on this resizing step, which guarantees compatibility and makes feature extraction from the input photos easier.

Consistency in the input data is preserved by shrinking photographs to 224x224 pixels only. This makes it possible for our neural network to pick up on subtleties and complex patterns found in the images. This standardised dimension maximises computing efficiency during training while simultaneously complying with our model's requirements.

As a result, only scaling photos to 224x224 pixels was concentrated on by our resizing technique, reflecting a conscious choice to prioritize model performance and resource efficiency during the training process.

The special features of our model architecture can be accommodated by our bespoke proposed CNN model. while integrating pre-trained features with ease thanks to this two-tiered scaling method. It displays a deliberate approach to preprocessing data, with each

resizing step selected with care to maximize the model's overall capabilities and performance, supporting our deep learning efforts.

3.9 Data Augmentation Techniques

For the deep convolutional models, to perform well without overfitting, a substantial amount of pictures must be used during training in this work. However, there aren't enough photos in our dataset. Once more, the system will be more robust the more variety there is in the training data. Data augmentation was employed to boost both the total number of training photos and the variance within those images. Additionally, this lessens the overfitting issue and improves prediction performance (Rahman, Hossain et al. 2021). We employed rotation, flipping, shearing, and zooming as the augmentation functions. The angle of random rotation was between 0 and 20 degrees. There was a 0.2 random zooming range. The shearing range was 0.2, meaning that one axis would lengthen 20% while the other would remain intact. It is extremely improbable that the legion will disappear from the image due to this slight degree of rotation, shearing, and zooming because it is located in the center of the image in both datasets. A few arbitrary enhanced pictures taken from the training dataset, Table 3. 2 Show the augmentation parameters.

Table 3. 2 Parameters for images augmentation

Parameters	value	Function
rotation_range	20	defines the range of degrees that can be used to rotate the photos at random.
width_shift_range	0.2	shows the range for adjusting the image's width at random.
height_shift_range	0.2	gives the range for the image's height to be changed at random.
shear_range	0.2	establishes the parameters for the image's shearing modifications.
zoom_range	0.2	gives the range within which the image can be randomly zoomed into or out of.
horizontal_flip	True	decides whether to flip photos horizontally at random.
fill_mode	Nearest	describes the filling process for newly produced pixels following transformations such as shifts in height or width.

3.9.1 Data balancing procedure

During the preprocessing phase, data augmentation approaches were used to correct class imbalances in the ISIC dataset. Specifically, a number of transformations were applied, such as rotation within a 20-degree range, horizontal flipping, shearing up to 20 degrees, and shifts in width and height up to 20% of the image's dimensions. By broadening the training set, these augmentation techniques are known to improve the model's ability to generalise across different kinds of skin lesions. Following augmentation, it was ensured that every category had a total of 1000 photographs in order to carefully balance each class. By avoiding bias towards overrepresented classes, this balanced dataset helps ensure that each skin condition is represented more fairly during model training. By exposing the model to a broader range of augmented photos, the augmentation procedure not only corrects class imbalances but also improves its robustness and performance by teaching it more comprehensive characteristics that enable effective classification across all skin conditions. The information about data augmentation and balancing shown in Table 3. 3.

Table 3. 3 Dataset details after Augmentation and balancing

ISIC	Train images	Validation images	Test images	Total images
actinic keratosis	631	167	202	1000
basal cell carcinoma	631	167	202	1000
Dermatofibroma	634	163	203	1000
Melanoma	649	149	202	1000
Nevus	633	164	203	1000
pigmented benign keratosis	647	153	200	1000
seborrheic keratosis	660	153	187	1000
squamous cell carcinoma	645	164	191	1000
vascular lesion	645	164	204	1000

3.10 Hardware and Software

During the training of my model, I utilized the following hardware and software resources:

Local Machine Specifications

Processor: 12th Gen Intel(R) Core(TM) i7-12650H 2.30 GHz

Storage: 500GB SSD NVMe

Memory: 16 GB DDR4

Graphics: NVIDIA GeForce RTX 3050 Laptop GPU

Online Resources :The Kaggle online notebook environment was utilized, specifically leveraging the GPU P-100 for efficient model training and experimentation.

Software: Jupyter Notebook was used for developing and testing the model, allowing for interactive coding and easy visualization of results. These resources provided a robust environment for the successful training and evaluation of my model.

4 PROPOSED METHOD

The technique for handling the ISIC (International Skin Imaging Collaboration) dataset is depicted in the diagram. The ISIC Dataset, which includes pictures of skin lesions, is where the process starts. Data preprocessing is the first set of operations to clean and prepare the data, including image resizing, normalisation, and noise or unnecessary information removal. The term "data augmentation" describes methods for synthesising new images into the dataset through manipulations such as flips, zooms, and rotations. An even distribution of classes within the dataset is guaranteed by data balancing. Next, the dataset is divided into test, validation, and training sets. The convolutional neural network created especially for this research is called the Proposed CNN Model. The dataset is also subjected to a variety of Pre-trained CNN Models, both in their unaltered and altered versions. These pre-trained models can be adjusted for the current task because they have already undergone extensive training on big datasets. The diagram illustrates how models are trained, validated, and tested iteratively. It also shows how data augmentation and balance are used to improve model performance. The architecture of the proposed method given in the Figure 4.1.

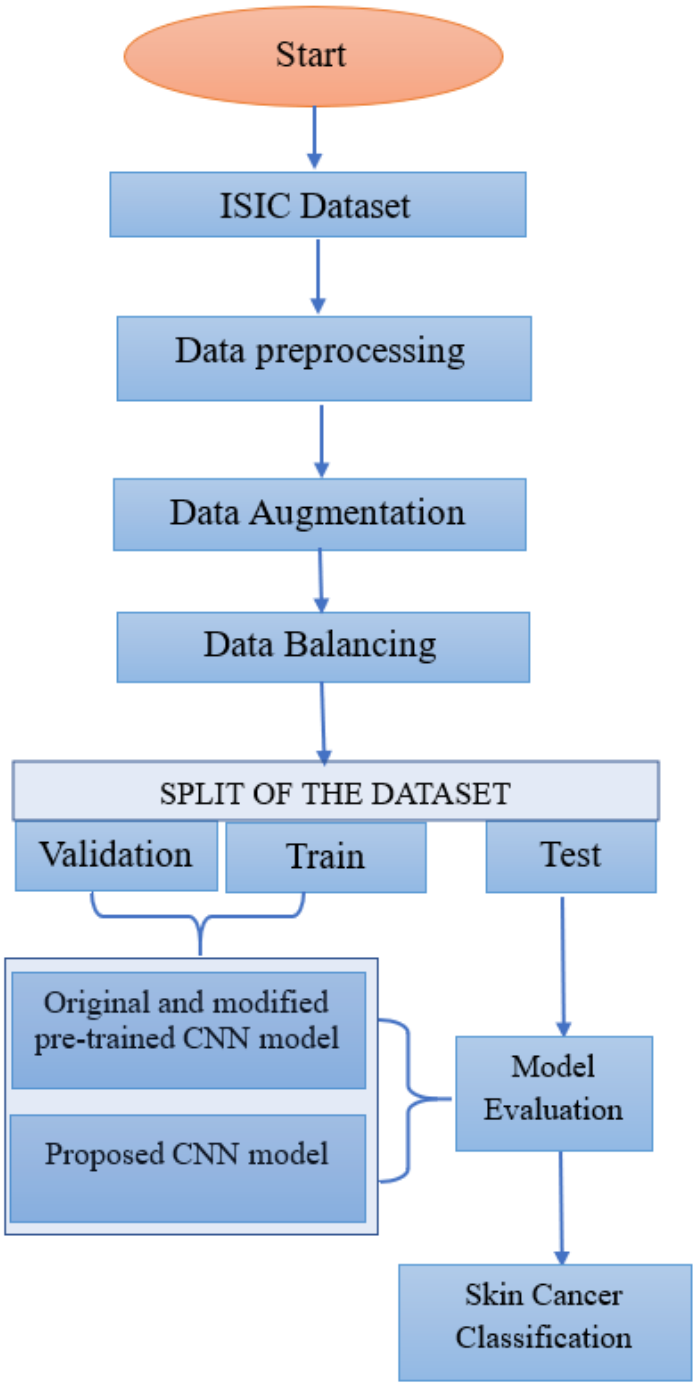

Figure 4.1.Proposed Method diagram

4.1 Proposed CNN Model

Convolutional neural networks (CNNs) like the one in this our model are made for image categorization applications. The first layer is an image-specific input layer that takes 224x224 pixel and 128x128 inputs with three RGB color channels. Convolutional, pooling, normalization, and dense layers make up the next layers, which are followed by a SoftMax output layer for multi-class classification. The convolutional layers use filters with sizes of 32, 64, and 128 in that order. To downsample the spatial dimensions, each filter is followed by a max-pooling layer. To stabilize and normalize the activations between layers, batch normalization layers are interspersed.Following the last max-pooling layer, two dense layers of 128 and 64 neurons, respectively, using ReLU activation functions, come next. A Global Average Pooling layer then fixes the spatial dimensions. After every dense layer, there are dropout layers with a rate of 0.3 to reduce overfitting during training. Assuming a nine-class classification task, the output layer's nine neurons use the SoftMax activation function to provide probability distributions over the nine classes. The model is constructed to reduce categorical cross-entropy loss while keeping an eye on accuracy as the assessment metric during training. It employs the RMSprop and Adam optimizer with a learning rate of 0.001. With convolutional processes, this architecture seeks to extract hierarchical features from input images, progressively reducing spatial dimensions, and ultimately generating classification probabilities for the target classes.With dropout to prevent overfitting during training, this CNN architecture combines convolutional, pooling, normalizing, and thick layers to learn hierarchical features from the input images. Ultimately, it is intended for image classification problems with nine classes.Batch normalization, dropout for regularization, convolutional layers, pooling layers, and other standard components for image classification applications are utilized in our architecture.Prior to the fully linked layers, the global average pooling layer reduces spatial dimensions, and the output layer's softmax activation makes multi-class classification easier. With the designated optimizer, assessment metric, and loss function, the model is assembled for training the Table 4. 1.shows the parameters of the Proposed CNN model, and Figure 4. shows the summary of the Proposed CNN model.

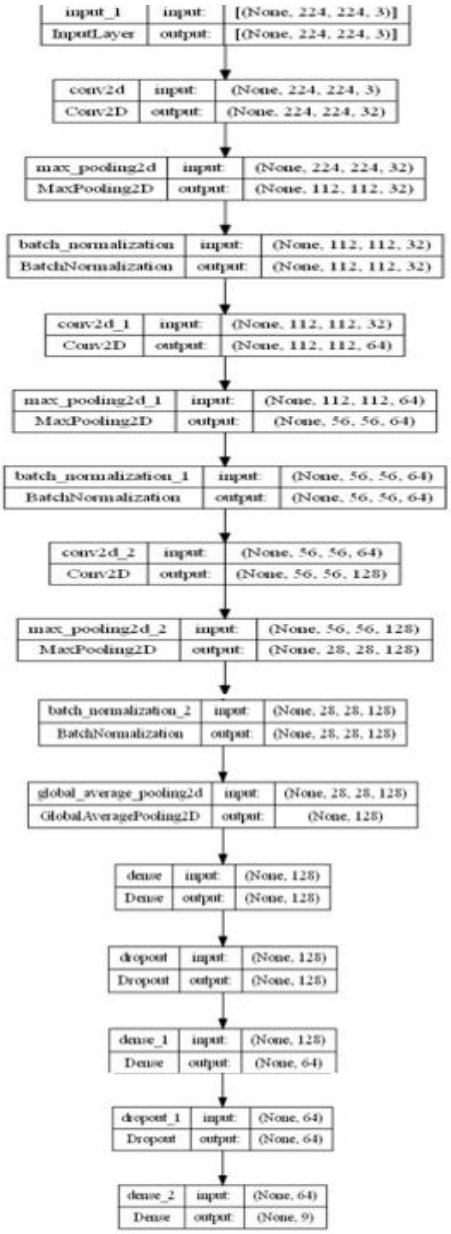

Figure 4.2 The proposed CNN Model architecture

Table 4. 1 The Parameters of the proposed CNN model

Layer name	**Output shape**	**Filter No.**	**Filter size**	**Stride No.**	**Learnable properties**	**No. of Parameters**
Input Layer	(224, 224, 3)	N/A	N/A	N/A	None	0
Conv2D Layer 1	(224, 224, 32)	32	(3, 3)	(1, 1)	Weights and biases	896
MaxPooling2D Layer 1	(112, 112, 32)	N/A	(2, 2)	(2, 2)	None	0
BatchNormalization 1	(112, 112, 32)	N/A	N/A	N/A	Scale and shift (gamma, beta)	128
Conv2D Layer 2	(112, 112, 64)	64	(3, 3)	(1, 1)	Weights and biases	18,496
MaxPooling2D Layer 2	(56, 56, 64)	N/A	(2, 2)	(2, 2)	None	0
BatchNormalization 2	(56, 56, 64)	N/A	N/A	N/A	Scale and shift (gamma, beta)	256
Conv2D Layer 3	(56, 56, 128)	128	(3, 3)	(1, 1)	Weights and biases	73,856
MaxPooling2D Layer 3	(28, 28, 128)	N/A	(2, 2)	(2, 2)	None	0
BatchNormalization 3	(28, 28, 128)	N/A	N/A	N/A	Scale and shift (gamma, beta)	512
GlobalAveragePooling2D	(128)	N/A	N/A	N/A	None	0
Dense Layer 1	(128)	N/A	N/A	N/A	Weights and biases	16,512
Dropout Layer 1	(128)	N/A	N/A	N/A	None	0
Dense Layer 2	(64)	N/A	N/A	N/A	Weights and biases	8,256
Dropout Layer 2	(64)	N/A	N/A	N/A	None	0
Output Layer	(9)	N/A	N/A	N/A	Weights and biases	585
Total params: 119,497 Trainable params: 119,049 Non-trainable params: 448						

4.2 Pretrained CNN model

4.2.1 Original pretrained model

The original pretrained models (VGG16, ResNet50, AlexNet, and InceptionV3) were used to solve a specific nine-class classification problem. The output layer was modified to fit the requirements of the classification challenge.

4.2.2 Modified pretrained model

4.2.2.1 VGG16

For feature extraction, a modified version of the original pre-trained VGG16 was used in the model's development. The performance of this pre-trained VGG16 model was improved for the particular task by adding more layers and fine-tuning it to better fit the dataset and project goals, Table 4.2 Show the summary of the modified VGG16 model parameters.

Table 4.2. Model summary for Modified pre-trained VGG16 convolutional basis for feature extraction and is a proprietary neural network architecture

Layer name	Output shape	Filter No.	Filter size	Stride No.	Learnable properties	No. of Parameters
input_1 (InputLayer)	(None, 224, 224, 3)	N/A	N/A	N/A	None	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	64	(3, 3)	(1, 1)	Weights and biases	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	64	(3, 3)	(1, 1)	Weights and biases	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	N/A	(2, 2)	(2, 2)	None	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	128	(3, 3)	(1, 1)	Weights and biases	73856

block2_conv2 (Conv2D)	(None, 112, 112, 128)	128	(3, 3)	(1, 1)	Weights and biases	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	N/A	(2, 2)	(2, 2)	None	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	256	(3, 3)	(1, 1)	Weights and biases	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	256	(3, 3)	(1, 1)	Weights and biases	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	256	(3, 3)	(1, 1)	Weights and biases	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	N/A	(2, 2)	(2, 2)	None	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	512	(3, 3)	(1, 1)	Weights and biases	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	512	(3, 3)	(1, 1)	Weights and biases	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	512	(3, 3)	(1, 1)	Weights and biases	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	N/A	(2, 2)	(2, 2)	None	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	512	(3, 3)	(1, 1)	Weights and biases	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	512	(3, 3)	(1, 1)	Weights and biases	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	512	(3, 3)	(1, 1)	Weights and biases	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	N/A	(2, 2)	(2, 2)	None	0
global_average_pooling2d (GlobalAveragePooling2	(None, 512)	N/A	N/A	N/A	None	0

D)						
dense (Dense)	(None, 1024)	N/A	N/A	N/A	Weights and biases	525312
batch_normalization (BatchNormalization)	(None, 1024)	N/A	N/A	N/A	Scale and shift (gamma, beta)	4096
dropout (Dropout)	(None, 1024)	N/A	N/A	N/A	None	0
dense_1 (Dense)	(None, 512)	N/A	N/A	N/A	Weights and biases	524800
batch_normalization_1 (BatchNormalization)	(None, 512)	N/A	N/A	N/A	Scale and shift (gamma, beta)	2048
dropout_1 (Dropout)	(None, 512)	N/A	N/A	N/A	None	0
dense_2 (Dense)	(None, 256)	N/A	N/A	N/A	Weights and biases	131328
batch_normalization_2 (BatchNormalization)	(None, 256)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1024
dropout_2 (Dropout)	(None, 256)	N/A	N/A	N/A	None	0
dense_3 (Dense)	(None, 9)	N/A	N/A	N/A	Weights and biases	2313
Total params: 15905609 (60.68 MB) Trainable params: 8266761 (31.54 MB) Non-trainable params: 7638848 (29.14 MB)						

This model takes input images of the shape (224, 224, 3) and uses VGG16 to extract features. Custom dense layers are then applied for additional feature refinement, and the input images are finally classified into one of nine classes.

A synopsis of the model architecture is shown, giving information about the number of trainable parameters, output forms, layer kinds, and structure.

The CNN model was trained for 100 epochs with a batch size of 32 during the training phase. The model processed 32 samples in each training step before changing its weights, and it went through the full training dataset 100 times. The purpose of selecting these hyper parameters was to maximize computing efficiency while guaranteeing that the model was fu lly exposed to the training data.

4.2.2.2 AlexNet

To enhance our work, this model makes use of an altered version of AlexNet, a foundational model in the deep learning field. AlexNet, a key contributor to the advancement of deep learning, is made up of convolutional layers that take features out of the input and greatly improve picture classification tasks. Customised AlexNet to better meet unique application requirements. The Table 4.3 shows an explanation of this updated architecture's layers and various components.

Table 4.3. Model summary for a customized convolutional neural network (CNN) architecture that was motivated by the deep learning industry's seminal model, AlexNet

Layer name	Output shape	Filter No.	Filter size	Stride No.	Learnable properties	No. of Parameters
Input Layer	(None, 224, 224, 3)	N/A	N/A	N/A	None	0
Conv2D Layer 1	(None, 54, 54, 96)	96	(11, 11)	(4, 4)	Weights and biases	34944
BatchNormalization 1	(None, 54, 54, 96)	N/A	N/A	N/A	Scale and shift (gamma, beta)	384

Activation 1	(None, 54, 54, 96)	N/A	N/A	N/A	None	0
MaxPooling2D Layer 1	(None, 26, 26, 96)	N/A	(3, 3)	(2, 2)	None	0
Conv2D Layer 2	(None, 26, 26, 256)	256	(5, 5)	(1, 1)	Weights and biases	614656
BatchNormalization 2	(None, 26, 26, 256)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1024
Activation 2	(None, 26, 26, 256)	N/A	N/A	N/A	None	0
MaxPooling2D Layer 2	(None, 12, 12, 256)	N/A	(3, 3)	(2, 2)	None	0
Conv2D Layer 3	(None, 12, 12, 384)	384	(3, 3)	(1, 1)	Weights and biases	885120
BatchNormalization 3	(None, 12, 12, 384)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1536
Activation 3	(None, 12, 12, 384)	N/A	N/A	N/A	None	0
Conv2D Layer 4	(None, 12, 12, 384)	384	(3, 3)	(1, 1)	Weights and biases	1327488
BatchNormalization 4	(None, 12, 12, 384)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1536
Activation 4	(None, 12, 12, 384)	N/A	N/A	N/A	None	0
Conv2D Layer 5	(None, 12, 12, 256)	256	(3, 3)	(1, 1)	Weights and biases	884992

BatchNormalization 5	(None, 12, 12, 256)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1024
Activation 5	(None, 12, 12, 256)	N/A	N/A	N/A	None	0
MaxPooling2D Layer 3	(None, 5, 5, 256)	N/A	(3, 3)	(2, 2)	None	0
Flatten Layer	(None, 6400)	N/A	N/A	N/A	None	0
Dense Layer 1	(None, 4096)	N/A	N/A	N/A	Weights and biases	26218496
BatchNormalization 6	(None, 4096)	N/A	N/A	N/A	Scale and shift (gamma, beta)	16384
Activation 6	(None, 4096)	N/A	N/A	N/A	None	0
Dropout 1	(None, 4096)	N/A	N/A	N/A	None	0
Dense Layer 2	(None, 4096)	N/A	N/A	N/A	Weights and biases	16781312
BatchNormalization 7	(None, 4096)	N/A	N/A	N/A	Scale and shift (gamma, beta)	16384
Activation 7	(None, 4096)	N/A	N/A	N/A	None	0
Dropout 2	(None, 4096)	N/A	N/A	N/A	None	0
Dense Layer 3	(None, 9)	N/A	N/A	N/A	Weights and biases	36873

BatchNormalization 8	(None, 9)	N/A	N/A	N/A	Scale and shift (gamma, beta)	36
Activation 8	(None, 9)	N/A	N/A	N/A	None	0
Total params: 46,822,189 (178.61 MB) Trainable params: 46,803,035 (178.54 MB) Non-trainable params: 19,154 (74.82 KB)						

In conclusion, this model has numerous convolutional and fully connected layers and is structured similarly to the AlexNet architecture. Dropout is used for regularization, along with batch normalization. Class probabilities are generated by the output layer, and the Adam optimizer with categorical crossentropy loss is used to train the model. With an emphasis on feature extraction using convolutional layers and subsequent classification via dense layers, this architecture is intended for image classification problems.

The model was trained for 100 epochs with a batch size of 32 during the training phase. The model processed 32 samples in each training step before changing its weights, and it went through the full training dataset 100 times. The purpose of selecting these hyperparam eters was to maximize computing efficiency while guaranteeing that the model was fully exposed to the training data.

4.2.2.3 ResNet

The modified version of ResNet50was used, a convolutional neural network (CNN) architecture that is well-known for its deep residual learning framework, as a feature extractor. Pre-trained on ImageNet, ResNet50 is built with 50 layers to efficiently address the vanishing gradient issue, enabling the creation of extremely deep networks.The changes to ResNet50 improve its capacity to extract and handle complex features unique to our use case. The following is an explanation of the layers and different components of this updated mode. This model has nine classes and is based on ResNet50, with additional unique

classification layers added for particular classification tasks. The remaining layers of the base ResNet50 model are frozen, however the final 10 layers can be trained for fine-tuning Table 4.4 show the model summary of modified ResNet50 architecture.

Table 4.4.Model summary of convolutional neural network (CNN) using modified ResNet50 architecture that has been pre-trained on ImageNet

Layer name	**Output shape**	**Filter No.**	**Filter size**	**Stride No.**	**Learnable properties**	**No. of Parameters**
Input Layer	(None, 224, 224, 3)	N/A	N/A	N/A	None	0
ResNet50 Base Model	(None, 7, 7, 2048)	N/A	N/A	N/A	Pre-trained weights	~23,587,712
GlobalAveragePooling2D Layer	(None, 2048)	N/A	N/A	N/A	None	0
Dense Layer 1	(None, 512)	N/A	N/A	N/A	Weights and biases	1,049,088
BatchNormalization 1	(None, 512)	N/A	N/A	N/A	Scale and shift (gamma, beta)	2,048
Dense Layer 2	(None, 256)	N/A	N/A	N/A	Weights and biases	131,328
Dropout 1	(None, 256)	N/A	N/A	N/A	None	0
BatchNormalization 2	(None, 256)	N/A	N/A	N/A	Scale and shift (gamma, beta)	1,024
Dropout 2	(None, 256)	N/A	N/A	N/A	None	0
Dense Layer 3	(None, 128)	N/A	N/A	N/A	Weights and biases	32,896
BatchNormalization 3	(None, 128)	N/A	N/A	N/A	Scale and shift (gamma, beta)	512

Dropout 3	(None, 128)	N/A	N/A	N/A	None	0
Dense Layer 4 (Output)	(None, 9)	N/A	N/A	N/A	Weights and biases	1,161
Activation (Softmax)	(None, 9)	N/A	N/A	N/A	None	0
Total params: 24811574 (94.65 MB) Trainable params: 24756644 (94.44 MB) Non-trainable params: 54930 (214.57 KB)						

The model was trained for 100 epochs with a batch size of 32 during the training phase. The model processed 32 samples in each training step before changing its weights, and it went through the full training dataset 100 times. The purpose of selecting these hyperparameters was to maximize computing efficiency while guaranteeing that the model was fully exposed to the training data.

4.2.2.4 InceptionV3

The modified InceptionV3 convolutional neural network architecture with parameters pre-trained on the ImageNet dataset is utilized in this model. The Table 4.5 show a brief synopsis of every component.The model's architecture is summarised, including information on each layer and the total number of parameters.

Overall, this model takes advantage of InceptionV3's tremendous feature extraction capabilities while also including bespoke thick layers for classification that are customised to a 9-class scenario.

The modified InceptionV3 model was trained for 100 epochs with a batch size of 32 during the training phase. The model processed 32 samples in each training step before changing its weights, and it went through the full training dataset 100 times. The purpose of selecting these hyperparameters was to maximize computing efficiency while guaranteeing that the model was fully exposed to the training data.

Table 4.5.Model summary InceptionV3 model is instantiated with pre-trained weights from ImageNet.

Layer Name	Output Shape	Filter No.	Filter Size	Stride No.	Learnable Properties	No. of Parameters
Input Layer	(None, 224, 224, 3)	N/A	N/A	N/A	None	0
InceptionV3 Base	(None, 5, 5, 2048)	N/A	N/A	N/A	Pre-trained weights	21,802,784
Flatten	(None, 51200)	N/A	N/A	N/A	None	0
Dropout 1	(None, 51200)	N/A	N/A	N/A	None	0
Dense Layer 1	(None, 512)	N/A	N/A	N/A	Weights and biases	26,214,912
Dropout 2	(None, 512)	N/A	N/A	N/A	None	0
Dense Layer 2	(None, 512)	N/A	N/A	N/A	Weights and biases	262,656
Dense Layer 3	(None, 9)	N/A	N/A	N/A	Weights and biases	4,617
Output Layer	(None, 9)	N/A	N/A	N/A	Softmax activation	0
Total params: 48,284,969 (184.19 MB) Trainable params: 48,250,537 (184.06 MB) Non-trainable params: 34,432 (134.50 KB)						

5 RESULT AND DISCUSSION

This section demonstrates the efficiency of the proposed CNN model in identifying each type of skin cancer using the ISIC dataset, which contains nine types of skin cancer.

5.1 Proposed CNN Model Result

For the purpose of the performance analysis, the dataset is split into three sets, one of which is the training set that is used to train the model. To prevent overfitting, the model's hyperparameters are adjusted using the validation set. And lastly, the testing set, which is used to evaluate the overall performance of the model. The training loss and validation loss gradually decreasing across the epochs indicates that the model is learning and generalizing successfully. After epoch 100, the validation loss stops decreasing, suggesting that the model has reached its peak performance. It is important to keep in mind that the appropriate number of epochs can vary depending on the size of the dataset and the complexity of the model.
The code snippet used the ReduceLROnPlateau class from the Keras package to construct a learning rate decrease callback. This callback is used to dynamically modify the learning rate when a neural network model is being trained. The callback instantiation specifies several parameters, such as the monitored metric ('val_loss'), a verbose setting of 1 to display information about the learning rate reduction, a reduction factor of 0.5 to indicate that the learning rate will be multiplied by this factor when triggered, and a minimum learning rate of 0.001 to prevent it from decreasing further. The patience value of 3 epochs indicates the number of epochs without improvement after which the learning rate will be reduced.

The fit approach was then used to train the model. The training process has a batch size of 32 and takes place across 100 epochs. The model is trained using the training data (x_train, y_train), and its performance is tracked during training using the validation data (x_validate, y_validate). By adding the ReduceLROnPlateau callback to the fit method's callbacks parameter, it is included into the training procedure. This guarantees that the learning rate is modified in accordance with the given parameters, which may lead to more efficient training and possible enhancements in model performance.

Figure 5. 1 shows the Proposed CNN models with image size (224*224) training and validation accuracy over a range of epochs, and Figure 5. 2 shows the model's loss analyses over the same number of epochs and Figure 5. 3 show the confusion matrix, in Table 5.1.the result of the trained proposed CNN model was displayed.

Table 5.1.Result of the Proposed CNN model with image size (224*224)

Class	**metrics**	**Accuracy**	**Precision**	**Recall**	**F1-score**
1	actinic keratosis	83.84%	0.76	0.91	0.83
2	basal cell carcinoma	88.33%	0.93	0.83	0.88
3	dermatofibroma	92.97%	0.86	1.00	0.92
4	melanoma	64.70%	0.80	0.49	0.61
5	nevus	77.03%	0.93	0.61	0.74
6	pigmented benign keratosis	80.17%	0.81	0.79	0.80
7	seborrheic keratosis	84.52%	0.69	1.00	0.82
8	squamous cell carcinoma	84.53%	0.83	0.87	0.84
9	vascular lesion	98.67%	0.98	1.00	0.99
Overall values		83%	84%	83%	82%

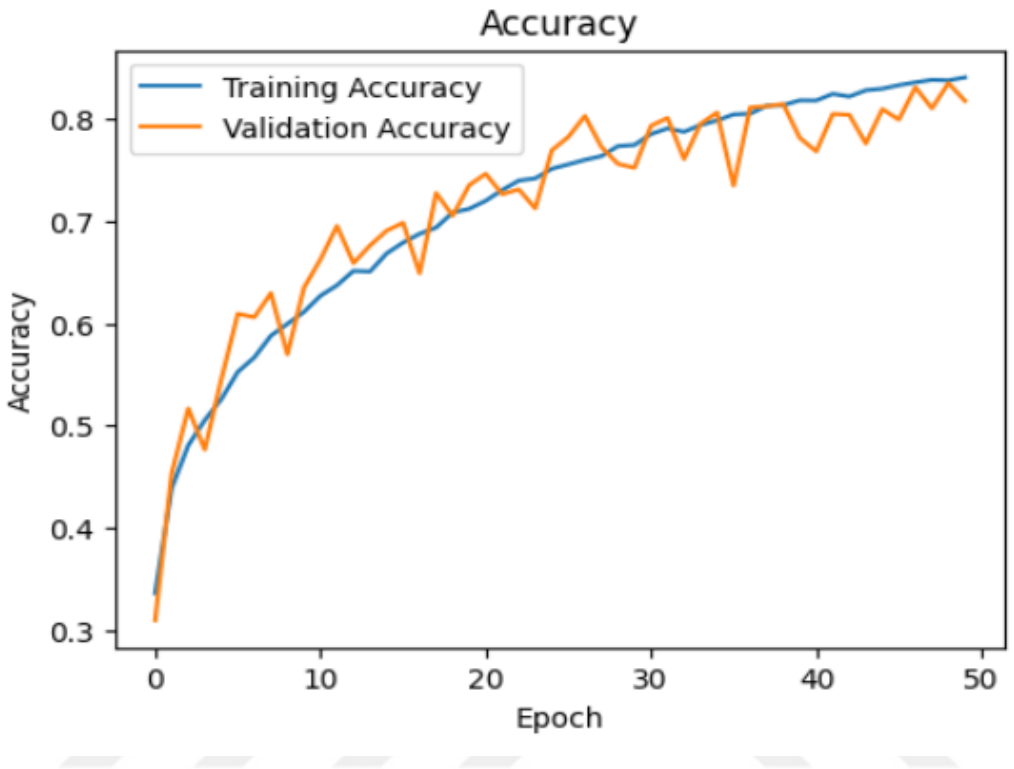

Figure 5. 1 This figure displays the classification accuracy during training and validation for the proposed CNN model with 224 by 224 image size across each training session.

In the Figure 5. 1 There are two plotted lines:

Training Accuracy: The blue line shows the accuracy attained on the training dataset at the end of each epoch.

Validation Accuracy: The orange line shows the accuracy attained on the validation dataset at the end of each session.

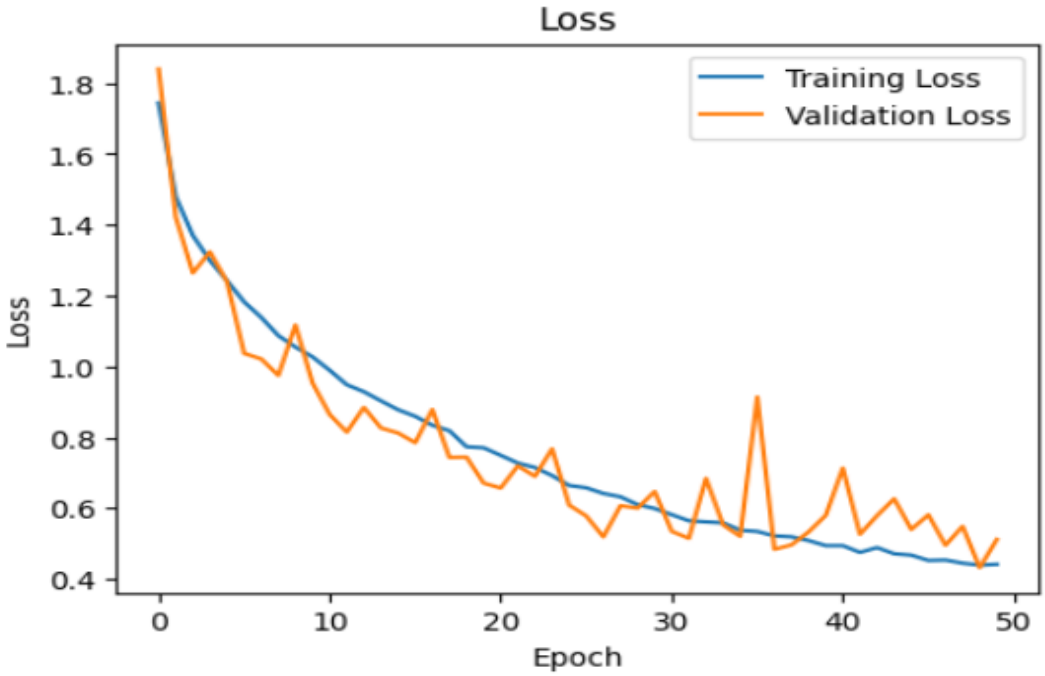

Figure 5. 2 The classification proposed CNN model's with 224 by 224 picture size training and validation losses throughout each training epoch are shown in this figure.

In the Figure 5. 2 There are two plotted lines:

Training Loss: The blue line shows the loss on the training dataset at the end of each epoch.

Validation Loss: The orange line shows the loss on the validation dataset at the end of each epoch.

Several metrics, including precision, F1-score, recall, and accuracy, have been used to assess the model's performance. The findings show that the accuracy performance of the Proposed CNN model was 83.22%. It implies that the model identified different kinds of skin lesions reasonably effectively.

All of these assessment measures suggest that the model performs admirably in skin lesion picture classification following training on the ISIC dataset. It demonstrates an excellent mix between reducing false positives and false negatives (balanced recall) and accurately recognizing various types of skin lesions (high accuracy). It's important to remember that when analyzing these indicators, the application's particular requirements and domain expertise should be taken into account. Additional examination, encompassing domain-specific validation and potentially examining cases that were incorrectly classified, may offer more profound understanding of the model's merits and demerits in identifying skin lesions.

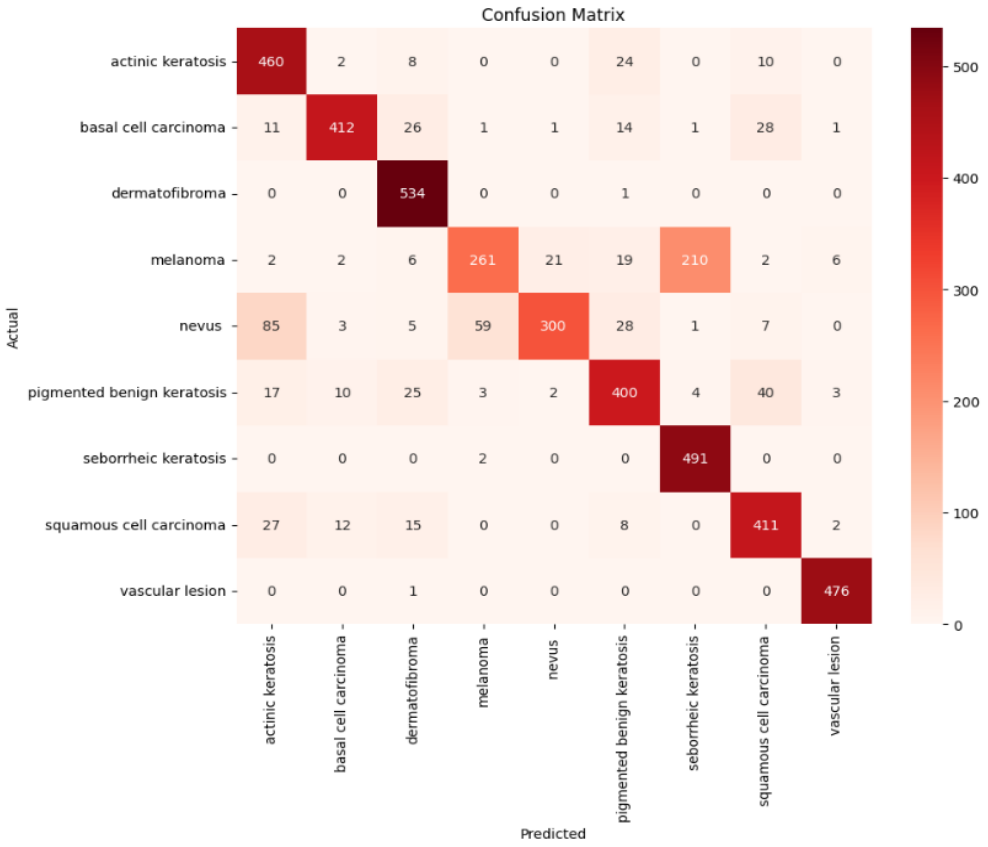

Figure 5. 3 This figure is shown the confusion matrix classification of proposed model's with image size (224*224) which shows how well the model differentiates between the nine classes.

In the matrix show in the Figure 5. 3, the instances of a predicted class are represented by each column, and the instances of an actual class are represented by each row. The number of successfully categorized cases for each class is represented by the diagonal elements, while the incorrectly classified instances are represented by the off-diagonal elements.
Higher values off the diagonal indicate classes where the model tends to make more mistakes, and higher values along the diagonal indicate classes where the model performs better. These can be identified with the aid of the confusion matrix. This thorough analysis is essential for comprehending the model's performance inside each class and for informing future model enhancements.

Table 5.2 illustrates that the accuracy of the suggested CNN model rose to 86% after the image size was reduced to 128x128 pixels.

Table 5.2.Result of proposed CNN model with image size (128*128)

Class	**metrics**	**Accuracy**	**Precision**	**Recall**	**F1-score**
1	actinic keratosis	85.89%	0.79	0.93	0.85
2	basal cell carcinoma	89.38%	0.87	0.92	0.89
3	dermatofibroma	97.70%	0.99	0.96	0.98
4	melanoma	74.51%	0.75	0.74	0.75
5	nevus	78.40%	0.85	0.72	0.78
6	pigmented benign keratosis	85.61%	0.82	0.90	0.85
7	seborrheic keratosis	84.52%	0.82	0.87	0.85
8	squamous cell carcinoma	89.01%	0.95	0.83	0.89
9	vascular lesion	98.04%	1.00	0.96	0.98
Overall values		86%	87%	86%	86%

In terms of the Proposed CNN model in image 128x128 Figure 5.5 and Figure 5.4 show the accuracy and loss of the Proposed CNN model from the ISIC dataset, respectively and Figure 5.6 represents the confusion matrix of the Proposed CNN model.

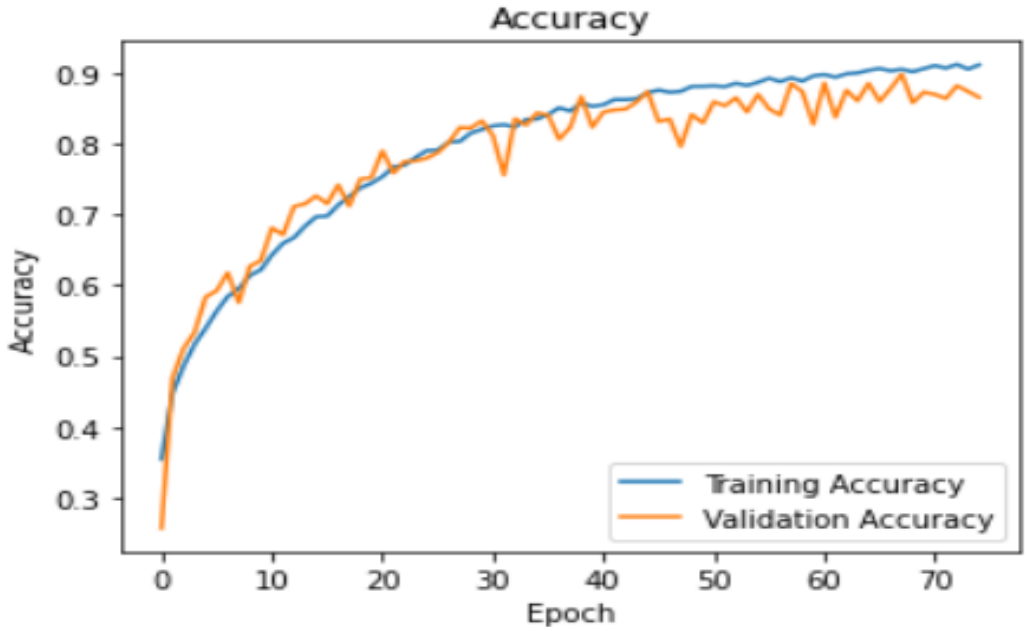

Figure 5.4 The Accuracy convergence graph obtained by the proposed CNN model with image size (128 by 128) during the training phase

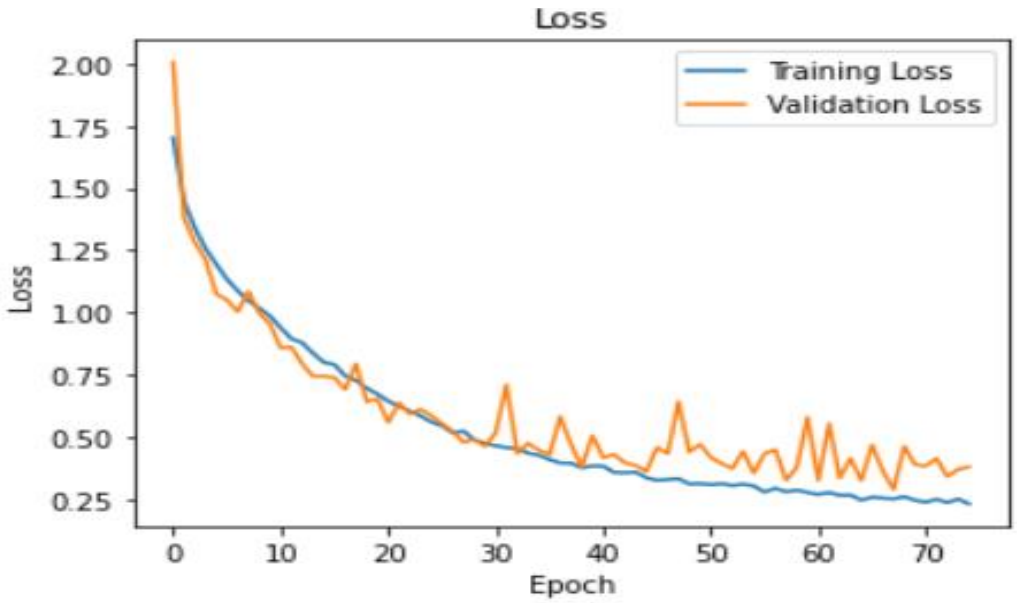

Figure 5.5 The Loss convergence graph obtained by the Proposed CNN model with image size (128 by 128) during the training phase

Figure 5.6. represents the confusion matrix of the Proposed CNN model from the ISIC dataset, which shows the model's performance.

Figure 5.6 The confusion matrix of the Proposed CNN model with image size (128 by 128)

5.2 Original Popular CNN models Result

A number of well-known Convolutional Neural Network (CNN) models was trained, including InceptionV3, AlexNet, ResNet, and VGG16, on the ImageNet dataset in order to tackle the classification task. For 9-class challenge, the early findings did not yield great accuracy, even with their outstanding feature extraction skills. In order to address this, each of these models was modified before using them again for classification task. Specifically, more thick layers and dropout layers were added to enhance performance and avoid overfitting, Table 5. 3 show the Original popular CNN model accuracy.

Table 5. 3.Original popular CNN model accuracy

Original Model	Accuracy
AlexNet	73%
VGG16	64%
ResNet50	53%
InceptionV3	64%

5.3 Modified Popular CNN models Result

Following the application of AlexNet on the ISIC dataset, which features nine distinct skin states and 224*224 image sizes, the model's performance While recall gauges the model's capacity to capture every instance of a given class, precision shows the accuracy of positive predictions. The F1-score offers a fair evaluation of a model's performance since it is the harmonic mean of precision and recall. Overall, the model achieved a precision of 89%, a recall of 89%, and an F1-score of 88%. with an accuracy of 89%, the Modified AlexNet model performs well. For vascular lesions and dermatofibroma, it shows especially excellent precision and recall, suggesting robust identification of these disorders. Melanoma and nevus, on the other hand, perform marginally worse, indicating potential for improvement in the differentiation of these classifications. Demonstrating a performance that is equitable across all classes, the Table 5. 4 show the result of modified AlexNet model,and Figure 5.7,Figure 5.8 show the accuracy and loss of the modified AlexNet,while Figure 5.9 show the confusion matrix of modified AlexNet model.

Table 5. 4.Result of modified AlexNet pretrained model

Class	Metrics	Accuracy	Precision	Recall	F1-score
1	actinic keratosis	93.66%	0.95	0.92	0.94
2	basal cell carcinoma	75.72%	0.81	0.70	0.75
3	Dermatofibroma	99.52%	0.99	1.00	1.00
4	Melanoma	86.64%	0.84	0.90	0.87
5	Nevus	95.97%	0.95	0.97	0.96
6	pigmented benign keratosis	96.02%	0.98	0.94	0.96
7	seborrheic keratosis	86.16%	0.82	0.90	0.86
8	squamous cell carcinoma	97.60%	0.95	1.00	0.98
9	Vascular lesion	75.84%	0.77	0.75	0.76
Overall values		89%	89%	89%	88%

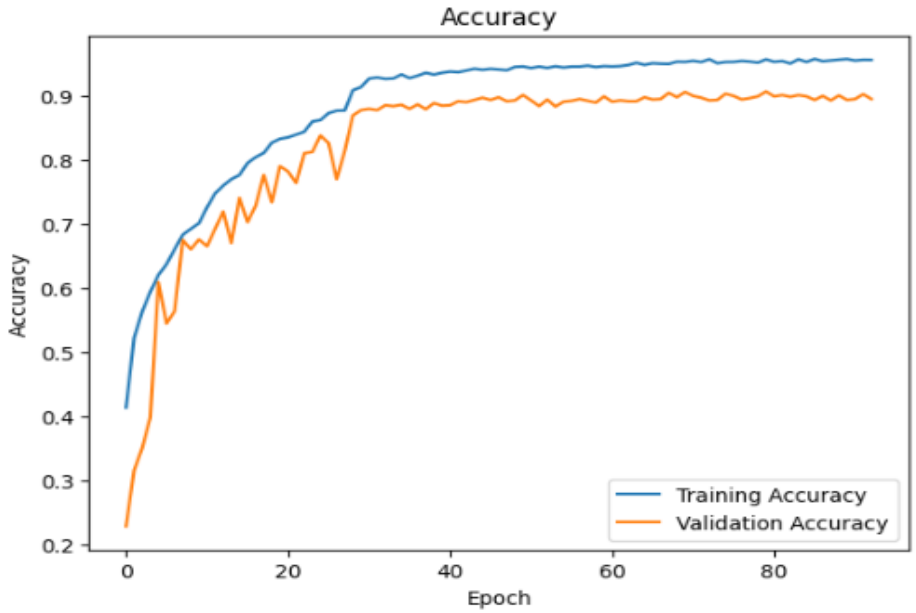

Figure 5.7 The modified AlexNet pretrained model's training and validation accuracy over each training period is depicted in this image.

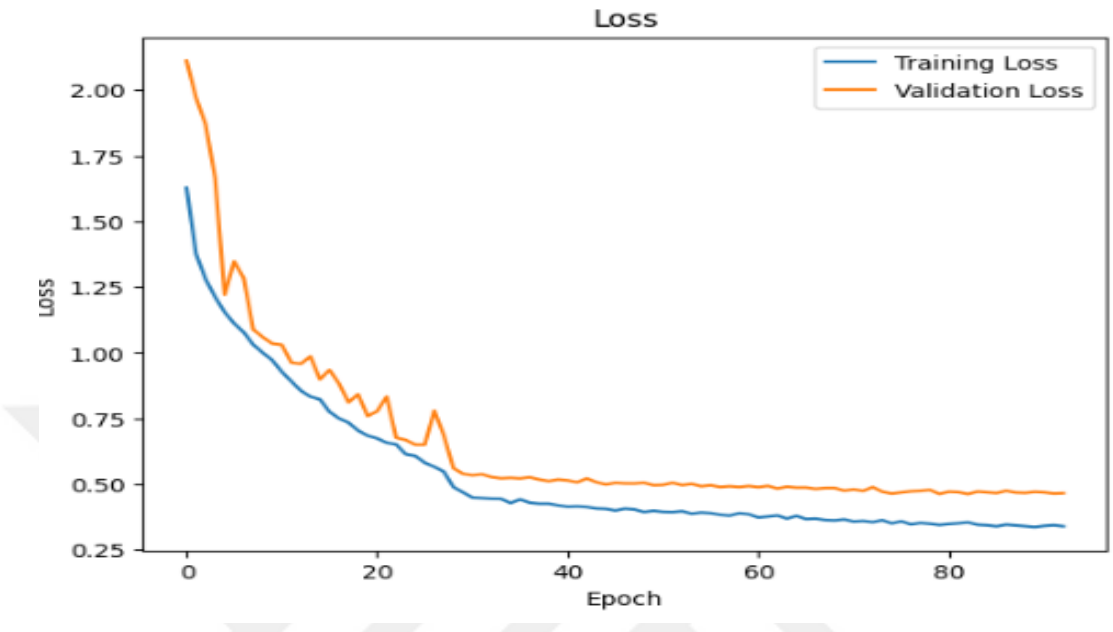

Figure 5.8 The modified AlexNet pretrained model's training and validation losses over each training period are depicted in this figure.

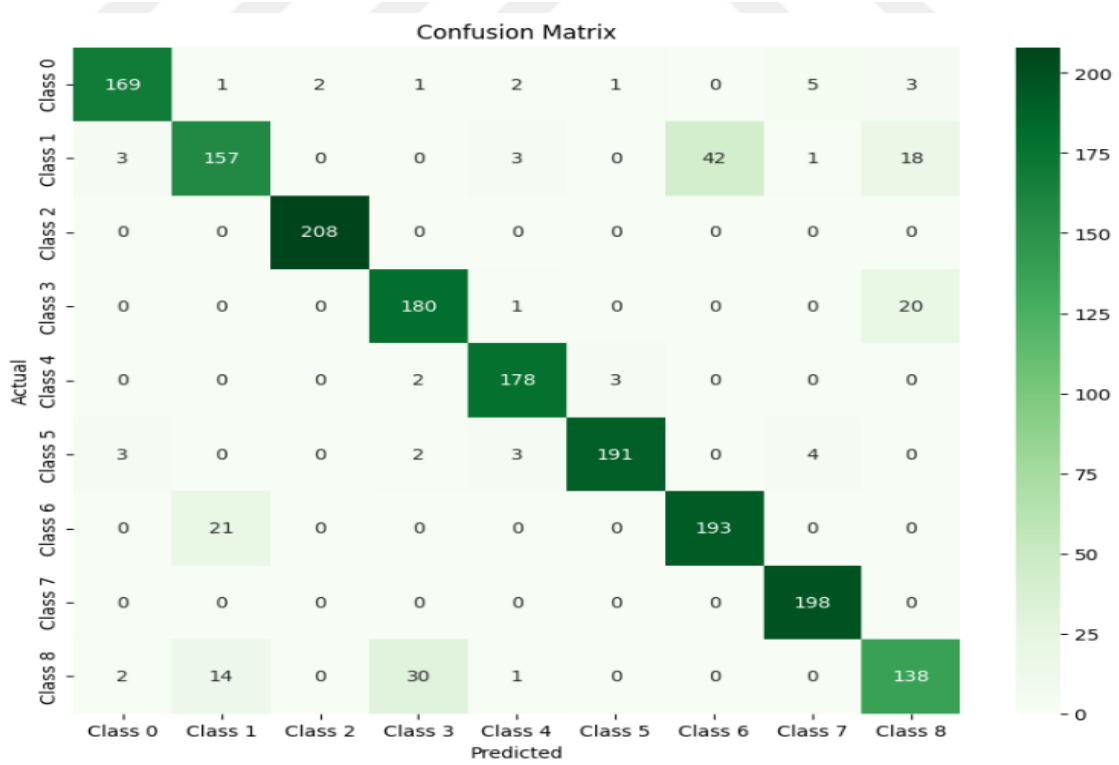

Figure 5.9 The confusion matrix for the updated AlexNet pretrained model is shown in this figure, which shows how well the model classified the nine distinct classes.

The performance and the result of the modified VGG16 pre-trained model during training is shown visually in Table 5. 5 and Figure 5.11, Figure 5.10,show the accuracy and the loss convergence of the modified VGG16 The accuracy trend is a good indicator of the model's learning curve and the Figure 5.10**.** show the confusion matrix for modified VGG16 model.

On the other hand, the loss curve shows how the model's inaccuracy decreased as it was trained. The model is approaching an ideal solution when the loss trend starts to decline.

Table 5. 5.Result of modified VGG16 pretrained model

Class	**metrics**	**Accuracy**	**Precision**	**Recall**	**F1-score**
1	actinic keratosis	86.35%	0.77	0.95	0.85
2	basal cell carcinoma	75.76%	0.69	0.82	0.75
3	dermatofibroma	98.99%	0.99	0.99	0.99
4	melanoma	83.46%	0.91	0.76	0.83
5	nevus	94.84%	0.93	0.96	0.95
6	pigmented benign keratosis	91.46%	0.98	0.85	0.91
7	seborrheic keratosis	82.12%	0.87	0.77	0.82
8	squamous cell carcinoma	96.66%	0.98	0.95	0.97
9	vascular lesion	74.70%	0.73	0.76	0.75
Overall Values		86%	87%	86%	84%

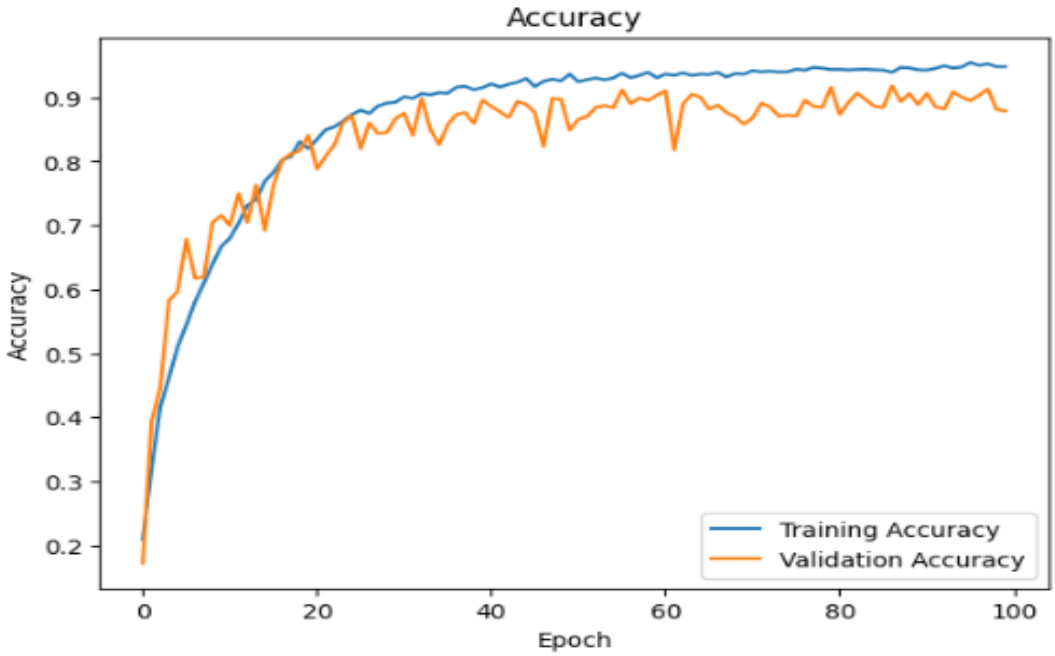

Figure 5.10 This figure shows the training and validation accuracy of the enhanced VGG16 pretrained model over each training session.

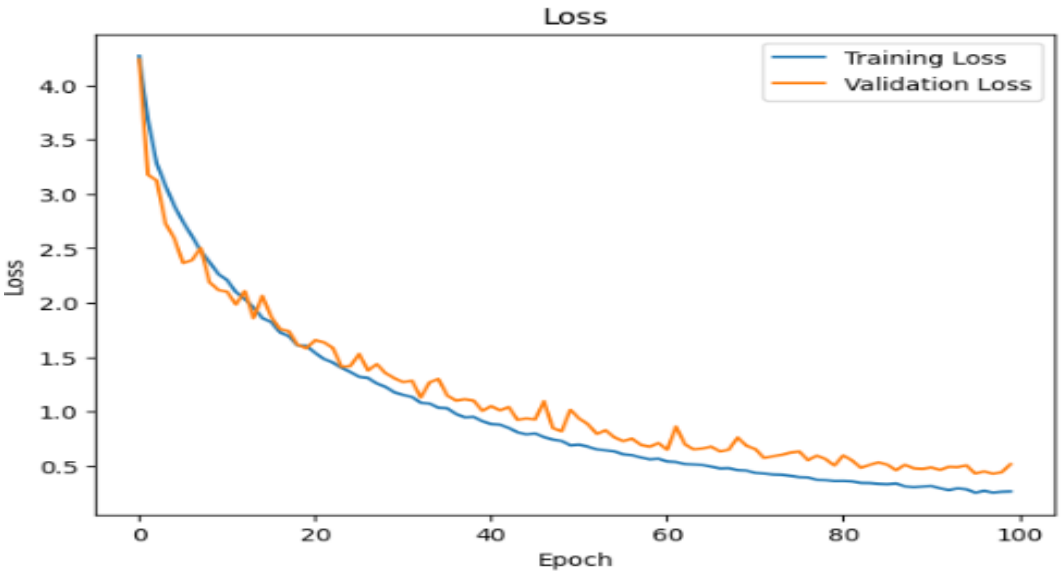

Figure 5.11 This convergence shows the training and validation loss for the modified VGG16 pretrained model for each training phase.

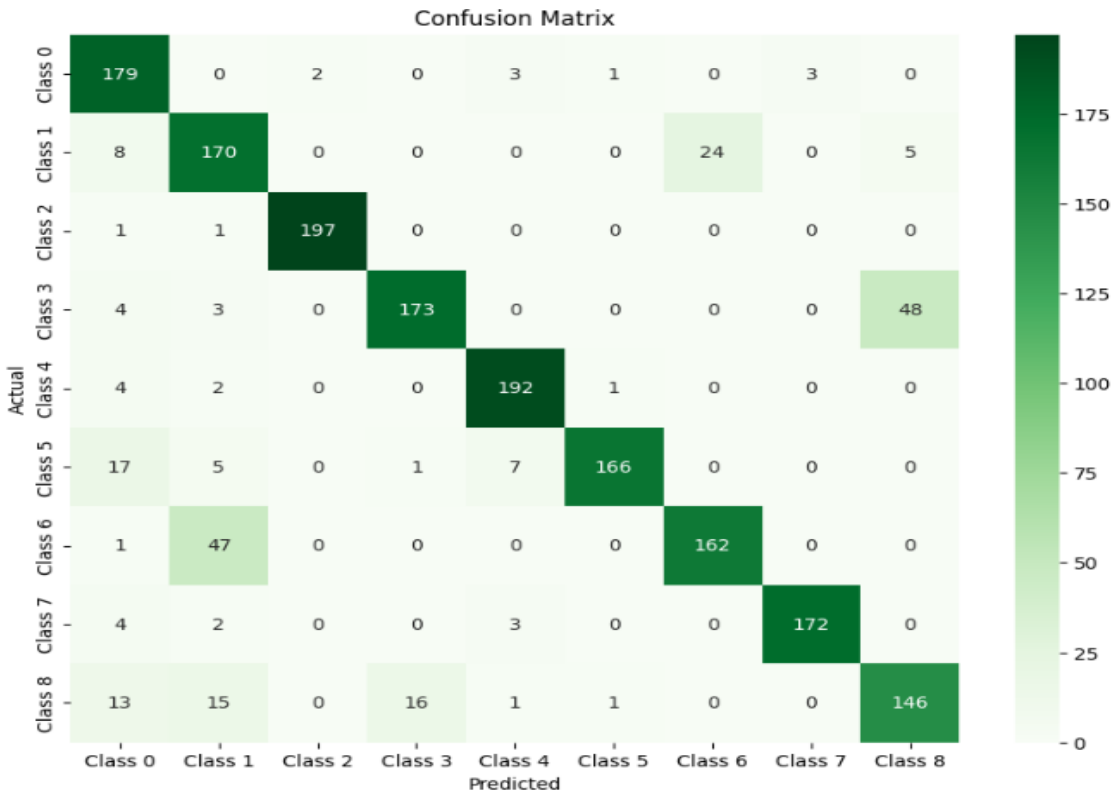

Figure 5. 12. The confusion matrix for the modified VGG16 pretrained model and illustrates how successfully the model categorised the nine different classes.

Performance of the ResNet50 model, attaining a remarkable 88% accuracy on a ISIC dataset. The model performs exceptionally well in terms of precision, recall, and F1-score in every class, suggesting a strong capacity to correctly categorize a range of skin diseases. Notably, the model achieves 89% precision, recall, and F1-score for vascular lesions and performs almost flawlessly for dermatofibroma and squamous cell carcinoma. The model's ability to differentiate between pigmented benign keratosis and basal cell carcinoma is demonstrated by its excellent recall and precision. Overall, the ResNet50 model exhibits balanced performance across classes and demonstrates good efficacy in classifying skin conditions, highlighting its applicability to dermatological image analysis applications, Table 5.6 show the result of the modified Resnet50 pretrained model.

Table 5.6.Result of modified ResNet50 pretrained model

Class	**metrics**	**Accuracy**	**Precision**	**Recall**	**F1-score**
1	actinic keratosis	93.42%	0.91	0.95	0.93
2	basal cell carcinoma	78.56%	0.82	0.75	0.78
3	dermatofibroma	98.80%	1.00	0.98	0.99
4	melanoma	78.84%	0.82	0.75	0.79
5	nevus	93.71%	0.96	0.91	0.94
6	pigmented benign keratosis	96.60%	0.95	0.98	0.97
7	seborrheic keratosis	87.65%	0.88	0.87	0.88
8	squamous cell carcinoma	100%	1.00	1.00	1.00
9	vascular lesion	74.80%	0.68	0.81	0.74
Overall Value		88%	89%	89%	87%

The accuracy and loss values from the modified ResNet50 model's training procedure are shown in Figure 5. 14,Figure 5. 13. These measures shed light on the model's convergence and performance throughout training and the confusion matrix show in the Figure 5. 15.

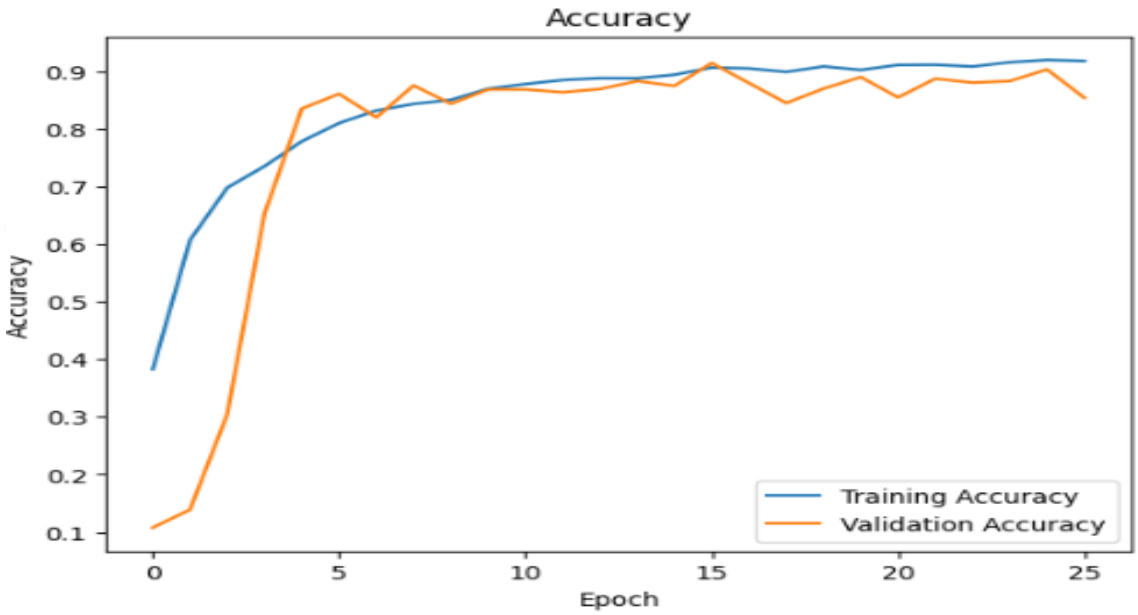

Figure 5. 14 Convergence graph of the ResNet50 modified model in the training phase This

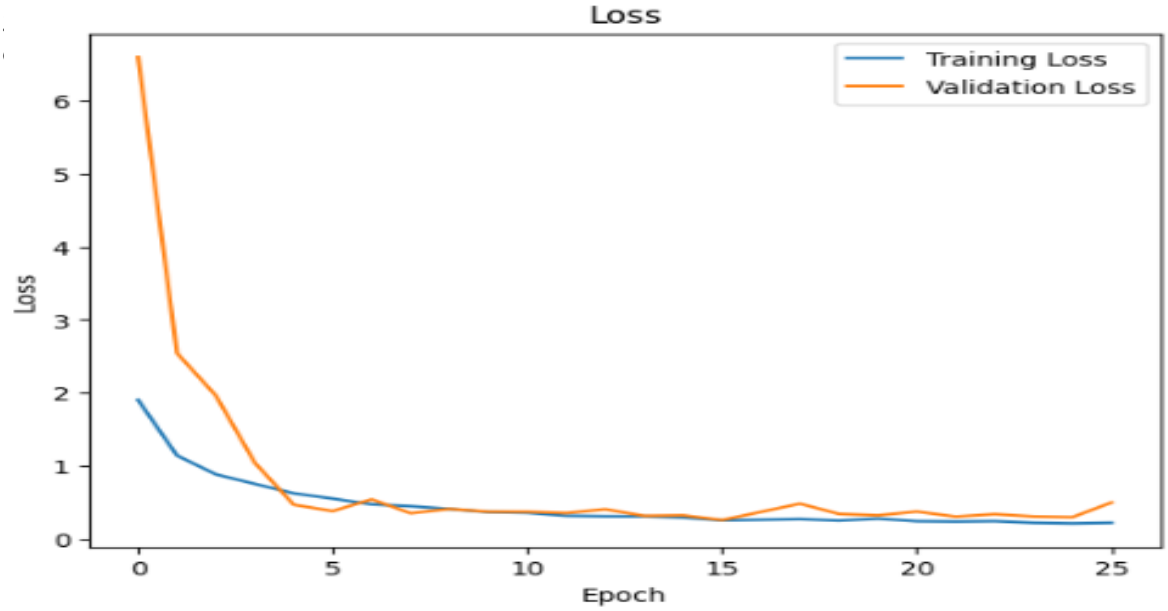

Figure 5. 13 This figure shows the training and validation losses for the modified ResNet50 pretrained model for each training phase.

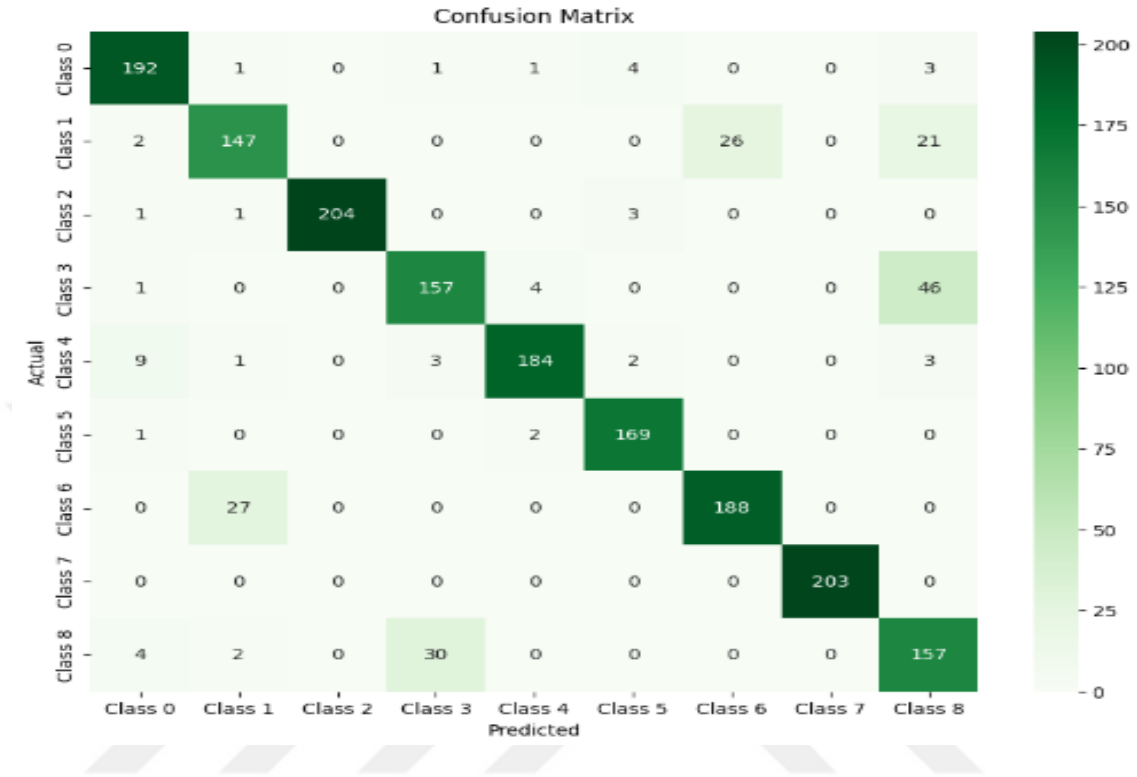

Figure 5. 15 This figure displays the confusion matrix for the modified ResNet50 pretrained model and illustrates how successfully the model categorised the nine different classes.

In Table 5.7 performance of the modified InceptionV3 model with an overall accuracy of 90% on the ISIC dataset. Each class's precision, recall, and F1-score metrics show a balanced performance in a range of skin conditions. The model performs exceptionally well in vascular lesion identification, achieving ideal scores in F1-score, recall, and precision of 90%. High precision and recall are also demonstrated by pigmented benign keratosis and dermatofibroma, demonstrating the model's ability to accurately identify these disorders. The precision and recall scores for melanoma classification are marginally lower than those of other classes, thus there is still opportunity for improvement. While there is significant variation in the model's performance across different classes, overall, the InceptionV3 model shows promise in the classification of skin disorders. It is particularly good at identifying particular dermatological conditions.

Table 5.7.Result of modified InceptionV3 pretrained model

Class	metrics	Accuracy	Precision	Recall	F1-score
1	actinic keratosis	94.03%	0.94	0.94	0.94
2	basal cell carcinoma	79.61%	0.77	0.82	0.80
3	dermatofibroma	99.75%	0.99	1.00	1.00
4	melanoma	89.01%	0.86	0.92	0.89
5	nevus	94.85%	0.91	0.98	0.95
6	pigmented benign keratosis	95.29%	0.97	0.93	0.95
7	seborrheic keratosis	83.96%	0.85	0.83	0.84
8	squamous cell carcinoma	99.17%	0.99	0.99	0.99
9	vascular lesion	81.62%	0.88	0.75	0.81
Overall value		90%	90%	90%	89%

The accuracy and loss values from the modified InceptionV3 model's training procedure are shown in Figure 5. 16, Figure 5. 17. These measures shed light on the model's convergence and performance throughout training and the confusion matrix show in the Figure 5. 18

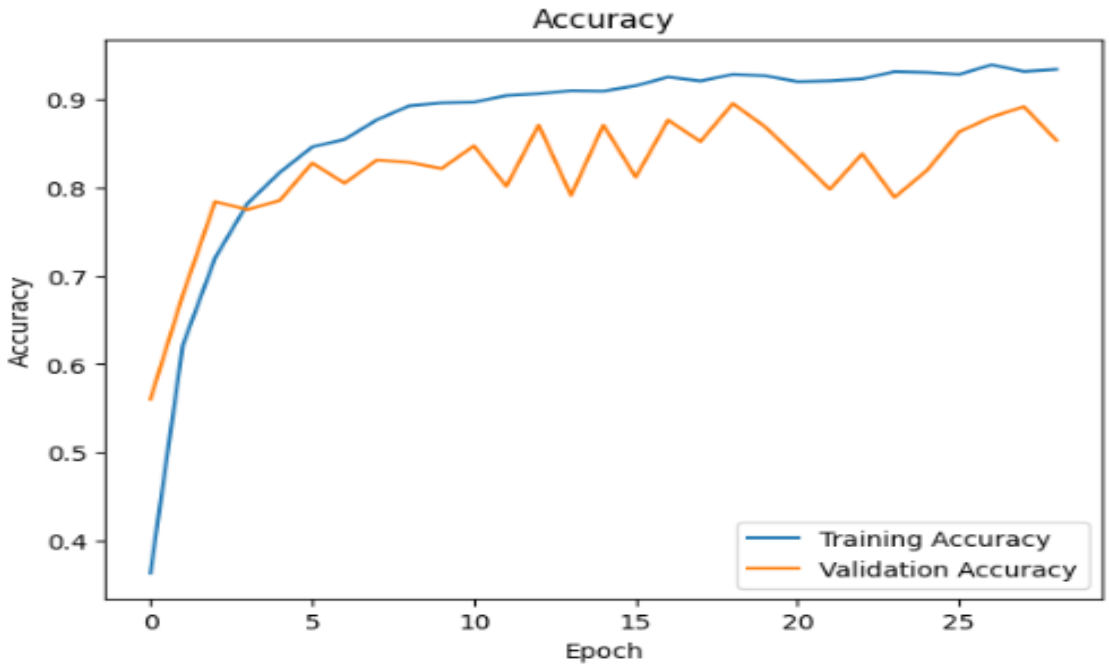

Figure 5. 17. The Accuracy convergence graph obtained by the modified InceptionV3 model during the training phase

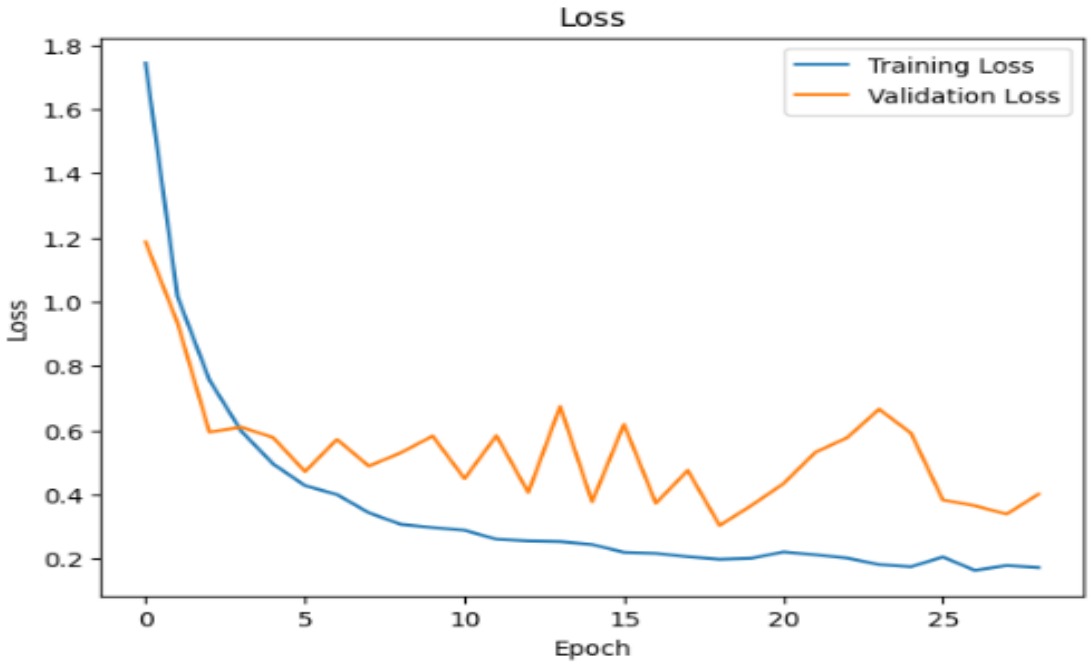

Figure 5. 16. The loss convergence graph obtained by the modified InceptionV3 model during the training phase

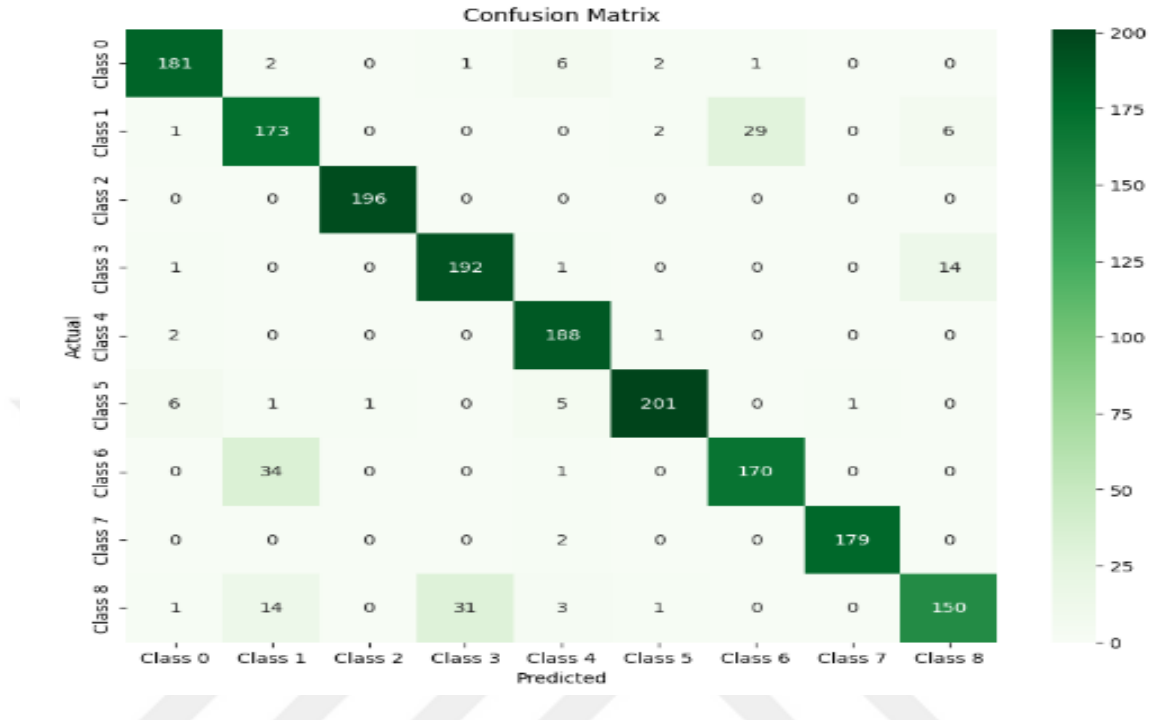

Figure 5. 18 This picture displays the confusion matrix for the modified InceptionV3 pretrained model and illustrates how successfully the model categorised the nine different classes.

During preprocessing, the data augmentation approaches was used to improve the model's generalization capabilities across various skin conditions and reduce class imbalances within the ISIC dataset. Applying a number of transformations, such as rotation, shifting the width and height, shearing, zooming, and horizontal flipping, was required for this. By enhancing the existing photos, these methods guaranteed that each class was adequately represented in addition to adding diversity to the training data. Next, the dataset was carefully balanced by limiting the number of photos to 1000 for each class. After that, several models were trained and assessed using the balanced and expanded dataset.

The performance of different models in comparable tasks is highlighted through comparison with the existing literature, which offers a framework for assessing our proposed models. The accuracy attained by various models as reported in recent studies is summarised, Table 5.8. comparing the performance of the proposed CNN models and modified pretrained CNN, this study expands on the previous comparison. These findings offer valuable information on the efficacy of our adjustments and the relative performance when compared to industry standards.

Table 5.8. comparison of the proposed method with similar studies in the literature

Ref.	**Model Name**	**Accuracy**
(Albahar 2019)	CNN and Novel Regularizer	97.49%
(Sanketh, Bala et al. 2020)	A CNN created with two convolutional layers, two max pool layers, and a final fully linked layer	98%
(Daghrir, Tlig et al. 2020)	hybrid approach	85.5%
(Vipin, Nath et al. 2021)	U-Net	88.7%
(Rahi, Khan et al. 2019)	CNN network with CGG16, RESNET50, DENSNET50	90%
(Jojoa Acosta, Caballero Tovar et al. 2021)	CNN combined with pretrained ResNet152	90.4%
(Rahi, Khan et al. 2019)	CNN with convolutional layers of two distinct strides	84.76%
proposed work	Proposed CNN Model_224	83%
	Proposed CNN Model _128	86%
	Modified AlexNet	89%
	Original AlexNet	73%
	Modified VGG16	86%
	Original VGG16	64%
	Modified ResNet50	88%
	Original ResNet50	53%
	Modified InceptionV3	90%
	Original InceptionV3	64%

6 CONCLUSION AND RECOMMENDATION

This thesis introduces the proposed CNN model, which uses skin cancer RGB images to categorize infected individuals. The model is tested using ISIC datasets, with image enhancement in pre-processing utilizing the Image class from the PIL or Pillow library to open and resize images, and np.asarray to translate the resized images into NumPy arrays. Data Augmentation Techniques are also used.

The suggested model does a fantastic job of classifying skin cancer patients from the ISIC dataset, achieving an accuracy of 83% and 86%, respectively, for image sizes of 224 by 224 and 128 by 128 pixels.

Additionally, a number of pre-trained modified CNN models were compared in this work using both original and customised models in order to compare their parameters and performance. The models ResNet, InceptionV3, VGG16, and AlexNet are used in the comparison. After doing a comparative analysis, updated VGG16 achieved 86%, AlexNet 89%, ResNet 88%, and InceptionV3 90%. In future work, to increase the model's capacity to generalise to changing skin tones and appearance changes, additional unique photos and revisions should be added to the training dataset. Optimising the current model, attempting more complex architectures, or experimenting with different transfer learning algorithms or pre-trained models will all yield better results.

7 REFERENCES

Adegun, A. A. and S. Viriri (2019). "Deep learning-based system for automatic melanoma detection." IEEE Access 8: 7160-7172.

Ahmed, S. G., F. Zeng, M. Alrifaey and M. Ahmadipour (2023). Skin Cancer Classification Utilizing a Hybrid Model of Machine Learning Models Trained on Dermoscopic Images. 2023 3rd International Conference on Emerging Smart Technologies and Applications (eSmarTA), IEEE.

Albahar, M. A. (2019). "Skin lesion classification using convolutional neural network with novel regularizer." IEEE Access 7: 38306-38313.

Albawi, S., T. A. Mohammed and S. Al-Zawi (2017). Understanding of a convolutional neural network. 2017 international conference on engineering and technology (ICET), Ieee.

ali Bagheri, M., G. A. Montazer and S. Escalera (2012). Error correcting output codes for multiclass classification: application to two image vision problems. The 16th CSI international symposium on artificial intelligence and signal processing (AISP 2012), IEEE.

Ali, M. S., M. S. Miah, J. Haque, M. M. Rahman and M. K. Islam (2021). "An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models." Machine Learning with Applications 5: 100036.

Amornsamankul, S., J. Promrak and P. Kraipeerapun (2011). "Solving multiclass classification problems using combining complementary neural networks and error-correcting output codes."

Anas, M., K. Gupta and S. Ahmad (2017). "Skin cancer classification using K-means clustering." International Journal of Technical Research and Applications 5(1): 62-65.

Attia, M., M. Hossny, S. Nahavandi and A. Yazdabadi (2017). Skin melanoma segmentation using recurrent and convolutional neural networks. 2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017), IEEE.

Chanda, D., M. S. H. Onim, H. Nyeem, T. B. Ovi and S. S. Naba (2024). "DCENSnet: A new deep convolutional ensemble network for skin cancer classification." Biomedical Signal Processing and Control 89: 105757.

Chen, H., A. Chen, L. Xu, H. Xie, H. Qiao, Q. Lin and K. Cai (2020). "A deep learning CNN architecture applied in smart near-infrared analysis of water pollution for agricultural irrigation resources." Agricultural Water Management 240: 106303.

Daghrir, J., L. Tlig, M. Bouchouicha and M. Sayadi (2020). Melanoma skin cancer detection using deep learning and classical machine learning techniques: A hybrid approach. 2020 5th international conference on advanced technologies for signal and image processing (ATSIP), IEEE.

Dhillon, A. and G. K. Verma (2020). "Convolutional neural network: a review of models, methodologies and applications to object detection." Progress in Artificial Intelligence 9(2): 85-112.

Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau and S. Thrun (2017). "Dermatologist-level classification of skin cancer with deep neural networks." nature 542(7639): 115-118.

Guo, F., R. He and J. Dang (2019). "Implicit discourse relation recognition via a BiLSTM-CNN architecture with dynamic chunk-based max pooling." IEEE Access 7: 169281-169292.

Harangi, B. (2017). "Skin lesion detection based on an ensemble of deep convolutional neural network." arXiv preprint arXiv:1705.03360.

He, K., X. Zhang, S. Ren and J. Sun (2016). Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition.

Hosny, K. M., M. A. Kassem and M. M. Foaud (2019). "Classification of skin lesions using transfer learning and augmentation with Alex-net." PloS one 14(5): e0217293.

Jaiswal, A., N. Gianchandani, D. Singh, V. Kumar and M. Kaur (2021). "Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning." Journal of Biomolecular Structure and Dynamics 39(15): 5682-5689.

Jayapriya, K. and I. J. Jacob (2020). "Hybrid fully convolutional networks-based skin lesion segmentation and melanoma detection using deep feature." International Journal of Imaging Systems and Technology 30(2): 348-357.

Jiang, Y., S. Cao, S. Tao and H. Zhang (2020). "Skin lesion segmentation based on multi-scale attention convolutional neural network." IEEE Access 8: 122811-122825.

Jojoa Acosta, M. F., L. Y. Caballero Tovar, M. B. Garcia-Zapirain and W. S. Percybrooks (2021). "Melanoma diagnosis using deep learning techniques on dermatoscopic images." BMC Medical Imaging 21(1): 1-11.

Khan, M. A., M. Sharif, T. Akram, S. A. C. Bukhari and R. S. Nayak (2020). "Developed Newton-Raphson based deep features selection framework for skin lesion recognition." Pattern Recognition Letters 129: 293-303.

Krizhevsky, A., I. Sutskever and G. E. Hinton (2012). "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25.

Kumar, A. and A. Vatsa (2022). "Untangling classification methods for melanoma skin cancer." Frontiers in big Data 5: 848614.

Liang, J., Y. Deng and D. Zeng (2020). "A deep neural network combined CNN and GCN for remote sensing scene classification." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13: 4325-4338.

Lin, C., Y. Li, H. Liu, Q. Huang, Y. Li and Q. Cai (2020). Power Enterprise Asset Estimation Algorithm Based on Improved InceptionV3. 2020 Chinese Automation Congress (CAC), IEEE.

Majtner, T., S. Yildirim-Yayilgan and J. Y. Hardeberg (2019). "Optimised deep learning features for improved melanoma detection." Multimedia Tools and Applications 78: 11883-11903.

Maqsood, S. and R. Damaševičius (2023). "Multiclass skin lesion localization and classification using deep learning based features fusion and selection framework for smart healthcare." Neural Networks 160: 238-258.

Marchetti, M. A., N. C. Codella, S. W. Dusza, D. A. Gutman, B. Helba, A. Kalloo, N. Mishra, C. Carrera, M. E. Celebi and J. L. DeFazio (2018). "Results of the 2016 international skin imaging collaboration isbi challenge: Comparison of the accuracy of computer algorithms to dermatologists for the diagnosis of melanoma from dermoscopic images." Journal of the American Academy of Dermatology 78(2): 270.

Mendonça, T., M. Celebi, T. Mendonca and J. Marques (2015). "Ph2: A public database for the analysis of dermoscopic images." Dermoscopy image analysis.

Mikołajczyk, A. and M. Grochowski (2018). Data augmentation for improving deep learning in image classification problem. 2018 international interdisciplinary PhD workshop (IIPhDW), IEEE.

Mukherjee, S., A. Adhikari and M. Roy (2019). Malignant melanoma classification using cross-platform dataset with deep learning CNN architecture. Recent Trends in Signal and Image Processing: Proceedings of ISSIP 2018, Springer.

Mukherjee, S., A. Malu, B. Ar and P. Bhattacharyya (2012). TwiSent: a multistage system for analyzing sentiment in twitter. Proceedings of the 21st ACM international conference on Information and knowledge management.

Nasr-Esfahani, E., S. Samavi, N. Karimi, S. M. R. Soroushmehr, M. H. Jafari, K. Ward and K. Najarian (2016). Melanoma detection by analysis of clinical images using convolutional neural network. 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE.

Nida, N., A. Irtaza, A. Javed, M. H. Yousaf and M. T. Mahmood (2019). "Melanoma lesion detection and segmentation using deep region based convolutional neural network and fuzzy C-means clustering." International journal of medical informatics 124: 37-48.

Nugroho, R. A., A. S. Nugraha, A. Al Rasyid and F. W. Rahayu (2021). "Improvement on KNN using genetic algorithm and combined feature extraction to identify COVID-19 sufferers based on CT scan image." TELKOMNIKA (Telecommunication Computing Electronics and Control) 19(5): 1581-1587.

Nyíri, T. and A. Kiss (2018). Novel ensembling methods for dermatological image classification. Theory and Practice of Natural Computing: 7th International Conference, TPNC 2018, Dublin, Ireland, December 12–14, 2018, Proceedings 7, Springer.

Pang, Y., J. Cao and X. Li (2016). "Learning sampling distributions for efficient object detection." IEEE transactions on cybernetics 47(1): 117-129.

Panwar, H., P. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj and V. Singh (2020). "A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-Scan images." Chaos, Solitons & Fractals 140: 110190.

Phung, V. H. and E. J. Rhee (2018). "A deep learning approach for classification of cloud image patches on small datasets." Journal of information and communication convergence engineering 16(3): 173-178.

Qian, W., Y. Huang, Q. Liu, W. Fan, Z. Sun, H. Dong, F. Wan and X. Qiao (2020). "UAV and a deep convolutional neural network for monitoring invasive alien plants in the wild." Computers and Electronics in Agriculture 174: 105519.

Rahi, M. M. I., F. T. Khan, M. T. Mahtab, A. A. Ullah, M. G. R. Alam and M. A. Alam (2019). Detection of skin cancer using deep neural networks. 2019 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), IEEE.

Rahman, Z., M. S. Hossain, M. R. Islam, M. M. Hasan and R. A. Hridhee (2021). "An approach for multiclass skin lesion classification based on ensemble learning." Informatics in Medicine Unlocked 25: 100659.

Rodrigues, D. d. A., R. F. Ivo, S. C. Satapathy, S. Wang, J. Hemanth and P. P. Reboucas Filho (2020). "A new approach for classification skin lesion based on transfer learning, deep learning, and IoT system." Pattern Recognition Letters 136: 8-15.

Sanghvi, K., A. Aralkar, S. Sanghvi and I. Saha (2020). "A survey on image classification techniques." Available at SSRN 3754116.

Sanketh, R. S., M. M. Bala, P. V. N. Reddy and G. P. Kumar (2020). Melanoma disease detection using convolutional neural networks. 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE.

Sarker, L., M. M. Islam, T. Hannan and Z. Ahmed (2020). "COVID-DenseNet: A deep learning architecture to detect COVID-19 from chest radiology images." preprint 2020050151.

Shah, A., M. Shah, A. Pandya, R. Sushra, R. Sushra, M. Mehta, K. Patel and K. Patel (2023). "A comprehensive study on skin cancer detection using artificial neural network (ANN) and convolutional neural network (CNN)." Clinical eHealth.

Shahin, A. H., A. Kamal and M. A. Elattar (2018). Deep ensemble learning for skin lesion classification from dermoscopic images. 2018 9th Cairo International Biomedical Engineering Conference (CIBEC), IEEE.

Shahsavari, A., T. Khatibi and S. Ranjbari (2023). "Skin lesion detection using an ensemble of deep models: SLDED." Multimedia Tools and Applications 82(7): 10575-10594.

Shustanov, A. and P. Yakimov (2019). Modification of single-purpose CNN for creating multi-purpose CNN. Journal of Physics: Conference Series, IOP Publishing.

Simonyan, K. and A. Zisserman (2014). "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556.

Singh, D., V. Kumar and M. Kaur (2021). "Densely connected convolutional networks-based COVID-19 screening model." Applied Intelligence 51: 3044-3051.

Singh, Y., C. Farrelly, Q. A. Hathaway, A. Choudhary, G. Carlsson, B. Erickson and T. Leiner (2023). "The role of geometry in convolutional neural networks for medical imaging." Mayo Clinic Proceedings: Digital Health 1(4): 519-526.

Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich (2015). Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition.

Vijayalakshmi, K., S. Uma, R. Bhuvanya and A. Suresh (2018). "A demand for wearable devices in health care." Int. J. Eng. Technol 7(1): 1-4.

Vipin, V., M. K. Nath, V. Sreejith, N. F. Giji, A. Ramesh and M. Meera (2021). Detection of melanoma using deep learning techniques: A review. 2021 International Conference on Communication, Control and Information Sciences (ICCISc), IEEE.

Weng, Y., T. Zhou, L. Liu and C. Xia (2019). "Automatic convolutional neural architecture search for image classification under different scenes." IEEE access 7: 38495-38506.

Wiharto, W., H. Kusnanto and H. Herianto (2015). "Performance analysis of multiclass support vector machine classification for diagnosis of coronary heart diseases." arXiv preprint arXiv:1511.02352.

Xiao-Feng, L., Z. Xue-ying and D. Ji-Kang (2010). Speech recognition based on support vector machine and error correcting output codes. 2010 First International Conference on Pervasive Computing, Signal Processing and Applications, IEEE.

Yan, Z. and Y. Yang (2014). "Application of ecoc svms in remote sensing image classification." The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 40: 191-196.

Yan, Z. and Y. Yang (2014). "Performance analysis and coding strategy of ECOC SVMs." International Journal of Grid and Distributed Computing 7(1): 67-76.

Yu, H., L. T. Yang, Q. Zhang, D. Armstrong and M. J. Deen (2021). "Convolutional neural networks for medical image analysis: state-of-the-art, comparisons, improvement and perspectives." Neurocomputing 444: 92-110.

Yu, Y., K. Adu, N. Tashi, P. Anokye, X. Wang and M. A. Ayidzoe (2020). "Rmaf: Relu-memristor-like activation function for deep learning." IEEE Access 8: 72727-72741.

Yuesheng, F., S. Jian, X. Fuxiang, B. Yang, Z. Xiang, G. Peng, W. Zhengtao and X. Shengqiao (2021). "Circular fruit and vegetable classification based on optimized InceptionV3." IEEE Access 9: 113599-113611.

Zhang, Y., S. Xiang, X. Guo, A. Wen and Y. Hao (2020). "The winner-take-all mechanism for all-optical systems of pattern recognition and max-pooling operation." Journal of Lightwave Technology 38(18): 5071-5077.

RESUME

Name and surname	Bafreen Mohammed
Personal Information	Nationality: Date and Place of Birth:
Contact information	Email:
Education Information	High school: 2008-2011 Jagarkheen High School Undergraduate: 2011-2015 Duhok University Science Faculty Computer Department
Work Experience	2016-present: Duhok Private Technical Institute (DPTI).