An Efficient Approach to Detect Melanoma Skin Cancer using a Custom CNN Model

by

K.M Saidur Rahman
22241036
Tanjim Bin Amin
17301220
Mahdi Sakib Rahman
21101349
G M Shadman Hossain Sakib
17301099

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

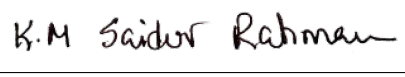
Department of Computer Science and Engineering
Brac University
September 2022

Declaration

It is hereby declared that

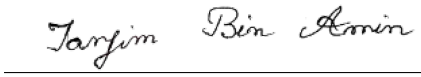
1. The thesis submitted is my/our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



K.M Saidur Rahman

22241036



Tanjim Bin Amin

17301220

Mahdi Sakib Rahman

21101349

G M Shadman Hossain Sakib

17301099

Approval

The thesis/project titled "An efficient approach to detect melanoma skin cancer using a custom CNN model" submitted by

1. K.M Saidur Rahman (22241036)
2. Tanjim Bin Amin (17301220)
3. Mahdi Sakib Rahman (21101349)
4. G M Shadman Hossain Sakib (17301099)

Of Summer, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on September 25, 2022.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Ashraful Alam, PhD

Assistant Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD

Assistant Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
BRAC University

Ethics Statement

Hereby, we, the members, consciously assure that for the manuscript /insert title/ the following is fulfilled:

• This content is the authors' original work that has never been published before.

• The authors' research and analysis are accurately and completely represented in the work.

• Co-authors and co-researchers are appropriately acknowledged in the study for their significant contributions.

• The findings are contextualized appropriately in light of previous and ongoing research.

• All sources are cited appropriately. Text that has been literally copied must be identified as such with quote marks and a suitable reference.

• All authors personally and actively contributed to the article's creation and will accept public responsibility for its content.

Violations of the Ethical Statement standards may result in serious consequences. We agree with the preceding declarations and certify that this submission complies with the rules of BRAC University.

Abstract

Despite only making up 1% of all occurrences of skin cancer, melanoma is one of the most prevalent forms to cause fatalities in recent years. Melanoma has a survival rate of more than 50% from the early stages to the end. To survive this type of cancer, it is essential to identify lesions on the skin early and to keep an eye out for any complications. If skin cancer is not detected and treated early, it is among the most fatal cancers. Of the skin cancers, which are among the deadliest, melanoma is the most unexpected. Like most other diseases, melanoma may be treatable if caught early enough. Due to the high cost of having a dermatologist screen every patient and the difficulty of human judgment, an automated system for melanoma diagnosis is required. Due to its promising pattern recognition skills, Convolutional Neural Network (CNN) models have recently gained a lot of interest in medical imaging. Melanoma diagnosis from dermoscopic skin samples automatically is a difficult task. In contrast to other types, melanoma ranks as the most serious type of skin cancer. However, those who are diagnosed early on have a better prognosis; several methods of spontaneous melanoma recognition and diagnosis have been researched by different researchers for the objective of providing a supplementary opinion to professionals. Building models using existing data has proven problematic due to the imbalance between classes. However, these issues may be solved by implementing a deep learning approach as a machine vision tool. The purpose of the current study was to determine how well dermoscopy and deep learning classified melanoma. In this paper, we introduce a brand-new deep learning model that was created to categorize melanoma skin cancer. And we have compared the result of our suggested model with pre-trained VGG16, VGG19, and AlexNet. According to experimental data, we discovered that our model worked well and could accurately categorize melanoma skin cancer. Also, the proposed system is competitive in the area of melanoma detection and superior in terms of accuracy and can be employed in the clinical decision-making procedure for melanoma skin cancer early detection.

Keywords: Convolutional Neural Networks(CNN), Classification, Skin cancer, Melanoma, VGG-16, VGG-19, AlexNet.

Dedication

This thesis is dedicated to our university's mentors, without whom we would not have been capable of completing this thesis. Our professors were more than just academic mentors; they were also a source of encouragement and support when we needed it most.

Acknowledgement

In the first place, we would like to thank Almighty Allah for allowing us to complete our thesis on time and without deterrent.

Having said that, we would like to express our gratitude to Dr. Md. Ashraful Alam (PhD), our distinguished instructor and supervisor, for his unwavering support and tenacious oversight, which allowed us to complete our project. In addition, we would like to thank our supportive friends who have been there for us during the difficult times. Lastly, to our parents, without their unwavering support, it may be impossible. With their gracious assistance and prayers, we are now on the verge of graduating.

Table of Contents

List of Figures

List of Tables

Nomenclature

AdaGrad Adaptive Gradient

ANN Artificial Neural Network

API Application Programming Interface

CNN Convolutional Neural Network

CUDA Compute Unified Device Architecture

DOG Derivative of Gaussian

DTEA Delhi Tamil Education Association

FC Fully Connected

FCL Fully Connected Layer

GAC Geodesic Active Contour

GPU Graphics Processing Unit

GVF Multi gradient Vector Flow

*HAM*10000 Human Against Machine with 10000 training images

IDE Integrated Development Environment

ILSVRC ImageNet Large Scale Visual Recognition Challenge

ISIC International Skin Imaging Collaboration

LSTM Long Short-Term Memory

ML Machine Learning

MLP Multilayer Perceptron

MNIST Modified National Institute of Standards and Technology

NLP Natural Language Processing

PIP Pip Install Packages

PyPI Python Package Index

RAC Region Based Contour

RAM Random Access Memory

ReLU Rectified Linear Activation Unit

RGB Red Green Blue

RMSProp Root Mean Squared Propagation

RNN Recurrent Neural Network

SGD Stochastic Gradient Descent

SNR Signal-to-Noise Ratio

UV Ultraviolet

VGG Visual Geometry Group

Chapter 1

Introduction

1.1 Introduction

Melanoma skin cancer incidences are increasing worldwide and are now being found in populations with darker skin due to increased UV exposure. Like many other types of cancer, the likelihood of effective treatment and cure is increased when cancer is discovered in its early stages. If discovered later, the chances of survival are greatly reduced. Image processing techniques for melanoma diagnosis have been developed to help dermatologists. Abnormal cell growth in the epidermis is the root cause of skin cancer, which leads to mutation and rapid expansion of skin cells, resulting in tumors. The most deadly form of skin cancer, melanoma, is responsible for roughly 75% of all skin cancer fatalities[5]. It is a kind of cancer that develops in the skin's pigment cells (melanocytes). It also has the highest rate of death among skin cancers. Whites had a 2.6% lifetime risk of having melanoma, 0.1 for blacks, and 0.6% for Hispanics, according to statistics[35]. Cutaneous melanoma, the most severe kind of illness, is responsible for 90% of all skin cancer deaths[14]. Fortunately, the prognosis is often favorable if the illness is identified and treated promptly[6]. UV radiation from the sun, or sunburn, is one of the factors contributing to the rise in melanoma cases [39]. After so much medical research and experimentation, it is undeniable that melanoma is caused by more than only UV light, as other body parts that are not exposed to sunlight are equally susceptible to melanoma. There are four different subtypes of this skin cancer. They are,

1. Superficial spreading
2. Nodular
3. Lentigo maligna
4. Acral lentiginous

It depends on the clinical appearance, progression, Anatomic size, and histologic appearance.

1.1.1 Superficial spreading melanoma

As it accounts for 70% to 80% of all melanomas, it is the most prevalent subtype. This type of melanoma is very much common in middle age. This type of melanoma is frequently found in the upper back of both males and females. Its length is 6 mm, flat and symmetric with variant colors. It begins in a non-specific manner and then changes its shape.

1.1.2 Nodular melanoma

Between 15% to 30% of all melanomas are caused by it, making it the second most common kind of skin cancer. The initial radial growth phase is absent in nodular melanoma, which is a de novo vertical growth tumor. It has a firm blue to black papule or nodule that can be ulcerated, and men over the age of 60 are more likely than women to have it. The trunk and head/neck region are preferred places.

1.1.3 Lentigo maligna

It is the third kind of skin cancer, accounting for about 5% to 10% of all melanomas. Intraepidermal melanoma is represented by Lentigo maligna. Lentigo maligna affects 5% of male and female patients, who are typically older. They progress throughout time. It usually impacts the face, neck, and arms. Melanoma of this type might have a complicated pattern. Although the color is more uniform than in Superficial spreading melanoma, red and white spots may appear later. Although the shape is flat, the outline is uneven.

1.1.4 Acral lentiginous

It's a type of lentiginous melanoma of the skin. People with darker complexions are more prone to it. Acral lentiginous is an uncommon type of skin cancer that impacts people with fair complexion. Acral lentiginous melanoma can be detected on the lips, tongue, oral mucosa, soles of the feet, and palms. Males are more likely than females to develop acral lentiginous melanoma.
Melanoma is a cancer that can be treated if caught early enough. Early melanoma detection is accomplished by biopsy, pathology results, and dermoscopy imaging analysis. A non-invasive imaging technique called dermoscopy is frequently used to detect melanoma early and increase survival rates. The technique of dermoscopy is capturing a high-resolution, enlarged photograph of a malignant tumor on the skin and having it examined by specialists for the melanoma diagnosis. Dermoscopy image analysis by dermatologists is costly and needs a high degree of skill to accurately diagnose the issue. The use of dermoscopy images to aid in the early diagnosis of melanoma has highlighted the requirement for precise computer-aided diagnostic techniques. First and foremost, visual similarity between diseased and non-cancerous cells is far too difficult to achieve. Melanoma and non-melanoma skin cancers are difficult to distinguish. Secondly, segmenting the skin lesion from certain skin patches is challenging because of the poor contrast. In addition, various people's skin problems cause cosmetically unique melanoma, even though melanoma and non-melanoma are essentially the same on the outside. Other artifacts that compound the concern by adding to blurriness and occlusions include color calibration charts, hair, ruler lines, and veins. To help doctors diagnose melanoma, a number of innovative techniques have been created in recent years. Two of the more effective initiatives are machine learning and deep learning.

1.2 Motivation

A malignant tumor is created when normal melanocytes undergo changes and expand out of control. Malignant describes a malignant tumor's capacity to grow and spread to other parts of the body. Melanoma can occasionally arise from a healthy mole that a person already has on their skin. This results in modifications to the mole that are often visible, such as adjustments to its size, color, form, or border. Sometimes melanoma can appear on skin that doesn't already have a mole. Melanoma frequently responds well to surgery when it is detected early. Melanoma is one of the worst types of skin cancer, and it can penetrate the skin deeply. The research is stated to make people more aware of the classifications and follow-ups of this type of skin disease. And also to identify melanoma in the least time. One of the main factors contributing to the development of melanomas is skin exposure to UV light. The structure of skin is examined using a technique known as dermoscopy. Dermoscopy pictures can be used in an observation-based detection method for melanoma. The dermatologist's training has an impact on the dermoscopy's accuracy. Dermoscopy is a method used by dermatologists to identify illnesses, although early diagnosis is still feasible. The system's diagnostic will contribute to improving the diagnosis' speed and precision. A computer will be able to extract some details, such as asymmetry, color variation, and texture aspects. These minute parameters might not be seen to the unaided sight of a human. So, in future if any person is affected by melanoma than he/she could identify it in the primary stage.

1.3 Problem Statement

The most crucial kind of skin cancer is melanoma. Every year, it kills over 10,000 people in the United States. Melanoma treatment necessitates early detection, reorganization, and treatment. It has the ability to save over 95% of the population. Because it can acquire high-resolution images of the skin without being interrupted by reflections from the surface, One of the most crucial techniques for analyzing skin lesions is dermoscopy. This high-resolution imaging, which can reach diagnostic accuracy of up to 80%[17], is utilized by specially trained specialists to determine the risk of melanoma early on. However, there are not enough experienced dermatologists in the world. They are
1. The database isn't quite up to par.
2. Computational power was limited in the past
Previous research relied on a tiny number of dermoscopy skin lesion images, therefore computers were unable to learn and extract useful data. Humans also lacked creative solutions for dealing with vast numbers of photos. Due to their superior computational power and access to large datasets that have been gathered and made publicly available as open sources, deep learning techniques have excelled in many domains and have been shown to outperform humans in tasks such as speech recognition, language translation, and strategic games like Go[17]. Because widely used pre-trained deep learning networks, such as ImageNet, are trained on different datasets, they can't be utilized to address skin cancer issues in general.

1.4 Research Objective

Wherever skin is exposed to sunlight and certain areas where it is not, skin cancer can develop. Skin cells are growing abnormally there. Up to 95% of human lives can be saved by early diagnosis. Numerous researchers are looking into this illness. There were so many incidents reported in the past. The patient can pass away if this disease is discovered at its most advanced state. Due to delayed detection, many people might die. Practitioners used to inspect the lesion area before determining if cancer was present or not. Patients may occasionally pass away if any of the assumptions made about them turn out to be false. The main goal of this study is to develop and apply a custom CNN model to identify melanoma more accurately than existing methods. To give you an example, Machine Learning (ML) has advanced tremendously in recent years. Many machine learning algorithms have been developed and are being used in a variety of fields to make life easier and less dangerous for humans. Machine learning plays a role in melanoma detection as well, and a lot of work is being done with it. In particular, Convolutional Neural Networks (CNN) have lately shown notable results in the analysis of medical images. The idea is to relieve medical specialists of some of their work burdens, allowing them to concentrate more on patient care. However, much work has still to be done in assembling various deep neural architectures, and there is still more to be discovered. So, in this study, we will introduce an efficient sequential algorithm that is performed sequentially, without interruption, once from beginning to end, as opposed to simultaneously or in parallel. Because the nature of each dataset's properties varies, so do the obstacles; lesion segmentation, illness categorization and detection, location of visual dermoscopy features/patterns, and lesion attribute identification are among these challenges[45].

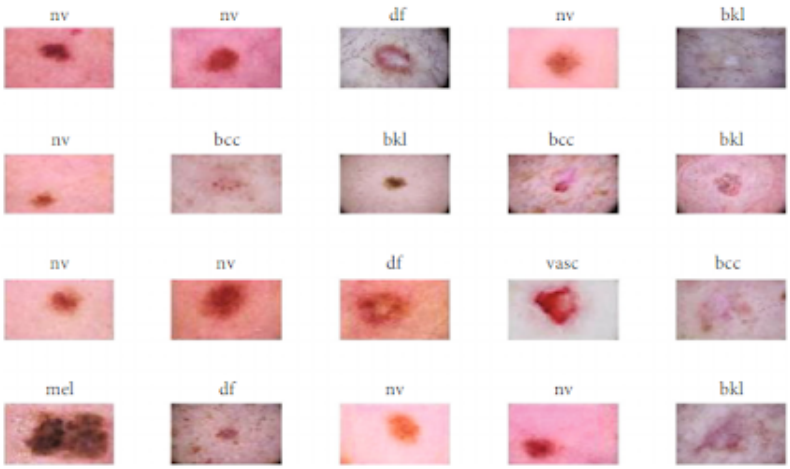


Figure 1.1: Different types of skin lesions

Figure 1.1 displays the various skin lesions based on the data of the HAM10000 dataset 2018. Most dermoscopic images may contain undesired particles such as thin and thick hair, air bubbles, gel, and occasionally various illumination effects. As a result, reliable methods for removing noise and undesirable particles became necessary. Some of these particles, such as air bubbles and oil, have grown less common as dermoscopy have evolved. The median filter is recommended for removing

unwanted elements. The most crucial step in accurately assessing a lesion is segmentation, as it determines the accuracy of all future phases. Precise segmentation is difficult since there are so many skin types and textures, as well as a wide range of lesion sizes, colors, and forms.

As a result, the proposed model intends to develop, test, and improve a very effective deep learning-based network for classifying melanoma. One of the features of the proposed study is the construction of numerous linked blocks to let considerable feature information travel directly across the network while building a DNN model for classifying skin lesions as benign or malignant on dermoscopic images. Again, a deep neural network is built by repeatedly iterating sub-blocks with a predetermined ratio on the validation set. Furthermore, in order to extract both low-level and high-level feature information from lesions, each block of the network makes use of a separate set of parameters, including the number of kernels, filter size, and stride. The suggested model performs better with fewer filters and learnable parameters on the HAM10000 datasets. As a result, it's a straightforward algorithm for classifying numerous skin malignancies.

Chapter 2

Literature Review

Many rules have been followed to detect melanoma over the years. Some of the most well-known ones are the ABCD rule, Pattern Analysis of Tumors, Menzies method, 7-point checklists, and 3-point checklists. Professional dermatologists utilize these procedures to distinguish between malignant and benign melanoma. Clinical application of skin surface microscopy was hampered until the dermatoscope was invented because stereo microscopes were rigid and expensive[1]. They made the procedure complicated and lengthy. The creation of the dermatoscope overcomes these challenges. It enabled quick surface microscopic analysis. Because novices find it challenging to understand photographs, Stolz et al. developed the ABCD rule, which relies on a semi-quantitative scoring system and a multivariate analysis of just four criteria. The four requirements are asymmetry, border, color, and differential structure. Perpendicular axes are employed to attain the lowest score for asymmetry, which is scored in terms of color and structure. The lesions were separated into eight segments for the border score, and the six possible lesion colors were counted for the color score, resulting in a score ranging from 1 to 6. These 5 points are counted for differential structure: network, homogenous areas, dots, globules, and streak. The study discovered that if the score from melanocytic pigmented skin lesions were higher than 5.45, it was malignant melanoma, but that a score between 4.75 and 5.45 could not rule out early melanoma.

Menzies Method contains three criteria, while Pattern Analysis includes eight[7]. Some pattern analysis criteria include the overall pattern, the pigment network, the globules, the streaks, the blue-white veil, the blotches, the hypopigmentation, the regression structures, and the vascular structures. Menzies's method criteria, on the other hand, include lesion color, pattern symmetry, and positive characteristics.

The 7-Point Checklist and the Golden Rule are two other traditional melanoma detection guidelines[3] and the 3-Point Checklist[8]. The 7-Point Checklist includes pigment networks, blue whitish veils, vascular structures, pigment streaks, patches, and globules, as well as regression structures. According to dermatology residents 5, it has a lower specificity but a better sensitivity than pattern analysis. The 3-Point checklist requirements, on the other hand, are asymmetry, unusual network, and blue-white structures.

Day proposes a method for determining the second border cut-off of the ABCD rule

by using the brightness gradient of the boundary lesion[4]. The author defines a sharp cut-off pigment as a considerable change in brightness away from the lesion in the skin slice. The slope of the lightness values, he added, may be utilized to determine the border cut-off. The problem with this strategy was that a substantial percentage of lesions in the set were marked "0" by dermatologists, making it difficult to interpret the results. Furthermore, running separate analyses on lesions marked "0" and lesions marked "1" in numerous lesions with comparable distribution does not help because it is very time-consuming clinically.

Artifacts and feature extraction, border detection, and noise reduction, pattern analysis, and lesion categorization are the required four automated dermoscopic image classification steps[10]. The hardest of these four steps is the border identification and categorization of pigmented skin lesions. Thresholding, region growth, clustering, Geodesic Active Contour(GAC) model, Region Based Contour(RAC) model, Multi gradient Vector Flow(GVF), and Dermatologist level algorithm for tumor extraction have all been created to produce an unsupervised segmentation technique for dermoscopy (DTEA). Inefficient algorithms for hair removal lead to over-segmentation and inadequate pattern analysis of the malignancy. Abaas et al. describe a comparative study of hair removal strategies such as linear interpolation, PDE nonlinear diffusion inpainting, and the exemplar-based inpainting approach in this research.

The DullRazor hair removal algorithm was proposed by Lee et al[2], which includes three steps: recognizing dark hair spots, bilinear interpolation to replace hair pixels, and an adaptive median filter to smooth the final result. This method only works if the tumors are smooth or devoid of hair pixels, which is unlikely to be the case with dermoscopic pictures. The nonlinear diffusion approach has the benefit over linear interpolation in that it fills large gaps while naturally using nearby information and maintaining a crisp border. However, when hair areas are filled, it results in some blur, which is noticeable. In addition, the computational complexity of PDE diffusion is higher than that of linear interpolation. However, because of its edge-preserving properties, it is excellent for tracing tumor boundaries. As a result, the authors believe it can be utilized as a pre-processing step. Non-linear PDE diffusion and texture synthesis are combined in the exemplar-based inpainting method. The authors also devised a quick marching system for hair removal. It entails three steps: identifying hair using the Derivative of Gaussian (DOG), refining using morphological methods, and fast marching using the inpainting method.

The partition of an image into discontinuous parts that are uniform in terms of brightness, color, and texture is known as segmentation[36]. Its purpose is to make an image simpler so that features may be extracted easily. Handcrafted feature-based methods such as threshold, edge, and region-based supervised segmentation, as well as intelligence-based supervised segmentation[44] are used. Threshold and clustering have the advantage of being quick to deploy and not requiring spatial properties of an image, yet being sensitive to noise. Edge and region-based techniques handle specific homogeneity constraints, though seed point initialization in some regions requires manual engagement. Although intelligence-based segmentation has a greater accuracy, it is still susceptible to noise and has some precision

uncertainty. When dealing with enormous amounts of data, these segmentation algorithms have demonstrated little success[22]. Deep Learning has recently revolutionized machine learning and computer vision, and with regard to body recognition, lesion detection, picture registration, segmentation, and classification, it is currently widely employed in medical image analysis. Kawahara et al. shown[16] for the categorization of skin lesions, features derived from a neural network pre-trained on ImageNet outperformed methods that relied on manually built human-engineered attributes. [21]Menegola et al. discovered that optimizing a neural network that has just been trained on ImageNet outperformed training a neural network from scratch in their investigation. As a result, deep neural networks are being used for segmentation. It is advantageous to transfer learning from one domain to another if data is accessible, and save time by using a pre-trained CNN as a feature extractor. In close-up skin photos, it could not be evident. Several automated processes have been suggested in recent years to help physicians diagnose melanoma. Among these strategies are traditional machine learning and deep learning-based methodologies[27] [40]. Deep learning-based algorithms for medical image analysis, such as segmentation, detection, and classification, have recently generated great results. As a result, deep learning-based approaches for melanoma diagnosis are receiving increased attention. To get better outcomes, different research used various CNN models for the categorization of melanoma[42]. Using pre-trained VGG-16 and VGG-19 models on their dataset, they were able to get a 76% accuracy, but that wasn't good enough. To tackle this challenge, reference[30] used deep learning architecture. Their research focused on detecting lesion attributes, segmenting lesion boundaries, and diagnosing lesions. AlexNet[11], Xception, ResNet[19], and VGGNet were among the pre-trained models used, with ResNet obtaining the greatest accuracy of 92.74 percent. On the ISIC-2017 dataset, Deep learning models were employed in a different study[15] to carry out the three tasks of segmentation, feature extraction, and classification. The results of the experiments show a high level of accuracy, with 75% for segmentation and 91% for classification.

In order to classify the melanoma kind and lower the false-positive rate, additional features of skin lesion pictures were extracted in[25]. Melanoma type classification and to reduce the false-positive rate. On the heraldic13 dataset, they used SVM, neural networks, and random forest classifiers, with the random forest classifier achieving the maximum accuracy of 90%. [33]employed an ensemble of Deep Neural Networks models, such as AlexNet, VGGNet, and GoogLeNet, to classify skin cancer and achieved an accuracy of 84.8% on the ISIC 2017 dataset. Other research[24] [41] [31] classified ISIC 2017 datasets with an accuracy of 76% using an ensemble of several approaches.

Prior studies[15] [48] [34] centered on categorizing skin lesion pictures into different cancer subtypes but could not go into great depth. Skin lesions were divided into melanoma and non-melanoma in the study [39]. The categorization of cutaneous lesions into several categories was the topic of another investigation[18]. For a precise diagnosis and to increase patient survival rates, melanomas must be divided into subgroups[21] [27]. The goal of this study is to detect acral lentiginous melanoma from dermoscopy images.

According to Haenssle HA[23], 300 high-quality photos were taken into consideration, 20% of which were melanomas and 80% of which were benign. There was no room for overlap between the training, validation, and testing datasets. Out of 300 photos, two dermatologists choose 100 for possible diagnosis. Set-100 was used to test CNN against dermatologists. A reader study at level 1 asked for melanoma or benign nevus diagnosis. Then, 4 weeks later, they provided their diagnosis and treatment in level-2, which included additional clinical data for 100 dermoscopy photos.

In the skin lesion analysis toward melanoma detection study, the authors presented two deep learning frameworks, the Lesion Indexing Network (LIN) and the Lesion Feature Network (LFN), to handle three key issues in the processing of skin lesion images: lesion segmentation, dermoscopic feature extraction, and lesion classification[26]. Lesion segmentation and classification were addressed simultaneously by the Lesion Indexing Network. The segmentation result and coarse performance of the classifier are produced by two extremely deep completely convolutional residual networks, specifically FCRN-88, which were trained with various training sets. The suggested lesion indexing calculation unit (LICU) measures the significance of a pixel for the categorization of lesions. The distance map produced by LICU is used to modify the coarse classification result[26].

The ISIC-2017 data set, which includes pictures of lesions from the three groups of melanoma, nevus, and seborrheic keratosis, was used by the authors to evaluate their method[28]. In the Supplementary section, further findings on the ISIC 2018 data set with photos of lesions from seven classes—actinic keratosis, basal carcinoma, benign keratosis, satiate, melanoma, lesion, and vascular lesion—are shown[28]. The authors search for and eliminate hairs and rulers on the photos using conventional data processing techniques. They improved the algorithm for removing hair. They suggest using decoupled DCGANs for data creation. We create 350 pictures of melanoma and 750 images of seborrheic keratosis using two distinct Deep Convolutional Generative Adversarial Networks (DCGANs) [54], as these two classes were significantly underrepresented in the ISIC 2017 data set in comparison to a much larger nevus class. We call this strategy "de-coupled DCGANs" since we employ different networks for each class[28].

The section analyzes the performance of the whole working technique presented in the research based on four key metrics: the ability to accurately extract features from lesion locations, the speed of computation, and the performance of segmentation[37]. For the purposes of detection and segmentation, three separate publically accessible datasets—PH2, ISBI 2017, and ISIC 2019—are employed in this case. On a computer running Ubuntu 18.04 and equipped with an i7 CPU, 32 GB of RAM, and a 4 GB GPU, all operations and computations were completed. The OpenCV image processing framework and Python were used to create the complete system[37]. The work, which was written by numerous authors, was conceptualized by S.B. under the guidance of R.B. and A.D. A.C. helped to develop the work's methodology in collaboration with Banerjee, who carried out a real investigation and thorough analysis of data[37].

Different neural network algorithms for skin cancer detection and classification have been covered in this systematic review research[46]. These methods are all non-invasive. The method of detecting skin cancer involves several steps, including preprocessing, picture segmentation, feature extraction, and classification. The categorization of lesion pictures using ANNs, CNNs, KNNs, and RBFNs was the main emphasis of this review. Each algorithm has benefits and drawbacks. The key to getting the best results is choosing the categorization method correctly. However, because it is more directly tied to computer vision than other neural networks, CNN performs better than other types of neural networks when categorizing picture data[46].

Chapter 3

Methodology

10,000 pictures in all are contained in this collection. The dataset is split into two parts: test 950 images, which is used to assess the correctness of the trained models, and train 9,050 images, which is employed to train the models. Malignant and benign cancer photos may be found in the train section (total of 9,050 images). And the test section has malignant (total of 475 images) and benign (total of 475 images) cancer images. The image sizes are (300×300). Each image is of a good caliber. All the data rights for this dataset are bound by The HAM10000.

Python is employed as a programming language to run these systems because of how extensive its library is. To display statistical analyses and create visual graphs, libraries like Pandas and Matplotlib are employed. As an IDE for these systems, Google Colab and Jupyter Notebook have been used. A core i5 7th generation laptop served as the workstation for this study. Google Colab's embedded RAM and GPU are utilized along with the laptop's integrated RAM (8 GB) and graphics zotac 1050ti (4 GB). TensorFlow was released as a Python library that is open source and free. Dataflow is used to perform machine learning methods using TensorFlow. It also aided in computing values and training models. The dataset was stored with the use of Google Drive.

One of the most elegant programming paradigms ever created is neural networks. The conventional approach to programming, which divides complex problems into several tiny, precisely defined tasks that the computer can easily carry out, allows anybody to tell the computer what to do. On the other hand, consumers cannot instruct a neural network on how to resolve an issue[29]. Instead, it derives a solution to the issue on its own by learning from observable data. The CNN weight-sharing tool, which decreases the number of network parameters that may be trained, aids in preventing model overfitting, and boosts generalization, is one of the main reasons to consider it in such a scenario. The concurrent loading of input by CNN's extracting features and classification layers enhances the output's dependence on the extracted features and boosts the output's order. Three key ideas constitute the foundation of convolutional neural networks: pooling, shared weights, and local receptive fields[29].

The underlying conceptual model of CNN is shown in Figure 3.1, and many layers of that model are described in the sections that follow. A CNN-based model essentially

consists of a few processing units which are adept at handling different levels of abstraction of incoming data. Sets of data are given as input. Then several layers, such as convolutional layers, are run, then a pooling layer. A completely linked layer and an output are both present at the conclusion. The input and Fully Connected (FC) levels are separated by a number of unnoticed layers. When compared to the buried layers, the initiatory components (with less abstraction) learn and retrieve high-level parts (with greater abstraction)[38].

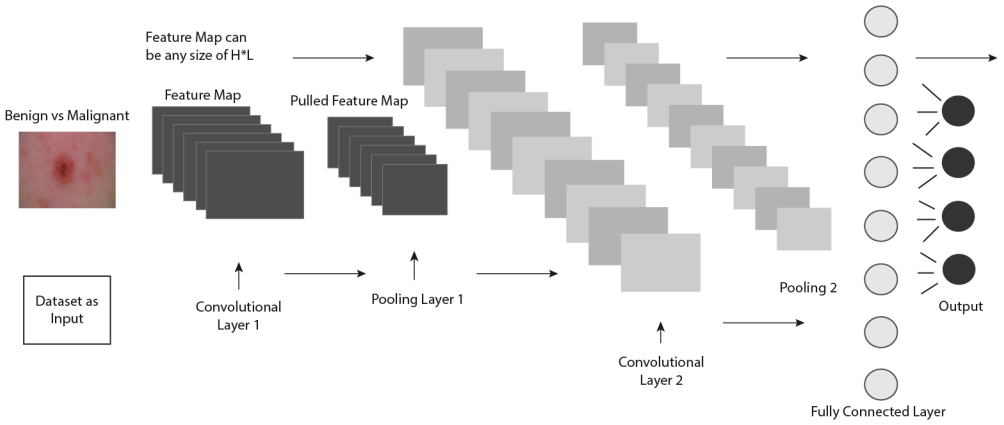


Figure 3.1: Conceptual module of CNN

Every CNN design starts with convolutional layers. It is made up of many intricately woven convolutional kernels (filters) that work together to transform the input data into an output feature map (N-dimensional parameters). An input volume exists for a single convolutional layer. Some data segments are removed from the input source, and the activation value is saved for those. The output activation volume receives the processed activation value after further processing. The use of kernel techniques is utilized for all communication. A kernel is a collection of constant or integer values, each of which corresponds to a kernel's weight. When a CNN model's training process starts, random numbers are assigned to each kernel weight. The kernel then learns to extract important information when the weights are adjusted with each training period.

Then, to break down larger picture features into smaller ones, feature maps are subsampled using pooling layers that are created using convolutional approaches. The most crucial characteristics (or information) for every pool stage are always preserved when the local features are compressed. The same steps used for convolution are used for pooling, including choosing the operation stride and the pooling region's size. Among the several distinct pooling algorithms used in various pooling layers, the maximum, average, minimum, gated, and tree pooling approaches are only a few. The most well-liked and often employed pooling strategy is max-pooling.

Fully linked layers, in which every cell in one layer is connected to every cell in the layer above it, constitute the final part of the convolution layer. The last layer of completely linked layers in the CNN architecture is the output unit. The term "FC layers" refers to a sort of feedforward artificial neural network (ANN) that performs similarly to a neural network like a perceptron (MLP) (FCLs). The fully connected

layers accept feedback based on a set of parameters following the last convolutional or pooling layer, which is then compressed to generate a variable and fed into the FC layer to produce the final CNN output.

We begin by attempting to obtain a data collection of skin lesion photos from which we can detect melanoma. The photos will next be subjected to data pre-processing. After that, we'll use Deep Neural Architectures like VGG-16 and VGG-19 to conduct image segmentation. The set of data will then be split into halves for the model's testing and training. In 2014's ILSVRC (Imagenet) competition, VGG-16, a convolutional neural network (CNN) architecture, took first place. This is frequently acknowledged as one of the most sophisticated vision model designs ever created. Throughout the design, the convolutional and max-pool layers are arranged in exactly the same way. Two completely interconnected layers and a softmax serve as the conclusion[32]. The fact that there are 16 layers with different values is indicated by the number 16 in VGG-16. This system is quite big, with an estimated 138 million components[32]. The whole VGG-16 model's 16 layers are seen in Figure3.2.

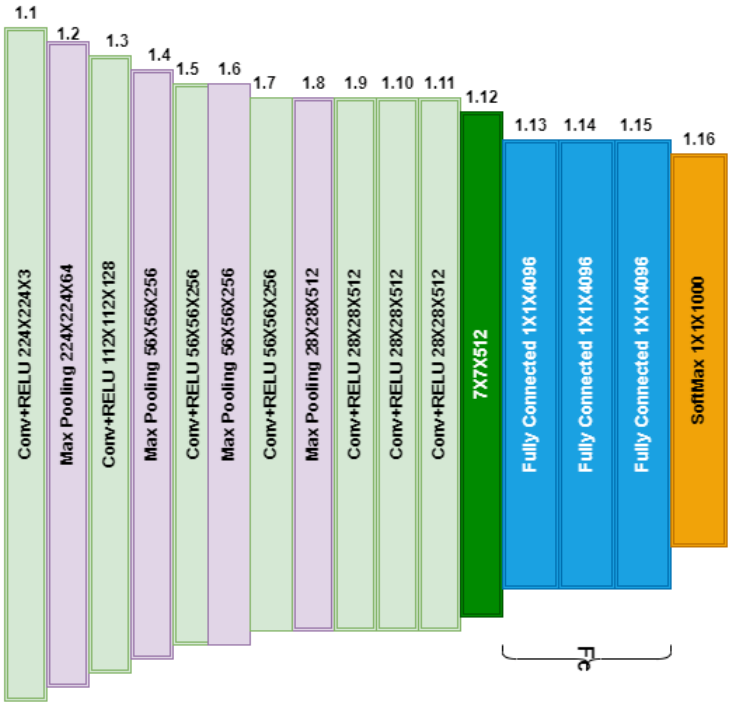


Figure 3.2: Basic architecture of VGG-16 model

There are 16 layers of VGG-16 model, and they are divided into 5 different types of layers (conv, ReLU, max-pooling, softmax, and fully connected). The convolutional layer uses a source RGB value of 224 × 224 pixels with a predetermined length. The data is altered using a number of convolutional layers, each with a 3 × 3 visual field (the least size required to maintain the concepts of top/bottom, right/left, and core). A layer called max-pooling comes after the convolutional layer. The data is processed using a combination of convolution and ReLu. In some setups, it additionally features 11 convolution filters, which are frequently seen as proportional modifications to the input network.

Our next attempt is the VGG-19 model containing 19 layers. According to reports, the VGG-19 CNN architecture can analyze huge picture datasets like ImageNet with great accuracy. The VGG-19 model, which has over 143 million parameters, was built using information from of the ImageNet dataset, which has 1.2 million object classification pictures in 1,000 different object categories[13]. There are 19 easy-to-train layers in the VGG-19, including convolutional and fully connected layers and also max pooling, dropouts, and fully connected layers. In our method, we combine a customized classification component with a learned convolution base, a densely-connected classifier, and a dropout layer for regularization. Figure 3.3 shows the VGG-19 model in its updated form.

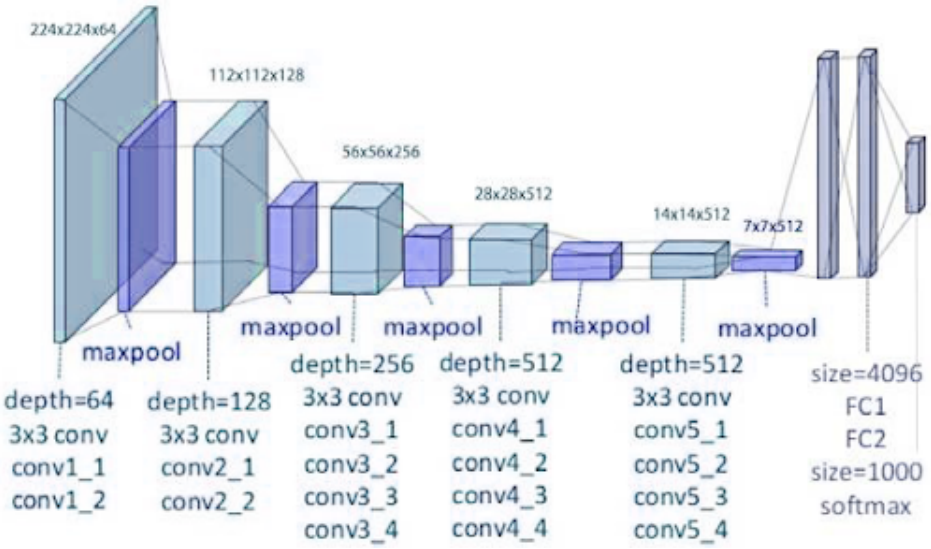


Figure 3.3: An example of the VGG-19 model's custom network design

The only preprocessing carried out was to subtract the mean RGB price from each component as calculated throughout the full training set. Utilizing kernels of (3×3) dimension with a stride size with one component, they were able to hide the whole notion of the image. Through spatial cushioning, the image's spatial resolution was maintained. A pair of max pooling was performed across a pair of element windows with stride. This model uses a rectified linear unit (ReLU) to integrate non-linearity, improve model classification, and accelerate processing, as contrasted to prior models that used tanh or sigmoid functions. Implemented three fully interconnected layers, the first two of which had a combined total parameter of 20,024,384, followed by a layer with 1,000 channels for 1000-way ILSVRC classification, and finally, a softmax performance.

Our next model is AlexNet where we implemented the same dataset for better accuracy results. In order to leverage the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)[9], Krizhevsky et al. [20] built AlexNet. The input picture is filtered using AlexNet's first layer. The required dimensions for the input picture are width (W), height (H), and depth (D); 2272273 where D = 3 denotes red, green, and blue. Stride is the name of the first convolutional layer that was used to filter the input color image. It comprises 96 kernels (K), a filter (F) of size 11×11, and 4 additional pixel(s). The stride in the kernel map is the separation between the responding field centers of adjacent neurons. The output size of the convolution

layer is calculated using the mathematical formula ((WF+2P)/S)+1, where P is the number of padded pixels, which in this case equals zero. Using this equation, the convolution layer's output volume is calculated to be ((22711+0)/4)+1 = 55. The input for the second convolutional layer will be[37] [46], and there are 256 filters in this layer. The load for each GPU is divided by two, since the work of this layer is divided between two GPUs. The convolutional layer comes next, and then the pooling layer. The pooling layer, where pooling may be Sum, Max, Average, etc., attempts to minimize the dimensionality of each feature map while maintaining key characteristics. Utilizing a max pooling layer is AlexNet. 256 filters are the layer's input. Each filter has a size of, 55256 and a stride of 2 pixels. Using two GPUs will split the work into 2727128 for each GPU, or 55/255/2256/2.

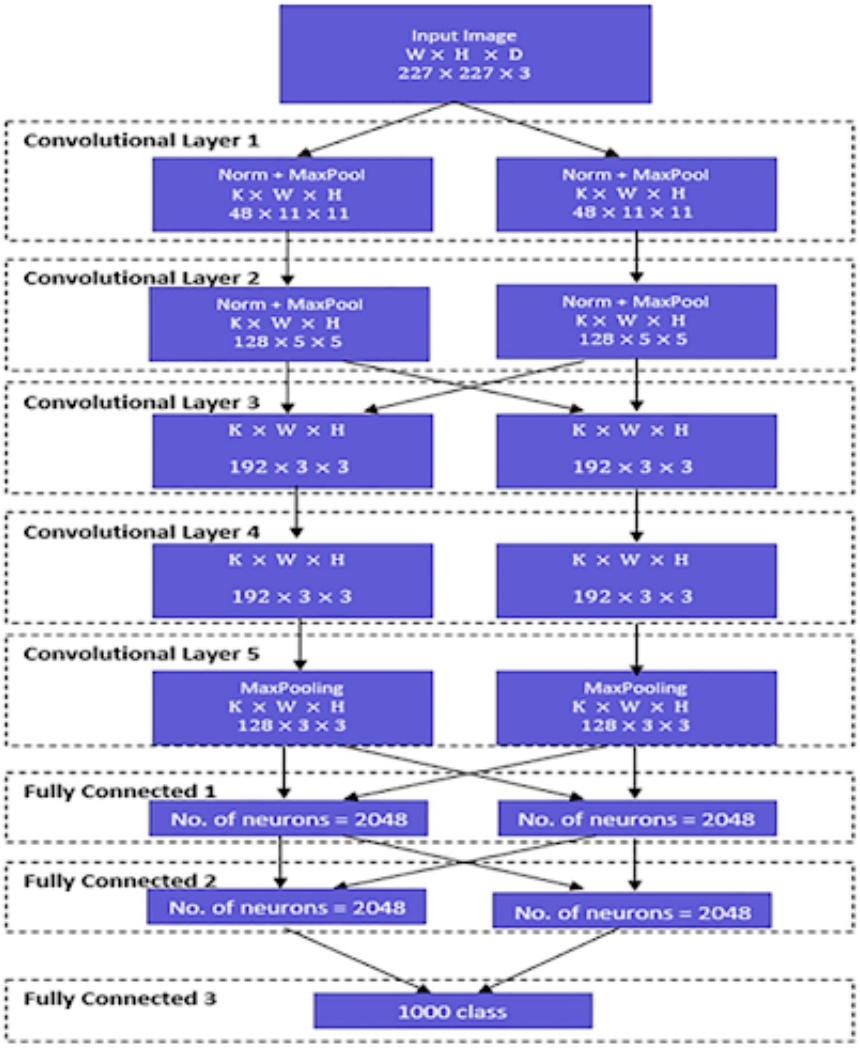


Figure 3.4: Architecture of AlexNet

The third layer is connected to the second convolutional layer's pooled and normalized output via 384 kernels, each of size 33. For the fourth convolutional layer, there are 384 kernels of size 33, and they will be split across 2 GPUs so that each GPU load is 33192. Each GPU load will be, 33128 since the fifth convolutional layer has 256 kernels, each of which is 33 in size. These kernels will be distributed across 2

GPUs. It should be emphasized that no normalizing or pooling layers are used while creating the third, fourth, or fifth convolutional layers. These 2 fully connected layers each have 4096 neurons, and they receive the output of these 3 convolutional layers as their input. Using the training dataset ImageNet[9], Figure 3.4 shows the general architecture of Alex-Net for classifying various classes.

Finally, comparing the results of the implementation, we used the custom CNN model to generate higher accuracy results. RGB images measuring 224 × 224 × 3 are used as the CNN model's input. Weights in each layer are equivalent to those in the one above it[47]. It is composed of consecutive layers that use convolutional and pooling techniques to extract patterns from images. Figure 6 depicts the construction of layers in the custom CNN model. As the depth of the network increases, convolutional layers frequently have more filters despite having fewer rows and columns. The model is constructed using layers like conv, ReLU, max-pooling, dense, and completely linked, similar to the structure described above. Conv1-Lu1-maxPool1 is a pooling layer that is always present after a convolutional and ReLU layer. As the system gets bigger, there are more parameters. The ease of use of this design is its key benefit.

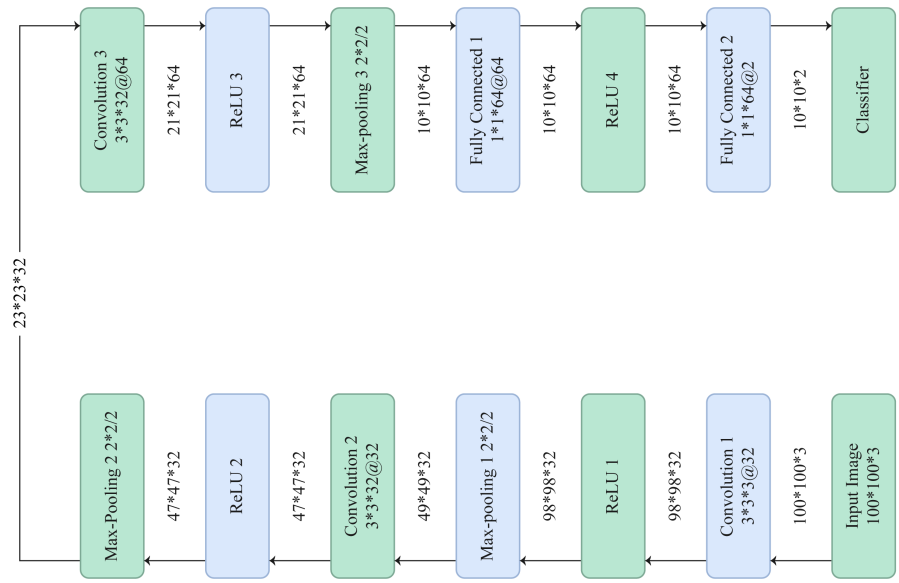


Figure 3.5: Architecture of custom CNN model

The first Conv2D parameter that is required is the amount of filter which the convolutional layer will learn. Layers early in the network design (i.e., closest to the actual input picture) will train fewer convolutional filters, whereas layers deeper inside the network (i.e., nearer to the output predictions) would learn more filters. Later Conv2D layers will learn more filters than early Conv2D layers, but fewer filters compared to the output will be learned by later layers.

Next, in neural networks, pooling is used to lower variance and simplify computation. Beginners frequently employ pooling techniques without understanding why they should. Sharp features may not be visible when using the average pooling approach, since it smoothens down the image. The brighter pixels in the picture are chosen via max pooling. It is helpful when we just care about the image's lighter pixels and the image's dark backdrop. As an illustration, the digits in the MNIST dataset are shown in white, while the background is black. Max pooling is used as a result. Again, the flatten function lowers the multi-dimensional data tensors to a single dimension in order to simulate the input layer, build the neural network model, and appropriately feed these inputs to each and every neuron in the model. Convolution layer to fully connected layer transitions frequently employs the flattened layer to reduce the multidimensional input to one dimension. Configure the layer differently depending on whether TensorSpace Model loads a pre-trained model before startup. Next, the dropout method is used in the model. By randomly removing units during training, the dropout method minimizes overfitting in neural networks. The trained model functions as an ensemble model made up of several neural networks as a consequence. To obtain the final model prediction at test time, the ensembled networks' predictions are averaged across all layers. We can model dropout as training numerous neural networks and average their prediction during testing, since certain sections of the network turn off during training.

The Rectified Linear Unit is the next most commonly used activation function in deep learning models (ReLU). The function returns 0 for any input that is negative and returns x for any input that is positive. Thus, it may be expressed as y = max (0,x). Some ReLU function characteristics. There are no difficult mathematical formulas involved, making it incredibly simple to grasp. It is not affected by the fading slope issue that mostly affects other activation functions, such as sigmoid or tanh. It features various built-in variations for certain challenging mathematical concepts, like Leaky ReLU and Parametric ReLU. Additionally, deep learning and machine learning also require activation functions, which essentially come in 7 different varieties. In machine learning, the sigmoid serves as an activation function that is used to introduce non-linearity into a model. It chooses which value to transmit as output and which not to pass, to put it another way.

$$Y = \left(\frac{1}{1 + e^{-z}} \right) \tag{3.1}$$

As a result, the expected value of y will be 1 if the value of z grows to positive infinity and 0 if it declines to negative infinity. Additionally, the label is classed as class 1 or a positive class if the sigmoid function result is larger than 0.5, and class 0 or a negative class if it is less than 0.5. Adam determines individualized education rates for different parameters using an efficient machine learning technique. Adaptive moment estimation, as Adam is known, uses estimations of the initial and second moment of the gradient to alter the learning rate for every weight of the neural network. Adam may be thought of as a mix of RMSprop and stochastic gradient descent with momentum. Similar to RMSprop, it adjusts the learning rate utilizing squared gradients and, like SGD with momentum, utilizes momentum by using the

gradient's trend line rather than the gradient itself.

The architecture of the deep learning model may be thought of as its layers. Different kinds of layers may be utilized in the models. Based on their attributes, each of these several strata has a particular significance. Convolutional layers are used in image processing, In problems involving NLP and time series analysis, among others, LSTM layers are commonly employed. The dense layer, also called as a completely connected layer, is used in the last phases of the neural network. In order to help the model more easily determine the link between both the values of the information it is working with, this layer helps to change the output's dimension from the layer before. A layer that is closely coupled to the layer above it in any neural network means that each and every neuron in that layer is related to all the neuron in the layer from above. The photos are categorized using a dense layer based on the output of the convolutional layers. Every layer of a neural network contains neurons, which calculate the weighted average of its input before processing it with a nonlinear function called an "activation function".

For the implementation of the model, we are required to install GPUs for achieving extra growth on the efficiency of the models. Installing the TensorFlow PIP package was the first step. The next step was to confirm that the CUDNN and CUDA Toolkit installations were complete. The Object detection API is more practical and effective for building models and enhancing performance. TensorFlow needs the NVIDIA GPU, CUDA Toolkit v11.2 cuDNN 8.1.0 to operate on our GPU. An environment has to be created. After that, we tested it on a different terminal to make sure everything was functional. The installation of TensorFlow is now finished. The TensorFlow Object Detection API has to then be configured on your machine. TensorFlow Model Garden was downloaded. In the TensorFlow Object Detection API, model and training parameters are set via protobuf. Before using the framework, the Protobuf library must be downloaded and built. We had to launch a fresh terminal for the modified environment variable values to take effect. Additionally, the COCO API was set up. By downloading and installing the object detection program, the Object Detection API may be set up. Run the following instructions from within TensorFlow models or research. Our installation seems to be operating as planned. By default, TensorFlow maps almost all the GPU memory of all GPUs that are visible to the process (subject to CUDA VISIBLE DEVICES). By minimizing memory fragmentation, this helps make greater use of the device's relatively valuable GPU memory resources. TensorFlow has been constrained to a certain subset of GPUs using the tf command.

Although TensorFlow can be installed and used without Anaconda, we choose it because of how simple it makes it to manage packages and create new virtual machines. Prior to version 20.3, pip installs a package together with any necessary Python packages without first checking for incompatibilities. Despite the continuing scenario, it would install a package and any required dependencies. TensorFlow might become unusable if another package other than TensorFlow through pip is installed, since a different version of the NumPy library is needed. Even when a package seems to be operating, several things might happen. The conda package manager has historically been distinct from pip because of this. There are numer-

ous platforms, such as Anaconda Cloud, PyPI, and other repositories, where one may distribute bespoke conda packages. Anaconda2 and Anaconda3 both contain Python versions 2.7 and 3.7, respectively. However, using Python versions that have been conda-packaged, new environments may be made. To install Anaconda, we had to open a new terminal window and enter the appropriate command. In subsequent phases, an Anaconda virtualization environment was required.

Chapter 4

Input and Training Dataset

4.1 Input Dataset

The modest size and lack of diversity of the current dermatoscopy image collection hinder the creation of neural networks again for automatic analysis of pigmented lesions. The authors dealt with this problem using the HAM10000 ("Human Versus Machine with 10000 trained photos") dataset. We have used HAM10000 dataset to put our thesis into practice[12]. The authors collected images of dermatoscopy taken and preserved using a variety of modalities from different populations. The final sample, which may be utilized as just a training dataset for machine learning based on the study, consists of 10,000 dermatoscopy images. The dataset is split into two parts: test (950 photos), which is used to assess the correctness of the trained models, and training (9,050 images), which is used to the models are trained. More than 50% of lesions were verified by histopathology (histo), with obey exams, expert opinion, and verification by in-vivo fluorescence microscope giving the ground truth (confocal) for the other cases[12]. Malignant and benign cancer photos may be found in the train section (a combined total of 4,525 images). Additionally, there are 475 photos of both malignant and benign cancer in the test section. The picture dimensions are (300×300). Each picture is of high quality. Lesions in the dataset with plenty of photos may be tracked using the lesion id field in the HAM10000 metadata file[12].

4.2 Training Dataset

During the primary analysis, we started our implementation part with the custom CNN model. The input libraries and layers that the model required were,

4.2.1 Conv2D

In order to generate a tensor of outputs, this layer generates a convolution kernel that is highly integrated with the input of the layer. If the utilize bias parameter is set to True, a bias vector is built and added to the outputs. Last but just not least, it is delivered to the outputs if activating is not None. In order to construct a tensor of outputs, this layer primarily generates a convolution kernel that's also combined with the layer input. Again, many of the convolutional layers were interleaved with

nonlinearity. ReLU is essentially used to filter information as it travels across the network in the forward direction. If your input is negative, it basically makes it zero by performing an element-wise operation on it.

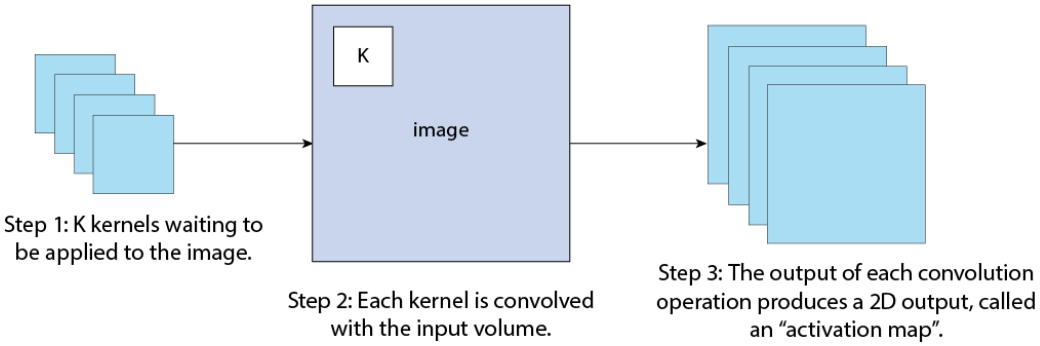


Figure 4.1: Conv2D architecture

4.2.2 MaxPool2d

The PyTorch class known as MaxPool2d is used in neural networks to pool across defined signal inputs that internally comprise different input planes. In the class definition, it accepts a number of options, such as dilation, ceil mode, size of kernel, stride, dilation, padding, and return indices. Max pooling helps to reduce overfitting by providing an abstracted representation. Additionally, by reducing the number of parameters that must be learned, it grants the internal representation basic translation invariance and reduces computational expense.

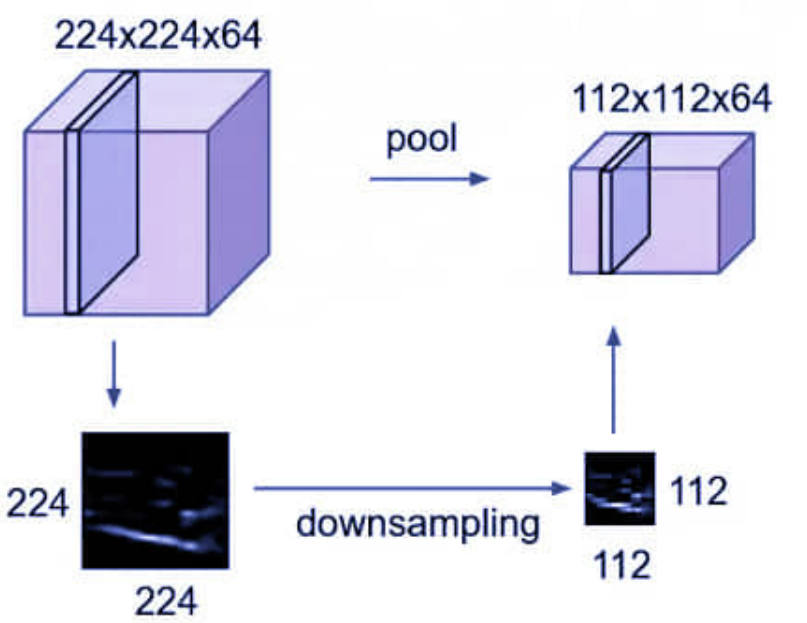


Figure 4.2: MaxPool2d workflow diagram

4.2.3 Flatten Layer

The input's depth perception is condensed to its developmental continuum using a flatten layer. It must have the shape of a linear vector with one dimension. Direct inputs cannot be of the rectangular or cubical form. And for this reason, we require layers that are totally linked and flat. Data is flattened when it is made into a 1-dimensional array for input into the following layer. The TensorFlow flatten function, which can be found in the library, lowers the input data's two dimensions to only one. The batch size is unaffected when doing this.

4.2.4 Dense layer

In any neural network, a layer that is highly linked to the one above it indicates that every neuron within this layer is connected to every neuron in the layer from above. In artificial neural network networks, this layer is the one that is most frequently utilized. As a result, each neuron in a dense layer is employed to change the vectors' dimension. Every neuron in the thick layer receives information from every neuron in the preceding layers, as was previously stated.

4.2.5 Dropout layer

Dropout, a training method, randomly disregards certain neurons. They drop out at random. As a result, any weight modifications are not provided to the cell on the return trip, and their effects are only briefly reduced on the activation of the following neurons. The Dropout layer arbitrarily changes input values to 0 at such a rate of rate at each step throughout training in order to prevent overfitting. Scaling non-zero inputs up by 1 maintains the sum of all inputs (1 – rate). In artificial neural networks, it largely avoids overfitting by randomly removing units during training.

Next, we implemented the model VGG-16 and VGG-19. The Oxford-based Visual Geometry Group is known as the VGG model. The model was simpler than AlexNet and had more depth. Two models with 16 and 19 layers each were shown in the article. Each and every CNN layer used 3 by 3 filters with stride, a pad of size 1, a maximum pooling size of 2, and stride 2. The 16 convolutional layers of VGGNet-16 are quite attractive, and its architecture is very consistent. It contains several filters but just 3×3 convolutions, like AlexNet. On 4 GPUs, it may be taught for two to three weeks. It is presently regarded by the community as the top method for drawing traits from photographs. The VGGNet's weight configuration is publicly available and has been used as a conventional feature representation in a variety of applications and problems. However, managing the 138 million parameters in VGGNet might be a bit challenging. VGG is possible with transfer learning. In which the parameters are updated for improved accuracy, and you may utilize the parameter values once the model has been pre-trained on a dataset. In spite of the fact that many new and enhanced scoring models have been created since VGG was initially introduced, data researchers and scientists from all over the world are still interested in VGG16. Here seem to be a few real-world examples of VGG16 applications.

4.2.6 Image Recognition

X-rays and MRIs are examples of medical imaging methods that may be used to identify illnesses using VGG16. While driving, it can also be utilized to read street signs.

4.2.7 Image Detection

In certain instances involving image detection, it can perform quite well.

4.2.8 Image Embedding Vectors

After blowing out the highest output layer of the Siamese network, the model could be employed to train to create image embed vectors that may be used for a job like a face recognition.
This model requires a Dense layer and dropout layer, which we already discussed earlier. Because it necessitates correlations between extracted features and categories, GlobalAveragePooling2D has also been employed in this model because it is more naturally compatible with the convolution structure. That is why the feature maps may be simply understood as confidence maps for categories.

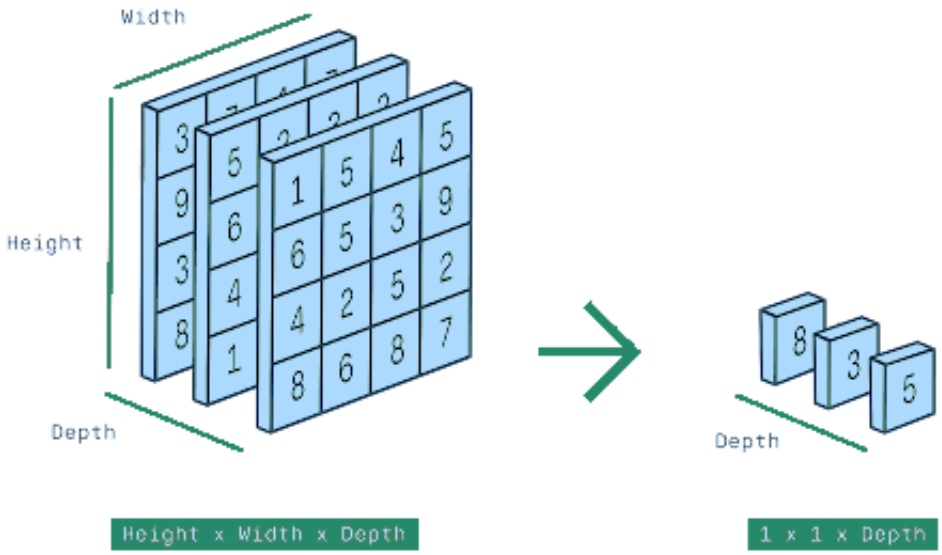


Figure 4.3: Operation of 2D global average pooling

Use global average pooling blocks instead of your convolutional neural network's final pooling block's flattening block. You may substitute 2D Global Average Pooling Blocks for CNN's fully linked blocks. Again, VGG-19 has also been used in the implementation to compare the results in identifications of melanoma. Modern CNN With layers that have previously undergone training, VGG19 has a solid understanding of the shape, color, and structural aspects of an image. For difficult classification tasks, the very deep VGG19 has indeed been trained on an enormous variety of images. The VGG-19 model is an improvement over the VGG-16. Convolutional neural network model with 19 layers. Convolutions are stacked together

to create the model, however a problem known as decreasing gradient restricts the model's depth. Deep convolutional networks are challenging to train due to this concern.

Later, we implemented AlexNet to show better performance in detecting melanoma. We used sufficient algorithms and libraries to generate better results. Such as,

4.2.9 ReLU

The perceptron most frequently utilized in deep learning models is Rectified Linear Units. The function returns 0 for any input that is negative and returns x for any input that is positive. ReLU is the maximum function given input vectors, such as a vector from a convolved picture. Then, ReLU sets all negative numbers in the matrix x to zero, while maintaining all other values constant. After the convolution, ReLU, a non-linear activation function like a sigmoid, is produced. If the user input is positive, the corrected activation function, or ReLU in short, will output the user input; if the user input is negative, it will create zero.

4.2.10 AdaptiveAvgPool2d

The stride and kernel size are effectively specified by the user in average-pooling or max-pooling, where they are set as hyper-parameters. If the input size changes, it will need to be reconfigured. On the other hand, using adaptive pooling, we specify the output size. Additionally, the stride and kernel size is automatically chosen to meet the requirements. Simply described, adaptive average pooling is an average pooling process that determines the appropriate kernel size required to create an output of the specified dimensionality from the input supplied, given any input and output dimensions.

4.2.11 Linear layer

The output features are produced by multiplying the linear layer's input properties by a weight matrix. The input characteristics are sent to a linear layer as a one-dimensional tensor that has been flattened, and they are then multiplied by the weight matrix. Linear layers without bias may learn the overall average of association between the input and the output. For instance, w would be positive if y and x have a positive correlation, and w will indeed be negative if y and x have a negative correlation. The value of w will be very close to 0 if y and x have no connection at all.

Furthermore, we also required the Conv2D layer for generating a convolution kernel, which is convolved with the layer input. Also used maxpool2D for lowering the parameters. Lastly, the dropout layer was also used for avoiding overfitting.

Lastly, the proposed model of our research is the custom CNN model. But the unique optimizer that is used for building the model with more efficiency and for higher accuracy is the Adam optimizer. Providing the "optimal" architecture in regard to a list of restrictions or priorities is the aim of optimization. The maximization of factors like production, tenacity, reliability, endurance, efficacy, and

utilization is among them. The stochastic gradient descent variant known as the Adam optimization technique has lately gained more popularity for deep learning applications. Rather than stochastic gradient descent, a new optimization method called Adam could be employed to train deep learning models. Adam brings together the best elements of the RMSProp and AdaGrad algorithms to produce an optimization method that can manage sparse gradient in noisy settings. The Adam optimizer is faster to calculate, has fewer tuning parameters, and often gives better results than traditional optimization techniques. As a result of all of that, Adam is recommended as that of the preferred optimization again for the majority of applications.

Chapter 5

Result Analysis

In this research for detecting melanoma, and it's classification, we have implemented our models with the HAM10000 dataset. We implemented VGG-16 and VGG-19, AlexNet and the custom CNN model. This system was run on Google Colab and Jupyter Notebook. The backend GPU of the Python 3 Google compute engine (12.69 GB of shared RAM). The dataset was put on Google Drive so that Google Colab could easily utilize it. The systems all functioned flawlessly and provided us with the outcomes we required, which are covered in this part.

First, we use a simple convolutional neural network VGG-16 model using transfer learning to address the melanoma classification. The model is trained with several hyper-parameters to achieve accuracy, and binary cross-entropy is employed as the loss function. Again, Adam optimizer is used to optimize each pre-trained model, with a batch size limit of 15 as well as a learning rate of 0.00001. (1104). The total number of epochs differs for the top four pre-trained CNN architectures. Thirty training epochs are used to train these pre-trained models. These hyper-parameters are all picked up via experience. The accuracy measure, which is the common metric for classification issues, is defined as follows:

$$Accuracy = \left(\frac{TruePositive + TrueNegative}{TruePositive + TrueNegative + FalsePositive + FalseNegative} \right) \tag{5.1}$$

When compared to other skin malignant disorders, ABCD rule identifies melanoma mostly by gaining the minimal modifications. Your general practitioner is worried about your skin, so if any changes take place, they will advise you to see a dermatologist who is skilled in identifying skin cancer. Asymmetry: Size and form differences between the two parts. Some of the skin's borders are distorted and uneven. Color: The skin has black, pink, and brown color variations. Diameter – Tell your doctor if the size of your melanoma changes at all. Typically, it is 6 mm in size. This method of VGG-16 using conv, maxpooling, softmax and fully connected layers achieved overall test accuracy of 74.05% respectively. The results are shown below,

This figure 5.1 depicts the accuracy results gained from VGG-16. We have run on

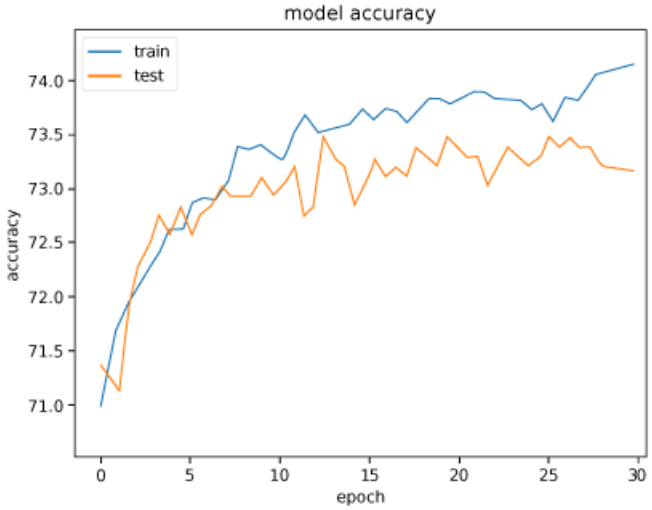


Figure 5.1: Accuracy graph of VGG-16

30 epochs and batch size 15 during the runtime. And overall we achieved a test accuracy of 0.7405.

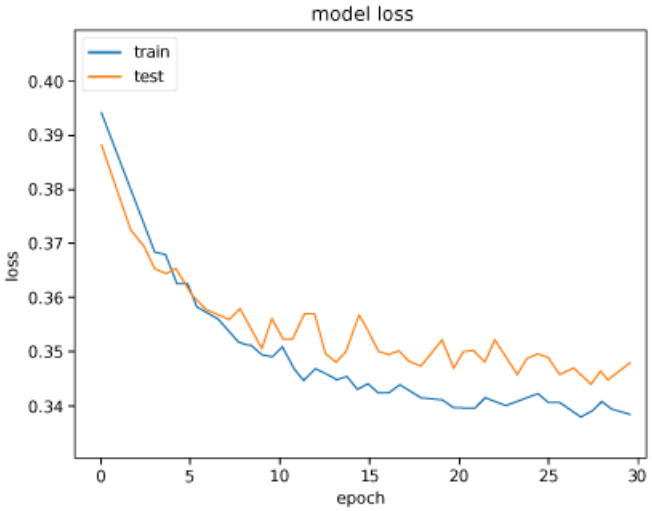


Figure 5.2: Loss function graph of VGG-16

The figure 5.2 depicts the loss function that we gained from VGG-16 during the runtime, which gradually decreased within the end of the epochs. In most cases, we see that the loss does not diminish in the initial few starting epochs when training a neural network. There may be several causes for this. Low learning rates are prevalent for high regularization parameter. It is observable that, after each epoch, the distance between train loss and test loss. This is due to the fact that once the network understands the data and lowers the regularization loss, there is barely any difference among test and train loss (model weights). The model is still more precise on the training set, though.

Next, we applied the VGG-19 using Convolution layers, max pool layers, softmax layers, Global Average Pooling, and fully connected layers achieving overall test accuracy of 77.36% respectively. The results are shown below,

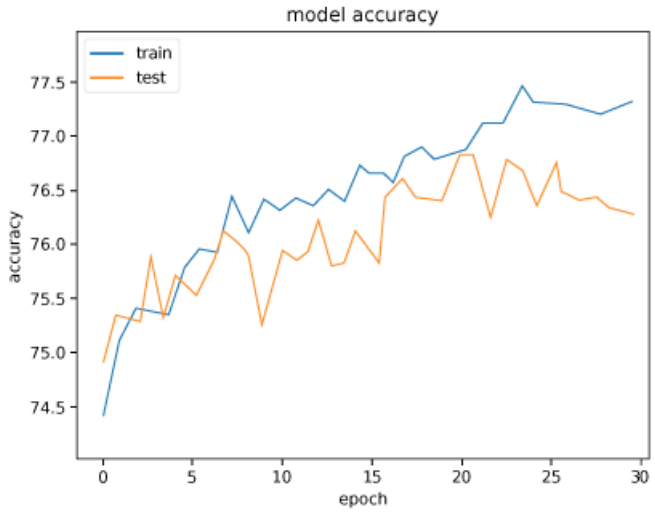


Figure 5.3: Accuracy graph of VGG-19

This figure 5.3 the accuracy results gained from the VGG-19 model, which scored 77.36%. Because the accuracy is less cumulative than, say, mean squared error, it can be seen to be leaking at some places. Because of this, accuracy is inconsistent while the loss is growing quickly. This essentially implies that a fraction of cases is categorized randomly, which results in oscillations since the proportion of accurate random guesses varies over time. Overfitting is typically defined as being sensitive to noise (when classification results in random results).

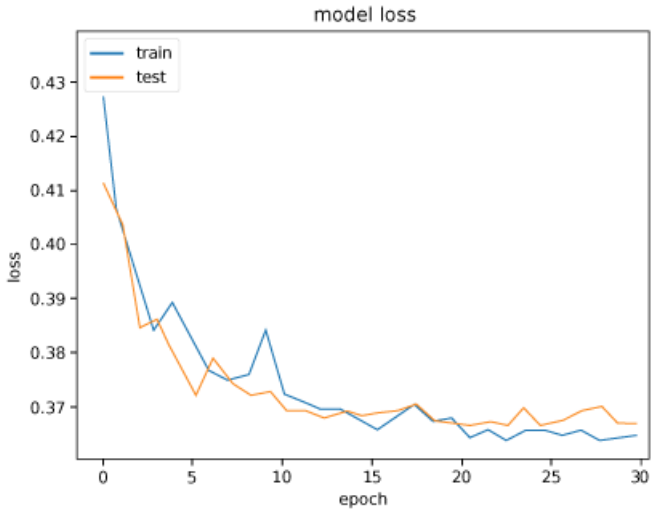


Figure 5.4: Loss function graph of VGG-19

While the training loss demonstrates how well the model matches training data in figure 5.4, the validation loss demonstrates how well it fits new data. The test loss finally drops below the train loss since normalization terms are only used when training the model just on the training dataset, inflating the training loss. Your loss function only includes prediction errors during training and testing, which often results in a lower loss than the training set. And to the contrary, this may point to underfitting in the model. Underfitting happens when a model is unable to accurately reflect the training collection of data, leading to significant errors.

Next, we implemented the convolutional neural network AlexNet model to address the melanoma classification. The model is trained with several hyper-parameters to achieve accuracy, and binary cross-entropy is employed as the loss function. All pre-trained models are optimized using Adam, with a batch size of 15. The model has five layers, and all of them aside from the output layer consist of a combination of max pooling, fully linked layers, and ReLU activation. The results are shown below,

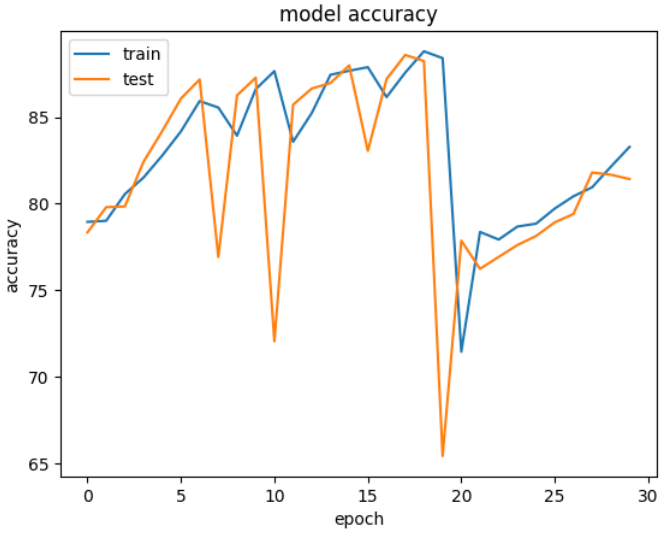


Figure 5.5: Accuracy graph of AlexNet

Beyond a certain point, dropout causes the model to no longer be able to fit data correctly. It makes sense that a larger dropout rate would cause some layers' variation to increase, which would worsen training. Dropout has the same effect on model capacity as all other types of regularization. That is why we can see many drops in the test function of the accuracy curve in AlexNet model, although we gained an accuracy result of 81.42%.

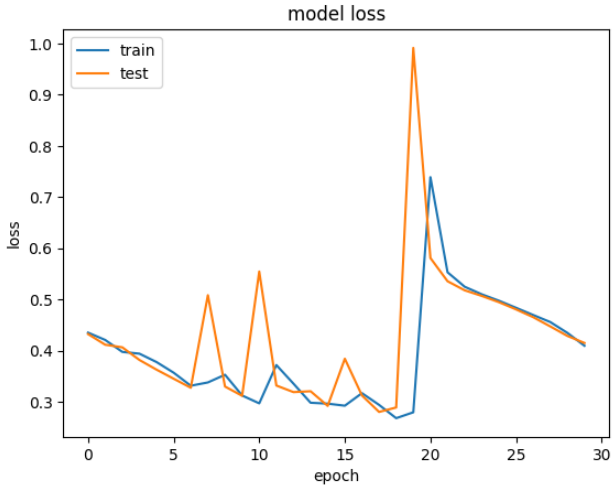


Figure 5.6: Loss function graph of AlexNet

As we can see, in the accuracy curve, the test, and train function falls correspond-

ingly during the approximated epoch 19. The same ways the loss function curve raised the same time due to the learning rate being high at that moment.

Next, we implemented the custom convolutional neural network model to address the melanoma classification. It is composed of consecutive layers that use convolutional and pooling techniques to extract patterns from images. Similar to the above-mentioned structure, the model is built utilizing layers like ReLU, conv, dense, max-pooling and entirely connected. After a convolutional and ReLU layer, there is usually a pooling layer called Conv1-Lu1-maxPool1.

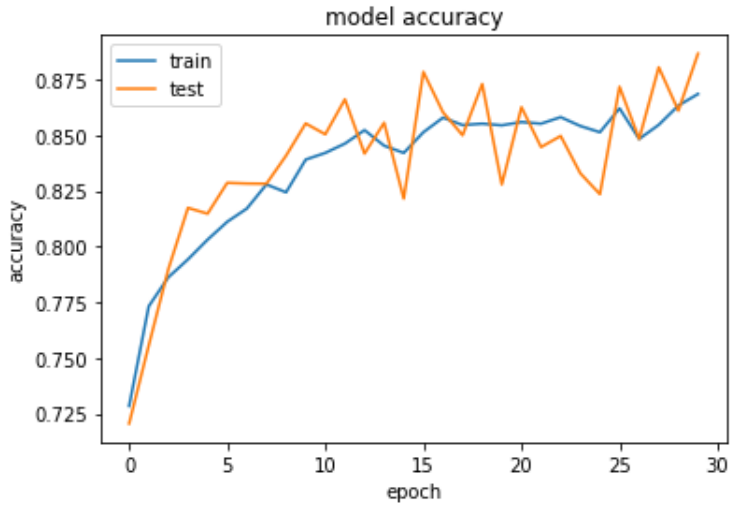


Figure 5.7: Accuracy graph of custom CNN

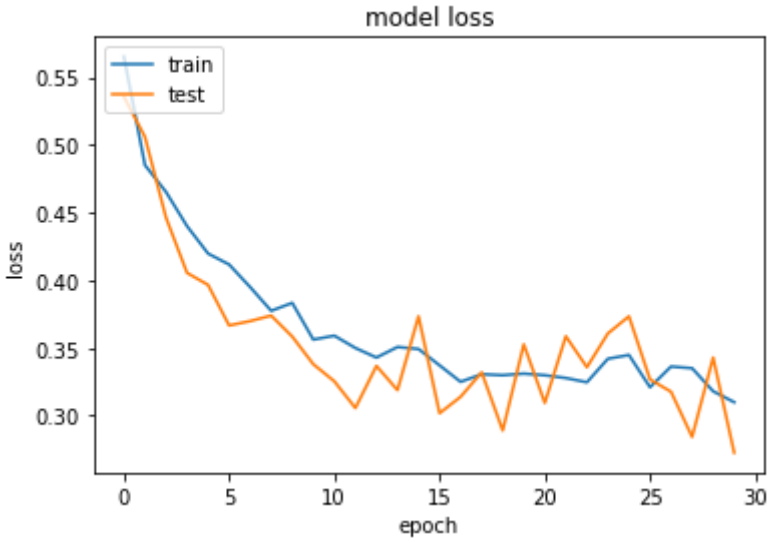


Figure 5.8: Loss function graph of custom CNN

By dividing the entire sample count by the number of reliable forecasts, we may calculate accuracy. According to the outcome, our model has an 88.66% accuracy rate on this classification problem. We must be aware that model performance encompasses much more than model correctness. Because the accuracy of the model

and its performance are inversely correlated, the more accurate the forecasts, the greater the model's performance.

When training a model, accuracy is typically taken into account and given attention. Loss on 5.8, on the other hand, requires equal attention. By design, the accuracy rating is the number of reliable forecasts. Values that deviate from the planned objective state are considered losses(s).

	precision	recall	f1-score
0	0.91	0.88	0.90
1	0.88	0.92	0.90
accuracy			0.90
macro avg	0.90	0.90	0.90
weighted avg	0.90	0.90	0.90

Table 5.1: Precision, recall and f1-score in custom CNN

The level of agreement between measurements of the same item is known as precision. Precision does not need accuracy. In other words, it is possible to be very exact but not particularly accurate, as well as accurate without being exact. The most accurate and precise scientific observations are made. Recall is defined as the proportion of Positive specimens among all Positive specimens that were correctly identified as Positive. Recall measures how well the model can differentiate Positive samples. As the more specimens are identified, the recall rises. The f1-score is also one of the most important evaluation metrics in machine learning. It succinctly distills a model's predictive power by merging accuracy and recall, two metrics that ordinarily compete with one another. The f1-score is the modulation index of recall and accuracy. It serves as a statistical tool for assessing performance. In other words, a person's performance is measured by their average of two factors: accuracy and recall. Here, in the table, we gain the precision value of benign(0) 0.91 and malignant(1) 0.88. Again, for recall value, we achieved 0.88 for benign and 0.92 for malignant. And finally, we gained the f1-score same for both types, which is 0.90. One parameter for assessing classification models is accuracy. The model's performance across all classes is often described by its accuracy metric. The average is returned by the macro avgerage, which computes F1 for each label in the dataset without taking the proportion into account. However, weighted average calculates F1 for each label in the dataset and returns the average while taking each label's share into account. The accuracy, macro avg, and weighted avg have the same value in all terms which is 0.90.
The performance of a classification model, often known as a classifier, using a collection of test data for which the true numbers were known is usually demonstrated using a confusion matrix. Whereas the confusion matrix alone is extremely simple to grasp, the terminology used to describe it could be more challenging. Using a confusion matrix to assess a classifier's performance on the data set. The diagonal parts display the percentage of points where the predicted label equals the actual label, and the off-diagonal portions reveal the percentage of inaccurate labels the classifier assigned. The diagonal values of the confusion matrix should be large, indicating many precise predictions. This confusion matrix above depicts the predicted label

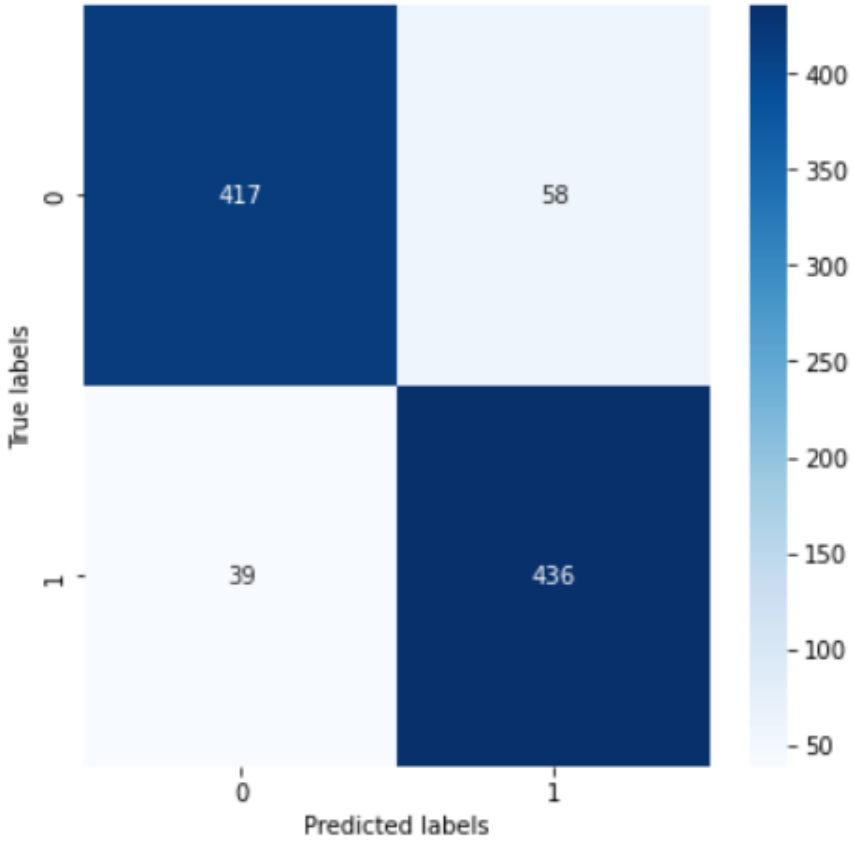


Figure 5.9: Confusion matrix of custom CNN

matches and incorrect labels during the classification of melanoma. Here, 0 states benign and 1 states malignant. The benign success rate is 417 whereas the negative is 58. On the other hand, malignant has a success rate of 436 and the negative is 39.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

A total of 91,270 American adults were diagnosed with cutaneous melanoma in 2015, according to data from the American Cancer Society. Each year, there are more than 90,000 new instances of melanoma in the European Union. Even though melanoma makes up just around 1% of all skin cancers, it is the most lethal. As one of the skin cancers with the quickest rate of growth, melanoma, early diagnosis is essential since it may aid and strongly propose particular and effective treatment regimens[43]. Melanoma is the most serious type of skin cancer, yet it can be fatal if caught early enough. Utilizing supplementary imaging methods that have been demonstrated to help in diagnosis is therefore imperative. These techniques are based on methods created by medical professionals to find melanoma at such an early stage. More than ever, it's critical to focus on the early diagnosis of skin conditions like melanoma when they spread to certain other body organs. Although this clue has been clinically proven to be a good criterion for detecting melanoma, it has yet to be investigated in autonomous melanoma detection systems. As a result, a study might be conducted to discover a strange skin lesion among a group of lesions. Those working in the field of medical imaging have faced a severe problem: a shortage of labeled data with which to train their systems. As a result, deep learning could be utilized to address this concern. We created a skin diagnosis system using deep learning algorithms in this study. Statistical analytic methods were used to examine and test the proposed methodology. In this study, we present an updated technique for melanoma skin cancer diagnosis that may be applied to any worrisome lesion. This research's objectives include determining the melanoma's precise prognosis and classifying skin cancer as either malignant or non-malignant melanoma with the right model. The obtained findings for melanoma skin cancer diagnosis show that the suggested technique performs well overall and with high accuracy: at the segmentation stage, the accuracy of VGG-16, VGG-19, AlexNet, custom CNN models are respectively 74.05%, 77.36%, 81.42%, and 88.66% respectively.

6.2 Future Work

In order to identify skin cancer from dermoscopy pictures, a deep convolutional neural network was developed in this paper. Melanoma and non-melanoma cancers were included in the data. The findings demonstrated the use of deep learning

in detecting skin cancer. This method can be used in cellphones to enable skin cancer self-diagnosis. Additionally, it can be used in smartphone-aided systems to help dermatologists find cancerous tumors. This comparison may be extended to more advanced convolutional neural network models in the future. Future work on the system will be focusing on improving the clinical measuring process and data processing, increasing the light collecting efficiency and SNR, and creating a probe that is smaller and more versatile. Future studies may benefit from the data on deep learning models for skin cancer obtained in this work to increase the accuracy of melanoma skin cancer detection. Additionally, we want to apply picture augmentation to boost accuracy and efficiency. Since only one form of skin cancer is the subject of this study article, additional classifications of skin cancer can be researched using the same techniques. The technology is capable of applying to big datasets. This will be helping with the creation of more accurate picture classification algorithms for skin cancer diagnosis.

Bibliography

[1] F. Nachbar, W. Stolz, T. Merkle, *et al.*, "The abcd rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions," *Journal of the American Academy of Dermatology*, vol. 30, no. 4, pp. 551–559, 1994.

[2] T. Lee, V. Ng, R. Gallagher, A. Coldman, and D. McLean, "Dullrazor®: A software approach to hair removal from images," *Computers in biology and medicine*, vol. 27, no. 6, pp. 533–543, 1997.

[3] G. Argenziano, H. P. Soyer, V. De Giorgio, *et al.*, "Interactive atlas of dermoscopy," 2000.

[4] G. Day, "How blurry is that border? an investigation into algorithmic reproduction of skin lesion border cut-off," *Computerized Medical Imaging and Graphics*, vol. 24, no. 2, pp. 69–72, 2000.

[5] A. F. Jerant, J. T. Johnson, C. D. Sheridan, and T. J. Caffrey, "Early detection and treatment of skin cancer," *American family physician*, vol. 62, no. 2, pp. 357–368, 2000.

[6] C. M. Balch, A. C. Buzaid, S.-J. Soong, *et al.*, "Final version of the american joint committee on cancer staging system for cutaneous melanoma," *Journal of Clinical Oncology*, vol. 19, no. 16, pp. 3635–3648, 2001.

[7] G. Argenziano, H. P. Soyer, S. Chimenti, *et al.*, "Dermoscopy of pigmented skin lesions: Results of a consensus meeting via the internet," *Journal of the American Academy of Dermatology*, vol. 48, no. 5, pp. 679–693, 2003.

[8] I. Zalaudek, G. Argenziano, H. Soyer, *et al.*, "Three-point checklist of dermoscopy: An open internet study," *British journal of dermatology*, vol. 154, no. 3, pp. 431–437, 2006.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database, in 'cvpr09'," *IEEE Computer Society, Miami, Florida, USA*, pp. 248–255, 2009.

[10] Q. Abbas, M. E. Celebi, and I. F. Garcıa, "Hair removal methods: A comparative study for dermoscopy images," *Biomedical Signal Processing and Control*, vol. 6, no. 4, pp. 395–404, 2011.

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

[12] E. S. Flores, M. Cordova, K. Kose, *et al.*, "Intraoperative imaging during mohs surgery with reflectance confocal microscopy: Initial clinical experience," *Journal of biomedical optics*, vol. 20, no. 6, p. 061 103, 2015.

[13] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating very deep convolutional networks for classification and detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 1943–1955, 2015.

[14] C. Garbe, K. Peris, A. Hauschild, *et al.*, "Diagnosis and treatment of melanoma. european consensus-based interdisciplinary guideline–update 2016," *European journal of cancer*, vol. 63, pp. 201–217, 2016.

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

[16] J. Kawahara, A. BenTaieb, and G. Hamarneh, "Deep features to classify skin lesions," in *2016 IEEE 13th international symposium on biomedical imaging (ISBI)*, IEEE, 2016, pp. 1397–1400.

[17] H. Chang, "Skin cancer reorganization and classification with deep neural network," *arXiv preprint arXiv:1703.00534*, 2017.

[18] N. C. Codella, Q.-B. Nguyen, S. Pankanti, *et al.*, "Deep learning ensembles for melanoma recognition in dermoscopy images," *IBM Journal of Research and Development*, vol. 61, no. 4/5, pp. 5–1, 2017.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[21] A. Menegola, M. Fornaciali, R. Pires, F. V. Bittencourt, S. Avila, and E. Valle, "Knowledge transfer for melanoma screening with deep learning," in *2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017)*, IEEE, 2017, pp. 297–300.

[22] Y.-Z. Yoon, J. M. Kang, Y. Kwon, *et al.*, "Cuff-less blood pressure estimation using pulse waveform analysis and pulse arrival time," *IEEE journal of biomedical and health informatics*, vol. 22, no. 4, pp. 1068–1074, 2017.

[23] H. A. Haenssle, C. Fink, R. Schneiderbauer, *et al.*, "Man against machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," *Annals of oncology*, vol. 29, no. 8, pp. 1836–1842, 2018.

[24] B. Harangi, A. Baran, and A. Hajdu, "Classification of skin lesions using an ensemble of deep neural networks," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, IEEE, 2018, pp. 2575–2578.

[25] Y. Li and L. Shen, "Skin lesion analysis towards melanoma detection using deep learning network," *Sensors*, vol. 18, no. 2, p. 556, 2018.

[26] Y. Li and L. Shen, "Skin lesion analysis towards melanoma detection using deep learning network," *Sensors*, vol. 18, no. 2, p. 556, 2018.

[27] C. Yu, S. Yang, W. Kim, *et al.*, "Acral melanoma detection using a convolutional neural network for dermoscopy images," *PloS one*, vol. 13, no. 3, e0193321, 2018.

[28] D. Bisla, A. Choromanska, R. S. Berman, J. A. Stein, and D. Polsky, "Towards automated melanoma detection with deep learning: Data purification and augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[29] Z. Cao, L. Duan, G. Yang, T. Yue, and Q. Chen, "An experimental study on breast lesion detection and classification from ultrasound images using deep learning architectures," *BMC medical imaging*, vol. 19, no. 1, pp. 1–9, 2019.

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

[31] M. A. A. Milton, "Automated skin lesion classification using ensemble of deep neural networks in isic 2018: Skin lesion analysis towards melanoma detection challenge," *arXiv preprint arXiv:1901.10802*, 2019.

[32] V. Pandiyan, P. Murugan, T. Tjahjowidodo, W. Caesarendra, O. M. Manyar, and D. J. H. Then, "In-process virtual verification of weld seam removal in robotic abrasive belt grinding process using deep learning," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 477–487, 2019.

[33] B. Parmar and B. Talati, "Automated melanoma types and stages classification for dermoscopy images," in *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, IEEE, vol. 1, 2019, pp. 1–7.

[34] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[35] R. A. Smith, K. S. Andrews, D. Brooks, *et al.*, "Cancer screening in the united states, 2019: A review of current american cancer society guidelines and current issues in cancer screening," *CA: a cancer journal for clinicians*, vol. 69, no. 3, pp. 184–210, 2019.

[36] E. Vocaturo, D. Perna, and E. Zumpano, "Machine learning techniques for automated melanoma detection," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, 2019, pp. 2310–2317.

[37] S. Banerjee, S. K. Singh, A. Chakraborty, A. Das, and R. Bag, "Melanoma diagnosis using deep learning and fuzzy logic," *Diagnostics*, vol. 10, no. 8, p. 577, 2020.

[38] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, "Fundamental concepts of convolutional neural network," in *Recent trends and advances in artificial intelligence and Internet of Things*, Springer, 2020, pp. 519–567.

[39] S. S. Han, I. J. Moon, W. Lim, *et al.*, "Keratinocytic skin cancer detection on the face using region-based convolutional neural network," *JAMA dermatology*, vol. 156, no. 1, pp. 29–37, 2020.

[40] S. Lee, Y. Chu, S. Yoo, *et al.*, "Augmented decision-making for acral lentiginous melanoma detection using deep convolutional neural networks," *Journal of the European Academy of Dermatology and Venereology*, vol. 34, no. 8, pp. 1842–1850, 2020.

[41] S. Osowski and T. Les, "Deep learning ensemble for melanoma recognition," in *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, pp. 1–7.

[42] N. Rezaoana, M. S. Hossain, and K. Andersson, "Detection and classification of skin cancer by using a parallel cnn model," in *2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*, IEEE, 2020, pp. 380–386.

[43] D. N. Thanh, V. Prasath, L. M. Hieu, and N. N. Hien, "Melanoma skin cancer detection method based on adaptive principal curvature, colour normalisation and feature extraction with the abcd rule," *Journal of Digital Imaging*, vol. 33, no. 3, pp. 574–585, 2020.

[44] A. Adegun and S. Viriri, "Deep learning techniques for skin lesion analysis and melanoma cancer detection: A survey of state-of-the-art," *Artificial Intelligence Review*, vol. 54, no. 2, pp. 811–841, 2021.

[45] F. W. Alsaade, T. H. Aldhyani, and M. H. Al-Adhaileh, "Developing a recognition system for diagnosing melanoma skin lesions using artificial intelligence algorithms," *Computational and mathematical methods in medicine*, vol. 2021, 2021.

[46] M. Dildar, S. Akram, M. Irfan, *et al.*, "Skin cancer detection: A review using deep learning techniques," *International journal of environmental research and public health*, vol. 18, no. 10, p. 5479, 2021.

[47] C. G. Pachón, D. M. Ballesteros, and D. Renza, "Fake banknote recognition using deep learning," *Applied Sciences*, vol. 11, no. 3, p. 1281, 2021.

[48] R. Raza, F. Zulfiqar, S. Tariq, G. B. Anwar, A. B. Sargano, and Z. Habib, "Melanoma classification from dermoscopy images using ensemble of convolutional neural networks," *Mathematics*, vol. 10, no. 1, p. 26, 2021.