## THOMPSON RIVERS UNIVERSITY

Skin Cancer Detection Using Deep Learning

By

Saeid Moradi

## A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science in Data Science

## KAMLOOPS, BRITISH COLUMBIA

August, 2024

SUPERVISOR

Dr. Mateen Shaikh

#### ABSTRACT

Skin cancer is one of the most prevalent cancer types worldwide currently, underscoring the significance of early detection and precise diagnosis for effective treatment. This study employs the HAM10000 dataset, comprising 10015 skin lesion instances across seven categories of pigmented skin lesions. Preprocessing techniques are applied, including image resizing and normalization, and data augmentation is implemented to address dataset imbalances. The research primarily employs supervised machine learning models for skin cancer detection, utilizing Convolutional Neural Networks (CNNs). Specifically, VGG16, VGG19, ResNet50, MobileNet, MobileNetV2, and MobileNetV3 are examined for their performance on the dataset. Results indicate that ResNet50, with 92.31% accuracy and 91.98% F1-score, demonstrates higher performance, while MobileNetV3, with about 13 minutes of training time, outperforms in terms of computational efficiency.

Key Words: Skin Cancer Detection; CNNs; VGG16; ResNet50; MobileNet, MobileNetV2, MobileNetV3.

#### ACKNOWLEDGEMENTS

Work was conducted on Secwepemcúl'ecw, the unceded territory of the Secwépemc. The TRU Kamloops campus operates on the traditional lands of the Tk'emlúps te Secwépemc.

This work acknowledges the support from the NSERC Discovery grant RGPIN-2018-06787.

I want to express my heartfelt gratitude to my thesis supervisor, Dr. Mateen Shaikh, for his unwavering support, guidance, and invaluable insights throughout the research process. His expertise and encouragement have been instrumental in shaping the direction of this thesis.

Beyond academics, I extend my most profound appreciation to my wife, Zohreh Moradi. Her support, patience, and understanding have been the cornerstones of my journey. Her encouragement during challenging moments and celebration during triumphs have made this academic pursuit a shared endeavor.

Thank you to everyone who contributed to and supported this thesis. Your influence has left an indelible mark on my academic and personal growth.

## Contents

1	Introduction			1
	1.1	What	is skin cancer?	1
		1.1.1	Types of skin cancer	2
	1.2	Summ	ary of contribution	10
	1.3	Artific	ial intelligence and skin cancer detection	11
		1.3.1	Skin cancer detection	11
		1.3.2	Artificial intelligence methods in medical imaging	12
	1.4	Data I	Description	14
		1.4.1	Exploratory Data Analysis - EDA	15
2	Lite	erature	Review	19
	2.1	Machin	ne Learning Techniques	20
		2.1.1	Decision trees	20

3

	2.1.2	Support Vector Machines	21
	2.1.3	Artificial Neural Network	22
	2.1.4	Naïve Bayes	24
	2.1.5	K-Nearest Neighbors	25
	2.1.6	Machine Learning Techniques Summary	25
2.2	Deep 1	learning	26
	2.2.1	Deep learning models in skin cancer detection	38
	2.2.2	Recurrent Neural Networks	39
	2.2.3	Long Short-Term Memory	41
	2.2.4	Generative Adversarial Network	43
	2.2.5	Convolutional Neural Network	44
	2.2.6	Related Works	46
	2.2.7	CNN Architecture	47
	2.2.8	Transfer Learning	68
Met	thodol	ogy	70
3.1	Main s	stages of the methodology	70
	3.1.1	Pre-processing	71

v

	3.1.2	Data Augmentation		71
	3.1.3	Model Architecture	•	72
	3.1.4	Evaluation Metrics		73
4 Res	sults &	Discussion		77
4.1	Trans	fer Learning and Data Augmentation	•	77
	4.1.1	Parameter Tuning and Implementation Details	•	77
	4.1.2	CNN models without data augmentation		79
	4.1.3	CNN models with data augmentation	•	88
4.2	Discu	ssion		98
5 Coi	nclusio	n	1	102
A Ap	pendix	: Accepted Paper	1	20

# List of Figures

1.1	Melanoma skin lesions examples [1]	4
1.2	Basal Cell Carcinoma (BCC) skin lesion instance [2]	6
1.3	Squamous cell carcinoma (SCC) skin lesions examples [3]	7
1.4	Actinic keratosis (AK) skin lesions examples [3]	8
1.5	Dysplastic Nevi skin lesions examples [4]	9
1.6	Skin lesion images of HAM10000 Dataset.	16
1.7	Distribution of Lesion Types in HAM10000 dataset	17
1.8	Distribution of skin lesion types in the HAM10000 dataset	18
2.1	Artificial Neural Networks (ANNs) basic architecture [5]	23
2.2	Shallow ANNs vs Deep neural networks [6]	26
2.3	Notations of neural network. [6]	28
2.4	Most Common Activation Functions.	29

2.5	A neural network with one hidden layer.	31
2.6	Calculation in each node on neural network	31
2.7	Computational graph of the model	35
2.8	Multiclass classification output layer	39
2.9	Recurrent neural networks architecture [7]	40
2.10	Long short term (LSTM) architecture [8]	42
2.11	Generative Adversarial Network (GAN) architecture [9]	44
2.12	CNNs Architecture.	49
2.13	Filtering with the stride of 2	50
2.14	Padding in filtering [10]	52
2.15	Convolution on RGB images.	53
2.16	Convolution on RGB images with 2 filters	53
2.17	One layer of a convolutional network	54
2.18	Pooling layer.	56
2.19	Dropout layer. [10]	58
2.20	VGG16 architecture. [11]	59
2.21	Residual Block.	61
2.22	Residual Network	61

2.23	ResNet Architecture. [12]	62
2.24	Normal Convolution Vs. Depthwise Separable Convolution. [6].	64
2.25	MobileNet architecture. [6]	64
2.26	MobileNet Version 2 architecture. [6]	65
2.27	MobileNet Version 3 architecture. [13]	66
2.28	MobileNetV3 activation functions. [13]	67
2.29	Transfer learning methodology.	69
3.1	Our methodology process steps	71
3.2	Model architecture.	73
4.1	Distribution of skin lesions in training dataset before augmen-	
	tation	89

## List of Tables

2.1	Summary of Related Works	48
4.1	Pre-trained CNNs without Data Augmentation.	80
4.2	Fine-tuned CNNs without Data Augmentation.	81
4.3	True labels of the test dataset	82
4.4	Confusion matrix of MobileNetV1 on the test dataset	82
4.5	Confusion matrix of MobileNetV2 on the test dataset	83
4.6	Confusion matrix of MobileNetV3 on the test dataset	84
4.7	Confusion matrix of VGG16 on the test dataset	85
4.8	Confusion matrix of VGG19 on the test dataset	86
4.9	Confusion matrix of ResNet50 on the test dataset	87
4.10	Frozen Weight CNNs with Data Augmentation.	90
4.11	Fine-tuned CNNs with Data Augmentation.	91

4.12	Confusion matrix of MobileNetV1 on the test dataset	after	
	data augmentation.		93
4.13	Confusion matrix of MobileNetV2 on the test dataset	after	
	data augmentation.		94
4.14	Confusion matrix of MobileNetV3 on the test dataset	after	
	data augmentation.		95
4.15	Confusion matrix of VGG16 on the test dataset after	data	
	augmentation.		96
4.16	Confusion matrix of VGG19 on the test dataset after	data	
	augmentation.		97
4.17	Confusion matrix of ResNet50 on the test dataset after	data	
	augmentation.		98
4.18	Summary of Model Performance, Strengths, and Weakne	esses .	99

## Chapter 1

## Introduction

## 1.1 What is skin cancer?

Skin cancer stands out as one of the most prevalent forms of cancer globally currently [14]. Skin cancer is mainly categorized into two major groups: melanoma, which is the dangerous type, and non-melanoma, which is more common but generally less deadly [15]. Based on the World Cancer Research Fund International (WCRFI) report, estimating the incidence of skin cancer poses a distinctive challenge for several reasons. The existence of various sub-types of skin cancer complicates the compilation of data. For instance, non-melanoma skin cancer is frequently not monitored by cancer registries, and registrations for this type of cancer are often incomplete as many cases are effectively treated through surgical procedures or ablation. Consequently, the reported global incidence of skin cancer is likely lower than the actual occurrence due to these factors. WCRFI reported that melanoma is the 17th most common cancer worldwide; it is the 13th most common cancer in men and the 15th most common cancer in women. Among women aged 30 to 35, skin cancer is the second most prevalent cancer, following breast cancer, and among women aged 25 to 29, it stands as the most common cancer [16]. In addition, WCRFI reports that there were more than 150,000 new cases of melanoma skin cancer worldwide in 2020, with a total of 324,635 cases around the world. Australia, New Zealand, Denmark, The Netherlands, and Norway were the five countries with the highest melanoma skin cancer rates in 2020. This is likely due to a combination of factors, including high sun and UV exposure levels, especially in Australia and New Zealand, and the predominance of light-skinned populations in these countries, who are more susceptible to UV-induced skin damage. The mortality of melanoma skin cancer around the world in 2020 was 57,043 deaths, where New Zealand, Norway, Montenegro, Slovakia, and Slovenia had the highest number of deaths. There were 1,198,071 cases of non-melanoma type in 2020, and the five countries with the highest rates were Australia, New Zealand, the US, Canada, and Switzerland. The worldwide mortality rate for non-melanoma skin cancer was 63,731 in 2020, whereas Papua New Guinea, Namibia, Mozambique, Zimbabwe, and Angola had the highest mortality rates [17].

### **1.1.1** Types of skin cancer

Melanoma and nonmelanoma represent the principal categories of skin cancer. This section outlines the specific subtypes within the families of skin cancers.

#### Melanoma

Melanoma is a dangerous form of skin cancer originating from melanocytes, which produce skin pigment, melanin [18]. Melanoma can potentially impact any region of the human body, with a common occurrence on sun-exposed areas like the hands, face, neck, and lips [18]. UV radiation from the sun can penetrate the skin and damage the DNA within skin cells, particularly in melanocytes, the cells responsible for producing the pigment melanin. This damage often takes the form of DNA mutations, such as the formation of thymine dimers, which can lead to errors during DNA replication. If these mutations affect genes that regulate cell growth and division—such as tumor suppressor genes or oncogenes—they can disrupt normal cellular functions and lead to uncontrolled cell proliferation. Over time, these changes can accumulate, transforming normal melanocytes into malignant melanoma cells. [18]. Timely diagnosis is crucial for effectively treating melanoma; otherwise, it can metastasize to other parts of the body and ultimately lead to death [19]. Prolonged exposure to specific forms of light, such as ultraviolet rays from the sun or tanning devices, constitutes the primary factor responsible for the development of both melanoma and non-melanoma skin cancers [20]. Additionally, several factors have been linked to an elevated risk of skin cancer, including radiation exposure, genetic predisposition, and family history, as well as variations in skin pigmentation [20]. In 2024, an estimated 200,340 cases are projected in the U.S., with 8,290 deaths expected [18]. Figure 1.1 indicates melanoma skin cancer image.



Figure 1.1: Melanoma skin lesions examples [1].

#### Non-Melanoma

Non-Melanoma Skin Cancer, alternatively referred to as keratinocyte cancer, originates in the skin's keratinocyte cells, and it has two major subtypes: Basal cell carcinoma (BCC) and squamous cell carcinoma (SCC) [21].

#### Basal Cell Carcinoma (BCC)

Basal cell carcinoma (BCC), the most prevalent form of skin cancer, originates in basal cells responsible for renewing skin cells in the lower epidermis [2]. While typically confined to sun-exposed areas and rarely metastatic, BCC can lead to disfigurement or, in rare instances, life-threatening spread [2]. According to the American Cancer Society (ACS), around 80 percent of all skin cancers are basal cell cancers [22]. BCC manifests on the skin's surface, resembling sores, growths, bumps, scars, or red patches. Diagnosed through visual inspection and biopsy, BCC, if untreated, may invade adjacent areas and recur [22]. Its occurrence in sun-exposed regions, such as the face, head, neck, and arms, is linked to long-term sun or UV exposure [2]. Figure 1.2 shows some examples of BCC skin lesions.

#### Squamous cell carcinoma (SCC)

Squamous cell carcinoma (SCC) is a common form of skin cancer, accounting for approximately 20% of all non-melanoma skin cancers [3]. It is characterized by abnormal growth of squamous cells. Since the primary cause of SCC is UV radiation, it typically appears as scaly patches or raised growths on sun-exposed areas but can occur anywhere on the body. Early detection



Figure 1.2: Basal Cell Carcinoma (BCC) skin lesion instance [2].



Figure 1.3: Squamous cell carcinoma (SCC) skin lesions examples [3].

is crucial for successful treatment, as advanced SCCs can become dangerous by invading deeper layers of the skin, underlying tissues, or even spreading (metastasizing) to lymph nodes and other organs, which can lead to significant complications and be life-threatening [3]. Regular self-examination and annual dermatologist visits are recommended, particularly for individuals at higher risk—such as those with a history of excessive sun exposure, fair skin, or a family history of skin cancer. These practices and sun safety measures can significantly reduce the risk of developing skin cancer [23]. Figure 1.3 shows some examples of SCC skin lesions.



Figure 1.4: Actinic keratosis (AK) skin lesions examples [3].

#### Actinic keratosis (AK)

Actinic keratosis (AK), known as solar keratosis, is a skin condition triggered by prolonged exposure to ultraviolet radiation, typically from sunlight [21]. Actinic keratosis (AK) is a pre-malignant skin growth that can potentially progress into squamous cell carcinoma (SCC). AKs usually emerge on skin areas exposed to the elements, such as the head, neck, hands, and forearms [23]. Figure 1.4 shows some examples of AK skin lesions.

### Dysplastic Nevi

Atypical moles, also known as dysplastic nevi, share similarities with regular moles but also display specific characteristics akin to melanoma. They often have an irregular shape or color and are larger than typical moles. Atypical moles can develop on skin that is usually covered, such as the buttocks or scalp, as well as on skin exposed to the sun [23]. Figure 1.5 shows some examples of dysplastic nevi skin lesions.



Figure 1.5: Dysplastic Nevi skin lesions examples [4].

## **1.2** Summary of contribution

This research is motivated by two primary goals. First, to improve the efficiency and accuracy of skin cancer diagnosis by developing an artificial intelligence-based screening system using dermoscopic images of skin lesions. Such a system could aid clinical screening tests, reduce diagnostic errors, and enhance early detection, which is critical for successful treatment. Second, this study aims to address the urgent need for reliable automated skin cancer detection systems, particularly in regions with limited access to dermatology specialists. By evaluating the classification performance of six CNN models and analyzing their training behavior and time requirements, this research provides a comprehensive assessment of AI-based solutions for skin cancer diagnosis. Ultimately, this study seeks to bridge diagnostic gaps, enable timely treatment, improve patient outcomes, and potentially save lives. Most recent studies focus on optimizing model accuracy without addressing the computational complexity, making them less suitable for real-time or mobile applications. Additionally, many approaches do not adequately address class imbalance in datasets, which can lead to biased models that underperform on minority classes. This study addresses these gaps by evaluating a diverse set of pre-trained CNN models, focusing on accuracy and computational efficiency. Moreover, by fine-tuning these models and analyzing their performance across a balanced dataset, this research aims to develop a practical, scalable solution for skin cancer detection that can be deployed in resource-limited settings.

A portion of this thesis was peer reviewed and accepted for publication to appear in the proceedings of The International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS) 2024. The accepted, not the final published, version of the manuscript, is provided in the Appendix.

In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of Thompson Rivers University's products or services. Internal or personal use of this material is permitted.

## 1.3 Artificial intelligence and skin cancer detection

### **1.3.1** Skin cancer detection

Early detection and accurate diagnosis are critical factors in treating skin cancer. Typically, physicians rely on the biopsy method for skin cancer detection, which involves extracting a sample from a suspected skin lesion for laboratory-based confirmation of cancer [24]. However, this process is often painful, slow, and time-consuming. A biopsy is usually conducted to confirm the diagnosis of a suspected lesion or to remove a lesion for cosmetic or therapeutic reasons [24]. Dermatologists can correctly classify skin cancer with an accuracy of 75% to 84% when diagnosing melanoma [25, 26]. However, globally, there is a shortage of skilled dermatologists in public healthcare systems, exacerbating the challenges in dermatological diagnosis and treatment [27], and demonstrating the need for fast and accurate diagnostic techniques that clinicians can easily employ.

## 1.3.2 Artificial intelligence methods in medical imaging

Artificial intelligence (AI), a domain within computer science characterized by using machines and programs to emulate intelligent human behavior through various technological approaches, stands as a pivotal catalyst driving the fourth industrial revolution [28]. Within this domain, Machine learning (ML) emerges as a prominent technique, employing statistical models and algorithms to progressively learn from data, enabling the prediction of characteristics of new samples and the execution of desired tasks [29]. ML trains computers to emulate human cognitive processes, learning from past experiences and expanding upon them with minimal human intervention. Its profound impact spans various societal domains, including production lines, healthcare, education, transportation, and food industries [29]. Indeed, machine learning is actively reshaping everyday life, and industries such as housing, automotive, retail, etc. Central to the objective of machine learning is the endowment of computers with the capacity to collect and interpret data, thereby facilitating informed decision-making processes based on past and present outcomes [30]. ML enables computers to gain insights from data through various paradigms such as supervised, unsupervised, semi-supervised, or reinforcement learning [31]. Supervised learning involves pattern recognition from labeled datasets containing descriptive features and corresponding class labels. In contrast, unsupervised learning algorithms discern patterns from unlabeled datasets, often applied in anomaly detection tasks [32]. Deep Learning (DL), as a subcategory of ML comprising deep neural networks, shares similarities with ML yet operates on a deeper level of complexity. DL techniques can be supervised, unsupervised,

or semi-supervised, demonstrating widespread application in medical imaging for tasks such as image segmentation, classification, and object detection due to their superior performance [33]. In recent decades, deep learning has profoundly transformed the field of machine learning. The significant increase in processing power has facilitated remarkable progress in computer vision technologies, notably by developing deep learning models like Convolutional Neural Networks (CNNs) [34]. The urgency for early skin cancer detection has intensified, and deep learning has emerged as a powerful tool in this endeavor. Studies have demonstrated that early identification of skin cancer using deep learning improves the performance of human specialists, ultimately leading to a reduction in mortality rates [35]. By incorporating efficient formulations into deep learning techniques, exceptional and state-of-the-art processing and classification accuracy can be achieved [36]. Computer-based technology presents a promising avenue for diagnosing skin cancer symptoms, offering advantages in comfort, cost-effectiveness, and speed [36]. Typically, the process of skin cancer detection entails several stages, starting with the acquisition of images of skin lesions. These images are then subjected to preprocessing techniques to enhance quality and remove noise [37]. Subsequently, relevant features are extracted from the preprocessed images, which are crucial inputs for classification algorithms. Finally, these algorithms utilize the extracted features to categorize skin lesions into their classes [38]. This approach leverages the capabilities of computer-based technology in the diagnosis process, enabling efficient and accurate identification of potential skin cancer symptoms.

This research tries to develop a skin lesion diagnosis model using the HAM10000 dataset [39], including a wide array of dermatoscopic images. The research methodology involves exploring and analyzing the HAM10000

dataset, focusing on harnessing the inherent complexities within the dermatoscopic images. Applying deep learning techniques and algorithms aims to develop a model that can effectively recognize patterns and characteristics within skin lesion images. The study aims to contribute to the progress of dermatological diagnostics, particularly in the classification of skin lesions.

## 1.4 Data Description

Quality data plays a pivotal role in the performance of machine learning models. Therefore, a diverse and comprehensive collection of dermoscopic images is necessary to assess the effectiveness of computer-based systems for skin cancer diagnosis. The HAM10000 dataset, which consists of highresolution dermoscopic images, is used in this research. The dataset consists of 10,015 dermatoscopic images obtained from different populations and acquired through various modalities. The dataset was gathered from two sources: Cliff Rosendahl's skin cancer practice in Queensland, Australia, and the Dermatology Department of the Medical University of Vienna, Austria. It includes representative cases of all significant diagnostic categories for pigmented lesions such as actinic keratoses and intraepithelial carcinoma (AKIEC), basal cell carcinoma (BCC), benign keratosis-like lesions (BKL), dermatofibroma (DF), melanoma (MEL), melanocytic nevi (nv), and vascular lesions (vasc) [36]. The dataset is publicly available through the Kaggle [40]. The resulting dataset includes 327 images of AKIEC, 514 images of basal cell carcinomas, 1099 images of benign keratoses, 115 images of dermatofibromas, 6705 images of melanomas, 1113 images of melanocytic nevi, and 142 images of vascular skin lesions [39]. Figure 1.6 indicates images from the dataset for seven lesion types.

## 1.4.1 Exploratory Data Analysis - EDA

Exploratory data analysis (EDA) involves analyzing and summarizing datasets to understand their characteristics better before formal modeling. The main goal of EDA is to identify patterns, trends, and relationships in the data that can inform further analysis and modeling. The following shows the analysis of the HAM10000 dataset to gain some insights into the data structures and samples.

Figure 1.7 indicates information about the distribution of lesion types in the HAM10000 dataset. The lesion types bar chart indicates that melanocytic nevi are the most diagnosed condition among people in this dataset among the various types of skin diseases. On the other hand, dermatofibroma is a benign skin lesion less common than other lesion types in the dataset. It shows a kind of imbalance in the HAM10000 dataset.

The pie chart in Figure 1.8 illustrates the distribution of skin lesion types in the HAM10000 dataset. Approximately 67% of the dataset comprises nevi lesions, while dermatofibroma skin lesions constitute only 1.2%. Such an imbalanced distribution may pose challenges in training models and potentially impact their generalization capabilities. We use data augmentation to solve the problem of imbalance in the dataset.



Figure 1.6: Skin lesion images of HAM10000 Dataset.



Figure 1.7: Distribution of Lesion Types in HAM10000 dataset.

17



Figure 1.8: Distribution of skin lesion types in the HAM10000 dataset.

## Chapter 2

## Literature Review

The rising incidence of skin cancer necessitates timely diagnosis and continuous monitoring, placing a strain on specialist medical services. This burden could be alleviated by promoting patient self-surveillance techniques and integrating decision support systems for less experienced physicians. Unlike human diagnosis, machine diagnosis is objective and remains unaffected by external factors, offering consistent results. If properly applied, leveraging AI for skin cancer detection and progression monitoring can potentially reduce the need for biopsies and detect cancers early before they progress.Additionally, training interventions can empower patients and their caregivers to conduct self-skin examinations, which can facilitate teledermoscopy — a process where images of skin lesions are captured using a smartphone or digital camera and then transmitted to a dermatologist for remote evaluation. This approach can reduce the frequency of in-person medical consultations while effectively monitoring skin conditions [28].

Finding an automatic classification system for skin cancer is challenging

due to the complexity and diversity of skin cancer images. First, it's important to note that different skin lesions often share significant similarities among various classes, increasing the risk of misdiagnosis [41]. Additionally, even within the same class, several skin lesions can vary in color, features, structure, size, and location [42].

## 2.1 Machine Learning Techniques

While traditional machine learning approaches perform well in specific skin cancer classification tasks, they often prove ineffective in handling complicated diagnostic problems. Typically, conventional machine learning methods for skin cancer diagnosis require extracting features from skin disease images and classifying these extracted features [43]. Commonly used features include the asymmetry, borders, color, and diameter of moles (known as ABCD features) [44], as well as 2D wavelet transformations [25] and the gray-level co-occurrence matrix (GLCM) features [45]. Various classification techniques like Support Vector Machines (SVM) [43], XGBoost [46], and decision trees [47] are frequently employed. Because of the limited number of selected features, machine learning algorithms may find it challenging to classify only a subset of skin cancer diseases. They may struggle to generalize to a broader spectrum of disease types [48].

#### 2.1.1 Decision trees

Decision trees, another machine learning technique that is a supervised learning method primarily employed for classification problems, offer an intuitive algorithm for assessing the long-term risk of non-melanoma skin cancer postliver transplant, utilizing variables linked to the peri-transplant period [49]. In a different context, [50] utilizes decision trees as a visual representation mode, dividing branches to depict various outcomes during a clinical procedure. This application involves assessing the cost-effectiveness of sentinel lymph node biopsy, a standard technique in melanoma and breast cancer treatment, specifically in the context of head and neck cutaneous squamous cell carcinoma, a subset of skin cancer.

Moreover, decision trees can function as an intermediate layer, as demonstrated in [51], which showcases their effectiveness in region extraction and skin cancer classification using deep convolutional neural networks. In this architecture, decision trees, support vector machines, and k-nearest neighbors are crucial in classifying most features.

Notably, the decision tree model in [49] reports a specificity of 42% and a sensitivity of 91%, while models akin to those in [50] exhibit a sensitivity of 77% with a reported 100% specificity. It's essential to recognize that decision tree model predictions are significantly influenced by the quality of the datasets they are trained on [50].

## 2.1.2 Support Vector Machines

Support Vector Machines (SVMs) are powerful supervised learning models widely used for classifying, predicting, and analyzing data. Within the domain of skin lesion classification, SVMs have proven effective. In [52], using ABCD features facilitates the extraction of critical attributes, including shape, color, and size, from clinical images. These features are then employed to classify skin lesions into distinct categories, such as melanoma, seborrheic keratosis, and lupus erythematosus, demonstrating the efficacy of the ABCD feature set when coupled with SVMs. In [53], preprocessing steps such as grayscale conversion, noise removal, and binarization are applied to the input image to enhance accuracy. Similarly, a bag-of-features approach incorporating spatial information is employed for skin cancer detection. SVMs are trained using histograms of oriented gradients, resulting in promising outcomes compared to existing algorithms [54]. A suggested methodology consisting of several phases, including pre-processing, segmentation, feature extraction, and classification, was proposed in [55]. Experimentation was conducted on a dataset comprising 1800 images, resulting in an accuracy 83% for a six-class classification task. This accuracy was attained using a support vector machine (SVM) with a quadratic kernel.

### 2.1.3 Artificial Neural Network

An artificial neural network (ANN) is a nonlinear and statistical prediction technique that draws its structural inspiration from the biological framework of the human brain. As shown in Figure 2.1 An ANN comprises three layers of neurons; the initial layer is called the input layer, where these input neurons transmit data to the second layer, often referred to as the intermediate or hidden layer. In a typical ANN, multiple hidden layers can exist. The intermediate neurons convey data to the third layer, consisting of output neurons. At each layer, computations are learned through backpropagation, which is employed to grasp the intricate associations and relationships between the input and output layers.



Figure 2.1: Artificial Neural Networks (ANNs) basic architecture [5].

Xie et al. [56] introduced a skin lesion classification system designed to categorize lesions into two primary classes: benign and malignant. The proposed system's classification results were benchmarked against various classifiers, including SVM, KNN, random forest, Adaboost, and others. The proposed model demonstrated an accuracy rate of 91.11%, outperforming the other classifiers by at least 7.5% in sensitivity.

Choudhari and Biday [45] introduced another skin cancer diagnostic system based on artificial neural networks (ANN). In their approach, images were segmented using a maximum entropy thresholding measure, and unique features of skin lesions were extracted using a gray-level co-occurrence matrix (GLCM). Subsequently, a feed-forward ANN was employed to classify the input images into either a malignant or benign stage of skin cancer, achieving an accuracy level of 86.66%.

### 2.1.4 Naïve Bayes

Naïve Bayes classifiers are another group of machine learning techniques that operate on Bayes' theorem and are probabilistic classifiers widely employed in skin cancer research to accurately classify clinical and dermatological images [57]. These models have demonstrated 70.15% and 73.33% for accuracy and specificity, respectively [57]. Expanding their utility, Naïve Bayes classifiers offer a method for detecting and segmenting skin diseases, as documented in [58]. The iterative process of obtaining posterior probability distributions for each output class enables efficient utilization of computational resources, minimizing the need for multiple training sessions. Results of this study indicate the diagnostic accuracy reached 72.7%. The Bayesian approach is valuable in various applications, including probabilistically predicting the nature of data points with high accuracy, as demonstrated in [59]. An iterative process obtains a posterior probability distribution for each output class, reducing the computational resources required and eliminating the need for multiple training sessions. This Bayesian sequential framework extends its utility to aiding models designed to detect melanoma invasion into human skin. In this context, three model parameters are estimated: the melanoma cell proliferation rate, the melanoma cell diffusivity, and a constant determining the degradation rate of melanoma cells in skin tissue. The algorithm learns from data sequentially, including a spatially uniform cell assay, a 2D circular barrier assay, and a 3D invasion assay. The versatility of this Bayesian framework allows for its extraction and application in various biological contexts beyond skin cancer detection.

### 2.1.5 K-Nearest Neighbors

The k-nearest neighbors algorithm (KNN) is a supervised classification method that leverages distance and proximity metrics to classify data points. KNNs have been utilized and assessed in skin cancer detection, with evaluations involving the generation of a confusion matrix to depict the model's accuracy [60]. The research shows an accuracy of 66.8% in terms of performance metrics. Furthermore, for positive predictions, the precision and recall stand at 71% and 46%, respectively. In [61], KNN is extended using the Radius Nearest Neighbors classifier to classify breast cancer, overcoming limitations posed by extreme values of k. Normalizing the radius value of each point helps effectively recognize outliers, mitigating sensitivity to outliers and underfitting issues.

Despite its effectiveness in skin cancer diagnosis, KNN classifiers necessitate continuous training and encounter challenges related to limited training data availability [60, 61].

### 2.1.6 Machine Learning Techniques Summary

Upon analyzing the diverse implementations of machine learning models in skin cancer diagnosis, it becomes evident that Support Vector Machines (SVMs) show better precise and accurate models [20]. However, their requirement for meticulous pre-processing of input data presents a significant challenge. For user flexibility, K-means clustering and K-nearest neighbors offer viable alternatives without substantial compromises in accuracy and performance. Nonetheless, K-nearest neighbors necessitate continuous train-


Figure 2.2: Shallow ANNs vs Deep neural networks [6].

ing as additional data points are introduced, which can be burdensome due to the unpredictable volume of input data [20]. In contrast, Naïve Bayes models exhibit the lowest accuracy among the studied machine learning techniques, likely because other methods, such as decision trees and random forests, build upon the foundational principles of the Naïve Bayes theorem [20].

# 2.2 Deep learning

Deep neural networks are ANNs with a higher number of hidden layers. Figure 2.2 represents the shallow neural networks with less than two hidden layers and deep neural networks with five hidden layers. Following this section, we go through the mathematics behind neural networks and deep learning and then describe the families of deep learning models commonly used in skin cancer detection.

Standard notations for neural networks and deep learning [6]

- superscript (i) will denote the  $i^{\text{th}}$  training example.
- m: number of examples in the dataset.  $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(m)}, y^{(m)})\}$
- $n_x$ : input size
- $n_y$ : output size (or number of classes)
- $n_h^{[l]}$ : number of hidden units of the  $l^{th}$  layer.
- L: number of layers in the network.

• 
$$X \in \mathbb{R}^{n_x \times m}$$
 is the input matrix.  $X = \begin{bmatrix} \vdots & \vdots & \vdots \\ x^{(1)} & x^{(2)} & \dots & x^{(m)} \\ \vdots & \vdots & & \vdots \end{bmatrix}$ 

•  $x(i) \in \mathbb{R}^{n_x}$  is the  $i^{th}$  example represented as a column vector.

• 
$$Y \in R^{n_y \times m}$$
 is the label matrix.  $Y = \begin{bmatrix} \vdots & \vdots & \vdots \\ y^{(1)} & y^{(2)} & \dots & y^{(m)} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{bmatrix}$ 

- $y(i) \in \mathbb{R}^{n_y}$  is the output label for the  $i^{th}$  example.
- W<sup>[l]</sup> ∈ R<sup>numberofunitsinnextlayer×numberofunitsinthepreviouslayer</sup> is the weight matrix, and subscription [l] indicates the layer.

Г

- $b^{[l]} \in \mathbb{R}^{numberofunitsinnext layer}$  is the bias vector in the  $l^{th}$  layer.
- ŷ ∈ R<sup>n<sub>y</sub></sup> is the predicted output vector. It can also be denoted a<sup>[L]</sup> where L is the number of layers in the network.
- $a = g^{[l]}(W_x x^{(i)} + b_1) = g^{[l]}(z_1)$  where  $g^{[l]}$  is the  $l^{th}$  layer activation function.
- General Activation Formula:  $a_j^{[l]} = g^{[l]}(\Sigma_k W_j k^{[l]} a_k^{[l-1]} + b_j^{[l]}) = g^{[l]}(z_j^{[l]})$



Figure 2.3: Notations of neural network. [6].

J(x, W, b, y) or J(ŷ, y) denote the cost function. Examples of cost function: J<sub>CE</sub>(ŷ, y) = −Σ<sup>m</sup><sub>i=0</sub>y<sup>(i)</sup>logŷ<sup>(i)</sup>
 J(ŷ, y) = −<sup>1</sup>/<sub>m</sub>Σ[y<sup>(i)</sup>logŷ<sup>(i)</sup> + (1 − y<sup>(i)</sup>)log(1 − ŷ<sup>(i)</sup>)]

Figure 2.3 indicates notations for a neural network with two hidden layers. In this representation, nodes represent inputs, activations, or outputs, and edges represent weights or biases.

#### Activation Function

Activation functions introduce non-linearity into the CNN architecture, enabling the networks to learn complex relationships in the data. Common activation functions include ReLU (Rectified Linear Unit), Leaky ReLu, sigmoid, and tanh. ReLU is widely used due to its simplicity and effectiveness



Figure 2.4: Most Common Activation Functions.

in mitigating the vanishing gradient problem. Figure 2.4 shows the most common activation functions in CNNs. The top left activation function is the sigmoid function, the top right function is tanh, the button left function is the Rectified unit function (ReLU), and the button right function is Leaky ReLu. The following indicates the formula for these functions.

$$a = g(z) = sigmoid(z) = \sigma(z) = \frac{1}{1+e^{-z}}$$
$$a = g(z) = tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$$
$$a = g(z) = ReLU(z) = max(0,z)$$
$$a = g(z) = LeakyReLU(z) = max(0.01z,z)$$

The derivatives of activation functions play a crucial role in neural network optimization. Thus, we present the derivatives corresponding to each activation function in the subsequent discussion.

$$a = g(z) = \sigma(z) = \frac{1}{1+e^{-z}} \to g'(z) = a(1-a)$$

$$a = g(z) = tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \rightarrow g'(z) = 1 - a^{2}$$

$$g(z) = ReLU(z) = max(0, z)$$

$$\rightarrow g'(z) = \begin{cases} 0, & \text{if } z < 0\\ 1, & \text{if } z > 0\\ undefined, & \text{otherwise} \end{cases}$$

$$g(z) = LeakyReLU(z) = max(0.01z, z)$$

$$\rightarrow g'(z) = \begin{cases} 0.01, & \text{if } z < 0\\ 1, & \text{if } z > 0\\ undefined, & \text{otherwise} \end{cases}$$

#### Calculation for shallow and deep neural networks

Let's start with doing calculations for a shallow neural network with one hidden layer for binary classification and then expand it to deep networks. Figure 2.5 indicates the architecture of the network. Figure 2.6 represents each node in the network, including two parts; z is the multiplication of weights and the input of the node in summation with bias, and a is the result of z through the activation function for that node.

In a neural network, we have two processes: forward propagation from input to output (left to right), where we propagate the input in different layers of the network to find the  $\hat{y}$ , and backward propagation from output to input (right to left) for updating the parameters w and b in a way minimizing the error. First, do the forward propagation calculation for one input with three features. For the hidden layer, we have the following calculations:



Figure 2.5: A neural network with one hidden layer.



Figure 2.6: Calculation in each node on neural network.

$$Z_1^{[1]} = w_1^{[1]T}x + b_1^{[1]}, a_1^{[1]} = \sigma(z_1^{[1]})$$
$$Z_2^{[1]} = w_2^{[1]T}x + b_2^{[1]}, a_2^{[1]} = \sigma(z_2^{[1]})$$
$$Z_3^{[1]} = w_3^{[1]T}x + b_3^{[1]}, a_3^{[1]} = \sigma(z_3^{[1]})$$
$$Z_4^{[1]} = w_4^{[1]T}x + b_4^{[1]}, a_4^{[1]} = \sigma(z_4^{[1]})$$

We can rewrite the above calculation in matrix form as below: The weight matrix  $W^{[1]}$  includes the weights in layer one for all nodes.

$$W^{[1]} = \begin{bmatrix} \dots & w_1^{[1]T} & \dots \\ \dots & w_2^{[1]T} & \dots \\ \dots & w_3^{[1]T} & \dots \\ \dots & w_4^{[1]T} & \dots \end{bmatrix}$$

The bias vector  $b^{[1]}$  includes all biases in the first layer.

$$b^{[1]} = egin{bmatrix} b_1^{[1]} \ b_2^{[1]} \ b_3^{[1]} \ b_4^{[1]} \end{bmatrix}$$

The multiplication and summation calculation vector as below:

$$z^{[1]} = \begin{bmatrix} z_1^{[1]} \\ z_2^{[1]} \\ z_3^{[1]} \\ z_4^{[1]} \end{bmatrix}$$

The same for a:

$$a^{[1]} = \begin{bmatrix} a_1^{[1]} \\ a_2^{[1]} \\ a_3^{[1]} \\ a_4^{[1]} \end{bmatrix}$$
$$a^{[1]} = \sigma(z^{[1]})$$

We can write the same calculation for the output layer, a node for binary classification. Ensuring the right dimension for the matrices and vectors in the calculation is pivotal.

The whole calculation for both layers in matrix form is like this:

$$\begin{aligned} z_{(4,1)}^{[1]} &= W_{(4,3)}^{[1]} x_{(3,1)} + b_{(4,1)}^{[1]} \\ a_{(4,1)}^{[1]} &= \sigma(z^{[1]}) \\ z_{(1,1)}^{[2]} &= W_{(1,4)}^{[2]} a_{(4,1)}^{[1]} + b_{(1,1)}^{[2]} \\ a_{(1,1)}^{[2]} &= \sigma(z^{[2]}) \end{aligned}$$

Now we can expand the calculation for m inputs with a "for loop" like the following:

for 
$$i = 1$$
 to  $m$ :  
 $z^{[1](i)} = W^{[1]}x^{(i)} + b^{[1]}$   
 $a^{[1](i)} = \sigma(z^{[1](i)})$   
 $z^{[2](i)} = W^{[2]}a^{[1](i)} + b^{[2]}$   
 $a^{[2](i)} = \sigma(z^{[2](i)})$ 

We also can use vectorization instead of using "for loop" in programming, which means writing all the calculations in matrix form and using dot product instead of using loops in programming to speed up the process.

$$Z^{[1]} = W^{[1]}X + b^{[1]}$$
$$A^{[1]} = \sigma(Z^{[1]})$$
$$Z^{[2]} = W^{[2]}A^{[1]} + b^{[2]}$$
$$A^{[2]} = \sigma(Z^{[2]})$$

Where X is the input matrix as below:

$$X = \begin{bmatrix} \vdots & \vdots & \vdots \\ x^{(1)} & x^{(2)} & \dots & x^{(m)} \\ \vdots & \vdots & & \vdots \end{bmatrix}$$
$$Z^{[1]} = \begin{bmatrix} \vdots & \vdots & & \vdots \\ z^{[1](1)} & z^{[1](2)} & \dots & z^{[1](m)} \\ \vdots & \vdots & & \vdots \end{bmatrix}$$
$$A^{[1]} = \begin{bmatrix} \vdots & \vdots & & \vdots \\ a^{[1](1)} & a^{[1](2)} & \dots & a^{[1](m)} \\ \vdots & \vdots & & \vdots \end{bmatrix}$$

These matrices indicate that we go through the examples or inputs horizontally and vertically through the units (nodes) in hidden layers. These are the calculations for the forward propagation.

Let's start the calculation for backward propagation from output to input. It's an optimization problem where we try to update the network pa-



Figure 2.7: Computational graph of the model.

rameters w and b to minimize the error so that the predicted value of  $\hat{y}$  is closer to the actual target variable y. Gradient descent is a popular method for optimizing the cost function in neural networks to update the model parameters. This method is iterative and tries to minimize the error in each iteration by updating the parameters. In our neural network model, we have four sets of parameters including  $W_{(n^{[1]},n^{[0]})}^{[1]}$ ,  $b_{(n^{[1]},1)}^{[2]}$ ,  $W_{(n^{[2]},n^{[1]})}^{[2]}$ , and  $b_{(n^{[2]},1)}^{[2]}$ , where  $n_x = n^{[0]}$ , and  $n^{[2]=1}$ . In gradient descent, we update the parameters by finding the derivative of the cost function concerning that parameter. We start with a computational graph to make the calculation easier. Figure 2.7 indicates the graph, where the black arrows show the forward propagation path and the red arrows show the backward propagation path from output to input.

We define the derivatives of the cost function to the parameters in the following:

$$dw^{[1]} = \frac{dJ}{dw^{[1]}}, \ db^{[1]} = \frac{dJ}{db^{[1]}},$$
$$dw^{[2]} = \frac{dJ}{dw^{[2]}}, \ db^{[2]} = \frac{dJ}{db^{[2]}},$$

and then updates the parameters in an iterative process like below:

Repeat {

compute predictions  $(\hat{y}, i = 1 \text{ to } m)$ compute  $dw^{[1]}, db^{[1]}, dw^{[2]}, db^{[2]}$   $W^{[1]} := W^{[1]} - \alpha dw^{[1]}, b^{[1]} := b^{[1]} - \alpha db^{[1]}$  $W^{[2]} := W^{[2]} - \alpha dw^{[2]}, b^{[2]} := b^{[2]} - \alpha db^{[2]}$ 

Where  $\alpha$  is the learning rate, below are the derivatives based on the computational graph in Figure 2.7. In this example, solve a binary classification, and the loss function is equal to  $L(a^{[2]}, y) = -[ylog(a^{[2]}) + (1-y)log(1-a^{[2]})].$ 

$$\begin{aligned} da^{[2]} &= \frac{dL(a^{[2]}, y)}{da^{[2]}} = -\frac{y}{a^{[2]}} + \frac{1-y}{1-a^{[2]}} \\ dz^{[2]} &= \frac{dL(a, y)}{dz^{[2]}} = \frac{dL(a, y)}{da^{[2]}} \frac{da^{[2]}}{dz^{[2]}} = da^{[2]} \frac{d\sigma(z^{[2]})}{dz^{[2]}} = \left[ -\frac{y}{a^{[2]}} + \frac{1-y}{1-a^{[2]}} \right] \left[ a^{[2]}(1-a^{[2]}) \right] = \\ a^{[2]} - y \\ dW^{[2]} &= \frac{dL(a^{[2]}, y)}{dw^{[2]}} = \frac{dL(a^{[2]}, y)}{da^{[2]}} \frac{da^{[2]}}{dz^{[2]}} \frac{dz^{[2]}}{dw^{[2]}} = dz^{[2]} a^{[1]T} = (a^{[2]} - y)a^{[1]T} \\ db^{[2]} &= \frac{dL(a^{[2]}, y)}{db^{[2]}} = \frac{dL(a^{[2]}, y)}{da^{[2]}} \frac{da^{[2]}}{dz^{[2]}} \frac{dz^{[2]}}{dz^{[2]}} = dz^{[2]} \\ dz^{[1]} &= dz^{[2]} \frac{dz^{[2]}}{da^{[1]}} \frac{da^{[1]}}{dz^{[1]}} = W^{[2]T} dz^{[2]} * g^{[1]'}(z^{[1]}) \\ dW^{[1]} &= dz^{[1]} x^{T} \\ db^{[1]} &= dz^{[1]} \end{aligned}$$

Where g(z) is the activation function layer one, and \* denotes the elementwise multiplication.

The vectorization for the above calculation can be written below:

$$dZ^{[2]} = A^{[2]} - Y$$

$$dW^{[2]} = \frac{1}{m} dZ^{[2]} A^{[1]T}$$

$$db^{[2]} = \frac{1}{m} [\Sigma_{j=1}^{m} (dZ_{i,j}^{[2]})]_{i=1}^{n_{h}^{[2]}}$$

$$dZ^{[1]} = W^{[2]T} dZ^{[2]} * g^{[1]'} (Z^{[1]})$$

$$dW^{[1]} = \frac{1}{m} dZ^{[1]} X^{T}$$

$$db^{[1]} = \frac{1}{m} [\Sigma_{j=1}^{m} (dZ_{i,j}^{[1]})]_{i=1}^{n_{h}^{[1]}}$$

After calculation for a shallow neural network, we can expand the calculation for deep neural networks in the following:

# Forward Propagation:

$$Z^{[1]} = W^{[1]}X + b^{[1]}$$

$$A^{[1]} = g^{[1]}(Z^{[1]})$$

$$Z^{[2]} = W^{[2]}A^{[1]} + b^{[2]}$$

$$A^{[2]} = g^{[2]}(Z^{[2]})$$

$$\vdots$$

$$A^{[L]} = g^{[L]}(Z^{[L]})$$

# Backward Propagation:

$$dZ^{[L]} = A^{[L]} - Y$$
$$dW^{[L]} = \frac{1}{m} dZ^{[L]} A^{[L-1]^T}$$
$$db^{[L]} = \frac{1}{m} [\sum_{j=1}^m (dZ_{i,j}^{[L]})]_{i=1}^{n_h^{[L]}}$$

$$dZ^{[L-1]} = W^{[L]^T} dZ^{[L]} * g'^{[L-1]} (Z^{[L-1]})$$
  

$$\vdots$$
  

$$dZ^{[1]} = W^{[2]T} dZ^{[2]} * g'^{[1]} (Z^{[1]})$$
  

$$dW^{[1]} = \frac{1}{m} dZ^{[1]} X^T$$
  

$$db^{[1]} = \frac{1}{m} [\sum_{j=1}^m (dZ^{[1]}_{i,j})]^{n^{[1]}_h}_{i=1}$$

These are the calculations from shallow to deep neural networks in a binary classification problem. In multiclass classification, all the processes are the same, but the output layer activation function is softmax. The following indicates the formula for the Softmax activation function.

$$z^{[L]} = W^{[L]} a^{[L-1]} + b^{[L]}$$
$$k = e^{z^{[L]}}$$
$$\hat{y} = a^{[L]} = g^{[L]} (z^{[L]}) = \frac{e^{z^{[L]}}}{\sum_{j=1}^{\# of classes} k_j}$$

This activation function takes a vector as input and produces a vector as output. Figure 2.8 illustrates a multiclass classification scenario with the softmax activation function utilized at the output layer. The following are the commonly used deep learning models in skin cancer detection.

# 2.2.1 Deep learning models in skin cancer detection

The discipline of deep learning within artificial intelligence is rapidly expanding, offering numerous potential applications. Deep learning is one of the most potent and extensively employed machine learning techniques based on



Figure 2.8: Multiclass classification output layer.

artificial neural networks, particularly for recognizing and categorizing images [62]. In recent years, deep learning algorithms have gained extensive usage for skin cancer classification. In contrast to traditional machine learning techniques, deep learning algorithms can accurately analyze data from large-scale datasets, enabling them to extract relevant features efficiently [63]. Deep learning techniques find applications in various domains, including speech recognition [64], computer vision and pattern recognition [65], and bioinformatics [66]. In recent years, diverse deep learning approaches, including Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Generative Adversarial Network (GAN), and Convolutional Neural Networks (CNN) have been employed for computer-based skin cancer detection.

# 2.2.2 Recurrent Neural Networks

A recurrent neural network (RNN) is a subset of artificial neural networks and has found application in melanoma skin cancer detection [67]. Figure 2.9 shows the architecture of RNN models. In [68], deep features are extracted from clinical images in a feature extraction process using the hamming dis-



Figure 2.9: Recurrent neural networks architecture [7].

tance approach and fed into a dual bidirectional long short-term memory (LSTM) network for feature learning and a SoftMax activation function for image classification. Similarly, ensemble models are employed for automating mammogram breast cancer detection, where features extracted through the grey-level co-occurrence matrix and grey-level run-length matrix are inputted into the RNN layer. The segmented tumor binary image is provided as input to the CNN layer, leading to improved diagnostic accuracy. Moreover, RNNs have been instrumental in segmenting various dermoscopic images [69]. The recurrent model's ability to train deeper and larger models enhances performance, ensuring better feature representation.

The modified RNNs proposed in [67] exhibit an average accuracy of around 90%, with an F1-score of 0.865. Similarly, RNNs in [70] achieve an accuracy of 98% and an F1-score of 0.745. The model in [69] reports a testing accuracy of 87.09% and an average F1-score of 0.86.

### 2.2.3 Long Short-Term Memory

Long Short-Term Memory (LSTM) is an artificial neural network architecture with feedback connections representing a specialized form of recurrent neural network architecture engineered to address the vanishing gradient challenge encountered in RNNs. Their capability to learn intricate temporal dependencies in sequential data makes them highly effective across various tasks, including time series prediction, natural language processing, and speech recognition. Figure 2.10 indicates the architecture of LSTM models.

Memory cells are central to the LSTM model's structure, which sustain a cell state capable of retaining information over extended durations. These memory cells incorporate a range of gates, including input, forget, and output gates, which manage the flow of information within the cell.

This model efficiently maintains stateful information, leading to accurate predictions and fast recognition of target regions while requiring fewer computations than previous algorithms. Including LSTM improves the prediction accuracy due to its ability to retain information from earlier timestamps.

LSTMs can predict cancer and tumors in irregular medical data, leveraging their superior performance in screening time-series data [71]. Skin disease classification models utilize deep learning approaches like LSTM, often enhanced with hybrid optimization algorithms such as the Hybrid Squirrel Butterfly Search Optimization algorithm (HSBSO) [72]. This modified LSTM, incorporating HSBSO and optimized parameters, maximizes classification accuracy and overall efficiency, achieving an average sensitivity of 53% and specificity of 80%.



Figure 2.10: Long short term (LSTM) architecture [8].

# 2.2.4 Generative Adversarial Network

A Generative Adversarial Network (GAN) is a powerful class of deep neural networks (DNN) inspired by zero-sum game theory [73]. GANs consist of two neural networks, a generator and a discriminator, which compete to analyze and capture the variance in a given dataset. The generator module creates fake data samples based on the data distribution, aiming to deceive the discriminator, while the discriminator distinguishes between real and fake data samples [74]. Through repeated iterations during training, both networks improve their performance as they compete against each other. GANs excel at generating fake samples resembling real ones, addressing the problem of insufficient training examples in deep learning. Figure 2.11 shows the architecture of generative adversarial networks.

Rashid et al. [64] proposed a GAN-based classification system, augmenting a training set with realistic-looking skin lesion images generated via GAN. A deconvolutional network was the generator, while the discriminator utilized a CNN classifier. The proposed system achieved an accuracy of 86.1% for skin lesion classification.

To address limitations in deep learning methods, such as the need for large, unbalanced datasets, [65] proposed a system combining data purification with GAN-based data augmentation. Decoupled deep convolutional GANs were employed for data generation, resulting in improved performance compared to the baseline ResNet-50 model.

These studies demonstrate the effectiveness of GANs in enhancing the performance of skin cancer diagnostic systems by addressing challenges re-



Figure 2.11: Generative Adversarial Network (GAN) architecture [9].

lated to dataset size and imbalance.

# 2.2.5 Convolutional Neural Network

A convolutional neural network (CNN) is a crucial subtype of deep neural networks extensively used in computer vision. CNNs are particularly adept at image classification, grouping, and recognition tasks. In CNNs, the convolution operation is a fundamental process that helps extract features from the input data, such as images. Equation 2.1 represents the mathematical expression for a 2D convolution.

$$S(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} K(m,n) \cdot I(i+m,j+n)$$
(2.1)

S(i, j) is the output feature map, I(i, j) is the input image, and K(m, n) is the convolutional kernel (filter) of size  $M \times N$ . The convolution operation slides the kernel K over the input image I. At each position, it computes the sum of element-wise products between the kernel and the corresponding

patch of the input image. The result is stored in the output feature map S.

In [75], Deep CNNs have been utilized to classify skin cancer into four categories: basal cell carcinoma, squamous cell carcinoma, actinic keratosis, and melanoma. The authors assess the performance using evaluation parameters such as accuracy, sensitivity, and specificity. Recent research has explored integrating patient data with CNNs to enhance diagnostic accuracy in dermatology [76]. The patient data typically included information such as sex, age, and lesion location, and one-hot encoding was used to incorporate this data. The decision to fuse image features with patient data was contingent on the complexity of each classification task. These studies highlight the potential advantages and benefits of incorporating patient data into deep CNN algorithms in dermatology. In [77], the pre-trained Inception v3 model has been fine-tuned on two different resolution scales of input lesion images: a coarse scale and a finer scale. The coarse scale captured the lesions' shape characteristics and overall contextual information. In contrast, the finer scale focused on gathering detailed texture information of the lesion, facilitating the differentiation between various skin lesions. In [78], a deep convolutional neural network (CNN) architecture was introduced to classify 12 distinct types of skin lesions. Initially, it was trained using 3797 lesion images; subsequently, data augmentation was applied, expanding the dataset 29 times through variations in lighting conditions and scale transformations. The proposed technique achieved an impressive AUC (Area Under the Curve) value of 0.99 for the classification of hemangioma lesions, pyogenic granuloma (PG) lesions, and intraepithelial carcinoma (IC) skin lesions.

### 2.2.6 Related Works

Previous studies have demonstrated the effectiveness of Convolutional Neural Networks (CNNs) in skin cancer classification. For instance, a study utilizing the HAM10000 dataset employed MobileNet for skin lesion detection, achieving an accuracy of 83% [79]. Another study introduced a Fully Convolutional Residual Network (FCRN) with 16 residual blocks for melanoma detection, achieving an accuracy of 85.5% with segmentation and 82.8% without segmentation [80]. Huang et al. developed two deep learning models using DenseNet and EfficientNet, achieving 89.5% accuracy in binary classification on the KCGMH dataset and 85.8% on the HAM10000 dataset [81]. Furthermore, using Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) for image enhancement, coupled with a modified ResNet-50 model, has improved classification metrics such as accuracy, precision, recall, and F1-score [82].

Another study focused on accurately classifying skin lesions into seven categories using the HAM10000 dataset by leveraging 13 deep transfer learning models. The research emphasizes the importance of early detection in reducing mortality rates. It highlights the potential of AI-based systems to enhance diagnostic accuracy, particularly in regions with limited access to dermatological care [83]. Most current state-of-the-art approaches rely on either hybrid models [[84], [85]] or ensembles of deep learning classifiers [[86], [87], [88]], which, despite their high accuracy, are often too resourceintensive for mobile applications. Developing a practical mobile application requires identifying a deep learning model that balances state-of-the-art performance with lightweight architecture. Therefore, this paper evaluates the performance of six different CNN models and analyzes their training time requirements.

Despite these advancements, several limitations remain. Many studies focus primarily on optimizing model accuracy without addressing computational complexity, which makes these models less suitable for real-time or mobile applications. Additionally, class imbalance in datasets is often not adequately addressed, leading to biased models that underperform on minority classes. This study addresses these gaps by evaluating a diverse set of pre-trained CNN models, focusing on accuracy and computational efficiency. Moreover, by fine-tuning these models and analyzing their performance across a balanced dataset, this research aims to develop a practical, scalable solution for skin cancer detection that can be deployed in resourcelimited settings. Table 2.1 summarizes related works, their limitations, and our contribution to this research.

# 2.2.7 CNN Architecture

In this research, we use the CNN family of deep neural networks to detect the skin lesions on our dataset. Therefore, we go through the architecture details and different layers of CNNs. The hidden layers of a CNN typically include convolution layers, nonlinear pooling layers, and fully connected layers [89]. Figure 2.12 shows the basic architecture of a CNN.

Dataset	Method	Accuracy	Limitations	Our Contribution				
HAM10000	MobileNet[79]	83%	Focuses on accuracy;	Evaluates models for accu-				
			lacks discussion on	racy, F1-Score and compu-				
			computational effi-	tational efficiency				
			ciency					
HAM10000	FCRN (16	85.5% with	Computationally in-	Proposes real-time applica-				
	residual	segmenta-	tensive, segmentation	tions models				
	blocks)[80]	tion, 82.8% without	requirement					
KCGMH	DenseNet,	89.5%	Focuses on binary	Evaluates performance on				
HAM10000	EfficientNet[81	] (KCGMH),	classification	multi-class classification				
		85.8%						
		(HAM10000)						
HAM10000	ESRGAN	86%	High resource usage	Developed a lightweight so-				
+ ResNet-				lution for mobile applica-				
	50[82]			tions				
HAM10000	13 deep	82.9%	Low accuracy; Com-	Improve Accuracy; Devel-				
	transfer		putationally expen-	ops a mobile-friendly solu-				
	learning		sive	tion				
	models[83]							

Table 2.1: Summary of Related Works

### **Convolutional Layer**

The convolutional layer is the core building block of CNNs. It applies a set of learnable filters to the input image to extract features. Each filter scans through the input image and produces a feature map by performing elementwise multiplication and summation. The output feature maps capture different aspects of the input image, such as edges, textures, and patterns. This



Figure 2.12: CNNs Architecture.

powerful aspect enables Convolutional Neural Networks to automatically extract essential features at each layer, eliminating the necessity for manual feature engineering or selection. CNNs inherently possess the capability to learn hierarchical representations of data, starting from low-level features such as edges and textures and progressing to higher-level features that capture complex patterns and structures. The output dimension of convolutional layers can be calculated using the equation 2.2.

$$\left[\frac{n_h + 2p - f}{s} + 1\right] \times \left[\frac{n_w + 2p - f}{s} + 1\right]$$
(2.2)

 $n_h$  and  $n_w$  are the input image height and width, f is the filter size (both height and width), p is the padding (both height and width), and s is the stride length. This formula computes the height and width of the output feature map produced by a convolutional layer based on the input image's parameters, filter size, padding, and stride length.

**Stride:** The stride is a parameter within the filter, influencing the extent of movement across an image. When employing a stride of 1, the network processes data pixel by pixel. Alternatively, setting a stride of 2 entails pro-

2	6	8	-1	0	5	3								
2	7	4	3	2	8	-1								
-1	1	9	0	3	6	4		-1	0	1		20	-18	-2
0	9	6	8	2	9	7	*	-2	0	2	=	18	-15	9
8	1	4	-1	3	6	1		-1	0	1		-3	2	-21
7	2	6	0	7	2	-1								
0	9	3	8	4	9	1								

Figure 2.13: Filtering with the stride of 2.

cessing data while skipping every other pair of adjacent pixels. Figure 2.13 indicates the calculation for filtering with the stride of 2. After any calculation, the filter skips one column and one row after completing all columns. Below is the calculation for the first row of the output matrix.

 $(2 \times (-1)) + (6 \times 0) + (8 \times 1) + (2 \times (-2)) + (7 \times 0) + (4 \times 2) + ((-1) \times (-1)) + (1 \times 0) + (9 \times 1) = 20$ 

 $(8 \times (-1)) + ((-1) \times 0) + (0 \times 1) + (4 \times (-2)) + (3 \times 0) + (2 \times 2) + (9 \times (-1)) + (0 \times 0) + (3 \times 1) = -18$ 

 $(0 \times (-1)) + (5 \times 0) + (3 \times 1) + (2 \times (-2)) + (8 \times 0) + ((-1) \times 2) + (3 \times (-1)) + (6 \times 0) + (4 \times 1) = -2$ 

**Padding:** Padding pertains to the augmentation of an image with additional pixels during kernel processing. For instance, when employing zeropadding in a CNN, extra pixels with zero value are appended to the image. Applying filters or kernels to scan the image often results in a size reduction. To retain the original image dimensions and extract low-level features effectively, it becomes necessary to prevent such size reduction by adding supplementary pixels around the image boundaries. Figure 2.14 shows the padding for a  $5 \times 5$  matrix in a  $3 \times 3$  filtering process.

Generating the results matrix involves multiplying each element of the  $3 \times 3$  filter with its corresponding neighbor in the input matrix and summing these products. As an illustration, the first row values of the result matrix are computed in the following manner:

 $(0 \times (-1)) + (0 \times 0) + (0 \times 1) + (0 \times (-2)) + (7 \times 0) + (1 \times 2) + (0 \times (-1)) + (2 \times 0) + (9 \times 1) = 11$ 

 $(0 \times (-1)) + (0 \times 0) + (0 \times 1) + (7 \times (-2)) + (1 \times 0) + (2 \times 2) + (2 \times (-1)) + (9 \times 0) + (3 \times 1) = -9$ 

 $(0 \times (-1)) + (0 \times 0) + (0 \times 1) + (1 \times (-2)) + (2 \times 0) + (4 \times 2) + (9 \times (-1)) + (3 \times 0) + (7 \times 1) = 4$ 

 $(0 \times (-1)) + (0 \times 0) + (0 \times 1) + (2 \times (-2)) + (4 \times 0) + (8 \times 2) + (3 \times (-1)) + (7 \times 0) + (6 \times 1) = 15$ 

 $(0 \times (-1)) + (0 \times 0) + (0 \times 1) + (4 \times (-2)) + (8 \times 0) + (0 \times 2) + (7 \times (-1)) + (6 \times 0) + (0 \times 1) = -15$ 

In this instance, with the stride value of one, we shift the filter one column to the right, compute the second value of the results matrix, and so on.

#### Convolutions on RGB images

The composition of color images involves three distinct channels—red, green, and blue—each represented by a pixel intensity values matrix. The fusion

0	0	0	0	0	0	0										
0	7	1	2	4	8	0					_	11	-9	4	15	-15
0	2	9	3	7	6	0		-1	0	1		26	0	-6	5	-20
0	5	7	8	2	1	0	*	-2	0	2	=	27	8	-14	-9	-23
0	2	4	1	2	3	0		-1	0	1		24	-2	-12	-2	-12
0	7	9	4	6	5	0					-	22	-7	-8	5	-14
0	0	0	0	0	0	0										

Figure 2.14: Padding in filtering [10].

of these channels generates an RGB image. Notably, convolution operations for RGB images deviate from those applied to 2D images with one channel. Precisely, in RGB image convolution, the filter or kernel matches the number of channels in the input RGB image.

Illustrated in Figure 2.15, an RGB image with the dimension of  $6 \times 6 \times 3$ undergoes convolution with a filter sized  $3 \times 3 \times 3$ . This convolution yields a resulting output of dimensions  $4 \times 4$ , constituting a 2D image. Each pixel in this output is computed by multiplying and summing the 27 values within the  $3 \times 3 \times 3$  filter, aligned with their respective pixels in the input image. For the present example, no padding is applied, and a stride of 1 is assumed.

Convolutional layers typically integrate multiple filters in practical convolutional neural network (CNN) implementations. Incorporating a greater number of filters facilitates the extraction of additional features from the input data. The output is a volume where the number of output channels equals the number of filters. Each channel within the output represents the feature maps associated with its corresponding filter, as depicted in Figure 2.16. Here, the outcomes derived from two distinct filters yield an output featuring two channels.



Figure 2.15: Convolution on RGB images.



Figure 2.16: Convolution on RGB images with 2 filters.



Figure 2.17: One layer of a convolutional network.

#### One Layer of a Convolutional Network

Examine a single layer within a convolutional neural network (CNN) and explore how neural network principles can illuminate its operations. Figure 2.17 illustrates such a layer, where the input is a  $6 \times 6 \times 3$  RGB image, and the output is a  $4 \times 4 \times 2$  feature map. Each channel in the output represents distinct features extracted by individual filters.

In neural network mathematics, these filters can be viewed as matrices of weights, and the sample calculations may be coerced into standard matrix algebra. When the input is convolved with each filter, the resulting outputs undergo nonlinear activation functions and bias addition. These processed outputs from each filter are then stacked together to form the final output.

#### Convolutional layer notation

A summary of notation in a convolutional layer in a CNN network. If layer l is a convolutional layer, the dimension and notation for this layer are as

follows:

$$\begin{split} f^{[l]} &= filter \; size, \; p^{[l]} = padding, \; s^{[l]} = stride, \; n_c^{[l]} = number \; of \; filters \\ Input \; size : \; n_H^{[l-1]} \times n_W^{[l-1]} \times n_c^{[l-1]} \\ each \; filter \; size : \; f^{[l]} \times f^{[l]} \times n_c^{[l-1]} \\ activations \; size \; (a^{[l]}) : \; n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]} \\ activations \; matrix \; size \; (A^{[l]}) : \; m \times n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]} \\ wieghts \; size : \; f^{[l]} \times f^{[l]} \times n_c^{[l-1]} \times n_c^{[l]}, \; \text{where} \; n_c^{[l]} \; \text{is number of filters in} \end{split}$$

layer l.

bias size : 
$$n_c^{[l]}$$
  
Output size :  $n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]}$   
 $n_H^{[l]} = \frac{n_H^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1$   
 $n_W^{[l]} = \frac{n_W^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1$ 

#### Pooling Layer

Pooling layers downsample, or reduce the dimension through sampling, of the feature maps generated by convolutional layers, reducing their spatial dimensions. Max pooling and average pooling are two commonly used pooling techniques. Max pooling selects the maximum value within each pooling region, while average pooling computes the average values. Pooling helps to reduce computational complexity, control overfitting, and increase the network's translational invariance [90]. The pooling layer does not have any



Figure 2.18: Pooling layer.

parameters for learning in the network. Figure 2.18 indicates the max and average pooling process.

The pooling layer hyperparameters are outlined as follows:

 $f: filter \ size, \ s: stride, \ and \ Pooling \ type: [max, average]$ 

In most instances, padding is not applied in pooling layers, except for certain special cases. Typically, common values for the stride (s) and filter size (f) parameters are set to 2. This configuration results in a halving of the input dimension in each pooling layer.

### Fully Connected (Dense) Layer

Fully connected layers are artificial neural network layers where each neuron is connected to every neuron in the previous layer. These layers integrate high-level features extracted by convolutional and pooling layers, which are then used for final classification or regression tasks. The outputs from fully connected layers are typically passed through activation functions.

#### Dropout

Deep neural networks are powerful tools in supervised learning but often face a significant challenge known as overfitting. Overfitting occurs when a model learns to perform exceptionally well on the training data but fails to generalize to new, unseen data. This issue is prevalent in deep networks due to their large number of parameters.

Dropout is a regularization technique designed to combat overfitting. It randomly removes a subset of neurons (along with their connections) from the neural network during training. This forces the network to learn more robust features, as no single neuron can rely on the presence of others. During training, dropout effectively generates numerous "thinned" networks. During testing, the averaging effect of these thinned networks is approximated by using a single network with scaled-down weights, significantly reducing overfitting and improving generalization [91]. Dropout is typically governed by a probability parameter p, which determines the percentage of neurons to exclude from the network, often ranging between 0.2 and 0.5. Figure 2.19 illustrates the dropout process in neural networks.

Please refer to the Evaluation metrics section (see Section 3.1.4) for a more detailed discussion of overfitting.

#### Convolutional neural networks used in this research

In this study, we employ pre-trained CNN models and adjust their parameters to address challenges in skin lesion detection. The CNN architectures utilized in this study include VGG16 [92], VGG19 [92], MobileNet [93], MobileNetV2



Figure 2.19: Dropout layer. [10].

[94], MobileNetV3 [13], and ResNet [12]. Below, we outline the architectures and distinctive features of these CNN families.

## VGG16 & VGG19

VGG16 and VGG19 are convolutional neural network models introduced by K. Simonyan and A. Zisserman from the Visual Geometry Group at the University of Oxford [92]. The numbers 16 and 19 indicate the number of weight layers in these models. These models gained prominence for their exceptional performance, achieving a top-5 test accuracy of 92.7% on the ImageNet dataset, comprising over 14 million images distributed across 1000 classes. VGG16 was notably submitted to the ILSVRC-2014 competition, where it showcased significant improvements over its predecessor, AlexNet. Figure 2.20 shows the VGG16 architecture.

One of the key advancements of VGG16 over previous models like AlexNet [95] is its utilization of multiple  $3 \times 3$  kernel-sized filters in place of larger kernel sizes (e.g.,  $11 \times 11$  and  $5 \times 5$  in the first and second convolutional layers of



Figure 2.20: VGG16 architecture. [11].

AlexNet). The network architecture consists of convolutional layers followed by ReLU activation functions, with a fixed input size of  $224 \times 224 \times 3$  RGB images. All the convolutional layers in VGG16 have  $3 \times 3$  filters, stride of 1, and padding of 1, so the input and output of each convolutional layer have the same size. VGG16 uses  $3 \times 3$  filters to capture spatial features effectively.

VGG16 incorporates spatial pooling through five max-pooling layers, which are interspersed among the convolutional layers. Max-pooling layers use  $2 \times 2$  filters with a stride of 2, aiding in downsampling and feature extraction.

VGG16 includes three fully connected (FC) layers following the convolutional layers. The first two contain 4096 channels each, and the third performs a 1000-way classification for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The final layer employs a softmax function for classification.

#### Residual Neural Network (ResNet)

The ResNet is a deep learning model designed for computer vision tasks. It introduced significant advancements in the ILVRSC 2015 competition. ResNet achieved unprecedented results by effectively addressing challenges associated with training profound neural networks. ResNet surpassed other architectures by a substantial margin, winning the image classification task in ILVRSC 2015 with an impressive top-five error rate of 3.57% [12].

One of the primary issues ResNet aims to tackle is the Vanishing/Exploding Gradient Problem commonly encountered in deeper neural networks. As the number of layers increases, gradients of the loss function to the weights may become either excessively small or excessively large during backpropagation, hindering effective learning.

The key components of the ResNet architecture include:

**Residual Block:** Residual blocks are fundamental components of Residual Neural Networks. Unlike in plain neural networks, where the input is transformed by convolutional layers and passed through an activation function, ResNet introduces a residual connection. In a residual block, the input to the block is added to the output, creating a residual connection:

$$a^{[l+2]} = g^{[l+2]}(z^{[l+2]} + a^{[l]})$$
(2.3)

Here,  $g^{[l+2]}$  represents the activation function in layer l + 2. Figure 2.21 represents the residual block.

Skip Connection: Skip connections play a crucial role in forming residual



Figure 2.21: Residual Block.



Figure 2.22: Residual Network.

blocks. They involve bypassing the residual block's input over the convolutional layer and adding it to the block's output.

Stacked Layers: ResNet architectures are constructed by stacking multiple residual blocks together. By leveraging these stacked residual blocks, ResNet can achieve remarkable depth. Various versions of ResNet, including those with 50, 101, and 152 layers, were introduced. Figure 2.22 represents stacking residual blocks to make a residual network.

Global Average Pooling (GAP): ResNet architectures typically employ Global Average Pooling as the final layer before the fully connected layer.


Figure 2.23: ResNet Architecture. [12].

GAP reduces spatial dimensions to a single value per feature map, providing a compact representation of the entire feature map. Figure 2.23 shows the architecture of ResNet models compared to VGG16 and plain networks.

### MobileNets

MobileNets represent a class of efficient models tailored for mobile and embedded vision tasks. They employ a streamlined architecture that relies on depthwise separable convolutions to construct lightweight deep neural networks [93]. One notable feature of MobileNets is the incorporation of two straightforward global hyperparameters, which effectively balance latency and accuracy. These hyperparameters offer model builders the flexibility to select an appropriately sized model that aligns with the constraints of their specific application.

MobileNets exhibit effectiveness across diverse applications and use cases, spanning object detection, fine-grained classification, analysis of facial attributes, and large-scale geo-localization tasks. This versatility underscores the adaptability and practical utility of MobileNets in various scenarios, particularly those requiring efficient and accurate vision processing on resourceconstrained devices.

The core concept underlying the MobileNet revolves around utilizing depthwise separable convolutions instead of conventional convolutions, aiming to diminish computational complexity and model size. This approach disassembles the standard convolution operation into two distinct stages: depthwise convolution and pointwise convolution.

Depthwise convolution conducts independent convolutions on each input channel, utilizing a single filter per channel. Compared to conventional convolutions, this segmentation minimizes the number of parameters and computational requirements. It employs a  $3 \times 3$  depthwise convolution with a stride of 1, followed by batch normalization and ReLU activation, thereby effectively capturing spatial information within each channel.

Pointwise convolution operates on the output of the depthwise convolution, employing a  $1 \times 1$  convolution to amalgamate information across channels. It uses a small number of  $1 \times 1$  filters to facilitate cross-channel feature combinations and dimensionality reduction, enabling the mixing and transformation of features from diverse channels. Figure 2.24 indicates the standard convolution and depthwise separable convolution.

MobileNet efficiently reduces parameters and computations while maintaining satisfactory accuracy by dividing the convolution process into specific stages. Utilizing depthwise separable convolutions enables the network to obtain concise representations of input data, rendering MobileNet suitable for



Figure 2.24: Normal Convolution Vs. Depthwise Separable Convolution. [6].



Figure 2.25: MobileNet architecture. [6].

resource-limited scenarios. The architecture of MobileNet, depicted in Figure 2.25, comprises 13 blocks of depthwise separable convolutional layers as described in the original paper by [93].

MobileNet version 2 (MobileNetV2) represents a significant advancement in mobile model performance across various tasks and benchmarks and in different model sizes [94]. The architecture of MobileNetV2 revolves around an inverted residual structure, which diverges from traditional residual models by employing thin bottleneck layers at the input and output of the residual block. Figure 2.26 represents the MobileNetV2 architecture.



Figure 2.26: MobileNet Version 2 architecture. [6].

MobileNetV2 adopts an inverted residual structure, where the input and output of the residual block consist of thin bottleneck layers. This design contrasts with conventional residual models that typically use expanded representations in the input.

Instead of employing standard convolutions, MobileNetV2 utilizes lightweight depthwise convolutions to filter features within the intermediate expansion layer. This approach helps reduce computational complexity and model size. So and so performed a comparison and found this approach reduced computational complexity and model size while maintaining effective feature extraction [94].

Removal of Non-linearities in Narrow Layers: MobileNetV2 removes nonlinear activation functions in the narrow layers to preserve the representational power, referring to its ability to capture and model the complex patterns and structures in the input data [94]. This design choice ensures that the model can capture intricate patterns and features, even in layers with fewer parameters.

Overall, MobileNetV2's innovative architectural design, incorporating inverted residual structures and lightweight depthwise convolutions, enhances performance across various tasks and model sizes. By prioritizing efficiency



Figure 2.27: MobileNet Version 3 architecture. [13].

without compromising accuracy, MobileNetV2 remains a powerful solution for mobile and embedded vision applications [94].

MobileNet version 3 (MobileNetV3) introduces complementary search techniques and innovative architectural designs. Tailored specifically for mobile phone CPUs, MobileNetV3 integrates hardware-aware network architecture search (NAS) [13] alongside the NetAdapt algorithm [13], further refined through novel architecture advancements. This iteration introduces two variants, MobileNetV3-Large and MobileNetV3-Small, catering to high and low-resource use cases. Figure 2.27 indicates the architecture of MobileNetV3.

Compared to MobileNetV2, MobileNetV3 incorporates the Squeeze and Excitation (SE) module, initially introduced in SENet [96], to enhance feature learning. To improve computational efficiency, MobileNetV3 replaces the sigmoid activation function in the SE module with the hard-sigmoid function, where the 2.4 and 2.5 indicate the equation for this function. Additionally, MobileNetV3 replaces the traditional ReLU activation function with the Swish activation function to enhance non-linearity [13]. 2.6 shows



Figure 2.28: MobileNetV3 activation functions. [13].

the swish function.

$$RELU6(x) = min(max(0, x), 6)$$
(2.4)

$$h - sigmoid(x) = \frac{RELU6(x+3)}{6}$$
(2.5)

$$h - swish(x) = x.h - sigmoid(x)$$
(2.6)

The swish activation can be computationally inefficient on mobile and embedded hardware [97, 13]. This issue was spurned the hard-swish (H-Swish) and was incorporated in mobilenetv3. H-Swish retains the non-linear properties of Swish while offering improved efficiency for mobile hardware implementations. This ensures that MobileNetV3 maintains high performance while being well-suited for deployment on mobile and embedded devices [13]. Figure 2.28 represents the activation functions of the MobileNetV3 model.

## 2.2.8 Transfer Learning

With the remarkable advancements in deep learning, transfer learning has become a central element in various computer vision fields, including multimedia [98], surveillance [99], and medical applications [100]. The concept involves leveraging pre-trained models originally trained on non-medical or natural image datasets. These models are then fine-tuned with new data to adapt to specific tasks [101]. Transfer learning is crucial in deploying convolutional neural networks for diagnostic imaging tasks such as skin cancer detection [37], Alzheimer's Disease diagnosis [102], and chest X-ray analysis [103].

Figure 2.29 illustrates the architectures used in the transfer learning approach. Typically, open-source pre-trained models are trained on extensive datasets containing numerous classes. For instance, the ImageNet dataset comprises 14 million images distributed across 1000 classes. Transfer learning allows us to modify pre-trained networks by replacing the top layer with an output layer tailored to our dataset. Depending on the size of our dataset, we can adjust or fine-tune the parameters of the pre-trained models to better suit our specific needs.



Figure 2.29: Transfer learning methodology.

# Chapter 3

# Methodology

This chapter focuses on detecting skin cancer in the HAM10000 dataset and the pre-trained CNN methods. Figure 3.1 visually depicts the critical steps of our methodology.

We use an ASUS TUF Gaming A15 system with AMD Ryzen 7 6800H processor information with Radeon Graphics, 3201 Mhz, 8 Core(s), 16 Logical Processor(s), and 16GB of RAM.

# 3.1 Main stages of the methodology

The methodology consists of four main steps: pre-processing, data augmentation, model architecture, and evaluation metrics.



Figure 3.1: Our methodology process steps.

### 3.1.1 Pre-processing

Pre-processing plays a crucial role in detecting skin cancer using deep learning models. This improves the performance of our models. The initial stage of our analysis involved reading images from the dataset and pre-processing them. The initial size of images is  $600 \times 450 \times 3$ , and we resized images to dimensions of  $244 \times 224 \times 3$ , ensuring compatibility with the convolutional neural network (CNN) architectures employed. Additionally, we use built-in functions in the Keras [104] library in Python for data normalization to enhance the uniformity of their pixel values, thus preparing them for subsequent training procedures. After pre-processing, we divided the dataset into training, validation (development), and test sets with an 80/10/10 ratio.

### 3.1.2 Data Augmentation

The HAM10000 dataset includes an imbalanced distribution, where some categories have many images while others have only a few. The imbalance is one of the significant challenges because classifiers tend to be influenced by the dominant class while neglecting the smaller ones [105]. This means that the classifier does not achieve the desired level of accuracy across all classes. The idea of resampling can be applied to tackle this problem. Data augmentation, achieved by applying transformations to images, is commonly employed to mitigate the challenges of imbalanced datasets. The available data can be diversified by augmenting the dataset through transformations such as rotation, flipping, scaling, and cropping, helping to address the class imbalance issue. This augmentation process enriches the dataset with variations of existing images, providing the model with a more comprehensive understanding of different instances within each class. Consequently, it enhances the model's ability to generalize effectively across all classes, even in scenarios where certain classes are underrepresented in the original dataset.

### 3.1.3 Model Architecture

We use pre-trained CNN models and try to fine-tune their parameters to alleviate skin lesion detection issues. Transfer learning gives us the power of flexibility in using all the parameters of these powerful CNN models or freezing the parameters and just using pre-trained weights. The exploration of transfer learning, utilizing pre-trained models such as VGG16 [92], VGG19 [92], MobileNet [93], MobileNet V2 [94], MobileNet V3 [13], and ResNet [12], is integral to the project. Transfer learning is employed to enhance the generalization ability of computer-aided diagnostic systems. Figure 3.2 represents the model architecture, where we drop the top layer of pre-trained models and add average pooling, dropout, and softmax layers with the number of classes in our dataset.



Figure 3.2: Model architecture.

In our skin lesion detection application, we employ transfer learning by adjusting the pre-trained weights of VGG16, VGG19, ResNet50, MobileNetV1, MobileNetV2, and MobileNetV3, which collectively have been trained on the ImageNet dataset including over 14 million images of 1000 classes. This allows us to capitalize on the rich feature representations these models learn from diverse categories. We fine-tune pre-trained model parameters, including weights and biases, based on the HAM10000 dataset, enhancing the performance and accuracy of our skin lesion detection systems. This process involves removing the top layer of the networks and replacing it with average pooling, dropout, and softmax layers tailored to classify our dataset's categories, including seven classes. By adapting the pre-trained weights to our unique classification tasks, we optimize the performance of our models for effective skin cancer lesion detection, saving computational resources and training time while leveraging the generalization power of these architectures.

### 3.1.4 Evaluation Metrics

Evaluating the performance of a skin cancer detection model is essential to assess its accuracy and effectiveness. In deep learning, it is crucial to ensure that a model performs well on the training data and generalizes effectively to unseen data. This is where the concepts of overfitting and underfitting become particularly important.

Overfitting and underfitting are two critical challenges in machine learning that directly impact model performance. Overfitting occurs when a model is too complex, capturing noise and outliers in the training data rather than the underlying distribution. This results in excellent performance on the training data but poor generalization to new data. Techniques such as dropout, cross-validation, and regularization are commonly employed to mitigate overfitting [106].

In contrast, underfitting happens when a model is too simple to capture the underlying patterns in the data. An underfit model fails to perform well even on the training data, leading to poor predictions. To address underfitting, one might consider increasing model complexity, using more sophisticated algorithms, or providing more features to the model [106].

Based on established evaluation metrics in the skin cancer image classification domain [15, 19, 56, 80], this thesis assesses the performance of the models using metrics such as accuracy and weighted F1-score. Overfitting or underfitting is monitored by comparing the performance of both the training and validation datasets.

Accuracy: Accuracy measures the proportion of correctly classified instances from the total number of cases in the dataset. Equation 3.1 indicates how we calculate the accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
(3.1)

Where:

TP (True Positives) are the instances correctly classified as positive. TN (True Negatives) are the instances correctly classified as negative. FP(False Positives) are the instances incorrectly classified as positive. FN (False Negatives) are the instances incorrectly classified as negative.

**Precision:** Precision measures the proportion of true positive predictions among all positive predictions made by the model. Equation 3.2 shows the formula for calculating the precision.

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

**Recall (Sensitivity)**: Recall measures the proportion of true positive predictions among all actual positive instances in the dataset. The formula for calculating recall is shown in equation 3.3.

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

**F1-Score:** The F1-score is the harmonic mean of precision and recall, balancing the two metrics. Equation 3.4 indicates the formula for F1-Score.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$
(3.4)

75

Weighted F1-Score: The weighted F1-score is a metric used to evaluate the performance of a classification model, particularly in scenarios where there is class imbalance [107]. It is computed as the weighted average of the F1-scores for each class, with the weights assigned based on the number of actual occurrences (true instances) of each class in the dataset. This weighting ensures that classes with more instances significantly influence the final score, which can be crucial in datasets where some classes are underrepresented.

**F1-Score for each class:** calculate F1-score for each class using 3.4, and the computing the weighted F1-score using

$$Weighted_F 1 = \sum_{i=1}^C F 1_i \times W_i \tag{3.5}$$

C denotes the total number of classes in the dataset, and  $W_i$  represents the weight for class i. Specifically,  $W_i$  is the proportion of true instances of class i relative to the total number of instances in the dataset. This means that classes with more true instances contribute more to the weighted F1score, reflecting their prevalence in the dataset.

**Training time:** We assess the models' performance by considering their training time alongside other evaluation metrics to determine which model is more efficient and suitable for real-world applications, particularly low-power devices.

# Chapter 4

# **Results & Discussion**

In this chapter, we present the findings of our investigation into skin lesion detection using deep learning models.

# 4.1 Transfer Learning and Data Augmentation

# 4.1.1 Parameter Tuning and Implementation Details

We fine-tuned the parameters for the pre-trained models, including MobileNetV1, MobileNetV2, MobileNetV3, VGG16, VGG19, and ResNet50. Fine-tuning was carried out on specific layers tailored to each architecture, as follows:

#### **Tuning Process:**

To optimize the performance of these models, we conducted extensive parameter tuning. The critical parameters adjusted during this process included:

**Batch Size:** We experimented with 16, 32, 64, 128, and 256 batch sizes to identify the optimal size for efficient learning and model convergence.

**Learning Rate:** Various learning rates, ranging from 0.00001 to 0.01, were tested to ensure the models converged effectively without overshooting the optimal point.

Number of Epochs: We varied the number of epochs, testing 10, 20, 50, 70, and 100 epochs to balance sufficient training and the prevention of overfitting.

Layers for Fine-Tuning: Depending on the architecture, various layers were tested to determine the best configuration for the model. Finally, specific layers were selected for fine-tuning based on performance.

**Dropout:** Different dropout probabilities ranging from 0.1 to 0.9 were used to prevent overfitting and find the most robust model.

#### Implementation:

The tuning process was implemented using Python's Keras [104] library. For each model, we monitored performance metrics on the validation set to identify the best combination of parameters. The final results were reported based on validation accuracy and loss.

Here are the parameters selected during the freezing and unfreezing stages:

#### Freezing Stage:

Optimizer: Adam with parameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and  $\alpha = 0.001$ . Dropout: p = 0.2Batch Size: 32 Epochs: 20 **Unfreezing Stage:** Learning Rate:  $\alpha = 0.0001$ Epochs: 50

This detailed tuning and implementation strategy ensured that each model was fine-tuned to achieve optimal performance on our skin cancer detection task.

### 4.1.2 CNN models without data augmentation

In this section, we present the results obtained from our experimentation with transfer learning techniques applied to pre-trained CNN models on the HAM10000 dataset. We initially explore the performance of these models without any data augmentation, focusing on the freezing and training of pre-trained weights while training the top layers on our dataset. Table. 4.1 presents the outcomes obtained from employing pre-trained models with frozen weights and without data augmentation, focusing on accuracy and F1-score evaluation metrics. Based on the results, ResNet50 achieves the highest accuracy among the considered models, indicative of its superior adaptability when utilizing frozen pre-trained weights and adjusting the upper layers to our dataset. Conversely, MobileNetV3 exhibits the most efficient runtime, emphasizing its potential suitability for real-time applications and low-power devices.

Model	Training	Training	Validation	Validation	Dun Time	
Model	Accuracy	uracy F1-Score Ad		F1-Score	Itun Inne	
VGG16	0.7679	0.5350	0.749	0.5289	128m 39s	
VGG19	0.7644	0.5777	0.754	0.5721	$162m\ 19s$	
ResNet50	0.8534	0.7531	0.8085	0.6451	$61m \ 48.5s$	
MobileNetV1	0.8156	0.6878	0.7944	0.6065	$20\mathrm{m}~51.4\mathrm{s}$	
MobileNetV2	0.8179	0.7039	0.7752	0.5469	$22\mathrm{m}~53.9\mathrm{s}$	
MobileNetV3	0.7843	0.6027	0.7772	0.5429	6m 57.6s	

Table 4.1Pre-trained CNNs without Data Augmentation.

#### Parameter Tuning and Implementation Details:

"

Following that, we proceed with fine-tuning the parameters of the pretrained models, focusing on specific layers tailored to each architecture. For MobileNetV1, MobileNetV2, MobileNetV3, VGG16, VGG19, and ResNet50, the fine-tuning of parameters commences from layers 50, 100, 120, 10, 13, and 120 out of a total of 86, 154, 157, 19, 22, and 175 layers, respectively. Table 4.2 showcases transfer learning results utilizing pre-trained CNN models, where weights are trained based on the HAM10000 dataset without resampling or data augmentation. ResNet50 is the top performer in training and validation accuracy and the F1-score. Furthermore, MobileNetV3 demonstrates rapid training, requiring less than 9 minutes, while achieving a training accuracy of 98.54% and a validation accuracy of 85.79%.

Model	Training	Training	Validation	Validation	Dun Time
Model	Accuracy	F1-Score	Accuracy	F1-Score	Kun 11me
VGG16	0.9842	0.8868	0.8427	0.7752	$263m \ 16.5s$
VGG19	0.9868	0.8903	0.8306	0.7511	$277\mathrm{m}~51\mathrm{s}$
ResNet50	0.9934	0.9233	0.8639	0.8131	$92m\ 40.1s$
MobileNetV1	0.9712	0.8561	0.833	0.761	$32m\ 21.2s$
MobileNetV2	0.9823	0.8839	0.8538	0.7782	$37m\ 22.5s$
MobileNetV3	0.9854	0.8871	0.8579	0.7802	8m 44.8s

Table 4.2Fine-tuned CNNs without Data Augmentation.

Examine the confusion matrix of the models on the test data. Table 4.3 illustrates the true labels of the test dataset, which comprises 1001 images from 7 classes.

Table 4.4 displays the confusion matrix results for MobileNetV1 on the test dataset without augmentation. The confusion matrix results reveal that MobileNetV1 successfully identifies the nv skin lesion family, achieving 642 correct predictions out of 654 instances. Bkl lesions also show a relatively high number of correct predictions (77). However, the model struggles with the vasc lesion family, which is frequently misclassified. Precisely, mel lesions are often mistaken for nv and bkl, with 64 and 30 instances, respectively. Akiec and df also show considerable misclassifications between these classes and others like bcc and nv. This indicates that MobileNetV1 has difficulty distinguishing between these classes.

Table 4.5 illustrates the confusion matrix results for MobileNetV2 on the

			Predicted Label							
		akiec	bcc	bkl	df	nv	mel	vasc		
~	akiec	37	0	0	0	0	0	0		
abe	bcc	0	51	0	0	0	0	0		
le L	bkl	0	0	108	0	0	0	0		
Tru	df	0	0	0	10	0	0	0		
	nv	0	0	0	0	654	0	0		
	mel	0	0	0	0	0	131	0		
	vasc	0	0	0	0	0	0	10		

Table 4.3: True labels of the test dataset.

Table 4.4: Confusion matrix of MobileNetV1 on the test dataset.

					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
~	akiec	19	3	9	0	6	0	0
abe	bcc	1	31	9	0	9	1	0
eĽ	bkl	0	2	77	2	26	1	0
2	df	0	1	3	4	2	0	0
	nv	2	2	7	0	642	1	0
	mel	2	1	30	0	64	33	1
	vasc	0	0	0	0	3	0	7

			Predicted Label							
		akiec	bcc	bkl	df	nv	mel	vasc		
-	akiec	20	1	5	0	10	1	0		
abe	bcc	6	24	5	0	12	3	1		
le L	bkl	5	0	71	0	28	4	0		
Tru	df	1	1	2	4	2	0	0		
	nv	3	3	13	0	628	6	1		
	mel	2	1	30	0	47	51	0		
	vasc	0	0	1	0	3	0	6		

Table 4.5: Confusion matrix of MobileNetV2 on the test dataset.

test dataset without augmentation. While MobileNetV2 performs well on identifying nv (628 correct predictions), it struggles with mel lesions correctly identifying only 51 out of 131 samples, frequently misclassifying them as nv and bkl. There are also notable confusions between akiec and mel with nv.

Table 4.6 indicates the confusion matrix for MobileNetV3 on the test dataset without augmentation. The model correctly identifies 16 instances of akiec but struggles with misclassifications, especially with bkl and nv. It correctly predicts 36 cases of bcc lesions, yet confusion remains with akiec, bkl, nv, and mel. While the model excels in identifying bkl with 64 correct predictions, it also misclassifies these as nv and mel. Df is relatively well-predicted, though minor confusions with other types persist. The model performs strongly in predicting nv with 602 correct predictions but faces challenges with misclassifications such as bkl and mel. Similarly, while it correctly identifies 65 instances of mel, significant misclassifications with bkl

			Predicted Label							
		akiec	bcc	bkl	df	nv	mel	vasc		
-	akiec	16	2	10	0	5	4	0		
abe	bcc	4	36	4	0	3	3	1		
le L	bkl	4	4	64	2	24	10	0		
Tru	df	1	2	0	6	1	0	0		
	nv	3	8	14	1	602	26	0		
	mel	2	1	22	1	40	65	0		
	vasc	0	1	0	0	1	0	8		

Table 4.6: Confusion matrix of MobileNetV3 on the test dataset.

and nv occur. Vasc identification is highly accurate, with eight instances correctly classified. Overall, MobileNetV3 demonstrates excellent performance in identifying nv and vasc, but struggles significantly with distinguishing mel and faces challenges with certain other class confusions.

Table 4.7 represents the confusion matrix for VGG16 on the test dataset without augmentation. The model identifies 18 akiec instances. However, it misclassified this akiec class with mel and nv. Regarding the bcc class, the model demonstrates a correct prediction rate of 33 cases; nevertheless, confusion persists with akiec, bkl, nv, and mel. While the model accurately predicts 65 bkl cases, it misclassifies some instances as nv and mel. Df prediction encounters minor confusion with other lesion types. The model showcases proficiency in predicting nv, accurately identifying 618 instances, albeit facing challenges with misclassifications as bkl, bcc, and mel. Similarly, while correctly identifying 87 mel instances, misclassifications with nv are

					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
-	akiec	18	2	3	0	5	9	0
abe	bcc	7	33	1	0	5	5	0
e Lé	bkl	5	3	65	1	16	18	0
Tru	df	0	1	1	7	0	1	0
	nv	0	5	6	1	618	23	1
	mel	3	0	5	0	36	87	0
	vasc	0	1	0	0	0	0	9

Table 4.7: Confusion matrix of VGG16 on the test dataset.

noted. Vasc classification is highly accurate, with nine instances correctly identified out of 10. In summary, while the VGG16 model performs well in identifying nv and vasc lesions, it struggles when classifying mel and akiec instances.

Table 4.8 depicts the confusion matrix for VGG19 on the test dataset without augmentation. The model successfully identifies 25 instances of akiec. However, it misclassifies some akiec cases as bcc, bkl, and nv. Regarding the bcc class, the model achieves an accuracy rate of 30 cases; nonetheless, confusion persists with akiec, bkl, nv, and mel. While accurately predicting 75 bkl cases, the model also misclassifies some as nv and mel. Confusion with other lesion types is observed in df prediction. Notably, the model demonstrates proficiency in predicting nv, correctly identifying 614 instances, despite encountering challenges with misclassifications as bkl, bcc, and mel. Similarly, while correctly identifying 76 mel instances, misclassifications with

					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
-	akiec	25	2	7	0	3	0	0
abe	bcc	7	30	3	0	10	1	0
e Li	bkl	5	2	75	0	15	11	0
Tru	df	3	0	1	5	1	0	0
	nv	3	5	12	0	614	19	1
	mel	5	0	14	0	34	76	2
	vasc	0	0	0	0	0	0	10

Table 4.8: Confusion matrix of VGG19 on the test dataset.

nv, bkl, and akiec are observed. Vasc classification is highly accurate, with all ten cases correctly identified. In conclusion, although the VGG19 model performs well in identifying vasc lesions, it faces difficulties in accurately classifying df instances.

Table 4.9 shows the confusion matrix for ResNet50 on the test dataset without augmentation. The model identifies 16 instances of akiec, but it incorrectly categorizes some as bkl and nv. For the bcc class, the model achieves an accuracy rate of 28 cases; however, confusion remains with other classes. While accurately predicting 63 bkl cases, it also misclassifies some instances as nv and mel. Df prediction, with six cases correct out of 10, experiences confusion with other lesion types. Notably, the model demonstrates proficiency in predicting nv, accurately identifying 630 instances, despite encountering challenges with misclassifications as bkl, bcc, vasc, and mel. Similarly, while correctly identifying 65 mel instances, misclassifications with nv

					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
-	akiec	16	2	7	2	9	1	0
abe	bcc	3	28	3	3	4	8	2
e Lé	bkl	2	0	63	2	18	21	2
Tru	df	0	0	1	6	3	0	0
	nv	0	5	5	1	630	9	4
	mel	2	1	14	0	45	65	4
	vasc	0	0	0	0	0	1	9

Table 4.9: Confusion matrix of ResNet50 on the test dataset.

and bkl are observed. Vasc classification stands out for its high accuracy, correctly identifying nine out of 10 cases. In summary, although the ResNet50 model performs well in identifying nv lesions, it faces difficulties in accurately classifying akiec instances.

In summary, the findings from Table 4.1 and Table 4.2 underscore the efficacy of transfer learning in leveraging pre-trained models for skin lesion detection, despite working with imbalanced datasets such as HAM10000. However, the confusion matrices provide deeper insights into how dataset imbalance impacts model classification challenges. Moreover, the results suggest that models perform better in classes with more training data, highlighting the importance of dataset balance in improving classification accuracy.

### 4.1.3 CNN models with data augmentation

After assessing the effectiveness of transfer learning in using pre-trained models across various image types and fine-tuning the weights based on our dataset, we try to address the imbalance issue in the HAM10000 dataset. To enhance the performance of our model, we applied transfer learning in conjunction with data augmentation techniques. Data augmentation helps to increase both the diversity and size of the training dataset without the need for additional data collection. Specifically, we employed the following augmentation methods:

### Geometric Transformations:

**Rotation:** Random rotations up to  $\pm 20$  degrees.

Horizontal Flip: Random flipping of images.

**Random Cropping:** Randomly cropping a portion of the image to simulate different perspectives.

### **Color Transformations**

**Brightness Adjustment:** Random adjustments to the brightness of the images.

These transformations were applied to generate additional instances for each class in the training dataset.

Figure 4.1 shows the distribution of images across the different classes before applying data augmentation. Each class is represented by numerical values where 0, 1, 2, 3, 4, 5, and 6 correspond to 'nv,' 'mel,' 'bkl,' 'bcc,'



Figure 4.1: Distribution of skin lesions in training dataset before augmentation.

'akiec,' 'vasc,' and 'df,' respectively. This figure highlights the class imbalance present in the original dataset.

After applying the augmentation techniques, the dataset was balanced by generating additional samples for each class. Specifically, each class contained exactly 2,000 samples, resulting in a balanced training dataset with a total of 14,000 images (2,000 samples per each of the 7 classes).

Table 4.10 presents the outcomes obtained using pre-trained CNN models without any fine-tuning on weights coupled with data augmentation. Once more, ResNet50 emerges as the top performer, demonstrating strong performance in training and validation accuracy and F1-score metrics. Specifically, by solely training the top layers and leveraging knowledge transferred from other datasets, ResNet50 achieves a validation accuracy of 84.94% and an F1-score of 83.59%. Conversely, MobileNetV3 demonstrates comparatively lower accuracy and F1-score than ResNet50 and different versions of the MobileNet model. However, it stands out in terms of runtime efficiency and computational costs.

Following data augmentation, we proceeded to fine-tune the parame-

Madal	Training	Training	Validation	Validation	Dun Time
Model	Accuracy	F1-Score	Accuracy	F1-Score	nun 1ime
VGG16	0.7753	0.7077	0.7668	0.6954	182m 33.6s
VGG19	0.7788	0.7268	0.7690	0.7055	$228m \ 11.3s$
ResNet50	0.8550	0.8494	0.8359	0.8171	86m 14.6s
MobileNetV1	0.8089	0.8013	0.8041	0.7663	$26\mathrm{m}~51.5\mathrm{s}$
MobileNetV2	0.8148	0.8056	0.8115	0.7834	$30\mathrm{m}~10.9\mathrm{s}$
MobileNetV3	0.7961	0.7770	0.7833	0.7581	10m 4.2s

Table 4.10Frozen Weight CNNs with Data Augmentation.

ters of the pre-trained models, incorporating the augmented data. Specifically, for MobileNetV1, MobileNetV2, MobileNetV3, VGG16, VGG19, and ResNet50, parameters were fine-tuned from specific layers within each architecture. These layers were chosen based on their position within the network architecture to strike a balance between retaining the learned features from the pre-trained model and adapting to the specifics of the target dataset. Table 4.11 showcases the outcomes obtained through transfer learning utilizing pre-trained CNN models, where the weights are trained based on the HAM10000 dataset with data augmentation.

After fine-tuning with augmented data, all methods exhibited commendable accuracy and F1-score performance. ResNet50, once again, emerged as a top performer, achieving 99.89% accuracy on the training dataset and 92.31% accuracy on the validation dataset. Following ResNet50, MobileNetV2, MobileNetV3, VGG16, VGG19, and MobileNetV1 demonstrated progressively better accuracy performance.

Runtime is a crucial metric for gauging the computational costs incurred

by the models. Larger networks such as VGG19, VGG16, and ResNet50 incurred significantly higher computational costs. Among them, VGG19, with almost 430 minutes, had the highest training time. Conversely, MobileNet models demonstrated notable efficiency in terms of computational costs. Among these, MobileNetV3 stood out with a training time of less than 13 minutes, making it an optimal choice for resource-constrained devices like smartphones.

This comprehensive evaluation underscores the effectiveness of transfer learning and data augmentation in addressing class imbalance and enhancing model performance across various CNN architectures. ResNet50 performs best, while MobileNetV3 offers an attractive balance between performance and computational efficiency.

Let's proceed to table 4.11 to delve into each model's detailed performance metrics and runtime statistics.

Madal	Training	Training	Validation	Validation	Dun Time
Model	Accuracy	F1-Score	Accuracy	F1-Score	run 1 me
VGG16	0.9936	0.9896	0.9106	0.9052	327m 23.1s
VGG19	0.9944	0.9902	0.9092	0.9064	430m 20.8s
ResNet50	0.9989	0.9957	0.9231	0.9198	133m 28.1s
MobileNetV1	0.9949	0.9815	0.9011	0.8981	48m 11.9s
MobileNetV2	0.9959	0.9911	0.9161	0.9131	38m 9s
MobileNetV3	0.9951	0.9903	0.9155	0.9112	13m 27.9s
11					

Table 4.11Fine-tuned CNNs with Data Augmentation.

The confusion matrices for the models on the test data after data augmentation provide valuable insights into their performance. Let's analyze each model's results:

Table 4.12 shows the confusion matrix for MobileNetV1 on the test dataset after augmentation. The model correctly identifies 29 instances of akiec, though it misclassifies several cases as bkl and nv. The model achieves 40 correct predictions for bcc cases, yet there is still notable confusion with nv and other classes. It accurately predicts 89 instances of bkl, but some are incorrectly classified as nv and bcc. For df class, the model correctly identifies 7 out of 10 instances, despite occasional confusion with other lesion types. The model predicts nv, with 650 correct identifications out of 654 cases. However, it struggles with misclassifications involving bkl, bcc, and akiec. While 58 melanoma lesions are correctly identified, many are misclassified as nv and bkl. The classification of vasc lesions is highly accurate, with 8 out of 10 instances correctly identified. Overall, MobileNet demonstrates a high accuracy rate of 99.38% for nv lesions but struggles significantly with mel lesions, achieving a detection rate of only 44.27%.

Table 4.13 displays the confusion matrix results for MobileNetV2 on the test dataset after data augmentation. The model accurately identifies 33 instances of akiec but misclassifies some as bkl and nv. It correctly predicts 34 bcc cases, though confusion with other classes like nv and akiec persists. The model predicts 86 instances of bkl accurately but misclassifies some as akiec and nv. The model achieves a high accuracy rate for df, correctly identifying 8 out of 10 cases, though it occasionally confuses other lesion types. MobileNetV2 is proficient in predicting nv, with 643 correct identifications out of 654 instances, but faces challenges with misclassifications involving bkl, mel, and akiec. It correctly identifies 80 mel instances but often misclassifies these as nv and bkl. The classification of vasc lesions is accurate, with 8 out

					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
-	akiec	29	2	4	0	2	0	0
abe	bcc	0	40	4	0	6	1	0
le L	bkl	0	2	89	1	15	1	0
Tru	df	0	1	1	7	1	0	0
	nv	1	1	2	0	650	0	0
	mel	3	3	20	0	45	58	2
	vasc	0	0	0	0	2	0	8

Table 4.12: Confusion matrix of MobileNetV1 on the test dataset after data augmentation.

of 10 instances correctly identified.

Table 4.14 indicates the confusion matrix of MobileNetV3 on the test dataset with data augmentation. The model correctly identifies 29 instances of akiec, though some are misclassified as bkl and nv. It achieves 46 correct predictions for bcc, yet confusion with other classes persists. The model accurately predicts 84 instances of bkl, but some are misclassified as nv, bcc, and akiec. It demonstrates a high accuracy rate for df, correctly classifying 9 out of 10 instances despite occasional confusion with other lesion types. MobileNetV3 excels in predicting nv, accurately identifying 644 out of 654 instances, although it faces challenges with misclassifications involving bkl, mel, and bcc. The model correctly identifies 92 instances of mel but frequently misclassifies these as nv and bkl. The classification of vasc lesions is highly accurate, with all 10 cases correctly identified, achieving 100% accu-

		Predicted Label						
		akiec	bcc	bkl	df	nv	mel	vasc
True Label	akiec	33	0	2	0	2	0	0
	bcc	5	34	3	0	7	1	1
	bkl	4	0	86	0	15	3	0
	df	0	0	1	8	1	0	0
	nv	2	1	5	0	643	2	1
	mel	1	1	18	0	31	80	0
	vasc	0	0	1	0	1	0	8

Table 4.13: Confusion matrix of MobileNetV2 on the test dataset after data augmentation.

racy for this lesion type. In summary, MobileNetV3 performs exceptionally well in detecting vasc and nv lesions, with a high accuracy rate of 98.47% for nv. However, it struggles with the mel lesion family, achieving a detection rate of only 70.23%.

Table 4.15 represents the confusion matrix for VGG16 on the test dataset with data augmentation. The model identifies 30 instances of akiec but misclassifies some as mel and nv. It achieves 41 correct predictions for bcc, yet confusion with other classes remains an issue. The model accurately predicts 81 instances of bkl but misclassifies some as nv, mel, bcc, and akiec. It shows a high accuracy rate for df, correctly classifying 9 out of 10 instances despite occasional confusion with other lesion types. VGG16 is proficient in predicting nv, accurately identifying 638 out of 654 cases, but faces challenges with misclassifications involving mel, bkl, and bcc. It correctly identifies 100

		Predicted Label						
		akiec	bcc	bkl	df	nv	mel	vasc
True Label	akiec	29	2	2	0	2	2	0
	bcc	2	46	1	0	1	1	0
	bkl	3	3	84	1	12	5	0
	df	0	1	0	9	0	0	0
	nv	0	2	3	0	644	5	0
	mel	1	1	11	0	26	92	0
	vasc	0	0	0	0	0	0	10

Table 4.14: Confusion matrix of MobileNetV3 on the test dataset after data augmentation.

mel instances but often misclassifies them as nv and bkl. The classification of vasc lesions is highly accurate, with all ten instances correctly identified, achieving 100% accuracy for this lesion type. While VGG16 achieves perfect accuracy in detecting vasc lesions, it shows lower accuracy in detecting bkl lesions.

Table 4.16 represents the confusion matrix for VGG19 on the test dataset with data augmentation. The model correctly identifies 35 instances of akiec but misclassifies some cases as bkl and nv. It achieves 38 correct predictions for bcc, though confusion with other classes persists. The model accurately predicts 89 instances of bkl but misclassifies some as nv, mel, and akiec. It shows a high accuracy rate for df, correctly classifying 8 out of 10 instances despite occasional confusion with other lesion types. VGG19 is proficient in predicting nv, accurately identifying 637 out of 654 cases, but faces chal-

		Predicted Label							
		akiec	bcc	bkl	df	nv	mel	vasc	
True Label	akiec	30	1	1	0	2	3	0	
	bcc	4	41	1	0	2	3	0	Ī
	bkl	3	2	81	1	9	12	0	Ī
	df	0	0	1	9	0	0	0	
	nv	0	2	3	1	638	10	0	
	mel	2	0	4	0	25	100	0	
	vasc	0	0	0	0	0	0	10	

Table 4.15: Confusion matrix of VGG16 on the test dataset after data augmentation.

lenges with misclassifications involving mel, bkl, akiec, and bcc. It correctly identifies 90 mel instances but frequently misclassifies them as nv and bkl. The classification of vasc lesions is highly accurate, with all ten instances correctly identified, achieving 100% accuracy for this lesion type. While VGG19 excels in detecting vasc lesions, it faces significant challenges in accurately detecting the mel skin lesion family.

Table 4.17 shows the confusion matrix for ResNet50 on the test dataset with data augmentation. The model successfully identifies 32 instances of akiec but misclassifies some as bkl and nv. It achieves 43 correct predictions for bcc, but confusion with other classes remains an issue. The model accurately predicts 88 instances of bkl but misclassifies some as nv and mel. It demonstrates a high accuracy rate for df, correctly classifying 9 out of 10 instances despite occasional confusion with other lesion types. ResNet50 is

		Predicted Label						
		akiec	bcc	bkl	df	nv	mel	vasc
True Label	akiec	35	0	1	0	1	0	0
	bcc	4	38	2	0	6	1	0
	bkl	3	1	89	0	9	6	0
	df	1	0	0	8	1	0	0
	nv	2	2	5	0	637	7	1
	mel	3	0	10	0	26	90	2
	vasc	0	0	0	0	0	0	10

Table 4.16: Confusion matrix of VGG19 on the test dataset after data augmentation.

proficient in predicting nv, accurately identifying 646 out of 654 cases, but faces challenges with misclassifications involving mel, bkl, and bcc. It correctly identifies 89 mel instances but frequently misclassifies them as nv and bkl. The classification of vasc lesions is highly accurate, with all ten instances correctly identified, achieving 100% accuracy for this lesion type. ResNet50 achieves the highest accuracy in detecting vasc lesions but struggles significantly with the mel skin lesion family.

In summary, the results confirm the effectiveness of transfer learning in combination with data augmentation for skin lesion detection. While ResNet50 demonstrates acceptable accuracy on both the dev and test datasets, MobileNetV3 is a promising choice for real-world deployment due to its efficient runtime and compatibility with low-power devices. These findings are consistent with the performance metrics presented in Table 4.10 and Table
					Predicted	Label		
		akiec	bcc	bkl	df	nv	mel	vasc
~	akiec	32	1	2	0	2	0	0
abe	bcc	1	43	1	1	2	3	0
eΓ	bkl	1	0	88	1	6	11	1
Tru	df	0	0	0	9	1	0	0
	nv	0	2	2	0	646	3	1
	mel	1	1	10	0	28	89	2
	vasc	0	0	0	0	0	0	10

Table 4.17: Confusion matrix of ResNet50 on the test dataset after data augmentation.

4.11, reinforcing the efficacy of the applied methodologies.

### 4.2 Discussion

The analysis presented in this thesis highlights the effectiveness of transfer learning and data augmentation techniques in improving the performance of deep learning models for skin lesion detection. By leveraging pre-trained models on the HAM10000 dataset, we were able to develop classifiers that achieve high accuracy and F1 scores across multiple skin lesion types. However, the performance of each model varied depending on the specific characteristics of the dataset and the model architecture, as summarized in Table 4.18.

Model	Accuracy	F1 Score	Strengths	Weaknesses
VGG16	90.81%	0.9065	- Strong performance	- Challenges in classi-
			in identifying nv and	fying mel and akiec.
			vasc lesions.	- Some confusion be-
				tween bcc and mel.
VGG19	90.61%	0.9036	- Effective in detecting	- Struggles with mis-
			nv and df.	classification between
				mel and nv.
${ m ResNet50}$	91.61%	0.9132	- Consistently strong	- Challenges with dis-
			across most classes,	tinguishing bkl from
			especially akiec and	mel and mel from nv.
			bcc.	
MobileNetV1	88.01%	0.8686	- Good at predicting	- High misclassifica-
			common classes like	tion rates in df and
			nv.	mel.
MobileNetV2	89.11%	0.8863	- High accuracy in de-	- Issues with distin-
			tecting df and nv.	guishing mel from nv
				and bkl.
MobileNetV3	91.31%	0.9102	- Excels in identifying	- Struggles signifi-
			nv and vasc with high	cantly with distin-
			accuracy.	guishing mel.
				- Misclassifies bkl and
				nv frequently.

Table 4.18: Summary of Model Performance, Strengths, and Weaknesses

As shown in the table, ResNet50 emerged as the most robust model, with an accuracy of 91.61% and an F1 score of 0.9132. Its strengths lie in

its ability to accurately classify akiec and bcc lesions, which are critical for identifying malignant conditions. However, it faces challenges distinguishing bkl from mel and mel from nv, indicating areas where further improvement is needed.

While slightly less accurate at 91.31%, MobileNetV3 demonstrated notable efficiency in runtime and performed exceptionally well in identifying nv and vasc lesions. However, it struggled significantly with mel lesions, highlighting the limitations of this model when dealing with visually similar classes. These findings suggest that while MobileNetV3 is a strong candidate for real-time applications, particularly in resource-constrained environments, further refinement is necessary to enhance its performance in more challenging classification tasks.

VGG16 and VGG19, with accuracies of 90.81% and 90.61%, respectively, also demonstrated strong performance, particularly in identifying nv and vasc lesions. However, both models exhibited difficulties in classifying mel and akiec; in some cases, there was confusion between bcc and mel. This highlights the potential need for advanced augmentation techniques or alternative model architectures to capture the subtle differences between these lesion types better.

While efficient, the MobileNetV1 and MobileNetV2 models showed lower accuracy than their counterparts, particularly in classifying df and mel lesions. Their performance underscores the trade-off between computational efficiency and classification accuracy, particularly in models designed for deployment on devices with limited processing power.

These results align with existing literature, suggesting that deeper models

like ResNet50 perform better on complex classification tasks. In contrast, more lightweight models like MobileNet are better suited for scenarios where computational resources are limited. The findings also emphasize the critical role of dataset characteristics, particularly class imbalance, in influencing model performance.

## Chapter 5

## Conclusion

In conclusion, the results of this study demonstrate the effectiveness of transfer learning and data augmentation in developing high-performance deeplearning models for skin lesion detection. ResNet50 consistently emerged as a top performer, achieving the highest accuracy and F1 scores across most lesion classes, making it a reliable model for clinical applications where accuracy is paramount.

On the other hand, MobileNetV3, with its impressive runtime efficiency, is particularly well-suited for deployment in real-time applications and on resource-constrained devices such as smartphones. However, its struggles with distinguishing mel lesions from other types underscore further refinement, perhaps through more sophisticated data augmentation techniques or by incorporating additional features to enhance its discriminative power.

The limitations of the HAM10000 dataset, including its size, demographic representation, and potential biases related to skin type, highlight the impor-

tance of using diverse and representative datasets in future research. These biases can impact the model's generalizability and effectiveness across different populations, emphasizing the need for careful consideration in both dataset selection and model development. Expanding the dataset to include images from varied populations and skin types and incorporating advanced augmentation techniques, such as Generative Adversarial Networks (GANs), could help address these limitations and improve model generalization.

Future research should also explore integrating domain adaptation techniques to enhance model adaptability across different datasets and real-world scenarios. Additionally, real-world validation through clinical trials and continuous learning systems will be crucial for maintaining the accuracy and relevance of these models over time.

Ultimately, the findings of this study contribute to the ongoing development of accurate and efficient diagnostic tools for skin lesion detection, with significant implications for clinical practice and patient care. By continuing to refine and optimize these models, focusing on diversity and bias mitigation, we can move closer to achieving reliable, real-time diagnostic systems that are both effective and accessible to a broad range of patients.

## Bibliography

- [1] "Skin cancer foundation-melanoma." https://www. skincancer.org/skin-cancer-information/melanoma/ melanoma-warning-signs-and-images/.
- [2] "Skin cancer foundation-bcc." https://www.skincancer.org/ skin-cancer-information/basal-cell-carcinoma/.
- [3] "Skin cancer foundation-scc." https://www.skincancer.org/ skin-cancer-information/squamous-cell-carcinoma/.
- [4] "National cancer institute." https://www.cancer.gov/types/skin/ moles-fact-sheet.
- [5] I. H. Sarker, "Deep cybersecurity: A comprehensive overview from neural network and deep learning perspective," SN Computer Science, vol. 2, p. 154, 2021.
- [6] "Deep learning specialization." https://www.deeplearning.ai/ courses/deep-learning-specialization/.
- [7] O. P. Ogunmolu, X. Gu, S. B. Jiang, and N. R. Gans, "Nonlinear systems identification using deep dynamic neural networks," *ArXiv*, vol. abs/1610.01439, 2016.

- [8] R. N. Toma, M. N. Hasan, A.-A. Nahid, and B. Li, "Electricity theft detection to reduce non-technical loss using support vector machine in smart grid," in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), pp. 1–6, 2019.
- [9] D. Xu, Y. Wang, S. Xu, K. Zhu, N. Zhang, and X. Zhang, "Infrared and visible image fusion with a generative adversarial network and a residual network," *Applied Sciences*, vol. 10, no. 2, 2020.
- [10] "Convolutional neural networks (cnn) architecture explained." https://medium.com/@draj0718/ convolutional-neural-networks-cnn-architectures-explained-716fb197b243.
- [11] M. Loukadakis, J. Cano, and M. F. P. O'Boyle, "Accelerating deep neural networks on low power heterogeneous architectures," 2018.
- [12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.
- [13] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for mobilenetv3," 2019.
- [14] R. Ashraf, S. Afzal, A. U. Rehman, S. Gul, J. Baber, M. Bakhtyar,
  I. Mehmood, O.-Y. Song, and M. Maqsood, "Region-of-interest based transfer learning assisted framework for skin cancer detection," *IEEE Access*, vol. 8, pp. 147858–147871, 2020.
- [15] M. Elgamal, "Automatic skin cancer images classification," International Journal of Advanced Computer Science and Applications, vol. 4, no. 3, 2013.

- [16] W. Gouda, N. U. Sama, G. Al-Waakid, M. Humayun, and N. Z. Jhanjhi, "Detection of skin cancer based on skin lesion images using deep learning," *Healthcare*, vol. 10, no. 7, 2022.
- [17] "World cancer research fund international: Skin cancer statistics." https://www.wcrf.org/cancer-trends/ skin-cancer-statistics/.
- [18] "Skin cancer foundation-melanoma." https://www.skincancer.org/ skin-cancer-information/melanoma/.
- [19] M. Q. Khan, A. Hussain, S. U. Rehman, U. Khan, M. Maqsood, K. Mehmood, and M. A. Khan, "Classification of melanoma and nevus in digital images for diagnosis of skin cancer," *IEEE Access*, vol. 7, pp. 90132–90144, 2019.
- [20] N. Melarkode, K. Srinivasan, S. M. Qaisar, and P. Plawiak, "Aipowered diagnosis of skin cancer: A contemporary review, open challenges and future research directions," *Cancers (Basel)*, vol. 15, no. 4, p. 1183, 2023.
- [21] "Canadian skin cancer foundation: Skin cancer." https://www. canadianskincancerfoundation.com/skin-cancer/?gad=1&gclid= CjwKCAjwjOunBhB4EiwA94JWsNs46V81Hivw4ZUdqzu6VP94k5HWbxWSqcmw\_ OKeAuv5MqsezY3orBoC\_sQQAvD\_BwE.
- [22] "What are basal and squamous cell skin cancers?." https://www.cancer.org/cancer/types/ basal-and-squamous-cell-skin-cancer/about/ what-is-basal-and-squamous-cell.html#:~:text=About%208% 20out%20of%2010,head%2C%20neck%2C%20and%20arms.

- [23] M. Naqvi, S. Q. Gilani, T. Syed, O. Marques, and H.-C. Kim, "Skin cancer detection using deep learning—a review," *Diagnostics*, vol. 13, no. 11, p. 1911, 2023.
- [24] A. Kilic, A. Kilic, A. Kivanc, and A. Sisik, "Biopsy Techniques for Skin Disease and Skin Cancer: A New Approach.," *Journal of Cutaneous* and Aesthetic Surgery, vol. 13, no. 3, pp. 251–254, 2020.
- [25] W. F. Cueva, F. Muñoz, G. Vásquez, and G. Delgado, "Detection of skin cancer "melanoma" through computer vision," in 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), pp. 1–4, 2017.
- [26] R. Marks, "Epidemiology of melanoma," Clinical and Experimental Dermatology, vol. 25, pp. 459–463, 11 2000.
- [27] M. A. Kadampur and S. Al Riyaee, "Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images," *Informatics in Medicine Unlocked*, vol. 18, p. 100282, 2020.
- [28] K. Das, C. J. Cockerell, A. Patil, P. Pietkiewicz, M. Giulini, S. Grabbe, and M. Goldust, "Machine learning and its application in skin cancer," *International Journal of Environmental Research and Public Health*, vol. 18, p. 13409, 2021.
- [29] T. Davenport and R. Kalakota, "The potential for artificial intelligence in healthcare," *Future healthcare journal*, vol. 6, pp. 94–98, 2019.
- [30] X. Du-Harpur, F. Watt, N. Luscombe, and M. Lynch, "What is AI? Applications of artificial intelligence to dermatology," *British Journal* of Dermatology, vol. 183, pp. 423–430, 09 2020.

- [31] A. S. Panayides, A. Amini, N. D. Filipovic, A. Sharma, S. A. Tsaftaris, A. Young, D. Foran, N. Do, S. Golemati, T. Kurc, K. Huang, K. S. Nikita, B. P. Veasey, M. Zervakis, J. H. Saltz, and C. S. Pattichis, "Ai in medical imaging informatics: Current challenges and future directions," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 7, pp. 1837–1857, 2020.
- [32] A. A. Patel, Hands-on unsupervised learning using Python: how to build applied machine learning solutions from unlabeled data. O'Reilly Media, 2019.
- [33] R. Aggarwal, V. Sounderajah, G. Martin, D. S. Ting, A. Karthikesalingam, D. King, H. Ashrafian, and A. Darzi, "Diagnostic accuracy of deep learning in medical imaging: a systematic review and metaanalysis," *NPJ digital medicine*, vol. 4, no. 1, p. 65, 2021.
- [34] M. Pandey, M. Fernandez, F. Gentile, O. Isayev, A. Tropsha, A. C. Stern, and A. Cherkasov, "The transformational role of gpu computing and deep learning in drug discovery," *Nature Machine Intelligence*, vol. 4, no. 3, pp. 211–221, 2022.
- [35] Z. Hu, J. Tang, Z. Wang, K. Zhang, L. Zhang, and Q. Sun, "Deep learning for image-based cancer detection and diagnosisa survey," *Pattern Recognition*, vol. 83, pp. 134–149, 2018.
- [36] M. Dildar, S. Akram, M. Irfan, H. U. Khan, M. Ramzan, A. R. Mahmood, S. A. Alsaiari, A. H. Saeed, M. O. Alraddadi, and M. H. Mahnashi, "Skin cancer detection: A review using deep learning techniques," *International Journal of Environmental Research and Public Health*, vol. 18, no. 10, p. 5479, 2021.

- [37] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, p. 115–118, 2017.
- [38] S. S. Han, I. Park, S. Eun Chang, W. Lim, M. S. Kim, G. H. Park, J. B. Chae, C. H. Huh, and J.-I. Na, "Augmented intelligence dermatology: Deep neural networks empower medical professionals in diagnosing skin cancer and predicting treatment options for 134 skin disorders," *Journal of Investigative Dermatology*, vol. 140, no. 9, pp. 1753–1761, 2020.
- [39] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific Data*, vol. 5, no. 1, 2018.
- [40] "Skin cancer mnist: Ham10000." https://www.kaggle.com/ datasets/kmader/skin-cancer-mnist-ham10000.
- [41] A.-R. Ali, J. Li, S. J. O'Shea, G. Yang, T. Trappenberg, and X. Ye, "A deep learning based approach to skin lesion border extraction with a novel edge detector in dermoscopy images," 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–7, 2019.
- [42] M. Goyal, T. Knackstedt, S. Yan, and S. Hassanpour, "Artificial intelligence-based image classification methods for diagnosis of skin cancer: Challenges and opportunities," *Computers in Biology and Medicine*, vol. 127, p. 104065, 2020.
- [43] J. L. Arroyo and B. G. Zapirain, "Automated detection of melanoma in dermoscopic images," *Series in BioEngineering*, p. 139–192, 2014.
- [44] T. Kanimozhi and D. A. Murthi, "Computer aided melanoma skin cancer detection using artificial neural network classifier," 2016.

- [45] S. Choudhari and S. Biday, "Artificial neural network for skincancer detection," 2014.
- [46] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.
- [47] S. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.
- [48] V. Pomponiu, H. Nejati, and N.-M. Cheung, "Deepmole: Deep neural networks for skin mole lesion classification," in 2016 IEEE International Conference on Image Processing (ICIP), pp. 2623–2627, 2016.
- [49] T. Tanaka and M. D. Voigt., "Decision tree analysis to stratify risk of de novo non-melanoma skin cancer following liver transplantation," *Journal of Cancer Research and Clinical Oncology*, vol. 144, pp. 607– 615, 2018.
- [50] P. L. Quinn, J. B. Oliver, O. M. Mahmoud, and R. J. Chokshi, "Costeffectiveness of sentinel lymph node biopsy for head and neck cutaneous squamous cell carcinoma," *Journal of Surgical Research*, vol. 241, pp. 15–23, 2019.
- [51] T. Saba, M. A. Khan, A. Rehman, and S. L. Marie-Sainte, "Region extraction and classification of skin cancer: A heterogeneous framework of deep cnn features fusion and reduction," *Journal of Medical Systems*, vol. 43, pp. 15–23, 2019.

- [52] K. Melbin and Y. Jacob Vetha Raj, "Integration of modified abcd features and support vector machine for skin lesion types classification," *Multimedia Tools Applications*, vol. 80, no. 6, p. p8909, 2019.
- [53] A. G. Neela, "Implementation of support vector machine for identification of skin cancer," *International Journal of Engineering and Manufacturing*, 2019.
- [54] G. Arora, A. K. Dubey, Z. A. Jaffery, and A. Rocha, "Bag of feature and support vector machine based early diagnosis of skin cancer.," *Journal* of Neural Computing Applications, vol. 34, no. 11, p. p8385, 2022.
- [55] N. Hameed, A. Shabut, and M. A. Hossain, "A computer-aided diagnosis system for classifying prominent skin lesions using machine learning," in 2018 10th Computer Science and Electronic Engineering (CEEC), pp. 186–191, 2018.
- [56] F. Xie, H. Fan, Y. Li, Z. Jiang, R. Meng, and A. Bovik, "Melanoma classification on dermoscopy images using a neural network ensemble model," *IEEE Transactions on Medical Imaging*, vol. 36, no. 3, pp. 849– 858, 2017.
- [57] O. F. Alwan, "Skin cancer images classification using naïve bayes," *Emergent: Journal of Educational Discoveries and Lifelong Learning* (*EJEDL*), vol. 3, p. 19–29, Apr. 2022.
- [58] V. Balaji, S. Suganthi, R. Rajadevi, V. Krishna Kumar, B. Saravana Balaji, and S. Pandiyan, "Skin disease detection and segmentation using dynamic graph cut algorithm and classification through naive bayes classifier," *Measurement*, vol. 163, p. 107922, 2020.

- [59] A. Mobiny, A. Singh, and H. Van Nguyen, "Risk-aware machine learning classifier for skin lesion diagnosis," *Journal of Clinical Medicine*, vol. 8, no. 8, p. 1241, 2019.
- [60] S. Alkhushayni, D. Al-Zaleq, L. Andradi, and P. Flynn, "The application of differing machine learning algorithms and their related performance in detecting skin cancers and melanomas.," *Journal of skin cancer*, vol. 2022, no. 2839162, 2022.
- [61] M. F. Ak, "A comparative analysis of breast cancer detection and diagnosis using data visualization and machine learning applications," *Healthcare (Basel, Switzerland)*, vol. 8, no. 2, p. 111, 2020.
- [62] T. Mazhar, I. Haq, A. Ditta, S. A. H. Mohsan, F. Rehman, I. Zafar, J. A. Gansau, and L. P. W. Goh, "The role of machine learning and deep learning approaches for the detection of skin cancer.," *Healthcare* (*Basel, Switzerland*), vol. 11, no. 3, p. 415, 2023.
- [63] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.
- [64] H. Rashid, M. A. Tanveer, and H. Aqeel Khan, "Skin lesion classification using gan based data augmentation," in 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 916–919, 2019.
- [65] 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 2019 IEEE/CVF Conference on Com-

puter Vision and Pattern Recognition Workshops (CVPRW), pp. 2720–2728, 2019.

- [66] A. Farag, L. Lu, H. R. Roth, J. Liu, E. Turkbey, and R. M. Summers, "A bottom-up approach for pancreas segmentation using cascaded superpixels and (deep) image patch labeling," *IEEE Transactions on Im*age Processing, vol. 26, no. 1, pp. 386–399, 2017.
- [67] D. Divya and T. Ganeshbabu, "Fitness adaptive deer hunting-based region growing and recurrent neural network for melanoma skin cancer detection," *International Journal of Imaging System and Technology*, vol. 30, pp. 731–752, 2020.
- [68] B. Ahmad, M. Usama, T. Ahmad, S. Khatoon, and C. M. Alam, "An ensemble model of convolution and recurrent neural network for skin disease classification.," *International Journal of Imaging Systems and Technology*, vol. 32, pp. 218–229, 2021.
- [69] M. Z. Alom, C. Yakopcic, M. S. Nasrin, T. M. Taha, and V. K. Asari, "Breast cancer classification from histopathological images with inception recurrent residual convolutional neural network.," *Journal of Digital Imaging*, vol. 32, no. 4, pp. 605–617, 2019.
- [70] R. Patil and N. Biradar, "Automated mammogram breast cancer detection using the optimized combination of convolutional and recurrent neural network.," *Evol. Intel*, vol. 14, p. 1459–1474, 2021.
- [71] X. Wu, H. Y. Wang, P. Shi, R. Sun, X. Wang, Z. Luo, F. Zeng, M. S. Lebowitz, W. Y. Lin, J. J. Lu, R. Scherer, O. Price, Z. Wang, J. Zhou, and Y. Wang, "Long short-term memory model a deep learning ap-

proach for medical data with irregularity in cancer predication with tumor markers.," *Computers in biology and medicine*, vol. 144, 2022.

- [72] M. A. Elashiri, A. Rajesh, S. Nath Pandey, S. Kumar Shukla, S. Urooj, and A. Lay-Ekuakille, "Ensemble of weighted deep concatenated features for the skin disease classification model using modified long short term memory," *Biomedical Signal Processing and Control*, vol. 76, p. 103729, 2022.
- [73] L. Gonog and Y. Zhou, "A review: Generative adversarial networks," in 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 505–510, 2019.
- [74] I. J. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," ArXiv, vol. abs/1701.00160, 2016.
- [75] U.-O. Dorj, K. K. Lee, J.-Y. Choi, and M. Lee, "The skin cancer classification using deep convolutional neural network," *Multimedia Tools* and Applications, vol. 77, p. 9909–9924, 2018.
- [76] J. Höhn, A. Hekler, E. Krieghoff-Henning, J. N. Kather, J. S. Utikal, F. Meier, F. F. Gellrich, A. Hauschild, L. French, J. G. Schlager, and et al., "Skin cancer classification using convolutional neural networks with integrated patient data: A systematic review (preprint)," *Journal* of Medical Internet Research, 2020.
- [77] T. Devries and D. Ramachandram, "Skin lesion classification using deep multi-scale convolutional neural networks," ArXiv, vol. abs/1703.01402, 2017.
- [78] D. B. Mendes and N. C. da Silva, "Skin lesions classification

using convolutional neural networks in clinical images," ArXiv, vol. abs/1812.02316, 2018.

- [79] S. S. Chaturvedi, K. Gupta, and P. S. Prasad, Skin Lesion Analyser: An Efficient Seven-Way Multi-class Skin Cancer Classification Using MobileNet, p. 165–176. Springer Singapore, May 2020.
- [80] L. Yu, H. Chen, Q. Dou, J. Qin, and P.-A. Heng, "Automated melanoma recognition in dermoscopy images via very deep residual networks," *IEEE Transactions on Medical Imaging*, vol. 36, no. 4, p. 994–1004, 2017.
- [81] H. Hsin-Wei, W.-Y. H. Benny, L. Chih-Hung, and S. T. Vincent, "Development of a light-weight deep learning model for cloud applications and remote diagnosis of skin cancers," *DERMATOLOGY*, vol. 48, no. 3, p. 310–316, 2020.
- [82] A. Ghadah, G. Walaa, H. Mamoona, and S. Najm, Us, "Melanoma detection using deep learning-based classifications," *Healthcare (Basel)*, vol. 10, no. 12, p. 2481, 2022.
- [83] F. Mohammad and F. Esraa, "On the automatic detection and classification of skin cancer using deep transfer learning," *Sensors (Basel)*, vol. 22, no. 13, p. 4963, 2022.
- [84] M. Roshni Thanka, E. Bijolin Edwin, V. Ebenezer, K. Martin Sagayam, B. Jayakeshav Reddy, H. Günerhan, and H. Emadifar, "A hybrid approach for melanoma classification using ensemble machine learning techniques with deep transfer learning," *Computer Methods and Programs in Biomedicine Update*, vol. 3, p. 100103, 2023.

- [85] L. Umesh, Kumar, S. Sarita, S. Yogesh, Kumar, K. Kuldeep, Singh, V. B. R. K, B, R. M. R. V, V, B. Anupam, B. Anchit, and A. Roobaea, "A precise model for skin cancer diagnosis using hybrid u-net and improved mobilenet-v3 with hyperparameters optimization," *Scientific Reports*, vol. 14, p. 4299, 2024.
- [86] T. Jitendra, V, H. Nachiketa, P. Hemprasad, Y, and D. Tausif, "Skin cancer detection using ensemble of machine learning and deep learning techniques," *Multimedia Tools and Applications*, vol. 82, p. 27501–27524, 2023.
- [87] H. Mehdi, H. Dildar, Z. M. Firas, Muhammad, A. Farhan, A, V. Amirhossein, Noroozi, A. Parvaneh, D. Aso, M. Mazhar, Hussain, and L. Sang, Woong, "A model for skin cancer using combination of ensemble learning and deep learning," *PloS one*, vol. 19, no. 5, 2024.
- [88] M. M. Hossain, M. M. Hossain, M. B. Arefin, F. Akhtar, and J. Blake, "Combining state-of-the-art pre-trained deep learning models: A noble approach for skin cancer detection using max voting ensemble," *Diagnostics*, vol. 14, no. 1, 2024.
- [89] A. W. Harley, "An interactive node-link visualization of convolutional neural networks," in Advances in Visual Computing (G. Bebis, R. Boyle, B. Parvin, D. Koracin, I. Pavlidis, R. Feris, T. McGraw, M. Elendt, R. Kopper, E. Ragan, Z. Ye, and G. Weber, eds.), pp. 867–877, Springer International Publishing, 2015.
- [90] H. Gholamalinezhad and H. Khosravi, "Pooling methods in deep neural networks, a review," ArXiv, vol. abs/2009.07485, 2020.

- [91] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929– 1958, 2014.
- [92] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.
- [93] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.
- [94] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," 2019.
- [95] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, pp. 84 – 90, 2012.
- [96] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2019.
- [97] R. Avenash and P. Viswanath, "Semantic segmentation of satellite images using a modified cnn with hard-swish activation function," in *VISIGRAPP*, 2019.
- [98] I. U. Haq, K. Muhammad, A. Ullah, and S. W. Baik, "Deepstar: Detecting starring characters in movies," *IEEE Access*, vol. 7, pp. 9265– 9272, 2019.
- [99] K. Muhammad, S. Khan, V. Palade, I. Mehmood, and V. H. C. de Albuquerque, "Edge intelligence-assisted smoke detection in foggy"

surveillance environments," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1067–1075, 2020.

- [100] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, and S. W. Baik, "Secure surveillance framework for iot systems using probabilistic image encryption," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3679–3689, 2018.
- [101] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans*actions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345– 1359, 2010.
- [102] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis, and D. Feng, "Early diagnosis of alzheimer's disease with deep learning," in 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), pp. 1015–1018, 2014.
- [103] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3462–3471, 2017.
- [104] F. Chollet et al., "Keras." https://keras.io, 2015.
- [105] S. Abokadr, A. Azman, H. Hamdan, and N. Amelina, "Handling imbalanced data for improved classification performance: Methods and challenges," in 2023 3rd International Conference on Emerging Smart Technologies and Applications (eSmarTA), pp. 1–8, 2023.
- [106] C. Aliferis and G. Simon, Overfitting, Underfitting and General Model Overconfidence and Under-Performance Pitfalls and Best Practices in

Machine Learning and AI, pp. 477–524. Cham: Springer International Publishing, 2024.

[107] D. M. W. Powers, "What the f-measure doesn't measure: Features, flaws, fallacies and fixes," 2019.

# Appendix A

# **Appendix: Accepted Paper**

## Transfer Learning Based Skin Cancer Detection Using Convolutional Neural Networks

Saeid Moradi<sup>1</sup>, Mateen Shaikh<sup>2\*</sup> Department of Mathematics and Statistics Thompson Rivers University Kamloops, Canada \*Corresponding author's email: mshaikh@tru.ca

Abstract— Skin cancer, a global health concern, requires early and accurate detection methods to improve patient outcomes. Despite significant advancements in deep learning, challenges like dataset imbalances and the trade-off between model accuracy and computational efficiency persist. This study introduces a comprehensive analysis of various Convolutional Neural Network (CNN) architectures for skin cancer detection using the HAM10000 dataset, comprising 10,015 dermatoscopic images of seven pigmented lesions. This research addresses class imbalances and enhances model robustness by implementing a data augmentation strategy combined with standard preprocessing techniques, such as image resizing and normalization. Six state-of-the-art CNN models-VGG16, VGG19. ResNet50, MobileNet, MobileNetV2, and MobileNetV3-are systematically evaluated to determine their effectiveness. The findings reveal that ResNet50 achieves the highest accuracy and F1-score, making it reliable for precise diagnosis. At the same time, MobileNetV3 excels in computational efficiency, suggesting its suitability for resourceconstrained environments or real-time applications. This study provides critical insights into the trade-offs between accuracy and efficiency in CNN-based skin cancer detection, offering a practical framework for selecting the appropriate model based on specific application needs.

Keywords—Skin Cancer Detection; CNNs; VGG16; ResNet50; MobileNet, MobileNetV2, MobileNetV3

#### I. INTRODUCTION

Skin cancer stands out as one of the most prevalent forms of cancer in the current decade [1]. It is mainly categorized into two major groups: melanoma and nonmelanoma skin cancer [2]. Based on the World Cancer Research Fund International (WCRFI) report, melanoma is the 17<sup>th</sup> most common cancer worldwide. It is the 13th most common cancer in men and the 15th most common cancer in women. The mortality of melanoma skin cancer around the world in 2020 was 57,043 deaths, where New Zealand, Norway, Montenegro, Slovakia, and Slovenia had the highest number of deaths. The worldwide mortality rate for non-melanoma skin cancer was 63,731 in

2020, whereas Papua New Guinea, Namibia, Mozambique, Zimbabwe, and Angola had the highest mortality rates.

Early detection and accurate diagnosis are pivotal factors in treating skin cancer. Typically, physicians rely on the biopsy method for skin cancer detection, which is often painful, slow, and time-consuming [3]. Studies have indicated that dermatologists exhibit classification performance values of 75% to 84% when diagnosing melanoma, drawing upon their professional experiences [4, 5]. Additionally, globally, there is a shortage of skilled dermatologists in public healthcare systems, exacerbating the challenges in dermatological diagnosis and treatment [6].

This research is motivated by two primary goals. First, to improve the efficiency and accuracy of skin cancer diagnosis by developing an artificial intelligence-based screening system using dermoscopic images of skin lesions. Such a system could aid clinical screening tests, reduce diagnostic errors, and enhance early detection, which is critical for successful treatment. Second, this study aims to address the urgent need for reliable automated skin cancer detection systems, particularly in regions with limited access to dermatology specialists. By evaluating the classification performance of six CNN models and analyzing their training behavior and time requirements, this research provides a comprehensive assessment of AI-based solutions for skin cancer diagnosis. Ultimately, this study seeks to bridge diagnostic gaps, enable timely treatment, improve patient outcomes, and potentially save lives.

Machine learning (ML) is a technique that employs statistical models and algorithms to learn from data progressively, enabling the prediction of characteristics of new samples and the execution of desired tasks [7]. ML's profound impact spans various societal domains, including production lines, healthcare, education, transportation, and food industries [7]. Deep Learning (DL), a subcategory of ML comprising deep neural networks, shares similarities with ML yet operates on a deeper level of complexity.

979-8-3503-6304-3/24/\$31.00 ©2024 IEEE



In recent decades, deep learning has profoundly transformed the field of machine learning. The significant increase in processing power has facilitated remarkable progress in computer vision technologies, notably by developing deep learning models like Convolutional Neural Networks (CNNs) [8]. Deep Learning has been widely successfully applied in a variety of classification problems, such as signal processing and radar systems [9, 10], autonomous vehicles [11, 12], cybersecurity [13, 14], and healthcare [15, 16].

The urgency for early skin cancer detection has intensified, and deep learning has emerged as a powerful tool in this endeavor. Studies have demonstrated that early identification of skin cancer using deep learning improves the performance of human specialists, ultimately leading to a reduction in mortality rates [17]. By incorporating efficient formulations into deep learning techniques, exceptional and state-of-the-art processing and classification accuracy can be achieved [18]. Computer-based technology presents a promising avenue for diagnosing skin cancer symptoms, offering advantages in comfort, cost-effectiveness, and speed [18].

Quality data plays a pivotal role in the performance of machine learning models. Therefore, a diverse and comprehensive collection of dermoscopic images is necessary to assess the effectiveness of computer-based systems for skin cancer diagnosis. The HAM10000 dataset [19] is used in this research. The dataset was gathered from two sources: Cliff Rosendahl's skin cancer practice in Queensland, Australia, and the Dermatology Department of the Medical University of Vienna, Austria. It comprises 10,015 dermatoscopic images obtained from different populations and acquired through various modalities. It includes representative cases of the most significant diagnostic categories for pigmented lesions, such as actinic keratoses and intraepithelial carcinoma (AKIEC), basal cell carcinoma (BCC), benign keratosis-like lesions (BKL), dermatofibroma (DF), melanoma (MEL), melanocytic nevi (NV), and vascular lesions (vasc). Each image is annotated with one of seven skin lesion types. Using dermoscopic images for training and applying AI models involves handling sensitive personal health information. To protect patient identities and prevent unauthorized access, all the images in the dataset are anonymized. The dataset is publicly available through the ISIC archive. HAM10000 dataset has an imbalance where it includes 327 images of AKIEC, 514 images of basal cell carcinomas, 1099 images of benign keratoses, 115 images of dermatofibromas, 6705 images of melanomas, 1113 images of melanocytic nevi, 142 images of vascular skin lesions. The imbalance is one of the significant challenges because classifiers tend to be influenced by the dominant class while neglecting the smaller ones [20].

#### II. LITERATURE REVIEW

Leveraging AI for skin cancer detection has the potential to significantly reduce the need for biopsies and empower patients to conduct self-examinations, facilitating teledermoscopy and decreasing the frequency of medical consultations [21]. However, developing an automatic classification system for skin cancer is challenging due to the complexity and diversity of skin cancer images. Skin lesions can share significant similarities across classes, increasing the risk of misdiagnosis [22]. Even within the same class, variations in color, features, structure, size, and location add to the difficulty of accurate classification [23].

CNNs are among the most powerful and widely used ML techniques for image recognition and categorization [24]. Their architecture typically includes convolutional layers, nonlinear pooling layers, and fully connected layers [25]. Fig. 1 shows the basic architecture of a CNN.



#### Fig. 1 CNN architecture.

Previous studies have demonstrated the effectiveness of CNNs in skin cancer classification. For instance, a study utilizing the HAM10000 dataset employed MobileNet for skin lesion detection, achieving an accuracy of 83% [25]. Another study introduced a fully convolutional residual network (FCRN) with 16 residual blocks for melanoma detection, achieving an accuracy of 85.5% with segmentation and 82.8% without segmentation [26]. Huang et al. developed two deep learning models using DenseNet and EfficientNet, achieving 89.5% accuracy in binary classification on the KCGMH dataset and 85.8% on the HAM10000 dataset [27]. Furthermore, using Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) for image enhancement, coupled with a modified ResNet-50 model, improved classification metrics such as accuracy, precision, recall, and F1-score [28].

Another study aims to accurately classify skin lesions into seven categories using the HAM10000 dataset by leveraging 13 deep transfer learning models. The research emphasizes the importance of early detection in reducing mortality rates. It highlights the potential of AI-based systems to enhance diagnostic accuracy, especially in regions with limited access to dermatological care [29].

Most current state-of-the-art approaches rely on either hybrid models [30, 31] or ensembles of deep learning classifiers [32, 33, 34], which are often too resource-intensive for mobile applications. Developing a practical mobile application requires identifying a deep learning model that balances state-of-the-art performance with lightweight architecture. Therefore, this paper evaluates the performance of six different CNN models and analyzes their training time requirements.

Despite these advancements, several limitations remain. Most studies focus on optimizing model accuracy without addressing the computational complexity, making them less suitable for real-time or mobile applications. Additionally, many approaches do not adequately address class imbalance in datasets, which can lead to biased models that underperform on



minority classes. This study addresses these gaps by evaluating a diverse set of pre-trained CNN models, focusing on accuracy and computational efficiency. Moreover, by fine-tuning these models and analyzing their performance across a balanced dataset, this research aims to develop a practical, scalable solution for skin cancer detection that can be deployed in resource-limited settings.

In this study, we employ pre-trained CNN models and finetune their parameters to address challenges in skin lesion detection. The CNN architectures utilized in this study include VGG16 [35], VGG19 [35], MobileNet [36], MobileNetV2 [37], MobileNetV3 [38], and ResNet [39].

VGG16 and VGG19 are convolutional neural network (CNN) models named for their 16 and 19 weight layers, respectively. A notable advancement of VGG16 over earlier models like AlexNet [40] is its use of multiple  $3\times3$  kernel-sized filters, which replaced larger kernels. The architecture of VGG16 consists of convolutional layers with ReLU activation functions designed for fixed input dimensions of  $224 \times 224 \times 3$  RGB images. Each convolutional layer employs  $3\times3$  filters with a stride of 1 and padding of 1. Following the convolutional layers, VGG16 incorporates three fully connected layers, with the first two containing 4096 channels each and the final layer performing 1000-way classification for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) using a softmax function for output classification.

ResNet is a deep learning model designed for computer vision tasks. It introduced significant advancements in the ILVRSC 2015 competition. One of the primary issues ResNet aims to tackle is the Disappearing/Exploding gradient problem commonly encountered in deeper neural networks.

The critical components of the ResNet architecture include Residual Block, Skip Connection, Stacked Layers, and Global Average Pooling (GAP). Fig.3 shows the architecture of ResNet models compared to VGG16 and plain networks.



Fig. 3 ResNet architecture [39].

MobileNets represent a class of efficient models tailored for mobile and embedded vision tasks. They employ a streamlined architecture that relies on depthwise separable convolutions to construct lightweight deep neural networks [36].

The core concept underlying the MobileNet revolves around utilizing depthwise separable convolutions and disassembling the standard convolution operation into two distinct stages: depthwise convolution and pointwise convolution. The architecture of MobileNet, depicted in Fig. 4, comprises 13 blocks of depthwise separable convolutional layers, as described in the original paper [36].



Fig. 4 MobileNet architecture [36].

MobileNet version 2 (MobileNetV2) significantly advances mobile model performance across various tasks and benchmarks and in different model sizes [37]. The architecture of MobileNetV2 revolves around an inverted residual structure, which diverges from residual models by employing thin bottleneck layers at the input and output of the residual block. Fig. 5 represents the MobileNetV2 architecture.



Fig. 5 MobileNetV2 architecture [37].

MobileNet version 3 (MobileNetV3) introduces complementary search techniques and innovative architectural designs. Tailored specifically for mobile phone CPUs, MobileNetV3 integrates hardware-aware network architecture search (NAS) [38] alongside the NetAdapt algorithm [38]. Fig. 6 indicates the architecture of MobileNetV3.





Compared to MobileNetV2, MobileNetV3 incorporates the Squeeze and Excitation (SE) module, initially introduced in SENet [41], to enhance feature learning. MobileNetV3 replaces the sigmoid activation function in the SE module with the hard-sigmoid function to improve computational efficiency.

With the remarkable advancements in deep learning, transfer learning has become a central element in various computer vision fields, including multimedia [42], surveillance [43], and medical applications [44]. The concept involves leveraging pre-trained models originally trained on non-medical or natural image datasets and then fine-tuning these models with new data to adapt to specific tasks [45]. Transfer learning plays a crucial role in deploying convolutional neural networks for diagnostic imaging tasks such as skin cancer detection [46], Alzheimer's Disease diagnosis [47], and chest X-ray analysis [48].

Fig. 7 illustrates the architectures used in the transfer learning approach. Typically, open-source pre-trained models



are trained on extensive datasets containing numerous classes. Transfer learning allows us to modify pre-trained networks by replacing the top layer with an output layer tailored to our dataset. Depending on the size of our dataset, we can adjust or fine-tune the parameters of the pre-trained models to better suit our specific needs.



Fig. 7 Transfer learning methodology.

#### III. METHODOLOGY

The methodology consists of four main steps: preprocessing, data augmentation, model architecture, and evaluation metrics.

Pre-processing: The initial stage of our analysis involved reading images from the dataset and pre-processing them. The initial size of images is  $600 \times 450 \times 3$ , and we resized images to dimensions of  $244 \times 224 \times 3$ , ensuring compatibility with the convolutional neural network (CNN) architectures employed. Additionally, we use built-in functions in the Keras library in Python for data normalization to enhance the uniformity of their pixel values, thus preparing them for subsequent training procedures.

Data augmentation: The HAM10000 dataset includes an imbalanced distribution of data. Resampling is applied to tackle this problem. The available data is diversified by augmenting the dataset through transformations such as rotation, flipping, scaling, and cropping, helping to address the class imbalance issue and enriching the dataset with variations of existing images.

Model architecture: In our skin lesion detection application, exploring transfer learning utilizing pre-trained models such as VGG16, VGG19, MobileNet, MobileNet V2, MobileNet V3, and ResNet is integral to the project. Fig. 8 represents the model architecture, where we drop the top layer of pre-trained models and add average pooling, dropout, and softmax layers with the number of classes in our dataset.



#### Fig. 8 Model architecture.

Evaluation metrics: Based on the evaluation metrics in the skin cancer image classification domain [49, 50, 51], our assessment will encompass standard metrics, including accuracy and Weighted F1-Score. Additionally, we evaluate

the proposed model based on training time for low-power devices.

#### IV. RESULTS & DISCUSSION

This research was done using an ASUS TUF Gaming A15 system with AMD Ryzen 7 6800H processor information with Radeon Graphics, 3201 Mhz, 8 Core(s), 16 Logical Processor(s), and 16GB of RAM.

#### A. CNN models without data augmentation

First, the results of transfer learning based models without any data augmentation are reported. We proceed with finetuning the parameters of the pre-trained models, focusing on specific layers tailored to each architecture. For MobileNetV1, MobileNetV2, MobileNetV3, VGG16, VGG19, and ResNet50, the fine-tuning of parameters commences from layers 50, 100, 120, 10, 13, and 120 out of a total of 86, 154, 157, 19, 22, and 175 layers, respectively. Table 1 showcases the results, where ResNet50 emerges as the top performer in training and validation accuracy and the F1-score. Furthermore, MobileNetV3 demonstrates rapid training, requiring less than 9 minutes, while achieving a training accuracy of 98.54% and a validation accuracy of 85.79%.

TABLE 1. FINE-TUNED CNNS WITHOUT DATA AUGMENTATION.

Model	Training	Training	Validation	Validation	Run Time
	Accuracy	F1-Score	Accuracy	F1-Score	
VGG16	0.9842	0.8868	0.8427	0.7752	263m 16.5s
VGG19	0.9868	0.8903	0.8306	0.7511	277m 51s
ResNet 50	0.9934	0.9233	0.8639	0.8131	92m 40.1s
Mobile Net	0.9712	0.8561	0.833	0.761	32m 21.2s
Mobile NetV2	0.9823	0.8839	0.8538	0.7782	37m 22.5s
Mobile NetV3	0.9854	0.8871	0.8579	0.7802	8m 44.8s

Examine the confusion matrix of the models on the test data. Fig. 9 illustrates the test dataset's true labels, comprising 1001 images from 7 classes.

	Predicted Label							
	akiec	bcc	bkl	df	nv	mel	vasc	
akiec	37	0	0	0	0	0	0	
bcc	0	51	0	0	0	0	0	
bkl	0	0	108	0	0	0	0	
df	0	0	0	10	0	0	0	
nv	0	0	0	0	654	0	0	
mel	0	0	0	0	0	131	0	
vasc	0	0	0	0	0	0	10	

#### Fig. 9 True labels of the test dataset.

Fig. 10 displays the confusion matrix results for MobileNetV1. The confusion matrix results reveal that MobileNetV1 successfully identifies the nv skin lesion family,

IEEE

achieving 642 correct predictions out of 654 instances. Bkl lesions also show a relatively high number of correct predictions (77). However, the model struggles with the vasc lesion family, which is frequently misclassified. Mel lesions are often mistaken for nv and bkl, with 64 and 30 instances, respectively. Akiec and df also show considerable misclassifications between these classes and others like bcc and nv. This indicates that MobileNetV1 has difficulty distinguishing between these classes.

				Predicted	Label		
	akiec	bcc	bkl	df	nv	mel	vasc
akiec	19	3	9	0	6	0	0
bcc	1	31	9	0	9	1	0
bkl	0	2	77	2	26	1	0
df	0	1	3	4	2	0	0
nv	2	2	7	0	642	1	0
mel	2	1	30	0	64	33	1
vasc	0	0	0	0	3	0	7

Fig. 10 Confusion matrix of MobileNet on the test dataset.

**True Label** 

Fig. 11 illustrates the confusion matrix results for MobileNetV2. The model shows significant misclassification between classes with likely visual similarities. Many akiec instances are misclassified as nv (10 out of 37), and some bcc instances as akiec or nv, indicating overlapping features. While bkl is correctly classified in 71 out of 108 cases, there are substantial errors with nv. The model poorly distinguishes df, with only 4 out of 10 correctly classified. It performs well on nv (628 out of 651), though some are misclassified as mel. The mel class has lower accuracy, with frequent misclassification into bkl and nv. Despite its small size, vasc is mostly correctly classified (6 out of 10).

		Predicted Label								
	akiec	bcc	bkl	df	nv	mel	vasc			
akiec	20	1	5	0	10	1	0			
bcc	6	24	5	0	12	3	1			
bkl	5	0	71	0	28	4	0			
df	1	1	2	4	2	0	0			
nv	3	3	13	0	628	6	1			
mel	2	1	30	0	47	51	0			
vasc	0	0	1	0	3	0	6			

Fig. 11 Confusion matrix of MobileNetV2 on the test dataset.

Figure 12 presents the confusion matrix for MobileNetV3 on the test dataset without augmentation. The model correctly identifies 16 instances of akiec but struggles significantly with misclassifications, particularly confusing akiec with bkl and nv. This confusion suggests that the model might focus on shared features, such as color and texture, which are not distinctive enough for accurate differentiation. Similarly, although the model accurately predicts 36 cases of bcc lesions, it often confuses them with akiec, bkl, nv, and mel, indicating a potential overlap in the feature space of these classes. The model performs well in predicting bkl with 64 correct identifications, but the confusion with nv and mel raises concerns about its ability to differentiate between lesions with subtle variations. The model's strong performance in predicting nv (602 correct identifications) reflects its effectiveness with more distinct classes. Yet, the misclassifications as bkl and mel suggest a need for more refined feature extraction. The high

accuracy in vasc identification (8 correct cases) highlights the model's proficiency with more easily distinguishable lesions.

				Predicted	Label		
	akiec	bcc	bkl	df	nv	mel	vasc
akiec	16	2	10	0	5	4	0
bcc	4	36	4	0	3	3	1
bkl	4	4	64	2	24	10	0
df	1	2	0	6	1	0	0
nv	3	8	14	1	602	26	0
mel	2	1	22	1	40	65	0
vasc	0	1	0	0	1	0	8

Fig. 12 Confusion matrix of MobileNetV3 on the test dataset.

Label

ruel

Figure 13 shows the confusion matrix for VGG16 on the test dataset without augmentation. The model identifies 18 instances of akiec but frequently misclassifies this class as mel and nv. This pattern of errors may stem from the model's struggle to capture the unique characteristics of akiec that distinguish it from other lesions, which is crucial given the clinical significance of akiec. While the model correctly predicts 33 cases for the bcc class, the ongoing confusion with akiec, bkl, nv, and mel suggests that the model might overly rely on shared features, leading to ambiguity. Although the model performs well in predicting bkl with 65 correct identifications, the misclassifications with nv and mel indicate that these classes might share overlapping features that the model is not effectively separating. Df prediction shows minor confusion with other types, possibly due to insufficient distinctive features being learned. The model performs strongly predicting nv (618 correct identifications), but in misclassifications such as bkl, bcc, and mel suggest difficulties differentiating between lesions with similar visual characteristics. The high accuracy in vasc classification (9 out of 10 cases) underscores the model's proficiency with more easily distinguishable classes.

				Predicted	Label		
	akiec	bcc	bkl	df	nv	mel	vasc
akiec	18	2	3	0	5	9	0
bcc	7	33	1	0	5	5	0
bkl	5	3	65	1	16	18	0
df	0	1	1	7	0	1	0
nv	0	5	6	1	618	23	1
mel	3	0	5	0	36	87	0
vasc	0	1	0	0	0	0	9

Fig. 13 Confusion matrix of Vgg16 on the test dataset.

Figure 14 depicts the confusion matrix for VGG19 on the test dataset without augmentation. The model successfully identifies 25 instances of akiec but struggles with misclassifications, particularly confusing akiec with bcc, bkl, and nv. This suggests that the model may be focusing on features that are not distinctive enough, leading to errors in classification, which is particularly concerning for a class as clinically crucial as akiec. While the model achieves an accuracy rate of 30 cases for the bcc class, the confusion with akiec, bkl, nv, and mel indicates that subtle visual similarities among these classes may challenge the model. The model performs strongly in predicting bkl with 75 correct identifications. Still, the misclassifications with nv and mel suggest that the model might benefit from more refined feature extraction or additional training data that better highlights the



True Labe

distinctions between these classes. While the model demonstrates proficiency in predicting nv (614 correct identifications), the persistent misclassifications as bkl, bcc, and mel highlight a need for more effective differentiation of these classes. The accurate classification of all 10 vasc cases showcases the model's strength with more visually distinct lesions.

				Predicted	Label		
	akiec	bcc	bkl	df	nv	mel	vasc
akiec	25	2	7	0	3	0	0
bcc	7	30	3	0	10	1	0
bkl	5	2	75	0	15	11	0
df	3	0	1	5	1	0	0
nv	3	5	12	0	614	19	1
mel	5	0	14	0	34	76	2
vasc	0	0	0	0	0	0	10

Fig. 14 Confusion matrix of Vgg19 on the test dataset.

Figure 15 presents the confusion matrix for ResNet50 on the test dataset without augmentation. The model correctly identifies 16 instances of akiec, but significant misclassifications with bkl and nv are observed. These errors suggest that the model may struggle to distinguish between akiec and other lesions with overlapping features, which could be problematic in clinical settings where accurate identification of akiec is critical. While the model achieves an accuracy rate of 28 cases for the bcc class, confusion with other classes remains, indicating that the model may need more discriminative features to improve its accuracy. The model performs well in predicting bkl with 63 correct identifications. Still, the confusion with nv and mel suggests that the model may not adequately capture the subtle differences between these classes. Df prediction shows moderate performance with 6 correct cases out of 10, indicating potential challenges in differentiating this class from others, possibly due to limited training data or insufficient feature representation. The model excels in predicting nv with 630 correct identifications, but the misclassifications as bkl, bcc, vasc, and mel highlight areas where the model could benefit from further refinement. The high accuracy in vasc classification (9 out of 10 cases) reflects the model's strength in identifying more distinct lesions.

	akiec	bcc	bkl	df	nv	mel	vasc
akiec	16	2	7	2	9	1	0
bcc	3	28	3	3	4	8	2
bkl	2	0	63	2	18	21	2
df	0	0	1	6	3	0	0
nv	0	5	5	1	630	9	4
mel	2	1	14	0	45	65	4
vasc	0	0	0	0	0	1	9

Predicted Label

Fig. 15 Confusion matrix of ResNet50 on the test dataset.

In summary, the findings from Table 1 demonstrate the effectiveness of transfer learning in utilizing pre-trained models for skin lesion detection, even when working with imbalanced datasets like HAM10000. However, the confusion matrices reveal critical insights into how this dataset imbalance exacerbates classification difficulties, particularly for less-represented classes. The results indicate that models perform significantly better in classes with more abundant training data.

This underscores the need for a balanced dataset to achieve optimal classification accuracy across all lesion types. This highlights the importance of addressing dataset imbalance through data augmentation, re-sampling, or advanced loss functions to mitigate the bias toward majority classes and improve overall model performance.

#### B. CNN models with data augmentation

After assessing the effectiveness of transfer learning in using pre-trained models across various image types and finetuning the weights based on our dataset, we try to address the imbalance issue in the HAM10000 dataset. To improve our results, we employ transfer learning again, incorporating the data augmentation concept. This involves techniques such as random flipping, rotation, adjustment of brightness and contrast, and cropping of images within the dataset to generate additional data instances.

After data augmentation, each class in the training dataset has been augmented to contain 2000 samples, resulting in a balanced training dataset.

Table 2 showcases the outcomes obtained through transfer learning utilizing pre-trained CNN models, where the weights are trained based on the HAM10000 dataset with data augmentation.

	TABLE 2. FIN	E-TUNED CN	NS WITH DAT.	A AUGMENTAT	ION
Model	Training	Training	Validation	Validation	Run Time
	Accuracy	F1-Score	Accuracy	F1-Score	
VGG16	0. 9936	0.9896	0.9106	0.9052	327m 23.1s
VGG19	0.9944	0.9902	0.9092	0.9064	430m 20.8s
ResNet 50	0.9989	0.9957	0.9231	0.9198	133m 28.1s
Mobile Net	0.9949	0.9815	0.9011	0.8981	48m 11.9s
Mobile NetV2	0.9959	0.9911	0.9161	0.9131	38m 9s
Mobile NetV3	0.9951	0.9903	0.9155	0.9112	13m 27.9s

After fine-tuning with augmented data, all methods exhibited commendable accuracy and F1-score performance. ResNet50, once again, emerged as a top performer, achieving 99.89% accuracy on the training dataset and 92.31% accuracy on the validation dataset. Following ResNet50, MobileNetV2, MobileNetV3, VGG16, VGG19, and MobileNetV1 demonstrated progressively better accuracy.

Runtime is a crucial metric for gauging the computational costs incurred by the models. Larger networks such as VGG19, VGG16, and ResNet50 incurred significantly higher computational costs. Among them, VGG19, with almost 430 minutes, had the highest training time. Conversely, MobileNet models demonstrated notable efficiency in terms of computational costs. Among these, MobileNetV3 stood out



True Labe

True Label

126

with a training time of less than 13 minutes, making it an optimal choice for resource-constrained devices like smartphones.

The confusion matrices for the models on the test data after data augmentation provide valuable insights into their performance.

Fig. 16 shows the confusion matrix for MobileNet on the test dataset after data augmentation, highlighting the model's strengths and weaknesses. While MobileNet accurately identifies 29 instances of akiec, it struggles with misclassifications, particularly confusing akiec with bkl and nv. The model correctly predicts 40 bcc cases, but confusion with nv and other classes remains significant. Although 89 instances of bkl are correctly identified, the model misclassifies several as nv and bcc, indicating challenges in distinguishing between these lesion types. The model shows moderate accuracy in classifying df lesions, correctly identifying 7 out of 10 cases, occasional misclassifications suggest room hut for improvement. MobileNet demonstrates a high accuracy rate (99.38%) for nv, successfully identifying 650 out of 654 cases; however, the misclassification of 4 instances, particularly as bkl, underscores the challenges in differentiating between these similar lesion types. Detecting melanoma (mel) is notably problematic, with only 44.27% accuracy, as the model frequently confuses mel with nv and bkl, which could have severe clinical implications. The classification of vasc lesions is commendable, with 8 out of 10 instances correctly identified.

			Predicted Label								
		akiec	bcc	bkl	df	nv	mel	vasc			
	akiec	29	2	4	0	2	0	0			
	bcc	0	40	4	0	6	1	0			
	bkl	0	2	89	1	15	1	0			
	df	0	1	1	7	1	0	0			
	nv	1	1	2	0	650	0	0			
	mel	3	3	20	0	45	58	2			
	vasc	0	0	0	0	2	0	8			
_											

Fig. 16 Confusion matrix of MobileNet on the test dataset after data augmentation.

Fig. 17 represents the confusion matrix for MobileNetV2 on the test dataset with data augmentation, revealing its overall strong performance, particularly with nv lesions, correctly identifying 643 out of 654 instances. However, the model encounters substantial difficulties distinguishing mel and akiec from classes like nv and bkl. This pattern of misclassification highlights the potential visual similarities between these lesions, which are often clinically significant. For instance, the frequent misclassification of mel as nv could lead to severe clinical consequences, emphasizing the need for more nuanced feature extraction or additional augmentation strategies. The results indicate that while MobileNetV2 is proficient in handling well-represented classes, it requires further refinement to improve the differentiation of lesions with overlapping features.

		Predicted Label								
		akiec	bcc	bkl	df	nv	mel	vasc		
	akiec	33	0	2	0	2	0	0		
	bcc	5	34	3	0	7	1	1		
L D	bkl	4	0	86	0	15	3	0		
	df	0	0	1	8	1	0	0		
	nv	2	1	5	0	643	2	1		
	mel	1	1	18	0	31	80	0		
	vasc	0	0	1	0	1	0	8		

Fig. 17 Confusion matrix of MobileNetV2 on the test dataset after data

18 demonstrates the confusion matrix for Fig. MobileNetV3 after data augmentation showcases the model's balanced performance across various classes. While it correctly identifies 29 instances of akiec, it still misclassifies some as bkl and nv, suggesting challenges in distinguishing between these classes. The model's 46 correct predictions for bcc indicate good performance, yet confusion with other classes persists. Although the model accurately predicts 84 instances of bkl, it struggles with misclassifications involving nv, bcc, and akiec, highlighting potential areas for improvement. MobileNetV3 demonstrates high accuracy for df, correctly identifying 9 out of 10 cases. The model excels in nv classification, with 644 out of 654 instances accurately identified, yet it continues to face challenges with bkl, mel, and bcc misclassifications. Melanoma detection shows improvement with 92 instances correctly identified, yet the model frequently misclassifies these as nv and bkl, indicating that subtle distinctions between these lesions remain challenging to capture. The classification of vasc lesions is perfect, with all ten instances correctly identified, achieving 100% accuracy. While MobileNetV3 performs exceptionally well in detecting nv and vasc lesions, its lower accuracy in detecting mel (70.23%) suggests further model refinement to better distinguish between closely related lesions.

	Predicted Label								
	akiec	bcc	bkl	df	nv	mel	vasc		
akiec	29	2	2	0	2	2	0		
bcc	2	46	1	0	1	1	0		
bkl	3	3	84	1	12	5	0		
df	0	1	0	9	0	0	0		
nv	0	2	3	0	644	5	0		
mel	1	1	11	0	26	92	0		
vasc	0	0	0	0	0	0	10		

Fig. 18 Confusion matrix of MobileNetV3 on the test dataset after data

Fig. 19 showcases the confusion matrix for VGG16 on the test dataset, with data augmentation providing a detailed view of the model's performance. VGG16 correctly identifies 30 instances of akiec but struggles with misclassifications, particularly with mel and nv, reflecting the challenges in distinguishing between these visually similar lesions. The model's 41 correct predictions for bcc indicate a solid performance, though misclassifications with other classes persist. The model accurately predicts 81 instances of bkl but also shows significant misclassifications as nv, mel, bcc, and akiec, suggesting that the model may benefit from further refinement in distinguishing between these classes. VGG16 demonstrates high accuracy in df classification, correctly identifying 9 out of 10 instances. The model is proficient in predicting nv, with 638 out of 654 cases correctly classified,

Label

True

but continues to face challenges with misclassifications involving mel, bkl, and bcc. Melanoma detection is relatively strong, with 100 instances correctly identified; however, frequent misclassifications as nv and bkl highlight the need for improved feature extraction. The classification of vasc lesions is perfect, achieving 100% accuracy. While VGG16 excels in detecting vasc lesions, its lower accuracy in detecting bkl lesions indicates the need for targeted improvements in model training.

		Predicted Label								
		akiec	bcc	bkl	df	nv	mel	vasc		
~	akiec	30	1	1	0	2	3	0		
ape	bcc	4	41	1	0	2	3	0		
eL	bkl	3	2	81	1	9	12	0		
Tra	df	0	0	1	9	0	0	0		
	nv	0	2	3	1	638	10	0		
	mel	2	0	4	0	25	100	0		
	vasc	0	0	0	0	0	0	10		

Fig. 19 Confusion matrix of Vgg16 on the test dataset after data

Fig. 20 shows the confusion matrix for VGG19 on the test dataset with data augmentation. The model correctly identifies 35 instances of akiec, yet misclassifications as bkl and nv persist, indicating the challenges in distinguishing these lesions. VGG19 achieves 38 correct predictions for bcc, though confusion with other classes remains an issue. The model accurately predicts 89 instances of bkl but struggles with misclassifications as nv, mel, and akiec, suggesting potential areas for improvement. VGG19 demonstrates a high accuracy rate for df, correctly identifying 8 out of 10 cases, although occasional confusion with other lesion types is observed. The model is proficient in predicting nv, correctly identifying 637 out of 654 instances, yet continues to face challenges with misclassifications involving mel, bkl, akiec, and bcc. Melanoma detection is relatively strong, with 90 cases correctly identified, but frequent misclassifications as nv and bkl highlight the need for enhanced feature differentiation. The classification of vasc lesions is perfect, achieving 100% accuracy. Although VGG19 excels in detecting vasc lesions, it faces significant challenges in accurately detecting the mel skin lesion family, suggesting further model adjustments.

	Predicted Label								
	akiec	bcc	bkl	df	nv	mel	vasc		
akiec	35	0	1	0	1	0	0		
bcc	4	38	2	0	6	1	0		
bkl	3	1	89	0	9	6	0		
df	1	0	0	8	1	0	0		
nv	2	2	5	0	637	7	1		
mel	3	0	10	0	26	90	2		
vasc	0	0	0	0	0	0	10		

Fig. 20 Confusion matrix of Vgg19 on the test dataset after data

Fig. 21 illustrates the confusion matrix for ResNet50 on the test dataset with data augmentation. ResNet50 successfully identifies 32 instances of akiec but misclassifies some as bkl and nv, indicating difficulties distinguishing these lesions. The model achieves 43 correct predictions for bcc, but confusion with other classes remains a challenge. ResNet50 accurately predicts 88 instances of bkl but misclassifies some as nv and mel, suggesting that the model may struggle with subtle

differences between these lesion types. The model demonstrates high accuracy in df classification, correctly identifying 9 out of 10 cases. ResNet50 excels in predicting nv, with 646 out of 654 instances accurately identified, yet faces challenges with misclassifications involving mel, bkl, and bcc. Melanoma detection is strong, with 89 cases correctly identified, but frequent misclassifications as nv and bkl highlight the need for improved model precision. The classification of vasc lesions is perfect, achieving 100% accuracy. While ResNet50 achieves the highest accuracy in detecting vasc lesions, its struggles with the mel skin lesion family underscore the need for targeted refinements to better distinguish between these critical lesion types.

		Predicted Label								
		akiec	bcc	bkl	df	nv	mel	vasc		
e Label	akiec	32	1	2	0	2	0	0		
	bcc	1	43	1	1	2	3	0		
	bkl	1	0	88	1	6	11	1		
2	df	0	0	0	9	1	0	0		
	nv	0	2	2	0	646	3	1		
	mel	1	1	10	0	28	89	2		
	vasc	0	0	0	0	0	0	10		

Fig. 21 Confusion matrix of ResNet50 on the test dataset after data

The results affirm the effectiveness of combining transfer learning with data augmentation for skin lesion detection. ResNet50 achieves substantial accuracy on both the development and test datasets, showcasing its robust performance across various lesion types. However, MobileNetV3 is optimal for real-world deployment, given its efficient runtime and suitability for low-power devices. This aligns with the performance metrics detailed in Tables 1 and 2, which underscore the effectiveness of the applied methodologies and their practical implications for deploying these models in resource-constrained environments.

#### V. LIMITATIONS AND FUTURE RESEARCH DIRECTIONS

The current approach and dataset have some problems. The HAM10000 dataset, while comprehensive, has limitations in size and demographic diversity, predominantly featuring images from specific population groups, which restricts the model's generalizability across broader populations. A notable challenge is the class imbalance, where benign lesions are more prevalent, potentially biasing the model and reducing its accuracy for less common, malignant lesions. Although transfer learning from pre-trained CNNs improves performance, the model may not generalize well to new datasets or real-world scenarios due to the specific features learned from HAM10000. Data augmentation efforts may mitigate overfitting, but the risk remains if the augmented images do not fully represent real-world variability.

Real-world deployment presents additional challenges, including the need for validation across diverse populations, seamless integration into clinical workflows, and adherence to regulatory standards. The potential for false positives or negatives also raises ethical concerns, underscoring the need for interpretable models that clinicians can trust.



Future research should expand and diversify the dataset, incorporate images from varied populations, and explore advanced augmentation techniques such as Generative Adversarial Networks (GANs). Fine-tuning more varied datasets and investigating domain adaptation techniques will be critical for improving adaptability. Real-world validation through clinical trials and the implementation of continuous learning systems will help maintain the model's accuracy and relevance over time.

#### VI. CONCLUSION

Based on the findings presented in this research, transfer learning and data augmentation are effective strategies for improving the performance of deep learning models in skin lesion detection tasks. Using the HAM10000 dataset and pretrained CNN models allowed for the development of classifiers that accurately identify skin lesions.

ResNet50 consistently emerged as a top performer in accuracy and F1-score metrics, demonstrating its adaptability and effectiveness in leveraging pre-trained weights for feature extraction. On the other hand, MobileNetV3 showcased notable efficiency in runtime, making it a viable option for real-time applications and resource-constrained devices.

Incorporating data augmentation techniques further enhanced model performance, particularly in mitigating issues related to dataset imbalance. The models were better equipped to generalize unseen data and improve classification accuracy by generating additional training instances through random transformations.

Overall, the results highlight the importance of thoughtful model selection and optimization strategies in achieving highperformance skin lesion detection systems. The findings have implications for clinical practice, where accurate and efficient diagnostic tools are essential for timely and effective patient care. Future research could investigate additional augmentation techniques like color shifting and explore advanced model architectures like vision transformers. Moreover, enhancing datasets could further improve performance and broaden the applicability of deep learning models in dermatology.

#### ACKNOWLEDGMENT

- Work was conducted on Secwepemc'ul'ecw, the unceded territory of the Secw'epemc. The TRU Kamloops campus operates on the traditional lands of the Tk'eml'ups te Secw'epemc.
- This work acknowledges the support received from the NSERC Discovery grant RGPIN-2018-06787.

#### REFERENCES

[1] R. Ashraf, S. Afzal, A. U. Rehman, S. Gul, J. Baber, M. Bakhtyar, I. Mehmood, O.-Y. Song, and M. Maqsood, "Region-of-interest based transfer learning assisted framework for skin cancer detection," IEEE Access, vol. 8, pp. 147858–147871, 2020.

- [2] M. Elgamal, "Automatic skin cancer images classification," International Journal of Advanced Computer Science and Applications, vol. 4, no. 3, 2013.
- [3] A. Kilic, A. Kilic, A. Kivanc, and A. Sisik, "Biopsy techniques for skin disease and skin cancer: A new approach.," Journal of Cutaneous and Aesthetic Surgery, vol. 13, no. 3, pp. 251–254, 2020.
- [4] W. F. Cueva, F. Mu<sup>\*</sup>noz, G. V'asquez, and G. Delgado, "Detection of skin cancer "melanoma" through computer vision," in 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), pp. 1–4, 2017.
- [5] R. Marks, "Epidemiology of melanoma," Clinical and Experimental Dermatology, vol. 25, pp. 459–463, 2000.
- [6] M. A. Kadampur and S. Al Riyaee, "Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images," Informatics in Medicine Unlocked, vol. 18, p. 100282, 2020.
- [7] T. Davenport and R. Kalakota, "The potential for artificial intelligence in healthcare," Future healthcare journal, vol. 6, pp. 94–98, 2019.
- [8] M. Pandey, M. Fernandez, F. Gentile, O. Isayev, A. Tropsha, A. C. Stern, and A. Cherkasov, "The transformational role of gpu computing and deep learning in drug discovery," Nature Machine Intelligence, vol. 4, no. 3, pp. 211–221, 2022.
- [9] S. Naoumi, A. Bazzi, R. Bomfin and M. Chafii, "Complex Neural Network based Joint AoA and AoD Estimation for Bistatic ISAC," in IEEE Journal of Selected Topics in Signal Processing, doi: 10.1109/JSTSP.2024.3387299.
- [10] M. Delamou, A. Bazzi, M. Chafii and E. M. Amhoud, "Deep Learningbased Estimation for Multitarget Radar Detection," 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), Florence, Italy, 2023, pp. 1-5, doi: 10.1109/VTC2023-Spring57618.2023.10200157.
- [11] K. Yang and L. Liu, "An Improved Deep Reinforcement Learning Algorithm for Path Planning in Unmanned Driving," in IEEE Access, vol. 12, pp. 67935-67944, 2024, doi: 10.1109/ACCESS.2024.3400159.
- [12] D. Chun, J. Choi, H. -J. Lee and H. Kim, "CP-CNN: Computational Parallelization of CNN-Based Object Detectors in Heterogeneous Embedded Systems for Autonomous Driving," in IEEE Access, vol. 11, pp. 52812-52823, 2023, doi: 10.1109/ACCESS.2023.3280552.
- [13] Y. Fang, Y. Zhang and C. Huang, "CyberEyes: Cybersecurity Entity Recognition Model Based on Graph Convolutional Network," in The Computer Journal, vol. 64, no. 8, pp. 1215-1225, Oct. 2020, doi: 10.1093/comjnl/bxaa141.
- [14] D. Arnold, M. Gromov and J. Saniie, "Network Traffic Visualization Coupled With Convolutional Neural Networks for Enhanced IoT Botnet Detection," in IEEE Access, vol. 12, pp. 73547-73560, 2024, doi: 10.1109/ACCESS.2024.3404270.
- [15] M. Zaabi, N. Smaoui, H. Derbel, and W. Hariri, "Alzheimer's disease detection using convolutional neural networks and transfer learning based methods," 2020 17th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 2020, pp. 939-943, doi: 10.1109/SSD49366.2020.9364155.
- [16] R. Lahoti, S. K. Vengalil, P. B. Venkategowda, N. Sinha and V. V. Reddy, "Whole Tumor Segmentation from Brain MR images using Multi-view 2D Convolutional Neural Network," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Mexico, 2021, pp. 4111-4114, doi: 10.1109/EMBC46164.2021.9631035.
- [17] Z. Hu, J. Tang, Z. Wang, K. Zhang, L. Zhang, and Q. Sun, "Deep learning for image-based cancer detection and diagnosis: a survey," Pattern Recognition, vol. 83, pp. 134–149, 2018.
- [18] M. Dildar, 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.
- [19] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," Scientific Data, vol. 5, no. 1, 2018.
- [20] S. Abokadr, A. Azman, H. Hamdan, and N. Amelina, "Handling imbalanced data for improved classification performance: Methods and

### IEEE

challenges," in 2023 3rd International Conference on Emerging Smart Technologies and Applications (eSmarTA), pp. 1–8, 2023.

- [21] K. Das, et al., "Machine learning and its application in skin cancer," International Journal of Environmental Research and Public Health, vol. 18, p. 13409, 2021.
- [22] A.-R. Ali, J. Li, S. J. O'Shea, G. Yang, T. Trappenberg, and X. Ye, "A deep learning based approach to skin lesion border extraction with a novel edge detector in dermoscopy images," 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–7, 2019.
- [23] M. Goyal, T. Knackstedt, S. Yan, and S. Hassanpour, "Artificial intelligence-based image classification methods for diagnosis of skin cancer: Challenges and opportunities," Computers in Biology and Medicine, vol. 127, p. 104065, 2020.
- [24] T. Mazhar, et al, "The role of machine learning and deep learning approaches for the detection of skin cancer.," Healthcare (Basel, Switzerland), vol. 11, no. 3, p. 415, 2023.
- [25] S. Chaturvedi, K. Gupta, P.S. Prasad, (2021). Skin Lesion Analyser: An Efficient Seven-Way Multi-class Skin Cancer Classification Using MobileNet. Advances in Intelligent Systems and Computing, vol 1141. Springer, Singapore. https://doi.org/10.1007/978-981-15-3383-9\_15.
- [26] L. Yu, H. Chen, Q. Dou, J. Qin, and P.-A. Heng, "Automated melanoma recognition in dermoscopy images via very deep residual networks," IEEE Transactions on Medical Imaging, vol. 36, no. 4, p. 994–1004, 2017.
- [27] H.-W. Huang, B.W. Hsu, C.-H. Lee, and V.S. Tseng, (2021), Development of a light-weight deep learning model for cloud applications and remote diagnosis of skin cancers. J. Dermatol., 48: 310-316. https://doi.org/10.1111/1346-8138.15683
- [28] G. Alwakid, W. Gouda, M. Humayun, and NU. Sama. "Melanoma Detection Using Deep Learning-Based Classifications," Healthcare (Basel). vol. 10, no. 12, p.2481,2022.
- [29] M Fraiwan, and E. Faouri, "On the Automatic Detection and Classification of Skin Cancer Using Deep Transfer Learning," Sensors (Basel). Vol. 22, no. 13, p.4963, 2022.
- [30] M. Roshni Thanka, et al, "A hybrid approach for melanoma classification using ensemble machine learning techniques with deep transfer learning," Computer Methods and Programs in Biomedicine Update, Volume 3, 2023, <u>https://doi.org/10.1016/j.cmpbup.2023.100103</u>.
- [31] K. Lilhore, et al. "A precise model for skin cancer diagnosis using hybrid U-Net and improved MobileNet-V3 with hyperparameters optimization." Sci Rep, Vol. 14, p. 4299, 2024, <u>https://doi.org/10.1038/s41598-024-54212-8</u>.
- [32] J.V. Tembhurne, et al. "Skin cancer detection using ensemble of machine learning and deep learning techniques," Multimed Tools Appl, Vol. 82, p. 27501–27524, 2023, <u>https://doi.org/10.1007/s11042-023-14697-3</u>.
- [33] M. Hosseinzadeh, et al. "A model for skin cancer using combination of ensemble learning and deep learning." PloS one, vol. 19, 5, 2024, doi:10.1371/journal.pone.0301275.
- [34] Md. Hossain, et al. "Combining State-of-the-Art Pre-Trained Deep Learning Models: A Noble Approach for Skin Cancer Detection Using Max Voting Ensemble" Diagnostics, Vol. 14, no. 1-89, 2024, https://doi.org/10.3390/diagnostics14010089.

- [35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.
- [36] A. G. Howard, et al, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.
- [37] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," 2019.
- [38] A. Howard, et al, "Searching for mobilenetv3," 2019.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.
- [40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, pp. 84 – 90, 2012.
- [41] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2019.
- [42] I. U. Haq, K. Muhammad, A. Ullah, and S. W. Baik, "Deepstar: Detecting starring characters in movies," IEEE Access, vol. 7, pp. 9265– 9272, 2019.
- [43] K. Muhammad, S. Khan, V. Palade, I. Mehmood, and V. H. C. de Albuquerque, "Edge intelligence-assisted smoke detection in foggy surveillance environments," IEEE Transactions on Industrial Informatics, vol. 16, no. 2, pp. 1067–1075, 2020.
- [44] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, and S. W. Baik, "Secure surveillance framework for iot systems using probabilistic image encryption," IEEE Transactions on Industrial Informatics, vol. 14, no. 8, pp. 3679–3689, 2018.
- [45] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010.
- [46] A. Esteva, et al, "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol. 542, no. 7639, p. 115–118, 2017.
- [47] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis, and D. Feng, "Early diagnosis of alzheimer's disease with deep learning," in 2014 IEEE 11<sup>th</sup> International Symposium on Biomedical Imaging (ISBI), pp. 1015– 1018, 2014.
- [48] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3462–3471, 2017.
- [49] L. Yu, H. Chen, Q. Dou, J. Qin, and P.-A. Heng, "Automated melanoma recognition in dermoscopy images via very deep residual networks," IEEE Transactions on Medical Imaging, vol. 36, no. 4, p. 994–1004, 2017.
- [50] F. Xie, H. Fan, Y. Li, Z. Jiang, R. Meng, and A. Bovik, "Melanoma classification on dermoscopy images using a neural network ensemble model," IEEE Transactions on Medical Imaging, vol. 36, no. 3, pp. 849– 858, 2017.
- [51] M. Q. Khan, et al, "Classification of melanoma and nevus in digital images for diagnosis of skin cancer," IEEE Access, vol. 7, pp. 90132– 90144, 2019.

