Classifying Melanoma Images with Ensembles of Deep Convolutional Neural Networks

by

Melina Tziomaka

Submitted

in partial fulfilment of the requirements for the degree of

Master of Artificial Intelligence

at the

UNIVERSITY OF PIRAEUS

March 2021

Author .....Melina..Tziomaka.....

II-MSc “Artificial Intelligence”

March 08, 2021

Certified by..... Ilias..Maglogiannis.....

Professor

Thesis Supervisor

Certified by.....Theodoros..Giannakopoulos. .....

Researcher

Member of Examination Committee

Certified by.....Orestis..Telelis.....

Assistant Professor

Member of Examination Committee

Classifying Melanoma Images with Ensembles of Deep Convolutional Neural Networks

By

Melina Tziomaka

Submitted to the II-MSc "Artificial Intelligence" on March 08, 2021, in partial fulfillment of the
requirements for the MSc degree

Abstract

Malignant melanoma is the deadliest form of skin cancer and is one of the most rapidly increasing cancers in the world. Proper diagnosis of melanoma at an earlier stage is crucial for a high rate of complete cure. Both patient and physician awareness regarding the signs and symptoms of early melanoma remains paramount. Hence, a reliable automatic melanoma screening system would provide a great help for clinicians to detect the malignant skin lesions as early as possible. In the last years, the efficiency of deep learning-based methods increased dramatically and their performances seem to outperform conventional image processing methods in classification tasks.

In this master thesis, the EfficientNet family of convolutional neural networks is utilized and extended for identifying malignant melanoma on a dataset of 58,457 dermoscopic images of pigmented skin lesions. A comparative study of the effects of different training configurations is conducted to reveal what contributes to improve performance, and all trained networks are aggregated with an ensembling strategy to further improve individual results.

The proposed method has been evaluated on the SIIM-ISIC Melanoma Classification 2020 dataset and the best ensemble model achieved 0.9404 area under the ROC curve score on hold out test data.

Thesis Supervisor: Ilias Maglogiannis
Title: Professor

Acknowledgments

First and foremost, I am very grateful to Prof. Ilias Maglogiannis of the Dept. of Digital Systems of the University of Piraeus for supervising my research and providing me the guidance and necessary resources needed for completing this thesis.

I would also like to express my gratitude to my parents and friends for their tremendous support, understanding and encouragement in the past year.

Any opinions, findings, conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the view of the University of Piraeus and Inst. of Informatics and Telecom. of NCSR “Demokritos”.

Contents

List of Figures

List of Tables

1 Introduction

In this section is mentioned the importance of computer aided diagnosis platforms based on deep learning techniques for an early detection of melanoma type skin cancer.

1.1 Motivation

Skin cancer is one of the most common cancers around the world, with the most harmful form of it being melanoma. Melanoma has been ranked at the ninth position among the most common types of cancer [1] and it's estimated that the number of new cases diagnosed in 2021 will increase by 5.8 percent. Approximately 207,390 cases of melanoma will be diagnosed in the U.S only. Of those, 106.110 cases will be noninvasive, restricted to the epidermis and 101.280 cases will be invasive, penetrating the epidermis into the skin's second layer. The number of new invasive melanoma cases diagnosed annually has increased by 44% in the past decade. Stage I melanoma patients treated within 30 to 59 days after diagnosis and stage I melanoma patients treated more than 119 days after diagnosis have 5% and 41% respectively higher risk of dying compared to those treated within 30 days [2]. This indicates the importance of early detection and treatment in order to increase the survival rate of patients with melanoma. One of the dermatologist's most popular imaging techniques is dermoscopy. The structure of the skin lesion becomes more visible for examination by magnifying the affected area. This technique is used by trained physicians and is based on the practitioner's experience [3]. With dermoscopy an expert dermatologist can achieve an average accuracy of 65% - 75% [4]. Accuracies can be further improved by capturing dermoscopic images with a high-resolution camera and a magnifying lens to improve visibility of the skin area affected. With this technological support the accuracy of a skin cancer diagnosis can be improved by an estimated 50% [5]. To automate the process of melanoma detection and change the unsettling situation of skin cancer mortality rate for the better, many efforts have been made for the development of computer aided diagnosis platforms, aiming to assist doctors in their day-to-day clinical routine, by allowing economical and quick access to life-saving diagnoses.

1.2 Deep Learning for Melanoma Detection

Gaining knowledge and actionable insights from complex, high-dimensional and heterogeneous health data remains a challenge. Various types of data have been emerging in modern biomedical research, including electronic health records, imaging, sensor data and text, which are complex, heterogeneous, poorly annotated and generally unstructured. Traditional data mining and statistical learning approaches typically need to first perform feature engineering to obtain effective and more robust features from these data, and then build prediction or clustering models on top of them. There are lots of challenges on both steps in a scenario of complicated data and lack of sufficient domain knowledge. The latest advances in deep learning technologies provide new effective paradigms to obtain end-to-end learning models from complex data.

One of the many medical fields that can benefit from the advantages of deep learning is detection of melanoma type of skin cancer. Utilizing deep learning in skin lesion images can ease the diagnosis of doctors among early detection of melanoma. Doctors can capture an image of the skin lesion and pass it through a deep learning architecture to return an outcome for each specific case immediately, assisting them to diagnose melanoma without lab tests or extra costs.

1.3 Structure of thesis

This thesis is subdivided in the following different topics:

Chapter 1: Introduction refers to the topic, and high-lights the scope and objectives of the present thesis.

Chapter 2: Background and Related Work provides a brief introduction to the field of machine learning and a closer look on a particular set of algorithms called convolutional neural networks. Furthermore, a brief overview of benchmark performances and techniques on deep neural networks in the medical domain are presented and the problem of skin lesion classification is analyzed.

Chapter 3: Methodology presents in detail the convolutional neural network architectures that are going to be used. It further discusses the proposed approach, the preprocessing steps and the performance measures used for the evaluation of the models.

Chapter 4: Experiments and Results contains the experimental setup and results of the selected approaches.

Chapter 5: Conclusion and future work is a summary of the accomplished work as well as a brief exploration of future research opportunities.

Bibliography provides a list of sources referred to this thesis, to further facilitate reader's access to the selected articles and books.

2 Background and Related Work

This section gives a brief introduction to machine learning and specifically deep learning, along with its applications in various medical image processing areas.

2.1 Machine Learning

The thoughts set forward by Alan Turing in the midst of the 20th century as to the ascent of machine learning are increasingly gaining momentum. The Turing Test states that an artificial system can be considered intelligent if it can interact with a human, either in written manner or combined with visual simulations, without the individual being able to understand the nature of the system (be it a machine or an actual human) [6]. Later Arthur Samuel characterized machine learning (ML) as a "field of concentrate that enables PCs to learn without being unequivocally customized" [7]. Be that as it may, ML was at long last characterized by Tom M. Mitchell: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E" [8]. This denotes a program that is able to get better at its task by trying the same task over and over again.

Traditional programming is fundamentally different from machine learning programming. In traditional programming, the program uses data in a logical way, which has been explicitly hard coded by a programmer, to achieve a task. Whereas, machine learning uses data to arrive at a logic that can further be used to predict patterns. There is no fixed logic coded into the program, instead the program is designed in such a way that the algorithm derives the logic based on the data and can evolve further as more data are provided. There are four categories that machine learning algorithms can be split into: supervised, unsupervised, semi-supervised and reinforcement learning.

2.1.1 Supervised Learning

Supervised learning is the most common subbranch of machine learning. The algorithms are designed to learn by example from a dataset, in which each example consists of a set of independent variables (features) paired with a dependent variable (target label). The aim of a supervised learning program is to approximate, through an iterative training process, a mapping function to predict the target variable of each example from its features. During the training process, the data are fed to the machine learning algorithm to predict the label. The error of predicting the correct label is then calculated by a cost function and finally adjustments to the parameters of the algorithm are made in order to minimize this error. The process is then repeated until the value error is adequately reduced. The main types of supervised learning problems include classification and regression problems.

Classification refers to the modeling problem of predicting a discrete class label output for a given example of input features. A model will use the training dataset and will calculate how to best classify the examples of the input data to the specific class labels. However, the training dataset must be sufficiently representative of the problem and have a satisfactory number of examples from each class label. The dataset can be compromised by binary class or multiclass examples. The classification predictive modeling algorithms are evaluated based on their results, with classification accuracy being a popular metric used to evaluate a model based on the predicted class labels. The Support vector machines (SVMs), Logistic Regression, Decision Tree, Naïve Bayes Classifier, K-Nearest Neighbors (KNNs) and Neural Networks are among the most popular classification algorithms.

A regression problem is when the target variable is a real or continuous value that needs to be approximated. The goal of a regression algorithm is to make predictions from data by learning the relationship between features of the data and the observed, continuous-valued response. Some widely used regression machine learning algorithms are Linear Regression, Lasso Regression, Support Vector Regression (SVR) and Multivariate Regression.

2.1.2 Unsupervised Learning

Unsupervised Learning is a machine learning technique in which the users do not need to guide the model with labeled data. To those tasks the target variable is unknown and the model works on its own to discover patterns and information that was previously undetected. The main type of problems that unsupervised machine learning is applied to are clustering problems.

Cluster analysis or clustering is the most commonly used technique of unsupervised learning. The most famous clustering algorithm is the k-means algorithm. It is a centroid-based algorithm used to find data clusters through patterns, such that each cluster has the most closely matched data. Its goal is to find groups in the data, (with the number of groups represented by the variable K), by working iteratively to assign each data point to one of the groups based on the features that are provided. The data points are clustered based on feature similarity by using a squared distance Euclidean formula as measurement. Other categories of clustering algorithms are the density-based, distribution-based and hierarchical-based algorithms.

2.1.3 Semi-Supervised Learning

Semi-supervised learning is an approach that falls between supervised learning and unsupervised learning, by combining a small amount of label data with a large amount of unlabeled data during training. Labeled data are used as insight information to help distinguish different classes present in the dataset. The unlabeled data are then used to find those different classes and possible other ones. This provides the benefits of both supervised and unsupervised learning by producing insights, while avoiding the challenges of gathering a large amount of labeled data, which is demanding and often expensive.

2.1.4 Reinforcement Learning

Apart from being a subfield of machine learning, reinforcement learning is also a general-purpose formalism for automated decision-making and AI. The machine learning models,

called agents are trained to make a sequence of decisions and achieve a goal in an uncertain and potentially complex environment. The computer employs a trial-and-error method to come up with a solution for a problem. To accomplish the objective, the agent receives either rewards or penalties for the actions it performs. Its goal is to maximize the total reward.

Although the designer sets the reward policy, he provides the model no hints or suggestions for how to solve the problem. It's up to the model to figure out how to perform the task to maximize the reward, starting from totally random trials and finishing with sophisticated tactics.

2.2 Deep Learning

This section gives a brief introduction to artificial neural networks and one of its subclasses, the convolutional neural type.

2.2.1 Neural Networks

Artificial neural networks are computational models that structurally and conceptually mimic the elegance of the human biological nervous system. The first and simplest type was the feedforward neural network. It consists of layers of computational units known as neurons, which are interconnected in a feed-forward way. Each of the network's neuron takes a group of weighted inputs, sums them up and applies an activation function to regulate the output. Figure 1 shows a fundamental representation of an artificial neuron. The inputs of a neuron can either be features from a dataset or outputs from a previous layer's neurons.

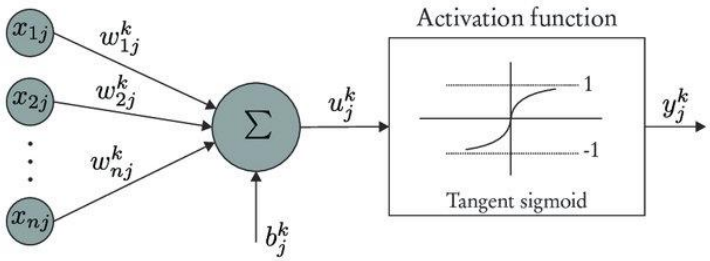

Figure 1: A basic artificial neuron

A key feature of neural networks is the iterative learning process in which data samples are presented to the network and the weights associated with the input values are adjusted each time. After all training samples are presented, the process usually starts over again. During this learning phase, the network adjusts the weights accordingly in order to correctly predict the class label of the input samples. The most popular algorithm to train neural networks is back-propagation [9], and was originally proposed in the 1980's. Perceptrons were the first neural networks that were developed [10]. They are composed of an input layer that is directly connected to an output layer and are capable to classify linearly separable patterns. To solve more complex patterns and capture nonlinear relationships, neural networks were introduced to additional layers, known as the hidden layers. Architectures that consist of multiple hidden layers are known as Deep Neural Networks. Compared to traditional machine learning algorithms that require feature extractors, which are usually designed in a handcrafted manner (based on domain knowledge), deep neural network architectures detect descriptive features from data in a hierarchical way. With an adequately large dataset and after successful training, they can learn all possible mappings and make predictions such as interpolations and extrapolations for unseen instances. Figure 2 shows the architecture of a feed forward neural network.

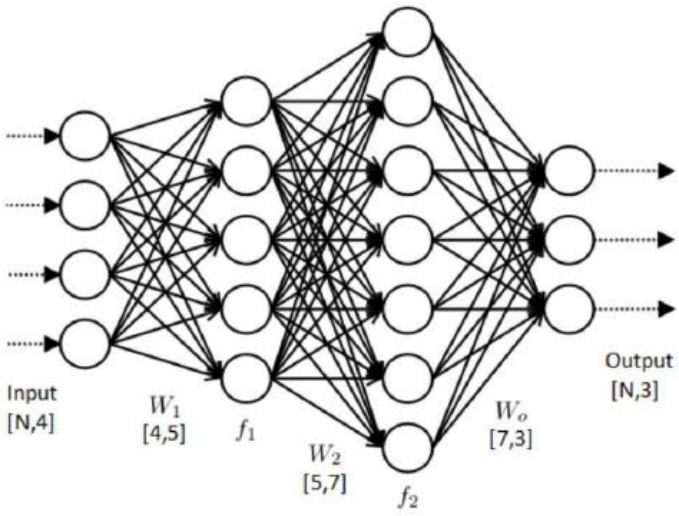

Figure 2: Representation of a neural network

2.2.2 Convolutional Neural Networks

In feedforward neural networks the inputs are in vector form, yet for images the information among neighboring pixels is a different source of data. Convolutional neural networks (CNNs) [11] were designed to exploit spatial configuration types of information in multidimensional regions. A digital image is a representation of visual data in a 2-dimensional form and contains a series of pixel values arranged in a grid-like structure, where each pixel denotes the brightness of a color. A typical convolutional neural network has three types of layers: the convolutional layers, the pooling layers and the fully connected layers.

The convolution layer detects local features at distinct regions of the image by performing a dot product between the set of learnable parameters known as a kernel, and the feature map. A window of pixels is used to connect to each hidden layer neuron, called local receptive field. The concept is for each hidden neuron to learn to analyze its local receptive field. The size of the local receptive field is a tunable hyperparameter that can be adjusted in any CNN architecture. After the first connection the receptive field is moved to scan all the input pixels by a fixed value called stride length. In the end all of the hidden neurons will correspond to a connection of a local receptive field of the input layer. As a result, the activation units of the convolution layer are computed based on the spatially contiguous subsets of the feature map from the previous layer, by convolving the kernels. This implies that if the input is slightly shifted, the activation of the units will be shifted at the same extent. Figure 3 is an illustration of the convolution operation.

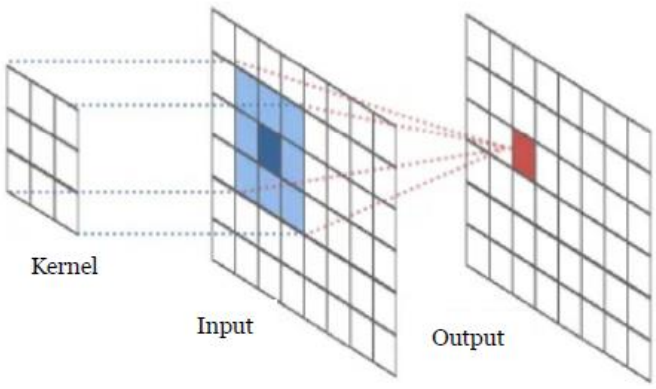

Figure 3: Illustration of the convolution operation

A pooling layer downsamples the feature map by deriving a summary-statistic of the output from the previous convolutional layer. Similar to the convolution layer, each receptive field is represented by a value, *e.g.*, maximum or average, among its units. This aids in dimensionality reduction by scaling down the spatial size of the representation and thus decreasing the amount of computation and weights.

Finally, a flattening operation is performed to convert the data into a 1-dimensional array, which will be passed through the final classification model, which is composed of fully-connected layers. Figure 4 shows the architecture of a convolutional neural network.

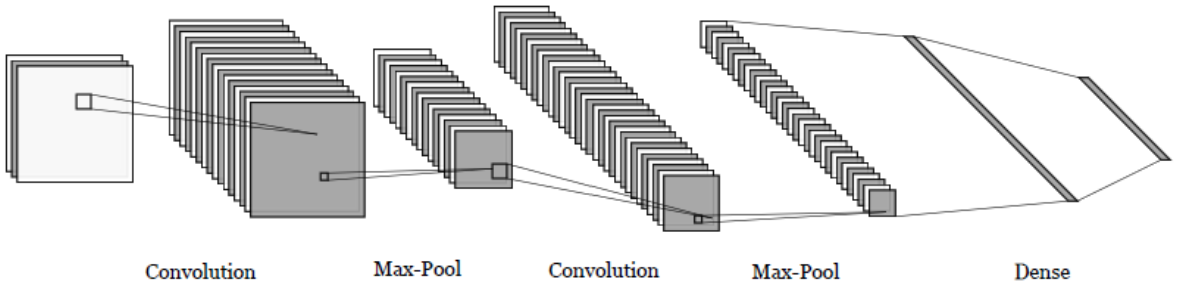

Figure 4: Graphical illustration of a convolution neural network architecture

The first convolutional layer captures the low-level features such as edges and curves, and the subsequent series of convolutional layers generate high-level features. For this reason, the architecture of a CNN should not be limited to one or a few convolutional layers. A network that performs more convolutions to the input data can extract with more precision the features that, according to the dataset, determine the output. Thus, to capture various levels of abstraction, a correspondingly deep architecture of multiple layers is required.

2.3 Deep Learning for Medical Images

This section gives an overview of the existing work in medical imaging using deep learning and mentions some of the basic methods being utilized.

2.3.1 Existing work

The impressive performance of deep learning in computer vision tasks have drawn the attention of researchers to explore their potential in medical imaging, *i.e.* radiological (X-Ray, CT and MRI scans) and pathology images. By allowing the automatic discovery of representations and not relying on problem-specific handcrafted features, which require a certain level of domain knowledge, deep learning has opened new doors in medical image analysis. Rajpurkar et al. [12] trained a modified version of DenseNet [13] with 121 convolutional layers called CheXNet, to classify 14 lung diseases from chest x-rays. CheXNet achieved state-of-the-art results and outperformed individual radiologists. Hosseini-Asl et al. [14] employed an autoencoder architecture of 3-D convolutional neural networks, which was pre-trained to capture anatomical shape variations in structural brain MRI scans. The fully connected upper layers of the 3-D convolutional neural network were then fine-tuned to discriminate scans between patients with Alzheimer's and patients with normal brain. Pratt et al. [15] utilized a deep convolutional neural network and achieved 75% accuracy in classifying diabetic retinopathy images into 5 clinically used classifications of severity.

The application of deep learning in medical imaging is one of the most promising areas of artificial intelligence, and as the number of hospitals and health authorities, that make data publicly available in an anonymized way is increasing, the deep learning research drives ahead. However, it is important to note that these, undeniably powerful techniques should be handled with care and understanding, which is even more imperative in the medical world.

2.3.2 Basic Methods

Deep learning can solve many challenging problems, but requires heavy computing power and a massive amount of data. The availability of data in medical imaging is often limited compared to other areas. Due to the sensitivity of the domain, annotation requires multiple expert opinions on the same data, which makes it expensive and time consuming. Although some deep convolutional neural network methods achieved impressive results in medical image domain, it is still a challenge to fully train deep

models with a limited number of labeled data. [16] Thus, overfitting, which is the modeling error that occurs when a function is closely fit to a limited set of datapoints, has always been a challenge in training deep models with limited examples, compared to their large number of parameters. To overcome these limitations, many transfer learning-based methods have been proposed in medical imaging classification problems.

Transfer learning is a machine learning method where a pre-trained model for a task is reused as the starting point for a new task. As humans inherent ways to transfer knowledge between tasks, by applying it from previous learning experiences to new, the objective of transfer-learning is to transfer knowledge obtained from a large number of labeled data to new conditions. Numerous publicly available deep models have been pre-trained on the ImageNet dataset [17], which consists of over 1.2 million images, and have been applied in many medical imaging classification problems with state-of-the-art results. Among the most popular transfer learning methods is using the pre-trained model for feature extraction. The pre-trained network works as feature extractor, by using the output from the layer prior to the output layer, as input to a new classifier. Another transfer learning technique is finetuning, where the weights of the pre-trained model work as an initialization scheme for the new task. There are various strategies in finetuning a pretrained network with a new dataset, such as training the whole initialized network or freezing some of the parameters and training the rest, usually by updating only the last layers of the network. Alternatively, the pre-trained model or a part of the model's architecture can be integrated to a new neural network and during the training of the new model the weights can again be frozen or updated. The use of the pre-trained models is unlimited and many researches have been conducted for exploiting the advantages of transfer learning in medical imaging.

To further prevent overfitting, regularization techniques like dropout [18] and batch normalization [19] can be utilized during training, to help the model adapt properly to new, previously unseen examples. Dropout refers to the regularization technique, that zeros out the activation values of randomly selected neurons during the training process. This prevents units from co-adapting and forces them to learn more robust features, by not relying on previous units. As a result, the network becomes less sensitive to specific parameters and generalizes better. Batch normalization is the process of subtracting the

batch mean from each activation and dividing the batch standard deviation. This technique has been shown to increase the stability and improve the performance of the neural networks.

Another method that has been also shown to further address the issue, is the pre-processing technique of data augmentation. Flipping, cropping, shifting, zooming, rotation and noise injection are some of the most popular types of image augmentations in general. The important effect of data augmentation is that, it increases the diversity of the data, since the model comes across a different version of the original image. However, a data augmentation method should be applied with caution, in order to preserve the label post-transformation and not affect the resulting classification. For example, in X-Ray images the heart is located on the right side of the body and a horizontal flip augmentation would create a medical condition called situs inversus [20]. Cropping, usually an area from the central part of the image, might also not be a label-preserving transformation for all cases, since information that defines the outcome may be outside the cropped patch. In those cases, rotation augmentation may be a better choice, since the safety is determined by the rotation degree parameter. In general, a combination of the mentioned augmentation methods can be applied. However, the combination of too many geometrical transformations is not guaranteed to be advantageous. It leads to a large number of similar versions from the original images and could result in further overfitting.

Skewed class distributions present another common challenge in effective medical image analysis and is referred to as an imbalance classification problem. Usually, data is collected from various different sources, and not all conditions are as prevalent as others, so the datasets are imbalanced more often than not. When training on an imbalanced dataset, the model tends to *'learn'* more from the dominant classes than the ones with fewer samples. While this is the case, accurately detecting minority class instances in medical imaging is of great importance, as they often relate to high-impact events. For this reason, the final accuracy of the model is not a descriptive measurement of performance when dealing with medical data. If the model's performance is poor on the minority class, but is performing well on the dominant class, the accuracy would still be high. Thus, in classification of health-care data other parameters such as sensitivity

(Recall / True Positive Rate), specificity (True Negative Rate), precision and f-scores are considered to analyze the performance of a trained model. The most effective method to tackle the class imbalance problem is to collect more instances for the minority classes, but for the vast majority of occasions is impossible. In that case, resampling strategies such as undersampling and oversampling, that attempt to rebalance class distribution, can be beneficial. Undersampling otherwise known as downsampling is the process of removing images from the dominant classes and make them comparable with the number of images from the minority classes. This process has the unavoidable consequence of loss of information. It can only be a solution for datasets with a great number of images, from which removing a few examples would not affect the overall performance of the model. Oversampling is the process of adding images to the minority class through random replication or augmentation techniques that were mentioned above. Utilizing a weighted loss function has also proven to provide good results when confronting the class imbalance problem. A weighted loss function refers to the process of penalizing some types of errors more than others. By computing the weights of class frequencies for the loss function, it is ensured that the misclassification of small class instances is penalized more than large-class instances.

2.4 The problem of Skin Lesion Classification

The incidence rate of melanoma worldwide continues to escalate quickly as it has been doing for the past 50 years and the main cause of it is exposure to ultraviolet radiation, where the risk increases drastically with prolonged or intense exposure [21]. However, melanoma that is found early can generally be treated successfully with surgery. Dermoscopy, which is one of the noninvasive skin imaging techniques, has become a key method in the diagnosis of melanoma, where the ABCDE rule was developed to facilitate the differentiation between benign and malignant melanocytic lesions. ABCDE stands for asymmetry, border, color, diameter and evolving. These are the characteristics of skin damage that doctors look for when diagnosing and classifying melanomas [22].

- *Asymmetry* – Melanoma is often asymmetrical, which means the shape isn't uniform. Non-cancerous moles are typically uniform and symmetrical in shape.
- *Border* – Melanoma often has borders that aren't well defined or are irregular in shape, whereas non-cancerous moles usually have smooth, well-defined borders.
- *Color* – Melanoma lesions are often more than one color or shade. Moles that are benign are typically one color.
- *Diameter* – Melanoma growths are normally larger than 6mm in diameter, which is about the diameter of a standard pencil.
- *Evolution* – Melanoma will often change characteristics, such as size, shape or color. Unlike most benign moles, melanoma tends to change over time.

Expert clinicians look for the presence of exclusive visual features to diagnose skin lesions correctly, in almost all of the clinical dermoscopy methods. However, in the case of an inexperienced dermatologist, diagnosis of melanoma can be very challenging and misdiagnosis or underdiagnosis of melanoma is another reason for many skin cancer-related fatalities [23]. The cause of these errors is usually due to the complexity of the subsurface structures and the subjectivity of visual interpretations [24], which indicates the necessity of computer-aided diagnosis (CAD) platforms.

Due to the astounding advancement of image capturing devices over the years, the data is quite large and image quality has been improved, attracting the interest of image analysts in the classification of dermoscopic images. As a result, extensive research with machine learning and computer vision techniques has been done the past decades on the development of CAD systems, that can detect melanoma and help physicians or primary care assistants to minimize the diagnostic errors.

The first approaches were based mostly on feature extraction methods and followed three primary steps: *i)* preprocessing and skin lesion segmentation, *ii)* feature extraction and selection, and *iii)* classification.

Fundamentally, the first step involves preprocessing of the image data, such as image resizing, contrast enhancement, noise reduction and hair removal [25], [26]. After

preprocessing, segmentation of skin lesions, *i.e.* regions of interest (ROIs) is performed, in order to exclude the lesional area from normal surrounding skin, by drawing an accurate border around it. The lesion segmentation literature covers a lot of different methods that can be implemented to tackle the problem, either individually or by combining multiple techniques, to achieve the best results. Some of these researched methods include: probabilistic modelling, active contours, clustering, histogram thresholding, edge detection and graph theory [27].

During the feature extraction process, a set of specific dermoscopic characteristics usually based on the ABCD rule, those visually recognized by expert dermatologists, such as border irregularity [28], [29], [30], asymmetry [31], [32], [33], color [34], [35] and texture [36], [37], [38] is computed from the segmented skin lesion to describe it. Finally, the extracted features from the skin lesion are used as inputs to a feature classification module to classify each skin lesion. Among the most commonly used classifiers for the task are the support vector machines [27], [39], bayesian classifiers [40], decision trees [39] and k-nearest neighbors [39], [40], [41]. Figure 5 shows the main framework of these methods.

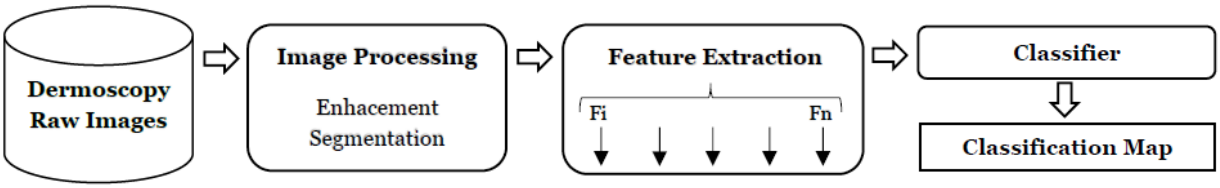

Figure 5: Feature extraction and classification system's framework for dermoscopy image classification

These systems use traditional machine learning techniques, therefore the chosen representation for the image and the quality of the extracted features can heavily affect their performance. Hence, a certain level of expertise is required for the feature extraction of the skin lesions. Deep learning has proven to outperform these methods. In the recent years, deep learning started to be utilized and is becoming the gold standard in melanoma detection and skin lesion classification, employing methods such as deep convolutional neural networks and transfer learning to achieve state-of-the-art results. The feature

extraction process becomes completely automated and depends up to the algorithm to find the more descriptive features of the dataset and train the model properly.

The first breakthrough on skin cancer classification came from Esteva et al [42], who utilized a pre-trained GoogleNet Inception v3 CNN model on a dataset of 129,450 clinical skin cancer images including 3,374 dermatoscopic images. They conducted two validation experiments for checking the performance of the classification rate of their network. The first test consisted of three prediction classes of benign lesions, malignant lesions and non-neoplastic lesions and the second validation test involved nine different classes of skin lesions. The results they observed of the two validation tests were 72.1% ± 0.9% and 55.4% respectively and were compared against certified dermatologists performing the same tasks under the same conditions, who received a peak accuracy of 66% and 55%. The aim of this study was to demonstrate a classification algorithm that is generalizable, and that the performance of their CNN achieved a level of classification competence matching real world expert dermatologists.

The work of Kawahara et al. [43] explored the idea of using a pretrained CNN as a feature extractor rather than training a CNN from scratch. Furthermore, the paper demonstrates that the use of filters from a CNN pretrained on natural images can be generalized into classifying 10 classes of non-dermoscopic skin images.

Liao's [44] work attempted to construct a universal skin disease classification by applying transfer learning on a deep CNN and fine-tuned its weights by continuing the backpropagation.

Y. Li and L. Shen [45] conducted their research utilizing deep learning for the detection of melanomas on a testing set, containing a total of 2000 images of dermoscopic images of different resolutions. Three tasks were performed: Lesion segmentation, feature extraction and classification, achieving accuracies of 92.2%, 91.4% and 85.2% respectively. They used a straight forward convolutional neural network for the feature extraction task, whereas the other two tasks (lesion segmentation and classification) were handled by two fully convolutional residual networks, that made up a deep learning framework.

In 2018 A. Rezvantalab et al [46] developed an algorithm using Support Vector Machines combined with a deep convolutional neural network for multiclass classification of clinical skin cancer images and Codella et al [47] reported new state-of-the-art results by utilizing an ensemble of deep convolutional neural networks to classify the clinical images of 12 skin diseases.

The International Skin Imaging Collaboration (ISIC) [48] has played a significant role in the adoption of new techniques. With the purpose to spread awareness regarding skin cancer and to drive the research in automated skin lesion classification ahead, the community has been providing dermoscopic image datasets with expert annotations, and organizing yearly challenges since 2016, where participants are asked to develop computer vision algorithms for the segmentation and classification of digital skin lesion images.

3 Methodology

In this chapter the proposed method is described in detail for tackling the problem of melanoma detection and the metrics used in the process of evaluating performance are being mentioned.

3.1 Datasets

In this section the dataset that was utilized is being presented along with its proposed method of preparation.

3.1.1 The ISIC Archive

Annually, ISIC makes publicly available new annotated images that add up to the datasets of the previous years, as a result the total number of the ISIC archive has grown significantly over the years, making it the largest publicly available skin lesion image dataset. Thus, for the application of the proposed method for melanoma classification, the datasets that were deployed are: the ISIC2019 Challenge Dataset: 'Skin Lesion Analysis Towards Melanoma Detection' [49], [50], [51], [52] and the ISIC 2020 Challenge Dataset: 'Skin Lesion Analysis Towards Melanoma Detection' [53], [54], which contain dermoscopy images that were collected from the Hospital Clínic de Barcelona, Medical University of Vienna, Memorial Sloan Kettering Cancer Center, Melanoma Institute Australia, University of Queensland, and the University of Athens Medical School.

The ISIC2019 dataset is composed of 25,331 labeled dermoscopy images, and their metadata, which include the site of the skin lesion, and the age and gender of the patient. The labels of the ISIC2019 dermoscopy images are among eight different diagnostic categories. Specifically, the diagnoses present in the dataset are: melanoma, melanocytic nevus, basal cell carcinoma, actinic keratosis, benign keratosis, dermatofibroma, vascular lesion and squamous cell carcinoma.

The ISIC2020 dataset consists of 32,542 benign and 584 malignant skin lesions from over 2,000 patients. The metadata of each image is also provided and include information about: the diagnosis and site of the lesion, the approximate age and gender of the patient,

and an anonymized patient identification number, which allows lesions from the same patient to be mapped together. The goal of the ISIC2020 Challenge is to classify benign and malignant lesions, with the ranking's evaluation metric being the ROC-AUC score. The benign images of the dataset are among 8 types (nevus, seborrheic keratosis, lichenoid keratosis, solar lentigo, lentigo NOS, cafe-au-lait macule, atypical melanocytic proliferation and unknown) and all malignant images are the melanoma type of diagnosis. Notably, no basal cell and squamous cell carcinoma cases are present in the dataset, as a result, this makes it a melanoma detection problem only. For evaluation purposes, a dataset composed of 10,982 unlabeled images along with their metadata, except from the diagnosis feature, is also available from ISIC. The evaluation of algorithms on the unlabeled images is completely automated by Kaggle [55], which hosted the ISIC2020 challenge and provides the ROC-AUC score metric for every submission.

The aforementioned datasets have no common instances, since the ISIC2019 dataset contains images from all the previous year challenges and ISIC2020 contains only the images that were generated by ISIC for the year 2020. Figure 6 shows sample images from the ISIC archive.

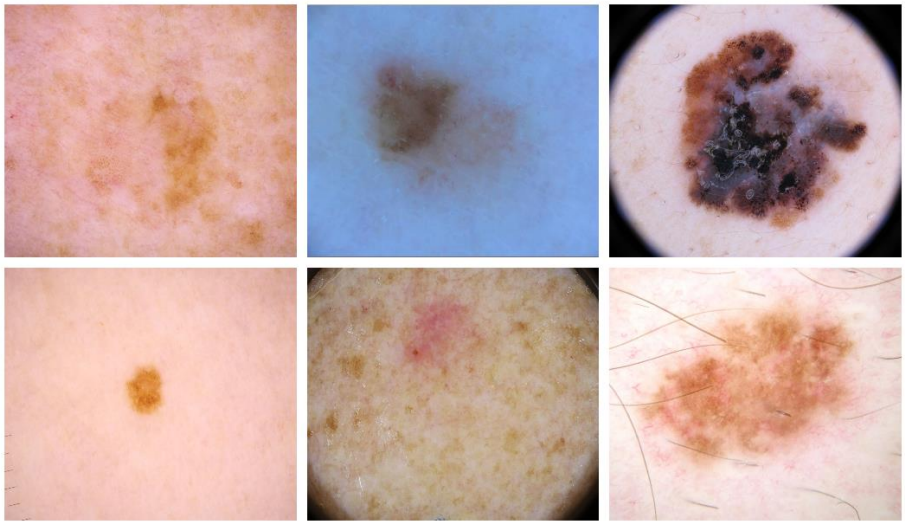

Figure 6: Melanoma (top) and non-melanoma (bottom) sample images from ISIC2019 and ISIC2020

3.1.2 Class Distributions

By calculating the frequency distribution of ISIC2020 dataset, the imbalance classification problem is distinct, as the presence of the positive class, *i.e* melanoma, is very low. The ratio of positive to negative instances is 18:1000 with only 584 positive samples, thus the deployment of last year's dataset was essential. Even though the ISIC2019 dataset is smaller, with 25,331 total images, 22% of its instances belong to the melanoma class, which is 12 times more the positive sample ratio of ISIC2020 data. By adding the melanoma examples to this year's dataset, the class distributions become less imbalanced, with a total of 5,106 melanoma instances and 32,542 non-melanoma instances. However, leaving the non-melanoma examples from ISIC2019 unexploited, results in loss of information. For this reason, all instances from both datasets were included to form the final dataset, with a total of 58,457 labeled dermoscopic images and a positive to negative ratio of 96:1000, which is an improvement from the initial ratio, but the problem of class imbalance is still present. Figure 7 illustrates the class distributions.

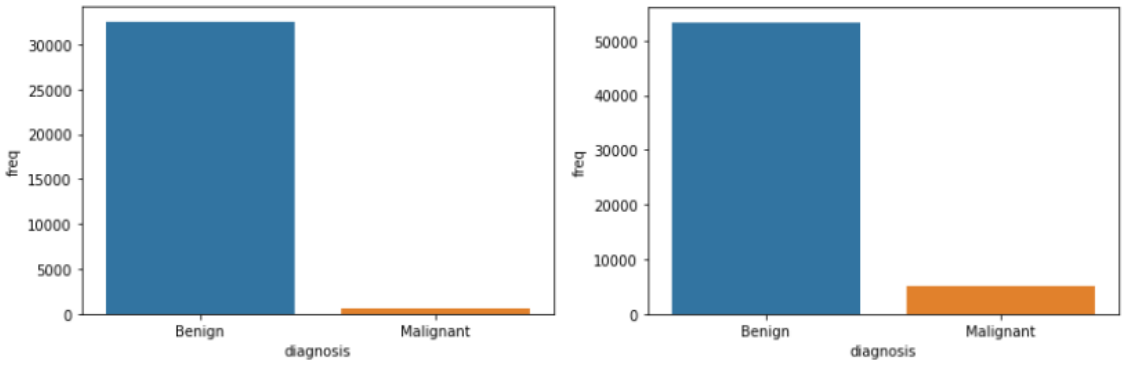

Figure 7: Before (left) and after (right) including the ISIC2019 dataset

3.1.3 Data and Metadata Preparation

The metadata of the ISIC2019 dataset had to be modified in order to be merged with ISIC2020's metadata. The categories, present in the 'anatomy site general' feature, were

mapped accordingly to this year's categories and the feature of 'patient identification number' was added with the value of 'unknown' for all instances.

After merging the metadata, a feature, containing information about the size of each image in bytes was added and the categorical variables of 'biological sex' and 'anatomy site general' were converted to binary vectors. Additionally, the feature 'patient id' was renamed to 'number of images', and was modified to contain the information about the number of all images from that patient of both the dataset and the unlabeled 10,982 images. These transformations were also applied to the unlabeled dataset's metadata, which were considered for the normalization of the numerical features: age, number of images and image size. The final metadata features that resulted after these transformations are: age, biological sex, site head/neck, site lower extremity, site oral/genital, site palms/soles, site torso, site upper extremity, site none, number of images and image size.

Lastly, for validation purposes, the dataset was partitioned into 3 subsets, with 80% of the original data going to the training set, 10% to the validation set and 10% to the test set. Table 1 shows the class frequency distributions of the train set and validation set.

Dataset	***Total***	***Non-Melanoma***	***Melanoma***
Train Set	46,765	42,684	4,081
Validation Set	5,846	5,332	514
Test Set	5,846	5,335	511

Table 1: The frequency distributions of the train, validation and test set

3.2 EfficientNet

The artificial neural networks that are going to be used in this thesis are based on the architecture and main principles of the EfficientNet Convolutional Neural Networks [56].

3.2.1 Compound Scaling

Convolutional Neural Networks are usually developed at a fixed resource budget, and then scaled up for better accuracy if more resources become available. Scaling a convolutional neural network refers to the modification of three architecture dimensions: *depth, width* and *resolution*. The depth refers to the total number of convolutional layers, while the width is related to the number of filters in each convolutional layer. Lastly, the resolution is simply the dimensions of the input image.

As mentioned, by adding more convolutional layers, *i.e.* scaling up the depth, allows the network to learn more complex features, yet deeper networks tend to suffer from vanishing gradients and become difficult to train. Although methods such as batch normalization and skip connections, *i.e.* connections between nodes in different layers, are effective in resolving this issue, the actual accuracy gains by just increasing the depth of the network, quickly saturate. Respectively, by increasing the width of the networks, the layers can learn more fine-grained features. In fact, this approach has been used in numerous works, *e.g.* Wide ResNet. However, as is the case of scaling up the depth, increasing only the width prevents the network from learning complex features, and results in accuracy gains that quickly diminish. Lastly, a higher input resolution provides a greater detail about the image and allows the network to extract finer patterns, but on its own, returns limited accuracy gains as well.

Scaling up by a combination of the three dimensions enhances more the model's predictive capabilities. The reason is that if the spatial resolution of an input image is increased, the number of convolutional layers should also be increased, so that the receptive field is large enough to span the entire image that contains more pixels. Still arbitrary scaling up the dimensions of a network does not guaranty better results, and to balance all three dimensions is a difficult task. In fact, the process often requires many trials to appropriately scale up the dimensions, in order to satisfy the resource constraints.

The concept of EfficientNet is to start with a high-quality and compact model, and to use a compound coefficient φ to uniformly scale all its three dimensions according to:

$$depth = \alpha^{\varphi}$$

$$width = \beta^{\varphi}$$

$$resolution = \gamma^{\varphi} \qquad , \qquad (1)$$

$$s.t.\ \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

where φ is a user-specified coefficient (integer) that controls how many resources are available to scale the model and α, β, γ are constants, determined by a grid search, to indicate how to assign these resources to each of the network's dimensions. The Floating-point Operations Per Second (FLOPS) of a convolution operation are proportional to $depth$, $width^2$, $resolution^2$, so as a result by scaling the network using equation 1 the FLOPS will increase by $(\alpha * \beta^2 * \gamma^2)^{\varphi}$. For this reason, the constraint of $(\alpha * \beta^2 * \gamma^2) \approx 2$ is applied to ensure that the total FLOPS don't exceed 2^{φ}. Figure 3 shows the structure change of a model after scaling each dimension separately and with compound scaling, which is generic and can be used with any architecture to effectively scale it and provide better accuracy.

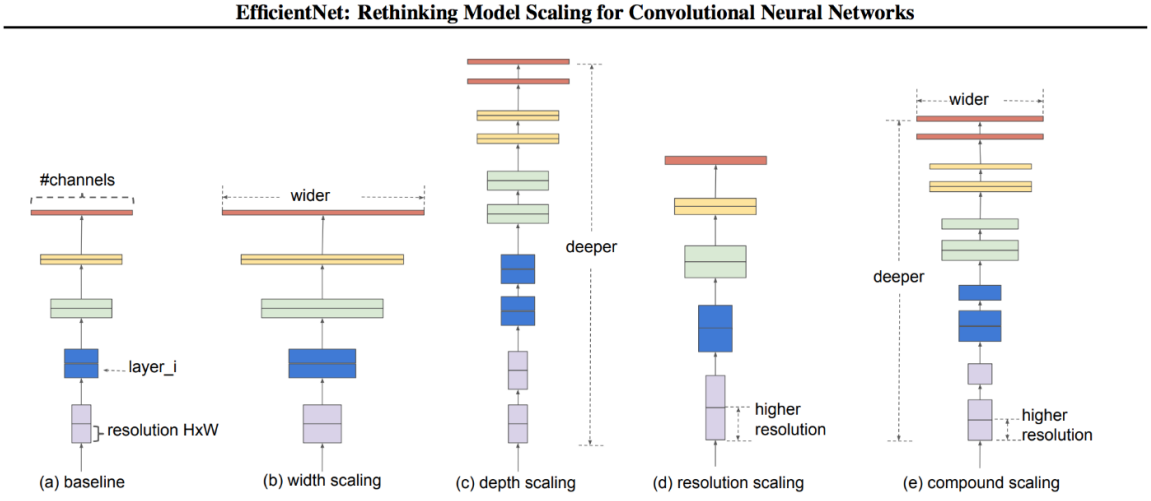

Figure 8: a) The baseline network, b) The network after increasing its width, c) The network after increasing its depth, d) The network that accepts higher resolution images, e) The baseline network that is expanded through compound scaling

3.2.2 EfficientNet-B0

In order for compound scaling to be more effective, a good baseline model is critical. Therefore, the authors of EfficientNet also introduced a new network, called EfficientNet-B0. They were inspired by the construction method used for MnasNet [57], whose architecture was developed using multi-object neural architecture search to optimize accuracy and real-world latency on mobile devices. Neural architecture search (NAS) is a technique that automates the design of artificial neural networks based on a given problem and the methods for NAS can be categorized according to three dimensions [58]:

- The *search space* defines the architectures that can be designed and optimized
- The *search strategy* describes how to explore the search space
- The *performance estimation strategy* evaluates the performance of a possible architecture from its design

To construct the architecture of MnasNet, a factorized hierarchical search space was used that factorizes a convolutional neural network into a series of blocks and uses a hierarchical search space to determine the operations and connections of each block. Along with the tuning of the hierarchical space, the search and performance estimation strategies are tuned in order to minimize the latency. However, since the authors of EfficientNet didn't target any specific hardware device, they utilized the method of multi-object neural architecture search on the ImageNet dataset to optimize accuracy and FLOPS rather than latency. Therefore, EfficientNet-B0, described in table 2, presents a similarity to MnasNet, but is slightly larger.

Stage	Operator	Resolution	#channels	#layers
1	Conv3x3	224x224	32	1
2	MBConv1,k3x3	112x112	16	1
3	MBConv6,k3x3	112x112	24	2
4	MBConv6,k5x5	56x56	40	2
5	MBConv6,k3x3	28x28	80	3
6	MBConv6,k5x5	14x14	112	3
7	MBConv6,k5x5	14x14	192	4
8	MBConv6,k3x3	7x7	320	1
9	Conv1x1/Pooling/FC	7x7	1, 280	1

Table 2: EfficientNet-B0 description

Its main component is known as the Mobile Inverted Bottleneck Conv (MBconv) Block [59] with the depth-wise separable convolution [60], [61]. Unlike the traditional convolution operation, which applies a 2-D depth filter to directly convolve the input in depth as well, depth-wise separable convolution uses each filter channel only at one input channel. Precisely, it breaks the filter and image into three different channels and applies the corresponding filter to the corresponding channel. Finally, it combines the output by applying a pointwise convolution. Figure 9 shows a regular vs a depth-wise separable convolution operation.

The Mobile Inverted Bottleneck Conv Block, flips the classic wide – narrow – wide approach, in which skip connections exist between wide parts of the network, to a narrow – wide – narrow approach with skip connections between narrow parts of the network. The first step is a 1x1 convolution, which increases the depth, then follows a depth-wise convolution, and lastly another 1x1 convolution squeezes the network in order to match the initial number of channels for the skip connection. The Mobile Inverted Bottleneck Conv Block is depicted in Figure 10.

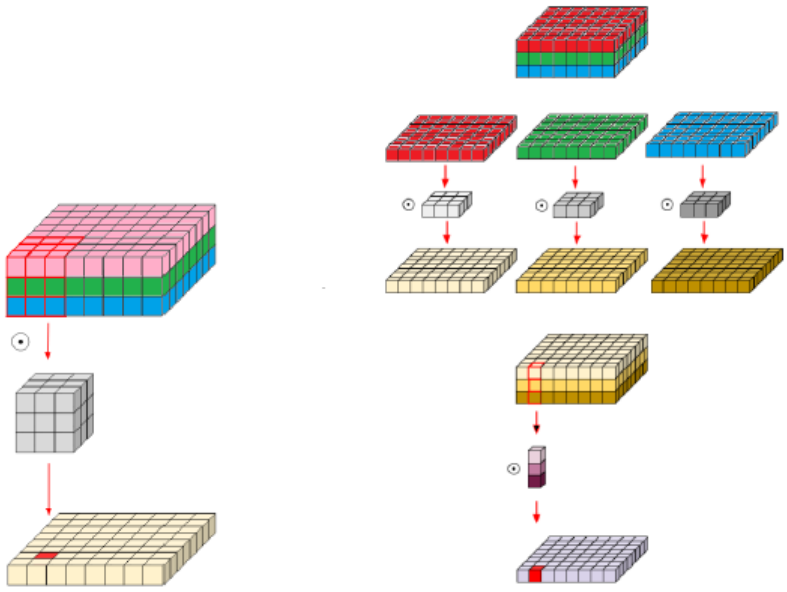

Figure 9: A regular convolution (left) vs a depth-wise separable convolution (right)

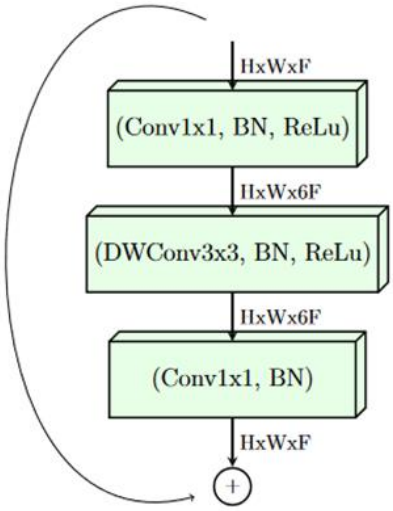

Figure 10: An illustration of the MBConv Block, where DWConv stands for depth-wise convolution and BN for batch normalization

3.2.3 The EfficientNet Family

The compound scaling method was applied to make up the family of EfficientNet. By starting with EfficientNet-B0 as baseline model and by fixing $\varphi = 1$, assuming twice more resources are available, the α, β, γ were determined by a small grid search. Once found, these parameters were fixed and the compound coefficient φ was increased to scale up the baseline model and construct the EfficientNet-B1 through EfficientNet-B7, with the integer in the end of their name indicating the value of compound coefficient.

The EfficientNet models present great results on the ImageNet dataset compared to their competitors, with EfficientNet-B7 achieving state-of-the-art 84.3% top-1 / 97.1% top-5 accuracy. The networks achieved both higher accuracy and better efficiency over existing models, reducing parameter size and FLOPS by an order of magnitude. Figure 11 shows a comparison between the EfficientNet models and other CNN's on the ImageNet dataset. Furthermore, as pretrained models on the ImagetNet dataset, they were also tested on 8 widely used transfer learning datasets, to which they also performed very well, with state-of-the-art results in 5 out of the 8.

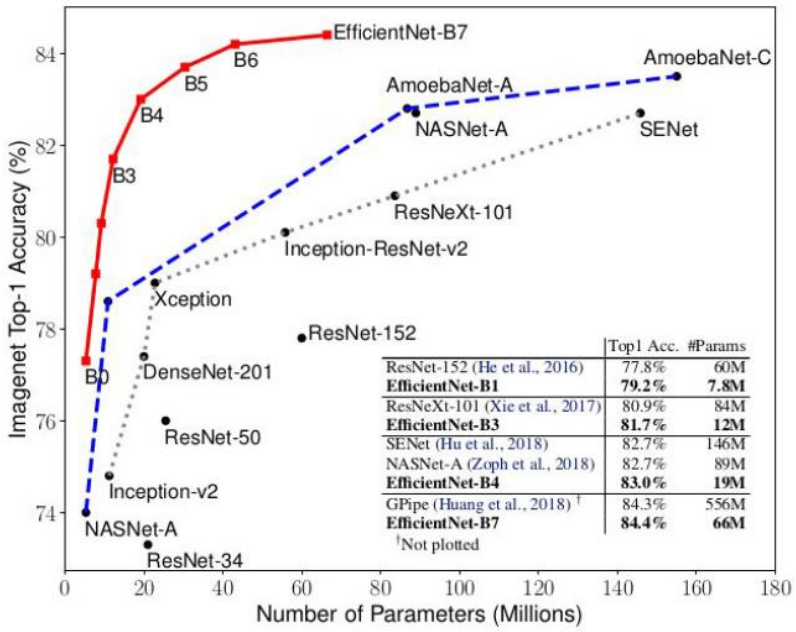

Figure 11: Accuracy results of the EfficientNet family networks on the ImageNet dataset compared to some popular neural network architectures

3.2.4 Swish Activation Function

The activation function used in EfficientNet models is the SiLU (Sigmoid-Weighted Linear Unit) [62], which is a specific version of the Swish activation function [56], [63]:

$$f(x;\beta) = x \cdot \sigma(\beta x) \qquad (2)$$

where $\sigma(x) = (1 + \exp(-x))^{-1}$ is the sigmoid function and β is a learnable parameter. However, most implementations do not use the learnable parameter β, in which case the activation function is simply $f(x) = x \cdot \sigma(x)$ and is referred as SiLU or Swish-1. The swish activation function was proposed as a better alternative to the successful and widely-used ReLU (Rectified Linear Unit): $f(x) = \max(x, 0)$ [64]. The reason ReLU stood out from other traditional activation functions, is its benefit of being unbounded above and therefore reducing the likelihood of vanishing gradients. Notably, ReLU is a non-smooth function that is not derivable at $x = 0$ and maps all negative inputs to zero, whereas for positive inputs, the function is linear. That makes the computation of it's derivative, which is required when updating the weights of a node during backpropagation, very efficient compared to the other functions. However, since the derivative zeroes out for all negative inputs, certain weights can become inactive. This can occur because, whenever there is a negative input into a given neuron, the backpropagated error can be cancelled out. As a result, the gradient will also be zero, which means that there is no way for the related weights to get updated towards the right direction. Such neurons can't contribute in discriminating the input and will become useless. The problem, is known as the dying ReLU problem and mainly impacts the learning process of deep models.

The SiLU activation function preservers ReLU's attributes of being bounded below and unbounded above, while being a smooth and non-monotonic function. In fact, the non-monotonic nature of swish is what sets this function apart from most activation functions. The derivative of SiLU is

$$
\begin{aligned}
f'(x) &= \sigma(x) + x \cdot \sigma(x)(1 - \sigma(x)) \\
&= \sigma(x) + x \cdot \sigma(x) - x \cdot \sigma(x)^2 \qquad (3) \\
&= x \cdot \sigma(x) + \sigma(x)(1 - x \cdot \sigma(x)) \\
&= f(x) + \sigma(x)(1 - f(x))
\end{aligned}
$$

and is also continuous and nonmonotonic. Figure 12 shows SiLU compared to the ReLU. In their work, Ramachandran et al. write that their "extensive experiments show that Swish consistently matches or outperforms ReLU on deep networks applied to a variety of challenging domains such as image classification and machine translation". It is difficult to prove the reason why an activation function outperforms another, but an attempt to explain this behavior can be based on observations. Notably, SiLU is bounded below and very negative weights are zeroed out, therefore benefits from sparsity similar to ReLU. But it is also smooth, so small negative values that may still be relevant for capturing patterns don't get canceled out. Instead, they have a smooth output landscape, which benefits the optimization of the model in terms of convergence towards the minimum loss. Lastly, SiLU preserves ReLU's benefit of being unbounded above and thus for large input values, the outputs don't saturate to the maximum values.

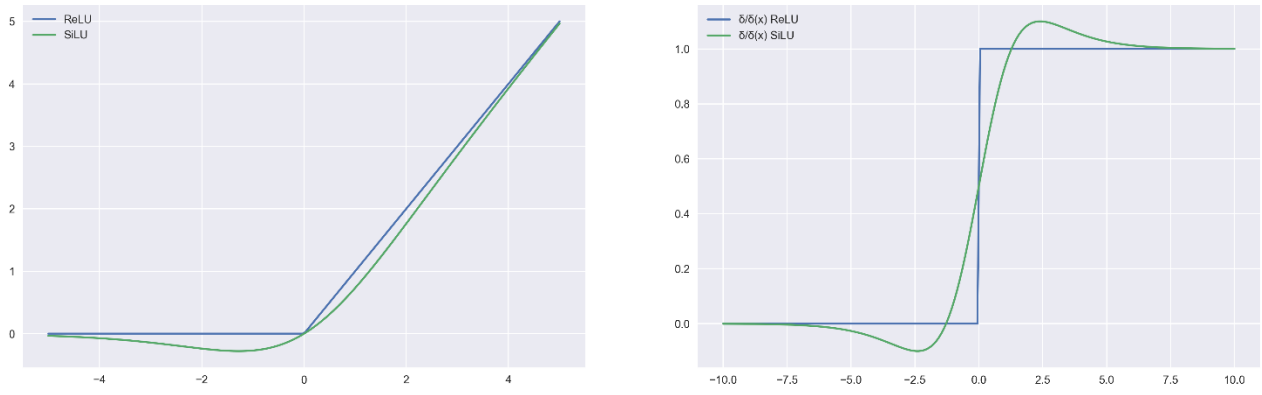

Figure 12: Comparison of ReLU and SiLU activation function. The plots of the activation functions (left), and their first derivatives (right)

3.3 Proposed Models

The EfficientNet family has state-of-the-art models of high performance and low computational cost, thus it was a natural choice. The models that were utilized and explored with the proposed method for melanoma detection are the architectures of the EfficientNet-B3 – B6 and each of the models was deployed with two approaches.

In the first approach, the only modification to the network architectures was the replacement of the final softmax-layer, which is specific for the ImageNet classification task (*i.e.* 1000-dim), by a two-neuron softmax layer to obtain probabilistic output for the melanoma and non-melanoma classes.

In the second approach, the models were deployed to a modified architecture, which takes into consideration the metadata and has a different activation function. The proposed model uses the Mish [65] activation function and has 3 additional layers. The first layer takes as input both the metadata and the output of the default EfficientNet model, and the second layer along with a final 2-neuron softmax layer for the output, perform the final classification task. The techniques of batch normalization (BN) and dropout with a 30% chance were also utilized for the two lower additional layers. The approach is illustrated in Figure 13.

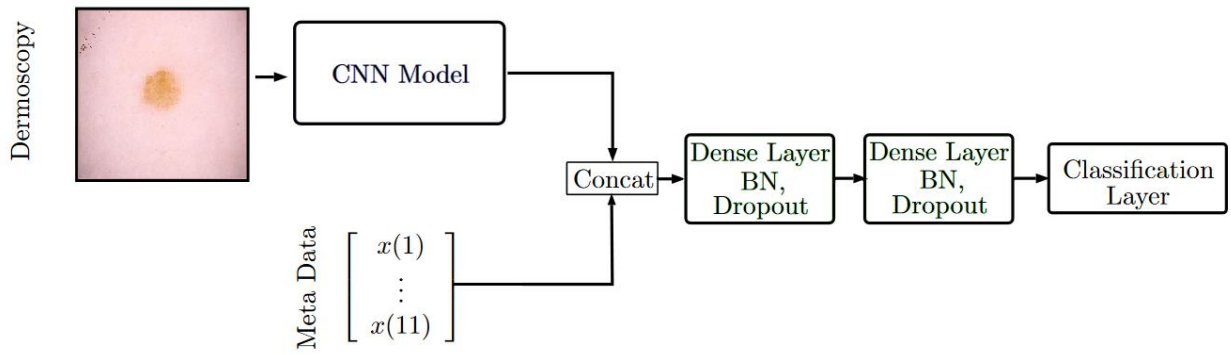

Figure 13: Proposed architecture

The activation function that replaced the default SiLU (Swish-1) is Mish. It is a recent activation function, that was introduced in the paper "Mish: A Self Regularized Non-Monotonic Neural Activation Function" and has outperformed Swish and ReLU in numerous tasks. The Mish activation function is defined as:

$$f(x) = x \cdot \tanh\big(\varsigma(x)\big) \qquad (4)$$

where, $\varsigma(x) = \ln(1 + e^x)$ is the softplus activation function [66]. It presents similarities to SiLU (Swish-1) and ReLU as it is also a smooth non-monotonic function, that is both bounded below and unbounded above. In fact, the graph of Mish is almost identical to Swish. However, the differences of the two functions are more prominent to their derivative graphs. Figure 14 shows the graphs of Mish and Swish and the graphs of their first and second derivatives.

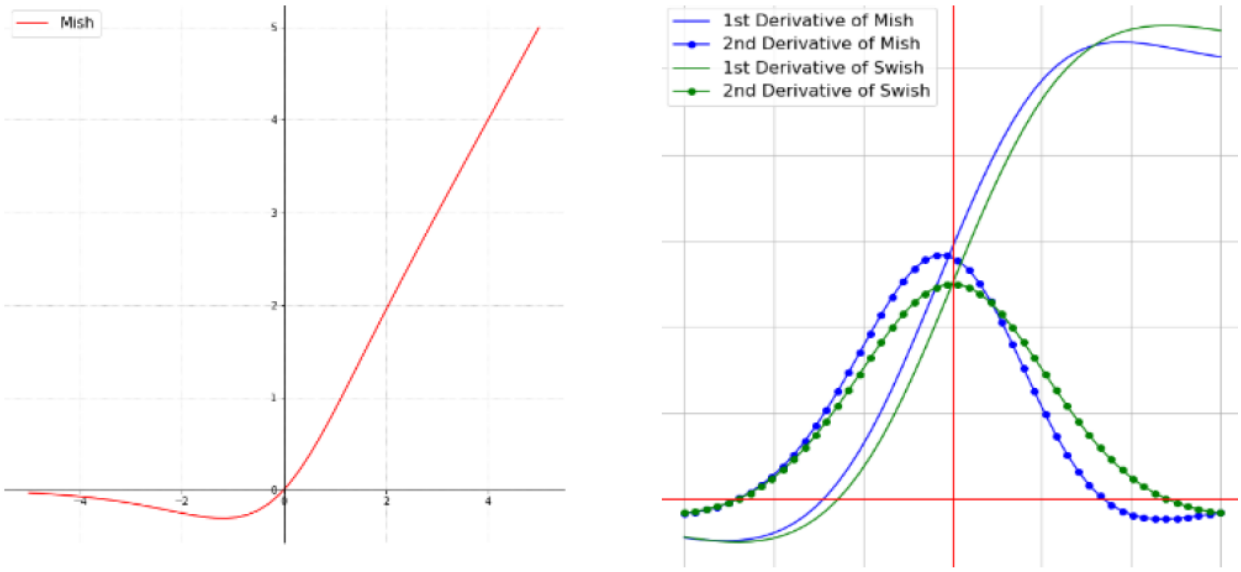

Figure 14: The Mish activation function (left) and the comparison between first- and second- derivative of Mish and Swish (right)

Mish avoids saturation due to near zero gradients, has strong regularization effects, and effective optimization and generalization. Although, Mish has the capability and robustness to improve a neural network's task, there is a certain trade-off of higher training time compared to the other functions.

3.4 Training

This section describes in detail the image preprocessing required to prepare the images, the data augmentation scheme and the hyper-parameter tuning.

3.4.1 Image Pre-Processing and Data Augmentation

Several preprocessing techniques may be applied to dermoscopic images for noise removal and image enhancement, such as hair removal, lesion segmentation or contrast enhancement. Considering this, such measures will not be implemented in this work. The only pre-processing techniques that were used, in order to prepare the images before passing them through the network, are resizing and normalization.

As mentioned in section 3.1.1, the images of the training set have been acquired from various sources. This presents inherent changes to the color constancy and resolution size of the skin lesion images, due to illumination and acquisition methods. Such variations in color can slow down and even disrupt the training process. To ensure that input parameters, *i.e.* pixels, have a similar data distribution, all images were normalized by subtracting the mean and dividing by the standard deviation RGB values of the ImageNet dataset, that were used to pretrain the EfficientNet models. The images were also resized in order to have a common dimension before training each network. Due to the fact that each EfficientNet model has been built to perform optimally on images of a specific size, *e.g.* EfficientNet-B3 on 300x300 and EfficientNet-B4 on 380x380 sized inputs, all images were resized according to each model's default input size. The resizing was implemented using bilinear interpolation, which is a resampling method that uses the distance weighted average of the nearest pixel values to estimate new ones. These preprocessing techniques were applied to the training, validation and test set.

Deep architectures trained on small sized datasets are more likely to see patterns that do not exist by overfitting the train set. Although it took several years for ISIC to create this dataset, which consists of tens of thousands of annotated dermoscopic images, it is still considered a small to medium size dataset for training deep architectures with millions of parameters. Thus, in order to increment the diversity and quantity of the train set without actually aggregating new data, the technique of data augmentation was used.

After experimenting with several different augmentation scenarios, the following augmentation pipeline was applied for the entire dataset during training: horizontal flip, vertical flip, color jitter and random erasing. Table 3 describes the probability of applying each augmentation and any details related to them. Color jitter makes a slight change in the color values of the image and random erasing [67] erases the pixels of a rectangular region in an image. Figure 15 is an illustration of the images before and after the proposed augmentation scheme. Augmentations were applied only to the training set.

Augmentation	**Probability / Details**
Horizontal Flipping	50% / -
Vertical Flipping	50% / -
Color Jitter	Each pixel's modification is chosen uniformly from *[max (0, 1 - brightness), 1 + brightness]* *[max (0, 1 - contrast), 1 + contrast]* *[max (0, 1 - saturation), 1 + saturation]* *[-hue, hue]* Where brightness, contrast, saturation and hue are the original values of each pixel
Random Erasing	50% / scaled uniformly from: *[0.02, 0.10]*

Table 3: The probabilities and details of the augmentations applied during training

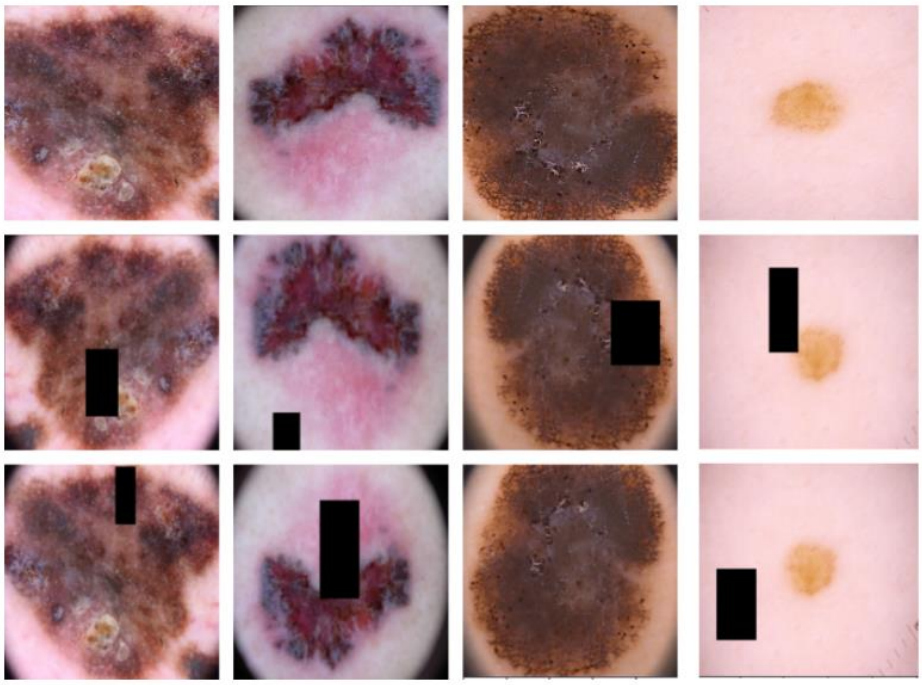

Figure 15: Training augmentation of the original images. First row: original images; second and third row: augmented images

3.4.2 Transfer Learning

The train set even after augmentation techniques applied, is still not adequately large for training models with millions of parameters. To compensate for the limited sized dataset, and accelerate and enhance the learning process, the method of transfer learning was employed as an initialization theme, with weights pretrained on ImageNet. The rationale is that the imported models already have large knowledge about many kinds of objects and by fine-tuning the weights, we allow the model to specialize to the problem in hands. The same method was applied to both approaches, where the additional layers of the second approach were initialized with the uniform LeCun method.

3.4.3 Hyper-Parameters Adjustment

To address the class imbalance problem, a weighted cross-entropy loss function [68] was used, where the class weights were computed by normalizing the inverse class frequencies from the training set. The choice of a weighted loss function during training over the

method of oversampling the minority class, was due to the fact that the models are also required to be more sensitive towards the melanoma-class. Thus, by penalizing the model with the factor of the inverse frequency, when melanoma images are misclassified, the model tends to avoid false negatives, over false positives.

All the layers, including lower convolutional layers, *i.e.* closer to the input, were fine-tuned using the Adam optimizer [69], which is a stochastic gradient descent algorithm, that computes individual adaptive learning rates for different parameters, by estimating the first- and second-order moments of the gradients. For each model the starting learning rate was selected according to Leslie Smith's 2017 paper "Cyclical Learning Rates for Training Neural Networks" [70] and a decay of 0.97 ratio was applied on each epoch, where one epoch is defined as an entire iteration over the training set.

The learning rate is one of the most important parameters when training a network and manually configuring it can be both time-consuming and error-prone. The method introduced in Leslie N. Smith's paper suggests to train the model with a few batches of samples, while letting the learning rate increase linearly between low and high values and after the training completes to plot the accuracy or the loss versus the learning rate. Figure 16 shows as an example the method applied to estimate the learning rate for a batch size of 32 with the EfficientNet-B3 model on the train set.

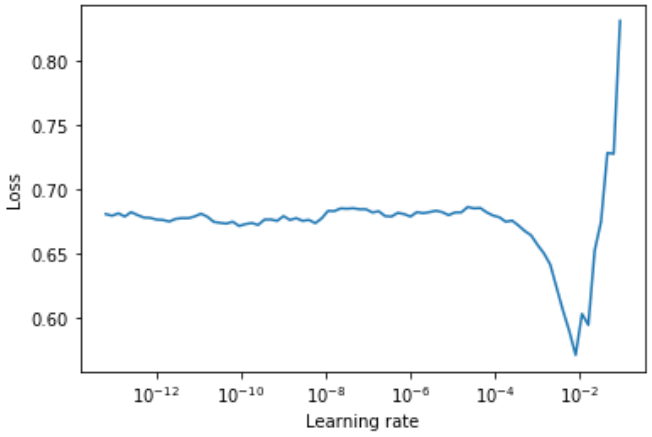

Figure 16: Loss vs Learning Rate (batch size = 32) for EffcientNet-B0

Notably, the learning rate value becomes ragged, when the loss starts to decrease and when the loss starts to diverge to infinity. A good choice for the learning rate lies between these values. In this work after choosing the largest batch size that the GPU ram could hold, the ideal learning rate for each model was near the point where the loss starts to decrease.

3.5 Ensemble Modeling

Ensemble modeling is a process where multiple diverse models are combined to solve a particular computational problem, either by using different modeling algorithms or using different training datasets. The ensemble model then totals the prediction of each model to result in a final prediction for unseen instances. The motivation for utilizing ensemble models is to reduce the variance of predictions and reduce generalization error, with a divide and conquer approach. The most popular ensemble modeling methods for neural networks, organized by each element of the method that may vary, are:

- The ensemble methods of varying training data, where a network is trained on different subsets, *e.g*. K-Fold cross validation ensemble, random training subset ensemble
- The ensemble methods of a varying network, where the same under-constrained model is trained on the same data with different initial conditions, *e.g*. hyperparameter tuning ensemble, horizontal epochs ensemble, snapshot ensemble
- The ensemble methods of varying networks, where different models are trained on the same dataset, *e.g*. max-voting ensemble, model averaging ensemble, weighted average ensemble, stacking ensemble

Extensive research has been done in the machine learning community on ensemble methods for both traditional machine learning algorithms and deep neural networks, with numerous papers devoted on how to combine models or model predictions, and how to reduce the model error that results.

By combining predictions, more accurate and robust models nearly always improve, without the need of a high-degree fine tuning which is required for single-model solutions. Typically, the models for the combination process are drawn from the same algorithm family and as the number of models in the combination increase, the ensemble model accuracy improves on average, though this not always the case. Alternatively, a better choice may be the selection of the best individual models or to determine which models combine best.

3.6 Evaluation and Metrics

The performance of the different models for melanoma detection is evaluated on the test set containing 5,846 samples, which has remained unused up to this point. Classification results for the binary classification task are obtained from inference of the probabilistic results from the output (probabilistic results from the two-class classification training).

As mentioned in section 2.3.2, caution must be taken regarding the interpretation of a model's performance measures. Due to the fact that the accuracy metric tends to be a less useful measurement for evaluating the performance, it will be reported as a mere illustration. Of high importance in the domain is the confusion matrix, depicted in Table 4, and its measurements, which provide a more detailed description of the model's performance.

		Predicted Label	
True Label	**Positive**	True Positive	False Negative
	Negative	False Positive	True Negative

Table 4: Confusion Matrix

Considering the goal of a CAD-system for melanomas, which focuses on early prevention measures, the implemented classification would target to not miss positive samples rather than negative samples. This can be evaluated with the sensitivity, also known as recall,

and precision measures. Sensitivity entails the amount of truly positive lesions that have been identified as such,

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

and precision is the amount of percentage of correctly positive identified lesions

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Whereas, specificity evaluates the performance on the negative lesions by measuring the proportion of negatives which are correctly identified

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive}$$

A way of combining the sensitivity and precision of the model is the *F1-Score*, and is defined as the harmonic mean of model's sensitivity and precision

$$F1 - Score = 2 \cdot \frac{precision \cdot sensitivity}{precision + recall}$$

To summarize with one metric the performance of a model trained on an imbalanced dataset, the *balanced accuracy* is commonly used. Unlike accuracy, it provides a better insight into the model's performance, and is calculated as the average of the proportion

corrects of each class individually *i.e.* the average of sensitivity and specificity in binary classification problems.

Another used measure for evaluating CAD-systems in the medical world is the *ROC-AUC* (Receiver Operating Characteristics Area Under Curve) score. ROC is a probabilistic curve that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. This can be visualized by depicting the true positive rate (sensitivity) versus the false positive rate (1 - specificity) at various discrimination thresholds. A diagonal ROC curve, (where TPR=FPR) depicts a random classifier, while moving upwards from this, indicates an increase in performance. The ROC-AUC measures the entire two-dimensional area underneath the ROC curve and is typically used as an overall measure to indicate the performance of a classifier, where a ROC-AUC score of 50% indicates a random classifier and a ROC-AUC score of 1 a perfect one.

During the training of the CNN architectures used throughout this work, observations were made over a sufficient number of epochs, to visualize overfitting on the evaluation dataset. The best performing models are subsequently chosen as those that either minimize validation loss or maximize validation ROC-AUC score.

3.7 Implementation

Python was used as programming language throughout this work. For the implementation of the neural networks, the PyTorch libray [71] was utilized, which is an open-source deep learning library inspired by Torch. It has been primarily developed by Facebook's artificial intelligence research group, and is built to be flexible and modular for research, with the stability and support needed for production deployment. The pretrained neural networks were loaded from the efficientnet-pytorch library [72] and are referenced *w.r.t.* the original authors throughout this work. All experiments concerning finetuning and training of neural networks were carried on a Nvidia GeForce RTX 2080 Ti 11GB GPU card.

4 Experiments and Results

This section presents the results from evaluating a baseline network using different configurations, and the results from the predictions of different models trained with the proposed configuration. Lastly, the results from ensemble methods of different combinations of the trained models are evaluated and compared.

4.1 Configuration Experiments

A configuration entails the model trained and evaluated using a unique combination of techniques, such as applying a specific pre-processing step with or without augmentation, or a certain technique to address the class imbalance problem. The baseline model that was utilized for the assessment of the impact of the different configurations explored in this work is the EfficientNet-B3.

A configuration without any attempt to address the class imbalance problem would be aimless, as the dataset is highly imbalanced to effectively train deep CNN's architectures with unbiased predictions towards the dominant class, *i.e.* non-melanoma examples. For this reason, no such attempt would be presented. Four different training configuration sets were analyzed: *i)* (Center Cropped Images + Oversampling) - the pre-processing step of center cropping the images to the networks input dimensions and oversampling of the minority class; *ii)* (Resized Images + Oversampling) – the pre-processing step of resizing the images and oversampling the minority class; *iii)* (Resized Images + Proposed Augmentation + Oversampling) same as the second configuration along with the proposed augmentation scheme to the entire dataset; *iv)* (Resized Images + Proposed Augmentation + Weighted Loss Function) – same as the last configuration mentioned with the utilization of a weighted loss function, instead of oversampling the melanoma instances, to both tackle the class imbalance problem and to avoid false negatives.

These configurations will involve evaluating the performance of the network, previously defined to reveal what contributes to improve the performance in this task. In all cases, the model that was chosen to be evaluated on the test set is that which minimized the

validation loss during a training of 20 epochs. Table 5 shows the results of EfficientNet-B3 trained with the different configuration sets.

	Bal. Acc.	Sensitivity	Specificity	ROC-AUC
Center Cropped Images + Oversampling	72.32%	46.4%	98.24%	81.3%
Resized Images + Oversampling	85.44%	75.15%	95.73%	95.8%
Resized Images + Proposed Augmentation + Oversampling	87.65%	79.1%	96.21%	96.24%
Resized Images + Proposed Augmentation + Weighted Loss Function	**87.38%**	**84.15%**	**90.61%**	**94.96%**

Table 5: EfficientNet-B3 results over the three configuration sets

One of the simplest augmentation methods that could have been a good candidate for this image classification problem, is the resampling scheme of random cropping from the center of the images. The process of resampling the images with a random center crop based on the input dimensions of each network, could have both augmented the data and replaced the preprocessing technique of resizing. But for the specific dataset it turned out

to be not a good technique. In general, the skin lesion should be located at the center of the image. However, due to the different image conditions, the skin lesion can be off center and in different scales and angles. As a result, important information that may define the diagnosis can be missing from the cropped image, *e.g*. the borders of the skin lesion. Figure 17 shows original image examples from the dataset and the resulted images after the resampling scheme of random center cropping was applied.

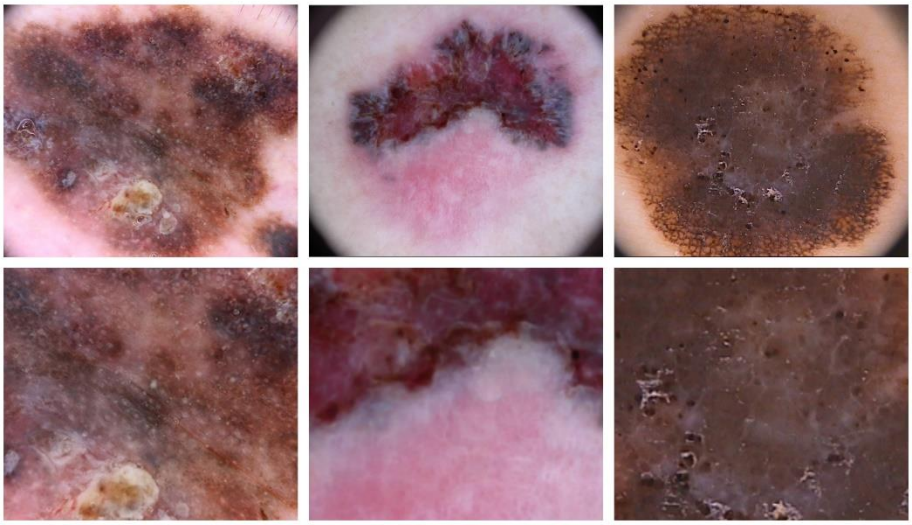

Figure 17: Examples of original images from the dataset (above) and the resulted images after random center cropping (below)

Yet, data augmentation consists of expanding the training set with transformations of the images, provided that the semantic information is not lost. Although the process of resizing distorts to a certain degree the dermoscopic images, it maintains the overall illustration of the lesion including all its areas and borders, and thus provides better results.

Notably, augmenting the entire dataset with different versions of the original images it provides further improved results for detecting melanoma, with the balanced accuracy score increasing by almost 4%. The final and proposed configuration setup uses a weighted loss function instead of the technique of oversampling the minority class. The

two approaches provide a similar balanced accuracy score, however, the performances of the two techniques differ in the sensitivity and specificity metrics. The technique of a weighted loss function improves the sensitivity score by a 5%, while the same time reduces the specificity score, which was achieved with the technique of oversampling. However, due to the fact that it's of great importance to avoid false negatives in the domain, a higher preference was given to the configuration that provided a model with improved sensitivity.

4.2 Detailed Results

The default and proposed architectures of EfficientNet-B3 – B6 were evaluated after being fine-tuned with the preferred configuration. The features of the models are summarized in table 6. The detailed results of all evaluated architectures are reported in Tables 7 - 10.

	Input Shape	#Params (M)	Memory Usage (Mb)
EfficientNet-B3	300, 300, 3	12	49
EfficientNet-B4	380, 380, 3	19	77
EfficientNet-B5	456, 456, 3	30	122
EfficientNet-B6	528, 528, 3	43	173

Table 6: Base models footprint details. (M = Millions, Mb = Megabytes)

	EfficientNet-B3	EfficientNet-B4	EfficientNet-B5	EfficientNet-B6
Accuracy	90.03%	90%	92.61%	93.89%
Bal. Accuracy	87.28%	87.84%	86.57%	87.89%
Sensitivity	83.95%	85.12%	79.25%	80.62%
Precision	46.12%	46.37%	55.4%	61.49%
Specificity	90.61%	90.57%	93.88%	95.16%
F1-Score	59.54%	60.05%	65.21%	69.77%
ROC-AUC	94.96%	94.9%	95.29%	95.94%
# Epochs	12	11	12	10

Table 7: Results of the default models that minimized the validation loss

	EfficientNet-B3	EfficientNet-B4	EfficientNet-B5	EfficientNet-B6
Accuracy	91.61%	90.81%	91.09%	91.57%
Bal. Accuracy	88.68%	88.06%	88.12%	89.71%
Sensitivity	85.12%	84.74%	84.54%	87.47%
Precision	51.24%	48.54%	49.43%	51%
Specificity	92.23%	91.40%	91.72%	91.95%
F1-Score	63.97%	61.72%	62.38%	64.45%
ROC-AUC	95.93%	95.72%	95.80%	95.85%
# Epochs	16	13	15	11

Table 8: Results of the proposed models that minimized the validation loss

	EfficientNet-B3	EfficientNet-B4	EfficientNet-B5	EfficientNet-B6
Accuracy	91.46%	90.61%	92.54%	94.5%
Bal. Accuracy	88.06%	87.25%	87.42%	88.5%
Sensitivity	83.95%	83.17%	81.21%	81.21%
Precision	50.71%	47.86%	54.97%	64.84%
Specificity	92.19%	91.32%	93.63%	95.78%
F1-Score	63.23%	60.76%	65.56%	72.11%
ROC-AUC	95.82%	95.67%	95.76%	96%
# Epochs	13	15	13	12

Table 9: Results of the default models that maximized the ROC-AUC score

	EfficientNet-B3	EfficientNet-B4	EfficientNet-B5	EfficientNet-B6
Accuracy	91.61%	91.94%	91.09%	93.94%
Bal. Accuracy	88.68%	87.53%	88.12%	88.36%
Sensitivity	85.12%	82.19%	84.54%	81.60%
Precision	51.24%	52.5%	49.43%	61.6%
Specificity	92.23%	92.88%	91.72%	95.12%
F1-Score	63.97%	64.07%	62.38%	70.2%
ROC-AUC	95.93%	95.82%	95.80%	96.43%
# Epochs	16	18	15	13

Table 10: Results of the proposed models that maximized the ROC-AUC score

Notably, the models with the proposed architecture required more training time, compared to the default models, and provided slightly improved balanced accuracy and ROC-AUC scores, with the best model being the proposed EfficientNet-B6 architecture (89.71% balanced accuracy and 96.43% ROC-AUC score).

4.3 The Ensemble Models Results

To further improve results, different combinations of the trained architectures were also evaluated on the test set with the average ensemble method. The three combinations analyzed are the ensemble method of the *i)* default models, *ii)* the proposed models and *iii)* EfficientNet-B3 – B6 with the proposed architecture and EfficientNet-B4 – B5 with the default architecture, which was the optimal ensemble method.

	Default Models	Proposed Models	Optimal
Accuracy	92.47%	94.15%	93.69%
Bal. Accuracy	90.83%	90.78%	91.14%
Sensitivity	88.84%	86.7%	88.1%
Precision	54.24%	61.79%	59.36%
Specificity	92.82%	94.86%	94.23%
F1-Score	67.35%	72.15%	70.93%
ROC-AUC	96.9%	97.4%	97.45%

Table 11: Results of the ensemble of models that minimized the validation loss

	Default Models	Proposed Models	Optimal
Accuracy	94.68%	95.06%	95.19%
Bal. Accuracy	90%	90.04%	90.64%
Sensitivity	84.34%	83.95%	85.12%
Precision	65.10%	67.45%	67.97%
Specificity	95.67%	96.12%	96.15%
F1-Score	73.48%	74.8%	75.59%
ROC-AUC	97.37%	97.55%	98.1%
ROC-AUC from Kaggle's Public Leaderboard	**93.4%**	**93.55%**	**94.04%**

Table 12: Results of the ensemble of models that maximized the ROC-AUC score

All ensemble models achieved superior metrics from individual networks, with the best model being the ensemble of the default EfficientNetB4 – B5 and EfficientNetB3 – B6 with the proposed architecture. The highest ROC-AUC score on the automatic evaluation system of Kaggle for the dataset of the 10,982 unlabeled images was provided by the optimal ensemble method of the best performing individual models in terms of ROC-AUC score (94.04%) and it's among the top-5% performances on the challenge.

4.4 Visualizations

The integrated gradients method [73] was utilized to calculate feature attributions for the default EfficientNet-B0 – B4. Figure 18 – 23 illustrate examples of melanoma and non-melanoma images and the corresponding integrated gradient attributions for the networks.

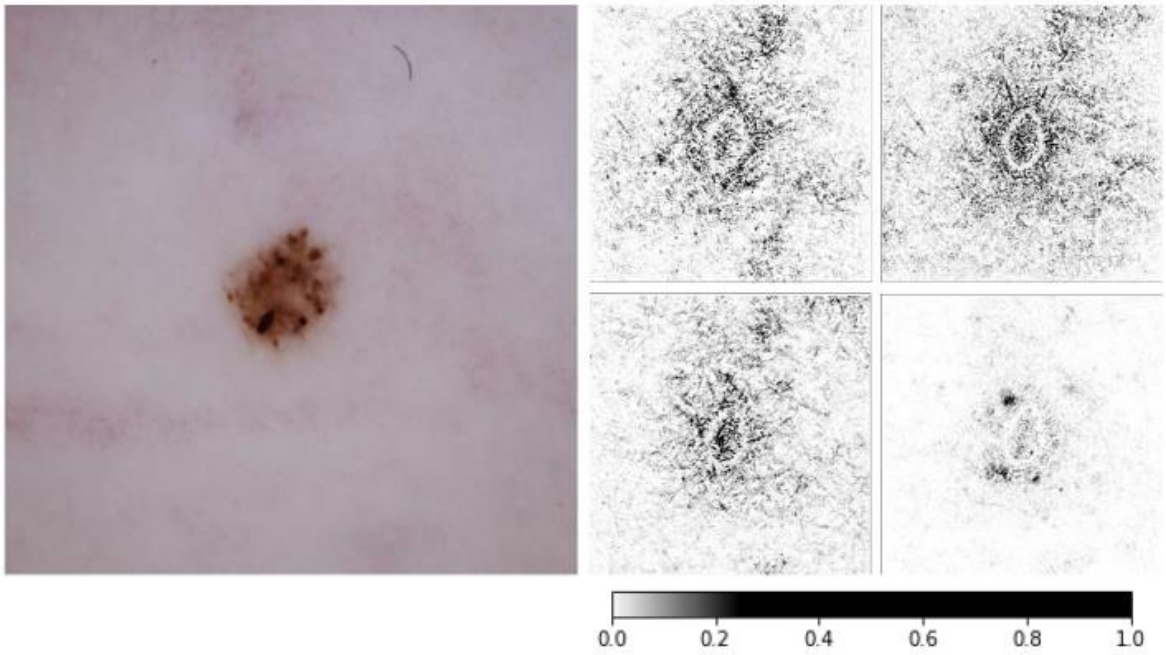

Figure 18: A non-melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

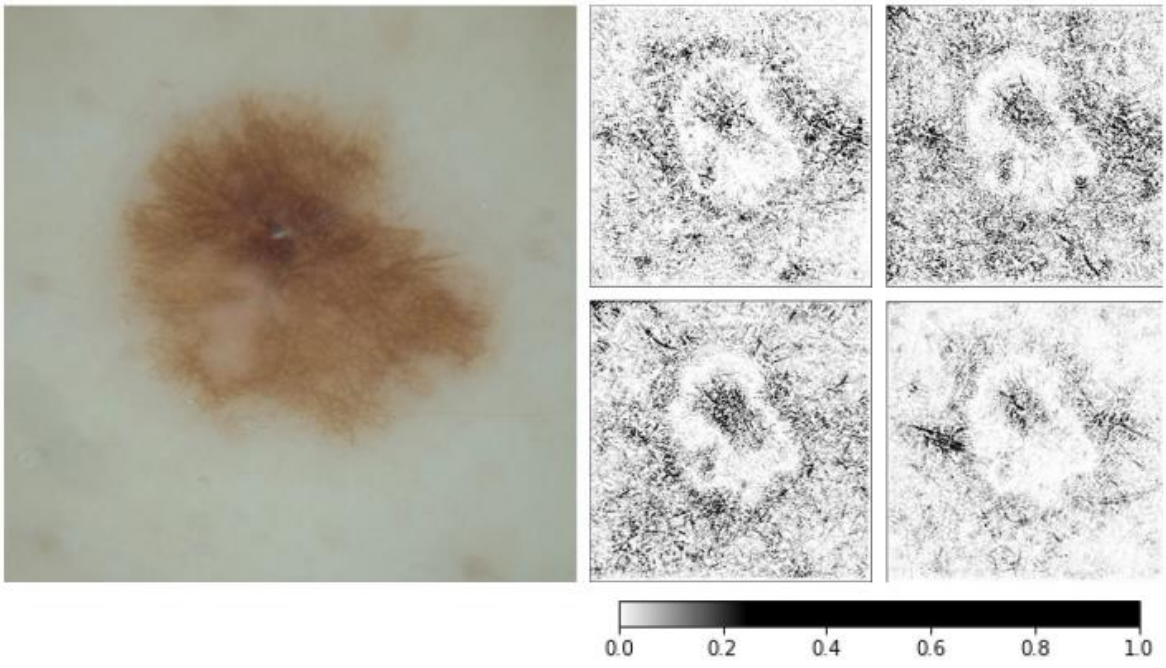

Figure 19: A non-melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

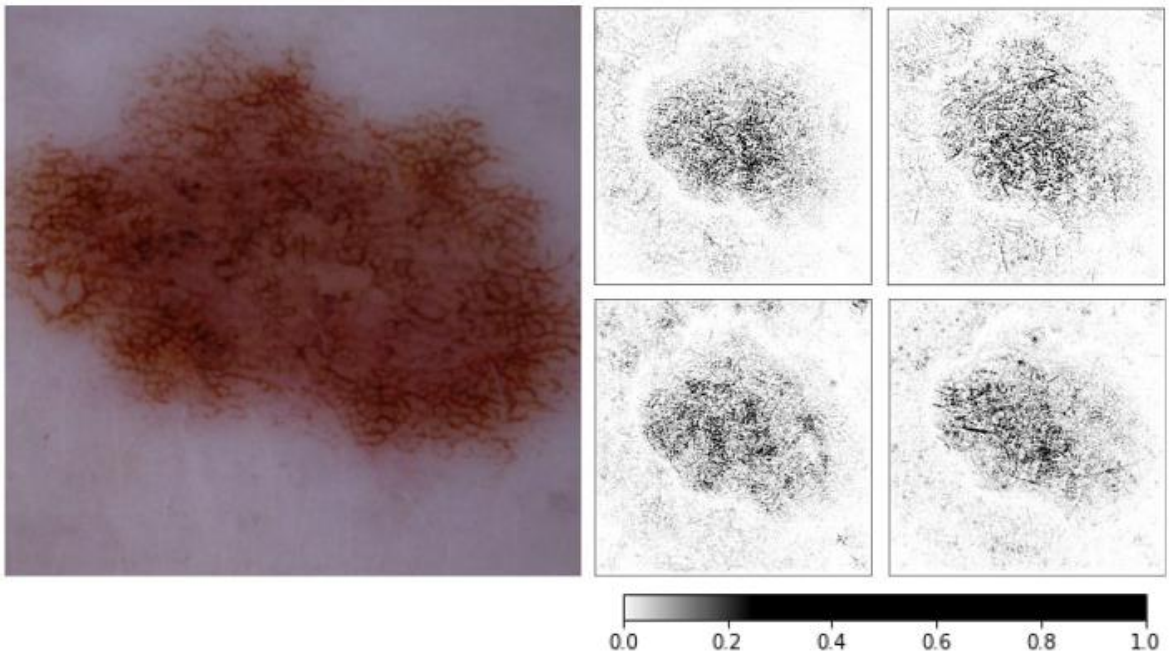

Figure 20: A non-melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

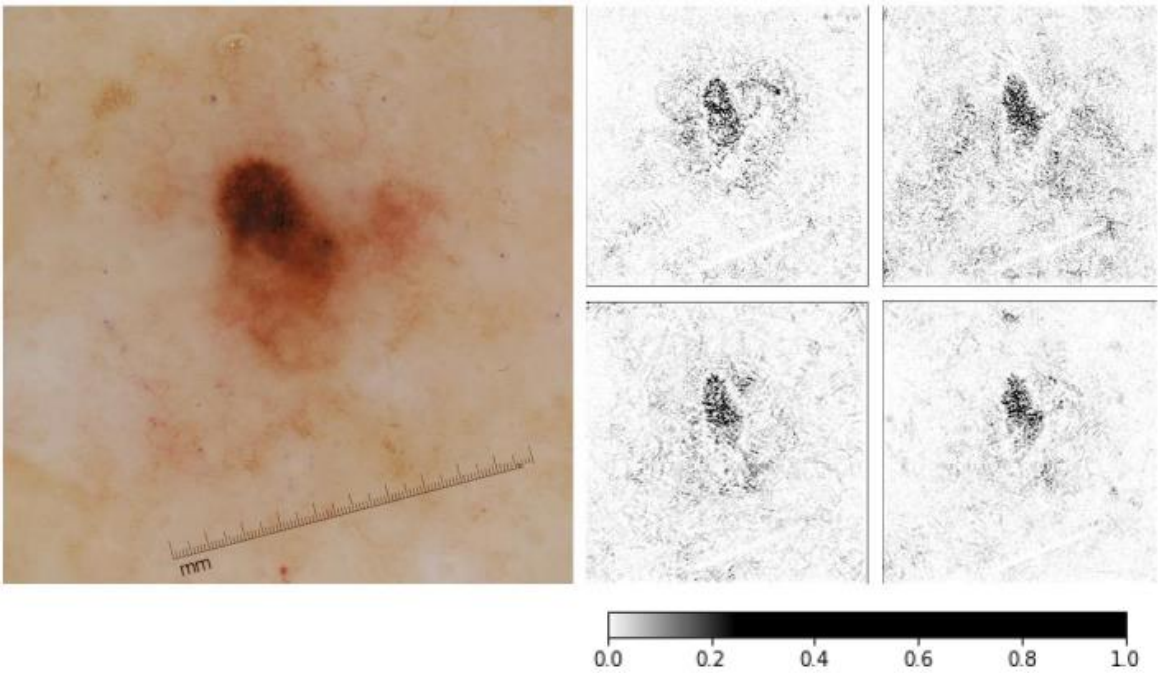

Figure 21: A melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

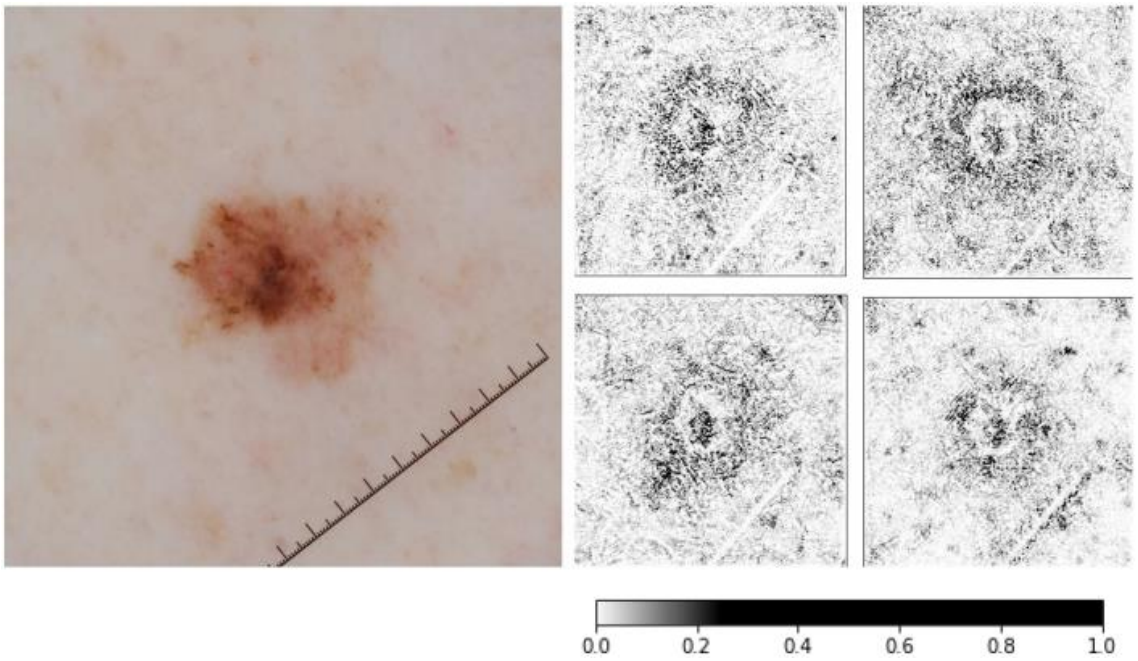

Figure 22: A melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

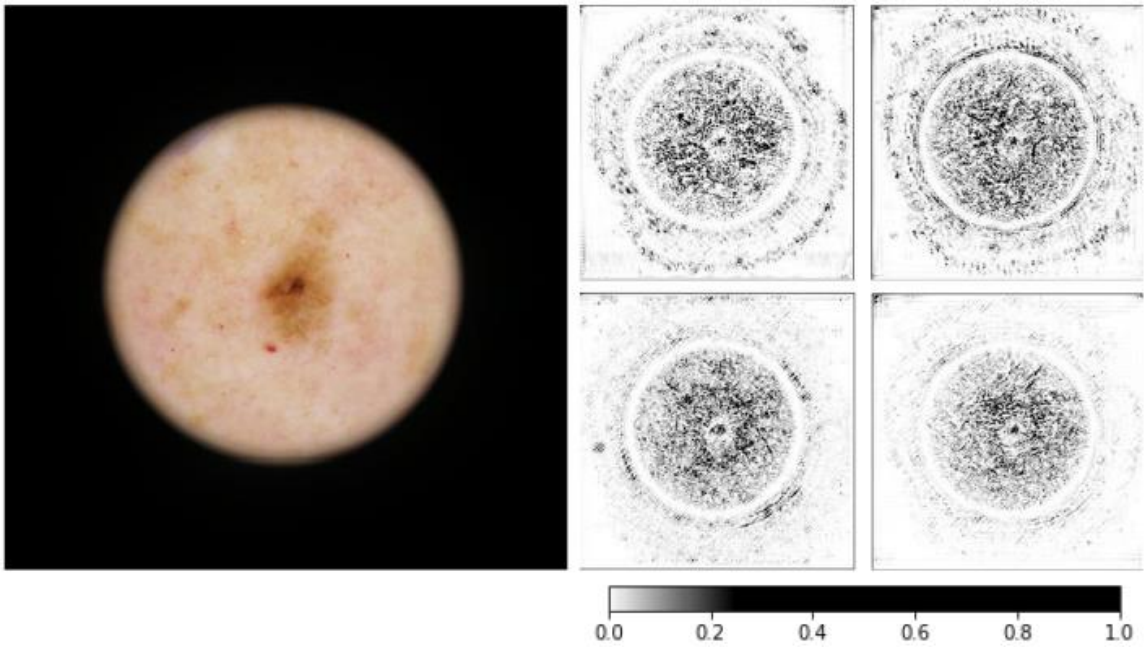

Figure 23: A melanoma test set image (left) and the corresponding integrated gradient attributions for EfficientNet-B3 – B6 (right)

The integrated gradients method computes the importance scores φ_i^{IG} by accumulating gradients interpolated between a baseline x'_i input (intended to represent the absence of data, in this case this is a black image) and the current input x_i .

$$\varphi_i^{IG} = (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\theta F(x' + a \times (x - x'))}{\theta x_i} d\alpha \qquad (5)$$

The CNN model is represented as F in equation 5.

It can be observed that the models tend to focus primarily on the edges of the skin lesions (Figure 4). This aligns with expectation, since uneven or notched edges are common in melanoma. Secondary to the edges, there is some importance to the lesion itself and surrounding skin for some of the networks. This is significant because melanomas can also show uneven texture or color [74].

4.5 Comparison with the State-of-the-Art

A comparison of results from the optimal ensembles of the models that maximized the ROC-AUC score and minimized the validation loss with state-of-the-art methods is displayed in table 13.

The results are only indicative of being in the same proximity in terms of accuracy, and not a precise and reliable performance comparison to existing research due to differences in the data and task. Every method in table 7 uses a unique dataset, some with or without dermoscopy images and a few classify more than two lesion classes. Lastly, because this thesis is based upon CNNs, table 7 specifically includes other state-of-the-art researches that employ CNNs as classification method.

	Dataset Size	Accuracy	ROC-AUC	Year
Esfahani et al. *[75]*	6,120	81%	-	2016
Kawahara et al. *[43]*	1,300	79,5%	-	2016
Esteva et al. *[42]*	129,450	72,1%	-	2017
Codella et al. *[51]*	2,150	93,1%	92,6%	2017
Haenssle et al. *[76]*	100,000	87,5%	95%	2018
Han et al. *[77]*	20,826	-	96%	2018
Optimal Ensemble on Test Set *(validation loss)*	58,457	**91.14%**	97.45%	2020
Optimal Ensemble on Kaggle Leaderboards *(ROC-AUC Score)*	58,457	-	**94.04%**	2020

Table 13: Comparison of the optimal ensemble method against state-of-the-art methods

5 Discussion

It is important to mention once again that the state-of-the-art deep learning networks, used in this thesis to diagnose melanoma, have originally been designed and evaluated on the ImageNet dataset using a specific resolution. They were not specifically created for a binary classification problem, but rather to recognize and classify objects into a thousand different classes such as trees, planes, busses, dogs and much more. All of the classes they categorize have many distinct inherent features which makes them more distinguishable from other classes in most cases, *e.g.* a cat versus a car. However, this is not always the case with a malignant melanoma versus non-melanoma lesion. Distinguishing a malignant melanoma from non-melanoma lesions is a substantially more difficult and complicated task due to the similarities they have in common and different variations, which correlates understandably towards why medical professionals in the field only achieve a diagnostic specificity of 59%, due to uncertainty when distinguishing melanoma from atypical lesions [78], [79]. When diagnosing melanoma, the standardized set of guidelines of the ABCDE rule is usually followed. This involves checking a melanoma for an asymmetrical shape, irregular border, the coloration (if there is multiple colors present or unusual color distribution), diameter and the evolution of the lesion over the span of time, the latter being the most important factor in deciding whether it is a melanoma. However, back to the point of the difficulty of distinction, a non-melanoma lesion may possess features normally attributed to a malignant melanoma such as an irregular border, asymmetrical shape or unusual coloration, and in turn a melanoma may possess a symmetrical shape and/or smooth border. Especially in the latter case when a melanoma is in its early stages, as it has not yet been allowed to evolve beyond a very small point which is when it is almost always curable. This is why this is a very challenging task to do with great precision in comparison to the previously mentioned image recognition tasks and should be kept in mind while going forward with discussing the results of the models tested.

Apart from the complexity of the task, the dataset is also a small to medium size dataset for training deep architectures, and has a class imbalance problem, with the minority class

of melanoma being only 5,106 examples. All these facts become apparent to the results of the fine-tuned networks. The augmentation method certainly improves the over-all metrics, but the balanced accuracy score then settles to a certain score and the additional different training techniques to further improve results, provide only a trade-off between sensitivity and specificity scores. However, these metrics show a slight improvement in models with the proposed architecture compared to models with the default architecture. This implies that the Mish activation function indeed improved the performance of the networks and indicates that the metadata have at least a rudimentary influence for the outcome.

Notably, the deeper and more complex networks with larger input resolutions don't provide that much greater results from the baseline models. This applies to both the default models and the proposed models. Yet, the ensemble methods of both approaches improve all metrics. In general, ensembles are known to perform better than single models and diversity has been identified as an important factor in explaining their success. An assumption as to the nature of the diversity present in the ensemble methods of this work, can be based on the fact that the EfficientNets have different scales and input resolutions. Thus, each pretrained network has learned, to a certain degree at least, different representations of the categories of the ImageNet dataset. Considering this with the fact that each network has then been finetuned on resized dermoscopic images, that illustrate lesions from different scales and angles, each resulting model considers also to a certain degree different features from an image to detect melanoma. As a result, some networks can be a better candidate for determining the outcome for images with certain attributes, while not being the most suitable for images with other attributes. Hence, an ensemble method which considers the confidence levels of these varying models, generalizes better to the unseen data. For example, the lesion depicted in figure 4 is a benign - non-melanoma lesion. The EfficientNet-B4 classified the lesion as malignant melanoma with a confidence level of 58.42%. However, the rest of the models, *i.e.* EfficientNet-B3, EfficientNet-B5 and EfficientNet-B6 correctly classified the lesion as non-melanoma, with confidence level of 55.77%, 89.9% and 67.38% respectively.

Additionally, the ensemble method of both default models and proposed models, includes networks trained with different activation functions, with the proposed models

diversifying their architecture even further from the default models, by taking into consideration also the metadata. Activation functions play a crucial role in deep learning as they define the output of every node of a network. Thus, by changing the activation function of a convolutional neural network architecture, the performance and the training dynamics of the network change as well. Taking this into account, and in order to make a more robust ensemble, that can generalize more to the unseen data, the proposed models, apart from taking into consideration the metadata, are also trained using a different activation function. In fact, the component that mainly diversifies the proposed models from the default members of the ensemble is the different activation function. Thus, the performance of the optimal ensemble method improves furthermore than the ensemble of just varying scale networks, with all metrics providing better results.

6 Conclusion and Future Work

In this thesis, it's demonstrated that an ensemble method of varying scale deep neural networks, trained with different activation functions, can achieve competitive classification performance in detecting melanoma from dermoscopic images, with the best model (in terms of ROC-AUC score) achieving a 94.04% ROC-AUC score in public leaderboards, which is among the top-5% performances. This could emerge as an automated melanoma classification system using dermoscopic images, that could be used along with experienced dermatologists.

The results showed the positive impact of using data augmentation for training melanoma classification models, where the typical problem of severe class imbalance was addressed with a loss balancing approach. To deal with multiple image resolutions, various EfficientNets were employed and to further diversify the ensemble method, the models were utilized again in an architecture with a different activation function that incorporates the metadata.

The next step towards increasing the performance of the classification is the forming of an ensemble with more models and with its members also trained on varying resolutions. By examining the leaderboard of the Kaggle's ISIC2020 challenge, the option of the ensemble method of varying resolutions seems to be an effective one as the top scorers deploy this technique. The winning solution achieved a ROC-AUC score of 94.90% by utilizing an ensemble method of 18 networks [80]. The models were the EfficientNet-B3 - B7, each trained on varying resolutions, and the SE-ResNext101 and ResNest101. This implies that networks trained with varying input resolutions have the characteristic of diversity to build an effective ensemble. Combining this technique with the methods described in this thesis can possibly improve even more results by further diversifying the ensemble method. However, such approaches require the training of many networks, making them computationally expensive and time consuming.

Bibliography

[1] N. C. Institute. [Online]. Available: https://www.cancer.gov/types/common-cancers.

[2] "Skin Cancer Foundation," [Online]. Available: https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/.

[3] R. L. Siegel, K. D. Miller and A. Jemal, "Cancer statistics, 2018," *CACancer J. Clinicians,* vol. 68, no. 1, pp. 7–30, doi: 10.3322/caac.21590, 2018.

[4] N. Nami, E. Giannini, M. Burroni, M. Fimiani and P. Rubegni, "Teledermatology: State-of-the-art and future perspectives," *Expert Rev.Dermatol.,* vol. 7, no. 1, pp. 1-3, doi: 10.1586/edm.11.79, 2012.

[5] H. Haenssle, C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, A. Blum, A. Kalloo, A. B. H. Hassen, L. Thomas, A. Enk and L. Uhlmann, "'Managainst machine: Diagnostic performance of a deep learning convolutionalneural network for dermoscopic melanoma recognition in comparison to58 dermatologists," *Ann Oncol.,* vol. 29, no. 8, pp. 1836-1842, doi: 10.1093/annonc/mdy166, 2018.

[6] A. M. Turing, "Computing machinery and intelligence," *Mind,* vol. 59, no. 236, pp. 433-460, 1950.

[7] A. L. Samuel, "Some studies in machine learning using the game of checkers," in IBM Journal of Research and Development," vol. 44, no. 1.2, pp. 206-226, doi: 10.1147/rd.441.0206, 2000.

[8] T. M. Michell, "Machine Learning," *New York: McGraw-Hill,* 2013.

[9] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning representations by back-propagating errors," *Nature,* vol. 323, pp. 533-536, doi: 10.1038/323533a0, 1986.

[10] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review,* vol. 65, no. 6, pp. 386–408, doi: 10.1037/h0042519, 1958.

[11] LeCun, Y. a. Bengio and Yoshua, "Convolutional Networks for Images, Speech, and Time Series," in *The Handbook of Brain Theory and Neural Networks*, Cambridge, MA: MIT Press, 1998, p. 255–258.

[12] A. Rajkomar, S. Lingam, A. G. Taylor, M. Blum and J. Mongan, "High-throughput classification of radiographs using deep convolu- tional neural networks," *Journal*

of digital imaging, vol. 30, no. 1, pp. 95–101, doi: 10.1007/s10278-016-9914-9, 2017.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[14] E. Hosseini-Asl, M. Ghazal, A. Mahmoud, A. Aslantas, A. M. Shalaby, M. F. Casanova, G. N. Barnes, G. Gimel'farb, R. Keynton and A. El-Baz, "Alzheimer's disease diagnostics by a 3D deeply supervised adaptable convolutional network," *Front Biosci (Landmark Ed),* vol. 23, pp. 584-596, doi: 10.2741/4606. PMID: 28930562, 2018.

[15] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding and Y. Zheng, "Convolutional Neural Networks for Diabetic Retinopathy," *Procedia Computer Science,* vol. 90, pp. 200-205, doi: 10.1016/j.procs.2016.07.014, 2016.

[16] D. Erhan, P.-A. Manzagol, Y. Bengio, S. Bengio and P. Vincent, "The difficulty of training deep architectures and the effect of unsupervised pre-training," in *International Conference on Artificial Intelligence and Statistics (ICAIS)*, 2009.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, " ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV),* vol. 115, no. 3, pp. 211-252, http://www.image-net.org/challenges/LSVRC/, 2015.

[18] Srivastava, N. a. Hinton, G. E. a. Krizhevsky, A. a. Sutskever, I. a. Salakhutdinov and Ruslan, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research,* vol. 15, pp. 1929-1958, 2014.

[19] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of International Conference on Machine Learning (ICML)*, arXiv:1502.03167, 2015.

[20] GARD. [Online]. Available: https://rarediseases.info.nih.gov/diseases/4883/situs-inversus.

[21] N. H. Matthews, W.-Q. Li, A. A. Qureshi, M. A. Weinstock and E. Cho, Chapter 1: Epidemiology of Melanoma., 2017.

[22] Beaumont, "ABCDE's of Melanoma," [Online]. Available: https://www.beaumont.org/conditions/melanoma/abcde's-of-melanoma.

[23] S. Urbancek, P. Fedorcova, J. Tomkova and R. Sutka, "Misdiagnosis of Melanoma: A 7 Year Single-Center Analysis," *Journal of Pigmentary Disorders, doi: 10.4172/2376-0427.1000208,* 2015.

[24] M. E. Celebi, Y. A. Aslandogan, W. V. Stoecker, H. Iyatomi, H. Oka and X. Chen, "Unsupervised border detection in dermoscopy images," *Skin Research and Technology,* vol. 13, no. 4, pp. 454-462, doi: 10.1111/j.1600-0846.2007.00251.x, 2007.

[25] P. R. Mahajan and Prof.Mrs.A.J.Vyavahare, "Artefact Removal and Contrast Enhancement for Dermoscopic Images Using Image Processing Techniques," *International Journal of Innovative Research in Electrical, Electronics, Instrumental and Control Engineering,* vol. 1, no. 9, 2013.

[26] I. Maglogiannis and K. Delibasis, "Hair removal on dermoscopy images," *Annu Int Conf IEEE Eng Med Biol Soc,* pp. 2960-3. doi: 10.1109/EMBC.2015.7319013, 2015.

[27] S. Bakheet, "An SVM Framework for Malignant Melanoma Detection Based on Optimized HOG Features," *Computation,* vol. 5, no. 1, pp. 4, doi: 10.3390/computation5010004 , 2017.

[28] H. Iyatomi, H. Oka, M. E. Celebi, M. Tanaka and K. Ogawa, "Parameterization of dermoscopic findings for the internet-based melanoma screening system," pp. 189 – 193, doi: 10.1109/CIISP.2007.369315, 2007.

[29] Q. Abbas, M. E. Celebi and I. Fond´on, "Computer-aided pattern classification system for dermoscopy images," *Skin research and technology,* vol. 18, no. 3, pp. 278-289, doi: 10.1111/j.1600-0846.2011, 2011.

[30] R. Erol, M. Bayraktar, S. Kockara, S. Kaya and T. Halic, "Texture based skin lesion abruptness quantification to detect malignancy," *BMC Bioinformatics,* pp. 51–60, doi: 10.1186/s12859-017-1892-5,, 2017.

[31] W. Stoecker, W. W. Li and R. Moss, "Automatic detection of asymmetry in skin tumors," *Computerized medical imaging and graphics: the official journal of the Computerized Medical Imaging,* vol. 16, no. 3, pp. 191–197, doi: 10.1016/0895-6111(92)90073-I, 1992.

[32] S. Seidenari, G. Pellacani and C. Grana, "Colors in atypical nevi: A computer description reproducing clinical assessment," *Skin research and technology,* vol. 11, no. 1, pp. 36–41, doi: 10.1111/j.1600-0846.2005.00097.x, 2005.

[33] K. Møllersen, M. Zortea, K. Hindberg, T. Schopf, S. Skrøvseth and a. F. Godtliebsen, Improved Skin Lesion Diagnostics for General Practice by Computer-Aided Diagnostics, 2015, pp. 247-292, doi: 10.1201/b19107-10.

[34] M. E. Celebi and A. Zornberg, "Automated quantification of clinically significant colors in dermoscopy images and its application to skin lesion classification," *IEEE*

Systems Journal, vol. 8, no. 3, pp. 980-984, doi: 10.1109/JSYST.2014.2313671, 2014.

[35] R. Stanley, W. Stoecker and R. Moss, "A relative color approach to color discrimination for malignant melanoma detection in dermoscopy images," *Skin research and technology,* vol. 13, no. 1, pp. 62-72, doi: 10.1111/j.1600-0846.2007.00192.x, 2007.

[36] H. Iyatomi, H. Oka, M. E. Celebi, K. Ogawa, G. Argenziano, H. P. Soyer, H. Koga, T. Saida, K. Ohara and M. Tanaka, "Computer-Based Classification of Dermoscopy Images of Melanocytic Lesions on Acral Volar Skin," *J. Invest. Dermatol.,* vol. 128, no. 8, pp. 2049-2054, doi: 10.1038/jid.2008.28, 2008.

[37] M. E. Celebi, H. A. Kingravi, B. Uddin, H. Iyatomi, Y. A. Aslandogan, W. V. Stoecker and R. H. Moss, "A methodological approach to the classification of dermoscopy images," *Comp. Med. Imag. and Graph.,* vol. 31, no. 6, pp. 362–373, doi: 10.1016/j.compmedimag.2007.01.003, 2007.

[38] M. Rastgoo, R. Garc´ıa, O. Morel and F. Marzani, "Automatic differentiation of melanoma from dysplastic nevi," *Comp. Med. Imag. and Graph.,* vol. 43, pp. 44–52, doi: 10.1016/j.compmedimag.2015.02.011, 2015.

[39] A. Victor and M. R. Ghalib, "Automatic Detection and Classification of Skin Cancer," *International Journal of Intelligent Engineering and Systems ,* vol. 10, no. 3, pp. 444-451, doi: 10.22266/ijies2017.0630.50, 2017.

[40] L. Li, Q. Zhang, Y. Ding, H. Jiang, B. H. Thiers and J. Z. Wang, "Automatic diagnosis of melanoma using machine learning methods on a spectroscopic system," *BMC Medical Imaging,* vol. 14, no. 36, 2014.

[41] N. Mishra and M. E. Celebi, "An Overview of Melanoma Detection in Dermoscopy Images Using Image Processing and Machine Learning," *arXiv:1601.07843,* 2016.

[42] 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, pp. 115–118, doi: 10.1038/nature21056, 2017.

[43] J. Kawahara, A. BenTaieb and G. Hamarneh, "Deep features to classify skin lesions," *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI),* pp. 1397-1400, doi: 10.1109/ISBI.2016.7493528, 2016.

[44] H. Liao, "A Deep Learning Approach to Universal Skin Disease Classification," *University of Rochester, Department of Computer Science,* 2015.

[45] Y. Li and L. Shen, "Skin Lesion Analysis towards Melanoma Detection Using Deep Learning Network," *Sensors (Basel),* vol. 18, no. 12, pp. 556, doi: 10.3390/s18020556, 2018.

[46] A. Rezvantalab, H. Safigholi and S. Karimijeshni, "Dermatologist Level Dermoscopy Skin Cancer Classification Using Different Deep Learning Convolutional Neural Networks Algorithms," *arXiv:1810.10348,* 2018.

[47] N. Codella, Q.-B. Nguyen, S. Pankanti, D. Gutman, B. Helba, A. Halpern and J. R. Smith, "Deep Learning Ensembles for Melanoma Recognition in Dermoscopy Images," *arXiv:1610.04662,* 2016.

[48] I. S. I. C. (ISIC). [Online]. Available: https://www.isic-archive.com.

[49] ISIC, "ISIC 2019 Skin Lesion Analysis Towards Melanoma Detection," [Online]. Available: https://challenge2019.isic-archive.com/.

[50] P. Tschandl, C. Rosendahl and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Sci Data 5,* Vols. 180161, doi: 10.1038/sdata.2018.161, 2018.

[51] N. C. F. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler and A. Halpern, "Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)," arXiv:1710.05006, 2017.

[52] M. Combalia, N. C. F. Codella, V. Rotemberg, B. Helba, V. Vilaplana, O. Reiter, A. C. Halpern, S. Puig and J. Malvehy, "BCN20000: Dermoscopic Lesions in the Wild," arXiv:1908.02288, 2019.

[53] ISDIS, "The ISIC 2020 Challenge Dataset," doi.org/10.34970/2020-ds01, 2020.

[54] V. Rotemberg, N. Kurtansky, B. Betz-Stablein, L. Caffery, E. Chousakos, N. Codella, M. Combalia, S. Dusza, P. Guitera, D. Gutman, A. Halpern, H. Kittler, K. Kose, S. Langer, K. Lioprys, J. Malvehy, S. Musthaq, J. Nanda, O. Reiter, G. Shih, A. Strat, P. Tschandl, J. Weber and P. Soyer, "A Patient-Centric Dataset of Images and Metadata for Identifying Melanomas Using Clinical Context," arXiv:2008.07360, 2020.

[55] "Kaggle," [Online]. Available: https://www.kaggle.com/.

[56] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *arXiv:1905.11946v5,* 2020.

[57] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard and Q. V. Le, "MnasNet: Platform-Aware Neural Architecture Search for Mobile," *arXiv:1807.11626*.

[58] T. Elsken, J. H. Metzen and F. Hutter, "Neural Architecture Search: A Survey," *arXiv:1808.05377v1,* 2018.

[59] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *arXiv:1801.04381*.

[60] L. Sifre, "Rigid-Motion Scattering For Image Classification," *PhD Thesis,* 2014.

[61] L. S. a. S. Mallat, "Rotation, Scaling and Deformation Invariant Scattering for Texture Discrimination," in *IEEE Conference on Computer Vision and Pattern Recognition*, Portland, 2013.

[62] S. Elfwing, E. Uchibe and K. Doya, "Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning," *arXiv:1702.03118 ,* 2017.

[63] P. Ramachandran, B. Zoph and Q. V. Le, "Searching for Activation Functions," *arXiv:1710.05941v2,* 2017.

[64] V. Nair, V. a. Hinton and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, Madison, 2010.

[65] D. Misra, "Mish: A Self Regularized Non-Monotonic Activation Function," *arXiv:1908.08681,* 2020.

[66] Q. Liu, Y. Chen and S. Furber, "Noisy Softplus: an activation function that enables SNNs to be trained as ANNs," *arXiv:1706.03609* .

[67] Z. Zhong, L. Zheng, G. Kang, S. Li and Y. Yang, "Random Erasing Data Augmentation," *arXiv:1708.04896v2,* 2017.

[68] Y. Ho and S. Wookey, "The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling," *IEEE Access,* vol. 8, pp. 4806-4813, 2020.

[69] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arxiv.org/abs/1412.6980,* 2015.

[70] L. N. Smith, "Cyclical Learning Rates for Training Neural Networks," *arXiv:1506.01186v6,* 2017.

[71] F. AI, "PyTorch," [Online]. Available: https://pytorch.org/.

[72] L. Melas-Kyriazi, "efficientnet-pytorch 0.7.0," [Online]. Available: https://pypi.org/project/efficientnet-pytorch/.

[73] M. Sundararajan, A. Taly and Q. Yan, "Axiomatic Attribution for Deep Networks," *arXiv:1703.01365 ,* 2017.

[74] C. McCourt, O. Dolan and G. Gormley, "Malignant Melanoma: A Pictorial Review," *The Ulster Medical Journal,* vol. 83, no. 2, pp. 103-110, 2014.

[75] E. Nasr-Esfahani, S. Samavi, N. Karimi, S. Soroushmehr, M. Jafari, K. Ward and K. Najarian, "Melanoma Detection by Analysis of Clinical Images Using Convolutional Neural Network," *Conference: 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), At Orlando, Florida, USA,* doi: 10.1109/EMBC.2016.7590963, 2016.

[76] H. A. Haenssle, C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, A. Blum, A. Kalloo, A. B. H. Hassen, L. Thoma, A. Enk and L. Uhlmann, "Man against machine: diagnostic performance of adeep learning convolutional neural network fordermoscopic melanoma recognition in comparisonto 58 dermatologists," *Annals of Oncology,* vol. 29, no. 8, pp. 1836 –1842, doi: 10.1093/annonc/mdy166, 2018.

[77] S. S. Han, M. S. Kim, W. Lim, G. H. Park, I. Park and S. E. Chang, "Classification of the Clinical Images for Benign and Malignant Cutaneous Tumors Using a Deep Learning Algorithm," *Journal of Investigative Dermatology,* vol. 138, no. 7, pp. 1529-1538, doi: 10.1016/j.jid.2018.01.028 , 2018.

[78] A. Bhattacharya, A. Young, A. Wong, S. Stalling, M. Wei and D. Hadley, "Precision Diagnosis Of Melanoma And Other Skin Lesions From Digital Images," *AMIA Jt Summits Transl Sci Proc,* pp. 220-226, 2017.

[79] H. Kittler, H. Pehamberger, K. Wolff and M. Binder, "Diagnostic accuracy of dermoscopy," *The Lancet Oncology,* vol. 3, no. 3, pp. 159-165, doi: 10.1016/S1470-2045(02)00679-4, 2002.

[80] Q. Ha, B. Liu and F. Liu, "Identifying Melanoma Images using EfficientNet Ensemble: Winning Solution to the SIIM-ISIC Melanoma Classification Challenge," *arXiv:2010.05351,* 2020.