Skin Cancer Recognition with Novel Deep Learning Methodology on Mobile Platform

Phillip Ly

Department of Computer Science,
California State University,
Fullerton, CA, USA
E-mail: phillipdly@csu.fullerton.edu

Abhishek Verma*

Department of Computer Science,
California State University,
Northridge, CA, USA
E-mail: abhishek.verma@csun.edu
* corresponding author

Doina Bein

Department of Computer Science,
California State University,
Fullerton, CA, USA
E-mail: dbein@fullerton.edu

Abstract: The goal of this research is to create mobile applications that can leverage the power of deep learning to detect skin cancer in the early phase and save lives. In this paper we present: (i) novel deep learning based methodology named as feature extraction with data augmentation and fine-tuning (FEDAFT) to develop compact mobile compatible model that perform effectively in both experimental and real-world situations. (ii) our methodology is based on advanced data augmentation, transfer learning, and fine-tuning techniques and obtained top-1 accuracy of 88.35%, which is better than several of the other researches on skin cancer dataset. (iii) The model is successfully deployed on the iOS and Android mobile systems. (iv) Furthermore, we create a composite dataset from several existing datasets for improved recognition accuracy.

Keywords: Deep Learning; Skin Cancer; Melanoma; Neural Network; CNN; PHDB.

Biographical notes: Phillip Ly received his Master of Science in Computer Science degree from California State University, Fullerton. He received undergraduate education at Santa Ana College and University of California,

Irvine. Currently, his research interests are in deep learning and computer vision.

Abhishek Verma received his Ph.D. in Computer Science from New Jersey Institute of Technology, NJ, USA. His research interests are in data science, big data analytics, machine learning, and deep learning on big datasets.

Doina Bein is Professor of Computer Science at California State University, Fullerton. She received BS and MS from the Al. I. Cuza University, Iasi, Romania, MS and PhD from the University of Nevada, Las Vegas, all in Computer Science. Her research interests include machine learning, Automatic Dynamic Decision-making, Computational Sensing, Distributed Algorithms, Energy-efficient Wireless Networks, Fault Tolerant Data Structures, Fault Tolerant Network Coverage, Graph Embedding, Multi-modal Sensor Fusion, Randomized Algorithms, Routing and Broadcasting in Wireless Networks, Secure Network Communication, Self-stabilizing Algorithms, Self-organizing Ad-hoc Networks, Urban Sensor Networks, Wireless Sensor Networks.

1 Introduction

Skin cancer is a ubiquitous type of cancer in the United States Guy et al. (2015). Based on estimates from American Academy of Dermatology Association one in five Americans will develop skin cancer in their lifetime. Skin cancer outnumbered all cancers including top cancers such as prostate cancer, breast cancer, lung cancer, and colorectal cancer American Cancer Society (2017). For instance, malignant melanoma is a dangerous type of skin cancer that can be benefited from early detection and widespread availability of effective diagnostic tools. Therefore, the main objectives of our research are to create a new type of skin lesion dataset and use powerful deep learning techniques in order to generate Convolutional Neural Network (CNN) models that have high accuracy of melanoma detection in the real world and can be deployed on various mobile systems such as iOS and Android.

People that have malignant melanoma must have the ability to detect it at an early stage and seek professional treatment in order to survive. If melanoma is detected in the early stage, people who contracted it will have 99% chance to survive as opposed to 14% survival rate in more advanced stages SCF (2023). Furthermore, preventive care is necessary to reduce the chance of skin cancer from occurring in the first place. The American Cancer Society recommends people to wear sunscreen, sunglasses, and long sleeve shirts to protect themselves from the ultraviolet rays from the sun ACS, UV (2019). However, millions of new cases of skin cancer still occur every year, thus we must also improve the diagnostic care for skin cancer. According to the skin cancer foundation, the yearly cost to treat skin cancer in the United States is approximately $8.1 billion SCF (2023). Therefore, it is very important to use the powerful capability of deep learning in order to improve diagnosis of skin cancer so that it is more affordable and available to everybody.

The primary goal of our research is to create mobile applications that can leverage the power of deep learning to detect malignant melanoma in the early phase and save lives. In this research, we present deep learning methodology that entails the construction of a new skin lesion dataset that is a combination of several existing datasets. We generate a new compact CNN model for the mobile platform with the uses of modern transfer learning and regularization techniques with excellent performances on computationally limited mobile

systems in the real world. Being able to do the inferencing on the edge devices makes the system more secure to use since sensitive patient data is not transmitted from mobile device to the remote server.

Moreover, based on estimates from World Economic Forum there are approximately 8.6 billion mobile device subscriptions worldwide in the year 2021 WEF (2022). Therefore, due to the ubiquity of mobile systems, deep learning models that can be deployed on mobile platforms can help to extend the reach of crucial diagnostic care for malignant melanoma detection. The next major step is to create a mobile app for skin cancer classification that is accessible to everyone and serves as an early warning tool for melanoma. Furthermore, dermatologists can also incorporate this app as an additional step in their standard verification process of diagnosing malignant melanoma.

The proposed deep learning methodology can generate mobile compatible models by rendering and training 80,192 high quality images. We performed rigorous experiments to attain top-1 accuracy of 88.35% with the uses of potent feature extraction and data augmentation methods. Additionally, our ChekSkin app is tested in real-world situations in which there are drastic variations in lighting conditions and image quality. We have considered tests in both experimental and real-world settings as important metrics for life-saving mobile applications.

This paper is organized as follows. Section 2 presents the related work for various skin cancer and deep learning researches. Next, we provide the descriptions of various datasets that are used to build a composite dataset, PHDB in Section 3. Section 4 describes the proposed methodology to generate compact deep learning model. Next, the experiment results and discussion are presented in Section 5. Conclusion and future work are presented in Section 6 .

2   Related Work

Transfer learning is a very popular deep learning technique that can be used for skin cancer recognition. In the past, researchers have used big Convolutional Neural Networks (CNNs) such as InceptionV3 Esteva et al. (2017), Microsoft ResNet-152 Han et al. (2018), VGG16 Kalouche (2016), GoogleNet, and VGG-19 Liao (2016) for skin cancer recognition. These CNN architectures are intricate and extensive in size. They have many layers to process various features of the input dataset (in the case of skin lesion datasets, the features can be the asymmetrical shapes, border, color, and diameter) in order to maximize top-1 or top-5 accuracies He et al. (2015); Simonyan and Zisserman (2014); Szegedy et al. (2015). However, these large CNNs have millions of parameters, which may lead to overfitting if the dataset is small and imbalanced Chollet (2016). Deep learning models can be considerable in size and hinder deployment to resource constrained mobile platforms. The input size of images from the skin lesion dataset must be pre-processed to be 224 x 224 pixels in order for the deep learning models to be deployable on popular mobile platforms such as iOS and Android. More recently, traditional machine learning approaches have been successfully applied for detection of cancer gene mutation Mandava et al. (2022). We work with the more efficient CNN architecture, MobileNet, which is specifically designed to accommodate deep learning on mobile systems Howard et al. (2017).

In most cases, if we use imbalanced datasets to train CNN models, then it is probable that the models would be biased toward the larger classes that have higher number of images Kalouche (2016). More specifically, the overall accuracy of CNN models for skin cancer

recognition is greatly influenced by how much the convolutional neural network (CNN) learned about the features of the images in the input dataset during the training process. Regarding skin cancer recognition research, in most cases, CNN usually learned more about the features of the benign class because most skin lesion datasets have higher number of images belong to the benign class as opposed to the malignant class. We need to balance the numbers for benign and malignant images during construction of the skin lesion dataset. Overall, it is intuitive to use a balanced dataset to optimize the overall accuracy of CNN model for skin cancer recognition.

This paper proposes a deep learning methodology that use MobileNet Howard et al. (2017) as the main architecture on mobile systems that utilize convolutional neural networks (CNNs) with various combinations of data augmentation, transfer learning, dropout, and other regularization techniques to solve the skin cancer classification problem. MobileNet's weights are trained on ImageNet, which includes 1.4 million labelled images that are divided into a multitude of categories to represent common objects and scene classes. MobileNet is more suitable for creation of deep learning applications on mobile systems because of its capability to generate deep learning models that are small in size and can be deployed on iOS or Android devices.

Convolutional neural networks need to be trained on large amount of training data to achieve generalization, which refers to the performance of the models on new data that CNNs have not seen before. The fundamental layers of deep convolutional neural networks include convolution, rectified linear unit (ReLU) Nair and Hinton (2010), pooling, and fully connected layers. In principle, these layers represent the building blocks for CNNs. Moreover, they are inspired by the works of Ciresan et al. (2011); Krizhevsky et al. (2012).

2.1   Convolution Layers

The primary purpose of the convolution layer in a typical CNN is to extract intrinsic features from input image. Furthermore, convolution layers facilitate the preservation of spatial relationships between images via a sliding window mechanism. More specifically, convolution layers constitute a set of image filters Gonzalez and Woods (2007). These filters represent the dimension of the patches to be extracted from the inputs, which can be 3 x 3 or 5 x 5. In this paper, the CNNs for skin cancer classification use 3 x 3 filters. Ultimately, these filters allow for the preservation of parallel relationships and produce a linear combination of every pixel value as specified by the size of the filters Moacir et al. (2017).

In the case of CNN for skin cancer classification, the convolution layers (near the input layer) help to preserve the generic feature maps of the input image such as the general shape and border of a given skin lesion image. These generic features can belong to both malignant and benign classes. As a skin lesion image is processed and passes through more layers (convolution and max pooling with ReLU activation), the neural network is able to learn the discriminatory features. For example, the convolution layers (near the output layer) learn the features that determine if a given skin lesion image is either malignant or benign.

2.2   Convolved Feature or Activation Map

A 3x3 filter or kernel is used to detect features. The convolved feature matrix is formed as the filter slides over the image and computes the corresponding dot product for the entire input image. Furthermore, the filter is the feature detector for the input image. The 'feature map' is also called 'activation map' or 'convolved feature'.

2.3   ReLU

All experiments in this paper use the rectified linear unit (ReLU) activation function. It is used in the convolution layers or fully connected layers in the CNNs Nair and Hinton (2010). ReLU is an element wise operation that is applied for every pixel of input skin lesion images. ReLU substitutes every negative pixel value in the activation map by zero and in the case of all positive pixel values, they are mapped using a linear function. ReLU solves the issues of vanishing gradients by introducing non-linearity, there by the network can learn complex patterns, it is computationally efficient in comparison with sigmoid and tanh functions.

2.4   Pooling

The pooling layer is a common layer to construct a convolutional neural network for skin cancer classification. It facilitates the simplification of skin lesion images by reducing the dimensionality of every generated feature or activation map. Furthermore, the pooling layer retains the most valuable information or feature of the skin lesion image (such as shape, color, border, diameter) and move them to subsequent layers for further discovery.

2.5   Fully Connected or Dense Layers

Typically, the output from the last pooling layer is the input for the fully connected layer (FC). The FC layer is a Multi-Layer Perceptron that uses a softmax activation function in the output layer. Furthermore, all neurons in the dense layers have connections with all neurons in previous layer LeCun et al. (1998). The main purpose of a fully connected layer is to perform the appropriate classification based on the processed features and compute the proper probabilities for each class. In this paper, the FC would output the probabilities for malignant or benign image in a range between the 0% to 100%.

2.6   MobileNet

We use the MobileNet architecture extensively as part of proposed methodology in Section 4. The MobileNet architecture is specifically designed to generate models for deployment to mobile platforms due to the use of core layers called depth wise separable convolutions Howard et al. (2017). Sifre Laurent first introduced depth wise separable convolution layers in Sifre and Mallat (2014). CNNs that use separable convolution layers can process data more quickly and require less data to attain better accuracy. The ability to obtain high accuracy with limited data is very important to solve skin cancer classification problem because malignant skin lesion images are rare. Typically, a normal convolution layer filters and combines inputs into new outputs in a single step. Conversely, depth wise separable layers are more efficient than standard convolution layers because they partitioned the filtering and combining steps into two separate layers. This method refers to the factorization of normal convolution layers. It reduces the required computation of CNNs and output smaller sized models. Such optimizations are expected to increase the overall accuracy of CNNs when training on a limited amount of data Chollet (2016).

3   Description of Skin Lesion Dataset

We construct a composite dataset of skin lesion images named 'PHDB'. This dataset comprises of four publicly available datasets: ISIC Archive ISIC (2018), Dermnet NZ DERMNET (2018), MED-NODE Giotis et al. (2015), and PH^2 Teresa et al. (2013). PHDB dataset is balanced, i.e., half of the images are from benign class and other half are from malignant class. We split the dataset into three partitions: 80% for training, 10% for validation, and 10% for testing. We use various data augmentation techniques to increase the size of the training set to 80,192. Details of data augmentation technique can be found in Section 4.

3.1   ISIC Dataset

The International Skin Imaging Collaboration (ISIC) is an academia-industry partnership that was established to facilitate digital skin imaging innovations. The main goal of ISIC is to reduce melanoma mortality worldwide. ISIC's dataset consists of 23,906 images in total. More specifically, there are 19,737 benign images and 2,286 malignant images. HAM10000 Philipp et al. (2018) was added to the ISIC archive in 2018. New images from HAM10000 contains 11,788 high quality images and 10,010 of them are publicly available for academic research.

Skin lesion images in the ISIC dataset can have very large dimensions (6,668 x 4,399) and images in the normal range (1,024 x 768). Moreover, larger images can occupy over 20,000 KB in physical size. Regarding the method of diagnosis, images from ISIC are confirmed via histopathology reports and/or biopsy-proven. Figure 1 shows sample images from the ISIC dataset.

3.2   Dermnet NZ Dataset

The Dermnet NZ dataset contains over 20,000 high quality images of many skin diseases DERMNET (2018). We have manually examined all Dermnet NZ's image and have extracted 234 malignant images to be used in the dataset for training neural networks.

3.3   MED-NODE Dataset

MED-NODE dataset consists of 70 melanoma and 100 nevi cases in total Giotis et al. (2015). Department of Dermatology of the University Medical Center Groningen (UMCG) is the main contributor for the MED-NODE dataset.

3.4   PH^2 Dataset

The PH^2 dataset has approximately 200 dermascopic images for melanocytic lesions Teresa et al. (2013). It includes 40 melanomas, 80 typical nevi, and 80 atypical nevi. Furthermore, professional dermatologists (from Dermatology Service of Hospital Pedro Hispano, Matosingos, Portugal) confirmed diagnosis for each skin cancer image by performing assessment based on various dermoscopic criteria such as colors, pigment network, regression areas, streaks, and globules.
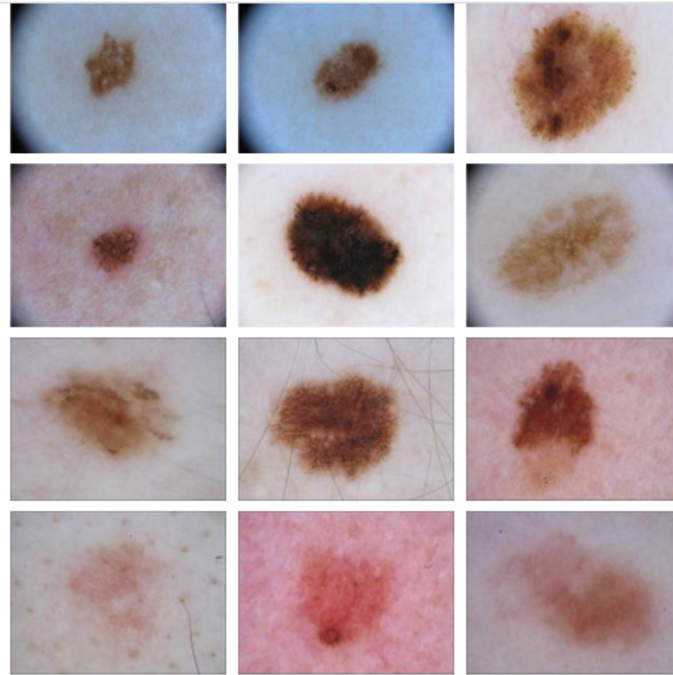
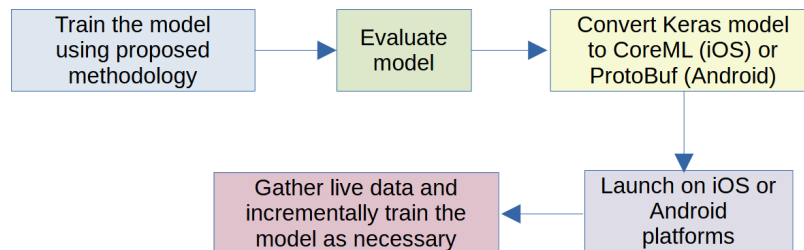**Figure 1**   Sample images from ISIC dataset ISIC (2018).

Figure 2   The overall work flow of the proposed system.

4   Proposed Methodology

The proposed methodology in this paper uses feature extraction with advanced data augmentation and fine-tuning techniques and utilizes transfer learning. The novelty of this paper is reinforced by orchestrating deep learning for skin cancer classification on mobile systems and the final product is a mobile app called ChekSkin. Furthermore, the work in this paper is novel from the perspective of building a composite PHDB dataset for training neural networks. Figure 2 presents the overall work flow of the proposed system. The process starts with training the model using the proposed methodology, the model is then evaluated, next we convert the Keras model to CoreML for deployment on iOS or ProtoBuf for deployment on Android platform, thereafter application runs on the mobile device and performs classification. If the user permits, then the app can gather live data that could be used for incrementally training the model to improve its performance.

4.1   Building Skin Cancer Classification Systems with MobileNet

A highly effective method to implement deep learning for skin cancer classification is to use a pretrained network. A pretrained network is already trained on a large dataset such as ImageNet Russakovsky et al. (2015). ImageNet was created for large-scale image-classification tasks. The ImageNet contains 1.4 million images that are organized into thousands of categories of various objects and scene classes. Thus, a model trained on ImageNet is considered to be a large and generic model such that its weights can be reused to solve different computer vision problems. For example, we can train a CNN on ImageNet (a massive dataset of images that contains mostly objects) and then repurpose the model via transfer learning to recognize cars in images. Such portability and reusability of deep learning models is an important feature that enables effective deep learning techniques to solve skin cancer classification (or the application of deep learning for medical images in general) problem where there is a scarcity of data.

The two major steps to use pretrained convolutional neural network (CNN) include implementation of feature extraction and fine-tuning techniques. Feature extraction entails using representations that are learned by a parent network to extract relevant features from new data or samples. Next, the new network will be trained from scratch and the extracted features will be processed as they represent the inputs for a new classifier.

The major components of typical CNN for image classification tasks consist of the following layers: 1) An input layer as the initial segment; 2) A sequence of convolution layers and pooling layers; 3) A densely connected classifier at the end. An important aspect of feature extraction is to train new data using a convolutional base of a pre-trained network with a new classifier on top. The pre-trained network of the convolutional base can be any popular CNN architecture. We exclusively deal with MobileNet because it can generate high performance compact models that are compatible with mobile systems.

4.2   Feature Extraction With MobileNet Convolutional Base

Feature Extraction with Data Augmentation and Fine-tuning (FEDAFT) deep learning process is depicted in Figure 3. It starts with instantiation of the MobileNet convolutional base model, followed by feature extraction, next the base model is extended by adding densely connected classifier, augmented data is used to train the unfrozen layers followed by the fine tuning-step. Figure 4 shows the visualization of the CNN model architecture using the FEDAFT methodology. Convolutional block 1 has an input layer, 2-D convolutional layer, followed by a batch normalization layer that normalizes its input close to a mean output of zero and the output standard deviation close to one. Next is a ReLU activation layer followed by layers that perform depth wise and point wise convolutions. Convolutional blocks 2-13 are next in sequence followed by the expanded convolutional base consisting of flatten and dense layers.

In-depth details of MobileNet's convolutional base, which can accommodate 3,228,864 parameters in total are shown in Table 1. Furthermore, there are 3,206,976 trainable parameters and 21,888 non-trainable parameters. As illustrated in Table 1, the shape of the final feature map is (None, 7, 7, 1024), where the four values are (batch size, height, width, output channels) respectively. Therefore, this is the precise location for the insertion of the densely connected classifier. The batch size for each epoch is the number of input images for training, it can be assigned just prior to training the model and depending on available memory. Height and width are the dimensionality of the input vector to each layer. The
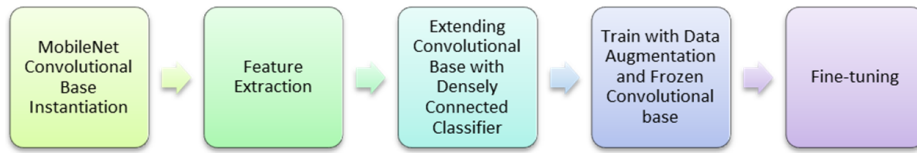
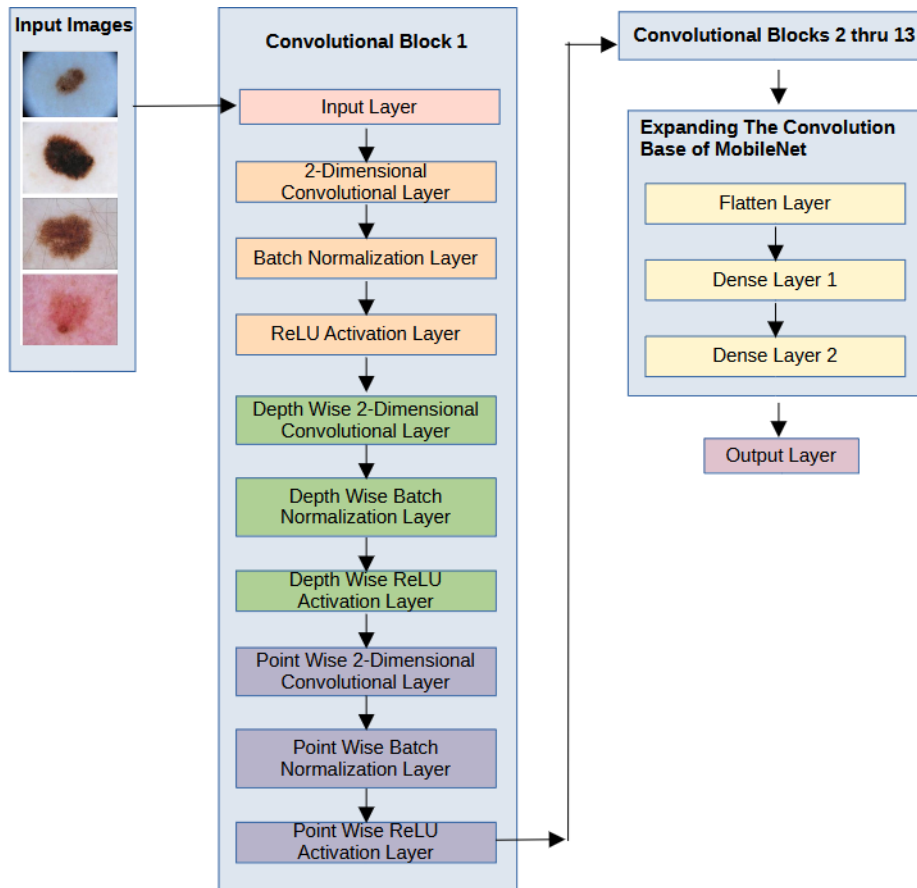**Figure 3** Deep learning steps for the proposed FEDAFT methodology.

Figure 4 Visualization of the CNN model architecture using FEDAFT methodology.

number of output channels is the result of applying the same number of convolutional filters to the input vector and forms the output feature map. Notice that with each successive block of CNN the height and width is reduced, and the number of feature maps increases thereby extracting the relevant features of the data. The parameter count is based on the filter shape, for instance the filter shape for [Block 1] 2-D Convolutional Layer is 3 x 3 x 3 x 32 resulting in 864 parameters for that layer. Further details of the filter shape and parameter count can be found in Howard et al. (2017).

As part of feature extraction, it is important to only reuse the convolutional base of a pre-trained network and not the densely connected classifier of the pre-trained network. The

representations learned by the convolutional base layers are considered to be more generic and reusable. Another crucial aspect of convnets' reusability is based on the depth of layers. For instance, in the case of skin cancer classification, subsequent layers after the input layer tend to extract more generic feature maps such as visual edges, colors, and textures of skin cancer images. Conversely, layers near the densely connected classifier tend to extract more abstract features from skin cancer images such as the symmetry or diameter of a skin lesion. For examples, benign skin lesions are usually symmetrical and are smaller than ¼ inch. Whereas malignant skin lesions are asymmetrical and are larger than ¼ inch. Moreover, the ISIC dataset contains skin cancer images that are quite different than images of common objects and scenes in the ImageNet dataset (which is what MobileNet was trained on). Thus, it is optimal to use only the convolutional base of the pre-trained CNN when performing feature extraction.

4.3   Expanding Convolutional Base

The next step is to extend the convolutional base with a new classifier and then initiate the training session with data augmentation techniques. Data augmentation refers to the generation of additional training data from existing training data. Augmentation of training data can be accomplished via random transformational techniques such as adjusting the skin lesion images using image rotation, horizontal flip, and random brightness. Thus, the model would be exposed to many aspects of the current training data and attain higher accuracy on validation and test data. Data augmentation is a crucial component that facilitates the mitigation of overfitting. Overfitting usually occurs when a deep learning model is trained on a small amount of data. Thus, it starts to memorize the internal representations of the training data. The memorization of training data's representations enables deep learning models to perform well during training (high training accuracy) but they would perform poorly (low validation accuracy) on the validation and test sets.

Table 1 gives the architecture of convolutional base of MobileNet (version 1.0, 224 x 224 images). Rows colored in blue show the architecture of the expanded convolutional base, the flatten layer changes the 7 x 7 x 1,024 input to 50,176 output. The densely connected classifier contains 12,845,569 parameters in total. This is a considerable number of parameters that are available for training. Regularization techniques must be used in future steps to prevent overfitting.

4.4   Freezing Convolutional Base

Before the training of neural network is initiated, the most important maneuver that we must perform is to freeze the convolutional base. This maneuver facilitates the prevention of large weight updates to a layer or a set of layers during training. If the convolutional base is not frozen when training is in session, propagation of large weight updates can occur and effectively annihilate learned representations.

4.5   Data Augmentation

The configurations to perform data augmentation using Image Data Generator in Keras (2018) are as follows: rescale = 1/255 to rescale an input image, rotation range = 60 which is the rotation angle in degrees, width shift range = 0.05 that is the shift in x direction, height shift range = 0.05 that is the shift in y direction, shear range = 0.01 that is the distortion of image perception angle along an axis, zoom range = 0.1, horizontal flip = True to flip an

image horizontally, vertical flip = True to flip an image vertically, and fill mode = 'nearest' to fill an empty area in an image with the pixel value of its nearest neighbor. In the fit generator method, we can specify the number of epochs for training. An epoch is one training cycle in which all the training data is processed. In this case, epochs = 50.

These configurations for data augmentation are set randomly as training proceeds from one step in an epoch to the next. For example, during step 1 in epoch 1, the data augmentation hyperparameters could be set to: rotation range = 59, width shift range = 0.04, height shift range = 0.04, shear range = 0.01, zoom range = 0.1, horizontal flip = True, vertical flip = True. When training proceeds to step 2, the image augmentation might be randomly reconfigured as: rotation range = 40, width shift range = 0.03, height shift range = 0.02, shear range = 0.01, zoom range = 0.05, horizontal flip = True, vertical flip = False. In order words, rotation range = 60 means that at each step in an epoch, a skin lesion image is randomly rotated within the range of 0 to 60. Therefore, if the duration of training is prolonged, then more data can be generated for a CNN to train on these varied representations.

4.6 Fine-tuning

The fine-tuning technique is widely used with feature extraction to further improve the accuracy of deep learning models. Fine-tuning entails unfreezing the most useful layers of the frozen convolutional base (from previous step) and then train these newly activated layers with the fully connected classifier end to end. In this case, the last few convolutional blocks near the end of MobileNet's convolutional base represent the most useful layers that are relevant to the skin cancer classification problem. More specifically, earlier layers of the convolutional base process generic features, whereas layers near the end handle more specialized features. Thus, these specialized features processing layers are more useful and can be repurposed as needed to solve the skin cancer classification problem.

As a very important part of the implementation, we must unfreeze the MobileNet's convolutional base from the [Block 12] depth wise 2-D convolutional layer as shown in Table 1. Regarding the training details, RMSProp optimizer is used with a very small learning rate (0.00001) to minimize the magnitude of changes to the representations of the active layers during the fine-tuning process. Otherwise, large updates would effectively destroy the learned representations.

4.7 Combining Data Augmentation With Transfer Learning for Mobile Platforms

We train deep convolutional neural network (CNN) with various transfer learning and data augmentation techniques. Contemporary image classification CNNs consist of millions of parameters that are vulnerable to overfitting. Furthermore, training CNNs from scratch requires a massive amount of computing power and a large amount of labeled training data (high quality). Thus, it is preferable to use innovative transfer learning techniques (via bottleneck features) in order to leverage the image feature extraction module with the instance of MobileNet architecture (Version 1.0) that is pretrained on ImageNet. Ultimately, performing transfer learning with data augmentation techniques using MobileNet architecture is a powerful technique. With such a complementary combination of modern deep learning methods, we have successfully deployed compact models, in the form of protocol buffer files (.pb), to mobile systems on Android and iOS platforms.

Table 1   The Detailed CNN Model Architecture Using FEDAFT Methodology

Layer Type in MobileNet v1 Covolutional Base	Output Shape (*batch size, height, width, output channels) *assigned at runtime	Parameter Count
[Block 1] Input Layer	(None, 224, 224, 3)	0
[Block 1] 2-D Convolutional Layer	(None, 112, 112, 32)	864
[Block 1] Batch Normalization Layer	(None, 112, 112, 32)	128
[Block 1] ReLU Activation Layer	(None, 112, 112, 32)	0
[Block 1] Depth Wise 2-D Convolutional Layer	(None, 112, 112, 32)	288
[Block 1] Depth Wise Batch Normalization Layer	(None, 112, 112, 32)	128
[Block 1] Depth Wise ReLU Activation Layer	(None, 112, 112, 32)	0
[Block 1] Point Wise 2-D Convolutional Layer	(None, 112, 112, 64)	2048
[Block 1] Point Wise Batch Normalization Layer	(None, 112, 112, 64)	256
[Block 1] Point Wise ReLU Activation Layer	(None, 112, 112, 64)	0
… [Blocks 2-11] ...	…	…
[Block 12] Depth Wise 2-D Convolutional Layer	(None, 7, 7, 512)	4,608
[Block 12] Depth Wise Batch Normalization Layer	(None, 7, 7, 512)	2,048
[Block 12] Depth Wise ReLU Activation Layer	(None, 7, 7, 512)	0
[Block 12] Point Wise 2-D Convolutional Layer	(None, 7, 7, 1024)	524,288
[Block 12] Point Wise Batch Normalization Layer	(None, 7, 7, 1024)	4,096
[Block 12] Depth Wise ReLU Activation Layer	(None, 7, 7, 1024)	0
[Block 13] Depth Wise 2-D Convolutional Layer	(None, 7, 7, 1024)	9,216
[Block 13] Depth Wise Batch Normalization Layer	(None, 7, 7, 1024)	4,096
[Block 13] Depth Wise ReLU Activation Layer	(None, 7, 7, 1024)	0
[Block 13] Point Wise 2-D Convolutional Layer	(None, 7, 7, 1024)	1,048,576
[Block 13] Point Wise Batch Normalization Layer	(None, 7, 7, 1024)	4,096
[Block 13] Depth Wise ReLU Activation Layer	(None, 7, 7, 1024)	0
Total Parameters: 3,228,864		
MobileNet v1 Convolutional Base	(None, 7, 7, 1024)	3,228,864
Expanded Convolutional Base is Below	**(batch size, output channels)**	
Flatten Layer	(None, 50176)	0
Dense Layer 1	(None, 256)	12,845,312
Dense Layer 2	(None, 1)	257
Total Parameters: 16,074,433		
Trainable Parameters: 12,845,569		
Non-Trainable Parameters: 3,228,864		

4.8   Additional Data Augmentation Techniques in Context of Real-World Situations

In addition to some of the default data augmentation techniques (random brightness and random contrast), additional data augmentation techniques such as random hue and random saturation could be incorporated into the system. Random brightness is especially important to mimic the drastic variations in lighting conditions in the real world (i.e., inside a house and outside on a sunny day). The flip up down and flip left right hyperparameters are crucial

in augmenting the PHDB dataset. The ability to use more data augmentation techniques is crucial because it helps to alleviate overfitting problem and enlarges the size of the PHDB dataset for training during runtime.

Dermatologists often perform additional procedures for examination such as adjusting the levels of illumination (for better view) and wiping skin lesions with rubbing alcohol. Thus, data augmentation techniques must be used to simulate examination conditions that occur during skin cancer classification procedures in the real world. We assume users would use the skin cancer classification app, ChekSkin, in real-world environment where there are variations in the lighting conditions and other external factors that can negatively affect image quality. Thus, the CNN for skin cancer classification must not only perform well in experimental settings but it also need to be flexible and make the right prediction in suboptimal situations. Ultimately, it is essential to consider accuracies obtained from both experimental and real-world tests as important metrics for life-saving mobile applications.

After the previous steps are implemented, a deep learning model in the form of a protocol buffer file (.pb) is generated. This model is less than 30MB when trained on the PHDB dataset. Protocol buffers are Google's extensible procedure for serializing structured data. Finally, we deploy the protocol buffer file to a skin image classification (in real time) app and test the system on android and iOS platforms.

4.9 Deploying Model to The Mobile Platform

The previously mentioned steps will generate deep learning model in h5 format. Next, we must convert Keras (in .h5 format) models to coreML models. This would ensure successful deployment of Keras' models to the iOS platform, which is a popular mobile operating system from Apple Inc. After successful conversion to coreML models, it is straightforward to deploy a coreML model on an iOS app to perform skin cancer classification in real-time. Screenshots of a simple mobile app for skin cancer classification are provided in Section 5. Ultimately, this is a very important step to transform Keras deep learning models into an actual product that can potentially reach the hands of many people.

5 Experiment Results and Discussion

5.1 Training Configurations

The learning rate of the FEDAFT model is 0.00002 with RMSprop as the optimizer. Moreover, the configurations for data augmentation are as follow: rescale = 1/255, rotation range = 60, width shift range = 0.05, height shift range = 0.05, shear range = 0.01, zoom range = 0.1, horizontal flip = True, vertical flip = True, and fill mode = 'nearest'. Data augmentation techniques such as rotation range, horizontal flip, and vertical flip are used to simulate real world situations in which users might use the ChekSkin app from different camera angles. Other hyperparameters such as width shift range. height shift range, and zoom range are used to ensure a certain level of uniformity for the dataset because some skin lesion images in the PHDB dataset are not properly centered in the same way.

5.2 Fine-Tuning Configurations

Early stopping is used to monitor the model's training accuracy and terminate the training process when the model overfits. RMSprop optimizer is used with a learning rate of 0.00001.
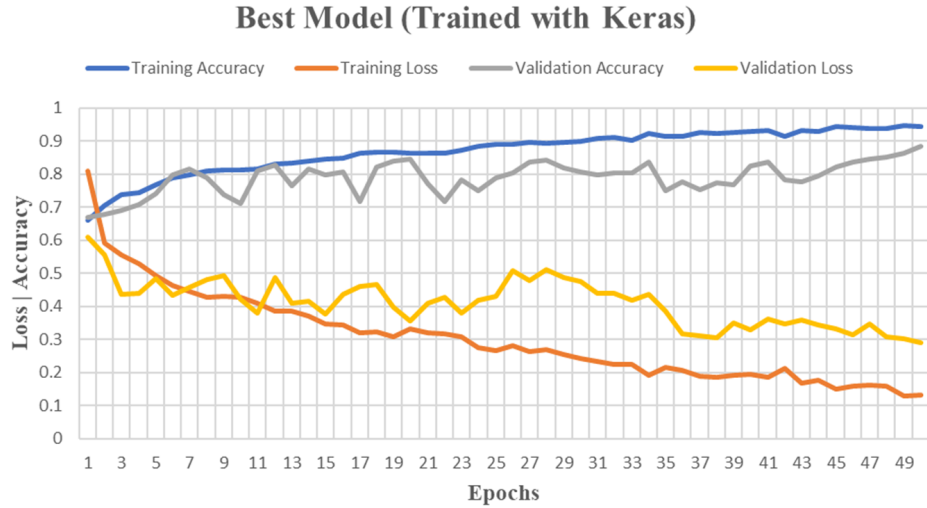
Best Model (Trained with Keras)

Figure 5 Training and validation performance of the proposed FEDAFT methodology.

Trainable layer for fine-tuning process is performed from the [Block 12] depth wise 2-D convolutional layer (see Table 1). Rows that are highlighted in red represent the layers of the MobileNet convolutional base to initiate the fine-tuning process. Likewise, rows that are highlighted in blue are the final layers (dense) for fine-tuning. The total number of trainable parameters is 12,845,569, which is a considerable number. However, the previously mentioned data augmentation techniques facilitate the alleviation of overfitting by enlarging the PHDB dataset to 80,192 images.

5.3   Results for Proposed FEDAFT Methodology

Figure 5 shows the training and validation performance of the proposed FEDAFT methodology with a top-1 accuracy of 88.35%. Both training and validation accuracy curves remained in close proximity throughout the duration of training. Such propinquity indicated that the model did not overfit. Overall, the model in Figure 5 was trained for 50 epochs. It is necessary to extend the duration for training because the longer the training time enables more generation of training data to train a CNN. The mixture of data augmentation methods has helped improve the performance.

Table 2 presents the results of our proposed methodology and results improve upon the previous researches in Esteva et al. (2017); Kalouche (2016); Han et al. (2018); Lopez et al. (2017). As shown, our model has the highest accuracy at 88.35% (top-1 accuracy) and is trained using transfer learning on a balanced dataset. Figure 6 shows confusion matrix of the proposed deep learning based FEDAFT methodology. As shown in the confusion matrix 90.4% of the malignant images and 86.8% of the benign images were correctly recognized. At 29 megabytes (MB) in size, our proposed deep learning model is small in size and deployable with a small memory footprint on resource constrained environments such as the iOS or Android systems.

Figure 7 shows situations in which the skin lesion images are tested in optimal imaging conditions, i.e., controlled environment using our proposed methodology. The model is able to correctly predict the two benign images with a high confidence level of 98.6% and

Table 2 Comparison of Performance of Author's Proposed Methodology With Other Researches from Skin Cancer Recognition That Use Transfer Learning

Author	Classification Accuracy	Neural Network	Input Size (Pixels)	Dataset Type	Size of Model (MB)	Mobile Platform
Proposed Methodology	**88.35%**	**MobileNet**	**224 x 224**	**Balanced**	**29**	**iOS and Android**
Esteva et al. (2017)	72.10%	Inception V3	299 x 299	Imbalanced	Not reported	Not reported
Lopez et al. (2017)	81.33%	VGGNet	224 x 224	Balanced	Not reported	Not reported
Kalouche (2016)	78.00%	VGG - 16	256 x 256	Imbalanced	Not reported	Not reported
Han et al. (2018)	57.30%	ResNet - 152	224 x 224	Imbalanced	Not reported	Not specified

		Malignant	123	13
Actual		Benign	18	**118**
			Malignant	Benign
			Predicted	

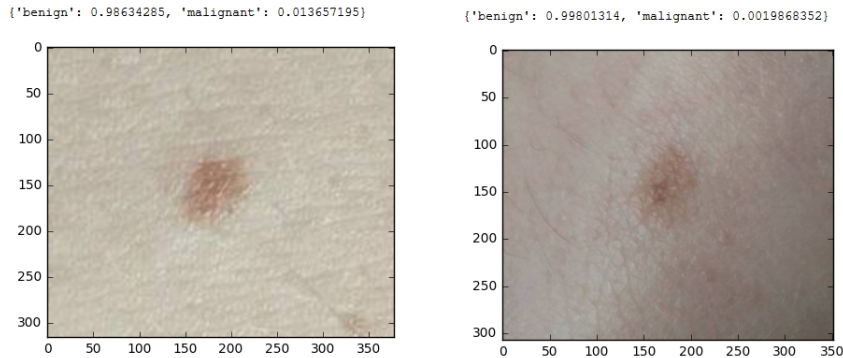**Figure 6** Confusion matrix of the proposed deep learning based FEDAFT methodology.

Figure 7 Tests of skin lesion images in optimal imaging conditions, i.e., controlled environment using proposed methodology.

99.8%. Figures 8 and 9 present various illustrations of tests in suboptimal conditions in the real world, notice that the model is able to correctly predict both the malignant and benign categories with a high level of confidence. Tests in optimal condition are performed via python code in Jupyter notebook, whereas tests in suboptimal conditions are carried out with the ChekSkin app on the iOS platform. Figures 7, 8, and 9 contain correct predictions for both malignant and benign cases that are verified via comparisons with official metadata of the ISIC dataset.

5.4 Testing ChekSkin

Figure 10 shows the ChekSkin app in action. All predictions are correct, the image used is the birthmark of a person, it is benign via confirmations from actual dermatologists and doctors. In the left image, person is wearing a beanie, but ChekSkin was flexible enough and correctly classified the birthmark. It is important to observe the variation in lighting conditions, which has a profound effect on the percentages for predictions, but the top-
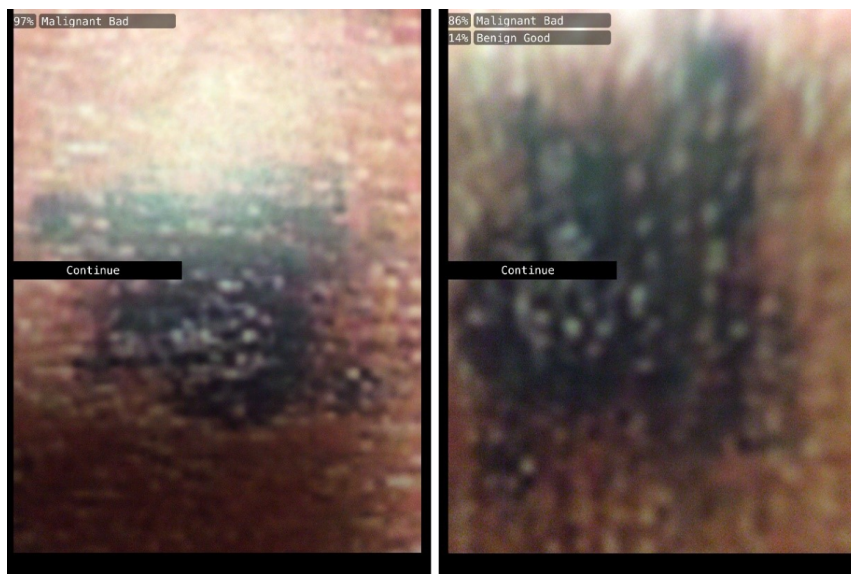
Figure 8 Tests of malignant skin lesion images in uncontrolled real-world situations using proposed methodology.
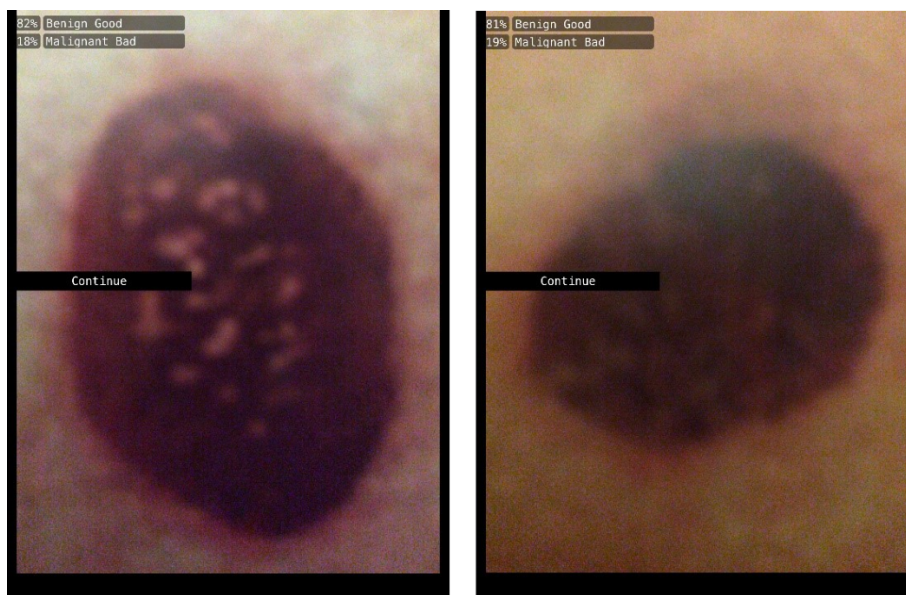

Figure 9 Tests of benign skin lesion images in uncontrolled real-world situations using proposed methodology.

1 accuracy is still higher than 50% in both cases. These examples represent the uses of ChekSkin in extreme real-world conditions. In most cases, ChekSkin's predictions are more stable as shown in Figures 8 and 9. Moreover, the presences of eyebrows, eyelashes, and hair pose significant challenges to the ChekSkin app, but it still managed to correctly identify

Figure 10 Testing ChekSkin mobile app on a benign birthmark.

the birthmark as benign. Furthermore, the 'Good' and 'Bad' labels are added to the original labels so that the prediction labels are understandable or user-friendly to more people.

6 Conclusion and Future Work

This paper presented a novel deep learning based methodology to develop compact mobile compatible model that performed effectively in both experimental and real-world situations. Our methodology is based on advanced data augmentation, transfer learning, and fine-tuning techniques and performed better than several of the other researches on skin cancer dataset. Additionally, the model is successfully deployed on the iOS and Android mobile systems. We created a composite dataset from several existing datasets for improved recognition accuracy.

A critical area of future work is to expand the dataset in order to train models with higher accuracy. Another important area of future work is to improve the performance of the models in uncontrolled conditions. Furthermore, advanced deep learning methodologies such as fusion of multiple models to improve the recognition performance could be considered as future work.

References

G. Guy, et al., "Vital signs: Melanoma incidence and mortality trends and projections—United States, 1982–2030," MMWR Morb Mortal Wkly Rep, vol. 64, no. 21, pp. 591-596, 2015.

American Cancer Society. (2017). "Don't Fry: Preventing Skin Cancer," [Online]. Available: https://www.cancer.org/research/infographics-gallery/skin-cancer-prevention.html.

American Cancer Society. (2019). "How Do I Protect Myself from Ultraviolet (UV) Rays?," [Online]. Available: https://www.cancer.org/cancer/risk-prevention/sun-and-uv/uv-protection.html.

Skin Cancer Foundation. (2023). "Skin Cancer Facts & Statistics," [Online]. Available: https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/.

World Economic Forum. (2022). "Charting The Rise of Mobile Device Subscriptions Worldwide," [Online]. Available: https://www.weforum.org/agenda/2022/10/mobile-device-subscription-rise-technology/.

A. Esteva, et al., "Dermatologist - level classification of skin cancer with deep neural networks," Nature, vol. 542, pp. 115-118, 2017.

S. Han, et al., "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," Journal of Investigative Dermatology, Vol. 138 (7), pp. 1529-1538, 2018.

S. Kalouche, "Vision-Based Classification of Skin Cancer using Deep Learning," 2016. [Online]. Available: https://www.semanticscholar.org/paper/Vision-Based-Classification-of-Skin-Cancer-using-Kalouche/b57ba909756462d812dc20fca157b3972bc1f533.

H. Liao, "A Deep Learning Approach to Universal Skin Disease Classification," 2016. [Online]. Available: https://pdfs.semanticscholar.org/af34/fc0aebff011b56ede8f46ca0787cfb1324ac.pdf.

K. He, et al., "Deep residual learning for image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1512.03385.

K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: https://arxiv.org/abs/1409.1556.

C. Szegedy, et al., "Rethinking the inception architecture for computer vision," 2015. [Online]. Available: https://arxiv.org/abs/1512.00567.

F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2016. Available: https://arxiv.org/abs/1610.02357

G. Howard, et al. (2017). "Mobilenets: Efficient convolutional neural networks for mobile vision applications." Available: https://arxiv.org/abs/1704.04861

V. Nair and G. Hinton. (2010). "Rectified linear units improve restricted Boltzmann machines," In Proceedings of the 27th International Conference on Machine Learning, pp. 807–814.

D. C. Ciresan, et al. (2011). "High performance convolutional neural networks for image classification," In International Joint Conferences on Artificial Intelligence, pp. 1237–1242.

A. Krizhevsky, et al. (2012). "ImageNet classification with deep convolutional neural networks". In Advances in neural information processing systems, pp. 1097–1105.

R. Gonzalez and R. Woods, (2007). Digital Image Processing. 3rd ed. Pearson.

P. Moacir, et al. (2017). "Everything you wanted to know about deep learning for computer vision but were afraid to ask". In SIBGRAPI Conference on Graphics, Patterns and Images Tutorials, pp. 1–25.

Y. LeCun, et al. (1998). "Gradient-based learning applied to document recognition." Proceedings of the IEEE, 86(11), 2278-2324.

L. Sifre and S. Mallat. (2014). "Rigid-motion scattering for texture classification." Accessible: https://arxiv.org/abs/1403.1687

ISIC Archive. [Online]. Available: https://www.isic-archive.com/.

I. Giotis, et al. (2015). "MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images," Expert Systems with Applications, 42 (2015), 6578-6585.

DermNet NZ (New Zealand Dermatological Society). (2018). "Dermnet Skin Cancer Dataset." Accessible: https://www.dermnetnz.org/topics/skin-cancer

M. Teresa, et al. (2013). "PH^2 - A dermoscopic image database for research and benchmarking." 35th International Conference of the IEEE Engineering in Medicine and Biology Society, July 3-7, 2013, Osaka, Japan.

T. Philipp, et al. (2018). "The HAM10000 Dataset: A Large Collection of Multi-Source Dermatoscopic Images of Common Pigmented Skin Lesions." Accessible: https://arxiv.org/abs/1803.10417

O. Russakovsky, et al. (2015). "ImageNet large scale visual recognition challenge." International Journal of Computer Vision, 115(3), 211–252.

Keras. (2018). Accessible: http://keras.io

A. R. Lopez, et al., "Skin lesion classification from dermoscopic images using deep learning techniques," 2017 13th IASTED International Conference on Biomedical Engineering (BioMed), Innsbruck, Austria, 2017, pp. 49-54.

J. S. Mandava, et al., "Machine Learning for Classification of Cancer Dataset for Gene Mutation Based Treatment," the 19th International Conference on Information Technology: New Generations, Apr. 11-12, 2022, Virtual.