



Multi-Features Extraction Based on Deep Learning for Skin Lesion Classification

Samia Benyahia, Boudjelal Meftah, Olivier Lézoray

► To cite this version:

Samia Benyahia, Boudjelal Meftah, Olivier Lézoray. Multi-Features Extraction Based on Deep Learning for Skin Lesion Classification. *Tissue and Cell*, 2022, 74, pp.101701. 10.1016/j.tice.2021.101701 . hal-03457466

HAL Id: hal-03457466

<https://hal.science/hal-03457466>

Submitted on 30 Nov 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-Features Extraction Based on Deep Learning for Skin Lesion Classification

Samia Benyahia^a, Boudjelal Meftah^b and Olivier Lézoray^c

^aDepartment of Computer Science, Faculty of Exact Sciences, University of Mascara, Mascara, Algeria

^bLRSBG Laboratory, University of Mascara, Mascara, Algeria

^cNormandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France

ARTICLE INFO

Keywords:

Feature extraction

Classification

Skin lesion

Convolutional Neural Networks

Dermoscopy images

ABSTRACT


For various forms of skin lesion, many different feature extraction methods have been investigated so far. Indeed, feature extraction is a crucial step in machine learning processes. In general, we can distinct handcrafted and deep learning features. In this paper, we investigate the efficiency of using 17 commonly pre-trained convolutional neural networks (CNN) architectures as feature extractors and of 24 machine learning classifiers to evaluate the classification of skin lesions from two different datasets: ISIC 2019 and PH2. In this research, we find out that a DenseNet201 combined with Fine KNN or Cubic SVM achieved the best results in accuracy (92.34% and 91.71%) for the ISIC 2019 dataset. The results also show that the suggested method outperforms others approaches with an accuracy of 99% on the PH2 dataset.

1. Introduction

Skin lesions are defects located on the surface of the skin or under the skin. They can be classified into two categories. First: benign skin tumors, that are lesions such as moles (nevi) or cysts, that are not particularly dangerous for the body. Second: malignant tumors, that refer to cancerous skin lesions such as melanoma, squamous cell carcinoma, basal cell carcinoma, etc. Skin lesions are widespread but it is complicated to characterize them, and the automatic recognition of malignant tumors from dermoscopy images is still a challenging task. This is why an effective machine-learning method is necessary for the identification of dermoscopic lesions. In recent years, deep learning has gained popularity in many image classification tasks since features can be learned from data automatically. The application of deep learning to skin lesion detection is an active area of research Celebi, Codella and Halpern (2019). Therefore, it is possible to acquire state-of-the-art performance by using deep trained models.

Deep learning roots in neuroscience. Inspired by neurons in the human brain, its concept comes from Artificial Intelligence (AI) techniques derived from machine learning (Goodfellow, Bengio and Courville, 2016). Deep learning proposes a learning method made of multiple layers of neurons interconnected with different weights and activation functions to model the relationship between input and output data. Deep learning is very interesting to acquire new feature representations on large amounts of data with a deep architecture having many hierarchical hidden layers with nonlinear modules. This enables to transform the raw input data with a suitable higher representation and to achieve feature extraction as well as classification. Deep learning layers are arranged into three types: the input layer presents the input data, the hidden layers transform the information, and the final layer of the network will then bring together the different information to deduce an answer. Deep learning has several architectures: Convolutional Neural Networks (CNNs), Long Short Term Memory Networks (LSTMs), Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs), Multilayer Perceptrons (MLPs), Deep Belief Networks (DBNs), Restricted Boltzmann Machines (RBMs), Autoencoders (AE). The most widely known are convolutional neural networks (CNN). Convolutional Neural Networks known as ConvNets or CNNs were reported by (LeCun, Jackel, Bottou, Cortes, Denker, Drucker, Guyon, Muller, Sackinger, Simard and Vapnik, 1995). It is a deep learning architecture based on a feedforward multi-layers hierarchical neural network. It uses a mathematical operation called convolution, that permits to extract features from an image with trainable filters. CNN is composed of two main components:

*Corresponding author

 boudjelal.meftah@univ-mascara.dz (B. Meftah)

ORCID(s):

- Hierarchical feature extraction component: divided into multiple layers where each layer learns from the previous layer. The convolutional feature extraction layers aims at detecting local features in input images. They are followed by non-linear layers, that apply an activation function. Then, pooling layers used to reduce the feature space at each level of the architecture. One can also use specific layers to enhance the learning such as dropout layers.
- The classification component is used to yield response. It consists of one or more fully connected layers and a final softmax layer.

Deep learning has recently received much interested in skin lesion classification. Indeed, skin lesion classification methods usually needed the extraction of specific handcrafted features such as statistical pixel-level features, shape features, texture features, and relational features from the images. A limitation of these traditionally employed feature extraction approaches is their dependence to a priori information on the classification task, making their design very complex. Many researchers have demonstrated deep learning efficiency as a feature extraction method in recent years (Kraus, Grys, Ba, Chong, Frey, Boone and Andrews, 2017). Moreover, many works in different tasks (Govindaswamy, Montague, Raicu and Furst, 2020), (Wang, Zhang and Hao, 2019), (Basly, Ouarda, Sayadi, Ouni and Alimi, 2020), (Alzubaidi, Fadhel, Al-Shamma, Zhang and Duan, 2020), (Aurelia, Rustam, Wibowo and Setiawan, 2020), (Mu and Qiao, 2019), (Suganthi and Sathiaselalan, 2020), (Oltu, Güney, Dengiz and Ağildere, 2021), (Bodapati and Veeranjanyulu, 2019), (Karungaru, Dongyang and Terada, 2021), (Öznur Özaltın and Özgür Yeniay, 2021) show the effectiveness of using ML classifier to classify the data based on features extracted through deep CNN compared to end-to-end deep learning.

Convolutional neural networks (CNN) have been widely applied, leading to improved classification. However, deep trained models were not used in an end-to-end manner, but only to extract features. The feature extraction process can be explained as a transition from raw images into N-dimensional vectors, where N is a hyper-parameter. The major difference between convolutional neural networks and conventional machine learning methods is that CNNs learn image features directly without manual feature extraction.

How to define the efficiency of classifiers is a challenge that affects which features are important for describing the interest patterns and skin lesion recognition is no exception to this rule.

The benefit of using a feature extractor is the ability to choose whichever classifier will best fit the data considering the problem, as well as the ability to combine the CNN with the different classifiers and the integration of innumerable characteristics.

CNN is considered as a feature extractor solution from images due to the difficulty of combining the image data with classifiers. The main reason to consider CNN as the best feature extractor is that the CNN obtains more characteristics compared to other methods. It has the ability to capture the higher quality and powerful features from images data and a large volume of raw data as well as the improvement of accuracy in less time. Choosing the right classifier and the right CNN architecture as feature extractors have always been a challenging task.

The answer to this question is the aim of writing this article, trying to acquire the features of dermoscopic images with a pre-trained CNN model and injecting the obtained characteristics into several classifiers. Extracted features will be classified using various algorithms: k-nearest neighbor (KNN), decision tree, Linear Discriminant, Naive Bayes, hybrid kernel-based SVM and finally ensemble classifiers. We have used 17 commonly pre-trained CNN architectures as feature extractors and 24 machine learning classifiers to evaluate the classification of skin lesions from two different datasets ISIC 2019 and PH2.

The next section exposes some works that supports the two-step pipeline of deep-learning feature extraction followed by classifier.

In (Govindaswamy et al., 2020), the authors employed a CNN for predicting physician gaze, where they focused on comparing the hand-crafted features and CNN-features extractor, in addition to analyzing the impact of feature extraction and fully connected layers in an end-to-end CNN model. The authors used the VGG16 as a feature extractor and a K-Nearest Neighbor and a Random Forest (RF) as a classifier. They concluded that the CNN features extractor showed a significant improvement over hand-crafted features with high-level features from the last convolutional layer of the CNN, also that the power of CNN comes from the features extraction part than the fully connected layers which can be replaced by another classifiers depending on the application.

In (Wang et al., 2019), the authors proposed to combine the CNN and the Extreme Learning Machine (ELM) algorithm to recognize Synthetic Aperture Radar (SAR) images The CNN model is considered as the feature extractor due to its high degree of invariance in extracting complex features of images, regardless any forms of deformation of

the image, and the ELM as a recognizer. The experiment result shows that the algorithm can alleviate the overfitting problem, speed up the convergence of the network, and reduce the time of the experiment.

The authors in (Basly et al., 2020) used a pre-trained CNN ResNet architecture as feature extractor combined with SVM classifier for human activity recognition task where they achieved a good recognition performance compared to state-of-art methods.

The authors in (Alzubaidi et al., 2020) have utilized three deep learning models as features extractor with multiclass SVM classifier. Where they obtained a high degree of accuracy with the proposed deep learning features extractor model compared to other classifiers.

In (Aurelia et al., 2020), the authors proposed a research for the performance of Convolutional Neural Network, and Convolutional Neural Network-Support Vector Machine with several kernel functions for classification of colon cancer. The results obtained show the performance of CNN-SVM Linear Kernel with the best accuracy compared to CNN.

The authors in (Mu and Qiao, 2019) proposed a method of image classification based on CNN and SVM. They extracted vital features and reduced redundant by preprocess the original images by using CNN. The extracted features will be injected to an optimized SVM model and evaluated on a mixed image set taken from Caltech image archive. The experimental results demonstrate the effectiveness of CNN features extractor with SVM classifier.

In (Suganthi and Sathiaselvan, 2020), the authors proposed a comparative study for the efficiency for classifying the image with different aspects of machine learning and CNN. They concluded that researches from different fields consider the combination of CNN model with different ML is more efficient than the normal CNN.

The authors in (Oltu et al., 2021) proposed a study for diagnosing a tuberculosis disease using CXR images based on VGG16, MobileNet networks as feature extractors and SVM is used as the classifier. The obtained results have outperformed most of the previous results.

The authors in (Bodapati and Veeranjanyulu, 2019) used CNN for two different tasks. In task one, they proposed to use CNN for feature extraction and classification. Contrariwise, in task two they proposed to use CNN for features extraction then using SVM for classification. From the experimental studies, they observed that the performance of SVM classification based on CNN features is better than the results obtained from using CNN for feature extraction and classification.

In (Karungaru et al., 2021), the authors proposed a vehicle detection and classification using an improved AlexNet architecture. From comparative results of three methods, original Alexnet, improved Alexnet with fully connected layer, and improved Alexnet as a feature extractor with SVM, the last model performs better than the full-connected network.

The authors in (Öznur Özaltın and Özgür Yeniay, 2021) proposed an hybrid CNN algorithm with 34 layers to perform a feature extraction automatically with SVM for classifying Electrocardiogram (ECG) signals. A comparison results indicated that proposed model represent the most successful result.

The rest of the paper is organized into four main sections. Section "Related work" presents the recent state-of-the-arts. Section "Methodology and materials" exhibits the proposed methodology for skin image classification and datasets used. Section "Experimental results and discussion" addresses the results and discussion. Finally, Section "Conclusion" concludes the research.

2. Related works

In the last decade, several automatic classification approaches for skin lesion images have been proposed. CNN approaches have completely dominated the skin lesion classification process and this related works will be just a drop in the bucket.

In (Yu, Chen, Dou, Qin and Heng, 2017), the authors proposed a framework with two-stage for melanoma recognition by employing a convolutional neural network (CNNs) to acquire discriminative features. First, a fully convolutional residual network incorporates multi-scale feature representations for the segmentation stage, then the last segmentation results are integrated into a residual network for the classification stage.

In (Mahbod, Ecker and Ellinger, 2019), the authors proposed a comparative study of the diagnosis of pigmented melanocytic lesions according to the accuracy between a proposed combined CNNs and medical personnel expert in dermoscopy. The proposed CNN combines the outputs of two CNNs trained on two different datasets, one with 7895 dermoscopic images and the other with 5829 clinical close-up skin lesion images. The study proves the efficiency of the CNN to classify dermoscopic images like experts in the field.

In (Tschandl, Rosendahl, Akay, Argenziano, Blum, Braun, Cabo, Gourhant, Kreusch, Lallas, Lapins, Marghoob, Menzies, Neuber, Paoli, Rabinovitz, Rinner, Scope, Soyer, Sinz, Thomas, Zalaudek and Kittler, 2019), a hybrid approach for skin lesion classification is proposed by using a deep feature generator from three deep learning pre-trained models: AlexNet, VGG16, and ResNet-18. A combination of multiple support vector machine classifiers is used in the last stage.

In (Jadhav, Ghontale and Shrivastava, 2019), authors extracted features of lesions using CNN, and then an SVM is used for classification. Without any preprocessing, the CNN eliminates the need for hand crafted features, and has been utilized in the proposed method for feature extraction.

Three classifiers are used in (Patil and Dongre, 2020) to detect melanoma images from a collection of PH2 dermoscopic images. For training and testing purposes, KNN, Naive Bayes, and SVM classifiers are used. The highest precision of the classifier with de-duplication techniques is shown by SVM.

In (Murugan, Nair and Kumar, 2019), to extract features from skin cancer images, the authors used three different approaches and classifiers. Shape, ABCD rule, and GLCM are the characteristics extracted. The kNN (k Nearest Neighbor), Random Forest, and SVM classifiers used are (Support Vector Machine).

In (Arora, Dubey, Jaffery and Rocha, 2020b), authors extracted shape, texture, and color features from skin lesions. The classification was made by binary a support vector machine (SVM). Patients with melanoma have been classified and differentiated from cases with non-melanoma.

In (Melbin and Raj, 2021), authors present an integrated approach based on modified ABCD features, and Support Vector Machine (SVM) has been used to detect and classify skin lesion images. Authors have taken three skin diseases, namely melanoma, seborrheic keratosis, and lupus Erythematosus, from three different three datasets. They obtained a better precision only for these three classes, but they do not consider the other classes.

An approach to identifying skin lesions most frequently found in a clinical environment was proposed by Robert et al. (Fisher, Rees and Bertrand, 2019). Using a K-nearest neighbor classifier, they used a mix of standardized and hand-crafted features, feature selection from a vast pool of possible features, and a hierarchical decision tree.

Surówka et al. (Surówka and Ogorzalek, 2019) have extracted wavelet features appropriate for the classification of dermoscopy images into two classes: pigment skin cancer (melanoma) and dysplastic (atypical but benign) skin lesion. In their work for feature extraction, the wrapper method will be used for classification with the Naive Bayes.

In (Mporas, Perikos and Paraskevas, 2020), authors present an architecture to classify pigmented skin lesions from dermatoscopic images. The color-based features have been extracted and the classifier used is AdaBoost with random forest. Zhou et al.

Yu and Zhuoyi (2013) investigated the use of multiple binary decision trees in the computer-assisted diagnosis of melanoma. The experimentation has been carried out on a dataset containing only 235 samples (125 benign and 110 malignant).

Massimo et al. (Aria, D' Ambrosio, Iorio, Siciliano and Cozza, 2020) proposed a dynamic classification tree for malignant melanoma identification in skin lesion dermoscopic images.

In (Dhivyaa, Sangeetha, Balamurugan, Amaran, Vetriselvi and Johnpaul, 2020), the authors used decision trees and random forest algorithms for skin lesion classification. They tested their approach on HAM10000 and ISIC 2017. Decision tree algorithms and random forests help generate better outcomes reasoning, making it a good goal to deal with the issues that also impact other regions. The findings result are considered to be superior to those that are similar in the literature.

In (Perez, Avila and Valle, 2019), authors proposed an evaluation study of the factors that affect the choice of the best CNN architecture for skin lesion analysis, and the evaluation of performance of simple ensemble models contrasting single models, based on the evaluate of 13 factors over nine architectures on the ISIC 2017 dataset. Obtained results was between 84% and 91%.

In (Valle, Fornaciali, Menegola, Tavares, Bittencourt, Li and Avila, 2020) authors explored 10 choices faced by researchers: use of transfer learning, model architecture, train dataset, image resolution, type of data augmentation, input normalization, use of segmentation, duration of training, additional use of SVMs, and test data augmentation. They concluded that advancing research on automated skin lesion analysis requires curating larger public datasets. Ensembles of models are a cost-effective alternative to the expensive full-factorial and to the unstable sequential designs.

In (Ozkan and Koklu, 2017), authors classified a PH2 datasets into three groups as normal, abnormal and melanoma using four different machine learning classifier: ANN, SVM, KNN and decision tree. They achieved 92.50%, 89.50%, 82.00% and 90.00% for ANN, SVM, KNN and decision tree respectively.

In (Ghalejoogh, Kordy and Ebrahimi, 2020a), authors proposed a stacking ensemble method based on the meta-learning algorithm for classifying skin lesions. The model used four classifiers: KNN, SVM, ENN, and MLP. Segmentation step used for different feature extraction is based on the shape, color and texture. They achieved 98.5% for melanoma and non-melanoma classes on PH2 dataset.

In (Chakravorty, Liang, Abedini and Garnavi, 2016), authors proposed a combination of three vectors feature extraction, based on the segmentation to capture the asymmetric distribution of shape, color and structure. The Geometry Features to captures the various shape related to the lesion, the Color Features using Kullback-Leibler to capture the divergence of the color distribution, and the Structural Features to capture the contrast and the difference in luminance. Obtained results was 83% for PH2 dataset.

In (Salido and Ruiz, 2018), authors used a preprocessing technique to remove the unwanted artifacts, and a convolutional neural network AlexNet for the classification. They tested the classifier using both preprocessed and unprocessed images on the PH2 dataset. They obtained an accuracy of 93% for two classes, and an accuracy of 67.5% for three classes.

In (P.V. Asha Deepika and R, 2020), authors used the preprocessing technique followed by segmentation and finally two machine learning classifier SVM and KNN for the classification. In addition, they used the image enhancement, CLAHE for the spatial domain and SWT for the frequency domain then CNN architecture as feature extractor followed by the SVM classifier. Obtained accuracy was 92.80% on PH2 dataset.

Khalid et al. proposed a fine-tuning AlexNet architecture by replacing the last layer with a softmax to classify the lesion on three different classes from the PH2 dataset (Hosny, Kassem and Foad, 2018). Obtained result was 98.61% on PH2 dataset.

In (Ahmed, Yanikoglu, Göksu and Aptoula, 2020), authors proposed a fusing and fine-tuning Xception, Inception-ResNet-V2, and NasNetLarge architecture on the ISIC2019 dataset and achieved an accuracy of 93.70%.

3. Methodology and Materials

This paper investigates the efficiency of using convolutional neural networks (CNN) architectures as feature extractors and many machine learning classifiers to evaluate the classification of skin lesions from two different datasets. Figure 1 depicts the proposed method's flowchart.

The process is made up of four successive parts. The first component is the input of dermoscopic images, the second is the feature extraction part with 17 commonly pre-trained convolutional neural networks (CNN) architectures. From the features obtained, two bases are created: a learning database and a test database. The third component of the system is the learning step by a set of classifiers, each one is performed individually. Finally, the last is the validation step from the test dataset.

There are three kinds of extracted features with the skin lesions dataset in the first section of the flowchart:

1. Extract features using the original dataset.
2. Extract features using the original dataset with image preprocessing.
3. Extract features with the augmented dataset.

3.1. Conventional Neural Networks Architectures

Several CNN architectures have been developed, like GoogLeNet, DarkNet, AlexNet, ResNet, NasNet beside others. Some of these architectures are available as pre-trained models that were initially trained on huge natural images.

Now, in this section, we present a brief overview of some of the most widely used CNNs. We will use them as extractor of features from skin lesion images.

3.1.1. GoogLeNet

GoogLeNet was implemented by Szegedy et al. (Szegedy, Liu, Jia, Sermanet, Reed, Anguelov, Erhan, Vanhoucke and Rabinovich, 2015), also known as Inception-V1. GoogLeNet aims at reducing the computation complexity compared to the traditional CNNs. It achieved a top-5 with an error rate of 6.67% and won the ILSVRC competition in 2014. GoogLeNet uses an essential module called the inception block. These modules present optimized variants of the classic convolution layers with various scales in order to avoid problems alignment of patches. Inception modules introduce partial connections inside a convolutional layer to reduce its dimensionality. These modules use variable size filters (1x1, 3x3, and 5x 5), which are applied on the same convolutional layer then the feature-maps results are stacked

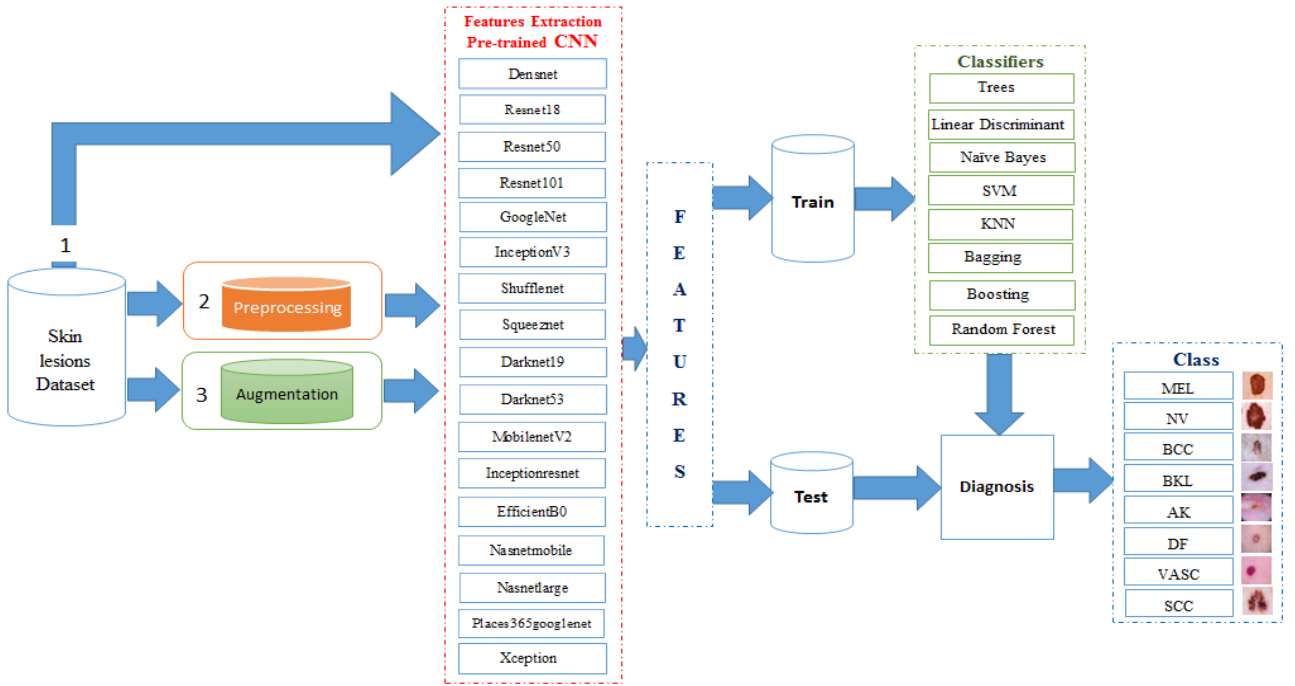


Figure 1: Proposed method's flowchart.

to form the next convolutional layer. GoogleNet presents a wide architecture with 22 total layers and seven million parameters with nine inception modules, four convolutional layers, four max-pooling layers, three average pooling layers, five fully-connected layers, and three softmax layers. In addition, it uses dropout regularization in the fully-connected layer and applies ReLU activation in all of the convolutional layers. Place365GoogLeNet is a pre-trained model with the same architecture as GoogLeNet but trained on the Places365 dataset, which classifies images into 365 different categories.

3.1.2. InceptionV3

InceptionV3 is an optimized version of the Inception-V1 network (Szegedy, Vanhoucke, Ioffe, Shlens and Wojna, 2016) with 24 million parameters. The idea of inception-V3 was the update of the inception module with tweaks in symmetric and asymmetric building blocks. It offers two techniques of factorization to optimize computational complexity and reduce the number of multiplications for the variants inception blocks. The first technique is the factorization of convolutions associated with large filters to reduce the number of parameters proposed and the size of the filters from 5×5 and 7×7 to 3×3 . The second technique is the spatial factorization in asymmetric convolutions. The architecture proposes to replace the classic convolutions ($N \times N$) with asymmetric convolutions ($N \times 1$ and $1 \times N$) in scope to reduce the computational cost.

3.1.3. ResNet

ResNet is an advanced version of a convolutional neural network based on residual blocks proposed by He et al. in 2015 (He, Zhang, Ren and Sun, 2016). This architecture aims at solving the issue of gradient degradation, that appears in very deep networks, where the accuracy begins to be saturated; thereafter, it degrades rapidly due to the decrease in gradient values. In order to solve this problem, ResNet introduces the concept of residual blocks by including shortcut connections between layers to increase the accuracy without increasing network depth and prevents corruption while the CNN deepens. ResNet was learned with a network depth of 152 layers and over eleven million parameters. It uses a 3×3 convolution filter, global average pooling, residual blocks, batch normalization followed by the classification layer. In order to reduce the computational cost and accelerate the training of ResNet, the bottleneck blocks defined with 3×3 convolution are replaced by 1×1 convolution followed by 3×3 convolution and another 1×1 convolution. ResNet proposes several variations with a different number of layers, ResNet18, ResNet34, ResNet50, ResNet101 and ResNet152.

ResNet-152 won the ILSVRC competition in 2015 and achieved a top-5 with an error rate of 3.57%.

3.1.4. SqueezeNet

Squeeze and Excitation Network (SE-Network) was proposed by Iandola et al. (Iandola, Moskewicz, Ashraf, Han, Dally and Keutzer, 2017). It offers dimensionality reduction techniques by adding a new block to select feature-maps known as SE-block before the convolution layer. SE block is based on two operations: squeeze and expand. Squeeze operation is used to get a global view of feature-maps by suppressing spatial information in the convolved input in order to reduce dimensionality and the depth of the maps; the excitation operation is used to capture spatial information. A fire module comprises a squeeze convolution layer with a convolution filter of 1x1 feeding into an expand layer which a mix of 1x1 and 3x3 convolution filters. SqueezeNet is composed of two standard convolutions, eight fire modules, three max-pooling layers, and one global average pooling followed in the end by a softmax layer.

3.1.5. DarkNet

DarkNet is a faster and accurate convolutional neural network used for object detection (Redmon, Divvala, Girshick and Farhadi, 2016). It presents the features extractor from YOLO architecture images with different versions of Darknet-19 (Redmon and Farhadi, 2017) DarkNet-53 (Redmon and Farhadi, 2018). The DarkNet-19 consists of 19 convolutional layers, 5 max-pooling layers, global average pooling layer, and softmax. It uses the identity mapping and 3x3 convolution filters instead of fully connected layers, a global average pooling to make predictions in front of the softmax layer. Batch normalization is used to regularize the model, speed up convergence, and stabilize the training without using any residual blocks, up sampling or skip connections. Redmon et al. proposed DarkNet-53 in 2018 (Redmon and Farhadi, 2018). It consists of the improved version of DarkNet-19 with more layers and the use of residual connections. It is composed of 53 layers with 3x3 and 1x1 convolution filters, five residual blocks to add the previous output values to the output of the current layer and average-pooling layer, and one fully connected layer. DarkNet-53 doesn't use any max-pooling layers; instead, convolution layers are used. DarkNet-53 allows the detection at three scales by down sampling input image dimensions by 32, 16, and 8.

3.1.6. DenseNet

Huang et al. proposed DenseNet (Huang, Liu and Weinberger, 2017). It is based on dense connections between layers of convolution. DenseNet aims to solve vanishing gradient issue and reduces the number of input and parameters as well as reinforcing features extraction. DenseNet architecture exploits the effects of shortcut connections where it is composed of a set of dense blocks, which are linked by transition layers. Each block contains a set of convolution layers, and each layer is connected to all preceding other layers belonging to the same block instead of adding them. All feature-maps from previous layers are passed on to successive layers, making the network thinner and more compact. DenseNet comprises convolution and pooling layers, transition layers, classification layers, and multiple dense blocks in series. DenseNet proposes several variations with a different number of layers: DenseNet-121, DenseNet-169, DenseNet-20, and DenseNet-246.

3.1.7. Xception

Chollet from Google proposed Xception in 2017 (Chollet, 2017). The Xception architecture consists of 36 convolutional layers structured into 14 modules. It processes 23 million parameters based on inception blocks' adaptation by replacing them with depthwise separable convolutions layers. With residual connections, the depthwise separable convolution consists of replacing the different spatial input dimensions (1x1, 5x5, 3x3) with a single dimension (3x3) followed by a pointwise convolution 1x1.

3.1.8. Inception-ResNets

The Inception-ResNet is proposed by Szegedy et al. in 2017 (Szegedy, Ioffe, Vanhoucke and Alemi, 2017) with 56 million parameters. Inspired by Inception and ResNet module's performance, it offers a hybridization between modules introduced into the InceptionV3/InceptionV4 architectures and Residual connections. It consists of replacing the Inception modules with Residual-Inception blocks. By adding convolution's output in the layers of inception to the input, to obtain the same dimensionality of the input and output, they use 1x1 convolutions after the original convolutions. This combination aims to speed up the Inception network's learning time and avoid the issue of degradation gradient.

3.1.9. *Shufflenet*

ShuffleNet is a convolutional neural network designed for mobile devices, introduced by Zhang et al. in 2017 (Zhang, Zhou, Lin and Sun, 2018). ShuffleNet is composed of stack ShuffleNet units grouped into three stages. The architecture utilizes the pointwise group convolution and channel shuffle, where group convolutions replace the first and second 1x1 convolutions, then a channel shuffle is applied after the first 1x1 convolution.

3.1.10. *Mobilenet*

MobileNet was proposed by Howard et al. in 2017 (Howard, Zhu, Chen, Kalenichenko, Wang, Weyand, Andreetto and Adam, 2017). It is designed for mobile devices and integrated vision systems. This network is composed of 28 layers, including 13 depthwise convolutions and 13 pointwise convolutions. The main purpose of depthwise convolution modules is to reduce the dimension of the network. In MobileNet, all layers are followed by batch normalization and ReLU, and the final layer is a fully connected layer that feeds to the softmax. The MobileNet network introduces two hyperparameters to realize a compromise between the response period, storage space, and precision. These hyperparameters are the width multiplier and the resolution multiplier. The role of the width multiplier is to reduce the number of feature maps in convolution layers. In contrast, the resolution multiplier reduces the computational cost by decreasing the input image's resolution, which automatically decreases the size of the convolution layers.

3.1.11. *Nasnet*

Nasnet is a scalable CNN architecture proposed by Google ML group (Zoph, Vasudevan, Shlens and Le, 2018). It introduces the concept of an optimized network using the reinforcement-learning method. Nasnet consists of basic building convolutional layers called cells repeated multiple times with an identical architecture and different weights. Each cell in the architecture presents the concatenation of different small units known as blocks not fixed in the number of the convolutional cells and filters in the convolutional, type, weights, or regularization methods size of input images. These blocks consist of regular convolutions, separable-convolutions, max-pooling, average pooling. Each block creates a residual connection by mapping two inputs to a single output feature map. Nasnet has two types of convolutional cells:

- Normal Cells return a feature map of the same dimension and contain three 3x3 convolution layers and two 5x5 depthwise separable convolutions,
- Reduction Cells that return a reduced feature map in the height and width by a factor of two. The Reduction Cells contain one 3x3, two 5x5, and two 7x7 depthwise separable convolutions

3.1.12. *EfficientNets*

EfficientNets was introduced by Tan et al. in 2019 (Tan and Le, 2019). It is characterized by optimizing the accuracy and efficiency by reducing parameter size compared to the other CNNs. The main idea of EfficientNets is about scaling different dimensions of deep neural network methods by using fixed compound scaling coefficients (width, depth, and image resolution) in a structured way. EfficientNets family contains eight different models, the baseline version B0 with 224x224 input size. The other version up to B7, derived from EfficientNet-B0 by increasing the resolution input image, a number of feature maps for each layer (width), and the number of layers (depth).

3.2. Transfer learning

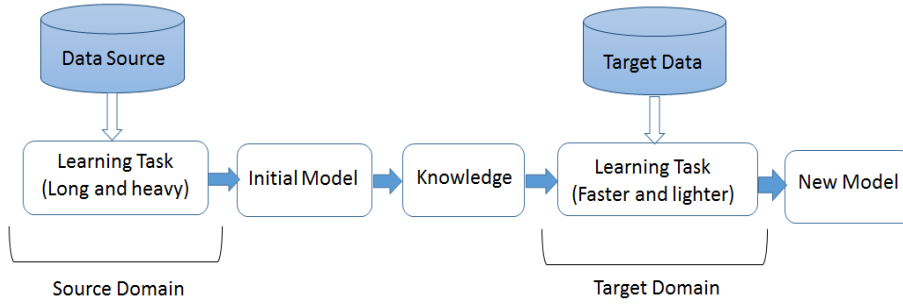
Unfortunately, we rarely have a large-scale imaging data set labeled to train CNN from randomly initialized parameters. In addition, the amount of processing time required to complete this learning task can be prohibitive. Transfer learning presents the best way to tackle these problems Tan, Sun, Kong, Zhang, Yang and Liu (2018).

Transfer Learning is a machine learning concept used in image classification tasks. It consists of reusing knowledge extracted from a trained CNN model with good results in the source domain. This trained model's weights are obtained on large labeled datasets such as ImageNet. Therefore, it was trained on million tagged training images with a high number of classes, on several high power computing facilities and for several days. Once the model has been trained, one can apply obtained weights to another specific image dataset in the new target domain. Transfer Learning can be used as a model of extracted convolutional features, where we take off the last fully connected layers (the output layer), and we considered the rest of the CNN as a "feature extractor" for the smaller dataset. Therefore, transfer learning will be used to classify skin lesion images since they are natural images. We can therefore use a pre-training on ImageNet and transfer the learned parameters.

Table 1

Overview of different CNNs architectures.

| Architecture | Year | Number of Parameters | Number of features | Developed by | Depth | Input image | Layers |
|-----------------|------|----------------------|--------------------|-----------------|-------|-------------|--------|
| Googlenet | 2014 | 4M | 1024 | Szegedy et al. | 22 | 224x224 | 144 |
| InceptionV3 | 2015 | 23.8M | 2048 | Szegedy et al. | 159 | 299x299 | 316 |
| ResNet18 | 2015 | 11.17M | 512 | He et al. | - | 224x224 | 72 |
| ResNet50 | 2015 | 25.6M | 2048 | He et al. | 168 | 224x224 | 177 |
| ResNet101 | 2015 | 44.7M | 2048 | He et al. | - | 224x224 | 347 |
| SqueezeNet | 2016 | 1.2M | 1000 | Iandola et al. | 152 | 227x227 | 68 |
| Darknet-19 | 2016 | 15.7M | 1000 | Redmon et al. | - | 256x256 | 64 |
| Darknet-53 | 2018 | 61.5M | 1000 | Redmon et al. | - | 256x256 | 184 |
| InceptionResNet | 2017 | 55.8M | 1536 | Szegedy et al. | 572 | 299x299 | 824 |
| Xception | 2017 | 22.9M | 2048 | Chollet | 126 | 299x299 | 171 |
| MobileNet | 2017 | 4.2M | 1280 | Howard et al. | 88 | 224x224 | 154 |
| DenseNet201 | 2017 | 20.2M | 1920 | Huang et al. | 201 | 224x224 | 709 |
| Shufflenet | 2017 | 3.4M | 544 | Xiangyu Zhang | - | 224x224 | 173 |
| NASNetMobile | 2018 | 5.3M | 1056 | Zoph et al. | - | 224x224 | 914 |
| NASNetLarge | 2015 | 88.9M | 4032 | Zoph et al. | - | 331x331 | 1244 |
| EfficientNetB0 | 2019 | 5.3M | 1280 | Mingxing and Le | - | 224x224 | 290 |

**Figure 2:** Transfer learning model.

Transfer Learning can also be used as fine-tune hyperparameters by freezing or unfreezing layers. Either by training an architecture with random initial weights by adding a custom classifier according to a new dataset, or by retraining the architecture weights only of some layers while freezing others.

3.3. Features extraction

A feature is described as "a relevant information for solving the different computational tasks related to a specific application". Features are categorized into two standard categories: local features and global features (Salahat and Qasaimeh, 2017). We generally base ourselves on a set of numerical criteria describing the object or phenomenon observed in order to develop a classification rule (supervised or not).

Several techniques have been created for feature extraction, and their operating principles are quite distinct from each other. However, no perfect feature extractor exists until today. Some of the elegant literature surveys are referred in (Salahat and Qasaimeh, 2017), (Guyon, Gunn, M. and Zadeh, 2006), (Storcheus, Rostamizadeh and Kumar, 2015), (Miksik and Mikolajczyk, 2012). Finding a suitable feature for skin lesion images is often a big challenge faced by a lot of research (Barata, Celebi and Marques, 2019). Generally, we can distinguish handcrafted from deep learning features (Barata et al., 2019). Different handcrafted features extraction methods have been suggested to detect skin lesion diseases, such as the 7-point checklist method (Walter, Prevost, Vasconcelos, Hall, Burrows, Morris, Kinmonth and Emery, 2013), ABCD rule, an acronym for Asymmetry, Border, Color and Dermoscopic structure (Nachbar, Stolz, Merkle, Cognetta, Vogt, Landthaler, Bilek, Braun-falco and Plewig, 1994), (Abbasi, Shaw, Rigel, Friedman, McCarthy, Osman, Kopf and Polsky, 2004), CASH algorithm (Henning, Dusza, Wang, Marghoob, Rabinovitz, Polsky

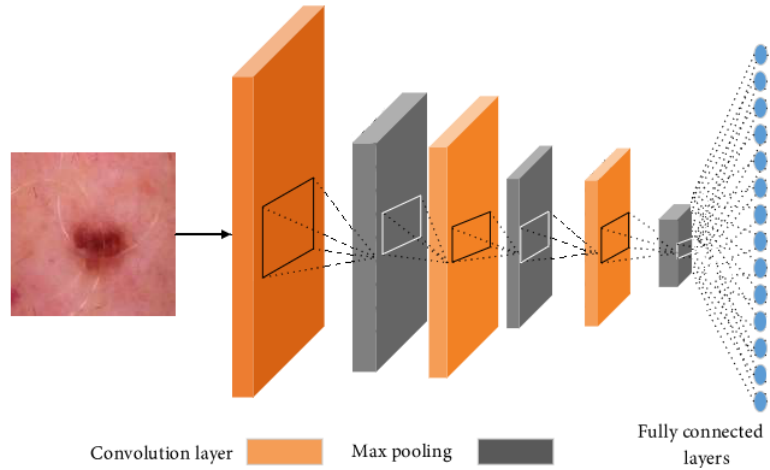


Figure 3: Feature extraction with CNN.

and Kopf, 2007), and three-point checklist (Zalaudek, Argenziano, Soyer, Corona, Sera, Blum, Braun, Cabo, Ferrara, Kopf, Langford, Menzies, Pellacani, Peris and Seidenari, 2006). These skin lesion detection methods are based on the shape, geometry, color, texture, and structure of skin lesions.

Many researchers have demonstrated deep learning efficiency as a feature extraction method in recent years (Kraus et al., 2017), (Wang, MacKenzie, Ramachandran and Chen, 2015). There are several variants of deep learning algorithms to characterize visual features, including the convolutional neural network (CNN (Abbas and Celebi, 2019)). To use the convolutional neural network (CNN), there are two distinct ways. Firstly, it can be used as a classification model and secondly, to extract features using Transfer Learning (Kassani and Kassani, 2019). CNNs learn image features directly without additional manual feature extraction (Figure 3).

There are other studies focusing on the hybridization of handcrafted and deep learning features. Filali et al. (Filali, Khoukhi, Sabri and Aarab, 2020) proposed a combination of handcrafted features (shape, skeleton, color, and texture) and features derived from four pre-trained CNNs: AlexNet, VggNet, GoogLeNet, and ResNet. They followed the process by a feature selection to keep only the best and relevant one using the genetic algorithm. Finally, based on the new features, SVM classifies skin cancer from the Ph2 dataset into melanoma or non-melanoma. The result got an accuracy of 87.8%.

In (Moura, Veras, Aires, Machado, Silva, Araújo and Claro, 2018), the proposed approach classifies skin lesions using a hybrid descriptor obtained by combining features of color, shape, texture, and pre-trained CNN. The result was found with a Multilayer Perceptron classifier with a 92.1% accuracy rate for PH2 and DermIS datasets.

Some of the recent literature surveys of features extraction in dermoscopy image analysis of skin lesion can be found in (Javed, Rahim, Saba and Rehman, 2019), (Barata et al., 2019), (Kolkur and Kalbande, 2016), (Abuzagheh, Faezipour and Barkana, 2015).

3.4. Supervised classification and sets of classifiers

This section describes the machine learning classifiers we use to classify skin lesion images.

3.4.1. Decision Tree

A decision tree is a graph that uses branching methods to illustrate a course of action and various outcomes (Smith and Koning, 2017). A decision tree works for both categorical and numerical variables since it doesn't require an assumption about the data distribution and classifier structure. Decision trees have the ability to accurately and efficiently provide classifications for large datasets (Avellaneda, 2019).

3.4.2. Gaussian Naive Bayes

The Naive Bayesian classifier (Neapolitan, 2007) is a simple yet robust classification approach which is based on Bayes' Theorem that describes the probability of an event from prior knowledge of the condition that potentially related to the event. The naive Bayes classifier calculates all the probabilities of all classes (values) for a target feature and

selects the one with the highest probability. Furthermore, Gaussian Naive Bayes assumes that the values associated with each class of each feature follow a Gaussian distribution. There exist multiple types of naive Bayes classifier. In this study, we use Gaussian Naive Bayes and Kernel Naive Bayes.

3.4.3. Support Vector Machines

Support Vector Machines (SVM) is a supervised learning model used for classification (Vapnik, 1998). Using the kernel approach, it can perform a non-linear classification that implicitly maps input data into a high-dimensional space of features. SVMs generally classify data by constructing a hyperplane which determines a straight line separating the space into two zones as homogeneous as possible. The set of data points closest to the hyperplane are referred to as the Support Vectors. When the data is linearly separable in a set, a linear SVM can only be utilized. When the data is not linearly separable, a kernel function can be selected to map the data into a higher dimensional space. The goal of this is to force the data points to become linearly separable if possible. We employed various SVM kernel functions: Linear, Quadratic, Cubic, Fine Gaussian, Medium Gaussian, and Coarse SVM.

3.4.4. K-Nearest Neighbor Classifiers

K-Nearest Neighbor (KNN) algorithm classifies unlabeled data to the nearest most similar labeled data. Due to its simplicity and efficiency, it is widely used to perform supervised classification in multivariate settings. The only parameters associated with a KNN classifier are the number of nearest neighbours to be considered, K. It is common to select K value between 3 and 10, to overcome the overfitting and the underfitting problems (Larose and Larose, 2014). There are many ways to measure the distance between the attributes in the testing set and the training set. The most significant ones are the Euclidean and the Manhattan Distances. Depending on the number of neighbors, we distinguish Fine (Xu, Zhu, Fan, Qiu, Chen and Liu, 2013), Medium, Coarse, Cosine, Cubic, and Weighted KNN.

3.4.5. Ensemble Methods

The two best-known classes of algorithms in the literature for building classifiers set are Boosting and Bagging. These two types of algorithms build sets of the same type differentiated at the level of the examples handled during training. On the other hand, at the "Characteristic" level, the most representative method is the "Random SubSpace" method. It represents a method that mixes the two techniques. We present these different types of algorithms in the following.

- Bagging ("bootstrap aggregating") (Breiman, 1996) consists of building a set of classifiers from different re-samplings of the same set of training data. This algorithm uses a method called "bootstrapping" to generate different sets. Bootstrapping (Efron and Tibshirani, 1993) is a resampling method with replacement. It consists of choosing examples from the training sample to create new sets. These are called "bootstrap samples". After generating "bootstrap" samples, a set of classifiers will be constructed. Each elementary classifier will be trained on one of the samples so that they are all trained on a different training set.
- "Boosting" (Schapire, 2005) designates a general learning principle making it possible to improve the precision of a given learning algorithm. The general principle is to combine linearly the results of so-called "weak" classifiers in order to construct a "strong" learning classifier from the original set and a method of combining classifiers constructed from each new set. To define his new technique of "Boosting", Shapire has proposed the idea that any weak classifier able to learn with a certain confidence and a classification error less than "0.5", can be transformed into a more confident classifier and with as small a classification error as desired. At each iteration, the algorithm seeks to find a weak classifier that can best correct the classifiers' errors in the preceding iterations. In the principle of "Boosting", this objective is achieved by means of a weighting of the training data.
- A Random Forest (Breiman, 2001) is the mixture of the two techniques of "Bagging" and "Random SubSpace" applied to decision trees. At each iteration, a "bootstrap" sample is chosen at random in order to build a binary decision tree. The search space for the construction of the nodes of the tree is limited by P characteristics drawn randomly. The performance of the method depends directly on the parameter P. A small value of P may degrade the performance of the classifier. In (Breiman, 2001), the author showed empirically that the optimal value of P is: $P = \sqrt{N}$, where N is the total number of characteristics. The random technique of the approach has shown its relevance and efficiency, especially on high dimensional data (with a high number of characteristics). This technique allows a better exploration of the representation space.

Table 2

Number of ISIC 2019 samples for each class

| Diagnostic Class | Number of Images | Percentage |
|------------------------------|------------------|------------|
| Melanoma (MEL) | 4522 | 17.85% |
| Melanocytic Nevus (NV) | 12875 | 50.83% |
| Basal Cell Carcinoma (BCC) | 3323 | 13.12% |
| Actinic Keratosis (AK) | 867 | 3.42% |
| Benign Keratosis (BKL) | 2624 | 10.36% |
| Dermatofibroma (DF) | 239 | 0.94% |
| Vascular Lesion (VASC) | 253 | 1.00% |
| Squamos Cell Carcinoma (SCC) | 628 | 2.48% |
| None of the others (UNK) | 0 | 0% |
| Total | 25331 | |

3.5. Dermoscopy images datasets

In this section, we will go through the datasets we have used: ISIC 2019 and PH2, as well as the pre-processing and data augmentation methods that can be considered.

3.5.1. ISIC 2019 dataset

The ISIC 2019 dataset Tschandl, Rosendahl and Kittler (2018), Gutman, Codella, Celebi, Helba, Marchetti, Mishra and Halpern (2018), Combalia, Codella, Rotemberg, Helba, Vilaplana, Reiter, Halpern, Puig and Malvehy (2019) consists in 25331 dermoscopic images with labels for training. ISIC 2019 aims at classifying dermoscopic images into nine distinct diagnostic classes: 1. Melanoma (MEL); 2. Melanocytic nevus (NV); 3. Basal cell carcinoma (BCC); 4. Actinic keratosis (AK); 5. Benign keratosis (BKL); 6. Dermatofibroma (DF); 7. Vascular lesion (VASC); 8. Squamous cell carcinoma (SCC); and 9. None of the others (UNK). The distribution of samples for each class in the training dataset is shown in table 2.

3.5.2. PH2 dataset

The Dermatology Service of Pedro Hispano Hospital in Portugal and the University of Porto collaborated to establish the Ph2 database Mendonça, Ferreira, Marques, Marçal and Rozeira (2013). There are 200 dermoscopic images in the PH2 database, with 80 common nevus, 80 atypical nevus, and 40 melanomas. These are 8-bit RGB color images with a 768 x 560 pixel resolution. The images were obtained from Hospital Pedro Hispano¹.

3.5.3. Preprocessing

The preprocessing phase aims at ameliorating the dermoscopic images' quality to ensure improved lesion detection efficiency (Figure 4). Throughout image acquisition, the presence of artifacts such as hair, bad lighting, and other noise can result in an imperfect diagnosis. There are common artifact removal algorithms such as DullRazor (Lee, Ng, Gallagher, Coldman and McLean, 1997), which is used to extract hair artifacts. The algorithm uses morphological filters to detect the position of the hair and replace the detected hair with its neighboring pixels. Adaptive median filters are used to smooth out the final image. We can also apply various filters for removing noises (Hoshyar and Al-Jumaily, 2014). In addition, to further improve the image quality, some image enhancement methods are also used. The most important of them are color correction or calibration (Wighton, Lee, Lui, Mclean and Atkins, 2011). We can find other techniques like illumination correction, contrast enhancement, and edge enhancement (Maglogiannis, Zafiroopoulos and Kyranoudis, 2006). Readers will find a summary of preprocessing methods related to dermoscopy images in (Celebi, Wen, Iyatomi, Shimizu, Zhou and Schaefer, 2015).

3.5.4. Data augmentation

Many authors believe that increasing data is fundamentally important for improving the performance of convolutional neural networks (Howard, 2014), (Krizhevsky, Sutskever and Hinton, 2012), (Wu, Yan, Shan, Dang and Sun, 2015). Data augmentation involves applying certain transformations to the initial training set to create new artificial

¹<https://www.fc.up.pt/addi/ph2%20database.html>

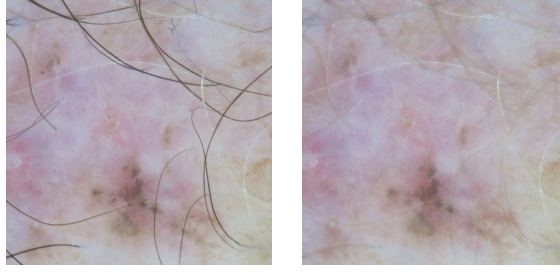


Figure 4: Example of skin image preprocessing.

examples. The applied transformations do not change the nature of the detected class and thus create new examples. The increase in data makes it possible to create invariances of the final network artificially and makes it possible to increase performance in generalization (Howard, 2014), (Wu et al., 2015). For skin lesion images, among the classic operations used for data augmentation, we can find rotation, zooming, cropping, addition of noise, scaling, and translation. More advanced transformations such as changing the contrast or brightness of the image can also be added depending on the user's needs (Perez, Vasconcelos, Avila and Valle, 2018). Figure 5 depicts some examples of data augmentation phase.

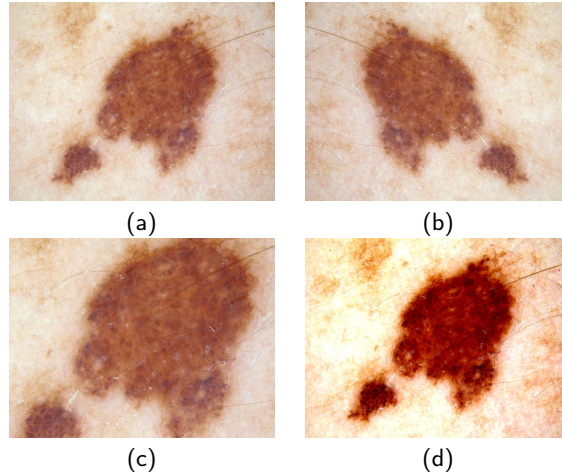


Figure 5: Example of data augmentation performed (a) original image; (b) rotation 180° (c) zoom (d) changing the contrast.

4. Experimental results and discussion

All experiments are performed on a desktop computer equipped with a Core i9 processor, 32 GB DDRAM with a GeForce GTX 1080 Ti NVIDIA GPU. CNN features extraction, and classifiers, training were carried out in x64-bit MATLAB 2020.

4.1. Evaluation metrics

We use a variety of standard evaluation metrics to compare the performance of the machine learning classifiers for skin lesions classification such as accuracy, sensitivity, specificity, precision, balanced accuracy and F-measure Grandini, Bagli and Visani (2020).

$$Accuracy = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \quad (1)$$

$$Sensitivity = \frac{T_P}{T_P + F_N} \quad (2)$$

$$Specificity = \frac{T_N}{T_N + F_P} \quad (3)$$

$$Precision = \frac{T_P}{T_P + F_P} \quad (4)$$

$$Balanced\ accuracy = \frac{Sensitivity + Specificity}{2} \quad (5)$$

$$F - measure = 2 \times \frac{precision \times recall}{precision + recall} \quad (6)$$

Where T_P , T_N , F_P , F_N , and refer to true positive, true negative, false positive and false negative respectively.

4.2. ISIC 2019 dataset Experiments

In our work, we define three kinds of performed experiments with the ISIC 2019 dataset:

1. evaluate the proposed method using the original dataset.
2. evaluate the proposed method using the original dataset with image preprocessing.
3. evaluate the proposed method with the augmented dataset.

This section presents the experimental results obtained from the three scenarios with 17 pre-trained CNN models and 24 classifiers. Based on ISIC 2019 dataset, where 80% is used for training and 20% is used for testing. The main aim of experiments in our study was to compare the difference in performance between treating the base as it is, doing preprocessing, and exploiting the data augmentation. In addition, to compare which is the best CNN architecture for feature extraction and which is the best combination between various CNN extractors and various classifiers.

All classifiers, regardless of architecture, were trained using the same training set. To simplify the presentation of results in a single table we limited to the first nine models in terms of accuracy. The performance analysis and evaluation of all experiments were carried based on the most common evaluation measure (accuracy).

4.2.1. Original ISIC 2019 dataset Experiments

The dataset contains a total number of 25331 images divided into two parts. The first part was 80% equals to 20265 images of the dataset for training and the second part was 20% equals to 5066 images used for testing.

To establish the effectiveness of the model, each classifier was evaluated on the same test set containing original images. Table 3 presents the accuracy results on the original ISIC 2019.

From the results of this experiment, when we compare the results obtained from each CNN separately with all 24 classifiers, we can observe that the precision obtained from the cubic SVM classifier is higher compared to all accuracy obtained from the other classifiers. On the other hand, when we compare precision obtained from the different CNN we observe that DenseNet201 obtain the highest accuracy with the best classifier. From the results for this experiment, we can observe that the combination of DenseNet201 and cubic SVM perform the highest accuracy with 80.87%.

4.2.2. Preprocessing ISIC 2019 dataset Experiments

To improve the quality of the dermoscopy images by extracting the hair artifacts, we used a public program². The second experiment was carried using 25331 improved images from ISIC 2019 dataset divided into two parts. The first part was 80% equals to 20265 images of the dataset for training and the second part was 20% equals to 5066 images used for testing.

To establish the effectiveness of the model, each classifier was evaluated on the same test set containing preprocessing images. Table 4 presents the accuracy results on the preprocessed ISIC 2019.

We can observe in this experiment that the result obtained from the cubic SVM classifier combined with DenseNet201 CNN is higher compared to all accuracy obtained from the other classifiers.

²<https://github.com/sunnyshah2894/DigitalHairRemoval>

Table 3
Accuracy results on the original ISIC 2019

| <i>Model</i> <i>Method</i> | DenseNet201 | Resnet18 | Resnet50 | Resnet101 | Shufflenet | Squeezenet | Darknet53 | Efficientb0 | InceptionResnet |
|-------------------------------|---------------|----------|----------|-----------|------------|------------|-----------|-------------|-----------------|
| Fine Tree | 59.24% | 58.40% | 59.60% | 60.09% | 58.40% | 59.20% | 57.50% | 58.30% | 57.70% |
| Medium Tree | 58.53% | 56.90% | 59.30% | 58.76% | 56.90% | 57.50% | 56.90% | 56.70% | 56.40% |
| Coarse Tree | 54.32% | 55.10% | 57.30% | 54.70% | 55.50% | 54.30% | 54.10% | 53.60% | 54.80% |
| Linear Discriminant | 73.41% | 66.50% | 71.10% | 71.97% | 67.40% | 67.40% | 69.70% | 69.40% | 67.90% |
| Gaussian Naive Bayes | 48.14% | 41.20% | 42.00% | 44.49% | 34.20% | 34.50% | 38.50% | 39.80% | 42.60% |
| Kernel Naive Bayes | 51.15% | 54.60% | 45.80% | 42.01% | 35.40% | 38.60% | 43.70% | 51.70% | 49.80% |
| Linear SVM | 72.40% | 68.00% | 71.80% | 73.31% | 69.10% | 67.60% | 68.10% | 71.30% | 69.20% |
| Quadratic SVM | 78.68% | 72.70% | 75.60% | 77.22% | 73.70% | 72.90% | 73.70% | 75.40% | 73.70% |
| Cubic SVM | 80.87% | 74.50% | 77.30% | 78.27% | 75.50% | 74.00% | 75.30% | 76.90% | 75.40% |
| Fine Gaussian SVM | 51.00% | 51.20% | 51.00% | 51.60% | 52.30% | 57.20% | 53.00% | 51.50% | 51.70% |
| Medium Gaussian SVM | 75.40% | 71.80% | 73.60% | 74.27% | 72.00% | 69.30% | 71.60% | 73.10% | 72.00% |
| Coarse SVM | 67.82% | 64.80% | 66.80% | 68.22% | 64.90% | 64.30% | 64.60% | 66.30% | 65.20% |
| Fine KNN | 77.28% | 72.30% | 73.70% | 73.88% | 70.90% | 69.20% | 70.70% | 73.60% | 70.20% |
| Medium KNN | 61.90% | 66.80% | 67.30% | 69.50% | 66.10% | 66.30% | 66.20% | 67.60% | 66.00% |
| Coarse KNN | 64.10% | 62.40% | 63.20% | 63.28% | 61.00% | 61.90% | 61.10% | 62.10% | 61.60% |
| Cosine KNN | 69.90% | 67.50% | 69.70% | 71.33% | 67.50% | 66.00% | 66.90% | 68.00% | 67.30% |
| Cubic KNN | 68.23% | 66.50% | 67.50% | 71.65% | 65.30% | 66.20% | 65.90% | 67.20% | 65.90% |
| Weighted KNN | 72.80% | 71.00% | 71.30% | 68.61% | 69.30% | 69.50% | 70.10% | 71.90% | 69.30% |
| Boosted Trees | 61.00% | 58.30% | 61.00% | 62.50% | 59.20% | 59.40% | 59.20% | 59.80% | 58.70% |
| Bagged Trees | 65.10% | 63.60% | 64.10% | 67.53% | 63.60% | 65.20% | 64.30% | 65.00% | 64.00% |
| Subspace Discriminant | 72.60% | 66.10% | 70.30% | 72.17% | 66.60% | 67.50% | 68.50% | 69.10% | 68.60% |
| Subspace KNN | 76.40% | 73.70% | 76.30% | 74.00% | 75.00% | 69.20% | 70.70% | 75.10% | 69.90% |
| RUSBoosted Trees | 45.00% | 43.10% | 44.60% | 46.88% | 47.90% | 40.00% | 40.00% | 41.80% | 40.20% |
| Random Forest | 65.77% | 63.48% | 64.17% | 64.07% | 63.11% | 65.52% | 64.88% | 65.84% | 64.8% |

4.2.3. Augmented ISIC 2019 dataset Experiments

It is clearly remarkable that the number of images in each class of the ISIC 2019 dataset is significantly different. There is a huge gap in the number of images in each class. The NV class contains 12875 images the contrary to AKIEC, DF, VASC, and the SCC. For example, the NV class is 54 times greater than the DF class. With the intention of increasing the number of training images and to reduce the negative effects of significant class imbalance and to improve CNN' efficiency, we have followed some of the augmentation techniques (section 3.5.4) such as rotations, zooming, shearing, cropping, Flips (top-bottom, left-right), skew-left-right and corner, contrasting, adding random color and random brightness.

Table 5 depicts the number of ISIC 2019 dataset samples before and after augmentation. After data augmentation the size of the ISIC 2019 dataset increases by a factor 2.5 with a total 63661 images in total. In this third experiment, we divided the new dataset obtained into two parts. The first part was 80% equals 50929 images of the dataset for training and the second part was 20% equals 12732 images used for testing.

To establish the effectiveness of the model, each classifier was evaluated on the same test set containing augmented ISIC 2019 images. Table 6 presents the accuracy results on augmented ISIC 2019.

When comparing the results obtained from each CNN separately with all 24 classifiers, we can observe that the accuracy obtained from the Fine KNN combined with DenseNet201 CNN classifier is higher compared to all accuracy obtained from the other classifiers.

Table 4
Accuracy results on the preprocessed ISIC 2019

| <i>Model</i> <i>Method</i> | DenseNet201 | Resnet18 | Resnet50 | Resnet101 | Shufflenet | Squeezenet | Darknet53 | Efficientb0 | InceptionResnet |
|-------------------------------|---------------|----------|----------|-----------|------------|------------|-----------|-------------|-----------------|
| Fine Tree | 60.70% | 58.40% | 60.40% | 60.40% | 58.20% | 58.20% | 58.70% | 58.70% | 57.40% |
| Medium Tree | 58.30% | 57.60% | 58.60% | 59.20% | 57.00% | 58.10% | 57.10% | 57.50% | 56.80% |
| Coarse Tree | 55.40% | 55.40% | 56.10% | 55.70% | 54.20% | 55.30% | 54.80% | 55.20% | 54.30% |
| Linear Discriminant | 72.50% | 66.10% | 69.70% | 70.00% | 66.40% | 66.80% | 68.90% | 69.80% | 68.50% |
| Gaussian Naive Bayes | 50.50% | 40.80% | 42.50% | 43.90% | 35.10% | 39.60% | 40.30% | 40.70% | 40.10% |
| Kernel Naive Bayes | 53.30% | 54.40% | 44.40% | 42.00% | 40.20% | 39.30% | 45.00% | 51.80% | 49.10% |
| Linear SVM | 72.00% | 67.70% | 70.70% | 71.60% | 67.50% | 67.20% | 68.80% | 70.90% | 68.60% |
| Quadratic SVM | 77.80% | 71.90% | 75.00% | 75.30% | 71.30% | 71.40% | 73.50% | 75.90% | 73.00% |
| Cubic SVM | 79.50% | 73.20% | 75.90% | 76.70% | 72.10% | 72.50% | 75.50% | 77.90% | 74.10% |
| Fine Gaussian SVM | 67.40% | 51.50% | 51.10% | 51.20% | 52.70% | 56.70% | 52.80% | 51.50% | 51.50% |
| Medium Gaussian SVM | 75.00% | 71.30% | 72.40% | 73.00% | 69.70% | 68.50% | 72.20% | 73.10% | 70.70% |
| Coarse SVM | 67.40% | 65.10% | 66.30% | 66.70% | 64.40% | 63.50% | 65.40% | 66.30% | 64.80% |
| Fine KNN | 76.20% | 71.90% | 73.10% | 72.80% | 70.60% | 68.20% | 71.40% | 71.80% | 69.00% |
| Medium KNN | 69.30% | 66.70% | 67.50% | 67.50% | 65.20% | 65.50% | 67.00% | 67.20% | 65.40% |
| Coarse KNN | 65.20% | 63.80% | 64.60% | 63.10% | 62.50% | 62.00% | 63.00% | 63.10% | 61.90% |
| Cosine KNN | 69.80% | 67.80% | 69.10% | 68.50% | 67.10% | 65.20% | 68.10% | 68.50% | 65.90% |
| Cubic KNN | 69.30% | 66.90% | 67.40% | 66.70% | 64.40% | 65.90% | 67.00% | 66.70% | 64.90% |
| Weighted KNN | 73.40% | 70.00% | 70.90% | 70.30% | 68.10% | 68.90% | 71.40% | 71.30% | 68.60% |
| Boosted Trees | 61.30% | 59.40% | 60.50% | 61.40% | 59.40% | 59.30% | 60.00% | 60.80% | 59.00% |
| Bagged Trees | 66.60% | 63.60% | 65.70% | 66.60% | 63.80% | 64.80% | 66.40% | 65.40% | 64.00% |
| Subspace Discriminant | 71.90% | 66.50% | 69.10% | 69.40% | 65.80% | 66.60% | 68.10% | 69.40% | 68.00% |
| Subspace KNN | 76.70% | 73.50% | 76.00% | 74.40% | 74.80% | 67.40% | 71.20% | 73.70% | 69.50% |
| RUSBoosted Trees | 47.80% | 44.70% | 47.20% | 45.50% | 44.70% | 42.10% | 41.00% | 43.5% | 40.50% |
| Random Forest | 68.18% | 65.20% | 67.24% | 67.14% | 65.54% | 67.02 % | 66.36% | 66.84% | 65.70% |

Table 5
Number of ISIC 2019 dataset samples before and after augmentation

| | Original dataset | Augmented dataset |
|-------|------------------|-------------------|
| MEL | 4522 | 9044 |
| NV | 12875 | 10000 |
| BCC | 3323 | 6646 |
| AK | 867 | 7803 |
| BKL | 2624 | 7872 |
| DF | 239 | 7170 |
| VASC | 253 | 7590 |
| SCC | 628 | 7536 |
| Total | 25331 | 63661 |

4.2.4. Discussion on ISIC 2019 dataset Experiments

Figure 6 depicts the accuracy for 17 pre-trained CNN models combined with a Cubic SVM classifier for the three scenarios used: treating the base as it is, doing a preprocessing and exploiting the data augmentation.

Through the evaluation results obtained (Table 3, Table 4 and Table 6), we find that the DenseNet201 model combined with the fine KNN or cubic SVM classifiers achieved even better results. Consequently, Table 7 depicts the accuracy according to DenseNet201 model and different classifiers on augmented ISIC 2019 dataset.

Table 6
Accuracy results on augmented ISIC 2019

| <i>Model</i> <i>Method</i> | DenseNet201 | Resnet18 | Resnet50 | Resnet101 | Shufflenet | Squeeznet | Darknet53 | Efficientb0 | InceptionResnet |
|-------------------------------|---------------|----------|----------|-----------|------------|-----------|-----------|-------------|-----------------|
| Fine Tree | 41.11% | 37.14% | 41.43% | 41.16% | 40.84% | 39.25% | 37.46% | 39.49% | 37.62% |
| Medium Tree | 35.46% | 34.90% | 37.7% | 38.76% | 34.8% | 35.6% | 34.97% | 32.82% | 34.42% |
| Coarse Tree | 26.72% | 26.3% | 29.46% | 26.37% | 27.8% | 26.4% | 32.12% | 26.14% | 26.83% |
| Linear Discriminant | 76.29% | 60.34% | 71.77% | 72.31% | 16.76% | 61.02% | 67.29% | 69.82% | 68.54% |
| Gaussian Naive Bayes | 48.6% | 41.4% | 42.3% | 44.45% | 34.6% | 34.54% | 38.71% | 40% | 42.76% |
| Kernel Naive Bayes | 38.5% | 42.2% | 33.84% | 29.4% | 24.12% | 26.86% | 31.7% | 39.3% | 37.4% |
| Linear SVM | 83.12% | 79.52% | 82.12% | 84.2% | 71.42% | 71.67% | 75.34% | 81.62% | 79.83% |
| Quadratic SVM | 88.52% | 79.82% | 84.35% | 84.10% | 75.79% | 79.68% | 80.56% | 85.65% | 82.46% |
| Cubic SVM | 91.71% | 83.79% | 86.87% | 86.95% | 81.06% | 82.87% | 84.98% | 89.36% | 86.86% |
| Fine Gaussian SVM | 52.3% | 52.6% | 51.62% | 52.97% | 53% | 58.04% | 53.6% | 51.87% | 52.43% |
| Medium Gaussian SVM | 76.52% | 72.84% | 74.62% | 75.12% | 72.94% | 71.3% | 72.07% | 73.4% | 72.9% |
| Coarse SVM | 86.69% | 81.04% | 68.85% | 80.1% | 82.36% | 80.7% | 81.76% | 80.71% | 81.46% |
| Fine KNN | 92.34% | 87.49% | 88.07% | 88.01% | 83.92% | 86.24% | 86.73% | 89.63% | 86.56% |
| Medium KNN | 70.4% | 74.8% | 81.7% | 77.5% | 74.1% | 77.2% | 74.2% | 75.6% | 74% |
| Coarse KNN | 73.24% | 71.4% | 76.3% | 72.28% | 70% | 75.3% | 70.1% | 71.1% | 70.6% |
| Cosine KNN | 77.72% | 74.5% | 70.2% | 78.33% | 74.5% | 68.9% | 73.9% | 75% | 74.3% |
| Cubic KNN | 78.03% | 75.5% | 78.7% | 80.65% | 74.3% | 75% | 74.9% | 76.2% | 74.9% |
| Weighted KNN | 79.2% | 77% | 73.5% | 74.61% | 75.3% | 72.2% | 76.1% | 77.9% | 75.3% |
| Boosted Trees | 69.4% | 65.3% | 78.3% | 69.5% | 66.2% | 76.5% | 66.2% | 66.8% | 65.7% |
| Bagged Trees | 74.24% | 72.60% | 70% | 76.53% | 72.6% | 68.4% | 73.3% | 74% | 73% |
| Subspace Discriminant | 73.22% | 57.94% | 79.45% | 69.20% | 59.47% | 58.82% | 63.53% | 66.96% | 65.94% |
| Subspace KNN | 77.52% | 75.64% | 78.4% | 74.62% | 76.43% | 71.55% | 72.94% | 76.00% | 70.14% |
| RUSBoosted Trees | 38.23% | 37.11% | 38.4% | 40.01% | 41.62% | 34.1% | 34.12% | 35.65% | 34.54% |
| Random Forest | 70.5% | 68.3% | 69.54% | 70.34% | 68.74% | 70.28% | 68.56% | 69.14% | 68.1% |

Table 8 shows that DenseNet201+Cubic SVM or DenseNet+Fine KNN are advantageous in comparison to a pure, simple, properly fine-tuned DenseNet201. Therefore, the various experimental results we have proposed have demonstrated that our approach can be able to improve the performance of a fine-tuned CNN.

Figure 7 depicts the accuracy of the ISIC 2019 dataset on the three scenarios experimented with according to the DenseNet201 model and different classifiers. Through this figure, we notice that the increase in data considerably improved the results for the SVM and KNN classifiers (with their different kernels). On the other hand, we remark that the results are weak for the tree and Bayesian classifiers.

From Figure 7, the statistics show that classifiers belonging to the KNN family perform the highest in terms of maximum classification accuracy of 92.34%. The SVM family is the second best in this domain, with a maximum classification accuracy of 91.71%. In terms of maximum classification precision, neither the Bayesian nor the Tree families show any improvement (44.61%, 41.11%).

Table 9 depicts the comparison of the proposed approach with the state-of-the-art models in terms of accuracy for the ISIC2019 dataset. It is obvious that our proposed approach achieves the best classification accuracy.

The results obtained by Zadeh et al. Alizadeh and Mahloojifar (2020) exceed our measurements because the authors validate their model on two classes while we validate our approach on eight classes. The obtained result in the work of Kassem et al. Kassem, Khalid Hosny and Fouad (2020) was 94.92% for 10% of the dataset only taken as a test but in our study we have used 20% and we obtain 92.34%.

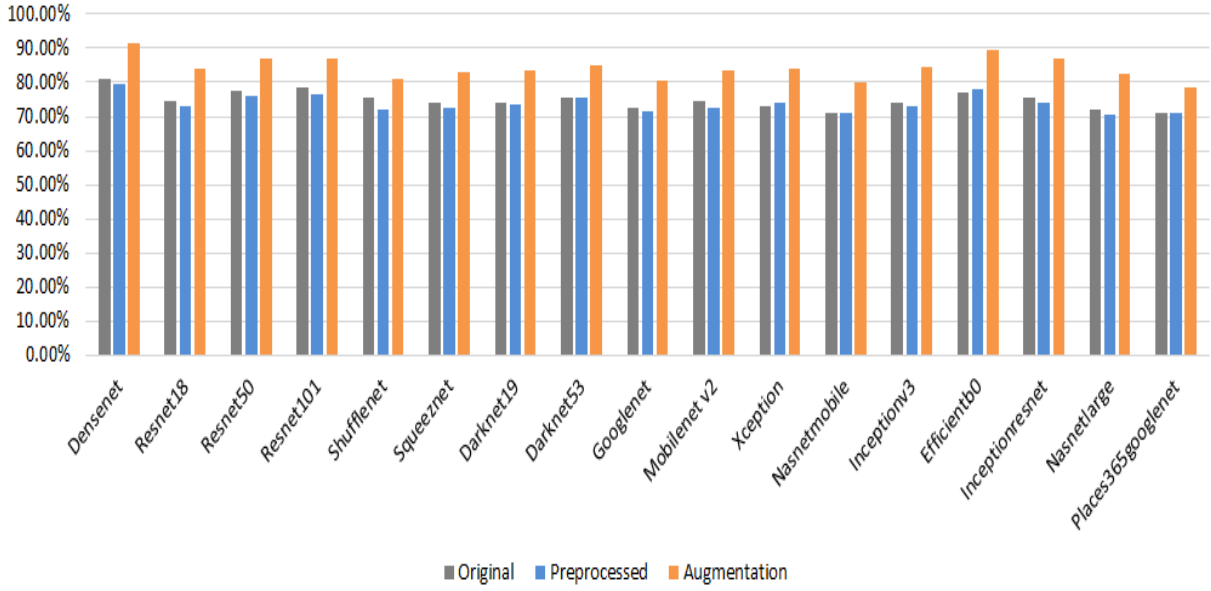


Figure 6: Accuracy according to different CNN model and Cubic SVM classifier.

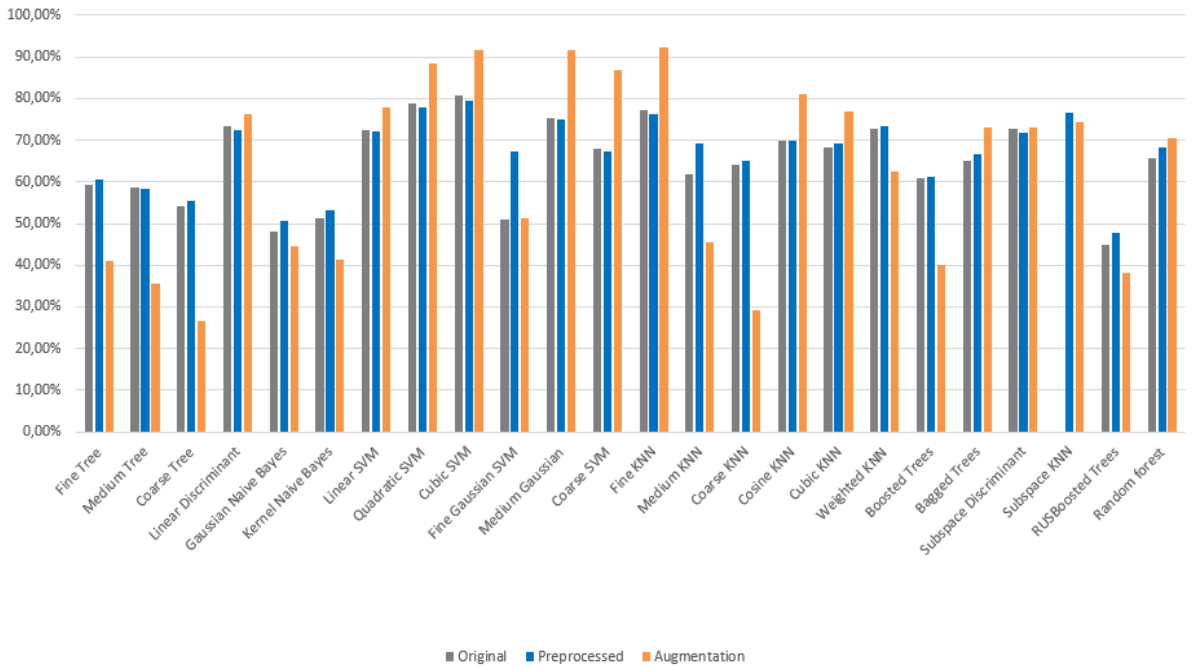


Figure 7: Accuracy according to DenseNet201 model and classifiers.

4.3. Ph2 dataset Experiment

By taking advantage of the previous experience carried out on the ISIC 2019 dataset, we applied several classifiers on the basis of the features obtained by the pre-trained DenseNet201 model. Table 10 presents the evaluation in terms of accuracy.

On the other side of our experiments, Table 11 depicts the results of the proposed approach compared to recent approaches for the PH2 dataset. The maximum classification accuracy obtained by previous works on the PH2 dataset

Table 7

Summary of different accuracy on augmented ISIC 2019 dataset according to DenseNet201 model and different classifiers

| <i>Method \ Metric</i> | Precision | Recall | Specificity | F1-score | Balanced Accuracy | Accuracy |
|------------------------|------------------|---------------|--------------------|-----------------|--------------------------|-----------------|
| Fine Tree | 39.38% | 39.83% | 91.18% | 37.8% | 39.83% | 41.11% |
| Medium Tree | 36.6% | 34.13% | 90.01% | 33.28% | 34.13% | 35.46% |
| Coarse Tree | 11.38% | 25.24% | 89.5% | 14.94% | 25.24% | 26.72% |
| Linear Discriminant | 71.61% | 76.38% | 94.83% | 73.12% | 76.38% | 76.29% |
| Gaussian Naive Bayes | 41.95% | 43.93% | 91.53% | 42.56% | 43.93% | 44.61% |
| Kernel Naive Bayes | 43.59% | 39.55% | 90.91% | 39.09% | 39.55% | 41.3% |
| Linear SVM | 72.24% | 77.89% | 94.72% | 73.62% | 77.89% | 77.74% |
| Quadratic SVM | 81.99% | 88.86% | 96.12% | 84.09% | 88.86% | 88.52% |
| Cubic SVM | 84.82% | 92.04% | 96.4% | 86.82% | 92.04% | 91.71% |
| Fine Gaussian SVM | 66.92% | 49.6% | 91.17% | 46.65% | 49.6% | 51.29% |
| Medium Gaussian | 84.18% | 91.82% | 96.26% | 86.02% | 91.82% | 91.56% |
| Coarse SVM | 31.33% | 76.69% | 80.75% | 44.49% | 86.93% | 86.69% |
| Fine KNN | 85.22% | 92.75% | 96.38% | 86.96% | 92.75% | 92.34% |
| Medium KNN | 19.43% | 76.52% | 69.29% | 30.99% | 43.63% | 45.53% |
| Coarse KNN | 34.38% | 27.01% | 88.51% | 18.31% | 27.01% | 29.12% |
| Cosine KNN | 75.4% | 81.43% | 95.8% | 77.31% | 81.43% | 80.97% |
| Cubic KNN | 72.11% | 77.11% | 94.89% | 73.58% | 77.11% | 76.96% |
| Weighted KNN | 74.47% | 60.74% | 92.77% | 58.11% | 60.74% | 62.43% |
| Boosted Trees | 37.89% | 38.75% | 91.08% | 37.93% | 38.75% | 39.96% |
| Bagged Trees | 68.65% | 73.15% | 94.85% | 70.34% | 73.15% | 73.04% |
| Subspace Discriminant | 68.79% | 73.18% | 94.46% | 70.18% | 73.18% | 73.22% |
| Subspace KNN | 69.82% | 74.53% | 94.54% | 71.2% | 74.53% | 74.4% |
| RUSBoosted Trees | 36.68% | 37.92% | 90.81% | 36.92% | 37.92% | 38.23% |

Table 8

Comparison between DenseNet201 (Fine-tuned), DenseNet201+Cubic SVM and DenseNet201+Fine KNN on augmented ISIC 2019

| <i>Method \ Metric</i> | Precision | Recall | Specificity | F1-score | Balanced Accuracy | Accuracy |
|--------------------------|------------------|---------------|--------------------|-----------------|--------------------------|-----------------|
| DenseNet201 (Fine-tuned) | 88.22% | 80.30% | 98.19% | 84.07% | 91.33% | 91.10% |
| DenseNet201+Cubic SVM | 84.82% | 92.04% | 96.4% | 86.82% | 92.04% | 91.71% |
| DenseNet201+Fine KNN | 85.22% | 92.75% | 96.38% | 86.96% | 92.75% | 92.34% |

using ABCD rule and artificial neural networks is 96.00%, whereas it is 99.00% using our proposed methodology.

5. Conclusion

In order to harness the power of deep models for the extraction of the features from skin lesion images we carried out several experiments by different classifiers. In this paper, we have proposed to use 17 commonly pre-trained convolutional neural networks (CNN) architectures as feature extractors and 24 machine learning classifiers to evaluate the classification of skin lesions from two different datasets ISIC 2019 and PH2. Three scenarios were used for the processing of the dataset: treating the base as it is, doing a preprocessing and finally exploiting the data augmentation.

Various experiment results have demonstrated that the DenseNet201 model combined with the fine Knn or cubic

Table 9

Results of the proposed approach compared to various approaches for ISIC 2019 dataset

| Authors | Method | No. of classes | Accuracy |
|------------------------------------------------------|------------------------------------------------------------|----------------|---------------|
| Alizadeh et al. Alizadeh and Mahloojifar (2020) | CNN, texture features | 2 | 96.30% |
| EL Khatib et al. El-khatib, Popescu and Ichim (2020) | Decision fusion, features based CNN, SVM | 2 | 93.00% |
| Kassem et al. Kassem et al. (2020) | GoogleNet+ SVM | 8 | 94.92% |
| Pacheco et al. Pacheco, Ali and Trappenberg (2019) | Ensemble of 13 CNNs | 8 | 89% |
| Ahmed et al. Sara Atito Ali Ahmed and Goksu (2020) | Ensembles (Xception, Inception-ResNet-V2, and NasNetLarge) | 8 | 90.6% |
| Guissoous Guissoous (2019) | Ensemble models (DenseNet201 and Inception V3) | 8 | 91.00% |
| Proposed method | DenseNet201, Cubic SVM | 8 | 91.71% |
| Proposed method | DenseNet201, Fine KNN | 8 | 92.34% |

Table 10

Summary of results on PH2 dataset

| Method | Accuracy |
|-----------------------|---------------|
| Fine Tree | 90.00% |
| Medium Tree | 90.00% |
| Coarse Tree | 90.00% |
| Linear Discriminant | 98.00% |
| Gaussian Naive Bayes | 80.00% |
| Kernel Naive Bayes | 88.00% |
| Linear SVM | 97.00% |
| Quadratic SVM | 99.00% |
| Cubic SVM | 99.00% |
| Fine Gaussian SVM | 90.00% |
| Medium Gaussian SVM | 95.00% |
| Coarse SVM | 90.00% |
| Fine KNN | 95.00% |
| Medium KNN | 88.00% |
| Coarse KNN | 95.00% |
| Cosine KNN | 90.00% |
| Cubic KNN | 90.00% |
| Weighted KNN | 88.00% |
| Boosted Trees | 90.00% |
| Bagged Trees | 83.00% |
| Subspace Discriminant | 93.00% |
| Subspace KNN | 93.00% |
| RUSBoosted Trees | 88.00% |

SVM classifiers is able to improve the performance of a fine-tuned DenseNet201.

Author contributions Every author mentioned, has provided direct and useful help, moreover, all authors read, rectified and approved this review. All authors have read and approved to submit it to your journal. There is no conflict of interest of any authors in relation to the submission. This paper has not been submitted elsewhere for consideration of publication.

Declaration of Competing Interest The authors declare that there are no conflicts of interest.

Table 11

Results of the proposed approach compared to various approaches for PH2 dataset

| Authors | Method | No. of classes | Accuracy |
|---------------------------------------------------------------------------|-------------------------------------------|----------------|---------------|
| Ballerini et al. Ballerini, Fisher, Aldridge and Rees (2013) | Color and texture, K-NN | 5 | 74.00% |
| Situ et al. Situ, Yuan, Chen and Zouridakis (2008) | Color histogram, Gabor filter, BoVW, K-NN | 2 | 82.20% |
| Mikos et al. Mikos, Sioulas, Sidiropoulos, Kalatzis and Cavouras (2012) | GLCM, PNN neural network | 2 | 69.50% |
| Wadhawan et al. Wadhawan, Situ, Hu, Lancaster, Yuan and Zouridakis (2011) | Color histogram, Haar wavelet, SVM | 2 | 76.40% |
| Upadhyay and Chandra Upadhyay and Chandra (2019) | GLOH, HSVcolor, BoVW, SVM | 6 | 78.00% |
| Arora et al. Arora, Dubey, Jaffery and Rocha (2020a) | BoF, SURF, SVM | 2 | 85.70% |
| Majumdera and Ullaha Majumder and Ullah (2019) | ABCD rule, ANN | 2 | 98.00% |
| Pereira et al. Pereira dos Santos and Ponti (2018) | MobileNet, Linear SVM | - | 95.00% |
| Alizadeh et al. Alizadeh and Mahloojifar (2020) | CNN, texture features | 2 | 97.50% |
| Ghalejoogh et al. Ghalejoogh, Kordy and Ebrahimi (2020b) | Hierarchical Structure, Stacking Approach | 2 | 96.00% |
| EL Khatib et al. El-khatib et al. (2020) | Decision fusion, features based CNN, SVM | 2 | 95.00% |
| Proposed method | DenseNet201, Cubic SVM | 2 | 99.00% |

Acknowledgement This work was completed as part of the Hubert Curien Partnership (PHC) TASSILI cooperation program between Algeria and France under the project code 19MDU212.

References

- Abbas, Q., Celebi, M.E., 2019. Dermodeep-a classification of melanoma-nevus skin lesions using multi-feature fusion of visual features and deep neural network. *Multimedia Tools and Applications* , 1–22.
- Abbasi, N.R., Shaw, H., Rigel, D., Friedman, R., McCarthy, W., Osman, I., Kopf, A., Polsky, D., 2004. Early diagnosis of cutaneous melanoma: revisiting the abcd criteria. *JAMA* 292 22, 2771–2776.
- Abuzaghlh, O., Faezipour, M., Barkana, B., 2015. A comparison of feature sets for an automated skin lesion analysis system for melanoma early detection and prevention. 2015 Long Island Systems, Applications and Technology , 1–6.
- Ahmed, S.A.A., Yanikoglu, B.A., Göksu, Ö., Aptoula, E., 2020. Skin lesion classification with deep cnn ensembles. 2020 28th Signal Processing and Communications Applications Conference (SIU) , 1–4.
- Alizadeh, S.M., Mahloojifar, A., 2020. Automatic skin cancer detection in dermoscopy images by combining convolutional neural networks and texture features. *International Journal of Imaging Systems and Technology* , 1–13.
- Alzubaidi, L., Fadhel, M., Al-Shamma, O., Zhang, J., Duan, Y., 2020. Deep learning models for classification of red blood cells in microscopy images to aid in sickle cell anemia diagnosis. *Electronics* 9, 427.
- Aria, M., D' Ambrosio, A., Iorio, C., Siciliano, R., Cozza, V., 2020. Dynamic recursive tree-based partitioning for malignant melanoma identification in skin lesion dermoscopic images 61, 1645–1661.
- Arora, G., Dubey, A., Jaffery, Z.A., Rocha, A., 2020a. Bag of feature and support vector machine based early diagnosis of skin cancer. *Neural Computing and Applications* , 1–8.
- Arora, G., Dubey, A.K., Jaffery, Z.A., Rocha, A., 2020b. Bag of feature and support vector machine based early diagnosis of skin cancer .
- Aurelia, J.E., Rustam, Z., Wibowo, V.V.P., Setiawan, Q.S., 2020. Comparison between convolutional neural network and convolutional neural network-support vector machines as the classifier for colon cancer. 2020 International Conference on Decision Aid Sciences and Application (DASA) , 812–816.
- Avellaneda, F., 2019. Learning optimal decision trees from large datasets. *ArXiv abs/1904.06314*.
- Ballerini, L., Fisher, R.B., Aldridge, B., Rees, J., 2013. A color and texture based hierarchical k-nn approach to the classification of non-melanoma skin lesions. *Springer Netherlands* , 63–86.
- Barata, C., Celebi, M.E., Marques, J., 2019. A survey of feature extraction in dermoscopy image analysis of skin cancer. *IEEE Journal of Biomedical and Health Informatics* 23, 1096–1109.
- Basly, H., Ouarda, W., Sayadi, F., Ouni, B., Alimi, A., 2020. Cnn-svm learning approach based human activity recognition. *Image and Signal Processing* 12119, 271 – 281.
- Bodapati, J., Veeranjanyulu, N., 2019. Feature extraction and classification using deep convolutional neural networks.
- Breiman, L., 1996. Bagging predictors. *Machine Learning* 24, 123–140.
- Breiman, L., 2001. Random forests. *Machine Learning* 4, 5–32.

- Celebi, M.E., Codella, N., Halpern, A., 2019. Dermoscopy image analysis: Overview and future directions. *IEEE Journal of Biomedical and Health Informatics* 23, 474–478.
- Celebi, M.E., Wen, Q., Iyatomi, H., Shimizu, K., Zhou, H., Schaefer, G., 2015. A state-of-the-art survey on lesion border detection in dermoscopy images.
- Chakravorty, R., Liang, S., Abedini, M., Garnavi, R., 2016. Dermatologist-like feature extraction from skin lesion for improved asymmetry classification in ph2 database. 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) , 3855–3858.
- Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 1800–1807.
- Combalia, M., Codella, N.C.F., Rotemberg, V., Helba, B., Vilaplana, V., Reiter, O., Halpern, A., Puig, S., Malvehy, J., 2019. Bcn20000: Dermoscopic lesions in the wild. *ArXiv abs/1908.02288*.
- Dhivyaa, C.R., Sangeetha, K., Balamurugan, M., Amaran, S., Vetrivel, T., Johnpaul, P., 2020. Skin lesion classification using decision trees and random forest algorithms .
- Efron, B., Tibshirani, R.J., 1993. An introduction to the bootstrap. Chapman & Hall, New York.
- El-khatib, H., Popescu, D., Ichim, L., 2020. Deep learning based methods for automatic diagnosis of skin lesions. *Sensors (Basel, Switzerland)* 20, 1–25.
- Filali, Y., Khoukhi, H.E., Sabri, M.A., Aarab, A., 2020. Efficient fusion of handcrafted and pre-trained cnns features to classify melanoma skin cancer. *Multimedia Tools and Applications* , 1–20.
- Fisher, R.B., Rees, J., Bertrand, A., 2019. Classification of ten skin lesion classes: Hierarchical KNN versus deep net, in: Zheng, Y., Williams, B.M., Chen, K. (Eds.), *Medical Image Understanding and Analysis*. Springer International Publishing, volume 1065, pp. 86–98. Series Title: *Communications in Computer and Information Science*.
- Ghalejoogh, G.S., Kordy, H.M., Ebrahimi, F., 2020a. A hierarchical structure based on stacking approach for skin lesion classification. *Expert Syst. Appl.* 145, 113–127.
- Ghalejoogh, G.S., Kordy, H.M., Ebrahimi, F., 2020b. A hierarchical structure based on stacking approach for skin lesion classification. *Expert Syst. Appl.* 145, 113–127.
- Goodfellow, I., Bengio, Y., Courville, A., 2016. *Deep Learning*. The MIT Press.
- Govindaswamy, A., Montague, E., Raicu, D., Furst, J., 2020. Cnn as a feature extractor in gaze recognition. 2020 3rd Artificial Intelligence and Cloud Computing Conference .
- Grandini, M., Bagli, E., Visani, G., 2020. Metrics for multi-class classification: an overview. *ArXiv abs/2008.05756*.
- Guissous, A.E., 2019. Skin lesion classification using deep neural network. *ArXiv abs/1911.07817*.
- Gutman, D., Codella, N.C.F., Celebi, M.E., Helba, B., Marchetti, M., Mishra, N., Halpern, A., 2018. 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). 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018) , 168–172.
- Guyon, I., Gunn, S., M., N., Zadeh, L., 2006. Feature extraction - foundations and applications, in: *Feature Extraction*.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 770–778.
- Henning, J., Dusza, S., Wang, S., Marghoob, A., Rabinovitz, H., Polsky, D., Kopf, A., 2007. The cash (color, architecture, symmetry, and homogeneity) algorithm for dermoscopy. *Journal of the American Academy of Dermatology* 56 1, 45–52.
- Hoshyar, A.N., Al-Jumaily, A., 2014. Comparing the performance of various filters on skin cancer images. *Procedia Computer Science* 42, 32–37.
- Hosny, K., Kassem, M.A., Foad, M.M., 2018. Skin cancer classification using deep learning and transfer learning. 2018 9th Cairo International Biomedical Engineering Conference (CIBEC) , 90–93.
- Howard, A., 2014. Some improvements on deep convolutional neural network based image classification. *CoRR abs/1312.5402*.
- Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv abs/1704.04861*.
- Huang, G., Liu, Z., Weinberger, K.Q., 2017. Densely connected convolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 2261–2269.
- Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W., Keutzer, K., 2017. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size. *ArXiv abs/1602.07360*.
- Jadhav, A.R., Ghontale, A.G., Shrivastava, V.K., 2019. Segmentation and border detection of melanoma lesions using convolutional neural network and SVM, in: Verma, N.K., Ghosh, A.K. (Eds.), *Computational Intelligence: Theories, Applications and Future Directions - Volume I*. Springer Singapore, volume 798, pp. 97–108. Series Title: *Advances in Intelligent Systems and Computing*.
- Javed, R., Rahim, M., Saba, T., Rehman, A., 2019. A comparative study of features selection for skin lesion detection from dermoscopic images. *Network Modeling Analysis in Health Informatics and Bioinformatics* 9.
- Karungaru, S., Dongyang, L., Terada, K., 2021. Vehicle detection and type classification based on cnn-svm. *International Journal of Machine Learning and Computing* 11, 304–310.
- Kassani, S.H., Kassani, P.H., 2019. A comparative study of deep learning architectures on melanoma detection. *Tissue & Cell* 58, 76–83.
- Kassem, M.A., Khalid Hosny, M., Fouad, M., 2020. Skin lesions classification into eight classes for isic 2019 using deep convolutional neural network and transfer learning. *IEEE Access* 8, 114822–114832.
- Kolkur, S., Kalbande, D., 2016. Survey of texture based feature extraction for skin disease detection. 2016 International Conference on ICT in Business Industry & Government (ICTBIG) , 1–6.
- Kraus, O.Z., Grys, B.T., Ba, J., Chong, Y., Frey, B., Boone, C., Andrews, B., 2017. Automated analysis of high content microscopy data with deep learning. *Molecular Systems Biology* 13.
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*

60, 84–90.

- Larose, D.T., Larose, C.D., 2014. Discovering Knowledge in Data: an Introduction to Data Mining. Wiley.
- LeCun, Y., Jackel, L., Bottou, L., Cortes, C., Denker, J., Drucker, H., Guyon, I., Muller, U., Sackinger, E., Simard, P., Vapnik, V., 1995. Learning algorithms for classification: A comparison on handwritten digit recognition , 261–276.
- Lee, T., Ng, V., Gallagher, R., Coldman, A., McLean, D., 1997. Dullrazor: a software approach to hair removal from images. Computers in biology and medicine 27 6, 533–43.
- Maglogiannis, I., Zafiropoulos, E., Kyranoudis, C., 2006. Intelligent segmentation and classification of pigmented skin lesions in dermatological images, in: SETN.
- Mahbod, A., Ecker, R., Ellinger, I., 2019. Skin lesion classification using hybrid deep neural networks. ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) , 1229–1233.
- Majumder, S., Ullah, M.A., 2019. A computational approach to pertinent feature extraction for diagnosis of melanoma skin lesion. Pattern Recognition and Image Analysis 29, 503–514.
- Melbin, K., Raj, Y.J.V., 2021. Integration of modified ABCD features and support vector machine for skin lesion types classification 80, 8909–8929.
- Mendonçan, T., Ferreira, P., Marques, J., Marçal, A., Rozeira, J., 2013. Ph2 - a dermoscopic image database for research and benchmarking. 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) , 5437–5440.
- Mikos, E., Sioulas, I., Sidiropoulos, K., Kalatzis, I., Cavouras, D., 2012. An android-based pattern recognition application for the characterization of epidermal melanoma.
- Miksik, O., Mikolajczyk, K., 2012. Evaluation of local detectors and descriptors for fast feature matching. Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012) , 2681–2684.
- Moura, N., Veras, R., Aires, K., Machado, V.P., Silva, R., Araújo, F.H.D., Claro, M., 2018. Combining abcd rule, texture features and transfer learning in automatic diagnosis of melanoma. IEEE Symposium on Computers and Communications (ISCC) , 508–513.
- Mporas, I., Perikos, I., Paraskevas, M., 2020. Color models for skin lesion classification from dermoscopic images, in: Hatzilygeroudis, I., Perikos, I., Grivokostopoulou, F. (Eds.), Advances in Integrations of Intelligent Methods. Springer Singapore. volume 170, pp. 85–98. Series Title: Smart Innovation, Systems and Technologies.
- Mu, N., Qiao, D., 2019. Image classification based on convolutional neural network and support vector machine. 2019 6th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS) , 248–252.
- Murugan, A., Nair, S.H., Kumar, K.P.S., 2019. Detection of skin cancer using SVM, random forest and kNN classifiers 43, 269.
- Nachbar, F., Stolz, W., Merkle, T., Cognetta, A., Vogt, T., Landthaler, M., Bilek, P., Braun-falco, O., Plewig, G., 1994. The abcd rule of dermatoscopy. high prospective value in the diagnosis of doubtful melanocytic skin lesions. Journal of the American Academy of Dermatology 30 4, 551–559.
- Neapolitan, R., 2007. Learning bayesian networks, in: KDD '07.
- Oltu, B., Güney, S., Dengiz, B., Ağildere, M., 2021. Automated tuberculosis detection using pre-trained cnn and svm. 2021 44th International Conference on Telecommunications and Signal Processing (TSP) , 92–95.
- Öznur Özaltn, Özgür Yeniay, 2021. Ecg classification performing feature extraction automatically using a hybrid cnn-svm algorithm. 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) , 1–5.
- Ozkan, I.A., Koklu, M., 2017. Skin lesion classification using machine learning algorithms. International Journal of Intelligent Systems and Applications in Engineering 5, 285–289.
- Pacheco, A.G.C., Ali, A.R., Trappenberg, T., 2019. Skin cancer detection based on deep learning and entropy to detect outlier samples. ArXiv abs/1909.04525.
- Patil, M., Dongre, N., 2020. Melanoma detection using HSV with SVM classifier and de-duplication technique to increase efficiency, in: Chaubey, N., Parikh, S., Amin, K. (Eds.), Computing Science, Communication and Security. Springer Singapore. volume 1235, pp. 109–120. Series Title: Communications in Computer and Information Science.
- Perez, F., Avila, S., Valle, E., 2019. Solo or ensemble? choosing a cnn architecture for melanoma classification. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) , 2775–2783.
- Perez, F., Vasconcelos, C., Avila, S., Valle, E., 2018. Data augmentation for skin lesion analysis, in: OR 2.0/CARE/CLIP/ISIC@MICCAI.
- P.V. Asha Deepika, B. Yamini, C.P.T.V., R, A., 2020. Classification of skin lesions using svm via deep learning feature network. International Journal of Advanced Science and Technology 29, 4526–4538.
- Redmon, J., Divvala, S., Girshick, R.B., Farhadi, A., 2016. You only look once: Unified, real-time object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 779–788.
- Redmon, J., Farhadi, A., 2017. Yolo9000: Better, faster, stronger. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 6517–6525.
- Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. ArXiv abs/1804.02767.
- Salahat, E., Qasaimeh, M., 2017. Recent advances in features extraction and description algorithms: A comprehensive survey. IEEE International Conference on Industrial Technology (ICIT) , 1059–1063.
- Salido, J.A.A., Ruiz, C.R., 2018. Using deep learning to detect melanoma in dermoscopy images, pp. 61–68.
- Pereira dos Santos, F., Ponti, M.A., 2018. Robust feature spaces from pre-trained deep network layers for skin lesion classification, in: 31st Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 33–40.
- Sara Atito Ali Ahmed, Berrin Yanikoglu, E.A., Goksu, O., 2020. Skin lesion classification with deep cnn ensembles. 28th Signal Processing and Communications Applications Conference (SIU) .
- Schapire, R., 2005. The strength of weak learnability. Machine Learning 5, 197–227.
- Situ, N., Yuan, X., Chen, J., Zouridakis, G., 2008. Malignant melanoma detection by bag-of-features classification. 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society , 3110–3113.
- Smith, C., Koning, M., 2017. Decision Trees and Random Forests: A Visual Introduction for Beginners. Blue Windmill Media.

- Storcheus, D., Rostamizadeh, A., Kumar, S., 2015. A survey of modern questions and challenges in feature extraction, in: FE@NIPS.
- Suganthi, M., Sathiaselvan, J., 2020. An exploratory of hybrid techniques on deep learning for image classification. 2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP), 1–4.
- Surówka, G., Ogorzalek, M., 2019. Naive bayes learning of dermoscopy images, in: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J.M. (Eds.), Artificial Intelligence and Soft Computing. Springer International Publishing. volume 11509, pp. 294–304. Series Title: Lecture Notes in Computer Science.
- Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning, in: AAAI.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1–9.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2818–2826.
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A survey on deep transfer learning, in: The 27th International Conference on Artificial Neural Networks (ICANN 2018).
- Tan, M., Le, Q.V., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. ArXiv abs/1905.11946.
- Tschanl, P., Rosendahl, C., Akay, B.N., Argenziano, G., Blum, A., Braun, R., Cabo, H., Gourhant, J., Kreusch, J., Lallas, A., Lapins, J., Marghoob, A., Menzies, S., Neuber, N., Paoli, J., Rabinovitz, H., Rinner, C., Scope, A., Soyer, H., Sinz, C., Thomas, L., Zalaudek, I., Kittler, H., 2019. Expert-level diagnosis of nonpigmented skin cancer by combined convolutional neural networks. JAMA Dermatology 155, 58–65.
- Tschanl, P., Rosendahl, C., Kittler, H., 2018. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific Data 5.
- Upadhyay, P.K., Chandra, S., 2019. An improved bag of dense features for skin lesion recognition. Journal of King Saud University - Computer and Information Sciences .
- Valle, E., Fornaciali, M., Menegola, A., Tavares, J., Bittencourt, F., Li, L., Avila, S., 2020. Data, depth, and design: Learning reliable models for skin lesion analysis. Neurocomputing 383, 303–313.
- Vapnik, V., 1998. Statistical learning theory.
- Wadhawan, T., Situ, N., Hu, R., Lancaster, K., Yuan, X., Zouridakis, G., 2011. Implementation of the 7-point checklist for melanoma detection on smart handheld devices. Annual International Conference of the IEEE Engineering in Medicine and Biology Society , 3180–3183.
- Walter, F., Prevost, A., Vasconcelos, J., Hall, P., Burrows, N., Morris, H.C., Kinmonth, A.L., Emery, J., 2013. Using the 7-point checklist as a diagnostic aid for pigmented skin lesions in general practice: a diagnostic validation study. The British journal of general practice : the journal of the Royal College of General Practitioners 63 610, e345–e353.
- Wang, J., MacKenzie, J., Ramachandran, R., Chen, D., 2015. Neutrophils identification by deep learning and voronoi diagram of clusters, in: Medical Image Computing and Computer-Assisted Intervention-MICCAI, pp. 226–233.
- Wang, P., Zhang, X., Hao, Y., 2019. A method combining cnn and elm for feature extraction and classification of sar image. J. Sensors 2019, 6134610:1–6134610:8.
- Wighton, P., Lee, T., Lui, H., Mclean, D., Atkins, M., 2011. Chromatic aberration correction: an enhancement to the calibration of low cost digital dermoscopes. Skin Research and Technology 17.
- Wu, R., Yan, S., Shan, Y., Dang, Q., Sun, G., 2015. Deep image: Scaling up image recognition. ArXiv abs/1501.02876.
- Xu, Y., Zhu, Q., Fan, Z., Qiu, M., Chen, Y., Liu, H., 2013. Coarse to fine k nearest neighbor classifier. Pattern Recognit. Lett. 34, 980–986.
- Yu, L., Chen, H., Dou, Q., Qin, J., Heng, P., 2017. Automated melanoma recognition in dermoscopy images via very deep residual networks. IEEE Transactions on Medical Imaging 36, 994–1004.
- Yu, Z., Zhuoyi, S., 2013. Binary decision trees for melanoma diagnosis, Springer International Publishing. volume 7872, pp. 374–385. Series Title: Lecture Notes in Computer Science.
- Zalaudek, I., Argenziano, G., Soyer, H., Corona, R., Sera, F., Blum, A., Braun, R., Cabo, H., Ferrara, G., Kopf, A., Langford, D., Menzies, S., Pellacani, G., Peris, K., Seidenari, S., 2006. Three-point checklist of dermoscopy: an open internet study. British Journal of Dermatology 154.
- Zhang, X., Zhou, X., Lin, M., Sun, J., 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. IEEE/CVF Conference on Computer Vision and Pattern Recognition , 6848–6856.
- Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V., 2018. Learning transferable architectures for scalable image recognition. IEEE/CVF Conference on Computer Vision and Pattern Recognition , 8697–8710.