Skin Cancer Recognition for Low Resolution Images

Chapter · January 2022

DOI: 10.1007/978-3-030-96600-3_10

4 authors, including:

Tatiana Jaworska
Instytut Badań Systemowych Polskiej Akademii Nauk
26 PUBLICATIONS **73** CITATIONS
SEE PROFILE

Maria Ganzha
Warsaw University of Technology
286 PUBLICATIONS **1,736** CITATIONS
SEE PROFILE

Marcin Paprzycki
Instytut Badań Systemowych Polskiej Akademii Nauk
472 PUBLICATIONS **3,899** CITATIONS
SEE PROFILE

Skin cancer recognition for low resolution images

Michał Kortała[1][0000−0003−0837−8633], Tatiana Jaworska[2][0000−0001−5399−8474], Maria Ganzha[1,2][0000−0001−7714−4844], and Marcin Paprzycki[2][0000−0002−8069−2152]

[1] Warsaw University of Technology, Warsaw, Poland
[2] Systems Research Institute Polish Academy of Sciences, Warsaw, Poland
firstname.lastname@ibspan.waw.pl

Abstract. Substantial body of work has been devoted to skin cancer recognition in high-resolution medical images. However, nowadays photos of skin lesions can be taken by mobile phones, where quality of image is reduced. The aim of this contribution is report on skin cancer recognition when applying machine learning to "low resolution" images. Experiments have been performed on the dataset from the ISIC 2018 Challenge.

Keywords: Skin cancer · Machine Learning · Deep Neural Network.

1 Introduction

Number of cases of skin cancer is raising. In 2017, in Poland, 7,000 new cases among women and 6,453 among men were detected, an increase of 9% compared to 2016 [37]. Currently, skin cancer diagnosis involves: (a) visual examination, and (b) skin biopsy. Too often, patients see the doctor in later stages of cancer, while early detection could ensure 5-year survival (with certainty of 99%). Beginning of treatment requires early cancer awareness. One possible solution would be a simple diagnosis tool, which would automate initial diagnosis.

With progress in image processing it should be possible to facilitate early skin cancer detection. Organizations, such as the International Skin Imaging Collaboration (ISIC), actively seek solutions by organizing competitions in this area, and engaging the research community. For instance, the ISIC 2018 Challenge [12,42] included diagnosis of skin cancer on the basis of image analysis.

This work proposes a diagnosis system for devices with restricted resources, such as mobile phones. This is achieved by reducing the resolution of photos, rescaled to size 100×100. This allows to speed the model training and reduces the size of the model achieved (e.g. to fit the memory of standard phones).

2 Related work

2.1 Image preprocessing

Before applying machine learning (ML) images have to be suitably prepared. Numerous image preprocessing techniques have been explored. Farooq et al. [17]

applied hair removal ([27]) to clean the image. Color space transformation were used to separate color from luminance. Here, Sarkar et al. [35] proposed transformation to CIEL*a*b, Abbas et al. [1] to CIECAM02, and Kumar et al. [25] to HSV and to YCbCr color spaces. Sarkar et al. [35] used color space transformation, together with luminescence channel enhancement, applying CLAHE algorithm. Preprocessing reported in Kumar et al. [25] explored feature extraction methods, e.g. conversion to grayscale, binary mask, sharpening filter, smooth filter, adjust histograms, median filter, RGB extraction and Sobel operator.

In computer vision, a region of interest (ROI) is a part of an image identified for a particular purpose. Abbas et al. [1], used region of interest selection, and structural feature extraction, with steerable pyramid transform. An approach presented in [26], used a threshold and statistical region merging (SRM) to generate a binary mask. Moreover, Karhunen-Loeve transform, histogram equalization, and contrast enhancement were applied. Sah et al. [34] proposed image augmentation via rotation, shift, and reflection. The preprocessing step often involves convolutional neural network (CNN; see, [3]). The role of CNN is to classify input images based on features extracted in a training process. In Sedigh et al. [36], dataset augmentation, by image generation, used a generative adversarial network, to solve the problem of insufficient number of images in the training set, by creating synthetic samples. Other preprocessing methods included grey image scale, noise filtering, principal component analysis, gray level co-occurrence matrix [4], border detection [38] and color normalization [8,7].

2.2 Classifiers

After extraction of features from preprocessed images, classification ensues. Typically it involves CNNs, sometimes with transfer learning. Following approaches have been reported (for details, readers are directed to the references):

- Inception-v3 [41,32,17,16]
- InceptionResNetv2 [40,2,32]
- ResNet152 [19,2,32]
- DenseNet201 [22,32,8]
- MobileNet [21,17]
- AlexNet [24,10]
- VGG16 [44,34]
- Custom architectures [36,23]

2.3 Ensembles

Models described in Section 2.2 were often grouped into an ensemble of models, aimed at enhancing prediction of its members. Numerous publications suggested using one of the following approaches for building hierarchical classifiers. The first solution consists of a neural network trained on each model's output. In the second one [23,29] voting was used. Sun et al. [39] proposed an approach consisting of a dedicated ensemble for each class predicted by a classifier. Summary of pertinent models is shown in Table 1.

Authors	Summary
Farooq et al. [17]	Inception-v3, MobileNet with transfer learning
Habib et al. [32]	CNN with transfer learning
Esteva et al. [16]	Inception-v3 with transfer learning
Sarkar et al. [35]	ResNet, CNN
Abbas et al. [1]	AdaBoost
Kumar et al. [25]	neural network, k-Near Neighbours, decision trees
Sah et al. [34]	VGG16 with transfer learning
Capdehurat et al. [11]	AdaBoost
Ahmad et al. [2]	CNN with triplet loss
Lau et al. [26]	neural network, auto-associative neural network
Masood et al. [30]	SVM with incremental use of unlabeled data
Sedigh et al. [36]	custom built CNN
Alquran et al. [4]	SVM
Suganya [38]	SVM
Barata et al. [8]	DenseNet-161 with transfer learning
Dorj et al. [15]	AlexNet with transfer learning, ECOC SVM

Table 1: Summary of proposed solutions in the area of skin cancer classification.

2.4 Datasets

There exist multiple datasets for skin cancer classification. They include from 2 to 10 classes. Most of them are small, e.g. DERMQUEST (76 images), PH2 (200), or Dermweb (320). However, Dermnet has 5500 images, while the dataset for the 2018 ISIC challenge consisted of 10,015 images.

2.5 Results reported in key publications

Results found in relevant publications have been obtained for different datasets. This makes it difficult to fairly compare prediction quality. Nevertheless, reported results can be presented collectively (in Table 2) to illustrate key trends.

Publications with best results (above 90% of accuracy) were obtained on relatively small datasets; e.g. Sarkar et al. ([35]) used a 700 image subset of the ISIC challenge dataset. Apart from models proposed in [16] and [13], all results were trained and tested on datasets smaller than the ISIC 2018 dataset. The dataset from the ISIC Challenge 2018 is described in [13]. Here, the training set consists of 10,015 images, while test set of 1,512 images (see, also Section 3). The best reported result, *in terms of balanced accuracy*, was 88.5%.

2.6 Existing software

Currently multiple software solutions are available, e.g. neural network libraries – Tensorflow, Pytorch, and Matlab toolbox. In the preprocessing stage, OpenCV implementation is often used, and the scikit-image Python package. Cited references used Tensorflow for neural networks [8,2,34] and Matlab [38,26]. In some works reported in Table 1 a software source was not specified [4,36,30,11].

Authors	Dataset size	Accuracy	Sensitivity	Specificity	F1	ROC AUC
Farooq et al. [17]	2637	86 %	89 %	83 %	84 %	-
Habib et al. [32]	10,015	-	-	-	89.01 %	0.987
Esteva et al. [16]	129,450	-	-	-	-	0.910
Sarkar et al. [35]	700	97.86 %	-	-	-	-
Abbas et al. [1]	1039	-	89.28 %	93.75 %	-	0.986
Kumar et al. [25]	N/A	95 %	-	-	-	-
Sah et al. [34]	5,500	82 %	83 %	82 %	-	-
Capdehurat et al. [11]	655	-	90 %	85 %	-	0.937
Ahmad et al. [2]	6,144	87.42 %	97.04 %	96.48 %	-	-
Lau et al. [26]	N/A	73 %	-	-	-	-
Masood et al. [30]	3,800	94 %	-	-	-	-
Sedigh et al. [36]	97	71 %	68 %	74 %	70 %	-
Alquran et al. [4]	N/A	92.1 %	-	-	-	-
Suganya [38]	320	96.8 %	95.4 %	89.3 %	-	-
Barata et al. [8]	2,000	70 %	-	-	-	0.876
Dorj et al. [15]	3,753	94.2 %	87.83 %	90.74 %	-	-
Codella et al. [13]	10,015	88.5 %	-	-	-	-

Table 2: Best results reported in listed contributions.

3 Dataset description

Results reported here concern multi-class classification, using data available in Task 3 of the ISIC 2018 Challenge competition [12,42]. This dataset includes dermatoscopic photos, from different populations, obtained and stored in different modalities. The training part of the dataset contains 10,015 images. Available photos present a representative collection of dermatologically relevant categories of pigmented skin lesions. Categories present in the dataset include:

1. Melanoma (mel)
2. Melanocytic nevus (nv)
3. Basal cell carcinoma (bcc)
4. Actinic keratosis / Bowen's disease (akiec)
5. Benign keratosis (bkl)
6. Dermatofibroma (df)
7. Vascular lesion (vasc)

Here, Basal cell carcinoma and melanoma are severe skin cancers. Actinic keratosis is the most common form of precancer that develops on skin damaged by ultraviolet rays, and may evolve into Bowen's disease. Melanocytic nevus (common mole), is a benign skin lesion that is visually similar to (and difficult to distinguish from) melanoma. Seborrheic keratosis is a noncancerous skin lesion that some people develop as they age. Dermatofibroma is also a benign skin lesion. Majority of vascular lesions are benign. However, rarely, malignant variants can materialize. Sample images of each class, are shown in Figure 1.

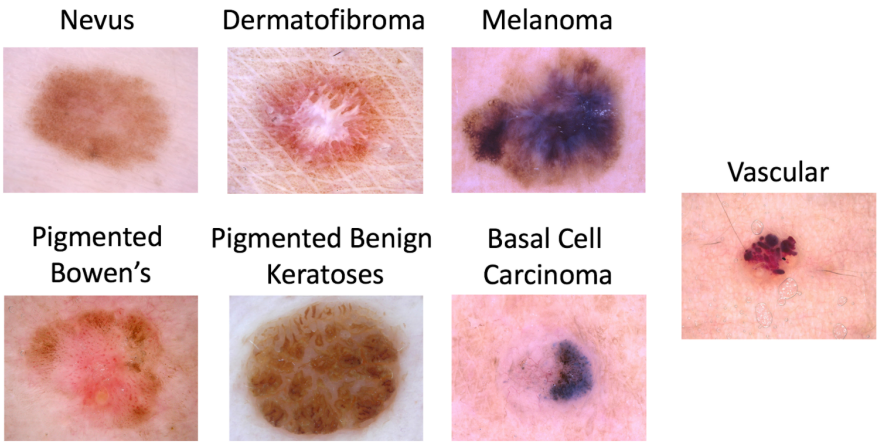


Fig. 1: Example of each database class.

Existence of seven different classes makes the problem a multi-class classification problem. This, limits the number of possible approaches, as not every "standard classifier" supports multi-class classification.

Another feature of the dataset is extreme class imbalance. The largest category is more than 58 times larger than the smallest one, and more than 6 times larger than the second largest. This imbalance is illustrated in Figure 2. This lack of balance adversely affects majority of classifiers, as they try to minimize the global error. Hence, the classifiers treat the minority classes as a "noise".

In this work, images have been resized to 100×100, from the original size of 450×600. In doing so, the images suffered some loss of details and a dimensional disturbance. However, would enable using such dataset on machines with restricted resources, and in a reasonable time. Moreover, the cost of collecting and storing data would be lowered. Finally, it would support the use of data acquired from devices from different manufacturers, which vary in size and proportions.

4 Image preprocessing

4.1 Hair removal

In the dataset, some skin lesions are partially covered by body hair, which can be a problem in the feature extraction and the region of interest segmentation, due to obscuring of essential features of the lesion. in order to eliminate this problem, a hair removal algorithm was developed, inspired by [27]. This algorithm was chosen, instead of one proposed in Lee et al. [27], because of is speed. The method consists of:

1. Grayscale image transformation.

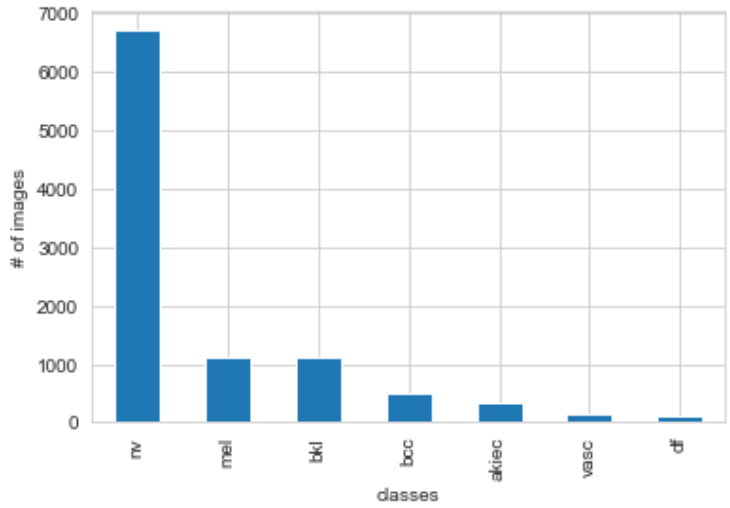


Fig. 2: Images numerosity in the dataset.

2. Performing blackhat morphological operation.
3. Creating a mask by applying thresholding.
4. Inpainting mask pixels, in the original image, using the region neighborhood.

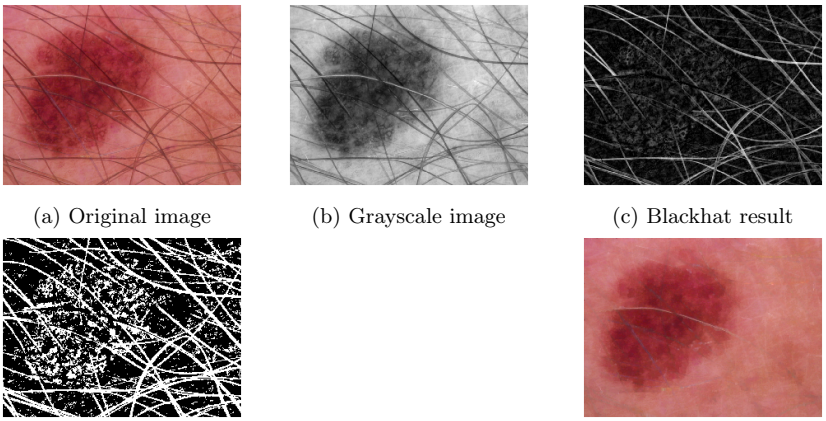
Here, when creating mask (applying thresholding), threshold value of 10 was used (every pixel with a value less than 10 was considered a skin pixel, rather than hair pixel). Pixels were filled starting at the region boundary, with the Fast Marching Method, to select the next pixel to fill the region. When hair is blonde and the skin lesion underneath it is dark, a low value of the threshold ensures that the hair will be removed, although small cavities may occur in the lesion. These will be fixed by refilling, on the basis of their neighborhood. Result of each step are shown in Figure 3.

Proposed algorithm successfully removes hair from skin lesion images, especially thick ones, which have the worst impact on lesion visibility. All of lesion surface is clearly visible, and none of its regions are hidden behind hair.

4.2 Color preprocessing

The images in the dataset have been acquired from multiple sources and "differ in appearance", among others, due to the light conditions. Leveling light conditions across the dataset improves the stability of classification systems [9]. To address this problem, the Shades of Gray color constancy algorithm [18], was applied. It is to adjust image colors to look like colors under a "standard illumination". The applied approach consists of two phases:

1. Color estimation of the illuminant in the RGB color space.



(a) Original image (b) Grayscale image (c) Blackhat result

(d) Mask (e) Hairless image

Fig. 3: Illustration of main steps of the hair detection and removal.

2. Transformation of image using the estimated light source.

Beneath, equation 1 was used to estimate the color of a standard light source.

$$\left(\frac{\int I_c(\mathbf{x})^p \, d\mathbf{x}}{\int d\mathbf{x}}\right)^{\frac{1}{p}} = K e_c. \tag{1}$$

where:

- I_c stands for c-th image channel of an image I,
- $\mathbf{x} = (i, j)$ is the point describing the position of a pixel,
- K is a normalization constant, which guarantees that $\mathbf{e} = [e_R \; e_G \; e_B]^T$ vector is a unit vector,
- p is a parameter of the Minkowski norm.

In the implementation, Minkowski norm with $p = 6$ was chosen, as it has delivered the best results in [9,18]. After calculating the $\mathbf{e}$ vector, the image can be transformed according to equation 2, which is applied to each pixel.

$$\begin{pmatrix} \mathrm{I}_R^* \\ \mathrm{I}_G^* \\ \mathrm{I}_B^* \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{3}e_R} & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}e_G} & 0 \\ 0 & 0 & \frac{1}{\sqrt{3}e_B} \end{pmatrix} \begin{pmatrix} I_R \\ I_G \\ I_B \end{pmatrix} \tag{2}$$

Example results of performing color constancy algorithm on four images of melanocytic nevus skin cancer class are presented in Figure 4.

In Figure 4, initially, skin color varied greatly between images. After the transformation, there is almost no difference in perception of similar colors; modified images look “similar”, independently of the original light conditions.

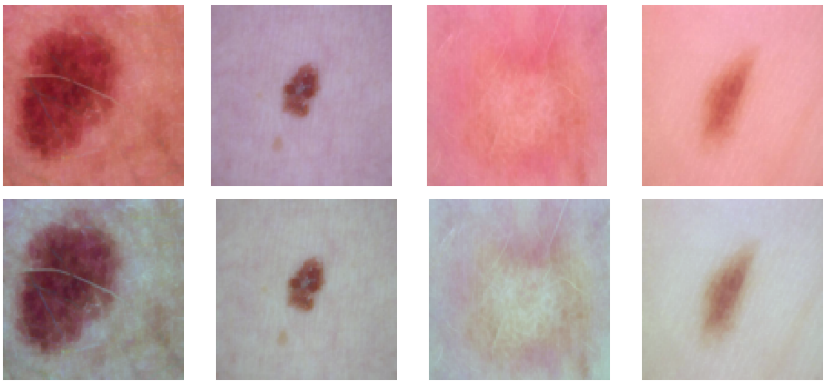


Fig. 4: Color constancy applied to melanocytic nevus images. First row contains original images, second row contains images after transformation.

4.3 Histogram equalization

Even after two steps of preprocessing, not every skin lesion had a clear border, or clearly visible details. Hence, the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm was applied. The CLAHE differs from classic histogram equalization algorithms in two key aspects:

- It operates on neighborhood regions, instead of the full image.
- It limits the intensification, by trimming the histogram at the preset value and redistributing that part of the histogram uniformly across all bins.

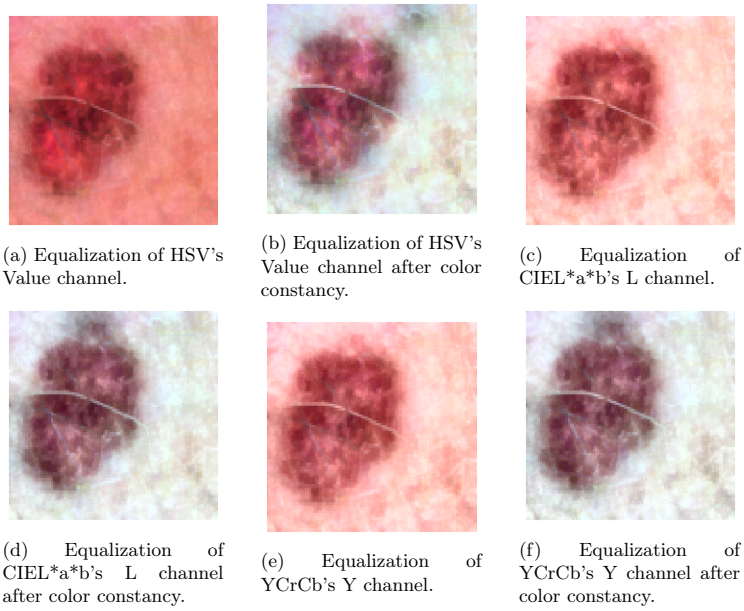
Equalizing histogram in regions, instead of the full image, tends to work better on images with regions that are considerably darker, or lighter, than the rest of the image, as it provides better enhancement of contrast. However, working on regions could over-amplify the contrast in near-constant regions, because the histogram is highly concentrated in such regions. Hence, contrast limiting is used. It clips the pixels from bins that exceed the limit value and redistributes that part of the histogram uniformly across other bins.

While original images are encoded in RGB color space, application of CLAHE requires using a color space with a brightness channel. Applying CLAHE to each RGB channel can introduce significant loss in color information. Hence, images were converted to one of the following color spaces:

- HSV
- YCrCb
- CIEL*a*b

Next, CLAHE was applied to the V, Y and L channels, before converting the image back to the RGB color space. This better preserved the color balance of the image, as above color spaces have channels that separate color information from their brightness. The results of applying CLAHE are shown in Figure 5.

Upon examination it is clear that the enhanced images have better contrast, resulting in better separation between skin lesion and the regular skin area.



(a) Equalization of HSV's Value channel.

(b) Equalization of HSV's Value channel after color constancy.

(c) Equalization of CIEL*a*b's L channel.

(d) Equalization of CIEL*a*b's L channel after color constancy.

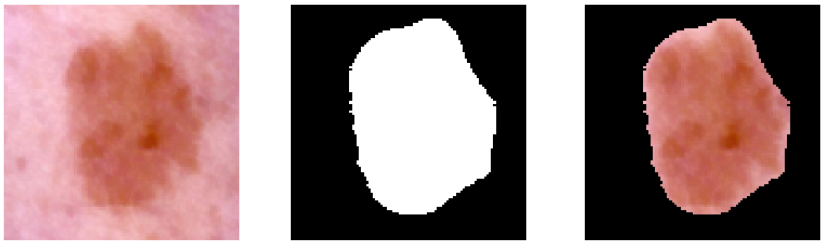
(e) Equalization of YCrCb's Y channel.

(f) Equalization of YCrCb's Y channel after color constancy.

Fig. 5: CLAHE algorithm results for the original image from Figure 4

4.4 Region of interest

In image processing, region of interest (ROI) is defined by a binary mask (same size as in the original image) with pixels having value 1 if they belong to ROI, and 0 otherwise. In skin lesions analysis, ROI masks indicate location (continuous region) of the (primary) skin lesion. For locating ROI, solution presented in [6,5] was used. It is a deep auto-encoder-decoder, extending the U-net neural network with bidirectional convolutional LSTM layers, to encode both semantic and high-resolution information. After establishing the mask, the region of interest was extracted from original images, as depicted in Figure 6.



(a) Original image (b) Binary mask (c) Region of interest

Fig. 6: Region of interest extraction steps.

5 Feature extraction

Preprocessed images were used as an input to feature extractors, such as Convolutional Neural Networks (CNN; [31]) and Histogram of Oriented Gradients (HOG; [14]). This allows comparing quality of features obtained by each of them, and measuring potential improvement resulting from their use.

5.1 CNNs as feature extractors

Modern feature extractors tend to use not only convolutional layers of various sizes but also multiple forms of connection – serial and parallel. Some of these extractors introduce other advanced types of connections, e.g. residual and bidirectional ones. The following network architectures were considered in this work:

- ResNet152V2 [19]
- InceptionV3 [41]
- MobileNet [21]
- DenseNet169 [22]
- InceptionResNetV2 [40]
- VGG16 [44]

A common problem in deep learning is the *vanishing gradient*. It occurs when, during gradient propagation, in the training process, consecutive multiplications result in smaller numbers. As a result, the calculated gradient diminishes almost to zero, stopping weights from improving, or drastically slowing the process. The ResNet152V2 architecture addresses this problem. It introduces a residual ("shortcut") connections between subsequent layers, which skip at least one layer in the network. Here, a network block, instead of learning an underlying function $\mathcal{G}(x)$, learns the residual $\mathcal{F}(x) = \mathcal{G}(x) - x$ where x denotes the input of a block. Such shortcut paths "carry the gradient" throughout the length of a network.

Inception network addresses the problem of extensive networks. Instead of elongating the network, it makes it "wider", and introduces parallel layers. The building block is an inception module. Each block extracts multiple types of features – globally extracted by 5×5 convolution, local features extracted by 3×3, and local features that distinguish themselves from their neighborhood.

MobileNet takes advantage of depthwise separable convolution. In the first step, a 2D convolutional filter is applied to each channel, separately. Next, resulting feature maps are stacked, and 1×1 convolutional filter is applied across channels. Depthwise separable convolution limits the number parameters and multiplications. This makes MobileNet faster and less prone to overfitting.

DenseNet, similarly to ResNet, reduces a vanishing gradient problem by concatenating outputs of each convolutional block with outputs of a preset number of previous convolutional blocks inside a dense block. Overall, DenseNet does not sum output feature maps, but concatenates them.

InceptionResNet architectures combine best features from Inception networks and ResNets. It extends regular inception block with a "shortcut connection" between an input and an output of a block, as in the residual block in ResNets.

VGG16 consists of 16 layers that aim to reduce the number of parameters by replacing larger convolutional kernels with multiple 3×3 ones. For example, a 5×5 kernel could be replaced by two 3×3 kernels, reducing the number of parameters from 25 to 18 (28% drop in size).

These architectures are used primarily in two ways. (1) They are trained from scratch, to extract features present in given dataset. (2) They are used jointly with transfer learning in which weights are ultimately "frozen". The networks with frozen weights are treated as external feature extractors. Typically, ImageNet [33] dataset was used for transfer learning. This approach was proven to be a good choice in [34], and in multiple ISIC 2018 competition submissions.

5.2 Histogram of oriented gradients

Another popular feature extractor is the histogram of oriented gradients (HOG) method [14]. Here, it is assumed that local object structure, and appearance, can be described by the distribution of intensity of oriented gradients. The algorithm consists of following four steps. (1) Calculation of horizontal and vertical gradients. To do so, the image is filtered with kernels presented in equation 3. These kernels extract edges in horizontal and vertical directions. On the basis of this information gradient is computed.

$$\begin{pmatrix} -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} \tag{3}$$

(2) The image is divided into cells with preset size. In each cell, histogram of gradients is calculated. (3) Cells are grouped into larger blocks and gradient strength is normalized inside the block. The block normalization is done by using the L2-hys norm, which is equal to the standard L2-norm, followed by clipping the vector values to the maximum value of 0.2 and renormalizing it. Equation 4 presents the normalization factor f of L2-norm.

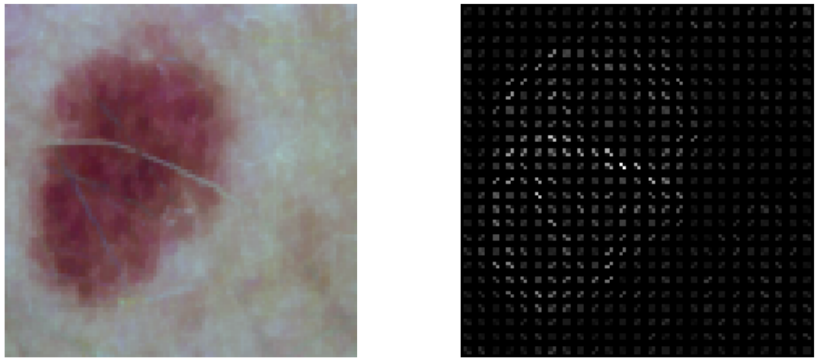
$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \tag{4}$$

where:

- v denotes the vector,
- e stands for a small constant value.

(4) The final feature vector is built by concatenating vectors obtained from each block. In the presented work, cells of size 4×4 are used, with 8×8 blocks, as they proved to achieve best results. The visualization of HOG feature descriptors is shown in Figure 7.

As can be seen, HOG detects a skin lesion on the image that is represented as a mask with marked feature descriptors. One can see the shape of the considered skin lesion with the naked eye.



(a) Original image (b) Hog descriptors

Fig. 7: Visualization of HOG feature descriptors.

6 Classification models

In order to decide which type of skin lesion a given photo contains, basis of the literature review, the following machine learning models were experimented with:

- Support Vector Machine,
- Decision Tree,
- Boosting(AdaBoost/XGBoost),
- Random Forrest,
- Fully Connected Artificial Neural Networks,
- Ensemble of experts.

All models were also considered for ensemble learning, described in what follows.

7 Ensemble learning

Ensemble learning is a process of using multiple machine learning models to improve their individual performance. The ensemble methods usually improve results when there is significant diversity among the member models. The main advantage of the ensemble is its ability to correct errors of some members on the basis of "opinions of other members". Diversity could be introduced in many ways, e.g. training models on different datasets or subsets of the original dataset or using different models and/or different hyperparameters. Ensemble learning is used to combine multiple heterogeneous models, where each performs full multiclass classification. This approach aims at polishing the already "good enough" classification of members, instead of training them from scratch. Here, only lightweight approaches were considered, as the proposed system aims at being used on machines with restricted resources, similar to that used in [28].

7.1 Stacking

Stacking [43] involves an additional machine learning model, trained "on top" of individual model predictions. In our case, the ensemble members are trained separately, to achieve best accuracy. Next, another model is trained, on predictions of ensemble members to deliver the final prediction.

7.2 Voting

Voting is the simplest way to create an ensemble. It acts solely on labels and does not attach importance to the source of prediction. As its result, voting ensemble chooses the label, which gets most votes. The voting process could be executed in two ways as majority voting or weighted majority voting. In majority voting all votes of members of an ensemble have the same weight.

In weighted majority voting, each vote is scaled taking into account perceived "strength" of individual models and "boost" these that perform better. While any metric can be used to generate weights, in what follows, a balanced accuracy metric was used.

8 Experimental results

In this section the results of a comprehensive set of experiments are presented and analysed. We start from the description of the experimental setup.

8.1 Experimental setup

Performance of all models has been evaluated against the original goal metric of the ISIC 2018 Challenge, the *balanced accuracy*, which is defined as an average of recalls obtained on each class along with standard accuracy. Recall performance of classification of each class is represented in the form of confusion matrix. For all these metrics, standard definitions are applied (see, for instance [20]).

Each reported method has been implemented using Python, with Tensorflow, OpenCV, scikit-learn and the Python Imaging Library.

Classifiers were tested on two datasets. The first one internally separated from the competition training dataset. This dataset was split into train (80%) and test (20%) sets, ensuring that both sets contain representatives of all classes. The second dataset was the official dataset of competition. Models were first evaluated on the internal test set and the ones with promising results were then checked against the official one. In all the experiments, the same hair removal algorithm has been applied (see, Section 4.1). Detailed parameters of each model are presented in subsequent sections, related to each experiment.

8.2 HOG-based feature extraction

The first set of experiments concerned the HOG algorithm, used as a feature extractor. Before extracting features, the color constancy algorithm was used. To compensate for the disproportion between the number of images in each class, simple oversampling has been applied. The images from classes with smaller sizes have been duplicated, to roughly match the number in the largest class. Next, random horizontal and vertical rotation were applied. HOG parameters were selected as described in Section 5.2. The results obtained on the internal dataset are presented in Table 3.

Model	Accuracy	Balanced accuracy
ANN	67.09	41.25
Decision Tree	78.44	14.28
AdaBoost	78.44	14.28
XGBoost	81.70	14.28
Random Forest	2.53	14.28
SVM	58.68	34.79

Table 3: Classification results; internal test set; HOG as a feature extractor.

The features generated using HOG contain mostly shape and geometric information. This proved to be insufficient for tree-based methods, as they classified images as belonging to the same category. In contrast, SVM and ANN achieved far better results. Note that, with limited size of the images, these models managed to outperform some of results submitted to the ISIC 2018 Challenge.

8.3 CNNs as feature extractors for tree-based classifiers

In subsequent experiments, convolutional neural networks were used as feature extractors. Overall, for tree-based classifiers CNNs were much better feature extractors than HOG. This is because they are able to extract more complex properties from an image. Their configuration, used in the experiments is summarized in Table 4. Both boosting algorithms used the same number of estimators inside. In each classifier a tree was limited to a depth level equal to 5 and a maximum number of leaf nodes equal to 200.

Model	max depth	max leaf nodes	# estimators	lr	alpha
Decision Tree	5	200	-	-	-
AdaBoost	5	200	50	0.005	-
XGBoost	5	200	50	0.5	0.3
Random Forest	5	200	100	-	-

Table 4: Configuration of tree-based models.

Before feature extraction, images have been processed using color constancy algorithm (best for tree-based methods). Classifiers were evaluated on features extracted by convolutional parts of ResNet152V2, DenseNet169 and MobileNet networks. Each CNN was used as an external extractor with weights obtained from training it on the ImageNet. We decided to use it despite of transfer learning rules which assume that the distribution of both sets (training and destinating) should have the same distribution.

Direct comparison of feature extractors is shown in Figure 8 along with the best result for each classifier presented in Table 5. In Figure 9 confusion matrices of XGBoost and Random Forest were presented as these models achieved the best results (balanced accuracy) equal to 55.86% and 52.92%, respectively.

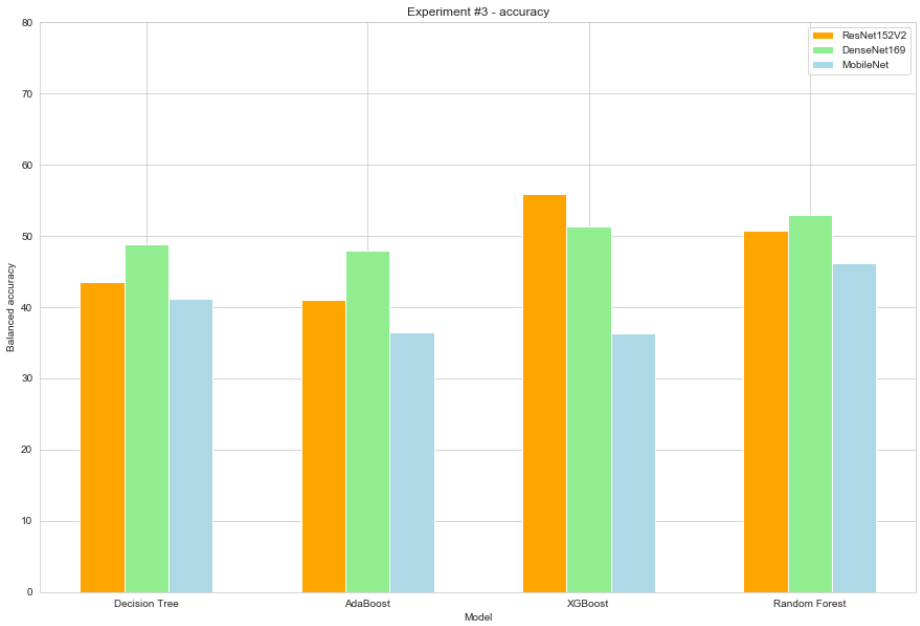
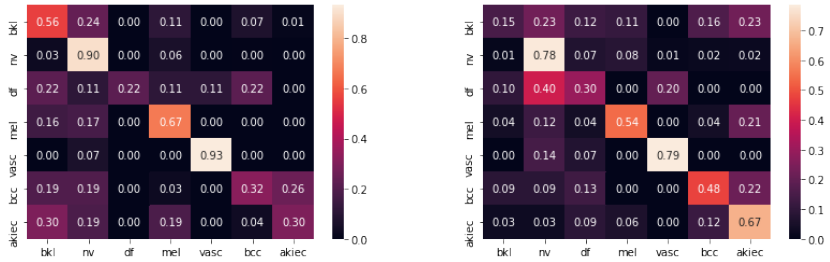


Fig. 8: Comparison of results achieved on the tree based model.

Model	Accuracy	Balanced accuracy
Decision Tree	53.40	48.89
AdaBoost	67.09	47.93
XGBoost	82.86	55.86
Random Forest	71.12	52.92

Table 5: Best result of tree-based classifiers.



(a) XGBoost (b) Random Forest

Fig. 9: Confusion matrices of the best tree-based solutions.

As can be seen from confusion matrices, XGBoost achieved high degree of recognition of images belonging to the vascular lesions and the melanoma classes, above 65% of recall. Random Forest recognized around 50% of skin lesions belonging to the most dangerous classes – melanoma and basal cell carcinoma. However, these two models made mistakes for other classes, e.g. actinic keratosis was classified with approximately 30% accuracy. These two classifiers outperformed the remaining tree-based ones by up to 7 percentage points. The overall performance of this class of models, trained with CNN extracted features, was unquestionably better than the ones using HOG as feature extractor.

8.4 SVM classifier

The SVM classifier has been tested similarly to the tree-based models. The classification model was run using RBF kernel, with $\gamma = \frac{1}{nx_{var}}$, where n denotes number of features in the training set and x_{var} stands for variance of elements in that set. Regularization parameter was set to 1. SVM has been trained on features provided by ResNet152V2, InceptionV3 and VGG16 networks. Metrics obtained on the internal test set are shown in Table 6.

Feature extractor	Accuracy	Balanced accuracy
ResNet152V2	56.48	36.48
IncpetionV3	60.11	50.04
VGG16	74.34	55.09

Table 6: Results of SVM with CNNs.

Convolutional parts of above-mentioned networks were compared in terms of quality of features they produced. VGG16 appears to produce features that are best for the SVM. With these features, SVM achieved 55.09% of balanced accuracy, improving upon result produced on HOG features by more than 37%.

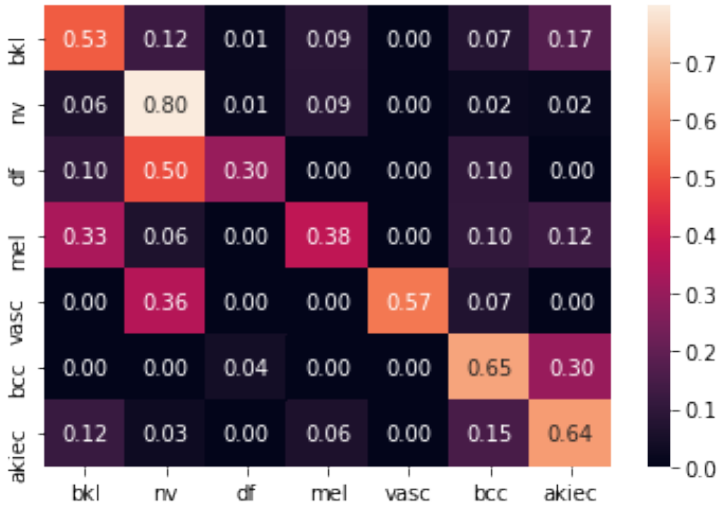


Fig. 10: Confusion matrix of SVM with VGG16.

In confusion matrix, shown in Figure 10, one can see that SVM has a much better result in diagnosing basal cell carcinoma, as it recognized 65% of all instances present in the dataset.

8.5 Comparison of imbalance handling methods

Due to the existing imbalance in the dataset, correct classification becomes a challenge. Hence, sample weighting and oversampling with random rotations were used. These methods have been evaluated on a fully connected ANN, built on the top of the ResNet152V2 feature extractor. Each time, the model has been trained for 100 epochs with a batch size of 96 images with categorical crossentropy as the loss function. The top-level network consisted of 3 hidden layers, each having 4096 neurons, with a sigmoid activation function and a dropout set equal to 0.5. The sample weights for sample s_C belonging to a class C were calculated as follows:

$$w_{s_C} = \frac{n}{N \cdot |C|} \tag{5}$$

where N – number of classes, n – number of samples in the dataset, $|C|$ – number of samples in class C. In Figure 11 confusion matrices corresponding to the results in Table 7 are shown. The results obtained with oversampling show certain improvement in contrast to using sample weights to counter the imbalance of the training data set. Models trained with oversampling tend to recognize either more classes, or have better average class recognition.

Table 7 shows that using oversampling is preferred over the application of sample weights. Results achieved on the augmented dataset allowed to improve

Imbalance handling	ImageNet	Random weights
Oversampling	65.29	42.14
Sample weight	62.43	41.29

Table 7: Balanced accuracy of ANN models with different imbalance handling.

the balanced accuracy of the model with transfer learning by 2.86% and of the one trained from scratch by 0.85%.

As can be seen in Figures 11 and 12 using oversampling increases the lowest nonzero recall by up to 10 percentage points, while increasing the average value of this metric for all classes.

8.6 Region of interest

Another experiment was performed using the ROI extraction described in Section 4.4. Separately, convolutional parts of networks described in Section 5.1 were used as feature extractors. The result of an experiment comparing preprocessed and full images is presented in Table 8. Here, top-level architecture consisted of three hidden layers with 4096 neurons, each with a sigmoid activation function and dropout of 0.5.

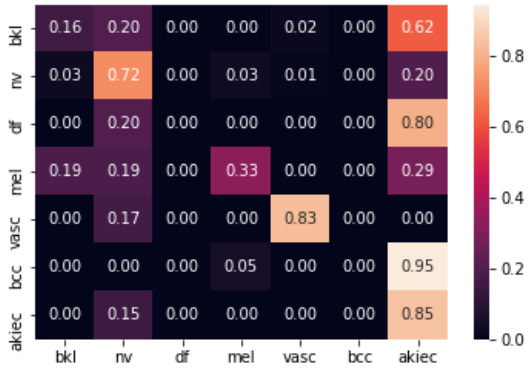
Imbalance handling	ROI	full image
ResNet152V2	26.43	43.86
InceptionV3	36.00	46.57
MobileNet	26.42	40.14
DenseNet169	45.57	43.85
InceptionResNetV2	36.01	41.86
VGG16	14.29	14.29

Table 8: Balanced accuracy of ANN models trained on ROIs and full images.

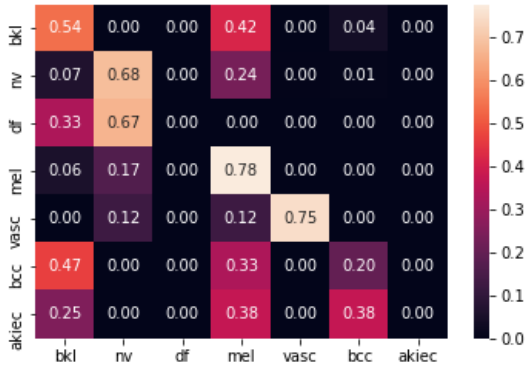
Extracting ROIs tends to perform worse than networks trained with exactly the same parameters and applied to full images. As can be seen (in Table 8) only DenseNet169 improves its predictions when trained on ROIs. This effect is due to the large number of black pixels introduced into the photo during the region of interest extraction. Other architectures do not seem to handle this properly as it likely results in "deactivation" of a big number of neurons.

8.7 Pretrained networks vs learning from scratch

Overall, the dataset consists of insufficient number images for CNN training from scratch. Even after oversampling with augmentation, the size of the dataset is equal to 23,450 records. To overcome this problem, pretrained on the ImageNet

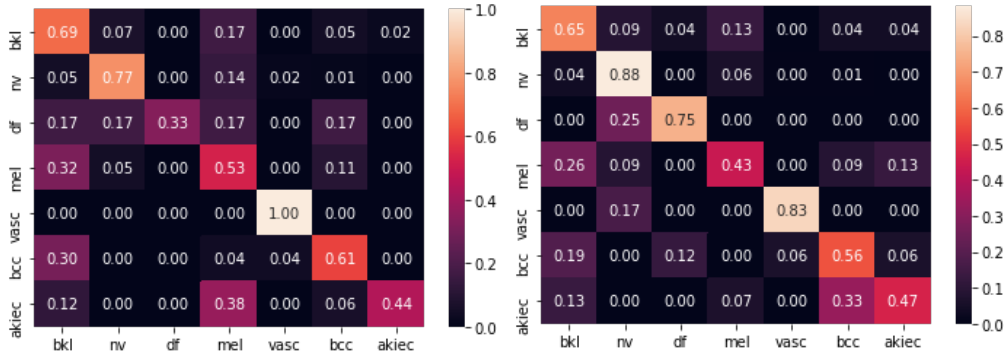


(a) From scratch with sample weight



(b) From scratch with oversampling

Fig. 11: Confusion matrices of networks trained from scratch.



(a) ImageNet with sample weight (b) ImageNet with oversampling

Fig. 12: Confusion matrices of networks with transfer learning

dataset weights were used. The primary motivation was to utilize low-level features that networks learned to recognize. In the learning process of a top-level dense network has been trained to build upon those features. This is also why time needed to complete training on the dataset of interest is shortened.

This approach proved to succeed in the original ISIC 2018 competition, though when applied to bigger images. In the summary of results presented in the Table 9 models that achieved below 30% of balanced accuracy in at least one category are omitted. One can see that ResNet152V2 performed extremely well using the weights trained on the ImageNet. It improved its result by 21.43 percentage points while keeping the training duration the same.

CNN model	ImageNet	Random weights
ResNet152V2	65.29	43.86
MobileNet	36.60	40.14
DenseNet169	51.56	43.85

Table 9: Balanced accuracy ANN models with respect to the CNN part and initial weights.

8.8 Histogram equalization

The next set of experiments was aimed at verification of usefulness of histogram equalization methods proposed in 4.3. These methods were evaluated to check, which approach is better to handle different light conditions for images acquired from multiple sources. Here, impact of histogram equalization on channels of different color spaces was measured and compared to the result obtained with the use of the color constancy algorithm. CLAHE algorithm was applied

to V, Y, L channels of HSV, YCrCb and CIEL*a*b color spaces, respectively, for both pure RGB images and for those preprocessed with the color constancy algorithm. The experiment was done with ResNet152V2 as a feature extractor, with the top-level dense layers with the same architecture as previously. Results of experiments are summarized in Table 10.

Enhancement	Balanced accuracy
color constancy	65.29
HSV color constancy	57.76
YCrCb color constancy	55.12
CIEL*a*b color constancy	57.23
HSV	48.22
YCrCb	46.67
CIEL*a*b	56.42

Table 10: Balanced accuracy of an ANN model trained on images enhanced by CLAHE.

One can see that the color constancy algorithm achieved the best result as it normalized colors present in the image and made them more comparable. Histogram equalization done by CLAHE on regular RGB images works a little bit worse. It noticeably increases brightness level but does not impact the color differences, leaving them "hard to compare". CLAHE applied to the color constancy images, as discussed in Section 4.3, can enhance some darker regions of the skin around the lesion, making it indistinguishable from the lesion and therefore confusing a classifier.

8.9 Ensemble of experts

In this section, problem of multiclass classification has been replaced with a group of simpler problems as described in [29]. As a result, an ensemble of experts is built consisting of models responsible for predicting only one class from the dataset. Table 11 presents the accuracy and the balanced accuracy of each expert along with results from the ensemble.

As one can see, higher results achieved by individual expert in its respective task confirm that simplifying the problem influences the result. Ensemble of experts classifier achieved over 60% of balanced accuracy on the internal test set, which makes it the second and the third best solution considered in this work.

Predictions made by experts were combined together using one of three techniques: expert voting, dense neural network and extreme gradient boosting. Confusion matrices with results from ensembles are shown in Figure 13. It is clearly visible that combining results of individual experts by the process of voting brings the best results. It has the highest average recall and the highest minimal

Expert	Accuracy	Balanced accuracy
bkl	84.42	84.69
nv	91.67	85.35
df	98.91	74.82
mel	77.35	83.44
vasc	96.01	65.02
bcc	96.01	72.77
akiec	96.20	85.91
ANN	78.26	63.20
XGBoost	84.60	57.72
Voting	82.97	61.50

Table 11: Results of individual experts and the full ensemble.

recall across all models proposed here. The results may be surprising because this approach is not very popular in the literature as stated in Chapter 2.

8.10 Summary of results for the ISIC 2018 Challenge

In Table 12 a summary of results achieved by each considered model is shown. Classifiers with balanced accuracy above 55% (on the internal dataset) were evaluated on the original ISIC 2018 Challenge test set. Italicized text denotes models trained on RGB images without color constancy.

Extractor	Model	Internal test set	External test set
ResNet152V2	XGBoost	55.86	41.40
ResNet152V2	ANN	65.29	49.60
ResNet152V2 CIELAB	ANN	56.52	48.0
ResNet152V2 CIELAB	ANN	57.23	42.1
ResNet152V2 HSV	ANN	57.76	46.2
ResNet152V2 YCrCb	ANN	55.12	39.0
VGG16	SVM	55.09	43.1
ResNet152V2 Expert	Voting	61.50	48.9
ResNet152V2 Expert	ANN	63.20	46.6
ResNet152V2 Expert	XGBoost	57.72	40.1

Table 12: Results summary of top 10 results.

8.11 Results obtained for ensemble learning

In this section, the comparison of results obtained with ensemble learning applied is presented. As ensemble members, models (reported above) with results above 42%, have been selected. Both versions of voting along with extreme

(a) Majority voting

(b) XGBoost

(c) ANN

Fig. 13: Confusion matrices of an ensemble of experts.

Type	External test set
Majority voting	52.8
Weighted majority voting	52.7
ANN stacking	45.3
XGBoost stacking	40.0

Table 13: Results of an ensemble learning with 7 best performing models as its members.

gradient boosting and artificial neural networks, have been tested. The results of these experiments are presented in Table 13.

As one can see the final ensemble achieved the highest balanced accuracy, by applying voting to combine predictions from member models. The use of an ensemble of models increased the final classification by 3 percentage points.

The results are promising in the context of computational resources available, as well as size of images. The proposed solution would take 60th place in the ISIC 2018 Challenge, ahead of 18 other solutions, including 3 using data from external datasets, thus increasing the number of photos at the training stage.

9 Concluding remarks

The aim of the reported work was to study application of machine learning to the skin cancer detection problem. In this project the target dataset was highly unbalanced, which introduced an additional level of difficulty to the problem.

The proposed approach differs significantly from those proposed in the literature. The difference materializes, primarily, in terms of size of the images, and the diversity of models used as members in the ensemble learning. The main idea of the proposed approach was to develop a diagnosis system that can run efficiently on mobile devices and can be trained without corporate-level computational resources (e.g. large Google/Amazon/Microsoft cloud). The modular architecture of the proposed system allows it to be easily expanded, as every stage could be replaced with another approach to its implementation, with possible further gains in performance.

Presented results show that using smaller images still allows achieving similar outcomes on comparable technical setup and computational power but reduces the complexity of the training process. The result of that is a faster learning stage and quicker predictions that could be used on mobile devices, achieving almost a real-time prediction. A single training epoch of best performing models took approximately between 57 to 85 seconds on NVIDIA GeForce GTX 1050 Ti 4GB.

Although the results presented in chapter 8 may seem worse than the best reported for the ISIC 2018 Challenge, in terms of resources used and learning time, they turn out to be more usable in everyday life. As a result, they offer hope for faster universal adaptation and greater availability of high-quality automated early diagnosis in the future.

As for the prospects for future work, the introduction of smaller, more condensed feature extractor, training on a larger dataset, with possible use of unlabeled data could be considered. Due to the proposed size of images, the execution time of an individual prediction should not rise with training performed on a bigger dataset, so the aspect of use on mobile devices could be maintained. Alongside proposed, slight enlargement of an image size aimed at restoring original image proportions could be considered.

References

1. Abbas, Q., Celebi, M., Serrano, C., Fondón García, I., Ma, G.: Pattern classification of dermoscopy images: A perceptually uniform model. Pattern Recognition **46**(1), 86–97 (2013)
2. Ahmad, B., Usama, M., Huang, C.M., Hwang, K., Hossain, M.S., Muhammad, G.: Discriminative feature learning for skin disease classification using deep convolutional neural network. IEEE Access **8**, 39025–39033 (2020)
3. ALEnezi], N.S.A.: A method of skin disease detection using image processing and machine learning. Procedia Computer Science **163**, 85 – 92 (2019). https://doi.org/https://doi.org/10.1016/j.procs.2019.12.090, http://www.sciencedirect.com/science/article/pii/S1877050919321295, 16th Learning and Technology Conference 2019Artificial Intelligence and Machine Learning: Embedding the Intelligence
4. Alquran, H., Qasmieh, I.A., Alqudah, A.M., Alhammouri, S., Alawneh, E., Abughazaleh, A., Hasayen, F.: The melanoma skin cancer detection and classification using support vector machine. In: 2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT). pp. 1–5 (2017)
5. Asadi-Aghbolaghi, M., Azad, R., Fathy, M., Escalera, S.: Multi-level context gating of embedded collective knowledge for medical image segmentation (2020)
6. Azad, R., Asadi-Aghbolaghi, M., Fathy, M., Escalera, S.: Bi-directional convlstm u-net with densley connected convolutions (2019)
7. Barata, C., Celebi, M.E., Marques, J.S.: Improving dermoscopy image classification using color constancy. IEEE Journal of Biomedical and Health Informatics **19**(3), 1146–1152 (2015)
8. Barata, C., Marques, J.S.: Deep learning for skin cancer diagnosis with hierarchical architectures. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019). pp. 841–845 (2019)
9. Barata, C., Celebi, M.E., Marques, J.S.: Improving dermoscopy image classification using color constancy. IEEE Journal of Biomedical and Health Informatics **19**(3), 1146–1152 (2015). https://doi.org/10.1109/JBHI.2014.2336473
10. Brinker, T., Hekler, A., Utikal, J., Grabe, N., Schadendorf, D., Klode, J., Berking, C., Steeb, T., Enk, A., Von Kalle, C.: Skin cancer classification using convolutional neural networks: Systematic review. Journal Of Medical Internet Research **20**(10) (2018)
11. Capdehourat, G., Corez, A., Bazzano, A., Alonso, R., Musé, P.: Toward a combined tool to assist dermatologists in melanoma detection from dermoscopic images of pigmented skin lesions. Pattern Recognition Letters **32**(16), 2187–2196 (2011)
12. Codella, N.C.F., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S.W., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M.A., Kittler, H., Halpern, A.:

Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (ISIC). CoRR **abs/1902.03368** (2019), http://arxiv.org/abs/1902.03368
13. Codella, N.C.F., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S.W., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M.A., Kittler, H., Halpern, A.: Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (ISIC). CoRR **abs/1902.03368** (2019), http://arxiv.org/abs/1902.03368
14. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 1, pp. 886–893 vol. 1 (2005). https://doi.org/10.1109/CVPR.2005.177
15. Dorj, U.O., Lee, K.K., Choi, J.Y., Lee, M.: The skin cancer classification using deep convolutional neural network.(report). Multimedia Tools and Applications **77**(8), 9909 (2018)
16. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**(7639), 115 (2017)
17. Farooq, M.A., Khatoon, A., Varkarakis, V., Corcoran, P.: Advanced deep learning methodologies for skin cancer classification in prodromal stages (2020)
18. Finlayson, G., Trezzi, E.: Shades of gray and colour constancy. pp. 37–41 (01 2004)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015)
20. Hossin, M., M.N, S.: A review on evaluation metrics for data classification evaluations. International Journal of Data Mining Knowledge Management Process **5**, 01–11 (03 2015). https://doi.org/10.5121/ijdkp.2015.5201
21. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications (2017)
22. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks (2018)
23. Kannojia, S.P., Jaiswal, G.: Ensemble of hybrid cnn-elm model for image classification. In: 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN). pp. 538–541 (2018)
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems. vol. 25. Curran Associates, Inc. (2012), https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
25. Kumar, V.B., Kumar, S.S., Saboo, V.: Dermatological disease detection using image processing and machine learning. In: 2016 Third International Conference on Artificial Intelligence and Pattern Recognition (AIPR). pp. 1–6 (2016)
26. Lau, H.T., Al-Jumaily, A.: Automatically early detection of skin cancer: Study based on nueral netwok classification. In: 2009 International Conference of Soft Computing and Pattern Recognition. pp. 375–380 (2009)
27. Lee, T., Ng, V., Gallagher, R., Coldman, A., McLean, D.: Dullrazor®: A software approach to hair removal from images. Computers in Biology and Medicine **27**(6), 533–543 (1997). https://doi.org/https://doi.org/10.1016/S0010-4825(97)00020-6, https://www.sciencedirect.com/science/article/pii/S0010482597000206

28. Lin, T.C., Lee, H.C.: Skin cancer dermoscopy images classification with meta data via deep learning ensemble. In: 2020 International Computer Symposium (ICS). pp. 237–241 (2020). https://doi.org/10.1109/ICS51289.2020.00055
29. Liu, W., Zhang, M., Luo, Z., Cai, Y.: An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors. IEEE Access **5**, 24417–24425 (2017)
30. Masood, A., Al-Jumaily, A.: Semi advised learning and classification algorithm for partially labeled skin cancer data analysis. In: 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE). pp. 1–4 (2017)
31. O'Shea, K., Nash, R.: An introduction to convolutional neural networks. CoRR **abs/1511.08458** (2015), http://arxiv.org/abs/1511.08458
32. Rezvantalab, A., Safigholi, H., Karimijeshni, S.: Dermatologist level dermoscopy skin cancer classification using different deep learning convolutional neural networks algorithms (2018)
33. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
34. Sah, A.K., Bhusal, S., Amatya, S., Mainali, M., Shakya, S.: Dermatological diseases classification using image processing and deep neural network. In: 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). pp. 381–386 (2019)
35. Sarkar, R., Chatterjee, C.C., Hazra, A.: A novel approach for automatic diagnosis of skin carcinoma from dermoscopic images using parallel deep residual networks. In: Singh, M., Gupta, P., Tyagi, V., Flusser, J., Ören, T., Kashyap, R. (eds.) Advances in Computing and Data Sciences. pp. 83–94. Springer Singapore, Singapore (2019)
36. Sedigh, P., Sadeghian, R., Masouleh, M.T.: Generating synthetic medical images by using gan to improve cnn performance in skin cancer classification. In: 2019 7th International Conference on Robotics and Mechatronics (ICRoM). pp. 497–502 (2019)
37. Statistics Poland, Social Surveys Department, S.O.i.K.: Health and health care in 2019 (2021)
38. Suganya, R.: An automated computer aided diagnosis of skin lesions detection and classification for dermoscopy images. In: 2016 International Conference on Recent Trends in Information Technology (ICRTIT). pp. 1–5 (2016)
39. Sun, H., Yang, J.: Domain-specific image classification using ensemble learning utilizing open-domain knowledge. In: 2019 International Conference on Computing, Networking and Communications (ICNC). pp. 593–596 (2019)
40. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning (2016)
41. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions (2014)
42. Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset: A large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific Data **5** (08 2018). https://doi.org/10.1038/sdata.2018.161
43. Wolpert, D.: Stacked generalization. Neural Networks **5**, 241–259 (12 1992). https://doi.org/10.1016/S0893-6080(05)80023-1
44. Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 (09 2014)