Skin Lesion Segmentation and Classification

Lluís Borràs Ferrís
Erasmus Mundus Joint Master in
Medical Imaging and Applications(MAIA)
Cassino, Italy
lluis.borrasferris@studentmail.unicas.it

Juan Andrés Cisneros Jácome
Erasmus Mundus Joint Master in
Medical Imaging and Applications(MAIA)
Cassino, Italy
juan.cisneros@studentmail.unicas.it

Nohemí Sofía León Contreras
Erasmus Mundus Joint Master in
Medical Imaging and Applications(MAIA)
Cassino, Italy
nohemisofia.leoncontreras@studentmail.unicas.it

***Abstract*—In the present work two different skin lesion segmentation approaches are presented on the ISIC 2017 challenge data (with training, validation and test sets of 2000/150/600), one using classic Image Processing and another one using Deep Learning. Additionally, classification of the skin lesions diagnosis (nevus, melanoma and seborrheic keratosis) is done with Machine Learning and Deep Learning.**

In the first segmentation method we propose the utilization of the K-means algorithm combined with constructed function to remove the undesirable hairs found in some images. We test our model on a 200-image subset of the ISIC 2017 challenge reaching a performance of 81.34% in the Jaccard Index. In the second approach for the segmentation we build a modified U-Net convolutional neuronal network (CNN) that reached a final Jaccard Index of 74.91% in the ISIC 2017 test set.

Using classic image processing and analysis, several geometrical (shape), texture and color features were extracted from the U-Net segmentations to train different Machine learning models to do the classification between nevus, melanoma and seborrheic keratosis. The best performance observed was 64.90% with SVM. Furthermore, the fine tuning of torchvision models (all of which have been pretrained on the 1000-class Imagenet data set) is done to classify the skin lesions, reaching a balanced multiclass accuracy of up to 68.09% with ConvNeXt.

***Index Terms*—Image analysis, Deep Learning, Machine Learning, U-Net, ConvNeXt, skin lesion, segmentation, classification, ISIC 2017.**

INTRODUCTION

Skin cancer incidence has increased rapidly in the last few years. The current lifetime risk of developing melanoma is 1 in 63 in the United States, and similar ratios are noted in other Western nations [1]. Even though melanoma is not the most common skin cancer, it is the deadliest form of skin cancer, accounting for 75% of skin cancer deaths [2]. According to the Skin Cancer Foundation, when melanoma is detected in early stages its 5-year survival rate is 99% [3]. For that reason, detecting melanoma in its early stages is of utmost importance to ensure the patient's best prognosis. Non-invasive imaging technology has proven to be a cost-effective tool for early detection of melanoma; this technology can be useful to help clinicians decide whether a biopsy is needed. However, much of the variability in diagnosis and prognosis for melanoma is due to the high inter- and intra- observer variability in visual observations used for melanoma diagnosis and prognosis, therefore, development of more objective methods to improve the accuracy and reproducibility in the field of melanoma diagnosis is needed.

In this work two segmentation algorithms and two classification algorithms for skin lesions are proposed by using the dermoscopic images from the ISIC 2017 challenge [4]. The image data set is composed of 2000 images used for training, 150 images used as a validation set and 600 test images. The images involve three unique diagnoses of skin lesions: melanoma, nevus, and seborrheic keratosis. Nevi are benign skin tumors derived from melanocytes, while seborrheic keratoses are benign skin tumors derived from keratinocytes commonly removed for cosmetic reasons.

The work will be presented in three parts: Image Processing and Analysis(IPA), Machine Learning (ML) and Deep Learning (DL). In the IPA part, preprocessing and segmentation over a subset of 200 images is done. Feature extraction over the full ISIC data set of the lesions is presented also in this part. In the ML part, the extracted features of the IPA part were fed into ML models to conduct classification of the lesions. In the DL part, a U-Net convolutional neural network (CNN) is implemented for segmentation of the skin lesions and different pretrained architectures were used for lesion classification using transfer learning over the full ISIC data set.

Performance of the segmentation task was evaluated with the jaccard index and performance of the classification task was evaluated with the balanced multiclass accuracy (BMA).

I. IMAGE PROCESSING

To process the images OpenCV library [5] was used in Python. A function was created to load the images to be processed. It returns the images in a list of arrays.

A. Methodology

Hair Removal

Some of the skin lesion images are cluttered with hairs, in order to correctly segment the lesions these hairs should be removed as part of the preprocessing stage. A function named *hair_removal* was made to detect and remove the hairs from the images.

To remove the hairs it was decided to use the image inpainting. In order to apply OpenCV's *inpaint()* function, we need to create a binary mask of the hairs found in the image where non-zero pixels correspond to areas that should be inpainted.

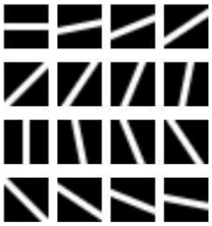

Fig. 1. Structuring elements used in the bottom hat operation

Firstly a structuring element, resembling_the appearance of a hair, was created. This kernel was rotated 16 times over 180° degrees, shown in Fig.1.

These kernels were used to apply a black hat transformation on the red channel of each image, in order to capture hairs in different directions. The results of each bottom hat transformation were then in a 16-bit unsigned integer array of the same height and width as the image. This array is scaled to 0 and 255 with minimum-maximum normalization and converted to an 8-bit unsigned integer 2-D array (Fig.2a). A grayscale opening operation is done to the array to get rid of small bright details. The image was then binarized with simple thresholding, in this way the hairs detected with the bottom hat transformation are set to white (Fig.2b). The binarized image was dilated to make the detected hairs more robust, connect them and fill in any gaps.

The hairs in the binarized image should be long connected lines in different directions, so a binary opening operation followed by a closing operation (to keep the connected lines and then fill them making them solid) is performed with a 16-time rotated structuring element, analogous to the one used in the bottom hat transformations (Fig 3). The results of the morphological operations (opening+closing) of each rotated kernel are accumulated in a 2-D array with the help of a for loop. The resulting mask was then dilated with an elliptical kernel (Fig.2c). The inpaint selected algorithm was INPAINT TELEA. All images with removed hair were stored as ID_nohairimage.jpg.

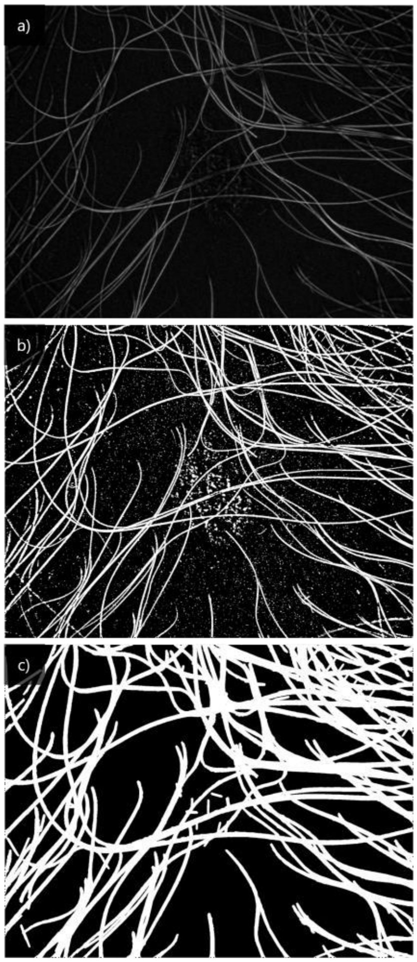

Fig. 2. a) Hair detection after normalizing the sum of bottom hats. b) Hair mask after doing binarization. c) Final mask used to inpaint the hairs in the original image

Watershed Segmentation

Any grayscale image can be viewed as a topographic surface where high intensity denotes peaks and hills while low intensity denotes valleys. Our first approach to segment the images was to apply the Watershed algorithm. The watershed algorithm starts by filling every isolated valley (local minima) with different colored water (labels). As the water rises, depending on the peaks (gradients) nearby, water from different valleys, hence with different labels, will start to merge. To avoid that, we set barriers in the locations where water merges. The water continues to be filled (and barriers build) until the peaks are under water. The created barriers give the segmentation result.

To implement watershed, we started by converting the no hair image from BGR to HSV and doing an inverted binary threshold. We then apply aggressive openings to try to eliminate spots that were not part of the lesion and then dilate the result to increase the lesion boundary to background. We apply a distance transform to obtain the regions of the image that are 'for sure' part of the lesion. By subtracting the image from the distance transform to the dilated image we

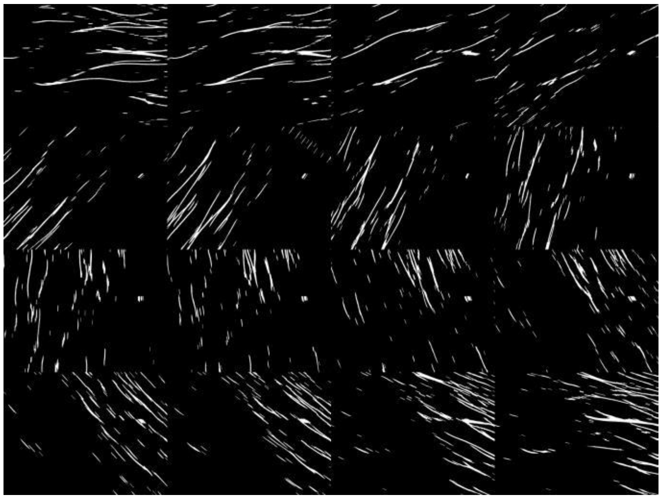

Fig. 3. The results of doing opening followed by closing in the binary image with the line_searcher_se kernel rotated in 16 different directions.

obtain a region that ’for sure’ is not foreground (in our case lesion). The regions we know for sure (whether foreground or not foreground) are labelled with any positive integers, but different integers, and the area we don’t know for sure are just left as zero. For this we use cv.connectedComponents(). We then make sure that the background is labelled as 1 and that the unknown regions of the image are labelled as 0. This image with markers is the one we use for the *watershed()* function alongside the original input image.To obtain the final lesion mask we look for the bigger area in the watershed result using OpenCV’s *findContours()* and *contourArea()* functions. Once the biggest area is found, we fill the area to ensure that there are no internal holes.

K-Means Segmentation

K-means is a methodology that divides the image into k clusters. Each observation (pixel intensity) would be assigned to one of the k clusters with the nearest mean. We propose a segmentation method with k = 2 clusters, background and skin lesion. The means are iteratively refined until convergence following two criteria: reach a maximum number of iterations or if every cluster center moved lessthan a defined epsilon.

An extra preprocessing step is performed into the image with no hairs before passing it to the K-means algorithm. Fist the RGB image is converted into a L*a*b color space, where L stand for lightness, the a component is the Red/Green value and the b component is the Blue/Yellow value. Then an adaptative histogram equalization is performed to improve the contrast of the images only in the L channel. The image is divided into small blocks, then each of these blocks are equalized. We set a contrast limit in order to avoid amplifying the noise of the image. Afterward, to reduce the possible small changes of intensity (noise) contained in the image, an average filter of kernel size 5x5 is applied to blur the image. Finally, to be able to perform image segmentation with the K-means algorithm we transform the image matrix to a vector of single pixel observations.

K-means algorithm from OpenCV library is applied with the criteria and the number of clusters mention before. Additionally, the algorithm performs 10 different runs with different kmeans++ center initialization (assigning the first two centroids randomly, and then the subsequent centroids are chosen from the remaining pixels based on a probability proportional to the squared). Once the 10 different attempts are performed, it is selected the classification with the best compactness.

The result of this algorithm, shown in Fig.4a, is the first step to obtain the final mask of the skin lesion. Because of the irregularities present on the images this algorithm does not return a perfect segmentation of the skin lesion and the background. In the result, there may be other areas in the image with similar intensities to the skin lesion. Four extra steps are performed in order to refine the result and obtain one final mask:

- First, we look for the presence of a field of view (FOV) in the K-means clustering result, since the K-means algorithm usually interprets the FOV as the same cluster as the skin lesions, due to the low intensities present in their pixels with respect to the rest of the skin (since all images seem to come from Caucasian patients). We check if the FOV is present in the image by looking for pixels with low intensities in the corners of the image. If true, we apply a classic binary threshold to generate a mask that eliminates de FOV (Fig.4b).
- Second, in order to binarize the image obtained in the k-means an OTSU threshold technique is applied. If FOV was detected in the image, we multiply the binary image (obtained in the step above) by the k-means binarized image to obtain a final binary image without FOV (Fig.4c).
- Third, we applied a closing operation in order to eliminate the holes present in the skin lesion possibly formed after the FOV removal operation. The remaining will be considered as skin lesion and some small areas due to some irregularities in the image.
- A last step is performed to obtain the final lesion mask by looking for the bigger area in the non-FOV binary image. Once the biggest area is found, we fill the area by using *drawContours()*.

After these steps, we have our final segmentation of the skin lesion. Studying the segmentation ground truth of the subset of the 200 masks we understood that these manual segmentations are not very sharpened masks. Therefore, to refine the results, we do 10 times an extra morphological dilation of this final mask with a morphological cross kernel of size 3x3, this result is shown in Fig.4d.

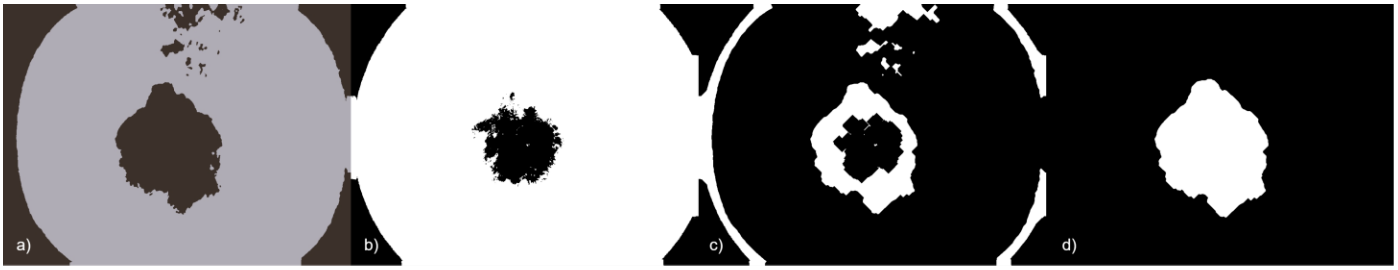

Fig. 4. a) Result obtained after applying K-means function. b) FOV mask obtained by binary thresholding. c) Result after multiplying the OTSU binarization with the FOV binary mask. d) Final segmentation obtained from the algorithm.

Feature Extraction

Methods and approaches used for the determination of the skin lesion features for ML classification will be discussed in this subsection. The first step in our feature extraction system, is the selection of the data set to work with. We choose to combine both train and validation sets (2150 images, 200 of which were previously analyzed in the work described above) in order to have more training data and to do cross-validation, which does not require a validation set. The Fig.5 shows the flowchart of our work.

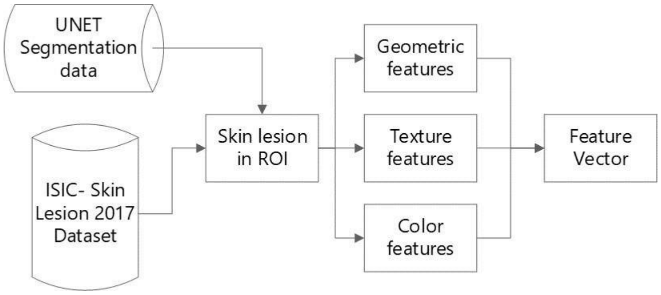

Fig. 5. Feature Extraction pipeline.

We take the segmented images from our Deep Learning segmentation task. Those images were trained, validated and tested with a modified U-Net model(methodology for obtaining these images is explained in detail in Deep Learning Section). The Jaccard Index obtained for the segmentation of the train, validation and test data sets were 85.43%, 76.41% and 74.91% respectively. We analyzed several criteria to determine which are the most important features that should be taken into account to obtain a good discrimination at the time of classification. In [6], they implement the ABCD (asymmetry, border irregularity, color and dermoscopic structure) rule which is a scoring method to quantify dermoscopy findings and effectively separate melanoma from benign lesions used by dermatologists. The main objective of this method is to identify each of the different characteristics present in a skin lesion, add them according to their importance with a respective weight and compare their final value (Total Dermoscopic Score, TDS) with an established one to know if it is benign or malignant.

$$TDS = A * 1.3 + B * 0.1 + C * 0.5 + D * 0.5$$

In our approach we will use some of these criteria to find the characteristics that best describe the behavior of a malignant and benign skin lesion. For our purposes we determined that the features that carry the most weight are color, texture and shape. As in [6], since these features correspond to those with the highest weight in the TDS formula.

1) *Geometric features:* Geometric moments refer to the most important features with respect to the shape of the skin lesion. Many different properties have been developed based on ways to support this process. It has been taught that it is important to divide these properties into two groups: in the first group there are properties that are invariant to translation, rotation and scaling, while in the other group there are properties that do not have these properties. The properties of the first set are usually simpler to subtract and require simpler methods [7]. The methods for subtracting these properties also have the ability to be split into two groups. The first one represents boundary-based methods that only focus on the outer edge of how they are retrieved. The other one represents region-based methods that operate on the entire surface as an object. The typical method of the first group is the use of Fourier descriptors, while the zonal set uses invariant moments more often [7].

- HU moments: Moment invariants are firstly introduced by Hu [8]. He derived six absolute orthogonal invariants and one skew orthogonal invariant based upon algebraic invariants, which are not only independent of position, size and orientation but also independent of parallel projection. As hu moments are rotation invariant the problem arises between the shapes that have similar shapes but in different angles. The moment invariants have been proved to be the adequate measures for tracing image patterns regarding the image's translation, scaling and rotation under the assumption of images with continuous functions and noise-free [9].
- Irregularity: It has been reported that border irregularity is the most relevant component in the diagnosis of melanoma. In contrast to benign pigmented wounds, which tend to present with regular borders, melanoma has an irregular border due to the uneven growth rate,

multidirectional spread of melanocytes, and regression of invasion and/or genetic instability of melanocytes [10].

- Eccentricity:The eccentricity feature is the value of comparison between the distance of the foci (two fixed, interior points) ellipses minor with the foci ellipses major in the object. [11].

2) *Texture features:* Texture features refer to visual patterns that have characteristics of uniformity that do not result from the existence of a single color or intensity. These properties contain fundamental information about the structural arrangement of ROI and their interaction with the surrounding surfaces.

- Gray Level Co-Occurrence Matrix:In [9], to subtract texture properties, the Gray Level Co-occurrence Matrix (GLCM) is first computed and then 4 texture properties are computed from it. GLCM is a matrix showing the distribution of simultaneous pixel values in a given motion. The functionality of GLCM is applied to characterize the texture of an image by calculating the appearance of pairs of pixels, and their spatial interaction is created in the images.
- Local Binary Patterns: LBP method is statistical texture descriptor that gives local structure. LBP provides a unified description including both statistical and structural characteristics of a texture patch, for that it is powerful for texture analysis. LBP is a simple and efficient texture operator that labels pixels in an image by thresholding the neighborhood of each pixel and treating it as a binary number [12]. Formally, given a pixel (center point) in the image, the LBP pattern is computed by comparing its value with those of its neighboring pixels.

3) *Color features:* Color histograms are one of the simplest methods for subtracting image properties. This method is based on looking at how much of a particular color is present in the current histogram. Therefore, to use this method, it is necessary to segment the image before processing it. In our case, we use the segmented images from Deep Learning and those will determine our ROI. Then for each image, in each color channel (RGB), we calculate the mean, standard deviation and skewness values. The mean reflects the overall intensity of the image. Standard deviation (or variance) gives the scatter of the data. A high contrast image has a high variance and a low contrast image has a low variance [13]. The skewness computes the asymmetry of the probability distribution of the histogram. Therefore, it reveals information about the shape of the distribution [13].

Once all the features have been extracted, we proceed with the creation of the feature vector, which will serve as a unique characteristic vector identifier of each image. This process will be carried out for the whole data set to be worked on, both training set (2150 images) and test set (600 images). The length of our feature vector will depend on the parameters we choose when extracting the texture features. The vector may vary from a dimension of 90 features with all uniform rotation invariant LBP descriptors, up to 132 features with both rotation invariant and uniform rotational invariant descriptors. The feature vector matrix of all images was saved in csv file for later use when working with the ML algorithms.

B. Results

We ran the above explained hair removal function with all the images of the ISIC 2017 subset (200 images). Overall, images cluttered with hairs had the hair removed while maintaining the details of the skin lesions. In most of the images without hair, the details of the lesions were also preserved. However, we still had some images where details of the lesion were lost by the inpaint process (see Fig 6). Since the inpaint algorithm was computationally slow, the images with the removed hairs were saved using the imwrite function of OpenCV in order to perform different trials with the two proposed segmentation methods.

We performed the segmentation of the subset composed by 200 no-hair images with the two different methodologies presented, K-means and Watershed. In the earlier stages of our implementation, we realized that K-means was performing better in average than Watershed by comparing the Jaccard Index and Dice coefficient. That's why we choose to use K-means as the final segmentation method. From this point on, we started refining the K-means to improve our results. First, we realized that the FOV was affecting badly the segmentation of the images where it was present. We added a FOV detection and removal feature that was explained above. Before running the K-means function, we applied an adaptative contrast and blurring to increase the level of noise. These modifications inserted in our K-means algorithm increased our overall performance. Either way, we present the average results obtained in both methodologies in the Table I. The table shows that K-means outperfoms the Watershed segmentations with an average Jaccard Index of 81.35% in comparasion with the average Jaccard obtained in the Wathershed of 67.41%. Clearly the best segmentation algorithm was the K-means, hence our final choice to be the segmentation model to be used in the image processing segmentation approach. We present the performance of our algorithm with two different images of the subset. The first one is the image ISIC_0000072, the Fig. 7 shows the performance with an image with no hairs

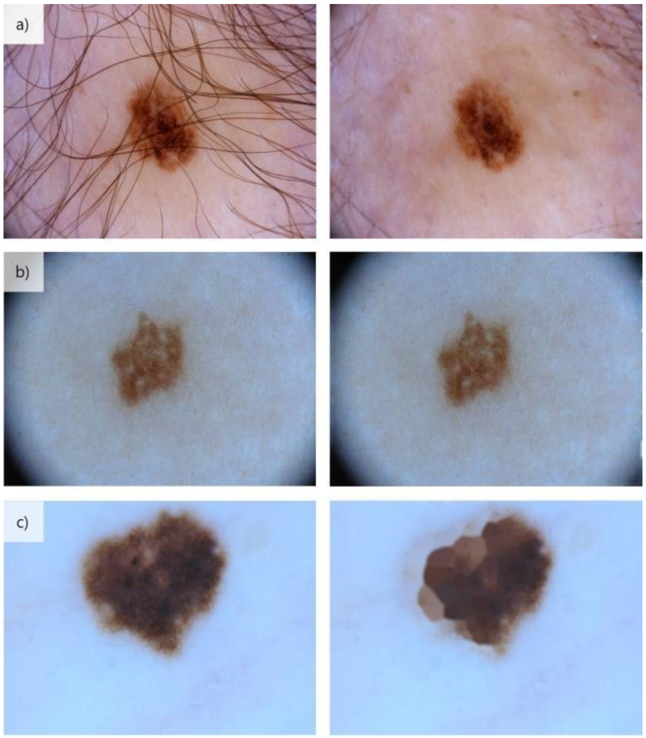

Fig. 6. On the left original images are shown, on the right the resulting image after hair_removal function being applied is shown. a) A satisfactory result where an image cluttered with hair has them removed while preserving the lesion. b) A satisfactory result where an image with no hair is unaltered after application of the hair_removal function. c) A result where the lesion details are damaged after the application of the hair _removal function.

TABLE I
AVERAGE SCORES FOR THE THE K-MEANS AND WATERSHED SEGMENTATIONS IN THE 200 IMAGES SUBSET

	Jaccard Index	*Dice Coefficient*
K-means	81.35	88.13
Wathershed	67.41	76.43

and FOV. It is possible to see the results obtained after each stage of our algorithm and comparing with the original image and its ground truth, the result after the hair removal, and a comparison between the segmentations. obtained with watershed and K-means. It is possible to see that in this case the image hair removal did not damage the lesion that didn't have hairs even having dark areas in the lesion. On the other hand, the second image ISIC_0000095 contain several dark hairs, some of them covering the lesion. It is possible to see in the Fig. 8 how the hair removal function eliminates effectively the hairs without damaging the lesion. Even though, it can be observed that both algorithms have good performances, K-means outperforms Watershed, with a Jaccard Index of 92.69% in contrast with a 76.41% obtained in the Watershed in the image ISIC_0000072 , and in the second image a Jaccard of 84.41% in comaparison with a score of 80.34% in the Watershed.

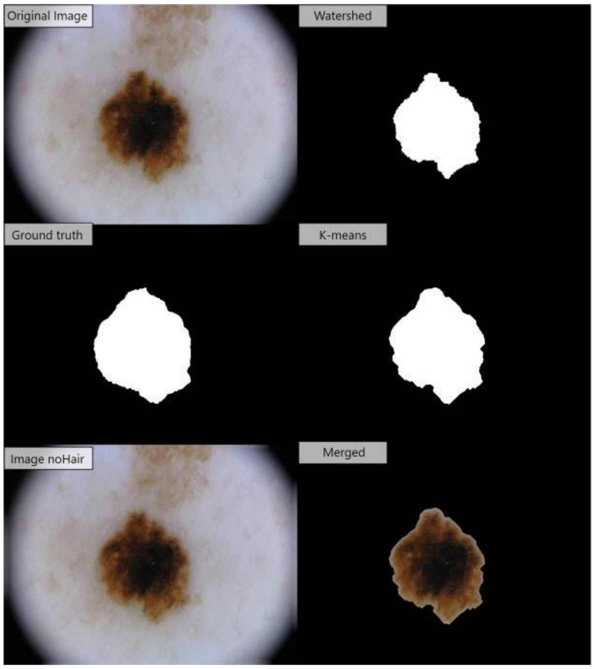

Fig. 7. Results obtained through the different parts of our algorithm in the image ISIC−0000072

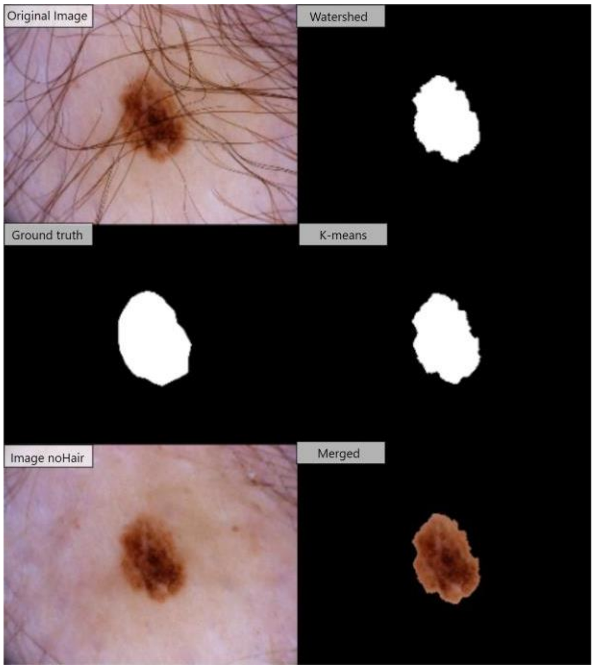

Fig. 8. Results obtained through the different parts of our algorithm in the image ISIC_0000095

II. Machine Learning

This section will discuss how a skin lesion classification system was implemented with machine learning algorithms. The diagnostic discrimination was performed for three classes of skin lesions: nevus, melanoma and seborrheic keratosis. We divide our work in 2 main steps:

1) Feature vector: Feature information from different data sets.
2) Classification: select, tune and train model classifiers in order to obtain the best performance classification.

The performance of this work is evaluated by its ability to correctly classify the different types of skin lesions. The performance measures are: accuracy, specificity, sensitivity, precision, balanced multiclass accuracy (BMA) and Receiver Operating Characteristic (ROC).

A. Methodology

We use the ISIC 2017 data set to get the feature vectors used for the classification task. We perform 2 different feature.csv files with images from this data set and the masks obtained from our U-Net segmentation for the three sets (training, validation and test), one .csv file training + validation composing the total data for training as cross validation is performed, and the other .csv with the features for the finaltest of our models.

Feature vector

A main part of working with machine learning algorithms for classification is the extraction of the most important features of the object to be discriminated. Three types of features were selected for skin lesion classification. The features are: geometric (shape), texture and color as mentioned above. The dimension of the final feature vector will depend on the parameters to be taken into account, being the texture feature, LBP (Local Binary Patterns), a delimiter. We have as input for the machine learning part, two matrices with feature vectors. One with three LBPs (all uniform) and the other with three LBPs (the first two rotation invariant and the last one uniform). The feature dimensions of both vectors are 132 and 90, respectively.

Classification

In order to classify an image using extracted features, an appropriate learning algorithm must be chosen. There are two types of algorithms accessible in machine learning, supervised learning algorithms and unsupervised learning algorithms. Supervised learning algorithms take the classified access data and train a model to produce predictions of the response to new unobserved data. Unsupervised algorithms are used when classified data are not available [13]. Since we do have labeled data our approach will be supervised learning. In which several algorithms will be tested to identify which one is the most optimal for our classification. It should be remembered that our objective is the classification of skin lesions between nevus, melanoma and seborrheic keratosis. This means that we have three classes as output.

is common to refer to a minority or “positive” class, and a majority or “negative” class. In several real applications it

The Fig. 9 shows the flowchart of the proposed ML (Machine Learning) classification algorithm. In our system the programming language used is Python 3. Dedicated libraries were used to work with machine learning algorithms such as sklearn [14]. We also used libraries for csv file management (pandas), scientific computing (numpy) and visualization of graphs and tables (matplotlib). All these libraries are loaded at the beginning of the code in order to be able to use their functions during the programming of the system. Next, we load our functions created to speed up the loading of documents in CSV format, the presentation of metrics and the model estimator that will score the models to take the best performing one.

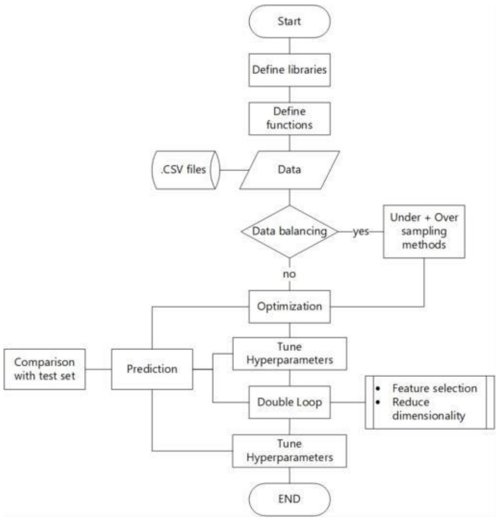

Fig. 9. Proposed ML classification algorithm

1) Data: To start the classification process, it is necessary to load the information collected on the features of each image. We proceed to read the .csv files, previously created (feature extraction explained in the Image Processing Section) for the two data sets to be worked on (training and test sets, 2150 and 600 images respectively). In Fig. 10 are presented the classes distribution for the training and test sets. Analyzing both figures, it can be clearly noticed that the class distribution is unbalanced, so one option to work on is to fix this unbalance.

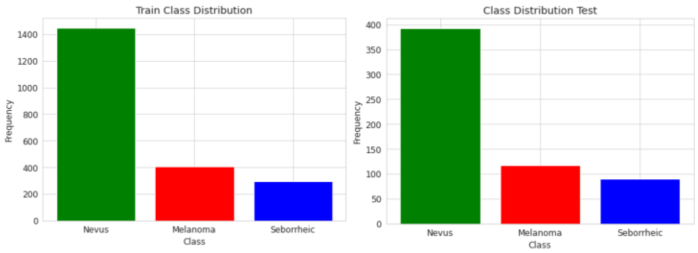

Fig. 10. Train and test data distribution

2) Data Balancing: Data imbalanced classification has been traditionally related to binary data sets. In particular, it

is normal to deal with more than two classes. Some clear examples are activity recognition, target detection, microarray research, protein classification, medical diagnosis and video mining. All of them have the same problem: the distribution of classes among examples is not homogeneous [15]. The simplest approach to address the problem of imbalanced learn- ing is the use of sampling methods. They consist in modify the data set using different procedures to provide balanced or better data distribution to the subsequent learning tasks. The advantages of use them are because they are simple and easily configurable and can be used in synergy with any learning algorithm. In this study we will work with both, raw data and balanced data.

With the application of this algorithm, it is possible to balance the data in a coherent way that will not affect significantly at the moment of the classification. In Fig. 11 is shown a data balancing.

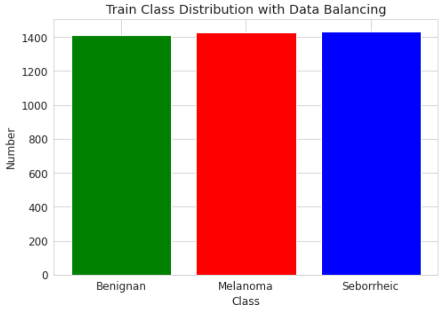

Fig. 11. Balance train Class Distribution

3) Optimization: For the optimization phase, a test is performed with several models, with various parameters, to find which one has the best classification performance. The metric used to measure the performance will be balanced accuracy.

Before continuing with the development, the metrics used to evaluate the performance of the algorithm should be explained. Accuracy is the percentage of correctly classified instances. It corresponds to the sum of diagonal elements in the confusion matrix, divided by the total number of instances.

Precision evaluates the fraction of correct classified instances among the ones classified as positive, while recall is the fraction of total positive instances correctly classified as positive. In imbalanced classification, misclassifying instances of a minority class is often much more costly than misclassifying instances of a majority class. In the medical domain, it is more expensive not to predict a rare pathology than a false alarm [15]. For this reason, the main objective is to minimize false negative cases. In order to have a better metric, the balanced accuracy is used instead of the accuracy.

Balance accuracy is the arithmetic average of true positive and true negative rates. The ROC curve is a plot of False Positive Rate (FPR) on the x-axis, and True Positive Rate (TPR) on the y-axis. In the ROC space, a good classifier should reach as close to the top left corner as possible. This corner corresponds to perfect classification. The upward diagonal indicates random performance. We select 8 different classifiers:

- Super Vector Machine
- Random Forest
- K-Neighbors
- Decision Trees
- Extra-Trees
- Gradient-Boosting
- Gauss Naive Bayes
- Voting

The output will be a graph corresponding to the balanced accuracies of each algorithm. In which it will be possible to detect which algorithms have the highest metric and which have the lowest.

4) Tuning: For the fine-tuning of the selected classification algorithm, a pipeline was developed, which will take as evaluation criteria the parameters of each algorithm. The output will be the best parameters, which together achieve the best accuracy.

5) Feature engineering: To reduce the dimension of the matrix and for feature selection, a double loop was created to allow both chi-square and PCA analysis.

- Feature reduction: The chi-square method was chosen for feature reduction. It is used in statistics to test the independence of two events. In feature selection is use it to test whether the occurrence of a specific feature and the target are independent or not. For each feature and target combination, a corresponding high chi-square score indicates that the target column is dependent on the feature column.
- Dimensionality reduction: In order to reduce the dimension of the features matrix, we chose to use PCA (Principal Component Analysis), which increases interpretability yet, at the same time, it minimizes information loss. It helps to find the most significant features in a data set. The output is the number of features (k) found with a specific number of components that achieved the best metric, balanced accuracy.

6) Second tuning: Once the number of features in the matrix has been reduced, we proceed with a final refinement of the parameters of the selected classification algorithm. The output are the parameters that obtained the best metrics when compared within a pipeline.

Each of the stages in which a change was made, from algorithm selection to the final tuning of the parameters, a prediction of results was made based on the features of the test set. The performance of the prediction was checked by comparing it with the test set classification vector (real vs. predicted value).

B. Results

This section will show the results obtained for each of the tests performed. We had two feature vector matrices as input, one with only LBP uniform which we will call Uniform features and another with rotation invariant LBP which we will call ROR features. Tests with and without data balancing will be performed for each Uniform and ROR feature matrix.

If any data balancing is not applied, the training set is maintained with the same class distribution (1450 images of Nevus, 404 of Melanoma and 296 of seborrheic keratosis) for any case.

Otherwise, if data balancing is applied, the distribution of classes changes according to the type of features being treated. For ROR features the distribution of classes is the following: 1414 images of Nevus, 1425 of Melanoma and 1435 of seborrheic keratosis, and for Uniform features is: 1415 images of Nevus, 1422 of Melanoma and 1435 of seborrheic keratosis.

The output of the optimization section is a figure comparing the performance of all the chosen algorithms (Balance accuracy is the used metric). Fig. 12 show the responses of the optimization phase depending on whether they are Uniform or ROR features and with or without data balancing It is observed that the algorithms that has the best-balanced accuracy (BMA) for imbalance data is the SVM (SVC) and for data balanced is Random Forest (RFC). Both models will be chosen to work for the following phases, in their respective sets.

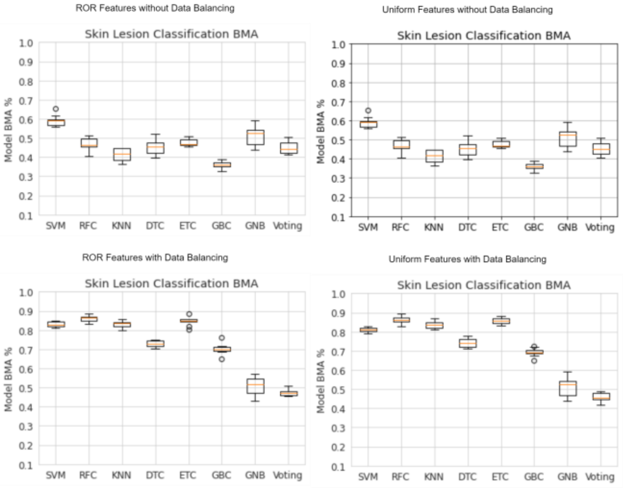

Fig. 12. Optimization output per set

For our proposed classification algorithm, the best result was obtained by applying an SVM classifier to our ROR feature without data balancing, with also feature and dimensionality reduction. The balanced accuracy value achieved was 64.90% (65%).