Skin Lesions Segmentation and Classification with Image Processing, Machine Learning and Deep Learning techniques

Kazi Saeed Alam, Gonzalo Esteban Mosquera Rojas and José Carlos Reyes Hernández
University of Cassino and Southern Lazio
Cassino, Italy
Erasmus Mundus Joint Master Degree in Medical Imaging and Applications (MAIA)
kazisaeed.alam@studentmail.unicas.it
gonzaloesteban.mosquerarojas@studentmail.unicas.it
josecarlos.reyeshernandez@studentmail.unicas.it

***Abstract*—The most fatal variety of skin cancer is the malignant melanoma. Increasing cases of malignant melanoma over the past few years are creating concerns among all the researchers and biologists. Despite being the deadliest form of skin cancer, melanoma can be treated if it is detected early in advance which is drawing increased attention to the researchers to help with the automatic detection of skin lesions from dermoscopic or microscopic images physicians detecting malignant melanoma with the aid of computer aided service. Skin lesion segmentation to detect the lesion areas and then classifying them into three most common types of categories (benign, melanoma and seborrheic keratoses). In this work, traditional unsupervised and supervised approaches have been analysed and proposed to segment and classify skin lesions properly. Dice Score and Jaccard score was used to compare the performance of the segmentation task where area under curve (AUC) and balanced accuracy specially for melanoma classes were considered for classification. For skin lesion segmentation, traditional unsupervised approach achieved around 72.1% Jaccard Score with a subset of 200 images from ISIC-2017 Dataset and around 55.4% Jaccard Score for the whole dataset of 2000 images. Whereas, UNET, the deep learning based approach achieved 65.4% Jaccard Score. On the other hand, for Skin Lesion Classification, among all the Machine Learning based classifiers applied, SVM yielded the best accuracy of 64.98% whereas deep learning based pretrained classifier ResNet outperformed by achieving an accuracy of 73% with 68% correct detection of melanoma class. A number of different feature extraction and selection techniques combined with various ML and DL models was studied and compared to get the best performing model.**

***Index Terms*—Skin Lesion, Melanoma, Lesion Segmentation, Lesion Classification, Traditional Approaches, Machine Learning, Deep Learning**

I. Introduction

A superficial growth or patch of skin that does not resemble the skin around it is known as a skin lesion. One in three cancers in the world are skin cancers, which frequently start in the epidermal layer of the skin mostly due to the ultraviolet light exposure. Skin Cancer Organization presented a report in 2019 that between 2008 and 2018, there were 53 percent rise in new melanoma cases detected each year, and one person worldwide passed away from the disease every hour [1]. The increasing cases of malignant has drawn attention of the researchers to find out an automatic segmentation and classification of the skin lesions with the aid of computed aided device to help the experts as early detection of these skin cancer can reduce the risk of death significantly.

Though the interset to perform research with skin lesion images has increased significantly over the years, segmenting and classifying skin lesions is a challenging task to do due to the different lesion characteristics, such as uneven color distribution, irregular shape, border, and textures. In this work, the dermoscopic and microscopic images of skin lesion examinations are used from ISIC-2017 Dataset [2]. Presence of artifacts like skin hairs, tapes, rulers, water bubbles, gel, markers, dark circles etc have made the task even more demanding. An overview of the different types of artifacts present in skin lesion images can be visualised in the Figure 1. These artifacts are one of the main obstacles to the improvement of the performance of the segmentation and classification tasks.

Various proposals have been made and also are being made to find out the techniques to segment and classify correctly the different types of cancerous and non-cancerous skin lesions. Some proposals are unsupervised using the tradiotional image processing based techniquues whereas recent advances shows an upward trend of applying deep learning based techniques. Traditional techniques are faster but perform poorly whereas deep learning based approaches have showed improved performance but require longer training time and large datasets.

In this work, both the unsupervised and supervised approaches have been applied and compared for the two different tasks: Skin Lesion Segmentation and Classification on ISIC 2017 Dataset (2000 images for training, 150 images for validating and 600 images for testing). Various image processing based techniques like Top-Hat transforms and morphological operations were performed to get rid of the artifacts discussed in the last paragraph. Then the prepossessed images are fed into the models to segment and classify the images properly.

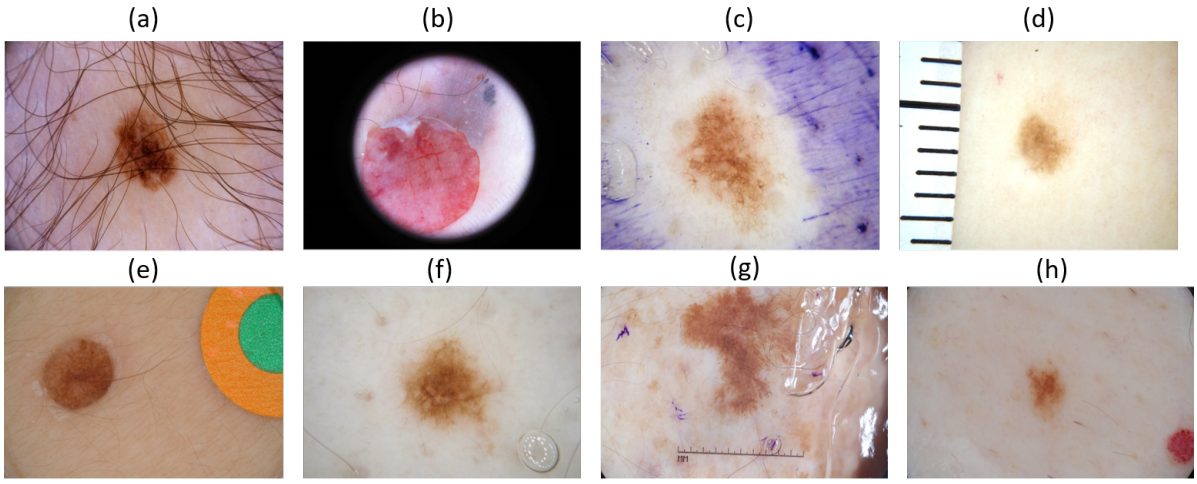

Fig. 1. Presence of various Artifacts in the Skin Lesion Images. (a) Skin Hairs, (b) Dark Border, (c) Markers, (d) Rulers, (e) tapes, (f) water bubbles, (g) gel, (h) non-uniform vignetting.

Unsupervised K-means clustering was applied to segment the skin lesions from randoly chosen 200 images from the dataset. To perform classification, various machine learning models like SVM, Logistic Regression, Gradient Boosting, Adaboost, Decision Trees, Random Forest, XGBoost, Naive Bayes, KNN, Bagging Classifiers, Stacking Classifiers have been applied for classification with 49 features extracted from the preprocessed images. An Extensive feature selection process was also taken into account before applying classification. Overall, SVM Outformed all the other models.

In terms of deep learning based approaches, UNET was chosen for segmentation whereas Transfer Learning using pretrained models like VGG16 and ResNet50 were considered for classification. Also, the hyperparameters were tuned properly with the validation set to get the best combination. Also various types of loss functions like Focal Loss, Cross-Entropy Loss, Dice Loss, IoU Loss were applied for training the model. The performance metrics used for segmentation were Jaccard Score and Dice Score, whereas AUC(Area Under the curve) and Balanced Accuracy were chosen for classification.

II. Literature Review

Since this project includes three different components: Advanced Image Analysis (AIA) for handcrafted segmentation, Machine Learning (ML) for classification, and Deep Learning (DL) for both automatic segmentation and classification of skin lesions, it was required to review the work done by some other authors in these fields. A summary of the consulted research papers is presented in this section.

Regarding the handcrafted algorithms for AIA, Hameed et al. [3] generated a binary mask of the skin lesions by performing a hybrid segmentation technique involving K-Means, Otsu's thresholding, and morphological operations. Before this stage, a preprocessing pipeline was followed. The hairs, Field of View (FOV), and skin bandages were removed by creating binary masks and setting different thresholds to detect these artifacts. Also, morphological operations and the inpainting technique were used to remove the remaining dark objects. Denoising the image was performed as the first step of this algorithm.

Hassan et al. [4] proposed two different segmentation algorithms: the first approach used watershed [5]; for the second approach, the mean-shift segmentation algorithm was implemented. The hairs and FOV were removed by applying inpainting and morphological operations, respectively. For these authors, the watershed approach obtained a higher Jaccard score than the algorithm involving mean-shift.

Hatem M [6] followed an adaptative thresholding segmentation technique after applying a preprocessing algorithm where the hairs were removed, and the image was enhanced by using morphological closing.

Salido et al. [7] also followed the segmentation technique described before, but for the hair removal, an algorithm involving denoising, BottomHat transform, morphological operations, and harmonic inpainting was performed.

Melli et at. [8] applied different color clustering segmentation techniques such as k-means, c-fuzzy means, and mean shift, where this last one resulted being the best technique.

Concerning the ML task, Capdegourat et al. [9] defined a set of features based on the ABCD (Asymmetry, Border, Color, and Diameter) rule, which specifies a list of visual characteristics that help to identify malignant skin lesions. Also, features related to shape (irregularity, distance, abnormality), texture (Gray Level Co-occurrence Matrix (GLCM)), and some statistical features (mean, standard deviation) were extracted. To perform the classification, AdaBoost and Decision tree classifiers were selected.

Russell et al. [10] implemented a more general set of statistical and texture features) that could be applied to all the different skin lesion images in the ISIC 2017 dataset to train an ANN with three hidden layers, leading to a 63.79 % accuracy.

Marugan et al. [11] used Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN) as classifiers, where SVM got the highest score on the test set.

Sumithra et al. [12] proposed a method applying a fusion of decisions of both SVM and KNN using OR rule. By doing this, a 61 % of F-measure was obtained.

Finally, for the DL part, Hassan et al. [13] propose a novel automatic segmentation network named Dermoscopic Skin Network (DSNet), where a depth-wise separable convolution to project the learned discriminating features onto the pixel's space at different stages of the encoder. Using this approach, the authors obtained 77.5% accuracy on the ISIC 2017 dataset.

Zabir et al. [14] implemented a U-Net for segmentation and DCNN-SVM for classification. For U-Net, the authors applied spatial dropout and data augmentation to solve the problem of overfitting due to unbalanced data. For DCNN-SVM, the transfer learning technique was used. The results for each task were 0.80 Jaccard Index and 92 % mean accuracy, respectively.

Amirreza et al. [15] applied three pre-trained networks such as VGG16, ResNet18, and AlexNet, as feature generators to train an SVM classifier. By implementing this approach, the authors obtained 83.83 % for melanoma classification and 97.55 % for seborrheic keratosis.

Bills et al. [16] interestingly presented an approach to automatic skin lesion segmentation using U-Net without the power of a GPU by implementing a histogram equalization-based preprocessing step.

Vandana et al. [17] implemented an EfficientNet-B0 for classifying seven different skin lesion categories. The authors found that this architecture outperformed the ResNet50 with few parameters since the transfer learning technique was used.

III. Traditional Advanced Image Analysis Approach for Skin Lesion Segmentation

A. Methodology

After having done the literature review and examined different approaches to performing the primary goal of this part of the project, a workflow for skin lesion segmentation was defined. There are three main parts of this workflow than can be identified: pre-processing, segmentation, and post-processing. Each of these stages will be explained in the following sections and subsections. It is important to mention that many techniques taught during the Advanced Image Analysis course were required to be applied for each stage of the overall algorithm. This algorithm can be summarized in Fig. 2.

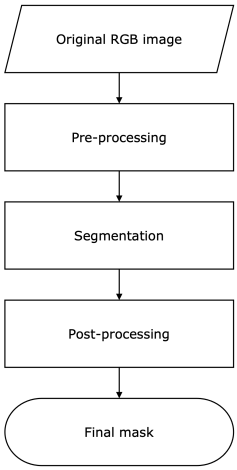

Fig. 2. Stages of Advanced Image Analysis.

B. Pre-processing

Since many variables can interfere when capturing the sample images, such as skin nature, environmental conditions, and capturing device, it is expected to find some artifacts in the images provided. Due to these variables, the artifacts found in the skin images are some hairs, noise, and the presence of the field of view (FOV) of the capturing device, among others. Hence, before applying any segmentation algorithm to the images, a pre-processing pipeline must be performed to remove these artifacts and any additional barriers. Thus, a decrease in the generated region of interest accuracy is prevented.

The algorithm of this stage is summarized in Fig. 3, and each step of it will be described in the following subsections.

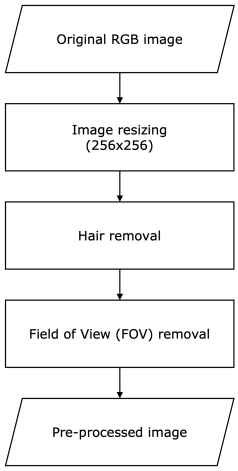

Fig. 3. Stages of Image Pre-Processing.

C. Hair removal

Given that the dataset used in this project contains high-resolution images with many different dimensions, the first step in this pipeline is to resize them all to 256x256x3. Thus, the images in the dataset are uniform, and the computational cost and processing time are reduced.

Once all the original RGB images are resized, a median filter with a kernel size of 3x3 is applied for denoising. Then, these smooth images are converted to grayscale before using Greyscale Morphology. As the hairs are dark objects in the images, all the pixels' intensities are inverted so that the Top Hat transform can detect them.

However, since the "getStructuringElement()" function of OpenCV only generates either horizontal or vertical rectangular structuring elements (SEs), and as the hairs appear in different orientations, a function for creating a series of rotated SEs at different angles is defined. Once these rotated SEs are created, all of them are used in a sum of TopHat transform applied to each image. So, all the hairs that fit into these SEs are removed, whereas the rest of the components of the images are enhanced. Afterward, both closing and dilation morphological operations are applied to a binarized image to create a mask that will be used in the following step [18].

Finally, the inpainting (TELEA) [19] method is used to get more homogeneous images and to refine the Top Hat hair removal. This function receives the dilated image obtained in the previous step as a mask to indicate the area that needs to be inpainted. By doing this, all the remaining undesirable dark details will be replaced by their neighboring pixels, making them look like the surrounding neighborhood. The step-by-step of this algorithm is shown in Fig 4. Also, the effect of each of these steps can be seen in Fig 5.

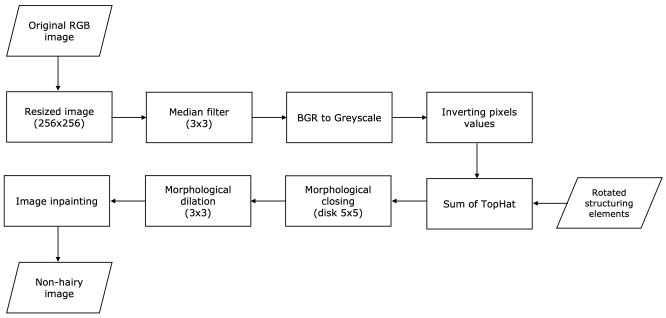

Fig. 4. Hair Removal Steps.

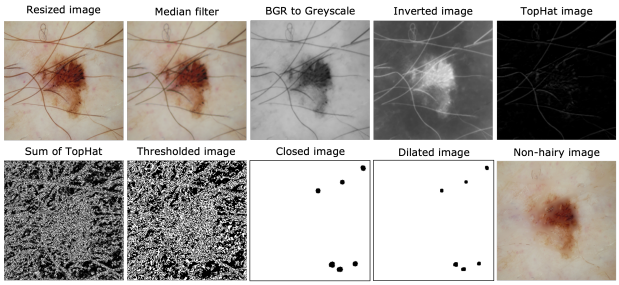

Fig. 5. Effect of each step of the hair removal algorithm on the image.

D. Field of View (FOV) removal

In some cases, the FOV of the capturing device can be observed as a black outer ring in the images. The size of this FOV also varies from image to image. To remove this artifact, the following algorithm is applied, having as the input the non-hairy image generated in the previous step:

First, a circular mask is created with a radius of the shortest measure of the dimensions (width and height) of the radius of the possible black ring present in the input image. Then, the intensities values of the pixels are counted depending on a defined threshold (T) to extract the number of low intensities (dark pixels). If this count is higher than T, there is a FOV in the image. Otherwise, there is not, and the input image is returned as an output. An iterative process of subtracting the circular mask from the input image is applied to all images whose black pixels values are higher than the threshold. After each iteration, the new count of black pixels is checked. If it is still higher than T, the process is repeated, and the radius of the circular mask is reduced to remove another black pixel region. Once this count is lower than T, the iterations are stopped, and the resulting image does not contain the initial black ring.

This process is explained in Fig 6. Also, some examples of this FOV removal are shown in Fig 7.

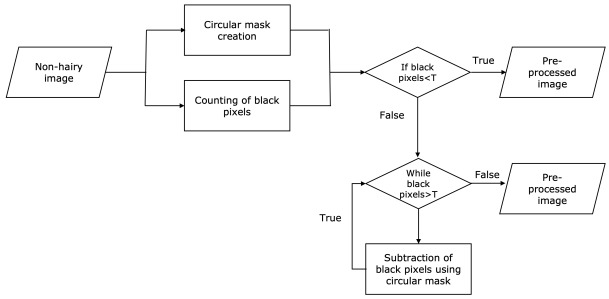

Fig. 6. FOV Removal Process.

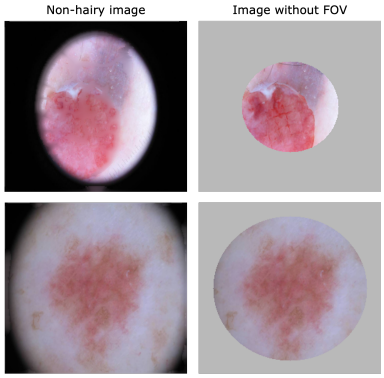

Fig. 7. Effect of each step of the hair removal algorithm on the image

E. Segmentation

The primary purpose of performing this segmentation is to generate a mask that will be used to extract the region of interest (ROI) from the pre-processed image. Extracting this ROI will be helpful for classification since it contains relevant information that characterizes each lesion, known as a feature.

The segmentation algorithm chosen in this project was K-means, described in the following subsection.

F. K-Means

The input for this stage is the pre-processed image obtained in the previous steps. This image must be converted into

a 2D array that contains the flattened height and width of the input image for each of the three RGB channels ((w*h), RGB). Then, the stopping criteria and the number of clusters must be defined. Since there is a significant number of pixels for each image, two stopping criteria are set: the k-means iterations will stop either when the maximum number of iterations is exceeded or if the specified accuracy (epsilon) is reached. These criteria are chosen to reduce the processing time. Regarding the number of clusters (k), it can be identified two principal regions: the lesion (foreground) and the rest of the skin (background).

Once the K-means algorithm stops, the image must be converted back to a 3D array to be able to visualize it. Next, a median filter is applied to denoising before getting the segmented region.

Since the segmentation result obtained at this point is an RGB mask image, and the final purpose of this process is to get a binary mask, the color mask is converted into a greyscale image to perform Otsu thresholding. Once this thresholding is done, the binary mask is obtained.

This segmentation process is explained in Fig 8, and Fig 9 shows some segmented color images and their respective binary masks.

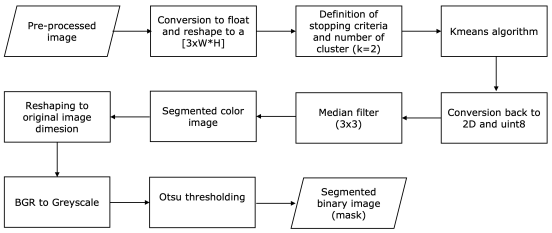

Fig. 8. Segmentation algorithm

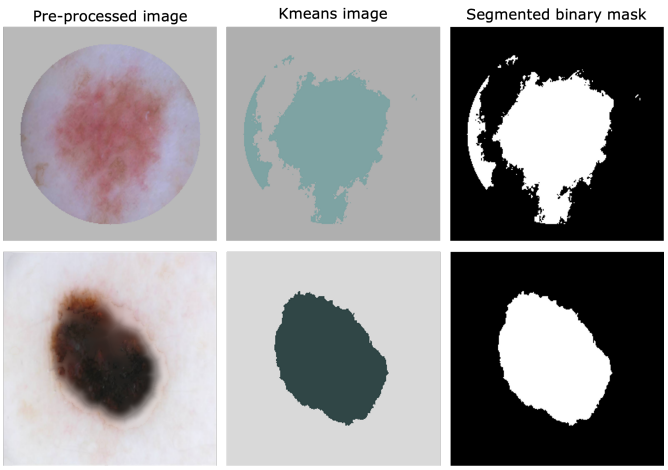

Fig. 9. Segmentation results: Colored and Binary Images.

G. Post-processing

As shown in Fig 9, there are some cases in which the previous segmented binary images have imperfections, such as holes that need to be filled or contours that do not belong to the target ROI. An additional step must be done after the segmentation stage to fix this. In this case, the largest connected component must be extracted from the resulting output of the previous segmentation process.

H. Extracting the largest connected component

Once the segmented binary images are input, the algorithm will start by searching each image's contours (connected components). The next step is to calculate the shape with the maximum area among all the possible ones and store it in a new temporary image. Afterwards, a black image of the same size as the input image is created, and the largest connected component found in the previous step is printed on it. Thus, the result is a binary mask without holes that need to be filled or false ROIs, and this constitutes the final output of the Advanced Image Analysis workflow. This last mask will be used to extract the ROI from all the preprocessed images to get what will be the base for the feature extraction part of the Machine Learning section of this project.

This algorithm can be visually understood by looking at Fig 10. Some post-processed images are shown in Fig 11.

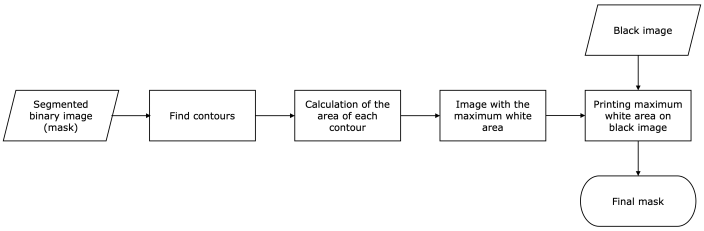

Fig. 10. Largest Connected Component Extraction Steps.

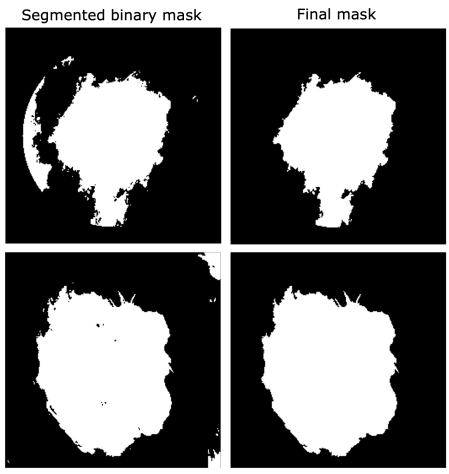

Fig. 11. Images after Largest Connected Component Extraction.

I. Results

To evaluate the performance of the skin lesion segmentation algorithm followed in this project, the Jaccard Index (JC) evaluation matrix was proposed to be used in the guidelines.

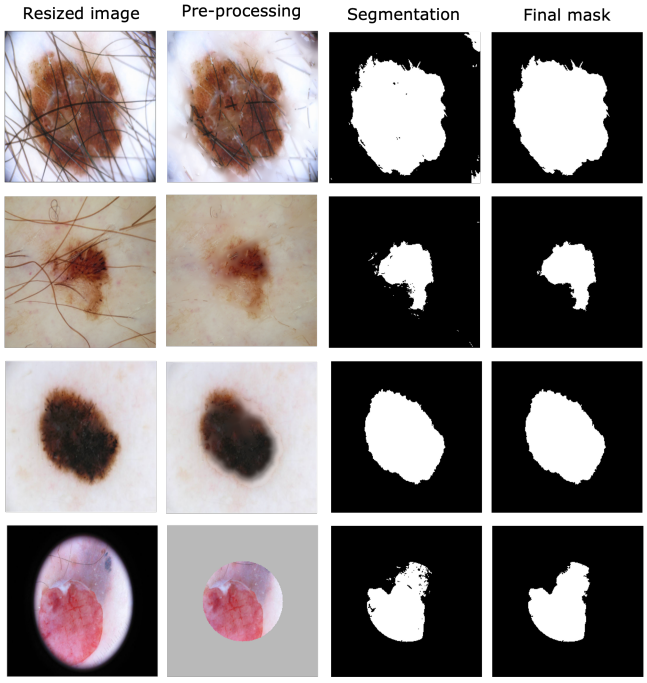

Fig. 12. Skin lesion segmentation Results.

This JC gives the similarity and diversity of samples sets, and it is defined as follows:

$$JC = \frac{X \cap Y}{X \cup Y} \tag{1}$$

Also, the Dice Coefficient was used as a comparison metric, and it is defined as follows:

$$DSC(X, Y) = \frac{|X \cap Y|}{|X| + |Y|} \tag{2}$$

For this Advanced Image Analysis part, a subset of the ISIC Challenge 2017 dataset consisting of 200 images was used to evaluate the performance. To do this evaluation, a dataset containing the ground truth mask of each skin lesion image was provided. Some of the results can be seen in Fig. 15.

The JC and DSC metric for every image in the dataset can be seen in Fig 13 and Fig 14, respectively. Also, the mean value for each metric is shown **(DSC=0.8109, JC=0.7040)** for 200 images, A subset of the ISIC-2017 Dataset. **(DSC=0.6209, JC=0.546)** were achieved after applying to all the 2000 images.

IV. Machine Learning Approaches for Skin Lesion Classification

The goal of the work for the Machine Learning part is to do the classification of the skin lesion images. For this purpose, the segmentation masks from the Deep Learning pipeline were used. The Fig. 16 presents a summary of the procedures followed in this pipeline, and in methodology section each of those stages are explained in detail.

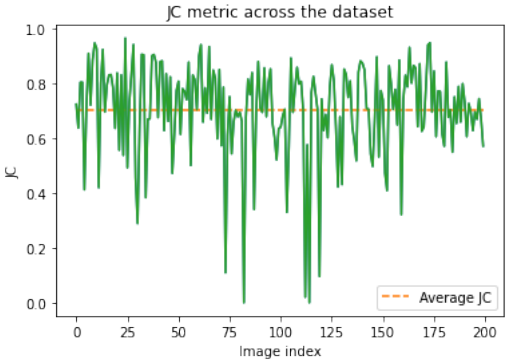

Fig. 13. JC metric across the dataset

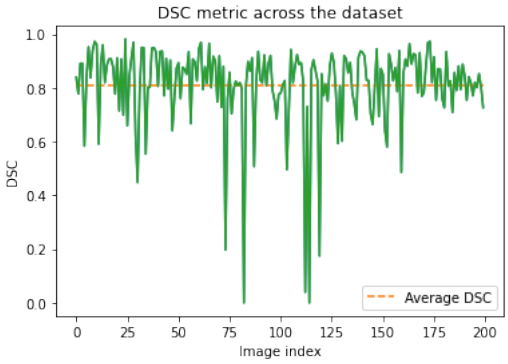

Fig. 14. DSC metric across the dataset

A. Methodology

1) Post-processing of segmentation masks from the Deep Learning pipeline: This process consisted on filling holes inside the lesion and eliminating small noisy elements that were not part of the region of interest. As explained in subsection H of the Image Processing pipeline, this is done by looking at the contours of each image, finding the one with largest area and painting it on a black image with the same size as the original image. This post-processing is required since it ensures that the regions of interest will correspond to the whole lesion that is meant to be extracted, with no loss of information as it would happen if the inner wholes of the initial segmentation images were kept. The Fig. 17 shows an example of the post-processing procedure.

2) Feature Extraction: When tackling with a classical Machine Learning classification problem, the first thing that must be done is the feature extraction from the raw data. Therefore, this was the first step of the pipeline. In order to decide which features to calculate, some research was done on what are the aspects clinicians focus on when evaluating a skin lesion image in their daily practice. According to the American Academy of Dermatology Association [20], when examining if a skin lesion could potentially be melanoma, the ABCDE rule must be taken into account. ABCDE stands for asymmetry, border, color, diameter and evolving, respectively. Once the lesion

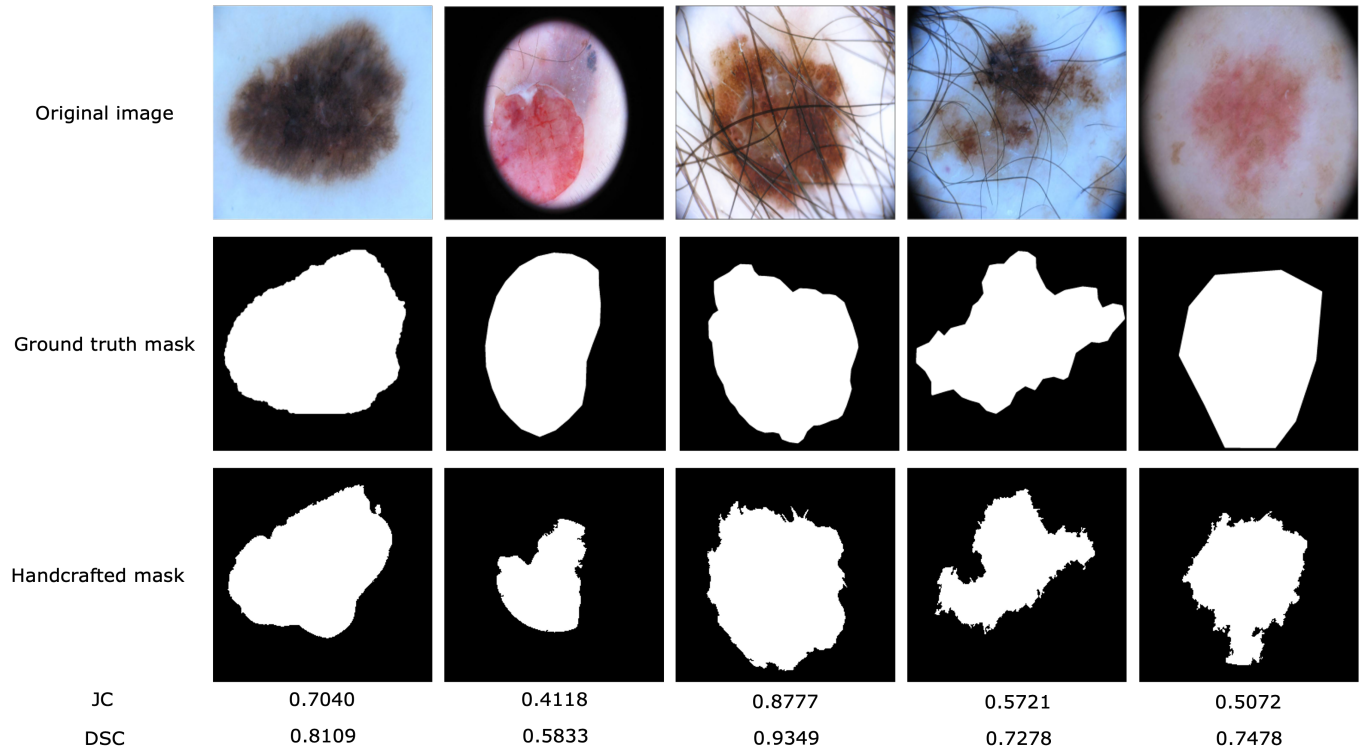

Fig. 15. Qualitative and quantitative results of handcrafted segmentation

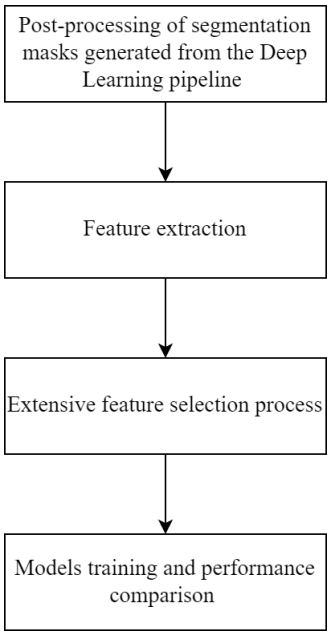

Fig. 16. General summary of the Machine Learning Pipeline processes.

region is located, the first aspect (asymmetry) divides the lesion in two parts and evaluates how alike the two halves are. The less alike, the higher asymmetry in the lesion and the higher risk it could represent. The second aspect studies

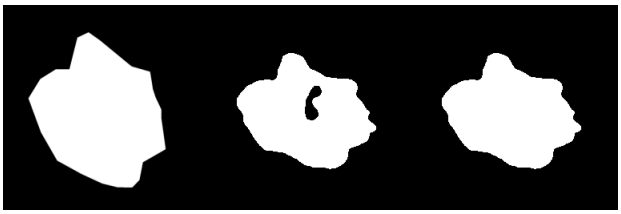

Fig. 17. Post-processing of Deep Learning Segmentation Images. From left to right: ground truth, Deep Learning segmentation mask before and after postprocessing.

irregularities in the border of the lesion. Color aspect focuses on studying the variety of colors present in the lesion. The diameter of the regions must be also studied, since typically, melanomas have diameters higher than 6 mm. Finally, the evolving aspect takes into account how the lesion changes in terms of size and shape over time.

The features calculated for this work were all based on the above mentioned rule. A total of 49 features were calculated and these were:

- Convexity Defects: in order to evaluate irregularities in the borders of the lesion, the convexity defects were taken into account. To calculate these features, a convex hull for each region is built. The convex hull is defined as the smallest convex polygon that contains a particular shape. The convexity defects will be all deviations of the shape from its convex hull. The number of important convexity

defects (those with norm greater than 15, a value that was found experimentally) are taken as features.

- Circularity: This feature is calculated in order to evaluate shape irregularity of the lesion. It is mathematically defined by the following expressions:

$$C = \frac{4\pi A}{P^2}, C = \frac{P^2}{4\pi A} \tag{3}$$

where A and P denote the area and perimeter of the lesion, respectively. In this work the first expression is used. This fraction is 1 for perfectly circular shapes and tends to 0 for highly non-circular shapes.

- Lesion Diameter
- Mean and standard deviation of the lesion intensity values on the blue, green and red channels. These features are called hereinafter as meanB, meanG, meanR, meanBGR, stdB, stdG, stdR, stdBGR.
- Maximum pixel intensities for B, G and R channels of BGR color spcae. These features are called hereinafter as maxB, maxG and maxR.
- Mean of the lesion intensity values on the H, S, and V channels for the image converted to HSV color space. These features are called hereinafter as meanH, meanS, meanV, meanHSV.
- Mean of the lesion intensity values on the Y, CR, CB for the image converted to YCRCB color space. These features are called hereinafter as meanY, meanCR, meanCB, meanYCRCB.
- Mean of the lesion intensity values on the L, A, B channels for the image converted to CIELAB color space. These features are called hereinafter as meanL, meanA, meanBLAB, meanLAB.
- Asymmetry features: to evaluate the asymmetry of the lesion, the following procedure was done: first, the lesion is divided in two symmetric halves, as shown in Fig 18. Then, the mean of pixel intensity values for the blue, green and red channel for each half of the lesion and the absolute difference of intensities for each chanel (B,G,R) are taken as features. These features are called hereinafter as meanBh1, meanGh1, meanRh1, meanBGRh1, meanBh2, meanGh2, meanRh2, meanBGRh2, meanBdiff, meanGdiff, meanRdiff, meanBGRdiff.
- Texture features: mean correlation, homogeneity, energy and contrast of the lesion obtained by the computation of the GLCM matrix.
- Skewness, entropy, kurtosis of the lesion.
- Total sum of pixel intensities within the lesion.
- Color variation of the feature in the blue, green and red channel, defined as follows [21]:

$$C_r = \frac{\sigma_r}{M_r}, C_g = \frac{\sigma_g}{M_g}, C_b = \frac{\sigma_b}{M_b} \tag{4}$$

where σ denotes the standard deviation of pixel intensities for the corresponding color space (R,G,B) and M denotes the maximum pixel intensity value in the color space.

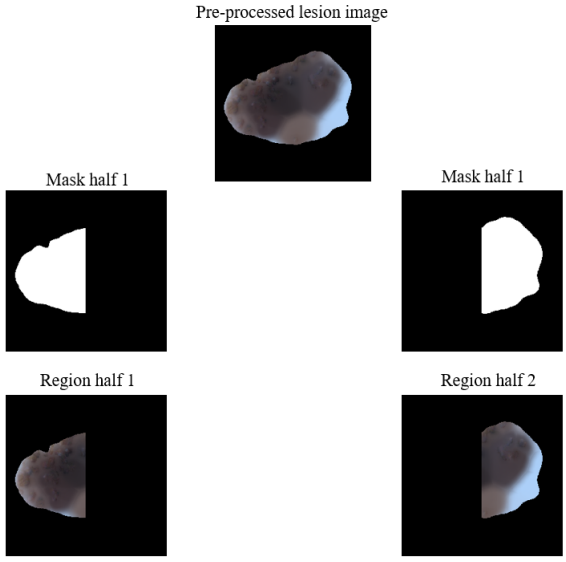

Fig. 18. Symmetric halves of the lesion and its corresponding mask

Once the features are calculated, they are scaled using standard scaler, which standardizes the features by making it to have zero mean and unit variance.

3) Extensive feature selection process: In order to perform extensive feature selection, two datasets were taken into account for the training of the models: the original training dataset and a a slightly bigger version of the training dataset, which includes the training and validation sets from the ISIC 2017 challenge, and sums up to 2150 samples. The figure 18 shows the sample distribution for each of the two datasets.For this case, 0 stands for beningn lesion, 1 for melanoma and 2 for seborrheic keratosis. The initial training set has the following distribution: 1372 samples for benign, 374 for melanoma and 254 for seborrheic keratosis. The validation set has the following distribution: 78 samples for benign, 42 for melanoma and 30 for seborrheic keratosis. Therefore, the training+validation set has 1450 samples for benign, 404 for melanoma and 296 for seborrheic keratosis. From the figure

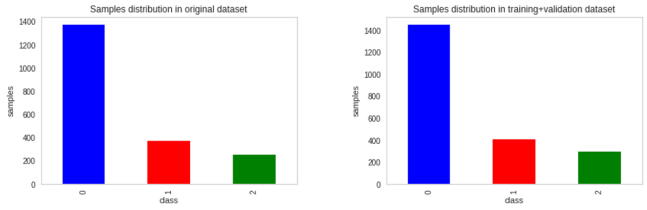

Fig. 19. Samples distribution for training and training+validation dataset

above it is noticeable that this is a class-imbalanced problem. Therefore, the working setup was based on doing extensive feature selection on 4 different scenarios and using Support Vector Machine as the base classifier. These setups employ two different methods to tackle class imbalance: class weighting and SMOTE oversampling, being the latter in which synthetic

data from minority classes is created based on KNeighbors algorithm. The training scenarios were the following:

- Original training set using class weighting.
- Training+validation set using class weighting.
- Original training set oversampled with SMOTE, where all classes have the same number of samples, 1372.
- Training+validation set oversampled with SMOTE, where all classes have the same number of samples, 1450.

A summary of the algorithm for extensive feature selection is depicted in Fig 20. As shown in the figure, there is a loop

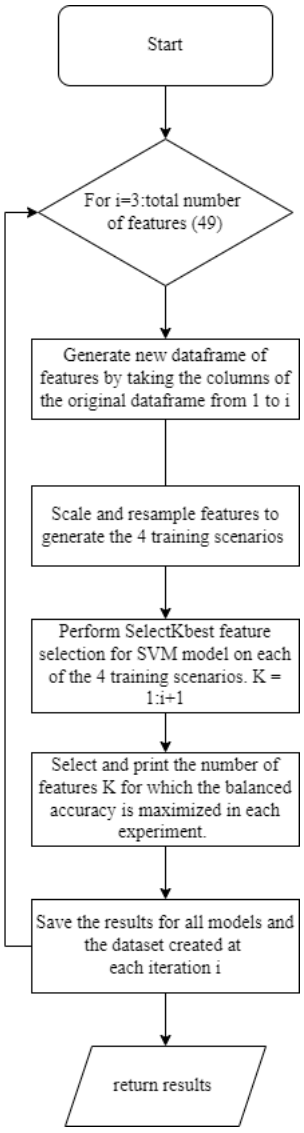

Fig. 20. Extensive feature search algorithm summary

that goes from 3 to the total number number of features (49 in this case) and will generate a new dataframe of features by taking the columns in the original dataframe that correspond to 1:i. For instance, if i=8, the new dataframe for this iteration will take the first 8 columns of the original feature dataframe and will perform SelectKbest feature selection, with K varying from 1:9 in each of the 4 training scenarios. The K best features are selected using the ANOVA F-value test. For each of the experiment the algorithm prints the name of the features that maximize the balanced accuracy score, and outputs all these results in a vector at the end of all iterations. In this way, the algorithm ensures that for every iteration the model with highest accuracy for each experiment is the one that is printed. A total of 4888 SVM models were trained during this process. The parameters for the model were fixed at RBF kernel, gamma='auto' and decision function shape One vs Rest.

4) Models training and performance comparison: Once the extensive feature selection algorithm finishes running, it will provide information on what is the features dataframe and the number of features within this dataframe for which the model gets the highest balanced accuracy. With this new dataframe of features other 10 models are trained and the SelectKbest feature selection is performed, to get a total of 11 final models for results comparison. The other trained classifiers were:

- Logistic Regression
- Gradient Boosting
- Adaboost
- Decision Trees
- Random Forest
- Extreme Gradient Boosting (XGBoost)
- Naive Bayes
- KNeighbors
- Bagging Classifier, using 10 estimators and SVM as base classifier
- Stacking classifier: a new classifier of the above mentioned models in stack.

B. Results

From the exhaustive feature selection search it was found that the highest balanced accuracy score was **0.6498**. This score was obtained from training the SVM model on the biggest dataset (training + validation) and using SMOTE as oversampling technique to handle the class imbalance. This results were obtained with a dataset of 41 features, from which 35 were selected. The 41 features were convexity defects, circularity index, lesion diameter, meanB, meanG, meanR, meanBGR (mean of the pixel intensities from all channels in BGR space), StdB, StdG, StdR, StdBGR (standard deviation of all pixels in all channels in BGR space), maxB, maxG, maxR (maximum pixel values from all channels in BGR space), mean correlation, mean homogeneity, mean energy, mean contrast, meanH, meanS, meanV, meanHSV (mean of pixel intensities from all channels in HSV space), meanY, meanCR, meanCB, meanYCRCB (mean of pixel intensities from all channels in YCRCB color space), skewness, kurtosis, entropy, value of the sum of pixel intensities,color variation in B, G and R channels, meanLAB, meanL, meanA, meanB (mean of pixel intensities from all channels in CIELAB color space), meanBh1, meanGh1, meanRh1, meanBGRh1 (mean pixel intensities from all channels in BGR space from the right half of the lesion). From this base feature space the rest

of the models was trained and SelectKBest feature selection was done using the ANOVA F-value test.

For all predictions presented in the next subsections, 0 stands for benign class, 1 for melanoma and 2 for seborrheic keratosis.

1) SVM classifier: For the SVM classifier, which was the base model used for extensive feature search, 35 features were finally selected out of the 41 preselected from the first level of the search algorithm, and these 35 features gave the maximum balanced accuracy, which was 0.6498. These features were: convexity defects, circularity index, lesion diameter, meanB, meanG, meanBGR, stdB,stdG, stdR, stdBGR, maxG, color variation in B,G,R channels, mean correlation, mean homogeinity, mean energy, mean contrast, meanS, meanHSV, meanY, meanCR, meanYCRCB, skewness, kurtosis, entropy, sum of pixel intensity values, meanLAB, meanL, meanA, meanBLAB (mean intensity values from CIELAB color space), meanBh1, meanGh1, meanBGRh1. Fig. 21 and Fig. 22 show the ROC curves and confusion matrix for the predictions of the model, respectively.

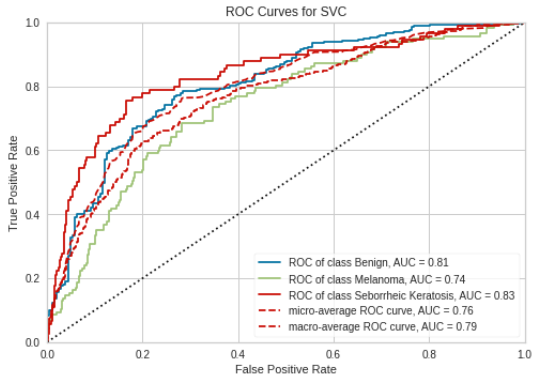

Fig. 21. ROC-AUC curve for SVM Classifier

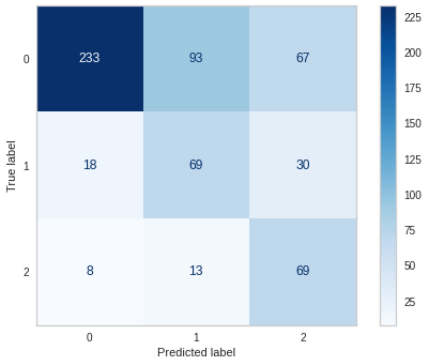

Fig. 22. Confusion Matrix for SVM Classifier

2) Logistic Regression: For the Logistic Regression classifier, the highest balanced accuracy score was 0.552, obtained by selecting 13 out of 41 features. These features were: convexity defects, lesion diameter, StdB, StdBGR, color variation in B, G and R channels of BGR color space, mean correlation, homogeneity and energy, skewness and entropy of the lesion. The Fig. 23 and Fig. 24 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

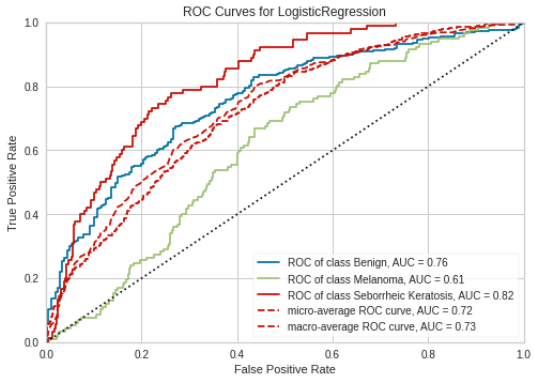

Fig. 23. ROC-AUC curve for Logistic Regression Classifier

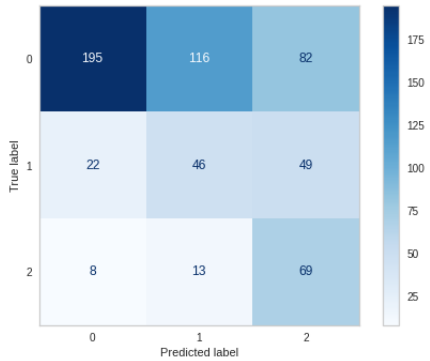

Fig. 24. Confusion Matrix for Logistic Regression Classifier

3) Gradient Boosting: For the Gradient Boosting classifier, the highest balanced accuracy score was 0.6251, obtained by selecting 28 out of 41 features. These features were: convexity defects, circularity index, lesion diameter, meanB, meanBGR, StdB, StdG, StdR, StdBGR, maxG, color variation in B,G,and R channels of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, meanHSV, meanCB, skewness, kurtosis, entropy, and sum of pixel intensities in the lesion. The Fig. 25 and Fig. 26 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

4) Adaboost Classifier: For the AdaBoost classifier, the highest balanced accuracy score was 0.6009, obtained by selecting all 41 features mentioned at the beginning of the results section. The Fig. 27 and Fig. 28 present the ROC-AUC curve and confusion matrix for the model predictions, respectively. respectively.

5) Decision Trees Classifier: For the Decision Trees classifier, the highest balanced accuracy score was 0.5277, obtained by selecting 35 out of 41 features. These features were: convexity defects, circularity index, lesion diameter, meanB,

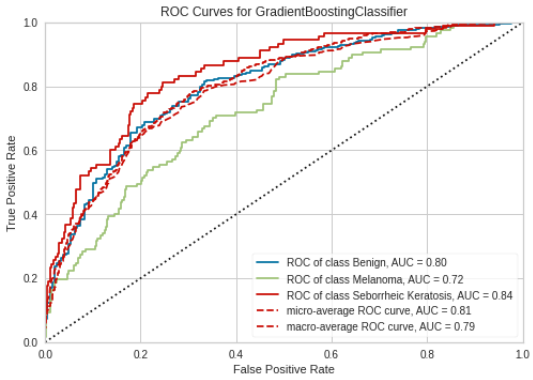

Fig. 25. ROC-AUC curve for Gradient Boosting Classifier

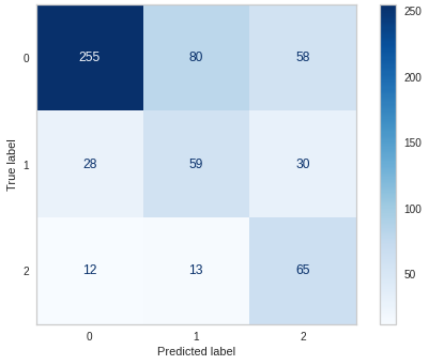

Fig. 26. Confusion Matrix for Gradient Boosting Classifier

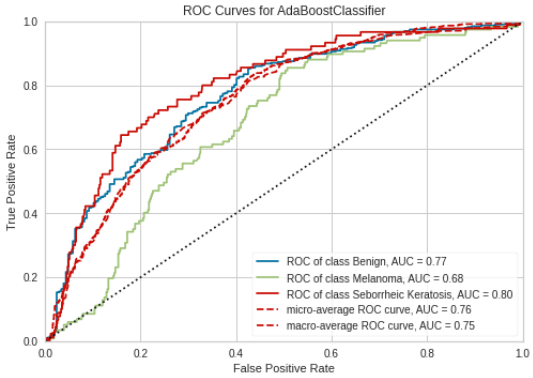

Fig. 27. ROC-AUC curve for Adaboost Classifier

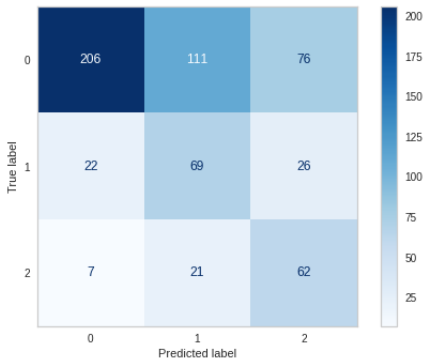

Fig. 28. Confusion Matrix for Adaboost Classifier

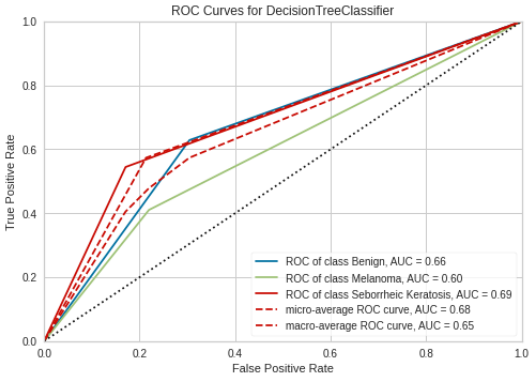

Fig. 29. ROC-AUC curve for Decision Tree Classifier

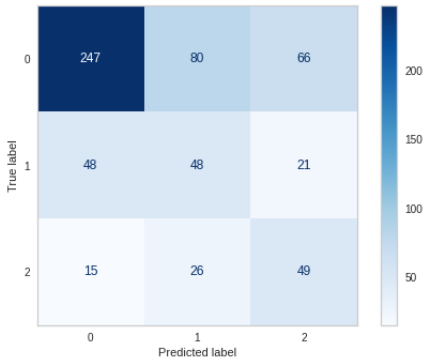

Fig. 30. Confusion Matrix for Decision Tree Classifier

meanG, meanBGR, stdB, stdG, stdR, stdBGR, maxG, color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, meanHSV, meanY, meanCR, meanCB, meanYCRCB, skewness, kurtosis, entropy, sum of pixel intensity values in the lesion, meanLAB, meanL, meanA, meanBLAB, meanBh1, meanGh1, meanBGRh1. The Fig. 29 and Fig. 30 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

6) Random Forest Classifier: For the Random Forest classifier, the highest balanced accuracy score was 0.5927, obtained by selecting 32 out of 41 features. These features were: convexity defects, circularity index, lesion diameter, meanB, meanBGR, stdB, stdG, stdR, stdBGR, maxG, color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, meanHSV, meanCR, meanCB, meanYCRCB, skewness, kurtosis, entropy, sum of pixel intensity values in the lesion, meanLAB, meanA, meanBLAB, meanBh1, meanGh1, meanBGRh1. The Fig. 31

and Fig. 32 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

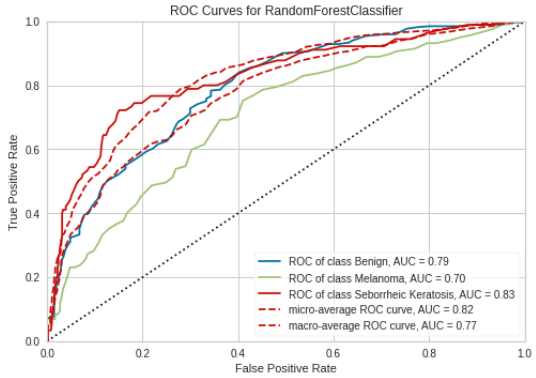

Fig. 31. ROC-AUC curve for Random Forest

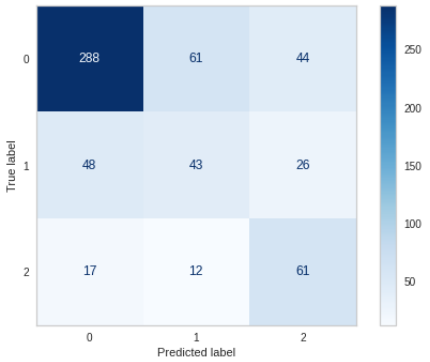

Fig. 32. Confusion Matrix for Random Forest

7) Extreme Gradient Boosting Classifier (XGBoost): For the XGBoost Classifier, the highest balanced accuracy score was 0.6247, obtained by selecting 28 out of 41 features. These features were: convexity defects, circularity index, lesion diameter, meanB, meanBGR, stdB, stdG, stdR, stdBGR, maxG, color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, meanHSV, meanCB, skewness, kurtosis, entropy, sum of pixel intensity values in the lesion, meanBLAB, meanBh1, meanGh1, meanBGRh1. The Fig. 33 and Fig. 34 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

8) Naive Bayes Classifier: For the Naive Bayes Classifier, the highest balanced accuracy score was 0.5376, obtained by selecting 39 out of 41 features. The discarded features were meanRh1 and meanV.The Fig. 35 and Fig. 36 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

9) KNeighbors Classifier: For the KNeighbors Classifier, the highest balanced accuracy score was 0.5596, obtained by selecting 19 out of 41 features. These features were: convexity defects, lesion diameter, stdB, stdG, stdR, stdBGR,

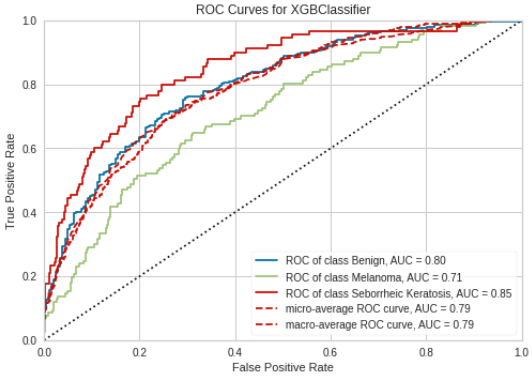

Fig. 33. Confusion Matrix for XGBoost Classifier

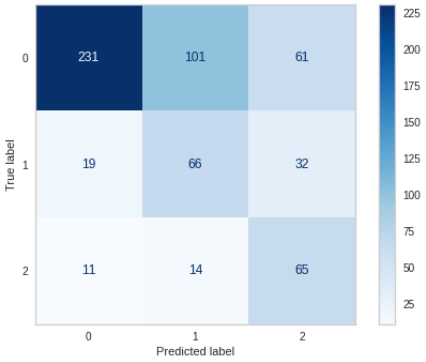

Fig. 34. Confusion Matrix for XGBoost Classifier

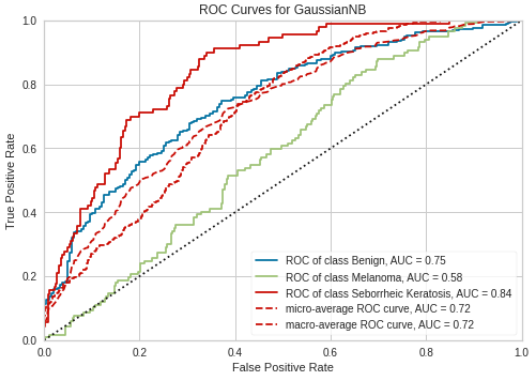

Fig. 35. Confusion Matrix for Naive Bayes Classifier

color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, skewness, kurtosis, entropy, sum of pixel intensity values in the lesion, meanBLAB. The Fig. 37 and Fig. 38 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

10) Bagging Classifier: For the Bagging Classifier, the highest balanced accuracy score was 0.6491, obtained by selecting 29 out of 41 features. These features were: convexity

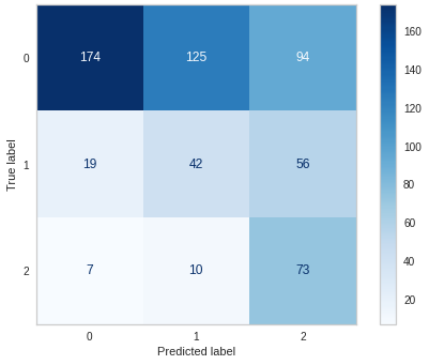

Fig. 36. Confusion Matrix for Naive Bayes Classifier

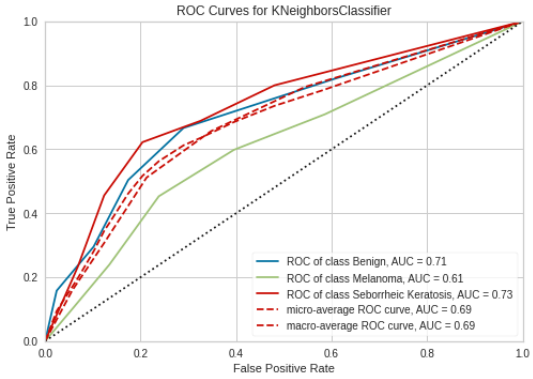

Fig. 37. ROC-AUC curve for KNeighbors Classifier

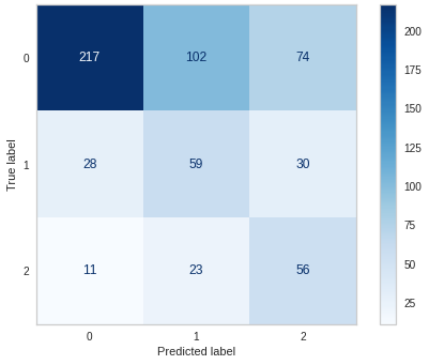

Fig. 38. Confusion Matrix for KNeighbors Classifier

defects, circularity index, lesion diameter, meanB, meanBGR, stdB, stdG, stdR, stdBGR, maxG, color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and contrast, meanS, meanHSV, meanCB, skewness, kurtosis, entropy, sum of pixel intensity values in the lesion, meanA, meanBLAB, meanBh1, meanGh1, meanBGRh1. The Fig. 39 and Fig. 40 present the ROC-AUC curve and confusion matrix for the model predictions, respectively.

11) Stacking Classifier: For the Stacking Classifier, the highest balanced accuracy score was 0.5422, obtained by

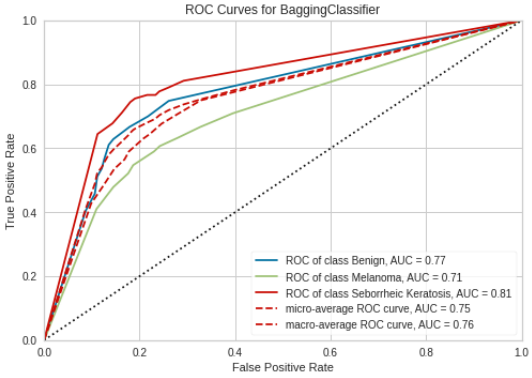

Fig. 39. Confusion Matrix for Bagging Classifier

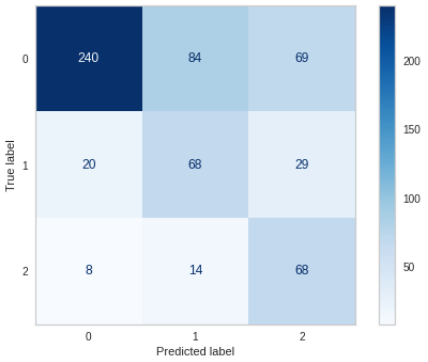

Fig. 40. Confusion Matrix for Bagging Classifier

selecting 12 out of 41 features. These features were: convexity defects, lesion diameter, stdB, stdG, stdBGR, color variation in channels B, G and R of BGR color space, mean correlation, homogeneity, energy and entropy. The Fig. 41 and Fig. 42 present the ROC-AUC curve and confusion matrix for the model predictions, respectively. Finally, Fig. 43 shows a

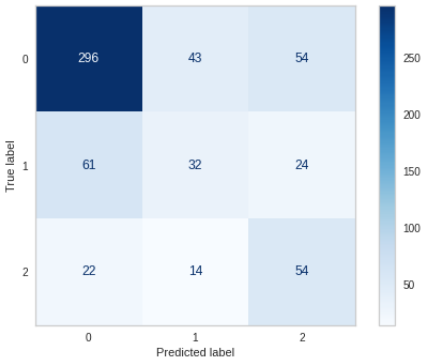

Fig. 41. Confusion Matrix for Stacking Classifier

summary of the balanced accuracy metric for all models, from which it is noticed that SVM is the model that performs best at classification, with almost 0.65 balanced accuracy score.

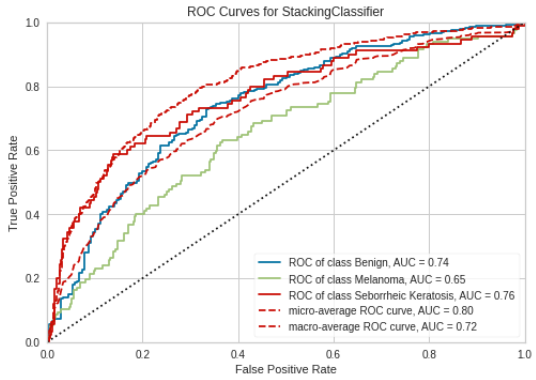

Fig. 42. ROC-AUC curve for Stacking Classifier

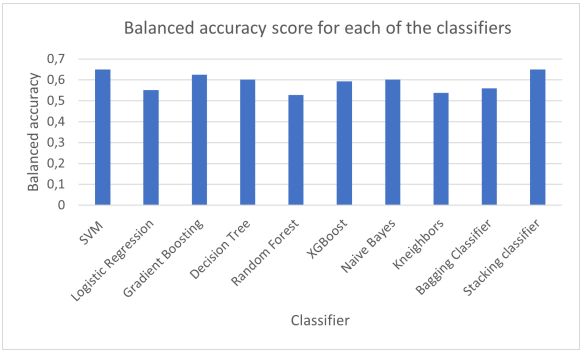

Fig. 43. Classifiers performance summary

V. Deep Learning Based Approaches for Segmentation and Classification

A. Segmentation

1) UNET architectures: For the segmentation part, UNET architecture proposed in [22] was used. The summary schematic of how UNET was used for this work is depicted in figure 44.

The UNET is a Deep Convolutional Neural Network that was initially designed for the segmentation of biomedical images. It consists of a contracting and an expansion path that together form the shape of an U, and that is the reason behind its name. The first path is a Convolutional Neural Network with successive convolutions followed by a nonlinear activation function (ReLU) and max pooling operations [22], as shown in Figure 44. It is called contracting because in each stage of the path the feature space gets reduced, since only the maximum response from every 2x2 sliding window is preserved. The goal of the expansion path is to create a higher resolution segmentation map, and get back to the original size of the image. This is done through successive transposed convolutions, also known as upconvolutions. This upsampling procedure makes the feature space bigger in every level of the path. In each stage of the expansion path the upsampled representations are concatenated with the high resolution features from its corresponding level in the contracting path. This concatenation is represented by the green arrow in Figure 44.

2) Model Implementation: For the model implementation, the dataset from ISIC-2017 [23] was used. It consists of 2000 and 600 images for training and test sets, respectively. Performance was measured through Jaccard index, defined in the section of the traditional image processing approach. For the model implementation, the following considerations were taken into account:

- First, the images are converted to square dimensions in order to be suitable for UNET. Two sizes were tested: 512x512 and 256x256. After running some experiments, it was found that using 256x256 as input images gave the highest results for the segmentation task.
- Two different types of optimizers were used: Stochastic Gradient Descent (SGD) and ADAM.
- Three different types of loss functions were taken into account: Binary Cross Entropy (BCE), Intersection over Union loss (IoU), and Dice-BCE loss. This latter is a combination of regular Dice and BCE loss.
- Six different combinations of optimizers and loss functions were used for the model, which leads to having 6 results to compare. These setups were: BCE loss + ADAM optimizer, IoU loss + ADAM optimizer, BCE loss + SGD optimizer, IoU loss + SGD optimizer, DICE-BCE loss + ADAM optimizer, DICE-BCE loss + SGD.
- The following parameters were tuned when training the model: learning rate, number of epochs, learning rate patience, early stopping patience, weight decay for the optimizer and momentum. The learning rate controls the speed at which the model learns. The number of epochs refers to an entire pass of the model through the training data. Learning rate patience acts as a scheduler parameter and it tells how many epochs to wait before decreasing the learning rate if there is no improvement in the performance of the model on the validation set. Early stop patience refers to the number of epochs to wait before early stop of the training there is no improvement in the performance on the validation sets. The weight decay is a regularization parameter in which a certain penalty is added to the loss function in order to shrink the weights during backpropagation. The values for these parameters that led to the best performance were 1e-05 for learning rate, 100 number of epochs, learning rate patience 5, early stopping patience 10, weight decay 1e-8 and momentum 0.9 which helps accelerate gradients vectors in the right directions, thus leading to faster converging.
- This is a binary classification problem for each of the pixels in the image, which means that they will be classified as either part of the foreground or the background, and this result will be given as a probability (floating point). Therefore, in order to convert this to a binary image a threshold is needed. The calculation of this threshold is done by Bayesian Optimization algorithm, which ensures that for this binary threshold the Jaccard index is maxi-

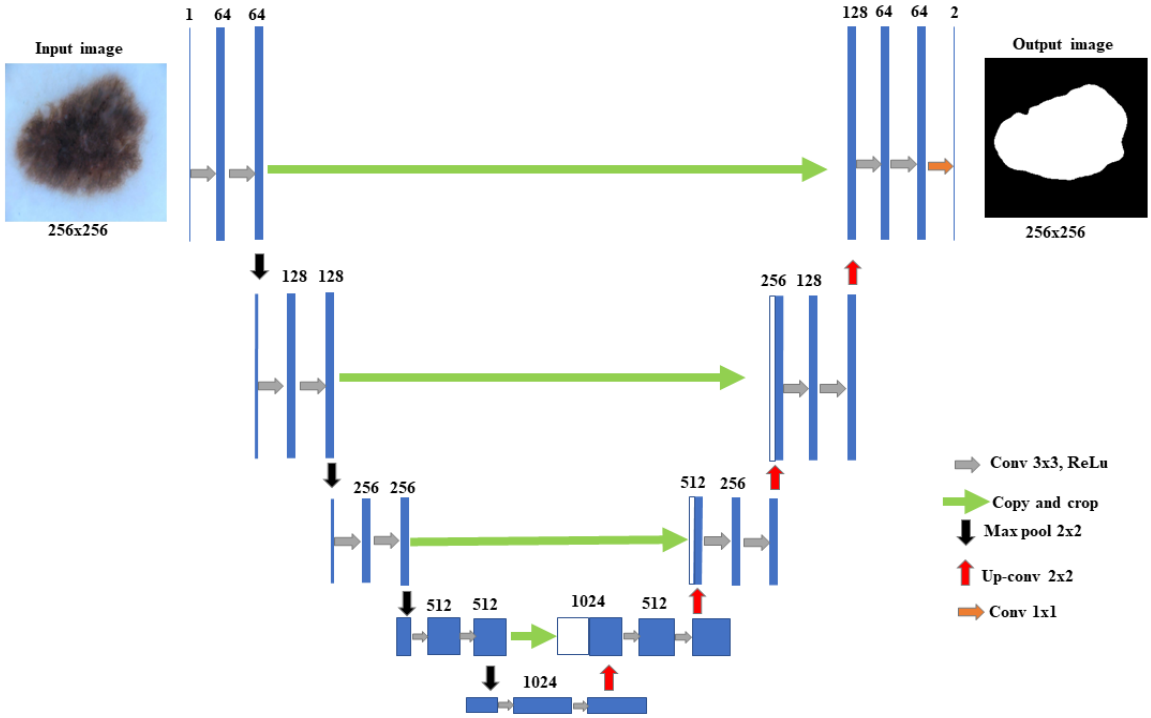

Fig. 44. Summary of UNET architecture scheme used for skin lesion segmentation task

mized. A number of 10 iterations are performed for this optimizer.

3) Results: As discussed before, to evaluate the performance of segmentation techniques, the Jaccard Score (JS) and Dice Score Coefficient (DSC) are considered. The model was trained with the 2000 images from ISIC-2017 Dataset and validated with 150 images. Finally the model's performance was tested on the totally unseen 600 images and measured the JC and DSC respectively. As, 6 different variant of UNET model was used for training with different loss functions and optimizers, they were results were compared to find the best model with all the hyperparameters. The hyperparameters were also tuned with the validation set images. The best results was achieved with the following hyperparameter set shown in Table I.

TABLE I
HYPERPARAMETER VALUES FOR SEGMENTATION TASK

Hyperparameters	Value
Image Dimension	256*256
Batch Size	10
Learning Rate	1.00E-05
Number Of Epochs	100
Leraning Rate Patience	5
Early Stopping Patience	10
Weight Decay	1.00E-08
Momentum	0.9

In terms of Loss Functions, both IOU Loss and Dice-BCE Loss outperformed BCE Loss. These two loss functions are popular for segmentation tasks. For Optimizers, Adam was superior to SGD. Overall, the model which yielded the best performance in terms of DSC and JS is the UNET model with Dice-BCE Loss and Adam Optimizer. IOU Loss combined with SGD and Adam achieved also very satisfactory results. The overall result comparison table is shown in the Table II.

TABLE II
COMPARISON TABLE FOR SEGMENTATION RESULTS

Model	Loss Function	Optimizer	Test Loss	Test Dice	Test Jaccard
UNET	BCE Loss	SGD	0.8359	0.431	0.4249
		Adam	0.8825	0.4175	0.4247
	IOU Loss	SGD	**0.5389**	**0.7461**	**0.6511**
		Adam	**0.5713**	**0.7575**	**0.6503**
	Dice-BCE Loss	SGD	0.5794	0.6272	0.5665
		Adam	**0.5052**	**0.7428**	**0.654**

The average jaccard score was computed over all the 600 test images both with resized image and masks and also with the original resolution. As the original image dimension was converted into squared dimension of size 256*256, the results or segmentation masks were also converted and upsampled back to the each respective sized as inputs. The average jaccard scores of **0.65** and **0.7** were obtained with the best model for these two cases (reduced and original dimension). A bar chart showing model vs average jaccard score on test data can be seen in Figure 46.

Finally, A set of example images and their respective segmentation results for the models are presented in the Figure 45. Clearly, Unet combined with Dice-BCE and Adam yielded

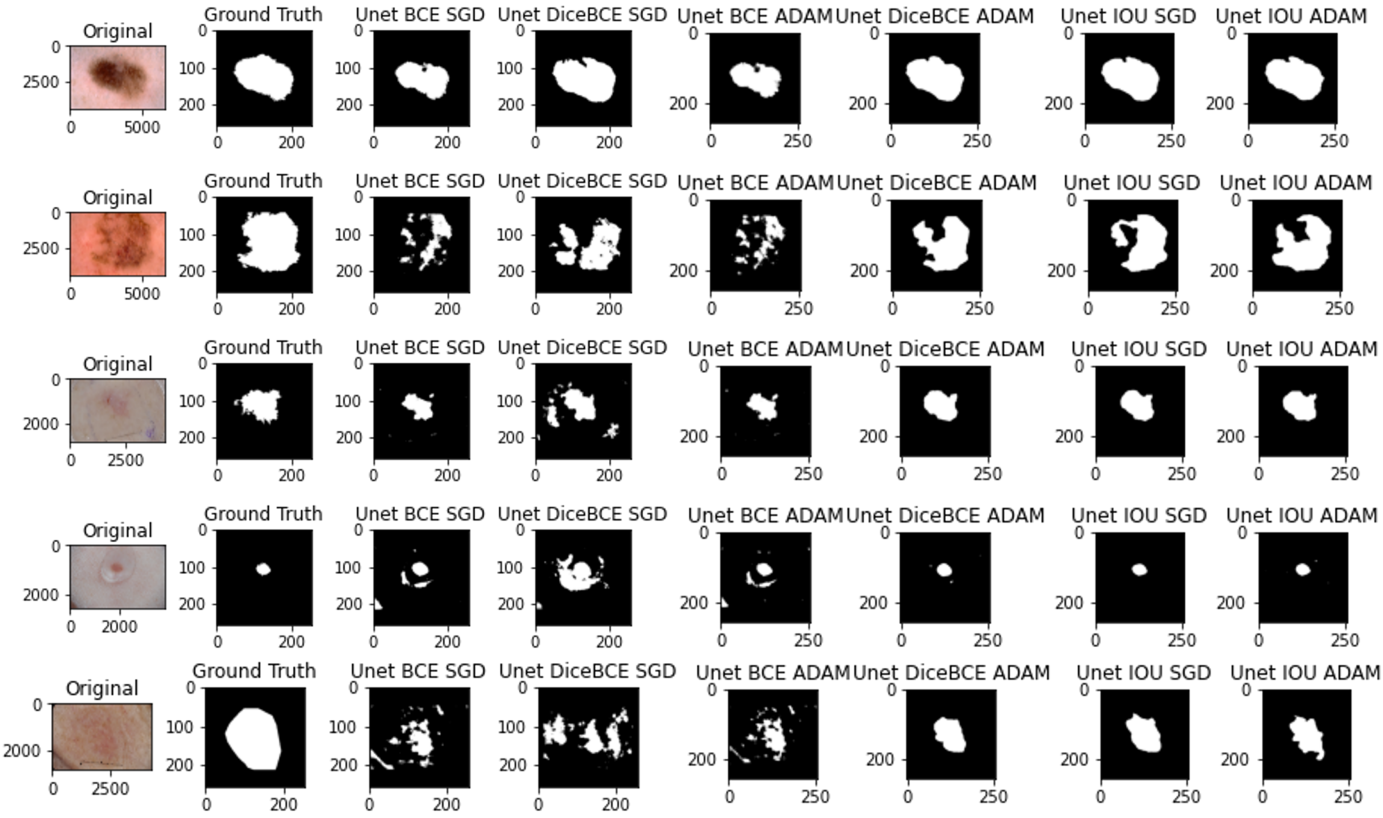

Fig. 45. Example Results for Segmentation

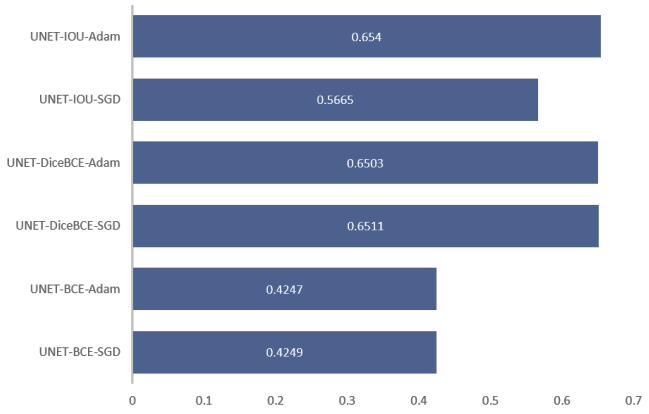

Fig. 46. Models vs Average Jaccard Score

the most satisfactory results.

B. Classification

1) Model architectures: Two different architectures were chosen to perform this classification: ResNet50 and VGG16. These two models are widely used for this task since they are very deep. They have more robust representation capabilities and a more significant number of parameters that make them better for image recognition, representing a higher classification accuracy.

Their architectures are described in the following section.

2) ResNet50 architecture: Residual networks (ResNet) are characterized because they use skip connections to reduce the vanishing gradient problem. This result is achieved by adding the original input to the following input of the next residual block, known as an identity block. A residual block is a stack of layers where the skip connections are performed. So, because of these skip connections, the value reached after executing backpropagation will not be that small. Hence, the overall accuracy obtained will be higher.

ResNet50 is a variant of the ResNet base model. The number "50" in its name indicates this network's number of layers. These layers are distributed in four different stages. The first stage receives an input image that has passed through the initial layer which contains convolution and max-pooling operations, with a stride=2 and kernels size 7x7, 3x3 respectively. This stage has three residual blocks with three convolutional layers each. The second, third, and fourth stages have four, six, and three residual blocks, respectively. The dimensions of the kernels of each convolutional layer of each block throughout the whole network are 1x1, 3x3, and 1x1, respectively.

Each time the input of each residual block enters a new stage, a convolutional operation with stride=2 is performed. Hence, the input size will be reduced to half, meaning that

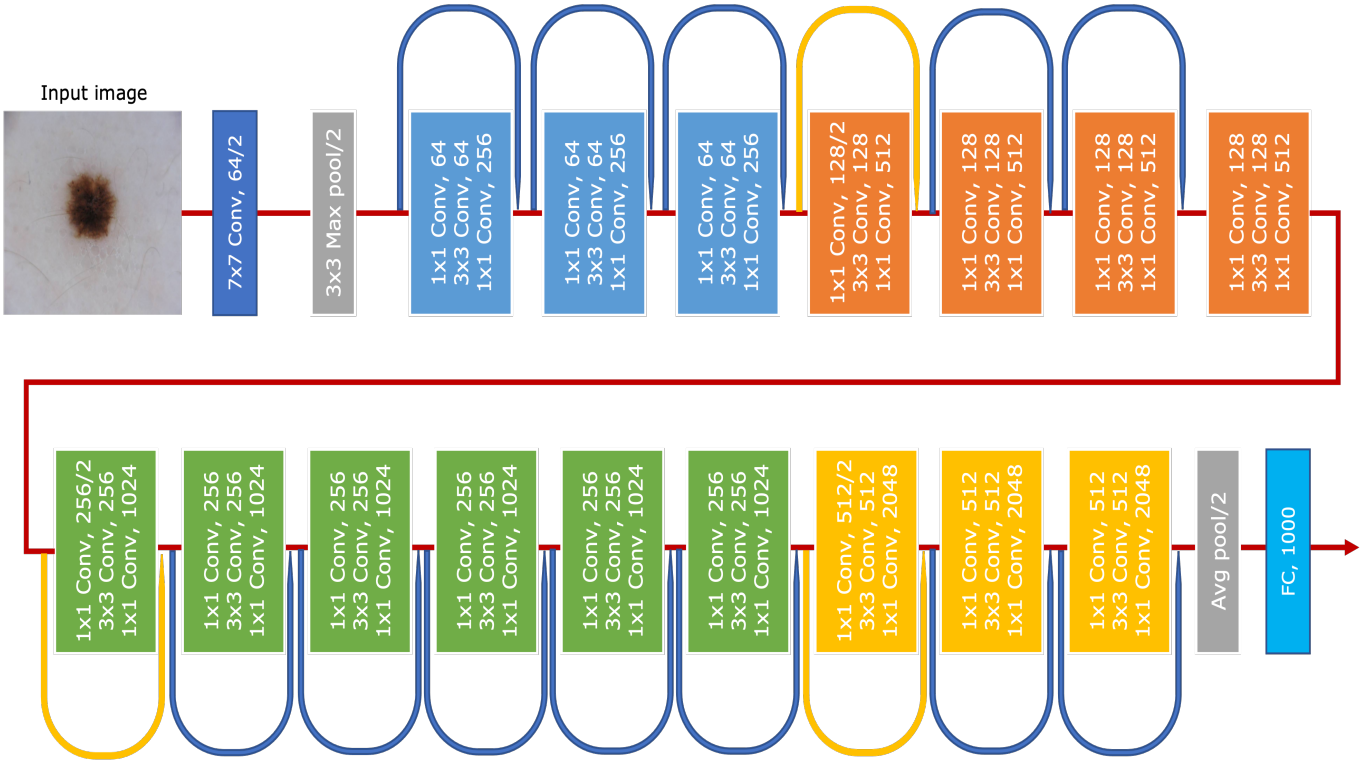

Fig. 47. ResNet50 architecture

the output size of this block will be different. To solve this issue, a convolutional operation is added to the identity block within the skip connection to restore the original size of the input. In this manner, the input size of the next residual block will be the same as the prior. Finally, at the end of these four stages, there is an averaging pooling operation followed by a fully connected layer that contains 1000 neurons. [24]

The architecture of this network is shown in Fig 47. The blue lines represent the skip connections with the identity block, whereas the yellow ones are skip connections with convolutional operations.

3) VGG16 architecture: As the ResNet50, the Visual Geometry Group (VGG)16 is an object detection 16 layers network mainly used for classification tasks. This network stands out for having a pyramidal form, with the top layers being deep and the lower layers, which are closer to the image, being wide. It consists of 21 layers in total—13 convolutional layers, 5 max-pooling layers, and 3 fully-connected layers—but only 16 are weight or learnable parameters layers. The most distinctive feature of VGG16 is that it prioritizes convolution layers of a 3x3 filter with stride one rather than a large number of hyper-parameters and consistently employs the same padding and max-pooling layer of a 2x2 filter with stride 2. These two types of layers are consistently arranged throughout the whole architecture.

The following stack of layers is the fully connected type, where two have 4096 channels, and the third one has 1000 since it performs 1000-way classifications. Finally, the last layer of this network is a SoftMax one. [25]

A visual representation of this architecture can be seen in Fig 48.

4) Implementation: Once defined the architecture to be is implemented, visualization of the dataset is an excellent step to start. By doing this, it can be seen that the dataset provided is highly unbalanced since it contains three types of skin lesions distributed in the following way: 1372 are labeled as benign, 374 as melanoma, and 254 as seborrheic keratosis, making a total of 2000 images. An unbalanced dataset creates a problem for the unbalanced class since optimized results cannot be reached because the algorithm never gets a sufficient look at it, resulting in a larger number of predictions for the majority class. To solve this issue, Data Augmentation is performed by creating some flipped horizontal and vertical copies of the images belonging to the unbalanced classes, resulting in a new dataset with 1122 melanoma and 762 seborrheic keratosis images and the same number of benign lesions.

Another step to deal with imbalance class problem is to assign weights to the classes. The primary purpose of this is to penalize the misclassification made by the minority class by giving a higher weight to it and reducing the assigned weight to the majority class. By doing this, the algorithm can focus on reducing errors in predicting the class with fewer samples which will help training with Focal Loss Paradigm.

Fig. 49 and Fig. 50 show the initial distribution of the classes and the resulting distribution after performing data augmentation, respectively.

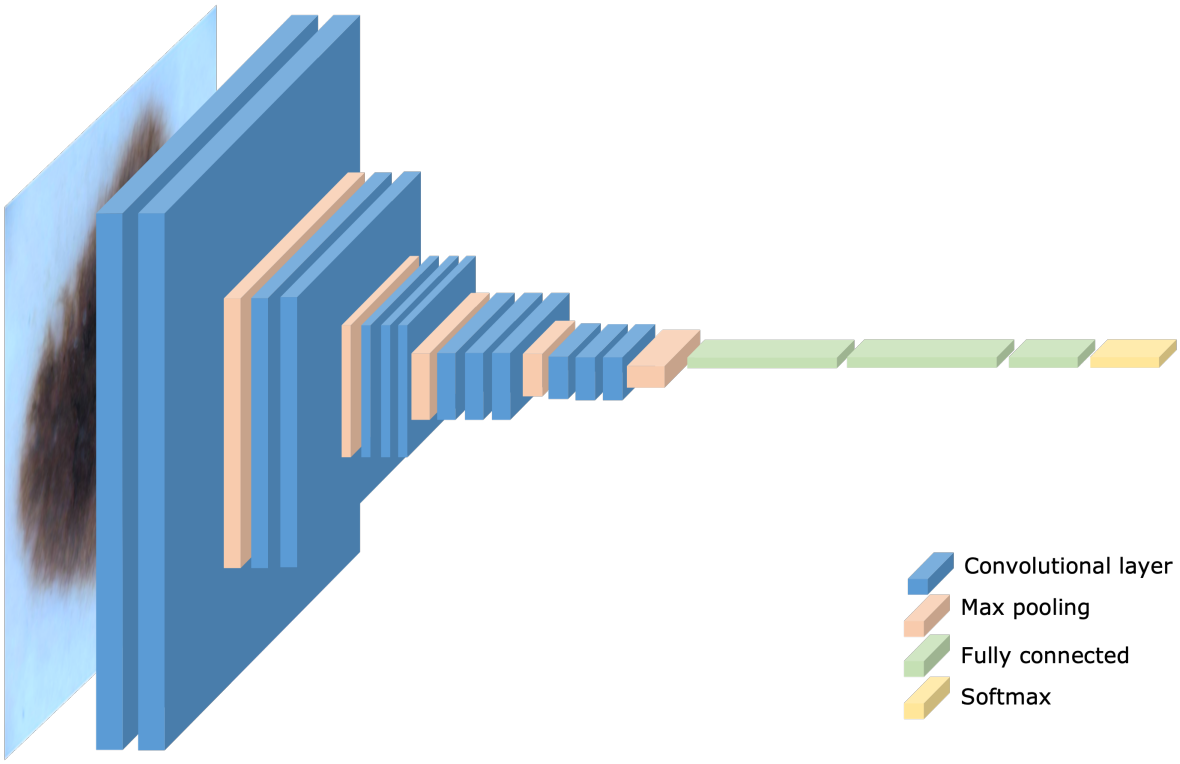

Fig. 48. VGG16 network architecture

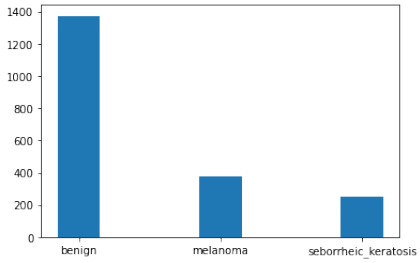

Fig. 49. Initial Training Data Distribution

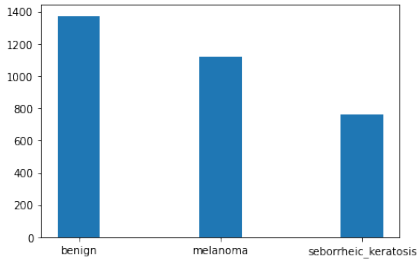

Fig. 50. Training Data Distribution After Data Augmentation

5) Transfer learning: Transfer learning is a technique for enhancing the effectiveness of our given task by using model weights trained on standard datasets like ImageNet, which contains more than a million images in 1000 different categories. By taking advantage of this technique, the base models for the architectures implemented in this project benefit by reducing the computational cost and processing training time.

6) Model optimization: The approach to optimizing the previously discussed architectures was to try a combination of models by changing their cost functions and optimizers. A cost function is an important parameter that defines how well a machine or deep learning model performs on a particular dataset. It calculates and expresses the difference between the expected and predicted values as a single real number. The two cost functions used in this project are Cross-Entropy (CE) and Focal Loss (FL). The first quantifies the difference in distance between probability distributions and penalize wrong predictions more than reward the right one. The second is an extension of CE that attempts to address the problem of class imbalance by giving harder or easily misclassified samples greater weight. Regarding optimizers, they aim to minimize the cost function by adjusting the model weights. The optimizers selected for this task are Stochastic Gradient Descent (SGD) and Adaptive Moment (ADAM) estimation.

On the one hand, SGD attempts to find the global minimum by adjusting the configuration of the network after each training point. Instead of decreasing the error or finding the gradient for the entire data set, this method merely reduces the error by approximating the gradient for a randomly selected batch (which may be as small as a single training sample).

On the other hand, the ADAM is an alternative optimization algorithm that provides more efficient network weights by

running repeated cycles of "adaptive moment estimation." ADAM extends SDG to solve non-convex problems faster while using fewer resources than other optimization programs. By doing this, eight different models are set to be trained:

- ResNet50, with CE and SGD
- ResNet50, with CE and ADAM
- ResNet50, with FL and SGD
- ResNet50, with FL and ADAM
- VGG16, with CE and SGD
- VGG16, with CE and ADAM
- VGG16, with FL and SGD
- VGG16, with FL and ADAM

It is important to mention that both architectures receive an input image with a 224x224x3 dimension. Also, the last layer of each architecture was replaced by the parameters trained on the ImageNet dataset using transfer learning, whereas the rest of the previous layers were frozen.

Finally, the model with the highest accuracy on the test set and the highest accuracy in predicting the melanoma class was selected as the best model.

7) Results: For Classification task, two important metrics considered to evaluate the performance of the classifier output were Model accuracy (balanced) and AUC (Area Under the curve) for each of the three classes (melanoma, benign and seborrheic-keratosis). Two different pre-trained models (ResNet50 and VGG16) were used combined with two types of loss functions (Cross-entropy loss and focal loss) and optimizers (SGD and Adam) as discussed above. AUC and Accuracy were measured for each of the considered scenarios and compared. Overall, with augmented data, 3256 images were used for the training, 150 images for validation and 600 images for testing. The hyperparameters were tuned using the validation dataset and the best tuned hyperparameters that were used for training can be seen in Table III.

TABLE III
HYPERPARAMETER VALUES FOR CLASSIFICATION TASK

Hyperparameters	**Value**
Image Dimension	224*224
Batch Size	10
Learning Rate	1.00E-05
Number of Epochs	15
Learning Rate Patience	2
beta1 (Adam)	0.9
beta2(Adam)	9.99E-01
Momentum	0.9
Weight Decay	1.00E-06

A comparison table is represented in the Table IV and V to better visualize and understand the performance of the each model. It can be seen that ResNet50 achieved overall better performance than VGG-16. Both focal loss and cross-entropy loss were considered but cross-entropy loss outformed as data augmentation solved the data imbalance problem. Adam and SGD both yielded satisfactory results combined with Resnet50 and cross-entropy loss. The overall balaced accuracy for the best model was **0.73** but the accuracy and auc value for the melanoma class was considered too to choose the best model as this is the main class to identify and main concern. ResNet50 achieved accuracy of **0.68** and auc of **0.572** for the melanoma class. A bar chart comparing the balanced accuracy for all the models can be found in the Figure 54. Also, the ROC curves for both pretained models Vgg16 and ResNet50 can be seen in the Figure 51 and 52. Clearly, the roc figure explains that resnet50 was superior to vgg16 in the classification task.

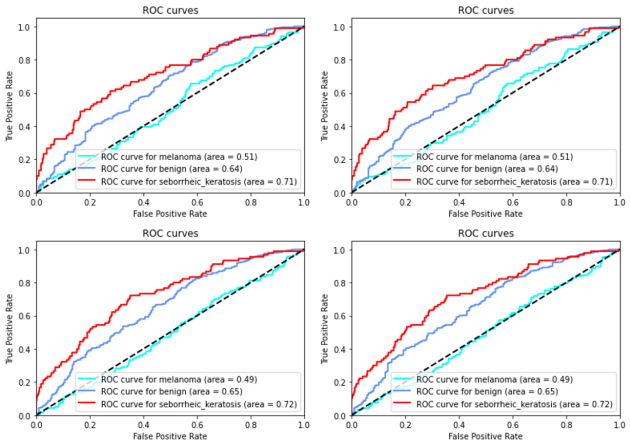

Fig. 51. ROC Curves (LT:VGG-CE-Adam, RT:VGG-CE-SGD, LB:VGG-FL-Adam, RB:VGG-FL-SGD)

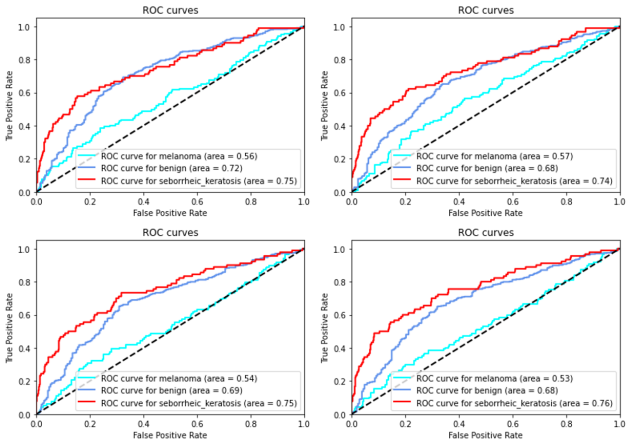

Fig. 52. ROC Curves (LT:ResNet-CE-Adam, RT:ResNet-CE-SGD, LB:ResNet-FL-Adam, RB:ResNet-FL-SGD)

A sample of example images can be seen in the Figure 53 with each predicted and actual classes for all the models.

VI. FUTURE WORKS AND CONCLUSION

Skin lesion segmentation and classification has become an unsolved challange for all the researchers as they have been trying to increase the effieciency as much as possible over the years. Variant features of lesion like uneven distribution of color, irregular shape, border and texture make this task challenging. In this work, traditional unsupervised and supervised

TABLE IV
ACCURACY COMPARISON TABLE FOR CLASSIFICATION TASK

Model	Loss Function	Optimizer	Accuracy(Melanoma)	Accuracy(Benign)	Accuracy(Seborrheic)	Overall Accuracy
VGG-16	Cross-Entropy	ADAM	0.41	0.65	0.66	0.6
		SGD	0.42	0.64	0.68	0.6
	Focal-Loss	ADAM	0.28	0.65	0.71	0.59
		SGD	0.38	0.65	0.69	0.6
ResNet50	Cross-Entropy	ADAM	**0.68**	**0.71**	**0.73**	**0.7**
		SGD	**0.6**	**0.78**	**0.71**	**0.73**
	Focal-Loss	ADAM	0.41	0.8	0.73	0.72
		SGD	0.56	0.74	0.74	0.7

TABLE V
AUC COMPARISON FOR CLASSIFICATION TASK

Model	Loss Function	Optimizer	AUC(Melanoma)	AUC(Benign)	AUC(Seborrheic)
VGG-16	Cross-Entropy	ADAM	0.509	0.642	0.709
		SGD	0.506	0.639	0.711
	Focal-Loss	ADAM	0.49	0.651	0.723
		SGD	0.38	0.65	0.69
ResNet50	Cross-Entropy	ADAM	**0.562**	**0.716**	**0.747**
		SGD	**0.572**	**0.68**	**0.739**
	Focal-Loss	ADAM	0.535	0.686	0.746
		SGD	0.534	0.685	0.757

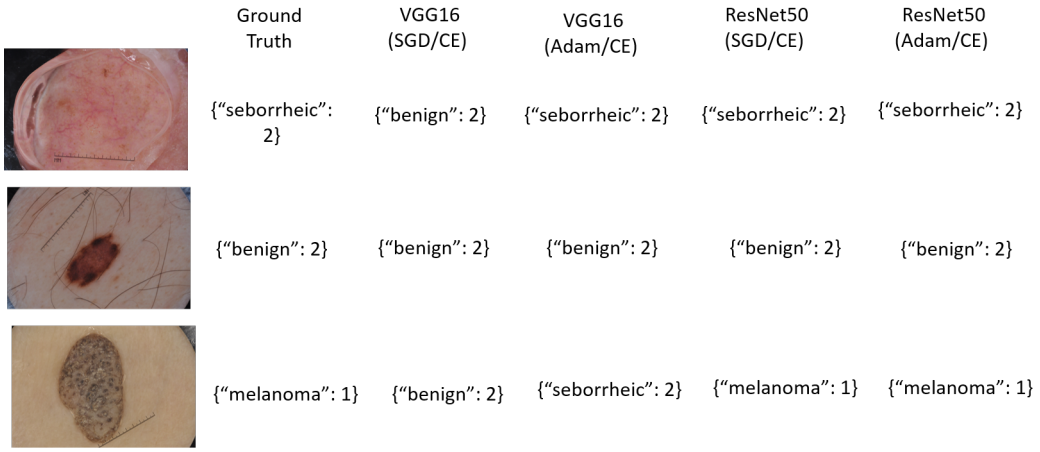

Fig. 53. Example Results for Classification

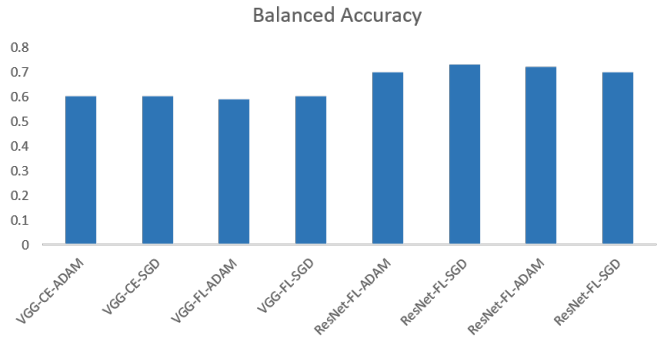

Fig. 54. Balanced Accuracy Comparison for Classification Models.

approaches are presented to analyse and compare the performance with the state of the art models. Though unsupervised approaches were faster in computation, supervised or deep learning based approaches took a longer time for training. For Segmentation, the results achieved with image analysis based techniques was satisfactory but it decreases with the increase of the data due to the present of various artifacts. So, to deal with them, deep learning based model Unet was applied for instantaneous based segmentation which achieved an improved accuracy of 64.5%. For Classification task, various features extracted for machine learning based classifiers and with the best selected features, SVM achieved highest accuracy among all the other 10 models studied. For Deep Learning based models, VGG16 and ResNet50, two pretrained models were applied with different tuning parameters but ResNet achieved the highest accuracy of 73% detecting 68% of the melanoma cases and treated as the best model for classification. Though the results achieved in both of the tasks were good but

still there is enough room for improvement. In future, the models can be trained with larger and more balanced dataset considering all the possible scenerio. Also there is still many models that can be analyzed specially for deep learning based classifiers to compare the performances. Mainly, the future tasks will comprise of trying to improve the jaccard score for segmentation and accuracy for classification in a faster approach without loosing much computational efficiency.

REFERENCES

[1] "Skin cancer facts & statistics, access: 22-june-2022." [Online]. Available: www.skincancer.org/skin-cancer-information/skin-cancer-facts

[2] N. C. F. Codella, D. A. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. K. Mishra, H. Kittler, and A. Halpern, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (ISIC)," *CoRR*, vol. abs/1710.05006, 2017.

[3] N. Hameed, A. M. Shabut, M. K. Ghosh, and M. A. Hossain, "Multi-class multi-level classification algorithm for skin lesions classification using machine learning techniques," *Expert Systems with Applications*, vol. 141, p. 112961, 2020.

[4] M. Hasan, B. Alyafi, F. I. Tushar *et al.*, "Comparative analysis of automatic skin lesion segmentation with two different implementations," *arXiv preprint arXiv:1904.03075*, 2019.

[5] K. L. Mon, "Automatic image segmentation using edge and marker controlled watershed transformation," Ph.D. dissertation, MERAL Portal, 2014.

[6] M. Q. Hatem, "Skin lesion classification system using a k-nearest neighbor algorithm," *Visual Computing for Industry, Biomedicine, and Art*, vol. 5, no. 1, pp. 1–10, 2022.

[7] J. A. Salido and C. Ruiz Jr, "Hair artifact removal and skin lesion segmentation of dermoscopy images," *Asian Journal of Pharmaceutical and Clinical Research*, vol. 11, no. 3, 2018.

[8] R. Melli, C. Grana, and R. Cucchiara, "Comparison of color clustering algorithms for segmentation of dermatological images," in *Medical Imaging 2006: Image Processing*, vol. 6144. SPIE, 2006, pp. 1211–1219.

[9] G. Capdehourat, A. Corez, A. Bazzano, and P. Musé, "Pigmented skin lesions classification using dermatoscopic images," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2009, pp. 537–544.

[10] R. C. Hardie, R. Ali, M. S. De Silva, and T. M. Kebede, "Skin lesion segmentation and classification for isic 2018 using traditional classifiers with hand-crafted features," *arXiv preprint arXiv:1807.07001*, 2018.

[11] M. A. Kassem, K. M. Hosny, and M. M. Fouad, "Skin lesions classification into eight classes for isic 2019 using deep convolutional neural network and transfer learning," *IEEE Access*, vol. 8, pp. 114 822–114 832, 2020.

[12] R. Sumithra, M. Suhil, and D. Guru, "Segmentation and classification of skin lesions for disease diagnosis," *Procedia Computer Science*, vol. 45, pp. 76–85, 2015.

[13] M. K. Hasan, L. Dahal, P. N. Samarakoon, F. I. Tushar, and R. Martí, "Dsnet: Automatic dermoscopic skin lesion segmentation," *Computers in biology and medicine*, vol. 120, p. 103738, 2020.

[14] Z. A. Nazi and T. A. Abir, "Automatic skin lesion segmentation and melanoma detection: Transfer learning approach with u-net and dcnn-svm," in *Proceedings of international joint conference on computational intelligence*. Springer, 2020, pp. 371–381.

[15] A. Mahbod, G. Schaefer, C. Wang, R. Ecker, and I. Ellinge, "Skin lesion classification using hybrid deep neural networks," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1229–1233.

[16] B. S. Lin, K. Michael, S. Kalra, and H. Tizhoosh, "Skin lesion segmentation: U-nets versus clustering," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–7.

[17] V. Miglani and M. Bhatia, "Skin lesion classification: A transfer learning approach using efficientnets," in *International Conference on Advanced Machine Learning Technologies and Applications*. Springer, 2020, pp. 315–324.

[18] M. Hashemzadeh and B. A. Azar, "Retinal blood vessel extraction employing effective image features and combination of supervised and unsupervised machine learning methods," *Artificial intelligence in medicine*, vol. 95, pp. 1–15, 2019.

[19] A. Telea, "An image inpainting technique based on the fast marching method," *Journal of graphics tools*, vol. 9, no. 1, pp. 23–34, 2004.

[20] "What to look for: Abcdes of melanoma, access: 22-june-2022." [Online]. Available: https://www.aad.org/diseases/skin-cancer/abcde-of-melanoma

[21] Z. She, Y. Liu, and A. Damatoa, "Combination of features from skin pattern and abcd analysis for lesion classification," *Skin Research and Technology*, vol. 13, no. 1, pp. 25–33, 2007.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[23] N. C. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler *et al.*, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic)," in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 2018, pp. 168–172.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.