Melanoma Skin Cancer Detection using Deep Neural Networks

Diogo F. Ferreira
Department of Electronics,
Telecommunications and Informatics
University of Aveiro
pdiogoferreira@ua.pt

Pedro B. Martins
Department of Electronics,
Telecommunications and Informatics
University of Aveiro
pbmartins@ua.pt

***Abstract*— Over the last few years, Deep Learning has been on the spotlight when it comes to solving many complex problems, namely Convolutional Neural Networks (CNN) in the field of image recognition. Since this Neural Networks can sometimes achieve more than 50 layers deep, Transfer Learning (TL) techniques are also getting popular to retrain some of the layers of Neural Networks that learned from a generic dataset, putting it to use in another case.**

In this work, we attempt using CNNs and TL tecnhiques on pre-trained Neural Networks to distinguish melanoma skin cancer from seborrheic keratosis and nevus benign tumors.

I. INTRODUCTION

The skin is the first line of defense of the human body against outer particles. It produces the melanin pigment, which provides our bodies with a shield against ultraviolet radiation. This pigment, as well as the underlying blood vasculature, is one of the biomarkers in the detection and follow up on melanoma skin cancer. Melanoma is considered one of the deadliest types of skin cancer, usually appearing due to DNA damage resulting from the sun ultraviolet radiation over-exposure.

According to the Skin Cancer Foundation, about 9,320 people are victims of melanoma in the US annually, and about 178,560 new cases of melanoma were initially predicted in 2018 alone [1]. While in its earliest stages, if correctly detected, melanoma is treatable, otherwise, it can spread to different parts of the body and become fatal. The accuracy of the methods currently used by the majority of dermatologists sits in about 75 to 85% [2], which can be considered a good value since most body marks, such as moles, brown spots and growths, are usually harmless and difficult to distinguish from melanoma. Benign or malignant pigments are usually told apart using the ABCDE signs of melanoma, standing for Asymmetry, Border irregularity, Color, Diameter and Evolving.

Although the ABCDE rule has proven to be very helpful to dermatologists over the years, the detection process usually takes some time and requires a high-skilled practitioner. Besides that, the rule makes it difficult to recognize birth marks from melanoma moles or even detect early stage small melanoma marks.

The first attempt to solve this problem was through image processing, extracting statistical features such as pixel density mean and standard deviation, and feeding those values to common shallow Neural Networks or SVMs for classification. However, with the recent trend of Deep Neural Networks (DNN), the use of Convolutional Neural Networks (CNN) might be very useful, since they are capable of learning by itself the details common image processing provides, saving us the trouble of feature engineering.

An even thorough approximation to solve these type of problems is to use Transfer Learning (TL), a technique which consists in adapting the data already stored and used to solve a given situation and using it to solve a related problem. An example is changing the last layers of a DNN trained for a certain scenario and retrain them, effectively updating their weights, according to the new scenario.

This paper focuses on using DNN architectures such as CNN and TL techniques to detect melanoma skin cancer through the analysis of dermatoscope images. The dataset used was provided by the International Skin Imaging Collaboration (ISIC)[3] while organizing a three part competition on melanoma identification in 2017. The dataset contains about 2,000 images, which 374 classified as ”melanoma” (malignant skin tumor, derived from melanocytes), 254 as ”seborrheic keratosis” (benign skin tumor, derived from melanocytes) and the remainder 1372 as ”nevus” (benign skin tumor, derived from keratinocytes). Besides the provided training dataset, ISIC also provided a validation and a test dataset, with 150 and 600 images, respectively. The original challenge was split into two classifications, *melanoma vs. all* and *seborrheic vs. all*. However, this paper will only focus on the detection of the malignant tumor, melanoma.

II. RELATED WORK

In recent years, Deep Learning has proved to be very effective in the classification and segmentation of medical imaging. There is an incredible amount of examples, such as mass detection using mammogram imaging [4] or even Alzheimer's disease detection [5].

A. Convolutional Neural Networks

CNNs are also a hot topic when it comes to medical imaging. They are used in a variety of fields, such as dermatology. Regarding melanoma detection, Nasr-Esfahani et al. obtained promising results using a dataset provided by Department of Dermatology of the University Medical Center Groningen (UMCG) [6]. The proposed solution started by a pre-processing step, in which the input images suffered an illumination correction and were the target of a Gaussian Filter in charge of smoothing the area outside

the lesion, as that information was considered irrelevant for melanoma detection. In order to expand the original dataset, each sample generated 36 others, resulting from the different combinations of cropping operations (different percentages of the image - 5 and 10% - and on different sides - top-left, top-right, etc.) and rotations by 0, 90, 180 and 270 degrees. Finally, the augmented dataset samples were resized to 188x188 pixels and fed to a 7-layer deep CNN. Besides the obvious input layer, the DNN was composed by two 5x5 convolutional layers and two max pooling layers, the first with a pool size of 4x4 and the second of 6x6, followed by two fully connected layers, the last corresponding to the output layer. The results were pretty satisfactory, as the model was able to score about 80% in test accuracy, sensitivity and specificity.

Aya Abu Ali and Hasan Al-Marzouqi [2] showed that, while developing a fairly simple CNN architecture, a well trained model is also capable of producing very accurate predictions of melanoma skin cancer. Rather than resizing and mean normalization, melanoma images are classified without applying lesion segmentation or complex image pre-processing. The developed model consisted of 17 layer CNN, based on 5 blocks built of a convolutional, a ReLU activation, pooling and dropout layers, although the final block replaced the last two components of the block by a fully connected layer and a softmax layer for the final classification. The first three convolutional layers were applying 5x5 filters, the fourth works with a 4x4 filter and the last with a 1x1 filter, all with a stride value equal to 2. The first worked with 32 kernels (filters), while the following three worked with 64, and the final with a mere 2. The pooling layers were defined with stride 4, while the dropout layers were configured with 0.1, 0.2, 0.3 and 0.5, respectively. After careful tuning of the considered hyperparameters - the batch size, which is the number of training images in one forward pass or backward pass, the number of epochs which is one forward and backward pass of all the training examples, and the learning rate - the model was able to score a testing error of 0.189, accompanied with a sensitivity value of 14.86% and a specificity of 98%. The authors mentioned that, accuracy wise, their model performed very similarly to the winner of the 2016 edition of the challenge, and that both accuracy and sensitivity were likely to increase if lesion segmentation was applied. One of the architectures used in the work presented on this paper is actually similar to the one Aya Abu Ali and Hasan Al-Marzouqi developed, as the Figure 1 shows.

The winners of the 2017 edition followed a different path than the majority of the participants. Kazuhisa Matsunaga et. al [7] developed a fairly complex solution by initially creating two independent classifiers, melanoma vs. all and seborrheic vs. all. After applying the luminance and color normalization to every sample of the dataset, those images are subjected to a combination of rotation, translation, scaling and flipping, and fed to an ensable of CNNs, consisting of a modified 50-layer ResNet, to be classified. Since both seborrheic and melanoma are generally rare at young ages, age and gender information is checked before outputting the

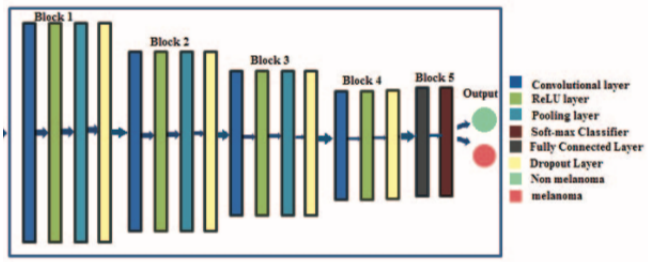


Fig. 1: CNN architecture used by Aya Abu Ali and Hasan Al-Marzouqi [2].

final result. This strategy was only applied on the seborrheic keratosis vs. all classifier, since no perceptive improvement was observed during their cross validation tests. They also noted their seborrheic keratosis vs. all classifier to be more reliable than the melanoma vs. all classifier, thus, if a sample had a very high hypothesis of being a seborrheic keratosis, then it is almost certain that it is not a melanoma. This was also taken into account before the final melanoma classifier output, as the Figure 2 depicts.

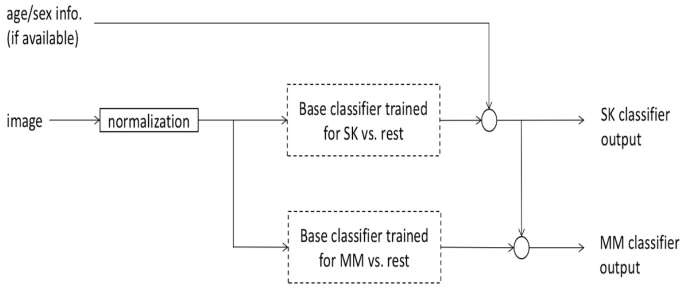


Fig. 2: Ensemble architecture of the 2017 ISIC winner [7].

Inspite of not showcasing their exact architecture, Haenssle et al. [8] also developed a CNN model to detect melanoma and put it to the test comparing its predictions against 58 dermatologists, and ended up concluding that their model was able to predict highly accurate diagnostics, sometimes even more accurate than a number of teams of experts in the field.

B. Transfer Learning

Transfer Learning (TL) techniques show that Deep Neural Networks (DNN) modeled and trained with sufficiently generic data, such as general image classification, are able to be modified to perform more specific tasks, like car models classification, whilst still retaining good performance.

Dan Cires et. al [9] performed an analysis on TL with DNN applied to multiple character recognition tasks. The objective of this work was to prove that is possible to transfer knowledge from a more clear and general task, to other much more specific and difficult to solve. The first approach was to train a CNN with (latin) digits and, ultimately, to classify latin uppercase characters. Having that in mind, in order to generate both the training set with digits, used as the initial

CNN training set, and the uppercase characters set, later used to re-train the network, some simple pre-processing was taken upon the NIST SD 19 dataset [10]. As the author states, it was possible to infer that "transferring the weights learned on the digit task to the uppercase letter task yields good results even if only the last two fully connected layers are retrained. In addition, learning from pretrained nets is much faster than learning from randomly initialized nets."

Following the same strategy, Qinghua Hu et. al [11] also showed that is possible to predict wind speed patterns by using TL techiques on a DNN trained over data-rich farms and then tune the mapping with data coming from newly-built farms. However, compared to the work mentioned before, it was used a Stacked Denoising Autoencoders (SDA) and Shared-hidden-layer DNN (SHL-DNN) instead of CNNs.

Hong-Wei Ng et. al [12] tried to perform emotion recognition by using a CNN architecture, trained over a big amount of facial expression images. The last two approaches in the previous paragraphs were different when it comes to the DNN architectures used to transfer learning. Instead of a "home made" architecture, Hong-Wei Ng et. al used two pre-trained and well known CNN architectures, AlexNet [13] and VGG-CNN-M-2048 [14]. In comparison to commonly used models, this approach was able to score a 16% better accuracy.

Also in the context of Transfer Learning, the ImageNet project [15] is a large visual database designed to be used in visual object recognition software research. This project runs an annual software contest, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), where software programs compete to correctly classify and detect objects and scenes. This project opened the path to the rise of some really good performant architectures, thus ending up receiving much support from the academic community. One of those is VGGNet [16], developed by Simonyan and Zisserman, a very uniform architecture composed by 16 convolutional layers (described in Figure 3) , a characteristic which made it so appealing to be used. It took almost 3 weeks to complete its training in 4 GPUs.

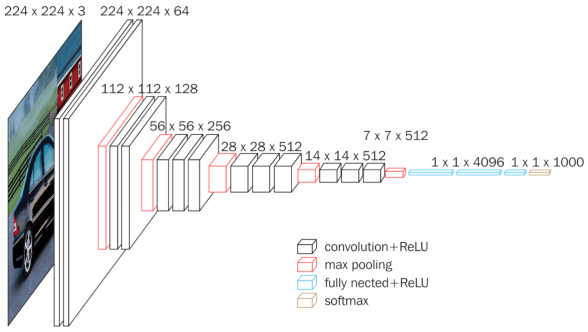


Fig. 3: VGG16 architecture

Another example derived from the ImageNet Project is the Residual Neural Network (ResNet) [17], developed by Kaiming He et al, introducing a novel architecture, based on *skip connections* and heavy batch normalization of features. The mentioned *skip connections* are also known as gated units or gated recurrent units, and have a strong similarity to recent successful elements applied in Recurrent Neural Networks (RNN). Due to this technique, they were able to train a 152 layer NN with a lower complexity than VGGNet. A top-5 error rate of 3.57% was achieved, beating human-level performance on this dataset. These big and complex Deep Neural Networks can be used to perform melanoma detection by applying Transfer Learning techniques.

III. METHODOLOGY

This paper intends to display two solutions to solving melanoma detection on the 2017 ISIC dataset. In this section, both of the proposed solutions and the preprocessing steps performed over the original data are detailed. The first solution consists of a DNN trained from scratch, namely a CNN, while the second uses TL techniques to apply ResNet and VGG19 pre-trained architectures in the resolution of this specific task. Both architectures were built and tested using the Keras framework.

Besides the different tecnhiques, two approaches for each one were taken. The first regards a multiclass classifier, in which the three classes (melanoma, seborrheic keratosis and nevus) were explicitly separated during the training process and, when applied to the test set, the last two classes were combined in a non-melanoma class. The second was merely a binary classifier, i.e., the dataset was already split in two classes, melanoma and non-melanoma, and the models learned to classify those two.

A. *Image pre-processing*

Since the dataset is highly inconsistent, due to microscopic borders, human hair, band aids and multiple other disturbances, all the images were first manually pre-processed, as they were cropped and resized to get rid of those nuisances that would most certainly unbalance the model while training, since it would most likely start to detect shapes, edges and colors not belonging to the lesion itself, but to these outliers. Given that the models input images were generally much smaller than the originally provided by the dataset, the cropping and resizing process did not cause problems.

Another step that was taken into account was to equalize the histograms of every image to improve image constrast and make the various skin tones look similar, making it easier to the model to detect the features relative to the multiple skin marks. However, this process can sometimes improve the constrast but not equalize the skin tones at all due to the original image quality, as the Figure 5 shows.

B. *Data augmentation*

Since the provided dataset size has proven to be insufficent, data augmentation techiques were applied in order to increase the number of training and validation examples. The orginal shape of the skin tumors are critically important, so the augmentation techniques should not be content destructive. To do so, only methods that keep the samples structure were applied, namely vertical and horizontal flips

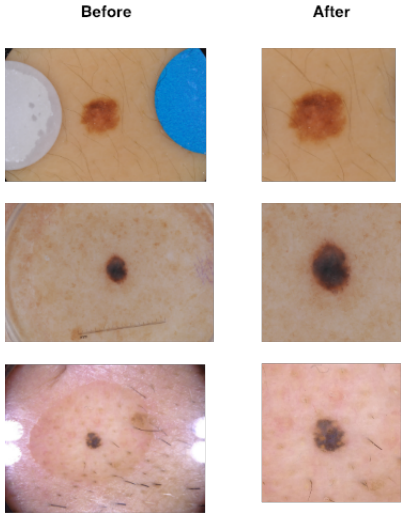


Fig. 4: Cropping and resizing of samples, focusing on the skin lesion.

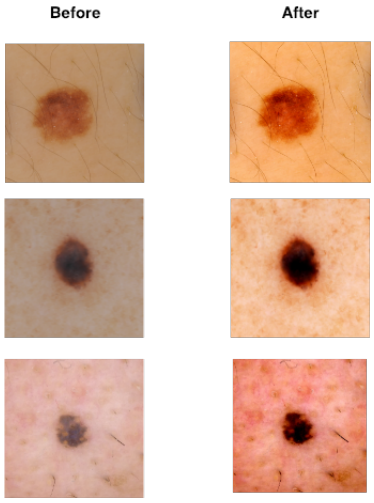


Fig. 5: Application of histogram equalization to the cropped samples.

and rotations with different angles. These transformations were randomly applied during the training process to the 2000 samples, resulting in about the quadruple of the original size, i.e., in about 8000 images.

Furthermore, since the number of melanoma samples is about 27.2% of the total amount and the seborrheic keratosis is about 18.5%, the training set had to be equalized. In an attempt to solve this issue was to feed the DNN with class weights. For example, considering 2 classes, the first with weight 1 and the second with weight 10. During the training process, the DNN will treat every sample of the class 2 as 10 samples, thus somewhat equalizing the dataset and forcing the model to become unbiased.

C. Convolutional Neural Network

As already mentioned, the first approach to solving this problem was to develop a CNN to effectively perform melanoma skin cancer detection. Even though a batch of architectures was developed, the proposed one lays on a DNN of 22 layers. It is important to notice that, regarding CNNs, literature often considers sequential convolutional, activation, pooling and dropout layers a single layer. However, regarding this architecture, they are considered independent layers.

The first layer receives a matrix with the dimension of 128 x 128 x 3, corresponding to RGB images of 128 by 128 pixels, and simply relies on an activation layer that normalizes the input data. The second layer is a convolutional layer that applies a filter of 5 x 5 to the input data and extracts 32 dimensions using the ReLU activation layer, while the fourth is a pooling layer using the maximum value of each 4 x 4 pool (MaxPooling), thus reducing the dimensions of the input data (in this case, four times smaller). The fifth layer is a dropout layer, which randomly discards about 20 percent of the neurons output to prevent overfitting. The sixth, seventh, eigth and nineth layers are somewhat similar to the previous, even though the convolutional layer uses now a 3 x 3 filter and the pooling layer uses a 2 x 2 pool. The tenth, eleventh, twelfth and thirteenth layers are equal to the last four. It is important to notice that, after each activation layer, the output data is normalized. After these layers, the data is flattened, following two batches of layers, made of a fully connected with ReLU activation layer and a 50% dropout layer, the first with 4096 neurons and the second with 1024. The last layer is also a fully connected layer, that uses Softmax activation, composed by 3 neurons, each corresponding to one of the classes the model is trying to predict: melanoma, nevus and seborrheic. The following figure depicts this architecture.

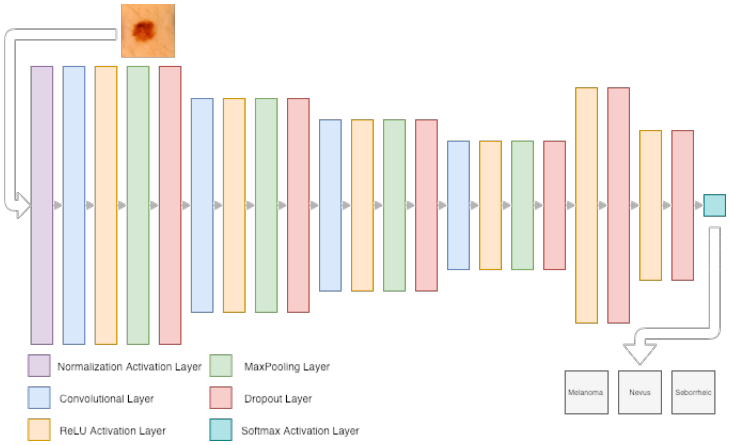


Fig. 6: Deep Convolutional Neural Network architecture.

D. Transfer Learning

On the context of TL applied to skin cancer classification, three main architectures were used - InceptionV3, ResNet50 and VGG19 - that have proved to be able to attain satisfactory results when it comes to image. The VGG19 network is 19 layers deep, while the ResNet50 is 50 layers deep and InceptionV3 is 48 layers deep, and all can classify images into 1000 object categories, such as balls, cars and variety of animals. VGG19 and ResNet50 demand an image input size

of 224 by 224 pixels, while the InceptionV3 architecture use an image input size of 299 x 299 pixels.

In order to perform melanoma detection, the hidden layers of the mentioned pre-trained architectures were frozen, so that their weights would not change during training. On the other hand, the output layer was changed to a densely-connected pooling layer with 3 or 2 units (regarding the different approaches), corresponding to the given context classes.

After the three above mentioned architetures were re-trained, various ensemble models were built regarding these examples. This type of approach can produce better results by taking into account weaker models. The goal was achieved by creating combinations between them (InceptionV3+VGG19+ResNet, InceptionV3+VGG19, ...), and averaging the values of the respective output layers.

E. Testbed

All the tests regarding the pure CNN approach were run in a Virtual Machine powered by a 6-core Intel Xeon X5670 clocked at 2.93GHz powered by 6GB of RAM. On the other hand, the TL tests were executed in a Virtual Machine powered by a 16-core Intel Xeon e5-2620 clocked at 2.10GHz running on 30GB de RAM. Even though the computational resources seem enough, it is important to notice that Deep Learning solutions are often executed on GPUs, running only on CPU-powered machines, which takes much more time.

IV. RESULTS

As already mentioned in the Methodology section of this paper, two different architectural approaches to detect melanoma skin cancer were taken, as well as two different ways of classification.

A. Implementation Details

Although it is often considered a hyperparameter, the batch size will be fixed at 32 samples, since the machine where the training will be running can only bear that much, due to memory limitations. The number of steps per epoch, i.e., the number of batches to be forward and backward passed into the Neural Network each epoch, will be four times the total number of samples on the training set divided by the batch size. Using Keras `ImageDataGenerator`, this will result in the already mentioned 8000 images used in the training of the CNN, due to randomly applied vertical and horizontal flips, rotations up to 40 degrees, and zoom up to 20%. The number of epochs considered was always 50, due to time constraints.

B. Multiclass approach

The first classification approach was, as mentioned, a multiclass classifier which would output one of three labels regarding melanoma, seborrheic keratosis or nevus.

1) Convolutional Neural Network: Although a variety of CNN architectures was tested, only a subset of those will be showcased in this section. Initially, it was tested some NN where it would vary the number of layers, the size and number of convolutional filters, the pool size on pooling layers, the dropout amount, and the number of output neurons on fully connected layers. It is important to notice that, due to time restraints, only very few combinations of these parameters were changed. As the architecture already depicted in Figure 6 has proven to obtain the best results, that is the one that went thorugh the majority of the tests.

As Table I shows, the results were somewhat disappointing, even though it is very hard to distinguish samples between these 3 types of skin marks, due to some similarities. Although the unbalanced dataset scored a higher accuracy on both validation and test scores, it performed worse than the balanced dataset on the test set across all metrics. Even though the training set accuracies are relatively higher than on the cross validation and test sets, indicating a possible overfit, the dropout values were already relatively high (e.g., 0.5 on the fully-connected layers) and, generally, it is not advised to cross the 0.5 threshold.

Obviously, this classifier would perform worse than a binary classifier, but the ultimate goal was to gather the seborrheic and nevus predicted labels into just one class, somewhat like an ensemble classifier. Even though seborrheic keratosis and nevus classes are joined into a single one, this is only applied in the test set. Accuracy wise, there was a considerable improvement (about 10%), however, both recall and precision dropped to nearly zero, indicating that this was definitely not a good approach.

Since this take on the melanoma detection problem was not efficient, it was not performed any hyperparameter tuning. The model was training using **Categorical Crossentropy** loss function with an **Adam** optimizer and a learning rate of 0.001.

2) Transfer Learning: The models used in TL were previously trained, so the number of parameters for fine-tuning is much smaller, when compared to an architecture created from scratch. Similar to the approach taken with CNNs, some hyperparameters variations were also tested, however, it showed that it did not produce a significant variation on the results. Taking this into account, the presented outcome will compare the differences between models trained with a balanced or an unbalanced dataset. The training process was executed using a **Categorical Crossentropy** loss function with a **Stochastic Gradient Descent (SGD)** optimizer with a Nesterov momentum and a learning rate of 0.01. The output layer was built with a **softmax** activation function and 40% for dropout amount.

In Table II, it is possible to find that, even though the ResNet50 with a balanced datset outperformed all others in the training data, it was VGG19 model trained with the same dataset that scored the best results in the test set metrics, except when regarding accuracy.

TABLE I: CNN Multiclass Results

Architecture	Training Set	Validation Set	Test Set			
	Accuracy	Accuracy	Accuracy	Precision	Recall	F1Score
CNN - Unbalanced Dataset	**0.77**	**0.64**	0.57	0.43	0.55	0.49
CNN - Balanced Dataset	0.73	0.59	**0.61**	**0.45**	**0.61**	**0.51**

TABLE II: TN Multiclass Results

Architecture	Training Set	Validation Set	Test Set			
	Accuracy	Accuracy	Accuracy	Precision	Recall	F1Score
VGG19 - Unbalanced Dataset	0.46	0.60	0.58	0.11	0.21	0.15
ResNet50 - Unbalanced Dataset	0.79	0.41	**0.68**	0.03	0.17	0.05
VGG19 - Balanced Dataset	0.68	**0.68**	0.58	**0.16**	**0.28**	**0.20**
ResNet50 - Balanced Dataset	**0.83**	0.55	0.63	0.02	0.13	0.04

C. *Binary approach*

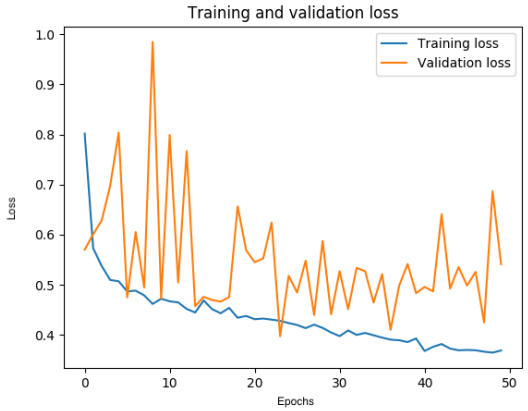
Due to the poor results obtained on the last approach, a new one, simply considering two classes (melanoma vs non-melanoma) right off the bat to be persued. Regarding the implementation, this approach follows the same lines as the multiclass one, since the important detail to consider is that the dataset is now split into only two classes, rather than the initial three. It is important to enhance, in this case, the positive label regards a non-melanoma situation, so the models are actually performing **non-melanoma detection**, due to Keras own interpretation of the dataset.

1) Convolutional Neural Network: As the Table III shows, the results obtained when compared to the original multiclass approach are far superior. Although limited, we were able to perform some hyperparameter tuning, namely the dropout rates and the used optimizer. The decrease of the dropout rate resulted in a so much slightly overfitted model and the use of the **Stochastic Gradient Descent (SGD)** actually increased the results, but after a careful examination of the confusion matrix, the model was outputting always the same label. In the end, the best solution was the initial binary one.

The Figure 7 showcases the behaviour of the training and validation accuracy and loss values throughout the training process of the best model. It is possible to observe that, even though the training accuracy and loss deviation is rather small, the validation values tend to have significant differences between epochs.

2) Transfer Learning: Based on the same architectures as the multiclass approach, with the additionally InceptionV3 pre-trained architecture, and modified so that its input size corresponds to the others (224, 224, 3), this solution has produced the best results amongst all of the previous attempts, as showcased on Table IV. The loss and accuracy values over the epochs are shown in Figures 8, 9 and 10.

Regarding the ensemble models approach, they were all built with the InceptionV3, VGG19 and ResNet50 retrained models (described above) and these models scored positive surprising results, proving to be a good approach to solve this problem, namely the ones that Ensemble3 (VGG19 + InceptionV3) and Ensemble4 (ResNet + InceptionV3) scored, reaching a F1Score value of 0.90.



(a) Loss

(b) Accuracy

Fig. 7: CNN loss and accuracy in the train and validation sets.

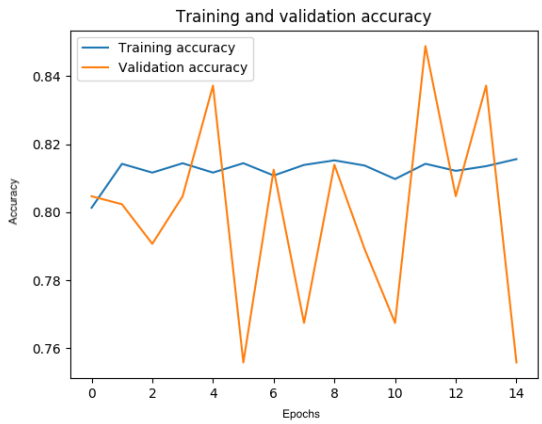
TABLE III: CNN Binary Results

Architecture	Training Set	Validation Set	Test Set			
	Accuracy	Accuracy	Accuracy	Precision	Recall	F1Score
CNN - 0.2&0.5 dropout - Adam	0.85	**0.83**	0.80	**0.72**	0.80	**0.73**
CNN - 0.1&0.3 dropout - Adam	**0.86**	0.77	0.74	0.70	0.73	0.72
CNN - 0.2&0.5 dropout - SGD	0.81	0.78	**0.81**	0.65	**0.81**	0.72

TABLE IV: TN Binary Results

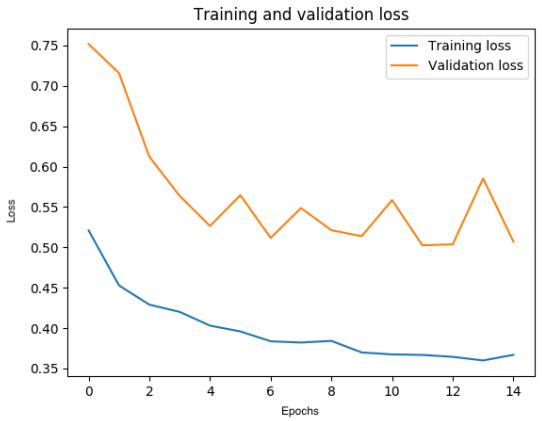
Architecture	Training Set	Validation Set	Test Set			
	Accuracy	Accuracy	Accuracy	Precision	Recall	F1Score
VGG19	0.81	0.76	0.80	0.60	0.80	0.71
ResNet50	**0.84**	**0.80**	0.81	0.67	0.82	0.74
InceptionV3	0.82	0.80	0.79	0.70	0.79	0.72
Ensemble1 (VGG19 + InceptionV3 + ResNet)	-	-	0.83	0.81	0.80	0.80
Ensemble2 (VGG19 + ResNet)	-	-	0.80	0.83	0.77	0.80
Ensemble3 (VGG19 + InceptionV3)	-	-	0.80	0.80	0.99	0.89
Ensemble4 (ResNet + InceptionV3)	-	-	0.80	0.80	1.0	**0.90**

(a) Loss

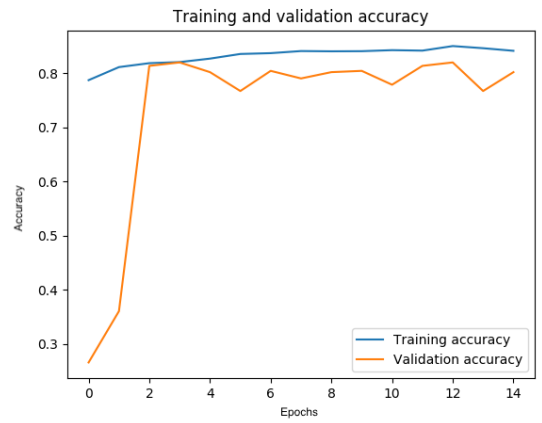


(b) Accuracy

Fig. 8: VGG19 loss and accuracy in the train and validation sets.



(a) Loss



(b) Accuracy

Fig. 9: ResNet50 loss and accuracy in the train and validation sets.

(a) Loss

(b) Accuracy

Fig. 10: InceptionV3 loss and accuracy in the train and validation sets.

V. CONCLUSIONS

Although most of the current literature regarding DNN melanoma detection uses almost exclusively binary solutions, we initially opted for a different approach by trying to classify all the three different classes, melanoma, seborrheic keratosis e nevus. This work revelead itself to be a failure, thus the obvious next step was to follow the binary route. Both the pure CNN and TL solutions performed relatively well, with an edge to the last one which was able to score a 0.90 F1Score on one of the tested models, mostly due to its most complex architecture.

Nevertheless, even the best results are non state-of-the-art level. One of the possible explanations is the fact that we did not take much time to heavily pre-process the dataset, which, according to the ISIC challenge, was one of the first steps of the challenge, and a total justified one, since the dataset was a collection of poor quality images and even only a small amount of them.

Due to time constraints, it was not also possible to further tune the hyperparameters of each of these solutions as thoroughly as possible. On a future work, that is one of the obvious steps to take. Another alternative is to train another classifier with a seborrheic keratosis vs. all classification and possible ensemble architecture just like the one used in [7].

REFERENCES

[1] Melanoma - SkinCancer.org. [Online]. Available: https://www.skincancer.org/skin-cancer-information/melanoma#panel1-5 (Accessed 2018-12-31).

[2] A. A. Ali and H. Al-Marzouqi, "Melanoma detection using regular convolutional neural networks," in *2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*. IEEE, pp. 1–5.

[3] Covalic. [Online]. Available: https://challenge.kitware.com/#phase/5840f53ccad3a51cc66c8dab (Accessed 2019-01-06).

[4] N. Dhungel, G. Carneiro, and A. P. Bradley, "Automated mass detection in mammograms using cascaded deep learning and random forests," in *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, pp. 1–8.

[5] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis, and D. Feng, "Early diagnosis of alzheimer's disease with deep learning," p. 4.

[6] E. Nasr-Esfahani *et al.*, "Melanoma detection by analysis of clinical images using convolutional neural network," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, pp. 1373–1376.

[7] K. Matsunaga, A. Hamada, A. Minagawa, and H. Koga, "Image classification of melanoma, nevus and seborrheic keratosis by deep neural network ensemble."

[8] H. A. Haenssle *et al.*, "Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," vol. 29, no. 8, pp. 1836–1842.

[9] D. C. Cireşan, U. Meier, and J. Schmidhuber, "Transfer learning for latin and chinese characters with deep neural networks," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6.

[10] S. G. Johnson. NIST special database 19. [Online]. Available: https://www.nist.gov/srd/nist-special-database-19 (Accessed 2019-01-08).

[11] Q. Hu, R. Zhang, and Y. Zhou, "Transfer learning for short-term wind speed prediction with deep neural networks," vol. 85, pp. 83–95.

[12] H.-W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, "Deep learning for emotion recognition on small datasets using transfer learning," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction - ICMI '15*. ACM Press, pp. 443–449.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., pp. 1097–1105.

[14] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets."

[15] ImageNet. [Online]. Available: http://www.image-net.org/ (Accessed 2019-01-06).

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition."

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition."