Gabriel Giorisatto De Angelo

# Skin cancer segmentation using deep learning for images acquired from smartphones

Vitória, ES

2018

Gabriel Giorisatto De Angelo

# Skin cancer segmentation using deep learning for images acquired from smartphones

Project presented to the Computer Engineering Course of the Department of Informatics of the Federal University of Espírito Santo, as a partial requirement to obtain a Degree in Computer Engineering.

Federal University of Espírito Santo – UFES

Technological Center

Department of Informatics

Supervisor: Prof. Dr. Renato A. Krohling

Co-supervisor: MSc. André G. C. Pacheco

Vitória, ES

2018

Gabriel Giorisatto De Angelo

# Skin cancer segmentation using deep learning for images acquired from smartphones

Project presented to the Computer Engineering Course of the Department of Informatics of the Federal University of Espírito Santo, as a partial requirement to obtain a Degree in Computer Engineering.

---

**Prof. Dr. Renato A. Krohling**
Advisor

---

**MSc. André G. C. Pacheco**
Co-advisor

---

**Prof. Dr. Thiago Oliveira dos Santos**
Graduate program in computer science - UFES

---

**MSc. Carlos Alexandre S. da Silva**
Federal Institute of Espírito Santo (IFES)

Vitória, ES
2018

*For those who believed in me.*

# Acknowledgements

I would like to express my deep gratitude to Professor Renato and André, my research supervisors, for their patient guidance, enthusiastic encouragement, and helpful criticisms of this research. Thanks also to all the researchers of the nature inspired computing lab (LABCIN) for the aids and contributions to my learning. In addition, I would like to thank all the members of PAD, in special Dra. Patricia Lira, Dra. Rachel Bertolani and the group of monitors from plastic surgery and dermatology.

Even during all the turbulence faced during graduation, I especially thank my friends Douglas and Luiz Otávio for the companionship and support given until the end. I also thank everyone who was part of my journey to the whole graduation.

Finally, I would like to thank the most important part of what I am, my family, especially my parents, my sister and all those who have become part of the family as well.

*"It is the time you have wasted for your rose that makes your rose so important."*
*(Antoine de Saint-Exupéry, The Little Prince)*

# Abstract

Skin cancer represents the most common group of malignant neoplasms in the white population around the world. In Brazil, it is one of the most serious public health issues. Malignant lesions at an early stage have more favourable conditions for being treated, and that is why early diagnosis is crucial to higher survival probability. In general, the diagnosis has two phases, first there is a screening step, then, the dermatologists search for certain features in the patient. In order to improve the diagnosis rate, dermatologists use dermoscopy in the screening step. However, some regions in Brazil do not have dermatologists available to the population. So, to help the dermatologists with tasks such as pre-diganostic and establishing a priority order for specialized consultations, this work proposes the use of a U-Net architecture combining color spaces, pre- and post-processing to segment skin cancer. First, the performance of each combination is compared to other works using a well-known dataset of dermoscopic images of skin lesions. Next, a case study is carried out using images taken by smartphones from a dataset created for this work, and the results with the same methodologies used before, including a transfer learning process, are presented. To collect the data for this dataset, we started a partnership with a dermatological assistance program (PAD) from Espírito Santo, in which we provided a system to collect data from patients, and a smartphone application to collect the skin cancer images.

**Palavras-chaves**: Skin cancer, Segmentation, U-Net, Conditional Random Fields, Color spaces, Smartphones, Dataset.

# List of Figures

# List of Tables

# List of abbreviations and acronyms

| | |
|---|---|
| IARC | International Agency for Research on Cancer |
| NMSC | Non-Melanoma Skin Cancer |
| CRF | Conditional Random Field |
| PAD | Dermatological Assistance Program |
| AI | Artificial Intelligence |
| BCC | Basal Cell Carcinoma |
| SCC | Squamous Cell Carcinoma |
| UFES | Federal University of Espirito Santo |
| AAML | Non-governmental Organization Martin Luther |
| CAD | Computer-Aided Diagnosis |
| ROI | Region Of Interest |
| ANN | Artificial Neural Network |
| CNN | Convolutional Neural Network |
| ReLU | Retified Linear Unit |
| FC | Fully-connected |
| FCNN | Fully Convolutional Neural Network |
| TL | Transfer Learning |
| ISIC | International Skin Imaging Collaboration |
| TP | True Positive |
| TN | True Negative |
| FP | False Positive |
| FN | False Negative |
| SEN | Sensitivity |

| | |
|---|---|
| SPE | Specificity |
| ACC | Accuracy |
| DIC | Dice Coefficient |
| JAC | Jaccard Index |
| FrCN | Full Resolution Convolutional Network |
| CDNN | Convolutional-deconvolutional neural network |
| DF | Direct Transfer |
| DT | Direct Training |
| FT | Fine-tune |

# Contents

# 1 Introduction

Artificial intelligence (AI) has been a subject of great interest in the recent years. According to a research by Stanford Artificial Intelligence Laboratory [1], the number of AI papers produced each year has increased by more than nine times since 1996. Moreover, the Scopus database contains over 200,000 papers with the key term "Artificial Intelligence". These researches have impacted in different applications, such as natural language processing, autonomous cars, data security, financial trading, marketing personalization and health care. This work focuses on the specific health care area of skin cancer. Early works using computers to automate pigmented skin cancer diagnosis have appeared in 1987 [2]. With the advance of AI, researchers started to use machine learning techniques to get better results. One example is the use of deep neural networks to classify skin cancer in a recent work by Stanford researchers [3], in which the automated diagnosis accuracy obtained is comparable to the dermatologist accuracy.

## 1.1 Overview

Skin cancer is the uncontrolled growth of abnormal cell in the skin. It occurs when unrepaired DNA damage to skin cells triggers mutations, or genetic defects, that lead the skin cells to multiply rapidly. It may be divided into two different categories, melanoma and non-melanoma skin cancer (NMSC). Basal cell carcinoma (BCC) and squamous cell carcinoma (SCC) are examples of NMSC and they represent almost every skin cancer occurrence. Nevertheless, they have low lethality risk due to rarely metastasize. On the other hand, melanoma has high levels of metastasis and it is the most harmful type of skin cancer [4].

According to the International Agency for Research on Cancer (IARC), melanoma was the 19th most common cancer worldwide in 2012, with more than 200,000 incidents and 55,488 deaths [5]. In Brazil, 85,170 new cases of NMSC, and 6,260 new cases of melanoma were estimated to be diagnosed in 2018 [6]. In 2015, 1,794 deaths in Brazil were caused by melanoma and 1,958 by NMSC. Even with low lethality risks, the higher NMSC incidence justify it having more deaths than melanoma.

## 1.2 Problem description

Malignant lesions at an early stage have more favourable conditions for being treated, and that is why early diagnosis is crucial to increase survival probability. In general, the diagnosis consists of two phases, first there is a screening step, then, the

Fig. 1. Example of dermatoscope, device used for dermoscopy.

dermatologists search for certain features in the patient. Such features for patients with suspected skin cancer may be: history of the skin lesion (duration, history of change, pain, itching and bleeding), skin type, sun exposure history, family ancestry, family history of skin cancer and some background medical history [7]. In pigmented lesions suspicious for melanoma it is usually used the ABCD-E rule [8], [9], in which the asymmetry, border irregularity, colour, diameter, enlarging and other features are the essentials features for the diagnostic [7]. In addition, there are some others scoring methods like the seven-point checklist [10], [11], three-point checklist [12], and Menzies [13]. Also, in order to improve the diagnosis rate, dermatologists use dermoscopy in the screening step. Dermoscopy is a non-invasive diagnostic technique that links clinical dermatology and dermatopathology by enabling the visualization of morphological features which are not discernible by screening with the naked eye [14]. An example of dermatoscope[1], used for dermoscopy, is shown in Fig. 1.

## 1.3 Motivation

The state of Espírito Santo has a dermatological assistance program (PAD, in portuguese, Programa de Assistência Dermatológica) since 1989. It is organized by the Federal University of Espirito Santo (UFES) in partnership with State Department of Health, Non-governmental organization Martin Luther (AAML, in Portuguese, Associacao Albergue Martim Lutero), city halls, municipal secretaries of Health and local parishes.

[1] instrument used in dermoscopy to allow the visualization of the subsurface structures of the skin revealing lesion details in colors and textures. https://www.healthandcare.co.uk/schuco-dermatoscopes-accessories/schuco-dermlite-fluid-dermatoscope.html

The program usually happens once in a month in a weekend and it takes place in 11 different countryside cities of the state. Usually, the population of these places cannot afford a medical treatment, as they, in general, are low incoming and the nearest specialist would take hours of travel. Then, dermatologists, plastic surgeon, and medical students are the ones responsible for going to these places and taking care of the patients, providing medical care for the needy community. In 2017, each city had on average 300 people attended by the program. Therefore, since 1989, thousands of lives were saved through this program, which provides a full treatment, from the screening to the surgical process.

As previously mentioned, some regions in Brazil do not have dermatologists available to the population. In addition, the availability of a dermatoscope is even more difficult as it is a costly device. These factors make it difficult, and sometimes impossible, to diagnose skin cancer with a good accuracy. Therefore, there was a motivation to develop a method that uses data that are not from dermatoscope. Since smartphones became a common tool among the population, it was proposed the development of a method that helps in the process of diagnosing skin cancer using images taken by smartphones. As segmentation is one of the first steps in the diagnosis, and can be used to generate essential features for the classification stage, such as some of the ABCD-E rule features, it was the stage studied in this work.

## 1.4 Objectives and expected results

This work proposes a methodology for skin cancer segmentation that uses a U-net architecture, a combination of color spaces and, in order to improve the segmentation results, the conditional random fields (CRF) as a post-processing tool. First, the performance of the methodologies is compared to other works using a well-known dataset of dermoscopic images of skin lesions. Next, a case study is done using images taken by smartphones. This type of image has several difficulties, such as zoom, angle and lighting, making detection substantially more challenging. Inspired by the good results of the deep learning approaches, this work studies the application of these methodologies in the dataset collected in the PAD.

## 1.5 Related works

The use of computer-aided diagnosis (CAD) systems for skin cancer analysis has been investigated in the last years and has proved to be a promising tool. Masood and Al-Jumaily [15] make a review of techniques and algorithms of CAD systems for skin cancer. The main proposal of those systems is to detect and extract features from the lesion, and occasionally classify the disease. This process includes four important stages namely pre-processing, segmentation, feature extraction and classification. Methods for

these steps are proposed in [3], [16]–[25], and a review is made in [26]. A CAD system needs to have some data as input. Usually, the input is one of the several types of imaging methods such as cutaneous photography, dermoscopy, confocal scanning laser microscopy, ultrasound, magnetic resonance imaging, optical coherence tomography, and multispectral imaging. The most common in the literature is the use of dermoscopic images, as in [16]–[21], [24], [25]. However, as mentioned before, most cities in countryside do not have dermatoscope. Therefore, the use of standard camera images is relevant, and some works got first results using it, as in [3], [22], [23].

# 2 Background on deep learning to segmentation

This chapter presents a theoretical foundation of the techniques used in this work. First, the pre-processing used in this paper is presented, along with a description of some difficulties present in skin cancer images. Next, it is described the concept of segmentation and how it is used in the domain of skin cancer images. Then, the artificial neural networks, together with the convolutional networks, are introduced, followed by the architecture used in this work. Also, some techniques to improve deep learning are detailed. Finally, the post-processing used in this work is presented.

## 2.1 Pre-processing

Computer aided systems for the diagnosis of skin cancer must differentiate the lesion from the healthy skin to work well. As shown in Fig. 2(a), transitions between lesion and skin around are almost imperceptible, making it hard to detect the lesion. Skin cancer images may also contain artifacts, such as hairs in Fig. 2(b), reflections in Fig. 2(c), shadows, skin lines and ink markings in Fig. 2(d), which may affect the accuracy of the image segmentation. So, it is important to have a pre-processing step to improve the segmentation.

In order to solve those problems, the first step is to choose between scalar (single channel) and vector (multichannel) processing. In scalar processing, the original image is converted to some single channel image such as grey-level image or only one channel of the image. In vector processing, the original RGB image may be used or converted to other color spaces, such as CIE L*a*b, CIE L*u*v, and HSV spaces. These color spaces are commonly used in literature, since they ensure approximate perceptual uniformity and achieve invariance to different imaging conditions such as viewing direction, illumination intensity, and highlights [16]. In this work, a combination of color spaces is made and described in Sec. 3.2, and its results are compared in Sec. 4. As described in Sec. 4.1, the images from the PAD dataset were also cropped to separate the lesion from other information in the images.

## 2.2 Segmentation

The main goal of segmentation is to partition the image into mutually exclusive regions to which can be subsequently attached meaningful labels [27]. In traditional

(a) Smooth transition (b) Hair

(c) Reflection (d) Ink marking

Fig. 2. Problems in skin cancer images.

systems, segmentation is the most important phase, as the whole process depends on how good it can detect the region of interest (ROI) of an image. In this work, the skin cancer is the ROI of the images. Fig. 3 presents some examples of segmentation, in which the contour represents the region of interest desired for each sample. It is desirable that the segmentation obtained by the algorithm is the closest to these regions of interest.

In the past, segmentation was essentially an unsupervised task, depending only on the input image to work, as the techniques described in Sec. 2.2.2. However, nowadays deep learning techniques are getting good results for this task, as described in Sec. 1.5. And, it is known that most algorithms based on machine learning have a supervised learning process, making the task dependent on the availability of a ground truth.

### 2.2.1 Supervised segmentation

Supervised segmentation algorithms require the ground truth of the images to work properly. In other words, in case of skin cancer segmentation, a specialist needs to determine where the lesion is in each image, as in the example shown in Fig. 4. The algorithms use the prior knowledge of the training set to learn how to generalize the targeting task. One of the most effective types of supervised algorithm is artificial neural networks (ANN) [28].

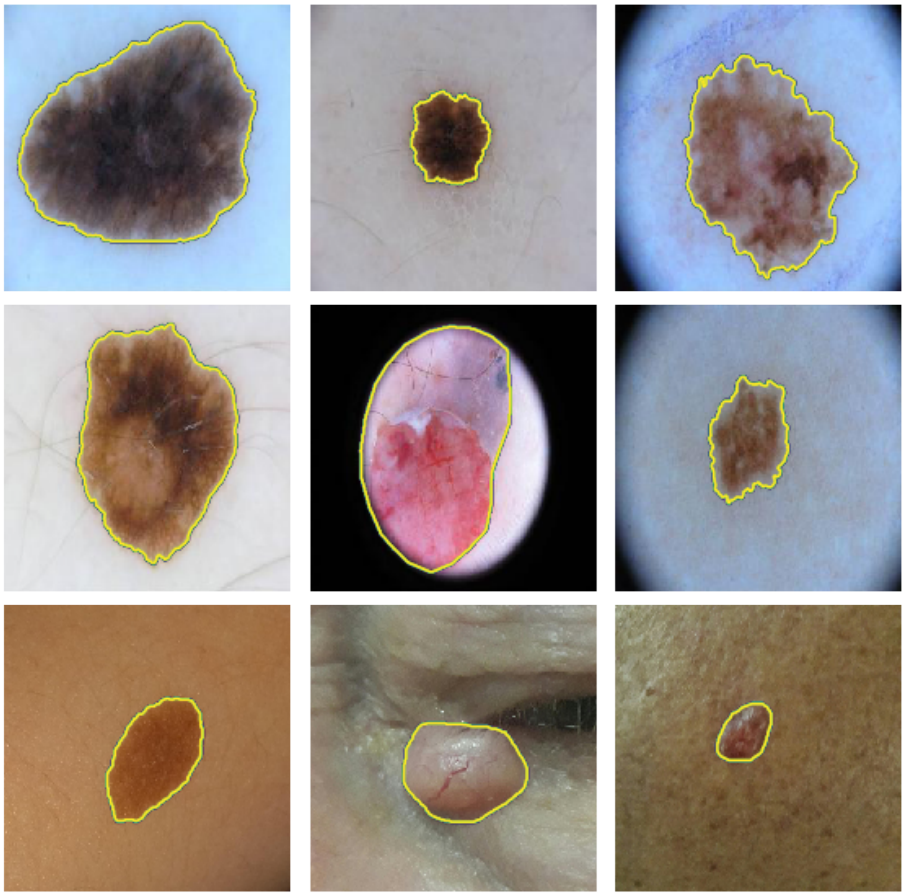


Fig. 3. Segmentation examples from the datasets used in this work. The first six images, from top to bottom, are from the ISIC 2017 Challenge dataset, and the last three images are from PAD dataset. The yellow line delimits the ROI of each image.

### 2.2.2 Unsupervised segmentation

Unlike supervised segmentation, unsupervised methods do not need a ground truth to work. These methods can be divided into some categories such as the border-based and region-based categories. Methods in these categories use respectively edge operators and region splitting or merging algorithms for segmentation [30], [31]. In addition, there are some methods based on histogram thresholding, which define a threshold value to get the region of interest. The Otsu [32] threshold method is one of the most used. Finally, there are algorithms based on active contours in which the initial curves move toward the boundary region of interest [33], [34].

## 2.3 Artificial neural network

An artificial neural network (ANN) is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use [35]. The most ANN models are compound by

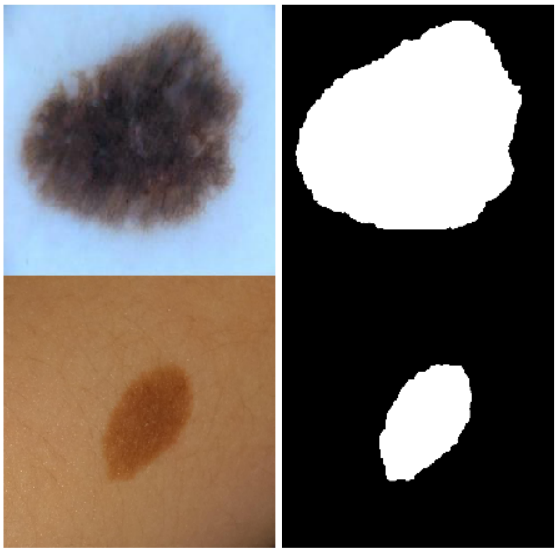


Fig. 4. Examples of ground truth for skin cancer images. In the top there is an example using ISIC 2017 Challenge dataset [29] and in the bottom using PAD dataset. The left image is the original one, and the right image is the ground truth.

artificial neuron, shown in Fig. 6. The output of the the neuron is described by Eq. 2.1

$$y = f(\sum_{i}^{m} x_i w_i + b) \tag{2.1}$$

where $\mathbf{x}$ are the inputs, $\mathbf{w}$ are the weights, $b$ is the bias and $f$ is the activation function.

An activation function that has become very popular is the ReLU (Rectified Linear Unit), described by Eq. 2.2 and illustrated in Fig. 5.

$$f(x) = max(0, x) \tag{2.2}$$

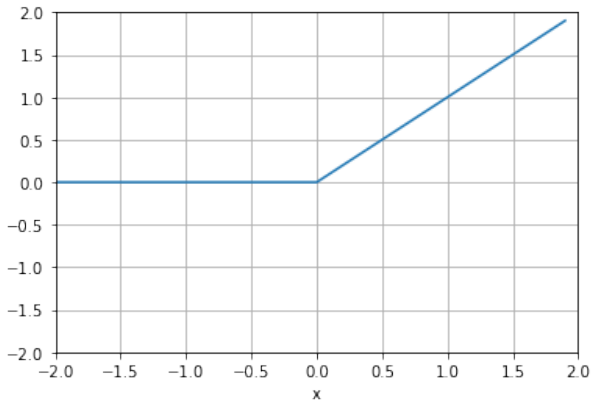



Fig. 5. ReLU activation function.

However, other activation functions such as the sigmoid and the tanh, respectively

described by Eqs. 2.3 and 2.4, are also used.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{2.4}$$

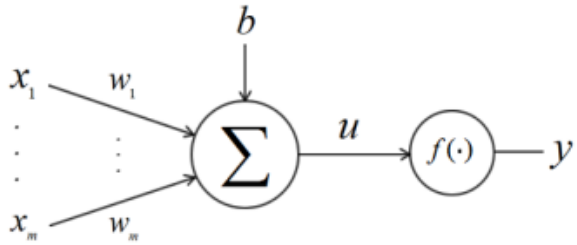



Fig. 6. The perceptron neuron model's representation [36]

The ANN architecture is divided in three different layers: input, hidden and output. The input and output layers represent respectively the data fed to the network and the resulting output in the problem context. The hidden layer is an essential part of an ANN, it is connected to the input layer and to another layer, which can be also a hidden layer, or, as shown in Fig. 7, an output layer. The best number of hidden layers is a currently studied subject and usually it is empirically determined [37].

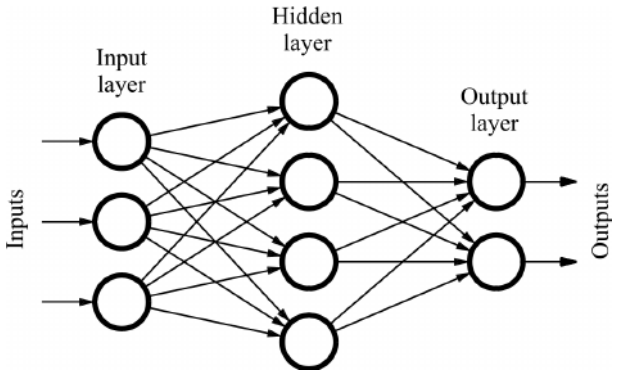



Fig. 7. A feedforward neural network representation [38]. The information always flows from the input nodes, through the hidden nodes to the output nodes.

Feedforward Neural Networks are a type of ANN where the information goes only in one way. So, the neurons in one layer can only affect the neurons in the next layer, having no feedback in this architecture. The ANN shown in Fig. 7 is a feedforward network with only one hidden layer.

In general, ANN learns from data how to generalize a function. So, given an input, the network should return a result close to the expected value. To do so, it is necessary a learning algorithm to determine the best weights and bias values to the architecture. The

most common learning algorithm is the backpropagation [39], described by the pseudocode in Algorithm 1. The main idea of this algorithm is to adjust the connection weights of the network aiming to minimize a loss function using the gradient descent. A common loss function used is the mean square error, described in Eq. 2.5. The gradient is the partial derivative of this measure with respect to the weights of the network, as described in Eq. 2.6.

$$E = \frac{1}{T}\sum_{t=1}^{T}(\hat{y}_t - y_t)^2 \tag{2.5}$$

$$\Delta\mathbf{W} = -\alpha\frac{\partial E}{\partial \mathbf{W}} \tag{2.6}$$

where $T$ is the number of samples in the dataset, $\hat{y}$ is the expected output, $y$ is the actual output of the network, $\mathbf{W}$ is the weight matrix and $\alpha$ is the learning rate. Therefore, the weight matrix is updated with Eq. 2.7 several times until a stopping criteria is met.

$$\mathbf{W} = \mathbf{W} + \Delta\mathbf{W} \tag{2.7}$$

**Algorithm 1** *backpropagation*

```
1:  procedure BACKPROPAGATION(T, N, k, α)
2:      T is the pairs input and output
3:      N is the number of hidden layers and k the number of hidden neurons for each
        layer
4:      α is the learning rate
5:      while stopping criteria is not met do
6:          for each sample of T do
7:              for each connection weight w do
8:                  calculate ΔW with Eqs. 2.5 and 2.6.
9:                  update W with Eq. 2.7
10:     return trained weights and bias
```

## 2.4 Convolutional neural network

Convolutional neural networks (CNN) are a special type of ANNs. They are composed by neurons that have learnable weights and biases, and also have a differentiable measure function. However, CNN considers the spacial distribution to take advantage of the fact that the input is usually an image, which have high correlation between pixels. The CNN architecture contains three main types of layer: convolutional layer, pooling layer, and fully-connected layer. Le Cun et al. [40] developed LeNet-5, a pioneering CNN architecture to handwritten digit recognition. The architectures from the original work is illustrated in Fig. 8. The LeNet-5 has seven layers, not counting the input. The input is

a $32 \times 32$ pixel image. The convolutional layers are labeled Cx, the subsampling layers are labeled Sx, and fully-connected layers are labeled Fx, where x is the layer index. The first layer, C1, is a convolutional layer with six feature maps of size $28 \times 28$. Next, there is S2, which is a subsampling layer with six feature maps of size $14 \times 14$. Layer C3 is a convolutional layer with 16 features maps of size $10 \times 10$. Layer S4 is a subsampling layer with 16 feature maps of size 5. Layer C5 is a convolutional layer with 120 feature maps of size $1 \times 1$. Layer F6 contains 84 units and is fully connected to C5. Finally, the output layer is composed of Euclidean radial basis function units, one for each class, with 84 inputs each.

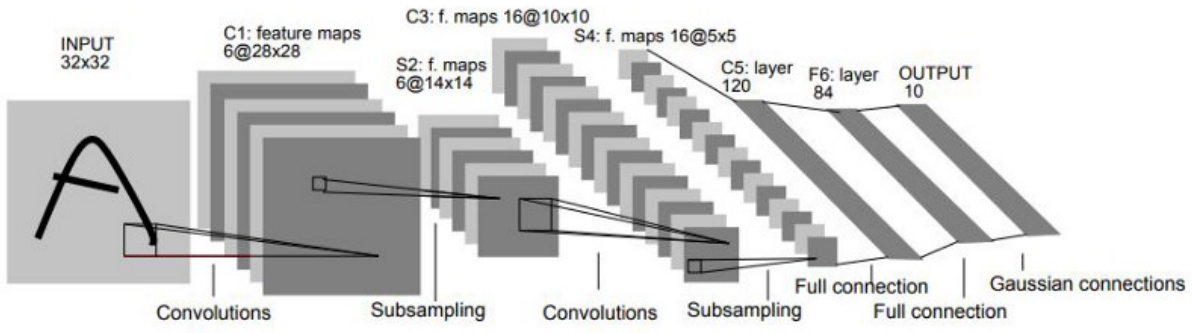



Fig. 8. The CNN architecture, LeNet-5, proposed by Le Cun et al. for digits recognition [40]. Each plane is a feature map and the operations are described at the bottom.

### 2.4.1 Convolutional layer

The convolutional layer is responsible to extract features from the image. The parameters in this layer consist of $K$ learnable two dimensional filters, where each filter has spatial size of $F$. During the forward pass, each filter is slided with stride of value $S$ across the width and height of the input volume and it is computed the dot products between the entries of the filter and the input at any position. This process produces a 2-dimensional activation map that gives the responses of each filter at every spatial position. Sometimes it is convenient to pad the input volume with $P$ zeros around the border, allowing to control the spatial size of the output volumes. As such, the output dimension is determined as:

$$D = \frac{(V - F + 2P)}{S} + 1 \tag{2.8}$$

where V is the volume size of the input.

To exemplify the convolutional step, let us consider a image with $V = 5$ represented by the blue matrix shown in Fig. 9. It is used a filter with dimension $F = 3$ and with its values shown in the red matrix in Fig. 9. It is considered $S = 1$, there is no zero padding ($P = 0$), and the bias is equal to one. The output dimension is calculated using Eq. 2.8,

and it is $D = 3$. Some states of this process is illustrated in Fig. 9, and the resulting output matrix is the green matrix shown in the final state.

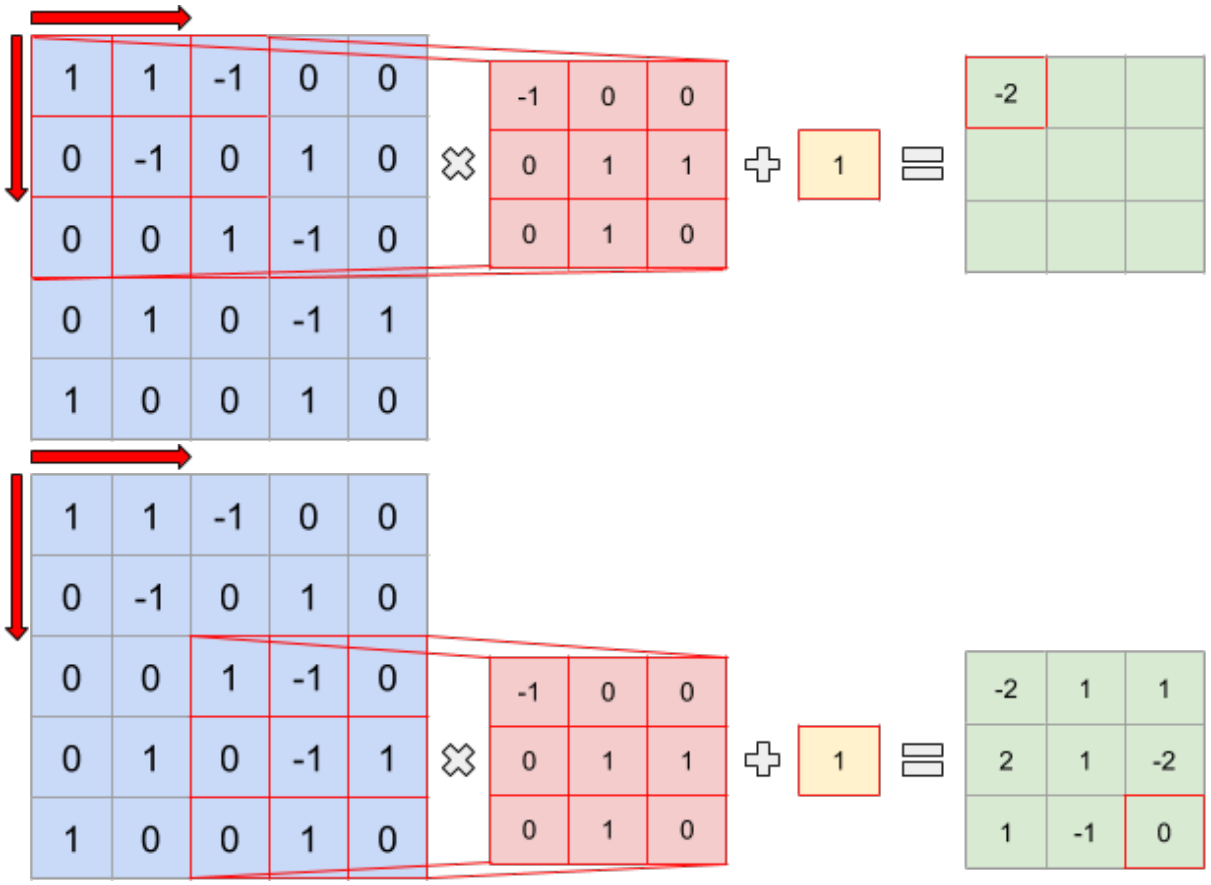



Fig. 9. The convolutional process is exemplified in two states: initial state (top) and final state (bottom). The blue matrix is the input matrix, the red matrix is the filter, the yellow box is the bias and the green matrix is the output matrix. There is no padding and the stride is equal to one.

### 2.4.2 Activation layer

Element-wise multiplication and summation are both linear operations that occur during the convolutional layer. However, it is desirable to introduce non-linearity into the system. And for this, it is common to have a non-linear layer (activation layer) just after each convolutional layer. The activation layer consists of a nonlinear activation function. The most common functions are: sigmoid, tanh and ReLU (Rectified Linear Unit). In convolutional networks, ReLU is usually used as an activation function due to its computational efficiency. Its mathematical formulation is described in Eq. 2.2, and its graph is illustrated in Fig. 5.

### 2.4.3 Pooling layer

After convolution, it is common to insert a pooling layer to reduce the dimensionality of each feature map, control overfitting, and help reduce the translation invariance. There are some types of spatial pooling, such as max, average and sum. The most common is the max pooling, in which it is defined a spatial neighborhood and the largest element is took from the rectified feature map within this window. A new matrix is used to exemplify this step using a $2 \times 2$ window with a stride equals 2. As shown in Fig. 10, each little $2 \times 2$ square with different color generate a max number in a resultant matrix.

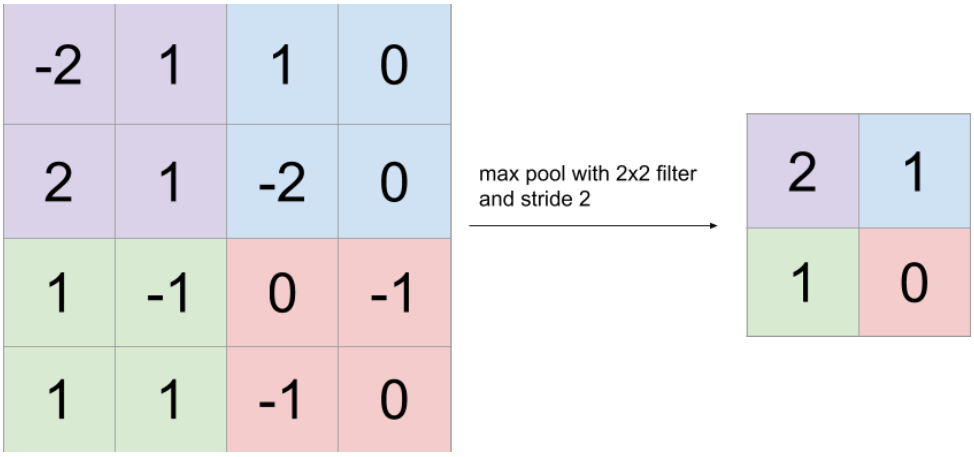



Fig. 10. Example of pooling operation using the max pooling with $2 \times 2$ filter and stride 2. The maximum value of each $2 \times 2$ colored square is selected for the output matrix in the position with the same color as the square.

### 2.4.4 Fully-connected layer

In order to execute tasks such as classification, the network has to be able to generate probabilities of a certain class for each input. So, after detecting high level features, usually it is attached a fully-connected (FC) layer to the end of the CNN, as in LeNet-5 architecture shown in Fig. 8. The way the fully-connected layer works is that it examines the output of the previous layer, since it has connections to all the previous activations, and determines which features are most correlated to a given class. The output of this layer is a N dimensional vector where N is the number of classes of the domain, and each value in this vector represents a probability of a given class.

### 2.4.5 Batch normalization

As described in Sec. 3.5, the architecture used in this work uses batch normalization after each convolutional layer, as well as after concatenation with the result of the convolution transposed. Batch normalization was initially proposed to reduce the internal covariate shift, which is caused by the distribution change of each layer's input as the

network parameters change during training [41]. This reduction allows the use of a higher learning rate and to be less careful about initialization, improving the training time. However, Santurkar et al. [42] recently state in their work that the advantage of using batch normalization is not related to the reduction of internal covariate shift, but to the smoothing of the optimization landscape, making the gradient more predictable and stable, decreasing the training time.

## 2.5 Fully convolutional networks

### 2.5.1 Overview

Fully Convolutional Network (FCN) is a specific type of CNN, where the final fully-connected layer is replaced by a convolutional layer. It started to receive attention after the good results in segmentation tasks in [43]. Since it was proposed for semantic segmentation, it is desirable that the final output layer has the same height and width as the input image, and the number of channels needs to be equal to the number of classes segmented. However, as shown in Fig. 11, as the image goes through the convolutional and pooling layers, the width and the height of the image are reduced. In order to have an output segmentation with the same size of the input, FCNs use the backward convolution called transposed convolution [43]. The transposed convolution, initially called deconvolution, had its first appearance, without receiving a specific name, in [44]. Next, in a later work by the same author, the term deconvolution layer was defined [45]. It can be used as a method of upsampling the pixels without a predefined interpolation method, as in Fig. 11. One way of implementing the transposed convolution is by reversing the forward and backward passes of the convolution, as described in [43]. Thus, in the same way as the convolutional layers, the transposed convolution also has weights that will be trained by the backpropagation. An excellent visual demonstration of the transposed convolution is done in [46], where the technique is discussed in more depth.

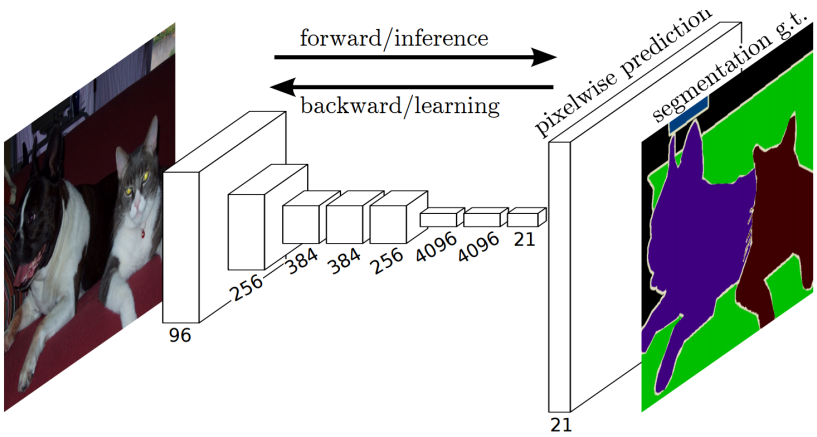



Fig. 11. The fully convolutional network presented in [43].

### 2.5.2 U-Net

The U-Net was first proposed by Ronneberger et al. [21] for biomedical image segmentation. The architecture is an extension of the fully convolutional network. As described previously, the upsampling layers in FCN increase the resolution of the output. The main idea in U-Net is to combine the high resolution features from the contracting path with the upsampled output. This process allows the network to propagate context information to higher resolution layers, making the expansive path symmetric to the contracting path as shown in Fig 12. The U-Net architecture used in this work has a different number of layers than originally proposed, however the main concept of the network is maintained. Sec. 3.5 describes the architecture used in greater detail.

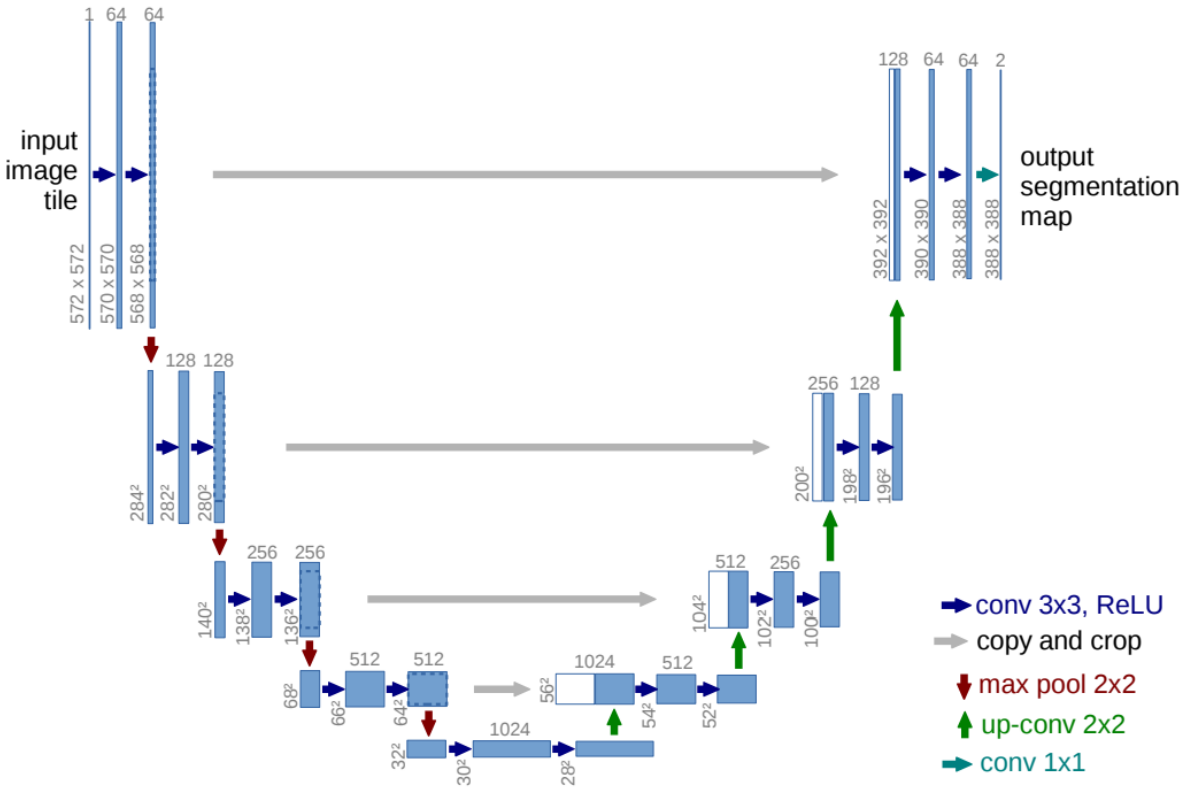



Fig. 12. U-Net architecture proposed by Ronneberger et al. [21]. Each blue box corresponds to a multi-channel feature map. The number of channels is on top of the box. The x-y-size is at the lower left edge of the box. The white boxes represent copied feature maps.

## 2.6 Transfer learning

When using deep learning techniques like CNN, the amount of data available should be large enough for the algorithm to learn the essentials features from the domain. However, there are some cases where the data acquisition is hard or costly, as in medical domains.

Transfer learning (TL) is the concept that tries to learn from a source domain where there are easier conditions, for example, more data or computational resource available, and transfer the knowledge for a target domain [47]. In case of convolutional neural networks, there are different types of TL, and it depends on the source and target datasets and how they are related [48]. When there is a small target dataset and it is similar to the base dataset, the most appropriated TL is to copy all the convolutional layers trained from the base network to the target network and freeze them. Then, the fully-connected layer is trained from the scratch with the target dataset. Another way of TL is to copy from the base network just the shallower convolutional layers and freeze them, while all the others layers are trained from the scratch. This is motivated by the fact that the first convolutional layers retain more generalized features from the images, such as color, texture and edges, while deeper layers parameters attempt to captures specific features from the target domain [49]. The choice of freezing or not some layers depends on the size of the target dataset and the number of parameters in the layers. If the target dataset is small and the number of parameters is large, fine-tuning the parameters can lead to overfitting, so it is better to let them frozen. On the other hand, if the target dataset is large, overfitting is not a problem, and fine-tuning the parameters may lead to better performances.

## 2.7 Data augmentation

Working in the same direction as transfer learning, data augmentation also attempts to improve the training phase in domains where there is insufficient data. The importance of increasing data for the analysis of skin cancer was shown in a work by the RECOD Lab, where 13 different data augmentation scenarios were investigated [50]. The scenarios include traditional color and geometric transforms, elastic transforms, random erasing and a novel augmentation that mixes different lesions. These scenarios are shown in Fig. 13. For this work, only some of the techniques presented in the RECOD Lab work were selected and are described in Sec. 3.3.

## 2.8 Conditional random fields

Although the use of fully convolutional networks has performed very well in segmentation tasks, the predictions from this type of model are still quite coarse, since the max-pooling stages in earlier parts of the network resulted in the loss of many spatial information. As a result, fine structures and object boundaries are usually poorly segmented [51]. Also, there are some times where the pixel is classified as an incorrect label in the middle of a blob of pixels from another label. However, we know that skin cancers are usually continuous and nearby pixels are likely to be part of the same object. Conditional

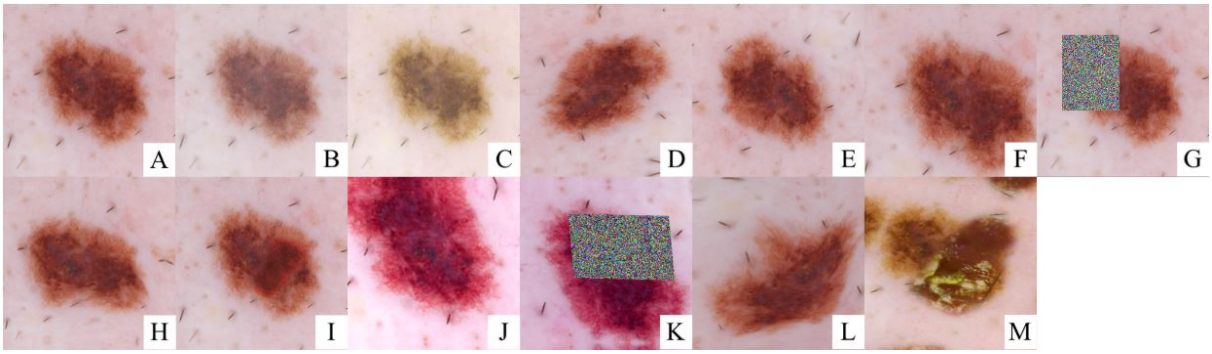



Fig. 13. Data augmentation scenarios presented in [50]. The scenarios can be defined from A to M in alphabetical order as: A - no augmentation; B - saturation, contrast and brightness; C - saturation contrast brightness and hue; D - affine; E - flips; F - random crops; G - random erasing; H - elastic; I - lesion mix; J - basic set; K - basic set + erasing; L - basic set + elastic; M - basic set + mix.

random fields (CRF) are models often used to help to improve performance of the final segmentation is such cases.

CRF models complex structures consisting of a large number of interrelated parts. For example, an image composed by separated pixels, in which there are some relations between the pixels. Each pixel $u$ has a set of possibles labels, such as lesion or non-lesion in the skin cancer segmentation task. The main idea of CRF is to find a configuration $\mathbf{x}$, where each pixel has a specific label, in which the total energy is minimized. The total energy is a combination between a unary and a pairwise energy. The unary energy $\psi_u(x_i)$, or unary cost, is the cost to assign a label $x$ to the pixel $i$. This cost is obtained from the final score of the CNN. The pairwise energy $\psi_p(x_i, x_j)$, or pairwise cost, is used to model interaction between pixels. The total energy is described by Eq. 2.9.

$$E(\mathbf{x}) = \sum_i \psi_u(x_i) + \sum_{i \neq j} \psi_p(x_i, x_j) \tag{2.9}$$

where $i$ and $j$ are indexes for the pixels from the image.

As it was used the fully connected CRF proposed in [52], where every pixel is related to every pixel in the image, the pairwise energy is described in Eq. 2.10.

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) k(\mathbf{f}_i, \mathbf{f}_j) \tag{2.10}$$

where each of the vectors $\mathbf{f}_i$, and $\mathbf{f}_j$ are feature vectors for pixels $i$ and $j$ in an arbitrary feature space and $\mu$ is a label compatibility function.

For the segmentation task, the kernel function $k(\mathbf{f}_i, \mathbf{f}_j)$ can be defined in terms of

the color vectors $I_i$ and $I_J$ and positions $p_i$ and $p_j$ by the Eq. 2.11.

$$k(\mathbf{f}_i, \mathbf{f}_j) = w^{(1)} exp(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2} + w^{(2)} exp - \frac{|p_i - p_j|^2}{2\theta_\gamma^2} \tag{2.11}$$

where $w^{(1)}$ and $w^{(2)}$ are linear combination weights, $exp(-\frac{|p_i-p_j|^2}{2\theta_\alpha^2} - \frac{|I_i-I_j|^2}{2\theta_\beta^2})$ is inspired by the fact that nearby pixels with similar color are likely to be in the same class. The parameters $\theta_\alpha$ and $\theta_\gamma$ control the nearness and similarity degree. The $exp - \frac{|p_i-p_j|^2}{2\theta_\gamma^2}$ part is responsible to remove small isolated regions [52].

The compatibility function $\mu$ used is given by the Potts model, $\mu(x_i, x_j) = [x_i \neq x_j]$, where the cost is zero when two neighboring pixels have the same label, and $\lambda$ if they have different labels [52].

# 3 Deep learning for skin cancer segmentation

This chapter details how the methods described in the Sec. 2 are used and combined to segment skin cancer images. There are two datasets used in this work, the first one is a public dataset provided by the International Skin Imaging Collaboration (ISIC) for the ISIC 2017 Challenge [29]. The dataset contains 2,000 dermoscopic images for training and 600 for testing, some examples from this dataset are shown in Fig. 14. As explained before, dermatoscope is not available in most places in Brazil, and the easiest way of capturing images would be using smartphones. However, training ML algorithms using dermoscopic images, and testing using images taken from smartphones seems not to work [53]. Therefore, to compare the results and ensure the difference between these types of image, we started to create a new dataset containing clinical images acquired from smartphones.

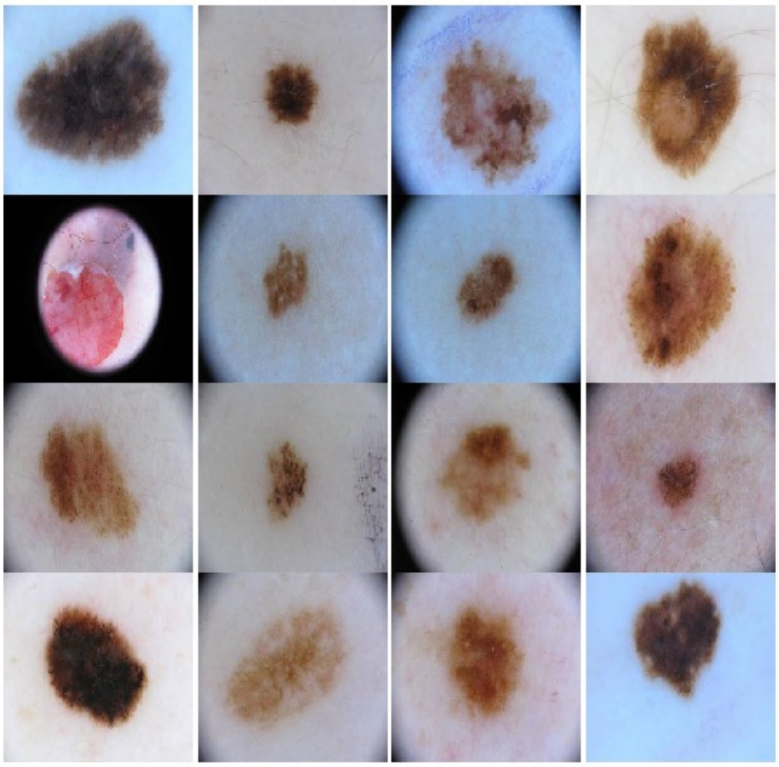


Fig. 14. Examples of desmoscopic images from the ISIC 2017 Challenge dataset [29].

## 3.1 PAD dataset

In the late of 2017, we started a partnership with the dermatological assistance program (PAD), by providing a system able to collect data from patients, present statistics,

and lesion tracking. To collect the images, a smartphone application was developed, and the students used it over the year during the program. Some screens from the app are shown in Fig. 15. It is all in Portuguese, because PAD occurs only in Brazil. Some of the images collected using the application are shown in Fig. 16. All data collected is secure and protected by password, and only allowed people can use the system. The app does not store any image or data from the patients.

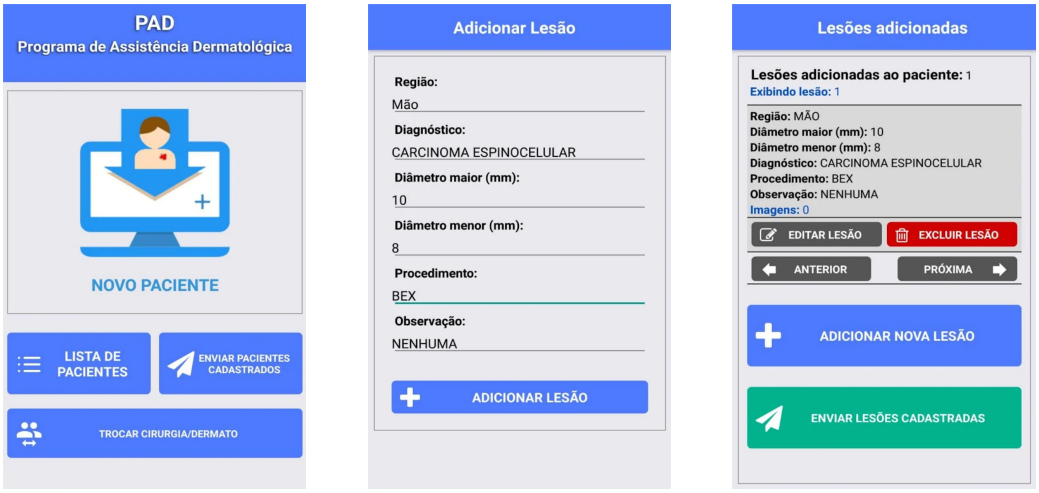



Fig. 15. Screens from the app developed for data collection. As the app is constantly updated, the screens may have been changed since the final version of this work. The data contained in these images are for illustrative purposes only.

As the deep learning algorithm used in this work has a supervised learning step, it is needed to create a ground-truth for each image. Researchers from LABCIN and PAD have used the experience acquired in the program to generate the masks for the entire dataset. However, as said before, segmentation is a subjective task, and the ground-truth created should be verified by specialists. But, as the creation of a dataset is a hard and long task, in the moment of this work, there was not yet a dermatologist participation, which can lead to some small divergences in the masks generated.

## 3.2 Color space combination

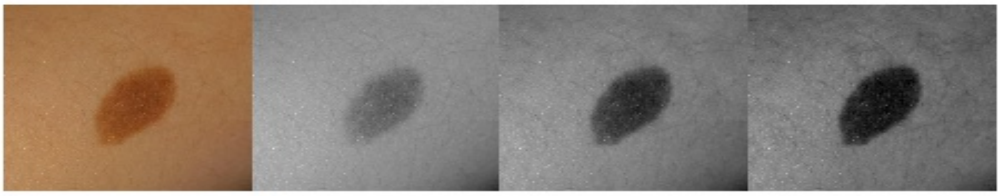
Color space is one of the first things to choose in a image segmentation problem. It is responsible for representing an image by saving its information in some specific model. The most common color space used in image processing is the RGB, which uses values of Red, Green and Blue to represent the image. Besides the preference for RGB, some works use others color spaces like HSV and CIE L*a*b [54]. The main justification in these cases is that HSV and CIE L*a*b are invariant to different imaging conditions such as illumination [16]. However, when using CNNs for segmentation, the color space with better results is not well defined, and it varies in accord to the domain. Moreover, a recent work

Fig. 16. Examples of skin cancers images from the PAD dataset. All the images were taken from smartphones using the app developed.

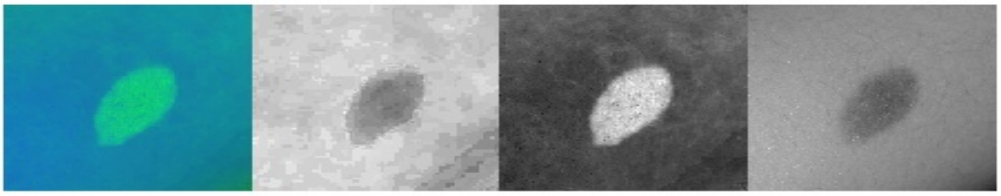
showed that the performance of convolutional neural networks is color-sensitive [55]. So, to define the best color space to this work domain, it is compared the RGB, HSV, CIE L*a*b and, inspired by the first place of ISIC 2018 segmentation task [56], some combinations between these spaces are also used. Fig. 17 shows the same image represented in different color spaces.

## 3.3 Data augmentation

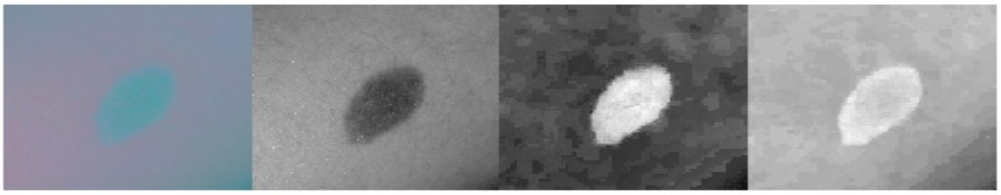
In this work, the data augmentation used was the Hue shift, horizontal flip, vertical and horizontal shifts. The augmentation is done in training time, so each image is randomly affected by the transformations in each epoch. Since some of the inputs are combinations of color spaces, having more than three dimensions, the augmentation happens before the color space transformation. That means that all augmentations are done in the RGB. Next, each augmentation is described, and some examples using ISIC and PAD datasets are shown in Figs. 18, 19 and 20. First, the Hue of the images were shifted by a value sampled from a uniform distribution of $[-0.1, 0.1]$. The Hue can be thought of as the shade of the colors in an image. This augmentation tries to prevent variations caused by how different devices can capture colors in different ways.



(a) RGB



(b) HSV



(c) CIE L*a*b

Fig. 17. For each color space, from left to right it is represented: the image composed by the three channels illustrated as a RGB image; the first channel of the color space in grayscale; the second channel; the third channel.

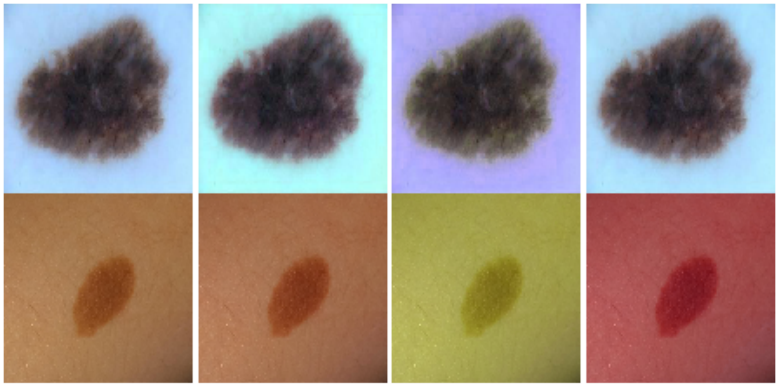


Fig. 18. In the top there are examples of random Hue shifts using ISIC 2017 Challenge dataset and in the bottom using PAD dataset. The first image on the left is the original one.

Next, each image was randomly flipped in the horizontal, increasing the robustness against variability caused by the angle in which the photo was taken.

For the last, the images were shifted in vertical and horizontal by a random factor

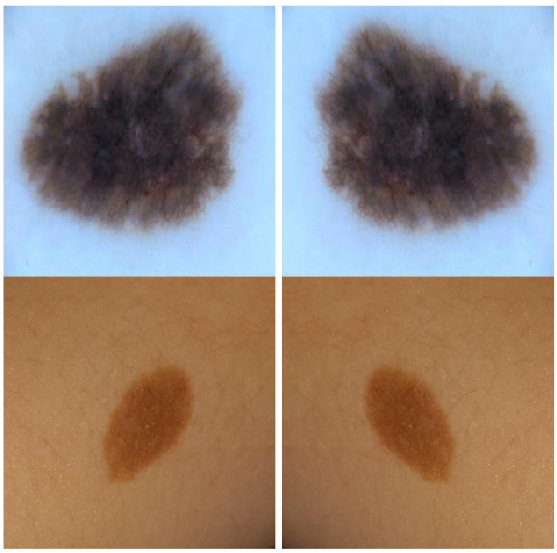


Fig. 19. In the top there is an example of horizontal flip using ISIC 2017 Challenge dataset and in the bottom using PAD dataset. The image on the left is the original one.

between $[-0.1, 0.1]$ of theirs original size.

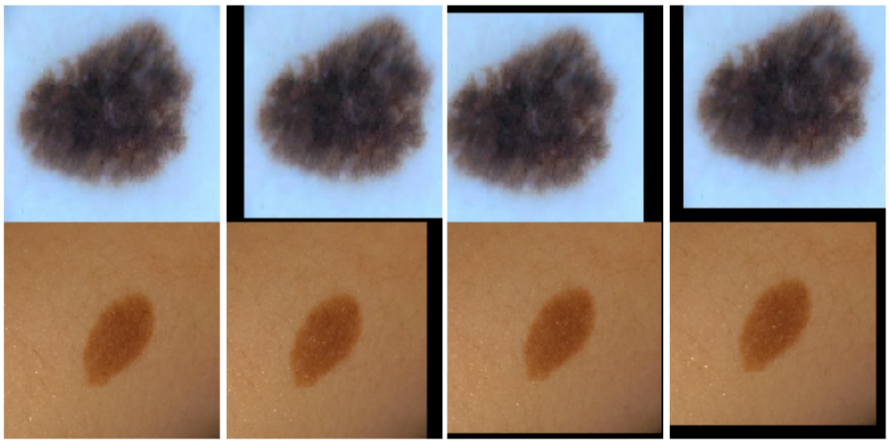


Fig. 20. In the top there are examples of random vertical and horizontal shifts using ISIC 2017 Challenge dataset and in the bottom using PAD dataset. The first image on the left is the original one.

## 3.4 Transfer learning

For transfer learning, the NN was first trained using the ISIC 2017 Challenge dataset. Then, it was used the same approach of [22], [57] where all CNN layers were fine-tuned using a learning rate 10 times smaller for the PAD dataset. Since the source domain is composed by dermoscopic images, and the target domain composed by images taken from smartphones, they have different characteristics, such as high variability caused

by lighting, zoom and camera model. Also, as explained before, the ground-truth of the datasets were obtained by different people, and it is known that segmentation is a very subjective activity, which can lead to a biased ground-truth. These are the reasons why fine-tuning seems to be needed for better learning the specifics features of the target domain for segmentation.

## 3.5 U-Net architecture

The architecture used in this work is the U-Net architecture proposed in [21] with some small changes in the number of layers and including batch normalization. The network consists of two main paths, one for contraction and one for expansion. The contraction path has several encoding blocks, where each block is composed by a convolution block followed by a pooling layer of pool size $2 \times 2$. The convolutional block consists of two sequences of convolutional layer, batch normalization and a ReLU activation layer. All convolutional layers, except the last layer of the network, have $3 \times 3$ filters, a padding such that the output has the same length as the original input and a stride equal to 1. The expansive path consists of decoding blocks, where each block is composed by a sequence of transposed convolution, concatenation with the result from the contraction path, batch normalization, and a convolutional block. All transposed convolutions have $2 \times 2$ filters, stride equal to 2 and a padding as the same type of the convolutional layers. The last layer is a convolutional layer containing only one 1x1 size filter and a sigmoid activation layer. Table 1 summarizes all the information described in this paragraph.

Table 1 – The U-Net Architecture used in this work. The input of each block is described in the Block Input column, where concatenated inputs are comma separated.

| Block | Layer/Block | Filter size | Block Input |
|---|---|---|---|
| Input data | 256 x 256 x Channels | - | Image |
| Encoder 1 | Conv. Block 1<br>Max Pooling 1 | 3 x 3 x 32<br>2 x 2 | Input data |
| Encoder 2 | Conv. Block 2<br>Max Pooling 2 | 3 x 3 x 64<br>2 x 2 | Encoder 1 |
| Encoder 3 | Conv. Block 3<br>Max Pooling 3 | 3 x 3 x 128<br>2 x 2 | Encoder 2 |
| Encoder 4 | Conv. Block 4<br>Max Pooling 4 | 3 x 3 x 256<br>2 x 2 | Encoder 3 |
| Encoder 5 | Conv. Block 5<br>Max Pooling 5 | 3 x 3 x 512<br>2 x 2 | Encoder 4 |
| Center | Conv. Block 6 | 3 x 3 x 1024 | Encoder 5 |
| Decoder 1 | Transpose. Block 1<br>Conv. Block 7 | 2 x 2 x 512<br>3 x 3 x 512 | Center, Encoder 5 |
| Decoder 2 | Transpose. Block 2<br>Conv. Block 8 | 2 x 2 x 256<br>3 x 3 x 256 | Decoder 1, Encoder 4 |
| Decoder 3 | Transpose. Block 3<br>Conv. Block 9 | 2 x 2 x 128<br>3 x 3 x 128 | Decoder 2, Encoder 3 |
| Decoder 4 | Transpose. Block 4<br>Conv. Block 10 | 2 x 2 x 64<br>3 x 3 x 64 | Decoder 3, Encoder 2 |
| Decoder 5 | Transpose. Block 5<br>Conv. Block 11 | 2 x 2 x 32<br>3 x 3 x 32 | Decoder 4, Encoder 1 |
| Output | Conv. Layer | 1 x 1 x 1 | Decoder 5 |

# 4 Experimental results

In this chapter, it is explained how the experiments were executed, and the results are presented. First, the ISIC 2017 Challange dataset is used and the results obtained are compared to the best results in the competition. Next, a case study is investigated using our own dataset created from PAD.

## 4.1 Pre-processing

Convolution neural networks can learn different types of filters, which is why they are so powerful. Relying on this premise, no filtering will be done on the images. The pre-processing done was cropping and resizing of the images, and the transformation of the color space. For the PAD dataset, a problem found due to the collection of images via smartphones was the freedom given to the user when taking the photo, even though the app gives the option to crop the image. As a consequence, several photos were taken outside of the standard, resulting in photos containing more than one lesion. So, it was necessary to crop all the images so that none of this was contained in the dataset. In addition, even though fully convoluted networks accept inputs of different sizes, all images are resized to 256x256, reducing computational cost. Finally, as described in Sec. 3.2, the images were converted to the chosen color space, i.e., RGB, HSV, CIE L*a*b and combinations between them.

## 4.2 Metrics

To evaluate and quantitatively compare the methodologies proposed in this work, we used some metrics widely used in the literature for segmentation [58]. Before defining such metrics, it is necessary to introduce the concept of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) in the domain of this work. In the segmentation of skin cancers, the lesions themselves are labeled as positive, and the part of the image where there is no lesion as negative. As shown in Fig. 4, where the white color represents the positive labels and the black the negative labels. Then, when a pixel is part of the lesion (positive label), and the proposed method classifies it as being part of the lesion (positive label), the so-called TP occurs. However, if the method classified this same pixel as not being part of the lesion (negative label), a case of FN would occur because the true pixel label is not negative. On the other hand, if a pixel is not part of the lesion (negative label), and the method classifies it as not part of the lesion (negative label), the TN occurs. However, if the method classified this same pixel as part of the lesion (positive

label), the so-called FP would happen because the true pixel label is not positive.

Based on these concepts, the metrics used in this work can be defined. The first one is the *sensitivity (SEN)*, which indicates the proportion of the pixels that are part of the lesion (positive label) and were correctly classified. The *sensitivity* is defined by the Eq. 4.1.

$$SEN = \frac{TP}{TP + FN} \tag{4.1}$$

Following the same logic, the *specificity (SPE)* informs the proportion of the pixels that are not part of the lesion (negative label) and were correctly classified. Eq. 4.2 defines the *specificity*.

$$SPE = \frac{TN}{TN + FP} \tag{4.2}$$

Next, to provide a more general view of the segmentation quality, it is used the *accuracy (ACC)*, defined by Eq. 4.3.

$$ACC = \frac{TP + TN}{TP + FN + TN + FP} \tag{4.3}$$

Two other metrics commonly used to quantify the quality of segmentation are the *Dice coefficient (DIC)* and the *Jaccard index (JAC)*. The dice coefficient measures how similar the segmentation of the lesions are when compared to the ground truths. The Jaccard index, on the other hand, is the intersection over union of the segmented lesion with the masks. The Jaccard index is the metric used to rank in the ISIC 2017 Challenge. These metrics are defined in Eqs. 4.4 and 4.5 respectively.

$$DIC = \frac{2.TP}{(2.TP) + FP + FN} \tag{4.4}$$

$$JAC = \frac{TP}{TP + FP + FN} \tag{4.5}$$

## 4.3 Training

The training of the network was carried out for 80 epochs, with batch size equal to 8. For the ISIC dataset, it was used the training partition provided in the competition, with 2.000 images. For the PAD dataset, the images were divided into 70% for training and 30% for testing. For all datasets, the training partition was divided into 80% for training and 20% for validation, where the model is chosen only when validation improvement is obtained. The tests using the PAD dataset were divided into three main methodologies.

The first, called direct transfer, is when the network is trained with the ISIC dataset (training + test) and tested directly with the PAD dataset. The second, called direct train, is when the network is trained with the PAD dataset. Finally, the fine-tune methodology, in which the network is trained with the ISIC dataset (training + test) and, with transfer learning, a fine-tune is done with the PAD dataset, then the tests are performed. The learning rate was set at 0.0001 for direct transfer and direct train methodologies, and at 0.00001 for fine-tune. The stochastic optimization method used was Adam, proposed in [59]. It is common that in skin cancer images, mainly in images taken by smartphones, the lesion occupies only a small portion of the image, which makes the number of positive labels (lesion) much smaller than the number of negative labels (not lesion). Experiments with unbalanced segmentations have shown that loss functions based on overlap measures present better results and appears to be more robust [60]. Therefore, based on these results, a combination between binary cross-entropy and dice loss is used for training the network in this work. This combination is described by the Eq. 4.6 below:

$$loss = \frac{1}{N}\sum_{n=1}^{N} y * log(\hat{y}) + (1-y) * log(1-\hat{y}) + 1 - \frac{\sum_{n=1}^{N} y * \hat{y} + \epsilon}{\sum_{n=1}^{N} y + \hat{y} + \epsilon} \tag{4.6}$$

where, $\frac{1}{N}\sum_{n=1}^{N} y * log(\hat{y}) + (1-y) * log(1-\hat{y})$ is the binary cross-entropy, $1 - \frac{\sum_{n=1}^{N} y*\hat{y}+\epsilon}{\sum_{n=1}^{N} y+\hat{y}+\epsilon}$ is the dice loss, $N$ is the number of pixels, $y$ and $\hat{y}$ are respectively the expected label and the score of a pixel, and $\epsilon$ is used to ensure the loss function stability by avoiding the numerical issue of dividing by 0.

## 4.4 Software and hardware

All code was implemented in Python using the TensorFlow framework, and are available at https://github.com/gabrielgiorisatto/segmentation_pad. The training and tests were run on two different computers, one containing an Intel (R) Core i7-6700 CPU @ 2.60 GHz, with 16 Gb of RAM and a NVIDIA GeForce GPU GTX 1070, and another containing an Intel (R) Core i7-7700 CPU @ 3.60 GHz, with 16 Gb of RAM and a NVIDIA Titan X GPU.

## 4.5 Results and discussions

In this section, it is presented the experimental results for the segmentation of skin cancers using the methodology described in Sec. 3. The experiments are divided into two parts. First, the ISIC 2017 Challenge dataset is used to perform the tests, and the results are compared with the results obtained by:

- Al-masni et al. [25]: The work proposes a full resolution convolutional networks (FrCN) method that learns the full resolution features of each individual pixel of the input data without the need for pre- or post-processing operations such as artifact removal, low contrast adjustment, or further enhancement of the segmented skin cancer boundaries. A comparison for the ISIC 2017 Challenge dataset is made between some latest deep learning segmentation approaches such as the fully convolutional network (FCN), U-Net, and SegNet.

- Yuan [24]: This work got the first place in the ISIC 2017 Challenge. The author proposes a framework based on deep fully convolutional-deconvolutional neural networks (CDNN). The network contains 29 layers with about 5M trainable parameters. Besides the original RGB channels, the author also includes the three channels in Hue-Saturation-Value color space, as well as the L channel (lightness) in CIELAB space. The loss function used was based on Jaccard distance. For post-processing, it was used a dual-thresholds method to generate a binary tumor mask from the CDNN output. A relatively high threshold (thH = 0.8) was firstly applied to determine the tumor center, which was calculated as the centroid of the region that has the largest mass among the candidates from thresholding. Then a lower threshold thL = 0.5 was applied to the output map. After filling small holes with morphological dilation, the final tumor mask was determined as the region that embraces the tumor center. Finally, a bagging-type ensemble strategy was implemented to combine outputs of 6 CDNNs to further improve the image segmentation performance on the testing images.

Next, a case study is done for the dataset collected during the dermatological assistance program (PAD). Sec. 3.1 describes the PAD dataset in more details.

To compare the methodologies, it was used the nonparametric Friedman followed by the Wilcoxon test. The Friedman test [61] verifies, in a group containing more than one sample, whether there are at least two samples representing populations with different median values. Thus, if the test returns a p-value lower than 0.05, there is statistical evidence that at least one of the sample is significantly different, that is, the null hypothesis for equality of medians between the populations is rejected [62]. Knowing that at least one method is statistically different from the other, it is necessary to do a pairwise Wilcoxon test between the methods. Similar to the Friedman test, the Wilcoxon test verifies the null hypothesis that two algorithms are similar. If the test returns the value of p less than 0.05, the null hypothesis is rejected, and there is statistical evidence that the two methods analyzed are significantly different [62]. In addition to these two statistical tests, as a way to complement the results, the A-TOPSIS algorithm proposed in [63] is used to select the best methods. The A-TOPSIS considers the mean and standard deviation of all metrics, weighted arbitrarily by the user, to rank the methods.

### 4.5.1 ISIC dataset

This section presents the results obtained with the ISIC 2017 Challenge dataset. As described in Sec. 3, the dataset consists of 2,000 dermoscopic images for training and 600 for testing. First, in order to check the best color space, the performance of various color space combinations for the segmentation of skin cancers is compared. Each combination is a concatenation between channels, for example, RGB-LAB means that all six channels are concatenated as input. Table 2 shows the mean and standard deviation obtained for each metric described in Sec. 4.2. Each experiment was run 10 times, and the training partition is shuffled for each time. The final segmentation without post-processing was obtained using a threshold of 0.55 for the U-Net output. In addition, for better visualization, Fig. 21 shows a boxplot of the Jaccard Index for each method, since it was the measure responsible for the ranking of the ISIC 2017 Challenge.

Table 2 – Segmentation performance of each color space combination with or without (w/o) post-processing for the ISIC data set. The best method ranked using A-TOPSIS and statistically significant based on the Jaccard index is in bold.

| C. Space | Post. | SEN | SPE | ACC | DIC | JAC |
|---|---|---|---|---|---|---|
| RGB | w/o | 0.7892 ± 0.0141 | 0.9788 ± 0.0031 | 0.9265 ± 0.0022 | 0.8322 ± 0.0088 | 0.7384 ± 0.0079 |
| | CRF | 0.7657 ± 0.0138 | 0.9808 ± 0.0029 | 0.9256 ± 0.0025 | 0.8241 ± 0.0099 | 0.7268 ± 0.0096 |
| HSV | w/o | 0.7948 ± 0.0157 | 0.9768 ± 0.0026 | 0.9238 ± 0.0019 | 0.8247 ± 0.0056 | 0.7315 ± 0.0080 |
| | CRF | 0.7732 ± 0.0180 | 0.9790 ± 0.0026 | 0.9233 ± 0.0021 | 0.8193 ± 0.0067 | 0.7242 ± 0.0108 |
| RGB-HSV | w/o | 0.7968 ± 0.0119 | 0.9760 ± 0.0035 | 0.9241 ± 0.0033 | 0.8280 ± 0.0080 | 0.7347 ± 0.0067 |
| | CRF | 0.7734 ± 0.0140 | 0.9785 ± 0.0036 | 0.9236 ± 0.0032 | 0.8219 ± 0.0080 | 0.7262 ± 0.0080 |
| LAB | w/o | 0.7940 ± 0.0190 | 0.9788 ± 0.0026 | 0.9274 ± 0.0026 | 0.8350 ± 0.0056 | 0.7417 ± 0.0112 |
| | CRF | 0.7661 ± 0.0181 | 0.9810 ± 0.0023 | 0.9264 ± 0.0028 | 0.8244 ± 0.0065 | 0.7275 ± 0.0129 |
| **RGB-LAB** | **w/o** | **0.8043±0.0157** | **0.9768±0.0039** | **0.9280±0.0023** | **0.8340±0.0069** | **0.7465±0.0089** |
| | CRF | 0.7852 ± 0.0155 | 0.9786 ± 0.0036 | 0.9275 ± 0.0024 | 0.8271 ± 0.0078 | 0.7397 ± 0.0100 |
| RGB-HSV-LAB | w/o | 0.7930 ± 0.0217 | 0.9763 ± 0.0052 | 0.9247 ± 0.0023 | 0.8275 ± 0.0061 | 0.7345 ± 0.0088 |
| | CRF | 0.7692 ± 0.0238 | 0.9788 ± 0.0048 | 0.9241 ± 0.0028 | 0.8202 ± 0.0079 | 0.7247 ± 0.0130 |
| RGB-HSV-L | w/o | 0.7929 ± 0.0130 | 0.9768 ± 0.0026 | 0.9248 ± 0.0016 | 0.8279 ± 0.0063 | 0.7341 ± 0.0079 |
| | CRF | 0.7700 ± 0.0138 | 0.9792 ± 0.0024 | 0.9242 ± 0.0015 | 0.8211 ± 0.0062 | 0.7249 ± 0.0101 |
| RGB-SV-LAB | w/o | 0.7911 ± 0.0198 | 0.9775 ± 0.0035 | 0.9266 ± 0.0034 | 0.8327 ± 0.0077 | 0.7381 ± 0.0106 |
| | CRF | 0.7679 ± 0.0210 | 0.9796 ± 0.0032 | 0.9257 ± 0.0035 | 0.8238 ± 0.0074 | 0.7268 ± 0.0132 |

Analyzing the results of Table 2 and the boxplot of figure 21, it is possible to note that the combination of colors that obtained the best Jaccard index was RGB-LAB. Also, it is interesting to note that the use of post-processing worsened the result in terms of the Jaccard index for all cases. In order to understand the reason for this event, it is necessary to visualize the output generated by the post-processing and compare it to the expected mask. Fig. 22 shows some comparisons between the results obtained without and with the CRF as post-processing. It is possible to see that the original masks from the 2017 ISIC Challenge dataset do not have contours close to the border of the lesion. However, the idea of using CRF is to obtain a result that has a contour as closer as possible to the border of the lesion. As the measures depends on the original mask, the results obtained with CRF appear to be quantitatively worse but visually better. To further complement

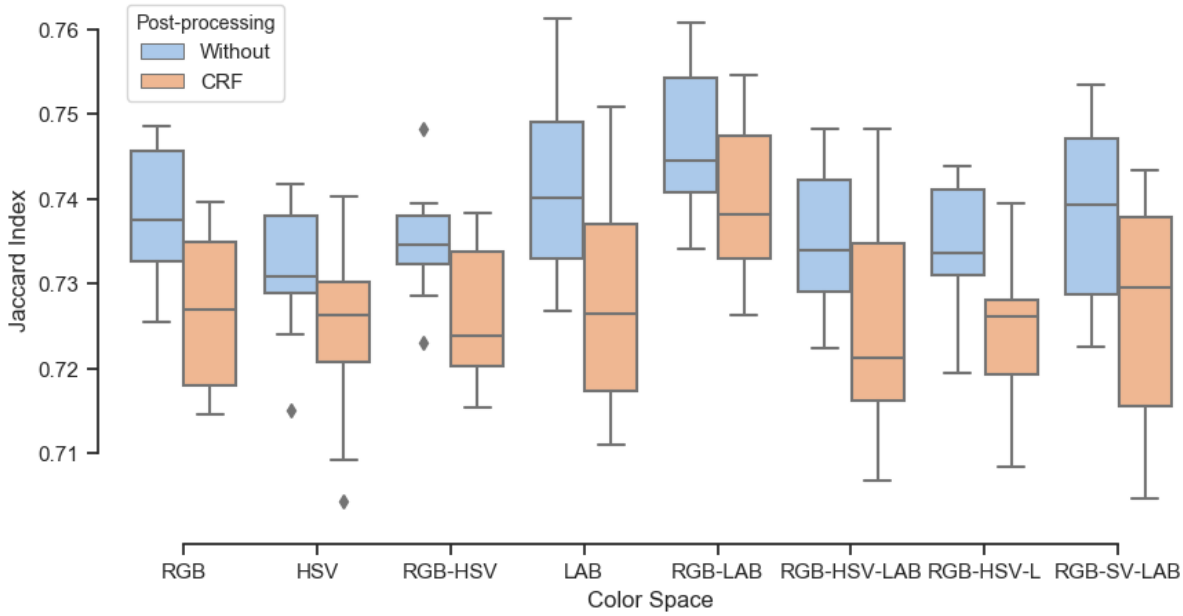



Fig. 21. Boxplot for the performance of each color space combination with or without (w/o) post-processing based on the Jaccard index for the ISIC dataset.

this issue, Fig. 23 presents a comparison of results obtained using different thresholds for the RGB-LAB combination. The lower the threshold, the lower the confidence required for the pixel to be classified as part of the lesion, so the lower the threshold, the less the prediction is tied to the border of the lesion, confirming the same problem described. Since the score output of the network is commonly greater than 0.8, the performance difference between values less than 0.8 is not as great as between 0.8 and 0.9, as shown in Fig. 23.

The Friedman test was applied to the results of Table 2 and it was obtained $p < 0.05$, so there is at least one method significantly different from the others. Then, as the best result visually obtained was the combination of RGB-LAB channels, the Wilcoxon test was done between this method and all the others to verify the statistical significance. Table 3 shows the result of this test, and it is possible to notice that only the comparisons with RGB and with LAB obtained $p > 0.05$, which means that it is not possible to assert the statistical significance between the RGB-LAB and these methods.

Finally, as a way of complementing the statistical results, the A-TOPSIS algorithm was used to generate a ranking of the methods. To use A-TOPSIS it is necessary to consider the mean and standard deviation of the performance of the algorithms. Table 4 describes the variation of the first 5 methods in ranking giving equal weight to both until disregarding the standard deviation. In all cases, the RGB-LAB was placed first, which confirms the previous results, reinforcing the idea that this is the best combination of channels. In order to represent the ranking visually, a weighting of 60% for mean and 40% for the standard deviation was chosen and the ranking obtained is shown in Fig. 24.

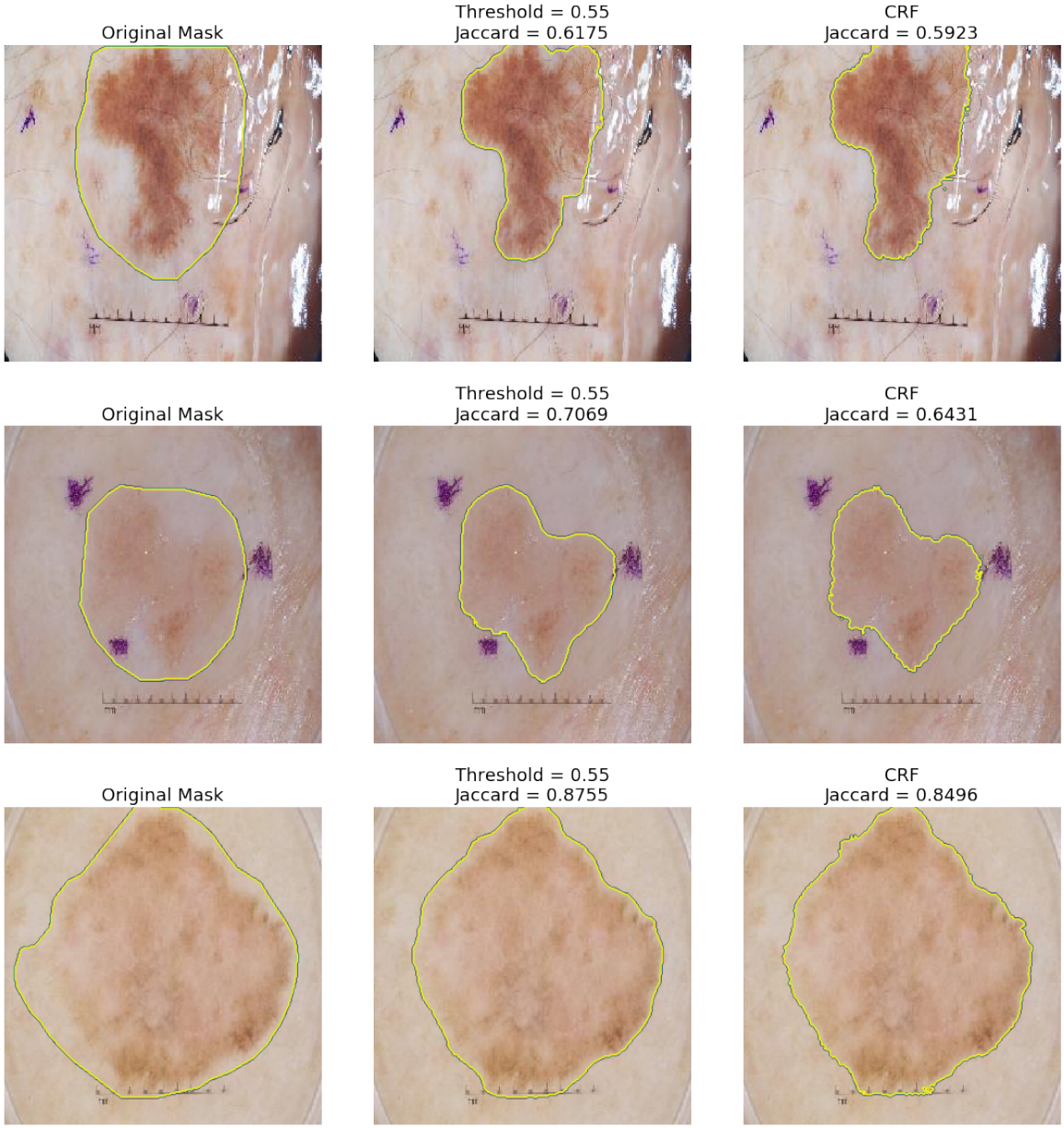



Fig. 22. Examples of the problem described about the masks contained in the ISIC 2017 Challenge dataset. From left to right: the original mask for the image; the mask obtained without post-processing using 0.55 as threshold; the mask obtained with CRF as the post-processing. For each mask obtained there is the Jaccard index resulted. The yellow line is the contour of the mask.

To compare the results obtained in this work with those obtained by the works described at the beginning of this section, only the Jaccard index will be used, since this was the measure used to rank participants in the ISIC 2017 Challenge. Therefore, the chosen methodology of this work was the combination of RGB-LAB channels, as it obtained the best Jaccard index. Moreover, due to the peculiarity previously described about the masks, the result used will be with the threshold of 0.2, without post-processing. Table 5 shows that the method used in this work is comparable with those in the literature,

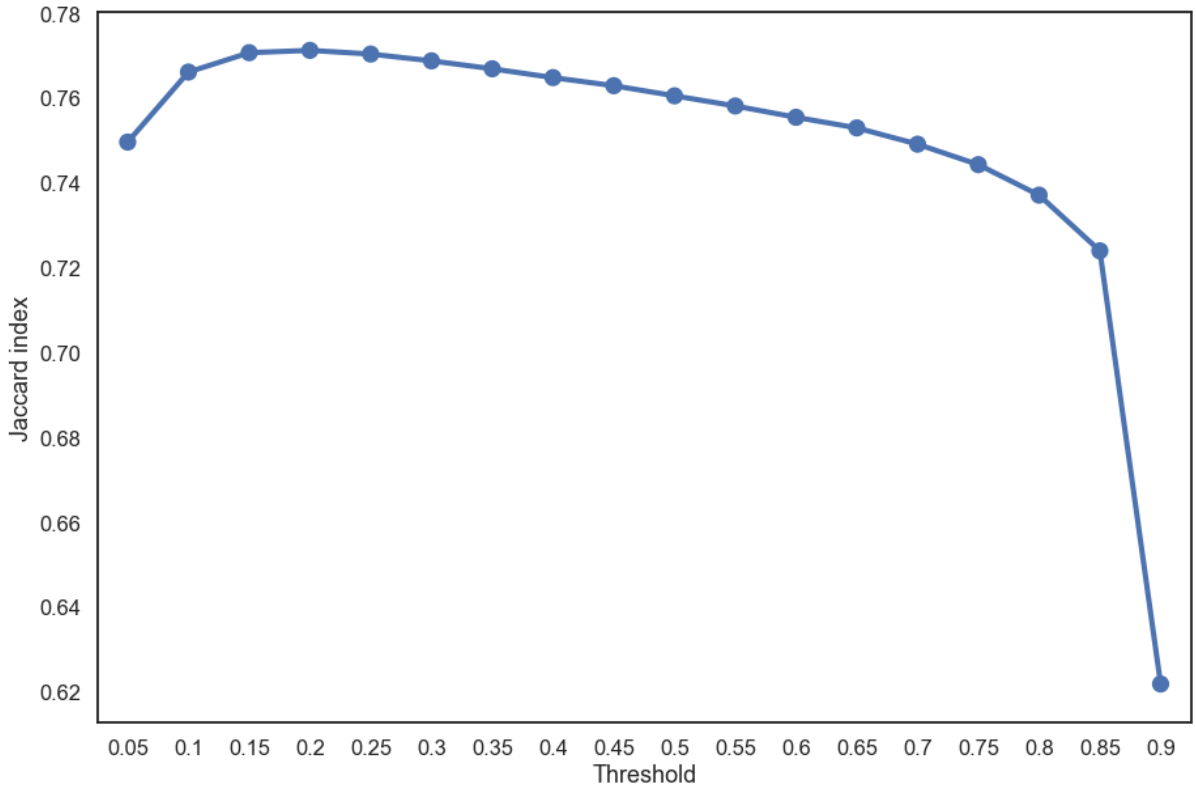



Fig. 23. The figure shows the graph of the Jaccard index in relation to various threshold values for the RGB-LAB methodology. As previously described, this image helps to visualize that the less the segmentation is tied to the lesion border, the higher the Jaccard index, since the final masks using low thresholds are more disperse.

Table 3 – Wilcoxon test between the RGB-LAB and the methods used in this work.

| Method compared | $p$ |
|---|---|
| RGB | **0.05933** |
| RGB + CRF | 0.00691 |
| HSV | 0.00691 |
| HSV + CRF | 0.00506 |
| RGB-HSV | 0.02841 |
| RGB-HSV + CRF | 0.00691 |
| LAB | **0.16880** |
| LAB + CRF | 0.01251 |
| RGB-LAB + CRF | 0.00506 |
| RGB-HSV-LAB | 0.03665 |
| RGB-HSV-LAB + CRF | 0.00934 |
| RGB-HSV-L | 0.01660 |
| RGB-HSV-L + CRF | 0.00691 |
| RGB-SV-LAB | 0.02841 |
| RGB-SV-LAB + CRF | 0.00506 |

having a Jaccard index superior to the first place of the competition [24], and also superior to other techniques used in the literature, having a result only inferior to that obtained by

Table 4 – Ranking of the classifiers obtained by A-TOPSIS for the ISIC 2017 Challenge dataset varying the weights of mean and standard deviation.

| [mean, std deviation] | Ranking |
|---|---|
| [0.5, 0.5] | RGB-LAB ≺ RGB-HSV ≺ RGB-HSV-L ≺ RGB ≺ LAB |
| [0.6, 0.4] | RGB-LAB ≺ RGB-HSV ≺ RGB-HSV-L ≺ RGB ≺ LAB |
| [0.7, 0.3] | RGB-LAB ≺ LAB ≺ RGB-HSV ≺ RGB ≺ RGB-HSV-L |
| [0.8, 0.2] | RGB-LAB ≺ LAB ≺ RGB-HSV ≺ RGB-SV-LAB ≺ RGB |
| [0.9, 0.1] | RGB-LAB ≺ LAB ≺ RGB-HSV ≺ RGB-SV-LAB ≺ RGB |
| [1.0, 0.0] | RGB-LAB ≺ LAB ≺ RGB-HSV ≺ RGB-SV-LAB ≺ RGB |

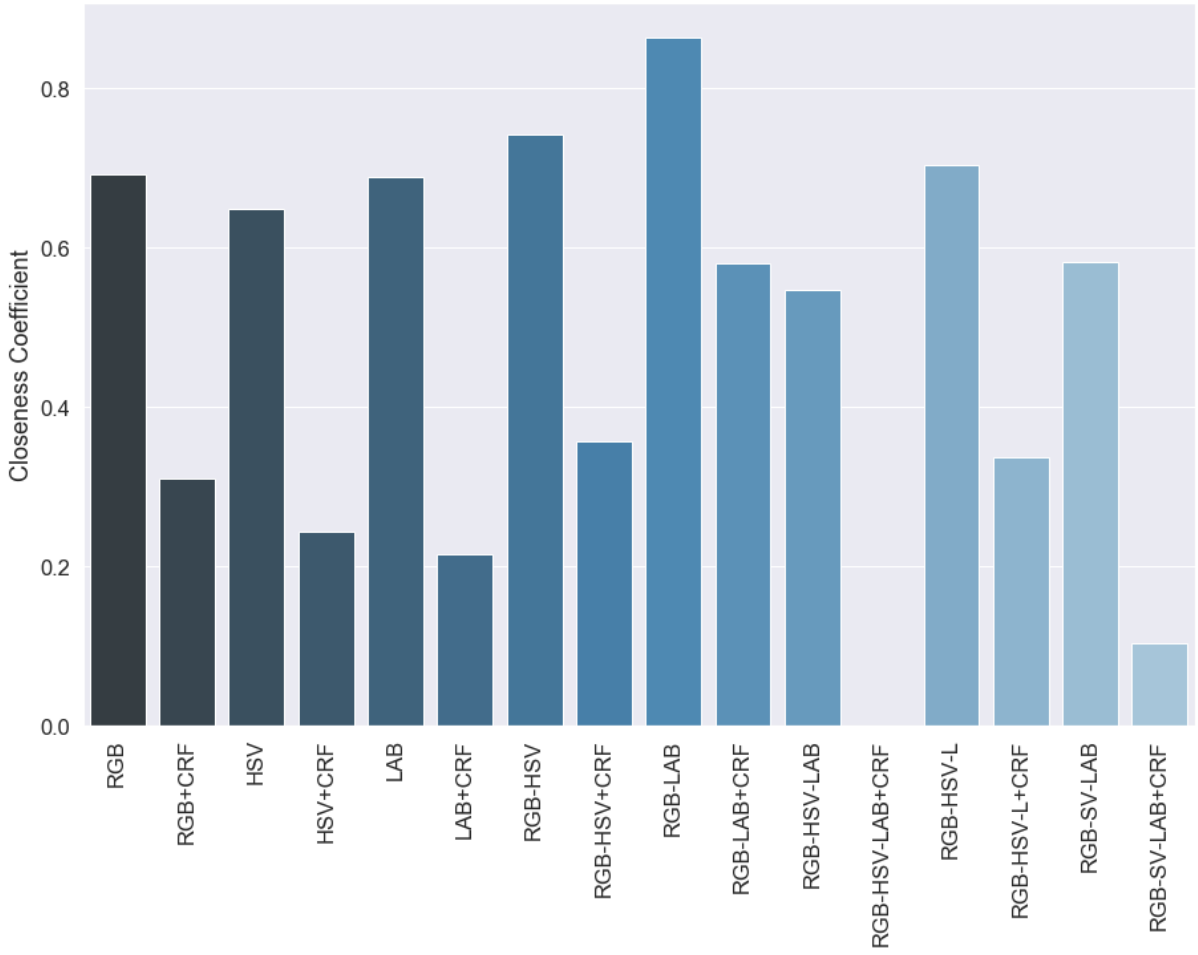



Fig. 24. Ranking of methods obtained by A-TOPSIS for the ISIC 2017 Challenge dataset. The best methodology in this rank is the RGB-LAB.

the FrCN method, proposed in [25].

### 4.5.2 PAD dataset - study case

This section presents the results obtained with the PAD dataset. As described in Sec. 3.1, the dataset has 258 skin cancer images taken from smartphones for training and 111 for testing. In the same way as in the previous section, in order to check the best color

Table 5 – Segmentation performance of the proposed method compared to methods presented in [24], [25] for the ISIC 2017 Challenge dataset.

| Method | Jaccard Index |
|---|---|
| Proposed Method | 0.771285 |
| FCN [25] | 0.7663 |
| U-Net [25] | 0.6972 |
| SegNet [25] | 0.7497 |
| FrCN [25] | 0.8128 |
| CDNN [24] | 0.765 |

space, the performance of various color space combinations for the segmentation of skin cancers is compared. However, as described in Sec. 3.4, there three different methodologies for training the network for the PAD dataset. Direct train uses only the data from PAD dataset, direct transfer trains the network only with the ISIC 2017 Challenge and fine-tune does a transfer learning from the ISIC dataset to the PAD dataset. Table 6 shows the mean and standard deviation obtained for each metric described in Sec. 4.2. Each experiment was run 10 times and the final segmentation without post-processing was obtained using a threshold of 0.55 for the U-Net output. In addition, for better visualization, Fig. 25 shows a boxplot of the Jaccard Index for each method, since it was the measure responsible for the ranking of the ISIC 2017 Challenge.

Table 6 – Segmentation performance of each color space combination with or without (w/o) post-processing for the PAD dataset. It is also presented the results using direct transfer (DF), direct train (DT) and fine tune (FN). The best method ranked using A-TOPSIS and statistically significant based on the Jaccard index is in bold.

Begin of Table

| C. Space | Method | Post. | SEN | SPE | ACC | DIC | JAC |
|---|---|---|---|---|---|---|---|
| RGB | DF | w/o | 0.9074 ± 0.0234 | 0.8083 ± 0.0271 | 0.8163 ± 0.0239 | 0.5805 ± 0.0229 | 0.4695 ± 0.0216 |
| | | CRF | 0.8915 ± 0.0268 | 0.8143 ± 0.0283 | 0.8207 ± 0.0249 | 0.5971 ± 0.0238 | 0.4919 ± 0.0217 |
| | DT | w/o | 0.8622 ± 0.0215 | 0.9689 ± 0.0036 | 0.9512 ± 0.0021 | 0.7496 ± 0.0074 | 0.6350 ± 0.0079 |
| | | CRF | 0.8414 ± 0.0288 | 0.9748 ± 0.0039 | 0.9549 ± 0.0021 | 0.7629 ± 0.0082 | 0.6552 ± 0.0084 |
| | FT | w/o | 0.8696 ± 0.0133 | 0.9701 ± 0.0034 | 0.9542 ± 0.0021 | 0.7634 ± 0.0067 | 0.6541 ± 0.0087 |
| | | CRF | 0.8591 ± 0.0135 | 0.9738 ± 0.0031 | 0.9567 ± 0.0021 | 0.7731 ± 0.0070 | 0.6676 ± 0.0080 |
| HSV | DF | w/o | 0.8946 ± 0.0256 | 0.8384 ± 0.0364 | 0.8419 ± 0.0317 | 0.6060 ± 0.0294 | 0.4967 ± 0.0274 |
| | | CRF | 0.8765 ± 0.0261 | 0.8446 ± 0.0391 | 0.8463 ± 0.0340 | 0.6213 ± 0.0315 | 0.5189 ± 0.0288 |
| | DT | w/o | 0.8789 ± 0.0180 | 0.9568 ± 0.0044 | 0.9426 ± 0.0029 | 0.7296 ± 0.0093 | 0.6137 ± 0.0091 |
| | | CRF | 0.8614 ± 0.0226 | 0.9637 ± 0.0037 | 0.9475 ± 0.0026 | 0.7458 ± 0.0137 | 0.6363 ± 0.0131 |
| | FT | w/o | 0.8450 ± 0.0126 | 0.9712 ± 0.0038 | 0.9521 ± 0.0024 | 0.7534 ± 0.0071 | 0.6425 ± 0.0071 |
| | | CRF | 0.8297 ± 0.0162 | 0.9750 ± 0.0041 | 0.9543 ± 0.0027 | 0.7602 ± 0.0104 | 0.6542 ± 0.0096 |

Continuation of Table 6

| C. Space | Method | Post. | SEN | SPE | ACC | DIC | JAC |
|---|---|---|---|---|---|---|---|
| RGB-HSV | DF | w/o | 0.9085 ± 0.0205 | 0.8189 ± 0.0306 | 0.8254 ± 0.0264 | 0.5934 ± 0.0221 | 0.4856 ± 0.0217 |
| | | CRF | 0.8921 ± 0.0255 | 0.8243 ± 0.0316 | 0.8291 ± 0.0271 | 0.6080 ± 0.0223 | 0.5055 ± 0.0217 |
| | DT | w/o | 0.8634 ± 0.0322 | 0.9615 ± 0.0100 | 0.9457 ± 0.0065 | 0.7347 ± 0.0134 | 0.6207 ± 0.0136 |
| | | CRF | 0.8443 ± 0.0354 | 0.9675 ± 0.0095 | 0.9497 ± 0.0058 | 0.7479 ± 0.0122 | 0.6393 ± 0.0121 |
| | FT | w/o | 0.8562 ± 0.0108 | 0.9702 ± 0.0038 | 0.9524 ± 0.0028 | 0.7549 ± 0.0101 | 0.6461 ± 0.0101 |
| | | CRF | 0.8441 ± 0.0122 | 0.9738 ± 0.0032 | 0.9546 ± 0.0023 | 0.7614 ± 0.0099 | 0.6563 ± 0.0094 |
| LAB | DF | w/o | 0.8985 ± 0.0133 | 0.8218 ± 0.0240 | 0.8280 ± 0.0210 | 0.6010 ± 0.0195 | 0.4910 ± 0.0194 |
| | | CRF | 0.8656 ± 0.0161 | 0.8242 ± 0.0256 | 0.8299 ± 0.0220 | 0.6030 ± 0.0194 | 0.4972 ± 0.0191 |
| | DT | w/o | 0.8854 ± 0.0182 | 0.9622 ± 0.0087 | 0.9483 ± 0.0064 | 0.7491 ± 0.0148 | 0.6346 ± 0.0149 |
| | | CRF | 0.8565 ± 0.0252 | 0.9694 ± 0.0078 | 0.9525 ± 0.0055 | 0.7617 ± 0.0160 | 0.6511 ± 0.0176 |
| | FT | w/o | 0.8609 ± 0.0154 | 0.9707 ± 0.0032 | 0.9543 ± 0.0026 | 0.7653 ± 0.0065 | 0.6572 ± 0.0068 |
| | | CRF | 0.8328 ± 0.0228 | 0.9754 ± 0.0030 | 0.9563 ± 0.0025 | 0.7670 ± 0.0099 | 0.6600 ± 0.0115 |
| **RGB-LAB** | DF | w/o | 0.8586 ± 0.0428 | 0.8681 ± 0.0315 | 0.8643 ± 0.0263 | 0.6248 ± 0.0315 | 0.5150 ± 0.0307 |
| | | CRF | 0.8772 ± 0.0383 | 0.8445 ± 0.0346 | 0.8461 ± 0.0295 | 0.6202 ± 0.0320 | 0.5143 ±0.0316 |
| | DT | w/o | 0.8807 ± 0.0355 | 0.9609 ± 0.0087 | 0.9468 ± 0.0046 | 0.7385 ± 0.0113 | 0.6229 ± 0.0138 |
| | | CRF | 0.8630 ± 0.0435 | 0.9676 ± 0.0081 | 0.9515 ± 0.0040 | 0.7551 ± 0.0125 | 0.6456 ± 0.0130 |
| | **FT** | w/o | 0.8729 ± 0.0455 | 0.9699 ± 0.0338 | 0.9545 ± 0.0281 | 0.7653 ± 0.0326 | 0.6566 ± 0.0310 |
| | | **CRF** | **0.8607±0.0406** | **0.9736±0.0373** | **0.9569±0.0317** | **0.7732±0.0335** | **0.6685±0.0324** |
| RGB-HSV-LAB | DF | w/o | 0.9222 ± 0.0220 | 0.8149 ± 0.0325 | 0.8232 ± 0.0279 | 0.5886 ± 0.0228 | 0.4790 ± 0.0209 |
| | | CRF | 0.9069 ± 0.0256 | 0.8212 ± 0.0330 | 0.8278 ± 0.0284 | 0.6047 ± 0.0229 | 0.5003 ± 0.0212 |
| | DT | w/o | 0.8675 ± 0.0215 | 0.9630 ± 0.0081 | 0.9470 ± 0.0046 | 0.7383 ± 0.0121 | 0.6228 ± 0.0120 |
| | | CRF | 0.8477 ± 0.0246 | 0.9694 ± 0.0077 | 0.9512 ± 0.0041 | 0.7518 ± 0.0123 | 0.6420 ± 0.0115 |
| | FT | w/o | 0.8554 ± 0.0124 | 0.9708 ± 0.0027 | 0.9535 ± 0.0020 | 0.7578 ± 0.0073 | 0.6484 ± 0.0083 |
| | | CRF | 0.8413 ± 0.0125 | 0.9745 ± 0.0020 | 0.9557 ± 0.0017 | 0.7639 ± 0.0082 | 0.6582 ± 0.0086 |
| RGB-HSV-L | DF | w/o | 0.9112 ± 0.0262 | 0.8261 ± 0.0369 | 0.8323 ± 0.0313 | 0.6031 ± 0.0237 | 0.4951 ± 0.0221 |
| | | CRF | 0.8951 ± 0.0281 | 0.8317 ± 0.0387 | 0.8362 ± 0.0328 | 0.6192 ± 0.0217 | 0.5167 ± 0.0199 |
| | DT | w/o | 0.8674 ± 0.0293 | 0.9609 ± 0.0098 | 0.9447 ± 0.0057 | 0.7339 ± 0.0142 | 0.6184 ± 0.0160 |
| | | CRF | 0.8455 ± 0.0326 | 0.9679 ± 0.0085 | 0.9495 ± 0.0046 | 0.7475 ± 0.0124 | 0.6384 ± 0.0127 |
| | FT | w/o | 0.8573 ± 0.0216 | 0.9692 ± 0.0052 | 0.9515 ± 0.0030 | 0.7542 ± 0.0068 | 0.6441 ± 0.0069 |
| | | CRF | 0.8437 ± 0.0259 | 0.9729 ± 0.0050 | 0.9537 ± 0.0029 | 0.7597 ± 0.0105 | 0.6531 ± 0.0102 |
| RGB-SV-LAB | DF | w/o | 0.9165 ± 0.0145 | 0.8008 ± 0.0167 | 0.8099 ± 0.0155 | 0.5793 ± 0.0141 | 0.4696 ± 0.0129 |
| | | CRF | 0.9007 ± 0.0154 | 0.8054 ± 0.0186 | 0.8131 ± 0.0173 | 0.5939 ± 0.0175 | 0.4895 ± 0.0158 |
| | DT | w/o | 0.8616 ± 0.0234 | 0.9666 ± 0.0061 | 0.9499 ± 0.0033 | 0.7453 ± 0.0128 | 0.6299 ± 0.0150 |
| | | CRF | 0.8429 ± 0.0273 | 0.9727 ± 0.0054 | 0.9539 ± 0.0028 | 0.7604 ± 0.0109 | 0.6510 ± 0.0118 |
| | FT | w/o | 0.8648 ± 0.0134 | 0.9695 ± 0.0036 | 0.9532 ± 0.0026 | 0.7594 ± 0.0091 | 0.6507 ± 0.0108 |
| | | CRF | 0.8528 ± 0.0130 | 0.9729 ± 0.0033 | 0.9555 ± 0.0024 | 0.7656 ± 0.0098 | 0.6608 ± 0.0113 |

Analyzing the results of Table 6 and the boxplot of figure 25, it is possible to note that, as for the ISIC dataset, RGB and RGB-LAB also had top results. In addition, it is interesting to note that the use of post-processing in this case improved the Jaccard index results for all cases. This is due to the fact that the masks marked in this dataset are tighter to the edges of the lesion, as shown in Fig. 26, favoring the results obtained with the CRF. Another point to emphasize is the importance of the information coming from the images taken by smartphones. The results of the direct transfer (DF) were much worse than the results of direct training (DT) and fine-tune (FT). This is mainly due to two factors, the first is the way the masks are made. Segmentation is a subjective task, so training the network in a domain marked by certain experts, and testing in another

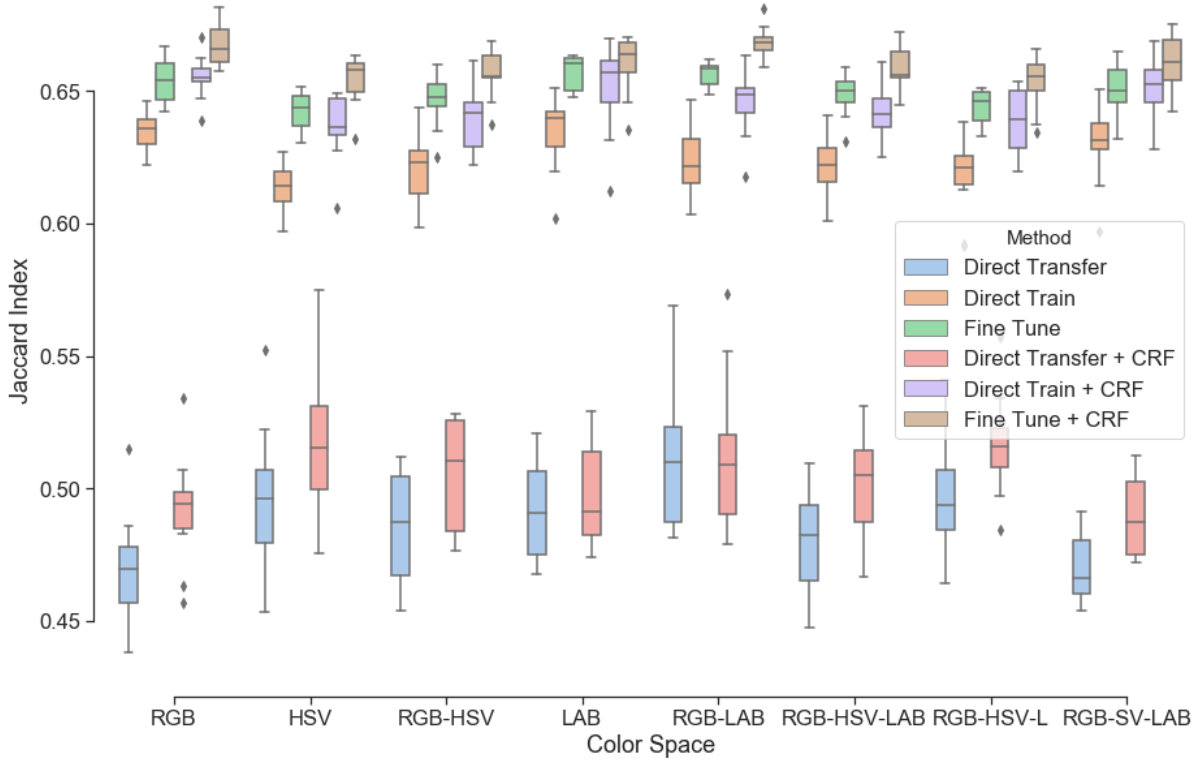



Fig. 25. Boxplot of the Jaccard index obtained from each method for the PAD dataset.

domain where masks were made by different people may not work. The second factor is the image itself, while the dermoscopic images are well controlled, do not have as much variation of luminosity and are very clear, the photos taken by smartphones, on the other hand, are much more sensitive to the variations of luminosity, have different zoom levels, can be taken with the flash on, and have a high chance of being blurred. Finally, it can be seen in Fig. 25 that fine-tune improved performance in terms of the Jaccard for all methods, showing the importance of transfer learning.

Analogously to the first part of the experiments, the Friedman test was applied and obtained p-value much smaller then 0.05, indicating that at least one algorithm is significantly different. Once again, the Wilcoxon test is performed, now for the RGB-LAB + CRF using the fine-tune methodology, and the result are shown in Table 7 only for the comparisons where the p-value obtained was greater then 0.05.

Table 7 – Results in which the Wilcoxon test with the RGB-LAB + CRF + FT method resulted in $p > 0.05$.

| Method compared | p |
|---|---|
| RGB + CRF + FT | 0.72127 |
| LAB + CRF + FT | 0.11412 |
| RGB-SV-LAB + CRF + FT | 0.24112 |

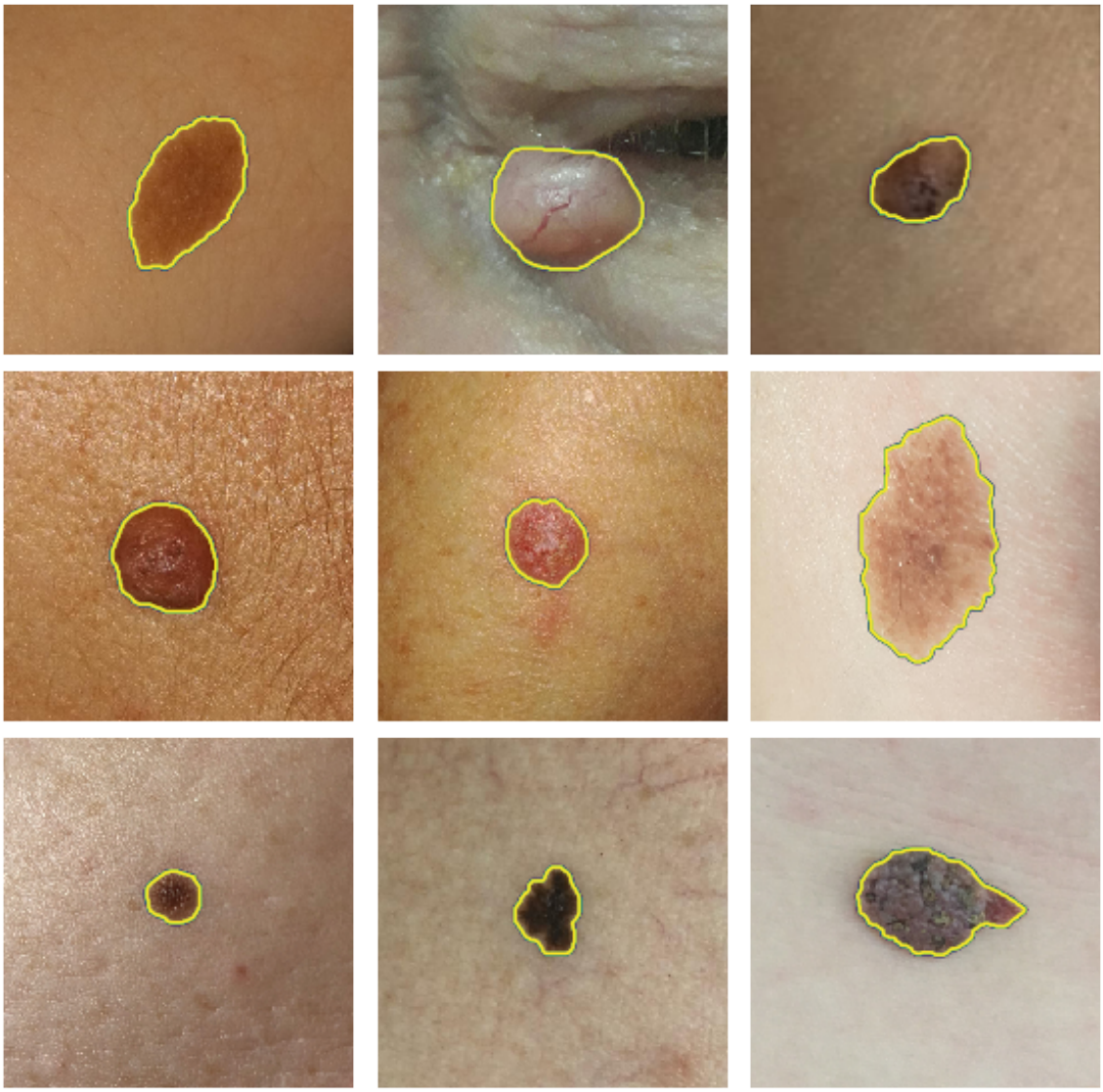


Fig. 26. Examples of ground-truth from the PAD dataset. It is possible to note that the contour, represented by the yellow line, is tied to the lesions border.

The statistical test was again complemented with the A-TOPSIS algorithm. Table 8 describes the ranking variation according to the mean weights and standard deviation of the performance. Due to the large number of methods, only the best 5 is displayed for each variation. In all cases, the RGB-LAB was placed first, which confirms the previous results, reinforcing the idea that this is the best combination of channels. In order to represent the ranking visually, a weighting of 60% for mean and 40% for the standard deviation was chosen and the ranking obtained is shown in Fig. 27.

As the dataset is still being created, and this is a pioneering work using it, there are no results to be compared. Therefore, one way of verifying the segmentation quality, in addition to the metrics already presented, is by visual inspection. Fig. 28 shows some segmentation obtained using the RGB-LAB color space, with post-processing and fine-tune.

Table 8 – Ranking of methods obtained by A-TOPSIS for the PAD dataset.

| [mean, std. dev.] | Ranking |
|---|---|
| [0.5, 0.5] | RGB-LAB+FT+CRF ≺ RGB-LAB+FT ≺ RGB-LAB+DT+CRF ≺ RGB-LAB+DT ≺ RGB-HSV-L+FT |
| [0.6, 0.4] | RGB-LAB+FT+CRF ≺ RGB-LAB+FT ≺ RGB-LAB+DT+CRF ≺ RGB-LAB+DT ≺ RGB-HSV-L+FT |
| [0.7, 0.3] | RGB-LAB+FT+CRF ≺ RGB-LAB+FT ≺ RGB-LAB+DT+CRF ≺ RGB-LAB+DT ≺ RGB-HSV-L+FT |
| [0.8, 0.2] | RGB-LAB+FT+CRF ≺ RGB-LAB+FT ≺ RGB-LAB+DT+CRF ≺ RGB-HSV-L+FT ≺ RGB-LAB+DT |
| [0.9, 0.1] | RGB-LAB+FT ≺ RGB-LAB+FT+CRF ≺ RGB-LAB+DT+CRF ≺ RGB+FT ≺ RGB+FT+CRF |
| [1.0, 0.0] | RGB-LAB+FT ≺ RGB+FT ≺ RGB-LAB+FT+CRF ≺ RGB+FT+CRF ≺ LAB+DT ≺ LAB+FT |

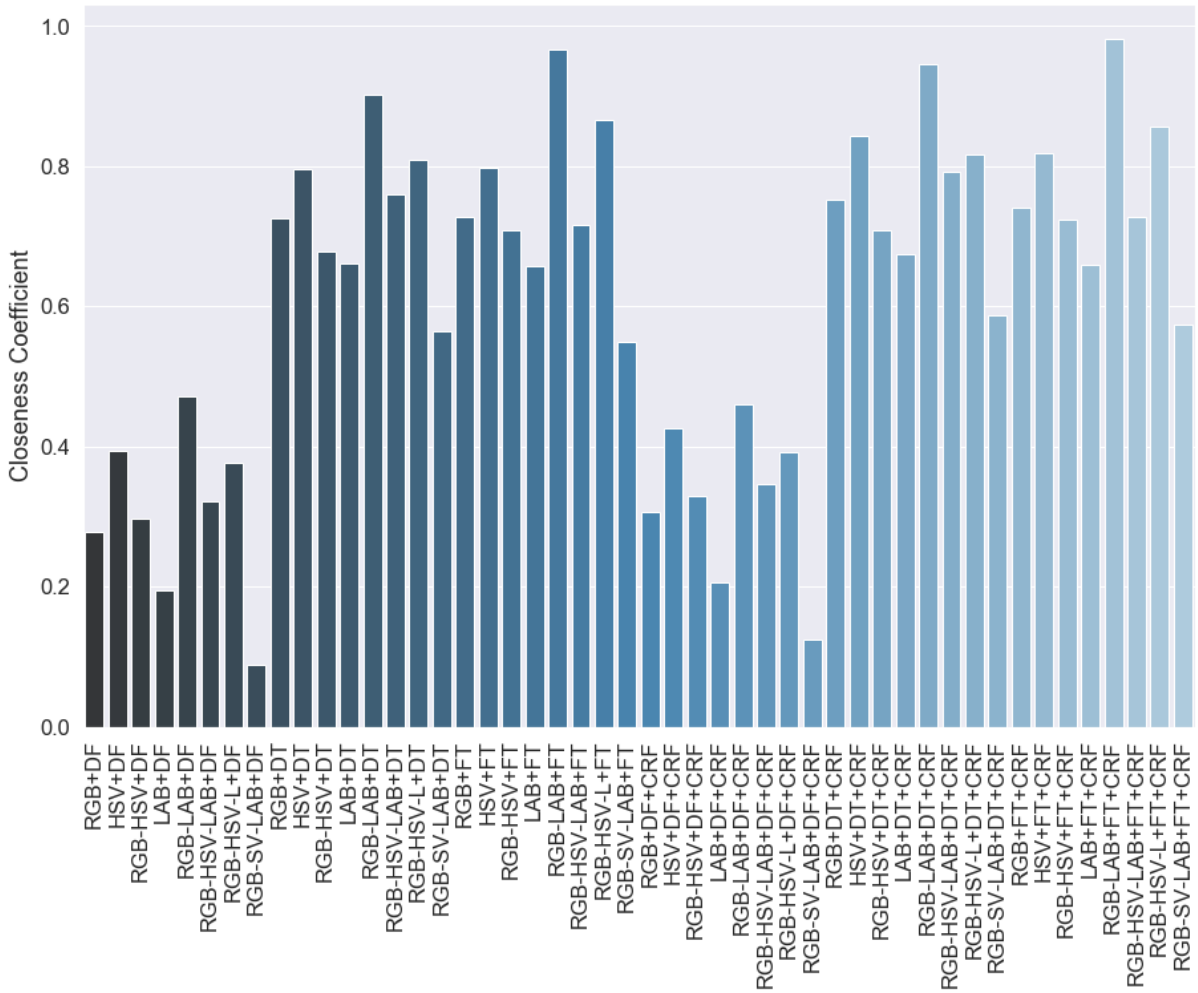



Fig. 27. Ranking of methods obtained by A-TOPSIS for the PAD dataset. The best methodology in this rank is the RGB-LAB+FT+CRF.

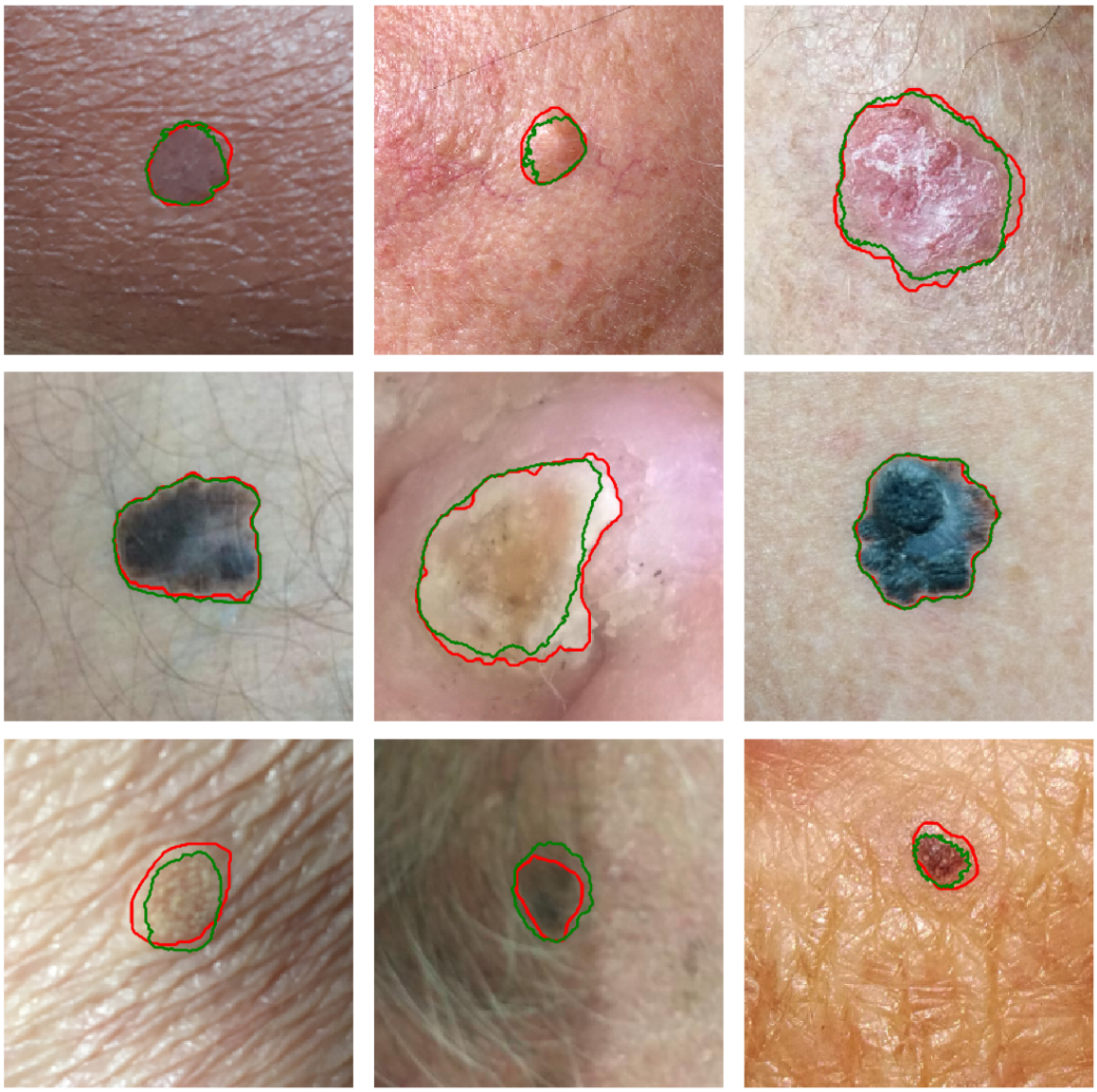


Fig. 28. Some examples of segmentation results obtained using the RGB-LAB, CRF, and fine-tuning color space for the PAD data set. The green line is the expected contour and the red line is the output obtained using the method. It may be noted that even in difficult scenarios, as some examples have shadows, low contrast, hair and different light conditions, the segmentation still performs well.

# 5 Conclusion

This work presented a study on skin cancer segmentation using the U-Net convolutional neural network. A new dataset of skin cancer images taken by smartphones was presented. Different combinations of color space and post-processing were proposed. First, the results were obtained using the existing ISIC dataset. An analysis of the results was then made and the problem of lesion marking was discussed in detail. After that, the results were compared to each other and the best method was compared with the results reported in the literature. To rank the methods proposed in this work, the A-TOPSIS algorithm was used, in addition to the statistical tests. Next, a real-world case study was investigated using the dataset described in this work, and the results were compared using the same methods previously used, but including learning transfer methodologies. The combination of RGB-LAB channels was the one that obtained the best results during all the experiments. The transfer learning, as well as the post-processing, improved practically all the results, showing the importance of these techniques. However, the results could not be compared, since the database is still being created. Therefore, to avoid the problems described in relation to the masks, it is necessary a greater validation of the dataset by specialists.

As future work, we intend to investigate the performance of other deep learning methods for segmentation, including ensembles, for the database presented in this work. In addition, in order to consolidate the PAD dataset, we intend to validate the data with greater confidence with the help of specialists in dermatology. It is also intended to increase the amount of data present in the database.

# Bibliography


[1] Y. Shoham, R. Perrault, E. Brynjolfsson, J. Clark, J. Manyika, and C. LeGassick, "AI Index 2017 Annual Report", Stanford University, Stanford, California 94305, Tech. Rep., Nov. 2017.

[2] N. Cascinelli, M. Ferrario, T. Tonelli, and E. Leo, "A possible new tool for clinical diagnosis of melanoma: the computer", *J Am Acad Dermatol*, vol. 16, no. 2, pp. 361–367, 1987.

[3] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks", *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[4] B. A. of Dermatologists. (2018). Skin cancer, [Online]. Available: http://www.bad.org.uk/for-the-public/skin-cancer/ (visited on 06/20/2018).

[5] F. J, S. I, E. M, and D. R. (2013). Cancer Incidence and Mortality Worldwide: IARC CancerBase No. 11. F. I. A. f. R. o. C. Lyon, Ed., [Online]. Available: http://globocan.iarc.fr (visited on 06/20/2018).

[6] INCA, "Estimativa 2018-Incidência de câncer no Brasil", Instituto Nacional de Câncer José Alencar Gomes da Silva, Rio de Janeiro, Tech. Rep., 2017.

[7] K. Lacy and W. Alwan, "Skin cancer", *Medicine (United Kingdom)*, vol. 41, no. 7, pp. 402–405, 2013.

[8] A. B. Cognetta, T. Vogt, M. Landthaler, O. Braun-Falco, and G. Plewig, "The ABCD rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions", *Journal of the American Academy of Dermatology*, vol. 30, no. 4, pp. 551–559, 1994.

[9] H. M. Shaw, D. S. Rigel, R. J. Friedman, W. H. Mccarthy, and A. W. Kopf, "Early Diagnosis of Cutaneous Melanoma: Revisiting the ABCD Criteria", vol. 292, no. 22, pp. 2771–2776, 2004.

[10] R. H. Johr, "Dermoscopy: Alternative melanocytic algorithms - The ABCD rule of dermatoscopy, menzies scoring method, and 7-point checklist", *Clinics in Dermatology*, vol. 20, no. 3, pp. 240–247, 2002.

[11] M. F. Healsmith, J. F. Bourke, J. E. Osborne, and R. A. C. Grahambrown, "An Evaluation of the Revised 7-Point Checklist for the Early Diagnosis of Cutaneous Malignant-Melanoma", *British Journal of Dermatology*, vol. 130, no. 1, pp. 48–50, 1994.

[12] I. Zalaudek, G. Argenziano, H. P. Soyer, R. Corona, F. Sera, A. Blum, R. P. Braun, H. Cabo, G. Ferrara, A. W. Kopf, D. Langford, S. W. Menzies, G. Pellacani, K. Peris, and S. Seidenari, "Three-point checklist of dermoscopy: An open internet study", *British Journal of Dermatology*, vol. 154, no. 3, pp. 431–437, 2006.

[13] S. W. Menzies, C. Ingvar, K. A. Crotty, and W. H. Mccarthy, "Frequency and Morphologic Characteristics of Invasive Melanomas Lacking Specific Surface Microscopic Features", *Archives of Dermatology*, no. 10, pp. 1178–1182, 1996.

[14] G. Argenziano and H. P. Soyer, "Dermoscopy of pigmented skin lesions – a valuable tool for early diagnosis of melanoma", *The Lancet Oncology*, vol. 2, no. 7, pp. 443–449, 2001.

[15] A. Masood and A. A. Al-Jumaily, "Computer aided diagnostic support system for skin cancer: A review of techniques and algorithms", *International Journal of Biomedical Imaging*, vol. 2013, p. 22, 2013.

[16] M. E. Celebi, H. Iyatomi, G. Schaefer, and W. V. Stoecker, "Lesion Border Detection in Dermoscopy Images", *Computerized Medical Imaging and Graphics*, no. 33, pp. 148–153, 2009.

[17] Q. Abbas, M. E. Celebi, and I. F. García, "Hair removal methods: A comparative study for dermoscopy images", *Biomedical Signal Processing and Control*, vol. 6, no. 4, pp. 395–404, 2011.

[18] J. Pereira and R. Fonseca-Pinto, "Segmentation Strategies in Dermoscopy to Follow-up Melanoma: Combined Segmentation Scheme", *The Online Journal of Science and Technology*, vol. 5, no. 3, pp. 56–61, 2015.

[19] A. Victor and M. Ghalib, "Automatic detection and classification of skin cancer", *International Journal of Intelligent Engineering and Systems*, vol. 10, no. 3, pp. 444–451, 2017.

[20] Q. Abbas, I. Fondón, and M. Rashid, "Unsupervised skin lesions border detection via two-dimensional image analysis", *Computer Methods and Programs in Biomedicine*, vol. 104, no. 3, e1–e15, 2010.

[21] O. Ronneberger, P.Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation", in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, ser. LNCS, vol. 9351, Springer, 2015, pp. 234–241.

[22] N. C. F. Codella, D. Anderson, T. Philips, A. Porto, K. Massey, J. Snowdon, R. Feris, and J. Smith, "Segmentation of both Diseased and Healthy Skin from Clinical Photographs in a Primary Care Setting", pp. 1–4, 2018. [Online]. Available: http://arxiv.org/abs/1804.05944.

[23] E. Flores and J. Scharcanski, "Segmentation of melanocytic skin lesions using feature learning and dictionaries", *Expert Systems with Applications*, vol. 56, pp. 300–309, 2016.

[24] Y. Yuan, M. Chao, and Y.-C. Lo, "Automatic skin lesion segmentation with fully convolutional-deconvolutional networks", *IEEE journal of biomedical and health informatics*, 2017.

[25] M. Al-masni, M. A. Al-antari, M.-T. Choi, S.-M. Han, and T.-S. Kim, "Skin lesion segmentation in dermoscopy images via deep full resolution convolutional networks", *Computer Methods and Programs in Biomedicine*, vol. 162, May 2018.

[26] R. B. Oliveira, M. E. Filho, Z. Ma, J. P. Papa, A. S. Pereira, and J. M. R. Tavares, "Computational methods for the image segmentation of pigmented skin lesions: A review", *Computer Methods and Programs in Biomedicine*, vol. 131, pp. 127–141, 2016.

[27] C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing*. 2011, p. 392, ISBN: 9780470844724.

[28] S. Yuheng and Y. Hao, "Image Segmentation Algorithms Overview", vol. 1, 2017. [Online]. Available: http://arxiv.org/abs/1707.02051.

[29] 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 *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, IEEE, 2018, pp. 168–172.

[30] M. Celebi, H. Kingravi, H. Iyatomi, Y. Aslandogan, W. Stoecker, R. Moss, J. Malters, J. Grichnik, A. Marghoob, H. Rabinovitz, and S. Menzies, "Border detection in dermoscopy images using statistical region merging", English, *Skin Research and Technology*, vol. 14, no. 3, pp. 347–353, Aug. 2008.

[31] Q. Abbas, M. E. Celebi, I. Fondón García, and M. Rashid, "Lesion border detection in dermoscopy images using dynamic programming", *Skin Research and Technology*, vol. 17, no. 1, pp. 91–100, Jan. 2011.

[32] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[33] M. Kass, a. Witkin, and D. Terzopoulos, "Snakes: Active contour models", *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[34] C. Xu, J. L. Prince, and B. Hall, "Snakes, Shapes, and Gradient Vector Flow 1 Introduction", *IEEE Trans. Image Proc.*, vol. 7, no. 3, pp. 1–23, 1997.

[35] S. Haykin, *Neural networks: a comprehensive foundation*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 2007.

[36] A. G. C. Pacheco, "Agregação de classificadores neurais via integral de Choquet com respeito a uma medida fuzzy", Master's thesis, Universidade Federal do Espírito Santo, Vitória, Jun. 2016.

[37] K. Gnana Sheela and S. N. Deepa, "Review on methods to fix number of hidden neurons in neural networks", *Mathematical Problems in Engineering*, vol. 2013, Jun. 2013.

[38] R. Quiza and J. Davim, "Computational methods and optimization", in. Jan. 2011, pp. 177–208.

[39] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors", *Nature*, vol. 323, pp. 533–536, 1986.

[40] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", in *ICML*, 2015.

[42] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?", *ArXiv e-prints*, May 2018.

[43] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, Jun. 2015.

[44] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks", in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2010, pp. 2528–2535.

[45] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning", in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 2018–2025.

[46] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning", *CoRR*, vol. abs/1603.07285, 2016.

[47] S. J. Pan and Q. Yang, "A survey on transfer learning", *IEEE Trans. on Knowl. and Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[48] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?", in *NIPS*, 2014.

[49] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks", in *ECCV*, 2014.

[50] F. Perez, C. Vasconcelos, S. Avila, and E. Valle, "Data augmentation for skin lesion analysis", in *OR 2.0/CARE/CLIP/ISIC@MICCAI*, 2018.

[51] A. Arnab, S. Zheng, S. Jayasumana, B. Romera-Paredes, M. Larsson, A. Kirillov, B. Savchynskyy, C. Rother, F. Kahl, and P. H. S. Torr, "Conditional random fields meet deep neural networks for semantic segmentation: Combining probabilistic graphical models with deep learning for structured prediction", *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 37–52, Jan. 2018.

[52] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with gaussian edge potentials", in *NIPS*, 2011.

[53] G. S. C. Ucelli, "Classificação de câncer de pele usando Redes Neurais Convolutivas embarcadas em dispositivos móveis", Federal University of Espírito Santo, 2017.

[54] D. J. Bora, A. K. Gupta, and F. A. Khan, "Comparing the performance of L*A*B* and HSV color spaces with respect to color image segmentation", *CoRR*, vol. abs/1506.01472, 2015.

[55] M. Engilberge, E. Collins, and S. Süsstrunk, "Color representation in deep neural networks", in *2017 IEEE International Conference on Image Processing (ICIP)*, Sep. 2017, pp. 2786–2790.

[56] C. Qian and H. Jiang, "ISIC 2018 – Skin lesion analysis", 2018.

[57] H. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning", *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016.

[58] D. Powers, "Evaluation: From precision, recall and FM measure to ROC, informedness, markedness and correlation", *Journal of Machine Learning Technologies*, vol. 2, pp. 37–63, Jan. 2007.

[59] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", *CoRR*, vol. abs/1412.6980, 2014.

[60] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations", in *DLMIA/ML-CDS@MICCAI*, 2017.

[61] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance", *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.

[62] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms", *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[63] R. A. Krohling and A. G. Pacheco, "A-TOPSIS – An approach based on TOPSIS for ranking evolutionary algorithms", *Procedia Computer Science*, vol. 55, pp. 308–317, 2015, 3rd International Conference on Information Technology and Quantitative Management, ITQM 2015.