

Article

Melanoma and Nevus Skin Lesion Classification Using Handcraft and Deep Learning Feature Fusion via Mutual Information Measures

Jose-Agustin Almaraz-Damian ^{1,†}, Volodymyr Ponomaryov ^{1,*,†}, Sergiy Sadovnychiy ^{2,†}
and Heydy Castillejos-Fernandez ^{3,†}

¹ Instituto Politecnico Nacional, Santa Ana Ave. # 1000, Mexico City 04430, Mexico; jalmarazd1401@alumno.ipn.mx

² Instituto Mexicano del Petroleo, Lazaro Cardenas Ave. # 152, Mexico City 07730, Mexico

³ Academic Area of Computer and Electronics, Institute of Basic Sciences and Engineering, Universidad Autonoma del Estado de Hidalgo, Pachuca–Tulancingo Highway Km. 4.5, Mineral de la Reforma, Hidalgo 42083, Mexico

* Correspondence: volodymyr.ponomaryov@gmail.com; Tel.: +52-55565620158

† These authors contributed equally to this work.

Received: 26 February 2020; Accepted: 20 April 2020; Published: 23 April 2020



Abstract: In this paper, a new Computer-Aided Detection (CAD) system for the detection and classification of dangerous skin lesions (melanoma type) is presented, through a fusion of handcraft features related to the medical algorithm ABCD rule (Asymmetry Borders-Colors-Dermatoscopic Structures) and deep learning features employing Mutual Information (MI) measurements. The steps of a CAD system can be summarized as preprocessing, feature extraction, feature fusion, and classification. During the preprocessing step, a lesion image is enhanced, filtered, and segmented, with the aim to obtain the Region of Interest (ROI); in the next step, the feature extraction is performed. Handcraft features such as shape, color, and texture are used as the representation of the ABCD rule, and deep learning features are extracted using a Convolutional Neural Network (CNN) architecture, which is pre-trained on Imagenet (an ILSVRC Imagenet task). MI measurement is used as a fusion rule, gathering the most important information from both types of features. Finally, at the Classification step, several methods are employed such as Linear Regression (LR), Support Vector Machines (SVMs), and Relevant Vector Machines (RVMs). The designed framework was tested using the ISIC 2018 public dataset. The proposed framework appears to demonstrate an improved performance in comparison with other state-of-the-art methods in terms of the accuracy, specificity, and sensibility obtained in the training and test stages. Additionally, we propose and justify a novel procedure that should be used in adjusting the evaluation metrics for imbalanced datasets that are common for different kinds of skin lesions.

Keywords: fusion; handcraft; deep learning; melanoma; convolutional neural networks; transfer learning; computer-aided systems; mutual information; balance; data

1. Introduction

Skin cancer has become one of the deadliest diseases for human beings. Globally, each year, between two and three million non-melanoma (less aggressive) cases occur, and over 130,000 melanoma (aggressive) types are diagnosed [1].

Melanoma is the deadliest type of skin cancer. Australia has the highest rates of skin cancer in the world. In 2018, melanoma accounted for about 22% of skin cancer diagnoses, and non-melanoma tumors accounted for about 78% [2]. Studies have shown that this disease is caused most of the time

by exposure to UV radiation in daylight, tanning on sunbeds, and skin color, among others. Physicians have suggested that the best way to detect a malignant skin lesion of any kind is early detection. The rate of survival increases to almost 99% over five years if the disease is spotted in the early stages.

Dermoscopy or Epiluminescence Microscopy (ELM) is a medical method that helps a physician to recognize if a skin lesion belongs to a benign or malignant type of the disease. This method uses a dermatoscope, a tool that consists of a light source and amplification lens to enhance the view of medical patterns such as ramifications, globs, pigmented networks, veils, and colors, among others.

Since the image processing techniques were developed, Computer-Aided Detection (CAD) systems and approaches in the classification [3–7] and segmentation [8] of a Pigmented Skin Lesion (PSL) have been improved, benefiting patient diagnoses in early stages of the disease without shocking or painful medical procedures.

In this work, we propose a novel approach in the detection of a skin lesion among melanoma or nevus types, using handcraft features that depend on shape, color, and texture, which represent the ABCD rule (Asymmetry Borders-Colors-Dermatoscopic Structures), and combining them with deep learning features; these latter features were extracted using the transfer learning method as a generic feature extractor. At the next step, the most important features according to the Mutual Information (MI) metric features should be selected using the fusion technique, aiming at the best performance by taking into account the influences of both sets of features on the binary classification result.

This paper is organized as follows: Section 2 presents a brief review of the methods used in CAD developments with fused features and their nature, Section 3 explains in detail the proposed method, the materials used, and the evaluation metrics employed, and Section 4 describes the experimental results and presents a brief discussion. The conclusions are detailed in Section 5.

2. Literature Survey

Medical detection algorithms are one of the first tools used for determining whether a skin lesion is malignant or benign [9–13]. Nachbar et al. [9] developed a subjective method based on the visual perception of the lesion. ABCD Rule is based on color, shape, and particular structures that appear on skin lesions. Due to the simplicity of the algorithm, it is one of the most practiced for evaluating a lesion by a naked eye exam or using a dermatoscope.

The ABCD medical algorithm is composed of the following parts:

- **Asymmetry A:** The lesion is bisected into two perpendicular axes at 90° of each other, so as to yield the lowest possible asymmetry score. In other words, whether the lesion is symmetrical or not is determined. For each axis, where asymmetry is found, one point is added.
- **Borders B:** The lesion is divided into slices by eight axes determining whether a lesion has abrupt borders. If one segment presents an abrupt border, one point is added.
- **Colors C:** The lesion can contain one or more of the following colors: white, brownish, dark brown, black, blue, and red. They are generated by vessels and melanin concentrations, so for each one color founded, one point per color is added.
- **Dermatoscopic Structures D:** The lesion has the appearance of the following structures: dots, globs, pigmented networks, and non-structured areas. A point is added for each structure spotted on the lesion.

The described features are weighted as follows:

$$TDS = 1.3 * A + 0.1 * B + 0.5 * C + 0.5 * D, \quad (1)$$

where TDS is the Total Dermatoscopic Score; if it is less than 4.75, it is concluded that the lesion is benign; if the score is between 4.75 and 5.45, the lesion is considered suspicious; if it is more than 5.45, then it is considered malignant.

This algorithm has subversions, where some elements such as dermatoscopic structures are changed by diameter or darkness [14,15]. Additionally, in [16,17], the addition of features known as

EFG properties has been suggested; E stands for elevation or evolution, F for firmness, and G for growth. These features work as complementary information obtained from a PSL. Modifications in the ABCD rule are based on simplifying the evaluation of a skin lesion such that anyone can evaluate themselves and record any change in the injury. If the lesion is not identified by these methods, physicians are obliged to initiate invasive methods such as a biopsy to determine their type.

Adjed et al. [18] proposed a method where the aim is the fusion of structural features using Curvelet and Wavelet transform employing the Fast Digital Curvelet Transform (FDCT) wrapping method, and statistical metrics and texture features such as local binary pattern are then computed. They fused around 200 features via concatenation using the PH2 dataset [19].

Hagerty et al. [20] developed a fusion method where deep features are extracted from images using transfer learning method based on the ResNET-50 Convolutional Neural Network (CNN) architecture. However, the question as to which handcraft features are used in their method is not clear. Moreover, they used a feature selection algorithm, in this case, the χ^2 method, for performance revision, employing two datasets: the private set and the second set (a modified version of the ISIC 2018 dataset) [21].

Li et al. [22] used a deep learning approach with the fusion of clinical criteria representations, where as a classifier and fusion method, a boosting tree-learning algorithm called LightGBM is used [23]. This method is applied for color properties (RGB and HSL features), texture properties (SIFT and LBP), and shape properties (solidity and circularity, image ratio, and area ratio). The deep learning features were obtained using the transfer learning method based on the ResNET-50 and DenseNET-201 CNN architectures. Data pertaining to 566 features were processed using the ISIC 2018 dataset [21].

Abbas and Celebi [24] proposed a CAD system where the lesion is processed by a Stack-Based Auto-Encoder (SAE), extracting the deep features from the pixels of a lesion while minimizing the information loss. The handcraft features are extracted for color (the Hill climbing algorithm (HCA)) and for texture (the speed-up robust features (SURF)). Applying a feature fusion approach, they used Principal Component Analysis (PCA), and in the concluding stage, Recurrent Neural Networks (RNN) and A Softmax Linear Classifier were employed.

Among the reviewed methods, most of them use handcraft features and deep learning features with the help of the transfer learning method based on well-known CNN architectures [25–31]. As one can see, the revised schemes tried to fuse the information extracted from the lesion images, gathering data via the concatenation of the feature vectors, classifiers, and feature selection. The main drawback of such methods is the unawareness of the medical information, which is relevant to physicians besides the extracted data from image processing algorithms. The analyzed methods above employ several possibilities in the fusion of features, but most of them do not consider the importance of each extracted feature according to their nature, which is relevant for a pertinence class. Moreover, some of them lack features based on medical algorithms due to their assumption of the weakness of the perceptual handcraft features based on the subjective visual human system. Modern image processing techniques and machine learning approaches are able to learn the patterns and implement those features, as in a vision scoring system. Finally, some of the reviewed methods attempt to perform a multiclass classification, where the system classifies a lesion image to a specific lesion category. Nevertheless, here a problem endures in the multiclass classification, because the data available for each class are limited, and some of the public databases are not well balanced to perform this classification. As a result, a designed system can perform incorrect classifications. Summarizing, we consider the importance of developing an intelligent system that is able to perform the correct classification of melanoma disease employing both types of features, where medical features have relevance on the classification with the aid of deep learning features, aiming for the best performance.

The novel method considers relevant information obtained from handcraft and deep learning features, improving performance quality presented by commonly used criteria: accuracy, specificity, and sensibility. Different from other schemes, our novel framework encourages the use of ABCD rule features, also known as perceptual features, with a set of features equivalent to or based on a similar medical nature.

Principal Contributions

The principal contributions of our novel approach in the classification of dermoscopic lesions are summarized as follows:

- A brief survey of computer-aided detection methods that employ fusion between handcraft and deep learning features is presented.
- Despite the new tendencies of avoiding the ABCD medical algorithm or any of its variations, we utilized descriptors based on them, such as shape, color, and texture, as a new aggregation, and the extraction of deep learning features were used afterwards.
- A balanced method was employed due to the inconsistency of the ISIC database with respect to classes. A SMOTE oversampling technique was applied, which in this work demonstrates an improvement in performance at the differentiation of melanoma and benign lesion images.
- A fusing method that employs relevant mutual information obtained from handcraft and deep learning features was designed, and it appears to demonstrate better performance in comparison with state-of-the-art CAD systems.

3. Materials and Methods

In this section, the proposed system is described. A brief conceptual block diagram of the system is illustrated in Figure 1. As an initial step, the pigmented skin lesion image is segmented from surrounding normal skin tissue and artefacts such as veils, hairs, and air bubbles, among others, by color space transformation, mean thresholding, and extraction of the Region of Interest (ROI). Subsequently, using the binary mask image, the ROI image and a set of handcraft features based on shape, color, and texture are extracted. Thereafter, deep learning features are obtained using a selected CNN architecture, which is pre-trained in an Imagenet classification task; this CNN is employed as a feature extractor. All extracted features are concatenated in one vector, which later is fused according to the MI criterion. The selected classifier is trained on the ISIC dataset comprised of both malignant and other benign skin lesion images. Finally, the trained classifier model is used to predict each unseen pigmented skin lesion image as a benign or malignant lesion. The details of each stage of the proposed method are described in the remainder of this section.

3.1. Preprocessing

An image $I(x, y)$ that is analyzed can contain a lesion with some artefacts such as veils, hairs, stamps, among others. In the first step, we apply a preprocessing stage, where an image is enhanced [8,32,33].

A Gaussian filter is applied, and this filter is used to blur the artefacts contained on the image as primary targets: hair, marks, and spots, among others. This maintains the geometric/shape form of the lesion.

The Gaussian filter is denoted as follows:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \quad (2)$$

where σ^2 is the variance of the spatial kernel, this step is shown in Figure 2.

CIE L*a*b* space is characterized to more closely approximate the human perception system, where there are channels: L stands for lightness; a* and b* stand for chroma channels, where a* is a parametric measure between magenta and green, and b* is a parametric measure between blue and yellow. The L channel presents values between [0, 100], and a* and b* chroma channels have values between [-30, 30]. This transformation is used to avoid the correlation between channels, while keeping the perceptual data intact, such as a pigmented skin lesion that is darker than healthy skin tissue, as one of the sub-variants of the ABCD algorithm states [14].

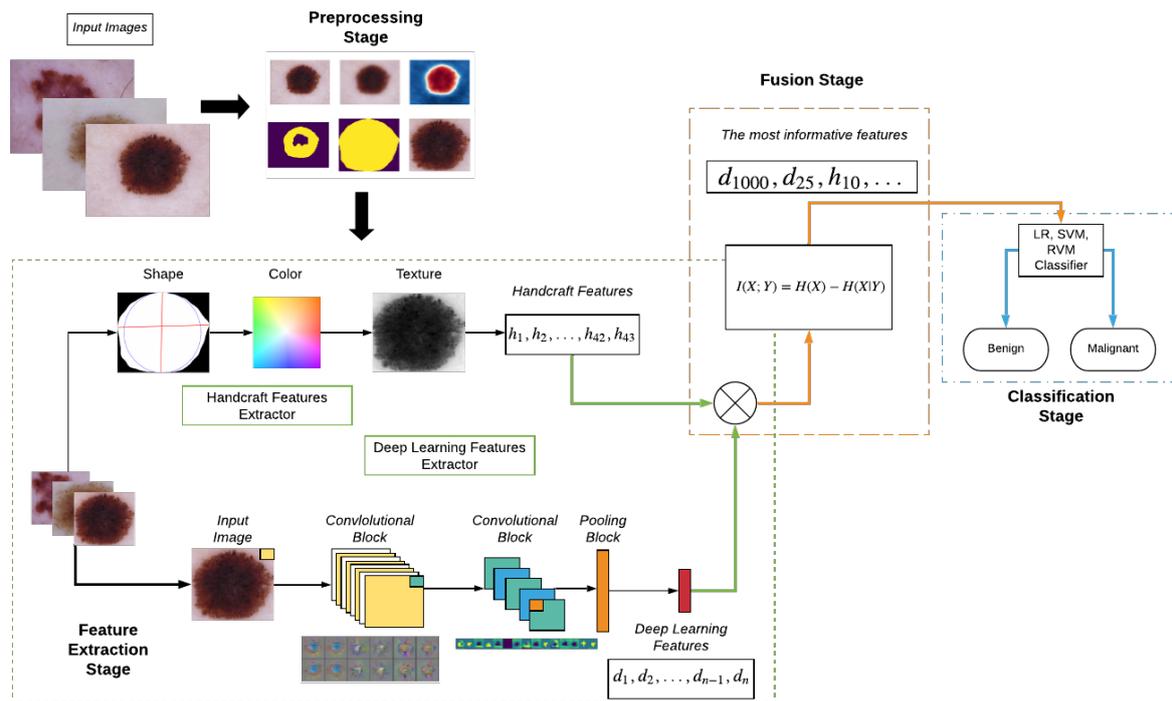


Figure 1. Block diagram of the novel Computer-Aided Detection (CAD) system.

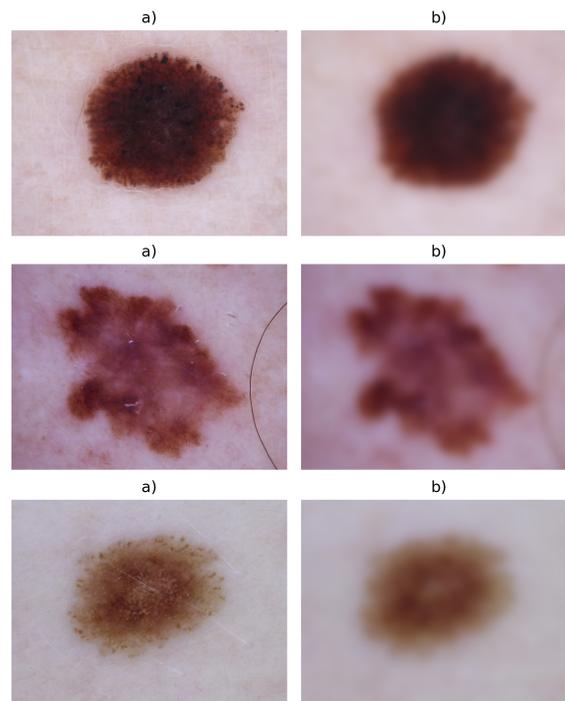


Figure 2. (a) Original image $I(x, y)$; (b) image (a) processed with a Gaussian filter.

In each channel of $CIEL^*a^*b^*$ for images I_L, I_a, I_b , the mean thresholding procedure is applied. Such thresholding allows one to differentiate skin tissue from lesion tissue. In Figure 3, one can see how the $CIEL^*a^*b^*$ space is able to visually separate this information. These mean values are calculated as follows:

$$\bar{I}_L = \sum_{x=1}^m \sum_{y=1}^n \frac{1}{mn} I_L(x, y), \tag{3}$$

$$\bar{I}_a = \sum_{x=1}^m \sum_{y=1}^n \frac{1}{mn} I_a(x, y), \quad (4)$$

$$\bar{I}_b = \sum_{x=1}^m \sum_{y=1}^n \frac{1}{mn} I_b(x, y), \quad (5)$$

where (x, y) are the spatial coordinates, m, n are the sizes of an image, and $\bar{I}_L, \bar{I}_a, \bar{I}_b$ denote the mean values. The thresholding operation is applied in each channel of the CIEL*a*b* space, forming the thresholded channel images as follows:

$$I_{Th_L}(x, y) = \begin{cases} 1, & I_L(x, y) \geq \bar{I}_L, \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

$$I_{Th_a}(x, y) = \begin{cases} 1, & I_a(x, y) \geq \bar{I}_a, \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

$$I_{Th_b}(x, y) = \begin{cases} 1, & I_b(x, y) \geq \bar{I}_b, \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

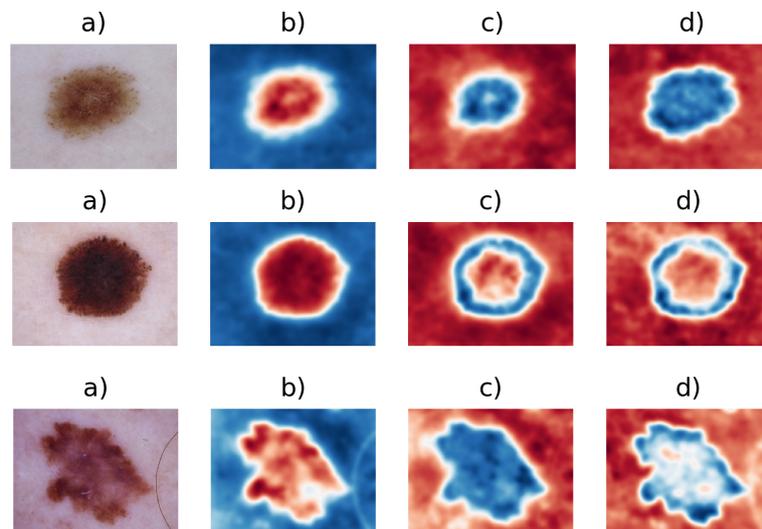


Figure 3. (a) Original image $I(x, y)$, (b) $I(x, y)$ on channel L, (c) $I(x, y)$ on channel a^* , and (d) $I(x, y)$ on channel b^* .

Afterwards, the following logic operation is applied on each binarized image, I_{Th_L} , I_{Th_a} , and I_{Th_b} , to form a binary mask $I_{bin}(x, y)$ of the image.

$$I_{bin}(x, y) = I_{Th_L} \cap I_{Th_a} \cap I_{Th_b}. \quad (9)$$

Example of extracted Binary Masks images are given in Figure 4. Finally, a median filter with a kernel 5×5 is applied to the $I_{bin}(x, y)$, removing the remaining artefacts, which resists thresholding. Next, a bounding box algorithm is performed. The bounding box [34] is a method that is used to compute an imaginary rectangle that completely encloses the given object. This rectangle can be determined by the x- and y-axis coordinates in the upper-left and lower-right corners of a shape. This method is commonly used in object detection tasks because it estimates the coordinates of the ROI in an image.

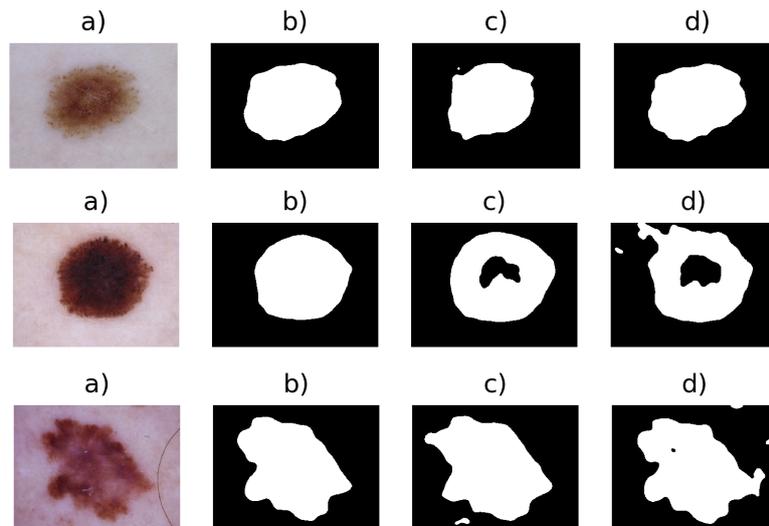


Figure 4. Results of the thresholding stage: (a) Original image $I(x, y)$, (b) binary image I_{Th_L} obtained from the threshold of the L channel, (c) binary image I_{Th_a} obtained from the threshold of the a* channel, (d) binary image I_{Th_b} obtained from the threshold of the b* channel.

Bissotto et al. [35] have shown the effect of the bias between different types of image segmentation, where those biases can negatively affect the performance of classification models. They consider that the usage of the bounding box algorithm to segment a lesion is appropriate because a CNN architecture can extract all the relevant features of a lesion and distinguish it from the surrounding healthy skin. Therefore, we consider this solution to reduce the bias of the classification model before processing.

3.2. Handcraft Features

The ABCD rule represents a set of perceptual features stated by the findings of patterns in PSLs. The ABCD method employs features that are based mostly on shapes, color, and texture. The selected features in this study are the representation of medical attributes using image processing algorithms.

Sirakov et al. [36] proposed a method to estimate the asymmetry of a lesion based on the binary mask I_{bin} , which is obtained from the previous thresholding step. Then, by rotating it through 180° , the symmetry mask $S_{I_{bin}}$ is formed, and the synthetic image A is calculated as follows :

$$A = I_{bin} \cup S_{I_{bin}} \quad (10)$$

where A is the generated image, which contains the non-overlapping regions of the lesion called false symmetry FS , therefore applying

$$Sym_{0^\circ} = 1 - (FS/A), \quad (11)$$

and this technique is applied on the 0° axis of the binary image.

In this study, a variation of the previous method is proposed, where from the A generated image is rotated from the major axis and the minor axis, by applying the same procedure again and finally, computing the average symmetry value between the two axes, as follows:

$$Symmetry = \frac{Sym_{0^\circ} + Sym_{90^\circ}}{2}. \quad (12)$$

The symmetry values belong to the interval $[0,1]$, where, if this index approaches the highest value (1), a lesion is more symmetric. The Figure 5 shown the extracted ROI images.

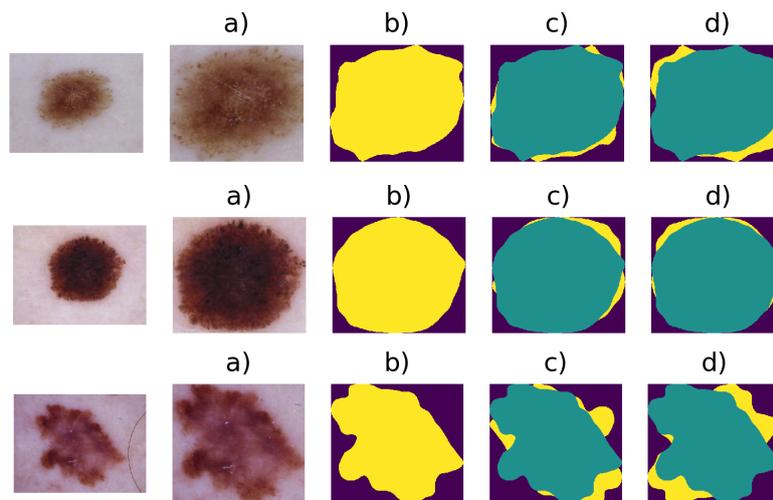


Figure 5. Results of the preprocessing stage: Original image $I(x, y)$, (a) Region of Interest (ROI) obtained, (b) segmented image $S_{I_{bin}}$, (c) obtained asymmetry at 0° , and (d) obtained asymmetry at 90° .

3.2.1. Shape Features

Shape features or geometric features [34] can describe an object or a form in numerical values to represent human perception.

For shape features, the following equations are employed:

$$Area = \sum_{x=1}^m \sum_{y=1}^n I_{bin}(x, y), \quad (13)$$

where m, n are the sizes of the image, and x, y are the spatial coordinates; therefore, the area consists of the amount of pixels contained in the ROI of a lesion.

$$Perimeter = \sum_{i=1}^m \sqrt{(x_1 - x_{i-1})^2 + (y_1 - y_{i-1})^2}, \quad (14)$$

where (x, y) are the spatial coordinates of the i -th pixel, which constructs the contour of the region, and the perimeter contains the amount of pixels around the ROI of a lesion.

$$Circularity = \frac{4\pi \cdot Area}{Perimeter^2}, \quad (15)$$

where the circularity shows the similarity between a shape and a circle.

$$Diameter = \sqrt{\frac{1}{2}(\mu_{2,0} + \mu_{0,2}) \pm \sqrt{4\mu_{1,1}^2 - (\mu_{2,0} - \mu_{0,2})^2}}, \quad (16)$$

where the diameter is formed by obtaining the length of the major axis and the minor axis of the shape, computed from the 2nd central moment. This measure connects two pairs of points on the perimeter of the shape.

$$Eccentricity = \frac{(\mu_{0,2} - \mu_{2,0})^2 + 4\mu_{1,1}}{A}, \quad (17)$$

which measures the aspect ratio of the length of the major axis to the length of the minor axis.

3.2.2. Colour Features

Medical algorithms, in particular the ABCD rule, try to present, as features, a set of colors contained on a PSL. Therefore, these features can be replaced by statistical characteristics obtained from color spaces. In this study, the following characteristics are used:

$$Min_{ch} = \min[I_{ch}(x, y)], \quad (18)$$

$$Max_{ch} = \max[I_{ch}(x, y)], \quad (19)$$

$$Var_{ch} = \text{var}[I_{ch}(x, y)], \quad (20)$$

$$Mean_{ch} = \overline{[I_{ch}(x, y)]}, \quad (21)$$

where $I_{ch}(x, y)$ is the image of a chosen channel for the PSL image in RGB and CIEL*a*b* color spaces.

3.2.3. Texture Features

Haralick et al. [37] proposed the Gray Level Co-occurrence Matrix (GLCM). This method analyzes the statistical texture features of an image. The texture features provide information about how the gray intensities of the PSL of the image are distributed. GLCM shows how often a gray level occurs at a pixel located in a fixed position, using $P_d(i, j)$ as the (i, j) element of the normalized GLCM; N_g is the number of gray levels; σ_x, σ_y and μ_x, μ_y are the standard derivations and the mean values among the i and j axes of the GLCM, and are expressed as follows:

$$\mu_x = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} iP_d(i, j), \quad (22)$$

$$\mu_y = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} jP_d(i, j), \quad (23)$$

$$\sigma_x = \sqrt{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu_x)^2 P_d(i, j)}, \quad (24)$$

$$\sigma_y = \sqrt{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (j - \mu_y)^2 P_d(i, j)}. \quad (25)$$

The 13 features used in this study are as follows:

$$ASM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_d^2(i, j). \quad (26)$$

The angular second moment measures consistency of the gray local values.

$$Contrast = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - j)^2 P_d(i, j), \quad (27)$$

This is the second moment. This characteristic measures the variations between pixels.

$$Correlation = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_d(i, j) \frac{(i - \mu_x)(j - \mu_y)}{\sigma_x \sigma_y}. \quad (28)$$

This is the linear dependency of the gray level values.

$$\text{Variance} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu)^2 P_d(i, j). \quad (29)$$

This is the second moment. It shows the spread around the mean in the surrounding neighborhood.

$$\text{IDM} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{1}{1 + (i - j)^2} P_d(i, j). \quad (30)$$

This is the Inverse Difference Moment (IDM). It shows how close the elements of the GLCM are in their distribution.

$$\text{Entropy} = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_d(i, j) \cdot \ln[P_d(i, j)]. \quad (31)$$

This is the measure of randomness of the gray values in an image.

Additional texture features that are used in this study are based on the difference statistics using the probability $P_{x-y}(k)$ that can be written as follows:

$$P_{x-y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_d(i, j), \quad k = 0, 1, \dots, N_g - 1, \quad (32)$$

where $P_d(i, j)$ is the (i, j) th element contained in the GLCM, and N_g is the number of gray levels.

$$\text{Sum Variance} = \sum_{k=2}^{2N_g} (k - \mu_{x+y})^2 P_{x+y}(k), \quad (33)$$

$$\text{Sum Entropy} = - \sum_{k=2}^{2N_g} P_{x+y}(k) \log[P_{x+y}(k)], \quad (34)$$

$$\text{Difference Variance} = \sum_{k=0}^{N_g-1} (k - \mu_{x-y})^2 P_{x-y}(k), \quad (35)$$

$$\text{Difference Entropy} = - \sum_{k=0}^{N_g-1} P_{x-y}(k) \log[P_{x-y}(k)], \quad (36)$$

$$\text{IMCorr}_1 = \frac{H(XY) - H(XY_1)}{\max[H(X)H(Y)]}, \quad (37)$$

$$\text{IMCorr}_2 = \sqrt{1 - \exp\{-2[H(XY_2) - H(XY)]\}}, \quad (38)$$

where $H(X)$, $H(Y)$, $H(XY)$, $H(XY_1)$ and $H(XY_2)$ are denoted as follows:

$$H(X) = - \sum_{i=1}^{N_g} P_x(i) \cdot \log[P_x(i)], \quad (39)$$

$$H(Y) = - \sum_{i=1}^{N_g} P_y(i) \cdot \log[P_y(i)], \quad (40)$$

$$H(XY) = - \sum_{i=1}^{N_g} P_d(i, j) \cdot \log[P_d(i, j)], \quad (41)$$

$$H(XY_1) = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_d(i, j) \cdot \log[P_x(i) \cdot P_y(j)], \quad (42)$$

$$H(XY2) = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_x(i) \cdot P_y(j) \cdot \log[P_x(i) \cdot P_y(j)]. \quad (43)$$

3.3. Deep Learning Features

Based on the discrete convolution operation,

$$W(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n) K(m, n), \quad (44)$$

the CNN [38,39] represents one type of method, namely, deep learning strategies [40], the basis of which is to obtain the information of an image $I(i, j)$, using filters $K(m, n)$, which are trained on a neural network as feed-forward and back-propagation, according to

$$y = z(W * X) + b, \quad (45)$$

where W are the computed values for the filters, z is the activation function, X is the input, and b is the bias [40,41].

The design of an architecture of a CNN is a rather complex task due to the statement of parameters such as the number and size of the filters, and the depth, even those that are task-related.

3.3.1. Transfer Learning

The main problem of using a deep learning approach is that a large amount of data is needed to train the network from scratch. Usually, to overcome this problem, the transfer learning method [42–44] is applied.

Transfer learning is a technique that can be defined as the generalization of a target task based on applied knowledge extracted from one or more source tasks [43]. The idea originates from human thinking: we do not learn exactly how to recognize a chair, a plane, or a book; we start by recognizing colors, shapes, and textures, and someone else then tells us how to differentiate a chair from an apple. This sharing of knowledge between beings helps us to understand the world as infants. Another idea is handling information collected for a task to solve related ones. Therefore, transfer learning can be defined as follows:

Assume a domain D , which consists of two components:

$$D = \{\chi, P(X)\}, \quad (46)$$

where χ is a feature space, and there is a marginal distribution: $P(X), X = \{x_1, \dots, x_n\}, x_i \in \chi$.

Given a task T with two components,

$$T = \{\gamma, P(Y|X)\} = \{\gamma, \eta\}; Y = \{y_1, \dots, y_n\}, y_i \in \gamma, \quad (47)$$

where γ is a label space with a predictive function; η is trained from $(x_i, y_i), x_i \in \chi, y_i \in \gamma$, for each feature vector in the D domain; η predicts the corresponding label $\eta(x_i) = y_i$.

The paper [43] states self-called scenarios:

Given a source domain and an objective domain D_S & D_T , where $D = \{\chi, P(X)\}$ and the related tasks are T_S & T_T , where $T = \{\gamma, P(Y|X)\}$, the conditions can vary as follows:

1. $\chi_S \neq \chi_T$: Feature spaces are different.
2. $P(X_S) \neq P(X_T)$: The marginal probabilities of the distributions are different.
3. $\gamma_S \neq \gamma_T$: The label spaces are different.
4. $P(Y_S|X_S) \neq P(Y_T|X_T)$: The conditional probabilities are different.

Transfer learning is defined in [44]. Given a source domain D_S with a corresponding source task T_S and a target domain D_T with a corresponding task T_T , transfer learning is the process of enhancement for a target predictive function $f_T(\cdot)$ using the related information from the domain source D_S and the task-related T_S .

CNN architectures are overlay filters that sample the data contained in an image. Therefore, these filters are hierarchical representations called feature maps when all filters learn all the features based on image data. These are connected to the last layer of the CNN architecture, which is a neural network classifier referred to in the literature as a fully connected layer [40].

Moreover, CNN architectures belong to a class of inductive learning algorithms, where the objective of these algorithms is to map input features between classes seeking the generalization of the data.

Therefore, inductive learning can be transferred from an architecture trained on the source task to the target class, and this is done by adjustments of the model space, correcting the inductive bias. Commonly, this can be performed by replacing the last layer of the model, which is the classifier, from the original one to a lightweight classifier, which should be trained on the generalized features.

In this study, we employed the transfer learning method on architectures that are pre-trained on a similar task. In this research, the Imagenet classification task [45] was employed, by the assumption that $D_S = D_T$ and the CNN Architecture should perform as a generic feature extractor.

3.3.2. Feature Extractor

In our case, the CNN Architecture was used as a deep feature extractor, where features are extracted by the following:

Consider an image $I(x, y)$ of the domain D_T that is mapped or transformed by

$$W = \{w_1, \dots, w_n\}; w_1 \in \mathbb{R}^{M \times N \times h}, \quad (48)$$

where W is the weight computed by the feature extractor, where M, N , and h are the proposed size of the CNN architecture. As a result, we can obtain

$$P = W(I(x, y)) = \{w_1(I(x, y)), \dots, w_n(I(x, y))\}; \in \mathbb{R}^{M \times N \times h}, \quad (49)$$

and the pooling transformations are based on

$$Q = f(0, P) = \{f(0, w_1(I(x, y))), \dots, f(0, w_n(I(x, y)))\}, \quad (50)$$

where $f(\cdot)$ is a mapping function.

3.3.3. Deep Learning Architectures

Below, we use the following architectures: VGG-16/VGG-19 [26], MobileNet [27], ResNet-50 [28], Inception v3 [29], Xception [30], and DenseNet-201 [31]. The selection of these architectures was based on the fact that they have been shown to obtain the Top-1 and Top-5 best accuracy and error rates on the Imagenet task proposed for [45], which is an image classification task.

In Table 1, the number of features extracted by each utilized architecture is explained.

The features for each selected architecture are concatenated with the extracted handcraft features, forming a unique feature vector:

$$F = D_{\text{deep}}[\#Features_{\text{Extracted}}] \cup H_{\text{and}}[Asymetry, Area, \dots, Entropy, Contrast, Std\dots, Max, Min, \dots], \quad (51)$$

where the size of vector F is equal to 43 for the handcraft features, plus the number of features extracted from the CNN architecture used.

Table 1. The number of features extracted by architecture.

CNN Architecture	#Features
VGG19	4096
VGG16	4096
ResNET-50	2048
Inception v3	2048
Mobilenet v1	1024
Mobilenet v2	1280
DenseNET-201	1920
Xception	2048

3.4. Algorithm Summary

Let the sum of all proposed and explained procedures be described in the form of an algorithm for extraction features for PSL Images. The proposed CAD system consists of four principal stages: (a) preprocessing, (b) handcraft features, (c) deep learning features, and (d) the fusion stage. In the first stage, artifacts are removed by using a Gaussian filter after a CIELab color transformation is employed. Mean thresholding per channel is employed afterwards. We then extract the ROI by using the boundary box algorithm. In the second stage, shape features are extracted using Equations (12)–(17), Equations (18)–(21) are computed for statistical color features, and texture features are then calculated from Equations (22)–(38). The ROI image is finally processed by the chosen CNN Architecture, whose features are concatenated for the following steps.

Algorithm 1 presents the details of the feature extraction process for PSL images.

3.5. Feature Selection

After extracting the deep learning and handcraft features, the number of features is reduced, which is a typical problem in machine learning algorithms, because high-dimensional data increases the computing time for a prediction.

Feature selection is one method to resolve this problem. In [46], filtering methods are applied, and the features are selected based on various statistical tests, such as χ^2 , ANOVA, Linear discriminant analysis (LDA) among others [47].

The extracted data can be represented in the form of a high-dimensional matrix defined by

$$\mathbf{X} \in \mathbb{R}^{n \times p} \quad (52)$$

where \mathbf{X} is the extracted data, n represents the instances or elements, and p represents the features extracted for each element.

The idea is to reduce the data as much as possible, such that the features are extracted by selecting a subset of each element that is relevant to the pertinence category or label y . This subset is defined as

$$\mathbf{X}_S \in \mathbb{R}^{n \times k} \quad (53)$$

where \mathbf{X}_S represents the reduced data, n is the same instance or element of the original data matrix, and k is the selected or reduced features based on $k \ll p$.

Mutual Information

In this work, we propose the MI metric to reduce the data of the extracted features. MI is a measure based on the entropy measure:

$$I(X; Y) = H(X) - H(X|Y) = \sum_{y \in Y} \sum_{x \in X} P(X \cap Y) \log_2 \left(\frac{P(X \cap Y)}{P(X)P(Y)} \right) \quad (54)$$

where $X = x_1, \dots, x_n$ and $Y = y_1, \dots, y_n$ in the multi-variable case, $H(X|Y)$ is the conditional entropy between two random variables, and $H(X)$ is the entropy of a random variable [48–50].

$$H(x) = - \sum_{i=1}^n P(x_i) \log_2 P(x_i) \quad (55)$$

Algorithm 1 Algorithm summary

Require: PSL Image

(a) Preprocessing

1: **Input:** I

Apply Gaussian Filter, Equation (2)

Apply RGB to CIEL*a*b* Color Transform

Separate CIEL*a*b* Image I_{Lab} in Image Channels I_L, I_a, I_b

Calculate Mean value of I_L, I_a, I_b , Equations (3)–(5)

2: **for all** $(i, y) \in I_L$ **do**

3: **if** $I_L(x, y) \geq \bar{I}_L$ **then**

4: Assign 1 to $I_{Th_L}(x, y)$

5: **else**

6: Assign 0 to $I_{Th_L}(x, y)$

7: **end if**

8: **end for**

9: **for all** $(i, j) \in I_a$ **do**

10: **if** $I_a(x, y) \geq \bar{I}_a$ **then**

11: Assign 1 to $I_{Th_a}(x, y)$

12: **else**

13: Assign 0 to $I_{Th_a}(x, y)$

14: **end if**

15: **end for**

16: **for all** $(i, j) \in I_b$ **do**

17: **if** $I_b(x, y) \geq \bar{I}_b$ **then**

18: Assign 1 to $I_{Th_b}(x, y)$

19: **else**

20: Assign 0 to $I_{Th_b}(x, y)$

21: **end if**

22: **end for**

Compute $I_{bin}(x, y)$ Applying Equation (9) to $I_{Th_L}, I_{Th_a}, I_{Th_b}$

Apply Median Filter, size = 5×5

Compute Bounding Box Algorithm to estimate coordinates of Region Of Interest

Crop $I_{roi}(x, y)$ from estimated coordinates on $I(x, y)$ & $I_{Lab}(x, y)$

23: **Output:** Region of Interest (ROI) Image $I_{roi}(x, y)$

(b) Handcraft Features

24: **Input:** I_{ROI}

Compute Area, Perimeter, Circularity, Diameter and Eccentricity from Equations (13)–(17)

Compute Asymetry from Equations (10)–(12)

Compute Color Features from Equations (18)–(21) from $I_{roi}(x, y)$

Compute Texture Features from Equations (22)–(43) from $I_{roi}(x, y)$

25: Concatenate the extracted features H

26: **Output:** H Handcraft features

(c) Deep Learning features

27: **Input:** I_{roi}

28: Load the weights W_i from selected CNN architecture

29: Apply the weights W_i to I_{roi}

30: Obtain the D deep learning features

31: **Output:** D deep learning features

(D) Wrapping Features

32: **Input:** D, H

33: Apply $H \cup D$ to the extracted features

34: **Output:** F Full set of extracted features

Ross, in [51], proposed an MI estimator for continuous and discrete data aiming at the relationship between datasets.

Based on the nearest neighborhood rule, the idea is to find a k -nearest neighbor between a point i among all the data points N_{xi} using the Chebyshev distance metric:

$$D_{chebyshev} = \max_i(|x_i - y_i|), \quad (56)$$

and the MI measure is then computed as follows:

$$I(X, Y) = \langle I_i \rangle = \psi(N) - \langle \psi(N_x) \rangle + \psi(k) - \langle \psi(m) \rangle, \quad (57)$$

where $\psi(\cdot)$ is the digamma function, $\langle N_x \rangle$ is the average of the data points, k is the number of k -closest neighbors to the point i , and $\langle m \rangle$ is the average counted neighbors among the full dataset.

If our dataset contains continuous data points, these are discretized using a binning method—grouping the data into bins—leading to a binned approximation of MI as follows:

$$I(X, Y) = \left\langle \log \frac{p(x_i, b_i)}{p(x_i)p(b_i)} \right\rangle_i. \quad (58)$$

After applying the MI method, a new vector is delivered, and it shows the value of MI per concrete feature among all obtained features. Next, the mean value is calculated for the MI among all features. We proposed to use this value as a threshold to discard the features with the lowest MI values and to keep the features with the highest MI values. Tables 2 and 3 expose examples of different parts of the fused feature vector.

$$L_{features}(x, y) = F(x, y) > \overline{M.I(x, y)} \quad (59)$$

The found subset contains the features with the highest mutual information values that we consider as the fused data of both sets of extracted features from a PSL. Below, as an example, we present the behavior of the features and their MI values.

Tables 2 and 3 expose, for illustration, several MI values between the extracted features for the dataset of images. Some of the deep learning features appear to demonstrate negligible MI values for the binary classification problem, according to the complete set of features extracted from the database, as one can see in Table 2.

Table 2. Example of features that expose the lowest Mutual Information (MI) values.

Inception v3 + Handcraft Features	
Feature	Mutual Info. value
1448	$3.813956037657107 \times 10^{-7}$
648	$1.4314941993776031 \times 10^{-5}$
1020	$3.4236070477255964 \times 10^{-5}$
804	$3.515106075036023 \times 10^{-5}$
333	$4.213506368255793 \times 10^{-5}$
562	$4.4971751217204314 \times 10^{-5}$
91	$5.133156269931938 \times 10^{-5}$
852	$6.514623080855486 \times 10^{-5}$
1689	$7.53828133426282 \times 10^{-5}$
1788	$7.605629690621285 \times 10^{-5}$

Table 3 exposes several features with significant values of MI that are merged with additional significant handcraft features, forming the final set of features for the proposed system. Therefore, the proposed fusion method based on MI measurements demonstrates the fusion for both types of features in accordance with their influence on the binary classification problem.

Therefore, in contrast with state-of-the-art techniques that use Concat, PCA-based, and χ^2 -based methods, among others, for the selection of significant features, the proposed approach employs the information measures, justifying the informative weight of each feature that is used in the classification stage.

Table 3. Example of features that expose the highest MI values.

Inception v3 + Handcraft Features	
Feature	Mutual Info. value
Mean_b	0.09829037284938669
Min_b	0.09536305749234275
Max_b	0.06834317593527395
578	0.06131147578510121
Min_G	0.05817685924318594
116	0.055293703553799256
389	0.05446628875169646
464	0.05424503079140042
Var_L	0.05420063226575533
288	0.053949117014718606

4. Results and Discussion

The classifiers: logistic regression, support vector machine with linear and rbf kernel [52], and finally, relevant vector machine [53,54] algorithms were employed in this work. The rationale for the use of different classifiers lies in the fundamental idea of transfer learning, whereby after extracting generic features, a shallow classifier must be applied to test the proposed method.

4.1. Experimental Results

4.1.1. Experimental Setup

The described method was performed on a PC with an Intel[®] Xeon E5 1230-V5 CPU, 32GB RAM, NVIDIA GeForce[®] 1080Ti with 11 GB RAM, running a Linux 64-bit operating system, Python 3.5, and the libraries: Keras 2.3 [55], Sklearn [56], Mahotas [57], thundersvm [58], and Imblearn [59].

4.1.2. Evaluation Metrics

In this study, we used commonly applied performance metrics: accuracy, sensitivity, specificity, precision, f-score, and the Matthews correlation coefficient:

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn}. \quad (60)$$

The accuracy value measures the appropriate classifications over the total elements evaluated.

$$Sensibility = \frac{tp}{tp + fn}. \quad (61)$$

The sensibility value, also known as recall, measures the number of positive elements that are correctly classified.

$$Specificity = \frac{tn}{tn + fp}. \quad (62)$$

The specificity value measures the number of negative elements that are correctly classified.

$$Precision = \frac{tp}{tp + fp}. \quad (63)$$

The precision value measures the number of elements that are correctly classified among all the positive elements to evaluate.

$$F_{Score} = \frac{2tp}{2tp + fp + fn}. \quad (64)$$

The F-score value measures the harmonic mean between precision and recall.

These criteria are described in terms of tp , tn , fp , and fn , which denote true positive, true negative, false positive, and false negative, respectively. Additionally, to characterize the classifier performance, we used the Matthews correlation coefficient [60]:

$$MCC = \frac{(tp \times tn) - (fp \times fn)}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}}, \quad (65)$$

where the MCC value measures the performance of the classification model as a coefficient between the predicted and the observed elements of the binary classification. It returns a value between $[-1, 1]$, where a value of 1 represents a perfect prediction, a value of 0 is no better than a random prediction, and -1 indicates total disagreement between prediction and observation.

4.1.3. Dataset

This study uses the public ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection grand challenge datasets [21] task 3, also known as HAM10000 [61], which contains 10,015 separate images, as shown in Figure 6, where AKIEC corresponds to Actinic Keratosis, BCC is Basal Cell Carcinoma, DF is a Dermatofibroma, MEL is Melanoma, NV is Nevus, BKL is Pigmented Benign Keratosis, and VASC is Vascular. This distribution was obtained from the Ground Truth file, and each image is on RGB space with a size of 450×600 .

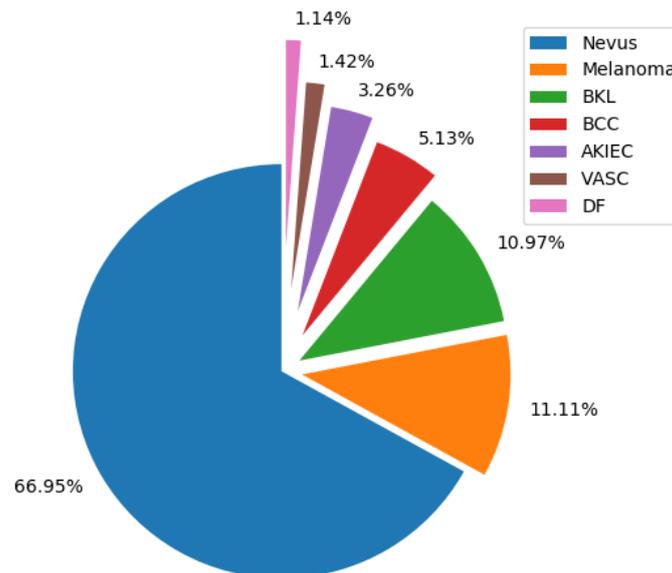


Figure 6. Distribution of classes on the ISIC2018/HAM10000 dataset.

As one can see in Figure 6, the ISIC dataset contains images that belong to different types of skin lesions that do not correspond to the melanoma-type lesion. Therefore, we decided to modify the dataset to develop a binary class classification by excluding all the classes, except for Melanoma and Nevus.

We split the ISIC dataset: 75% for the training set and 25% for test set. The features extracted were processed by Z-score normalization:

$$Z = \frac{x - \mu}{\sigma}. \quad (66)$$

4.1.4. Balance Data

The adjustment of the ISIC dataset mentioned above means that the data are unbalanced, where one class contains more data than others. In most machine learning techniques, when they employ unbalanced data, this can result in a lower performance of the minority class, which can cause misclassification of the data.

SMOTE [62] is a data augmentation method that can oversample the data of the minority class, compensating with the majority class. This method is based on K-NN clustering and Euclidean distance, selecting two points of the minority class and computing a new one based on them. This method is iterative until it reaches an equivalent amount of information of the majority class.

The SMOTE technique has been employed in several studies [63,64], where extracted features that belong to an unbalanced dataset are oversampled to compensate for the number of instances between classes. In this work, we apply this method to compensate the data of the melanoma class against nevus ones to the selected features by the MI criterion, and as result, a balanced dataset with the fused features can be obtained.

The study [65] introduces new metrics to overcome this problem, where geometric mean attempts to maximize the accuracy of each of the two classes similarly, a performance metric that correlates both objectives:

$$G_{mean} = \sqrt[2]{Sensitivity \cdot Specificity}. \tag{67}$$

Dominance is aimed at quantifying the prevalence relation between the majority and minority classes and is used to analyze the behavior of a binary classifier:

$$Dominance = Sensitivity - Specificity. \tag{68}$$

The Index of Balanced Accuracy (IBA) is a performance metric in classification that aims to make it more sensitive for imbalanced domains. This metric is defined as follows:

$$IBA = (1 - Dominance) \cdot G_{mean}^2. \tag{69}$$

The objective of this procedure is to moderately favor the classification models with a higher prediction rate of the minority class, without underestimating the relevance of the majority class.

4.1.5. Experimental Results

In the following tables, the experimental results of binary classification for balance data are presented.

The experimental results in Table 4 show that the designed system appears to demonstrate sufficiently good performance when different CNN architectures are fused with handcraft features in accordance with the MI metric that can seek relevant information among features against concatenating or discriminative analysis.

Table 4. Performance results of the proposed method using selected deep learning architectures fused with handcraft features.

	Acc. Train	Acc. Test	Sensibility	Specificity	Precision	F-Score	AUC	G-Mean	IBA	MCC
VGG16	88.60	84.90	79.23	0.85	88.74	83.71	84.79	0.85	0.72	0.7012
VGG19	90.23	87.14	82.46	0.87	90.44	86.26	87.05	0.87	0.76	0.7451
Mobilenet v1	91.48	89.32	84.04	0.89	93.49	88.51	89.21	0.89	0.79	0.7898
Mobilenet v2	92.40	89.71	86.41	0.90	92.08	89.16	89.64	0.90	0.80	0.7953
ResNET-50	90.67	87.86	81.24	0.88	93.09	86.76	87.72	0.87	0.77	0.7624
DenseNET-201	91.10	88.54	83.25	0.88	92.61	87.68	88.44	0.88	0.78	0.5985
Inception V3	91.33	88.10	84.87	0.88	90.59	87.42	88.02	0.88	0.77	0.7632
Xception	90.47	87.53	83.19	0.87	90.58	86.73	87.44	0.87	0.76	0.7525

Table 4 provides the experimental results obtained using the selected CNN architectures, where the Mobilenet v2 appears to demonstrate the best performance in comparison to the before-mentioned architectures. The proposed method shows notable evaluation metrics such as accuracy, Area Under Curve (AUC), and IBA metrics. The selected features contain the fused features that are the most relevant for the classification of a lesion with the MI metric.

Presented in Table 5, experimental results for different criteria show that the designed system outperforms several state-of-the-art methods. The experimentally justified performance, in our view, is due to the fusion technique employed, where a mutual information metric seeks relevant information among features upon concatenating or discriminative analysis. Moreover, the IBA metric is employed, achieving a value of 0.80, which confirms the stability and robustness of the system when balanced data are used.

Table 5. Comparison between our novel CAD and state-of-the-art CADs.

Metric	[18]	[24]	[22]	Proposed Method Using Mobilenet v2 Architecture
Accuracy	86.07	95	85.55	92.40
Sensibility	78.93	93	86	86.41
Specificity	93.25	-	-	90
Precision	-	93	85	92.08
F-Score	-	-	86	89.16
G-Mean	-	-	-	0.90
IBA	-	-	-	0.80
MCC	-	-	-	0.7953
Imbalance Data	Yes	No	No	No
Fused Data	Yes	Yes	Yes	Yes
Type of Classification	Binary	Binary	Multiclass	Binary

The proposed method achieves an accuracy of 92.40%, sensitivity of 86.41%, AUC of 89.64%, and an IBA of 0.80. In [24], the authors proposed the usage of the complete PSL image, which contains healthy skin and artefacts. Whereby, there is a probability of generating misclassification of the recognizing patterns that belong to these objects. By contrast, the proposed system extracts the proposed features from the region of interest of an image only for the entire classification process. This guarantees that the feature extraction is performed exactly in the lesion image.

Moreover, our novel CAD employed the ISIC 2018 database [21] that contains more than 10,000 dermoscopy images that are authenticated by experts. Additionally, because of the unbalanced data that are present in this database, we applied the data augmentation procedure given in Section 4.1.4. This guarantees the robustness of the obtained classifications results. In contrast, the CAD DermoDeep system [24] obtained the experimental results using a synthetic database constructed by four different databases (private and public). In this case, an equal number of melanoma and benign skin lesions were subjectively selected from each of the four databases. This system showed slightly improved performance results as those reported in this study. In our opinion, such an approach does not guarantee that the same high performance can be repeated using data that are not previously preselected.

The proposed system used a data augmentation technique and presented the performance analysis for all images contained in the database and not only those that have been preselected according to a subjective criterion that has no statistical justification.

Finally, our proposed system was developed with medical-based and deep learning features, whereby the system employed data from both sets of features and merged them, applying the MI criterion. As a result, the system enhances the recognition of melanoma and nevus lesions compared to the use of a fully deep learning approach that is extremely computationally expensive to train, requires substantial amounts of labeled data and does not recognize dermoscopic features established in the ABCD algorithm.

5. Conclusions and Future Work

In this study, a novel competitive CAD system was designed to differentiate melanoma from nevus lesions. Different from commonly proposed CAD systems, the novel method employs handcraft features based on the medical algorithm ABCD rule and deep learning features and applies a transfer learning method as a feature extractor. Additionally, in the proposed system, the set features are fused using an MI metric that, in contrast with state-of-the-art systems, can select the most significant features in accordance with their influence on binary classification decisions.

The performance of the proposed system has been evaluated; the system achieved an accuracy of 92.4%, an IBA of 0.80, and an MCC of 0.7953 using a balanced dataset. The system is competitive against the performance of other, state-of-the-art systems.

The proposed CAD system can help inexperienced physicians to visually distinguish the medical features to be applied. Furthermore, it could be used to provide a second opinion to a dermatologist. Our future work will consist of designing a method for multiclass classification using both sets of features, thus permitting the diagnosis of several diseases found in the ISIC challenge dataset.

Author Contributions: Methodology: J.-A.A.-D. and V.P.; software: J.-A.A.-D. and H.C.-F.; formal analysis: J.-A.A.-D., V.P. and S.S.; investigation: J.-A.A.-D. and V.P.; resources: J.-A.A.-D. and V.P.; data curation: J.-A.A.-D. and V.P.; writing—original draft preparation: J.-A.A.-D. and V.P.; writing—review and editing: J.-A.A.-D., V.P., S.S. and H.C.-F. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: The authors would like to thank the Instituto Politecnico Nacional (Mexico) and the Consejo Nacional de Ciencia y Tecnologia (Mexico) for their support in this work.

Conflicts of Interest: The authors declare that there is no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ELM	Epiluminance Microscopy
MD	Medical Doctor
PSL	Pigmented Skin Lesion
ROI	Region of Interest
GLCM	Gray Co-occurrence Matrix
DL	Deep Learning
CAD	Computer-Aided Detection
SVM	Support Vector Machine
RVM	Relevant Vector Machine
LR	Logistic Regression
CNN	Convolutional Neural Network
TF	Transfer Learning
MI	Mutual Information
K-NN	K-Nearest Neighborhood
MCC	Matthews Correlation Coefficient

References

1. Skin Cancers. Available online: <http://www.who.int/uv/faq/skincancer/en/index1.html> (accessed on 15 January 2020).
2. Skin Cancer. Available online: <https://www.wcrf.org/dietandcancer/skin-cancer> (accessed on 15 January 2020).
3. Baldi, A.; Quartulli, M.; Murace, R.; Dragonetti, E.; Manganaro, M.; Guerra, O.; Bizzi, S. Automated Dermoscopy Image Analysis of Pigmented Skin Lesions. *Cancers* **2010**, *2*, 262–273. [[CrossRef](#)]
4. Almaraz-Damian, J.A.; Ponomaryov, V.; Rendon-Gonzalez, E. Melanoma CADe based on ABCD Rule and Haralick Texture Features. In Proceedings of the 2016 9th International Kharkiv Symposium on Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves (MSMW), Kharkiv, Ukraine, 20–24 June 2016; pp. 1–4.

5. Li, Y.; Shen, L. Skin Lesion Analysis towards Melanoma Detection Using Deep Learning Network. *Sensors* **2018**, *18*, 556. [[CrossRef](#)]
6. Lopez, A.R.; Giro-i-Nieto, X.; Burdick, J.; Marques, O. Skin lesion classification from dermoscopic images using deep learning techniques. In Proceedings of the 2017 13th IASTED International Conference on Biomedical Engineering (BioMed), Innsbruck, Austria, 20–21 February 2017; pp. 49–54.
7. Hosny, K.M.; Kassem, M.A.; Foad, M.M. Classification of skin lesions using transfer learning and augmentation with Alex-net. *PLoS ONE* **2019**, *14*, e0217293. doi:10.1371/journal.pone.0217293. [[CrossRef](#)]
8. Castillejos, H.; Ponomaryov, V.; Nino-De-Rivera, L.; Golikov, V. Wavelet Transform Fuzzy Algorithms for Dermoscopic Image Segmentation. *Comput. Math. Method. Med.* **2012**, *2012*, 578721. [[CrossRef](#)] [[PubMed](#)]
9. Nachbar, F.; Stolz, W.; Merkle, T.; Cognetta, A.B.; Vogt, T.; Landthaler, M.; Bilek, P.; Braun-Falco, O.; Plewig, G. The ABCD Rule of Dermatoscopy. *J. Am. Acad. Dermatol.* **1994**, *30*, 551–559. [[CrossRef](#)]
10. Zalaudek, I.; Argenziano, G.; Soyer, H.P.; Corona, R.; Sera, F.; Blum, A.; Braun, R.P.; Cabo, H.; Ferrara, G.; Kopf, A.W.; et al. Three-point checklist of dermoscopy: An open internet study. *Br. J. Dermatol.* **2005**, *154*, 431–437. [[CrossRef](#)] [[PubMed](#)]
11. Henning, J.S.; Dusza, S.W.; Wang, S.Q.; Marghoob, A.A.; Rabinovitz, H.S.; Polsky, D.; Kopf, A.W. The CASH (color, archi-ecture, symmetry, and homogeneity) algorithm for dermoscopy. *J. Am. Acad. Dermatol.* **2007**, *56*, 45–52. [[CrossRef](#)] [[PubMed](#)]
12. Stolz, W.; Riemann, A.; Cognetta, A.B.; Pillet, L.; Abmayr, W.; Hölzel, D.; Bilek, P.; Nachbar, F.; Landthaler, M.; Braun-Falco, O. ABCD rule of dermatoscopy: A new practical method for early recognition of malignant melanoma. *Eur. J. Dermatol.* **1994**, *4*, 521–527.
13. Argenziano, G.; Fabbrocini, G.; Carli, P.; De Giorgi, V.; Sammarco, E.; Delfino, M. Epiluminescence Microscopy for the Diagnosis of Doubtful Melanocytic Skin Lesions: Comparison of the ABCD Rule of Dermatoscopy and a New 7-Point Checklist Based on Pattern Analysis. *Arch. Dermatol.* **1998**, *134*, 1563–1570. [[CrossRef](#)]
14. Melanoma Education Foundation. Finding Melanoma Early: Warning Signs & Photos. Available online: <https://www.skincheck.org/Page4.php> (accessed on 2 February 2020).
15. MoleMap NZ Official Site. The EFG of Nodular Melanomas. Available online: <https://www.molemap.co.nz/knowledge-centre/efg-nodular-melanomas> (accessed on 2 February 2020).
16. Jensen, J.D.; Elewski, B.E. The ABCDEF Rule: Combining the “ABCDE Rule” and the “Ugly Duckling Sign” in an Effort to Improve Patient Self-Screening Examinations. *J. Clin. Aesthet. Dermatol.* **2015**, *8*, 15.
17. Kalkhoran, S.; Milne, O.; Zalaudek, I.; Puig, S.; Malvehy, J.; Kelly, J.W.; Marghoob, A.A. Historical, Clinical, and Dermoscopic Characteristics of Thin Nodular Melanoma. *Arch. Dermatol.* **2010**, *146*, 311–318. [[CrossRef](#)]
18. Adjed, F.; Gardezi, S.J.S.; Ababsa, F.; Faye, I.; Dass, S.C. Fusion of structural and textural features for melanoma recognition. *IET Comput. Vis.* **2018**, *12*, 185–195. [[CrossRef](#)]
19. Mendonça, T.; Ferreira, P.M.; Marques, J.; Marcal, A.R.S.; Rozeira, J. PH²—A dermoscopic image database for research and benchmarking. In Proceedings of the 35th International Conference of the IEEE Engineering in Medicine and Biology Society, Osaka, Japan, 3–7 July 2013.
20. Hagerty, J.R.; Stanley, R.J.; Almubarak, H.A.; Lama, N.; Kasmi, R.; Guo, P.; Stoecker, W.V. Deep Learning and Handcrafted Method Fusion: Higher Diagnostic Accuracy for Melanoma Dermoscopy Images. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 1385–1391. [[CrossRef](#)]
21. Codella, N. Rotemberg, V.; Tschandl, P.; Celebi, M.E.; Dusza, S.; Gutman, D.; Helba, B.; Kalloo, A.; Liopyris, K.; Marchetti, M.; et al. Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). *arXiv* **2018**, arXiv:1902.03368.
22. Li, X.; Wu, J.; Jiang, H.; Chen, E.Z.; Dong, X.; Rong, R. Skin Lesion Classification Via Combining Deep Learning Features and Clinical Criteria Representations. *bioRxiv* **2018**, bioRxiv:382010.
23. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Y.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 3149–3157.
24. Abbas, Q.; Celebi, M.E. DermoDeep—A classification of melanoma-nevus skin lesions using multi-feature fusion of visual features and deep neural network. *Multimed. Tool. Appl.* **2019**, *78*, 23559–23580. [[CrossRef](#)]
25. Krizhevsky, A. *Learning Multiple Layers of Features from Tiny Images*; Technical Report TR-2009; University of Toronto: Toronto, ON, Canada, 2009.

26. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
27. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
29. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826.
30. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2016; pp. 1800–1807.
31. Huang, G.; Liu, Z.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2016; pp. 2261–2269.
32. Orea-Flores, I.Y.; Gallegos-Funes, F.J.; Arellano-Reynoso, A. Local Complexity Estimation Based Filtering Method in Wavelet Domain for Magnetic Resonance Imaging Denoising. *Entropy* **2019**, *21*, 401. [[CrossRef](#)]
33. Goritskiy, Y.; Kazakov, V.; Shevchenko, O.; Mendoza, F. Model of Random Field with Piece-Constant Values and Sampling-Restoration Algorithm of Its Realizations. *Entropy* **2019**, *21*, 792. [[CrossRef](#)]
34. Yang, M.; Kpalma, K.; Joseph, R. A Survey of Shape Feature Extraction Techniques. In *Pattern Recognition Techniques, Technology and Applications*; Yin, P.-Y., Ed.; InTech: London, UK, 2008. Available online: http://www.intechopen.com/books/pattern_recognition_techniques_technology_and_applications/a_survey_of_shape_feature_extraction_techniques (accessed on 10 January 2020). [[CrossRef](#)]
35. Alceu, B.; Fornaciali, M.; Valle, E.; Avila, S. (De) Constructing Bias on Skin Lesion Datasets. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–20 June 2019; pp. 2766–2774.
36. Sirakov, N.M.; Mete, M.; Chakrader, N.S. Automatic boundary detection and symmetry calculation in dermoscopy images of skin lesions, Department of Mathematics, 2 Department of Computer Science. *IEEE Intern. Conf. Image Process.* **2011**, *1*, 1637–1640.
37. Haralick, R.; Shanmugam, K.; Dinstein, I. Textural Features for Image Classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *3*, 610–621. [[CrossRef](#)]
38. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
39. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intell. Neurosci.* **2018**, *2018*, 7068349. [[CrossRef](#)] [[PubMed](#)]
40. Chollet, F. *Deep Learning with Python*, 1st ed.; Manning Publications Co.: Greenwich, CT, USA, 2017.
41. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In *Proceedings of the 27th International Conference on Neural Information Processing Systems—Volume 2*; MIT Press: Cambridge, MA, USA; pp. 3320–3328.
42. Rawat, W.; Wang, Z. Deep convolutional neural networks for image classification: A comprehensive review. *Neural Comput.* **2017**, *29*, 2352–2449. [[CrossRef](#)] [[PubMed](#)]
43. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [[CrossRef](#)]
44. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*. [[CrossRef](#)]
45. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**. [[CrossRef](#)]
46. Bommert, A.; Sun, X.; Bischl, B.; Rahnenführer, J.; Lang, M. Benchmark for Filter Methods for Feature Selection in High-Dimensional Classification Data. *Comput. Stat. Data Anal.* **2020**, *143*, 106839. [[CrossRef](#)]
47. Karczmarek, P.; Pedrycz, W.; Kiersztyn, A.; Rutka, P. A study in facial features saliency in face recognition: An analytic hierarchy process approach. *Soft Comput.* **2017**, *21*, 7503–7517. [[CrossRef](#)]
48. Kozachenko, L.F.; Leonenko, N.N. Sample Estimate of the Entropy of a Random Vector. *Probl. Peredachi Inf.* **1987**, *23*, 9–16.

49. Kraskov, A.; Stögbauer, H.; Grassberger, P. Estimating mutual information. *Phys. Rev. E* **2004**, *69*, 066138. [[CrossRef](#)] [[PubMed](#)]
50. Houghton, C. Calculating mutual information for spike trains and other data with distances but no coordinates. *R. Soc. Open Sci.* **2015**, *2*. [[CrossRef](#)] [[PubMed](#)]
51. Ross, B.C. Mutual information between discrete and continuous data sets. *PLoS ONE* **2014**, *9*. [[CrossRef](#)]
52. Vapnik, V. *Statistical Learning Theory*; John Wiley: New York, NY, USA, 1998.
53. Tipping, M.E. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.* **2001**, *1*, 211–244.
54. Bishop, C. Probabilistic graphical models and their role in machine learning. In Proceedings of the NATO ASI–LTP 2002 Tutorial, Leuven, Belgium, 8–19 July 2002.
55. Chollet, F. Keras. 2015. Available online: <https://keras.io> (accessed on 15 January 2020).
56. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
57. Coelho, L.P. Mahotas: Open source software for scriptable computer vision. *J. Open Res. Softw.* **2013**, *1*, e3. [[CrossRef](#)]
58. Wen, Z.; Shi, J.; Li, Q.; He, B.; Chen, J. ThunderSVM: A fast SVM library on GPUs and CPUs. *J. Mach. Learn. Res.* **2018**, *19*, 797–801.
59. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *J. Mach. Learn. Res.* **2017**, *18*, 559–563.
60. Powers, D.M. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *J. Mach. Learn. Technol.* **2011**. [[CrossRef](#)]
61. Tschandl, P.; Rosendahl, C.; Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci Data* **2018**, *5*, 180161. [[CrossRef](#)] [[PubMed](#)]
62. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [[CrossRef](#)]
63. Celebi, M.E.; Kingravi, H.A.; Uddin, B.; Iyatomi, H.; Aslandogan, Y.A.; Stoecker, W.V.; Moss, R.H. A methodological approach to the classification of dermoscopy images. *Comput. Med. Imag. Graph.* **2007**, *31*, 362–373. [[CrossRef](#)] [[PubMed](#)]
64. Capdehourat, G.; Corez, A.; Bazzano, A.; Alonso, R.; Musé, P. Toward a combined tool to assist dermatologists in melanoma detection from dermoscopic images of pigmented skin lesions. *Pattern Recognit. Lett.* **2011**, *32*, 2187–2196. [[CrossRef](#)]
65. García, V.; Mollineda, R.A.; Sánchez, J.S. *Index of Balanced Accuracy: A Performance Measure for Skewed Class Distributions*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Berlin/Heidelberg, Germany, 2009; Volume 5524, pp. 441–448. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).