Article

A Standardized Approach for Skin Detection: Analysis of the Literature and Case-Studies

Loris Nanni [1,*], Andrea Loreggia [2], Alberto Dorizza[1], Alessandra Lumini[3]

[1] Department of Information Engineering, University of Padova, Italy; loris.nanni@unipd.it alberto.dorizza@studenti.unipd.it
[2] Department of Information Engineering, University of Brescia, Italy; andrea.loreggia@unibs.it
[3] DISI, Università di Bologna, Via dell'università 50, 47521 Cesena, Italy; alessandra.lumini@unibo.it
[*] Correspondence: loris.nanni@unipd.it;

Abstract: Skin detection, the process of distinguishing between skin and non-skin regions in a digital image, is widely used in a variety of applications ranging from hand gesture analysis to body part tracking to facial recognition. Skin detection is a challenging problem that has received a lot of attention from experts and proposals from the research community in the context of intelligent systems, but the lack of common benchmarks and unified testing protocols has hampered fairness among approaches. Comparisons are very difficult. Recently, the success of deep neural networks has had a major impact on the field of image segmentation detection, resulting in various successful models to date. In this work, we survey the most recent research in this field and propose fair comparisons between approaches using several different datasets. The main contributions of this work are: (i) a comprehensive literature review of approaches to skin color detection and a comparison of approaches that may help researchers and practitioners choose the best method for their application; (ii) a comprehensive list of datasets that report ground truth for skin detection; (iii) a framework for evaluating and combining different skin detection approaches. Moreover, we proposed an ensemble of convolutional neural networks and transformers that obtains state of the art performance. All the code is made publicly available at https://github.com/LorisNanni

Keywords: skin classification; skin detection; skin segmentation; skin database; neural networks

1. Introduction

Skin texture and color are important cues that people use to understand different cultural aspects of each other (health, ethnicity, age, beauty, wealth, etc.). The presence of skin color in an image or video indicates the presence of a person in such media. Therefore, over the past two decades, extensive research in the context of professional and intelligent systems has focused on video and image skin detection. Skin detection is the process of distinguishing between "skin" and "non-skin" regions in a digital image and consists of performing binary classification of pixels and performing fine segmentation to define skin region boundaries. Skin detection is now an advanced process, involving not only model training but many additional methods including data pre- and post-processing.

This survey is a revised version of [1]. The aim of this study is to cover the recent literature in deep-learning-based skin segmentation by providing a comprehensive review with specific insights into different aspects of the proposed methods. This includes the training data, the network architectures, loss functions, training strategies, and specific key contributions. Moreover, we propose a new ensemble, based on convolutional neural networks and transformers, that provides state-of-the-art performance.

Skin detection is used as a preparatory step for medical imaging, such as detection of skin cancer [2], [3], skin diseases in general [4], [5], or skin lesions in general [6], [7]. It is

doi:10.20944/preprints202211.0495.v1

also adopted for face detection [8] and body tracking [9], hand detection [10], biometric authentication [11], and many others [12]–[14].

This article provides an extensive review about the ways techniques from artificial intelligence, deep learning, and machine learning systems are designed and developed to resolve the problem of skin detection.

A characteristic that helps distinguish between skin pixels and non-skin pixels is the pixel color. Still, achieving skin tone consistency in different lighting, different ethnicities, and different sensing devices, is a very difficult task.

Moreover, when used as a preliminary step for other applications, skin detection is computationally efficient, invariant to geometric transformations, partial occlusions, or changes in body pose/expression, and can be applied to complex or simulated skin. It is not affected by the background of the capture device.

Pixel intensity depends on scene conditions, such as reflectance and light, strongly influencing color consistency, that results to be the most influential factor in determining skin color [15]. To be effective when lighting conditions change rapidly, some skin detection approaches use image preprocessing strategies based on color constancy (i.e. color correction methods based on luminance estimation) and/or dynamic adaptation methods (i.e. transformations of skin color patterns under changing lighting conditions. A possible solution is to consider additional data not in the visible spectrum (i.e., infrared images [16] or spectral images [17]), but these sensors are not suitable for all applications and requires a higher acquisition cost that limits their use for specific problems.

A more specific application for skin detection is hand segmentation, which aims at segmenting the hand profile: this task becomes particularly challenging when the segmentation of a hand is over the face or other portions of skin. Recent approaches to solve these problems are adopting very deep neural network structures and collecting new large-scale datasets on real-life scenes to increase the diversity and complexity [18], [19]. New studies try to reduce the size of the network models, refining existing ones, in order to perform with few parameters and increasing the inference speed while achieving high accuracy during the hand segmentation process [19].

Recent surveys are almost all focused on the adoption of artificial intelligence techniques for the early detection of skin cancer. They observed the increasing interest of researchers for deep learning techniques [20], [21]. A key point that emerges from this analysis is the amount of studies focusing on the automatic detection of lesions [22] or cancer. This is reported in a recent systematic literature review [23] which identified 14,224 studies to the early diagnosis of skin cancer published between Jan 1, 2000, to Aug 9, 2021 in MEDLINE, Embase, Scopus, and Web of Science. Another systematic review [24] identified 21 open access datasets containing 106 950 skin lesion images which can be used for training and testing algorithm for skin cancer diagnosis.

The major contributions of this research work are:

- An exhaustive literature review of skin color detection approaches with a detailed description of methods freely available.
- Collection and study of virtually any real skin detection dataset available in the literature.
- A framework for comparing different approaches for skin detection.
- Four different deep learning architectures have been trained for skin detection. The proposed ensemble obtains state of the art performance.

1. Skin Detection Approaches

Some skin detection approaches are based on the assumption that the skin color can be detected in a specific color space from the background color using clustering rules. This

doi:10.20944/preprints202211.0495.v1

assumption holds true in constrained environments where both the ethnicity and background color of the people are known, but in complex images taken under unconfined conditions, where the subject has a wide range of human skin tones, it's a very difficult task [25]. There are a lot of challenging factors that influence the performance of a skin detector:

- Ethnicity, age and other human characteristics. Skin color ranges from white to dark brown among human racial groups, the transition from fresh skin to dry skin related to the age determines a strong variation of tones.
- Shooting conditions connected with camera characteristics and lighting variations have a large effect on the appearance of skin. In general, changes in lighting level or light source distribution determine the presence of shadows and changes in skin color.
- Skin Paint: tattoos and makeup affect the aspect of the skin.
- Complex Background: The presence of skin-colored objects in the background can fool the skin detector.

Existing skin detection models can be classified according to several aspects of the procedure:

1. The presence of preprocessing steps intended to reduce the effects of different acquisition conditions, such as color correction and light removal [26] or dynamic adjustment [27];
2. The selection of the most suitable skin color model [28]. The performance of different color models are considered [25], [29], [30] (e.g., RGB, normalized RGB, the perceptual model, creating new color spaces, and others).
3. The formulation of the problem based on either segmenting the image into human skin regions or treating each pixel as skin or non-skin, regardless of its neighbors. There are few area-based skin color detection methods [31]–[34] including some recent methods (e.g., [35], [36]) based on convolutional neural networks.
4. The type of approach [37]: rule-based methods define explicit rules for determining skin color in an appropriate color space; machine learning approaches use nonparametric or parametric learning approaches to estimate the color distribution of the training.
5. According to other taxonomies from the field of machine learning [38], that extend simple classification to parametric and non-parametric approaches. Statistical methods include parametric methods based on Bayes' rule of mixed models [39]. Neural network models [40], [41] can be used to segment color images based on color and texture information. Diffusion-based methods [42], [43] extend the analysis to adjacent pixels to improve classification performance. Adaptive techniques [44] rely on coordination patterns to adapt to specific conditions (e.g., lighting, skin color, background). Model calibration often provides performance benefits but increases computation time. Hyperspectral models [45] are based on acquisition instruments with hyperspectral capabilities. Despite the benefits of the availability of spectral information, this approach is not considered in this work, as it only applies to specially collected datasets. SVM-based systems are parametric models based on SVM classifiers. When the SVM classifier is trained by active learning, this class also repeats the adaptive method [14]. Blending methods are methods based on combining different machine learning approaches [46].
6. Deep learning methods have shown outstanding potential in dermatology for skin lesion detection and identification [6], however, they usually require annotations beforehand and can only classify lesion classes seen in the training set. Moreover, large-scale, open-sourced medical datasets normally have far fewer annotated classes than in real life, further aggravating the problem.

When the detection conditions are controlled, the identification of skin regions is fairly straightforward, for example, in some gesture recognition applications, hand images are captured using flatbed scanners and have a dark, unsaturated background [47]. For this reason, several simple rule-based methods have been proposed, in addition to approaches based on sophisticated and computationally expensive techniques. These methods are preferred in some applications because they are easier to understand, implement, and reuse, more efficient, and ready to use. Effective enough at the same time. Simple rule-based methods are typically not even tested against pure skin detection benchmarks, but as a step in more complex tasks (face recognition, hand gesture recognition, etc.). An example of a method belonging to this class is the work of [47], who conducted a study of different color models and concluded that there is no obvious advantage to using a uniform color space for perception, therefore proposing an approach based on a simple RGB look-up table. First, simple approaches were based on parameterizing color spaces as a preliminary step to detect skin regions [48] or to improve the learning phase allowing for reduced amount of data in the training phase [49]. More complex approaches perform spatial permutations to deal with the problem of light variations [50]. The creation of new color spaces is reached by introducing linear and nonlinear conversions of RGB color space [30] or applying Principal Component Analysis and a Genetic Algorithm to discover the optimal representation [51]. Recent studies mimic alternate representations of images by developing color-based data augmentations to enrich the dataset with artificial images [29].

When skin detection is performed in uncontrolled situations the current state of the art is obtained by deep learning methods [36], [52], [53]. Often, convolutional neural networks are preferred and implemented in a variety of computer vision tasks, for instance applying different structures to identify the best suitable for skin detection [35], [53].

In [52] a patch-wise approach for skin segmentation, based on deep neural networks that use image patches as processing units instead of pixels, is proposed. An image patch dataset is properly collected for training purposes, and the trained deep skin samples are embedded in a sliding window framework, to provide competitive performance in skin region detection and skin detection.

Another approach [36] integrates fully convolutional neural networks with recurrent neural networks to develop an end-to-end network for human skin detection.

The main problem identified in the analysis of the literature is the heterogeneity of protocols adopted in training and assessing the proposed models. This makes the comparison with traditional existing approaches very difficult, due to the different testing protocols. For instance, recently, a research study compares different deep learning approaching on different datasets using different training sets [54]. In this work we adopt a standard protocol to train the models and validate the results.

Now, we list some of the most interesting approaches proposed in the last twenty years.

- GMM [39] is simple skin detection approach based on Gaussian mixture model trained to classify non-skin and skin pixels in RGB space.
- Bayes [39] is a fast and effective Bayesian classifier trained in the RGB color space using 2000 images from the ECU data set.
- SPL [55] is a pixel-based skin detection approach that uses a look-up table (LUT) to determine skin probabilities in the RGB domain. For the test image, the probability that each pixel x is occluded and then apply a threshold τ to determine whether it is not occluded/nose.
- Cheddad [56] is a pixel-based and real-time approach which reduces the RGB color space to a 1D space derived from differentiating the grayscale map and

the non-red encoded grayscale version. The classification is performed using a skin probability which delimits the lower and upper bounds of the skin cluster and the final decision depends on a classification threshold τ.

- Chen [41] is a statistical skin color model, which is specifically suited to be implemented on hardware. The 3D skin cube is represented as three 2D sides calculated as the difference of two-color channels: sR=R-G, sG=G-B, sB=R-B. The skin cluster region is delineated in the transformed space.
- SA1 [57] is a skin detection method based on spatial analysis. Starting with the skin probability map obtained with the pixel color detector, the first step in spatial analysis is to correctly select high-probability pixels as skin seeds. The second step is to find the shortest path to propagate the "shell" from each seed to each individual pixel. During the enhancement process, all non-adjacent pixels are marked as non-skin.
- SA2 [44] is a method based on spatial analysis which uses both color and textural features to determine the presence of skin. It extracts the textural features from the skin probability maps rather than from the luminance channel: therefore simple textural statistics are computed from each pixel's neighborhood in the probability map using kernels of different sizes. Then skin and non-skin pixels are transformed into two classes of feature vectors whose size is reduced by Linear Discriminant Analysis (LDA) to increase their discriminating power. Finally, the spatial analysis method proposed in [57] and described above is used for seed extraction and propagation using the distance transform.
- SA3 [58] It is a self-adaptive method that combines probabilistic mapping and local skin color patterns generated by spatial analysis to describe skin regions. It is an evolution of methods based on spatial analysis [44].
- DYC [59] is a skin detection approach which works in the YCbCr color space and takes into account the lighting conditions. The method is based on the dynamic generation of the skin cluster range both in the YCb and YCr subspaces of YCbCr and on the definition of correlation rules between the skin color clusters.
- In [1] and [60] several deep learning segmentation approaches are compared, SegNet, U-Net, DeepLabv3+, HarD-NetMSEG (Harmonic Densely Connected Network)[1] [61] and Polyp-PVT [62] a deep learning segmentation model based on a transformer encoder, i.e. PVT (Pyramid Vision Transformer)[2].
- ALDS [63] is a framework based on probabilistic approach that initially utilizes active contours and watershed merged mask for segmenting out the mole and later SVM and Neural Classifier are applied for the classification of the segmented mole.
- DNF-OOD [6] applies a non-parametric deep forest-based approach to the problem of out-of-distribution (OOD) detection
- SANet [64] contains two sub-modules: superpixel average pooling and superpixel attention module. Authors introduce a superpixel average pooling to reformulate the superpixel classification problem as a superpixel segmentation problem, and a superpixel attention module is utilized to focus on discriminative superpixel regions and feature channels.

[1] https://github.com/james128333/HarDNet-MSEG - Last access on November 5, 2022

[2] https://github.com/DengPingFan/Polyp-PVT - Last access on November 5, 2022

- OR-Skip-Net [65] is an outer residual skip connection designed and implemented to deal with skin segmentation in challenging environments irrespective of skin color, and to eliminate the cost of the preprocessing. The model is based on deep convolutional neural network.
- In [29] a new approach for skin detection that performs a color-based data augmentations, to enrich the dataset with artificial images to mimic alternate representations of the image, is proposed. Data augmentation is performed in the HSV (Hue, Saturation, Value) space. For each image in a dataset, this approach creates fifteen new images.
- In [30] a different color space is proposed, its goal is to represent the information in images introducing a linear and nonlinear conversion of RGB color space through a conversion matrix (W matrix). W matrix values are optimized to meet two conditions: firstly, maximizing the distance between centers of skin and non-skin classes, and secondly minimizing entropy of each class. The classification step is done with the adoption of neural networks and an adaptive neuro-fuzzy inference system called ANFIS.
- SSS-Net [66] captures the multi-scale contextual information and refines the segmentation results especially along object boundaries. It also reduces the cost of the preprocessing as well.
- SCMUU [67] stands for skin color model updating units, it performs skin detection by using the similarity of adjacent frames in a video. The method is based on the assumption that the face and other parts of the body have a similar skin color. The color distribution is used to build chrominance components of the YCbCr color space referring facial landmarks.
- SKINNY [68] is a U-net based model. The model has more depth levels, it uses wider convolutional kernels for the expansive path, and employs inception modules alongside dense blocks to strengthen feature propagation. In such a way, the model is able to increase the multi-scale analysis range.

A rough classification of the most used methods is reported in Table 1.

Table 1. Rough classification of the tested approaches

	GMM	*Bayes*	*SPL*	*Cheddad*	*Chen*	*SA1*	*SA2*	*SA3*	*DYC*	*SegNet*	*U-Net*	*DeelLabv3+*	*PVT*	*HSN*

Preprocessing steps														
None	x	x	x	x	x					x	x	x	x	x
Dynamic adaptation						x	x	x	x					
Color space														
Basic color spaces	x	x	x							x	x	x	x	x
Perceptual color spaces						x	x	x						
Orthogonal color spaces									x					
Other (e.g. Color ratio)				x	x									
Problem formulation														
Segmentation based						x	x	x		x	x	x	x	x
Pixel based	x	x	x	x	x				x					
Type of pixel classification														
Rule based				x	x				x					
Machine learning: parametric	x	x												
Machine learning: non-parametric			x											
Type of classifier														
Statistical		x	x											
Mixture techniques	x													
Adaptive methods						x	x	x						
CNN										x	x	x		
Transformer													x	x

1.1. Hand Segmentation

As is the case in skin detection, deep learning methods are used for hand segmentation to achieve cutting-edge performance. Current state-of-the-art approaches for human hand detection [69] have achieved great success by making good use of multiscale and contextual information, but still remain unsatisfactory for hand segmentation, especially in complex scenarios. In this context, deep approaches have faced some difficulties, such as the clutter in the background that hinders the reliable detection of hand gestures in real-world environments. Moreover, frequently the task described in literature is not clear: for instance, some studies report a hand segmentation task but in the empirical analysis the authors used a mask to recognize the whole arm [70], this affects the final results as makes the goal being a skin segmentation task rather than a hand detection.

Among the several recent studies focused on hand segmentation we cite:

- Refined U-net [19], authors propose a refinement of U-net that perform with few parameters and increases the inference speed while achieving high accuracy during the hand segmentation process.

- CA-FPN [69], it stands for Context Attention Feature Pyramid Network, is a model designed for human hand detection. In this method, a novel Context Attention Module (CAM) is inserted into the feature pyramid networks. The CAM is designed to capture relative contextual information for hands and build long-range dependencies around hands.

In this work, we do not make a complete survey of hand segmentation, but we treat the task as a subtask for skin segmentation and use some datasets collected for this task to show the robustness of the proposed ensemble of skin detectors. We show that the proposed method gives good performance in this domain without an ad-hoc training.

2. Materials and Methods

This section presents some of the most interesting models and methods for training used in the field of skin detection. We also report a brief overview of all the main available loss functions developed for skin segmentation. Some of the following approaches have been included for the creation of the proposed ensemble.

2.1. Deep Learning for Semantic Image Segmentation

In order to solve the problem of semantic segmentation, several deep learning models have been proposed in the specialized literature.

Semantic segmentation aims to identify objects in an image and their relative boundaries. Therefore, the main purpose is to assign classes at the pixel level, which is a task achieved thanks to FCN (Fully Convolutional Networks). FCN has very high performance and unlike CNN, it uses a fully convolutional last layer instead of a fully connected layer. [71]. FCN and autoencoder are combined to obtain a deconvolutional network like U-Net. U-Net represents the first attempt to use autoencoders in image segmentation operations. Autoencoders can shrink the input while increasing the number of features used to describe the input space. Another symbolic example can be found in SegNet [72].

DeepLab [73] is of a set of autoencoder models provided by Google and has shown excellent results in semantic segmentation applications [73]–[76]. The key features included to ensure better performance comprehend an advanced convolution to reduce merging and transition effects and significantly increase resolution; information is obtained by Atrous Spatial Pyramid Pooling of different scales, and a combination of CNNs and probabilistic graphical models can determine object boundaries. In this work, we adopted an extension of the suite developed by Google DeepLabV3+ [75]. We found two major innovations in DeepLabV3+. First, a 1x1 Convolution and Packet Normalization in Atrous Spatial Pyramid Pooling. Second, a set of parallel and cascaded convolution scaling modules. One of the main features of this extension is a depth roll and spot roll decoder. Different depths at the same location but different channels use the same channel at different locations in a point. We can consider other features of the model structure to achieve a different design for your framework. In fact, the architecture model itself is only a used choice. Here, we consider ResNet101 [77] as backbone for DeepLabV3+, ResNet101 is a very popular CNN that obtains residual functions by referencing block inputs (for a complete list of CNN structures please refer to [78]). It is pre-trained on the VOC segmentation dataset and then tuned using the parameters specified on the github page[3]. We adopted the same parameters to prevent overfitting (i.e. same parameters in all the training datasets):

- initial learning rate=0.01;
- number of epoch=10 (using the simple data augmentation approach DA1, see section 3.3) or 15 (the latter more complex data augmentation approach DA2,

[3] https://github.com/matlab-deep-learning/pretrained-deeplabv3plus

see section 3.3, since the slower convergence using this larger augmented training set);

- momentum=0.9;
- L2Regularization=0.005;
- Learning Rate Drop Period=5;
- Learning Rate Drop Factor=0.2;
- Shuffle training images every-epoch;
- Optimizer=SGD (stochastic gradient descent).

We present an ensemble based on DeepLabV3+, HarDNet-MSEG [61], Polyp-PVT [62], and Hybrid Semantic Network (HSN) [79]. HarD-Net-MSEG (Harmonic Densely Connected Network) [61] is a model influenced by densely connected networks, that can reduce memory consumption by diminishing aggregation with the reduction of most connection layers to the DenseNet layer. Also, the input/output channel ratio is balanced (due to increased connections) as the layer channel width increases.

Polyp-PVT [62] is based on a pure convolutional network of transformers that aims to achieve high-resolution displays from microscopic inputs. The computational cost of the model decreases with the depth of the model through progressive pyramidal reduction. The Spatial Reduction Focusing (SRA) layer was introduced to further reduce the computational complexity of the system. The decoder part is based on a cascaded fusion module (CFM) used to collect the semantic and location information of foreground pixels from high-level features, a camouflage identification module (CIM) is applied to capture skin information disguised in low-level features and a similarity aggregation module (SAM) is used to extend the pixel features of the skin area with high-level semantic position information to the entire image, thereby effectively fusing cross-level features.

Hybrid Semantic Network [79] leverages transformers and convolutional neural networks. HSN includes Cross Semantic Attention Module (CSA), Hybrid Semantic Complement Module (HSC), and Multi-Scale Prediction Module (MSP). The authors introduced a new CSA module, which fills the gap between low-level and high-level functions by an interactive mechanism that replaces the two semantics of different NNs. Moreover, HSN adopts a new HSC module that captures both long-range dependencies and local scene details using the two-way architecture of Transformer and CNN. In addition, the MSP module can learn weights for combining prediction masks at the decoder stage.

All the different network topologies are trained using the Structure loss function, which is the sum of weighted IoU loss and weighted binary cross entropy (BCE) loss, where weights are related to pixel importance (which is calculated according to the difference between the center pixel and its surroundings). We employed the Adam or SGD optimization algorithms for HardNet-MSEG and AdamW for PVT-Polyp and HSNet. The learning rate is 1e−4 for HardNet and PVT-Polyp and 5e-5 for HSNet (decaying to 5*10-6 after 30 epochs. The whole network is trained in an end-to-end manner for 100 epochs with a batch size of 20 for HardNet-MSEG and 8 for PVT-Polyp and HSNet. The output prediction map is generated after a sigmoid operation.

Moreover, as in the original code of PVT and HSN each output map is normalized between [0,1], we have performed the same normalization also for HardNet-MSEG and DeepLabV3+.

2.2. Loss functions

Loss functions play an important role in any statistical model; they define what is and what is not a good prediction, so the choice of the right loss function determines the quality of the estimator.

In general, loss functions affect the training duration and model performance. In semantic segmentation operations, pixel cross-entropy is one of the most common and loss functions. It works at the pixel level and checks whether the predicted signature of a given pixel matches the correct answer.

An unbalanced dataset with respect to labels is one of the main problems for this approach, that can be solved adopting a counterweight. A recent study allows a comprehensive review of image segmentation and loss functions [80].

In this section, we detail some of the most used loss functions in segmentation field. Table 2 reports all the mathematical formulation of the following loss functions:

- Dice Loss is a commonly accepted measure for models used for semantic segmentation. It is derived from the Sorensen-Dice ratio Coefficients that tests how similar two images are. The value range is [0, 1].
- Tversky Loss [81] deals with a common problem in machine learning and image segmentation that manifests as unbalanced classes in dataset, meaning one class dominates the other.
- Focal Tversky Loss: The cross-entropy (CE) function is designed to limit the inequality between two probability distributions. Several variants of CE have been proposed in the literature, including, for example, focal loss [82] and binary cross-entropy. The first uses a modulation coefficient y > 0 to allow the model to focus on rough patterns rather than correctly classified patterns. The second is an adaptation of CE applied to a binary classification problem (i.e., a problem with only two classes).
- Focal Generalized Dice Loss allows to focus on limited ROI to reduce the weight of ordinary samples. This is done by regulating the modulating factor.
- Log-Cosh Type Loss is a combination of Dice Loss and Log-Cos. Log-Cosh function is commonly applied with the purpose of smoothing the curve in regression applications.
- SSIM Loss [83] is obtained from the Structural similarity (SSIM) index [84], usually adopted to evaluate the quality of an image.
- Cross Entropy: The cross-entropy loss (CE) function provides a measure of the difference between two probability distributions. The aim is to minimize these differences and avoid deviations between small and large areas. This can be problematic when working with unbalanced datasets. Thus, a weighted cross-entropy loss and a better-balanced classification for unbalanced scenarios were introduced [85]. The weighted binary cross entropy formula is given in (14).
- Intersection over Union (IoU) loss is another well-known loss function, which was introduced for the first time in [86].
- Structure Loss is based on the combination of weighted Intersect over Union and weighted binary-crossed entropy. In Table 2, formula (19) refers to structure loss, while formula (20) is a simple variation that want to give more importance to the binary-crossed entropy loss.
- Boundary Enhancement Loss is a loss proposed in [87] which explicitly focus on the boundary areas during training. This loss has very good performances as it does not require neither any pre- or post-processing of the image nor a particular net in order to work. In [60] it is proposed to combine it with Dice Loss and weighted cross-entropy loss.
- Contour-aware Loss was proposed for the first time in [88]. It consists in a weighted binary cross-entropy loss where the weights are obtained with the aim of giving more importance to the borders of the image. In the loss was employed a morphological gradient edge detector. Basically, the difference between the dilated and the eroded label map is evaluated. Then, for smoothing purposes, the Gaussian blur was applied.

In the Table 2, T represents the image of the correct answer, Y is the prediction for the output image, K is the number of classes, M is the number of pixels. T_{km} and Y_{km} are, respectively, the ground truth value and the prediction value for the pixel *m* belonging to the class *k*.

Table 2. Mathematical formalization of the adopted loss functions

Name	Formula		Parameters description
Dice Loss	$L_{GD}(Y,T) = 1 - \frac{2 * \sum_{k=1}^{K} w_k * \sum_{m=1}^{M} Y_{km} * T_{km}}{\sum_{k=1}^{K} w_k * \sum_{m=1}^{M}(Y_{km}^2 + T_{km}^2)}$	(1)	The weight w_k aims to help focus the network on a limited area (so inversely proportional to the frequency of symbols for a given class k).
	$w_k = \frac{1}{\left(\sum_{m=1}^{M} T_{km}\right)^2}$	(2)	
Tversky Index	$TI_k(Y,T) = \frac{\sum_{m=1}^{M} Y_{pm} T_{pm}}{\sum_{m=1}^{M} Y_{pm} T_{pm} + \alpha \sum_{m=1}^{M} Y_{pm} T_{nm} + \beta \sum_{m=1}^{M} Y_{nm} T_{pm}}$	(3)	α and β are two weighting factors used to balance false negative and false positive. n is the negative class and p is the positive class. In the special case, for α=β=0.5, we reduced the Tversky exponent to the equivalent Dice factor.
Tversky Loss	$L_T(Y,T) = \sum_{k=1}^{K} (1 - TI_k(Y,T))$	(4)	We fix $\alpha = 0.3$ and $\beta = 0.7$. We use these values in order to put attention on false negatives.
Focal Tversky Loss	$L_{FT}(Y,T) = L_T(Y,T)^{\frac{1}{\gamma}}$	(5)	we choose $\gamma = 4/3$
Focal Generalized Dice Loss	$L_{FGD}(Y,T) = L_{GD}(Y,T)^{\frac{1}{\gamma}}$	(6)	we choose $\gamma = 4/3$
Log-Cosh Generalized Dice Loss	$L_{lcGD}(Y,T) = \log(\cosh(L_{GD}(Y,T)))$	(7)	
Log-Cosh Focal Tversky Loss	$L_{lcFT}(Y,T) = \log(\cosh(L_{FT}(Y,T)))$	(8)	
SSIM Index	$SSim(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$	(9)	Here, μ_x, μ_y are the local means, σ_x, σ_y, are the standard deviations, and σ_{xy}, is the cross-covariance for images x, y, while C_1, C_2 are regularization constants
SSIM Loss	$L_S(Y,T) = 1 - SSim(Y,T)$	(10)	L_MS (Y,T), it defined as L_S but instead of SSIM we use the Multiscale structural similarity (MS-SSIM) index.
Different Functions Combined Loss	$Comb_1\,(Y,T) = L_{FGD}(Y,T) + L_{FT}(Y,T)$	(11)	
	$Comb_2(Y,T) = L_{lcGD}(Y,T) + L_{FGD}(Y,T) + L_{lcFT}(Y,T)$	(12)	
	$Comb_3\,(Y,T) = L_S(Y,T) + L_{GD}(Y,T)$	(13)	
Weighted Cross Entropy Loss	$L_{WBCE} = -\sum_{k=1}^{K} \sum_{i=1}^{M} w_{ki} * T_{ki} * \log(Y_{ki})$	(14)	w_{ik} is the weight given to the i-th pixel of the image for the class k. These weights were calculated by using an average pooling over the mask with a kernel 31x31 and a stride of 1 in order to consider also nonmaximal activations.
Intersection over Union	$IoU = \frac{\lvert Y \cap T \rvert}{\lvert Y \cup T \rvert}$	(15)	
	$IoU' = \frac{\lvert Y * T \rvert}{\lvert Y + T - Y * T \rvert}$	(16)	

	$L_{IoU} = 1 - IoU'$ (17)	
Weighted Intersect over Union loss	$L_{WIOU} = 1 - \frac{\vert w * Y * T \vert}{\vert w * (Y+T) - w * Y * T \vert}$ (18)	The weights w_{ik} are calculated as aforementioned.
Dice Boundary Enhancement loss	$\mathcal{L}(\mathrm{x},\mathrm{y}) = \frac{\partial^2 S}{\partial x^2} + \frac{\partial^2 S}{\partial y^2}$ (18) $L_{BE} = \lVert \mathcal{L}(T) - \mathcal{L}(Y) \rVert_2 = \left\lVert \frac{\partial^2 (T-Y)}{\partial x^2} + \frac{\partial^2 (T-Y)}{\partial y^2} \right\rVert_2$ (19) $L_{DiceBES} = \lambda_1 L_{Dice} + \lambda_2 L_{BE} + L_{Str}$ (20)	Where $\lVert \cdot \rVert_2$ is the l_2 norm. Best results were achieved by using $\lambda_1 = 1$ and $\lambda_2 = 0.01$
Contour-aware Loss	$M^C = Gauss\left(K \cdot \left(dilate(T) - erode(T)\right)\right) + \mathbb{1}$ (21) $L_C = -\sum_{i=1}^{N} M_i^C * (T_i * \log(Y_i) + (1 - T_i) * \log(1 - Y_i))$ (22) $L_{CS} = L_C + L_{Str}$ (26)	$dilate(T)$ and $erode(T)$ are dilation and erosion operations with a 5 × 5 kernel. K is a hyperparameter for assigning the high value to contour pixels which was set to 5 empirically. $\mathbb{1}$ is the matrix with 1 in every position.

Some works [89]–[91] show that varying the loss function is a good technique for generating diversity among outcomes and creating robust ensembles.

2.3. Data Augmentation

Different methods can be applied to the original data set to increase the amount of data available for training the system. We applied these techniques to the training set on both input samples and masks. We adopt the two data augmentation techniques defined in [60], in particular:

- DA1, base data augmentation consisting in horizontal and vertical flip, 90° rotation.
- DA2, this technique performs a set of operations to the original images in order to derive new ones. These operations comprehend shadowing, color mapping, vertical or horizontal flipping, and others.

3. Performance Evaluation

3.1. Performance Indicators

Since skin segmentation and hand segmentation are binary classification problems, we can evaluate their performance using standard measures for general classification problems [92]. Such as, precision, accuracy, recall, F1 measure, kappa, ROC curve, area under the curve, etc. However, due to the specific nature of this problem, which relies on pixel-level classification and disproportionate distribution, the following metrics are usually considered for performance evaluation: Confusion matrix, F1 measure (Dice), Intersection over Union (IoU), true positive rate (TPR) and false positive (FPR) .

The confusion matrix is obtained by comparing the results to the ground truth and determining the number of true negatives (tn), false negatives (fn), true positives (tp) and false positives (fp) at the pixel level. Precision is the percentage of correctly classified pixels out of all pixels classified as skins, recall measures the model's ability to detect positive samples.

In Table 3, we report the mathematical formalization of the metrics.

Table 3. Performance Indicators

Name	Formula
Precision	$precision = \frac{tp}{(tp + fp)}$
Recall	$recall = \frac{tp}{(fn + tp)}$
F1 Measure/Dice	$F1 = {}^{2tp}/_{(2tp + fn + fp)}$
IoU	$IoU = {}^{tp}/_{(tp + fn + fp)}$
True Positive Rate (TPR)	$TPR = recall$
Fale Positive Rate (FPR)	$FPR = {}^{fp}/_{(fp + tn)}$

We used F1/Dice in this paper for skin segmentation and IoU for hand segmentation, because they are widely used in the related literature.

3.2. Skin Detection Evaluation: Datasets

To assist research in the area of skin detection, there are some well-known color image datasets provided with ground truth. The use of a standard and representative benchmark is essential to execute a fair empirical evaluation of skin detection techniques.

In Table 4 some of the most used datasets are summarized and, in this section, a brief description of each of them is given.

- Compaq [39] is one of the first and most widely used large-scale skin datasets, consisting of images collected from web browsing. It consists in 9731 images containing skin pixels and 8965 images with no skin pixels. Only 4675 skin images have been segmented and included in the ground truth.
- TDSD [93] contains 555 images with highly imprecise annotations produced with automatic labeling.
- Chile [94] contains 103 images with different lighting conditions and complex backgrounds. The ground truth is manually interpreted with moderate accuracy.
- The ECU Skin dataset [95] is a collection of 4,000 color images with a relatively high ground truth annotation. It is particularly challenging because they contain a wide variety of lighting conditions, background scenes, and skin types.

- Schmugge [96] is a collection of 845 images with accurate annotations on the 3 classes skinned/non-skinned/unrelated. Images come from different face datasets (i.e., the UOPB dataset, the AR face dataset, and the University of Chile database).

- Feeval [15] is a dataset based on 8991 frames from 25 low-quality online videos. The annotations are imprecise.

- MCG skin database [97] contains 1000 images selected from the Internet, including blurred backgrounds, various ambient lights, and various human beings. Ground truth have been obtained by hand marking, but it is not accurate, as sometimes eyes, eyebrows and even wrists are marked with skin.

- VMD [98] contains 285 images, it is usually implemented to recognize human activity. The images cover a wide range of lighting levels and conditions.

- SFA dataset [99] contains 1118 manually labeled images (with moderate accuracy).

- Pratheepan [100] contains 78 images randomly downloaded from Google.

- HGR [58] contains 1558 images representing Polish and American Sign Language gestures with controlled and uncontrolled backgrounds.

- SDD [101] contains 21,000 images, some images taken from a video and some others taken from a popular face dataset with different lighting conditions and with different skin colors of people around the world.

- Abdominal Skin Dataset [18] consists of 1400 abdominal images collected using Google image search and then manually segmented. The dataset preserves the diversity of different ethnic groups and avoids the racial bias implicit in segmentation algorithms. To him, 700 images represent dark-skinned people and 700 images represent light-skinned people. Additionally, 400 images represent individuals with high BMI, evenly distributed between light and dark categories. The dataset also took into account other inter-individual variation, such as hair and tattoo coverage, and external variation, such as shadows, when preparing the dataset.

Table 4. Some of the most used datasets per skin detection

Name	Ref	Images	Ground truth	Download	Year
Compaq	[39]	4675	Semi-supervised	ask to the authors	2002

TDSD	[93]	555	Imprecise	http://lbmedia.ece.ucsb.edu/research/skin/skin.htm	2004
UChile	[94]	103	Medium Precision	http://agami.die.uchile.cl/skindiff/	2004
ECU	[95]	4000	Precise	http://www.uow.edu.au/~phung/download.html (currently not available)	2005
Schmugge	[96]	845	Precise (3 classes)	https://www.researchgate.net/publication/257620282_skin_image_Data_set_with_ground_truth	2007
Feeval	[15]	8991	Low quality, imprecise	http://www.feeval.org/Data-sets/Skin_Colors.html	2009
MCG	[97]	1000	Imprecise	http://mcg.ict.ac.cn/result_data_02mcg_skin.html (ask to authors)	2011
Pratheepan	[100]	78	Precise	http://web.fsktm.um.edu.my/~cschan/downloads_skin_dataset.html	2012
VDM	[98]	285	Precise	http://www-vpu.eps.uam.es/publications/SkinDetDM/	2013
SFA	[99]	1118	Medium Precision	http://www1.sel.eesc.usp.br/sfa/	2013
HGR	[44], [58]	1558	Precise	http://sun.aei.polsl.pl/~mkawulok/gestures/	2014
SDD	[101]	21000	Precise	Not available	2015
Abdominal Skin Dataset	[18]	1400	Precise	https://github.com/MRE-Lab-UMD/abd-skin-segmentation	2019

3.3. Hand detection evaluation: Datasets

Similar to the skin detection task, we adopted some well-known color image datasets equipped with ground truth for hand detection. Notice, that we do not want to review the datasets of hand segmentation, but rather choosing two known ones to show strength of the proposed ensemble. In Table 5 two datasets are summarized and, in this section, a brief description of each of them is given.

- EgoYouTubeHands (EYTH) [70] dataset: it comprehends images extracted from YouTube videos. Specifically, authors downloaded three videos with an egocentric point of view and annotated one frame every five frames. The user in the video interacts with other people and performs several activities. The dataset has 1290 frames with hand annotation at the pixel-level, where the environment, number of participants, hand sizes and other factors varies among different images.

- GeorgiaTech Egocentric Activity dataset (GTEA) [102]: the dataset contains images from videos about 4 different subjects performing 7 daily activities. Originally, the dataset was built for activity recognition in the same environment. The original dataset has 663 images with pixel-level hand annotations considering hand till arm. Arms have been removed for a fair training as already done in previous works (e.g., [70]**)**.

It is important to notice that the use of the GTEA dataset is far from homogeneous in the literature and this creates several issues in the comparison of the results among different studies. For instance, some research studies do not remove arms in the training phase. This makes the task a skin segmentation task in which the performance is higher but that should not be compared with results about hand segmentation. We emphasize the importance of a single, standard protocol for these cases, which should be adopted by all those proposing a solution for this problem.

Table 5. Some of the most used datasets per hand detection

Name	Ref	Images	Ground truth	Download	Year
EYTH	[70]	1290	Precise	https://github.com/aurooj/Hand-Segmentation-in-the-Wild	2018
GTEA	[102]	663	Precise	https://cbs.ic.gatech.edu/fpv/	2015

4. Experimental Results

We perform an empirical evaluation to assess the performance of our proposal compared with the state-of-the-art models. We adopted the same methods for both skin and hand segmentation.

The performance of classifiers is affected by the amount of data used for the training phase and ensembles are no exception. In this work, we employed DA1 and DA2 (see section 3.3) on the training set and maintaining the test sets as they are. Notice that for skin segmentation only the first 2000 images of ECU are used as training set, the other images of ECU are one the test sets used for assessing the performance.

HardNet-MSEG is trained with two different optimizers, Stochastic gradient descent (SGD) denoted as H_S and Adam denoted as H_A; the ensemble FH is the fusion of HarDNet-MSEG trained with both the optimizers. PVT and HSN are trained using AdamW optimizer (as suggested in their original papers). The loss function for HarDNet-MSEG, HSN and PVT is the same of the original papers (Structure Loss).

- PVT(2), sum rule between PVT combined with DA1 and PVT combined with DA2;
- HSN(2) is similar to PVT(2), i.e sum rule between one HSN combined with DA1 and one HSN combined with DA2;
- FH(2), sum rule among two H_S (one combined with DA1, the latter with DA2) and two H_A (one combined with DA1, the latter with DA2);
- FH(4) computes FH(2) twice and the output is aggregated using sum rule.
- FH(2)+2×PVT(2), weighted sum rule between PVT(2) and FH(2), the weight of PVT(2) is assigned so that its importance in the ensemble is the same of FH(2) (notice that FH(2) consists of four networks while PVT(2) is built by only two networks).
- FH(4)+4×PVT(2), weighted sum rule between PVT(2) and FH(4), the weight of PVT(2) is assigned so that its importance in the ensemble is the same of FH(4).
- AllM=ELossMix2(10)+(10/4)×FH(2)+(10/2)×PVT(2), weighted sum rule among ElossMix2(10), FH(2) and PVT(2); as in the previous ensemble, the weights are assigned so that each ensemble member have the same importance. ELossMix2(10) is an ensemble, combined by sum rule, of ten stand-alone DeepLabV3+ segmentators with Resnet101 backbone (pretrained as detailed before using VOC); the ten networks are obtained coupling five loss, vix. L_{GD}, $L_{DiceBES}$, Comb1, Comb2, Comb3 (see Table 2 for loss definitions) one time using DA1 and another time using DA2.
- AllM_H=ELossMix2(10)+(10/4)×FH(2)+(10/2)×PVT(2)+(10/2)×HSN(2), similar to the previous one but with the add-on of HSN(2).

4.1. Skin Segmentation

A fair comparison among different approaches is very difficult due to the lack of a universal standard in evaluation: most of published works are tested on self-collected datasets which often are not available for further comparison; in many cases the testing protocol is not clearly explained, many datasets are not of high quality and the precision of the ground truth is questionable since sometimes lips, mouth, rings and bracelets have been labelled as skin. Table 6 reports the performance of the different models.

Table 6. Performance (Dice in the skin detection problem)

	DA	**PRAT**	**MCG**	**UC**	**CMQ**	**SFA**	**HGR**	**SCH**	**VMD**	**ECU**	**VT**	***AVG***
H_S	DA1	0.903	0.880	0.903	0.838	0.947	0.964	0.793	0.744	0.941	0.810	0.872
H_S	DA2	0.911	0.884	0.903	0.844	0.950	0.968	0.776	0.683	0.943	0.835	0.870
H_A	DA1	0.913	0.880	0.900	0.809	0.951	0.967	0.792	0.717	0.945	0.799	0.867
H_A	DA2	0.909	0.886	0.893	0.848	0.951	0.968	0.775	0.707	0.944	0.832	0.871
FH(2)	DA1/DA2	0.920	0.892	0.913	0.859	0.953	0.971	0.793	0.746	0.951	0.839	0.884
FH(4)	DA1/DA2	0.920	0.892	0.916	0.862	0.954	0.971	0.795	0.765	0.951	0.831	0.886
PVT	DA1	0.920	0.888	0.925	0.851	0.951	0.966	0.792	0.709	0.951	0.828	0.878
PVT	DA2	0.923	0.892	0.908	0.863	0.951	0.968	0.776	0.709	0.952	0.848	0.879
PVT(2)	DA1/DA2	0.925	0.892	0.925	0.863	0.952	0.970	0.781	0.719	0.954	0.850	0.883
HSN	DA1	0.927	0.893	0.920	0.851	0.953	0.966	0.777	0.704	0.951	0.800	0.874
HSN	DA2	0.924	0.896	0.889	0.860	0.953	0.969	0.781	0.690	0.953	0.855	0.877
HSN(2)	DA1/DA2	0.928	**0.897**	0.915	0.860	0.955	0.970	0.775	0.671	0.953	**0.860**	0.879
FH(2)+2×PVT(2)	DA1/DA2	0.927	0.894	0.932	0.868	0.954	0.971	0.797	0.767	0.955	0.853	0.893
FH(4)+4×PVT(2)	DA1/DA2	0.926	0.894	0.933	0.869	0.954	0.971	0.798	0.768	0.955	0.847	0.892
ElossMix2(10)	DA1/DA2	0.924	0.893	0.929	0.850	**0.956**	0.970	0.789	0.739	0.952	0.829	0.883
AllM	DA1/DA2	0.929	0.895	0.939	0.868	**0.956**	**0.972**	**0.800**	0.770	0.956	0.846	0.893
AllM_H	DA1/DA2	**0.931**	**0.897**	**0.941**	**0.869**	**0.956**	**0.972**	0.799	**0.773**	**0.957**	0.854	**0.895**

Clearly, combining different topologies boost the performance, the best result is obtained by AllM_H that combines transformers (i.e., PVT and HSN) with HardNet/DeepLabV3+ (i.e., CNN based models).

It is interesting to observe the behavior of ensembles with PVT: PVT with DA1 got higher performance on the UC dataset than its counterpart PVT with DA2; the opposite happens on the CMQ dataset where PVT with DA2 got higher performance than its counterpart PVT with DA1. While the fusion of these two PVTs performs as the best of the two approaches on both situations.

In Table 7 we report a comparison of our methods with some methods proposed in the literature. This is useful to give an idea of how the performance improve along the years. Notice that, we have used only a subset of the datasets used in the previous test for matching the results already proposed in the literature.

Table 7. Comparison with the literature

Method	Year	*Pratheepan*	*MCG*	*UChile*	*Compaq*	*SFA*	*HGR*	*Schmugge*	*VMD*	AVG
Bayes	2002	0.631	0.694	0.661	0.599	0.760	0.871	0.569	0.252	0.630
SA3	*2014*	0.709	0.762	0.625	0.647	0.863	0.877	0.586	0.147	0.652
U-Net	*2015*	0.787	0.779	0.713	0.686	0.848	0.836	0.671	0.332	0.706

SegNet	*2017*	0.730	0.813	0.802	0.737	0.889	0.869	0.708	0.328	0.734
[67]	*2020*	0.812	0.841	0.829	0.773	0.902	0.950	0.714	0.423	0.781
[83]	*2021*	0.926	0.888	0.916	0.842	0.955	0.971	**0.799**	0.764	0.883
AllM_H	*2022*	***0.931***	***0.897***	***0.941***	***0.869***	***0.956***	***0.972***	***0.799***	***0.773***	***0.892***

From the Table 7, we can notice that the big leap in the performance is mainly due to the adoption of deep learning in this domain, while methods from 2002 and 2014 reports similar results.

4.2. Hand Segmentation

In this section, we report the results about the empirical analysis performed for the hand segmentation task. We also provide an ablation study that shows the importance of adopting an ensemble based on DeepLabV3+; this ablation study, for the skin segmentation, have been already reported in [60].

Each ensemble is made up of *N* models (*N*=1 denotes a stand-alone model) which differ only for the randomization in the training process. We employed the standard Dice loss for all the methods. As a standard metric adopted in literature to evaluate the different models, in Table 8 we report the resulting IoU. In particular, we tested the following approaches:

- RN18 a stand-alone DeepLabV3+ segmentators with backbone Resnet18 (pretrained in ImageNet);
- ERN18(N) is an ensemble of N RN18 networks (pretrained in ImageNet);
- RN50 a stand-alone DeepLabV3+ segmentators with backbone Resnet50 (pretrained in ImageNet);
- ERN50(N) is an ensemble of N RN50 networks;
- RN101 a stand-alone DeepLabV3+ segmentators with backbone Resnet101 (pretrained as detailed in before using VOC);
- ERN101(N) is an ensemble of N RN101 networks.

Table 8. Performance (IoU) of the proposed ensembles in the five benchmark datasets, the last column AVG reports the average performance. We report the resulting IoU because this is the standard metric adopted to evaluate the different models.

IoU	EYTH	GTEA
RN18	0.759	0.761
RN50	0.782	0.808
RN101	0.806	**0.841**
ERN18(10)	0.778	0.777
ERN50(10)	0.796	0.812
ERN101(10)	**0.821**	**0.841**

It is possible to notice from the results that the ensembles are performing well but not surprisingly. In this set of experiments, ERN101 is the best model.

In Table 9 the performances of RN101, with different loss functions, are reported and compared with the dice loss as baseline and DA1 as data augmentation method. The following methods are reported (see Table 2 for loss definitions):

- ELoss101(10) is an ensemble, combined by sum rule, of 10 RN101 each coupled with data augmentation DA1 and a given loss function, the final fusion is given by: 2×L_{GD}+ 2×L_T+ 2× Comb1 + 2× Comb2+2×Comb3; where with 2×L_x we mean two different RN101 trained using L_x loss function.
- ELossMix(10) is an ensemble similar to the previous one, but here data augmentation is used to increase diversity: the networks coupled with the loss used in ELoss101(10) (L_{GD}, L_T, Comb1, Comb2, Comb3) are trained one time using DA1 and another time using DA2 (i.e. 5 networks each trained two times, so we have an ensemble of 10 networks);
- ELossMix2(10), it is similar to the previous ensemble, but using $L_{DiceBES}$ instead of L_T.

Table 9. Performance of RN101 with different loss functions

IoU	LOSS	EYTH	GTEA
ERN101(10)	L_{GD}	0.821	0.841
ELoss101(10)	Many loss	0.821	0.849
ELossMix(10)	Many loss	0.819	**0.852**
ELossMix2(10)	Many loss	**0.823**	**0.852**

Table 10. Performance of different models on the two datasets

IoU	DA	EYTH	GTEA
H_S	DA1	0.745	0.757
H_S	DA2	0.760	0.769
H_A	DA1	0.802	0.831
H_A	DA2	0.802	0.826

FH(2)	DA1/DA2	0.810	0.826
FH(4)	DA1/DA2	0.810	0.826
PVT	DA1	0.799	0.819
PVT	DA2	0.814	0.830
PVT(2)	DA1/DA2	0.808	0.837
HSN	DA1	0.818	0.833
HSN	DA2	0.815	0.836
HSN(2)	DA1/DA2	0.812	0.843
FH(2)+2×PVT(2)	DA1/DA2	0.824	0.840
FH(4)+4×PVT(2)	DA1/DA2	0.824	0.840
ELossMix2(10)	DA1/DA2	0.823	**0.852**
AllM	DA1/DA2	0.831	0.847
AllM_H	DA1/DA2	**0.834**	0.848

In Table 10 the previous ensembles are compared with the different models considered in Table 6 for the skin detection problem. It can be noticed from the results that ELossMix2(10) get better results than HardNet, HSN and PVT. The ensemble is the best tradeoff considering both skin and hand segmentation.

We also compare our models with some baselines (see Table 11). In particular, we noticed that:

- Some approaches adopt ad hoc pre-training for hand segmentation, so performance improves, but it becomes difficult to tell whether the improvement is related to model choice or better pre-training;
- Others use additional training images, making performance comparison unfair.

Table 11. Performance comparison with state-of-the-art

	EYTH	**GTEA**
AllM_H	0.834	0.848
[82]	0.688	0.821
[81]	0.897	---
RRU-Net [74]	0.848 / 0.880	---

The proposed ensemble approximates the state of the art, without optimizing the model or performing any domain-specific tuning for hand segmentation. Comparisons among different methods in this case is not easy. As already mentioned before, many methods have higher performance because during the pretraining phase they do not omit other parts of the body (e.g., arms or head) or they add different images during the training phase, making the comparison among performance unfair. For example, [74] reports an IoU of 0.848 without external training data and 0.880 adding examples to the original training data; moreover, in [74] for GTEA dataset also the skin of forearms is considered as foreground. In [76] their method is pretrained using PASCAL person parts (more suited for this specific task); even in [103], for GTEA dataset also the skin of forearms is considered as foreground.

5. Conclusion and Future Research Directions

In this paper, we propose a new framework for evaluating and combining different skin detector approaches, and a comprehensive evaluation of different approaches is performed on several different datasets. We review the latest available approaches, train and test four popular deep learning models for data segmentation on this classification problem, proposing a new ensemble that obtains state of the art performance for skin segmentation.

Empirical evidence provides that CNNs/Transformers work very well for skin segmentation and outperforms all previous methods based on hand-crafted approaches. Furthermore, the proposed ensemble performs very well compared to other previous approaches.

In conclusion, we show that skin detection is a very difficult problem that cannot be solved by individual methods. The performance of many skin detection methods depends on the color space used, the parameters used, the nature of the data, the characteristics of the image, the shape of the distribution, the size of the training sample, the presence of data noise, etc. New methods based on deep learning are less affected by these problems.

The advent of deep learning has led to the rapid development of image segmentation with new models introduced in recent years [76]. These new models require a lot of data with respect to traditional computer vision techniques. Therefore, it is recommended to collect and label large data sets with people from different regions of the world for future research.

Moreover, further research is needed to develop lightweight architectures that can run on resource-constrained hardware without compromising performance.

Author Contributions: "Conceptualization, L.N. and A.L; software, A.L.; writing—review and editing, all the authors. All authors have read and agreed to the published version of the manuscript."

Funding: "This research received no external funding"

Institutional Review Board Statement: "Not applicable"
Informed Consent Statement: "Not applicable."

Data Availability Statement: links are provided in the paper

Acknowledgments: We would like to acknowledge the support that NVIDIA provided us through the GPU Grant Program. We used a donated TitanX GPU to train CNNs used in this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] A. Lumini and L. Nanni, "Fair comparison of skin detection approaches on publicly available datasets," *Expert*

Syst. Appl., vol. 160, p. 113677, Dec. 2020, doi: 10.1016/J.ESWA.2020.113677.

[2] S. S. Han *et al.*, "Augmented Intelligence Dermatology: Deep Neural Networks Empower Medical Professionals in Diagnosing Skin Cancer and Predicting Treatment Options for 134 Skin Disorders," *J. Invest. Dermatol.*, vol. 140, no. 9, pp. 1753–1761, Sep. 2020, doi: 10.1016/J.JID.2020.01.019.

[3] J. R. H. Lee, M. Pavlova, M. Famouri, and A. Wong, "Cancer-Net SCa: tailored deep neural network designs for detection of skin cancer from dermoscopy images," *BMC Med. Imaging*, vol. 22, no. 1, p. 143, 2022, doi: 10.1186/s12880-022-00871-w.

[4] M. Maniraju, R. Adithya, and G. Srilekha, "Recognition of Type of Skin Disease Using CNN," in *2022 First International Conference on Artificial Intelligence Trends and Pattern Recognition (ICAITPR)*, 2022, pp. 1–4, doi: 10.1109/ICAITPR51569.2022.9844199.

[5] M. Zhao, J. Kawahara, K. Abhishek, S. Shamanian, and G. Hamarneh, "Skin3D: Detection and longitudinal tracking of pigmented skin lesions in 3D total-body textured meshes," *Med. Image Anal.*, vol. 77, p. 102329, Apr. 2022, doi: 10.1016/J.MEDIA.2021.102329.

[6] X. Li, C. Desrosiers, and X. Liu, "Deep Neural Forest for Out-of-Distribution Detection of Skin Lesion Images," *IEEE J. Biomed. Heal. Informatics*, p. 1, 2022, doi: 10.1109/JBHI.2022.3171582.

[7] L. M. Pfeifer and M. Valdenegro-Toro, "Automatic Detection and Classification of Tick-borne Skin Lesions using Deep Learning," Nov. 2020, doi: 10.48550/arxiv.2011.11459.

[8] R. L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, doi: 10.1109/34.1000242.

[9] A. A. Argyros and M. I. A. Lourakis, "Real-time tracking of multiple skin-colored objects with a possibly moving camera," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2004, doi: 10.1007/978-3-540-24672-5_29.

[10] K. Roy, A. Mohanty, and R. R. Sahay, "Deep Learning Based Hand Detection in Cluttered Environment Using Skin Segmentation," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017, pp. 640–649, doi: 10.1109/ICCVW.2017.81.

[11] H. Sang, Y. Ma, and J. Huang, "Robust Palmprint Recognition Base on Touch-Less Color Palmprint Images Acquired," *J. Signal Inf. Process.*, vol. 04, no. 02, pp. 134–139, 2013, doi: 10.4236/jsip.2013.42019.

[12] M. De-La-Torre, E. Granger, P. V. W. Radtke, R. Sabourin, and D. O. Gorodnichy, "Partially-supervised learning from facial trajectories for face recognition in video surveillance," *Inf. Fusion*, 2015, doi: 10.1016/j.inffus.2014.05.006.

[13] J.-S. Lee, Y.-M. Kuo, P.-C. Chung, and E.-L. Chen, "Naked image detection based on adaptive and extensible skin color model," *Pattern Recognit.*, vol. 40, pp. 2261–2270, 2007, doi: 10.1016/j.patcog.2006.11.016.

[14] J. Han, G. M. Award, A. Sutherland, and H. Wu, "Automatic skin segmentation for gesture recognition combining region and support vector machine active learning," in *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 237–242, doi: 10.1109/FGR.2006.27.

[15] J. Stöttinger, A. Hanbury, C. Liensberger, and R. Khan, "Skin paths for contextual flagging adult videos," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5876 LNCS, no. PART 2, pp. 303–314, doi: 10.1007/978-3-642-10520-3_28.

[16] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi, "Recent advances in visual and infrared face recognition - A review," *Computer Vision and Image Understanding*, vol. 97, no. 1. pp. 103–135, 2005, doi: 10.1016/j.cviu.2004.04.001.

[17] G. Healey, M. Prasad, and B. Tromberg, "Face recognition in hyperspectral images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1552–1560, 2003, doi: 10.1109/TPAMI.2003.1251148.

[18] A. Topiwala, L. Al-Zogbi, T. Fleiter, and A. Krieger, "Adaptation and Evaluation of Deep Learning Techniques for Skin Segmentation on Novel Abdominal Dataset," in *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2019, pp. 752–759, doi: 10.1109/BIBE.2019.00141.

[19] T. H. Tsai and S. A. Huang, "Refined U-net: A new semantic technique on hand segmentation," *Neurocomputing*, vol. 495, pp. 1–10, Jul. 2022, doi: 10.1016/J.NEUCOM.2022.04.079.

[20] E. Goceri, "Automated Skin Cancer Detection: Where We Are and The Way to The Future," in *2021 44th International Conference on Telecommunications and Signal Processing (TSP)*, 2021, pp. 48–51, doi: 10.1109/TSP52935.2021.9522605.

[21] V. Rawat, D. P. Singh, N. Singh, P. Kumar, and T. Goyal, "A Comparative Study of various Skin Cancer using Deep Learning Techniques," in *2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 2022, pp. 505–511, doi: 10.1109/CISES54857.2022.9844409.

[22] A. Afroz, R. Zia, A. O. Garcia, M. U. Khan, U. Jilani, and K. M. Ahmed, "Skin lesion classification using machine learning approach: A survey," in *2022 Global Conference on Wireless and Optical Technologies (GCWOT)*, 2022, pp. 1–8, doi: 10.1109/GCWOT53057.2022.9772915.

[23] O. T. Jones *et al.*, "Artificial intelligence and machine learning algorithms for early detection of skin cancer in community and primary care settings: a systematic review," *Lancet Digit. Heal.*, vol. 4, no. 6, pp. e466–e476, Jun. 2022, doi: 10.1016/S2589-7500(22)00023-1.

[24] D. Wen *et al.*, "Characteristics of publicly available skin cancer image datasets: a systematic review," *Lancet Digit. Heal.*, vol. 4, no. 1, pp. e64–e74, Jan. 2022, doi: 10.1016/S2589-7500(21)00252-1.

[25] P. Kakumanu, S. Makrogiannis, and N. Bourbakis, "A survey of skin-color modeling and detection methods," *Pattern Recognit.*, vol. 40, no. 3, pp. 1106–1122, 2007, doi: 10.1016/j.patcog.2006.06.010.

[26] B. D. Zarit, B. J. Super, and F. K. H. Quek, "Comparison of five color models in skin pixel classification," *Proc.*

Int. Work. Recognition, Anal. Track. Faces Gestures Real-Time Syst. Conjunction with ICCV'99 (Cat. No.PR00378), pp. 58–63, 1999, doi: 10.1109/RATFG.1999.799224.

[27] N. B. Ibrahim, M. M. Selim, and H. H. Zayed, "A Dynamic Skin Detector Based on Face Skin Tone Color," in *8th International Conference on In Informatics and Systems (INFOS)*, 2012, pp. 1–5.

[28] S. Naji, H. A. Jalab, and S. A. Kareem, "A survey on skin detection in colored images," *Artif. Intell. Rev.*, Nov. 2018, doi: 10.1007/s10462-018-9664-9.

[29] H. Xu, A. Sarkar, and A. L. Abbott, "Color Invariant Skin Segmentation." pp. 2906–2915, 2022.

[30] K. Nazari, S. Mazaheri, and B. S. Bigham, "Creating A New Color Space utilizing PSO and FCM to Perform Skin Detection by using Neural Network and ANFIS," Jun. 2021, doi: 10.48550/arxiv.2106.11563.

[31] W. C. Chen and M. S. Wang, "Region-based and content adaptive skin detection in color images," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 21, no. 5, pp. 831–853, 2007, doi: 10.1142/s0218001407005715.

[32] R. P. K. Poudel, J. J. Zhang, D. Liu, and H. Nait-Charif, "Skin Color Detection Using Region-Based Approach," *Int. J. Image Process.*, vol. 7, no. 4, p. 385, 2013.

[33] H. Kruppa, M. A. Bauer, and B. Schiele, "Skin Patch Detection in Real-World Images," in *Annual Symposium for Pattern Recognition of the DAGM*, 2002, p. 109f, doi: 10.1007/3-540-45783-6.

[34] N. Sebe, I. Cohen, T. S. Huang, and T. Gevers, "Skin detection: a Bayesian network approach," *Proc. 17th Int. Conf. Pattern Recognit.*, vol. 2, pp. 2–5, 2004, doi: 10.1109/ICPR.2004.1334405.

[35] Y. Kim, I. Hwang, and N. I. Cho, "Convolutional neural networks and training strategies for skin detection," in *IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3919–3923, doi: 10.1109/ICIP.2017.8297017.

[36] H. Zuo, H. Fan, E. Blasch, and H. Ling, "Combining Convolutional and Recurrent Neural Networks for Human Skin Detection," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 289–293, 2017.

[37] A. Kumar and S. Malhotra, "Pixel-Based Skin Color Classifier: A Review," *Int. J. Signal Process. Image Process. Pattern Recognit.*, vol. 8, no. 7, pp. 283–290, 2015.

[38] M. R. Mahmoodi and S. M. Sayedi, "A Comprehensive Survey on Human Skin Detection," *Int. J. Image, Graph. Signal Process.*, vol. 8, no. 5, pp. 1–35, 2016, doi: 10.5815/ijigsp.2016.05.01.

[39] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *Int. J. Comput. Vis.*, vol. 46, no. 1, pp. 81–96, 2002, doi: 10.1023/A:1013200319198.

[40] L. Chen, J. Zhou, Z. Liu, W. Chen, and G. Xiong, "A skin detector based on neural network," in *Communications, Circuits and Systems and West Sino Expositions, IEEE 2002 International Conference on*, 2002, vol. 1, pp. 615–619 vol.1, doi: 10.1109/ICCCAS.2002.1180694.

[41] Y. H. Chen, K. T. Hu, and S. J. Ruan, "Statistical skin color detection method without color transformation for

real-time surveillance systems," *Eng. Appl. Artif. Intell.*, vol. 25, no. 7, pp. 1331–1337, 2012, doi: 10.1016/j.engappai.2012.02.019.

[42] M. R. Mahmoodi and S. M. Sayedi, "Leveraging spatial analysis on homogonous regions of color images for skin classification," in *4th International eConference on Computer and Knowledge Engineering (ICCKE)*, 2014, pp. 209–214.

[43] R. Nidhu and M. G. Thomas, "Real Time Segmentation Algorithm for Complex Outdoor Conditions," *Int. J. Sci. Technoledge*, vol. 2, no. 4, p. 71, 2014.

[44] M. Kawulok, J. Kawulok, and J. Nalepa, "Spatial-based skin detection using discriminative skin-presence features," *Pattern Recognit. Lett.*, vol. 41, no. 1, pp. 3–13, 2014, doi: 10.1016/j.patrec.2013.08.028.

[45] A. S. Nunez and M. J. Mendenhall, "Detection of Human Skin in Near Infrared Hyperspectral Imagery," *Int. Geosci. Remote Sens. Symp.*, vol. 2, pp. 621–624, 2008, doi: 10.1109/IGARSS.2008.4779069.

[46] Z. Jiang, M. Yao, and W. Jiang, "Skin Detection Using Color, Texture and Space Information," in *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, 2007, pp. 366–370, doi: 10.1109/FSKD.2007.518.

[47] F. E. Sandnes, L. Neyse, and Y.-P. Huang, "Simple and practical skin detection with static RGB-color lookup tables: A visualization-based study," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 2370–2375.

[48] W. Song, D. Wu, Y. Xi, Y. W. Park, and K. Cho, "Motion-based skin region of interest detection with a real-time connected component labeling algorithm," *Multimed. Tools Appl.*, pp. 1–16, 2016.

[49] S. Jairath, S. Bharadwaj, M. Vatsa, and R. Singh, "Adaptive Skin Color Model to Improve Video Face Detection," in *Machine Intelligence and Signal Processing*, Springer, 2016, pp. 131–142.

[50] A. Gupta and A. Chaudhary, "Robust skin segmentation using color space switching," *Pattern Recognit. Image Anal.*, vol. 26, no. 1, pp. 61–68, 2016.

[51] M. M. Oghaz, M. A. Maarof, A. Zainal, M. F. Rohani, and S. H. Yaghoubyan, "A hybrid Color space for skin detection using genetic algorithm heuristic search and principal component analysis technique," *PLoS One*, vol. 10, no. 8, 2015, doi: 10.1371/journal.pone.0134828.

[52] T. Xu, Z. Zhang, and Y. Wang, "Patch-wise skin segmentation of human body parts via deep neural networks," *J. Electron. Imaging*, vol. 24, no. 4, p. 043009, 2015, doi: 10.1117/1.JEI.24.4.043009.

[53] C. Ma and H. Shih, "Human Skin Segmentation Using Fully Convolutional Neural Networks," in *IEEE 7th Global Conference on Consumer Electronics (GCCE)*, 2018, pp. 168–170, doi: 10.1109/GCCE.2018.8574747.

[54] A. Dourado, F. Guth, T. E. de Campos, and W. Li, "Domain adaptation for holistic skin detection," *CoRR*, vol. abs/1903.0, 2019.

[55] C. Ó. Conaire, N. E. O'Connor, and A. F. Smeaton, "Detector adaptation by maximising agreement between independent data sources," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern*

Recognition, 2007, doi: 10.1109/CVPR.2007.383448.

[56] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, “A skin tone detection algorithm for an adaptive approach to steganography,” *Signal Processing*, vol. 89, no. 12, pp. 2465–2478, 2009, doi: 10.1016/j.sigpro.2009.04.022.

[57] M. Kawulok, “Fast propagation-based skin regions segmentation in color images,” in *10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition.*, 2013, doi: 10.1109/FG.2013.6553733.

[58] M. Kawulok, J. Kawulok, J. Nalepa, and B. Smolka, “Self-adaptive algorithm for segmenting skin regions,” *EURASIP J. Adv. Signal Process.*, no. 1, pp. 1–22, 2014, doi: 10.1186/1687-6180-2014-170.

[59] N. Brancati, G. De Pietro, M. Frucci, and L. Gallo, “Human skin detection through correlation rules between the YCb and YCr subspaces based on dynamic color clustering,” *Comput. Vis. Image Underst.*, vol. 155, pp. 33–42, 2017, doi: 10.1016/j.cviu.2016.12.001.

[60] L. Nanni, A. Lumini, A. Loreggia, A. Formaggio, and D. Cuza, “An Empirical Study on Ensemble of Segmentation Approaches,” *Signals*, vol. 3, no. 2, pp. 341–358, Jun. 2022, doi: 10.3390/signals3020022.

[61] C.-H. Huang, H.-Y. Wu, and Y.-L. Lin, “HarDNet-MSEG: A Simple Encoder-Decoder Polyp Segmentation Neural Network that Achieves over 0.9 Mean Dice and 86 FPS,” Jan. 2021, doi: 10.48550/arxiv.2101.07172.

[62] B. Dong, W. Wang, D.-P. Fan, J. Li, H. Fu, and L. Shao, “Polyp-PVT: Polyp Segmentation with Pyramid Vision Transformers,” Aug. 2021, doi: 10.48550/arxiv.2108.06932.

[63] M. A. Farooq, M. A. M. Azhar, and R. H. Raza, “Automatic Lesion Detection System (ALDS) for Skin Cancer Classification Using SVM and Neural Classifiers,” *Proc. - 2016 IEEE 16th Int. Conf. Bioinforma. Bioeng. BIBE 2016*, pp. 301–308, Dec. 2016, doi: 10.1109/BIBE.2016.53.

[64] X. He, B. Lei, and T. Wang, “SANet:Superpixel Attention Network for Skin Lesion Attributes Detection,” Oct. 2019, doi: 10.48550/arxiv.1910.08995.

[65] M. Arsalan, D. S. Kim, M. Owais, and K. R. Park, “OR-Skip-Net: Outer residual skip network for skin segmentation in non-ideal situations,” *Expert Syst. Appl.*, vol. 141, p. 112922, Mar. 2020, doi: 10.1016/J.ESWA.2019.112922.

[66] K. Minhas *et al.*, “Accurate Pixel-Wise Skin Segmentation Using Shallow Fully Convolutional Neural Network,” *IEEE Access*, vol. 8, pp. 156314–156327, 2020, doi: 10.1109/ACCESS.2020.3019183.

[67] K. Zhang, Y. Wang, W. Li, C. Li, and Z. Lei, “Real-time adaptive skin detection using skin color model updating unit in videos,” *J. Real-Time Image Process.*, vol. 19, no. 2, pp. 303–315, Apr. 2022, doi: 10.1007/S11554-021-01186-9/TABLES/5.

[68] T. Tarasiewicz, J. Nalepa, and M. Kawulok, “Skinny: A Lightweight U-Net for Skin Detection and Segmentation,” *Proc. - Int. Conf. Image Process. ICIP*, vol. 2020-October, pp. 2386–2390, Oct. 2020, doi: 10.1109/ICIP40778.2020.9191209.

[69] Z. Xie, S. Wang, W. Zhao, and Z. Guo, “A robust context attention network for human hand detection,” *Expert*

Syst. Appl., vol. 208, p. 118132, Dec. 2022, doi: 10.1016/J.ESWA.2022.118132.

[70] A. U. Khan and A. Borji, "Analysis of Hand Segmentation in the Wild." pp. 4710–4719, 2018.

[71] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017, doi: 10.1109/TPAMI.2016.2572683.

[72] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017, doi: 10.1109/TPAMI.2016.2644615.

[73] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2699184.

[74] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," Jun. 2017, doi: 10.48550/arxiv.1706.05587.

[75] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, doi: 10.1007/978-3-030-01234-2_49.

[76] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3523–3542, Jul. 2022, doi: 10.1109/TPAMI.2021.3059968.

[77] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." pp. 770–778, 2016.

[78] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif. Intell. Rev.*, 2020, doi: 10.1007/s10462-020-09825-6.

[79] W. Zhang, C. Fu, Y. Zheng, F. Zhang, Y. Zhao, and C. W. Sham, "HSNet: A hybrid semantic network for polyp segmentation," *Comput. Biol. Med.*, vol. 150, p. 106173, Nov. 2022, doi: 10.1016/J.COMPBIOMED.2022.106173.

[80] S. Jadon, "A survey of loss functions for semantic segmentation," *2020 IEEE Conf. Comput. Intell. Bioinforma. Comput. Biol. CIBCB 2020*, Oct. 2020, doi: 10.1109/CIBCB48159.2020.9277638.

[81] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, "Tversky loss function for image segmentation using 3D fully convolutional deep networks," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10541 LNCS, pp. 379–387, 2017, doi: 10.1007/978-3-319-67389-9_44/COVER.

[82] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection." pp. 2980–2988, 2017.

[83] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-Aware Salient Object Detection." pp. 7479–7489, 2019.

[84] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004, doi: 10.1109/TIP.2003.819861.

[85] Y. S. Aurelio, G. M. de Almeida, C. L. de Castro, and A. P. Braga, “Learning from Imbalanced Data Sets with Weighted Cross-Entropy Function,” *Neural Process. Lett.*, vol. 50, no. 2, pp. 1937–1949, Oct. 2019, doi: 10.1007/S11063-018-09977-1/TABLES/9.

[86] M. A. Rahman and Y. Wang, “Optimizing intersection-over-union in deep neural networks for image segmentation,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10072 LNCS, pp. 234–244, 2016, doi: 10.1007/978-3-319-50835-1_22/COVER.

[87] D. Yang, H. Roth, X. Wang, Z. Xu, A. Myronenko, and D. Xu, “Enhancing Foreground Boundaries for Medical Image Segmentation,” May 2020, doi: 10.48550/arxiv.2005.14355.

[88] Z. Chen, H. Zhou, J. Lai, L. Yang, and X. Xie, “Contour-Aware Loss: Boundary-Aware Learning for Salient Object Segmentation,” *IEEE Trans. Image Process.*, vol. 30, pp. 431–443, 2021, doi: 10.1109/TIP.2020.3037536.

[89] L. Nanni, D. Cuza, A. Lumini, A. Loreggia, and S. Brahnam, “Deep ensembles in bioimage segmentation,” *CoRR*, vol. abs/2112.12955, 2021.

[90] L. Nanni, S. Brahnam, M. Paci, and S. Ghidoni, “Comparison of Different Convolutional Neural Network Activation Functions and Methods for Building Ensembles for Small to Midsize Medical Data Sets,” *Sensors 2022, Vol. 22, Page 6129*, vol. 22, no. 16, p. 6129, Aug. 2022, doi: 10.3390/S22166129.

[91] L. Nanni, D. Cuza, A. Lumini, A. Loreggia, and S. Brahman, “Polyp Segmentation with Deep Ensembles and Data Augmentation,” pp. 133–153, 2023, doi: 10.1007/978-3-031-11154-9_7.

[92] D. M. W. Powers, “Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation,” *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011, doi: 10.1.1.214.9232.

[93] Q. Zhu, C.-T. Wu, K. Cheng, and Y. Wu, “An adaptive skin model and its application to objectionable image filtering,” in *Proceedings of the 12th annual ACM international conference on Multimedia*, 2004, p. 56, doi: 10.1145/1027527.1027538.

[94] J. Ruiz-Del-Solar and R. Verschae, “Skin detection using neighborhood information,” in *Proceedings - Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, 2004, pp. 463–468, doi: 10.1109/AFGR.2004.1301576.

[95] S. L. Phung, A. Bouzerdoum, and D. Chai, “Skin segmentation using color pixel classification: Analysis and comparison,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 148–154, 2005, doi: 10.1109/TPAMI.2005.17.

[96] S. J. Schmugge, S. Jayaram, M. C. Shin, and L. V. Tsap, “Objective evaluation of approaches of skin detection using ROC analysis,” *Comput. Vis. Image Underst.*, vol. 108, no. 1–2, pp. 41–51, 2007, doi: 10.1016/j.cviu.2006.10.009.

[97] L. Huang, T. Xia, Y. Zhang, and S. Lin, “Human skin detection in images by MSER analysis,” *18th IEEE Int. Conf.*

Image Process., pp. 1257–1260, 2011, doi: 10.1109/ICIP.2011.6115661.

[98] J. C. Sanmiguel and S. Suja, “Skin detection by dual maximization of detectors agreement for video monitoring,” *Pattern Recognit. Lett.*, vol. 34, no. 16, pp. 2102–2109, 2013, doi: 10.1016/j.patrec.2013.07.016.

[99] J. P. B. Casati, D. R. Moraes, and E. L. L. Rodrigues, “SFA: A human skin image database based on FERET and AR facial images,” in *IX workshop de Visao Computational, Rio de Janeiro*, 2013.

[100] W. R. Tan, C. S. Chan, P. Yogarajah, and J. Condell, “A Fusion Approach for Efficient Human Skin Detection,” *Ind. Informatics, IEEE Trans.*, vol. 8, no. 1, pp. 138–147, 2012, doi: 10.1109/TII.2011.2172451.

[101] M. R. Mahmoodi, S. M. Sayedi, F. Karimi, Z. Fahimi, V. Rezai, and Z. Mannani, “SDD: A skin detection dataset for training and assessment of human skin classifiers,” in *Knowledge-Based Engineering and Innovation (KBEI), 2015 2nd International Conference on*, 2015, pp. 71–77.

[102] Y. Li, Z. Ye, and J. M. Rehg, “Delving Into Egocentric Actions.” pp. 287–295, 2015.

[103] W. Wang, K. Yu, J. Hugonot, P. Fua, and M. Salzmann, “Recurrent U-Net for Resource-Constrained Segmentation.” pp. 2142–2151, 2019.