All-in-one "*HairNet*": A Deep Neural Model for Joint Hair Segmentation and Characterization

Diana Borza
Babeş Boylai University,
Cluj-Napoca, Romania, 400000
dianaborza@cs.ubbcluj.ro

Ehsan Yaghoubi
Instituto de Telecomunicações
University of Beira Interior, 6201-001 Covilhã, Portugal
Ehsan.yaghoubi@ubi.pt

João Neves
TomiWorld
3500-106 Viseu, Portugal
JoaoNeves@tomiworld.com

Hugo Proença
Instituto de Telecomunicações
University of Beira Interior, 6201-001 Covilhã, Portugal
hugomcp@di.ubi.pt

Abstract

The hair appearance is among the most valuable soft biometric traits when performing human recognition at-a-distance. Even in degraded data, the hair's appearance is instinctively used by humans to distinguish between individuals. In this paper we propose a multi-task deep neural model capable of segmenting the hair region, while also inferring the hair color, shape and style, all from in-the-wild *images. Our main contributions are two-fold: 1) the design of an all-in-one neural network, based on depth-wise separable convolutions to extract the features; and 2) the use convolutional feature masking layer as an attention mechanism that enforces the analysis only within the 'hair' regions. In a conceptual perspective, the strength of our model is that the segmentation mask is used by the other tasks to perceive - at feature-map level - only the regions relevant to the attribute characterization task. This paradigm allows the network to analyze features from non-rectangular areas of the input data, which is particularly important, considering the irregularity of hair regions. Our experiments showed that the proposed approach reaches a hair segmentation performance comparable to the state-of-the-art, having as main advantage the fact of performing multiple levels of analysis in a* single-shot *paradigm.*

1. Introduction

Visual surveillance has grown astonishingly in the last decade at a worldwide level: more than 350 billion surveillance cameras were reported in 2016 [6]. Despite popular belief, a reliable, fully-automated visual surveillance system has not been yet developed, and state of the art artificial intelligence based models still struggle with high false-positive rates. Standard face biometric measures cannot be properly analyzed in surveillance systems, due to the poor image quality (low resolution, blurred, off-angle and occluded subjects), and soft biometric cues are often used to assist classical recognition systems. Despite this, research on external face features (hair, head and face shape) has been neglected in favor of other features, such as irises, eyes, mouth, etc.). [33] has shown that hair cues (namely, the hair length and color) are amongst the most discriminative soft biometric labels when dealing with person recognition at a distance. Moreover, neuroscience research studies confirm this hypothesis: the human visual system seems to perceive the face holistically [30], with emphasis on the head structure and hair feature, rather than internal cues [29, 31].

Automated hair analysis is undoubtedly a difficult task, as the hair structure, shape and visual appearance largely vary between individuals, as depicted in Fig. 1. Unlike internal face features (e.g., the eyes or mouth), it is hard to establish the appropriate region of interest for hair pixels. It is difficult to define a hair shape as a variety of hairstyles exist; defining hair texture and color is difficult too. Individuals naturally tend to have different hair colors and styles, but some tend to change their hair color and styles that affects the hair properties.

In this paper, we propose an all-in-one convolutional neural network (CNN) designed for complete hair analysis (segmentation, color, shape and hairstyle classification), which uses only depth wise separable convolutions [9], making it suitable for running on devices with limited

978-1-7281-9186-7/20/$31.00 @2020 European Union

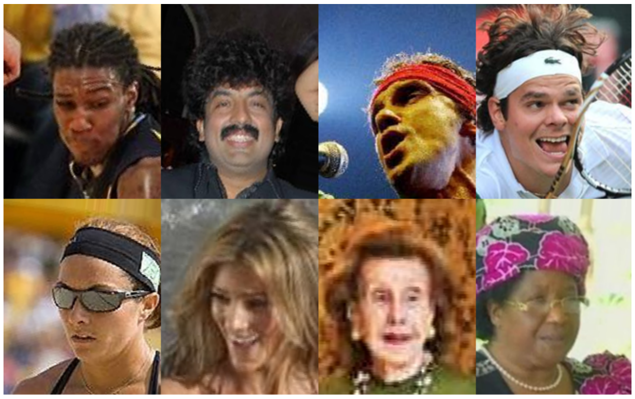

Figure 1. Samples that illustrate the complexity of the *in-the-wild* hair analysis. Subjects have varying poses, with hair of varying shapes, densities and colors, often partially occluded and hard to distinguish from the background.

computational resources. The original architectural features of the network are: (1) the use of *convolutional feature masking* layers in order to keep the convolutions ”focused” only on the hair pixels and (2) *convolutional feature selection* by using skip layers and feature-map masking. The network operates on images captured in uncontrolled, *in-the-wild* environments; the only constraint imposed on the input is that the head area is detectable by a state of the art face detector [22]. A cohesive perspective on the proposed solution is depicted in Fig. 2.

The remainder of this paper is organized as follows: in Section 2, we discuss the related work, and in Section 3 we detail the network architecture and the learning phase of the proposed method. Section 4 describes our experiments and, finally, the conclusions are given in Section 5.

2. Related Work

Early works on hair segmentation operated mainly on frontal images with relatively simple backgrounds. In [36], the positions of the face and eyes are used to establish the region of interest (ROI) for the hair analysis; next, based on spatial (anthropomorphic proportions) and color information a list of seeds is obtained, and region growing is performed to obtain the hair mask. Therefore, hair segmentation is problematic if the background has a similar texture to the hair area. The method also extracts several properties of the hair (volume, length, dominant color) using classical image processing techniques. The method described in [21] defines the hair ROI starting from the positions of the eyes and mouth. Next, the authors devised a region growing algorithm to distinguish the hair pixels from the skin and background pixels. The region growing algorithm operates on a set of 45 features, which includes color, gradient (Canny magnitude), and frequency descriptors. In [27] a raw localization of the hair area is obtained by fusing frequency and color information (YCrCb color space). [14] segments the hair based on the appearance of the upper hair region. First, this region is extracted using active shape and contour models. Based on the appearance parameters of this region, the entire hair region is extracted at pixel level using texture analysis. [17] operates on video sequences; the head area is computed using face detection and background subtraction. Within this region, a skin segmentation mask is obtained using flood-fill algorithm. Finally, the hair region is estimated as the difference between the head and skin pixels. Similarly, [35] extracts the head from video sequences, and the hair region is segmented through histogram analysis and k-means clustering. The hair length is determined through line scanning. [18] uses learned mixture models of color and location information to infer the hypothesis of the hair, face, and background regions. [34] relies on a *coarse hair probability map*, in which each pixel encodes the probability of belonging to the hair class. The hair segmentation map is inferred through regression techniques by finding pairs of isomorphic manifolds. In [28], the authors apply a shape detector to establish a ROI for the hair area; then, to extract the hair-pixels, they use graph-cuts based on solely color cues in the YCbCr color-space. Finally, k-means is applied as a post-processing step to ensure homogeneity between neighboring hair patches.

In [24], a two-layered hierarchical Markov Random Field (MRF) architecture is proposed for the segmentation and labeling of hair and facial hairstyles. The first layer operates at pixel level, modeling local interactions, while the latter extracts higher level, object information, providing coherent solutions for the segmentation problem. The method was tested on degraded images captured by an outdoor visual surveillance system.

Recently, the problem of hair segmentation was approached from a deep learning perspective ([19, 1, 13]); these methods achieve state of the art performance. The segmentation neural networks begin with ”contracting” path, in which a sequence of convolutional layers extract meaningful features, but also reduce the spatial information. Next, a set of deconvolutional layers expands these condensed features into segmentation maps. To preserve high-resolution details, skip-connections are inserted to concatenate feature maps from the beginning of the network (higher level of details) with those from the expanding part of the network (higher semantic level). In [19], the loss function is tuned to preserve the high-frequency information of the hair by adding a term that penalizes the discrepancy between the gradients of the input image and those of the predicted hair mask.

2.1. Multi-task Convolutional Neural Networks

Multi-task learning (MTL) has been successfully used in machine learning as a strategy to improve generalization

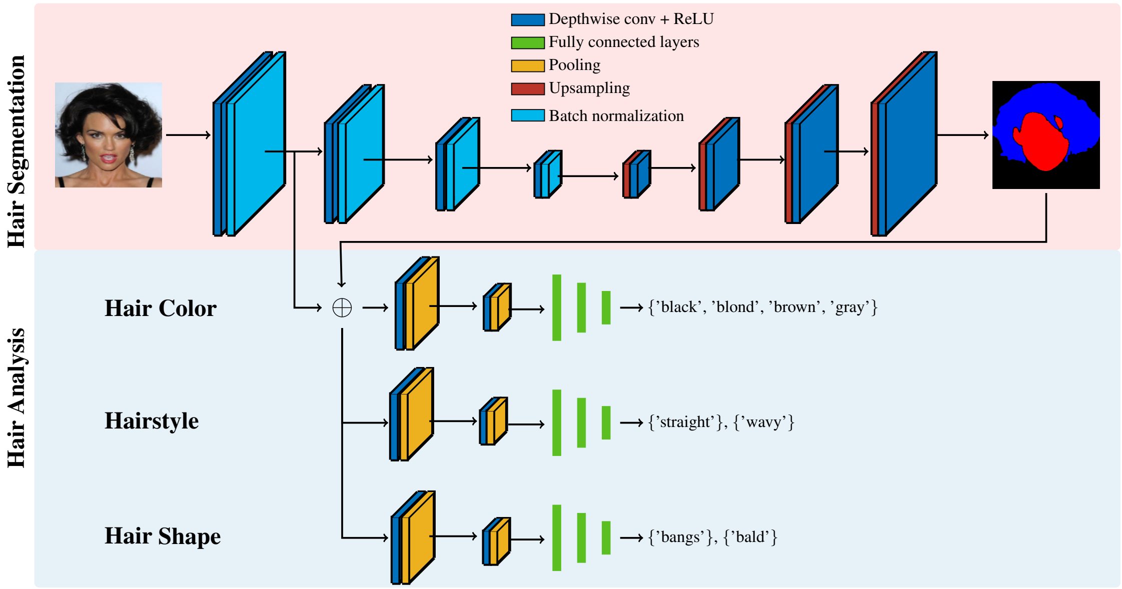

Figure 2. Solution outline: the network comprises several classification branches for the following hair attributes: hair-skin segmentation mask, hair color, shape and style. The segmentation output is used by the other classification branches to select, at feature map level, only the hair pixels via convolutional feature masking.

by learning several classification tasks at once while maintaining a shared representation of the data. A detailed description, including theoretical analysis and applications of multi-task learning, can be found in [2]. One of the pioneering works to performed multi-task facial attribute analysis using a single CNN in an end-to-end manner is [26]. The network simultaneously performs face detection and alignment, pose estimation, gender recognition, smile detection, age estimation, and face recognition. In this framework, the filters in the first convolutional layers of the network are shared between all the classification tasks, constraining a shared representation among the tasks, and reducing the risk of overfitting in these layers. Deep multi-task learning has also been applied for emotion analysis. In [3], the authors propose a deep learning framework for the tasks of facial attribute recognition, action unit detection, and valence-arousal estimation.

2.2. Feature Selection in Convolutional Neural Networks

Deep neural networks achieved state of the art performance on (almost) every field of computer vision and are often used as generic feature extractors. This adaptability of CNNs is also proved by transfer learning: features learned by the network on a (large) database can be successfully applied to other classification tasks. However, classical CNN architectures operate holistically, in the sense that the features are extracted globally, from the entire image, and thus capturing (potentially) irrelevant information. Therefore: How could the network be *guided* to *see* and extract features within some predefined ROI? This question arose for the problem of object detection, in which a bounding box and a class must be inferred for every object in an image. Clearly, the classification part should only analyze the region of interest of the localized object.

The R-CNN (Regions with CNN features) architecture solves this problem in a straightforward manner: a region proposal extraction step is first applied to extract potential objects, and each of these regions is fed to the CNN. Its successor, Fast R-CNN object detector [7], analyzes the entire image to extract a convolutional feature map. Next, each ROI is mapped to this feature map, warped into square regions of predefined size (ROI pooling), and fed to the object classification layer.

Similarly, the SPP-Net architecture [8] introduced the Spatial Pyramid Pooling layer (SPP layer), which masks convolutional feature maps by a rectangular region (i.e., zeros-out the features outside the ROI) and extracts a fixed-length feature vector out of each ROI. In [4], bounding boxes, which can be seen as coarse segmentation masks, are used to ”supervise” the training of CNNs for semantic image segmentation. A step forward is taken by [5]; here, input masks of irregular shapes are used to eliminate irrelevant regions of the feature map. The input binary masks are projected into the domain of the convolutional feature maps: each activation is mapped to the input image domain as the center of its receptive field (similar to [8]), and each pixel of the input mask is assigned to the nearest projected receptive field. However, this approach requires an additional step to generate the input masks with the region proposals. In

[5], the proposal regions are extracted by grouping several super-pixels of the input image. In our approach, the masks are segmented directly by the neural network, therefore no pre-processing steps are required.

3. Proposed Method

3.1. Network Architecture

Formally, let X_i denote the feature vector (RGB-pixels) of the i^{th} sample, Y_i the corresponding annotations, and $\hat{Y}_i$ the network's prediction. The data associated to each image (Y_i or $\hat{Y}_i$) comprises the following attributes: $\{M_i, cl_i, st_i, wv_i, bg_i, bd_i\}$, where M_i is the face/hair segmentation mask, cl_i $\in$CL = {'black', 'blond', 'brown', 'gray'} denotes the hair color label, st_i and wv_i are binary values which indicate if the *hairstyle* is 'straight' or 'wavy' respectively. Finally, the values bg_i, bd_i compose the *hair shape* classification branch, indicating whether the person has bangs, or is bald respectively. The output of the network was chosen in accordance with the hair attribute information provided by CelebA database [23], which is, to the best of our knowledge, the largest image dataset providing multiple hair attributes annotations. We chose separate, binary attributes to describe the *hairstyle* (st_i and wv_i) and the *hair shape* (bg_i, bd_i), instead of a single multi-label classification layer, for two main reasons. First of all, not all the samples from the dataset are annotated with this information, or, on the other hand, some samples are annotated with multiple labels from the same logical group. The latter case results in a contradiction with the multi-label classification, which assumes that each example is appointed to *one and only one* label. Secondly, the annotations provided for these attributes are not exhaustive: for example, the *hair shape* analysis could also include one of the following: "long hair", "medium hair", "short hair", etc.

The backbone of the network is inspired by the lightweight MobileNet [9], on top of which we added several classification branches.

3.2. Hair Segmentation

The output of the hair segmentation branch $\hat{M}_i \in \mathbb{R}^{224\times224\times2}$ is a bi-dimensional, two-channel, mask of the same size as the input. The two channels (M_i^0, M_i^1) contain, for each pixel, the probability for belonging to the skin or hair class, respectively. The facial skin area is also segmented, as it provides essential information regarding the hair shape and length: one cannot make any inference about the shape of the hair without correlating its area to the face.

To obtain the segmentation mask, the feature map of the last convolutional layer in the network backbone is fed to a decoder. As suggested in [19], rather than using transposed convolutional layers, the upsampling is accomplished by a 2× upsampling operation, followed by depth-wise and point-wise convolutions. Three such blocks are concatenated to obtain a mask of the same size as the input image. Similar to [19], skip connections to the corresponding, equal-sized layers in the network backbone are added such that the output includes information about the high resolution, but yet weak, features extracted by these layers.

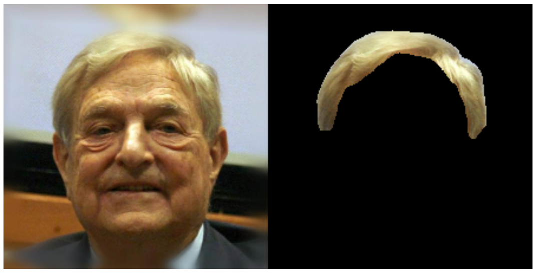

Figure 3. Hair color perception is a contextual phenomenon and cannot be decoupled from the surrounding scene colors and light sources. Also, demographic attributes can influence the hair color estimation process.

Finally, the segmentation output is obtained by adding a 1×1 convolution with two filters (i.e., two output channels: one for the hair and one for the skin pixels) with *softmax* activation.

During training, we aim at minimizing the binary cross entropy loss (1) between the ground truth mask M_i and the predicted segmentation mask $\hat{M}_i$:

$$L_{seg}(M_i, \hat{M}_i) = -(M_i \cdot log(\hat{M}_i) + (1 - M_i) \cdot log(1 - \hat{M}_i)). \tag{1}$$

At test time, the single channel, output mask is obtained by assigning each pixel to the class (hair or skin) with the highest probability, given that it is larger than a threshold t, or to background otherwise:

$$M_{out} = \begin{cases} \text{arg max}(M^0, M^1) + 1, & \text{if} \max(M^0, M^1) > t \\ 0 \text{ (background)}, & \text{otherwise} \end{cases}, \tag{2}$$

where $t = 0.5$ was used in all our experiments.

3.3. Hair Color Inference

Color perception is a complex process, as the appearance of an object is highly dependent on the environmental context (both spatially and temporally) [12]. It is practically impossible to distinguish the apparent color of a patch, without having additional information regarding the surrounding colors and light sources. In the context of hair color estimation, demographic cues (such as gender and age) are also crucial in deciding the hair tone. An illustrative example is depicted in Figure 3.

Therefore, when deciding on the color tone, the network should use, not only information about the hair tone but also

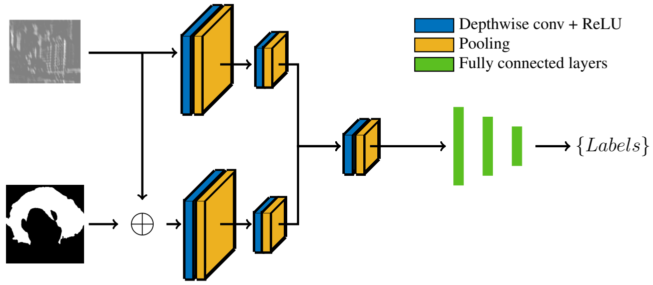

Figure 4. Hair color analysis module. Two separate convolutional branches analyze the image's feature map: the first captures information about the global scene lighting, while the second one focuses only on the hair region using convolutional feature masking.

some cues regarding the surrounding lighting conditions and light sources. With this in mind, the hair color classification task combines two convolutional branches (Fig. 4), which operate on the feature map extracted by the network backbone. The first analyzes the entire feature map, thus extracting information about the overall scene lighting conditions, while the latter masks this feature map using the output of the hair segmentation, to put emphasis solely on the hair features.

Finally, they are merged into a single feature vector FVC, which is flattened and passed to a fully-connected layer with *softmax* activation:

$$sm(FVC_i) = \frac{e^{FVC_i}}{\sum_{j=1}^{K} e^{FVC_j}}, \tag{3}$$

where FCV_i is the feature vector of the *i*-$^{\text{th}}$ sample.

As mentioned above, the hair color analysis module distinguishes the hair tone into one of the following classes CL = {'black', 'blond', 'brown', 'gray'}.

The loss function to be optimized in this case is the categorical cross-entropy loss:

$$L_{color} = \sum_{i} \sum_{j=1}^{|CL|} -cl_{ij} \cdot log(\widehat{cl_{ij}}), \tag{4}$$

where CL is the number of hair labels, cl_i is the one-hot encoding of the ground truth hair color, and $\hat{cl_i}$ are the predicted class probabilities.

3.4. Hairstyle Inference

The hairstyle analysis module comprises two separate binary classification layers, specialized for the 'wavy' or 'straight' structures respectively.

To decide on these tasks, the network should only analyze the hair pixels. Therefore, the input of each classification branch consists of a feature map extracted from the network backbone, masked with the hair segmentation mask, such that only the deemed hair regions are considered. Let FM be the feature map extracted from the network backbone and HS the binarized hair segmentation map. The input I of each of these branches is given by:

$$I = FM \ominus HS, \tag{5}$$

where Θ is the feature map masking operator as defined in Section 2.2. This input is passed to 2 convolutional layers, flattened and then fed to a fully convolutional classification layers. As we are dealing with binary attributes, the activation function for the output neurons O_b is the *sigmoid* function:

$$P(O_b) = \frac{1}{1 + e^{-O_b}}. \tag{6}$$

The loss function of these layers is the binary cross-entropy loss function:

$$L_a(a, \hat{a}) = -\frac{1}{N} \sum (a \cdot log(\hat{a}) + (1 - \hat{a}) \cdot log(1 - \hat{a})), \tag{7}$$

where $a = 1$ if the hair has the attribute and $a = 0$ otherwise; $\hat{a}$ is the predicted probability for the hair attribute.

3.5. Hair Shape Inference

The hair shape analysis task consists of two classification branches, each having a binary outcome: *Bangs* and *Bald*.

Intuitively, a piece of essential information in inferring these shape characteristics is the relationship between the face area and the hair area. Therefore, when applying the convolutional feature masking operation, we keep the hair pixels, as well as the facial skin pixels to better capture this relationship.

As the predictions are binary values, the activation and loss functions for the hair shape classification layers are identical to the ones used for hairstyle classification (Section 3.4).

4. Experiments and Discussion

4.1. Datasets and Experimental Setup

The main dataset used to train and validate the proposed model was CelebAMask-HQ [15], a subset of CelebA database [23]. CelebA [23] is suitable for training our model as it contains more than 200k images captured in real-world scenarios (blurred, occluded subjects and with large pose variations); in addition, each image is labeled with 40 binary attributes, including information about the hair color attributes {'black', 'blond', 'brown', 'gray'}, hairstyle attributes {'straight', 'wavy'} and shape attributes {'bangs', 'bald'}.

For the segmentation task, we used CelebAMask-HQ [15] which contains 30k images, selected from CelebA, together with manually annotated masks of face components (skin, nose, eyes, eyebrows, ears, mouth, lip, hair, hat) and other accessories (eyeglass, earring, necklace, neck, and cloth).

In addition, to demonstrate the generalization ability of the proposed method, we also tested the segmentation module on three additional databases: (a) Labeled Parts in the Wild [11], (b) Figaro-1k [32] and (c) another subset of CelebA, independently annotated by [1]. Images from these datasets were not used at all in the training part. Labeled Parts in the Wild [10] (the *funnelled* version) is a subset of Label Faces in the Wild (LFW) [11] database; it contains 2927 face images segmented into hair/skin/background labels. The segmentation is performed at a coarse level: first the images are divided into super-pixels, and then each super-pixel is manually assigned to a label. Figaro-1k [32] contains 1050 images annotated with hair masks, gathered from the Internet, for the purpose of hair analysis in the wild.

4.2. Learning and Parameter Tuning

As annotated data (with hair masks and hair attributes) is limited, we used transfer learning to make sure that the network won't overfit the training data. So, instead of randomly initializing the weights of the neural network, the training starts from some weight values computed on a different task, for which larger datasets are available; this assumes that the low-level features extracted (edges, textures, gradients, etc.) are relevant across tasks. Therefore, the backbone of the network and the segmentation branch is first trained to segment objects from the COCO dataset [20]. COCO is a large scale image database, which comprises approximately 330K images, designed for object detection and segmentation. The dataset comprises more than 1.5 million object instances, captured in real-world scenarios, grouped into 80 object categories, thus providing enough generalization and data variance.

Next, we conduct the following training scheme:

1. Train the hair segmentation branch on CelebAMask-HQ dataset using the loss function described in L_{seg} (1). The segmentation branch is first trained, as the attribute classification problems use the segmentation mask to establish the ROIs (in the convolutional feature masking layers). Having a good estimate of the hair and face region would greatly speed-up the training process.

2. Freeze the shared layers of the network backbone and individually train all the hair analysis branches using their corresponding loss functions.

3. Finally, the neural network is trained on all the tasks, in an end-to-end manner, such that the common knowledge (filter values) is shared across all the classification problems. At this stage, the individual loss functions are combined into a weighted average as described in equation (8):

$$L = \sum_{i=0}^{T} \lambda_i \cdot L_i, \tag{8}$$

where T is the total number of tasks, $L_i \in \{L_{color}, L_{seg}, L_{straight}, L_{wavy}\}$ and λ_i are the loss value and weight for task i.

In all cases, the weights are optimized using Adam [16] optimizer. The initial learning rate α is set to $\alpha = 0.0001$ when training individually the classification branches, and decreased to $\alpha = 0.00001$ for the final, end-to-end training; in all cases, the exponential decay rate for the first moment estimates β_1 is set to 0.9, and the exponential decay rate for the second-moment estimates β_2 is fixed at 0.99.

4.3. Results

4.3.1 Hair Segmentation

Let n_{cl} be the number of segmentation classes ($n_{cl} = 2$ in our case), n_{ij} be the number of pixels belonging to class i but predicted to class j, and t_i the number of pixels in the ground truth annotation belonging to class i. For the numerical evaluation of the proposed method, we report the mean Intersection over Union ($mIOU$) and the mean pixel accuracy ($mAcc$), as defined in equations (9) and (10).

$$mIoU = \frac{1}{n_{cl}} \sum_i \frac{n_{ii}}{t_i + \sum_j n_{ij} - n_{ii}}. \tag{9}$$

The $mAcc$ metric defines the percentage of correctly classified pixels of a class, averaged over all the segmentation classes.

$$mAcc = \frac{1}{n_{cl}} \frac{\sum_i n_{ii}}{\sum_i t_i}. \tag{10}$$

A fraction of 3000 images (10%) of the CelebAHQ-Mask dataset, which were not used in the training process, are used to validate the proposed approach. The results of the proposed method compared to other state of the art works are reported in Table 1. The results are discussed in Section 4.3.1, with some of the predictions of the proposed method depicted in Fig. 5.

Baseline methods Table 1 displays the hair segmentation performance on CelebAHQ-Mask, Labeled Parts and Figaro-1k databases, compared to other state-of-the-art methods based on deep learning frameworks. In Table 1, CelebA* refers to the subset of CelebA dataset annotated by [1].

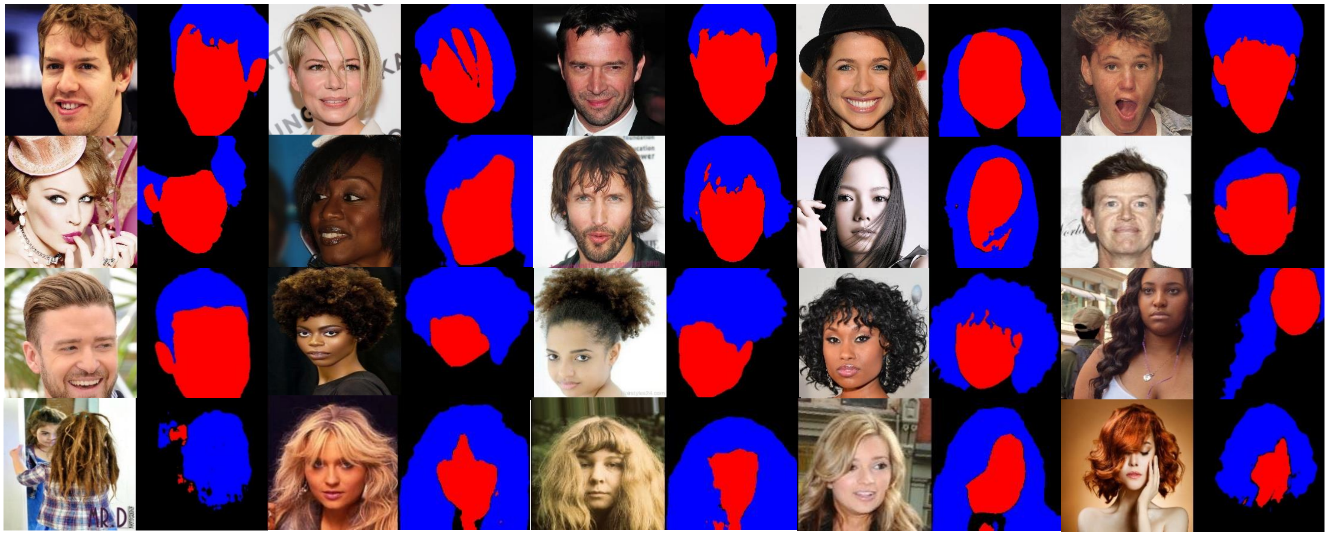

Figure 5. Segmentation masks obtained by the proposed solution on different datasets. The predicted hair pixels are depicted in blue, skin pixels appear in red and background pixels in black. Last row: some failure cases.

Table 1. Comparison of hair segmentation performance with respect to the state-of-the-art.

Method	Database	Pixel accuracy	IoU
[19]	LFW	**97.69**	NA
[1]	LFW	97.01	0.871
[25]	LFW	97.32	NA
Proposed	LFW	95.30	0.864
[1]	CelebA*	97.06	0.920
Proposed	CelebA*	**97.55**	0.881
Proposed	CelebA-MaskHQ	**98.79**	0.939
[1]	Figaro-1k	90.28	0.778
Proposed	Figaro-1k	**97.61**	0.903

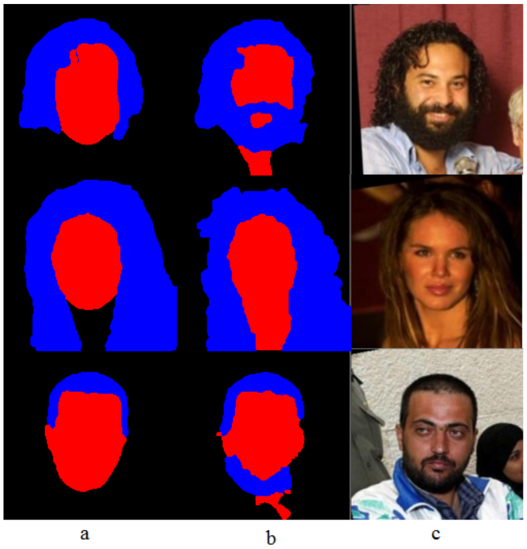

Figure 6. Examples of predicted segmentation masks (LFW dataset): **a)** predicted; **b)** ground truth; **c)** input image.

Overall, the proposed method achieves high performance for the task of hair segmentation, even if it is surpassed by the other methods on the LFW dataset. In our view, this was due to the fact that most of these methods are intended for various fashion, visagisme or hair coloring applications, in which the hair shape needs to be accurately captured by the segmentation mask. [19] uses a secondary loss function besides binary cross-entropy to obtain accurate segmentation masks from coarse annotation data. This loss function enforces the consistency between the input image and the predicted mask edges. In [1] a more complex (VGG-16) fully convolutional neural networks, while [25] combines fully convolutional neural networks with conditional random fields to obtain an accurate hair matting result. Also, the lower performance in LFW might be due to the proposed method hasn't been trained on the LFW parts dataset and the segmentation masks provided by this database are quite different from the ones of CelebA-MaskHQ. First of all, they are provided at super-pixel level, so are not accurate enough for high-accuracy evaluation. In addition, as opposed to CelebA-MaskHQ, the hair class also includes facial hair (moustache and beard), while the skin class comprises the neck area. Fig. 6 displays some ground truth segmentation masks versus predicted masks on the LFW dataset.

The proposed method is not intended for virtual try-on applications, where highly accurate hair segmentation masks are required, but for soft biometrics analysis in visual surveillance systems. Therefore, we are not interested in perfectly segmenting all the hair strands or contour details. Moreover, as discussed in the introductory section,

Table 2. Hair attributes classification performance of the proposed method

Metric	**Feature masking**	**Hair color**	**Hairstyle**		**Hair shape**	
			'wavy'	**'straight'**	**'bangs'**	**'bald'**
Accuracy	✗	88.16	93.20	92.10	92.71	**98.40**
Precision	✗	88.13	94.26	92.87	96.32	97.45
Recall	✗	88.16	92.00	91.2	88.82	99.40
F1 Score	✗	88.01	93.11	92.02	92.41	98.41
Accuracy	✓	**93.45**	**94.30**	**94.60**	**94.41**	98.10
Precision	✓	93.50	95.48	95.88	97.23	97.23
Recall	✓	93.45	93.00	93.20	91.41	99.00
F1 Score	✓	93.43	94.22	94.52	94.23	98.12

images captured by security cameras are often low resolution and blurred, and these hair details would be impossible to distinguish. Even so, from Figure 5 it can be observed that the proposed network is capable of capturing the overall hair shape by accurately segmenting larger strands of hair covering the face or bangs.

4.3.2 Hair Attributes Inference

To evaluate the classification branches, we randomly selected test images from the CelebA dataset (which are not a part of CelebA-MaskHQ) such that the number of samples in each class is the same. The standard metrics: acc - accuracy, pr - precision, rec -recall and F_1 - F1 score are used to numerically express the performance of the proposed solution. Table 2 summarizes the performance of our network in hair attribute characterization, with and without using convolutional feature masking (to prove the efficiency of the proposed convolutional feature masking layer). In the latter case, the network was trained as described in Section 4.2, but the input masks of the hair segmentation module are set to 1, such that the entire image is analyzed for classifying the hair shape.

For each hairstyle and shape classes we randomly selected 1,000 images from the CelebA dataset that are not part of CelebA-HQ. Our experiments showed that, except for the bald detection task, the convolutional feature masking resulted in an increase of the classification performance. For the bald attribute, the accuracy values between the masked and unmasked implementations are comparable (a difference of only 0.3%).

The hair color analysis branch was evaluated on 6,000 images (1,500 samples for each color class) randomly selected from the CelebA dataset. The proposed method uses *softmax* as a final classification layer for predicting the hair color, and considers the class with the highest probability as the hair color prediction. However, some images from the CelebA dataset are not labeled with any of the hair color attributes, or, on the other hand, are labeled with multiple colors (e.g., 'blond' and 'gray'). To be fair in the comparison, both for training and for testing, we randomly selected solely images that contain one and only one annotation of the hair color classes. Overall, the majority of confusions are between the 'brown'/'blonde', 'brown'/'gray' and 'blonde'/'gray' labels. In our view, this was mostly due to the subjective perception of hair color, with light brown/dark-blonde colors being easily mistaken with blond/light blonde color when performing the manual annotation of ground truth data.

The inference step (hair segmentation and hair attribute classification), takes, on average 350 milliseconds on an third generation iPad Pro device.

5. Conclusions

This paper described an *all-in-one* model for hair segmentation and attribute analysis, able to jointly extract the hair-facial skin segmentation mask while also inferring information about the hair color, shape and style. Also, as the proposed architecture uses only depth-wise separable convolutions, it is straightforward to running it in real time, even on devices with limited computational power (e.g., smartphones). To limit the influence of background and irrelevant features on the prediction of the network, an attention mechanism based on convolutional feature masking layers is proposed. Therefore, in our architecture, the inferred segmentation masks are used by the classification branches to determine, at the feature map level, any irregular shaped patches that might correspond to the hair pixels, which enables it to ignore the remaining regions that are deemed as irrelevant to the analysis problem. This feature masking strategy is preferred over traditional ROI-Pooling layers, as if we try to enclose the hair area into a rectangle, a large portion of that patch will be "filled" by the face area, which introduces irrelevant (but salient) features to the analysis problem.

Our experiments were performed in challenging *in the wild* datasets (CelebA, LFW and Fiagro-1k), obtaining high performance (similar or higher than the state of the art), at a lower computational cost.

References

[1] D. Borza, T. Ileni, and A. Darabant. A deep learning approach to hair segmentation and color extraction from facial images. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 438–449. Springer, 2018.

[2] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, Jul 1997.

[3] W.-Y. Chang, S.-H. Hsu, and J.-H. Chien. Fatauva-net: An integrated deep learning framework for facial attribute recognition, action unit detection, and valence-arousal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 17–25, 2017.

[4] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015.

[5] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3992–4000, 2015.

[6] S. Feldstein. The global expansion of ai surveillance. *Carnegie Endowment. https://carnegieendowment.org/2019/09/17/global-expansion-of-ai-surveillance-pub-79847*, 2019.

[7] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[9] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[10] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[11] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[12] A. Hurlbert and Y. Ling. Understanding colour perception and preference. In *Colour Design*, pages 169–192. Elsevier, 2017.

[13] T. Ileni, D. Borza, and A. Darabant. Fast in-the-wild hair segmentation and color classification. In *14th International Conference on Computer Vision Theory and Applications*, pages 59–66, 2019.

[14] P. Julian, C. Dehais, F. Lauze, V. Charvillat, A. Bartoli, and A. Choukroun. Automatic hair detection in the wild. In *2010 20th International Conference on Pattern Recognition*, pages 4617–4620. IEEE, 2010.

[15] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] A. Krupka, J. Prinosil, K. Riha, J. Minar, and M. Dutta. Hair segmentation for color estimation in surveillance systems. In *Proc. 6th Int. Conf. Adv. Multimedia*, pages 102–107, 2014.

[18] K.-c. Lee, D. Anguelov, B. Sumengen, and S. B. Gokturk. Markov random field models for hair and face segmentation. In *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*, pages 1–6. IEEE, 2008.

[19] A. Levinshtein, C. Chang, E. Phung, I. Kezele, W. Guo, and P. Aarabi. Real-time deep hair matting on mobile devices. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 1–7. IEEE, 2018.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[21] U. Lipowezky, O. Mamo, and A. Cohen. Using integrated color and texture features for automatic hair detection. In *2008 IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, pages 051–055. IEEE, 2008.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[23] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[24] H. Proenca and J. C. Neves. Soft biometrics: Globally coherent solutions for hair segmentation and style recognition based on hierarchical mrfs. *IEEE Transactions on Information Forensics and Security*, 12(7):1637–1645, 2017.

[25] S. Qin, S. Kim, and R. Manduchi. Automatic skin and hair masking using fully convolutional networks. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 103–108. IEEE, 2017.

[26] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa. An all-in-one convolutional neural network for face analysis. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 17–24. IEEE, 2017.

[27] C. Rousset and P.-Y. Coulon. Frequential and color analysis for hair mask segmentation. In *2008 15th IEEE International Conference on Image Processing*, pages 2276–2279. IEEE, 2008.

[28] Y. Shen, Z. Peng, and Y. Zhang. Image based hair segmentation algorithm for the application of automatic facial caricature synthesis. *The Scientific World Journal*, 2014, 2014.

[29] P. Sinha. Last but not least. *Perception*, 29(8):1005–1008, 2000.

[30] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11):1948–1962, 2006.

[31] P. Sinha and T. Poggio. ’united’we stand. *Perception*, 31(1):133, 2002.

[32] M. Svanera, U. R. Muhammad, R. Leonardi, and S. Benini. Figaro, hair detection and segmentation in the wild. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 933–937. IEEE, 2016.

[33] P. Tome, J. Fierrez, R. Vera-Rodriguez, and M. S. Nixon. Soft biometrics and their application in person recognition at a distance. *IEEE Transactions on information forensics and security*, 9(3):464–475, 2014.

[34] D. Wang, S. Shan, H. Zhang, W. Zeng, and X. Chen. Isomorphic manifold inference for hair segmentation. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–6. IEEE, 2013.

[35] Y. Wang, Z. Zhou, E. K. Teoh, and B. Su. Human hair segmentation and length detection for human appearance model. In *2014 22nd International Conference on Pattern Recognition*, pages 450–454. IEEE, 2014.

[36] Y. Yacoob and L. S. Davis. Detection and analysis of hair. *IEEE transactions on pattern analysis and machine intelligence*, 28(7):1164–1169, 2006.