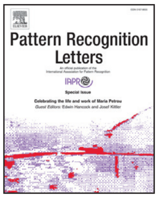
Pattern Recognition Letters 68 (2015) 250–259

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Extracting human attributes using a convolutional neural network approach[☆]

Hugo Alberto Perlin[a,*], Heitor Silvério Lopes[b]

[a] Paraná Federal Institute – Paraná, Paranaguá (PR), Brazil
[b] Federal University of Technology – Paraná, Curitiba (PR), Brazil

ARTICLE INFO

Article history:
Available online 26 July 2015

Keywords:
Computer vision
Machine learning
Soft-biometrics
Convolutional Neural Network
Gender recognition
Clothes parsing

ABSTRACT

Extracting high level information from digital images and videos is a hard problem frequently faced by the computer vision and machine learning communities. Modern surveillance systems can monitor people, cars or objects by using computer vision methods. The objective of this work is to propose a method for identifying soft-biometrics, in the form of clothing and gender, from images containing people, as a previous step for further identifying people themselves. We propose a solution to this classification problem using a Convolutional Neural Network, working as an all-in-one feature extractor and classifier. This method allows the development of a high-level end-to-end clothing/gender classifier. Experiments were done comparing the CNN with hand-designed classifiers. Also, two different operating modes of CNN are proposed and compared each other. The results obtained were very promising, showing that is possible to extract soft-biometrics attributes using an end-to-end CNN classifier. The proposed method achieved a good generalization capability, classifying the three different attributes with good accuracy. This suggests the possibility to search images using soft biometrics as search terms.

1. Introduction

In the computer vision community research agenda, extracting high level information from digital images and videos is still an open issue. The main idea is to extract semantically meaningful concepts from images or video, similarly to those that would be extracted and understood by a human. Such procedure, if possible and done automatically (without human intervention) would, certainly, allow a better usage of the huge amount of media (images and videos) currently recorded and stored.

In fact, the interpretation of visual contents can lead to several different outcomes, since it may vary according to the context the observer is immersed. For instance, one might be interested in finding all rectangular or circular shapes present an image, while other might be interested to find complex and highly variable objects, such as cars, animals, or people. An interesting visual content for most people is to identify other people by means of their physical appearance, and this is a key point to various important applications, such as surveillance.

Possibly, the most important and open issue about this extraction process is known as the semantic gap. It represents a kind of distance between the low level information (pixels, edges, shapes, texture) and its high level meaning. Most researchers claim that this information is not available inside the image, but it is observer-dependent. If this is really true, intelligent computational methods are required to deal with that gap.

Biometrics is a research field concerned with the metrics related to human features. There are several types of biometrics ranging, for instance, from physiological (such as fingerprint, DNA, retina) to behavioral (such as voice and gait). In most cases, the acquisition of such kind of biometrics requires the cooperation of the target person [1]. On the other hand, there is another kind of biometric data that can be extracted from images/videos. This kind of information, known as soft biometrics, is related to unspecific human attributes, such as clothing, gender, and ethnicity, for instance. By using these attributes, it is not possible to identify a person unambiguously. However, it is possible to reduce the range of possibilities when searching for a given individual. The great advantage of this kind of biometrics is that the cooperation of the subject is not needed for acquiring the data [2], therefore it fits perfectly for surveillance purposes.

In this paper, the focus is to classify people images according to three different attributes: upper clothes, lower clothes and gender. More specifically, the objective is, giving an image of a person, to identify the type of upper and lower clothes he/she is using, as well as his/her gender. The main motivation for the development of methods to deal with this problem is the improvement of video surveillance systems. Such methods would enable to search a video database

[☆] This paper has been recommended for acceptance by G. Sanniti di Baja.

[*] Corresponding author. Tel.: +55 041 33104697.
E-mail address: hugo.perlin@ifpr.edu.br, haperlin@gmail.com (H.A. Perlin).

http://dx.doi.org/10.1016/j.patrec.2015.07.012
0167-8655/© 2015 Elsevier B.V. All rights reserved.

using high level queries, such as “Find all men with blue t-shirt and black pants”, thus saving many hours of human effort to analyze and classify those images.

This is a very hard problem in several ways: there is a high variance in the way that people are dressed as well as in environmental illumination; people can be in many different poses; they also can be partially occluded by other objects or other people; finally, the background in which a target subject is can be complex and different from scene to scene, imposing more difficulties to the identification of the person.

Most classical approaches proposed in the literature are based on a pair: feature extractor and classifier. The former includes a large variety of general/specialized color, texture, and shape descriptors; and the latter uses machine-learning algorithms, such as a Support Vector Machine (SVM). The major problem with these approaches remains in the fact that they are strongly dependent on the human design and, thus, they are far from being done automatically. In this paper a different approach is proposed.

Instead of choosing which kind of feature extractor and classifier will be used, a Convolutional Neural Network (CNN) is employed. This is a deep learning method, where the feature extractor and the classifier are build in a supervised way, tailored according to the nature of the problem. Deep learning methods, such as CNN, have been used as solution to solve some interesting problems in computer vision, especially those related to pattern recognition, such as Optical Character Recognition (OCR), object recognition, pedestrian detection, among others. For some computer vision problems this methodology has achieved the state-of-the-art results.

The main contribution of this work is to create a method to develop an end-to-end supervised classifier capable of extracting high level (semantic) information from images. The content of the paper is as follows. In Section 2, we summarize some results from the recent literature related to the extraction and classification of soft biometrics. In Section 3, a brief review about the concepts of the CNN method is shown. The proposed methodology to extract and classify soft biometrics is presented in details in the Section 4. Next, in Section 5, a hand-designed feature extractor and classifier is reported for comparison purposes. The experiments done and the results obtained are reported in Section 6. The conclusion of the work and future research directions are discussed in Section 7.

2. Related work

Some soft biometrics methods for extracting semantic information from people were developed in the last years. Different methodologies are available taking advantage of certain properties of the problem.

The work of Hansen et al. [3] focused on annotating the humans’ features in surveillance videos. A person is described using the primary color of the hair, upper an lower body clothing, as well as his/her height. However, no classification of clothes was done. The proposed methodology includes a background subtraction algorithm, a color descriptor based on the Hue-Saturation-Value (HSV) color model, a height estimator, and a head direction evaluator.

Zhang et al. [4] proposed a methodology for clothes recognition, but restricted only to t-shirts. They present a survey to evaluate which kind of detail/pattern is the most relevant for classifying t-shirts. Based on this survey, some methods were proposed to evaluate the sleeve length, recognize collar and placket, color analysis, pattern recognition and shirt style recognition. More specifically, sleeve length recognition is based on face and skin detector, then, color segmentation and a one-level decision tree classifier.

The development of a robust color detection framework able to identify the colors of clothing in video under real illumination conditions was proposed by D’Angelo and Dugelay [5]. The objective was the identification of hooligans and the prevention of clashes in soccer matches. The proposed algorithm included stages of color constancy, color-space transformation and color matching. The results showed that the proposed methodology was efficient for the specific purpose.

Bourdev et al. [6] proposed a system to describe clothes of people using nine binary attributes. The detection and classification process relies on the Poselet detector, which uses a fully annotated training set. Besides Poselets, a strategy for skin detection and segmentation was also employed. Using a similar method, Weber et al. [7] proposed a clothing segmentation approach. In this case, the H3D dataset was employed to construct a segmentation mask based on the Poselets detection.

Bo and Fowlkes [8] proposed a methodology for pedestrian image segmentation based on hierarchical composition of parts and sub-parts of image segments. The candidate parts and sub-parts were derived from a superpixels segmentation code. For each candidate segment a score was calculated to determine if it is part of a pedestrian. To do this, authors used a shape descriptor and color and texture histograms. Although the reported results suggested a promising methodology, it turned out to have some drawbacks. The concept of superpixels was also used later by Yamaguchi et al. [9] for clothes labeling, based on pose estimation.

Chen et al. [10] presented a fully automatic system capable of learning 23 binary and 3 multi-class attributes for human clothing. Human pose estimation was performed to find the location of upper torso and arms. Then, 40 features were extracted and quantized. This set of features was used to train a SVM classifier for each desired attribute. A Conditional Random Field (CRF) was calculated to extract the mutuality between attributes.

Employing the well-known Viola–Jones face detector, a modification of the GrabCut segmentation algorithm, the MPEG-7 color descriptor, the HOG shape descriptor and skin color detection, Cushen and Nixon [11] presented a methodology for segmenting the upper body clothes in a mobile platform.

In Dong et al. [12] authors define Parselets as a group of semantic images obtained by low-level over-segmentation, having strong and consistent semantic meaning. With this representation, a deformable mixture parsing model was proposed, based on a set of hand-designed feature descriptors. Using this method those authors managed to parse a human image, obtained by pixel segmentation.

A fully automated clothing suggestion approach was proposed by Kalantidis et al. [13]. The authors used segmentation and hash process to extract and classify the clothes from a digital image. They used color quantization and LBP (Local Binary Patterns) as descriptors, and claimed that the proposed method was scalable and very time-efficient.

Sharma et al. [14] proposed a model for recognizing human attributes and actions. The model was based on a bag-of-words representation and SIFT (Scale-Invariant Feature Transform) features at multiple scales. The results related to human attributes were evaluated as good, but showed that this problem is far from being solved.

Most of the above-cited methodologies are based on the classical approach: some kind of segmentation method and/or hand-designed feature extractors followed by a classifier. In the following sections the main contribution of this paper will be presented: a end-to-end soft biometrics classification framework that needs no segmentation or image pre-processing.

3. Convolutional neural networks

The common process for automatic classification is based, mainly, in two factors, a feature extractor (FE) and a classifier. Usually, the feature extractor consists of hand-designed transformations of the raw input image, seeking to make the classification process more efficient. There are many image descriptors cited in the literature used as features for the classifier. Some usual methods are: HOG (Histogram of Oriented Gradient) [15], Speeded-up Robust Feature (SURF) [16],

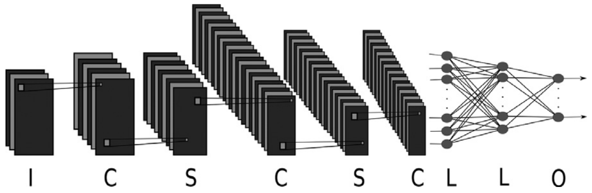


Fig. 1. An example of a CNN architecture. The dashed rectangle represents the feature extractor layers, and the continuous one, the classifier layers. Layers (I) represent the raw input image. Layers (C) represent the convolution operations. Layers (S) perform sub-sampling operations. Linear layers are represented by (L), and (O) represents the output layer.

Local Binary Pattern (LBP) [17], Oriented FAST and Rotated BRIEF (ORB) [18], and Scale-Invariant Feature Transform (SIFT) [19], among other. For the classifier, the most usual in computer vision is SVM, although one can find frequently Random Forests, Decision Trees, Multi-Layer Perceptrons (MLP) and others. In general, the choice for the feature extractor has a high impact in the classification results.

A hand-designed feature extractor requires a human expert to find the right data manipulations that will lead to good classification performance for the specific sort of images and objects to be detected. To overcome this drawback, the features could be learnt from the input data, such that the extractor is adjusted to fit the requirements for a given classification task. A method conceived to address these issues is the Convolutional Neural Network (CNN) [20]. A CNN is a special type of feed-forward neural network, where many hidden layers are employed, such that the CNN is a case of a deep learning method. The main characteristic of this method is the ability to learn several levels of features, allowing a more abstract representation of the input data. Thus, raw images can be presented to the network without any kind of preprocessing or previous feature extraction. Therefore, a CNN is an end-to-end classifier. Another advantage is that the feature extractors are constructed based on the data used for training. That is, the network learns to manipulate the input data in a manner to perform the classification task [21]. The architecture of an CNN is designed to include both the feature extractor and the classifier, as shown in Fig. 1.

The feature extraction module is composed by multiple stages, and their inputs and outputs are called feature maps. Each output map represents a particular feature extracted by a three-operations procedure: convolution, non-linear filtering and sub-sampling.

The input of a convolution layer is a three-dimensional (3D) array, with n_1 2D *feature maps* of size $n_2 \times n_3$. Each feature map is represented by x_i, and each component is x_{ijk}. The output is also a 3D array, y with m_1 maps of size $m_2 \times m_3$. A $k \times k$ trainable kernel is used to connect input features maps x to output feature map y. This connection is performed by a 2D discrete convolution operator and a trainable bias b_j. Eq. 1 represents this operation, where $*$ represents the 2D convolution.

$$y_i = b_j + \sum_i k_{ij} * x_i \tag{1}$$

The trainable kernel is called receptive field, and the idea here is to restrict connections from a given neuron to a small neighborhood in the immediately preceding layer, thus forming a tiny filter for feature extraction. This restriction was inspired by the complex arrangement of cells within the human visual cortex.

Since a common feature may be located at different positions in the input image, it is interesting to perform the extraction in the full image. The weight sharing mechanism allows neurons of the same feature map to have the same weights associated with different receptive fields. This allows extracting features independently of the position and reduces the number of parameters to be trained [21].

The non-linear filtering consists of the application of a squashing function to all components of each feature map. The common choice is the hyperbolic tangent function $tanh(x)$, where x is each component of the feature map. Recent works have shown that other functions can be more effective, such as the Rectified Linear Unit (ReLU), which is an activation function in the form of $f(x) = max(0, x)$ [22]. This kind of activation produces similar results compared to *tanh*, but requires less computation.

The sub-sampling mechanism involves the reduction of a feature map by a constant factor, providing invariance to small distortions and translations to the network. This reduction is performed over a $f_1 \times f_2$ neighborhood from the previous layer using an operator, such as sum, average or maximum. For instance, if an input feature map has 28 × 28 components, and the neighborhood size is 4 × 4, the output feature map will have 7 × 7 components, being reduced by a factor of 4 [20,23].

The last module of a CNN consists of the classifier itself, in which the inputs are the outputs from the last feature maps. In general, a common linear feed-forward MLP is employed. The amount of input neurons, hidden layers and output neurons is problem-dependent.

For adjusting the weights of the network, a variation of the backpropagation learning algorithm was proposed by LeCun et al. [20]. Thus, the layers are trained in a supervised way, seeking to reduce the error between the predicted and the expected results. Usually, this is done by the Stochastic Gradient Descent (SGD) algorithm.

Thanks to their high capacity of learning mid-level features by a trainable feature extractor, CNNs are versatile and robust. Consequently, they have been used for several computer vision problems, for instance, handwritten characters recognition [20]; generic object classification [22]; face detection and recognition [24].

4. The proposed approach

In this work we aim at extracting and classifying automatically soft biometric information from images containing a person. The classification includes the gender, as well as the upper and lower clothes wearing by the individual, based on their appearance. For the upper clothes (UC) there are two possible classes: short and long sleeves, and for the lower clothes (LC), pants and shorts. We are also interested in a more difficult task: determining the gender of the subject into two classes (male and female), based on the full body appearance. By now, we assume that the classifier will operate after a people detector module, that is, a frame containing a person will be the input for the classifier.

In this work, we propose two different operation modes (OM) for the CNN classifier. The first one (OM #1) consists of three independent trained sub-classifiers, each one dealing with a different classification problem: an UC classifier, a LC classifier, and a gender classifier. The output of each classifier is a vector with two units, and a winner-takes-all strategy is used to indicate which is the class of the image for each soft biometric.

The second mode (OM #2) deals with the three soft biometrics at the same time, describing each sample based on the answer of just one classifier. In this way, the classifier is trained to take into account all the semantic aspects within the input image. Here, the output of the classifier is a continuous vector with three units, where each unit represents a soft biometric with values in the range $[-1..+1]$. Therefore, the output of the classifier is a multilabel answer, allowing three different attributes to be evaluated at the same time. This is not usual for CNNs, since, in most cases, just a single class would be expected for each image.

Since the CNN allows the construction of an end-to-end classifier, the proposed approach for the semantic description of a person is based only on the raw input image, without any preprocessing. Table 1 shows some details of the CNN architecture used in this work, which is the same for both OM #1 and OM #2. The difference is only in the number of outputs. The architecture parameters were

Table 1
Description of the architecture of the CNNs used in this work. Each column represents a layer. ∗ For OM#1, the number of outputs is 2 for each classifier. In OM#2 the number of outputs is 3.

	Conv1	Maxpool1	Conv2	Maxpool2	Conv3	Maxpool3	Linear1	Linear2
Input maps	3	32	32	64	64	128	1152	768
Input size	128 × 128	60 × 60	30 × 30	24 × 24	12 × 12	6 × 6	-	-
Output maps	32	32	64	64	128	128	768	∗
Output size	60 × 60	30 × 30	24 × 24	12 × 12	6 × 6	3 × 3	-	-
Kernel size	10 × 10	2 × 2	7 × 7	2 × 2	7 × 7	2 × 2	-	-
Stride	2 × 2	2 × 2	1 × 1	2 × 2	1 × 1	2 × 2	-	-

Table 2
Main parameters of the hand-designed classifiers.

C4.5	
Confidence factor	0.25
Minimum # objects per leaf	10
Pruning	True
SVM	
Cost parameter for C-SVM	1.0
Kernel degree	3
Tolerance for termination	0.001
MLP	
Auto build/connect hidden layers	True
Number of hidden layers	$(\#classes + \#features)/2$
Learning rate	0.3
Momentum	0.2
Training epochs	500

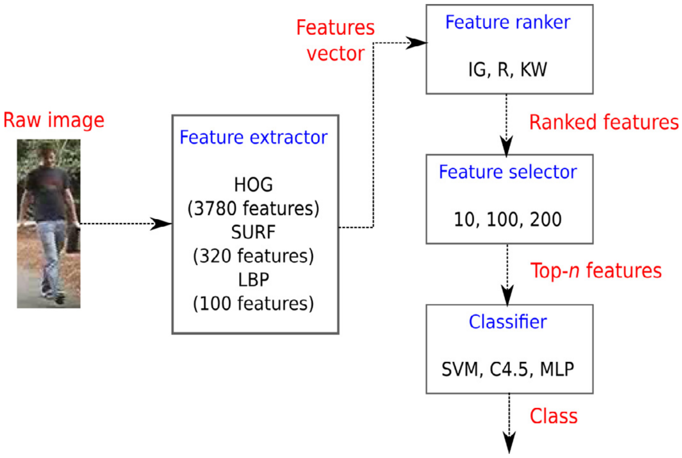


Fig. 2. Hand-designed feature extraction and classification process.

adjusted based on the current literature, as well as by preliminary experiments.

The input for the networks is a 3D array with 3 × 128 × 128 elements, defining a 128 × 128 RGB image.

The first convolution layer has 32 feature maps obtained by a set of 10 × 10 convolution kernels, using 2 × 2 strides. The number of trainable parameters in this layer is 832 (trainable kernel plus the bias). After the application of the ReLU squashing function, the next step is the sub-sampling, accomplished by maxpooling. The neighborhood size used was 2 × 2. These three steps compose the first stage of the network.

The second stage is composed by a convolution layer with 64 feature maps and a 7 × 7 kernel. Again, after the convolution, both the ReLU function and the maxpooling operation were applied. The number of trainable parameters in this layer is 3200.

The third stage and the last part of the so-called feature extractor is another convolution step, which generates 128 feature maps using 7 × 7 kernels. Following, the convolution the ReLU and the maxpooling operation were applied.

The output generated by the last sub-sampling operation is 128 × 3 × 3 large, rearranged as a 1152 unidimensional vector. This is the input to the linear stage. It is possible to observe that the convolution and sub-sampling layers work together for reducing dimensionality.

The linear stage is composed by the input layers, followed by a hidden layer containing 768 neurons. The output layer has 2 neurons for OM #1 and 3 neurons for OM #2. The number of neurons in the hidden layer was determined using the relationship 2/3 of the size of the preceding layer.

For OM #1, a winner-takes-all strategy was used to determine the classifier output. For OM #2, each output was evaluated separately.

During the training process, the network neurons tend to co-adapt their weights based on the neighborhood elements. This phenomenon can lead the network to lose its generalization capability. To prevent this phenomenon, some randomly chosen neurons are omitted from the network, with probability 0.5, so the elements cannot rely on other units. Basically, this is the dropout strategy proposed by [25].

The adjustment of the network's weights was carried by the back-propagation of the error using the negative log-likelihood for OM #1, and the mean squared error for OM #2, as loss metric and the SGD algorithm as the optimization strategy. The learning rate starts at 0.005 and decreases by half every 30 training epochs. The momentum was set to 0.9 so as to help the convergence of the method. The SGD optimization process was executed for a maximum of 1000 epochs.

The CNN training process is very time consuming, not only because the network has many layers and neurons but, also, because many computations are required at each step of the training. Considering the algorithmic issues and the high level parallelism of the operations involved, an interesting strategy is the use of GPGPU (General-Purpose Graphics Processing Unity) as the horse power to reduce the total training time. In order to obtain efficiency from the GPGPU, a mini-batch scheme should be used. Here, the size of mini-batch was set to 128 as suggested by [22].

5. Hand-designed image feature extractor and classifier

A CNN approach does not require the definition of a feature extractor or a classifier before being used. Therefore, we created a hand-designed feature extractor and classifier, shown in Fig. 2, for comparing its performance with the CNN.

The first part of the diagram is the feature extractor that uses the raw image (supposedly containing the image of a person with clothes) to compute a 4200-long feature vector. This vector is composed by three groups of elements extracted from the image: 3780 features using HOG; 320 features from the top-5 interest points (each one produces a 64 float-point vector) using SURF; and 100 features using LBP. HOG is a very popular method that, for years, was considered the state-of-art people detector [15]. SURF is also a popular

Table 3
Number of original images per class used in the experiments.

	Gender		Lower Clothes		Upper Clothes	
	Male	Female	Long	Short	Pants	Shorts
H3D	755	600	320	287	438	447
ViPer	636	624	236	182	526	456
HATdb	2615	2615	492	492	771	771
Total	4006	3839	1048	961	1735	1674

method [16], very efficient for many applications, from object recognition to face detection and recognition [26]. LBP [17] is a simple, but robust, texture descriptor, and it was successfully applied to face recognition problems [27] and pedestrian detection [28]. The LBP descriptor is a histogram of the detected binary patterns in an image. In this paper the histogram size was defined as 100 bins. All 4200 elements of the feature vector were locally normalized.

Next, the predictive power of each feature regarding the class is accessed by applying some well-known statistical methods: Information Gain (IG), Spearman's rank correlation coefficient R, and the Kruskal–Wallis score (KW).

IG, also known as Mutual Information in some domains, is an entropy-based measure that evaluates the gain of information with respect to the class when a specific feature is chosen. Given the entropies of two features, $H(X)$ and $H(Y)$, $IG(X, Y)$ is given by $IG(X, Y) = H(X) - H(X|Y)$, where the last term is the conditional entropy of X given Y. Considering X as the values of a feature for the several image samples, and Y as the corresponding classes to which they belong to, therefore $IG(X, Y)$ gives a measure of how predictable a class is considering only that feature. This is the rationale for using IG for feature selection [29].

Spearman's rank correlation coefficient R is a nonparametric measure between two variables [30]. This measure of correlation can be used for both continuous or discrete variables and it does not require that data has normal distribution. If one of the variables is a feature and its values, and the other the classes, R can be used as a method for ranking sets of features according to their predictability.

The KW score (also known as H test) is a nonparametric statistic frequently used for comparing means of more than two populations when they are not normally distributed and their variances are not equal [31]. Data values are transformed into ranks and considered as a whole and separated by classes of samples. If the samples of a class differ significantly from those of other classes, this statistic will give a high value and, if the differences are small, so is the result of the test.

When applied to feature selection, the higher the value of the score, the better. For features extracted from image data, this means that a given feature has quite different values for a given class.

The result of the feature ranker are three 4200-long vectors which elements are ranked according to the values computed by IG, R and KW, as described above.

Next, a threshold is applied to cut out the length of the ranked feature vector. Three values, empirically chosen, were used for selecting the top features: 10, 100 and 200. This was done to limit the length of the feature vectors to be used by the classifiers (next block), aiming at eliminating possible noise and improving performance.

Finally, the top-n features ($n = 10, 100$ or 200) were applied to classifiers. Three different classifiers were used here: C4.5, SVM and MLP. C4.5 is a well-known tree-based classifier based on IG that has been used for years as the baseline for comparison of classification methods [32]. SVM [33] is one of the most used classifiers not only in computer vision but, also, in pattern recognition and data mining. MLP is a feed-forward neural network with multiple hidden layers that is frequently used in general classification tasks, including images [34]. The main parameters of the classifiers used are shown in Table 3.

6. Experiments and results

6.1. Evaluation protocol

To evaluate the performance and robustness of the proposed approach, several experiments were done. The main concern is to verify the ability of the CNN to correctly extract the soft biometrics from raw input images. This requires a CNN capable of learning both a feature extractor and a classifier, in a data-driven way. We compared the results achieved by the CNN against multiple classifiers trained using a hand-designed feature extractor.

The three soft biometrics to be classified in this work are binary. To use the same language when dealing with binary classifiers, one class will be considered positive and the other negative. Therefore, four different measures can be computed using the outcomes of a classifier for a set of instances, in this case, input images:

- *true positive (tp):* number of positive instances that were correctly classified as positive;
- *true negative (tn):* number of negative instances that were correctly classified as negative;
- *false positive (fp):* number of negative instances that were wrongly classified as positive;
- *false negative (fn):* number of positive instances that were wrongly classified as negative;

From the above measures, several useful ones can be derived, including accuracy (Acc) (Eq. 2), which is the proportion of correctly classified instances, regardless of they are positive or negative.

$$Acc = \frac{(tp + tn)}{(tp + tn + fp + fn)} \tag{2}$$

A graphical aid to visualize the performance of the classifiers used in this work is the ROC (Receiver Operating Characteristics) curve [35]. Points of a ROC curve represent the trade-off between true-positive and false-positive rates for a given discrimination threshold. Therefore, when plotting the performance of a classifier, the best operation point would be that at the top-left corner, representing the optimal ability of the classifier to separate classes. As the curve approximates to the ascending diagonal, which represents the random guessing, it indicates that the classifier has a poor ability to separate classes. Using this method, the quantitative evaluation of a classifier is given by the Area Under the Curve (AUC), which is the area in the ROC space covered by the classifier curve. The AUC varies from $[0 \ldots 1]$, where 1 indicates the best possible performance.

For all cases, the classifiers were trained using a 10-fold cross-validation strategy. This is done seeking to access their generalization ability.

6.2. Implementation

The CNNs were implemented using the Torch7 library [36] that is a very flexible and efficient package for implementing neural networks and machine learning algorithms. An important feature of this library is its capability for using GPGPU. It provides an easy and almost transparent way to switch between CPU and GPGPU environments, allowing a significant reduction of development and training time of a CNN-based solution.

The hand-designed image descriptor was implemented using the OpenCV computer vision library, version 2.4.9.

Other classifiers were trained and tested using WEKA (Waikato Environment for Knowledge Analysis) software, version 3.6.11 [37].

The running environment was a Linux-based desktop computer equipped with an Intel i7 quad-core processor, 32 GBytes of RAM and a Nvidia GTX 660 GPU board running CUDA 6.0. The CNN training process exhaustively used the GPGPU power, and the other classifiers took advantage of parallel processing by means of multi-threads.

6.3. Image datasets

During the experiments, three datasets were used for training and the evaluating the classifiers. The image set was composed by a mix of the H3D dataset, provided by Bourdev et al. [6], the ViPer dataset created by Gray and Tao [38], and the HATdb made available by [39]. The idea behind using a mix of different image datasets is to indirectly improve the generality of the proposed method for real-world images.

The H3D includes a large variety of people, wearing different clothes, standing in different poses, sometimes subject to occlusion and cluttered background. Therefore, this dataset is challenging for pattern recognition methods. The original dataset contains 4013 images annotated with 9 binary attributes (is male, has long hair, has glasses, has hat, has t-shirt, has long sleeves, has shorts, has jeans, has long pants). From this dataset we selected only those images with people standing up, thus remaining 1355 images. This was done by adjusting bounding boxes around a person with an aspect ratio greater than 2. Then, all examples were normalized in size to fit a frame of 128 × 64 pixels. Seeking to use just representative examples, a careful manual selection an annotation was done.

The ViPer dataset is composed by 1264 images of 128 × 48 pixels taken from people walking in various places. Each subject has two different images taken from different positions. Both images were used to create the training set. This dataset does not have the desired labels to be used during the network training. Consequently, a manual annotation was necessary.

The HATdb dataset is composed by 9344 images, annotated with 27 attributes. We selected a subset, only using images classified as standing people, and used the bounding box information to crop the person and resize it to 128 × 64 pixels. We also performed an additional revision of the annotation provided.

Considering that the classifiers were designed to deal with 128 × 128 images, the size of all the images was standardized by padding with 0's to achieve the desired dimensions.

The distribution of images per class in each dataset is shown in Table 3.

Since the number of available images was small for effectively training the CNN, we expanded the dataset by using random transformations, such as translations (up to ± 10 pixels for x and y), scaling (from 0.98 to 1.1), rotation (up to ± 10^o) and horizontal flip.

This data augmentation strategy is applied in an on-line way, i.e., for each epoch in the CNN training, random transformations are applied for each training sample. This is done in such a way that the total number of examples shown to the CNN at each epoch is around to 10,000 samples per class.

Similarly, the same data augmentation procedure was used for the other classifiers, reaching the same number of 10,000 samples per class.

6.4. Hand-designed image classifier

Considering the methodology of the hand-designed image classifier, as described in Section 5, a factorial experiment was done for each classification problem: {IG, R, KW} × {10, 100, 200} × {C4.5, SVM, MLP}. Considering the three problems: gender, top and bottom clothes, we performed 3 × 3 × 3 × 3 experiments. For each combination of options, a 10-fold cross-validation procedure was performed. Therefore, a total of 810 different runs were done. The mean accuracy and the standard deviation for upper clothes is shown in Fig. 3. Similar behavior occurred with the other two soft biometrics.

It is noticed that the length of the feature vector has an important influence in the performance of the classifiers. As the number of features used increases, the overall performance also increases. For all the three classification problems, using only the top-10 features, results are close to a random guess (∼50%). On the other hand, the best results were achieved using the subset of top-200 features. Analyzing the results from the point of view of the feature ranking methods, it is possible to see that the top features ranked by the KW method leaded to the best results. Also, the best-performing classifier was the standard MLP.

Therefore, we selected the configuration of top-200 features selected by KM classified by MLP to report the results in details, shown in Table 4.

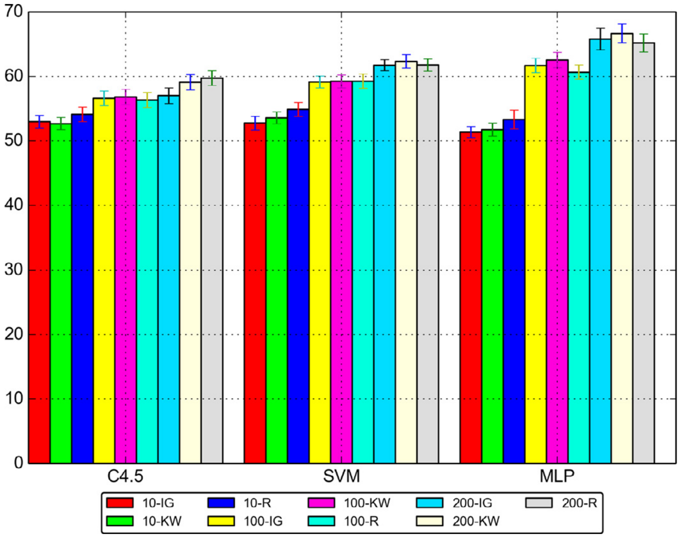


Fig. 3. Comparison of hand-designed classifiers according to ranking methods and number of top features, for the Upper Clothes attribute.

Table 4
Mean classification accuracy for the OM #1 and #2 CNNs and Hand-designed classifiers.

		CNN OM #1	CNN OM #2	Hand-designed
		Acc(%) ± *std*	*Acc(%)* ± *std*	*Acc(%)* ± *std*
Upper Clothes	Long	78.23 ± 5.83	82.82 ± 4.10	66.70 ± 1.19
	Short	78.01 ± 8.80	77.53 ± 3.61	66.60 ± 1.63
Lower Clothes	Long	83.92 ± 6.73	80.42 ± 5.70	69.98 ± 1.65
	Short	86.71 ± 2.41	89.39 ± 2.27	70.12 ± 1.32
Gender	Male	74.43 ± 9.54	62.80 ± 8.21	70.10 ± 1.24
	Female	61.63 ± 8.36	59.29 ± 8.74	57.00 ± 2.35

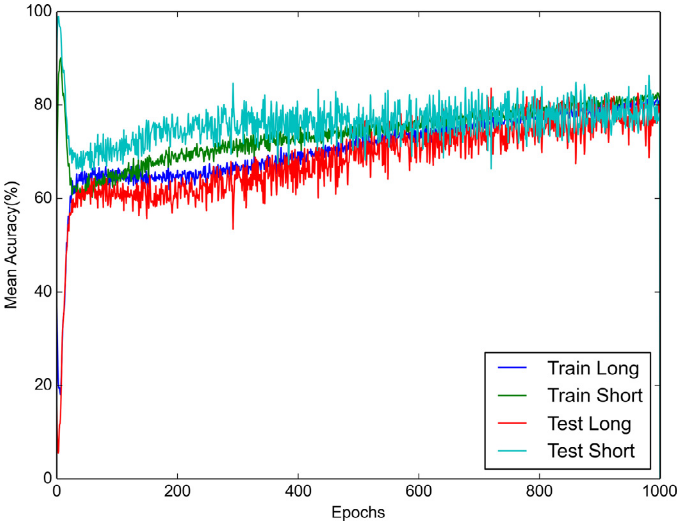


Fig. 4. Accuracy curve during OM #1 CNN training for the Upper Clothes attribute.

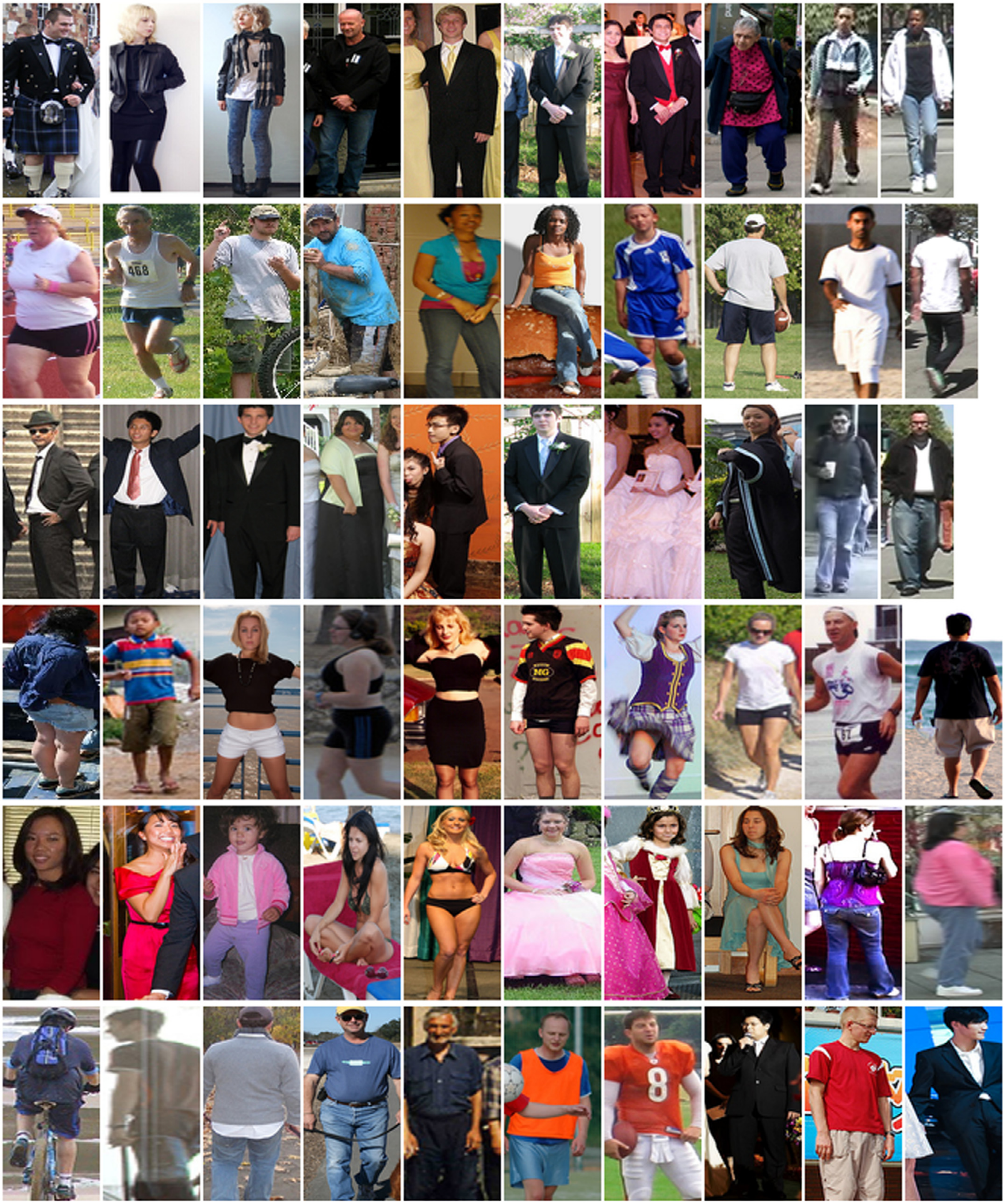


Fig. 5. The top-10 OM #1 CNN results for each class. *First and second rows:* samples classified as Long and Short for Upper Clothes. *Third and fourth rows:* samples classified as Long and Short for Lower Clothes. *Fifth and sixth rows:* samples classified as Female and Male for Gender.

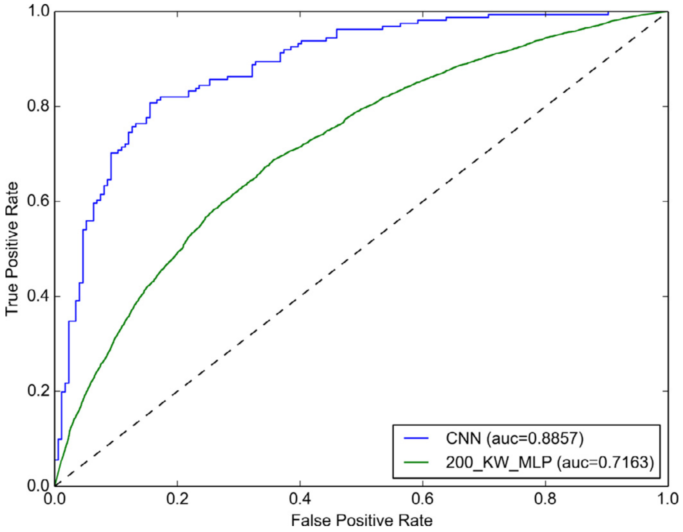


Fig. 6. ROC plot comparing the OM #1 CNN and Hand-designed classifiers for the Upper Clothes attribute.

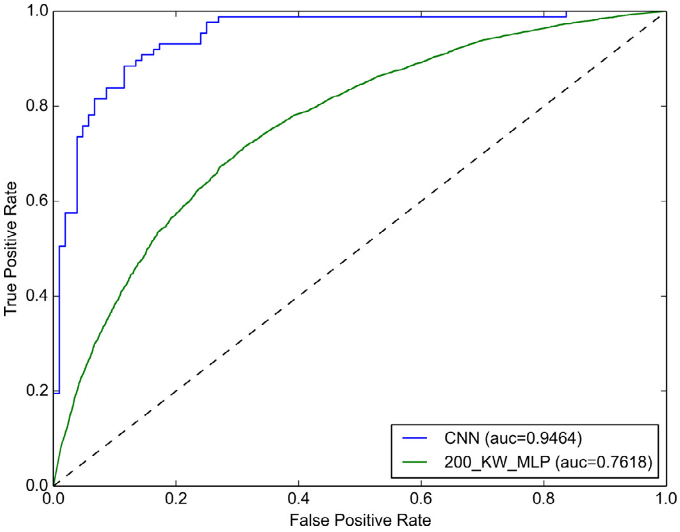


Fig. 7. ROC plot comparing the OM #1 CNN and Hand-designed classifiers for the Lower Clothes attribute.

6.5. Classification results for OM #1

In order to visualize the evolution of the CNN's learning along the training process of the three different CNN classifiers, the average *Acc* in the 10-fold cross-validation process was plotted, for both, the training and the testing steps. Fig. 4, shows the curves for upper clothes. Similar curves were obtained for other two soft biometrics.

Observing these curves, it is noticed that the CNN has an interesting learning ability. The usage of the image augmentation strategy at each training epoch introduces a challenge to the classifier, since it rarely will face the same training example multiple times. This seems to influence the behavior of the classifier in a good manner, preventing it from overfitting.

The final predictive mean accuracy achieved in the test by the three OM #1 CNN classifiers is reported in Table 4.

The top-10 highest output values for the best CNN trained for each class is shown in Fig. 5. It is noticed that, in all classes a high variability is present in the images (poses, clothes appearance, background, etc). However, it is also noticed that the main characteristic of the classes is present. For instance, for clothes classification, gender information is not taken into account, since there are male and female individuals in the images. Another interesting detail is related to the results for short upper clothes. Among the results, three individuals are wearing long pants and short sleaves. This indicates that the CNN spotted the right information to make the decision.

The CNN classifiers were compared with another classifiers using the hand designed descriptors mentioned in Section 5. Observing the AUC values from the ROC curves of Figs. 6–8, it is found that CNN achieved better performance than the other classifiers, for the three different attributes.

The gender, as expected, seems to be the most difficult soft biometric to be classified, since, actually, it is also difficult in the real-world. All the classifiers tested produced poor results, even so, CNN was better than the other classifiers.

6.6. Classification results for OM #2

In OM #2, for each image instance presented, the classifier should give values representing the classification for all three attributes (upper clothes, lower clothes and gender) at the same time. This is significantly more difficult than the previous case in OM #1 with three independent classifiers. In this experiment we seek to compare the performance of the two operation modes.

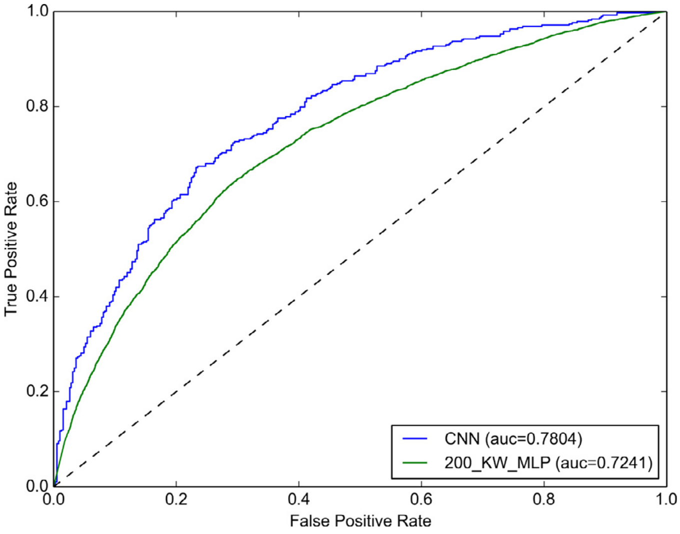


Fig. 8. ROC plot comparing the OM #1 CNN and Hand-designed classifiers for the Gender attribute.

Similarly as before, the classifier was trained using a 10-fold cross-validation scheme. The results obtained, represented by the mean accuracy and the standard deviation for each attribute are shown in Table 4.

We also plotted the ROC curves for each attribute obtained by both OMs in Figs. 9–11, .

These plots shows that the results for both OMs are similar, for upper and lower clothes, suggesting that the CNN was able to learn how separate the classes simultaneously. The performance of OM#2 with gender attribute is worse than OM#1, which is an intriguing fact, and will be addressed in depth in the future.

7. Discussion and conclusions

In the computer vision area, the automatic extraction of high-level information from visual data still remains a challenge, since no method has yet reputed as a gold standard for most problems.

A sort of human-focused data is soft biometrics, which is related to the appearance and features of a person. The development of methods to extract and classify this type of data allows a better usage of the vast amount of stored images and videos, making

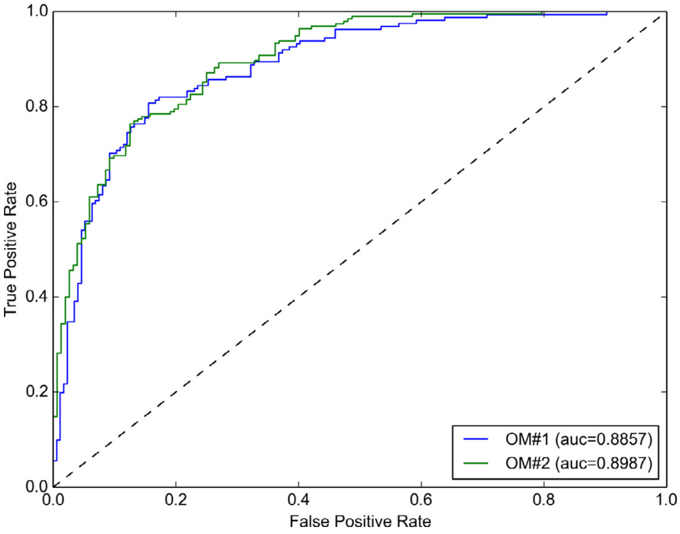


Fig. 9. ROC curves comparing the two OMs #1 and #2 CNNs for the Upper Clothes attribute.

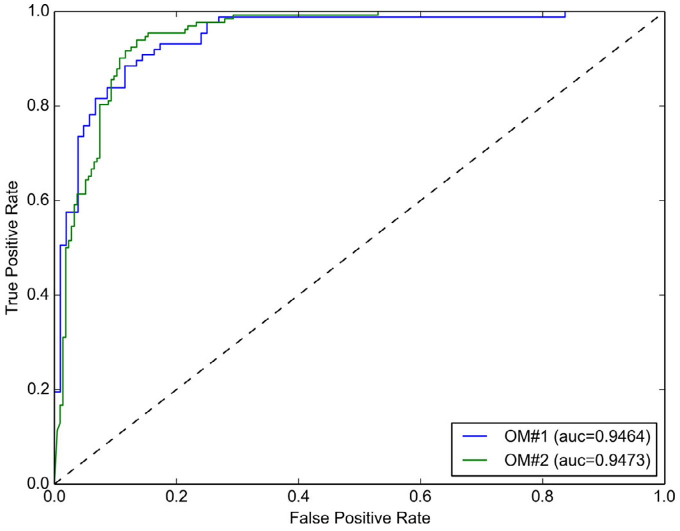


Fig. 10. ROC curves comparing the two OMs #1 and #2 CNNs for the Lower Clothes attribute.

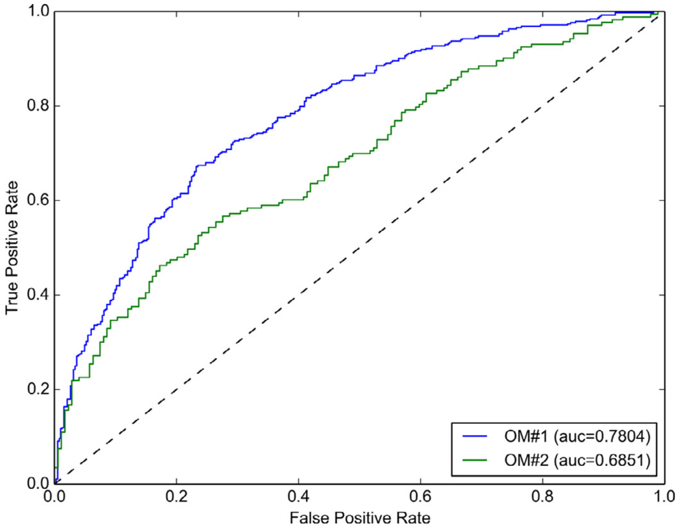


Fig. 11. ROC curves comparing the two OMs #1 and #2 CNNs for the Gender attribute.

it possible to search by the contents of the image. This kind of information is very difficult to be correctly annotated and classified, not only due to its large variability, but also, due to its multiple semantic meaning.

In recent years, several methods were proposed in the literature using many preprocessing tools, feature extraction and classifiers. Choosing the right combination of these methods and the correct tuning of parameters is a hard task, and it is usually done by a trial-and-error process. This procedure demands intensive computation and the ability of experts to adjust the parameters of the system, as we showed before in this paper.

Seeking alternatives to this difficult computer vision problem, this paper investigates the performance of a CNN to build an end-to-end classifier, in a data-driven manner. Basically, it learns how to extract features from the raw images, so that they can be useful for the subsequent classification of the images. There is no need for preprocessing, meaning that the raw images can be used directly as inputs. This capability of the CNN allow the construction soft biometric classifiers in a relative simple way.

Amongst the drawbacks encountered during this work, the most important to mention is the great difficulty in finding appropriate image datasets, concerning size, quality, and variability. Mainly, this due to the need for a human to analyze, classify and annotate images.

Overall, the results of our experiments show that the proposed CNN approach provided promising results, achieving a reasonable accuracy. For clothes classification the performance can be considered good. Notwithstanding, for gender, which is a very challenging problem, even for humans, there is still the need for further improvement.

Another interesting result is related to the performance of the CNN when trained in the two different operation modes. Although, the final performance of both modes were similar for clothes classification, the computation effort demanded by the OM#2 is lower than OM#1, since the classifier was trained to deal with the three attributes in the same time.

As future work we intend to improve the results using other image datasets, as well as to study the tuning of the CNN parameters. Other work will focus on the application of the proposed approach to automatic video surveillance annotation. Since the proposed method uses as input the annotated raw data, it could be easily extended to other soft biometrics.

Acknowledgments

This work was partially supported by the Brazilian National Research Council (CNPq) and Fundação Araucária/CAPES for grants no. 308803/2013-2 to H.S. Lopes, and no. 45354 to H.A. Perlin.

References

[1] M.-G. Kim, H.-M. Moon, Y. Chung, S.B. Pan, A survey and proposed framework on the soft biometrics technique for human identification in intelligent video surveillance system, J. Biomed. Biotechnol. 2012 (2012), doi:10.1155/2012/614146.

[2] D. Reid, S. Samangooei, C. Chen, M. Nixon, A. Ross, Chapter 13 - soft biometrics for surveillance: An overview, in: C. Rao, V. Govindaraju (Eds.), Handbook of Statistics Machine Learning: Theory and Applications, Handbook of Statistics, 31, Elsevier, 2013, pp. 327– 352. http://dx.doi.org/10.1016/B978-0-444-53859-8.00013-8.

[3] D.M. Hansen, B.K. Mortensen, P.T. Duizer, J.R. Andersen, T.B. Moeslund, Automatic annotation of humans in surveillance video, in: Proceedings of the Fourth Canadian Conference on Computer and Robot Vision., IEEE Press, Piscataway, NJ, 2007, pp. 473–480, doi:10.1109/CRV.2007.12.

[4] W. Zhang, B. Begole, M. Chu, J. Liu, N. Yee, Real-time clothes comparison based on multi-view vision, in: Proceedings of the Second ACM/IEEE International Conference on Distributed Smart Cameras., IEEE Press, Piscataway, NJ, 2008, pp. 1–10, doi:10.1109/ICDSC.2008.4635727.

[5] A. D'Angelo, J.L. Dugelay, Color based soft biometry for hooligans detection, in: Proceeding of the 2010 IEEE International Symposium on Circuits and Systems., IEEE Press, Piscataway, NJ, 2010, pp. 1691–1694, doi:10.1109/ISCAS.2010.5537508.

[6] L. Bourdev, S. Maji, J. Malik, Describing people: A poselet-based approach to attribute classification, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE Press, Piscataway, NJ, 2011, pp. 1543–1550, doi:10.1109/ICCV.2011.6126413.

[7] M. Weber, M. Bauml, R. Stiefelhagen, Part-based clothing segmentation for person retrieval, in: Proceedings of the IEEE International Conference on Advanced Video and Signal-Based Surveillance, IEEE Press, Piscataway, NJ, 2011, pp. 361–366, doi:10.1109/AVSS.2011.6027351.

[8] Y. Bo, C.C. Fowlkes, Shape-based pedestrian parsing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition CVPR '11., IEEE Computer Society, Washington, DC, USA, 2011, pp. 2265–2272, doi:10.1109/CVPR.2011.5995609.

[9] K. Yamaguchi, M.H. Kiapour, L.E. Ortiz, T.L. Berg, Parsing clothing in fashion photographs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012., IEEE Press, Piscataway, NJ, 2012, pp. 3570–3577, doi:10.1109/CVPR.2012.6248101.

[10] H. Chen, A. Gallagher, B. Girod, Describing clothing by semantic attributes, in: Proceedings of the 12th European Conference on Computer Vision - Volume Part III ECCV'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 609–623, doi:10.1007/978-3-642-33712-3-44.

[11] G. Cushen, M.S. Nixon, Mobile visual clothing search, in: IEEE International Conference on Multimedia and Expo Workshops., IEEE Press, Piscataway, NJ, 2013, pp. 1–6, doi:10.1109/ICMEW.2013.6618404.

[12] J. Dong, Q. Chen, W. Xia, Z. Huang, S. Yan, A deformable mixture parsing model with parselets, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE Press, Piscataway, NJ, USA, 2013, pp. 3408–3415, doi:10.1109/ICCV.2013.423.

[13] Y. Kalantidis, L. Kennedy, L.-J. Li, Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos, in: Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, ACM, New York, NY, USA, 2013, pp. 105–112, doi:10.1145/2461466.2461485.

[14] G. Sharma, F. Jurie, C. Schmid, Expanded Parts Model for Human Attribute and Action Recognition in Still Images, in: Proceedings of the International Conference on Computer Vision and Pattern Recognition, IEEE, Portland, OR, USA, 2013, pp. 652–659, doi:10.1109/CVPR.2013.90.

[15] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Washington, DC, USA, 2005, pp. 886–893, doi:10.1109/CVPR.2005.177.

[16] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vis. Imag. Underst. 110 (3) (2008) 346–359, doi:10.1016/j.cviu.2007.09.014.

[17] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987, doi:10.1109/TPAMI.2002.1017623.

[18] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to SIFT or SURF, in: IEEE International Conference on Computer Vision, 2011, pp. 2564–2571, doi:10.1109/ICCV.2011.6126544.

[19] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, doi:10.1023/B:VISI.0000029664.99615.94.

[20] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324, doi:10.1109/5.726791.

[21] Y. LeCun, K. Kavukcuoglu, C. Farabet, Convolutional networks and applications in vision, in: Proceedings of the IEEE International Symposium on Circuits and Systems, IEEE Press, Piscataway, NJ, USA, 2010, pp. 253–256, doi:10.1109/ISCAS.2010.5537907.

[22] A. Krizhevsky, I. Sutskever, G. Hinton, ImageNet classification with deep convolutional neural networks, in: P. Bartlett, F. Pereira, C. Burges, L. Bottou, K. Weinberger (Eds.), Advances in Neural Information Processing Systems, 25, NIPS, Nevada, USA, 2012, pp. 1106–1114.

[23] Y.-L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, in: J. Furnkranz, T. Joachims (Eds.), Proceedings of the 27th International Conference on Machine Learning, Omnipress, Madison, WI, USA, 2010, pp. 111–118.

[24] H. Khalajzadeh, M. Mansouri, M. Teshnehlab, Hierarchical structure based convolutional neural network for face recognition, Int. J. Comput. Intell. Appl. 12 (03) (2013) 13500181, doi:10.1142/S1469026813500181.

[25] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, CoRR (2012).

[26] C. Chidambaram, H. Vieira Neto, L.E.B. Dorini, H.S. Lopes, Multiple face recognition using local features and swarm intelligence, IEICE Trans. Inf. Syst. E97-D (6) (2014) 1614–1623, doi:10.1587/transinf.E97.D.1614.

[27] T. Ahonen, A. Hadid, M. Pietikainen, Face description with local binary patterns: Application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 28 (12) (2006) 2037–2041, doi:10.1109/TPAMI.2006.244.

[28] G. Gan, J. Cheng, Pedestrian detection based on HOG-LBP feature, in: Seventh International Conference on Computational Intelligence and Security., 2011, pp. 1184–1187, doi:10.1109/CIS.2011.262.

[29] C. Lee, G.G. Lee, Information gain and divergence-based feature selection for machine learning-based text categorization, Inf. Process. Manage. 42 (1) (2006) 155–165.

[30] A.G. Bluman, Elementary Statistics, 7th, McGraw-Hill Higher Education, New York, USA, 2009.

[31] G.W. Corder, D.I. Foreman, Nonparametric Statistics: a step-by-step approach, J. Wiley & Sons, New York, NY, USA, 2014.

[32] J.R. Quinlan, Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, USA, 1993.

[33] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297, doi:10.1007/BF00994018.

[34] Y.V. Venkatesh, On the classificaton of multispectral satellite images using the multilayer perceptron, Pattern Recognit. 36 (9) (2003) 2161–2175, doi:10.1016/S0031-3203(03)00013-X.

[35] T. Fawcett, An introduction to roc analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874, doi:10.1016/j.patrec.2005.10.010.

[36] R. Collobert, K. Kavukcuoglu, C. Farabet, Implementing neural networks efficiently, in: G. Montavon, G. Orr, K.-R. Muller (Eds.), Neural Networks: Tricks of the Trade, Lecture Notes in Computer Science, 7700, Springer, 2012, pp. 537–557.

[37] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The weka data mining software: An update, SIGKDD Explor. Newsl. 11 (1) (2009) 10–18, doi:10.1145/1656274.1656278.

[38] D. Gray, H. Tao, Viewpoint invariant pedestrian recognition with an ensemble of localized features, in: Proceedings of the 10th European Conference on Computer Vision: Part I, in: Lecture Notes in Computer Science, 5203, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 262–275, doi:10.1007/978-3-540-88682-2-21.

[39] G. Sharma, F. Jurie, Learning discriminative spatial representation for image classification, in: Proceedings of the British Machine Vision Conference, BMVA Press, 2011, pp. 6.1–6.11.