IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 7, 2019, accepted December 4, 2019, date of publication December 24, 2019, date of current version January 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962010

Clustering Facial Attributes: Narrowing the Path From Soft to Hard Biometrics

ANDREA F. ABATE[1], (Member, IEEE), PAOLA BARRA[1], (Member, IEEE), SILVIO BARRA[2], (Member, IEEE), CRISTIANO MOLINARI[3], MICHELE NAPPI[1], (Member, IEEE), AND FABIO NARDUCCI[4], (Member, IEEE)

[1]Department of Computer Science, University of Salerno, 84084 Fisciano, Italy
[2]Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari, Italy
[3]Polo Scientifico, Presidenza del Consiglio dei Ministri, 00187 Rome, Italy
[4]Department of Science and Technology, University of Naples Parthenope, 80143 Naples, Italy

Corresponding author: Silvio Barra (silvio.barra@unica.it)

This work was supported in part by the Grant PRIN 2015 COSMOS: "COntactlesS Multibiometric mObile System in the wild," and in part by the Ministry of Education, University and Research, under Grant 201548C5NT.

ABSTRACT Despite the success obtained in face detection and recognition over the last ten years of research, the analysis of facial attributes still represents a trend topic. Keeping the full face recognition aside, exploring the potentials of soft biometric traits, i.e. singular facial traits like the nose, the mouth, the hair and so on, is yet considered a fruitful field of investigation. Being able to infer the identity of an occluded face, e.g. voluntary occluded by sunglasses or accidentally due to environmental factors, can be useful in a wide range of operative fields where user collaboration cannot be considered as an assumption. This especially happens when dealing with forensic scenarios in which is not unusual to have partial face photos or partial fingerprints. In this paper, an unsupervised clustering approach is described. It consists in a neural network model for face attributes recognition based on transfer learning whose goal is grouping faces according to common facial features. Moreover, we use the features collected in each cluster to provide a compact and comprehensive description of the faces belonging to each cluster and deep learning as a mean for task prediction in partially visible faces.

INDEX TERMS Clustering methods, face detection, principal component analysis, Eigenfaces, convolutional neural networks.

I. INTRODUCTION

Different application scenarios benefit from facial attributes analysis for the purposes of person verification and identification. Face detection and recognition have shown an incredible high accuracy in laboratory conditions and in all related scenarios where the user collaborates with the biometric recognition system (e.g., security control for accessing restricted areas like airports and train stations). On the other hand, once the user collaboration lacks or the user is not meant to be aware of being acquired by the sensors in the surrounding environment, this task becomes particularly challenging. This is typical of smart cities and sensitive places like banks and airports [1] and videosurveilled areas [2], but also applicable to all those contexts whose aim is granting the user's identity [3] like learning platforms [4] or smart devices applications [5], [6]. Whereas in collaborative scenarios the face variations like illumination and pose can be effectively eliminated or greatly reduced, even the detection of the face becomes challenging in unconstrained environments [7]. These adverse conditions affecting facial attribute analysis are also called PIE-issue, that are:

- **Occlusions**: induced by environmental or worn objects that make the face partially visible. Are examples of this issue the hats, eyeglasses and scarfs which tend to hide facial features;
- **Pose**: this is typically hard to address in uncontrolled scenarios and regards the alignment between the face and the acquiring sensor. Adverse poses can introduce significant distortion in face appearance thus distorting its attributes as well;
- **Illumination**: which is responsible for shadows and noise. Artificial illumination in particular creates the conditions for a difficult detection of the features;

The associate editor coordinating the review of this manuscript and approving it for publication was Zhen Qin.

- **Expressions**: which naturally alters the face morphology thus reducing the chances for a correct analysis. Moreover, even if expressions can be effectively classified, they cannot be always inverted to go back to neutral expression, which represents the ideal condition for biometric recognition.

When approaching to the difficulties induced by natural behaviour of human beings interacting with the smartcities, the biometric face recognition can keep benefit from a collection of soft biometrics, like the single facial features are. This is particularly true in those cases where the normal pose of a subject may not enable an exhaustive capturing of the whole face. In many cases, a pre-processing step consisting in face normalisation is exploited to go further with the facial analysis. Over the years, several approaches have been proposed in the literature for pose estimation and occlusion detection, most of them based on model of mixture of trees or landmarks estimation [8]–[10]. However, state-of-the-art approaches rely nowadays on the effective and robust training performed via convolutional neural networks (CNNs) [11]–[13]. Particularly significant are the achievements of Multi-Task Cascade Convolutional Neural Network (MTCNN) [14] for face alignment and detection, which demonstrated superior results compared to similar approaches in the literature. The approach here proposed also relies on CNNs that are indeed used as a facial feature extractor only (to extract features more robust to hand-made features like HOG, Haar, LBP). Such features are then used as input for a clustering method which in turns can be used to ease the successive biometric recogntion tasks based on faces.

The main contribution of this paper can be summarised as follows:

1) transfer learning has been used on a neural network model for facial attributes detection and estimation.
2) deep features have been used as input for clustering techniques to discover distinctive features for grouping people with common biometric traits.
3) main applications of the proposed approach can be found in the forensic scenario as a support to identikit recognition. In this direction, the witness can describe the look of a person and the method can output a set of potential similar subjects.

II. RELATED WORKS

Biometric recognition based on facial traits has been extensively explored over last years and significant robust solutions have been proposed in the literature. On the contrary, the unsupervised approaches dealing with clustering of facial features are notably less discussed. This section aims at providing a comprehensive presentation of related work in this field, starting from facial attribute detection and prediction methods and going through clustering techniques available for the proposed task. The section concludes with transfer learning approaches aimed at using the pre-trained deep neural network models nowadays available and customise it for the purpose of the task discussed in this work.

A. ATTRIBUTE PREDICTION

Attribute prediction in the field of face detection/recognition deals with the problem of inferring missing data starting from visible features of the user's face. The analysis of human's face, as well as the reconstruction of missing information, takes benefit from the symmetric proportion of the face features [15]. This means that even in case of partially occluded face, it is still possible to perform biometric processing with reliable results. In [11], the authors proposed a combination of two convolutional neural network models, that is LNet and ANet, specifically trained for face attribute prediction. However, the two models are differently pre-trained: the former is meant for face localisation and hence trained on massive categories of general objects, the second one specifically trained on facial attributes. Then, either are trained and jointly fine-tuned with attributes tags. Learning massive amount of facial features allows the trained ANet model to cope with many complex face variations (due to illumination and pose changes) thanks to the face features learnt at training time.

An alternative way to deal with this problem is learning the discriminative face representation. Researches in [16], [17], proposed a model to achieve a compact face representation by synthesising its attributes. It is called *face embedding*, that on a proper training phase it is able to take apart faces belonging to the same subject (identity). During the training process, the model implicitly learns enough features to distinguish face identities (the embeddings related to the same identity will lie on the same hyperplane). However the resulting face embedding is hard to interpret because it hides the learnt facial features, consequently losing any relations among attributes.Thus, the embedding-based approaches are extremely good at maximising the performance of a single feature; in this case the feature is the face identity. Indeed these are one of the most effective way of performing accurate face recognition tasks (which are based on a single high-level concept: identity).

B. CLUSTERING METHODS

Three clustering techniques have been considered in this study, that are K-means, Agglomerative Clustering and DBSCAN.

The well-known K-Means algorithm [18] clusters data by trying to separate samples in n groups of equal variance, minimising the inertia criterion (the average squared distance between points in the same cluster). A significant aspect of this algorithm is that it is based on *a-priori* assumption that is the number of clusters to be formed with input data. On the other hand, it scales up really well on large number of samples and for this reason it has been intensively used across a large range of application domains.

A variant of K-means, which overpasses the limitation of knowing in advance the number of clusters, known as ISODATA algorithm (Iterative Self-Organizing Data Analysis Technique Algorithm) [19] has been also considered. It consists in a clustering method using a self-organising

approach for data analysis and pattern recognition. Being self-organising, the number of clusters to form does not depend on a given parameter but it is automatically achieved by the clustering method on convergence. The convergence depends on the metrics chosen to assign a sample to a cluster rather than another.

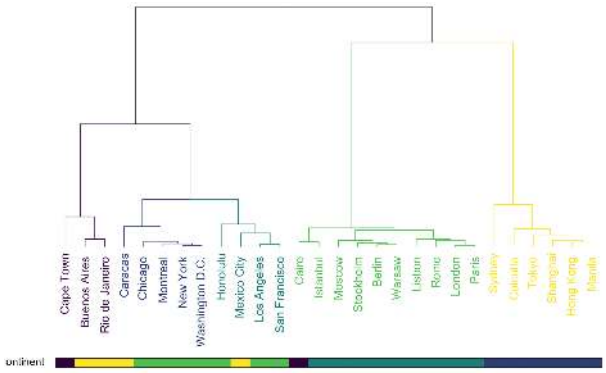

FIGURE 1. Typical hierarchical clustering depicted in form a dendrogram. The dendrogram allows to both realise the distance between single input samples than between clusters formed at different heights of the tree-structure.

The Agglomerative Clustering approach [20] falls within the Hierarchical clustering family. These methods work by building a tree-like structure called dendrogram (see figure 1. The benefit of such a hierarchy of clusters is that the grouping formed from the input samples can be both described in terms of samples themselves that in terms of difference between clusters.

Similarly to the ISODATA algorithm introduced above, the DBSCAN algorithm (Density-Based Spatial Clustering of Applications with Noise) [21] is able to discover clusters of arbitrary shape thus not requiring the number of clusters as input parameter. It is based on the concept density-connectedness, therefore it assures that all points belonging to a cluster are density-connected to each other and thus any point which is reachable from a point in a specific cluster belongs to the same cluster too. Moreover, DBSCAN is designed to require a minimal knowledge of the domain data, together with a good efficiency on large databases.

The problem of clustering millions of faces into thousands of clusters was proposed by Otto et al. [22]. The objective was of grouping faces so that the formed cluster would group similar identities. They proposed an approximate Rank-Order clustering algorithm that outperformed popular clustering algorithms, like k-Means and Spectral [23], both in terms of accuracy and runtime complexity. Other interesting evaluation was carried out on the work done by Rosebrock [24]. The author proposes an algorithm that extracts a 128-value array consisting in real numbers inferred from a face. In turn, these features array are used to create clusters. Each cluster contains a set of faces with similar facial features. As it can be expected, in case of DBSCAN or ISODATA algorithms the number of clusters depend on the input data and may differ from an algorithm to another, which are designed to return the optimal number of clusters withoyt a priori knowldege of the data distribution. Even in this case it does not fit our needs perfectly, as another goal that has been set is the possibility of independently choosing the number of clusters to be displayed.

C. TRANSFER LEARNING

The complexity of the Machine Learning tasks increases as the research goes further. Consequently, the model architectures become tend to become particularly big and both time and computing demanding. In turn, this implies the need for enormous processing power and longer training time duration. This point is particularly true for recent Convolutional Neural Networks models [25], which require a huge amount of data and computational power. Thanks to the ImageNet classification challenge [26], the submitted AlexNet model [25] marked a turning point in 2012 for deep learning in computer vision. The models that followed, like VGGNet [27], InceptionNet [28], and ResNet [29] are examples nowadays very used and useful as solver for a wide range of computer vision tasks and more. The success and the accuracy achieved by these models have assessed, over the years, the tendency of using them as feature extractors, rather than as a solution for classification or regression problems. The Transfer Learning [30] has so achieved huge consideration, for the benefit of using pre-trained models like off-shelf solutions which do not required to be trained from scratch. Thus, recycling a model trained for a specific task on a new similar task reduces significantly the overall training time to cope with the new problem. In this work, this technique is exploited to fine-tune the proposed model.

III. OUR APPROACH

More details about the proposed detection-clustering pipeline are provided in this section. The workflow can be mainly divided in the following three parts:

- **Inference step**: it consists in the preliminary step of fine-tuning the pre-trained model on 37 facial features taken from CelebA dataset [11].
- **Clustering step**: the prediction of the model above, i.e. the labels assigned to the given input, are in turn provided as input to a clustering algorithm (refer to section II-B for details on them). The clustering is meant to compute the grouping of closest input faces.
- **Analysis and visualisation step**: the occurrences of the attributes in a given cluster allow to quantitatively measure the accuracy and the significance of faces collected. By the analysis of close features in a cluster, we provide a compact and exemplary representation of the collected faces.

A. THE CELEBA DATASET

The experimentation has been conducted on the CelebA dataset [31], a large-scale face attributes dataset with more than 200K celebrity images each with 40 attribute annotations. It is a very challenging dataset (compared to similar one like LFW [32] or UTKFace [33] because of its amount

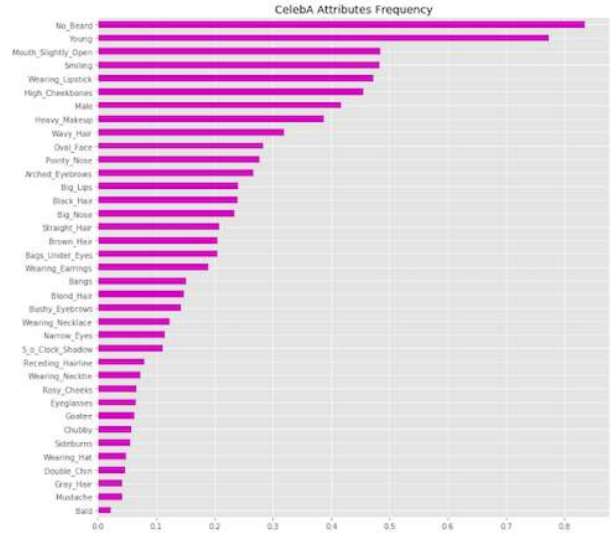

FIGURE 2. Frequencies of CelebA attributes: the chart clearly shows the discrepacy about the occurrence of the attributes.

of different identities but also for the large range of environmental and behavioural factors, like poses, age and gender, expressions, occlusion variations and so on. Moreover, by looking at the bar plot in figure 2 it can be observed that the CelebA dataset is heavily imbalanced in terms of its attributes. A third of attributes consists of extremely rare facial features (10% frequency or less), and only a couple of them are very common (by occurring in more than 70% of cases). The just mentioned imbalance negatively affects many loss functions in a significant way when adopted at training time.

B. THE FINE-TUNING OF THE MODEL

The baseline model explored in this study is MobileNetV2 [34], an efficient deep neural network model implementing depth-wise separable convolutions. First a depth-wise convolution layer filters the input that is it performs convolutions on image colour channels separately. This step is followed by a 1×1 (also called point-wise) convolution layer that combines the filtered values and creates new features. Compared to a regular convolution, where filtering and combining task occur at the same time, the depth-wise convolutions achieve higher running performance due to the separation of the two tasks that can then be implemented with parallel architectures. The choice falls on MobileNetV2 architecture after having considered competing and different model architectures, even with 10x more parameters (our model has only 4.3M of parameters). In terms of final accuracy, the performances equal each other but at the cost of a much slower training. As discussed in the following sections, the proposed model achieves 90.95% testing accuracy.

We implement the transfer learning approach by removing the top classification layers (see figure 3) and opportunely fine-tuning the weights by a fast training stage on training data.

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

FIGURE 3. MobileNetV2: Each line describes a sequence of 1 or more identical (modulo stride) layers, repeated *n* times.

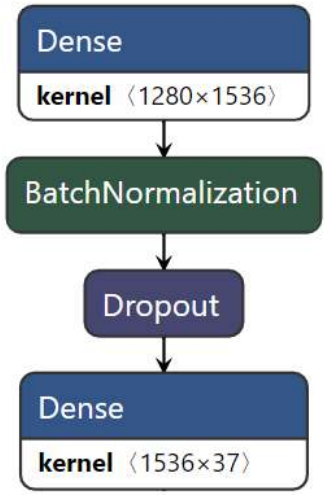

FIGURE 4. Top Layers: a first Fully-Connected (Dense) Layer with 1536 neurons, normalised by Batch-Normalisation and regularised by applying a dropout on 30% of the connections. The last Dense layer outputs the labels for every attribute (thus, requires 37 neurons *sigmoid*-activated).

Our top layers (figure 4), simply consists of a Fully Connected layer, followed by a Batch Normalisation operation before the final multi-label Dense layer that outputs the facial attributes of the input sample. Therefore, the output of the model consists in binary 37-*d* vector (we decided to drop three attributes: attractiveness, pale skin and blurry).

A technique for Data Augmentation has been involved during the experimentation to achieve a higher level of generalisation of the results. The augmentation mainly consists in introducing new samples starting from the available ones. In our case, the augmentations implemented can be summarised as follows:

- *Rotation*: a maximum of 20 degrees rotation is applied to image training data.
- *Shift*: consisting in a translation of image pixels by rows and cols, within a maximum 0.2 translation of the entire image width/height.

- *Shear*: a random distortion is applied with an effect of 0.2.
- *Zoom*: the magnification of image pixels is applied and limited to 20% of their total dimension.
- *Flipping*: considering the input data, i.e. the faces, only a horizontal flipping is applied to images. A vertical flipping would introduce upside down faces which could not contribute to effectively augment the training-set.

The data augmentation has the benefit of increasing the training data keeping a consistency with the original one. During the training, the model will see just slightly different samples in each epoch which still present possible adverse acquisition conditions. Moreover, any bias is introduced for using samples that would not be ever acquired in real conditions.

It is important to observe that common loss functions, e.g., mean absolute error (MAE) and mean squared error (MSE), notably fails during training on sparse binary multi-labelled data. When the position of a single bit is as much informative as the its presence the above mentioned loss functions does not work properly. To be more precise, different vectors with the same amount of 1s but in different positions (e.g. [0, 0, 1, 0] and [1, 0, 0, 0]) would produce the same response (the loss value) as for vectors that have the same amount of 1s but differently indexed (e.g., [0, 0, 1, 0] and [0, 1, 0, 0]). As a consequence for that, using these class of loss function leads to a *dumb model* during training: it predicts an array of 0's for whatever input instance regarding the belonging attributes. The implication of this observation is that the model performance drop down dramatically when trained with rare features like those characterising the CelebA dataset, which by definition creates a sparse representation of the features. For sake of clarity, the following example explains what has described above: let suppose the model needs to predict if the input face is of a young person or not and the features to discriminate this are available only for 5% of the samples. A fixed and blind prediction that assigns 0 to all samples results in an accuracy of 95% even if the model has not learned the feature at all.

This is the reason why relying just on accuracy may be misleading. Taking this into account, our model is double-checked with qualitative results and misprediction rate. Finally, since a good loss function should understand the difference between two samples, the cosine proximity has been adopted in this proposal. The cosine proximity formula is reported in equation 1

$$L = -\frac{y \cdot \hat{y}}{\|y\|_2 \cdot \|\hat{y}\|_2} = -\frac{\sum_{i=1}^{n} y^{(i)} \cdot \hat{y}^{(i)}}{\sqrt{\sum_{i=1}^{n} (y^{(i)})^2} \cdot \sqrt{\sum_{i=1}^{n} (\hat{y}^{(i)})^2}} \quad (1)$$

where $y = (y^{(1)}, y^{(2)}, \ldots, y^{(n)}) \in \mathbf{R}^n$ and $\hat{y} = (\hat{y}^{(1)}, \hat{y}^{(2)}, \ldots, \hat{y}^{(n)}) \in \mathbf{R}^n$ represent two samples. During training process, $\hat{y}$ is the ground truth label and y is the predicted label. This function treats a binary array as a vector in a multidimensional space. The similarities between true and predicted labels are computed in terms of the *arcos* of the angle formed by the two vectors. When two vectors are orthogonal (there is a 90° angle between them), means that they are completely different (the loss value is at its maximum). Instead, when two vectors overlap (the angle is 0°), they result the same (the loss value is at its minimum).

In 1) the facial attributes recognition results are shown. The achieved results have been compared to the model proposed by Yang et. al [31] consisting in the combination of LNet and ANet, discussed at section II-A.

IV. EXPERIMENTAL RESULTS

As preliminary discussed in section III-A, the CelebA dataset has been split in three partitions as the dataset's authors suggest: (i)training set, (ii) validation set and (iii) testing set. During training stage, the entire training set has been provided ($160k$ samples resized to 224×224 and augmented according to the augmentation stragegy mentioned in previous section III-B). The validation occurred at the end of each epoch on a total number of $20k$ samples, constituting the validation set. The batch size has been set at 64 images due to resource limitation. On the other hand, light computation demand aside, having small batch size helps to reduce the generalisation error of the network model [38]. The AdaDelta optimiser [39] has been chosen as rule for cosine proximity loss minimisation. This choice depends on the fact that the AdaDelta optimiser requires less hyperparameter-tuning compared to other optimiser but, more importantly, converges in many conditions more rapidly to sub-optimal solutions compared to others (like the traditional Stochastic Gradient Descent or Momentum or Adam optimisers, to mention a few of them). Moreover, AdaDelta features some interesting properties, that the the following:

- it adapts the learning rate by a moving window of gradient updates. This prevents from stopping the learning process after many iterations.
- in turn, it features a high convergence speed. The adaptive learning allows to accelerate the direction of learning (i.e. by increasing the learning rate) when the results is promising and to slow down when the loss does not improve over successive iterations.

A. CLUSTERING

The way the clustering techniques have been exploited represents one of the main contribution of this work. In digital forensics is very useful to match the identity of a subject with potential similar users rather than looking for a strict and highly robust identification (which in many conditions might be impossible to achieve or unfeasible). The idea here presented is that of grouping faces according to the visible facial features so that building up an *identikit* of the subjects of interest. In unconstrained environment, where occlusions may occur with high probability, the association of an identity to partially visible faces represents a big advantage in many sensitive areas. The cluster synthesis, by which the dimensional space of representation of the features is reduced by preserving distinctive facial attributes only, makes possible

to simplify the potential identikit recognition, very useful in forensics.

The steps of the proposed clustering approach for grouping facial attributes consists of the following steps:

- *Number of cluster*: Choosing the number of clusters (no fixed values);
- *Dimensional reduction*: removal of non discriminating attributes by selecting a subset of facial features;
- *Visualisation*: graphical representation of the clusters achieved from the input population;
- *Synthesising*: for each cluster, turn the facial attributes into a reliable and realistic face, also known as *eigenface*.

An example of the output of the approach is shown in figure 6, in which, for each cluster (4, in the example), the population of the cluster is shown (the first element), the representative element of the cluster, obtaining by applying *eigenface* to the cluster (the second element), and the occurrence of each attribute within the cluster components (the third element).

According to these requirements, the *silhouette score* of different clustering algorithm has been computed as a measure to estimate the quality and significance of the groups of faces formed. Graphically, a plot of the occurrence of the attributes in each cluster is provided as well as reconstruct the eigenface from the synthesis of facial attributes in a cluster. This helps to better fix the faces gathered from the clustering algorithm in each cluster.

In order to get a fair comparison among the clustering methods considered in this study, the optimal number of clusters discovered by DBSCAN has been applied to all others that need for such a parameter be set a priori.

More in details, the K-Means algorithm divides a set of n samples $x_1, x_2, \ldots, x_n$ into K disjointed clusters $C_1, C_2, \ldots, C_K$, each described by the mean $\mu_j, j = 1, \ldots, K$ of the samples in the cluster. The computed means represent the *centroid* of the cluster itself, which not necessarily matches with one sample of the input population. The K-Means algorithm aims at choosing centroids which minimise the inertia, or within-cluster sum-of-squares criterion:

$$\sum_{i=0}^{n} \min_{\mu_j \in C_j} (||x_i - \mu_j||^2) \tag{2}$$

In particular, as regards Agglomerative Clustering, various input parameters have been analysed in order to optimise the performances achieved. In particular, the list below summarise the linkage criteria and the metric adopted:

- **Single linkage**, consisting in the minimisation of the distance between the closest observations of pairs of clusters.
- **Maximum or complete linkage**, consisting in the minimisation of the maximum distance between observations of pairs of clusters.
- **Average linkage**, consisting in the minimisation of the average of the distances between all observations of pairs of clusters.
- **Ward**, consisting in the minimisation of the sum of squared differences within all clusters. This makes the Agglomerative Clustering similar to the K-Means objective function but exploiting an agglomerative hierarchical approach.

Also, the linkage criteria determines the metric used for merging strategy. Here have been considered the following distances: Euclidean, $\ell 1$, $\ell 2$, Manhattan and Cosine. When the linkage is set to *ward*, the Euclidean distance is solely accepted.

The performances of both K-Means and Agglomerative Clustering on a variable number of clusters has been compared with those of DBSCAN (figure 5). The best combination of metric and linkage criteria resulted from the experimental evaluation is complete linkage and Manhattan distance. On the other side, however, the best achieved result in agglomerative clustering resulted inferior inferior to that of K-Means. In fact, in similar and comparable conditions the Agglomerative Clustering achieves a silhouette score of 0.73, which is lower the value of 0.83 obtained with K-Means clustering. Despite the better performances of DBSCAN, with a silhouette score of 0.87, we have decided to use K-means, since, as can be evaluated from the charts, it proves to be the best among the algorithms that allow you to choose the number of clusters.

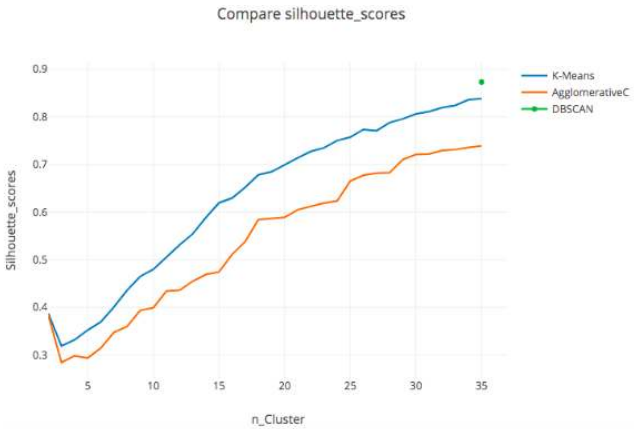

FIGURE 5. Comparison of clustering algorithms: it can be observed the superior results achieved by K-Means over Agglomerative Clustering.

On the output of the clustering process, the significance of the clusters obtained has been analysed both in terms of features and the corresponding eigenfaces. Hand-checking the facial attributes in a given cluster is an error-prone process, not mentioning the cost of doing such an inspecting analysis with a large amount of faces. For these reasons, the eigenfaces are computed (by standard dimensionality reduction methods, like PCA [40]) to get a meaningful and representative visual description of each cluster.

Regarding this objective of achieving a visual representation of the clusters, two simple strategies have been considered:

TABLE 1. Performance comparison of attribute prediction models.

	5 Shadow	Arched Eyebrows	Bags Under Eyes	Bald	Bangs	Big Lips	Big Nose	Black Hair	Blond Hair	Brown Hair	Bushy Eyebrows	Chubby	Double Chin	Eyeglasses	Goatee	Gray Hair	Heavy Makeup	High Cheekbones	Male
Our approach	95	81	85	99	96	71	84	90	96	89	92	96	96	99	97	98	91	87	98
[35]	77	83	83	91	91	78	83	91	97	76	83	75	80	91	83	87	95	88	94
[36]	94	83	85	99	96	71	84	90	96	89	93	96	96	99	97	98	91	87	98
[37]	91	79	79	98	95	68	78	88	95	80	90	91	92	99	95	97	90	87	88
[31]	91	79	79	98	95	68	78	88	95	80	90	91	92	99	95	97	90	87	98

	Mouth Slightly open	Mustache	Narrow Eyes	No Beard	Oval Face	Pointy Nose	Receding Hairline	Rosy Checks	Sideburns	Smiling	Straight Hair	Wavy Hair	Wearing Earrings	Wearing Hat	Wearing Lipstick	Wearing Necklace	Wearing Necktie	Young	**Average**
Our approach	94	97	87	96	76	76	93	95	98	93	83	82	90	99	91	88	97	88	**91**
[35]	81	94	81	80	75	83	86	82	82	90	77	77	95	90	95	90	81	86	**87**
[36]	93	97	87	96	77	94	96	97	92	83	84	90	99	94	86	96	88	91	**90**
[37]	92	95	81	95	66	72	89	90	96	92	73	80	82	99	93	71	93	87	**87**
[31]	92	95	81	95	66	72	89	90	96	92	73	80	82	99	93	71	93	87	**87**

FIGURE 6. Clusters with weights. The attributes selected are "Eyeglasses", "Male", "Bald" and "Young".

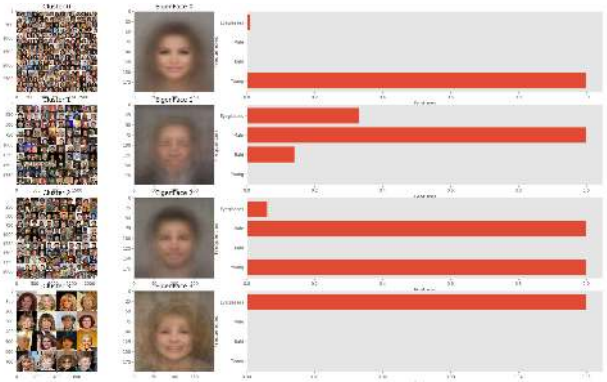

FIGURE 7. Clusters without weights. The attributes selected are "Eyeglasses", "Male", "Bald" and "Young.

1) By using a chart like the one in figure 6, it is possible to determine the occurrences of the facial attributes within a given cluster and thus inferring the frequency of each attributes in it. By observing with attention, it is possible to separate the noisy attributes (those with low occurrence) from the prominent ones (those with a very high occurrence).
2) An Eigenface represents the total amount of faces in a cluster in a single reliable exemplary face.From a given cluster, the corresponding eigenface is obtained by a traditional implementation of Principal Component Analysis (PCA) [41] by flattening the images belonging to that cluster. Being obtained by a dimensionality reduction techniques, the generated face includes only the prominent features of the cluster. In other terms, this allows to determine are the relevant facial attributes constituting the cluster.

To further improve the achieved results, a weighting criterion has been adopted. It consists in assigning a weight to the most frequent, or alternatively the less frequent, facial features. The impact of such a weighted approach is shown in figures 6 and 7. It can be observed that this trick can drastically reduce the quantity of partial features within the cluster.

V. CONCLUSION

Biometric recognition is affected by several critical factors in unconstrained scenarios, which make it a challenging practice to address in an efficient and effective way. Lighting changes, pose variations and occlusions, do not permit to achieve accurate results in terms of detection and, hence, the consequent biometric recognition results in a challenging task. Moreover, due to the high variability of factors affecting the existing solutions in the literature, a totally precise and

accurate recognition is not possible without a manual intervention where an operators assists the association between a given identity and the data acquired by the sensors, it might be the face of a person, his/her eyes, fingerprint, body shape and so on. From this point of view, in this work it has been proposed a framework for facial features recognition by means of MobileNet-like convolutional neural network, and face clustering based on K-means and Eigenface as a mean for graphically describe the clusters. Two interesting findings are highlighted by the conducted experiments: the former is related to the improvement of the results in the state of the art like shown in Table 1. The latter, instead reveals the possibility of addressing the clustering process by means of the weights learnt by a CNN. Also, the encouraging results show that the clustering based on facial features can provide useful insights to build up an identikit of a subject, especially useful in digital forensics when data are partially available or corrupted. The recognition accuracy is higher than similar neural networks in the literature proposed for this task. Qualitative and quantitative results have been proposed, also showing the advantages of the K-means with respect to DBSCAN and Agglomerative Clustering techniques.

ACKNOWLEDGMENT

The authors would like to thank the students Luca Anzalone, Marialuisa Trere and Simone Faiella for having conducted the experiments and proposed the model.

REFERENCES

[1] S. Barra, A. Castiglione, F. Narducci, M. De Marsico, and M. Nappi, "Biometric data on the edge for secure, smart and user tailored access to cloud services," *Future Gener. Comput. Syst.*, vol. 101, pp. 534–541, Dec. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X19303188

[2] J. C. Neves, G. Santos, S. Filipe, E. Grancho, S. Barra, F. Narducci, and H. Proença, "Quis-Campi: Extending in the wild biometric recognition to surveillance environments," in *New Trends in Image Analysis and Processing*, V. Murino, E. Puppo, D. Sona, M. Cristani, and C. Sansone, Eds. Cham, Switzerland: Springer, 2015, pp. 59–68.

[3] S. Barra, M. De Marsico, C. Galdi, D. Riccio, and H. Wechsler, "FAME: Face Authentication for Mobile Encounter," in *Proc. IEEE Workshop Biometric Meas. Syst. Secur. Med. Appl.*, Sep. 2013, pp. 1–7.

[4] G. Fenu, M. Marras, and M. Meles, "A learning analytics tool for usability assessment in moodle environments," *J. e-Learn. Knowl. Soc.*, vol. 13, no. 3, pp. 23–34, 2017.

[5] G. Fenu and M. Marras, "Leveraging continuous multi-modal authentication for access control in mobile cloud environments," in *New Trends in Image Analysis and Processing—ICIAP 2017*, S. Battiato, G. M. Farinella, M. Leo, and G. Gallo, Eds. Cham, Switzerland: Springer, 2017, pp. 331–342.

[6] G. Fenu and M. Marras, "Controlling user access to cloud-connected mobile applications by means of biometrics," *IEEE Cloud Computing*, vol. 5, no. 4, pp. 47–57, Jul./Aug. 2018.

[7] H. Proença, J. C. Neves, S. Barra, T. Marques, and J. C. Moreno, "Joint head pose/soft label estimation for human recognition in-the-wild," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 12, pp. 2444–2456, Dec. 2016.

[8] X. Zhu and D. Ramann, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2879–2886.

[9] P. Barra, C. Bisogni, M. Nappi, and S. Ricciardi, "Fast quadtree-based pose estimation for security applications using face biometrics," in *Network and System Security*, M. H. Au, S. M. Yiu, J. Li, X. Luo, C. Wang, A. Castiglione, and K. Kluczniak, Eds. Cham, Switzerland: Springer, 2018, pp. 160–173.

[10] A. F. Abate, P. Barra, C. Bisogni, M. Nappi, and S. Ricciardi, "Near real–time three axis head pose estimation without training," *IEEE Access*, vol. 7, pp. 64256–64265, 2019.

[11] Z. Liu, P. Luo, X. Wang, and X. Tang, "Large-scale celebfaces attributes (celeba) dataset," in *Multimedia Laboratory*, Hong Kong: Chinese Univ., 2015.

[12] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: A deep multi–task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2019.

[13] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

[14] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5325–5334.

[15] G. Passalis, P. Perakis, T. Theoharis, and I. A. Kakadiaris, "Using facial symmetry to handle pose variations in real–world 3D face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 1938–1951, Oct. 2011.

[16] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. 2015 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015.

[17] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, *Sphereface: Deep Hypersphere Embedding for Face Recognition*. Atlanta, Georgia: Georgia Institute of Technology, 2017.

[18] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable k-means++," *Proc. VLDB Endowment*, vol. 5, no. 7, pp. 622–633, 2012.

[19] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vols. PAMI–1, no. 2, pp. 224–227, Apr. 1979.

[20] I. Davidson and S. Ravi, "Agglomerative hierarchical clustering with constraints: Theoretical and empirical results," in *Proc. Eur. Conf. Princ. Data Mining Knowl. Discovery*. Berlin, Germany: Springer, 2005, pp. 59–70.

[21] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "Density-based spatial clustering of applications with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, vol. 240, 1996, p. 6.

[22] C. Otto, D. Wang, and A. K. Jain, "Clustering millions of faces by identity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 289–303, Feb. 2018.

[23] H. Jia, S. Ding, X. Xu, and R. Nie, "The latest research progress on spectral clustering," *Neural Comput. Appl.*, vol. 24, nos. 7–8, pp. 1477–1486, Jun. 2014.

[24] A. Rosebrock. (2018). *Face Clustering With Python*. [Online]. Available: https://www.pyimagesearch.com/2018/07/09/face-clustering-with-python/

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Imagenet Classification With Deep Convolutional Neural Networks*. Toronto, ON, Canada: Univ. Toronto, 2012.

[26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016.

[30] *S231n Convolutional Neural Networks for Visual Recognition*. [Online]. Available: http://cs231n.github.io/transfer-learning#tf

[31] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015.
[32] 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," Univ. Massachusetts Amherst, Amherst, MA, USA, Oct. 2007.
[33] Z. Zhang, Y. Song, and H. Qi, "Age progression/regression by conditional adversarial autoencoder," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017.
[34] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018.
[35] R. Rothe, R. Timofte, and L. V. Gool, "DEX: Deep expectation of apparent age from a single image," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 10–15.
[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[37] A. H. Marblestone, G. Wayne, and K. P. Kording, "Toward an integration of deep learning and neuroscience," *Frontiers Comput. Neurosci.*, vol. 10, p. 94, Sep. 2016.
[38] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural Networks: Tricks Trade*. Berlin, Germany: Springer, 2012, pp. 437–478.
[39] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701
[40] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[41] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, nos. 1–3, pp. 37–52, 1987.