Face Recognition Using Smoothed High-Dimensional Representation

Juha Ylioinas, Juho Kannala, Abdenour Hadid, and Matti Pietikäinen

Center for Machine Vision Research, PO Box 4500,
FI-90014 University of Oulu, Finland

Abstract. Recent studies have underlined the significance of high-dimensional features and their compression for face recognition. Partly motivated by these findings, we propose a novel method for building unsupervised face representations based on binarized descriptors and efficient compression by soft assignment and unsupervised dimensionality reduction. For binarized descriptors, we consider Binarized Statistical Image Features (BSIF) which is a learning based descriptor computing a binary code for each pixel by thresholding the outputs of a linear projection between a local image patch and a set of independent basis vectors estimated from a training data set using independent component analysis. In this work, we propose application specific learning to train a separate BSIF descriptor for each of the local face regions. Then, our method constructs a high-dimensional representation from an input face by collecting histograms of BSIF codes in a blockwise manner. Before dropping the dimension to get a more compressed representation, an important step in the pipeline of our method is soft feature assignment where the region histograms of the binarized codes are smoothed using kernel density estimation achieved by a simple and fast matrix-vector product. In detail, we provide a thorough evaluation on FERET and LFW benchmarks comparing our face representation method to the state-of-the-art in face recognition showing enhanced performance on FERET and promising results on LFW.

1 Introduction

Automatic face recognition from images is a major research area in computer vision. The high societal impact and practical significance of face recognition technologies is evident given the ever-increasing digital image databases and wide availability of cameras in various consumer devices (e.g. smart phones, Google Glasses). Thus, face recognition has many applications in several areas, including, for example, content-based image retrieval, security and surveillance (e.g. passport control), web search and services (e.g. automatic face naming in services like Facebook) and human computer interaction.

The first studies on automatic face recognition emerged already in 1970's and since then various methods have been developed resulting in continuous improvements in performance. Examples of well known early techniques include the *Eigenfaces* and *Fisherfaces* methods [1, 2]. A comprehensive review of the field from its early days until the beginning of 2000's is presented in [3]. However, despite decades of research and all the developments and efforts, there is still a clear gap in accuracy and robustness between the automatic face recognition systems and human level of performance. In

fact, the problem of automatic face recognition is still a very active research topic and there are plenty of recent developments [4–13]. Important driving forces behind the recent progress are public datasets and benchmarks that are used for comparing and evaluating different methods. Examples of well known and widely used benchmarks are the Facial Recognition Technology (FERET) database [14] and the Labeled Faces in the Wild (LFW) database [15].

A typical face recognition system consists of detection, alignment, representation, and classification steps. In this paper, along with many other recent studies, our focus is on face representation. Usually, face representation is composed of two distinct steps where (i) a certain kind of face representation is first generated from a normalized input face image and then (ii) subspace analysis is performed to produce a significantly lower dimensional representation [16]. The step (i) can be performed by common signal processing techniques, such as Gabor wavelets and Discrete Fourier Transform, whereas the step (ii) by applying Principal Component Analysis (PCA) or Linear Discriminant Analysis (LDA). The way the input face is processed divides the face representation methods further into holistic and so called local methods. Especially, the use of methods based on local image descriptors has resulted in a great success, the notable ones including gradient based Scale Invariant Feature Transform (SIFT), Histograms of Oriented Gradients (HOG), and Local Binary Patterns (LBP). Especially binarized local image descriptors, like LBP, have gained a great favour in a wide spectrum of face analysis studies. An essential part of local face description methods is the use of statistical histograms over a discrete vocabulary of the resulting local descriptor features.

In this paper, we propose a novel *unsupervised* face representation method which builds on the recently proposed local image description method called Binarized Statistical Image Features (BSIF) [17]. BSIF is a learning based method which computes a binary code for each pixel by thresholding the inner products between a vectorized local image patch outputs of a linear projection between a local image patch and a set of basis vectors which are learnt via Independent Component Analysis (ICA) from training image patches. The BSIF method is inspired by LBP and its derivatives, but in contrast to many of them, BSIF does not use a manually predefined set of filters but learns them by utilizing the statistics of images under interest. In particular, in this paper we show that it is beneficial to learn separate sets of linear filters for different face regions by utilizing training image patches from face images. Thus, the binary codes for pixels of a certain face region are obtained by using the corresponding filters, specifically learnt to describe patches in that region. This is in contrast to [17] which uses the same filters for all pixels and learns them from natural images (but not specifically from face images). The face regions are finally represented by histograms of the resulting binary codes and the final face descriptor is obtained by concatenating the histograms of different regions, as in [18] and [17]. As an important step of our representation, before compressing the histogram based face representation to a lower dimensional subspace using Whitening PCA (WPCA), we propose to smooth each region histogram using a kernel method suitable for n-dimensional binary data. Most importantly, we show that the smoothing can be accomplished by a simple matrix-vector product. Unlike other recent learning based descriptors, our approach does not need hand-crafted local pixel patterns and vector quantization, for example, using *k-means* during learning and, most

importantly, it avoids cumbersome large look-up tables at test time. Our contributions include: (i) we advance descriptor based methods; (ii) we provide insights to benefits of unsupervised application specific descriptor learning; (iii) we introduce a practical method for local descriptor soft-assigments; and (iv) we show the importance of soft-assigment as a predecessor for dimensionality reduction.

Based on our evaluations, by using the face representation method proposed herein one is able to gain the state-of-the-art performance on the widely used FERET dataset. We also validate our proposal on more challenging conditions using the popular LFW dataset showing promising results.

2 Related Work

There is a lot of previous research on different aspects of automatic face recognition systems [19]. Some studies focus on the first stages of the pipeline, i.e. face detection and alignment [20], whereas others focus on learning classifiers or similarity metrics for chosen face representations [10]. However, this paper concentrates on the representation problem as it has been shown to be a crucial component for robust performance in challenging real-world scenarios [15, 13]. We next review some recent works that we consider to be the most closely related to our work.

High-dimensional features have been found very potential in designing representations in object recognition. For example in [21], it was shown that together with the chosen feature itself, equally important is to consider how many of them to use and how dense or in how many scales they are extracted. Indeed, these are the key elements that are acknowledged in many studies for obtaining an informative representation. The only setback is that a method which embodies all of these elements usually outputs a very high-dimensional representation. To compress the representation for a more practical usage, efficient dimension reduction methods are needed as noted in [7].

In [7], Chen et al. discussed about benefits of a high-dimensional face representation and practically showed that the increase in dimensionality has a positive impact on the accuracy while applied to face verification. Their method was based on face landmark detection following encodings of the detected keypoints (such as eyes, nose, and mouth corners). They compared several local descriptors which all ended up to improved recognition accuracy while the feature dimension was increased by varying landmark numbers and sampling scales. In [11], a high-dimensional representation was constructed concatenating LBP histograms computed from the whole face area using overlapping blocks and different kinds of LBP parameter configurations. However, the authors argued that the added accuracy in high-dimensional face representation can be revealed only after dimensionality reduction, which they showed using whitening PCA among other methods.

Although the claimed pivotal role of high-dimensionality, there are some clear hints that the underlying feature extraction method has its own impact and should be taken into careful consideration. To this end, learning encoders for hand-crafted descriptors has lately been shown to yield outperforming results compared to completely hand-crafted ones. For example, the best performing descriptor in [22], an LBP-based descriptor combined with unsupervised codebook learning via *k-means* outperformed such descriptors as the conventional LBP and HOG, in all given settings. In turn, a quite

recent supervised descriptor, proposed in [8], is based on first fixing the LBP-like sampling strategy and then learning discriminative filters and so-called soft-sampling using a formulation similar to two-dimensional Linear Discriminant Analysis. Finally, the method was shown to outperform many of the existing LBP-based, completely hand-crafted descriptors.

Our approach connects all aspects of the methods discussed above for producing a discriminative face representation. The desired high-dimensionality [7] is reached at the descriptor level, using a local binary descriptor called BSIF [17]. In our method, we basically learn descriptors that produce higher dimensional histograms but, unlike LBP, in a more justified manner without sacrificing further loss of information during the encoding of pixel neighborhoods. Like in [8], we learn the descriptors in a blockwise manner from aligned face images, but apart from that, our approach does not need hand-crafted local pixel patterns and vector quantization using large look-up tables. We use overlapping blocks, like in [11], but before compression we further propose to smooth each region based histogram using kernel density estimation. Finally, an efficient compression is achieved by projecting the whole representation into a lower dimension using the whitening PCA method.

3 Our Method

In this section, we review the most important steps of our face representation method. We first introduce the BSIF descriptor and then present the utilized soft-assignment method suitable for binary descriptors. Finally, we introduce the WPCA dimensionality reduction method and provide some discussion about the used face matching methods together with other related details.

Binarized Statistical Image Features. Binarized Statistical Image Features (BSIF) is a data-driven local image description method which is widely inspired by LBP and its derivatives. In BSIF, a predefined number of linear filters are learnt using a set of training image patches using a criterion which aims to maximize the statistical independence between the responses of the convolutions of each individual filter and the given image patches [17]. Evidently, by maximising the statistical independence one is able to learn the most optimal set of filters with respect to the following independent quantization of the response vector coordinates, which is the fundamental part of all local binary descriptor methods. Moreover, the maximization of the statistical independence results in entropy growth between the coordinates leading to an effective description process in overall. This is also the main difference to the LBP method where the derivative pixel neighborhood tests are usually set without taking any criterion into consideration. This discussion above should justify the reason why we call BSIF as an optimal binarized descriptor. If a binarized descriptor is used to produce a high-dimensional representation, it is highly important to take the full advantage of the descriptor's encoding capability.

One BSIF operation is a linear matrix vector product, $\mathbf{s} = \mathbf{W}\mathbf{x}$, where $\mathbf{x}$ is a vector containing all the pixels of a local image patch of a size $w \times w$ (i.e. $\mathbf{x} \in \mathbb{R}^{w \times w}$), and $\mathbf{W}$ is a matrix of a size $n \times w^2$ containing the n linear filters which are stacked row by row. The output, vector $\mathbf{s} \in \mathbb{R}^n$, is then binarized by thresholding each of its elements s_i at zero finally yielding an n-bit long binary string treated as a codeword characterising the contents of the local neighborhood area on a certain location in the image.

For learning the linear filters, one needs to sample a training set consisting of image patches of the same size than the window of the desired descriptor. In the original paper [17], the training set was sampled from natural images, but the images can also be sampled from application-specific images, like it is done in this study. In the very beginning, the mean luminance is removed from each patch. Then, the linear filters are learnt so that the matrix $\mathbf{W}$ is first decomposed into two parts by $\mathbf{W} = \mathbf{UV}$, where $\mathbf{V}$ is a whitening transformation matrix learnt from the same training image patches and $\mathbf{U}$ is then finally estimated using Independent Component Analysis (ICA). The whitening transformation, usually accomplished via PCA, may also contain the reduction of dimensionality which in general lightens our computations but also reduces the effects caused by different image artefacts in image patches which are usually recorded by the last principal components. Here, we reduce the dimension of our training vectors to the length equal to the desired number of filters. Finally, to accomplish ICA we applied FastICA [23].

Soft-assigned BSIF descriptors. Originally, the idea of descriptor-space soft assignment was to tackle the problems caused by hard assignment of descriptors to discrete visual codewords. In this procedure, also known as the bag-of-visual-words representation, two image feature descriptors are treated identical if they are assigned to the same visual codeword of the visual vocabulary generated by some clustering algorithm, such as *k-means*. As noted in [24], such a hard quantization leads to errors as even a small variation in the feature value may cause totally different assignments. In soft-assignment the objective is to describe an image patch by a weighted combination of visual words. In general, soft-assignment has been investigated in both with using visual vocabularies generated by some clustering algorithm [24] and with binarized local descriptors [25]. The soft-assignment method we are using is based on kernel density estimation. The normal kernel, proposed by [26], is given by

$$K_\lambda(l|l') = \lambda^{n-d(l,l')}(1-\lambda)^{d(l,l')}, \tag{1}$$

where l and l' are both n-dimensional binary codewords, $d(l,l')$ is the Hamming distance between the codewords, and $\lambda \in [\frac{1}{2}, 1)$ is the bandwidth (smoothing) parameter.

The smoothing operation is put into action by first constructing a kernel matrix $\mathbf{S}$ so that

$$\mathbf{S}_\lambda = \begin{bmatrix} K_\lambda(0|0) & \dots & K_\lambda(0|2^n-1) \\ \vdots & \ddots & \vdots \\ K_\lambda(2^n-1|0) & \dots & K_\lambda(2^n-1|2^n-1) \end{bmatrix}. \tag{2}$$

Descriptor space soft-assignment is then accomplished by introducing a matrix-vector product $\mathbf{S}_\lambda\mathbf{h} \in \mathbb{R}_+^{2^n}$ where $\mathbf{h}$ is a histogram (in column format) of binary codewords on a certain image area and n is the number of filters.

The amount of weighting among the codewords is controlled by the smoothing parameter λ. Letting $\lambda = 1$ coincides with the naive estimator, i.e. the basic histogram of codewords. In that case the kernel matrix $\mathbf{S}_\lambda$ equals the identity matrix. On the contrary, by setting $\lambda = 1/2$ all codewords are given the same weight 2^{-n} which finally yields to evenly distributed codewords. It is noteworthy that the kernel in (1) is analogous with

the well-known Gaussian kernel that operates in continuous domain. Although soft assignment with binarized descriptors is quite well-known, we show its efficiency while combined with dimensionality reduction which, to the best of our knowledge, was not considered in previous studies.

Whitening PCA. To compress the high-dimensional representation we use Whitening Principal Component Analysis (WPCA). WPCA has proven to provide extra boost to the face recognition performance in many studies [27, 6, 8, 11]. The first benefit of using WPCA is the resulting reduced dimension of the final representation. In our algorithm, for example, it turns out useful as the length of the descriptor histogram is 2^n, where n is the number of filters. If the input face is divided into 7×7 blocks the final representation yields $2^n \times 7 \times 7$ which can finally prove too large in certain circumstances. The second benefit comes from the whitening part where the features projected along the principal components are divided by their standard deviations. It has been shown that the whitening part is important in order to equalize the influence of the principal components to the matching process which is often performed using the Cosine similarity. The PCA part is accomplished using the Turk-Pentland strategy where instead of calculating the covariance matrix $\mathbf{A}\mathbf{A}^\mathbf{T}$, where $\mathbf{A} = [\mathbf{a}_1\mathbf{a}_2 \dots \mathbf{a}_M]$ collects mean-subracted feature vectors in column format, we calculate $\mathbf{A}^\mathbf{T}\mathbf{A}$ which is a matrix of a much smaller size. The eigenvectors of $\mathbf{A}\mathbf{A}^\mathbf{T}$ are then $\mathbf{u}_i = \mathbf{A}\mathbf{v}_i / \left\|\mathbf{A}\mathbf{v}_i\right\|_2$, where $\mathbf{v}_i$ are the eigenvectors of $\mathbf{A}^\mathbf{T}\mathbf{A}$ [1].

In general, PCA may suffer with sparse and high-dimensional data leading to the overfitting problem [28]. One reason of this problem in our case is that while using BSIF with increasing number of filters the resulting histograms of descriptor labels will become larger and sparser since the number of descriptor label occurrences is always constrained according to the block size. The result is that applying PCA most likely overfits as the correlation between the possible pairs of coordinates is most probably represented by only a few samples in the data. Based on our results, it seems that the overfitting problem can be most likely alleviated by introducing the smoothing operation which ensures that there are much less non-zero elements in the concatenated representation than in the non-smoothed one.

Matching faces. For matching faces, we used the Hellinger distance. According to [29], this distance can be calculated as

$$d(\mathbf{x}, \mathbf{y})^2 = \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 - 2\mathbf{x}^\top\mathbf{y} = 2 - 2\mathbf{x}^\top\mathbf{y}, \tag{3}$$

where both $\mathbf{x}$ and $\mathbf{y}$ are properly preprocessed face representations. If we used the $L2$ normalized representations of $\mathbf{x}$ and $\mathbf{y}$, we would be measuring the Euclidean distance between them. To measure the distance based on the Hellinger kernel, before applying (3), we first (i) $L1$-normalize both representations and then (ii) replace all coordinate values by their square roots (see the detailed reasoning in [29]). Following the majority of previous face recognition studies, for the WPCA projected representation we use the Cosine distance. To accomplish this, we further $L2$-normalize the WPCA compressed representation and apply (3), which can be easily shown to be equal as measuring the Cosine distance between the input representations.

In our FERET evaluation, we straightly calculate the distance between two input representations using the steps given above. However, in the LFW evaluation, we use an additional step which has been used in some recent studies to gain some additional boost in performance. In detail, we use the *flip-free* strategy described in [30]. That means, instead of direct distance calculations of two input representations we horizontally flip all images before feature extraction and calculate the average of the distances between all possible four combinations of the representations stemming from the original and horizontally flipped images.

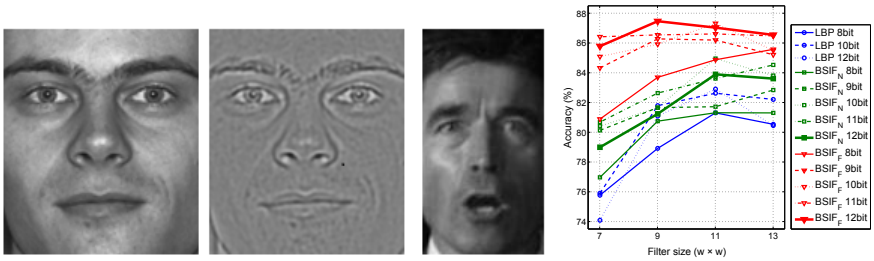
4 Experiments

We use the Face Recognition Technology (FERET) [14] and the Labeled Faces in the Wild (LFW) [15] datasets. To better understand the possible benefits of using application-specific images in learning the descriptor for high-dimensional representation, we compare the face-based BSIF descriptor to the one which was learnt using natural images. Our baseline is the popular LBP descriptor with different bit lengths and several different radii. Finally, we compare our proposal to the state-of-the-art methods which were reported using the given two datasets. For LFW experiments, we evaluate our method barely in the unsupervised evaluation category using the recently updated protocol [15].

Setup. FERET [14] is a standard dataset for benchmarking face recognition methods in constrained imaging conditions. FERET is composed of several different subsets with varying pose, expression, and illumination. We are interested in the frontal profile images of it, which are divided into five sets known as *fa*, *fb*, *fc*, *dup1*, and *dup2*. For gallery, we use *fa* containing 1,196 images of 1,196 subjects. For probes, we use the rest four subsets, where *fb* contains 1,195 images covering varied expressions, *fc* contains 194 images with varied illumination conditions, *dup1* contains 722 images taken later in time, and *dup2*, which contains 234 images taken at least one year after the corresponding gallery images. LFW [15], regarded as *de-facto* evaluation benchmark for face verification in unconstrained conditions, consists of 13,233 images of 5,749 subjects. LFW is organized in to two disjoint subsets called View 1 and 2. View 1 is a development set containing 2,200 face image pairs for training and 1,000 pairs for testing. View 2, which is meant to be used in reporting the final performance, is a 10-fold cross-validation set of 6,000 face pairs. Herein we use the LFW aligned (LFW-a) [31] version where all the original LFW images are aligned using a commercial face alignment system.

As it will be seen, we evaluate the proposed method on two face recognition modes, namely in identification and verification. For the former we use the FERET dataset whereas for the latter we use the LFW dataset. In this paper, the major part of the parameters used in the LFW evaluation is set based on the results of the preceding FERET evaluation. We are also trying to utilize as much as possible the existing knowledge on different LFW experiments found from the literature to minimise parameter tuning.

Face identification on FERET. We first align all face images based on the provided eye coordinates and rescale them to the size of 150×130 pixels. We further preprocess all face images by applying the method proposed in [32] (see Fig. 1 (a)). Then, we



(a) (b)

Fig. 1. (a) An example of a cropped and preprocessed face used in the FERET experiments, and a crop used in the LFW experiment. (b) The effect of the descriptor's length (in bits) and the support area. The curves are for raw representations each having a length of $7 \times 7 \times 2^n$, where n is the number of bits of the descriptor *dup2*.

divide the face into 7×7 blocks which are of 30×28 pixels using a vertical and horizontal overlap of 10 and 11 pixels, respectively. Finally we apply the given local binary descriptor, record the frequencies of the resulting codewords in the given block, and store them to separate histograms. After processing the whole face area, the resulting block-based histograms are concatenated to form the final representation of the face. While the soft-assignment of the codewords is applied, it is done separately for each block-based histogram before they are concatenated.

To see whether application-specific learning is beneficial for constructing local binary descriptors for describing faces, we form a specific training set and perform descriptor learning locally for each separate block resulting in a bank of descriptors, each of them specialized in describing some particular facial region. Indeed, it has been shown many times that different face regions provide different contributions to face recognition [18, 8]. In practice, we use a standard training set, which contains altogether 762 faces, provided by the CSU package [33]. Using the desired descriptor window size we then randomly sample each separate face region by evenly taking 50,000 image patches from the given 762 images and perform the descriptor learning. As we use the method given in [32], the training images are preprocessed accordingly. We compare the resulting descriptors with the corresponding ones learnt from 13 natural images [17]. For the baseline, we use the circular 8, 10 and 12 bit LBP descriptors with several different radii. To compare all these representations we use the nearest neighbor classifier applying the Hellinger distance. To take a stand on the issue of the fast growth in the representation based on local binary descriptors, we reduce the length of the final representation to 1,195 via WPCA and finally report the results using the Cosine distance. For computing the WPCA transform, we use all faces in the gallery.

The parameters we must tune are the window size of the descriptor and its dimension. For LBP, the dimension equals to the number of neighborhood pixels used in the feature calculation. For BSIF, the dimension is the number of filters, or statistically independent basis vectors. To see the influence of the window size and the dimension, we show the mean accuracy of *dup1* and *dup2* in Fig. 1 (b). The results indicate that BSIF clearly outperforms LBP and that the same window size 11×11 performs consistently

well for all tested descriptors. The best dimension, however, seems to differ as for the BSIF descriptor based on natural image patches (BSIF_N) it seems to be 10, and for the BSIF based on face image patches (BSIF_F), all four starting from 9 to 12 bits, seem to perform well. From now on, we fix the number of filters as 11, for both BSIF_N and BSIF_F descriptors.

For comprehensiveness, we report the results on all subsets using 11-bit BSIF descriptors. We also attest the usefulness of soft-assignment, setting λ as 0.9, before compressing (WPCA and Cosine distance) the representation. Based on the results in Table 1, using face image patches in descriptor learning clearly benefits. One also observes that the result of compression is remarkable. Moreover, the utilized soft-assignment method further boosts up the performance while combined with compression. In general, we noticed that by using BSIF_F combined with soft-assignments and compression, the performance was better in 16 out of 20 test cases (from 8 to 12 bits and from 7×7 to 13×13 size of filters) compared with its compressed non-smoothed version. The mean accuracies over all parameter combinations and over all subsets for the smoothed and non-smoothed BSIF_F were 96.1% and 95.5%, respectively.

Table 1. Comparative results on FERET using 11-bit and 11×11 size of descriptors. The first two columns are for raw features with the Hellinger distance metric, and *sa* refers to soft-assignment.

	BSIF_N	BSIF_F	BSIF_F + WPCA	BSIF_F^{sa} + WPCA
fb	97.9	99.0	99.7	99.7
fc	100	100	100	100
dup1	84.3	88.2	93.9	95.2
dup2	82.9	85.0	91.9	94.4
mean	91.3	93.1	96.4	97.3

Comparing our best result to the state-of-the-art, shown in Table 2, we can observe that the accuracy of our method is the best one. It must be noted that the earlier best methods, the DFD and LGXP descriptors, are based on supervised learning. Moreover, at least POEM, I-LQP, and G-LQP uses horizontal image flipping to further boost the performance, whereas our method does not use any flipping strategies in this experiment. Finally, according to [6], G-LQP is based on fusion on decision level, which has also shown to provide some gain in performance compared to using descriptors separately.

Face verification on LFW. In this experiment, after geometrical alignment we rescale all faces to the size of 150×81 using a slightly different cropping than in the previous experiment, see Fig. 1 (a). This time the face is divided into 14×8 blocks which are of 20×18 pixels using a horizontal and vertical overlap of 10 and 9, respectively. These selections are made largely based on the results provided by [34]. Based on the FERET experiment, we use 11-bit coding for both the BSIF_N and BSIF_F descriptors. For BSIF_F, we learn the descriptors locally for each block this time resulting in 112 specialized face descriptors. Soft-assignment is performed like previously but for the compression the final dimension is fixed to 2000, like in [6]. Unlike in the previous experiment, we do not apply preprocessing as it did not seem to provide any improvement based on the evaluations on View 1.

The setting of the LFW protocol forces us to learn the descriptor bank 10 times, separately for each fold. For learning a BSIF bank for one fold under evaluation, we

Table 2. Comparison to the state-of-the-art on FERET. The first value is for raw features and the second (after slash mark) is for compressed features. All but LGXP uses WPCA for compression. LGXP uses supervised Fisher Linear Discriminant (FLD) approach.

	POEM [27]	DFD [8]	LGOP [35]	LGXP [16]	I-LQP [6]	G-LQP [6]	Ours
fb	97.6 / 99.6	99.2 / 99.4	98.8 / 99.2	98.0 / 99.0	99.2 / 99.8	99.5 / 99.9	99.0 / 99.7
fc	95.0 / 99.5	98.5 / 100	99.0 / 99.5	100 / 100	69.6 / 94.3	99.5 / 100	100 / 100
dup1	77.6 / 88.8	85.0 / 91.8	83.5 / 89.5	82.0 / 92.0	65.8 / 85.5	81.2 / 93.2	88.2 / 95.2
dup2	76.5 / 85.0	82.9 / 92.3	83.8 / 88.5	83.0 / 91.0	48.3 / 78.6	79.9 / 91.1	85.0 / 94.4
mean	86.7 / 93.2	91.4 / 95.9	91.3 / 94.2	90.8 / 95.5	70.7 / 89.6	90.0 / 96.0	93.1 / 97.3

randomly picked a set of 1800 face images from the rest nine folds. This procedure confirms that we do not learn from those persons that appear in the testing set. Also, for computing the WPCA transform we used only those images that belong to the nine training folds. According to the updated protocol for evaluating methods under unsupervised paradigm, we report the performances in terms of ROC curves and by measuring the area under these curves (AUC).

The results, shown in Fig. 2 and in Table 3, indicate that our method is comparable with other methods reported in the literature. If we compared only raw features (without compression) our method would actually produce the highest AUC value. However, comparing our proposed approach to the top-performers, Pose-adaptive filtering (with WPCA according to [36]) and MRF-Fusion-CSKDA [37], it should be noticed that these methods use different kind of pose correction and therefore the results are not directly comparable in terms of image features. Moreover, it should be noted that among all of those methods using the aligned version of the LFW dataset (LFW-a), our proposed representation yields the best result.

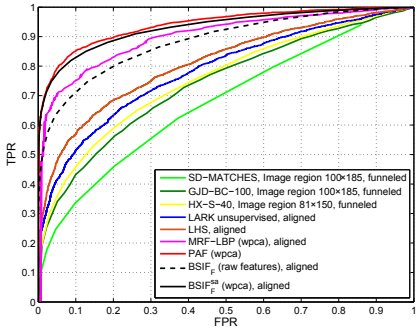


Fig. 2. ROC curves averaged over 10 folds of View 2 under *unsupervised* evaluation category.

Table 3. Comparison to the state-of-the-art methods on LFW in the unsupervised evaluation category.

method	AUC
SD-MATCHES, 125×125, funneled [34]	0.5407
GJD-BC-100, 122×225, funneled [34]	0.7392
H-XS-40, 81×150, funneled [34]	0.7547
LARK unsupervised, aligned [38]	0.7830
LHS, aligned [39]	0.8107
MRF-LBP (WPCA) [40], aligned	0.8994
Pose Adaptive Filtering (WPCA) [36]	0.9405
MRF-Fusion-CSKDA (WPCA) [37]	0.9894
BSIF$_N$, aligned	0.8026
BSIF$_F$, aligned	0.8843
BSIF$_F^{sa}$ (WPCA), aligned	0.9318

5 Conclusions

Recent studies have pointed out the importance of high-dimensional features for improving the accuracy of face recognition. In this paper, we contributed to this aspect

by presenting an optimal way of learning local image descriptors that we applied in building unsupervised face representations. The descriptor, which our face representation builds on, is based on the recent Binarized Statistical Image Features (BSIF). We showed that by learning the BSIF descriptors regionally from distinct face parts results in a very discriminative representation. In boosting up the recognition performance, we empirically approved the remarkable role of the whitening PCA (WPCA) transformation. To boost up it even further, before applying WPCA, we proposed a preprocessing step that we named histogram smoothing. We showed the histogram smoothing operation is accomplishable via a simple matrix-vector product.

Our proposed face representation yielded outperforming results compared with the current state-of-the-art on the widely known FERET benchmark. This was achieved without any kind of feature fusion or image flipping strategies that are used by most of the earlier best methods. To complement this, after slight modifications, the face representation proved highly competitive in more demanding face recognition scenarios. As for these scenarios, the method was inspected following the guidelines of the unsupervised evaluation category of the updated LFW benchmark protocol yielding promising results.

References

1. Turk, M., Pentland, A.P.: Face recognition using eigenfaces. In: CVPR. (1991) 586–591
2. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE TPAMI **19**(7) (1997) 711–720
3. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: A literature survey. ACM Comput. Surveys **35**(4) (2000) 399–458
4. Huang, G.B., Lee, H., Learned-Miller, E.: Learning hierarchical representations for face verification with convolutional deep belief networks. In: CVPR. (2012)
5. Chen, D., Cao, X., Wang, L., Sun, J.: Bayesian face revisited: A joint formulation. In: ECCV. (2012) 566–579
6. Hussain, S.u., Napoleon, T., Jurie, F.: Face recognition using local quantized patterns. In: BMVC. (2012)
7. Chen, D., Cao, X., Wen, F., Sun, J.: Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In: CVPR. (2013) 3025–3032
8. Lei, Z., Pietikäinen, M., Li, S.Z.: Learning discriminant face descriptor. IEEE TPAMI **36**(2) (2014) 289–302
9. Sun, Y., Wang, X., Tang, X.: Hybrid deep learning for face verification. In: ICCV. (2013) 1489–1496
10. Cao, X., Wipf, D., Wen, F., Duan, G.: A practical transfer learning algorithm for face verification. In: ICCV. (2013) 3208–3215
11. Barkan, O., Weill, J., Wolf, L., Aronowitz, H.: Fast high dimensional vector multiplication for face recognition. In: ICCV. (2013) 1960–1967
12. Parkhi, O.M., Simonyan, K., Vedaldi, A., Zisserman, A.: A compact and discriminative face track descriptor. In: CVPR. (2014)
13. Taigman, Y., Yang, M., Ranzato, M.A., Wolf, L.: Deepface: Closing the gap to human-level performance in face verification. In: CVPR. (2014)
14. Phillips, P., Wechsler, H., Huang, J., Rauss, P.J.: The FERET database and evaluation procedure for face recognition algorithms. IVC **16**(5) (1998) 295–306
15. Huang, G.B., Learned-Miller, E.: Labeled faces in the wild: Updates and new reporting procedures (2014) Technical report UM-CS-2014-003, University of Massachusetts, Amherst.

16. Xie, S., Shan, S., Chen, X., Chen, J.: Fusing local patterns of Gabor magnitude and phase for face recognition. IEEE TIP **19**(5) (2010) 1349–1361
17. Kannala, J., Rahtu, E.: BSIF: Binarized statistical image features. In: ICPR. (2012) 1364–1366
18. Ahonen, T., Hadid, A., Pietikäinen, M.: Face description with local binary patterns: Application to face recognition. IEEE TPAMI **28**(12) (2006) 2037–2041
19. Li, S.Z., Jain, A.K., eds.: Handbook of Face Recognition, 2nd Edition. Springer (2011)
20. Huang, G.B., Mattar, M., Lee, H., Learned-Miller, E.: Learning to align from scratch. In: NIPS. (2012) 773–781
21. Coates, A., Lee, H., Ng, A.: An analysis of single-layer networks un unsupervised feature learning. Ann Arbor **1001** (2010) 48109
22. Cao, Z., Yin, Q., Tang, X., Sun, J.: Face recognition with learning-based descriptor. In: CVPR. (2010) 2707–2714
23. Hyvärinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE TNN **10**(3) (1999) 626634
24. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR. (2008) 1–8
25. Ylioinas, J., Hadid, A., Hong, X., Pietikäinen, M.: Age estimation using local binary pattern kernel density estimate. In: ICIAP. (2013) 141–150
26. Aitchison, J., Aitken, C.: Multivariate binary discrimination by the kernel method. Biometrika **63**(3) (1976) 413–420
27. Vu, N.S., Caplier, A.: Enhanced patterns of oriented edge magnitudes for face recognition and image matching. IEEE TIP **21**(3) (2012) 1352–1365
28. Raiko, T., Ilin, A., Karhunen, J.: Principal component analysis for large scale problems with lots of missing values. In: ECML. (2007) 691–698
29. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR. (2012)
30. Huang, C., Zhu, S., Yu, K.: Large scale strongly supervised ensemble metric learning, with applications to face verification and retrieval TR115, 2007.
31. Wolf, L., Hassner, T., Taigman, Y.: Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. IEEE TPAMI **33**(10) (2011) 1978–1990
32. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE TIP **19**(6) (2010) 1635–1650
33. Bolme, D.S., Beveridge, J.R., Teixeira, M., Draper, B.A.: The CSU face identification evaluation system: Its purpose, features, and structure. In: ICVS. (2003) 304–313
34. Ruiz-del Solar, J., Verschae, R., Correa, M.: Recognition of faces in unconstrained environments: A comparative study. EURASIP Journal on Advances in Signal Processing (2009)
35. Lei, Z., Yi, D., Li, S.Z.: Local gradient order pattern for face representation and recognition. In: ICPR. (2014)
36. Yi, D., Lei, Z., Li, S.Z.: Towards pose robust face recognition. In: CVPR. (2013)
37. Arashloo, S., Kittler, J.: Class-specific kernel fusion of multiple descriptors for face verification using multiscale binarised statistical image features. IEEE TFS (2014)
38. Seo, H.J., Milanfar, P.: Face verification using the lark representation. IEEE TIFS **6**(4) (2011) 1275–1286
39. Sharma, G., ul Hussain, S., Jurie, F.: Local higher-order statistics (lhs) for texture categorization and facial analysis. In: ECCV. (2012)
40. Arashloo, S.R., Kittler, J.: Efficient processing of mrfs for unconstrained-pose face recognition. In: BTAS. (2013)