Journal of Visual Communication and Image Representation 51 (2018) 155–161

Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

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

Short Communication

# On Fisher vector encoding of binary features for video face recognition[☆]

Yoanna Martínez-Díaz[a], Noslen Hernández[b,*], Rolando J. Biscay[c], Leonardo Chang[e], Heydi Méndez-Vázquez[a], L. Enrique Sucar[d]

[a] Advanced Technologies Application Center (CENATAV), Havana, Cuba
[b] University of Sao Paulo (USP), Sao Paulo, Brazil
[c] Centro de Investigación en Matemáticas (CIMAT), Guanajuato, Mexico
[d] Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Puebla, Mexico
[e] Tecnológico de Monterrey, Campus Estado de México, Estado de México, Mexico

ARTICLE INFO

*Keywords:*
Fisher vector
Binary features
Face recognition
Video

ABSTRACT

Several approaches have been proposed for face recognition in videos. Fisher vector (FV) encoding of local Scale-Invariant Feature Transforms (SIFT) is among the best performing ones. Aiming at speed up the computation time of this approach, a method based on FV encoding of binary features was recently introduced. By using Binary Robust Independent Elementary Features (BRIEF), this method gained in efficiency but lost in accuracy. FV representation of binary features demands appropriated mathematical tools, which are not as easy available as for continuous features. This paper introduces a new way for obtaining FV encoding of binary features that is still efficient and also accurate. We show that BRIEF combined with FV are discriminative enough, and provide as good performance as the one obtained by using SIFT features for video face recognition. Besides, we discuss several insights and promising lines of future work in regard to FV encoding of binary features.

## 1. Introduction

Video face recognition has attracted the attention from both academy and industry in the last years, mainly due to its relevance for real-world applications (e.g., video-surveillance, human-computer interaction) [1–3]. For such applications it is necessary to develop algorithms that operate with limited computing resources, maintaining a high accuracy and a processing speed close to real-time.

Recently, state-of-the-art results in video face recognition have been achieved by using convolution neural networks [4–6]. However, the main limitation of these approaches is the requirement of a massive amount of labeled training data (millions of images) to achieve good generalization, demanding thus a lot of training time and computing resources. Thus, “shallow” (not deep) learning methods are still under the attention of researchers [7–10]. Among these approaches, those based on the Fisher vector (FV) representation have shown competitive performance on unconstrained video face recognition [10,11]. FV representation was first used for face recognition in images [12], and later extended for video face recognition giving rise to the Video Fisher Vector Faces (VF$^2$) descriptor [10].

In general, FV representation is obtained by densely computing local descriptors (e.g., SIFT) at different scales and locations, which are aggregated into a high-dimensional vector and encoded by the derivatives of the log-likelihood of a parametric generative model (e.g., Gaussian Mixture Model) for such descriptors, evaluated at the fitted parameters on the basis of a training sample. In particular, VF$^2$ representation uses SIFT features and has a high computational cost. For this reason, despite its good accuracy, it is extremely hard to use it in real-time applications. Focused on improving the computation time of VF$^2$ and gaining in efficiency, we recently introduced the Binary Video Fisher Vector Faces (BinVF$^2$) descriptor [13]. It was elaborated using binary features and Bernoulli Mixture Model (BMM) in the FV encoding scheme. BRIEF descriptor [14] was chosen as binary representation because of its simple construction and compact storage in memory. In addition, it is known that BRIEF is two orders of magnitude faster than SIFT with comparable distinctiveness [14]. A considerable speed-up was achieved with BinVF$^2$, but the accuracy decreased in comparison to the one obtained by VF$^2$.

Several issues considered in the VF$^2$ representation, aimed at improving its effectiveness, remain unsolved for the proposed BinVF$^2$, mainly due to novel particularities involved in the use of binary features. One of them is the inclusion of the spatial information of the local descriptors in the encoding. This is usually done in two different ways. The first one is by using a Spatial Pyramid approach [15], which

[☆] This paper has been recommended for acceptance by Zicheng Liu.
[*] Corresponding author.
*E-mail addresses:* noslenh@gmail.com, noslenh@ime.usp.br (N. Hernández).

https://doi.org/10.1016/j.jvcir.2018.01.017
Received 5 July 2017; Received in revised form 4 December 2017; Accepted 27 January 2018
Available online 31 January 2018

consists of dividing an image into a number of cells and then stacking the FVs computed for each of them. However, this strategy results in a notably high dimension representation, not desirable if we are looking for an efficiency improvement. The second one consists of augmenting each local descriptor with their spatial coordinates; but this, which is theoretically justified for Gaussian Mixture Model (GMM) and continuous features [16], can not be straightforwardly accomplished with binary features and BMM. Another issue is the decorrelation (and optionally, dimensionality reduction) of local descriptors before computing the FV, in order to fulfill the assumptions of generative models like GMM and BMM. In VF$^2$, this is easily solved by applying Principal Component Analysis (PCA) to SIFT descriptors.

Finding ways to deal with these issues in the case of FV encoding of binary features constitutes a promising line of research. A basic question that arises in this respect is whether binary features (BRIEF, in particular) contain enough information to provide a FV encoding that achieves an accuracy similar to the one obtained when SIFT descriptors are encoded. In this paper we report some findings that support an affirmative answer to this question for the case of video face recognition. We introduce a method named Logistic Binary Video Fisher Vector Faces (LBinVF$^2$), which offers a solution that achieves similar effectiveness as VF$^2$ but with greater efficiency. LBinVF$^2$ consists of applying a Logistic Principal Component Analysis (LPCA) approach [17] to BRIEF descriptors, and then doing the FV encoding of the projected descriptors augmented with their spatial information. In addition, we discuss several issues concerning the understanding of the different steps involved in the FV encoding process. Finally, we point out some future work aimed at obtaining a FV representation of binary features on the basis of more suitable mathematical tools.

The reminder of this paper is organized as follows. Section 2 introduces the proposed LBinVF$^2$ representation for video face recognition. Experiments on two public benchmarks (YouTube Faces and COX Face) are presented in Section 3, as well as discussion on the results and future work. Conclusions are given in Section 4.

## 2. The LBinVF$^2$ representation

This section describes in detail the steps involved in the proposed LBinVF$^2$ representation. First, BRIEF descriptors are extracted densely at multiple scales and locations from all the video frames. Then, a Logistic PCA method is used to project the binary descriptors to a real-valued vector space. This allows us to reduce the dimension of local descriptors and to include the spatial coordinates by simply stacking them to the projected features. Finally, the FV encoding of such augmented features provides the representation. The overview of LBinVF$^2$ is illustrated in Fig. 1.

### 2.1. Binary representation

Recently, the use of local binary descriptors has become a suitable option for real-time applications [14,18–20]. We have chosen BRIEF descriptor as binary representation since it is very efficient to compute as well as to store in memory. Moreover, BRIEF has shown to be significantly faster than other local descriptors such as SIFT and SURF, reporting comparable accuracy in some pattern recognition tasks [21].

BRIEF creates a bit vector resulting from a set of binary test responses computed over a smoothed image patch. Formally, a binary test $\tau$ on a patch $I$ of size $M\times M$ $(M\in\mathbb{N})$ is defined as:

$$\tau(I;u,v) := \begin{cases} 1 & \text{if } I(u) < I(v) \\ 0 & \text{otherwise,} \end{cases}$$

where $u$ and $v$ $(u,v\in\mathbb{N}\times\mathbb{N})$, are locations in the patch $I$, and $I(v)$ denotes the pixel intensity of $I$ at $v$. The locations of the pixels are preselected randomly according to a Gaussian distribution around the patch center. Finally, using a set of $\eta\in\mathbb{N}$ binary tests, we obtain a bitstring of length $\eta$, whose decimal representation is defined by:

$$f_\eta(I) = \sum_{1\leqslant i\leqslant\eta} 2^{i-1}\tau(I;u_i,v_i). \tag{1}$$

Similar to our previous work in [13], we use $\eta = 256$ and BRIEF descriptors are densely sampled on a regular grid at multiple scales, without needing a sophisticated landmarks detector. This process is performed for every video frames, hence the importance of using local visual descriptors that can be computed very fast.

### 2.2. Logistic principal component analysis

Once we have computed all the BRIEF descriptors corresponding to a video, we transform them into a set of continuous features by using a PCA-like method for binary data. This strategy has two fold benefits. First, transforming from binary to continuous features gives us a simple way to incorporate the spatial information into the FV encoding, by just stacking the spatial coordinates to the projected features and using a GMM model. Secondly, due to the advantages that a PCA-like method bring to fulfill the assumptions of generative models used in the FV encodings.

Dimensionality reduction of binary data has been studied for some time. Different generalizations of classical PCA have been proposed for this end [22–24]. Recently, a new approach called Logistic PCA was introduced [17,25]. This method extends PCA to binary data by re-interpreting Pearson's formulation of PCA [26] as a projection of the natural parameters of the saturated model to minimize the Gaussian deviance. Logistic PCA has a nice advantage over the previous ones which is, indeed, very suitable for our problem (and interesting in general for real-time applications): it allows a fast and easy evaluation of principal component scores for new data. Unlike other approaches that involve additional optimizations to find the scores for each new data, Logistic PCA (like the standard PCA) reduces this operation to just a matrix multiplication.

Let $\mathbf{X}$ be a matrix of $N$ observations of $D$-dimensional binary data. Assume that each $x_{ij}$, $1\leqslant i\leqslant N$, $1\leqslant j\leqslant D$, is from Bernoulli$(p_{ij})$, with natural parameter $\theta_{ij} = \operatorname{logit}(p_{ij})$. Let $\widetilde{\theta}_{ij} = mq_{ij}$ (with $q_{ij} = 2x_{ij}-1$ and $m$ a large number) be a convenient approximation of the natural parameters of the saturated model.

The natural parameters are estimated by $\widehat{\theta}_i = \boldsymbol{\mu}-\mathbf{U}\mathbf{U}^T(\widetilde{\theta}_i-\boldsymbol{\mu})$, where $\boldsymbol{\mu}$ is a $D$-dimensional vector and $\mathbf{U}$ is a $D\times Q$ orthonormal matrix. The objective function to minimize is the Bernoulli deviance of the estimated natural parameters matrix $\widehat{\mathbf{\Theta}}$ from the data matrix $\mathbf{X}$,

$$\min_{\boldsymbol{\mu},\mathbf{U}:\mathbf{U}^T\mathbf{U}=\mathbf{I}_Q} D(\mathbf{X};\widehat{\mathbf{\Theta}}) = \sum_{i=1}^{N}\sum_{j=1}^{D} 2\log(1+\exp(\widehat{\theta}_{ij}))-2x_{ij}\widehat{\theta}_{ij}. \tag{2}$$

Let $\widehat{\mathbf{U}}$ be the principal components loadings resulting from the optimization problem in Eq. (2). The principal component scores, $\mathbf{s}\in\mathbb{R}^Q$, of a new data $\mathbf{x}^*\in\{0,1\}^D$ are determined by

$$\mathbf{s} = \widehat{\mathbf{U}}^T\widetilde{\theta}^*, \tag{3}$$

where $\widetilde{\theta}^* = m(2\mathbf{x}-1) = m\mathbf{q}^*$ is the approximate natural parameter for $\mathbf{x}$ under the saturated model. Note that the principal component scores are linear combinations of transformations of the data, as in the classical PCA.

It is important to point out that the optimization problem defined in Eq. (2) is solved off-line and only on the training step. The projection of BRIEF descriptors done during the computation of each LBinVF$^2$ representation involves only Eq. (3).

### 2.3. Fisher vector encoding

Fisher vector encoding assumes that the generation process of local descriptors can be modeled by a probability density function. The gradient of the log-likelihood is a useful vector-valued signature of the

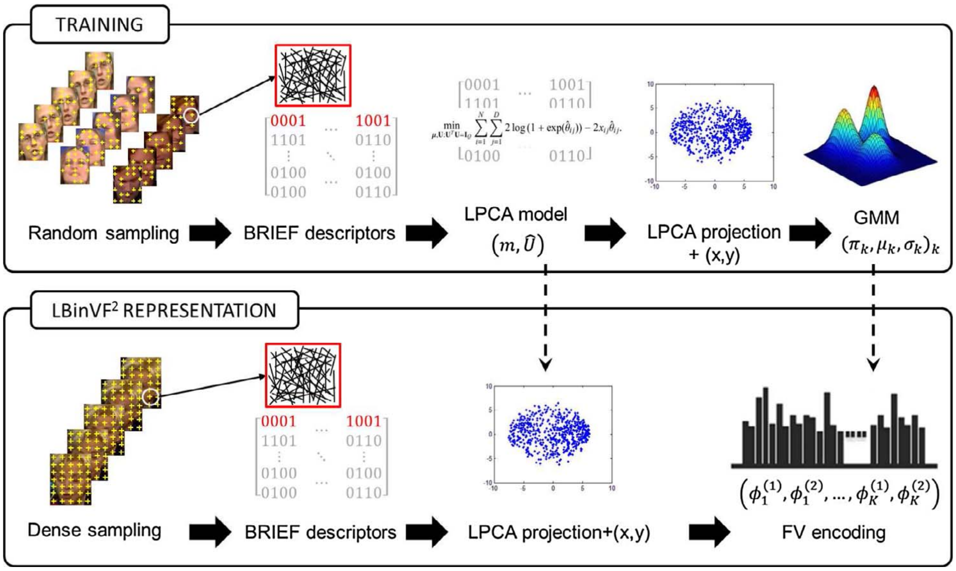

**Fig. 1.** The pipeline of LBinVF$^2$ representation.

generation process. On this basis, the FV representation is defined by assuming a parametric generative model for the data and stacking the derivatives of its log-likelihood with respect to all its parameters.

As it was mentioned before, BRIEF descriptors are first projected to a $Q$-dimensional real-valued vector space by using LPCA. So, we can use the well-established formulation of FV based on the GMM. Let $(\mathbf{s}_1,\mathbf{s}_2,\ldots,\mathbf{s}_N),\mathbf{s}_i \in \mathbb{R}^Q$ be a set of $N$ projected descriptors. The FV representation of this set is given by $F = (\boldsymbol{\phi}_1^{(1)},\boldsymbol{\phi}_1^{(2)},\ldots,\boldsymbol{\phi}_K^{(1)},\boldsymbol{\phi}_K^{(2)}) \in \mathbb{R}^{2QK}$ where the entries $\phi_{kq}^{(1)}$ and $\phi_{kq}^{(2)},q = 1,\ldots,Q$, of the vectors $\boldsymbol{\phi}_k^{(1)}$, and $\boldsymbol{\phi}_k^{(2)}$ are determined by

$$\phi_{kq}^{(1)} = \frac{1}{N\sqrt{\pi_k}} \sum_{n=1}^{N} \alpha_k(\mathbf{s}_n)\left(\frac{s_{nq}-\mu_{kq}}{\sigma_{kq}}\right), \tag{4}$$

$$\phi_{kq}^{(2)} = \frac{1}{N\sqrt{2\pi_k}} \sum_{n=1}^{N} \alpha_k(\mathbf{s}_n)\left[\left(\frac{s_{nq}-\mu_{kq}}{\sigma_{kq}}\right)^2-1\right]. \tag{5}$$

Here, $s_{nq}$ refers to the $q$th element of vector $\mathbf{s}_n$. $(\pi_k,\boldsymbol{\mu}_k,\boldsymbol{\sigma}_k)_k$ are the estimated weights, mean vectors and diagonal covariances parameters of a GMM model with $K$ components, respectively. These parameters are learned beforehand and off-line on a given training set. $\alpha_k(\mathbf{s}_n)$ is the posterior soft assignment of the data $\mathbf{s}_n$ to the $k$th component. More details on Eqs. (4) and (5) can be found in [27].

In order to incorporate the spatial information, we augment each projected feature $\mathbf{s}_i$ with their normalized spatial coordinates $(u_i^x,u_i^y)$ and use these augmented features, $(\mathbf{s}_i,u_i^x,u_i^y)$, in the FV encoding. The dimension of the resulting Fisher vector will be , where $P = Q + 2$. Following [27], the performance of FV is further improved by applying a signed square-rooting and L2 normalisation.

To decide if two videos belong to the same person or not, we compare their LBinVF$^2$ representations by using a Joint Bayesian approach for face verification [28]. This approach models the joint distribution of features vectors of a pair of face images being the same or different persons and uses the log-likelihood ratio of intra- and inter-classes probabilities as the similarity measure.

## 3. Experimental results and discussion

In this section we evaluate the performance of the introduced LBinVF$^2$ representation for video face recognition. First, the effectiveness of LBinVF$^2$ is tested and compared with state-of-the-art methods on two publicly available databases: YouTube Faces and COX Face. Second, based on the framework of the experiments carried out in the YouTube Face database, we analyze the role of both PCA step and the inclusion of the spatial information in the FV encoding process. Finally, the efficiency of LBinVF$^2$ is evaluated.

For the two databases, every video frame was center cropped to 150 × 150 pixels. BRIEF descriptors were computed densely over 24 × 24 patches using 2-pixel spacing at 5 levels of the scale-space pyramid with scale factor $\sqrt{2}$.

### 3.1. Effectiveness assessment on YouTube Faces

The challenging YouTube Faces (YTF) database [29] is designed for unconstrained face verification in videos. It contains 3425 videos of 1595 subjects with significant variations on expression, illumination, pose, resolution and background. The length of the videos is 181.3 frames on average. 5000 video pairs are available from YTF database, which are divided into 10 splits. Each split contains 250 positive pairs (same subject) and 250 negative pairs (different subjects).

We follow the restricted protocol defined in the YTF database, where a 10-fold cross-validation procedure is performed selecting at each time 9 splits for training and the remaining split for testing. All learning tasks involved in our method, i.e., the estimation of the LPCA model and the Gaussian Mixture Model, were performed independently for each fold. We use one million of BRIEF descriptors randomly sampled from the videos in the training set to fit the LPCA model, using Q = 32 components (the selection of this value is explained in the next section). To train the GMM, we set the number of components to $K = 512$ and used $10^6$ LPCA-projected descriptors augmented with their normalized spatial coordinates. Three evaluation measures are considered to report the verification results on the 10 splits [29]: recognition rate (accuracy), area under curve (AUC) and equal error rate (EER).

The verification results under the restricted protocol of LBinVF$^2$ and state-of-the-art methods are shown in Table 1. In addition, the corresponding ROC curves are plotted in Fig. 2. Under this protocol, the subject identity labels of YouTube Faces database are not allowed to be used for training, neither the use of outside training data. For these reasons, deep learning-based methods, which have an outstanding performance on this dataset but under the unrestricted protocol, were not included in the comparisons.

It can be noticed that LBinVF$^2$ achieves the best results, outperforming all the methods included in the comparison in two of the three evaluation measures used (EER and AUC). It achieves a performance very close to that of the state-of-the-art method VF$^2$. Also, LBinVF$^2$ significantly improves our previous approach BinVF$^2$ in about 5%, in terms of ERR and AUC, and in more than 3% in terms of accuracy.

**Table 1**
Performance on the YTF database under the restricted protocol.

| Method | EER | AUC | Accuracy ± SE |
|---|---|---|---|
| MGBS (mean) LBP [29] | 25.3 | 82.6 | 76.4 ± 1.8 |
| MBGS + SVM [30] | 21.2 | 86.9 | 78.9 ± 1.9 |
| APEM_FUSION [31] | 21.4 | 86.6 | 79.1 ± 1.5 |
| STFRD + PMML [32] | 19.9 | 88.6 | 79.5 ± 2.5 |
| VSOF + OSS [8] | 20.0 | 89.4 | 79.7 ± 1.8 |
| DDML(combined) [33] | 18.5 | 90.1 | 82.3 ± 1.5 |
| EigenPEP [34] | 15.5 | 92.6 | **84.8 ± 1.4** |
| VF$^2$ [10] | 14.9 | 93.0 | 84.7 ± 1.4 |
| BinVF$^2$ [13] | 19.9 | 87.8 | 79.8 ± 1.6 |
| **LBinVF$^2$** | **14.6** | **93.2** | 83.3 ± 1.8 |

Bold values refer to the best result in each evaluation measure.

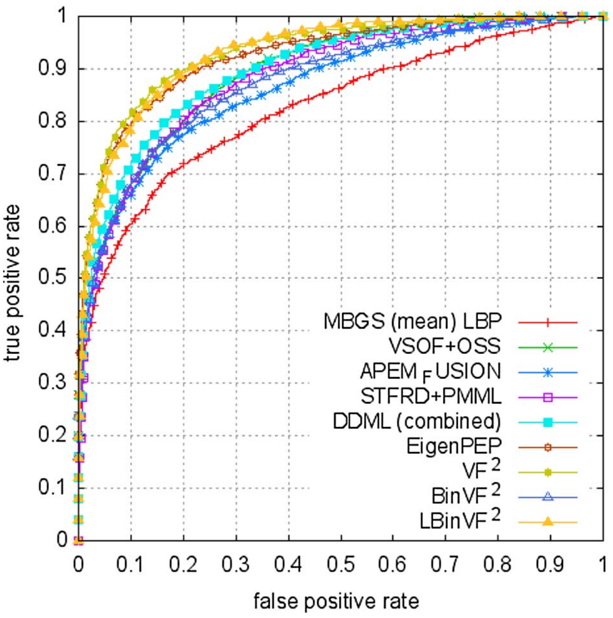

**Fig. 2.** ROC curves on the YouTube Faces database under the restricted protocol.

### 3.2. Effectiveness assessment on COX Face

The COX Face database [35] was designed to simulate applications like video surveillance, containing both still images and videos of 1000 Chinese subjects. For each subject, the database contains one high-quality still image and three video clips each captured from a different camera (Cam1, Cam2 and Cam3). The videos have natural variations in pose, expression, lighting, blur, and face resolution.

The proposed method is evaluated following the defined face verification and face identification protocols for video-to-video (V2V) scenario [35]. The identification performance is presented by rank-1 recognition rate while the verification rate is reported by ROC curve.

For the V2V evaluation scenario, the database provides 10 random partitions of the 1000 subjects. In each partition the corresponding videos of 300 subjects are used for training, while the videos of the remaining 700 subjects are used for testing [35]. Since there are three videos per subject, in order to form either target set or query set, six experiments are defined.

Besides state-of-the-art methods reported in the COX Face database, we also included in the comparisons BinVF$^2$ and VF$^2$ methods. Table 2 shows the rank-1 identification rates of testing algorithms under V2V scenario. The Vi-Vj denotes the test using the videos taken from the ith camera as query, and the videos taken from the jth camera as target. It can be observed that LBinVF$^2$, VF$^2$ and TBE-CNN achieved a very similar performance, being the first three best algorithms. Although our previous approach BinVF$^2$ not reach the state-of-the-art, it obtains a comparable results, even very close to that of the VGG-Face deep network.

Fig. 3 shows the ROC curves for LBinVF$^2$, BinVF$^2$ and VF$^2$ methods. As it can be appreciate, LBinVF$^2$ and VF$^2$ perform very similar. Moreover, LBinVF$^2$ obtains better performance than BinVF$^2$, which is consistent with previous results. The top verification rates reported by [35] in this scenario for the experiments V2-V1/V1-V2, V3-V1/V1-V3 and V3-V2/V2-V3 are 66.67%, 75.69%, 70.19% at FAR = 0.001, and 87.26%, 90.77%, 87.14% at FAR = 0.01. As we can see, FV encoding-based methods significantly improve these results.

### 3.3. On the influence of PCA and the spatial information

In this section we study the influence of some steps of the pipeline of VF$^2$ and LBinVF$^2$ methods. We carried out several experiments on the YTF database that involved applying or not dimensionality reduction and, incorporating or not the spatial information of local descriptors during the FV encoding of dense SIFT and dense BRIEF features. Hereafter we use the acronym (L)PCA that should be read as LPCA or PCA depending on the context. For SIFT-based methods, we used the VLFeat Library [40] provided by the authors, which implements the core functionalities i.e., dense SIFT descriptor, PCA and FV encoding based on GMM.

Table 3 shows the obtained results. We first note that for both SIFT-based and BRIEF-based encodings, the (L)PCA step improves the performance. When (L)PCA is not applied the results are far worse, which means that the (L)PCA step is quite influencing.

FV encoding based on GMM (under the standard assumption of diagonal covariance matrices) ignores dependences between variables in each mixture component. To some extent this restrictive assumption is compensated by using PCA. Indeed, PCA takes into account the correlation structure among the original SIFT descriptors, giving place to new no correlated variables for which the mathematical assumptions of GMM with diagonal covariance matrices holds.

In the case of FV encoding of BRIEF features, a similar situation arises. When LPCA is not used (result that corresponds to BinVF$^2$ as highlighted in Table 3), a BMM is used as generative model. BMM assumes independence between variables (which implies no correlation), and this is highly violated by BRIEF descriptors. Regrettably, in this case we do not have a simple mathematical tool to compensate for this too restrictive assumption of BMM, analogously as it is done for SIFT features and GMM. That is to say, there is no method to decorrelate the binary features and, at the same time, keep the data in the binary space. A kind of PCA approach that transforms binary features into new no correlated binary features, would be a convenient tool to be used with generative models for binary data in which assumptions on independence are made (e.g., BMM). This remains as a challenging open problem.

Another open problem that exists on the FV encoding of BRIEF features is how to incorporate the spatial information. For example, BMM as generative model does not allow us to take into account the spatial information in an easy way. Therefore, new generative models are required for modeling binary features jointly with the spatial information. No proposal is currently available for this challenging goal, but LBinVF$^2$ provides a simple way to overcome this difficulty.

It can be noticed in Table 3 that including the spatial information always improve the results. Each component in the mixture encodes the behavior of local features (i.e., how its density is) in a different spatial region of the face image (whose size varies depending on the component). When the derivative vector of the log-likelihood with respect to its parameters (i.e., the Fisher vector) is computed, each component in the mixture (more specifically, its derivative) leads to a distinct sub-vector in the FV. In this way, the FV contains consecutive blocks corresponding to different spatial regions. This happens even when no spatial information is taken into account. Nevertheless, including information about the location of patches (through additional entries into

**Table 2**
Comparison of testing methods for rank-1 identification rates (%) under V2V face recognition scenario on the COX Face database. The bold values correspond to the first three highest performances that consist of both mean and standard deviation over 10 random runs.

| Method | V2-V1 | V3-V1 | V3-V2 | V1-V2 | V1-V3 | V2-V3 |
|---|---|---|---|---|---|---|
| PSCL-SS [35] | 57.70 ± 1.40 | 73.17 ± 1.44 | 67.70 ± 1.70 | 62.77 ± 1.02 | 78.26 ± 0.97 | 68.91 ± 2.28 |
| LERM-SS [36] | 65.94 ± 1.97 | 78.24 ± 1.32 | 70.67 ± 1.88 | 64.44 ± 1.55 | 80.53 ± 1.36 | 72.96 ± 1.99 |
| GGDA [37] | 70.80 ± 1.24 | 76.23 ± 1.25 | 71.99 ± 1.05 | 69.17 ± 1.01 | 76.77 ± 1.57 | 77.43 ± 1.41 |
| CDL [38] | 78.43 ± 1.01 | 85.31 ± 0.97 | 79.71 ± 1.47 | 75.56 ± 1.95 | 85.84 ± 0.86 | 81.87 ± 1.14 |
| VGG-Face [5] | 94.51 ± 0.47 | 95.34 ± 0.32 | 96.39 ± 0.42 | 93.39 ± 0.56 | 96.10 ± 0.27 | 96.60 ± 0.52 |
| TBE-CNN [39] | **98.07 ± 0.32** | **98.16 ± 0.23** | **97.93 ± 0.20** | **97.20 ± 0.26** | **99.30 ± 0.16** | **99.33 ± 0.19** |
| VF$^2$ [10] | **98.27 ± 0.39** | **99.34 ± 0.24** | **99.06 ± 0.29** | **98.33 ± 0.34** | **99.47 ± 0.23** | **99.23 ± 0.20** |
| BinVF$^2$ [13] | 93.37 ± 0.63 | 96.81 ± 0.71 | 95.93 ± 0.57 | 93.17 ± 0.59 | 96.56 ± 0.56 | 96.39 ± 0.64 |
| **LBinVF$^2$** | **97.83 ± 0.39** | **98.96 ± 0.29** | **98.63 ± 0.35** | **97.99 ± 0.41** | **98.93 ± 0.23** | **98.87 ± 0.30** |

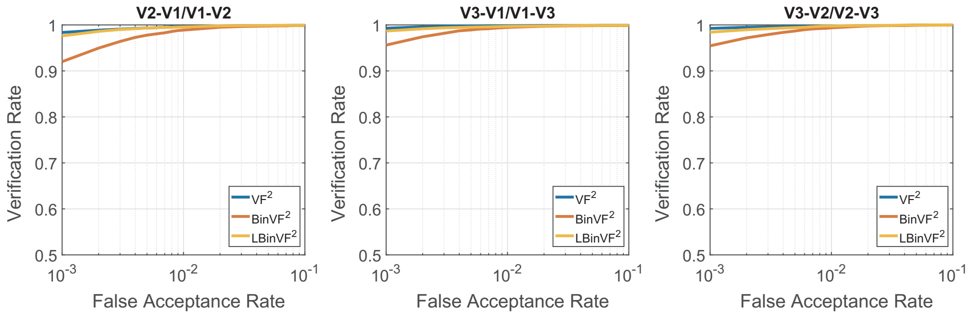

**Fig. 3.** Face verification ROC curves for the different face methods on V2V scenario of the COX Face database.

**Table 3**
Performance in terms of EER for different FV encoding of SIFT and BRIEF features.

| | SIFT-based encoding | | BRIEF-based encoding | |
|---|---|---|---|---|
| | No spatial inf. | Spatial inf. | No spatial inf. | Spatial inf. |
| No PCA | 20.4 | 20.0 | 19.9[b] | Open problem |
| (L)PCA | 16.8 | 14.9[a] | 17.0 | 14.6[c] |

[a] VF$^2$.
[b] BinVF$^2$.
[c] LBinVF$^2$.

the descriptor vector) leads to a better spatial distribution of the different components around the face region.

Fig. 4 shows the spatial distribution of the mixture components over the face region for GMM trained with PCA-projected SIFT features and with LPCA-projected BRIEF features, when the spatial information is included and not. Each mixture component is represented by a red ellipse with center (green cross) and shape parameters (semi-major axis and semi-minor axis) defined by the spatial mean and the spatial variances of the component, respectively. We can see from Fig. 4(a) and (c) that when the spatial information of local descriptors is not considered, the components are located on different regions of the image but the tendency is to have large variances, taking into account the information in a quite global way. On the contrary, in Fig. 4(b) and (d), a more coherent spatial distribution of the components is obtained when spatial information is incorporated, leading to components with small variances on informative parts of the face image and components with greater variances in non-informative regions.

Finally, we would like to draw the attention to another important aspect in the FV encodings using GMM and the spatial information stacked to the (L)PCA-projected features. So far we have described the role of (L)PCA for obtaining new variables that meet the usual mathematical assumptions of the generative models. But, is it mandatory to worry about the dimension of the new features?

When using GMM with the spatial information aggregated to the features, the log-likelihood is a product of a term which is a density for the spatial variables (usually two) and a term that is a density for the non-spatial variables (e.g., SIFT, (L)PCA-projections of BRIEF or SIFT). When there are many non-spatial variables, its corresponding term begins to dominate because (under independence assumption) this factor is equal to the product of the densities of each variable (or equivalently, its logarithm is a sum of many summands, each one corresponding to one variable). As a consequence, a large number of non-spatial variables implies a loss in the role of the spatial information.

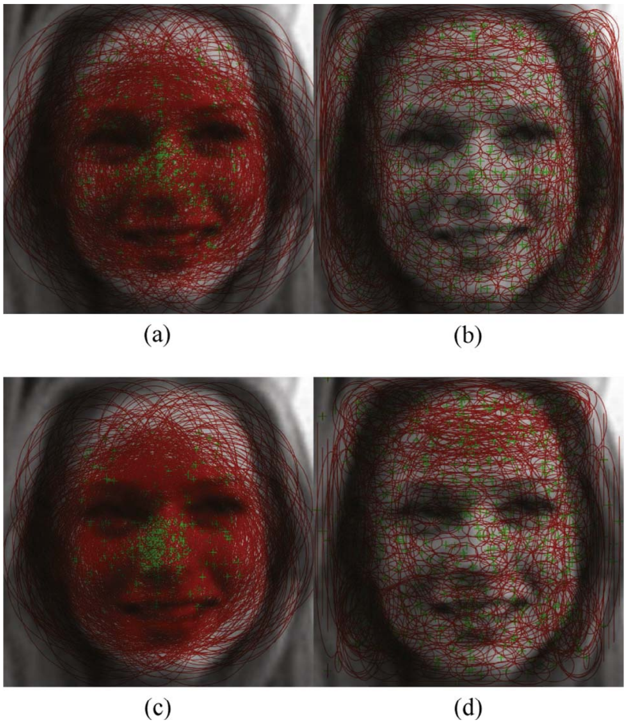

**Fig. 4.** GMM clusters of (a) PCA-projected SIFT features without spatial information, (b) PCA-projected SIFT features with spatial information, (c) LPCA-projected BRIEF features without spatial information and (b) LPCA-projected BRIEF features with spatial information.

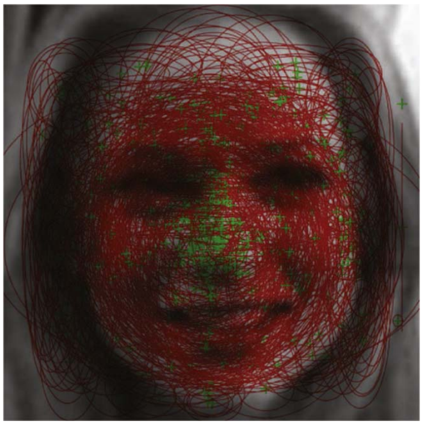

**Fig. 5.** GMM clusters of LPCA-projected BRIEF features on a 250 dimensional space with spatial information.

The number of variables, for which the spatial term is negligible compared to the terms associated to non-spatial variables, depends on the data and the specific model. For example, experiments in [27] show that the effectiveness of the method remains stable when the number of principal components increases above 64 until 128. This means that in that case, 128 appears to be the number of variables for which this phenomenon has not been evidenced yet. But in general, it is something to consider, especially when the dimension is particularly large. For the BRIEF descriptors used in this work, whose original dimension (256) is twice the one of SIFT, the spatial information begins to lose influence for a bigger number of variables. Fig. 5 illustrates the spatial distribution of components when BRIEF features are projected to a 250-dimensional space. Note that the spatial pattern is less clear and has less resolution than that obtained in Fig. 4(d). Besides, it can be appreciate in Fig. 6 that the accuracy begins to deteriorate as the dimension increases. This offers another reason why dimensionality reduction tools (like (L)PCA) are effective when they are combined with generative models.

### 3.4. Computational efficiency assessment

Since YTF database has more videos compared to COX Face, we select it in order to evaluate and compare the computational efficiency of our proposal. For this aim, we followed the experimental setup described in [13], where the average time for video face representation over all the videos is reported. We compared the computation time for the different steps involved in the VF$^2$, BinVF$^2$ and LBinVF$^2$ methods. Table 4 reports the computation time obtained for feature extraction, dimensionality reduction and FV encoding steps; as well as the total time of processing.

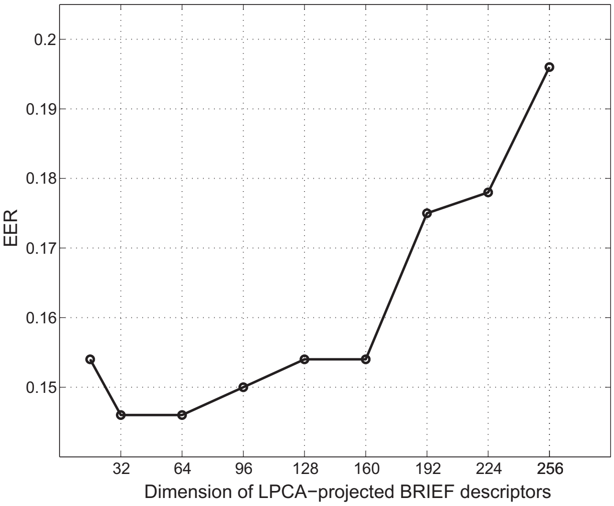

**Fig. 6.** Influence of the dimensionality reduction of BRIEF descriptors in LBinVF$^2$ representation on the YTF database.

**Table 4**
Computation time comparison of FV representations on the YouTube Faces database. All times are expressed in seconds and represent the average computation time for a video.

| Method | Feat. extraction | Dim. reduction | FV encoding | Total |
|---|---|---|---|---|
| VF$^2$ | 11.20 | 3.75 | 3.09 | 18.04 |
| BinVF$^2$ | 0.72 | – | 3.50 | 4.22 |
| LBinVF$^2$ | 0.72 | 2.74 | 2.58 | 6.04 |
| Speed-up (BinVF$^2$ vs. VF$^2$) | 15.5x | – | 0.9x | 4.2x |
| Speed-up (LBinVF$^2$ vs. VF$^2$) | 15.5x | 1.37x | 1.2x | 3x |

It can be seen in Table 4 that in the feature extraction step, both BinVF$^2$ and LBinVF$^2$ significantly outperform the computational time of VF$^2$, with a speed-up over 15x. This improvement is achieved thanks to the use of efficient binary features (i.e., BRIEF) instead of SIFT features. In the dimensionality reduction step no significant speed-up is obtained since, as it was explained in Section 2.2, LPCA and PCA are similar in regards to project a new data from the computational point of view. During the FV encoding both, LBinVF$^2$ and VF$^2$, execute the same operations, so the improvement of 1.2x achieved on this step is only due to the use of fewer LPCA components (32 instead of 64). This reduction on the number of retained principal components also give us a memory and disk storage improvement by a factor of two because the length of the resulting representation is $2K(Q+2)$, where $Q$ is the dimension of the projected descriptors.

On the other hand, we can also see from Table 4 that the overall speed-up of LBinVF$^2$ with respect to VF$^2$ is lower than that achieved by BinVF$^2$ in our previous work (it is reduced from 4.2x to 3x). This drop is a result of the inclusion of the dimensionality reduction step and also the loss of the possibility to operate with binary data during the FV encoding step. Nevertheless, LBinVF$^2$ is still three times faster than VF$^2$ with a better performance, which was not achieved by BinVF$^2$.

Both BinVF$^2$ and LBinVF$^2$ implementations were developed in C++. For the BRIEF descriptor we used the implementation provided in OpenCV 2.4.9. For the VF$^2$ method, we developed a C++ implementation using the VLFeat Library [40] that, as we mentioned before, implements the core functionalities i.e., PCA, FV encoding and dense SIFT descriptor. Note that the VLFeat Library implements a fast version of SIFT optimized for dense extraction which is 30–70 times faster compared to the original SIFT, according to VLFeat authors. Thus, we are here comparing the introduced LBinVF$^2$ in terms of efficiency with an optimized implementation of VF$^2$. All the experiments were conducted on a PC with an Intel i5-3470 CPU at 3.20 GHz and 8 GB of RAM.

## 4. Conclusions

In this work we proposed a novel representation, named LBinVF$^2$, for face recognition in videos. The proposal is based on FV encoding of binary descriptors (i.e., BRIEF), and it is a step forward into the research initiated in a previous work [13], where BinVF$^2$ was introduced achieving a considerable speed-up w.r.t. state-of-the-art methods but with a worst accuracy. Here, we managed to insert two fundamental steps in the encoding process of binary features: (1) incorporating the spatial information of the local descriptors, and (2) decorrelating them (and reduce their dimension) in order to fulfill the assumptions of generative models. Through an extensive experimental evaluation on YTF and COX databases, we have shown the influence of these two steps and demonstrated that the new proposal is both efficient and accurate, achieving state-of-the-art results with a considerable speed up.

This proposal is just a first step in the research on FV encoding of binary features that should be further elaborated in the future by means of well-founded binary modeling, as it was discussed in this work. Meanwhile, the given solution constitutes an appealing alternative to add to the toolkit of available methods for video face recognition due to its simplicity and good performance. In addition, the proposal could be extended to other applications such as large-scale image recognition.

## Acknowledgments

The second author was supported by FAPERJ/CAPES (E45/2013). This work is supported in part by FONCICYT (CONACYT and European UNION) Project No. 272727.

## References

[1] M. Hassaballah, S. Aly, Face recognition: challenges, achievements and future directions, IET Comput. Vis. 9 (4) (2015) 614–626.
[2] Z. Mahmood, N. Muhammad, N. Bibi, T. Ali, A review on state-of-the-art face recognition approaches, Fractals 25 (02) (2017) 1750025.
[3] P. Grother, G. Quinn, M. Ngan, Face in Video Evaluation (Five) Face Recognition of Non-cooperative Subjects, Interagency Report 8173, National Institute of Standards and Technology, March 2017. http://dx.doi.org/10.6028/NIST.IR.8173.
[4] J. Yang, P. Ren, D. Chen, F. Wen, H. Li, G. Hua, Neural aggregation network for video face recognition, in: CVPR, In press, 2017.
[5] O.M. Parkhi, A. Vedaldi, A. Zisserman, Deep face recognition, in: British Machine Vision Conference, vol. 1, 2015, pp. 1–12.
[6] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: closing the gap to human-level performance in face verification, in: CVPR, 2014, pp. 1701–1708.
[7] Y. Wen, L. Zhang, K.M. von Deneen, L. He, Face recognition using discriminative locality preserving vectors, Digit. Sign. Process. 50 (2016) 103–113.
[8] H. Méndez-Vázquez, Y. Martínez-Díaz, Z. Chai, Volume structured ordinal features with background similarity measure for video face recognition, in: ICB, 2013, pp. 1–6.
[9] H. Li, G. Hua, Hierarchical-pep model for real-world face recognition, in: CVPR, 2015, pp. 4055–4064.
[10] O.M. Parkhi, K. Simonyan, A. Vedaldi, A. Zisserman, A compact and discriminative face track descriptor, in: CVPR, 2014, pp. 1693–1700.
[11] J.-C. Chen, V.M. Patel, R. Chellappa, Landmark-based fisher vector representation for video-based face verification, in: ICIP, 2015, pp. 2705–2709.
[12] K. Simonyan, O.M. Parkhi, A. Vedaldi, A. Zisserman, Fisher vector faces in the wild, in: British Machine Vision Conference, 2013, pp. 1–13.
[13] Y. Martínez-Díaz, L. Chang, N. Hernández, H. Méndez-Vázquez, L.E. Sucar, Efficient video face recognition by using Fisher vector encoding of binary features, in: ICPR, 2016, pp. 1436–1441.
[14] M. Calonder, V. Lepetit, C. Strecha, P. Fua, Brief: binary robust independent elementary features, in: ECCV, 2010, pp. 778–792.
[15] S. Lazebnik, C. Schmid, J. Ponce, Beyond bag of features: spatial pyramid matching for recognizing natural scene categories, in: CVPR, 2006, pp. 2169–2178.
[16] J. Sánchez, F. Perronnin, T. de Campos, Modeling the spatial layout of images beyond spatial pyramids, Pattern Recogn. Lett. (2012) 2216–2223.
[17] A. Landgraf, Y. Lee, Dimensionality reduction for binary data through the projection of natural parameters. Available from: < arXiv:1510.06112 > .
[18] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: an efficient alternative to sift or surf, in: ICCV, 2011, pp. 2564–2571.
[19] S. Leutenegger, M. Chli, R.Y. Siegwart, Brisk: binary robust invariant scalable keypoints, in: ICCV, 2011, pp. 2548–2555.
[20] A. Alahi, R. Ortiz, P. Vandergheynst, Freak: fast retina keypoint, in: CVPR, 2012, pp. 510–517.
[21] J. Lankinen, V. Kangas, J.-K. Kamarainen, A comparison of local feature detectors and descriptors for visual object categorization by intra-class repeatability and matching, in: ICPR, 2012, pp. 780–783.
[22] M. Collins, S. Dasgupta, R.E. Schapire, A generalization of principal component analysis to the exponential family, in: Advances in Neural Information Processing Systems, 2001, pp. 617–624.
[23] A.I. Schein, L.K. Saul, L.H. Ungar, A generalized linear model for principal component analysis of binary data, in: Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics, 2003, pp. 14–21.
[24] J. de Leeuw, Principal component analysis of binary data by iterated singular value decomposition, Comput. Stat. Anal. 50 (1) (2006) 21–39.
[25] A.J. Landgraf, Generalized Principal Component Analysis: Dimensionality Reduction through the Projection of Natural Parameters, Ph.D. thesis, The Ohio State University, 2015.
[26] K. Pearson, On lines and planes of closest fit to systems of points in space, Lond. Edinburgh Dublin Philos. Mag. J. Sci. 2 (11) (1901) 559–572.
[27] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the fisher vector: theory and practice, Int. J. Comput. Vis. 105 (3) (2013) 222–245.
[28] D. Chen, X. Cao, L. Wang, F. Wen, J. Sun, Bayesian face revisited: a joint formulation, ECCV (2012) 566–579.
[29] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, in: CVPR, 2011, pp. 529–534.
[30] L. Wolf, N. Levy, The SVM-minus similarity score for video face recognition, in: CVPR, 2013, pp. 3523–3530.
[31] H. Li, G. Hua, Z. Lin, J. Brandt, J. Yang, Probabilistic elastic matching for pose variant face verification, in: CVPR, 2013, pp. 3499–3506.
[32] Z. Cui, W. Li, D. Xu, S. Shan, X. Chen, Fusing robust face region descriptors via multiple metric learning for face recognition in the wild, in: CVPR, 2013, pp. 3554–3561.
[33] J. Hu, J. Lu, Y.-P. Tan, Discriminative deep metric learning for face verification in the wild, in: CVPR, 2014, pp. 1875–1882.
[34] H. Li, G. Hua, X. Shen, Z.L. Lin, J. Brandt, Eigen-pep for video face recognition, in: ACCV, vol. 9005, 2014, pp. 17–33.
[35] Z. Huang, S. Shan, R. Wang, H. Zhang, S. Lao, A. Kuerban, X. Chen, A benchmark and comparative study of video-based face recognition on cox face database, IEEE Trans. Image Process. 24 (12) (2015) 5967–5981.
[36] Z. Huang, R. Wang, S. Shan, X. Chen, Learning euclidean-to-riemannian metric for point-to-set classification, in: CVPR, 2014, pp. 1677–1684.
[37] M.T. Harandi, C. Sanderson, S. Shirazi, B.C. Lovell, Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching, in: CVPR, 2011, pp. 2705–2712.
[38] L.S.D.R. Wang, H. Guo, Q. Dai, Covariance discriminative learning: a natural and efficient approach to image set classification, in: CVPR, 2012, pp. 2496–2503.
[39] C. Ding, D. Tao, Trunk-branch ensemble convolutional neural networks for video-based face recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2017).
[40] A. Vedaldi, B. Fulkerson, Vlfeat: an open and portable library of computer vision algorithms, in: 18th ACM International Conference on Multimedia, 2010, pp. 1469–1472.