Face Verification Using the LARK Representation

Hae Jong Seo, *Student Member, IEEE,* Peyman Milanfar, *Fellow, IEEE,*

Abstract—We present a novel face representation based on locally adaptive regression kernel (LARK) descriptors [1]. Our LARK descriptor measures a self-similarity based on "signal-induced distance" between a center pixel and surrounding pixels in a local neighborhood. By applying principal component analysis (PCA) and a logistic function to LARK consecutively, we develop a new binary-like face representation which achieves state of the art face verification performance on the challenging benchmark "Labeled Faces in the Wild" (LFW) dataset [2]. In the case where training data are available, we employ one-shot similarity (OSS) [3], [4] based on linear discriminant analysis (LDA) [5]. The proposed approach achieves state of the art performance on both the unsupervised setting and the image restrictive training setting (72.23% and 78.90% verification rates) respectively as a single descriptor representation, with no preprocessing step. As opposed to [4] which combined 30 distances to achieve 85.13%, we achieve comparable performance (85.1%) with only 14 distances while significantly reducing computational complexity.

Index Terms—Face Verification, Locally Adaptive Regression Kernels, Matrix Cosine Similarity, One-Shot Similarity, Labeled Faces in the Wild

1 Introduction

FACE recognition has been of great research interest [6], [7], [3], [8], [9], [10], [11], [12], [13] in recent years. Face recognition is mainly divided into two tasks: 1) face identification and 2) face verification. The goal of face identification is to place a given test face into one of several predefined sets in a database, whereas face verification is to determine if two face images belong to the same person. In general, the face verification task is more difficult than face identification because a global threshold is required to make a decision. There are also many papers on face detection such as [14], [15] and [16], which is considered as a pre-processing step for face recognition.

According to the face recognition grand challenge (FRGC) [17], face identification rates under well-constrained environments have been saturated (almost perfect with a small false alarm rate.) Nevertheless, face recognition in uncontrolled settings is still an open problem due to the large variations caused by different pose, lighting condition, facial expression, occlusion, misalignment, etc. With the advent of a standard benchmark dataset "Labeled Faces in the Wild (LFW) [2]", the face verification problem in unconstrained settings has recently attracted much research effort [3], [8], [9], [10], [18], [19], [12], [20], [13]. This challenging dataset contains a collection of annotated faces captured from news articles, and exhibits all the variations mentioned above. There are three evaluation protocols for this dataset: 1) the image *un*restricted training setting, 2) the image restricted training setting, and 3) the unsupervised (no training) setting.

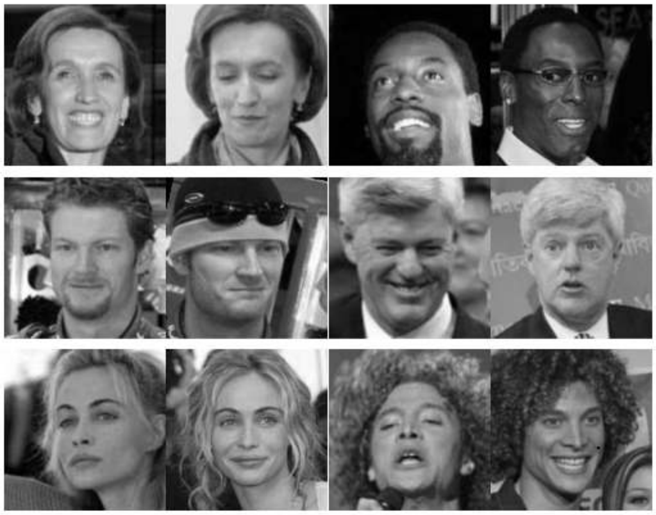

Fig. 1. Example faces from Labeled Faces in the Wild (LFW) [2]: faces belonging to the same person may look very different from each other due to the large variation caused by different pose, light condition, facial expression, and etc.

In this paper, we address the face verification problem in uncontrolled environments (on LFW dataset). The main task is to decide whether the images of two faces belong to the same individual. Among three evaluation settings, we focus on the last two (the unsupervised and the image restricted training) which are more realistic in practice. In our earlier work [1], we have tackled the generic object detection problem by employing local steering kernels (LSK) as visual descriptors, in conjunction with the matrix cosine similarity (MCS) measure.

- *H. J. Seo and P. Milanfar are with University of California, Santa Cruz, CA, 95064.*
E-mail: {rokaf,milanfar}@soe.ucsc.edu

This combination has led to state of the art detection performance from a single query, and without any further training. In fact, face detection is in nature very similar to face verification in the sense that both are binary classification problems and require two major components: 1) a face representation and 2) a similarity measure. In this paper, we provide some insights into how the face detection method in [1] can be extended to face verification.

Recently, face representation based on local image descriptors such as local binary pattern (LBP) and its variants [21], [3] and histogram of gradient descriptors (SIFT [22] and HOG [23]) have been proven to be effective for face verification. These descriptors encode local geometric structures by using either a quantized version of local gray level patterns or quantized codes of the image gradients. In this paper, we propose a more powerful alternative, namely, locally adaptive regression kernel (LARK) as a visual descriptor. LARK essentially measures a self-similarity based on *the geodesic distance* between a center pixel and surrounding pixels in a local neighborhood. LARK provides much more rich and detailed information than other local descriptors [23], [22], [24]. After reducing the dimension of LARK by performing PCA, we apply a logistic function to the result. The role of the logistic function here is to make LARK become more or less binarized by stretching values to extreme ends. We demonstrate that the use of matrix cosine similarity, combined with our feature representation results in the best performance in unsupervised settings of LFW benchmark.

For the image restricted setting, we employ one-shot similarity (OSS) measure [4] based on linear discriminative analysis (LDA). The OSS with the proposed feature representation achieves state of the art performance as a single descriptor and obtains results comparable with [4] when jointly used with other descriptors (with many fewer distances: 14 (ours) vs. 30 [4]). A block diagram of the proposed face verification system is given in Fig. 2.

The proposed face representations achieve not only state of the art performance on LFW dataset, but are also applicable to subspace learning based face recognition algorithms such as locality preserving projection (LPP), neighborhood preserving embedding (NPE), and spectral regression (SR) [25]. We demonstrate the effectiveness of the proposed representation with kernel spectral regression on FRGC 2.0 dataset in Section 3.5.

1.1 Related Work

1.1.1 Face Verification

A texture descriptor called local binary patterns (LBP) [21] has been shown to be effective for face recognition. Ever since LBP was introduced, such variants of LBP as three-patch LBP (TPLBP), and four-patch LBP (FPLBP) have been proposed by Wolf et al. [3]. These descriptors were combined with the one-shot similarity (OSS) [18] measure motivated by the growing body of "One-Shot Learning" techniques [26]. Wolf et al. [18] also applied the OSS in the framework of support vector machine (SVM) [5] by modifying the OSS to a conditional positive definite kernel. The OSS was extended to two-shot similarity (TSS) in [27]. In [3], [18], [4], it has been further shown that combining multiple descriptors and multiple measures can boost the overall verification performance.

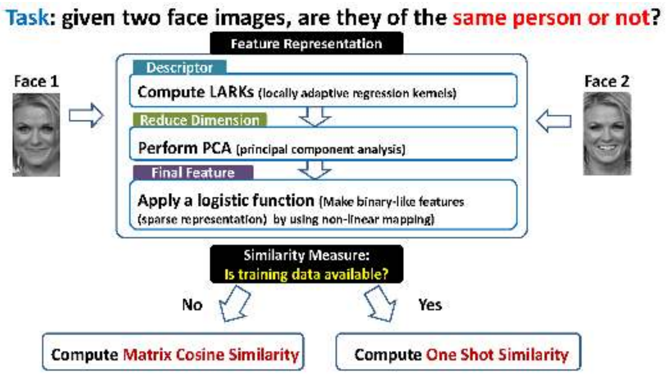
Task: given two face images, are they of the same person or not?
Feature Representation
Face 1
Face 2
Descriptor
Compute LARKs (locally adaptive regression kernels)
Reduce Dimension
Perform PCA (principal component analysis)
Final Feature
Apply a logistic function (Make binary-like features (sparse representation) by using non-linear mapping)
Similarity Measure: Is training data available?
No
Yes
Compute Matrix Cosine Similarity
Compute One Shot Similarity

Fig. 2. System block diagram. The system mainly consists of two stages: feature representation and similarity measure.

Guillaumin et al. [10] proposed two methods called 1) logistic discriminant metric learning and 2) marginalized k-nearest neighbor. They focused on finding a metric based on learning the Mahalanobis distance. Independently from [3], [18], [4], they also showed that combination of descriptors and metrics improves upon using only one metric and one descriptor.

Hua and Akbarzadeh [19] recently proposed an elastic and partial matching metric which robustly measures distance between two sets of descriptors by using a nonparametric significance test. In their work, they revealed that a simple difference of Gaussian (DoG) filtering on face images works better than the more often utilized photometric rectification methods such as self-quotient image [28] in handling lighting variations.

Motivated by the observation that humans perform very well on the LFW dataset, Kumar et al. [12] proposed two classifiers called "attribute" and "simile" classifiers for face verification. While the attribute classifiers are binary classifiers trained to recognize the presence or absence of visual aspects such as gender, race, age, and hair color, simile classifiers are binary classifiers trained to recognize the similarity of faces. This method achieved state of the art performance (85.29% verification rates on the LFW dataset), but requires a combination of many (more than 70) classifiers.

Distinguished from aforementioned works, Cao et al. [13] introduced a learning-based encoding method based on unsupervised learning techniques such as k-means, kd-tree, and random projection tree [29]. In [13], they focused on learning uniform descriptors from a collection of histogram-based low-level descriptors. They claimed that the uniformity of features is important when the L_2 or L_1 distances are used as similarity

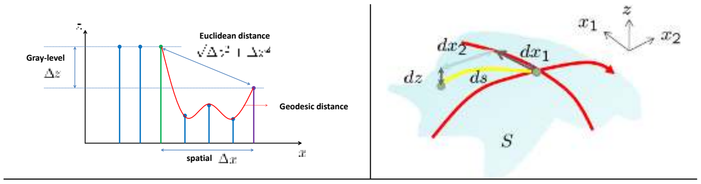

Fig. 3. Left: Euclidean distance vs. geodesic distance (the shortest path along the manifold) in 1-D signal. Right: the geodesic distance in 2-D surface can be computed as squared arclength and is given by $ds^2 = dx_1^2 + dx_2^2 + dz^2$.

metrics. By using multiple learning-based descriptors from nine fiducial areas and pose-adaptive matching, they have achieved a verification rate of 84.45% on the LFW dataset.

While all the works [3], [18], [4], [12], [13], [19] above evaluated their methods in the image restricted training setting, Ruiz-del-solar et al. [20] carried out a comprehensive study of existing image matching methods such as LBP matching, Gabor-Borda count, and SIFT matching in the unsupervised setting (without any training). They empirically compared these methods by changing the image crop size, parameter settings of descriptors, and image block size.

1.1.2 Training-free Face Detection

In our earlier work [1], we proposed a training-free non-parametric detection framework by employing local steering kernels (LSKs) as visual descriptors, in conjunction with the matrix cosine similarity (MCS) measure. By employing principal component analysis (PCA), LSKs were transformed to compact feature vectors. MCS between two matrices composed of resulting sets of feature vectors, $\mathbf{F}^{(i)} = [\mathbf{f}_1, \cdots, \mathbf{f}_n]^{(i)}, \mathbf{F}^{(j)} = [\mathbf{f}_1, \cdots, \mathbf{f}_n]^{(j)}$ is defined as:

$$\begin{aligned} \rho(\mathbf{F}^{(i)}, \mathbf{F}^{(j)}) &= \sum_{\ell=1}^{n} \frac{{\mathbf{f}_\ell^{(i)}}^\top \mathbf{f}_\ell^{(j)}}{\|\mathbf{F}^{(i)}\| \|\mathbf{F}^{(j)}\|}, \\ &= \sum_{\ell=1}^{n} \underbrace{\rho(\mathbf{f}_\ell^{(i)}, \mathbf{f}_\ell^{(j)})}_{\text{cosine similarity}} \underbrace{\frac{\|\mathbf{f}_\ell^{(i)}\| \|\mathbf{f}_\ell^{(j)}\|}{\|\mathbf{F}^{(i)}\| \|\mathbf{F}^{(j)}\|}}_{\text{relative weights}}, \quad (1) \end{aligned}$$

where n is the number of features. This MCS is a weighted sum of the cosine similarities of local features $\mathbf{f}_\ell$. The optimality properties of this approach were described in [1] within a naive Bayes framework. Robustness to local deformation, presence of noise, and occlusion is implicitly attained by the relative weights which play a key role in finding interest points in the face image (see Fig. 7). This is particularly useful for face detection where the goal is to separate faces from the background in a given image. In order for this framework to be extended to face verification, the role of the relative weights should be adjusted accordingly, as we will describe in Section 2.2.

1.2 Contributions of this paper

This paper presents a novel approach to the face verification problem with state-of-the art performance. We extend the face detection framework [1] so that it can be well adapted to the face verification task. This paper is distinguished from [1] in the sense that 1) we learn an overall, *fixed* PCA of general faces whereas PCA bases were learned from only one image in [1], 2) a novel binary-representation by using a logistic function is employed to adjust relative weights in MCS measure, 3) we also address a supervised setting where resulting representations can be useful in learning based framework.

Our contributions to the face verification task are three-fold. First, we introduce LARK which robustly captures local geometric structures. This LARK in conjunction with PCA provides a very compact face representation, desirable for real-time applications. Second, we extend face detection to face verification by introducing a binary-like face representation. The proposed representation along with both MCS and OSS achieves the best performance on the unsupervised and image restricted settings as a single descriptor. Lastly, we show that a very simple idea (namely the addition of a mirror image of a query) remarkably boosts the overall performance[1].

In the following sections, we first present the locally adaptive regression kernels (LARK) and the proposed binary-like face representation for face verification in the MCS measure in Section 2. We demonstrate comprehensive experimental results for both the unsupervised setting and the image restricted setting on the LFW dataset, and for FRGC 2.0 Experiment 4 in Section 3. We conclude the paper in Section 4.

2 A New Face Representation

As outlined in the previous section, our new face representation is derived in two stages: 1) computation of LARK and 2) producing binary-like features after dimension reduction. Below, we describe each of these steps in detail.

2.1 Locally Adaptive Regression Kernels (LARK)

LARK effectively and efficiently captures local geometric structure by taking advantage of self-similarity based on gradients. In order to measure the similarity of two pixels, in general, we can naturally consider both the spatial distance and the gray level distance (See Fig. 3 (left).) The most simple way to incorporate the two Δ's is the Euclidean distance between points. However, a much more effective way to combine the two Δ's is to define a "signal-induced" distance [31] which basically stands for a distance between the points measured along the

[1] This is a computational strategy that takes advantage of the fact that objects in our world frequently present mirror-symmetric views [30]

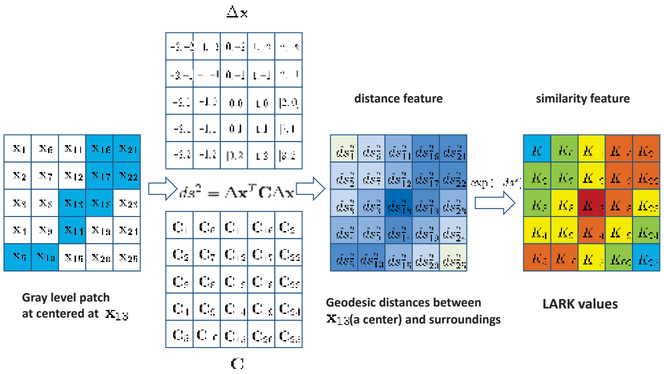

Fig. 4. How to compute LARK (5 × 5) values centered at $\mathbf{x}_{13}$. First of all, geodesic distance (middle) between $\mathbf{x}_{13}$ and surrounding pixels are computed and transformed to a similarity (right). The darker blue colors (middle), the smaller distances are. The red color (right) means higher similarity whereas blue color represents smaller similarity.

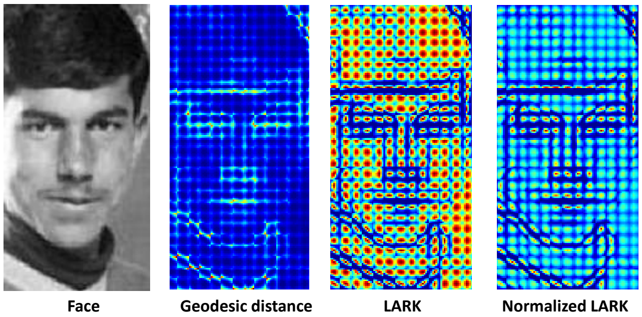

Fig. 5. Examples of geodesic distance, LARK, and normalized LARK. We show these in non-overlapping patches of a face image for a graphical purpose.

shortest path on the signal manifold (a.k.a. the geodesic distance).

Suppose that we consider the parameterized image surface $S(x_1, x_2) = \{x_1, x_2, z(x_1, x_2)\}$, embedded in the Euclidean space $\mathbb{R}^3$ as shown in Fig. 3 (right). The differential arclength on the surface is given by $ds^2 = dx_1^2 + dx_2^2 + dz^2$. Applying the chain rule, we have

$$dz(x_1, x_2) = \frac{\partial z}{\partial x_1} dx_1 + \frac{\partial z}{\partial x_2} dx_2 = z_{x_1} dx_1 + z_{x_2} dx_2, \qquad (2)$$

where z_{x_1}, z_{x_2} are first derivatives along x_1, x_2 respectively. Plugging $dz(x_1, x_2)$ into the arclength definition, we have

$$\begin{aligned} ds^2 &= dx_1^2 + dx_2^2 + dz^2, \\ &= dx_1^2 + dx_2^2 + (z_{x_1} dx_1 + z_{x_2} dx_2)^2, \\ &= (1 + z_{x_1}^2) dx_1^2 + 2 z_{x_1} z_{x_2} dx_1 dx_2 + (1 + z_{x_2}^2) dx_2^2, \\ &= [dx_1 dx_2] \begin{bmatrix} z_{x_1}^2 + 1 & z_{x_1} z_{x_2}, \\ z_{x_1} z_{x_2} & z_{x_2}^2 + 1 \end{bmatrix} \begin{bmatrix} dx_1 \\ dx_2 \end{bmatrix}, \\ &= \Delta\mathbf{x}^\top \mathbf{C} \Delta\mathbf{x} + \Delta\mathbf{x}^\top \Delta\mathbf{x}, \end{aligned} \qquad (3)$$

where $\Delta\mathbf{x} = [dx_1, dx_2]^\top$, and $\mathbf{C}$ is the local gradient covariance matrix (a.k.a. structure tensor).

We measure this arclength between a center pixel and surrounding pixels in a local window (see Fig. 4.) In a particular example of computing LARK of size 5 × 5 shown in Fig. 4, $\Delta\mathbf{x}_{13}$ is $[0, 0]^T$ since $\mathbf{x}_{13}$ is the center pixel. $\mathbf{C}_{13}$ is an average 2×2 covariance matrix computed from the patch Ω_{13} of size 5×5 centered at $\mathbf{x}_{13}$. The effect of $\Delta\mathbf{x}^\top \Delta\mathbf{x}$ in the local small window is trivial and data-independent, thus we only consider $ds^2 \approx \Delta\mathbf{x}^\top \mathbf{C} \Delta\mathbf{x}$.

We define LARK as a self-similarity between a center and its surroundings as follows:

$$K(\mathbf{C}_l, \Delta\mathbf{x}_l) = \exp\left(-ds^2\right) = \exp\left\{-\Delta\mathbf{x}_l^\top \mathbf{C}_l \Delta\mathbf{x}_l\right\}, \qquad (4)$$

where $l \in [1, \cdots, P]$, P is the total number of samples in a local analysis window around a sample position at the pixel of interest $\mathbf{x}$.

In theory, $\mathbf{C}_l$ is based on gradients (z_{x_1}, z_{x_2}) in one pixel. However, this $\mathbf{C}_l$ is unstable and prone to noise components in the data. Therefore, we use a collection of first derivatives of the visual signal $\mathbf{z}_l$, which contain the values of a patch Ω_l of pixels centered at position l, along spatial (x_1, x_2) axes. Then, the matrix $\mathbf{C}_l \in \mathbb{R}^{(2 \times 2)}$ can be written as follows:

$$\mathbf{C}_l = \sum_{m \in \Omega_l} \begin{bmatrix} z_{x_1}^2(m) & z_{x_1}(m) z_{x_2}(m) \\ z_{x_1}(m) z_{x_2}(m) & z_{x_2}^2(m) \end{bmatrix}. \qquad (5)$$

This can be interpreted as averaging geodesic distances in a patch to obtain a robust estimation even in the presence of noise and other perturbations.

Another key aspect of LARK lies in the fact that we implicitly smooth the image surface so that the local geodesic distance can be computed in a stable way. Specifically, we perform eigen-decomposition on the ("average") covariance matrix $\mathbf{C}_l$ as follows:

$$\mathbf{C}_l = \lambda_1 \mathbf{u}_1^\top \mathbf{u}_1 + \lambda_2 \mathbf{u}_2^\top \mathbf{u}_2 = s_1 s_2 \left(\frac{s_1}{s_2} \mathbf{u}_1^\top \mathbf{u}_1 + \frac{s_2}{s_1} \mathbf{u}_2^\top \mathbf{u}_2\right), \qquad (6)$$

where λ_1, λ_2 are eigenvalues[2], $\mathbf{u}_1, \mathbf{u}_2$ are eigenvectors, and $s_1 = \sqrt{\lambda_1}, s_2 = \sqrt{\lambda_2}$ are singular values.

Singular values s_1, s_2 are regularized to avoid numerical instabilities, while both eigenvectors remain the same. Namely,

$$\mathbf{C}_l^{reg} = (s_1 s_2 + \epsilon)^\alpha \left(\frac{s_1 + \tau}{s_2 + \tau} \mathbf{u}_1^\top \mathbf{u}_1 + \frac{s_2 + \tau}{s_1 + \tau} \mathbf{u}_2^\top \mathbf{u}_2\right), \qquad (7)$$

where ϵ, τ, α are set to $10^{-7}, 1, 0.5$ respectively, and they are fixed throughout the paper. This can be thought of a non-linear mapping to the eigenvalues in order to turn the structure tensor into a Riemannian metric [32].

LARK in (4) with $\mathbf{C}^{reg}$ is densely calculated from the image and normalized to a unit vector (i.e., a unit norm) to be more robust to illumination changes. We will call a collection of LARKs from an image $\mathbf{K} = [\mathbf{k}_1, \cdots, \mathbf{k}_n] \in$

[2] In practice, eigenvalues and eigenvectors of the covariance matrix can be efficiently computed in the closed form as follows: $\lambda_\pm = \frac{(C_{11}+C_{22}) \pm \sqrt{(C_{11}-C_{22})^2 + 4 C_{12} C_{21}}}{2}$, $\theta = \tan^{-1}\left(-\frac{C_{11}+C_{21}-\lambda_1}{C_{22}+C_{12}-\lambda_1}\right)$, $\mathbf{u}_1 = [\cos\theta, \sin\theta]^\top$, $\mathbf{u}_2 = [-\sin\theta, \cos\theta]^\top$.

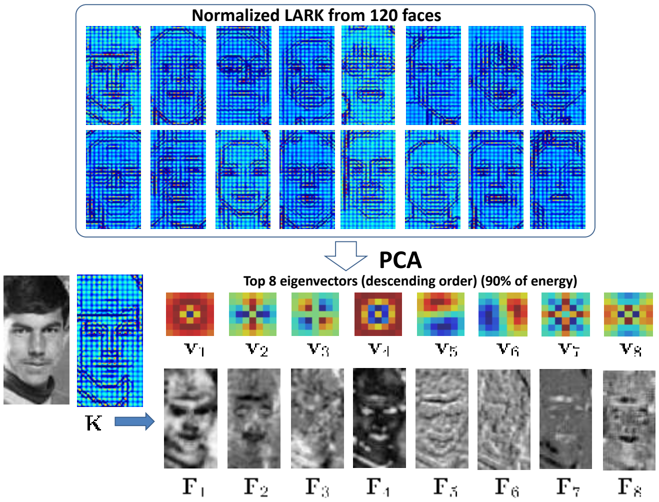
Normalized LARK from 120 faces
PCA
Top 8 eigenvectors (descending order) (90% of energy)
$\mathbf{v}_1$ $\mathbf{v}_2$ $\mathbf{v}_3$ $\mathbf{v}_4$ $\mathbf{v}_5$ $\mathbf{v}_6$ $\mathbf{v}_7$ $\mathbf{v}_8$
$\mathbf{K}$
$\mathbf{F}_1$ $\mathbf{F}_2$ $\mathbf{F}_3$ $\mathbf{F}_4$ $\mathbf{F}_5$ $\mathbf{F}_6$ $\mathbf{F}_7$ $\mathbf{F}_8$

Fig. 6. A collection of normalized LARK descriptors are very informative, but contains a redundant information as well. PCA is applied to not only reduce the dimensionality of LARK, but also to retain only the salient characteristics of the LARKs. After applying PCA to LARKs of size 7×7 collected from 120 face images, we obtained 8 eigenvectors corresponding to the top 8 eigenvalues which preserve 90 % of energy.

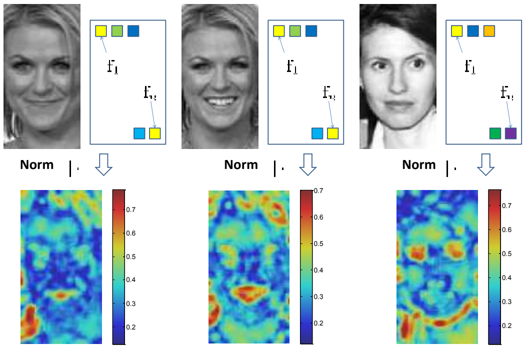

Fig. 7. $||\mathbf{f}||$: magnitude of $\mathbf{f}$ reveals interest points in the face images.

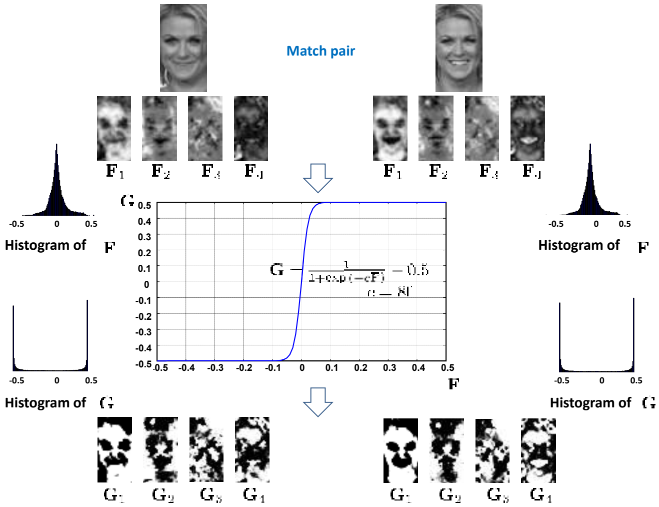
Match pair
$\mathbf{F}_1$ $\mathbf{F}_2$ $\mathbf{F}_3$ $\mathbf{F}_4$
Histogram of $\mathbf{F}$
Histogram of $\mathbf{G}$
-0.5 0 0.5
$\mathbf{G}$
$\mathbf{F}$
$\mathbf{G} = \frac{1}{1+\exp(-c\mathbf{F})} - 0.5$
$c = 80$
0.5 0.4 0.3 0.2 0.1 0 -0.1 -0.2 -0.3 -0.4 -0.5
-0.5 -0.4 -0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5
$\mathbf{G}_1$ $\mathbf{G}_2$ $\mathbf{G}_3$ $\mathbf{G}_4$

Fig. 8. We employ a logistic function $\frac{1}{1+\exp(-c\mathbf{F})} - 0.5$ in order to make binary-like representation.

$\mathbb{R}^{P\times n}$ where $\mathbf{k}$ is a vectorized version of K and n is the number of LARKs in the image (see Fig. 5.)

It is worth noting that LARK calculates similarity between oriented gradients without any quantization of gradients whereas HOG and SIFT use quantized values of oriented gradients. The quantization of oriented gradients, while useful in reducing computational complexity, can lead to a significant degradation in discriminative power of descriptors [33]. This effect is particularly severe in the case where there are no training samples available, which as we will show, is the case in the unsupervised settings of the LFW dataset in Section 3.2.

2.2 Face Verification by Binary-like Representation and MCS

These densely computed LARKs are highly informative, but taken together are be over-complete (redundant). Therefore, we derive features by applying dimensionality reduction (namely PCA) to $\mathbf{K}$, in order to retain only the salient characteristics of the LARKs. Applying PCA to $\mathbf{K}$ we can retain the top d principal components which form the columns of a matrix $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_d] \in \mathbb{R}^{P\times d}$. Since we focus on finding stable (but less specific) bases which can represent basic characteristics of general faces, we used LARKs collected from 120 face images to learn an overall, *fixed*, PCA basis for faces whereas the goal of [1] was to find similar images to a particular query, thus PCA bases were learned from only one image. (see Fig. 6.)

Typically, d is selected to be a small integer such as 7 or 8 so that 80 to 90% of the information in the LARKs would be retained. (i.e., $\frac{\sum_{i=1}^{d}\lambda_i}{\sum_{i=1}^{P}\lambda_i} \geq 0.8$ (to 0.9) where λ_i are the eigenvalues.) Next, the lower dimensional features are computed by projecting $\mathbf{K}$ onto $\mathbf{V}$ as follow:

$$\mathbf{F} = [\mathbf{f}_1, \cdots, \mathbf{f}_n] = \mathbf{V}^\top\mathbf{K} \in \mathbb{R}^{d\times n}. \qquad (8)$$

It is worth noting that the use of the PCA here is not critical in the sense that any unsupervised subspace learning method such as Kernel PCA, 2DPCA [34], [35], LLE [36], LPP [37] CDA [38], CEA [39], kd-tree, and random projection tree [29] can be used.

Denoting $\mathbf{F}$ as the feature representation, we measure the similarity score by MCS between two images. As shown in Fig. 7, $||\mathbf{f}||$ reveals interest points (e.g., eyes, mouth, hairs, jaws, etc.) in the face images. If the task is to detect faces from background, focusing on interest points would be helpful for robustness to occlusion, misalignment, pose, and local deformation. Even though these properties are also important for face verification, relying too much on these relative weights tends to weaken the ability to discriminate *between* two faces. In order to alleviate this problem, we need to make these weights relatively spread out (somewhat uniform). This can be realized by applying a nonlinear mapping to features $\mathbf{F}$. Specifically, we apply a logistic function element-by-element to the feature matrices $\mathbf{F}$ as follows:

$$\mathbf{G} = \frac{1}{1+\exp\left(-c\mathbf{F}\right)} - 0.5. \tag{9}$$

The role of this logistic function is to make the features ($\mathbf{F}$) become more or less binary-like by stretching values to extreme ends (-0.5,0.5). This nonlinear mapping also plays a role in making features sparse [40]. As shown in Fig. 8, histograms of $\mathbf{G}$ have two peaks around (-0.5,0.5) whereas histograms of $\mathbf{F}$ are centered around 0. After applying the logistic function, the dominance of large relative weights in $\mathbf{F}$ is removed and discriminative power of $\mathbf{G}$ compared to $\mathbf{F}$ is increased as shown in Fig. 9. It is worth noting that this idea is somewhat related to [13] in which the uniformity of histogram-based feature values is considered important, and [4] in which the Hellinger distance outperforms L_2 distance. That is, ensuring that feature values stay in a small range enhances the discriminative power of any distance measure.

As defined in (1), the MCS between two matrices $\mathbf{G}^{(i)} = [\mathbf{g}_1, \cdots, \mathbf{g}_n]^{(i)}, \mathbf{G}^{(j)} = [\mathbf{g}_1, \cdots, \mathbf{g}_n]^{(j)}$ is as follows:

$$\rho(\mathbf{G}^{(i)}, \mathbf{G}^{(j)}) = \sum_{\ell=1}^{n} \underbrace{\rho(\mathbf{g}_\ell^{(i)}, \mathbf{g}_\ell^{(j)})}_{\text{cosine similarity}} \underbrace{\frac{\|\mathbf{g}_\ell^{(i)}\|\|\mathbf{g}_\ell^{(j)}\|}{\|\mathbf{G}^{(i)}\|_F\|\mathbf{G}^{(j)}\|_F}}_{\text{relative weights}}. \tag{10}$$

This $\rho(\mathbf{G}^{(i)}, \mathbf{G}^{(j)})$ can be efficiently implemented by column-stacking the matrices $\mathbf{G}^{(i)}, \mathbf{G}^{(j)}$ and simply computing the cosine similarity between two long column vectors as follows:

$$\rho(i,j) = \rho(\mathrm{colstack}(\mathbf{G}^{(i)}), \mathrm{colstack}(\mathbf{G}^{(j)})) \in [-1,1], \tag{11}$$

where $\mathrm{colstack}(\cdot)$ means an operator which column-stacks (rasterizes) a matrix.

The MCS measure along with the $\mathbf{G}$ provides robustness to many small deformation, but tends to fail when there are large variations due to out-of-plane rotation, which is common in the LFW dataset. To deal with off-frontal (out-of-plane rotated) faces, we use a very simple (but novel) idea of additionally using mirror-reflect version of $\mathbf{G}$ (see Fig. 10.) We take a maximum value between the two resulting MCS scores as a final MCS score[3].

$$\mathrm{MCS}(i,j) = \max(\rho(\mathbf{G}^{(i)}, \mathbf{G}^{(j)}), \rho(\overline{\mathbf{G}}^{(i)}, \mathbf{G}^{(j)})), \tag{12}$$

where $\overline{\mathbf{G}}$ is a mirror-reflect version[4] of $\mathbf{G}$.

3 Experimental Results

Up to now, we have described the proposed face representation for the face verification task. In this section, we demonstrate the performance of the proposed method with comprehensive experiments on the challenging labeled faces in the wild (LFW) [7] dataset and FRGC 2.0 dataset [41].

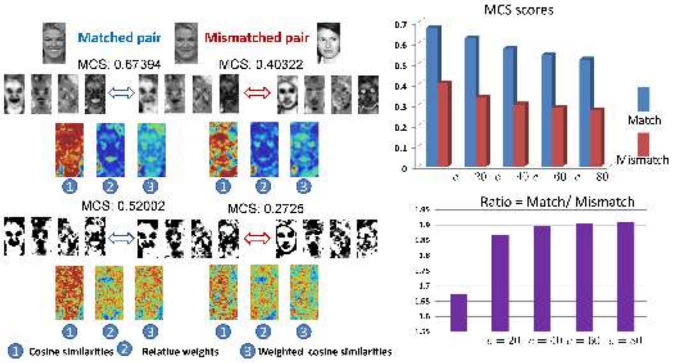

Fig. 9. MCS scores for a matched pair and a mismatched pair. Top-Left: MCS by using $\mathbf{F}$ 1) $\rho(\mathbf{f}_\ell^{(i)}, \mathbf{f}_\ell^{(j)})$, 2) $\frac{\|\mathbf{f}_\ell^{(i)}\|\|\mathbf{f}_\ell^{(j)}\|}{\|\mathbf{F}^{(i)}\|\|\mathbf{F}^{(j)}\|}$, 3) $\rho(\mathbf{f}_\ell^{(i)}, \mathbf{f}_\ell^{(j)})\frac{\|\mathbf{f}_\ell^{(i)}\|\|\mathbf{f}_\ell^{(j)}\|}{\|\mathbf{F}^{(i)}\|\|\mathbf{F}^{(j)}\|}$, Bottom-Left: MCS by using $\mathbf{G}$, 1) $\rho(\mathbf{g}_\ell^{(i)}, \mathbf{g}_\ell^{(j)})$, 2) $\frac{\|\mathbf{g}_\ell^{(i)}\|\|\mathbf{g}_\ell^{(j)}\|}{\|\mathbf{G}^{(i)}\|\|\mathbf{G}^{(j)}\|}$, 3) $\rho(\mathbf{g}_\ell^{(i)}, \mathbf{g}_\ell^{(j)})\frac{\|\mathbf{g}_\ell^{(i)}\|\|\mathbf{g}_\ell^{(j)}\|}{\|\mathbf{G}^{(i)}\|\|\mathbf{G}^{(j)}\|}$. The relative weights of $\mathbf{G}$ are more useful than those of $\mathbf{F}$ because they are not just focusing on texture region. Right: The effect of coefficient c in the logistic function. The ratio of MCS scores of match pair to those of mismatch pair is maximized at $c = 80$. Also see Fig. 12.

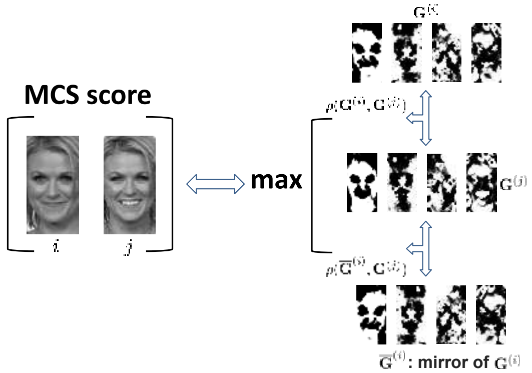
MCS score
i
j
max
$\mathbf{G}^{(i)}$
$\rho(\mathbf{G}^{(i)}, \mathbf{G}^{(j)})$
$\mathbf{G}^{(j)}$
$\rho(\overline{\mathbf{G}}^{(i)}, \mathbf{G}^{(j)})$
$\overline{\mathbf{G}}^{(i)}$: mirror of $\mathbf{G}^{(i)}$

Fig. 10. MCS score between face images (i) and (j) is a maximum score of two MCS scores computed between $\mathbf{G}^{(j)}$ and $\mathbf{G}^{(i)}$, and $\overline{\mathbf{G}}^{(i)}$: mirror-reflect version of $\mathbf{G}^{(i)}$ respectively.

3.1 Labeled Faces in the Wild (LFW) Dataset

The LFW database [7] consists of 13,233 face images of 5,749 different persons, obtained from news articles on the web. The images in the LFW database have a very large degree of variability in the facial expression, age, race, pose, occlusion, and illumination conditions (see Fig. 1). The task is to determine if a pair of face images belong to the same individual or not. We test on the "View 2" which includes 3,000 matched pairs and 3,000 mismatched pairs. The data are equally divided into 10 sets. The final verification performance is reported as the mean recognition rate and standard error about the mean over 10-fold cross-validation.

We also provide the receiver operating characteristic

[3] Interestingly, we found that this simple idea has not been utilized before, but remarkably boosts the overall performance.

[4] $\overline{\mathbf{G}}$ is not computed from mirror-reflected face images, but is the reflected version of $\mathbf{G}$. This helps us compute LARK features just once.

(ROC) curves for the sake of completeness. The true positive rate (TPR), the false positive rate (FPR), and the verification rate (VR) are defined as follows:

$$\begin{aligned}\text{TPR} &= \frac{\sharp \text{ correctly accepted matched pairs}}{\sharp \text{ total matched pairs}}, &(13)\\ \text{FPR} &= \frac{\sharp \text{ incorrectly accepted mismatched pairs}}{\sharp \text{ total mismatch pairs}}, &(14)\\ \text{VR} &= \frac{\sharp \text{ correctly classified pairs}}{\sharp \text{ total pairs}}. &(15)\end{aligned}$$

We compute the TPR and FPR by changing the threshold values to draw the ROC curves and report the best VR across the ROC curves.

As mentioned earlier, there are three evaluation settings : 1) the image unrestricted training setting, 2) the image restricted training setting, and 3) the unsupervised setting. The unsupervised setting is the most difficult one among these because there are no training examples available. On the other hand, the other two settings allow us to utilize available image pair information in the training set. The image unrestricted setting further provides the identity information of each pair. The official LFW website[5] provides all the state of the art results on the three settings.

In this paper, we only focus on the two most challenging settings: the unsupervised setting and the image restricted setting, because these scenarios are more realistic in practice. We use the aligned version of the LFW dataset available from the website[6] The images were cropped to a size of 184×97 so that images include more or less faces only[7].

3.2 Unsupervised Setting

In this section, we examine the efficacy of the proposed method in the unsupervised setting where we do not use any training examples. We compute LARKs (K) of size 7×7 (based on Ω_l of size 7×7) densely from each face image. We end up with features $\mathbf{G}$ by reducing dimensionality from 49 to 8 and employing a logistic function with $c = 80$ (performance converges as we increase c value (see Fig. 9)). The MCS score described in Fig. 10 is computed from each of 6,000 pairs. [20] conducted comprehensive experiments to find the best combination among various state of the art descriptors (i.e., LBP, PCALBP, Gabor jets, and SIFT) and similarity measures (i.e, histogram intersection, Chi-square, Borda count, and Euclidean distance). They reported that LBP with Chi-square achieves the best performance (69.45% VR). We computed TPR and FPR by changing the threshold to draw a ROC curve. The proposed method achieves (72.23% VR) and outperforms previous state of the art methods reported in the LFW website as shown in Fig. 11. Even before employing the logistic function, the proposed approach outperforms state of the art methods. We can see in Fig. 12 that the higher parameter c is, the better performance is. It clearly shows that binary-like features $\mathbf{G}$ are superior to the direct use of $\mathbf{F}$ for face verification task. We observe that there is no further improvement above $c = 80$. We also analyzed the effect of using mirror-reflect version of $\mathbf{G}$. This simple idea led to a nontrivial improvement ($1 \sim 2\%$) which is more pronounced in the range of smaller c. The overall performance of the proposed method is not a strong function of 1) LARK window size P and 2) LARK patch size $|\Omega_l|$ within a desirable range (for instance, $P = 7{\times}7 \sim 11{\times}11$) as shown in Table I. If we choose too large a window size, the geodesic distance we measure is not accurate. However, if we set the window size to be too small, we end up with insufficient local geometric information. In other words, local fiducial features such as eye, nose, and mouth will not be captured appropriately. Similarly, the patch size for computing average covariance matrix $\mathbf{C}_l$ should be set properly. The use of large patch size can over-smooth $\mathbf{C}_l$. Conversely, the use of excessively small patch size results in inaccurate estimation of $\mathbf{C}_l$ due to noise. We found that the size of patch Ω_l should be less than or equal to the window size P.

3.3 Image Restricted Setting

In this section, we deal with the case where there are training image pairs available. More specifically, in the training set, it is known whether an image pair belongs to the same person or not, while identity information is not used at all. We employ one-shot similarity (OSS) [4] based on linear discriminative analysis (LDA). We briefly review OSS and explain how we use the proposed feature representation in the OSS framework.

3.3.1 One Shot Similarity (OSS)

The key idea behind the OSS is to use negative examples. Suppose that there are two classes (positive (+) and negative (-)) and we have many negative examples while there is only one positive example. In binary LDA case, the goal is to find out a projection direction $\mathbf{w}$ which maximizes the Raleigh quotient:

$$\widehat{\mathbf{w}} = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^\top S_B \mathbf{w}}{\mathbf{w}^\top S_W \mathbf{w}}, \tag{16}$$

where S_B is the "between-class scatter matrix" and S_W is the "within-class scatter matrix." The definitions of the scatter matrices are as follows:

$$\begin{aligned}S_B &= (\mathbf{m}_+ - \mathbf{m}_-)(\mathbf{m}_+ - \mathbf{m}_-)^\top,\\ S_W &= S_+ + S_-,\\ S_k &= \sum_k (\mathbf{G}_k - \mathbf{m}_k)(\mathbf{G}_k - \mathbf{m}_k)^\top,\end{aligned}$$

[5] http://vis-www.cs.umass.edu/lfw/results.html

[6] http://www.openu.ac.il/home/hassner/data/lfwa/ and It is worth noting that Wolf et al. [4] used a commercial face alignment system based on localization of fiducial points. They reported that the aligned version significantly improved the performance of all the descriptors compared to both the standard and the funneled versions.

[7] A slight difference in crop size makes no difference for the overall performance. For example, consider $184{\times}97$ vs. $186{\times}94$. More detailed discussion about the choice of image crop size in LFW dataset can be found in [42].

TABLE 1
Verification rates as functions of parameters 1) window size P, 2) patch size $|\Omega_l|$

P	3	5	7	9	11	7	7	7	7	7
$\|\Omega_l\|$	7	7	7	7	7	3	5	7	9	11
Verification rate	0.6933	0.7052	0.7232	0.7223	**0.7238**	0.7228	0.7218	**0.7232**	0.7163	0.7125

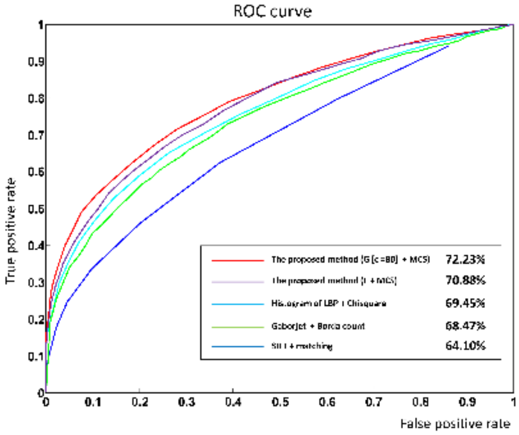

Fig. 11. ROC curves and VR computed from 10 folds of View 2 (the unsupervised setting). The proposed method performs the best among all.

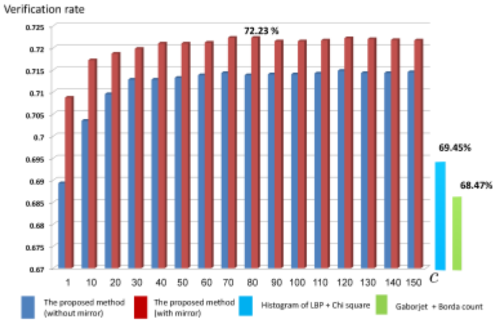

Fig. 12. Verification rates as a function of the parameter c in the proposed representation with comparison to state of the art methods in the unsupervised setting of the LFW dataset (view 2). Adding mirror-reflect improves the overall performance ($1 \sim 2\%$).

where $k \in \{+, -\}$ and $\mathbf{m}_+, \mathbf{m}_-$ are mean sample vectors of the positive class and the negative class respectively. It can be shown [5] that this maximization leads to a generalized eigenvalue problem:

$$S_B \mathbf{w} = \lambda S_W \mathbf{w}. \tag{17}$$

This problem can be solved very easily because $S_B\mathbf{w}$ is always in the direction of $(\mathbf{m}_+ - \mathbf{m}_-)$. Since there is only one positive example, that is, $S_+ = 0$, the within-class scatter matrix boils down to S_- which can be precalculated. $\mathbf{w}$ can be computed as follow:

$$\mathbf{w} \propto S_W^{-1}(\mathbf{m}_+ - \mathbf{m}_-) = S_-^{-1}(\mathbf{m}_+ - \mathbf{m}_-). \tag{18}$$

The benefit of using LDA is that the training step consists mainly of a vector difference followed by a matrix multiplication.

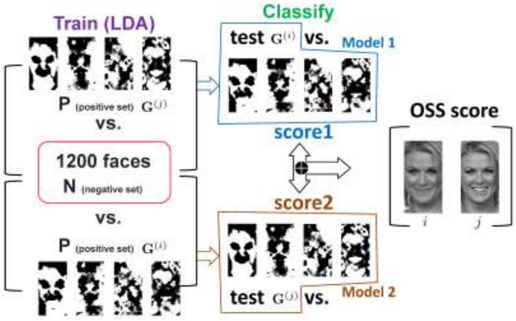

Fig. 13. The original OSS score between two images (i,j) is an average of two scores from model 1 and model 2. We use a negative set composed of 1,200 faces which are exclusive to the test image pairs.

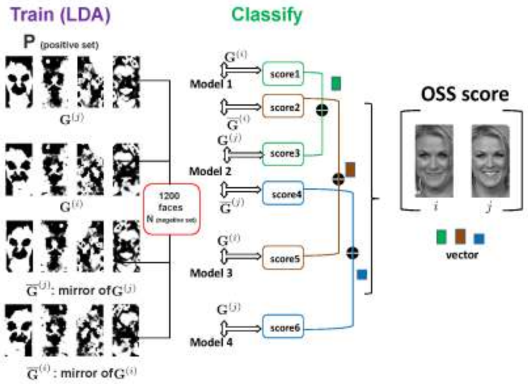

Fig. 14. We have 4 models that are learned from $\mathbf{G}^{(i)}$, $\mathbf{G}^{(j)}$, $\overline{\mathbf{G}}^{(i)}$, $\overline{\mathbf{G}}^{(j)}$ respectively. In this case, the OSS score is not a scalar, but a vector of three elements.

Fig. 13 describes how an OSS score between a pair of images (i) and (j) is computed as given in [4]. As negative examples, we used 1,200 faces which are not included in the test image pairs. First, by learning model 1 between (j) and negative set (N) and classifying (i) based on model 1, we obtain a score 1. Then we switch the role of (i) and (j) and learn model 2 and classify (j) on model 2 in order to get a score 2. The final score is an average of these two scores. We used the Matlab code available from the website[8].

3.3.2 One Shot Similarity (OSS) with the Proposed Representation

We use the same parameters for binary-like face representation as the ones explaned in Section 3.2. By using the mirror-reflect version of $\mathbf{G}^{(i)}, \mathbf{G}^{(j)}$, we construct 4 models instead of 2 models and obtain three scores

[8] http://www.openu.ac.il/home/hassner/projects/Ossk/

TABLE 2
Test set, Negative set, and Training sets in 10-fold validation (view 2)

Fold	1	2	3	4	5	6	7	8	9	10
Test set	3∼10	1, 4∼10	1,2, 5∼10	1∼3, 6∼10	1∼4, 7∼10	1∼5, 8∼10	1∼6, 9∼10	1∼7, 10	1∼8	2∼9
Negative set	2	3	4	5	6	7	8	9	10	1
Train set	1	2	3	4	5	6	7	8	9	10

TABLE 3
Mean verification rates (10-fold) on the LFW dataset (view 2) . sqrt means $\sqrt{\text{descriptors}}$ which is Hellinger distance and Mirror means that mirror-reflect version is added.

Descriptors	L2 distance	L2 + sqrt	MCS	MCS + sqrt	OSS	OSS+ sqrt
LBP	67.86%	68.53%	67.98%	68.18%	74.48%	74.41%
LBP (Mirror)	68.33%	69.08%	71.0%	67.61%	75.65%	76.05%
TPLBP	68.28%	68.78%	68.35%	67.76%	74.7%	74.58%
TPLBP (Mirror)	68.98%	69.38%	71.66%	68.6%	77.08%	76.1%
SIFT	71.01%	71.05%	70.65%	70.96%	73.13%	76.4%
SIFT (Mirror)	71.3%	71.08%	71.26%	71.3%	73.2%	78.2%
	L2 distance	L2 + logistic(c=80)	MCS	MCS + logistic(c=80)	OSS	OSS+ logistic(c=80)
Ours	65.81%	70.98%	68.25%	71.08%	75.81%	76.45%
Ours (Mirror)	66.28%	**73.23**%	71.26%	**73.3**%	76.38%	**78.9**%

TABLE 4
Mean verification rates (10-fold) on the LFW dataset (view 2) . standard set VS. aligned set

Dataset	**standard**	**aligned**	**standard**	**aligned**
Descriptor	MCS + logistic(c=80)	MCS + logistic(c=80)	OSS+ logistic(c=80)	OSS+ logistic(c=80)
Ours	66.65%	71.08%	70.97%	76.45%
Ours (Mirror)	68.35%	**73.3**%	71.96%	**78.9**%

TABLE 5
Mean verification rate (10-fold) comparison between [4] and our best result. TSS means the two shot similarity [4]. Numbers mean the number of descriptors used.

Method	L_2	L_2 + sqrt	TSS	TSS + sqrt	OSS	OSS + sqrt
Wolf et al. [4] (30) 85.13 ±0.37%	LBP, Gabor FPLBP, TPLBP SIFT (5)	LBP, Gabor FPLBP, TPLBP SIFT (5)	LBP, Gabor FPLBP, TPLBP SIFT (5)	LBP, Gabor FPLBP, TPLBP SIFT (5)	LBP, Gabor, FPLBP, TPLBP SIFT (5)	LBP, Gabor FPLBP, TPLBP SIFT (5)
Method	L2 distance	L2 + logistic(c=80)	MCS	MCS + logistic(c=80)	OSS	OSS+ logistic(c=80)
Ours (14) 85.10 ±0.59%	(0)	(0)	TPLBP (3) SIFT, pcaLARK	TPLBP (3) SIFT, pcaLARK	LBP, TPLBP SIFT, pcaLARK (4)	LBP, TPLBP SIFT, pcaLARK (4)

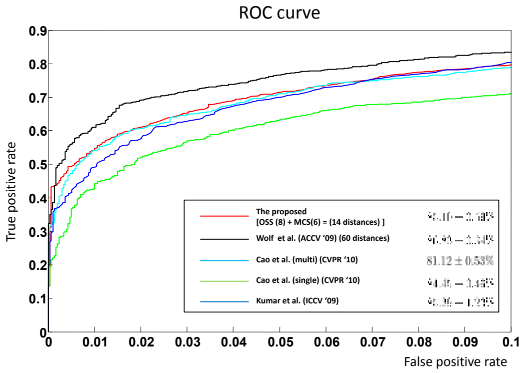

Fig. 15. ROC curves and VR averaged over 10 folds of View 2. We achieve state of the arts performance with the much less number of distances than 30 distances in [4].

instead of a single score (see Fig. 14). We treat these three scores as a single vector and feed these vectors into a support vector machine (SVM) [5]. As shown in Table 2, we use (1 set) as a negative set, train the support vector machine (SVM) with 4,800 (8 sets) OSS scores, and test 600 OSS scores (1 set).

We compare our feature representation with state of the art descriptors such as TPLBP[9], LBP[10], and SIFT[11]. The parameters of all descriptors were copied from [4]. We used either the descriptor vectors or their square roots (i.e., the Hellinger distance) for other descriptors. L_2 distance and MCS were also shown in Tables 3 for a comparison. Table 4 demonstrates that verification rates on the aligned version are much higher than ones on the standard set, which is consistent with the results from [4].

Consistent with the unsupervised setting in Fig. 12, the use of mirror-reflect lead to $1 \sim 3\%$ improvement to all descriptors as described in Table 3. As we can see from Table 3, the proposed representation ($c = 80$) outperforms all the other (single) descriptors when used with OSS. Consistent with the results in the previous section (unsupervised setting), addition of mirror-reflect boosts the overall performance as well. Wolf et al. [4] reported that they achieve 85.13% with a total of 30

[9] http://www.openu.ac.il/home/hassner/projects/Patchlbp/
[10] http://www.ee.oulu.fi/research/imag/texture/download/lbp.m
[11] http://people.csail.mit.edu/ceilu/ECCV2008

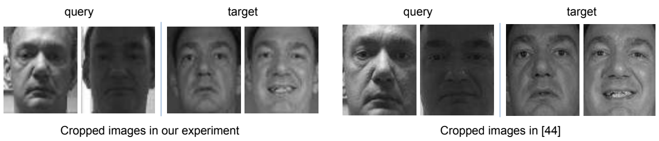

Fig. 16. Cropped images in our FRGC experiments contain background around hair and neck while those in [44] only have facial components.

distances, but we are able to get the same performance with only 14 distances (vectors)(see Table 5.)

3.4 Discussion

It is worth noting that state of the art descriptors such as LBP, TPLBP, and SIFT use preprocessing steps as suggested in [4]. Accordingly, they applied a noise-removal filter (Matlab's wiener2 function) to the cropped images and saturated 1% of values at the low and high intensities for these descriptors. After computing descriptors from preprocessed images, descriptors were normalized to unit length. Then, these values are truncated at 0.2 and once again normalized to unit length. On the other hand, the proposed LARK descriptor does not require any preprocessing steps and is directly normalized to a unit vector. We acknowledge that recognition rates of LBP, TPLBP, and SIFT in Table 3 do not coincide with ones in [4]. This slight difference may come from the image crop size, the sizes of the blocks, and how they are distributed within the crop size. However, we believe that the results shown in Table 3 in the same image crop size are a fair comparison because we followed the optimal parameter settings the authors reported.

The MCS (0.01 sec per pair) and OSS (0.37 sec per pair: Matlab implementation on Intel Pentium CPU 2.66 Ghz machine) in conjunction with the proposed features is computationally efficient. Since the proposed method is based on a fixed set of bases, the extension of this methods to a large-scale face dataset would be straightforward. To this end, we could benefit from an efficient searching method (coarse-to-fine search) and/or a fast nearest neighbor search method (e.g., vantage point tree [12] and kernelized locality-sensitive hashing [43].)

3.5 FRGC 2.0 Experiment 4

In this section, we evaluate the proposed method on the experiment 4 in the Face Recognition Grand Challenge (FRGC) v2.0 [17]. The training set for this experiment consists of 12,766 images from 222 individuals while there are 8,014 query images (from the uncontrolled setting) and 16,028 target images (from the uncontrolled setting) out of 466 subjects that are exclusive to the training set (see Fig. 16.). There are three types of ROC curves: ROC-I, ROC-II, and ROC-III, corresponding to images collected within a semester, within a year, and between semesters respectively. In this paper, we report results only for ROC-III which is the most challenging.

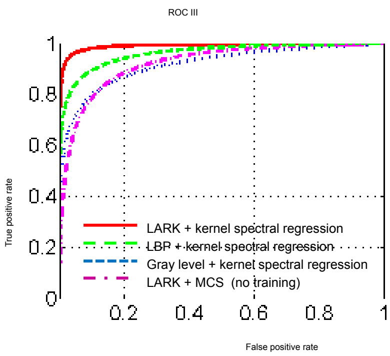

Fig. 17. FRGC 2.0 Experiment 4 face verification performance (ROC-III curves) for LARK, LBP, and gray level value in the kernel spectral regression framework. For a comparison, we inserted the performance of LARK without any training.

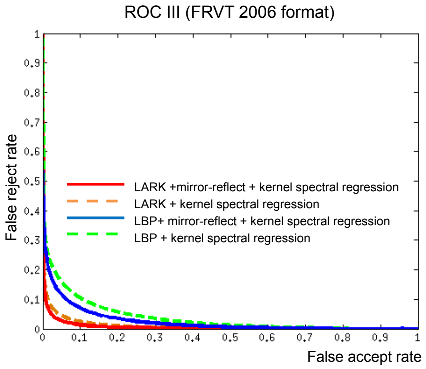

Fig. 18. Adding mirror-reflect versions to the system improves the overall face verification performance. FRVT 2006 report format version of results (false reject rate vs. false accept rate).

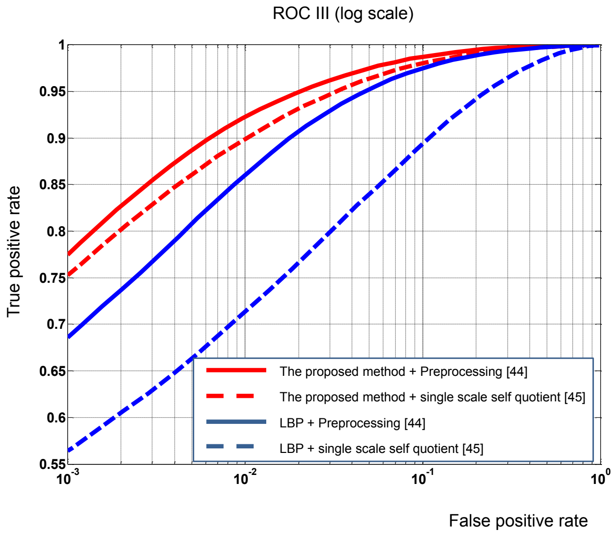

Fig. 19. State of the art preprocessing method [44] further improves the performance of the proposed method and LBP, but the proposed method still outperforms LBP.

The FRGC dataset provides fiducial feature points such as eyes, nose, and mouth positions. We cropped the face images by using given eye labels and resized the images to the size of 112×96 pixels. In order to alleviate a large illumination variation contained in FRGC dataset, we applied an illumination normalization method, single scale self quotient image [45] as a preprocessing. Differently from LFW image restricted setting, we do not have image pairs as a training set in FRGC 2.0 experiment 4. Instead, we have 12,766 individual face images for the training set. In order to take advantage of the huge number of training examples, we employed an efficient regularized subspace learning method called kernel spectral regression[12] [25]. The kernel spectral regression casts the problem of learning the projective functions into a regression framework, which avoids eigen-decomposition of dense matrices. The kernel spectral regression is fast because the regression framework reduces the complexity of the optimization problem of the linear graph embedding from cubic-time to linear-time. In order to show superiority of LARK[13] to LBP[14], we applied the kernel spectral regression to LARK, LBP, and gray level values separately. By projecting features (LARK, LBP, gray level values) computed from the query set and target set to the corresponding subspaces, we generated similarity matrices by using the cosine similarity. We computed the TPR and FPR by changing the threshold values to draw the ROC curves. Fig. 17 shows that LARK outperforms LBP and gray level values in the framework of kernel spectral regression. More specifically, LARK achieves 75.6 % TPR at 0.1 % FPR whereas LBP and gray pixel values obtain 57 % TPR and 39 % TPR at 0.1 % FPR respectively. For a comparison, we also show the performance of LARK without any training process. Fig. 18 demonstrates that adding mirror-reflect version[15] improves overall performance of LARK and LBP respectively in terms of false rejection rate vs. false accept rate curves by following FRVT 2006 report format[16]. Fig. 19 shows that with state of the art preprocessing method [44], both our method and LBP[17] achieve better performance than those with single scale self quotient image [45]. However, the proposed method still outperforms LBP. In order to be consistent with cropped regions in LFW dataset (see Fig. 2), we have included neck and hair region in FRGC dataset (see Fig. 16). We believe that this led to a huge performance gap between state of the art methods from FRVT 2006 and the proposed method. We expect the proposed method to perform better if we register faces after manually cropping and ensuring that only facial components are present. However, the point was that our method can perform reasonably well even when the face alignment is not perfect.

4 Conclusion and Future Work

In this paper, we have proposed a novel binary-like face representation for the face verification task. The LARK is derived from the geodesic distance and robustly captures underlying geometric structure even in the presence of noise. In order to make LARK as compact as possible and adapted to face verification, we developed a binary-like representation by applying PCA to LARK to develop a *fixed* basis, followed by a logistic function. Experiments on the LFW dataset, a challenging database of real-world human faces, demonstrated that the proposed method yields state of the art results in the unsupervised setting.

Since the proposed feature representation does not involve any quantization in the computation of descriptors (unlike LBP, TPLBP, and SIFT,) LARK conveys more rich and discriminative power than other descriptors. In the image restricted setting, we took advantage of SVM learning by employing the OSS and utilizing negative sets. The proposed approach with OSS achieves a high recognition accuracy and improves upon other state-of-the-art descriptors. A simple idea of additionally using mirror-reflect version of images led to a significant improvement (on average 2%). As opposed to [4] which used 30 distances to reach 85% accuracy, we were able to achieve the same performance with just 14 distances. It would be interesting to see how far the performance can be improved if we involve the two-shot similarity (TSS) and the OSS based on the SVM (not LDA) which are claimed to further improve the overall performance in [4]. In the FRGC 2.0 Experiment 4, we employed kernel spectral regression [25] to learn a subspace from the training set and show that the proposed LARK in conjunction with the use of mirror reflection consistently outperforms other methods.

We have shown that face verification is a problem akin to face detection. We believe that these problem can be dealt with in a unified framework. In our earlier detection work [1], we assumed that there is no training available. We can benefit from the idea of the OSS which utilizes a negative set in order to increase detection accuracy as well. These aspects of the work are the subject of future research.

[12] Downloadable from http://www.zjucadcg.cn/dengcai/Data/code/KSR.m. The parameters for KSR are set as follows: ReguAlpha = 0.1, ReguType = Ridge, KernelType = Gaussian, t = 5.

[13] We used the same parameters for LARK as in LFW dataset in previous section.

[14] Downloadable from http://www.ee.oulu.fi/research/imag/texture/image_data/__matlab.html. Please note that we consistently used the same LBP implementation code as done in experiments on LFW dataset. As far as the number of non-overlapping regions of LBP is concerned, we used the same setting (35 non-overlapping blocks) as suggested in Wolf et al.[3]

[15] Faces are generally not symmetric. Face asymmetry has been studied in [46]. Their results supported previous work in Psychology that facial asymmetry contributes to human identification. This also justifies the idea of mirror-reflection that improves overall performance.

[16] http://www.frvt.org/FRVT2006/docs/FRVT2006andICE2006LargeScaleReport.pdf

[17] We used 256 overlapping blocks of LBP instead of 35 non-overlapping blocks.

ACKNOWLEDGMENTS

This work was supported by AFOSR Grant FA 9550-07-01-0365 and NSF Grant CCF-1016018.

REFERENCES

[1] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, Sep 2010.

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

[3] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Faces in Real-Life Image Workshop in European Conference on Computer Vision (ECCV)*, 2008.

[4] ——, "Similarity scores based on background samples," in *Asian Conference on Computer Vision (ACCV)*, 2009.

[5] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd Edition*. New York: John Wiley and Sons Inc, 2000.

[6] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *IEEE Conference on Computer Vision and Computer Vision (CVPR)*, 1991.

[7] G. B. Huang, V. Jain, and E. Leonard-Miller, "Unsupervised joint alignment of complex images," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.

[8] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *International Conference on Biometric (ICB)*, 2009.

[9] N. Pinto, J. J. Dicarlo, and D. D. Cox, "How far can you get with a modern face recognition test set using only simple features?" in *IEEE Conference on Computer Vision and Computer Vision (CVPR)*, 2009.

[10] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[11] Y. Taigman, L. Wolf, and T. Hassner, "Multiple one-shots for utilizing class lable information," in *British Machine Vision Conference (BMVC)*, 2009.

[12] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayer, "Attribute and simile and classifiers for face verification," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[13] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *IEEE Conference on Computer Vision and Computer Vision (CVPR)*, 2010.

[14] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision (IJCV)*, vol. 57, no. 2, pp. 137–154, 2004.

[15] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 20, no. 1, pp. 23–36, 1998.

[16] M. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 24, no. 1, pp. 34–58, 2002.

[17] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, C. Jin, K. Hoffman, J. Marques, M. Jaesik, and W. Worek, "Overview of the face recognition grand challenge," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[18] L. Wolf, T. Hassner, and Y. Taigman, "The one-shot similarity kernel," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[19] G. Hua and A. Akbarzadeh, "A robust elastic and partial matching metric for face recognition," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[20] J. R. del Solar, R. Verscheae, and M. Correa, "Recognition of faces in unconstrained enviornments: A comparative study," in *EURASIP Journal on Advances in Signal Processing (Recent Advances in Biometric Systems: A Signal Processing Perspective), Vol. 2009, Article ID 184617, 19 pages*.

[21] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution grayscale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 24, no. 7, pp. 971–987, 2002.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision (IJCV)*, vol. 20, pp. 91–110, 2004.

[23] N. Dalal and B. Triggs, "Histogram of oriented gradietns for human detection," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[24] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[25] D. Cai, X. He, and J. Han, "Spectral regression for efficient regularized subspace learning," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.

[26] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 28, no. 4, pp. 594–611, 2006.

[27] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix," *Annals of Mathematical Statistics (Abstract)*, vol. 20, p. 621, 1949.

[28] H. Wang, S. Z. Li, and Y. Wang, "Generalized quotient image," *IEEE Conference on Computer Vision and Computer Vision (CVPR)*, 2004.

[29] Y. Freund, S. Dasgupta, M. Kabra, and N. Verma, "Learning the structure of manifolds using random projections," *In Advances in Neural Information Processing Systems (NIPS)*, 2007.

[30] C. E. Connor, "A new viewpoint on faces," *Science*, vol. 330, pp. 764–765, 2010.

[31] R. Kimmel, *Numerical Geometry of Images*. Springer, 2003.

[32] G. Peyre and L. Cohen, "Geodesic methods for shape and surface processing," *In Advances in Computational Vision and Medical Image Processing: Methods and Applications (springer)*, vol. 13, pp. 29–56, 2008.

[33] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[34] J. Yang, D. Zhang, A. F. Frangi, and J. Yang, "Two-dimensional pca:a new approach to appearance-based face representation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 26, pp. 131–137, 2004.

[35] J. Meng and W. Zhang, "Volume measure in 2dpca-based face recognition," *Pattern Recognition Letters*, vol. 28, pp. 1203–1208, 2007.

[36] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[37] X. He, S. Yan, Y.Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Transacions on Pattern Analaysis and Machine Intelligence (PAMI)*, vol. 27, no. 3, pp. 328–340, 2005.

[38] Y. Ma, S. Lao, E. Takikawa, and M. Kawade, "Discriminant analysis in correlation similarity measure space," *IEEE International Conference on Machine Learning (ICML)*, 2007.

[39] Y. Fu, S. Yan, and T. S. Huang, "Correlation metric for generalized feature extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 30, no. 12, pp. 2229–2235, 2008.

[40] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun, "Learning invariant features through topographic filter maps," *IEEE Conference on Computer Vision and Computer Vision (CVPR)*, 2009.

[41] P. J. Philips, P. J. Flynn, T. Scruggs, K. W. Bower, and W. Worek, "Preliminary face recognition grand challenge results," *In IEEE Conference in Automatic Face and Gesture Recognition*, 2006.

[42] J. R. del Solar, R. Verschae, and M. Correa, "Recognition of faces in unconstrained environments: A comparative study." *EURASIP Journal on Advances in Signal Processing (Recent Advances in Biometric Systems: A Signal Processing Perspective)Volume 2009 (2009), Article ID 184617, 19 pages, doi:10.1155/2009/184617*.

[43] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[44] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing (TIP)*, vol. 19, no. 6, pp. 1635 – 1650, June 2010.

[45] V. Struc, "Inface: A toolbox for illumination invariant face recognition," in *University of Ljubljana*, 2009.

[46] Y. Liu, K. L. Schmidt, J. F. Cohn, and S. Mitra, "Facial asymmetry quantification for expression invariant human identification." *Computer Vision and Image Understanding*, vol. 91, pp. 138–159, 2003.