Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions

Xiaoyang Tan, William Triggs

Enhanced Local Texture Feature Sets for Face Recognition under Difficult Lighting Conditions

Xiaoyang Tan and Bill Triggs

Abstract— **Making recognition more reliable under uncontrolled lighting conditions is one of the most important challenges for practical face recognition systems. We tackle this by combining the strengths of robust illumination normalization, local texture based face representations, distance transform based matching, kernel-based feature extraction and multiple feature fusion. Specifically, we make three main contributions: (*i*) we present a simple and efficient preprocessing chain that eliminates most of the effects of changing illumination while still preserving the essential appearance details that are needed for recognition; (*ii*) we introduce Local Ternary Patterns (LTP), a generalization of the Local Binary Pattern (LBP) local texture descriptor that is more discriminant and less sensitive to noise in uniform regions, and we show that replacing comparisons based on local spatial histograms with a distance transform based similarity metric further improves the performance of LBP/LTP based face recognition; and (*iii*) we further improve robustness by adding Kernel PCA feature extraction and incorporating rich local appearance cues from two complementary sources – Gabor wavelets and LBP – showing that the combination is considerably more accurate than either feature set alone. The resulting method provides state-of-the-art performance on three data sets that are widely used for testing recognition under difficult illumination conditions: Extended Yale-B, CAS-PEAL-R1, and Face Recognition Grand Challenge version 2 experiment 4 (FRGC-204). For example, on the challenging FRGC-204 data set it halves the error rate relative to previously published methods, achieving a Face Verification Rate of 88.1% at 0.1% False Accept Rate. Further experiments show that our preprocessing method outperforms several existing preprocessors for a range of feature sets, data sets and lighting conditions.**

I. Introduction

Face recognition has received a great deal of attention from the scientific and industrial communities over the past several decades owing to its wide range of applications in information security and access control, law enforcement, surveillance and more generally image understanding [53]. Numerous approaches have been proposed, including (among many others) eigenfaces [37,43], fisherfaces [5] and laplacianfaces [15], nearest feature line-based subspace analysis [32], neural networks [22,38], elastic bunch graph matching [46], wavelets [46], and kernel methods [48]. Most of these methods were initially developed with face images collected under relatively well controlled conditions and in practice they have difficulty in dealing with the range of appearance variations that commonly occur in unconstrained natural images due to illumination, pose, facial expression, ageing, partial occlusions, *etc*.

This paper focuses mainly on the issue of robustness to lighting variations. For example, a face verification system for a portable device should be able to verify a client at any time (day or night) and in any place (indoors or outdoors). Unfortunately, facial appearance depends strongly on the ambient lighting and – as emphasized by the recent FRVT and FRGC trials [33] – this remains one of the major challenges for current face recognition systems. Traditional approaches for dealing with this issue can be broadly classified into three categories: appearance-based, normalization-based, and feature-based methods. In direct appearance-based approaches, training examples are collected under different lighting conditions and directly (*i.e*. without undergoing any lighting preprocessing) used to learn a global model of the possible illumination variations, for example a linear subspace or manifold model, which then generalizes to the variations seen in new images [6,4,23,9,50]. Direct learning of this kind makes few assumptions but it requires a large number of training images and an expressive feature set, otherwise it is essential to include a good preprocessor to reduce illumination variations (*c.f*. Fig.16).

Normalization based approaches seek to reduce the image to a more "canonical" form in which the illumination variations are suppressed. Histogram equalization is one simple example, but purpose-designed methods often exploit the fact that (on the scale of a face) naturally-occuring incoming illumination distributions typically have predominantly low spatial frequencies and soft edges so that high frequency information in the image is predominantly signal (*i.e*. intrinsic facial appearance). For example, the Multiscale Retinex (MSR) method of Jobson *et al*. [19] cancels much of the low frequency information by dividing the image by a smoothed version of itself. Wang *et al*. [44] use a similar idea (with a different local filter) in the Self Quotient Image (SQI). More recently, Chen *et al*. [10] improved SQI by using Logarithmic Total Variation (LTV) smoothing, and Gross & Brajovic (GB) [13] developed an anisotropic smoothing method that relies on the iterative estimation of a blurred version of the original image. These methods are quite effective but their ability to handle spatially non-uniform variations remains limited. Shan *et al*. [35] and Short *et al*. [36] gave comparative results for these and related methods.

The third approach extracts illumination-insensitive feature sets [8,1,2,3,46,14] directly from the given image. These

Xiaoyang Tan is with the Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, P.R. China. Bill Triggs is with the CNRS and Laboratoire Jean Kuntzmann, BP 53, 38041 Grenoble Cedex 9, France. The work was financed by the European Union research project CLASS and the National Science Foundation of China (60773060). Part of it was undertaken at INRIA Grenoble. Corresponding author: Xiaoyang Tan (x.tan@nuaa.edu.cn).

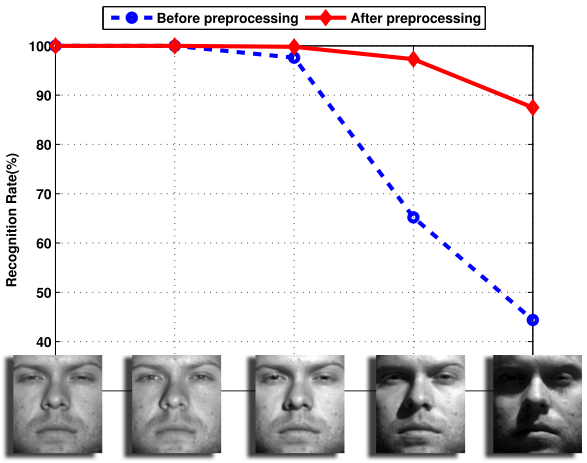

Fig. 1. (Lower curve) Degradation of the performance of LBP descriptors with nearest-neighbour classification under the increasingly extreme illumination conditions of subsets 1-5 of the Yale database [5]. Example images are shown on the horizontal axis. (Upper curve) Adding our preprocessing chain greatly improves the performance under difficult illumination.

Fig. 2. The stages of our full face recognition method.

feature sets range from geometrical features [8] to image derivative features such as edge maps [1], Local Binary Patterns (LBP) [2,3], Gabor wavelets [46,49,31], and local autocorrelation filters [14].

Although such features offer a great improvement on raw gray values, their resistance to the complex illumination variations that occur in real-world face images is still quite limited. For example, even though LBP features are completely invariant to monotonic global gray-level transformations, their performance degrades significantly under changes of lighting direction and shadowing – see Fig. 1. Similar results apply to the other features. Nevertheless, it is known that complete illumination invariants do not exist [9] so one must content oneself with finding representations that are more resistant to the most common classes of natural illumination variations.

In this paper, we propose an integrative framework that combines the strengths of all three of the above approaches. The overall process can be viewed as a pipeline consisting of image normalization, feature extraction and subspace representation, as shown in Fig. 2. Each stage increases resistance to illumination variations and makes the information needed for recognition more manifest. The method centres on a rich set of robust visual features that is selected to capture as much as possible of the available information. A well-designed image preprocessing pipeline is prepended to further enhance robustness. The features are used to construct illumination-insensitive subspaces, thus capturing the residual statistics of the data with relatively few training samples (many fewer than traditional raw-image-based appearance based methods such as [23]).

We will investigate several aspects of this framework:

1) **The relationship between image normalization and feature sets.** Normalization is known to improve the performance of simple subspace methods (*e.g.* PCA) or classifiers (*e.g.* nearest neighbors) based on image pixel representations [35,44,10], but its influence on more sophisticated feature sets has not received the attention that it deserves. A given preprocessing method may or may not improve the performance of a given feature set on a given data set. For example, for Histogram of Oriented Gradient features combining normalization and robust features is useful in [11], while histogram equalization has essentially no effect on LBP descriptors [3], and in some cases preprocessing actually hurts performance [12] – presumably because it removes too much useful information. Here we propose a simple image preprocessing chain that appears to work well for a wide range visual feature sets, eliminating many of the effects of changing illumination while still preserving most of the appearance details needed for recognition.
2) **Robust feature sets and feature comparison strategies.** Current feature sets offer quite good performance under illumination variations but there is still room for improvement. For example, LBP features are known to be sensitive to noise in near-uniform image regions such as cheeks and foreheads. We introduce a generalization of LBP called Local Ternary Patterns (LTP) that is more discriminant and less sensitive to noise in uniform regions. Moreover, in order to increase robustness to spatial deformations, LBP based representations typically subdivide the face into a regular grid and compare histograms of LBP codes within each region. This is somewhat arbitrary and it is likely to give rise to both aliasing and loss of spatial resolution. We show that replacing histogramming with a similarity metric based on local distance transforms further improves the performance of LBP/LTP based face recognition.
3) **Fusion of multiple feature sets.** Many current pattern recognition systems use only one type of feature. However in complex tasks such as face recognition, it is often the case that no single class of features is rich enough to capture all of the available information. Finding and combining complementary feature sets has thus become an active research topic, with successful applications in many challenging tasks including handwritten character recognition [16] and face recognition [27]. Here we show that combining two of the most successful local face representations, Gabor wavelets and Local Binary Patterns (LBP), gives considerably better performance than either alone. The two feature sets are complimentary in the sense that LBP captures small appearance details while Gabor wavelets encode facial shape over a broader range of scales.

To demonstrate the effectiveness of the proposed method we give results on the Face Recognition Grand Challenge version 2 experiment 4 dataset ("FRGC-204"), and on two other face datasets chosen to test recognition under difficult illumination

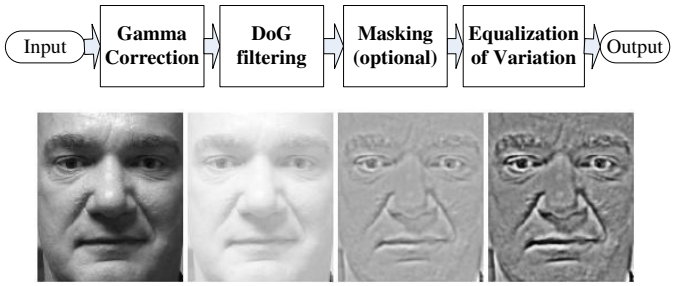

Fig. 3. (Top) the stages of our image preprocessing pipeline, and (bottom) an example of the effect of the three stages – from left to right: input image; image after Gamma correction; image after DoG filtering; image after robust contrast normalization.

conditions. FRGC-204 is a challenging large-scale dataset containing 12 776 training images, 16 028 controlled target images and 8 014 uncontrolled query images. To the best of knowledge this is the first time that a preprocessing method has been systematically evaluated on such a large-scale database, and our method achieves very significant improvements, achieving a Verification Rate of 88.1% at 0.1% False Acceptance Rate.

The rest of the paper is organized as follows: Section II presents our preprocessing chain, Section III introduces our LTP local texture feature sets, Section IV describes our multiple-feature fusion framework, Section V reports experimental results, and Section VI concludes. A preliminary description of the methods was presented in the conference papers [40,41].

II. Illumination Normalization

A. The Preprocessing Chain

This section describes our illumination normalization method. This is a preprocessing chain run before feature extraction that incorporates a series of stages designed to counter the effects of illumination variations, local shadowing and highlights while preserving the essential elements of visual appearance. Fig. 3 illustrates the three main stages and their effect on a typical face image. Although it was motivated by intuition and experimental studies rather than biology, the overall chain is reminiscent of the first few stages of visual processing in the mammalian retina and LGN. In detail, the stages are as follows.

Gamma Correction is a nonlinear gray-level transformation that replaces gray-level I with I^γ (for $\gamma > 0$) or $\log(I)$ (for $\gamma = 0$), where $\gamma \in [0, 1]$ is a user-defined parameter. This enhances the local dynamic range of the image in dark or shadowed regions while compressing it in bright regions and at highlights. The underlying principle is that the intensity of the light reflected from an object is the product of the incoming illumination L (which is piecewise smooth for the most part) and the local surface reflectance R (which carries detailed object-level appearance information). We want to recover object-level information independent of illumination, and taking logs makes the task easier by converting the product into a sum: for constant local illumination, a given reflectance step produces a given step in $\log(I)$ irrespective of the actual intensity of the illumination. In practice a full log transformation is often too strong, tending to over-amplify the noise in dark regions of the image, but a power law with exponent γ in the range $[0, 0.5]$ is a good compromise[1]. Here we use $\gamma = 0.2$ as the default setting.

Difference of Gaussian (DoG) Filtering. Gamma correction does not remove the influence of overall intensity gradients such as shading effects. Shading induced by surface structure is a potentially useful visual cue but it is predominantly low spatial frequency information that is hard to separate from effects caused by illumination gradients. High pass filtering removes both the useful and the incidental information, thus simplifying the recognition problem and in many cases increasing the overall system performance. Similarly, suppressing the highest spatial frequencies potentially reduces both aliasing and noise without destroying too much of the underlying recognition signal. DoG filtering is a convenient way to achieve the resulting bandpass behaviour. Fine details remain critically important for recognition so the inner (smaller) Gaussian is typically quite narrow ($\sigma_0 \leq 1$ pixel), while the outer one might have σ_1 of 2–4 pixels or more, depending on the spatial frequency at which low frequency information becomes misleading rather than informative. Given the strong lighting variations in our datasets we find that $\sigma_1 \approx 2$ typically gives the best results, but values up to about 4 are not too damaging and may be preferable for datasets with less extreme lighting variations. LBP and LTP features do seem to benefit from a little smoothing ($\sigma_0 \approx 1$), perhaps because pixel based voting is sensitive to aliasing artifacts. Below we use $\sigma_0 = 1.0$ and $\sigma_1 = 2.0$ by default[2].

We implement the filters using explicit convolution. To minimize boundary effects, if the face is part of a larger image the gamma correction and prefilter should be run on an appropriate region of this before cutting out the face image. Otherwise, extend-as-constant boundary conditions should be used: using extend-as-zero or wrap-around (FFT) boundary conditions significantly reduces the overall performance, in part because it introduces strong gradients at the image borders that disturb the subsequent contrast equalization stage. Prior gamma normalization is still required: if DoG is run without this, the resulting images suffer from reduced local contrast (and hence loss of visual detail) in shadowed regions.

Masking. If facial regions (hair style, beard, ...) that are felt to be irrelevant or too variable need to be masked out, the mask should be applied at this point. Otherwise, either strong artificial gray-level edges are introduced into the DoG convolution, or invisible regions are taken into account during contrast equalization.

Contrast Equalization. The final stage of our preprocessing

[1] Shot noise – the dominant noise source in modern CCD sensors – is proportional to the square root of illuminance so $\gamma = 0.5$ makes it approximately uniform: $\sqrt{I + \Delta I} \approx \sqrt{I} + \frac{\Delta I}{2\sqrt{I}} = \sqrt{I} + \text{const.}$ for $\Delta I \propto \sqrt{I}$.

[2] Curiously, for some datasets it also helps to offset the center of the larger filter by 1–2 pixels relative to the center of the smaller one, so that the final prefilter is effectively the sum of a centered DoG and a low pass spatial derivative. The best direction for the displacement is somewhat variable but typically diagonal. The effect is not consistent enough to be recommended practice, but it might repay further investigation.

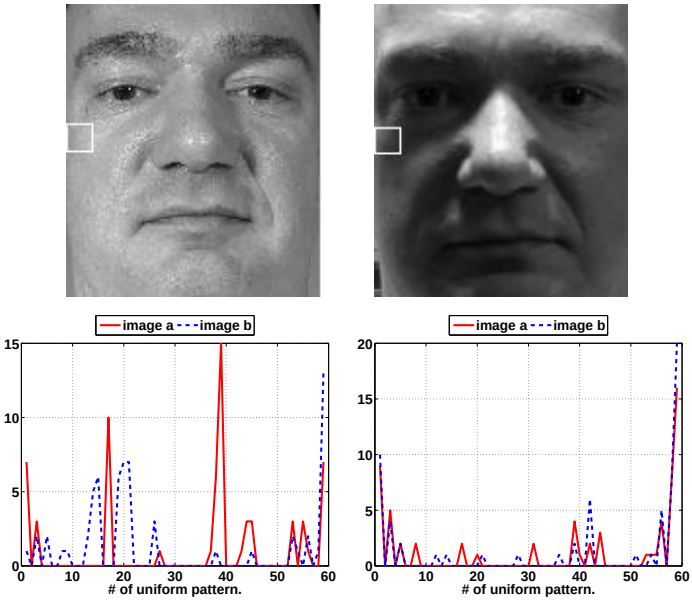

Fig. 4. (Top) Two images of the same subject from the FRGC-204 dataset. (Bottom) The LBP histograms of the marked image regions, (left) without preprocessing, (right) after preprocessing. Note the degree to which preprocessing reduces the variability of the histograms of these relatively featureless but differently illuminated facial regions.

chain rescales the image intensities to standardize a robust measure of overall contrast or intensity variation. It is important to use a robust estimator because the signal typically contains extreme values produced by highlights, small dark regions such as nostrils, garbage at the image borders, *etc*. One could use (for example) the median of the absolute value of the signal for this, but here we have preferred a simple and rapid approximation based on a two stage process:

$$I(x,y) \leftarrow \frac{I(x,y)}{(\text{mean}(|I(x',y')|^a))^{1/a}} \quad (1)$$

$$I(x,y) \leftarrow \frac{I(x,y)}{(\text{mean}(\min(\tau, |I(x',y')|)^a))^{1/a}} \quad (2)$$

Here, a is a strongly compressive exponent that reduces the influence of large values, τ is a threshold used to truncate large values after the first phase of normalization, and the mean is over the whole (unmasked part of the) image. By default we use $\alpha = 0.1$ and $\tau = 10$.

The resulting image is well scaled but it can still contain extreme values. To reduce their influence on subsequent stages of processing, we apply a final nonlinear mapping to compress over-large values. The exact functional form is not critical. Here we use the hyperbolic tangent $I(x,y) \leftarrow \tau \tanh(I(x,y)/\tau)$, thus limiting I to the range $(-\tau, \tau)$.

B. Robustness and Computation Time

To illustrate the need for preprocessing we demonstrate its effect on part of an LBP histogram feature set. Fig. 4 (top) shows a matching target-query pair chosen randomly from the FRGC-204 dataset. We chose a relatively featureless – and hence not particularly informative – face region (white squares) and extracted its LBP histograms, both without (bottom left) and with (bottom right) image preprocessing. Without

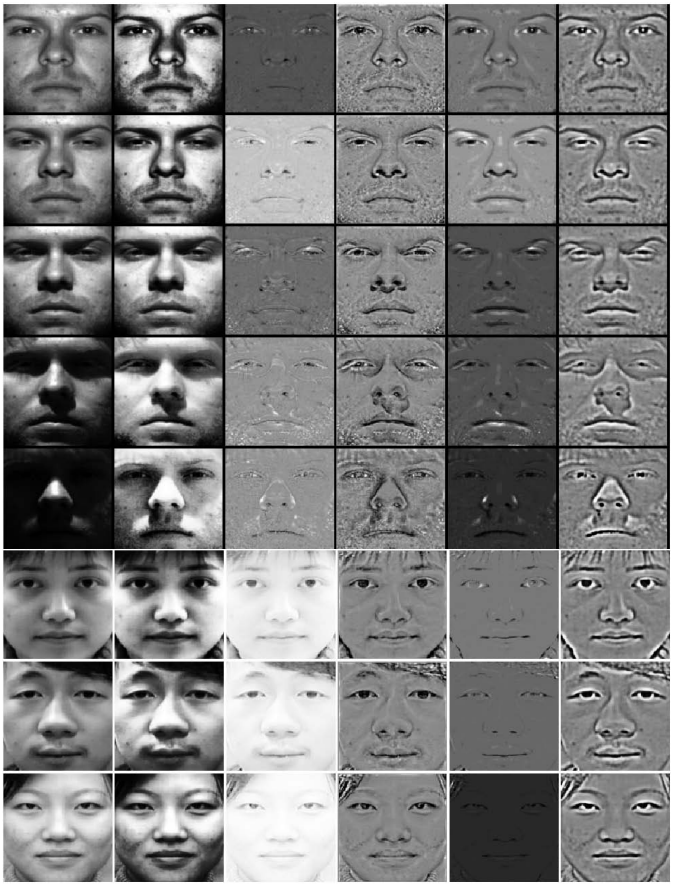

Fig. 5. Examples of the effects of the different preprocessing methods. Rows 1-5 respectively show images of one subject from subsets 1-5 of the Yale-B data set, and rows 6-8 show images of different subjects from the CAS-PEAL data set, with from left to right: (None) no preprocessing; (HE) Histogram Equalization; (MSR) Multiscale Retinex; (GB) Gross & Brajovic method; (LTV) Logarithmic Total Variation; (TT) Our preprocessing method.

preprocessing the two histograms are both highly variable and very different, but preprocessing significantly reduces these differences, quantitatively decreasing the χ^2 inter-histogram distance from 93.4 to 25.0.

Run time is also a critical factor in many applications. Our method uses only simple closed-form image operations so it is much more efficient than ones that require expensive iterative optimizations such as Logarithmic Total Variation (LTV, [10]) and anisotropic diffusion (GB, [13]). Our (unoptimized Matlab) implementation[3] takes only about 50 ms to process a 128×128 pixel face image on a 2.8 GHz P4, allowing face preprocessing to be performed in real time and thus providing the ability to handle large face databases. In comparison, the current implementation of GB is about 5 times slower and LTV is about 300 times slower.

C. Competing Methods

Below we will use recognition rates under several feature sets and data sets to compare the performance of our

[3]A Matlab implementation is publicly available on the author' homepage *http://parnec.nuaa.edu.cn/xtan*.

preprocessing chain with that of several competing methods. We will not describe the methods tested in detail owing to lack of space, but briefly they are: Histogram Equalization (HE); Multiscale Retinex (MSR) [19]; Gross & Brajovic's anisotropic smoothing (GB) [13]; and Logarithmic Total Variation (LTV) [10]. The implementations of these algorithms were based in part on the publicly available *Torch3Vision* toolbox (http://torch3vision.idiap.ch). We would also like to thank Terrence Chen for making his implementation of LTV [10] available to us.

To illustrate the effects of the different preprocessors, Fig. 5 shows some example images from the Yale-B and CAS-PEAL data sets, with the corresponding preprocessor outputs. As the images suggest, and the experiments below confirm, point transformations such as Histogram Equalization are not very effective at removing spatial effects such as shadowing. In contrast, GB and our method TT (which are the best performers below) remove much of the smooth shading information and hence emphasize local appearance. The LTV images appear washed out owing to the presence of small but intense specularities and dark peaks in the output. This is partly a display issue – MATLAB normalizes images based on their extreme values – but we have not corrected it to emphasize that many feature sets and image comparison metrics are also sensitive to such peaks. In contrast, in GB the peaks tend to diffuse away, while in our method they undergo strong nonlinear compression.

III. Local Ternary Patterns

A. Local Binary Patterns (LBP)

Ojala *et al*. [29] introduced Local Binary Patterns (LBP) as a means of summarizing local gray-level structure. The LBP operator takes a local neighborhood around each pixel, thresholds the pixels of the neighborhood at the value of the central pixel and uses the resulting binary-valued image patch as a local image descriptor. It was originally defined for 3×3 neighborhoods, giving 8 bit integer LBP codes based on the 8 pixels around the central one. Formally, the LBP operator takes the form

$$LBP(x_c, y_c) = \sum_{n=0}^{7} 2^n \, s(i_n - i_c) \tag{3}$$

where in this case n runs over the 8 neighbors of the central pixel c, i_c and i_n are the gray-level values at c and n, and $s(u)$ is 1 if $u \geq 0$ and 0 otherwise. The LBP encoding process is illustrated in Fig. 6.

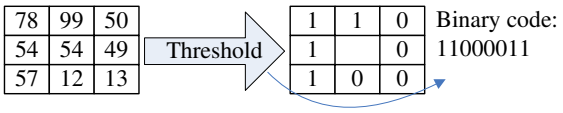

Fig. 6. Illustration of the basic LBP operator.

Two extensions of the original operator were made in [30]. The first defined LBP's for neighborhoods of different sizes, thus making it feasible to deal with textures at different scales. The second defined the so-called *uniform patterns*: an LBP is 'uniform' if it contains at most one 0-1 and one 1-0 transition when viewed as a circular bit string. For example, the LBP code in Fig. 6 is uniform. Uniformity is important because it characterizes the patches that contain primitive structural information such as edges and corners. Ojala *et al*. observed that although only 58 of the 256 8-bit patterns are uniform, nearly 90 percent of all observed image neighbourhoods are uniform and many of the remaining ones contain essentially noise. Thus, when histogramming LBP's the number of bins can be reduced significantly by assigning all non-uniform patterns to a single bin, typically without losing too much information.

B. Local Ternary Patterns (LTP)

LBP's have proven to be highly discriminative features for texture classification [29] and they are resistant to lighting effects in the sense that they are invariant to monotonic gray-level transformations. However because they threshold at exactly the value of the central pixel i_c they tend to be sensitive to noise, particularly in near-uniform image regions, and to smooth weak illumination gradients. Many facial regions are relatively uniform and it is legitimate to investigate whether the robustness of the features can be improved in these regions.

This section extends LBP to 3-valued codes, *Local Ternary Patterns* (LTP), in which gray-levels in a zone of width $\pm t$ around i_c are quantized to zero, ones above this are quantized to $+1$ and ones below it to -1, *i.e*. the indicator $s(u)$ is replaced with a 3-valued function:

$$s'(u, i_c, t) = \begin{cases} 1, & u \geq i_c + t \\ 0, & |u - i_c| < t \\ -1, & u \leq i_c - t \end{cases} \tag{4}$$

and the binary LBP code is replaced by a ternary LTP code. Here t is a user-specified threshold – so LTP codes are more resistant to noise, but no longer strictly invariant to gray-level transformations. The LTP encoding procedure is illustrated in Fig. 7. Here the threshold t was set to 5, so the tolerance interval is $[49, 59]$.

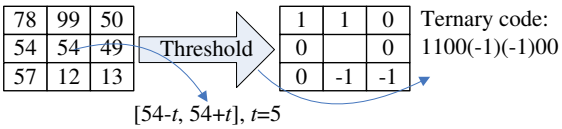

Fig. 7. Illustration of the basic LTP operator.

When using LTP for visual matching we could use 3^n valued codes, but the uniform pattern argument also applies in the ternary case. For simplicity, the experiments below use a coding scheme that splits each ternary pattern into its positive and negative halves as illustrated in Fig. 8, subsequently treating these as two separate channels of LBP descriptors for which separate histograms and similarity metrics are computed, combining the results only at the end of the computation.

LTP's bear some similarity to the texture spectrum (TS) technique from the early 1990's [45]. However TS did not include preprocessing, thresholding, local histograms or uniform pattern based dimensionality reduction and it was not tested on faces.

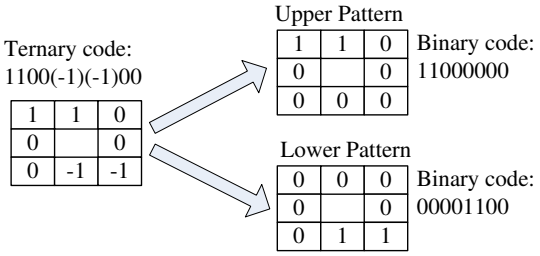

Fig. 8. Splitting an LTP code into positive and negative LBP codes.

C. Distance Transform based Similarity Metric

Ahonen *et al*. [2] introduced an LBP based method for face recognition that divides the face into a regular grid of cells and histograms the uniform LBP's within each cell, finally using nearest neighbor classification in the χ^2 histogram distance for recognition:

$$\chi^2(p,q) = \sum_i \frac{(p_i - q_i)^2}{p_i + q_i} \tag{5}$$

Here p, q are image region descriptors (histogram vectors), respectively.

This method gave excellent results on the FERET dataset. However subdividing the face into a regular grid seems somewhat arbitrary: the cells are not necessarily well aligned with facial features, and the partitioning is likely to cause both aliasing (due to abrupt spatial quantization of descriptor contributions) and loss of spatial resolution (as position within each grid cell is not coded). Given that the overall goal of coding is to provide illumination- and outlier-robust visual correspondence with some leeway for small spatial deviations due to misalignment, it seems more appropriate to use a Hausdorff-distance-like similarity metric that takes each LBP or LTP pixel code in image X and tests whether a similar code appears at a nearby position in image Y, with a weighting that decreases smoothly with image distance. Such a scheme should be able to achieve discriminant appearance-based image matching with a well-controllable degree of spatial looseness.

We can achieve this using Distance Transforms [7]. Given a 2-D reference image X, we find its image of LBP or LTP codes and transform this into a set of sparse binary images b^k, one for each possible LBP or LTP code value k (*i.e*. 59 images for uniform codes). Each b^k specifies the pixel positions at which its particular LBP or LTP code value appears. We then calculate the distance transform image d^k of each b^k. Each pixel of d^k gives the distance to the nearest image X pixel with code k (2D Euclidean distance is used in the experiments below). The distance or similarity metric from image X to image Y is then:

$$D(X,Y) = \sum_{\text{pixels } (i,j) \text{ of } Y} w(d_X^{k_Y(i,j)}(i,j)) \tag{6}$$

Here, $k_Y(i,j)$ is the code value of pixel (i,j) of image Y and

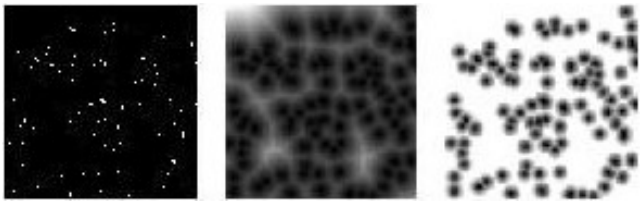

Fig. 9. From left to right: a binary layer, its distance transform, and the truncated linear version of this.

$w()$ is a user-defined function[4] giving the penalty to include for a pixel at the given spatial distance from the nearest matching code in X. In our experiments we tested both Gaussian similarity metrics $w(d) = \exp\{-(d/\sigma)^2/2\}$ and truncated linear distances $w(d) = \min(d, \tau)$. Their performance is similar, with truncated distances giving slightly better results overall. For 120×120 face images in which an iris or nostril has a radius of about 6 pixels and overall global face alignment is within a few pixels, our default parameter values were $\sigma = 3$ pixels and $\tau = 6$ pixels.

Fig. 9 shows an example of a binary layer and its distance transforms. For a given target the transform can be computed and mapped through $w()$ in a preprocessing step, after which matching to any subsequent image takes $\mathcal{O}$(number of pixels) irrespective of the number of code values.

IV. A Framework for Illumination-Insensitive Face Recognition

A. The method

This section details our robust face recognition framework introduced in Section I (*c.f*. Fig. 2). The full method incorporates the aforementioned preprocessing chain and LBP or LTP features with distance transform based comparison. However, as mentioned above, face recognition is a complex task for which it is useful to include multiple types of features, and we also need to build a final classification stage that can handle residual variability and learn effective models from relatively few training samples.

The selection of an expressive and complementary set of features is crucial for good performance. Our initial experiments suggested that two of the most successful local appearance descriptors, Gabor wavelets [21,46,26] and LBP (or its extension LTP), were promising candidates for fusion. LBP is good at coding fine details of facial appearance and texture, whereas Gabor features encode facial shape and appearance over a range of coarser scales[5]. Both representations are rich in information and computationally efficient, and their complementary nature makes them good candidates for fusion.

In face recognition, it is widely accepted that discriminant based approaches offer high potential performance and improved robustness to perturbations such as lighting variations

[4] w is monotonically increasing for a distance metric and monotonically decreasing for a similarity one. In D, note that each pixel in Y is matched to the nearest pixel with the same code in X. This is not symmetric between X and Y even if the underlying distance d is, but it can be symmetrized if desired.

[5] Gabor features have also been used as a preprocessing stage for LBP feature extraction [52].

(*e.g.* [5]) and that kernel methods provide a well-founded means of incorporating domain knowledge in the discriminant. In particular, Kernel Linear Discriminant Analysis (KLDA [28]) has proven to be an effective method of extracting discriminant information from a high dimensional kernel feature space under subspace constraints such as those engendered by lighting variations [26]. We use Gaussian kernels $\mathbf{k}(p,q) = e^{-\mathrm{dist}(p,q)/(2\sigma^2)}$, where $\mathrm{dist}(p,q)$ is $\|p-q\|^2$ for Gabor wavelets and χ^2 histogram distance (5) for LBP feature sets.

We now summarize our modified KLDA method. Let $\phi()$ be the mapping to the implicit feature space, $\mathbf{\Phi} = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_m)]$ be the operator mapping the m training examples to the feature space, and $\overline{\mathbf{\Phi}} = \mathbf{\Phi}\,\mathbf{\Pi} = [\ldots, \phi(\mathbf{x}_i) - \boldsymbol{\mu}, \ldots]$ be the operator of centred training examples, where $\mathbf{\Pi} = \mathbf{I} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top$ and $\boldsymbol{\mu} = \frac{1}{m}\mathbf{\Phi}\mathbf{1}_m$. To perform LDA, we need explicit orthogonal coordinates for this implicit feature space. If we had $\overline{\mathbf{\Phi}}$ in explicit form we could find its thin SVD $\mathbf{W}\,\mathbf{D}\,\mathbf{U}^\top$ and project to coordinates using $\mathbf{W}^\top = (\overline{\mathbf{\Phi}}\,\mathbf{U}\,\mathbf{D}^{-1})^\top$. We cannot do this directly, but we can find $\mathbf{U}$ and $\mathbf{D} = \mathbf{\Lambda}^{1/2}$ from the thin eigendecomposition of the centred kernel matrix of the training examples $\overline{\mathbf{K}} = \mathbf{\Pi}\,\mathbf{K}\,\mathbf{\Pi} = \mathbf{\Pi}\,\mathbf{\Phi}^\top\mathbf{\Phi}\,\mathbf{\Pi} = \mathbf{U}\,\mathbf{\Lambda}\,\mathbf{U}^\top$. This allows the projection of any example $\mathbf{x}$ to be calculated using $\mathbf{\Lambda}^{-1/2}\,\mathbf{U}^\top\mathbf{\Pi}\,\mathbf{k}_\mathbf{x}$ where $\mathbf{k}_\mathbf{x} = \mathbf{\Phi}^\top\phi(\mathbf{x})$ is the kernel vector of $\mathbf{x}$ against the training examples. Using these coordinates, we find the projected within-class and between-class scatter matrices $\mathbf{S}_W, \mathbf{S}_B$, from which a basis $\mathbf{V}$ for the kernel discriminative subspace is obtained by solving the thin LDA eigendecomposition $(\mathbf{S}_W + \varepsilon\, I)^{-1}\,\mathbf{S}_B\,\mathbf{V} = \mathbf{V}\,\mathbf{\Xi}$ for eigenvectors $\mathbf{V}$ and eigenvalues $\mathbf{\Xi}$. Here, ε is a small regularization constant (10^{-3} below) and $\mathbf{I}$ is the identity matrix. The optimal projection operator is then $\mathbf{P} = \overline{\mathbf{\Phi}}\,\mathbf{U}\,\mathbf{\Lambda}^{-1/2}\,\mathbf{V}$ and test examples $\mathbf{x}$ can be projected into the optimal discriminant space by

$$\Omega_\mathbf{x} \;=\; \mathbf{P}^\top\phi(\mathbf{x}) \;=\; \mathbf{V}^\top\mathbf{\Lambda}^{-1/2}\,\mathbf{U}^\top\mathbf{k}_\mathbf{x}. \tag{7}$$

The projected feature vectors $\mathbf{\Omega}_{\mathrm{test}}$ are classified using the nearest neighbour rule and the cosine 'distance'

$$\mathbf{d}_{\cos}(\mathbf{\Omega}_{\mathrm{test}}, \mathbf{\Omega}_{\mathrm{template}}) \;=\; -\frac{\mathbf{\Omega}_{\mathrm{test}}^\mathbf{T}\,\mathbf{\Omega}_{\mathrm{template}}}{\|\mathbf{\Omega}_{\mathrm{test}}\|\;\|\mathbf{\Omega}_{\mathrm{template}}\|} \tag{8}$$

where $\mathbf{\Omega}_{\mathrm{template}}$ is a face template in the gallery set. Other similarity metrics such as L_1, L_2 or Mahalanobis distances could be used, but [26] found that the cosine distance performed best among the metrics it tested on this database, and our initial experiments confirmed this.

When a face image is presented to the system, its Gabor wavelet and LBP features are extracted, separately projected into their optimal discriminant spaces (7) and used to compute the corresponding distance scores (8). Each score s is normalized using the '*z-score*' method [17]

$$z = \frac{s-\mu}{\sigma} \tag{9}$$

where μ, σ are respectively the mean and standard deviation of s over the training set.

Finally the two scores z_{Gabor} and z_{LBP} are fused at the decision level. Notwithstanding suggestions that it is more effective to fuse modalities at an earlier stage of processing [17], our earlier work found that although feature-level and decision-level fusion both work well, decision-level fusion is better in this application [41]. Kittler *et al.* [20] investigated a number of different fusion schemes including product, sum, min and max rules, finding that the sum rule was the most resilient to estimation errors and gave the best performance overall. Thus we fuse the Gabor and LBP similarity scores using the simple sum rule: $z_{\mathrm{Gabor}} + z_{\mathrm{LBP}}$. The resulting similarity score is input to a simple Nearest Neighbor (NN) classifier to make the final decision. A similar strategy was independently proposed by [47].

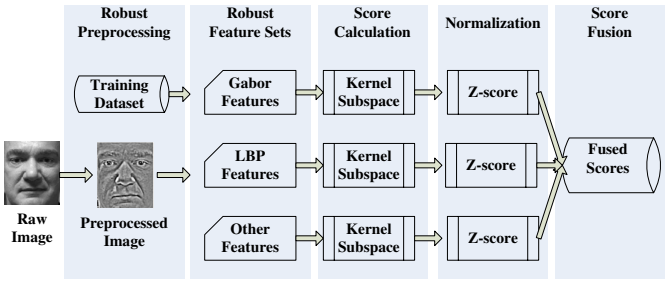

Fig. 10. The overall architecture of our multi-feature subspace based face recognition method.

Fig. 10 gives the overall flowchart of the proposed method. We emphasize that it includes a number of elements that improve recognition in the face of complex lighting variations: (*i*) we use a combination of complementary visual features – LBP and Gabor wavelets; (*ii*) the features are individually both robust and information-rich; (*iii*) preprocessing – which is usually ignored in previous work on these feature sets [2,3,46,25] – greatly improves robustness; (*iv*) the inclusion of kernel subspace discriminants increases discriminativity while compensating for any residual variations. As we will show below, each of these factors contributes to the overall system performance and robustness.

B. Tensor-based Feature Representation

The LBP and Gabor feature sets described here are both very high dimensional (usually over 10 000), and it would be useful to be able to represent them more compactly without sacrificing too much performance. Inspired by recent work on tensor-based decompositions, we tested tensor-based representations for LBP and Gabor features using General Tensor Discriminant Analysis (GTDA) [42] as a dimensionality reduction method. The resulting reduced tensors are written as vectors, optionally subjected to additional stages of feature extraction, then fed to the classifier. Note that there are many other dimensionality reduction methods that could be applied – notably local or manifold-based representations such as [51] – but in this paper we will focus on global linear vector and tensor reductions.

V. Experiments

We illustrate the effectiveness of our methods by presenting experiments on three large-scale face data sets with difficult lighting conditions: Extended Yale B, CAS-PEAL-R1, and Face Recognition Grand Challenge version 2 Experiment 4.

For each data set we use its standard evaluation protocol in order to facilitate comparison with previous work.

We divide the results into two sections, the first focusing on nearest neighbour classification with various LBP/LTP based feature sets and distance metrics, and the second on KLDA based classifiers with combinations of LBP and Gabor features. Note that unlike subspace based classifiers such as KLDA, the Nearest Neighbour methods do not use a separate training set – they simply compare probe images directly to gallery ones using a given (not learned) feature set and distance metric. They are thus simpler, but in general less discriminant than methods that learn subspaces, feature sets or distance metrics.

In both cases we compare several different preprocessing methods. The benefits of preprocessing are particulary marked for Nearest Neighbour classifiers. We only show results for LBP/LTP here, but additional experiments showed that our preprocessing method substantially increases the performance of Nearest Neighbour classifiers for a wide variety of other image descriptors including pixel or Gabor based linear or kernelized eigen- or Fisher-faces under a range of descriptor normalizations and distance metrics.

A. Data Sets

Fig. 11 shows some example images from our three datasets, with the corresponding output of our standard preprocessing chain.

Extended Yale-B. The Yale Face Dataset B [5] containing 10 people under 64 different illumination conditions has been a de facto standard for studies of recognition under variable lighting over the past decade. It was recently updated to the Extended Yale Face Database B [23], containing 38 subjects under 9 poses and 64 illumination conditions. In both cases the images are divided into five subsets according to the angle between the light source direction and the central camera axis $(12°, 25°, 50°, 77°, 90°)$. For our experiments, the images with the most neutral light sources ('A+000E+00') were used as the gallery, and all frontal images of each of the standard subsets 1–5 were used as probes (in all, 2 414 images of 38 subjects). The Extended Yale-B set only contains 38 subjects and it has little variability of expression, ageing, *etc*. However its extreme lighting conditions still make it a challenging task for most face recognition methods.

CAS-PEAL-R1. The CAS-PEAL-R1 face database contains 30 863 images of 1 040 individuals (595 males and 445 females, predominantly Chinese). The standard experimental protocol [12] divides the data into a training set, a gallery set and six frontal probe sets. There is no overlap between the gallery and any of the probes. The gallery contains one image taken under standard conditions for each of the 1 040 subjects, while the six probes respectively contain images with the following basic classes of variations: expression, lighting, accessories, background, distance and ageing. Here we use the lighting probe, which contains 2 243 images. The illumination conditions are somewhat less extreme than those of Yale-B, but the induced shadows are substantially sharper, presumably because the angular light sources were less diffuse. This makes it harder for all high-pass based preprocessors to separate shadows from facial details.

When training KLDA on CAS-PEAL-R1, we use the standard CAS-PEAL-R1 protocol and training set, which contains 4 frontal images each of 300 subjects who were randomly selected from the full 1 040-subject data set.

FRGC-204. The Face Recognition Grand Challenge version 2 Experiment 4 data set [33] is the largest data set studied here. It contains 12 776 training images, 16 028 target images and 8 014 query images. The targets were obtained under controlled conditions but the probes were captured in uncontrolled indoor and outdoor settings, including many images of poor quality that pose a real challenge to any recognition method. FRGC-204 is the most challenging data set studied here, owing to its large size and to the wide range of natural variations that it contains including large lighting variations, ageing and image blur.

We use the standard FRGC experimental protocol based on the Biometric Experimentation Environment (BEE) evaluation tool[6], reporting performance in terms of Receiver Operating Characteristic (ROC) curves of Face Verification Rate (FVR) versus False Accept Rate (FAR). BEE allows three types of curves to be generated – ROC-I, ROC-II and ROC-III – corresponding respectively to images collected within a semester, within a year, and across years. Below we report only ROC-III, the most challenging and most commonly reported results. To facilitate comparison with previous publications on FRGC-204, we used only Liu's 6 388-image subset [26] of the full FRGC-204 training set for training.

TABLE I
DEFAULT PARAMETER SETTINGS FOR OUR METHODS.

Procedure	Parameter	Value
Gamma Correction	γ	0.2
DoG Filtering	σ_0	1
	σ_1	2
Contrast Equalization	α	0.1
	τ	10
LTP	t	0-0.2
LBP/LTP χ^2 cell size		8×8
σ for KLDA kernels with LBP	σ	10^5

B. Experimental Settings

We restrict attention to geometrically aligned frontal face views but allow lighting, expression and identity to vary. Geometric alignment includes conversion to 8 bit gray-scale, rigid image scaling and rotation to place the centers of the two eyes at fixed positions, and image cropping to 128×128 pixels. The eye coordinates are those supplied with the original data sets.

Unless otherwise noted, the parameter settings listed in table I apply to all experiments. The exact setting of the preprocessor parameters is not critical: the method gives similar results over a broad range of settings.

[6]This evaluates the entire (16028×8014) all-pairs similarity matrix between the query images and the targets – a very expensive calculation that requires more than 128 million face comparisons.

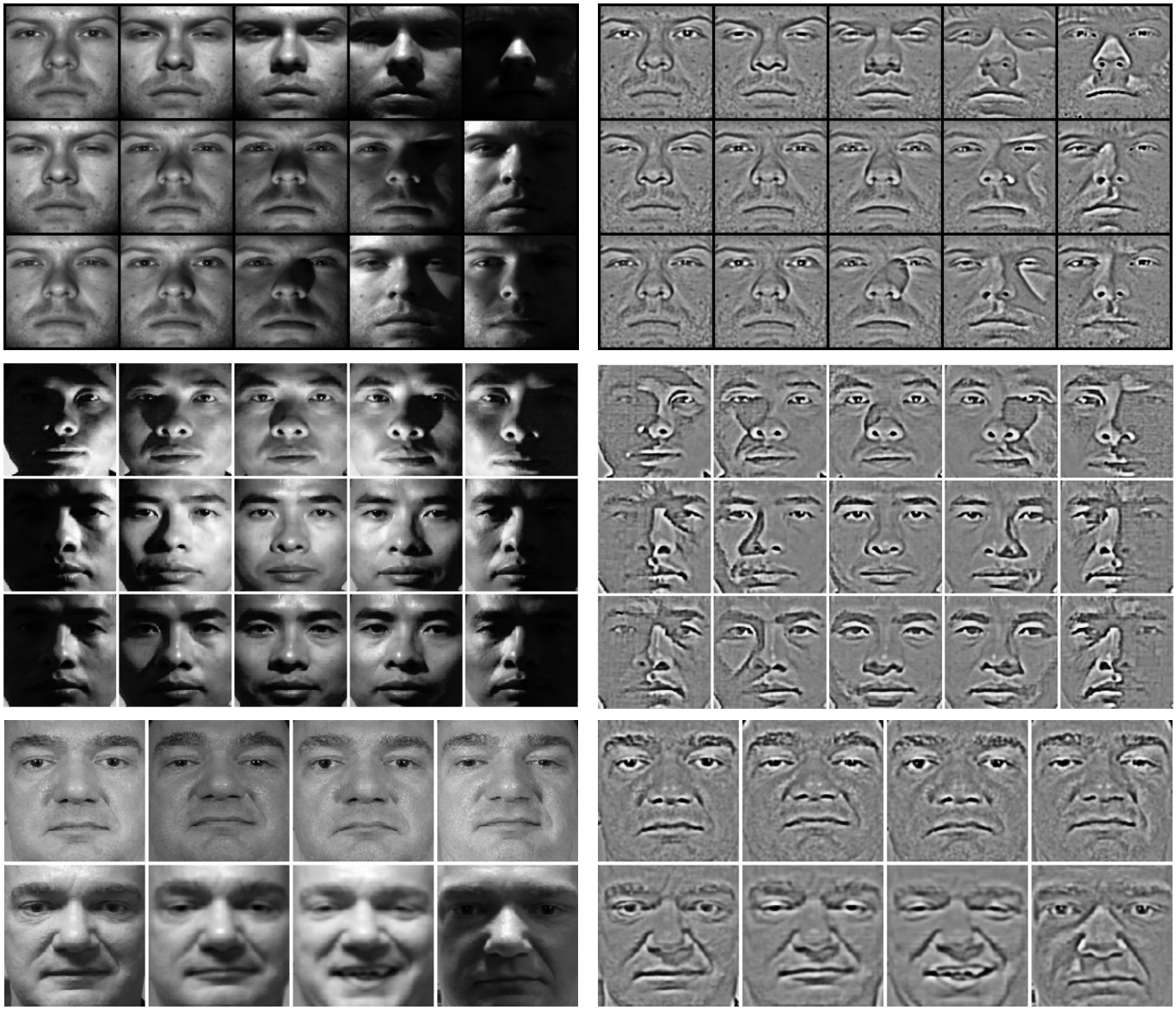

Fig. 11. Example images from the three data sets used for testing: (top) frontal images of a subject from Extended Yale-B – columns 1–5 respectively contain samples from illumination subsets 1–5; (middle) a subject from the CAS-PEAL probes, with illumination ranging from left to right and from below to above; (bottom) a subject from FRGC-204 – the first row shows controlled gallery images and the second one uncontrolled query images. In each case we show raw (geometrically normalized) images on the left, and the corresponding output of our standard preprocessing method on the right. As the experiments below confirm, preprocessing greatly reduces the influence of lighting variations, although it can not completely remove the effects of hard shadowing.

The GB and LTV preprocessors have a data fidelity parameter λ to set (*c.f.*, Eq. 10 below). For GB we set $\lambda = 1$ for all experiments. For LTV we set $\lambda = 0.75$ (as recommended in [10]) for Yale-B, but found that $\lambda = 0.5$ worked better for CAS-PEAL and FRGC2.

For General Tensor Discriminant Analysis (GTDA) [42] based dimensionality reduction we used the following settings. For the Gabor features we applied 40 Gabor filters (5 scales and 8 directions) to each 128×128 face image, taking the modulus of the output and down-sampling it to 16×16 to provide a $16 \times 16 \times 8 \times 5$ tensor. This is input to GTDA over the training set. The best results were obtained by retaining 99.9% of the overall energy, giving an output tensor of size $14 \times 15 \times 7 \times 4$ (a 43% reduction in overall feature dimension). We also tested the decompositions discussed in [42]: resizing the image to 64×64 and decomposing the tensor as $64 \times 64 \times 8$ ('GaborS', with sum of coefficients over all orientations), $64 \times 64 \times 5$ ('GaborD', with sum over all scales) and 64×64 ('GaborSD', with sum over all scales and orientations), in each case setting the output dimension in each mode to preserve a fixed proportion of the overall energy. However these latter settings gave poor recognition rates. For example, TT/GaborD-GTDA/NN gave 22.0% and TT/GaborD-GTDA/KLDA/NN gave 39.6% on CAS-PEAL.

For the LBP features, the image contains 16×16 LBP blocks, each with a 59-D histogram, so we have a $16 \times 16 \times 59$ tensor. We found that it was best to express this as a 256×59 2-mode tensor. After reduction with GTDA, the output tensor had a size of 248×47.

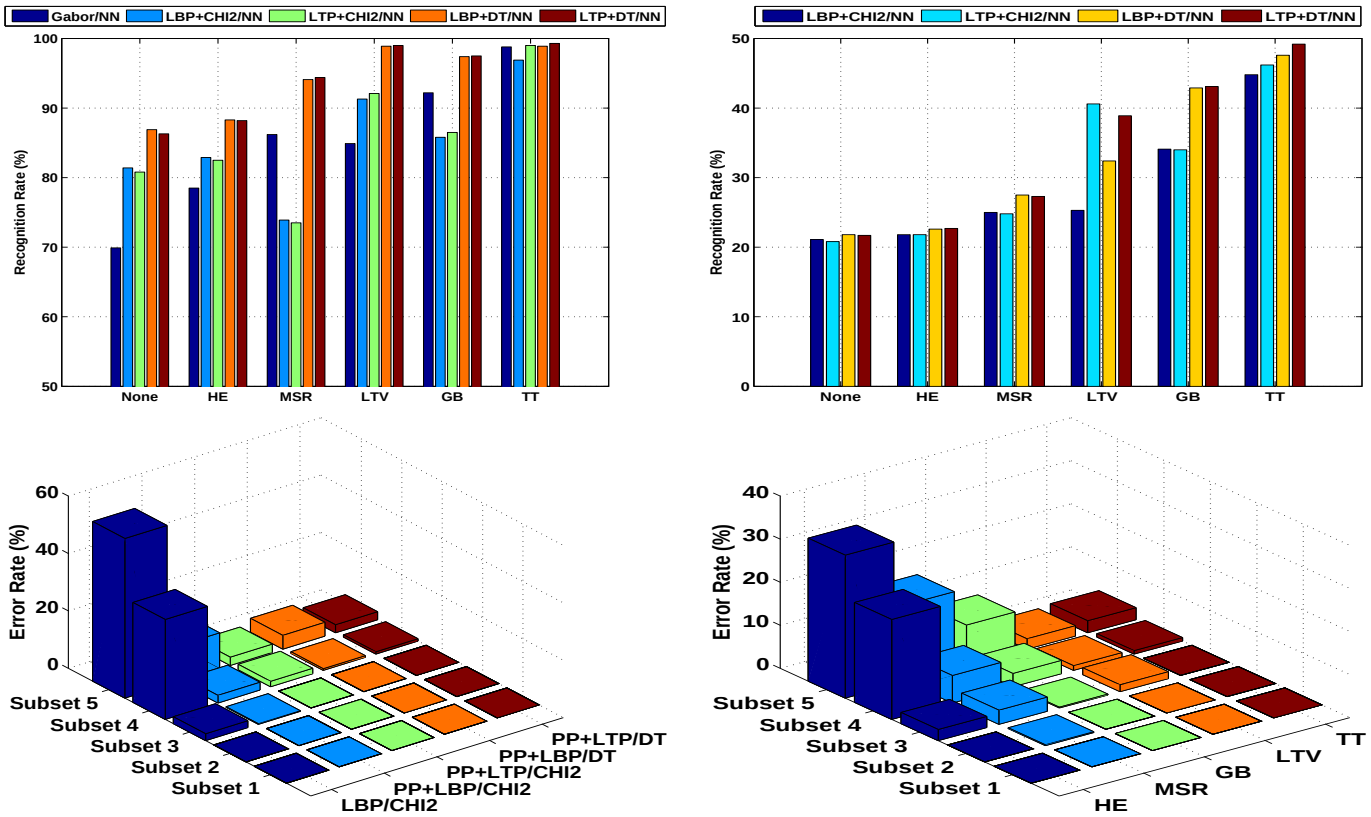

Fig. 12. (Top) Overall nearest-neighbor recognition rates (%) on (left) Extended Yale-B and (right) CAS-PEAL-R1, using the proposed LBP based and Gabor features and various preprocessing methods. (Bottom) Breakdown of error rates on the five Extended Yale-B subsets for (left) the various feature sets with our standard preprocessing, and (right) the various preprocessing methods with LTP/DT features.

C. Results for Nearest Neighbour Classification

Fig. 12 (top) shows the extent to which nearest neighbour based LBP face recognition can be improved by combining three of the enhancements proposed here: using preprocessing (PP); replacing LBP with LTP; and replacing local histogramming and the χ^2 histogram distance with the Distance Transform based similarity metric (DT). On Extended Yale-B (top left), the absolute recognition rate is increased by about 23.5% relative to standard unpreprocessed LBP/χ^2. Preprocessing alone boosts the performance by 20.2% (from 75.5% to 95.7%). Replacing LBP with LTP improves the recognition rate to 98.7% and adding DT further improves it to 99.0%. Similarly, on CAS-PEAL-R1 (top right), our preprocessor improves the performance by over 20.0% for LBP, and replacing LBP/χ^2 with LTP/DT improves it by another 5.0%. In each case our preprocessing method is the most effective tested, followed by GB.

Fig. 12 (bottom) illustrates how the various feature sets and preprocessing methods degrade with the increasingly extreme illumination of Extended Yale-B sets 1–5. Even without image preprocessing, our system performs quite well under the mild lighting changes of subsets 1–3. However preprocessing is required for good performance under the more extreme conditions of subsets 4–5. For the most difficult subset 5, preprocessing improves the performance by 43.1%, while including either LTP or the distance transform respectively increases performance over PP+LBP/χ^2 by about 10.0% and 8.0%. Again our preprocessing method predominates, although LTV catches up as the lighting becomes more difficult and equals our method on subset 5. In contrast, GB does well on the easier subsets but has trouble with subsets 4 and 5.

To aid comparison with previous work, note that on the (older and smaller) 10 subject Standard Yale-B set our PP+LTP/DT method gives perfect results for all 5 illumination subsets. In contrast, on subsets 2–4: Harmonic Image Exemplars gives 100, 99.7, 96.9% [50]; nine points of light gives 100, 100, 97.2% [24]; and Gradient Angle gives 100, 100, 98.6% [9]. None of these authors test on the most difficult set, 5.

Fig. 13 (left) shows how the size of the histogram pooling neighborhood influences the performance of the LBP and LTP feature sets, here with TT preprocessing on Extended YaleB subset 5. The performance is optimal for sizes of around 6×6, however such small neighborhoods result in high dimensional face descriptors (and hence high computation cost) and potentially increased sensitivity to spatial misalignment. In practice, we preferred to use 8×8 as the default size below. Moreover, for LBP/LTP learning methods based on inter-example distances, the DT similarity measure provides slightly better performance than binning without the need to choose the neighborhood size.

The quantization threshold t is an important parameter for LTP. It depends on the noise or the range of pixel values in the preprocessed images, so we set it using a heuristic $t = \rho\,\mathrm{Median}\{\sigma(x_i)|i = 1, \ldots, N\}$ where x_i are the preprocessed

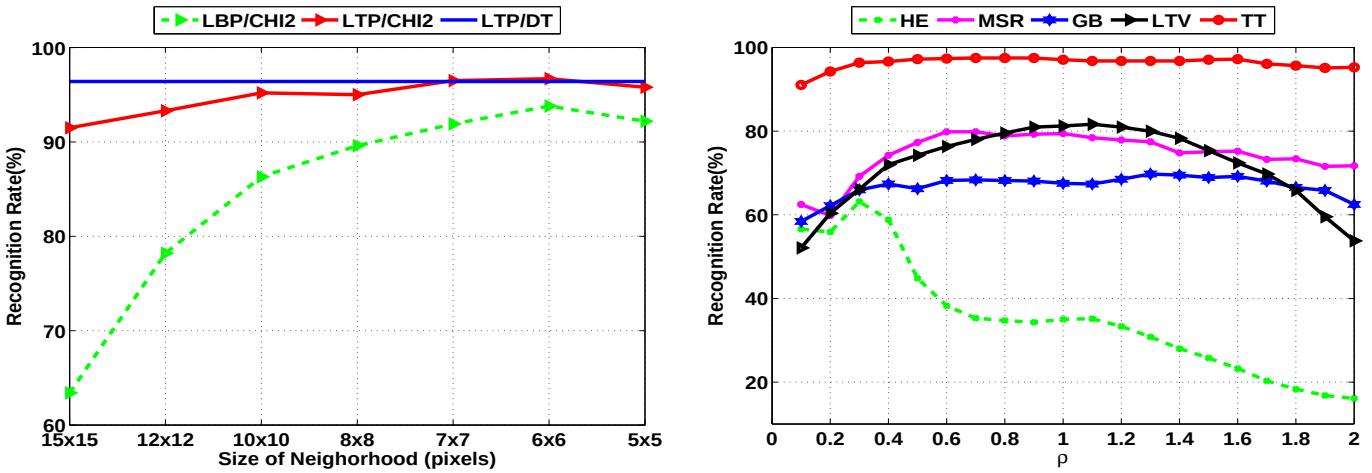

Fig. 13. (Left) The influence of the size of the LBP/LTP histogram binning neighborhood on recognition rate, here with TT preprocessing. (The rate for LBP/DT is 95.7% – only slightly below that shown for LTP/DT). (Right) the effect of different LTP quantization thresholds ρ on recognition rate (see the text for the details). Both graphs are for Extended YaleB subset 5.

training images, $\sigma()$ is their pixel standard deviation, and ρ is a small constant that needs to be set by hand for each preprocessor. In practice, we set ρ to a value between $[0.1, 2]$ for all preprocessors except LTV, for which we use $[0.01, 0.2]$ owing to the peakiness of the LTV output. Fig. 13 (right) shows the resulting performance on Extended Yale B set 5 (for convenience, the ρ values for LTV have been multiplied by 10 for display). We see that apart from HE, for which small values are preferred, all of the illumination methods tested are relatively insensitive to ρ. TT is both the best performer and the least sensitive to ρ.

D. Results for KLDA Subspace based Classifiers

CAS-PEAL-R1. The above Nearest Neighbour based classifiers give almost perfect results on Extended Yale B, but the best of them only scores 49.2% on CAS-PEAL-R1. This is in line with the state of the art – the best method tested in [12], LGBPHS [52], scored slightly more than 50%, while pixel based eigenfaces and fisherfaces respectively scored only 8.2% and 21.8% – but it is not very satisfying in absolute terms. CAS-PEAL-R1 is more difficult both because it contains 27 times more subjects than Extended Yale B, and because it has a greater degree of intrinsic variability owing to its more natural image capture conditions (less perfectly controlled illumination, pose, expression, *etc.*).

To do better, we replaced the Nearest Neighbour classifier with a kernel subspace (KLDA) based one and also generalized the feature set to contain both LBP and Gabor features. See section IV for a description of the resulting recognition framework. Fig. 14 (bottom left) shows the resulting overall face search performance (recognition rate within the first r responses). Including both Gabor and LBP features increases the rank-1 recognition rate by 30% relative to LBP features alone and by 10% relative to Gabor features alone, which suggests that the two feature sets do indeed capture different and complementary information. The resulting rank-1 recognition rate of 72.7% is more than 20% higher than the previous best method on this data set [52].

Fig. 14 (top left) presents rank-1 recognition rates on CAS-PEAL for the various preprocessing methods and feature sets. The combination of LBP and Gabor features gives better performance than the individual features under all six preprocessors (including 'None'). For Gabor features, MSR, GB, LTV and TT (our method) all significantly improve the performance relative to no preprocessing, whereas for LBP/LTP features, only our method has a clear positive effect (perhaps due to its inclusion of DoG filtering, which enhances small facial details). Histogram equalization (HE) actually reduces the performance for both feature sets, and LTV also reduces it for LBP/LTP features. Our preprocessor is the best method overall, beaten only by GB for pure Gabor features, and GB is again the second best.

Fig. 15 shows the performance of the various illumination preprocessors using Generalized Tensor Discriminative Analysis (GTDA) feature extraction with nearest neighbor and kernel subspace based classifiers. For comparison, the results for vector representation based KLDA (Vec/KLDA/NN) are also shown. We see that (at least with these settings) the tensor-based representations have slightly lower performance than the vector-based ones.

FRGC-204. Similar conclusions hold for the FRGC-204 data set. Fig. 14 (bottom right) shows that the proposed Gabor+LBP method increases the FVR at 0.1% FAR from about 80% for either Gabor or LBP features alone to 88.1%. This exceeds the state of the art on FRGC-204 [26] by over 12%, thus halving the error rate on this important data set.

Fig. 14 (top right) shows how the various preprocessing methods affect the performance of several combinations of visual features and learning methods on FRGC-204. Replacing Nearest Neighbours with KLDA greatly improves the performance of both LBP and Gabor features under all preprocessors, and in each case the combination of Gabor+LBP outperforms either of the corresponding individual features. Gabor+LBP also outperforms [26] for all preprocessors except LTV. In general, the inclusion of KLDA and/or multiple features decreases the performance differences between the

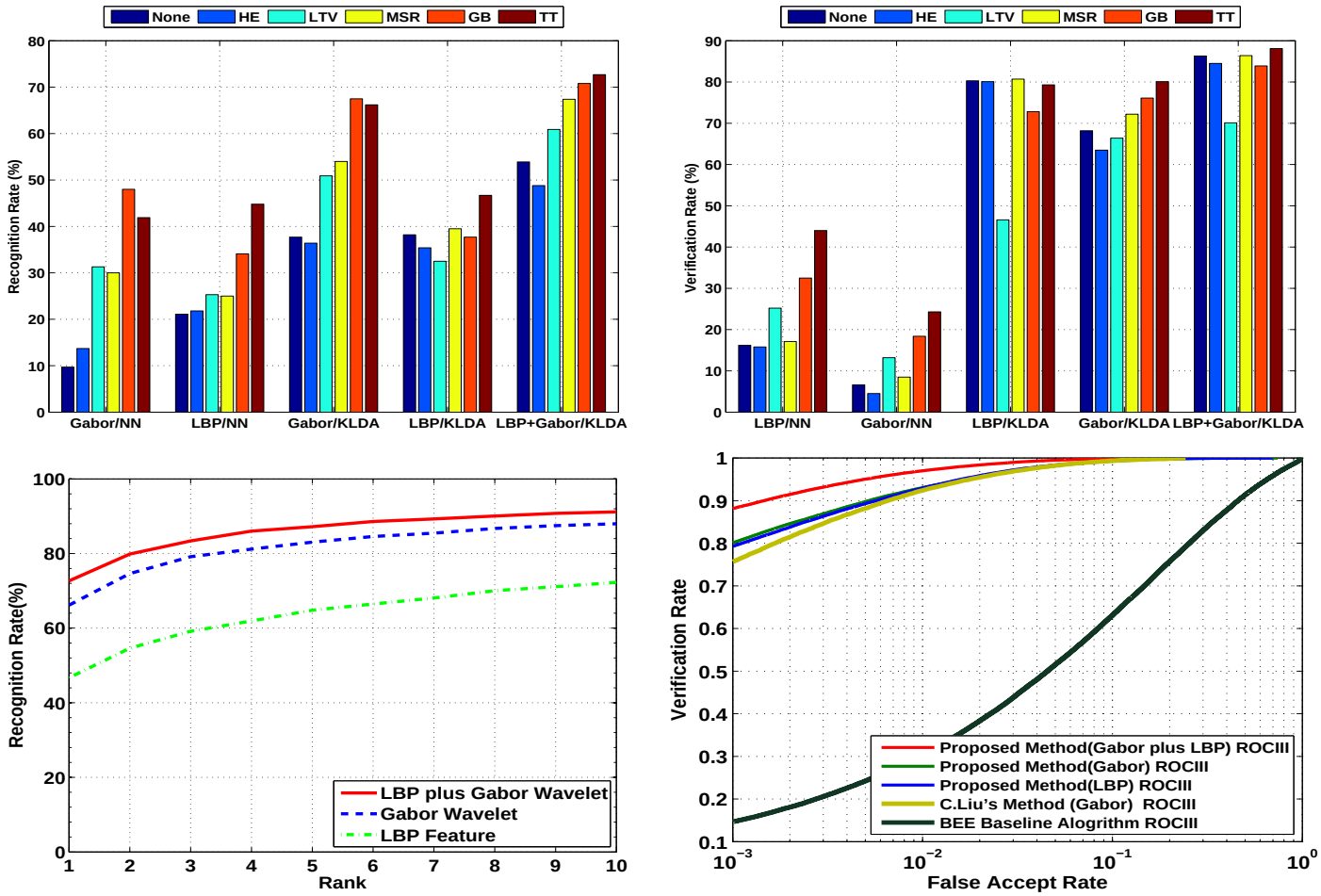

Fig. 14. Performance of the full KLDA-based methods on (left) CAS-PEAL-R1 and (right) FRGC-204. (Top) Recognition rates for several combinations of visual features and learning methods, under various preprocessing options, for (left) CAS-PEAL-R1, (right) FRGC-204 (FVR at 0.1% FAR). (Bottom left) Search performance (% of cases with the correct subject within the first N matches) on CAS-PEAL-R1 for the KLDA Gabor, LBP and Gabor+LBP methods. (Bottom right) ROC-III face recognition performance on FRGC-204 for the KLDA Gabor, LBP and Gabor+LBP methods. The BEE baseline and Liu's method [26] are also shown for comparison.

different preprocessing methods (or no preprocessing at all), except that the LTV preprocessor uniformly decreases the performance of all KLDA methods. Overall TT (our preprocessor) still does best, although under KLDA, MSR is marginally better than TT on LBP features and unpreprocessed images perform surprisingly well for both LBP and LBP+Gabor features (but not for Gabor alone).

Fig. 16 shows the influence of training set size on FRGC validation rates for the KLDA based methods. Keeping all 222 subjects while reducing the number of training images per subject has relatively little effect on performance until there are fewer than about 10 images per subject. Conversely, reducing the number of subjects while keeping a fixed number of images per subject causes a much more rapid deterioration. This suggests that (with our robust descriptors) the principal degree of variation in this dataset is identity not lighting related appearance changes.

Finally, we very briefly illustrate the contributions of the individual stages of our preprocessing chain on the FRGC-204 data set for various features and learning methods[7]. Fig 17 illustrates the effect of removing each of the four main stages of preprocessing in turn while leaving the remaining stages in place (the comparison is thus against our full preprocessor, not against no preprocessing). In general, each stage of preprocessing is beneficial and (not shown) the results are cumulative over the stages, but the benefits are much greater for Nearest Neighbour classifiers than for KLDA ones. The only case in which omitting a single stage of preprocessing actually improves the results is for DoG filtering with LBP features under KLDA, and the improvement in this case is slight (and to be contrasted with the large decrease that occurs when DoG is omitted from LBP under Nearest Neighbour classification). Also note that the last two stages of preprocessing involve monotone gray-level transformations and hence (as expected) have no effect on LBP features. We nevertheless include them

[7]We only present a small selection of our experimental results on preprocessing owing to lack of space. In general the experiments show that under nearest neighbour classification, each stage of preprocessing is beneficial for a broad range of features and distance metrics including pixel-based representations such as eigen- or fisher-faces, local filters such as Gabor features and texture histograms such as LBP/LTP.

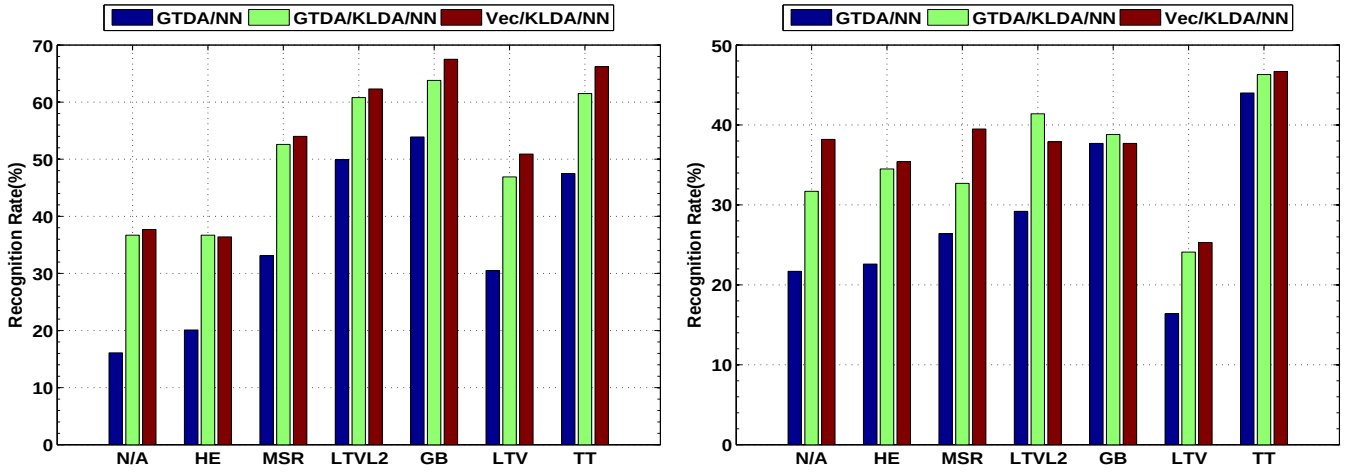

Fig. 15. Comparative performance of various illumination processors for (left) Gabor features and (right) LBP features, using Generalized Tensor Discriminative Analysis (GTDA) for dimensionality reduction in Nearest Neighbor (GTDA/NN) and KLDA-based Nearest Neighbour (GTDA/KLDA/NN) classifiers on the CAS-PEAL face dataset.

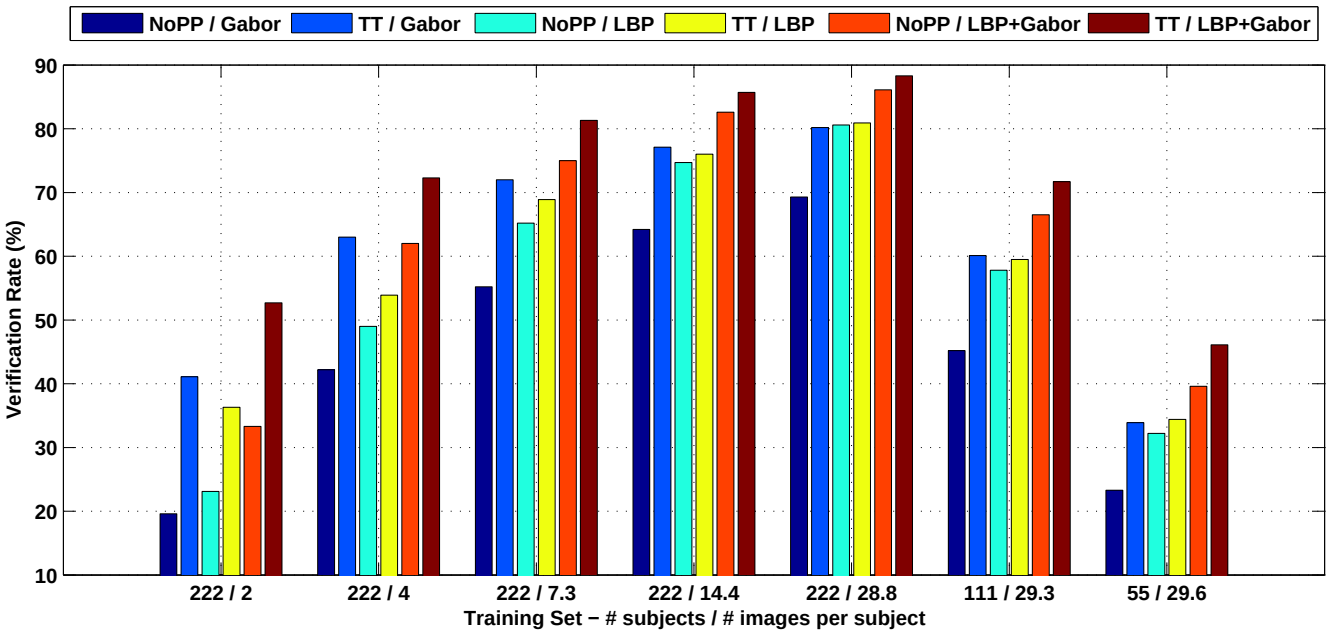

Fig. 16. Influence of the size of the training set on FRGC validation rates, for KLDA based methods with the Gabor, LBP and combined LBP+Gabor feature sets, with ('TT') and without ('NoPP') preprocessing. The full FRGC training set contains 222 subjects with an average of about 29 images per subject ('222 / 28.8' – the fifth group in the plot). The first four groups show that if we use all 222 subjects but reduce the number of training images by randomly selecting respectively 2, 4, an average of about 7 or an average of about 14 images per subject, the performance gradually decreases, but quite good results can still be obtained with about 10 images per subject. In contrast, the final two groups show that reducing the number of subjects quickly reduces the performance, even if we use all of the available images for them. In general, the performance differences between the different feature sets and preprocessing methods increase as the amount of training data is reduced: KLDA is quite effective at reducing these differences, but only when it has sufficient training data.

in our default preprocessor because they cause no harm for LBP and have a very beneficial effect on Gabor wavelets and a number of other feature sets for which we do not show results including LTP and pixel-based features such as eigen- and Fisher-faces.

E. Discussion

KLDA *vs*. preprocessing. The substantial performance gains produced by replacing nearest neighbour classification with KLDA on CAS-PEAL-R1 and FRGC-204 are welcome but not particularly surprising. These data sets have many sources of natural variation besides illumination (other imaging conditions, expression, ageing, *etc*.) and their galleries contain very few examples of each individual, giving Nearest Neighbour methods based on generic (non-learned) features and distance metrics little opportunity to generalize across these variations.

On the other hand, like Nearest Neighbours, KLDA is based on an underlying image similarity metric: the feature space distance embedded in its Gaussian kernel. Given the extent to which preprocessing improves Nearest Neighbours by providing a more illumination-resistant distance metric for comparisons, one might have hoped for analogous improve-

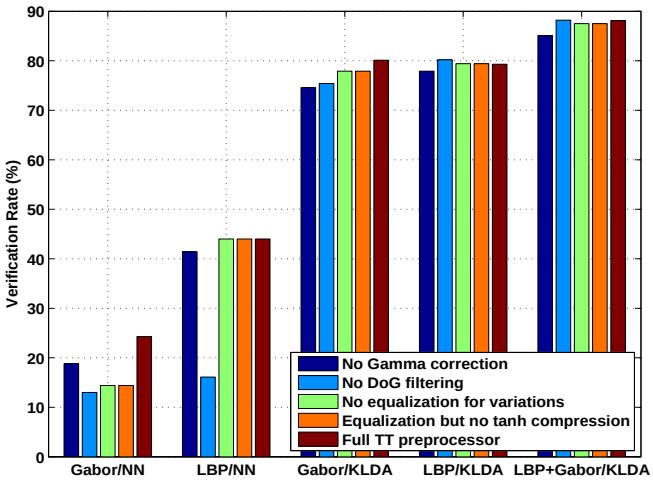

Fig. 17. Influence of the individual stages of our preprocessing chain. For various features and learning methods on the FRGC-204 data set, we compare the recognition rates (%) with our full preprocessing method to rates when each of the four main steps of preprocessing is removed in turn, leaving the remaining steps in place.

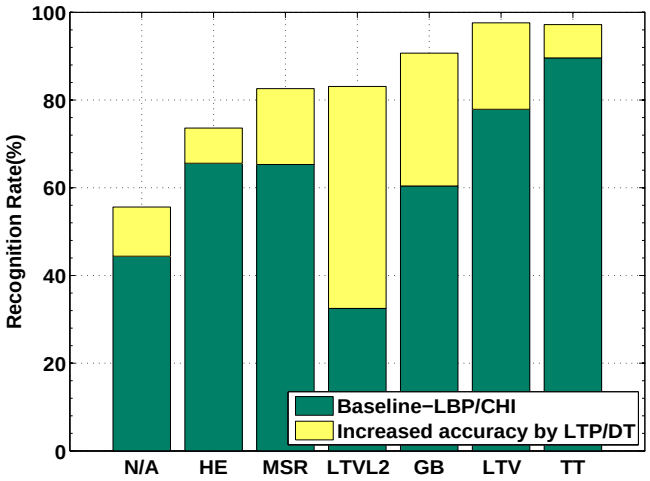

Fig. 18. Improvements due to replacing LBP/χ^2 with the less noise-sensitive features LTP and the more robust similarity metric DT, for various illumination preprocessors on Extended Yale B set 5.

ments under KLDA. In this respect – even allowing for the fact that facial lighting variations can be described quite well by rather low-dimensional models, *c.f. e.g.* [5,6,4] – KLDA's ability to compensate for the absence of preprocessing is somewhat surprising. Presumably, even though the supplied training data was not designed to systematically span the space of lighting variations, KLDA implicitly learns a nonlinear descriptor space lighting model that is more accurate than the default models that are implicitly embodied in the various preprocessors tested, thus producing a more accurate implicit "projection to an illumination invariant description". Saying this another way, rather than comparing each incoming example to a relatively large but stable set of illumination-invariant support vectors (nearby training examples after preprocessing), it seems to be better to compare them to a smaller but more variable set of non-invariant support vectors with similar lighting (nearby unpreprocessed training examples).

Choice of preprocessor. Both GB and LTV belong to the variational Retinex framework in which a plausible illumination image u is estimated by minimizing a functional combining smoothness and fidelity terms

$$u \;=\; \arg\min \int_{\text{image}} \|\nabla u\|^p + \lambda\,|f-u|^q \qquad (10)$$

and then the albedo image v (the preprocessor output) is estimated using Land's retinex formula $v = f/u$ (or equivalently ($\log v = \log f - \log u$). Here f is the input image, p, q are the orders of the regularization and fidelity norms, and λ is a data fidelity or roughness parameter. For GB, $p = q = 2$ (and the fidelity term is further weighted by Weber's contrast), while for LTV $p = q = 1$.

Retinex based approaches have proven very effective for illumination-invariant face recognition [13,10], but they do tend to amplify noise in dark areas owing to the division by u. For example this can be observed for GB in Fig. 5. Hence, it is preferable to use noise-insensitive feature sets with them – *e.g.* LTP or Gabor rather than LBP, *c.f.*, Fig. 12 (top left), Fig. 14 (top left) – and also robust similarity metrics – *e.g.*, DT, *c.f.*, Fig. 12 (top).

The 2-norms used in GB strongly penalize both large deviations $|f(x) - u(x)|$ and large gradients $\|\nabla u\|$ so the solution u tends to be determined predominantly by the large discontinuities in $f(x)$. This can lead to artificats in v such as haloing around edges.

In contrast, the 1-norms used in LTV give less haloing, but (depending on the setting of λ and the darkness of the image) they often allow a significant amount of fine scene texture to leak into u, thus suppressing it in the output image v. As Fig. 5 column 5 shows, even though LTV preserves facial details such as eyes to some extent, its output is dominated by a few patches of very high variation and many of the finer textures needed for recognition are suppressed, particularly for the more extreme illuminations. Despite this, with features that are robust to the size of illumination variations such as LBP and LTP, LTV does perform well on the challenging extended YaleB subset 5 tests.

In contrast, our TT preprocessor takes a signal processing approach motivated more by bottom-up human perception than by retinex theory, trying to estimate at least one useful part of the intrinsic (albedo) signal directly without passing via an illumination image u. In practice it manages to remove many of the illumination artifacts and provide a well-normalized output, while still preserving much of the textural detail that is needed for recognition.

To further illustrate these points, we give some additional results comparing LBP/χ^2 with LTP/DT for various preprocessors on Extended YaleB subset 5. For comparison we include results for the p=1, q=2 preprocessor, called TVL^2 or ROF in the literature [34] – here we call it LTVL2 (*c.f.* standard LTV is LTVL1). Fig. 18 shows that replacing LBP/χ^2 with the less noise-sensitive LTP features and the more robust DT similarity metric significantly improves the performance for all preprocessors.

Overall, our TT preprocessor seems to be the best choice:

it provides the best performance among the methods tested in almost all of our experiments, and it is also fast (at least a factor of 10 faster than GB and LTV) and very simple to implement. GB is the second choice for Nearest Neighbour or KLDA classification on datasets with relatively mild illumination variations (CAS-PEAL-R1, FRGC-204 and Extended Yale-B subsets 1-3). LTV performs poorly on these sets, but becomes competitive for Nearest Neighbour classification on sets with extreme lighting variations such as Yale-B subsets 4-5 – *c.f*. Fig. 12 (bottom right).

Facial alignment. Good alignment of the input image is essential for most feature sets for face recognition. Throughout our experiments, we simply aligned each image using a 2D similarity transform based on the eye coordinates shipped with the face database. Ruiz-del-solar *et al*. [18] have recently investigated the robustness of LBP-like feature sets to errors in eye positions, concluding that their performance remains acceptable so long as the relative position error is below about 5%. This should hold for all of the datasets used here. Note that none of the preprocessors studied here require accurate alignment to work. Thus, besides face recognition, they are likely to be useful tools for both face alignment and face detection under strong lighting variations, *c.f*. [39].

VI. Summary and Conclusions

We have presented new methods for face recognition under uncontrolled lighting based on robust preprocessing and an extension of the Local Binary Pattern (LBP) local texture descriptor. There are following main contributions: (*i*) a simple, efficient image preprocessing chain whose practical recognition performance is comparable to or better than current (often much more complex) illumination normalization methods; (*ii*) a rich descriptor for local texture called Local Ternary Patterns (LTP) that generalizes LBP while fragmenting less under noise in uniform regions: (*iii*) a distance transform based similarity metric that captures the local structure and geometric variations of LBP/LTP face images better than the simple grids of histograms that are currently used; and (*iv*) a heterogeneous feature fusion-based recognition framework that combines two popular feature sets – Gabor wavelets and LBP – with robust illumination normalization and a kernelized discriminative feature extraction method. The combination of these enhancements gives the state of the art performance on three well-known large-scale face datasets that contain widely varying lighting conditions.

Moreover, we empirically make comprehensive analysis and comparison with several state of the art illumination normalization methods on the large-scale FRGC-204 dataset, and investigate their connections with robust descriptors, recognition methods and image quality. This provides new insights into the role of robust preprocessing methods played in dealing with difficult lighting conditions and thus being useful in the designation of new methods for robust face recognition.

References

[1] Y. Adini, Y. Moses, and S. Ullman, "Face recognition: The problem of compensating for changes in illumination direction," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 19, no. 7, pp. 721–732, 1997.

[2] T. Ahonen, A. Hadid, and M. Pietikainen, "Face recognition with local binary patterns," in *European Conf. Computer Vision*, Prague, 2005, pp. 469–481.

[3] ——, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 28, no. 12, 2006.

[4] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 25, no. 2, pp. 218–233, February 2003.

[5] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[6] P. Belhumeur and D. Kriegman, "What is the set of images of an object under all possible illumination conditions," *Int. J. Computer Vision*, vol. 28, no. 3, pp. 245–260, 1998.

[7] G. Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics & Image Processing*, vol. 34, no. 3, pp. 344–371, 1986.

[8] R. Brunelli and T. Poggio, "Face recognition: Features versus Templates," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 15, no. 10, pp. 1042–1052, 1993.

[9] H. Chen, P. Belhumeur, and D. Jacobs, "In search of illumination invariants," in *CVPR*, 2000, pp. I: 254–261.

[10] T. Chen, W. Yin, X. Zhou, D. Comaniciu, and T. Huang, "Total variation models for variable lighting face recognition," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 28, no. 9, pp. 1519–1524, 2006.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, Washington, DC, USA, 2005, pp. 886–893.

[12] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The CAS-PEAL large-scale chinese face database and baseline evaluations," *IEEE Trans. Systems, Man and Cybernetics, Part A*, vol. 38, no. 1, pp. 149–161, 2008.

[13] R. Gross and V. Brajovic, "An image preprocessing algorithm for illumination invariant face recognition," in *AVBPA*, 2003, pp. 10–18.

[14] F. Guodail, E. Lange, and T. Iwamoto, "Face recognition system using local autocorrelations and multiscale integration," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 18, no. 10, pp. 1024–1028, 1996.

[15] X. He, X. Yan, Y. Hu, P. Niyogi, and H. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[16] Y. S. Huang and C. Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 17, no. 1, pp. 90–94, 1995.

[17] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognition*, vol. 38, no. 12, pp. 2270–2285, 2005.

[18] R. Javier, V. Rodrigo, and C. Mauricio, "Recognition of faces in unconstrained environments: a comparative study," *EURASIP J. Adv. Signal Process*, vol. 2009, pp. 1–19, 2009.

[19] D. Jobson, Z. Rahman, and G. Woodell, "A multiscale retinex for bridging the gap between color images and the human observation of scenes," *IEEE Trans. Image Processing*, vol. 6, no. 7, pp. 965–976, 1997.

[20] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[21] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Transactions Computers*, vol. 42, no. 3, pp. 300–311, 1993.

[22] S. Lawrence, C. Lee Giles, A. Tsoi, and A. Back, "Face

recognition: A convolutional neural-network approach," *IEEE Trans. Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.
[23] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 27, no. 5, pp. 684–698, 2005.
[24] ——, "Nine points of light: Acquiring subspaces for face recognition under variable lighting," in *CVPR*, 2001, pp. I:519–526.
[25] C. Liu, "Gabor-based kernel pca with fractional power polynomial models for face recognition," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 26, no. 5, pp. 572–581, 2004.
[26] ——, "Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 28, no. 5, pp. 725–737, 2006.
[27] C. Liu and H. Wechsler, "A shape- and texture-based enhanced fisher classifier for face recognition," *IEEE Trans. Image Processing*, vol. 10, no. 4, pp. 598–608, 2001.
[28] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller, "Fisher discriminant analysis with kernels," in *Neural Networks for Signal Processing IX*, Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, Eds. Piscataway, NJ: IEEE, 1999, pp. 41–48.
[29] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.
[30] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invarianat texture classification with local binary patterns," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
[31] Y. Pang, Y. Yuan, and X. Li, "Gabor-based region covariance matrices for face recognition," *IEEE Trans. Circuits & Systems for Video Technology*, vol. 18, no. 7, pp. 989–993, 2008.
[32] ——, "Iterative subspace analysis based on feature line distance," *IEEE Trans. Image Processing*, vol. 18, no. 4, pp. 903–907, 2009.
[33] P. J. Phillips, P. J. Flynn, W. T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. J. Worek, "Overview of the face recognition grand challenge," in *CVPR*, San Diego, CA, 2005, pp. 947–954.
[34] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1-4, pp. 259–268, 1992.
[35] S. Shan, W. Gao, B. Cao, and D. Zhao, "Illumination normalization for robust face recognition against varying lighting conditions," in *AMFG*, Washington, DC, USA, 2003, p. 157.
[36] J. Short, J. Kittler, and K. Messer, "A comparison of photometric normalization algorithms for face verification," in *IEEE Int. Conf. Automatic Face & Gesture Recognition*, 2004, pp. 254–259.
[37] L. Sirovich and M. Kirby, "Low dimensional procedure for the characterization of human faces," *J. Optical Society of America*, vol. 4, no. 3, pp. 519–524, 1987.
[38] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, "Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft kNN ensemble," *IEEE Trans. Neural Networks*, vol. 16, no. 4, pp. 875–886, 2005.
[39] X. Tan, F. Song, Z-H.Zhou, and S.Chen, "Enhanced pictorial structures for precise eye localization under uncontrolled conditions," in *CVPR*, 2009.
[40] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," in *AMFG*, 2007, pp. 168–182.
[41] ——, "Fusing gabor and LBP feature sets for kernel-based face recognition," in *AMFG*, 2007, pp. 235–249.
[42] D. Tao, X. Li, X. Wu, and S. Maybank, "General tensor discriminant analysis and gabor features for gait recognition," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 29, no. 10, pp. 1700–1715, 2007.
[43] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neurscience*, vol. 3, no. 1, pp. 71–86, 1991.
[44] H. Wang, S. Li, and Y. Wang, "Face recognition under varying lighting conditions using self quotient image," in *IEEE Int. Conf. Automatic Face & Gesture Recognition*, 2004, pp. 819–824.
[45] L. Wang and D. He, "Texture classification using texture spectrum," *Pattern Recognition*, vol. 23, pp. 905–910, 1990.
[46] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.
[47] S. Yan, H. Wang, X. Tang, and T. Huang, "Exploring feature descriptors for face recognition," in *IEEE Intl Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 2007, pp. 629–632.
[48] J. Yang, A. F. Frangi, J.-Y. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: A complete kernel fisher discriminant framework for feature extraction and recognition," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. 27, no. 2, pp. 230–244, 2005.
[49] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition," *IEEE Trans. Image Processing*, vol. 16, no. 1, pp. 57–68, 2007.
[50] L. Zhang and D. Samaras, "Face recognition under variable lighting using harmonic image exemplars," in *CVPR*, vol. 01, Los Alamitos, CA, USA, 2003, pp. 19–25.
[51] T. Zhang, D. Tao, X. Li, and J. Yang, "Patch alignment for dimensionality reduction," *IEEE Trans. Knowledge & Data Engineering*, vol. 21, no. 9, pp. 1299–1313, 2009.
[52] W. Zhang, S. Shan, W. Gao, and H. Zhang, "Local Gabor Binary Pattern Histogram Sequence (LGBPHS): A novel non-statistical model for face representation and recognition," in *ICCV*, Beijing, China, 2005, pp. 786–791.
[53] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys*, vol. 34, no. 4, pp. 399–485, 2003.

Xiaoyang Tan received his B.S. and M.S. degree in computer applications from Nanjing University of Aeronautics and Astronautics (NUAA) in 1993 and 1996, respectively. Then he worked at NUAA in June 1996 as an assistant lecturer. He received a Ph.D. degree from Department of Computer Science and Technology of Nanjing University, China, in 2005. From Sept.2006 to Oct.2007, He worked as a postdoctoral researcher in the LEAR (Learning and Recognition in Vision) team at INRIA Rhone- Alpes in Grenoble, France. His research interests are in face recognition, machine learning, pattern recognition, and computer vision.

Bill Triggs is a CNRS researcher who works mainly on machine learning based approaches to understanding images and other sensed data. He leads the AI (Apprentissage et Interfaces) team in the Laboratoire Jean Kuntzmann (LJK) in Grenoble, France, and he is also the deputy director of LJK, coordinator of the EU research project CLASS on unsupervised image and text understanding, and coordinator of the CNRS partner of the EU network of excellence PASCAL 2.