Preprocessing and Feature Sets for Robust Face Recognition

Xiaoyang Tan and Bill Triggs
LJK-INRIA, 655 avenue de l'Europe, Montbonnot 38330, France
Xiaoyang.Tan@inria.fr, Bill.Triggs@imag.fr

Abstract

Many face recognition algorithms have been developed over the past few years but the problem remains challenging, especially for images taken under uncontrolled lighting conditions. We show that the robustness of several popular linear subspace methods and of Local Binary Patterns (LBP) can be substantially improved by including a very simple image preprocessing stage based on gamma correction, Difference of Gaussian filtering and robust variance normalization. LBP proves to be the best of these features, and we improve on its performance in two ways: by introducing a 3-level generalization of LBP, Local Ternary Patterns (LTP), and by using an image similarity metric based on distance transforms of LBP image slices. We give experimental results on two face recognition sets chosen for their difficult lighting conditions: version 1 of the Face Recognition Grand Challenge experiment 4, and the full Yale-B dataset.

1. Introduction

There have been several decades of work on automatic face recognition [18]. Many algorithms have been developed and performances on benchmark datasets have steadily improved, albeit often at the cost of substantially increased computation and storage. However we believe that in the race for ever more sophisticated representations and learning methods, the community has tended to lose sight of the extent to which "trivial" image preprocessing influences the overall system performance. To illustrate this we present an empirical study of the effects of a preprocessing chain that we have developed on the performance of several popular face recognition algorithms on the Face Recognition Grand Challenge version 1 experiment 1.0.4 ('FRGC-104') [11] and Extended Yale illumination face database B ('Yale-B') [7, 8] datasets. These datasets were selected for their difficult lighting conditions, similar to those encountered in natural images taken under uncontrolled conditions. We also make two contributions to feature sets for visual recognition: a 3-valued generalization of Local Binary Patterns (LBP) [2] called Local Ternary Patterns (LTP) and a distance transform based similarity metric for LBP and LTP images, both of which improve on LBP's state-of-the-art recognition performance.

One conclusion from the FRVT and FRGC tests [11] was that illumination variations are one of the most important bottlenecks for practical face recognition systems. One can cope with such variations in three ways [15]. The first seeks local image features that are resistant to illumination changes, such as geometric features [5], image derivatives and 2D Gabor-like filtering methods [3]. The second uses training examples to learn a global model of the possible variations, for example a linear subspace or manifold model, which then generalizes to the variations seen in new images. This approach is currently very popular [7, 8]. The third approach seeks conventional image processing transformations that reduce the image to a more "canonical" form in which the variations are suppressed. It has the merit of ease of application to real images and the lack of a need for comprehensive training data. Traditional techniques such as histogram equalization belong to this category. Recently, several illumination normalization algorithms have been proposed [13] and normalization methods for component-based face verification are investigated in [14].

In this paper we adopt the "canonical form" philosophy. The key steps of our proposed normalization method are strong gamma compression followed by Difference of Gaussian (DoG) filtering, robust normalization of the range of output variations, and sigmoid-function based compression of any remaining signal highlights. We test the method with several appearance-based recognition algorithms on the FRGC-104 and Yale-B datasets, in all cases obtaining substantial increases in performance. The methods tested include classics such as Principal Component Analysis (PCA) [16, 17] with various different weighting functions and Linear Discriminant Analysis (LDA) [4], and local histograms of Local Binary Pattern (LBP) [2]. LBP is

Draft originally submitted to CVPR'07. This work was supported by the European Union funded research project CLASS.

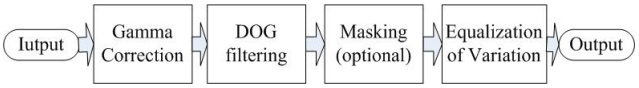


Figure 1. The block diagram of the proposed cascade of preprocessing steps.

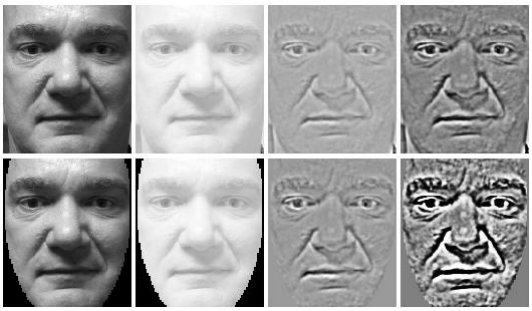


Figure 2. An example of the stages of our preprocessing chain. Upper row: a raw image is processed in turn by Gamma correction, DoG filtering and contrast equalization. Lower row: the same as the upper row except that the mask is applied after each step.

currently one of the best feature sets, giving excellent results on the FERET dataset [1]. It is invariant to monotonic gray-level transformations so it is generally considered that no preprocessing is needed when using it, but our results show that gamma compression and DoG filtering substantially improve its performance.

The rest of the paper is organized as follows. Our proposed preprocessing method is described in Section 2, and Local Ternary Patterns in Section 3. Section 4 presents our experimental results, and Section 5 concludes the paper.

2. Image Preprocessing for Robust Recognition

Our image preprocessing method is based on a series of steps that are designed to counter the effects of illumination variations, local shadowing and highlights, while still preserving the essential elements of visual appearance for use in recognition. The processing chain is shown in figure 1, and its effect on a typical face image is shown in figure 2. Although not done by design, the final chain is reminiscent of certain preprocessing stages found in the mammalian visual cortex. In detail, the steps are as follows. We resize the input images 80x88 before processing, and all of the pixel measurements below are relative to images of this size.

Gamma Correction. This 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. It has the effect of enhancing the local dynamic range of teh image in dark or shadowed regions, while compressing it in bright regions and at highlights. The basic principle is that the intensity of the light reflected from an object is the product of the incoming illumination (which is piecewise smooth for the most part) and the local surface reflectance (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. 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 potentially a useful visual cue but it is predominantly low frequency spatial 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 reduces aliasing and noise, and in practice often manages to do so without destroying too much of the underlying signal on which recognition needs to be based. DoG filtering is a convenient way to obtain the resulting bandpass behaviour. Spatial detail is important for recognition so typically the inner (smaller) Gaussian has a standard deviation σ_0 of one pixel or less, while the outer one might have a standard deviation σ_1 of $2 - 4$ pixels or more, depending on how reliable the low frequency information is. Given the strong lighting variations in our datasets we find that $\sigma_1 \approx 2$ typically gives the best performance, although values up to about 4 are not too damaging. The PCA and LDA methods need little or no image smoothing ($\sigma_0 \lesssim 0.25$), while for LBP and LTP $\sigma_0 \approx 1$ is preferred, perhaps because pixel based voting is sensitive to aliasing artifacts. Below we use $\sigma_1 = 2$ and $\sigma_0 = 0.25$ or 1.0 depending on the method.

We implement the filters using explicit convolution. 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[1]. Using extend-as-zero or wrap-around (FFT) boundary conditions

[1] As a curiosity, in some of our experiments with PCA and LDA we find that it helps to offset the centre of the larger filter by 1–2 pixels relative to the centre of the smaller one, so that the final prefilter is effectively the sum of a centred 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 futher investigation.

significantly reduces the overall performance, in part because it introduces strong gradients at the image borders which disturb the subsequent contrast equalization stage.

If DoG is run without prior gamma normalization, the resulting images clearly show the extent to which local contrast (and hence visual detail) is reduced in shadowed regions.

Masking. If a mask needs to be applied to suppress facial regions that are felt to be irrelevant or too variable, it should be applied after DoG filtering and before contrast equalization. Otherwise, either strong artificial grey-level edges are introduced into the convolution, or invisible regions are taken into account during contrast equalization. Some of our experiments on the FRGC-104 dataset use the standard elliptical mask supplied with the data to remove all but the central face region, but as the experiments show, some useful information is inevitably lost during this process.

Contrast Equalization. The final step of our preprocessing chain globally renormalizes the image intensities to standardize a robust measure of the overall contrast or intensity variation. We need to use a robust estimator of signal scatter because the signal typically still contains a small admixture of extreme values produced by highlights, garbage at the image borders and small dark regions such as nostrils. One could use, e.g. 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 optionally apply a nonlinear function that compresses over-large values. Here we use the hyperbolic tangent function $I(x,y) \leftarrow \tau \tanh(I(x,y)/\tau)$, which has the effect of limiting I to the range $(-\tau, tau)$.

3. Local Ternary Patterns (LTP)

3.1. Local Binary Patterns (LBP)

Ojiala et al. [9] introduced the Local Binary Pattern operator in 1996 as a means of summarizing local gray-level structure. The 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 codes based on the 8 pixels around the central one. Formally, the LBP operator takes the form

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

in this case, where n runs over the 8 neighbors of the central pixel c, i_c and i_n are respectively the grey-level values of c and n, and $s(u)$ is 1 if $u \geq 0$ and 0 otherwise.

LBP's are resistant to lighting effects in the sense that they are invariant to monotonic gray-level transformations, and they have been shown to have high discriminative power for texture classification [9]. However because they threshold exactly at i_c, they tend to be sensitive to noise, especially in near-uniform image regions. The LBP encoding process is illustrated in Fig.3.

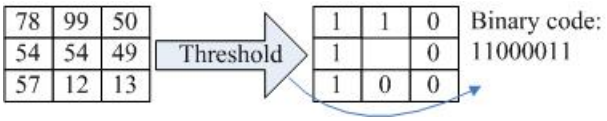


Figure 3. Illustration of the basic LBP operator.

Two important extensions to the original operator were made in [10]. The first defines LBP's for neighborhoods of different sizes, thus making it feasible to deal with textures at different scales. The second defines the so-called *uniform patterns*. An LBP is called 'uniform' if the it contains at most one 0-1 and one 1-0 transition when viewed as a circular bit string. For example, the LBP code shown in Fig.3 is uniform. Ojala et al. observed that although there are only 58 uniform 8-bit patterns, nearly 90 percent of all image neighbourhoods are uniform. In methods based on histogramming LBP's, the number of bins can thus be significantly reduced without much loss of information by assigning all non-uniform patterns to a single bin.

In [1], T. Ahonen et al. 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 a nearest neighbor classifier over the χ^2 histogram distance for recognition. Excellent results were reported on the FERET dataset [1]. Some recent advances in using LBP for face recognition are summarized in [2].

3.2. Local Ternary Patterns (LTP)

This section extends LBP to 3-valued codes, Local Ternary Patterns. It is motivated by the observation that because they threshold exactly at i_c, LBP's tend to be sensitive to noise, especially in uniform regions [12]. To alleviate this

problem, we introduce a zone $i_c \pm t$ around i_c in which the neighbour i_n is quantized to zero, quantizing it to $+1$ if it is above this and to -1 if it is below this, i.e. the indicator $s(u)$ is replaced by a 3-valued function

$$s(u) = \begin{cases} 1, & u \geq i_c + t \\ 0, & i_c - t < u < i_c + t \\ -1, & u \leq i_c - t \end{cases} \quad (4)$$

and the binary LBP code is replaced by a ternary LTP code. Here t is a user-specified threshold (so LTP codes more resistant to noise, but no longer invariant to gray-level transformations). The LTP encoding procedure is illustrated in Fig. 4, in which the threshold t is set to 5, so that the tolerance interval is $[49, 59]$.

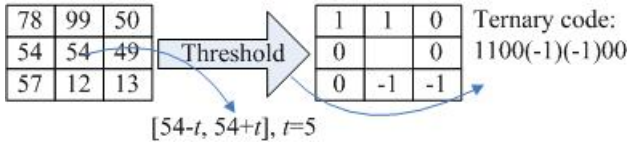


Figure 4. Illustration of the basic LTP operator.

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 we have used an LBP-based coding scheme in the experiments below, where each ternary pattern is split into positive and negative parts, each part is encoded as a separate LBP pattern, and the resulting patterns are concatenated, as illustrated in Fig.5.

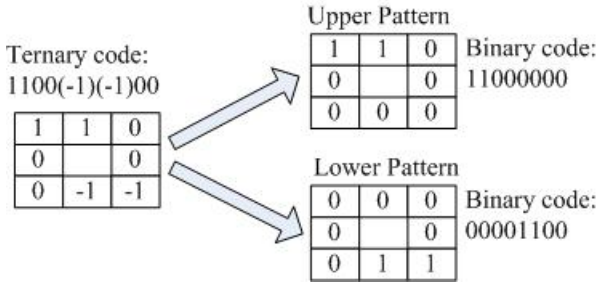


Figure 5. Illustration of the splitting of one basic LTP code into two standard LBP codes.

3.3. Distance Transform based Similarity Metric

We have also investigated a distance transform based metric for measuring the similarity between images represented by LBP or LTP codes. Each image is mapped to its image of LBP or LTP codes and viewed as a set of sparse binary layers, one for each LBP or LTP code value. A distance transform is run (separately) in each layer of the reference image, and the resulting distance values d are then mapped through a user-defined function $f()$. To compare an example image to the reference, the LBP or LTP layers from the example image and the resulting bitmaps are used to look up the corresponding $f()$ value in the corresponding reference layer. The $f()$ values are then summed over all image positions and layers to produce the similarity or distance metric. The advantage of this Hausdorf-distance like method is that it allows a degree of spatial looseness when matching the images, without introducing the abrubt spatial quantization inherent in grids of LBP histograms. In our experiments we have used the Euclidean distance transform for d and tested both similarity metrics based on Gaussians $f(d) = e^{-d^2/(2\sigma^2)}$ and distance metrics based on truncated linear distance $f(d) = \min(d, \tau)$. The performance is similar with the truncated distance method being slightly better overall. The default parameter values are $\sigma = 3$ pixels and $\tau = 6$ pixels.

4. Experiments

We now present experiments to evaluate the effectiveness of the proposed preprocessing method and the LTP representation. Two publicly available face recognition databases were used, Face Recognition Grand Challenge version 1 experiment 1.0.4 ('FRGC-104') [11] and the Extended Yale face database B ('Yale-B') [8]. Both databases cover a range of illumination conditions. Before presenting the results, we briefly review the recognition algorithms used.

4.1. Recognition Algorithms and Distance Measures

Three families of appearance-based face recognition algorithms were used to evaluate the proposed preprocessing chain. In all cases, a simple nearest neighbor rule is used to match the probes to the gallery, using a suitable distance metric on the extracted image features. The first two families, Principal Components Analysis (PCA) [16, 17] and Linear Discriminant Analysis (LDA) [4] are classical linear-subspace approaches. For LDA the usual L2 distance metric is used, while for PCA four different metrics were tested: L1, L2, cosine angle(COS), and whitened cosine angle (WC). These are defined as follows for two vectors x, y :

$$d_{L1}(x, y) = |x - y| \quad (5)$$

$$d_{L2}(x, y) = \|x - y\|^2 \quad (6)$$

$$d_{cos}(x, y) = -\frac{x \cdot y}{\|x\| \cdot \|y\|} \quad (7)$$

$$d_{WC}(x, y, W, \Sigma) = \frac{x^t \Sigma^{-1} y}{\|W^t x\| \|W^t y\|} \quad (8)$$

where W, Σ are respectively the whitening transformation and the covariance matrix of all of the samples [6].

LBP and LTP form the third family of methods. Three distance metrics were tested, the L1 and χ^2 distances and

Figure 6. Examples of images from FRGC-104. Upper row: target images. Middle row: query images. Lower row: preprocessed query images.

the above distance transform metric. The χ^2 distance is defined as follows:

$$\chi^2(x,y) = \Sigma_i \frac{(x_i - y_i)^2}{x_i + y_i} \quad (9)$$

4.2. Results on FRGC-104

The Face Recognition Grand Challenge (FRGC) provided several benchmark experiments for the face recognition community. Here we use only the experiment 4 of version 1 of FRGC. This experiment is challenging because – although the gallery images were obtained under carefully controlled conditions – the probe images were captured in uncontrolled indoor and outdoor settings. Some example images are shown in Fig.6. Specifically, the data is divided into 3 standard subsets: target and query sets each containing 943 images from 275 subjects, and a training set contain 366 face images from 123 subjects. All of the images in this data were used in our experiments, but only the PCA and LDA methods actually use the training set.

Fig. 7 demonstrates the overall value of our preprocessing chain by showing the rank-1 recognition performance of each method with and without proprocessing. The improvement is statistically significant for all methods. By way of comparison, the previous state of the art performance on this dataset (unfortunately for a different and hence not strictly comparable experimental setup) is 50.0% [6].

To illustrate the contribution of each individual preprocessing step, we removed each step in turn using otherwise optimal parameter settings. The results are shown in Fig. 8, for both unmasked and masked images. All of the methods benefit from gamma correction, but the benefit is greater for the linear subspace methods. As expected, the LBP methods are insensitive to the final equalization step, but in return they benefit markedly from the incorporation of DoG filtering. In contrast, LDA does not benefit from DoG and PCA is actually harmed by it in the L1 case, while both LDA and PCA benefit from equalization, especially the $\tanh()$ compression step. The introduction of masking reduces the overall performance, but the conclusions remain similar to those for the unmasked case.

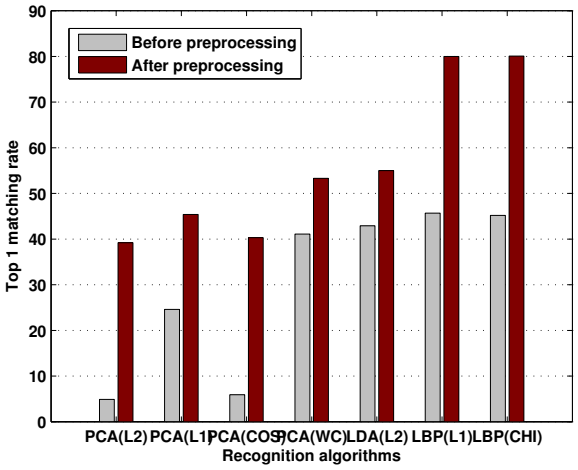


Figure 7. Comparison of the overall recognition performance on unmasked FRGC-104 data with and without preprocessing.

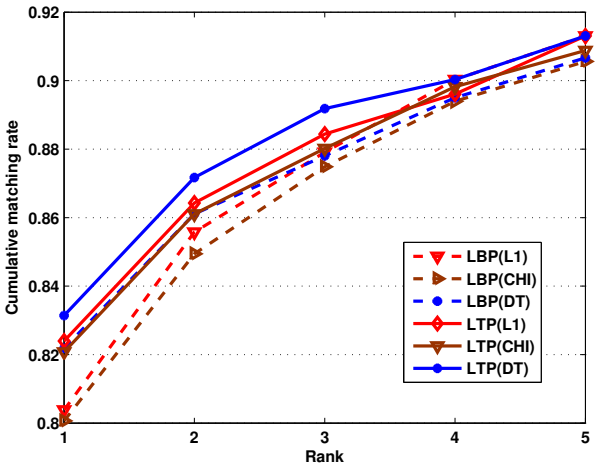
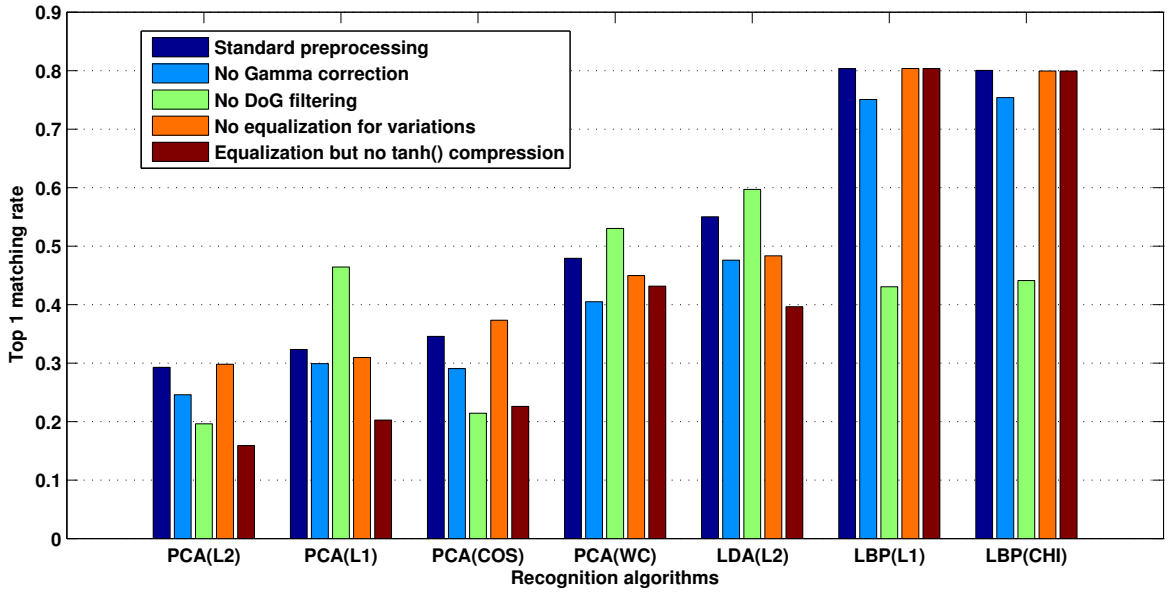


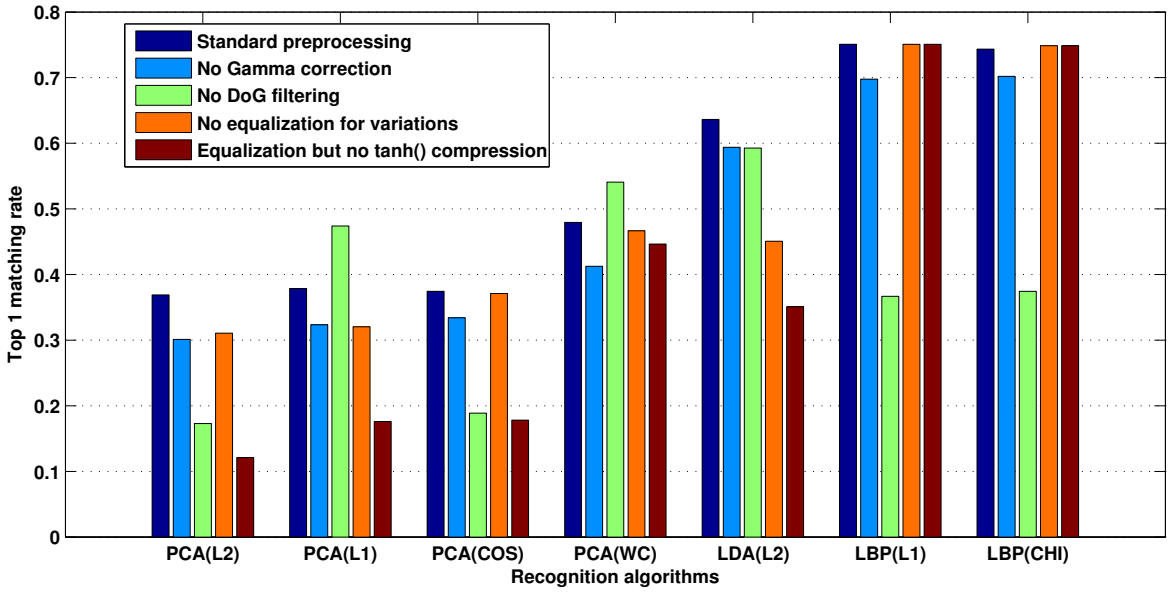
Figure 9. The benefits of replacing Local Binary Patterns with Local Ternary Patterns, and L1 or χ^2 similarity metrics with distance transform based similarity metrics.

Finally, we emphasize that different algorithms and distance measures have different degrees of sensitivity to the different stages of preprocessing: a preprocessing method can not be evaluated independently of its corresponding face recognition algorithm, and vice versa.

Our final experiment on FRGC-104 demonstrates the benefits of replacing Local Binary Patterns with Local



(a)



(b)

Figure 8. Influence of each normalization procedure on various recognition algorithms. (a) preprocessing without mask (b) preprocessing with mask.

Ternary Patterns, and L1 or χ^2 similarity metrics with distance transform based similarity metrics. The results are presented in Fig.9, which shows the percentage of trials for which the correct match is in the top k as a function of the rank k. Note that the LTP methods improve on LBP irrespective of the distance metric used. Here, the parameters for LTP are the same as those for LBP, with an LTP threshold of 0.09. This is for images preprocessed with our method – we find that LTP can sometimes be sensitive to the value of its threshold, so good contrast equalization is needed to benefit from it. The distance transform similarity function here is linear distance truncated at 6 pixels.

Figure 10. Examples of images of one person from the Yale-B frontal database. The rows respectively give images from subsets 1 to 5.

4.3. Results on Yale-B

To verify our conclusions on the FRGC-104 data, we conducted similar experiments on the Extended Yale face dataset B. The full dataset contains 21888 images of 38 human subjects under 9 poses and 64 illumination conditions. In this paper we are mainly concerned with illumination variations, so we used only the 2432 image subset containing the frontal images (38 people under 64 illuminations). Examples of one person in frontal pose are shown in Fig.10. The images are divided into five subsets according to the angle that the light source direction makes with the camera axis ($12°, 25°, 50°, 77°, 90°$). See [7] for details.

For our experiments on Yale-B, the "A+000E+00" images (the images with the most neutral light sources) for all 38 subjects were used as the gallery, and all images of each of the standard subsets 1-5 in turn were used as probes. Thus we have a small (38 image) gallery but several large probe sets that progressively become more and more challenging as the lighting angle increases. The results are shown in Table 1. Note that LDA could not be tested as there is only one training sample per person. As the table shows, our preprocessing method significantly improves the performance of the various algorithms, particularly the PCA based methods under extreme lighting conditions (subsets 4 and 5). Similarly, LTP and the distance transform similarity fucntion continue to improve on the performance of LBP under these conditions.

Table 1. A comparison of % Recognition Rates on the Yale-B database with ('P') and without ('N') preprocessing.

		Subset No. (number of probes)				
		1 (363)	2 (456)	3 (455)	4 (526)	5 (714)
PCA(L2)	N	93.0	89.0	20.0	3.8	2.8
	P	89.4	100	**85.1**	**87.8**	**88.0**
PCA(L1)	N	99.0	92.0	22.9	3.9	2.8
	P	90.9	100.0	**85.3**	**87.8**	**84.2**
PCA(cos)	N	95.0	89.0	20.9	4.8	2.9
	P	90.9	100.0	**87.0**	**88.6**	**87.1**
PCA(WC)	N	100.0	97.0	43.5	8.4	3.4
	P	95.4	100.0	**89.7**	**90.9**	**86.1**
LBP(L1)	N	100.0	100.0	96.9	63.3	51.3
	P	97.7	100.0	**97.6**	**98.2**	**92.7**
LBP(CHI)	N	100.0	100.0	97.4	66.7	54.6
	P	97.7	100	**98.0**	**98.3**	**93.3**
LTP(L1)	P	97.7	100.0	**98.0**	**99.0**	**94.0**
LTP(CHI)	P	97.7	100.0	**98.0**	**99.2**	**94.1**
LBP(DT)	P	100.0	100.0	**100.0**	**99.2**	**97.2**
LTP(DT)	P	100.0	100.0	**100.0**	**99.2**	**97.2**

5. Conclusion

We presented a general preprocessing method for face images that significantly improves the performance of a number of standard recognition algorithms. The main components are gamma correction, Difference of Gaussian filtering, robust contrast equalization and nonlinear clipping of extreme values. We demonstrated the contribution of each step and studied the relationships between the preprocessor and the recognition algorithm, concluding that these can not be chosen independently.

Secondly, we proposed a novel feature set for recognition called Local Ternary Patterns, that extends the state-of-the-art Local Binary Pattern features by including 3-valued coding with a threshold for improved resistance to noise. We also introduced a distance transform based similarity metric for LBP and LTP images that significantly improves on the χ^2 histogram distance.

In future work, we will extend these methods to problems requiring robust face alignment, and explore their potential in other applications such as texture recognition.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *Proceedings of the 2004 European Conference on Computer Vision (ECCV'04)*, pages 469–481, Prague, 20054. 2, 3

[2] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), 2006. 1, 3

[3] O. Arandjelovic and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 860–867, Washington, DC, USA, 2005. 1

[4] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997. 1, 4

[5] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993. 1

[6] C.Liu. The bayes decision rule induced similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, to appear, 2006. 4, 5

[7] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001. 1, 7

[8] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 27(5):684–698, 2005. 1, 4

[9] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29, 1996. 3

[10] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invarianat texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. 3

[11] 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 *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 947–954, San Diego, CA, 2005. 1, 4

[12] Y. Rodriguez and S. Marcel. Face authentication using adapted local binary pattern histograms. In *9th European Conference on Computer Vision (ECCV)*, 2006. 3

[13] S. Shan, W. Gao, B. Cao, and D. Zhao. Illumination normalization for robust face recognition against varying lighting conditions. In *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, page 157, Washington, DC, USA, 2003. 1

[14] J. Short, J. Kittler, and K. Messer. Photometric normalisation for component-based face verification. pages 114–119, 2006. 1

[15] T. Sim and T. Kanade. Combining models and exemplars for face recognition: An illuminating example. In *Proceedings of the CVPR 2001 Workshop on Models versus Exemplars in Computer Vision*, December 2001. 1

[16] L. Sirovich and M. Kirby. Low dimensional procedure for the characterization of human faces. *Journal of Optical Society of America*, 4(3):519–524, 1987. 1, 4

[17] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neurscience*, 3(1):71–86, 1991. 1, 4

[18] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Survey*, 34(4):399–485, 2003. 1