Face Recognition with Learning-based Descriptor

Zhimin Cao[1] Qi Yin[2*] Xiaoou Tang[1,3] Jian Sun[4]

[1]The Chinese University of Hong Kong [2]ITCS, Tsinghua University [3]Shenzhen Institutes of Advanced Technology Chinese Academy of Sciences, China [4]Microsoft Research Asia

Abstract

We present a novel approach to address the representation issue and the matching issue in face recognition (verification). Firstly, our approach encodes the micro-structures of the face by a new learning-based encoding method. Unlike many previous manually designed encoding methods (e.g., LBP or SIFT), we use unsupervised learning techniques to learn an encoder from the training examples, which can automatically achieve very good tradeoff between discriminative power and invariance. Then we apply PCA to get a compact face descriptor. We find that a simple normalization mechanism after PCA can further improve the discriminative ability of the descriptor. The resulting face representation, learning-based (LE) descriptor, is compact, highly discriminative, and easy-to-extract.

To handle the large pose variation in real-life scenarios, we propose a pose-adaptive matching method that uses pose-specific classifiers to deal with different pose combinations (e.g., frontal v.s. frontal, frontal v.s. left) of the matching face pair. Our approach is comparable with the state-of-the-art methods on the Labeled Face in Wild (LFW) benchmark (we achieved 84.45% recognition rate), while maintaining excellent compactness, simplicity, and generalization ability across different datasets.

1. Introduction

Recently, face recognition has attracted much research effort [9, 25, 26, 10, 11, 13, 19, 20, 30, 32] due to the progresses of local descriptors [6, 16, 17, 22, 27, 28, 29, 30] and increasing demands of real-world applications, such as face tagging on the desktop [5] or the Internet[1]. There are two main kinds of face recognition tasks: face identification (who is who in a probe face set, given a gallery face set) and face verification (same or not, given two faces). In this paper, we focus on the verification task, which is more widely applicable and is also the foundation of the identification task.

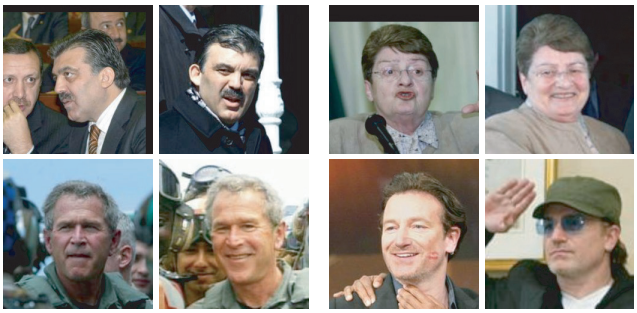

Figure 1. Images from the same person may look quite different due to pose (upper left), expression (upper right), illumination (lower left), and occlusion (lower right).

Since face verification is a binary classification problem on an input face pair, there are two major components of a verification approach: face representation and face matching. The extracted feature (descriptor) is required to be not only discriminative but also invariant to apparent changes and noise. The matching should be robust to variations from pose, expression, and occlusion, as shown in Figure 1. These requirements render face verification a challenging problem.

Currently, descriptor-based approaches [10, 20, 31] have been proven to be effective face representations producing best performance [8, 18, 12]. Ahonen *et al*. [1] proposed to use the histogram of Local Binary Pattern (LBP) [17] to describe the micro-structures of the face. LBP encodes the relative intensity magnitude between each pixel and its neighboring pixels. It is invariant to monotonic photometric change and can be efficiently extracted. Since LBP is encoded by a handcrafted design, many LBP varieties [21, 30, 33] have been proposed to improve the original LBP. SIFT [16] or Histogram of Oriented Gradients (HOG) [6] are other kinds of effective descriptors using handcrafted encoding. The atomic element in these descriptors can be viewed as the quantized code of the image gradients. Essentially, different encoding methods and descriptors have to balance between the discriminant power

*Yin is affiliated with the Institute for Theoretical Computer Science (ITCS).

[1]Picasa Web Albums, *http://picasaweb.google.com/*

and the robustness against data variance.

However, existing handcrafted encoding methods suffer two drawbacks. On one hand, manually getting an optimal encoding method is difficult. Usually, using more contextual pixels (higher dimension vector) can generate a more discriminative code. But it is non-trivial to manually design an encoding method and determine the codebook size to achieve reasonable tradeoff between discrimination and robustness in a high dimension space. In addition, handcrafted codes are usually unevenly distributed as shown in Figure 2. Some codes may rarely appear in real-life face images. It means that the resulting code histogram will be less informative and less compact, degrading the discriminant ability of the descriptor.

In this paper, to tackle the aforementioned difficulties, we present a learning-based encoding method, which uses unsupervised learning methods to encode the local microstructures of the face into a set of discrete codes. The learned codes are more uniformly distributed (as shown in Figure 2) and the resulting code histogram can achieve much better discriminative power and robustness tradeoff than existing handcrafted encoding methods. Furthermore, to pursue the compactness, we apply the dimension reduction technique, PCA, to the code histogram. And we find a proper normalization mechanism after PCA can improve the discriminative ability of the code histogram. Using two simple unsupervised learning methods, we obtain a highly discriminative and compact face representation, the learning-based (LE) descriptor.

Many recent researches also apply learning approaches in face recognition, such as subspace learning [25, 26], metric learning [9], high-level trait learning [13], discriminant model learning [20, 30, 31], but few of these works focus on the issue of local feature encoding [14, 24] and the study of descriptor compactness. Though Ahonen *et al*. [2] tried K-means cluster to build local filter response codebook, they argued manual thresholding is faster and more robust.

Besides the representation, the matching also plays an important role. In most practices, the face is aligned by a similarity or affine transformation using detected face landmarks. Such 2D holistic alignment is not sufficient to handle large pose deviations from the frontal pose. Further, the large localization error of any landmark will result in misalignment of the whole face. 3D alignment [3] is more principled but error-prone and computationally intensive. Wright *et al*. [32] recently encoded the geometric information into descriptors and used an implicit matching algorithm to deal with the misalignment and pose problem. Gang [10] demonstrated that a simple elastic and partial matching metric can also handle pose change and clutter background.

To explicitly handle large pose-variance, we propose a pose-adaptive matching method. We found that a specific face component contributes differently when the pose combinations of input face pairs are different. Based on this observation, we train a set of pose-specific classifiers, each for one specific pose combination, to make the final decision.

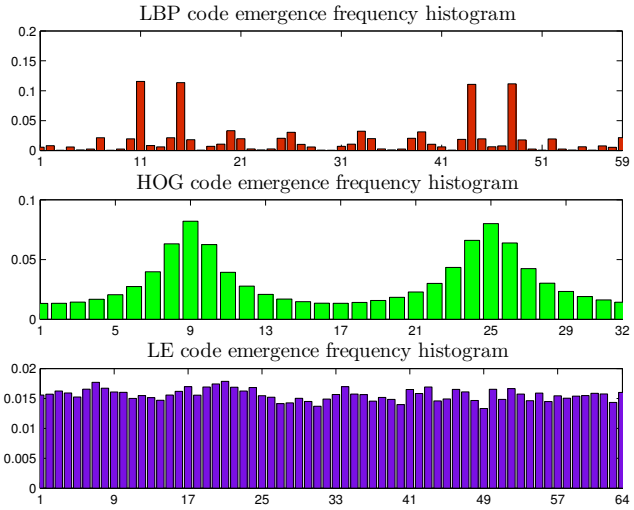

Figure 2. The code uniformity comparison of LBP, HOG, and the proposed LE code. We computed the distribution of code emergence frequency for LBP (59 uniform codes), HOG (32 orientation bins) and LE (64 codes) in 1000 face images. Clearly, the histogram distribution is uneven for LBP and HOG while our LE code is close uniform.

Combining a powerful learning-based descriptor and a pose-adaptive matching scheme, our system achieves the leading performance on both the LFW [12] and the Multi-PIE [8] benchmarks.

2. Overview of framework

Pipeline overview. Our system is a two-level pipeline: the upper-level is the learning-based descriptor pipeline while the bottom-level is the pose-adaptive face matching pipeline.

As shown in Figure 3, we first use a standard fiducial point detector [15] to extract face landmarks. Nine different components (e.g., nose, mouth) are aligned separately based on detected landmarks. The resulting component images are fed into a DoG filter (with $\sigma_1 = 2.0$ and $\sigma_2 = 4.0$) [10] to remove both low-frequency and high-frequency illumination variations. In each component image, a low-level feature vector is obtained at each pixel and encoded by our learning-based encoder. The final component representation is a compact descriptor (LE descriptor) generated by the concatenated patch histogram of the encoded features after PCA reduction and normalization. The component similarity is measured by L_2 distance between corresponding LE descriptors of the face pair. The resulting 9 component similarity scores are fed into a pose-adaptive classifier,

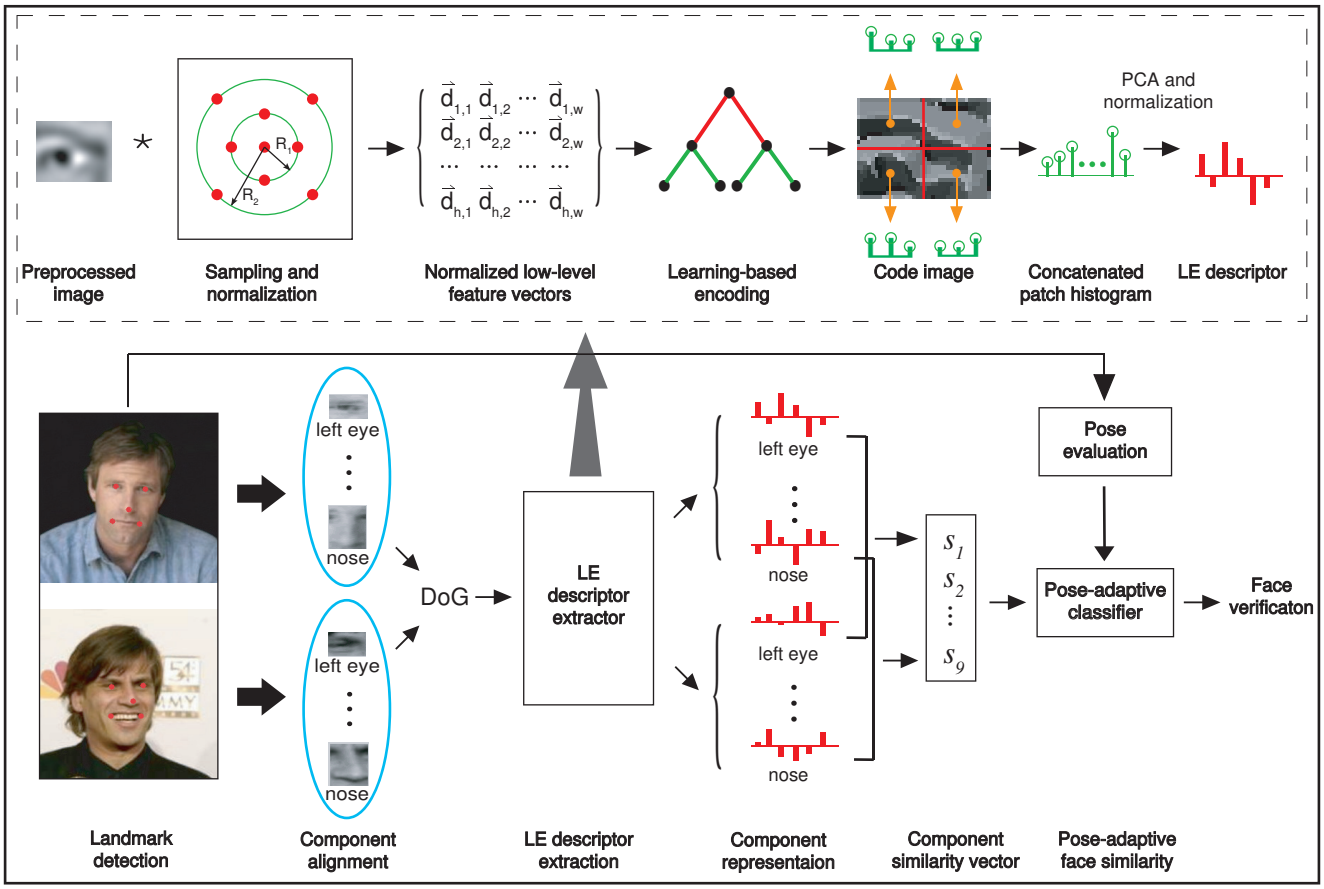

Figure 3. The proposed LE descriptor pipeline and the pose-adaptive face matching framework.

consisting of a set of pose-specific classifiers. The pose-specific classifier optimized to the pose combination of the matching pair gives the final decision.

Experiment overview. We mainly use the LFW benchmark [12] in our experiments and follow their protocol. The LFW standard test set consists of ten subsets and each subset contains 300 intra-personal/extra-personal pairs. The recognition algorithm needs to run ten times for formal evaluation purpose. At each time, one subset is chosen for testing and the other nine are used for training. The final average recognition performance serves as the evaluation criterion.

3. Learning-based descriptor extraction

In this section, we describe the critical steps in the learning-based (LE) descriptor extraction. In order to study the LE descriptor's power precisely, all the experiments in this section are conducted in holistic face level, without using component-level pose adaptive matching.

3.1. Sampling and normalization

At each pixel, we sample its neighboring pixels in the ring-based pattern to form a low-level feature vector. We sample $r * 8$ pixels at even intervals on the ring of radius r. Figure 4 shows four effective sampling patterns we found in an empirical manner. We extensively varied the parameters (*e.g.*, ring number, ring radius, sampling number of each ring) but found the differences among good patterns are not significant - no more than 1% on the LFW benchmark. The 2^{nd} pattern in Figure 4 is our best single pattern and we use it as our default sampling method.

Although the performances of single patterns are similar, combining them together may give us a chance to exploit the complementary information captured by different sampling methods. We will discuss the use of multiple patterns later in this section.

After the sampling, we normalize the sampled feature vector into unit length. Such normalization combined with DoG preprocessing makes the feature vector invariant to local photometric affine change.

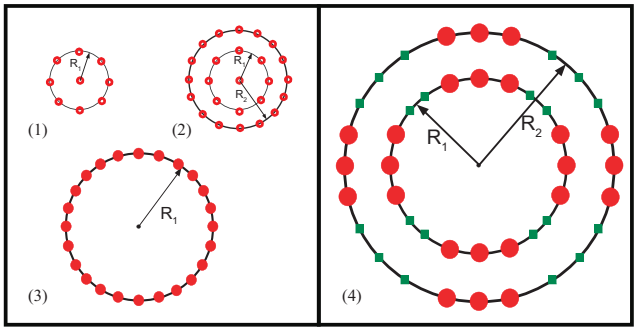

Figure 4. Four typical sampling methods used in our experiments: (1) $R_1 = 1$, with center; (2) $R_1 = 1, R_2 = 2$, with center; (3) $R_1 = 3$, no center; (4) $R_1 = 4, R_2 = 7$, no center. (The sampling dots on the green-square labeled arcs are omitted for better visuality).

3.2. Learning-based encoding and histogram representation

Next, an encoding method is applied to encode the normalized feature vector into discrete codes. Unlike many handcrafted encoders, in our approach, the encoder is specifically trained for the face in an unsupervised manner from a set of training face images. We have tried three unsupervised learning methods: K-means, PCA tree [7], and random-projection tree [7]. While K-means is commonly used to discover data clusters, random-projection tree and PCA tree are recently proved effective for vector quantization. In our implementation, random-projection tree and PCA tree recursively split the data based on uniform criterion, which means each leaf of the tree is hit by the same number of vectors. In other words, all the quantized codes have a similar emergence frequency in the vector space (as shown in Figure 2).

After the encoding, the input image is turned into a "code" image (Figure 3). Following the method described in Ahone *et al.*'s work [1], the encoded image is divided into a grid of patches (5×7 patches for the holistic face (84×96) used in this section). A histogram of the LE codes is computed in each patch and the patch histogram is concatenated to form the descriptor of the whole face image.

The choice of the learning method and the code number are important for our learning-based encoding. Figure 5 shows the performance comparison of the three learning methods under different code number setting. We select 1,000 images from the LFW training set to train our learning-based encoders. On each image, a number of 8,064 (=84×96) feature vectors are sampled as training examples. We varied the code number from 4 to 131,072 (=2^{17}) and plotted the recognition rate (we stopped testing K-means after reaching 2^9 codes since the computation becomes intractable). Notice that random-projection tree slightly outperforms the other two and thus is adopted in the following as default. We compare our LE descriptor with LBP (59-bin), HOG (8-bin), and Gabor [29] on the LFW. Our LE descriptors start to beat existing descriptors (LBP 72.35%, HOG 71.25%, and Gabor 68.53%) when the code number reaches 32. And our LE descriptor achieves 77.78% rate when the code number reaches 2^{15}.

3.3. PCA dimension reduction

If we use the concatenated histogram directly as the final descriptor, the resulting face feature may be too large (e.g., 256 codes $\times$ 35 patch = 8,960 dimension). A large feature not only limits the number of faces which can be loaded into memory, but also slows down the recognition speed. This is very important for the applications that need to handle a large number of faces, for example, recognizing all face photos on a desktop. To reduce the feature size, we apply Principle Component Analysis (PCA) [23] to compress the concatenated histogram, and call the compressed descriptor as our final learning-based (LE) descriptor.

Surprisingly, we found that PCA compression substantially improves the performance if a simple normalization is applied after the compression. Figure 6 shows the recognition rates of LE descriptors with different normalization methods. Without the normalization, the compressed feature is inferior to the uncompressed one by 6% points. But with L_1 or L_2 normalization, the PCA version can be 5% higher. This result reveals the angle difference between features is most essential for the recognition in the compressed space. To confirm this key observation, we also tried to apply PCA compression to LBP. We repeated the same compression and normalization operations and also found simple normalization can boost uncompressed LBP's performance 3% points while skipping such step will detract it 5% points.

To obtain the optimal setting for the LE descriptor, we extensively studied the parameter combination of code number and PCA dimension. For large code number shows little performance advantage after PCA compression, we choose 256 code and 400 PCA-dimension as our default setting in the following experiments.

Our default LE descriptor achieves recognition rate as high as 81.22%, which significantly outperforms previous descriptors, using only 400-dimension feature vector for the holistic face, about 20% the size of the 59-code LBP descriptor. This demonstrated that our descriptor extraction pipeline (pre-processing, sampling and normalizing, learning-based encoding, and dimension reduction) is very effective for producing a compact and highly discriminative descriptor.

3.4. Multiple LE descriptors

As discussed in Section 3.1, our flexible sampling method enables us to generate a class of complementary LE

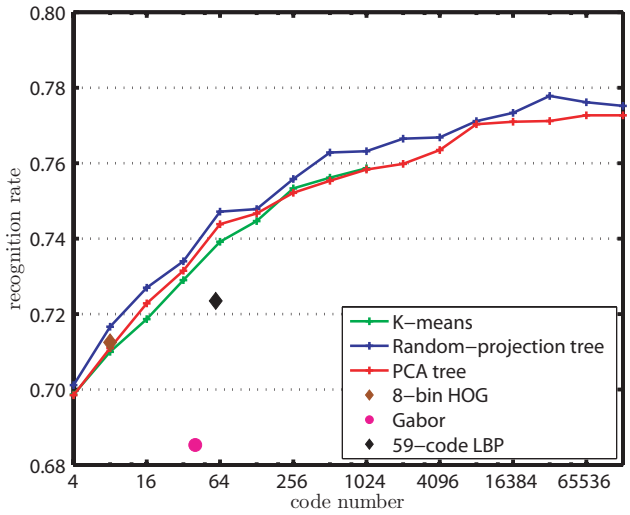

Figure 5. Performance comparison vs. learning method. We studied the recognition performance of the LE descriptors using three learning methods (random projection tree, PCA-tree, and K-means) under different code number settings. We also gave several existing descriptors' results for comparison.

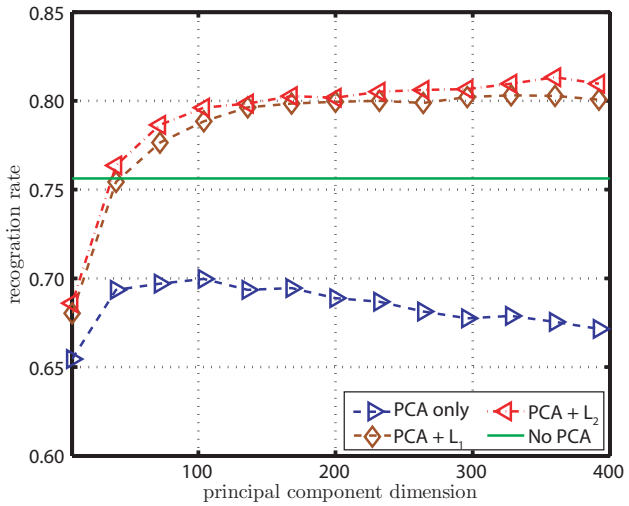

Figure 6. Investigate the effects of the PCA dimension with different normalization methods. After applying PCA compression to the concatenated patch histogram vector, we normalize the resulting vector with different normalization methods and then compute the similarity score with L_2 distance.

descriptors, and the combination of multiple LE descriptors may achieve better performance. In this paper, we take a simple approach by training a linear SVM [4] to combine the similarity scores generated by different LE descriptors. Generally, the combination can always achieve better result. In our experiments, the combination of four LE descriptors (shown in Figure 4) obtained the best performance on the LFW. Figure 7 gives the comparison curves of different descriptors.

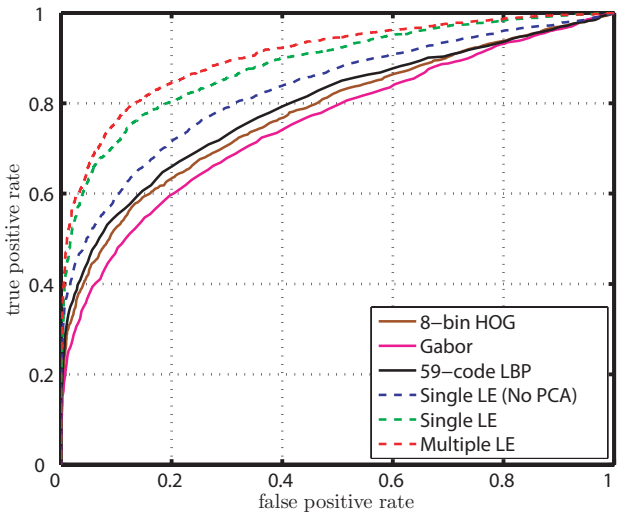

Figure 7. ROC curve comparison between our LE descriptors and existing descriptors.

4. Pose-adaptive matching

In the previous section, we use 2D holistic alignment and matching for the comparison purpose. In this section, we will show that a pose-adaptive matching at the component-level can effectively handle large pose variation and further boost the recognition accuracy.

4.1. Component-level face alignment

Instead of using a 2D holistic (similarity) alignment on the whole face, we align 9 face components (shown in Figure 8) separately using similarity transform. For each component, two landmarks are selected from the five detected fiducial landmarks (eyes, nose, and mouth corners) to determine the similarity transformation (details in Table 1). Compared with the 2D holistic alignment, the component-level alignment presents advantages in large pose-variant case. The component-level approach can more accurately align each component without balancing across the whole face. And the negative effect of landmark error will also be reduced. Figure 8 shows aligned components and Table 2 compares the performance of different alignment methods.

4.2. Pose-adaptive matching

Using the component-level alignment, the face similarity score is the sum of similarities between corresponding components. We found that each component contributes differently for the recognition when the pose combination of the matching pair is different. For example, the left eye is less effective when we match a frontal face and a left-turned face. Based on this observation, we take a simple pose-adaptive matching method.

Firstly, we categorize the pose of the input face to one of

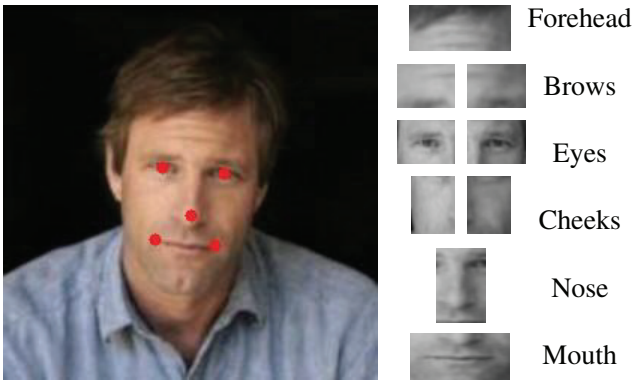

Figure 8. Fiducial points and component alignment.

Component	Selected landmarks
Forehead	left eye + right eye
Left eyebrow	left eye + right eye
Right eyebrow	left eye + right eye
Left eye	left eye + right eye
Right eye	left eye + right eye
Nose	nose tip + nose pedal*
Left cheek	left eye + nose tip
Right cheek	right eye + nose tip
Mouth	two mouth corners

Table 1. Landmark selection for component alignment. (* means the pedal of the nose tip on the eye line.)

Alignment mode	Recog. rate
2-Point holistic	$79.85\% \pm 0.42\%$
5-Point holistic	$81.22\% \pm 0.53\%$
Component	$82.73\% \pm 0.43\%$

Table 2. Recognition rate vs. alignment mode.

three poses (frontal (F), left (L), and right (R)). To handle this pose category, three images are selected from the Multi-PIE dataset, one image for each pose, and the other factors in these three images, such as person identity, illumination, expression remain the same. After measuring the similarity between these three gallery images and the probe face, the pose label of the most alike gallery image is assigned to the probe face.

Given the estimated pose of each face, the pose combinations of a face pair could be {FF, LL, RR, LR (RL), LF (FL), RF (FR)}. Our final pose-adaptive classifier consists of a set of linear SVM classifiers, each trained by a subset of training pairs with a specific pose combination. The "best-fit" classifier having the same pose combination with the input matching pair makes the final decision. Through pose-adaptive matching, we explicitly handle the large pose variation by this "divide-and-conquer" method.

Component	Image size	Patch division
Forehead	76×24	7×2
Left eyebrow	46×34	4×3
Right eyebrow	46×34	4×3
Left eye	36×24	3×2
Right eye	36×24	3×2
nose	24×76	2×7
Left cheek	34×46	3×4
Right cheek	34×46	3×4
Mouth	76×24	7×2

Table 3. Patch division for face components.

4.3. Evaluations of pose-adaptive matching

To best evaluate the ability of pose change handling, we constructed a new test set from the LFW dataset by randomly sampling 3,000 intra-personal/extra-personal pairs for each pose combination. The total pair number in our new test set is $3,000 \times 6 = 18,000$. Note that this new test set is more challenging than the standard test data in the LFW due to the larger pose difference between the matching pair. We use half of them as the training set and the rest as the test set. Subjects are mutually exclusive in these two sets. And the patch division in component-level setting is shown in Table 3. Recognition performances were compared before ($76.20\% \pm 0.41\%$) and after ($78.30\% \pm 0.42\%$) pose-adaptive matching was adopted and results showed that the proposed technique is useful in such large pose-variant case.

5. Experimental results

In this section, we report our final face recognition performance on the LFW benchmark systematically and then validate the excellent generalization ability of our system across different datasets.

5.1. Results on the LFW benchmark

We present our recognition results on the LFW benchmark in the form of ROC curves. Figure 9 shows comparison results for the validation of our proposed individual techniques. In Figure 9, "single LE + holistic" means that we only use the single best LE to represent the holistic face, and it is the baseline to show the power of LE without other techniques. "single LE + comp" indicates the application of component-level, pose-adaptive matching to the baseline single LE. Multiple LE descriptors are combined to form "multiple LE + holistic". And "multiple LE + comp" is our best performer. The accuracies for these four methods are $81.22\% \pm 0.53\%$, $82.72\% \pm 0.43\%$, $83.43\% \pm 0.55\%$, and $84.45\% \pm 0.46\%$. Despite the strong discriminant ability of the LE descriptor itself, the pose-adaptive matching and

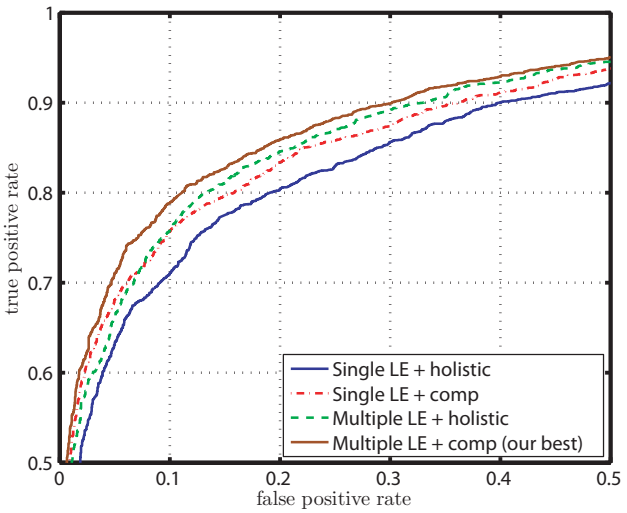

Figure 9. Demonstrate the effects of our proposed techniques on the LFW benchmark. Here, "holistic" means using holistic face representation while "comp" means component-level, pose-adaptive matching.

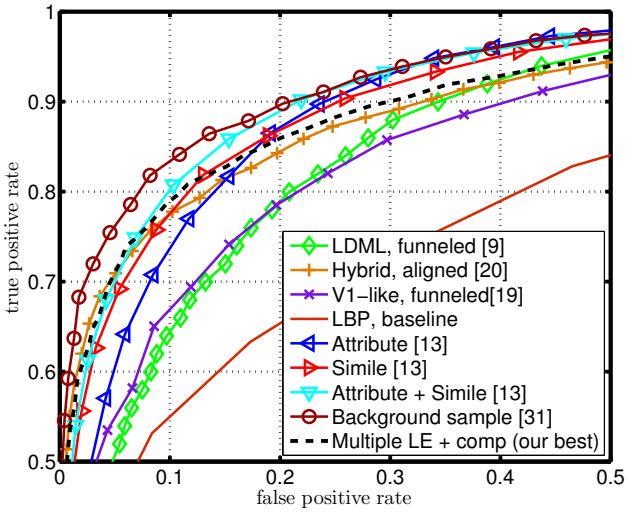

Figure 10. Face recognition comparison on the LFW benchmark in restrict protocol.

multiple descriptor combination further enhance the recognition performance of our system.

Our best ROC curve is comparable with previous state-of-the-art methods, as shown in Figure 10. On the LFW benchmark, two new algorithms show the leading performance. Wolf *et al.*'s work [31] adopts the background learning by using the identity information within the training set. Kumar *et al.* [13] used the supervised learning to train high-level classifications through a huge volume of training images outside of the LFW dataset. These two methods [13, 31] both use additional information outside the LFW test protocol. So the comparison with other methods (including ours) in Figure 10 is not really fair. Additional training data or information may also improve other approaches.

Descriptor	Recog. rate on Multi-PIE
59-code LBP	$84.30\% \pm 0.89\%$
8-bin HOG	$84.02\% \pm 0.66\%$
Gabor	$86.42\% \pm 0.85\%$
single LE + holistic	$\mathbf{91.58\% \pm 0.50\%}$
single LE + comp	$\mathbf{92.12\% \pm 0.52\%}$
multiple LE + holistic	$\mathbf{92.20\% \pm 0.49\%}$
multiple LE + comp	$\mathbf{95.19\% \pm 0.46\%}$

Table 4. Recognition performance on the Multi-PIE dataset.

Our system achieves the best performance when the standard test protocol is strictly respected [12]. More importantly, our work focuses on low-level face representation, which can be easily combined with previous algorithms to produce better performance.

5.2. Results on Multi-PIE

We also perform extensive experiments on the Multi-PIE dataset to verify the generalization ability of our approach. The Multi-PIE dataset contains face images from 337 subjects, imaged under 15 view points and 19 illumination conditions in 4 recording sessions. Large differences exist between LFW and Multi-PIE, considering the pose compositions, illumination variance, and resolution. Moreover, Multi-PIE is collected under a controlled setting systematically simulating the effects of pose, illumination, and expression. On the other hand, the LFW is more close to the real-life setting since its faces are selected from news images. For these reasons, training on one dataset and testing on the other can better demonstrate the generalization ability of a recognition system.

Similar to the LFW benchmark, we randomly generate 10 subsets of face images with Multi-PIE, each has 300 intra-personal and 300 extra-personal image pairs. The identities of subjects are mutually exclusive among these 10 subsets, and cross-validation mode similar to LFW is applied. The default "single LE" descriptor and "multiple LE" descriptors trained on the LFW benchmark are adopted in the experiments.

As shown in Table 4, the single LE with holistic face representation outperforms the commonly used descriptors more than 5 points, and pose-specific classifiers trained on the LFW dataset also perform well on the Multi-PIE dataset. All these results demonstrated the excellent generalization ability of our system.

6. Conclusion and discussion

We have introduced a new approach for face recognition using learning-based (LE) descriptor and pose-adaptive

matching. We validated our recognition system on the LFW benchmark and demonstrated its excellent generalization ability on Multi-PIE.

In this work, the face micro-pattern encoding is learned but the pattern sampling is still manually designed. Automating this step with learning techniques [27] may produce a more powerful descriptor for face recognition.

Acknowledegments: This work is performed when Cao and Yin visited Microsoft Research Asia. It was supported in part by the National Natural Science Foundation of China Grant No.60553001, and the National Basic Research Program of China Grant Nos.2007CB807900, 2007CB807901.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face Recognition with Local Binary Patterns. *Lecture Notes in Computer Science*, pages 469–481, 2004.

[2] T. Ahonen and M. Pietikäinen. Image description using joint distribution of filter bank responses. *Pattern Recognition Letters*, 30(4):368–376, 2009.

[3] V. Blanz and T. Vetter. Face recognition based on fitting a 3 D morphable model. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1063–1074, 2003.

[4] C. Chang and C. Lin. LIBSVM: a library for support vector machines, 2001. *Software available at http://www. csie. ntu. edu. tw/cjlin/libsvm*, 2001.

[5] J. Cui, F. Wen, R. Xiao, Y. Tian, and X. Tang. EasyAlbum: an interactive photo annotation system based on face clustering and re-ranking. In *Proc. of the SIGCHI conference on human factors in computing systems*, page 376, 2007.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005.

[7] Y. Freund, S. Dasgupta, M. Kabra, and N. Verma. Learning the structure of manifolds using random projections. In *NIPS*, 2007.

[8] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. In *International Conference on Automatic Face and Gesture Recognition*, 2008.

[9] M. Guillaumin, J. Verbeek, C. Schmid, I. LEAR, and L. Kuntzmann. Is that you? Metric learning approaches for face identification. In *Proc. ICCV*, 2009.

[10] G. Hua and A. Akbarzadeh. A robust elastic and partial matching metric for face recognition. In *Proc. ICCV*, 2009.

[11] P. Hua, G. Viola and S. Drucker. Face recognition using discriminatively trained orthogonal rank one tensor projections. In *Proc. CVPR*, 2007.

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

[13] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and Simile classifiers for face verification. In *Proc. ICCV*, 2009.

[14] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.

[15] L. Liang, R. Xiao, F. Wen, and J. Sun. Face Alignment via Component-based Discriminative Search. In *Proc. ECCV*, 2008.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[17] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[18] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The FERET evaluation mthodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1090–1104, 2000.

[19] N. Pinto, J. DiCarlo, and D. Cox. How far can you get with a modern face recognition test set using only simple features. In *Proc. CVPR*, 2009.

[20] Y. Taigman, L. Wolf, T. Hassner, and I. Tel-Aviv. Multiple One-Shots for utilizing class label information. In *BMVC*, 2009.

[21] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *Lecture Notes in Computer Science*, 4778:168, 2007.

[22] E. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. In *Proc. CVPR*, 2008.

[23] M. Turk and A. Pentland. Face Recognition using Eigenfaces. In *Proc. CVPR*, 1991.

[24] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? In *Proc. CVPR*, 2003.

[25] X. Wang and X. Tang. A unified framework for subspace face recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 26(9):1222–1228, 2004.

[26] X. Wang and X. Tang. Random sampling for subspace face recognition. *International Journal of Computer Vision*, 70(1):91–104, 2006.

[27] S. Winder and M. Brown. Learning local image descriptors. In *Proc. CVPR*, 2007.

[28] S. Winder, G. Hua, and M. Brown. Picking the best DAISY. In *Proc. CVPR*, 2009.

[29] L. Wiskott, J. Fellous, N. Krüger, and C. Von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):775–779, 1997.

[30] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Faces in Real-Life Images Workshop in ECCV*, 2008.

[31] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *Proc. ACCV*, 2009.

[32] J. Wright and G. Hua. Implicit elastic matching with random projections for pose-variant face recognition. In *Proc. CVPR*, 2009.

[33] L. Zhang, R. Chu, S. Xiang, S. Liao, and S. Li. Face detection based on multi-block lbp representation. *Lecture Notes in Computer Science*, 4642:11, 2007.