Robust Face Recognition via Accurate Face Alignment and Sparse Representation

Hanxi Li[1,2] Peng Wang[1,2,3] Chunhua Shen[1,2]
[1]NICTA Canberra Research Laboratory, Australia; [2]Australian National University; [3]Beihang University, China

***Abstract*—Due to its potential applications, face recognition has been receiving more and more research attention recently. In this paper, we present a robust real-time facial recognition system. The system comprises three functional components, which are face detection, eye alignment and face recognition, respectively. Within the context of computer vision, there are lots of candidate algorithms to accomplish the above tasks. Having compared the performance of a few state-of-the-art candidates, robust and efficient algorithms are implemented. As for face detection, we have proposed a new approach termed Boosted Greedy Sparse Linear Discriminant Analysis (BGSLDA) that produces better performances than most reported face detectors. Since face misalignment significantly deteriorates the recognition accuracy, we advocate a new cascade framework including two different methods for eye detection and face alignment. We have adopted a recent algorithm termed Sparse Representation-based Classification (SRC) for the face recognition component. Experiments demonstrate that the whole system is highly qualified for efficiency as well as accuracy.**

***Keywords*-Sparse representation; face recognition; greedy sparse linear discriminant analysis.**

I. Introduction

Face recognition might be one of the most well-known and attracting applications in computer vision. Because of the outstanding face recognition capability of human being and the numerous potential applications of face recognition technology, face recognition technology has been receiving more and more attentions. Over the past decades, researchers have focused on developing fully automatic facial recognition systems, which means, not only the recognition process but also face detection and feature extraction are all supposed to be handled by the computer automatically. As a result, a practical facial recognition system should at least include the three aforementioned parts.

Face detection is the first key step prior to recognition. Most work on segmentation was focused on single-face segmentation from a simple or complex background, while significant progresses have been made recently. The new methods [1]–[3], which train the face detector on large numbers of samples have achieved superior results compared with the traditional feature-based and template-matching approaches. The method devised by Viola and Jones, which using the Haar-like features and AdaBoost to train the classifier, is so successful that makes it a milestone in computer vision. In the system this paper presented, having compared with Viola Jones' approach, we use a similar but better face localization algorithm to detect faces in images.

Feature extraction plays an important role in a face recognition system, especially in a real-time system because feature-extracted classification is usually computationally cheaper and more robust than methods that use all the potential features. There are two categories of features in the face recognition domain. The first one is local facial features and their variants. These features are typically used in template-based or structural-based methods [4], [5]. Within the context of holistic methods, where all the pixels of a face image rather than a part of it can be seen as candidate features, abundant work has been devoted to investigate various methods projecting the high dimensional facial image into lower dimensional feature spaces. Eigenfaces [6], Fisherfaces [7] and Laplacianfaces [8] are the most popular examples of them. However, in many circumstance, even holistic method needs the local facial features to make sure their robustness. In the our system, eyes' locations are first obtained for aligning face and then the Eigenface for each test sample is extracted for recognition.

Besides the face detection and feature extraction, face recognition is essentially a classification process. Machine learning methods such as artificial neural network (ANN), support vector machine (SVM) and boosting are usually applied for training the recognition classifiers [9], [10]. Because of the special characteristics of face recognition, *e.g.*, multiple classes and similar distribution for each class of training data, instance-based learning methods such as k-Nearest Neighbor (k-NN) [11], nearest subspace (NS) [12] and nearest feature line (NFL) [13] are also very popular in this field. Wright and Yang *et al*. proposed a new algorithm for facial recognition recently [14]. Their method is somewhat similar to NS and NFL approaches but more robust to the noise and occlusion. This novel algorithm is implemented in our system and our experiments show that it is indeed a very promising algorithm for facial recognition.

The organization of the remaining content is described as follows. Section 2 is the introduction to the face detection algorithm that we have tested and the one we adopt finally. In Section 3, we show the important step for facial recognition: eye alignment. The sparse representation algorithm for face recognition is discussed in Section 4. Experiments are presented in Section 5, where all the details of our system, *e.g.*, data sets, experiment design and the results are discussed.

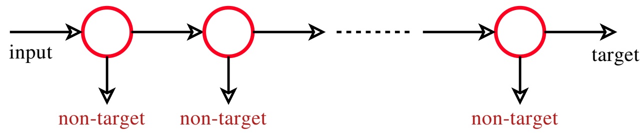


Figure 1: The Cascade Structure of Viola and Jones' Framework

II. Face Detection

As it is introduced in Section 1, face detection is the first essential step to face recognition system. The method proposed by Paul Viola and Michael Jones might be one of the most frequently used face detection approaches around the world. In their framework for face detection (illustrated in Fig.1), Haar-like features are selected at the same time when the weak classifiers are trained by AdaBoost. The cascade framework allows most non-face patches to be rejected quickly far before reaching the final node, resulting in fast performance. Only the candidate who go through all the nodes will be reported as a face. The cascade scheme accelerates face detection process while maintains an acceptable detection rate.

However, when the standard AdaBoost is implemented, one assumes that negative training data and positive training data are close in the terms of number. This assumption, however, doesn't hold in the context of face detection since negative samples, *i.e.*, the non-facial training data are usually much more than the face samples which are commonly considered as positive data. Paisitkriangkrai *et al.* proposed a novel face detection algorithm termed Boosted Greedy Sparse Linear Discriminant Analysis (BGSLDA) [15]. This algorithm inherits the cascade framework and reweighing scheme of AdaBoost based detectors but use LDA to choose features. It is well known that LDA takes the numbers of each class into consideration. This way, BGSLDA can yield higher accuracy. In our system, we have used BGSLDA as our face detection method.

The scheme of BGSLDA can be illustrated as follows. In each round of training a cascade layer (strong classifier), positive training set D^+ is kept the same, while negative training set D^- is constructed by collecting those negative samples misclassified by current cascaded classifier. Therefore, each strong classifier are trained for rejecting different types of negative samples. For BGSLDA, the training of one weak classifier can be split into two phases: Haar feature selection and decision stump training. BGSLDA utilize the class-separability criterion of Sparse LDA (which is shown in (1)) to select Haar features.

$$\underset{\omega}{\text{maximize}} \; \frac{\omega^T S_b \omega}{\omega^T S_w \omega}, \text{ subject to } \mathbf{Card}(\omega) = k. \tag{1}$$

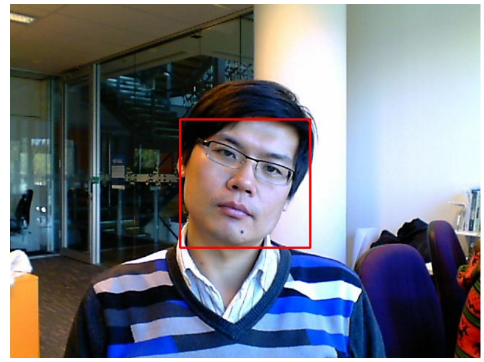


Figure 2: An example of face detection in a typical office environment.

Here, we define S_b as the between-class covariance matrix, while S_w is the within-class covariance matrix and k is the target number of non-zero elements of ω.

The decision stump for each Haar feature is learned on the training set reweighed according to the strategy of AdaBoost.

Having been detected by the cascade classifier, the face with some background is captured and processed by the eye alignment algorithm presented in next section. An example of face detection procedure can be seen in Fig. 2.

III. Face Alignment

A. Design of Face Alignment Algorithm

As discussed, face alignment plays a crucial role in a face recognition system since most face recognition approaches are very sensitive to the face pose and scale. Poorly-aligned face would deteriorate the recognition performance remarkably [7], [16]. The main idea for aligning face is to localize the facial features in the detected face image. Due to the importance, numerous approaches have been proposed in the literature [17]–[19]. These approaches can be grouped into two types: context-free methods and context-dependent methods. The former type of approaches are always faster but inferior in terms of the accuracy while the later one usually shows high robustness but suffering from expensive computation.

Considering the various characteristics of these two different approaches, we have used two eye detection algorithms, which belong to the two different types respectively, to form a two-layer eye localization cascade.

B. Two-layer Eye Localization Cascade

Fast Radial Symmetry Transform (FRST) [20] is a fast gradient-based and context-free operator which looks for points of high radial symmetry. The transformation is calculated at one or more radii $n \in N$, where N is the set of radii of the radially symmetric features to be detected. At each radius n, an orientation projection image O_n and a magnitude projection image M_n are formed according to the gradient g at each point p from the original image. Then the radial symmetry contribution at radius n is defined as the convolution:

$$S_n = F_n * A_n, \tag{2}$$

where

$$F_n(p) = \frac{M_n(p)}{k_n}\left(\frac{|\tilde{O}_n(p)|}{k_n}\right)^{\alpha} \tag{3}$$

and

$$\tilde{O}_n(p) = \begin{cases} O_n(p) & \text{if } O_n(p) < k_n \\ k_n & \text{otherwise.} \end{cases} \tag{4}$$

The full transformation is defined as the average of the symmetry contributions over all the radii considered,

$$S = \frac{1}{|N|}\sum_{n \in N} S_n. \tag{5}$$

The low value of a point in transformed image indicates it could be the location of a pupil. After transformation, We crop some areas centered around the points with the lowest value. These areas are called "region of interest" (ROI) which are considered as the potential eye locations and will be the searching area of next layer. The FRST is quite suitable for being a preprocessor thanks to its highly computational efficiency.

In the second layer, the exact coordinates for two pupils are detected by the Viola and Jones approach. There are many available trained classifier models with Viola and Jones' framework to detect facial elements such as eyes, nose and mouth. Castrillon *et al*. implemented most of them and compared the results in their paper [21]. According the performance and efficiency, we adopt the classification model [22]. Searching in the limited "region of interest" yielded by the first layer, the detection in the second layer is much faster and more accurate than using the same model to detect eyes over whole image.

C. Alignment Transform

Given the position of two pupils, eye alignment is an affine transformation of the original detected face image. Suppose that we have acquired two pupils with coordinates (x_1, y_1), and (x_2, y_2), the expected distance between two eyes on aligned face is d, the rotation center's position is calculated as $(center_x, center_y) = (x_1 - x_2, y_1 - y_2)$ and the rotation angle $\theta = \arctan((y_2 - y_1)/(x_2 - x_1))$. Then the mapping matrices A_m and $B_m$for rotation can be created as follows.

$$A_m = \begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix} \tag{6}$$

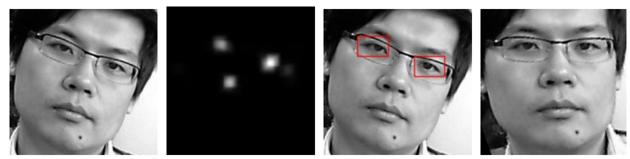
$$B_m = \begin{bmatrix} (1-\alpha)\cdot center_x - \beta \cdot center_y \\ (1-\alpha)\cdot center_y - \beta \cdot center_x \end{bmatrix}, \tag{7}$$

where $\alpha = d \cdot \cos\theta$ and $\beta = d \cdot \sin\theta$.

The aligned location (x', y') for the original point in (x, y) can be obtained by:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A_m \cdot \begin{bmatrix} x \\ y \end{bmatrix} + B_m. \tag{8}$$

All the points in original image are processed in above manner so the face could be aligned as our wish. A demonstration for eye alignment is shown in Fig. 3.



(a) (b) (c) (d)

Figure 3: A demonstration for eye alignment, (a) is original gray image; (b) illustrates the image after FRST, the bright parts are the potential locations of pupils; (c) shows the found eye; image (d) is the aligned face acquired by eyes' location information.

As is mentioned before, face alignment can affect the performance of face recognition remarkably. The affection is illustrated in Section V.

IV. Face Recognition

Different from face detection or eye detection, face recognition is always a multi-class classification problem. Furthermore, the data points belonging to each class usually cluster together. As a result, instance-based classifiers, who can handle this type of tasks well, are very popular in face recognition community.

Wright *et al*. proposed a novel classifier named Sparse Representation-based Classification (SRC) [14]. This algorithm, strikes a balance between k-NN and NS since the former one classifies the test sample based on the best representation in terms of a single training sample, whereas NS classifiers based on the best linear representation in terms of all the training samples in each class. The novel SRC, focus on choosing the appropriate sparse representation across all the training data. The procedure of SRC is illustrated in Algorithm 1.

We have also implemented other popular face recognition approaches for comparing with SRC. In the terms of performance, SRC beats all the other candidates with over 80% correct rate for YaleB data set. The high accuracy is the essential reason why we choose it. The details about the comparison is shown in the next section.

V. Experiments

In this section, the dataset processing, design of our experiment as well as the parameters and results of the experiments are presented and discussed.

Algorithm 1: Sparse Representation-based Classification

Input:

- A matrix of training samples $A = [A_1, A_2, \cdots, A_k] \in \mathbb{R}^{m\times n}$ for k classes
- An optional error tolerance $\varepsilon > 0$;
- An test sample $y \in \mathbb{R}^m$

1) Normalize the columns of A to have unit l^2-norm.
2) Solve the l^1minimization problem:

$$\hat{x_1} = \underset{x}{\operatorname{argmin}} \|x\|_1 \text{subject to} \|Ax - y\| \leq \varepsilon. \quad (9)$$

3) Compute the residuals

$$r_i(y) = \|y - A\delta_i(\hat{x}_1)\|_2 \quad (10)$$

Output:

$$identity(y) = \underset{i}{\operatorname{argmin}}\, r_i(y)$$

A. Design of the Experiment

1) Running Process of the System: Given that we have obtained sufficient training data to implement the recognition, the scene of a random selected subject (persons whose face images are store in our datasets) captured with a USB video camera. Frames of the video stream is processed one by one.

The face detection algorithm is applied on the image in the first place. Occasionally, multiple face are detected. But only the largest faces is selected for the following steps. If no face is detected, the next frame of image will be fetched. A rectangular subwindow is cropped given the position of the largest face in the image. The eye detection algorithm is conducted on the subwindow. Generally, two eyes are detected, otherwise this subwindow will be abandoned. Then the face subwindow is aligned by the transformation according to the algorithm shown in Section 3. Finally, the aligned face will be distinguished by the SRC classification algorithm.

2) Data Sets: Three data sets have been adopted or composed for test. The first one is Yale-B which is a open database consists of 2414 frontal-face image of 38 individuals. The face images in this data set were captured under various laboratory-controlled lighting conditions [23]. NICTA-A is the second data set which we obtained from AVI videos of 14 subjects. The AVI streams were made with USB camera (with 30 million hardware pixels) under different circumstances. NICTA-A comprises two parts *i.e.*, training data and testing data. Training data which contains 4200 faces is captured from videos which were shot approximately at 10 am and indoors. The testing videos were made at around 3 pm in the same place and yield also 4200 facial images. In this way, the robustness of our system can be estimated. Both training and testing faces in NICTA-A are aligned. NICTA-B is the unaligned version of NICTA-A, in other words, the samples in NICTA-B are the original face images corresponding to the aligned ones in NICTA-A. This data set is formed for assessing the influence of eye alignment process. Some examples of NICTA-A and NICTA-B are illustrated in Fig.4. Meanwhile, Fig.4 could also be regarded as a survey of our eye alignment's performance.

(a) NICTA-A

(b) NICTA-B

Figure 4: Examples from NICTA-A and NICTA-B. Fig.(a) shows 64 examples from NICTA-A while (b) contains the corresponding unaligned faces in NICTA-B.

All the face images in above three data sets are frontal. The faces in Yale-B are normalized images at resolution of 32×32 which means the dimension of primary face space is 1024. Pictures in NICTA-A and NICTA-B are all with resolution of 52×52 so the dimension is 2704.

3) Test Details : As to the recognition test, the aligned face image is firstly processed by a histogram equation operation which can diminish the affection of illumination variation to a great extent. We then implement the SRC on the extracted-features yielded by using some popular dimension reduction approaches, *i.e.*, PCA, LDA, Uncorrelated LDA (ULDA) [24], Orthogonal LDA (OLDA) [25], and Semidefinite Programming LDA (SDP) [26]. These methods can reduce feature dimension significantly in order to guarantee the efficiency of our system. The reduced dimensions of Eigenface are 25, 50 and 75 for Yale-B and 50, 75, 100 for NICTA-A respectively. They are all much smaller than the original 1024 or 2704 while still reserves no less than 80% energy of primary space. Nonetheless, due to the size of classes, the dimension for LDA, ULDA and OLDA are set to 20 for Yale-B and 13 for NICTA-A and NICTA-B. SDP project the original face space into feature space with dimension 30. The experiment is divided into two parts which are recognition algorithm comparison and eye alignment evaluation respectively.

Besides SRC, some conventional classifiers for face recognition, namely, k-NN and SVM are run based on various reduced spaces in order to compare with SRC in terms of performance. As to the parameters selection, the error tolerance ε for SRC is fixed to 0.05 which is the value chosen in [14]. The parameter k which defines number of the nearest neighbors is set 3 in k-NN. In our experiment, SVM is implemented with Gaussian kernel whose parameter γ is chosen by cross validation among the parameter set $[2^{-15}, 2^{-12}, 2^{-9}, 2^{-6}, 2^{-3}, 2^{0}, 2^{3}]$ while the parameter C for realization is selected from set $[2^{-15}, 2^{-12}, 2^{-9}, 2^{-6}, 2^{-3}, 2^{0}, 2^{3}]$. Every algorithm has been run for five rounds.

In order to evaluate the effect of eye alignment, we run the face recognition algorithms again (also for 5 rounds) on aligned faces (NICTA-A) as well as unaligned faces (NICTA-B). The performances of recognition algorithms on NICTA-A and NICTA-B is shown in Table II.

Our test is conducted on PC (Intel Quad Core 2.6G, 4G RAM) with Matlab 2008b.

B. Experimental Results

The result for recognition comparison based on Yale-B and NICTA-A is shown in Table I. We can find that SRC outperforms nearly all (except on OLDA with dimension 75) the other competitors in the test for Yale-B. The result shows the high robustness of SRC. However, in the self-made data test, SRC with the parameter $\varepsilon = 0.05$ ranks first for 7 times which is a bit less than k-NN's score but still far surpasses SVM. Though it remains the best for individual item (0.017 for LDA with dimension 100), SRC can only be considered as comparable with k-NN. This might partly because that we copy the value of ε set in John Wright's paper [14] which only tests the algorithm on Yale-B and AR data sets. If we want better outcomes, a cross validation procedure should be appended into experiment. Other approaches might improve SRC in the reduced space will be discussed in Section VI.

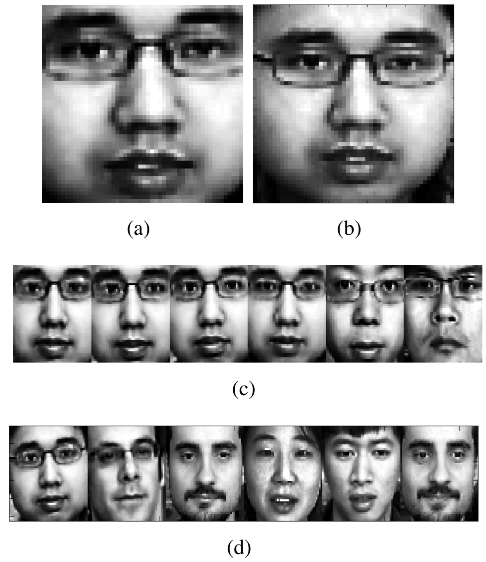


Figure 5: Comparison between aligned and unaligned facial recognition using SRC. (a) is a test image randomly selected from NICTA-A and (b) is its unaligned version in NICTA-B. (c) and (d) show the 6 most weighted training faces from NICTA-A and NICTA-B respectively. They are used by SRC to represent the test face linearly on two data sets. It can be seen that SRC select 4 same-label faces out of total 6 to represent the test image (a) linearly. Nonetheless, without the face alignment, only one correct face is chosen by SRC.

With regard to the evaluation of eye alignment, from Table II we can find that the error rates increase for around 2 to 10 times which means the performances all deteriorate remarkably due to the absence of eye alignment. It proves that a good eye alignment algorithm is of great importance in the whole facial recognition system.

Another example for illustrating the importance of eye alignment is shown in Fig.5. Fig. 5.a is a test image randomly selected from NICTA-A and Fig. 5.b is its unaligned version in NICTA-B. Fig. 5.c (corresponding to NICTA-A) and Fig. 5.d (corresponding to NICTA-A) both show 6 most weighted training faces which are used by SRC to represent the test face linearly on two data sets. It can be seen that SRC select 4 same-group faces out of total 6 to represent test image (a) linearly. On the contrary, without the face alignment, there is only one right image being chosen to form the representation. Predictably, SRC yields right result on NICTA-A but fails on NICTA-B.

VI. Conclusion and Future Work

This paper introduces a novel real-time facial recognition system which assembles some of latest computer vision technologies such as: BGSLDA-based face detection; face alignment using cascade framework combining FRST as well as Viola and Jones' method; latest face recognition algorithm named SRC. We integrated them into a practical system with acceptable performance.

The BGSLDA-based face detection algorithm can detect real-time face image from video stream with high robustness. It has been proved by the experiment that the face detection algorithm is totally qualified for our system.

The layered face alignment approach offers a great precondition for later recognition task with minor computation. Our experiment also shows that a good eye alignment preprocessor will ameliorate recognition performance remarkably (from 2 to 10 times).

A novel instance-based classification approach which termed SRC is adopted as our recognition method. It beat k-NN and SVM on the Yale-B data set and yield comparable robustness with respect to k-NN. Plus, SRC is a great recognition algorithm with marvelous improving space. In the future, a standard n-fold cross validation will be applied to select parameter ε for SRC. Furthermore, since some of the dimension reduction methods fix the target dimension, *e.g.*, LDA-based approaches are always with dimension of 13 for NICTA-A, the dimension is sometimes too low to run the SRC. In the future, the number of categories will be increased by adding more individuals to the NICTA-A data set. In this way, the dimension for LDA-based methods will grow to a reasonable size with respect to the original image size. In addition, the dimension reduction method which is not curbed by the number of categories, *i.e.*, SDP in this case, will be paid more attention.

The future work also includes harnessing fast linear-programming algorithm to accelerate the recognition procedure. It would allow us to run SRC on higher-dimensional feature space, which usually indicated the better performance. Apparently, the completion of those future task can improve the robustness of our system significantly.

ACKNOWLEDGMENTS

NICTA is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council.

This research was also supported in part by the Australian Research Council through its Special Research Initiative in Bionic Vision Science and Technology grant to Bionic Vision Australia.

REFERENCES

[1] L. Gu, S. Z. Li, and H. J. Zhang, "Learning probabilistic distribution model for multiview face detection," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2001, pp. 116–112.

[2] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2001, pp. 511–518.

[3] B. Heisele, T. Serre, M. Pontil, and T. Poggio, "Component-based face detection," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2001, pp. 657–662.

[4] P. W. Hallinan, "Recognizing human eyes," in *Geometric Methods In Computer Vision*. 1991, pp. 214–226, SPIE.

[5] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Automatic face identification system using flexible appearance models," *Image Vis. Comp.*, vol. 13, pp. 393–401, 1995.

[6] M. Kirby and L. Sirovich, "Application of karhunen-loeve procedure for the characteriszation of human faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, pp. 103–108, 1990.

[7] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class secific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, pp. 711–720, 1997.

[8] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang, "Face recognition using laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, pp. 328–340, 2005.

[9] J. Bigun, B. Duc, F. Smeraldi, S. Fischeer, and A. Markarov, *Multi-miodal person authentication*, pp. 26–50, Springer-Verlag, 1998.

[10] F. Samaria and S. Young, "Hmm based architecture for face identification," *Image Vis. Comp.*, vol. 12, pp. 537–583, 1994.

[11] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, Wiley & Sons, 2001.

[12] J. Ho, M. Yang, J. Lim, K. Lee, and D. Kriegman, "Clustering apperances of objects under varing illumination conditions," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2003, pp. 11–18.

[13] S. Z. Li and J. Lu, "Face recognition using the nearest feature line method," in *Proc. Int. Conf. Artificial Neural Networks*, 1999, pp. 439–443.

[14] J. Wright, A. Y. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 537–583, 2008.

[15] S. Paisitkriangkrai, C. Shen, and J. Zhang, "Efficiently training a better visual detector with sparse eigenvectors," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, Miami, USA, 2009.

[16] M. Turk and A. Pentland, "Eigenfaces for recognition," *Cogitive Neuroscience*, vol. 3, pp. 71–86, 1991.

[17] P. Wang, M. B. Green, Q. Ji, and J. Wayman, "Automatic eye detection and its validation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2005, pp. 164–164.

[18] P. Wang, Lam Cam Tran, and Qiang Ji, "Improving face recognition by online image alignment," in *Proc. Int. Conf. Patt. Recogn.*, 2006, pp. 311–314.

[19] W. Zhang, H. Chen, P. Yao, B. Li, and Z. Zhuang, "Precise eye location with adaboost and fast radial symmetry," in *Proc. Int. Conf. Computational Intelligence and Security*, 2006, pp. 725–730.

[20] G. Loy and A. Zelinsky, "Fast radial symmetry for speed sign detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, pp. 959–973, 2003.

[21] M. Castrillo-Santana and O. D. Suarez, "Real-time detection of multiple face ar different resolutiion in video streams," *Journal of Visual Communication and Image Representation*, pp. 130–140, 2007.

[22] M. Castrillo-Santana and O. D. Suarez, "Precise eye location with adaboost and fast radial symmetry," in *Proc. Int. Conf. Comp. Vis. Theory and Application*. 2008, pp. 145–150, Springer.

[23] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 643–660, 2001.

[24] Z. Jin, Y. Yang, Z. M. Tang, and Z. S. Hu, "A therorem on the uncorrelated optimal discriminant vectors," *Pattern Recogn. Lett.*, vol. 34, pp. 2041–2047, 2001.

[25] J. Ye, "Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems," *J. Mach. Learn. Res.*, vol. 6, pp. 483–502, 2005.

[26] C. Shen, H.D. Li, and M. J. Brooks, "Supervised dimensionality reduction via sequential semidefinite programming," *Pattern Recogn.*, vol. 41, pp. 3644–3652, 2008.

Table I: Comparison between SRC kNN and SVM with various dimension reduction methods. Numbers in the table is the error rates and their stand deviation yield by running the corresponding algorithm for 5 times. Base is the result for recognition algorithms run on the original feature space. PCA, LDA, OLDA, ULDA and SDP are the different dimension reduction approaches.

Data	Algorithm	Base	PCA	LDA	OLDA	ULDA	SDP
YaleB-25	kNN	0.314 ± 0.024	0.605 ± 0.0164	0.167 ± 0.028	0.605 ± 0.016	0.217 ± 0.040	0.605 ± 0.016
	SRC	NA	$\mathbf{0.192 \pm 0.030}$	$\mathbf{0.144 \pm 0.015}$	$\mathbf{0.192 \pm 0.031}$	$\mathbf{0.122 \pm 0.018}$	$\mathbf{0.194 \pm 0.033}$
	SVM	NA	0.384 ± 0.097	0.412 ± 0.064	0.483 ± 0.114	0.492 ± 0.071	0.291 ± 0.043
YaleB-50	kNN	0.327 ± 0.013	0.448 ± 0.029	0.119 ± 0.007	0.142 ± 0.008	0.132 ± 0.019	0.102 ± 0.012
	SRC	NA	$\mathbf{0.111 \pm 0.015}$	$\mathbf{0.117 \pm 0.005}$	$\mathbf{0.094 \pm 0.007}$	$\mathbf{0.093 \pm 0.018}$	$\mathbf{0.059 \pm 0.014}$
	SVM	NA	0.183 ± 0.046	0.216 ± 0.055	0.176 ± 0.018	0.192 ± 0.033	0.192 ± 0.057
YaleB-75	kNN	0.307 ± 0.025	0.402 ± 0.013	$\mathbf{0.104 \pm 0.022}$	0.104 ± 0.014	0.119 ± 0.005	0.082 ± 0.013
	SRC	NA	$\mathbf{0.089 \pm 0.017}$	0.114 ± 0.017	$\mathbf{0.072 \pm 0.008}$	$\mathbf{0.081 \pm 0.009}$	$\mathbf{0.049 \pm 0.006}$
	SVM	NA	0.110 ± 0.011	0.155 ± 0.042	0.145 ± 0.022	0.134 ± 0.037	0.157 ± 0.038
NICTAA-50	kNN	0.105 ± 0.037	0.102 ± 0.027	0.048 ± 0.014	0.042 ± 0.009	0.044 ± 0.013	0.048 ± 0.002
	SRC	NA	$\mathbf{0.1 \pm 0.031}$	$\mathbf{0.036 \pm 0.013}$	$\mathbf{0.041 \pm 0.020}$	0.048 ± 0.005	$\mathbf{0.042 \pm 0.019}$
	SVM	NA	0.067 ± 0.031	0.052 ± 0.025	$\mathbf{0.041 \pm 0.028}$	$\mathbf{0.040 \pm 0.032}$	0.052 ± 0.020
NICTAA-75	kNN	0.077 ± 0.017	$\mathbf{0.080 \pm 0.016}$	$\mathbf{0.026 \pm 0.005}$	$\mathbf{0.029 \pm 0.007}$	$\mathbf{0.040 \pm 0.019}$	$\mathbf{0.030 \pm 0.005}$
	SRC	NA	0.095 ± 0.021	0.029 ± 0.017	0.035 ± 0.009	0.055 ± 0.019	0.039 ± 0.019
	SVM	NA	0.086 ± 0.023	0.039 ± 0.009	0.055 ± 0.026	0.041 ± 0.005	0.067 ± 0.028
NICTAA-100	kNN	0.092 ± 0.012	0.095 ± 0.008	0.033 ± 0.004	0.032 ± 0.006	$\mathbf{0.029 \pm 0.008}$	0.033 ± 0.010
	SRC	NA	0.080 ± 0.007	$\mathbf{0.017 \pm 0.007}$	$\mathbf{0.023 \pm 0.008}$	0.039 ± 0.003	$\mathbf{0.026 \pm 0.004}$
	SVM	NA	$\mathbf{0.070 \pm 0.037}$	0.048 ± 0.017	0.032 ± 0.003	0.035 ± 0.003	0.066 ± 0.008

Table II: The evaluation for eye alignment. NICTA-A is the data set in which faces are aligned which NICTA-B is unaligned version of NICTA-A.

Algorithm	Data	Base	PCA	LDA	OLDA	ULDA	SDP
k**NN**	NICTA-A-50	0.105 ± 0.037	0.102 ± 0.027	0.048 ± 0.014	0.042 ± 0.009	0.044 ± 0.013	0.048 ± 0.002
	NICTA-B-50	0.254 ± 0.044	0.235 ± 0.025	0.172 ± 0.045	0.192 ± 0.041	0.198 ± 0.046	0.146 ± 0.034
	NICTA-A-75	0.077 ± 0.017	0.080 ± 0.016	0.026 ± 0.005	0.029 ± 0.007	0.040 ± 0.019	0.030 ± 0.005
	NICTA-B-75	0.261 ± 0.050	0.271 ± 0.059	0.145 ± 0.030	0.173 ± 0.030	0.173 ± 0.012	0.135 ± 0.043
	NICTA-A-100	0.092 ± 0.012	0.095 ± 0.008	0.033 ± 0.004	0.032 ± 0.006	0.029 ± 0.008	0.033 ± 0.010
	NICTA-B-100	0.240 ± 0.021	0.244 ± 0.012	0.136 ± 0.039	0.133 ± 0.037	0.153 ± 0.018	0.138 ± 0.010
SRC	NICTA-A-50	NA	0.1 ± 0.031	0.036 ± 0.013	0.041 ± 0.020	0.048 ± 0.005	0.042 ± 0.019
	NICTA-B-50	NA	0.225 ± 0.037	0.220 ± 0.018	0.227 ± 0.014	0.238 ± 0.014	0.188 ± 0.010
	NICTA-A-75	NA	0.095 ± 0.021	0.029 ± 0.017	0.035 ± 0.009	0.055 ± 0.019	0.039 ± 0.019
	NICTA-B-75	NA	0.234 ± 0.022	0.166 ± 0.014	0.173 ± 0.021	0.207 ± 0.030	0.197 ± 0.023
	NICTA-A-100	NA	0.080 ± 0.007	0.017 ± 0.007	0.023 ± 0.008	0.039 ± 0.003	0.026 ± 0.004
	NICTA-B-100	NA	0.189 ± 0.019	0.177 ± 0.013	0.172 ± 0.025	0.192 ± 0.025	0.176 ± 0.019
SVM	NICTA-A-50	NA	0.067 ± 0.031	0.052 ± 0.025	0.041 ± 0.028	0.040 ± 0.032	0.052 ± 0.020
	NICTA-B-50	NA	0.275 ± 0.038	0.202 ± 0.054	0.235 ± 0.034	0.196 ± 0.075	0.286 ± 0.029
	NICTA-A-75	NA	0.086 ± 0.023	0.039 ± 0.009	0.055 ± 0.026	0.041 ± 0.005	0.067 ± 0.028
	NICTA-B-75	NA	0.296 ± 0.006	0.254 ± 0.078	0.223 ± 0.033	0.223 ± 0.098	0.295 ± 0.085
	NICTA-A-100	NA	0.070 ± 0.037	0.048 ± 0.017	0.032 ± 0.003	0.035 ± 0.003	0.066 ± 0.008
	NICTA-B-100	NA	0.264 ± 0.040	0.164 ± 0.012	0.2 ± 0.026	0.159 ± 0.023	0.214 ± 0.085