Expression-Invariant Face Recognition with Expression Classification

Xiaoxing Li, Greg Mori and Hao Zhang
School of Computing Science
Simon Fraser University, Burnaby, BC, V5A 1S6 Canada
E-mail: {xli1, mori, haoz}@cs.sfu.ca

Abstract

Face recognition is one of the most intensively studied topics in computer vision and pattern recognition. Facial expression, which changes face geometry, usually has an adverse effect on the performance of a face recognition system. On the other hand, face geometry is a useful cue for recognition. Taking these into account, we utilize the idea of separating geometry and texture information in a face image and model the two types of information by projecting them into separate PCA spaces which are specially designed to capture the distinctive features among different individuals. Subsequently, the texture and geometry attributes are re-combined to form a classifier which is capable of recognizing faces with different expressions. Finally, by studying face geometry, we are able to determine which type of facial expression has been carried out, thus build an expression classifier. Numerical validations of the proposed method are given.

1. Introduction

Face recognition is an intensively studied research problem. A comprehensive review of the related works can be found in [1] and [2]. It has been noticed that facial expression usually affects the performance of a face recognition system. To deal with it, some existing methods rely on extracting stable face features, e.g., extracted line segment [3] and geometric invariants [4]. [5] is based on dynamic link matching which is robust under face rotation and deformation. One of the drawbacks with these works is that there is less reliable invariants when faces carry heavy expressions.

Recently, some newly-developed face recognition systems based on video sequences can handle a wider range of facial expressions [6, 7]. Compared to a single face image, a video sequency consists of a large number of succesive images and carries much more information, making the recognition task easier. However, a conventional face recognition system based on single image is still valuable in many real applications, e.g., creating personal photobook from digital image collections, where video sequences consistently capturing a face are not generally available.

Besides face recognition, expression classification has also drawn much attention. The facial action coding system (FACS) [8] is designed to capture the variations of face features under facial expressions given a single image frame. [9] and [10] are examples of classifying facial expressions by analyzing multiple image frames in a video sequence. However, these works do not consider the problem of recognizing faces.

Our aim in this paper is to design a single-image-based face recognition system which is capable of, given an expressioned testing face image, not only recognizing which individual the testing face belongs to, but also determining which type of facial expression has been carried out. This is a challenging task, since the expression information needs to be separated from the intrinsic face features. An intuitive observation is that even under different expressions, the texture of one's face skin is relatively invariant, with only slight changes due to local lighting effects, or particular changes such as blushing. In addition, certain geometry, e.g., the length of nose and the distance between eyes, is also relatively invariant under different expressions. On the other hand, some geometry can have obvious changes under different expressions. For example, an opened mouth can prolong the chin. In this paper, our proposed face recognition system is based on constructing separate PCA spaces for face texture and those invariant geometry features. In addition, a novel expression PCA space is also constructed for the purpose of classifying expressions.

The rest of this paper is organized in the following way. Section 2 compares our method with existing works. Section 3 describes the proposed method. Section 4 provides experimental results. Concluding remarks are given in Section 5, followed by some future research issues.

2 Comparison with related methods

The adverse effect of facial expressions on the performance of face recognition systems can be mostly attributed

0-7695-2542-3/06 $20.00 © 2006 IEEE

to misalignment. In typical recognition systems, faces are aligned globally. As a result, under different facial expressions, face features are not simultaneously and accurately aligned, which dramatically affects the performance of an appearance-based recognition system [11, 12].

The idea of separately modeling texture and geometry information has been applied in Active Shape Model and Active Appearance Model (ASM/AAM) [13, 14]. In ASM, face geometry is defined via a set of feature points. In doing so, the geometry of a face can be described by a mean shape and a set of PCA basis representing geometric variations. In AAM, face texure can be warped to the mean shape to acquire *shape-normalized* faces. The textures of these normalized faces can be used to calculate a mean texture and a set of PCA basis modeling texture variations. Using these geometry and texture models, given a testing face, ASM/AAM can represent it, as well as detect and register its corresponding feature points.

In our proposed method, we have a face gallery which is a collection of unexpressioned faces (in the following we call them *reference faces*). Given an expressioned testing face, we first recognize it by finding its matching reference face in the face gallery. This process begins with fitting a generic mask to reference faces as well as the testing face, where the mask nodes are feature points. We then warp the testing face to each of the reference faces. If we have warped an expressioned face to the reference face of the same individual, we call this warping *natural warping*; otherwise we call it *artificial warping*. The warped texture is then used as the texture information of the testing face, while the fitted mask carries the geometric information. The face recognition task is then to distinguish natural warping from artificial warpings by jointly utilizing the two pieces of information.

To compare the texture of a testing face with that of each reference face, we project their textures onto the Eigenfaces [11] spanned by the training faces. The distance between their projections indicates their texture dissimilarity. As for the geometry attribute, we focus on stable face geometry features by eliminating the geometric change caused by expressions. Specifically, in training stage, we investigate the geometric change in natural warpings and find out those inner angles of the fitted mask which are relatively stable. Larger weights are assigned to these angles when spanning a angle PCA space. In this angle PCA space, the distance between two faces' projections indicates the dissimilarity between their intrinsic geometry. The distances in the two PCA spaces will be appropriately combined to identify a natural warping for each testing face, i.e., accomplish the task of face recognition. Next, the geometric change of the testing face during the identified natural warping can then be used to determine which type of expression has been carried out on the testing face. For example, faces with an opened mouth can be detected by a prolonged chin, which probably implies the expression "happy" or "surprise". In this case, we focus on the unstable angles during natural warpings and build an expression PCA space for expression classification.

By comparing our method with ASM/AAM, we clearly see that the design targets and the set of PCA spaces are quite different. In ASM/AAM, the target is to build representative models which can be used to represent, detect or register different faces. Therefore, all the faces are warped to an average face. On the other hand, our target is to capture the distinctive features among different individuals. Thus, testing faces are warped to each reference face. In the warping of AAM, a face will be aligned with the average face where the intrinsic face geometry has been changed and the expression is also removed. In our warping, however, we separately capture the intrinsic geometry and expression, which is specially designed for the purpose of face recognition and expression classification.

3 Method

3.1 Face mask

The idea of using a mask for registration in a face recognition system is not new. For example, [15] uses a 3D mask to register the frontal view with the profile view. Some existing masks, such as [16], are well designed for fitting deformable surface according to muscle actions. However, when working with planar face images, it is hard to achieve a good registration for all the densely placed vertexes. Besides, although quad-based masks [16] make morphing flexible, triangulated masks are advantageous in texture mapping. Consequently, we use a simplified and trianglated mask, as shown in Fig. 1. On this mask, the grey triangles correspond to regions of the eyebrows, eyes, nose and

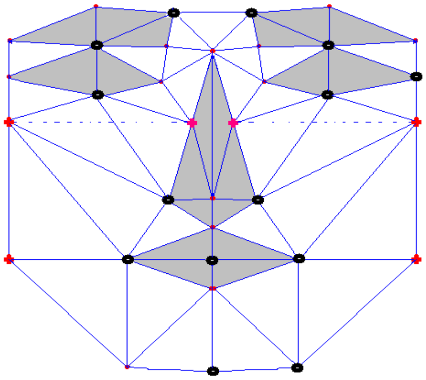

Figure 1. Face mask.

mouth. They are intentionally set to smaller sizes so as to capture more detailed features. This mask contains only 34 vertexes and 51 triangles, which will be denoted by v and t, respectively.

3.2 Input data

We arrange the input data from a given database of face images as follows:

$$F_1 : r_1, e_1^1, e_1^2, \ldots, e_1^n$$
$$F_2 : r_2, e_2^1, e_2^2, \ldots, e_2^n$$
$$\vdots$$
$$F_k : r_k, e_k^1, e_k^2, \ldots, e_k^n$$

where F_i denotes the set of faces for the ith individual, for $i = 1, 2, \ldots, k$. Within F_i, there is a frontal-looking face without expression, namely normal face, denoted by r_i. In addition, there are faces with n types of expressions, denoted by e_i^j where $j = 1, 2, \ldots, n$ indexes different types of expressions. Since we are particularly interested in recognizing expressioned faces, we use all the normal faces to form a face gallery and each of these normal faces is a reference face.

3.3 Mask fitting and warping

Given a testing face, we first fit the mask onto it, by manually selecting 14 markers to register important face features. These markers are shown as the dark dots in Fig. 1. Then, all the other vertexes on the mask can be fitted using the symmetry and common knowledge of face structures. Fig. 2 gives two examples of such a registration process.

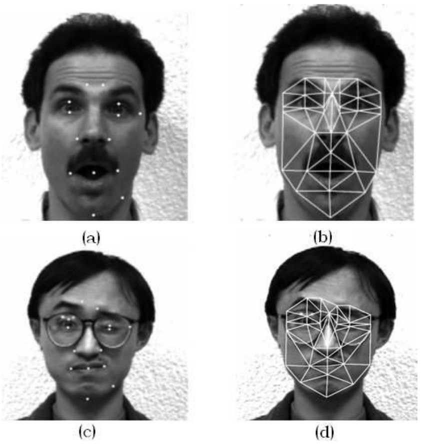

Figure 2. Fitting masks. (a), (c): manually placed markers (white dots); (b), (d): fitted masks.

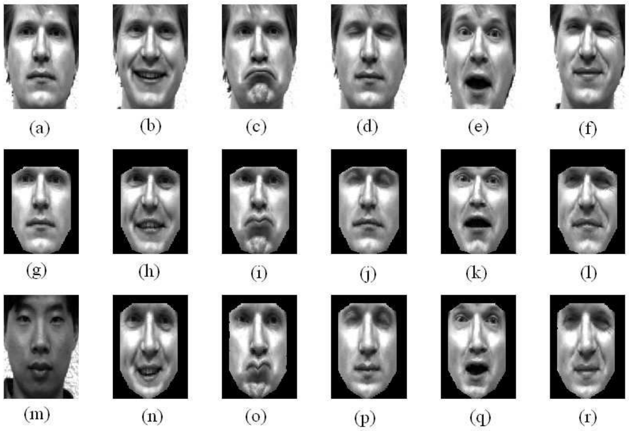

Figure 3. Examples of face warping. (a): reference face; (b), (c), (d), (e), (f): expressioned faces of the same individual; (g): cropped reference face; (h), (i), (j), (k), (l): natural warpings; (m): reference face of another individual; (n), (o), (p), (q), (r): artificial warpings to (m).

The region outside of the fitted mask is then set to zero intensity, since backgrounds, hair and shoulders are not stable features. After fitting masks, we have established a one-to-one correspondence among the vertexes on fitted masks for different faces. We then warp the texture in each triangle t' on an expressioned face onto the corresponding triangle t on a reference face by operating an affine triangle warping. Each vertex $v'(x', y')$ in t' will be transformed to the corresponding vertex $v(x, y)$ in t. This warping can be expressed as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \tag{1}$$

where a_1, a_2, a_4 and a_5 define scaling, rotation and shearing, etc., while a_3 and a_6 represent translation. Substituting the coordinates of all the vertexes in t and t', the above six parameters can be determined.

After warping, expressioned faces have been morphed to have the same geometry as the reference face. Such examples are shown in Fig. 3. As we can see, the geometries of expressioned faces, shown in Fig. 3 (b) to (f), are similar to that of the reference face of the same individual in Fig. 3 (a) in most facial parts, with only certain geometry changes, such as prolonged chin or closed eyes. However, when compared to the reference face of a different individual in Fig. 3 (m), the geometries differs greatly almost for the entire face. In the warping process, the warped face preserves texture features, while its geometry has been changed. In natural warpings, i.e., warping expressioned faces to the reference face of the same individual, the warped texture will be close to that of the reference face, as shown in Fig. 3 (h) to (l). On the other hand, if expressioned faces are warped to the reference face of a different individual, leading to artificial warpings, the warped texture will be quite different from that of the reference face, as shown in Fig. 3 (n) to (r).

3.4 Face geometry and angle residual

An example illustrating the geometry change of an expressioned face during a warping process is given by Fig. 4.

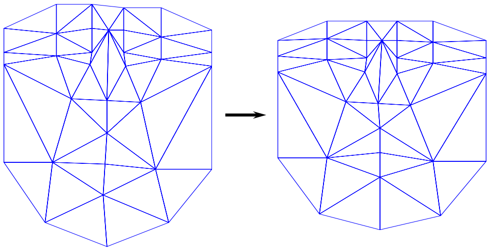

Figure 4. Geometry change, when warping a surprised face to a reference face.

After warping, the resulting face will have the same geometry as the reference face. Consequently, the warped face may appear similar to the reference face if their skin textures are close to each other. However, the testing face may belong to another individual whose face geometry differs greatly from that of the reference face. This implies that face geometry carries valuable information. To characterize this, we select the *inner angles* of each triangle on the fitter mask as a descriptor of its geometry. In addition, this descriptor is invariant under uniform scaling, translation and rotation, which can be caused by inaccurate calibration. These angles are arranged into an *angle vector* x_{ang}, denoted by

$$x_{ang} = [\theta_1 \; \theta_2 \; \theta_3 \; \ldots \; \theta_{151} \; \theta_{152} \; \theta_{153} \;]^T$$

where θ_1 to θ_3 belong to the first triangle, θ_4 to θ_6 belong to the second triangle, etc. There are altogether 51 triangles, yielding a total of 153 angles for each mask.

We record the geometric change during the warping process. As the mask of an expressioned face is warped, its angle vector changes. We calculate a vector x_{res} to record this angle change, referred to it as *angle residual* hereafter, given by

$$x_{res} = x^e_{ang} - x^r_{ang}. \tag{2}$$

where x^e_{ang} and x^r_{ang} are the angle vectors of the expressioned face and the reference face, respectively.

3.5 Face recogntion and expression classification

Our face recognition system is developed by jointly utilizing texture and geometry information. A diagram illustrating the various components and the flow of our recognition system is shown in Fig. 5.

3.5.1 Training stage

Given a set of training faces F_1 to F_k, we fit a mask to each of these faces, then crop each normal face by only keeping the region inside of the fitted mask. For the jth expressioned face of the ith individual e^j_i, we warp it to its normal face r_i. For all the cropped normal faces and the warped faces, we obtain a column vector x_{tex} for each of them. We use x_{tex} to construct a texture PCA space, denoted by V_{tex}, to calculate eigenfaces. The eigenfaces serve as the basis of texture projection.

When constructing the angle PCA space, denoted by V_{ang}, the angle vectors of the fitted masks are used. However, based on the observation that only a small number of angles have notable changes under facial expressions (such as the ones around mouth and eyes), we use relatively invariant angles to capture the intrinsic face geometry in discriminating individuals. This can be achieved by assigning larger weighting factors to these angles. Specifically, we arrange a matrix E, whose column vectors are the angle residual vectors x_{res} for all faces. We then calculate the variance of each row in E, and use the inverse of the variance as the weighting factor for the angles on that row. The angle weights acquired in this way are arranged into a column vector s. $s(p)$ characterizes the stability of the pth angle, i.e., the larger $s(p)$, the more stable the pth angle is. The weighted angle vector, denoted by x^*_{ang}, with its pth element calculated by

$$x^*_{ang}(p) = x_{ang}(p) \cdot s(p), \tag{3}$$

will be used to construct V_{ang}. The eigenvectors in this space serve as the basis of geometry projection.

To build a classifier for face recognition, we jointly utilize the two distances among faces in V_{tex} and V_{ang}. As a result, a weight w is needed for appropriately combining the two attributes. We use the same recognition scheme as will be explained later in the testing stage. Note that, for different values of w, we will obtain different recognition ratios on the training data. Thus, a curve plotting the recognition ratio versus the weight w can give the best w to be used in the testing stage. However, in practice, this curve may be jaggy, or in other words, have more than one values of w leading to maximum recognition ratio. To smooth the curve, we fit a least-square parabola to the plotted curve and select w where the parabola reaches its peak.

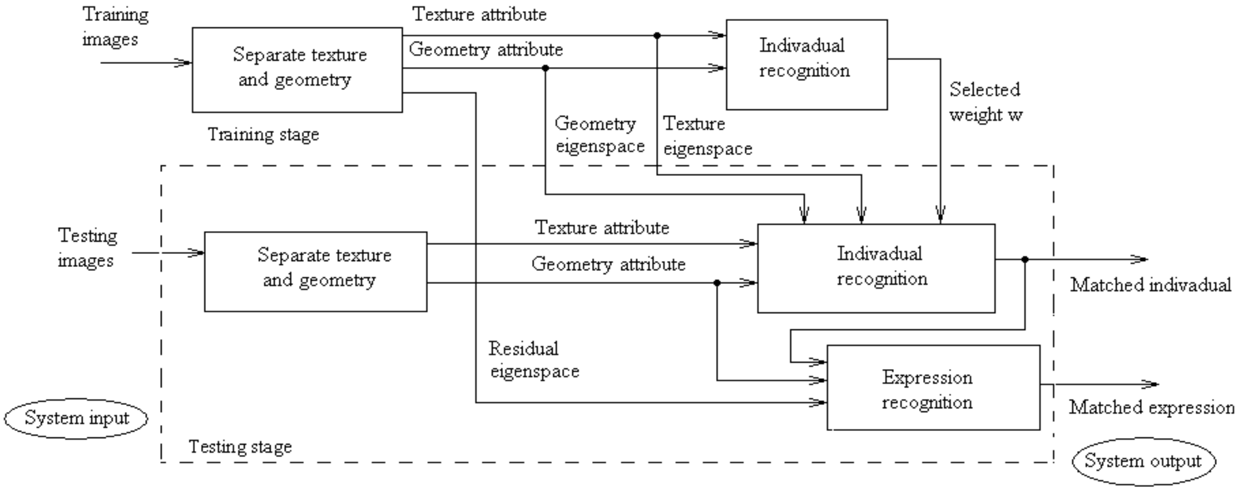

Figure 5. Diagram of the recognition system.

For the expression classification, we calculate angle residuals for expressioned faces. The angle residual PCA space, denote by V_{res}, can be constructed accordingly. Then, all the angle residuals are projected to V_{res}. These projections will serve as *prototypes* in the testing stage. Now, each type of expression can be characterized by a set of prototypes from different individuals.

3.5.2 Testing stage

Given a testing face, we fit a mask to it and warp it to each of the reference faces in the face gallery. The warped faces are projected to V_{tex}. After that, we calculate an Euclidian distance d_{tex} between the testing face and the reference face that it has been warped to. The angle vector of the testing face x_{ang} is weighted by the angle weighting vector s, leading to x^*_{ang}, and then projected to V_{ang}. The distance between the testing face and a reference face in the angle PCA space is calculated as d_{ang}. In a warping process, the testing face preserves its texture, but its geometry will be completely changed to that of the reference face. Therefore, we infer that texture and geometry attributes are uncorrelated to each other, which means they can be linearly combined to form a new classifier. Such a combined distance between a testing face and a reference face is then calculated as

$$d_{comb} = w \times d_{tex} \ + \ (1 - w) \times d_{ang} \tag{4}$$

where $0 \leq w \leq 1$ is the weighting parameter. Finally, we can identify a reference face for each testing face by selecting the one with the smallest d_{comb}.

So far, we have identified a reference face for each testing face. We then collect the angle residual x_{res} for each testing face with respect to its reference face. We project x_{res} to V_{res} constructed in the training stage, and calculate the Euclidean distances between the projections of the testing face and all the prototypes. Then, we can obtain an average value of the distances between the testing face and those prototypes belonging to the same type of expression. Indeed, this average distance is able to tell us how far the expression on the testing face is from each type of expression and thus can be used to classify expressions.

4 Results

The Yale Faces database [17] is used for our experiments. In this database, there are 90 face images with/without expressions, for 15 individuals. Each individual has six different types of facial expressions, namely: "normal", "happy", "sad", "sleepy", "surprise" and "wink".

Due to the limited number of individuals in the database and to ensure cross-validation, we arrange these faces to conduct three experiments. Specifically, in the first experiment, we place the first five individuals' expressioned faces into the testing set. Their normal faces form the face gallery. The remaining ten individuals' faces are placed into the training set. In the second experiment, we place the expressioned faces of individuals numbered from 6 to 10 into the testing set; the face gallery and the training set are formed in a similar way as the first experiment. In doing so, each face will be included in the testing set only once.

4.1 Face recognition

To compare with typical face recognition methods, we first use Eigenface to recognize the original faces without warping. To align the faces, we manually pick the nose tip. A 150 × 150 square centered at the nose tip is cut off and taken as the aligned face. Fig. 2 (a) to (f) and (m) are such examples. Following the same experiment setup as mentioned above, we divide the input data to perform three experiments and in each experiment, we place five individuals' expressioned faces into the testing set. All other individuals' faces are put into the training set and used to construct a face PCA space.

To compare the performances when using different attributes in discriminating individuals, we conduct three sets of experiments, which are based on texture attribute, geometry attribute and combined attributes, respectively. Both weighted and un-weighted angle vectors, i.e., x^*_{ang} and x_{ang}, will be used to model face geometry. In the set of experiments based on combined attributes, the weight w is obtained from the training stage. As mentioned above, we use the training set to simulate a recognition process and find a w corresponding to maximum recognition ratio. An exemplary curve plotting the recognition ratio versus w for the training set is shown in Fig. 6. The w corresponding to the peak of the fitted parabola will be used in the testing stage.

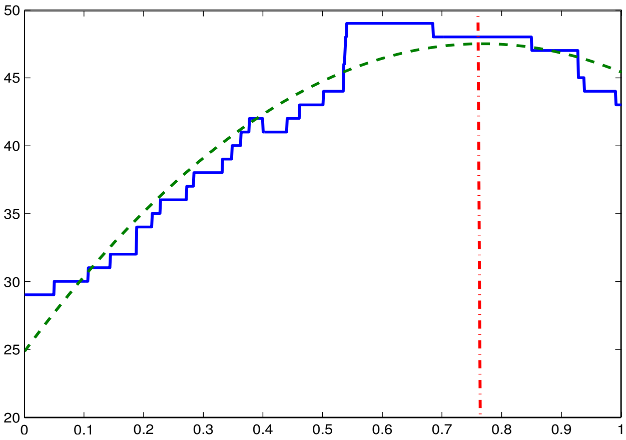

Figure 6. Training for w. The solid line plots the recognition ratio versus w. The dash line is the fitted parabola. The dash-dot line cuts the fitted parabola at its peak. From left to right, larger weight is given to texture attribute.

		experiment 1	experiment 2	experiment 3	over all
(1)	original face	22/25	17/25	23/25	82%
(2)	texture attribute	24/25	23/25	23/25	93.3%
(3)	geometry attribute	12/25	12/25	15/25	52%
(4)	weighted geometry attribute	14/25	17/25	18/25	65.3%
(5)	Combined recognition	25/25	23/25	24/25	96%

Table 1. Results for face recognition. (1) shows the recognition ratio when directly using the Eigenface method on original faces. (2) is the recognition ratio when using texture attribute only. (3) and (4) are the recognition ratios when using geometry attribute only, without and with angle weighting, respectively. (5) is the final combined recognition ratio.

As can be seen from Fig. 6, we achieve a recognition ratio of 29/50 by using geometry attribute only ($w = 0$). On the other hand, if we only use texture attribute, a recognition ratio of 43/50 ($w = 1$) can be achieved. When the two attributes are combined, we will obtain a recognition ratio of 48/50 at the peak of the parabola, $w = 0.7677$.

The recognition ratios for all the experiments have been listed in Table 1. From Table 1, we observe that using texture attribute only, the recognition system has better performance than directly recognizing the original faces, with an improvement of the recognition ratio from 82% to 93.3%, which demonstrates the adverse effect of facial expressions. Using geometry attribute only does not provide satisfactory results, due mainly to the geometry changes under different expressions. However, using weighted geometry attribute, higher recognition ratios are observed in all three experiments. This proves that certain geometry information is relatively stable under different expressions. However, the recognition ratio of using geometry attribute is much lower than that of using texture attribute. This result is reasonable, because face geometries of different individuals can be easily similar. In contrast to the systems using single attribute only, the performance of the proposed system further improves if the two attributes are jointly utilized. It is shown in [18] that both face texture and global configuration serve as attributes in a face recognition system. Our experiment corroborates that a machine recognition system also benefits from combining the two attributes properly.

It is also interesting to investigate how the two attributes contribute proportionally to a machine recogntion system. In Table 2, we list the trained weights w in three experiments, together with its standard deviation. We find that w does not change too much over three experiments, which suggests that the contributions from the two attributes are

	experi- ment 1	experi- ment 2	experi- ment 3	std
w	0.8283	0.7677	0.7676	0.0350

Table 2. The trained weight w in three experiments, the standard deviation is calculated in the last column.

related by a relatively stable ratio. This might be a good indication that our system will have a stable performance when working on larger data sets.

4.2 Expression classification

In the training stage, for each of the five types of facial expressions, we have obtained ten prototypes. In Fig. 7, we plot an exemplary distance matrix between the prototypes obtained from a training set.

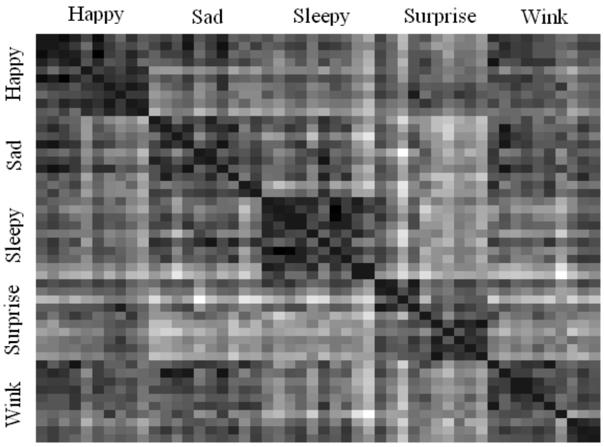

Figure 7. One distance matrix between the prototypes of five types of expressions.

In Fig. 7, the first ten columns/rows correspond to the expression "happy" for individuals 1 to 10. The next ten columns/rows correspond to the expression "sad", and so on. It is easy to observe that there are five 10×10 dark diagonal blocks, indicating that faces carrying the same type of expression have smaller distances in the angle residual PCA space.

However, as can be seen from Fig. 7, there are outliers for almost all types of expressions, shown by the bright dots in the diagonal blocks. We believe that this phenomenon might be attributed to the fact that the ways people annotate an expression may be quite different. For example, one can have one or two eyes blinking under "wink". One can also have a closed or opened mouth under "happy". Such examples can be found in Fig. 8. As can be seen, Figs. 8 (a) and (b) have a widely opened mouth and a slightly opened mouth, respectively, even though they are both defined as "happy". Similar observation can be made for Figs. 8 (c) and (d), which are both defined as "sad". As a result, for each type of expression, there may be several faces that are quite different from each other. To minimize the effect of these outliers, when calculating the average distance from a testing face to each type of expression, we only count the distances between the testing face and its k-nearest neighbors, instead of using all the prototypes.

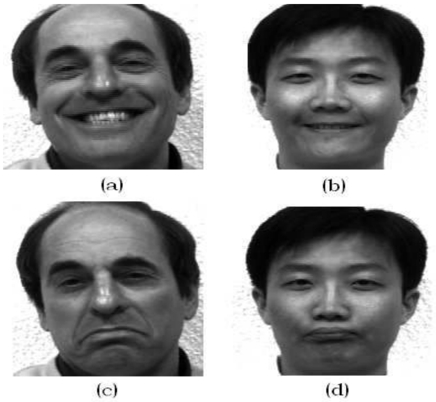

Figure 8. Different faces carrying the same expression: (a), (b): happy faces; (c), (d): sad faces

The expression classification ratio over three experiments is plotted in Fig. 9, where different values of k are used. From Fig. 9, we find that the classification ratio varies

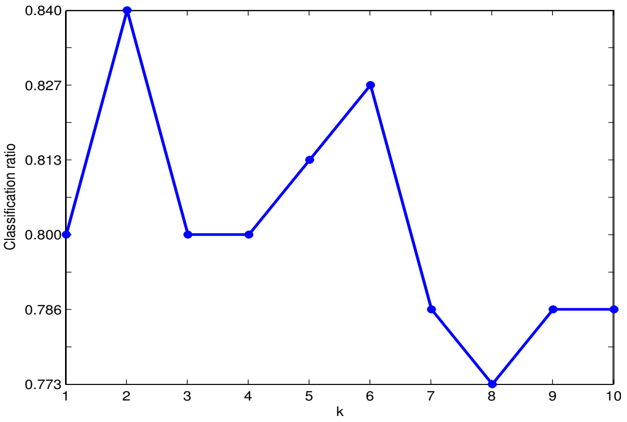

Figure 9. Expression classification ratio. The ratio varies with respect to different k

from the lowest 77.3% to the highest 84%, with different settings of k. The best ratio is obtained when two-nearest neighbors are used.

The expression classification does not provide a ratio as high as face recognition. As mentioned above, it might be caused the ambiguity in defining an expression. In this case, a larger training set for each type of expression may improve the system performance. Another reason is, we may have mis-recognition in the face recognition step. If a testing face is matched to a wrong reference face, the resulting angle residual is no longer meaningful and the mistake will propagate into the expression classification step.

5 Conclusion and future works

In this paper, we constructed three PCA spaces separately modeling face texture, intrinsic geometry and expression information by fitting a generic mask and warping the texture. Based on a combination of texture and appropriately defined geometric attributes, superior recognition performance can be achieved. After face recognition, expressions can be quantitively modeled, enabling our system to classify expressions as well.

Our current mask fitting scheme is based on 14 manually picked markers. The usage of markers makes the system less automatic and likely to cause error. Considering that there are existing methods on automatic face feature registration, the combination of our method with these automatic registration algorithms will further highlight the advantages of our method.

In our experiments, the definition of expression types has to follow what has been provided by the database. However, it may not be an accurate way of categorizing expressions in practice, e.g., faces defined as "sad" do not have clearly distinguishable features and likely resemble faces defined as "normal". In addition, for the database with which we have conducted our experiments, the number of prototypes in each type of expression is quite limited and insufficient to cover different ways people define expressions. Thus, we anticipate that a physiological expression categorizing method and more prototypes representing each type of expression will be helpful in classifying different expressions. Furthermore, in this database, faces do not usually carry heavy expressions. We would expect that, with a specially captured database which include more heavily expressioned faces, our proposed method will further demonstrate its advantages.

References

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *Source ACM Computing Surveys (CSUR)*, 35(4):399–458, Dec. 2003.

[2] S. Z. Li and A. K. Jain. *Handbook of Face Recognition*. Springer. New York. ISBN 0-387-40595-x, 2005.

[3] B. Park, K. Lee, and S. Lee. Face recognition using face-arg matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 27(12):1982–1988, Jul. 2005.

[4] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Expression-invariant 3d face recognition. *Proc. Audio- and Video-based Biometric Person Authentication (AVBPA), Lecture Notes in Comp.*, (2688):62–69, 2003.

[5] L. Wiskott and C. von der Malsburg. Face recognition by dynamic link matching. Lateral Interactions in the Cortex: Structure and Function, www.cs.utexas.edu/users/nn/web-pubs/htmlbook96, 1995.

[6] D. O. Gorodnichy. Video-based framework for face recognition in video. In *The 2nd Canadian Conference on Computer and Robot Vision (CRV'05)*, pages 330–338, May. 2005.

[7] U. Park, H. Chen, and A. Jain. 3d model-assisted face recognition in video. In *The 2nd Canadian Conference on Computer and Robot Vision (CRV'05)*, pages 322–329, May. 2005.

[8] P. Ekman and E. e. Rosenberg. *The facial action coding system: a technique for the measurement of facial movement*. Consulting Psychologists press, 1978.

[9] I. Essa and A.P.Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 19(7):757–763, Jul 1997.

[10] J. Cohn, T. Kanade, T. Moriyama, Z. Ambadar, J. xiao, J. Gao, and H. Iamura. A comparative study of alternative faces coding algorithms. Technical report CMU-RI-TR-02-06, Robotics Institute, CMU, Pittsburgh, Nov. 2001.

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

[12] X. He and S. Yan and Y. Hu and P. Niyogi and H. Zhang Face recognition using Laplacianfaces. *IEEE Trans. on Patt. Anal. and Machine Intel.*, 27(3):328–340, 2005.

[13] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models – their training and application. *Comput. Vis. Image Understand*, (61):18–23, 1995.

[14] T. Cootes, G. J. Edwards, and C. Taylor. Active appearance models. *IEEE Trans. Patt. Anal. Mach. Intell*, (23):681–685, 2001.

[15] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 25(9):1–11, Sept. 2003.

[16] S. M. Platt and N. I. Badler. Animating facial expressions. In *Proceedings of the 8th annual conference on Computer graphics and interactive techniques*, pages 245–252, 1981.

[17] Yale database. http://cvc.yale.edu.

[18] J. S. L. A. Schwaninger and S. M. Collishaw. Role of featural and configural information in familiar and unformiliar face recognition. In *Lecture notes in computer science 2525*, pages 643–650, 2002.

0-7695-2542-3/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY