

Facial Recognition using Hausdorff- Shape- Radon Transform

Rerkchai Fooprateepsiri and Werusak Kurutach

Faculty of Information Science and Technology, Mahanakorn University of Technology

51 Cheum-sampan rd., Nongchok, Bangkok, 10530, Thailand

{*rerkchai*, [werasak](mailto:werasak@mut.ac.th)}@mut.ac.th

doi: 10.4156/jdcta.vol3.issue2.fooprateepsiri

Abstract

This paper proposes a robust method for face recognition with variant illumination, scaling and rotation. Techniques introduced in this work are composed of two stages. First, the feature of face is to be detected by the combined of Trace Transform and Fast Active Contour. Then, in the second stage, the Hausdorff distance and Modified Shape Context are employed to measure and determine of similarity between models and test images. Finally, our method is evaluated with experiments on the AR, ORL and Yale face database using 6,325 face images and compared with other related works (e.g. Eigen face). The extensive experimental results show that the average of accuracy rate of face recognition with variant illumination, scaling and rotation is higher than 84%.

Keywords

Facial Recognition System, Radon Transform, Hausdorff distance, Shape context, Active Contour

1. Introduction

Recognition of faces in digital photographs remains a challenging problem despite of over three decades of research efforts. Facial Recognition [1-6] is substantially different from classical pattern recognition problems, such as object recognition. The shapes of the objects are usually different in an object recognition task, while in face recognition one always identifies objects with the same basic shape. This is very difficult for a face recognition system when one tries to discriminate faces all of which have the same shape with minor texture differences. The face recognition therefore depends heavily on the particular choice of face representation. The aim of feature selection in face representation method is to suppress the variations of face images and simultaneously provide enhanced discriminatory power. It has many image representations proposed for face recognition such as eigenface and fisherface methods. The goal of the eigenface method is to linearly projecting the

image space to a feature space of lower dimensionality [7]. One can reconstruct a face-like image by using only a few eigenvectors which correspond to the largest eigenvalues. Eigenface is an optimal reconstruction method in the sense of minimum mean square error, which projects the image on the directions that maximize the total scatter across all classes of face images. This means that the eigenface is not the optimal method in the sense of discrimination ability, which depends on the separation between different classes rather than the spread of all classes. For the problems of class separation, a method based on class specific linear projection was proposed by Belhumeur et al. [8]. This method tries to find the eigenvectors for which the ratio of the between-class scatter and the within class scatter is maximized. In general, it is common to use principal component analysis in conjunction with linear discriminate analysis (LDA) to overcome the complication of singular eigenvectors. This is achieved by using PCA for reducing the dimensionality of the image space to obtain the eigenspace, and then applying the standard. Fisher LDA (FLD) to further reduce the dimensionality of the eigenspace. It is known that the fisherface is superior to the eigenface approach when the training images contain much variation in an individual class; otherwise the performances of fisherface and eigenface are not significantly different [9].

This paper presents a face feature extraction and recognition method that employs the texture representation derived from the Trace transform. A distance measure is also proposed by incorporating the weighted Trace transform in order to select only the significant features from the Trace transform. The organization of this paper is as follows. Section 2 introduces a method for tracing line on an image and some trace functional we used in this paper. We introduce a shape matching measure in section 3. In section 4, we introduce a shape similarity measure that integrates spatial as well as structural information on the shape by which a more accurate shape matching measure can be achieved. Section 5 presents our experimental results. Finally, we conclude in section 6.

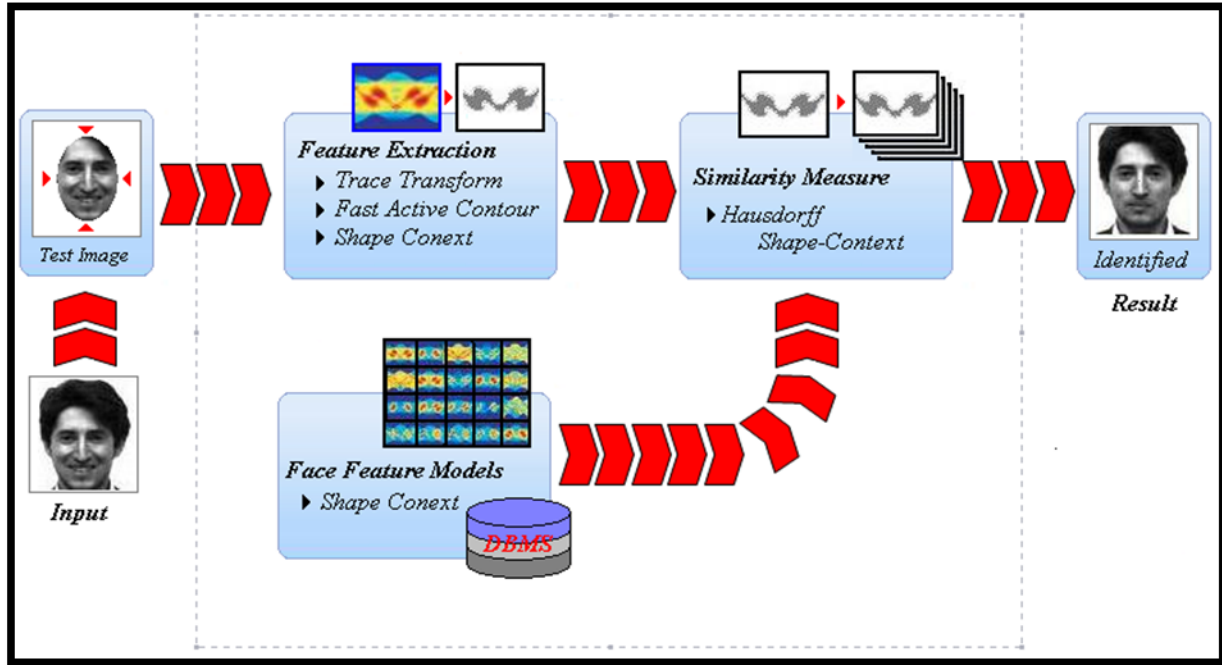


Figure1. Facial Recognition Systems.

2. Feature Extraction Section

2.1 The Trace Transform

The Trace transform [10,11], a generalization of the Radon transform, is a new tool for image processing which can be used for recognizing objects under transformations, e.g. rotation, translation and scaling. To produce the Trace transform one computes a functional T along tracing lines of an image. Each line is characterized by two parameters, namely its distance ρ from the centre of the axes and the orientation ϕ the normal to the line has with respect to the reference direction.

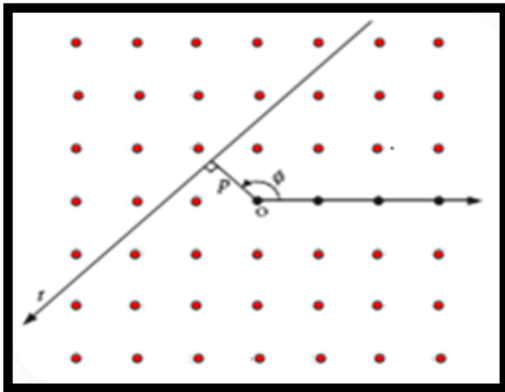


Figure 2. Tracing line with parameters ϕ , ρ and t

In addition, we define parameter t along the line with its origin at the foot of the normal. The definitions of these three parameters are shown in Figure 2. With the Trace transform the image is transformed to another “image”, which is a 2-D function $g(\phi, \rho)$. Different Trace transforms can be produced from an image using different functional T . In our current implementation we have 22 different trace functional, 3 of which are shown below.

$$T(f(t)) = \int_0^{\infty} f(t) dt \quad (1)$$

$$T(f(t)) = \int_0^{\infty} \sqrt{|t-c|} f(t) dt \quad (2)$$

$$T(f(t)) = \int_0^{\infty} (t-c)^2 f(t) dt \quad (3)$$

The first one is just the integral of the image function along the tracing line. This produces the Radon transform of the image. The second and third are some robust moments of the image along the tracing line.

2.2 Fast Active Contour

In this section, we propose an object boundary detection [12], a gradient vector flow (GVF) is defined

to be the vector $V_G(x, y) = [u(x, y), v(x, y)]$, that points into the object boundary, where (x, y) represents pixels coordinates and u and v are projects of a vector in x and y direction, respectively. We assume that each pixel in the snake curve moves from $P_{i,j}$ to $P_{i,j+1}$, where i is the pixel index in snake curve and j is the iteration number. The $P_{i,j+1}$ is computed by:

$$P_{i,j+1} = P_{i,j} + [V_{\text{int}}(P_{i,j}) + \Gamma V_{\text{ext}}(P_{i,j})], \quad (4)$$

where V_{int} is the internal force vector that minimizes the energy function

$$E(V_{\text{int}}) = \alpha E_{\text{cont}}(V_{\text{int}}) + \beta E_{\text{curv}}(V_{\text{int}}), \quad (5)$$

E_{cont} and E_{curv} are continuity and curvature forces, respectively, due to discontinuities or bending. V_{ext} is the external force vector that directs the points of curve to object boundary. We employ the 8-neighborhood average in the GVF field to be the external force vector field. That is

$$V_{\text{ext}}(x, y) = \frac{1}{8} [V_G(x-1, y-1) + V_G(x, y-1) + V_G(x+1, y-1) + V_G(x-1, y) + V_G(x+1, y) + V_G(x-1, y+1) + V_G(x, y+1) + V_G(x+1, y+1)] \quad (6)$$

This scheme can resist more level of noise. As for α , β and Γ they are adjustable coefficients. Form our simulation, $\alpha = \beta = 0$ and $\Gamma = 1$ are adopted. i.e., $V_{\text{int}} = 0$ in (4). The model then becomes

$$P_{i,j+1} = P_{i,j} + V_{\text{ext}}(P_{i,j}), \quad (7)$$

This means that the internal force is zero and the motion of snake entirely depends on external force. The GVF provides enough information to pull a snake approaching object boundary. Each snake pixel moves to a new position in each iteration with the guidance of GVF. When a snake pixel reaches true object boundary, it will stop even though the iteration is continuous. Therefore, we can set the number of iterations larger than the distance between the initial snake contour and true object boundary. The computational complexity in each iteration is then $O(n)$ and faster than $O(nm^3)$ of original snake

algorithm [13] and $O(nm)$ of previously proposed greedy snake [14]. Where n the number of points in snake is curve and m is the number of neighborhoods to be searched in each iteration. We can also sample the snake points to further reduce the computational complexity. Interpolation and deletion may be necessary to avoid large spacing or heavy clustering of snake point.

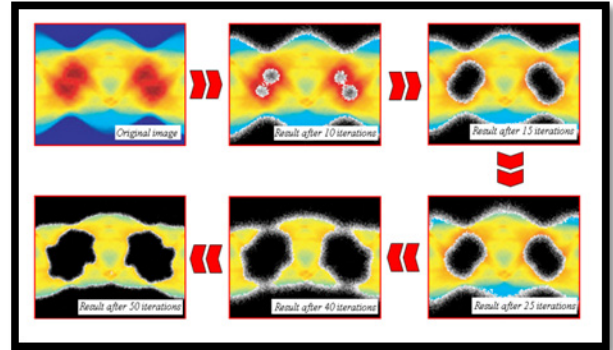


Figure 3. Boundary detection by Fast Active Contour

3. The Shape Context

In our approach, a shape is represented by a discrete set of points obtained from an object boundary detector. Let us denote by $P = \{p_1, \dots, p_n\}$, $p_i \in \mathcal{R}^2$ the set of n edge pixels. The shape context at a reference pixel is computed from the set of the remaining $n-1$ points [15]. In other words, at an edge pixel p_i , we compute a coarse histogram h_i of the relative coordinates of all other edge points in the image, with respect to the pixel p_i :

$$h_i(k) = \#\{q \neq p_i \mid (q - p_i) \in \text{bin}(k)\}, \quad (8)$$

The histogram h_i is therefore defined as the shape context of p_i . Each shape context is a log-polar histogram of the coordinates of the rest of the point set measured using the reference point as the origin. More practically, we count the number of points that fall inside each bin k . The log-polar histogram bins are used in computing the shape context as shown in Figure 4. As shown in Figure 4, the shape context is a compact representation of the distribution of points relative to each point. This log-polar histogram is more sensitive to positions of nearby sample points than to those of points farther away. The cost of matching

between two points p_i and q_i is denoted by $C(p_i, q_i)$ and is computed by using the χ^2 test statistic:

$$C(p_i, q_i) = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_i(k)]^2}{h_i(k) - h_i(k)}, \quad (8)$$

where p_i is a point on shape P and q_i a point on shape Q , and $h_i(k)$ denote the K-bin normalized histogram at p_i and q_i , respectively.

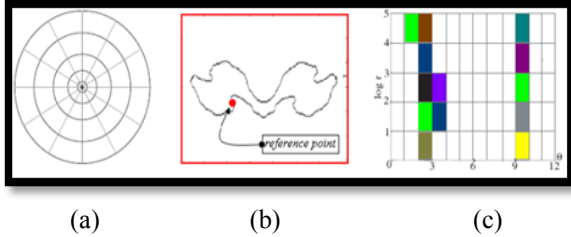


Figure 4. (a) Diagram of log-polar histogram. (b) Shape Trace transform computed from trace functional 1. (c) Shape context example of the point marked by \circ in (b) (dark = large value). Here we use 5 bins for ϕ and 12 bins for r

3.1 Shape Context with 2nd Order Derivative and Distance Measure

The shape context is a rich descriptor providing an image representation which describes a shape characteristic around a specific point. It is however not powerful enough to use the shape context alone in order to find the correspondences in distorted objects, such as shape of face boundary. We propose here a new approach to improve the performance by incorporating new features to the shape context. Let us define by $C_k = C_{sc}(p_i, q_i)$ the shape context cost at position $k = (i, j)$, and then the previous position $C_{k-1} = C_{sc}(p_{i-1}, q_{i-1})$ and the next position $C_{k+1} = C_{sc}(p_{i+1}, q_{i+1})$ are the closest points to position k . In an image processing, a second-order derivative of the one-dimensional function $f(x)$ is the difference, i.e. $\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$. We then define a second-order derivative as the difference between the closest points at position k .

$$C_{diff}(p_i, q_i) = C_{k+1} + C_{k-1} - 2C_k, \quad (10)$$

where the assumption comes from the fact that the closest points $k-1$ and $k+1$ should also have the similar shape context. In addition, the corresponding points between two similar shapes, e.g. two points on the original and distorted versions of the same object, tend to close to each other. The distance between two points can then be defined as

$$C_{dist}(p_i, q_i) = \begin{cases} 0 & \text{if } \|p_i - q_i\| \leq \tau \\ \|p_i - q_i\| & \text{Otherwise,} \end{cases} \quad (11)$$

where $\|\bullet\|$ is a norm between two points, and τ is some threshold. Finally, the shape context cost is a weighted summation

$$C(p_i, q_i) = \omega_1 C_{sc}(p_i, q_i) + \omega_2 C_{diff}(p_i, q_i) + \omega_3 C_{dis}(p_i, q_i), \quad (12)$$

where ω_1, ω_2 and ω_3 are weighting parameters with $\sum \omega_i = 1$. These additional features can be used to improve the performance of the shape context in which the more correct corresponding points can be achieved. The better the corresponding points are known, the better the transformation between two shapes can be accomplished.

3.2 Thin Plate Spline Model

The points on one shape can be warped to other points on the other shape by Thin Plate Spline model [16]. This can be achieved by finding the corresponding points between two shapes by shape context cost matching, and then using these correspondences as the key points for geometric transformation. These points may be transformed by affine model

$$T(x) = Hx + b, \quad (13)$$

where H is a matrix of homogeneous coordinates and b is the translational offset vector. The least squares solution $\hat{T} = (\hat{H}, \hat{b})$ is then computed by

$$H = (Q + P)^t, \quad (14)$$

$$\hat{b} = \frac{1}{n} \sum_{i=1}^n (p_i - q_{\pi}(i)), \quad (15)$$

where P and Q contain coordinates on shape P and Q , respectively, i.e.,

$$P = \begin{bmatrix} 1 & p_{11} & p_{12} \\ \vdots & \vdots & \vdots \\ 1 & p_{n1} & p_{n2} \end{bmatrix}, Q = \begin{bmatrix} 1 & q_{11} & q_{12} \\ \vdots & \vdots & \vdots \\ 1 & q_{n1} & q_{n2} \end{bmatrix}$$

Let us define by the matrix of this system, i.e.,

$$L = \begin{bmatrix} K & P \\ p^t & 0 \end{bmatrix},$$

where L is a $(n+3) \times (n+3)$ matrix. In a 2-dimensional homogeneous coordinate representation, points are transformed from coordinate $p_i = (x_i, y_i)$ to $v_i = (x'_i, y'_i)$ with the matrix operation

$$\begin{bmatrix} v \\ 0 \end{bmatrix} = \begin{bmatrix} K & P \\ p^t & 0 \end{bmatrix} \cdot \begin{bmatrix} w \\ a \end{bmatrix},$$

where

$$K = \begin{bmatrix} 0 & \cdots & U(r_{1n}) \\ \vdots & \ddots & \vdots \\ U(r_{n1}) & \cdots & 0 \end{bmatrix} \quad (16)$$

with $r_{i,j} = \|(x_i, y_i) - (x'_j, y'_j)\|$, v and v are column vector constructed from w_i and v_i , a is column vector with elements a_1, a_x, a_y and 0 is a 3×3 matrix of zeros. Let us define by the target function values at corresponding point $p_i = (x_i, y_i)$ in the plane, where v_i is usually set to (x'_i, y'_i) . We can write Thin Plate Spline model interpolate model $f(x, y)$ to minimize the bending energy

$$I_f = \iint_{R^3} \left(\frac{\partial^2 f}{\partial x^2} \right)^2 + \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 dx dy \quad (17)$$

which has the form

$$f(x, y) = a_1 + a_x x + a_y y + \sum_{i=1}^n w_i U(\|(x_i, y_i) - (x, y)\|) \quad (18)$$

where $U(r)$ is a kernel function and defined by $U(r) = r^2 \log r^2$, with the following constrain solutions:

$$\sum_{i=1}^n w_i = 0 \text{ and } \sum_{i=1}^n w_i x_i = 0, \sum_{i=1}^n w_i y_i = 0.$$

Finally, the matrix v of target point is obtained at each iteration; we then use the matrix as the initial point for the next iteration by which the bending energy function in (13) is minimized.

4. Similarity Measure

4.1 The Classical Hausdorff Distance

Given two point sets A and B , the Hausdorff distance [17] between A and B is defined as

$$H(A, B) = \max(h(A, B), h(B, A)), \quad (19)$$

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|, \quad (20)$$

$$h(B, A) = \max_{b \in B} \min_{a \in A} \|a - b\|, \quad (21)$$

where $\|\bullet\|$ denotes some norm of points of A and B . This measure indicates the degree of similarity between two point sets. It can be calculated without an explicit pairing of points in their respective data sets. The conventional Hausdorff distance, however, is not robust to the presence of noise. Dubuisson et. al.[18] have studied 24 different variations of the Hausdorff distance in the presence of noise. A modified Hausdorff distance (MHD) using an average distance between the points of one set to the other set gives the best result. This measure is the most widely used in the task of object recognition and defined as

$$h(B, A) = \frac{1}{n} \sum_{a \in A} \min_{b \in B} \|a - b\|, \quad (22)$$

with $h(B, A)$ defined similarly. This modified Hausdorff distance is less sensitive to noise than the conventional one. It is possible, however, to end show the Hausdorff distance with even more attractive features as it is shown in the next section.

4.2 The Hausdorff-Shape Context

In this section, we propose a shape similarity measure, the ‘‘Hausdorff-Shape Context’’, based on the combination of the Hausdorff distance and the shape context. The Hausdorff distance measures the distance from point a to all points of set B , $d(a, B)$, then, selects the one at the minimum distance among them. In this case, the candidate point marked by \square is selected and, then, the distance between them is used as the result. The minimum distance is therefore based only on the spatial information. This is not useful when using the Hausdorff distance in the presence of noise, when we have to deal with the broken point problem caused by segmentation and edge detection errors, etc. To the best of our knowledge, there is no work in the point matching Hausdorff distance with structural point information. We propose an alternative way to find the minimum distance between point a and set B to overcome the above problem. Instead of finding the nearest distance, in our approach, the point descriptor, shape context, is used to find the best matching between point a and set B . We, therefore, call this shape similarity measure as ‘‘Hausdorff-Shape context’’.

$$h_{HSC}(A, B) = \sum_{a \in A} \omega(a, b') \min_{b \in B} C_{sc}(a, N(b)), \quad (23)$$

$$\omega(a, b') = \frac{D(a, b')}{\sum_{a \in A} D(a, b')} \quad (24)$$

and

$$\sum \omega(a, b') = 1 \quad (25)$$

where $b' = \arg \min_{b \in B} C_{sc}(a, N(b))$ and $C_{sc}()$ is χ^2 test statistic as define in (9). In the example shown in Figure 5 the candidate point b' is the one

marked by \square which is the correct corresponding point between point a and a point in set B .

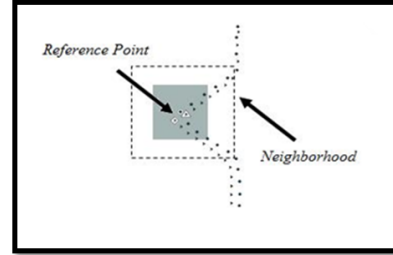


Figure 5. The grey shade indicates the neighborhood area. The point marked by \circ is a sample point a of the first shape A . The points marked by \blacktriangle and \square are the candidate matching points of the second shape B

The cost of matching between two points a and b , $C_{sc}(a, N(b))$ is weighted by their distance, (a, b') . Therefore $\omega()$ is a normalized distance between points a and b' over the entire distance between sets A and B . Furthermore, the neighborhood $N()$ is designed to reduce the computation time of the shape matching, since it finds the best point matching only in the neighborhood area. Thus faster performance improvement can be achieved. The $h_{HSC}(B, A)$ is defined in a similar way. The shape similarity measure in (23) with the maximum Hausdorff matching in (19) is defined to be a confidence level of matching:

$$dist_{HSC}(A, B) = 1 - H(A - B) \quad (26)$$

5. Experimental Results

In this section, we describe a face database we used and then present a face recognition result under rotation, size variation and facial expression. Our proposed method was implemented on the AR [20], ORL [21] and Yale [22] face databases.

In the real world applications, the face recognition should be invariant to rotation, size variation and facial expression. In our proposed method, we use 3 images from the face database for training. The face images for testing were generated by applying a random scaling and rotation factors to the face images, which was distributed within [1-50, 1+50]% and [0, 360] degree.



Figure 6. Examples of face images in AR, ORL and Yale databases.

Examples of test images are shown in Figure 7. The rightmost image is an extreme case of rotated image which is hard to recognize even for the human ability. In general, the face registration method was used in the preprocessing step to locate the faces in an image. The located faces were then normalized (in scale and orientation) for which the two eyes were aligned at the same position. In table 1, we also show the results of facial expression in 3 cases: smile, anger and scream. Smiling and angry expression decrease the recognition rate by some amount. The scream expression is a special case of local deformation of face images in which the geometric location (e.g. mouth position) and texture information are destroyed.



Figure 7. Examples of cropped face images under rotation, size variation and facial expression.

In this case we obtain a recognition rate of 69.87%. In summary, our proposed method is robust to rotation, size variation and facial expression. From the inspection of the table 1, it was found that our proposed method performed better than the eigenface method in all cases. Such robustness comes from the use of masked

Trace transform, weighted Trace transform and matching measure in (26). It also comes from the fact that only the flagged line is used rather than entire face representation which helps us maximize the matching between reference and test images.

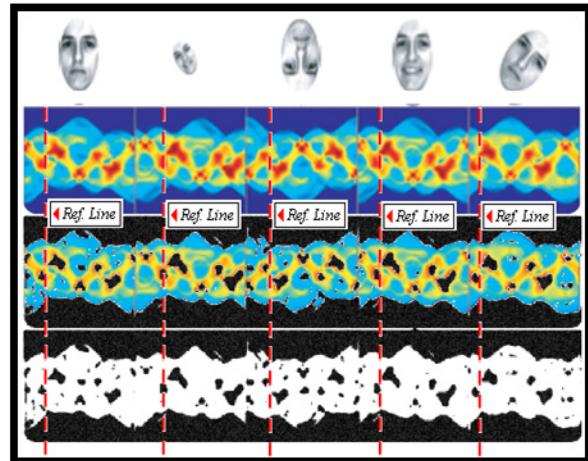


Figure 8. the face images under rotation, size variation and facial expression. The corresponding face representations are shown in bottom row.

Another advantage of our approach is that, when new subjects are added to the system we do not need to retrain on the whole face database, in fact only images of the new subject are used to find the optimal parameter of the algorithm. This may not be the case for eigenface: when new subjects are added to the face database, these systems must be retrained over the whole face database, which is a barrier for real applications.

6. Conclusions

We have presented a highly robust method for face authentication. Techniques introduced in this work are composed of two stages. Firstly, the feature of face is to be detected by the principle of Trace Transform. Then, in the second stage, the Hausdorff distance and modified Shape Context are employed to measure and determine of similarity between models and test images. From the experimental result of 6,325 images, the average of accuracy rate is higher than 87%.

Table 1. Performance of our method.

Condition	Success rate (%)	
	Eigen Face	Our Method
Scaling	68.87	88.96
Rotation	55.43	97.13
Scaling + Rotation	46.86	83.11
Smiling	82.05	91.72
Angry	73.33	91.53
Screaming	45.14	69.87

7. Acknowledgement

This research was supported by 100 Year Shell Scholarship Foundation. The authors would like to thank AR, ORL and Yale face databases for providing testing images.

8. References

- [1] R. Fooprateepsiri and W. Kurutach, "Face Verification Base-on Hausdorff-Shape Context," The 2009 International Asia Conference on Informatics in Control, Automation and Robotics, Thailand, pp. 240-244, 2009.
- [2] A.Thammano and C.Rungruang, "Hausdorff ARTMAP for Human Face Recognition," WSEAS Transactions on Computers, Issue 3, Vol. 3. pp. 667-672.2004.
- [3] M. Martínez, M. Yang and D. J. Kriegman, "Special Issue on Face Recognition," Computer Vision and Image Understanding, Vol. 91, No. 1-2. Academic Press.1-5, 2003.
- [4] Liu and H. Wechsler, "A Shape- and Texture-Based Enhanced Fisher Classifier for Face Recognition," IEEE Transactions on Image Processing, Vol. 10, No. 4, pp. 598-608, Apr. 2001.
- [5] Tefas, C. Kotropoulos and I. Pitas, "Using Support Vector Machines to Enhance the Performance of Elastic Graph Matching for Frontal Face Authentication," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 7, pp. 735-746, Jul. 2001.
- [6] B. Duc, S. Fischer and J. Bigun, "Face Authentication with Gabor Information on Deformable Graphs," IEEE Transactions on Image Processing, Vol. 8, No. 4, pp. 504-516, Apr. 1999.
- [7] L. Kotropoulos, A. Tefas and I. Pitas, "Frontal Face Authentication using Discriminating Grids with Morphological Feature Vectors," IEEE Transactions on Multimedia, Vol. 2, No. 1, pp. 14-26, Mar. 2000.
- [8] M. Turk and A. Pentland, "Eigenfaces for Recognition," Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71-86, 1991.
- [9] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp. 711-720, Jul. 1997.
- [10]Liu and H. Wechsler, "A Shape- and Texture-Based Enhanced Fisher Classifier for Face Recognition," IEEE Transactions on Image Processing, Vol. 10, No. 4, pp. 598-608, Apr.2001.
- [11]Kadyrov and M. Petrou,"The Trace Transform and Its Applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, pp. 811-828,2001.
- [12]M. Petrou and A. Kadyrov, "Affine invariant features from the trace transform," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, pp. 30-44, 2004.
- [13]S. Srisuk, M. Tamsri, R. Fooprateepsiri and P. Sookavatana, "A New Shape Matching Measure for Nonlinear Distorted Object Recognition," International Conference on Digital Image Computing: Techniques and Applications (DICTA'03),Sydney, Australia, pp. 339-348,Dec. 10-12,2003.
- [14]Xu and J. L. Prince,"Snakes, shapes, and gradient vector flow," IEEE Transactions on Image Processing, Vol. 7, No. 3, pp. 359-369, 1998.
- [15]J. Williqms and M. Shah, "A fast algorithm for active contours and curvature estimation," CVGIP: Image Understanding, Vol. 55, No. 1, pp. 14-26, 1992.
- [16]S. Belongie, J. Malik and J. Puzicha, "Shape Matching and Object Recognition using Shape Context," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 24, No. 24, pp. 509-522, 2002.
- [17]L. Bookstein, "Principal Warps: Thin-Plate Splines and Decomposition of Deformations," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 11, No. 6, pp. 567-585, 1989.
- [18]P. Huttenlocher, G. Klanderman and W. Rucklidge, "Comparing Images using the Hausdorff Distance," IEEE Trans.PAMI, Vol. 15(9), pp. 850-863, 1993.
- [19]M. Dubuisson and A. K. Jain, "A Modified Hausdorff Distance for Object Matching," In Proc. ICPR, pp. 566-568, 1994.
- [20]S. Srisuk and W. Kurutach,"A New Robust Face Detection in Color Images,"IEEE Conf.on Automatic Face and Gesture Recognition, Washington D.C., USA, pp. 306-311, May 20-21, 2002.
- [21]AR: http://cobweb.ecn.purdue.edu/face_DB.html
- [22]ORL Face DB: Retrieved from <http://www.uk.research.att.com/facedatabase.html>.
- [23]Yale Face Database: Retrieved from <http://cvc.yale.edu/projects/yalefaces.html>.
- [24]S. Srisuk, M. Petrou, R. Fooprateepsiri, K. Sunat,W. Kurutach and P. Chopaka, "A Combination of Shape and Texture Classifiers for a Face Verification System," Lecture Notes in Computer Science, Vol. 3072, pp. 44-51, 2004.
- [25]S. Srisuk, R. Fooprateepsiri, M. Petrou, S. Waraklang and K. Sunat, "A General Framework for Image Retrieval using Reinforcement Learning," The Image and Vision Computing New Zealand 2003, Massey University, New Zealand, pp. 36-41,Nov. 26-28, 2003.