

NGHIÊN CỨU ỨNG DỤNG EIGENFACES VÀ ĐẶC TRƯNG CỤC BỘ LBP CHO BÀI TOÁN NHẬN DẠNG MẶT NGƯỜI

A STUDY ON APPLYING EIGENFACES AND LBP LOCAL CHARACTERISTICS
TO THE HUMAN FACIAL RECOGNITION PROBLEM

Dương Anh Hùng

Công ty TNHH Xuân Hùng

Email: duonganhhung@gmail.com

TÓM TẮT

Trong những năm gần đây, các nghiên cứu và ứng dụng của trí tuệ nhận tạo (*Artificial Intelligence*) và học máy (*Machine Learning*) thu hút được sự quan tâm của rất nhiều nhà khoa học. Một trong những lĩnh vực liên quan tới công nghệ trí thức mà hiện nay được ứng dụng rất nhiều vào trong cuộc sống là nhận dạng mẫu (*Pattern Recognition*). Trong các bài toán được khá nhiều người quan tâm cho đến thời điểm này là nhận dạng khuôn mặt (*Face Recognition*). Bài báo đã xây dựng thành công hệ thống nhận dạng khuôn mặt trên đặc trưng cục bộ Local Binary Pattern (*LBP*) và subspace Principle Component Analysis (*PCA*). Hệ thống đã được kiểm tra trên bộ cơ sở dữ liệu MBGC, bên cạnh đó bài báo còn xây dựng được một bộ cơ sở dữ liệu của người Việt phục vụ cho việc nhận dạng.

Từ khóa: thuật toán LBP; thuật toán PCA; thuật toán eigenface; không gian con; nhận dạng khuôn mặt

ABSTRACT

In recent years, the studies and applications of artificial intelligence (*Artificial Intelligence*) and machine learning (*Machine Learning*) have drawn the interest of many scientists. One of the intellectual technology- related fields whose applications are present in many aspects of life is Pattern Recognition. Face Recognition has, up to this point, been among problems gaining the most significant attention. The report successfully completes the facial recognition system in Local Binary Pattern (*LBP*) and subspace Principle Component Analysis (*PCA*). The system has been tested on the MBGC database. In addition, the topic builds up a database for Vietnamese people's faces in reality.

Key words: LBP's algorithm; PCA's algorithm; eigenface's algorithm; subspace; face recognition

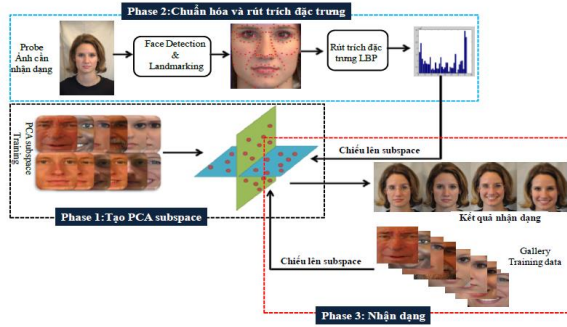
1. Đặt vấn đề

Trong những năm gần đây, các nghiên cứu và ứng dụng của trí tuệ nhận tạo (*Artificial Intelligence*) và học máy (*Machine Learning*) thu hút được sự quan tâm của rất nhiều nhà khoa học. Một trong những lĩnh vực liên quan tới công nghệ trí thức mà hiện nay được ứng dụng rất nhiều vào trong cuộc sống là nhận dạng mẫu (*Pattern Recognition*). Các hệ thống nhận dạng phổ biến hiện nay như: nhận dạng chữ viết (đánh máy hoặc viết tay), nhận dạng chữ ký, nhận dạng vân tay, nhận dạng tròng mắt (*iris*), nhận dạng mặt người,...[1][2].

Một trong các bài toán được khá nhiều người quan tâm cho đến thời điểm này là nhận dạng khuôn mặt (*Face Recognition*). Khuôn mặt

đóng vai trò quan trọng trong quá trình giao tiếp giữa người với người và cũng mang một lượng thông tin giàu có, chẳng hạn như có thể xác định giới tính, tuổi tác, chủng tộc hay trạng thái cảm xúc,... Nhận dạng khuôn mặt không phải là bài toán mới nhưng nó vẫn là một thách thức lớn vì một bài toán nhận dạng mặt người chứa nhiều các bài toán khác như: phát hiện mặt người (*face detection*), đánh dấu (*facial landmarking*), rút trích đặc trưng (*feature extraction*), gán nhãn, phân lớp (*classification*). Ngoài ra, ảnh khuôn mặt trong thực tế chứa đựng nhiều vấn đề như: độ sáng, độ nhòe/mờ, độ nhiễu, độ phân giải, góc ảnh,...[5].

2. Giải quyết vấn đề



Hình 1. Hướng tiếp cận nhận dạng

Bước 1: Tạo PCA subspace dựa trên bài toán eigen

Bước 2: Đầu vào của bước này là ảnh có chứa mặt người cần nhận dạng. Đầu ra của bước này là đặc trưng dùng trong quá trình nhận dạng. Bước này được thực hiện thông qua nhiều giai đoạn như sau: Tiền xử lý, phát hiện khuôn mặt, landmarking, cắt vùng khuôn mặt, rút trích đặc trưng.

Bước 3: Trong bước này, đặc trưng của ảnh cần nhận dạng và đặc trưng của dữ liệu học mẫu được chiếu lên PCA subspace. Dựa vào hệ số chiếu, ta sẽ có được kết quả nhận dạng [9].

2.1. Đặc trưng LBP

2.1.1. LBP cơ bản

Thông tin LBP của pixel tại trung tâm của mỗi khối ảnh sẽ được tính dựa trên thông tin của các pixel lân cận. Có thể tóm tắt các bước tiến hành như sau: [4]

Bước 1: Xác định bán kính làm việc.

Bước 2: Tính giá trị LBP cho pixel ở trung tâm (x_c, y_c) khối ảnh dựa trên thông tin của các pixel lân cận:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=1}^P s(g_p - g_c) 2^p$$

Trong đó, (g_p) là giá trị grayscale của các pixel lân cận, (g_c) là giá trị grayscale của các trung tâm và (s) là hàm nhị phân được xác định như sau: $s(z) = 1$ nếu giá trị $z \geq 0$.

$$LBP \text{ code} = 1*2^0 + 1*2^1 + 1*2^2 + 1*2^3 + 0*2^4 + 0*2^5 + 0*2^6 + 0*2^7 = 15$$

47	51	65	-23	-19	-5	0	0	0
62	70	70	-8	*	0	0	*	1
80	83	78	10	13	8	1	1	1

Hình 2. Tính giá trị LBP với một ma trận 3x3

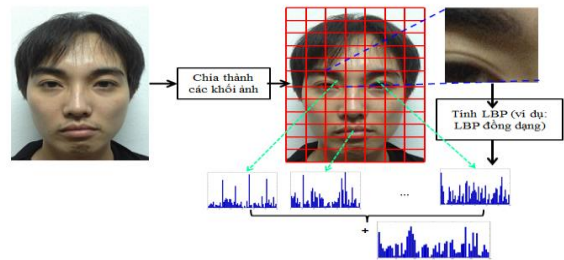
2.1.2. Rút trích đặc trưng LBP

Quá trình rút trích đặc trưng histogram của LBP được tóm tắt như sau: [4][9]

Bước 1: Giả sử ta có 1 bức ảnh số có kích thước là 100x100, ta chia nhỏ bức ảnh ra làm nhiều khối ảnh (blocks) khác nhau có kích thước là 10x10.

Bước 2: Ta tiếp tục chia khối ảnh 10x10 thành các ma trận nhỏ hơn là 3x3 chồng lấp nhau (overlapping) và tiến hành thực hiện LBP lên các ma trận này. Như vậy chúng ta có được 64 ma trận 3x3 từ khối ảnh 10x10. Tuy nhiên thông thường ta dùng kỹ thuật nối (padding) để có thể có được số lượng khối ảnh bằng với kích thước ảnh. Chính vì vậy, như ta đã thấy trong phần LBP cơ bản, đặc trưng LBP có kích thước cùng với kích thước của ảnh gốc. Giả sử trong trường hợp này, ta dùng histogram của LBP (LBP đồng dạng).

Bước 3: Tổng hợp histogram của LBP từ các khối ảnh bằng cách cộng dồn.



Hình 3. Ví dụ minh họa cho rút trích đặc trưng LBP trên khuôn mặt

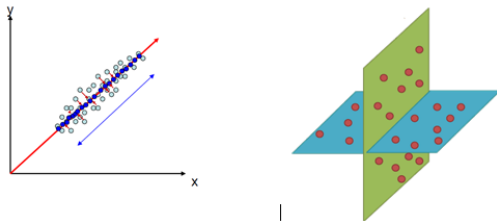
2.2. Phương pháp PCA

2.2.1. Giới thiệu về PCA

Phương pháp phân tích thành phần chính (PCA) là một công cụ rất phổ biến trong việc nhận dạng, được biết đến như là Eigenfaces trong ứng dụng nhận dạng mặt người sử dụng PCA. Cụm từ Eigenfaces được hình thành dựa trên ứng dụng nhận dạng mặt người và bài toán PCA giải bài phương pháp phân tích trị riêng

(eigenvalue). PCA được xem như là phương pháp chuẩn dùng để đánh giá mức độ hiệu quả của các phương pháp khác (baseline benchmark).

Hai phương pháp cơ bản dùng để giải quyết bài toán PCA là: phân tích trị riêng (eigenvalue) và Singular Value Decomposition (SVD). Trong khi SVD giải quyết bài toán PCA dựa trên ma trận dữ liệu (data matrix) thì phương pháp phân tích trị riêng (eigenvalue) làm việc trên hiệp phương sai của dữ liệu. Trong khuôn khổ bài báo, bài toán PCA được giải quyết dựa trên phương pháp phân tích trị riêng (eigenvalue).[6][9]



Hình 4. Ví dụ minh họa cho PCA

2.2.2. Các khái niệm trong PCA

Trị trung bình (mean) của 1 vector là một giá trị đơn (scalar) và được tính theo công thức:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i.$$

Trong đó x_i là thành phần trong vector x . Ví dụ x là 1 vector với giá trị như sau: $x = [5 \ 10 \ 20 \ 50 \ 70]$, trung bình của vector này là 31.

Trong khuôn khổ của bài toán PCA, vector trung bình cột được sử dụng. Cách tính vector trung bình cột hoặc dòng được miêu tả trong ví dụ sau

Vector trung bình cột				Vector trung bình dòng			
1	2	3		1	2	3	
4	5	6		4	5	6	
7	8	9		7	8	9	
10	11	12		10	11	12	
			2				5.5
			5				6.5
			8				7.5
			11				

Hình 5. Ví dụ minh họa cho trị trung bình theo cột và dòng

Độ lệch chuẩn (standard deviation): cho biết độ khác biệt hay độ lệch của các phần tử

trong dữ liệu so với trị trung bình. Cho trước dữ liệu X và trị trung bình của X là μ , độ lệch chuẩn của X được tính như sau:

$$\sigma = \sqrt{E[X - \mu]^2}.$$

Ma trận hiệp phương sai (covariance matrix): giả sử cho trước một ma trận $X = [X_1 \ X_2 \ \dots X_N]$, trong đó X_i là vector. Ma trận hiệp phương sai, ký hiệu là Σ , được định nghĩa như sau:

$$\Sigma = \text{cov}(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)].$$

2.2.3. Bài toán Eigen

Trong ngữ cảnh của bài toán eigen, nếu tồn tại vector x thỏa điều kiện song song với Ax thì x được gọi là vector riêng (eigenvectors) của ma trận A . Tính song song được biểu diễn như sau:

$$Ax = \lambda x$$

Trong đó λ là 1 trị đơn (scalar) và được xem là trị riêng (eigenvalue) của ma trận A .

Cách tính vector riêng (eigenvectors) và trị riêng (eigenvalues) được tổng quát như sau:[10]

Tính định thức (determinant) của $A - \lambda I$.

Tìm trị riêng (eigenvalues) bằng cách giải quyết

$$\det(A - \lambda I) = 0, \text{ trong đó } \lambda \text{ là nghiệm.}$$

Tương ứng với mỗi giá trị trị riêng (eigenvalue) λ_i , ta tính được vector riêng (eigenvector) bằng cách giải phương trình:

$$Ax = \lambda x.$$

2.2.4. Đặt vấn đề và giải pháp của PCA

Đặt vấn đề

Một trong những câu hỏi cơ bản trong bài toán PCA là: cho trước 1 tập dữ liệu, tìm trục chiếu mà có thể phân biệt được dữ liệu tốt nhất, tức là biến thể (variation) lớn nhất. Xem xét ví dụ sau đây: mục đích của bài toán PCA là đi tìm trục chiếu tốt nhất ở hình vẽ bên phải.



Hình 6. Ví dụ về phép chiếu từ không gian 2D xuống 1D

Giải quyết vấn đề: gọi ω là trục chiếu cần tìm. Phép chiếu của dữ liệu x trên trục chiếu này là $\omega^T x$. Mục đích của PCA là tìm ω sao cho biến thiên của $\omega^T x$ lớn nhất. Bài toán được biểu diễn như sau: $\arg \max_{(\omega)} \{\text{var}(\omega^T x)\}$.

Giá trị biến thiên của phép chiếu dữ liệu x lên trục ω được thể hiện như sau:

$$\begin{aligned} \text{Var}(\omega^T x) &= E(\omega^T x - E(\omega^T x))^2 \\ &= \omega^T E(x - E(x)) \times (x - E(x))^T \omega \quad (**). \end{aligned}$$

Trong đó $E[x]$ là trị trung bình của x .

Đặt $\Sigma = E(x - E(x)) \times (x - E(x))^T$ là ma trận hiệp phương sai (covariance matrix).

Công thức (**) tương đương

$$\begin{aligned} \text{Var}(\omega^T x) &= \omega^T \Sigma \omega, \text{ với} \\ \Sigma &= E(x - E(x)) \times (x - E(x))^T \text{ và } \omega^T \omega = 1. \end{aligned}$$

Lưu ý là trong ngữ cảnh của bài toán, ω được chuẩn hóa về vector đơn vị (unit norm), có nghĩa là $\omega^T \omega = 1$.

Mục tiêu của bài toán PCA là đi tìm 1 trục chiếu ω sao cho giá trị biến thiên của phép chiếu $\omega^T x$ lớn nhất (tức là đi tìm $\max(\text{Var}(\omega^T x))$). Như vậy ta có thể phát biểu bài toán như sau:

$$\arg \max_{(\omega)} \{\text{var}(\omega^T x)\}, \text{ s.t. } \omega^T \omega = 1 (***)$$

Giải pháp đơn giản nhất để giải quyết (***) là sử dụng phương pháp tối ưu Lagrangian (Lagrangian optimization).

Từ (***) ta đưa về dạng hàm số

$$\text{Lagrangian: } L(\omega, \lambda) = \omega^T \Sigma \omega - \lambda(\omega^T \omega - 1),$$

Trong đó λ là hệ số Lagrangian. Lấy đạo hàm theo ω , đặt nó có giá trị bằng không (0), ta có công thức sau:

$$\frac{\partial L(\omega, \lambda)}{\partial \omega} = 0$$

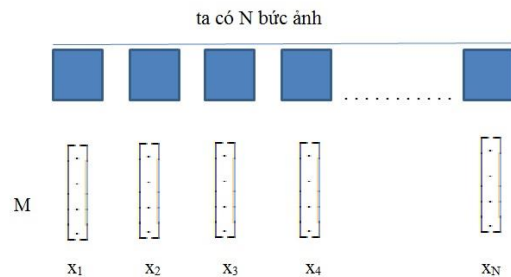
$$\Leftrightarrow 2 \Sigma \omega - 2 \lambda \omega = 0$$

$$\Leftrightarrow \Sigma \omega - \lambda \omega = 0$$

$$\Leftrightarrow \Sigma \omega = \lambda \omega$$

Đây chính là bài toán eigen, trong đó λ là trị riêng (eigenvalues), PCA subspace ω là vector riêng (eigenvector) của ma trận hiệp phương sai $\Sigma = E(x - E(x)) \times (x - E(x))^T$.

2.2.5. Dùng PCA để xây dựng subspace



Hình 7. Minh họa ma trận $m \times n$

Cho trước tập huấn luyện (training data) gồm N ảnh ($x_1, x_2, \dots, x_N$) mỗi ảnh được biểu diễn dưới dạng vector cột có M chiều ($x_i \in \mathbb{R}^M$). Sắp xếp các ảnh lại ta được 1 ma trận X . PCA subspace được xây dựng qua các bước như sau:

$$\text{Bước 1: Chuẩn hóa dữ liệu: } \bar{X} = \frac{1}{N} \sum_{i=1}^N x_i$$

và

$$X = (X - \bar{X}) = \begin{bmatrix} x_1 - \bar{X} & x_2 - \bar{X} & \dots & x_N - \bar{X} \end{bmatrix}$$

Bước 2: Tính ma trận hiệp phương sai của X : $\Sigma = X^T X$.

Bước 3: Tìm nghiệm của phương trình $\Sigma \omega = \lambda \omega$ và được giải bằng phương pháp eigen. Ma trận hiệp phương sai Σ kích thước $N \times N$ nên sẽ có N nghiệm khi tìm lời giải cho bài toán eigen. Tập hợp của các vector riêng (eigenvector) sẽ hình thành 1 PCA subspace.

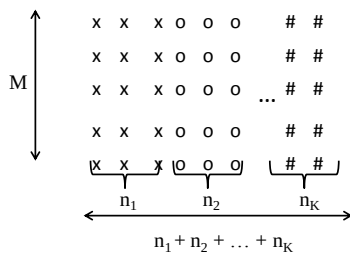
2.2.6. Ứng dụng PCA trong bài toán nhận dạng mặt người

Bước 1: Thực hiện xây dựng không gian con PCA (PCA subspace):

$\omega = \omega_1 \omega_2 \dots \omega_m \in \mathbb{R}^{M \times m}$ từ tập ảnh huấn luyện X.

Bước 2: Thực hiện nhận dạng.

Đầu vào: Dữ liệu huấn luyện Y (khác với dữ liệu huấn luyện X được sử dụng trong xây dựng subspace). Giả sử dữ liệu này gồm có K lớp (class), mỗi lớp đại diện cho 01 đối tượng và mỗi đối tượng được thể hiện bằng nhiều ảnh khác nhau. Giả sử lớp thứ k (gọi là C_K) chứa n_k ảnh đại diện cho người K này. Mỗi ảnh được thể hiện là 1 vector cột có M chiều ($x_i \in \mathbb{R}^M$). Dữ liệu huấn luyện $Y \in \mathbb{R}^{M \times (n_1 + n_2 + \dots + n_K)}$ được thể hiện như sau:



Để tiện sử dụng, ta đặt $S = n_1 + n_2 + n_3 + \dots + n_K$, khi đó $Y \in \mathbb{R}^{M \times S}$

Xây dựng không gian con PCA (PCA subspace) $\{\omega_i\} \in \mathbb{R}^{M \times m}$ đã xác định trước đó. Đưa ảnh cần nhận dạng về vector cột có M chiều ($y \in \mathbb{R}^M$).

Đầu ra: Xác định y thuộc về lớp nào trong những lớp $\{C_1, C_2, \dots, C_K\}$

Tiến hành: Chiếu tập huấn luyện Y lên PCA subspace ω , ta sẽ có S hệ số $\alpha_1, \alpha_2, \dots, \alpha_S$, trong đó

$$\alpha_1 = \omega^T Y_1, \alpha_2 = \omega^T Y_2, \dots, \alpha_{n_i} = \omega^T Y_{n_i}$$

Và S là tổng số các mẫu có trong tập huấn luyện Y. Chiếu dữ liệu được kiểm tra y lên PCA subspace ω : $\alpha_y = \omega^T y$.

Thực hiện so sánh hệ số (coefficients) giữa α_y và $\alpha_1 \alpha_2 \dots \alpha_S$.

3. Kết quả thực hiện

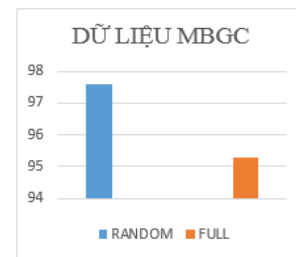
3.1. Kết quả trên dữ liệu chuẩn

Tập training: gồm có 860 bức ảnh của 43 đối tượng trong cơ sở dữ liệu chuẩn MBGC

Tập testing: chạy random đối với 213 bức ảnh của 43 đối tượng.

Hiện tại thí với cơ sở dữ liệu chuẩn MBGC (Multiple Biometric Grand Challenge Database) thì kết quả nhận dạng đạt được trên 90% (>90%) độ chính xác.

	DỮ LIỆU MBGC
Random	97.6 %
FULL	95.3 %



Hình 8. Kết quả nhận dạng dữ liệu MBGC

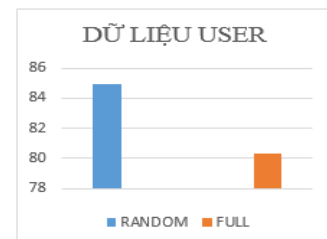
3.2. Kết quả trên dữ liệu thực tế

Tập training: gồm có 121 bức ảnh của 11 đối tượng cụ thể.

Tập testing: được chạy random 53 bức ảnh của 11 đối tượng.

Các kết quả dữ liệu thực tế được học viên thu thập trong điều kiện ngoại cảnh khác nhau, ảnh hưởng của ánh sáng, góc chụp, các biểu hiện trên khuôn mặt và chất lượng của camera cũng như chất lượng bức ảnh không đồng đều nên kết quả đạt được tầm từ 70% - 80%.

	DỮ LIỆU USER
Random	84.9 %
FULL	80.3 %



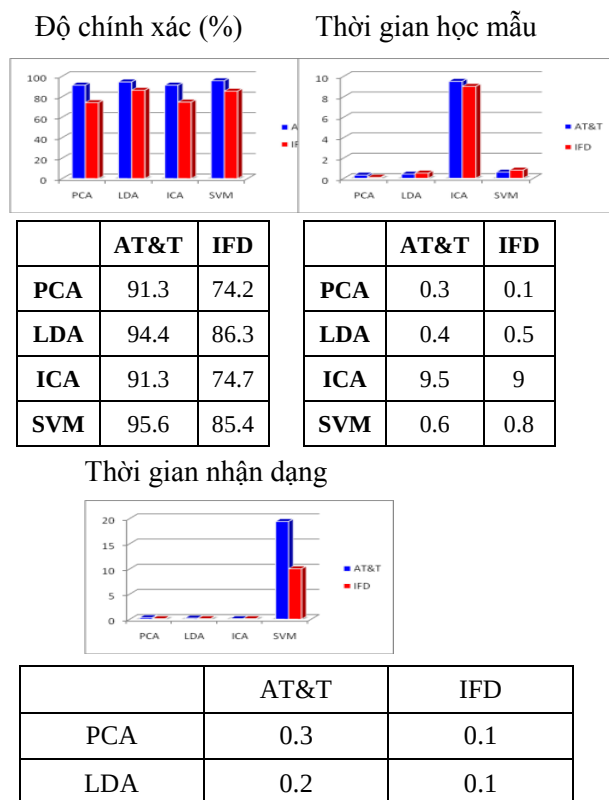
Hình 9. Kết quả nhận dạng dữ liệu USER

3.3. So sánh đánh giá

Như đã phân tích và trình bày ngay từ

đầu bài báo, PCA là thật toán tốt, chạy ổn định trên mọi kiểu dữ liệu, chính là vì PCA được chọn làm baseline.[3][7][8]

Bảng 1. Bảng so sánh các phương pháp về độ chính xác, thời gian học mẫu và thử nghiệm dựa trên bộ cơ sở dữ liệu



ICA	0.1	0.1
SVM	19.39	10

4. Kết luận

Mục đích của bài báo là tìm hiểu và đánh giá một số thuật toán về xử lý ảnh, nhận dạng để từ đó xây dựng một hệ thống nhận dạng khuôn mặt hoàn chỉnh nhằm phục vụ cho ứng dụng trong thực tế. Nội dung bài báo đã thực hiện đầy đủ việc tìm hiểu và nghiên cứu một số thuật toán tiêu biểu tại từng giai đoạn xử lý của quá trình nhận dạng khuôn mặt. Trọng tâm nghiên cứu là các thuật toán có khả năng thực hiện một cách chính xác nhất cho quá trình nhận dạng. Cụ thể là tìm hiểu và trình bày hai thuật toán quan trọng nhất là nhị phân cục bộ LBP nhằm rút trích đặc trưng cho khuôn mặt và bộ phân lớp PCA cho quá trình nhận dạng.

Về thực tế thì bài báo đã xây dựng được một công cụ nhận dạng khuôn mặt hoàn chỉnh, có thể đem ứng dụng trực tiếp vào thực tế. Hệ thống nhận dạng đã hoạt động khá chính xác, cho kết quả đạt được là hơn 90% với bộ thư viện chuẩn MBGC và đạt yêu cầu đề ra là từ 70-80% cho bộ dữ liệu mà học viên tự mình thu thập được.

TÀI LIỆU THAM KHẢO

- [1] PGS.TS. Nguyễn Quang Hoàng (2006), *Bài giảng Xử lý ảnh*, NXB Thống kê, Hà Nội.
- [2] Hoàng Kiếm (1992), *Nhận dạng – Các phương pháp và ứng dụng*, NXB Thống kê, Hà Nội.
- [3] Ahonen. T, Hadid. A, and Pietikainen. M (2004), *Computer Vision - ECCV 2004*, (3021).
- [4] Anagha V. Malkapurkar¹, Rupali Patil², Prof. Sachin Murarka (2011), *A New Technique for LBP Method to Improve Face Recognition*.
- [5] Guo, G, Li S. Z. and Chan K (2000), *Face Recognition by Support Vector Machines*, *Proc. of 4th IEEE Intern. Conf. on. Automatic Face and Gesture Recognition*, 106–201.
- [6] Jolliffe, I.T. (2002), *Principal Component Analysis, second edition* (Springer).
- [7] Ojala. T, Pietikainen. M and Harwood. M (1996), *A Comparative Study of Texture Measures with Classification Based on Feature Distributions*, *Pattern Recognition*.
- [8] Rafael C. Gonzalez, Richard E. Woods (2007), *Digital Image Processing*
- [9] Stan Z. Li, Anil K. Jain (2011), *Handbook of Face Recognition, 2nd Edition* (Springer)
- [10] Turk. M. and Pentland. A (1991), *Eigenfaces for recognition*. *J. Cogn. Neurosci.*

(BBT nhận bài: 01/08/2013, phản biện xong: 23/08/2013)
