

Fisherface 알고리즘과 Elastic Graph Matching을 이용한 얼굴 인식

이형지, 이완수, 정재호
 인하대학교 전자공학과

Hyung-Ji Lee Wan-Su Lee Jae-Ho Chung
 Dept. of Electronic Engr., Inha Univ.
 E-mail : g1991177@inhavision.inha.ac.kr

요약문

본 논문에서는 K-L 변환을 기반으로 한 Fisherface 알고리즘과 Elastic Graph Matching 방법을 사용하여 보다 효율적인 얼굴 인식 방법을 제안하고자 한다. 즉, 얼굴의 모양 정보뿐만 아니라, 영상 픽셀의 그레이 정보를 동시에 이용할 수 있는 Elastic Graph Matching 방법과 통계학적으로 신호의 차원을 줄일 뿐만 아니라 주위 환경의 변화에 강인한 Fisherface 알고리즘을 효율적으로 결합하고자 한다. 본 연구에서는 얼굴을 인식하는데 있어서 주로 문제가 되는 광원의 위치에 따른 조도의 변화, 얼굴 표정 변화, 안경 착용 여부 등을 고려한 Yale대학 데이터베이스에 대해 기존의 여러 얼굴 인식 알고리즘을 실험 및 비교하였다. 또한, 이들을 효율적으로 결합할 수 있는 인식 알고리즘을 제안함으로써 인식을 및 속도 면에서 비교적 만족할 만한 결과를 얻을 수 있었다.

서론

최근 정보화 사회가 발달함에 따라 전화, 인터넷 등의 다양한 네트워크를 이용한 사용자의 시스템에 대한 정보처리 요구와 접근이 급격히 증가하고 있으며, 이에 따른 정보의 보안 문제가 심각해지고 있다. 이러한 이유로 도난이나 위조의 문제점이 거의 없고 네트워크 시대에 쉽게 적용되어 사용자의 신원을 확인, 인증할 수 있는 방법으로 생체 측정학이 사용될 수 있다. 생체 측정학에 있어서 주로 얼굴, 지문, 음성, 손등, 홍채 인식 등이 사용되고 있는데, 특히 얼굴 인식의 경우, 사용자에게 거부감을 주지 말아야 하는 장소와 사용자의 협조적인 자세가 불가능한 환경에 적용될 수 있다. 즉, 변별력과 친화성을 동시에 만족하는 이상적인 생체 측정 수단이라 할 수 있다.

얼굴 인식의 일반적인 인식 과정 중에 제일 중요한 부분은 어떠한 특징 벡터를 어떻게 사용하여 인식하는가이다. 1990년대 초반 K-L 변환을 이용한 얼굴 인식 방법[1]이 MIT에서 제안되었으며 이를 응용한 Fisherface방법[2]과 가버(Gabor) 필터 응답을 이용한 Elastic graph matching[3]과 같은 방법이 Vissage, Miros, Visionics 사와 같은 상업적인 회사에서 주요한 알고리즘으로 사용되고 있다. 최근에는 ICA(independent component analysis)를 이용한 얼굴 인식 알고리즘[4]이 소개되기도 한다.

본 논문에서는 K-L변환을 이용한 PCA(principle component analysis), Fisherface 알고리즘 및 Elastic graph matching 알고리즘에 대한 실험을 하고 이들을 효율적으로 결합한 얼굴 인식 방법을 제안하고자 한다.

I. K-L 변환을 이용한 얼굴인식

Fisherface 알고리즘은 LDA(linear discriminant analysis)를 사용하는 방법으로서 MDF(most discriminating feature) 방법이라고도 알려져 있다. 이는 사람마다 얼굴을 2개 이상씩 그룹을 지어 학습하는데 그 과정은 다음과 같다.

우선, K-L 변환을 사용하는 PCA 방법으로부터 압축된 영상 정보를 within-class scatter matrix의 rank 수 N-C차원에 해당하는 만큼의 특징벡터 $\Omega = \{y_n, n = 1, 2, \dots, N - C\}$ 를 받는다. N은 전체 얼굴 영상의 개수이고 C는 인식하고자 하는 사람의 수(클래스)이다. 그러면 서로 다른 K 개의 얼굴 영상을 사용한다고 가정하면, within-class scatter matrix와 between-class scatter matrix는 다음과 같다.

$$S_w = \sum_{m=1}^K \sum_{k=1}^K (\Omega_m^k - E\{\Omega_m\})(\Omega_m^k - E\{\Omega_m\})^T \quad (1)$$

$$S_b = \sum_{m=1}^C (E(\Omega_m) - E(\Omega))(E(\Omega_m) - E(\Omega))^T \quad (2)$$

여기서 $E(\Omega_m)$ 은 각각의 클래스 평균이고, $E(\Omega)$ 는 클래스의 전체 평균이다. 이 때 between-class scatter matrix와 within-class scatter matrix의 determinant 비를 최대화하기 위하여 $(S_w)^{-1}S_b$ 의 고유 벡터를 구한 후, 고유치 크기 순서대로 정렬한다[5]. 여기서 구한 고유 벡터 행렬로부터 특징 벡터 행렬 W_j 를 구한다.

최종적으로 얼굴을 인식하기 위해 입력된 얼굴영상 F_{in} 로부터 $\Psi_{in} = F_{in} - E(F)$ 를 계산하고, PCA 방법에서 구한 고유 벡터 행렬을 이용하여 특징 벡터 Ω_{in} 을 얻는다. 그리고 Fisherface 방법으로부터 구한 고유 벡터 행렬 U_j 를 이용하여 $U_j^T \Omega_{in}$ 의 모든 $j=1, 2, \dots, C-1$ 에 대해 특징 벡터를 계산한다. 그런 후 학습에 사용된 특징 벡터와 유클리드 거리를 비교함으로써 인식을 수행한다.

II. Elastic Graph Matching을 이용한 얼굴 인식

그래이 레벨 정보와 물체의 모양 정보를 동시에 이용하는 접근 방법 중의 하나를 소위 DLA(dynamic link architecture)라고 한다. 이러한 패턴 인식 방법은 신경 회로망에서와 같이 자기 스스로의 발전을 기본으로 하며 학습과정과 적응과정이 구성된다. 학습 과정에서, 얼굴 영역은 여러 개의 노드(node)와 에지(edge)로 나타내게 되는데 이는 여러 개의 오리엔테이션(orientation)과 크기 인자(scale)를 갖는 가버 필터의 응답으로부터 얻어진다. 즉, 가버 필터의 응답이 각 노드에서의 특징 벡터를 구성한다. 2차원 가버 필터[6]는 다음과 같이 나타낼 수 있다.

$$f(x, y, \theta_k, \lambda, \sigma_x, \sigma_y) = \exp\left[-\frac{1}{2}\left\{\frac{R_1^2}{\sigma_x^2} + \frac{R_2^2}{\sigma_y^2}\right\}\right] \times \exp\left[i\frac{2\pi R_1}{\lambda}\right] \quad (3)$$

$$R_1 = x \cos \theta_k + y \sin \theta_k \quad R_2 = -x \sin \theta_k + y \cos \theta_k$$

λ : wave length θ_k : orientation of the sinusoidal plane

$$\theta_k = \pi(k-1)/m, \quad k=1, 2, \dots, m$$

m : the number of orientation

σ_x, σ_y : standard deviation of gaussian envelope

적용 과정에서는 미리 만들어진 사람의 모델은 새로운 사람의 얼굴 영상과 비교되며 그 비용함수나

dissimilarity 함수가 최소화되도록 변형된다. 그런 후, 비용함수나 dissimilarity 함수의 분포(distribution)를 비교함으로써 얼굴을 인식할 수 있다.

즉, Elastic graph matching은 미리 만들어진 학습 모델의 그래프를 새로운 영상의 그래프와 비교하는 것으로부터 시작된다. 그림 1은 노드와 에지로 구성된 학습 모델의 예를 보여준다. 비교하고자 하는 두 그래프 사이의 거리(비용함수)는 각 노드와 에지가 가지고 있는 변형 정보로 구성된 특징 벡터들을 고려한 dissimilarity 함수에 의하여 구해질 수 있는데, 이는 각 노드들과 에지를 독립적으로 생각하여 측정할 수 있으며 그 식은 다음과 같다.

$$d(G, R) = \sum_{i=1}^{N_n} d_n(G_{n_i}, R_{n_i}) + \lambda \sum_{j=1}^{N_e} d_e(G_{e_j}, R_{e_j}) \quad (4)$$

여기에서 G_{n_i} 는 학습 모델 그리드(Grid) G의 i 번째 노드를 나타내며, R_{e_j} 는 입력된 영상 그리드의 j 번째 에지를 나타낸다. 그리고, N_n 과 N_e 는 각각 노드와 에지의 전체 개수이며 λ 는 그래프의 강도(stiffness)를 특징지어주는 가중치 인자이다.

얼굴을 인식하기 위한 매칭 알고리즘은 다음의 두 단계로 구성된다. 먼저 그래프를 변형시키지 않고 비교될 영상의 가장 적절한 위치를 찾는다. 이를 전체적 이동(Global Move)이라고 한다. 그런 다음 식 (4)를 최소화시키기 위하여 그래프를 변형시키는 작업이 이어지는데 이를 지역적 이동(Local Move)이라고 한다. 변형은 $d(G, R)$ 의 최소 값을 얻을 수 있는 장소를 현재 위치한 노드의 주위에서 찾음으로써 현재 노드 위치를 그 곳으로 대체한다. 이 과정은 각 노드마다 변형이 일어나지 않을 때까지 계속해서 수행되는데, 결국 이러한 과정을 수행한 후 계산된 비용함수를 비교함으로써 얼굴을 인식할 수 있다.

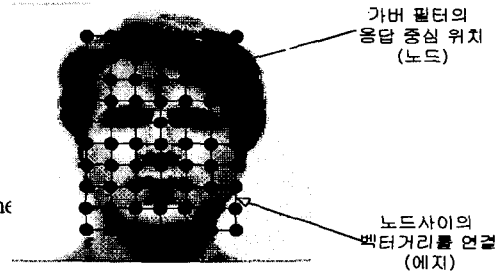


그림 1 얼굴 영역을 구성하는 노드와 에지의 예

III. 제안한 알고리즘

본 연구에서 제안하는 얼굴 인식 방법은 다음과 같이 크게 3단계로 나누어진다;

- (1) Gabor 웨이블릿 변환을 통한 그리드의 구성 및 조정 단계
- (2) Labeled Graph 벡터(LG 벡터)의 PCA 및 Fisherface 알고리즘 적용 단계
- (3) 특징 벡터의 매칭 알고리즘 적용 단계

그림 2는 제안하는 얼굴 인식 과정을 보여준다. 얼굴 영역의 그리드를 구성하기 위하여 가버 웨이블릿 변환을 한다. 가버 웨이블릿의 응답으로부터 얻어진 원래의 LG 벡터는 Fisherface(Principle Component Analysis 및 Linear Discriminant Analysis의 적용) 알고리즘을 거친다. 이 알고리즘은 LG 벡터의 차원을 감소로 인한 속도 향상과 더불어 같은 사람의 특징 벡터들을 밀접하게 연결시키는 성질을 가지고 있다.

PCA 및 Fisherface 알고리즘을 거쳐서 나온 특징 벡터는 데이터베이스에 저장되어 있는 얼굴 영상들의 특징 벡터와 비교한다. 이러한 매칭 과정에서 단순히 얼굴의 모양(shape)를 고려하지 않는 경우와 지역적 이동을 함으로써 얼굴의 모양을 고려하는 방법이 있는데 본 논문은 전자만을 실험하였다.

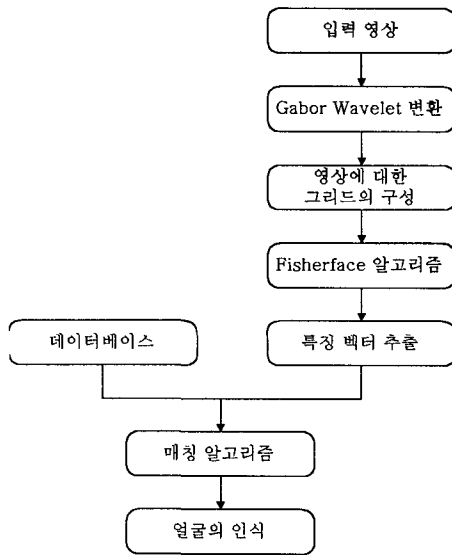


그림 2 제안하는 얼굴 인식 알고리즘

IV. 실험 및 결과

실험에 사용된 프로그램은 C-언어를 사용하여 펜티엄 프로세서 433MHz, 메모리 128MB의 PC에서 수행하였다. 실험에 사용된 데이터베이스는 인터넷을 통해 얻은 Yale대학 얼굴 영상을 사용하였으며, 이는 총 15명이고 각 사람 당 광원의 위치에 따른 조도의 변화, 얼굴 표정 변화, 안경 착용 여부에 따라 10장의 얼굴 영상으로 구성되었다. 본 논문에서는 원래 320X243 크기의 영상을 128X128 크기로 바꾸어 총 5가지의 얼굴 인식 방법을 실험하였다. 그림 2는 Yale대학 데이터베이스의 예이다.

표 1은 Leaving-one-out 방법으로 실험한 5가지의 얼굴 인식 알고리즘에 대한 인식률의 결과를 보여주고 있다. PCA 방법의 경우 고유 벡터의 개수는 전체 고유치 값의 95%를 점유하는 개수를 사용하였다. 그리고 Fisherface 방법의 경우 특징벡터로써의 고유 벡터 개수는 14개를 이용하였다. 또한 Elastic graph matching을 사용하는데 있어서 전체적 이동의 경우 일정한 위치를 지정해줌으로써 그 과정을 생략하였다. 즉, 일률적으로 150개의 얼굴 영상에 대해 좌표 값 (30, 24)의 위치를 첫 번째 노드 위치로 지정하였다. 또한 가로로 7개, 세로 10개의 노드로 얼굴영상 그리드를 구성했으며 노드 간의 거리는 11픽셀로 지정하였다. 또한, 그래프의 강도를 특징지어주는 λ 값은 3×10^{-3} 이며 가버 필터의 오리엔테이션과 크기인자는 각각 4와 3인 것을 사용하였다.

표 2는 Hold-out 방법으로 실험한 것으로서 각 사람마다 3장의 얼굴 영상으로 학습을 한 후 나머지 영상들을 이용하여 얼굴을 인식한 결과이다.

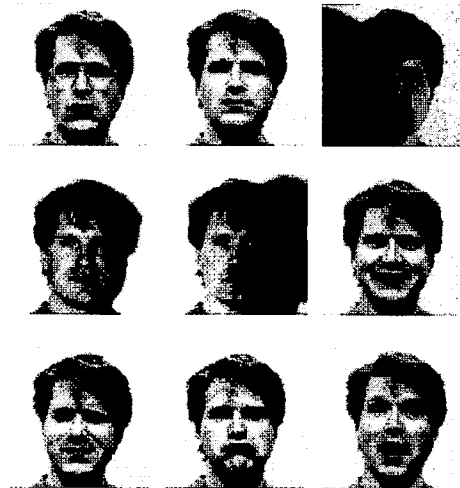


그림 3 Yale대학 데이터베이스의 예

표 1 인식을 실험 결과 (Leaving-one-out 방법)

인식 방법	인식률(%)
PCA	79.3
Fisherface	99.3
Elastic graph matching	99.3
Elastic graph matching + PCA	88.7
Elastic graph matching + Fisherface	99.3

표 2 인식을 실험 결과 (Hold-out 방법)

학습에 사용된 영상	PCA	Fisherface	Elastic graph matching + PCA	Elastic graph matching + Fisherface
1, 2, 3 번째	81.9	93.3	89.5	94.3
3, 4, 5 번째	75.2	85.7	85.7	94.3
5, 6, 7 번째	82.9	87.6	86.7	93.3
7, 8, 9 번째	72.4	77.1	81.0	87.6
1, 9, 10번째	70.5	72.4	75.2	84.8
인식률 (평균)	76.6	83.2	83.6	90.9

실험 결과 본 논문에서 제안한 elastic graph matching과 Fisherface방법을 결합한 얼굴 인식 알고리즘의 인식률이 가장 높았으며, 학습과정을 제외하고 얼굴 영상 하나를 인식하는데 걸리는 시간은 수초 정도가 걸렸다. 특히, 표1에서 elastic graph matching의 경우 한 영상을 인식하는데 거의 20분 이상이 소요되었다.

V. 결론 및 향후 방향

본 연구에서는 Elastic graph matching과 Fisherface 알고리즘을 결합함으로써 기존 얼굴 인식에 있어서 인식률을 떨어뜨리는 주요 요인(광원의 위치에 따른 조도의 변화, 얼굴 표정 변화, 안경 착용 여부 등)에 강인한 얼굴 인식 알고리즘을 제시하였다. 또한 elastic graph matching 과정에서 Fisherface 알고리즘의 적용으로 인한 LG 벡터의 차원 감소로 인하여 연산 처리 시간을 단축할 수 있다는 장점을 실험을 통해 알 수 있었다. 다시 말하자면, K-L 변환만 이용한 얼굴 인식 알고리즘의 경우 인식 속도는 빠르지만, 영상의 변화에 대한 인식률은 가버 필터를 이용한 얼굴 인식 방법보다 상대적으로 떨어짐을 알 수 있었다.

향 후 연구해야 할 과제로써 먼저 전체적인 이동을 고려한 elastic graph matching 방법과 얼굴 영상의 효율적인 학습 모델 구성에 관한 연구가 필요하다. 또한 본 논문에서 제안한 알고리즘의 매칭 과정에서 얼굴의 모양(shape)을 고려한 지역적 이동을 함으로써 인식률을 높일 수 있으리라 사려된다.

참고문헌

- [1] A. P. Pentland and M. A. Turk, "Face recognition using eigenfaces," in *Proc. Int. Conf. on Pattern Recognition*, pp.586-591, 1994.
- [2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.
- [3] B. Duc, S. Fisher, and J. Bigun, "Face authentication with Gabor information on deformable graphs," *IEEE Trans. on Image Processing*, vol. 8, no. 4, April 1999.
- [4] 김종규, 장주석, 김영일, "독립 성분 분석을 이용한 얼굴인식," 전자공학회논문지, 제35권, C편, 제10호, 1998, 10.
- [5] D.L. Swets, and J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831-836, August 1996.
- [6] B.S. Manjunath, and W.Y. Ma, "Texture features for browsing and retrieval of image data" *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837-842, August 1996.