

Eigenface를 이용한 얼굴인식에서의 영상등록 오차 보정

정회원 문지혜*, 이병욱**

Registration Error Compensation for Face Recognition Using Eigenface

Ji-Hye Moon*, Byung-Uk Lee** *Regular Member*

요약

얼굴 인식에서는 입력 영상에서 얼굴을 검출한 후에 데이터베이스의 영상과 위치와 크기를 일치시키는 등록 과정이 필요하다. 본 논문에서는 영상의 등록 과정에서 발생하는 얼굴영상의 이동, 회전, 혹은 크기의 차이를 eigenspace에서 보정하는 알고리즘을 제안하였다. 이를 위하여 얼굴 영상의 수직, 수평 이동, 회전, 크기 변환 등의 등록오차를 선형보간 행렬로 근사하였다. 각 변환행렬을 사용하여 등록오차에 따른 미분계수를 eigenspace에서 구하면 subpixel 단위의 등록 오차를 보정할 수 있다. 제안된 방법은 공간 영역에서 오차를 보정하는 것보다 계산량이 훨씬 더 적다. 오차 보정 후 얼굴 인식률이 크게 향상되는 것을 실험으로 확인하였다.

Key Words : Eigenface, Subpixel, Database

ABSTRACT

The first step of face recognition is to align an input face picture with database images. We propose a new algorithm of removing registration error in eigenspace. Our algorithm can correct for translation, rotation and scale changes. Linear matrix modeling of registration error enables us to compensate for subpixel errors in eigenspace. After calculating derivative of a weighting vector in eigenspace we can obtain the amount of translation or rotation without time consuming search. We verify that the correction enhances the recognition rate dramatically.

1. 서론

얼굴 인식은 영상인식 분야에서 가장 주목 받는 분야 중에 하나이다. 이러한 관심 속에 지난 수년 동안 얼굴 인식에 대한 연구가 활발히 진행 되었으며, 여러 얼굴 인식의 기법과 체계적인 평가 방법들이 제시되었다^[1].

일반적으로 얼굴 인식의 문제는 정지 혹은 움직이는 영상으로부터 얼굴을 찾아내는 얼굴 검출 (face

detection)과 얼굴이 검출 되었을 경우 검출된 얼굴이 누구의 얼굴인지를 가려내는 얼굴의 분류로 나눌 수 있다^[2].

그러나 얼굴 영상이 카메라의 각도, 안경이나 수염 등의 존재, 얼굴의 표정과 조명의 밝기 등에 따라 많은 변수를 갖고 있기 때문에 얼굴을 정확하게 검출하는 것은 매우 어려운 문제이다. 이러한 문제

* 이화여자대학교 정보통신학과 영상 처리 연구실 (jihye.moon@magnachip.com),

** 이화여자대학교 정보통신학과 영상 처리 연구실 (bulee@ewha.ac.kr)

논문번호 : 040171-0507, 접수일자 : 2004년 5월 7일

※본 연구는 대학 IT연구센터 육성지원사업의 연구결과로 수행되었음.

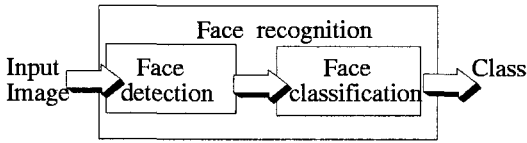


그림 1. 얼굴 인식 시스템 개념도

때문에 대부분의 연구는 얼굴 검출, 얼굴 분할, 얼굴의 배치까지 모두 정확하게 이루어 졌다는 가정 하에 얼굴영상의 분류 부분에 집중되어 왔다³⁾.

시험영상의 얼굴의 위치와 크기가 데이터베이스 얼굴의 좌표와 다른 경우 인식률이 현저하게 낮아진다. 본 논문에서는 영상간의 위치와 크기가 다른 영상등록오차를 eigenface에서 보정하는 방법을 제안하였다. 본 방법은 계산량이 작으면서도 인식률을 크게 향상시키는 것을 실험으로 확인하였다.

Eigenface를 사용한 얼굴인식은 dimension을 크게 줄이면서도 높은 성능을 갖기 때문에 최근 많이 사용되는 방법이다⁴⁾.

II. Eigenface를 이용한 얼굴 인식

Principal component analysis(PCA)는 상관관계가 높은 데이터를 선형 변환을 통해 redundancy를 제거하는 기법이다. 이러한 PCA 변환을 사용하여 얼굴인식을 하는 것은 1991년 Pentland⁵⁾에 의해 제안되었다. PCA에서 data set의 공분산 행렬의 eigenvector가 데이터 사이의 correlation을 제거하는 기저벡터가 된다. 특별히 얼굴영상 data를 이용하여 구한 eigenvector를 eigenface라고 하며 eigenface의 가중치 계수를 나타내는 공간을 eigenspace라고 한다. 각 얼굴영상은 eigenface의 선형결합으로 완벽하게 복원될 수 있다. Training set로부터 구할 수 있는 eigenface의 갯수는 training 얼굴영상의 갯수와 같다. 여기에서 가장 큰 eigenvalue에 해당하는 eigenface는 얼굴영상의 training set에서 가장 큰 변화를 갖는 기저벡터이며, 가장 작은 eigenvalue에 해당하는 eigenface는 얼굴영상에 가장 작은 변화를 나타내므로 얼굴영상은 상위 몇 개의 eigenface를 사용하여 구별할 수 있다. 따라서 얼굴영상의 수보다 적은 수의 eigenface를 사용하여 얼굴인식을 하는 방법으로서 매우 효율적이다.

얼굴영상으로부터 eigenface를 구하고 가중치 벡터를 얻는 과정을 식으로 나타내면 다음과 같다. 세로의 화소 수가 Height 이고 가로 크기가 Width 인 2차원의 얼굴 영상을 T_i 라고 할 때, 이 영상을

column stacking한 $N \times 1$ 의 벡터를 t_i 라고 하자. 여기에서 $N = \text{Height} \times \text{Width}$ 가 된다. Training set의 평균 영상을 m 으로 표시하면, 각 training 영상에서 평균을 뺀 영상은 $\phi_i = t_i - m$ 으로 나타낸다. 이때 공분산 행렬 C 는

$$C = \frac{1}{M} \sum_{i=1}^M \phi_i \phi_i^T$$

이며, 이 공분산 행렬의 eigenvector e_i 를 eigenvalue의 크기에 따라 정렬한 것이 $V = [e_1 \dots e_M]$ 이다. 이렇게 구해진 eigenspace에 평균을 뺀 얼굴영상을 입력하여 다음과 같이 투영시키면,

$$w = V^T \phi$$

와 같이 eigenface 영상 인식에 사용되는 가중치 벡터 w 를 구할 수 있다.

III. Eigenspace에서 영상 등록 오차 보정

이 장에서는 영상 등록 과정 중에 생기는 수직 및 수평 이동, 회전, 크기의 변환에 대한 오차를 eigenspace에서 구하고, subpixel 단위의 오차를 보상하는 알고리즘에 대해 제안한다. 일반적으로 얼굴 인식에서 얼굴영상을 매우 작은 크기로 정규화 시켜 사용하는 점에 착안하면 대부분의 영상등록 오차는 subpixel 단위로 나타나게 된다. 따라서 영상 등록 오차를 subpixel 단위로 가정한 것은 매우 합리적이다.

3.1 얼굴 영상의 등록 오차의 정의

본 논문에서는 얼굴 인식을 위해 시스템에 들어 오게 되는 시험 영상을 column stacking 시킨 $N \times 1$ 크기의 벡터를 x , 이상적인 training set와의 등록 오차를 행렬 A 로 나타낸다. 즉 영상분류 시스템으로 입력되는 변형된 영상은 Ax 가 된다. 여기에서 다루는 영상의 변형을 수직 및 수평이동, 회전, 크기로 인한 변화로 한정하면, 각 변환 행렬을 영상의 가로와 세로 방향의 선형 보간으로 모델링할 수 있다. 여기서 영상등록 오차를 한 화소 이하로 제한하면 각 변화를 선형 보간 행렬 A 로 나타낼 수 있다.

얼굴영상의 등록과정 중 오차가 발생하였을 때 분류 시스템으로 입력되는 시험 영상의 가중치 벡터는 (식 1)과 같다.

$$\mathbf{w}_{test} = \mathbf{V}^T(\mathbf{A}\mathbf{x} - \mathbf{m}) \quad (\text{식 1})$$

여기에서 영상의 등록 오차를 가진 가중치 벡터와 오차가 없는 가중치 벡터 사이의 차이를 $\Delta\mathbf{w}$ 라고 하면, $\Delta\mathbf{w}$ 는 (식 2)와 같다.

$$\begin{aligned} \Delta\mathbf{w} &= \mathbf{w}_{test} - \mathbf{w} \\ &= \mathbf{V}^T\mathbf{A}\mathbf{x} - \mathbf{V}^T\mathbf{x} \\ &= \mathbf{V}^T(\mathbf{A} - \mathbf{I})\mathbf{x} \end{aligned} \quad (\text{식 2})$$

위의 식에서 볼 수 있는 바와 같이 영상변형 행렬 \mathbf{A} 가 identity matrix로부터 차이가 나는 정도에 따라서 가중치 벡터가 변하게 된다.

3.2 영상 등록오차의 선형 근사화

미분계수는 함수 $f(x)$ 가 x 의 매우 미세한 변화에 대하여 선형적으로 변하는 기울기를 의미한다. 이와 같은 개념을 eigenspace의 가중치 벡터에 적용시켜 보면 다음과 같이 정리할 수 있다. 영상이 수평 방향으로 h ($|h| < 1$) 만큼 이동한 경우의 변환을 $\mathbf{A}(h)$ 로 표시하면 이 변환으로 인하여 발생하는 영상 vector \mathbf{x} 의 eigenspace에서 변화를 $\mathbf{w}'(x)$ 는 다음과 같다

$$\begin{aligned} \mathbf{w}'(x) &= \lim_{h \rightarrow 0} \frac{\Delta\mathbf{w}}{h} \\ &= \lim_{h \rightarrow 0} \frac{\mathbf{V}^T(\mathbf{A}(h)\mathbf{x} - \mathbf{m}) - \mathbf{V}^T(\mathbf{x} - \mathbf{m})}{h} \end{aligned} \quad (\text{식 3})$$

따라서 h 가 작은 경우 이 미분 계수를 사용하여 선형 근사화 하면,

$\Delta\mathbf{w} \approx h\mathbf{w}'(x) = \mathbf{V}^T(\mathbf{A}(h)\mathbf{x} - \mathbf{m}) - \mathbf{V}^T(\mathbf{x} - \mathbf{m})$, for $h \approx 0$ 가 된다. 즉 변화량이 작은 경우, 즉 영상의 subpixel 변화에 대해 가중치 벡터의 차이는 가중치 벡터의 미분계수를 사용하여 선형 근사화 될 수 있다. 수직 및 수평이동, 회전, 크기 변화에 대한 영상 변형에 선형 보간을 사용하면 가중치 벡터의 미분계수는 위의 식을 사용하여 구할 수 있다.

영상을 수직방향으로 h 만큼 이동시키는 선형 변환행렬로 나타내면 (식 4)와 같다.

$$\mathbf{A}_v(h) = \mathbf{I} + h\mathbf{A}_v^d$$

$$\mathbf{A}_v^d = \begin{bmatrix} \mathbf{H} & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{H} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{O} \\ \mathbf{O} & \cdots & \mathbf{O} & \mathbf{H} \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -1 \end{bmatrix} \quad (\text{식 4})$$

위의 식에서 $h = 1$ 이면 영상이 수직방향으로 한 화소 아래로 이동한 것이다. 여기에서 아래 첨자 v 는 수직 (vertical)을 의미하며 위첨자 d 는 변화량 (delta)을 의미한다. (식 4)에서 확인할 수 있는 바와 같이 h 가 0에 가까워질수록 \mathbf{A}_v 는 단위행렬에 가까워진다. 또한 (식 3)에 적용시켜 보면 $|h| < 1$ 일 때 h 에 관계없이 아래와 같이 \mathbf{w}' 을 구할 수 있다.

$$\begin{aligned} \mathbf{w}'(x) &= \lim_{h \rightarrow 0} \frac{\mathbf{V}^T(\mathbf{A}_v(h) - \mathbf{I})\mathbf{x}}{h} \\ &= \mathbf{V}^T \lim_{h \rightarrow 0} \frac{(\mathbf{I} + h\mathbf{A}_v^d) - \mathbf{I}}{h} \mathbf{x} \\ &= \mathbf{V}^T \mathbf{A}_v^d \mathbf{x} \end{aligned} \quad (\text{식 5})$$

수직 이동으로 인한 영상의 변화를 선형 보간으로 모델링하였을 때, 미분계수는 $|h| < 1$ 구간에서 상수가 된다. 따라서 이동으로 인한 변형을 정확하게 보정할 수 있다. 기타 수평이동, 회전, 크기 변화도 비슷한 방법으로 선형변환으로 근사화 할 수 있다. 회전이나 크기 변화는 화소의 위치에 따라서 수평과 수직으로 이동하는 양이 변하는 것으로 모델링할 수 있다.

3.3 이동, 회전, 크기변화를 모두 고려한 경우

3.2절에서는 이동, 회전, 크기 변화 중 한 가지만 일어났을 경우에 대해서만 $\Delta\mathbf{w}$ 를 고려하였다. 하지만 실제 등록 과정에서 일어날 수 있는 변화는 이동, 회전, 크기의 변화는 모든 변환행렬의 곱으로 나타낼 수 있다. 이러한 변환을 거친 얼굴 영상의 경우 $\Delta\mathbf{w}$ 는 (식 6)과 같이 나타낼 수 있다.

$$\begin{aligned} \Delta\mathbf{w} &= \mathbf{V}^T(\mathbf{A}_h\mathbf{A}_v\mathbf{A}_r\mathbf{A}_s)\mathbf{x} - \mathbf{V}^T\mathbf{x} \\ &= \mathbf{V}^T(\mathbf{A}_h\mathbf{A}_v\mathbf{A}_r\mathbf{A}_s - \mathbf{I})\mathbf{x} \end{aligned} \quad (\text{식 6})$$

\mathbf{A}_h : 수평방향 이동변환 \mathbf{A}_v : 수직방향 이동변환
 \mathbf{A}_r : 회전 변환 \mathbf{A}_s : 크기변환

위에서 아래첨자 h 는 수평(horizontal), r 은 회전(rotation), s 는 크기(scale factor)를 나타낸다. (식

6)의 각 변환 행렬은 원래 영상에서 미세한 변화를 나타내므로 identity matrix에서 약간 벗어난 것으로 생각할 수 있다. 따라서 아래의 (식 7)과 같이 나타낼 수 있다.

$$\begin{aligned}
 & \mathbf{A}_b \mathbf{A}_v \mathbf{A}_r \mathbf{A}_s \\
 &= (\mathbf{I} + \mathbf{A}_b^d)(\mathbf{I} + \mathbf{A}_v^d)(\mathbf{I} + \mathbf{A}_r^d)(\mathbf{I} + \mathbf{A}_s^d) \\
 &= \mathbf{I} + \mathbf{A}_b^d + \mathbf{A}_v^d + \mathbf{A}_r^d + \mathbf{A}_s^d + \mathbf{A}_b^d \mathbf{A}_v^d + \mathbf{A}_v^d \mathbf{A}_r^d + \mathbf{A}_r^d \mathbf{A}_s^d + \\
 & \quad \dots + \mathbf{A}_b^d \mathbf{A}_v^d \mathbf{A}_r^d \mathbf{A}_s^d \\
 & \approx \mathbf{I} + \mathbf{A}_b^d + \mathbf{A}_v^d + \mathbf{A}_r^d + \mathbf{A}_s^d
 \end{aligned}
 \tag{식 7}$$

위의 식에서 각 변환의 정도가 아주 작은 경우에 2 차항 이상은 전부 무시하고 1 차항 만 남겼다. (식 7)의 결과를 이용하여 (식 8)을 다음과 같이 정리할 수 있다.

$$\begin{aligned}
 \Delta \mathbf{w} &= \mathbf{V}^T (\mathbf{A}_b \mathbf{A}_v \mathbf{A}_r \mathbf{A}_s - \mathbf{I}) \mathbf{x} \\
 &\approx \mathbf{V}^T ((\mathbf{I} + \mathbf{A}_b^d + \mathbf{A}_v^d + \mathbf{A}_r^d + \mathbf{A}_s^d) - \mathbf{I}) \mathbf{x} \\
 &= \mathbf{V}^T (\mathbf{A}_b^d + \mathbf{A}_v^d + \mathbf{A}_r^d + \mathbf{A}_s^d) \mathbf{x} \\
 &= \mathbf{V}^T (\mathbf{A}_b - \mathbf{I}) \mathbf{x} + \mathbf{V}^T (\mathbf{A}_v - \mathbf{I}) \mathbf{x} \\
 & \quad + \mathbf{V}^T (\mathbf{A}_r - \mathbf{I}) \mathbf{x} + \mathbf{V}^T (\mathbf{A}_s - \mathbf{I}) \mathbf{x} \\
 &= \Delta \mathbf{w}_b + \Delta \mathbf{w}_v + \Delta \mathbf{w}_r + \Delta \mathbf{w}_s
 \end{aligned}
 \tag{식 8}$$

따라서, 영상에서 이동, 회전, 크기 변환이 모두 일어난 경우에도 각 변환으로 인한 변화량의 합으로 간주할 수 있다.

IV. 실험 과정 및 실험 결과

4.1 Database

본 실험에서 사용한 얼굴 영상은 MIT Media Laboratory에서 제공하는 얼굴영상을 사용하였다^[6]. 남자 10명에 대해 한 사람당 다른 조명에서 찍은 사진 3개씩을 포함하여 training set을 구성하였으며 얼굴영상을 16x16 크기로 정규화 시켰다. 얼굴 영상을 이동하거나 회전할 때에 영상 외부 화소 값의 영향을 없애기 위하여 얼굴 영역을 제외한 모든 부분의 밝기를 128로 바꾸어 실험하였다.

4.2 실험 결과

본 논문에서 제안한 영상의 등록과정 중에 발생하는 오차를 보정하기 위한 실험은 정규화를 거친 training set의 영상에 수직 및 수평이동, 회전, 크기 변환한 영상을 시험영상으로 하였다. Training set은

eigenface를 구할 때 사용되며 얼굴 검색의 database를 구성한다. 변형된 시험 영상을 사용하여 변형전의 database의 변형전의 영상을 찾을 수 있는지 실험하였다. 등록오차를 가진 시험영상에 대해 본 논문의 오차보정 알고리즘을 적용하였을 때와 적용하지 않았을 때 인식률을 비교하였다. 본 논문에서는 영상 등록 오차 보정에 관한 연구이므로, 영상의 분류과정에서 Euclidian distance를 사용하였으며 제일 가까운 class로 분류하였다. Support Vector Machine을 사용하면 인식률을 향상시킬 수 있다.

미리 정의된 모든 class에 대한 weight 벡터와 시험 영상의 weight 벡터의 차이 $\Delta \mathbf{w}$ 는 (식 9)를 사용하여 미분계수 \mathbf{w}' 로 근사화할 수 있다.

$$\begin{aligned}
 h &= \frac{\Delta \mathbf{w} \cdot \mathbf{w}'}{\|\mathbf{w}'\|^2} \\
 \Delta \mathbf{w} &\approx h \mathbf{w}'
 \end{aligned}
 \tag{식 9}$$

실험에서 이동, 회전, 스케일 변화로 인한 가중치 벡터의 차이 $\Delta \mathbf{w}$ 를 보정하는 순서는 영상의 전체가 움직여서 오차가 큰 수직, 수평이동에 대해 먼저 오차를 보정한 후, 크기와 회전 변환의 차례로 정하였다.

본 논문에서는 제안한 알고리즘은 이동의 한계를 subpixel로 가정하였으므로, 여기서 고려하는 최대 이동 범위는 수직, 수평방향으로 각각 -1~1이다. 본 실험에서 영상의 최외각에서 한 화소 이하의 이동 범위를 갖기 위해서는 크기 변환의 경우 원래 영상과의 크기 비에 대해 0.875~1.125, 회전변환의 경우 -7~7 도의 범위를 가질 수 있다.

[그림 2]는 각각 이동, 회전, 크기변화 정도를 달리하며 오차보정 알고리즘을 적용하기 전과 후의 correct recognition rate (CRR)을 비교한 것이다. 실험 (a)는 영상의 중심을 기준으로 7 도 회전, 크기를 1.125배로 한 후 수직방향으로 1 만큼 이동한 상태에서 수평방향의 이동 정도에 따라 얼굴인식률을 비교한 것이다. 오차 보정전의 얼굴인식률은 수평이동 정도가 커질수록 급격히 떨어지지만 본 논문에서 제안한 방법으로 오차를 보정한 후 얼굴인식률이 수평방향의 이동에 크게 영향을 받지 않고 모두 CRR이 0.9이상 되는 것을 볼 수 있다. 실험 (b)는 회전각을 달리하여 영상을 변화시킨 후 1.125 배 크기변환과 각각 1 씩 수직, 수평방향으로 이동 변환시킨 영상에 대해 CRR을 비교한 것이다. 이 경우 회전각에 크게 상관없이 오차보정 전의 CRR

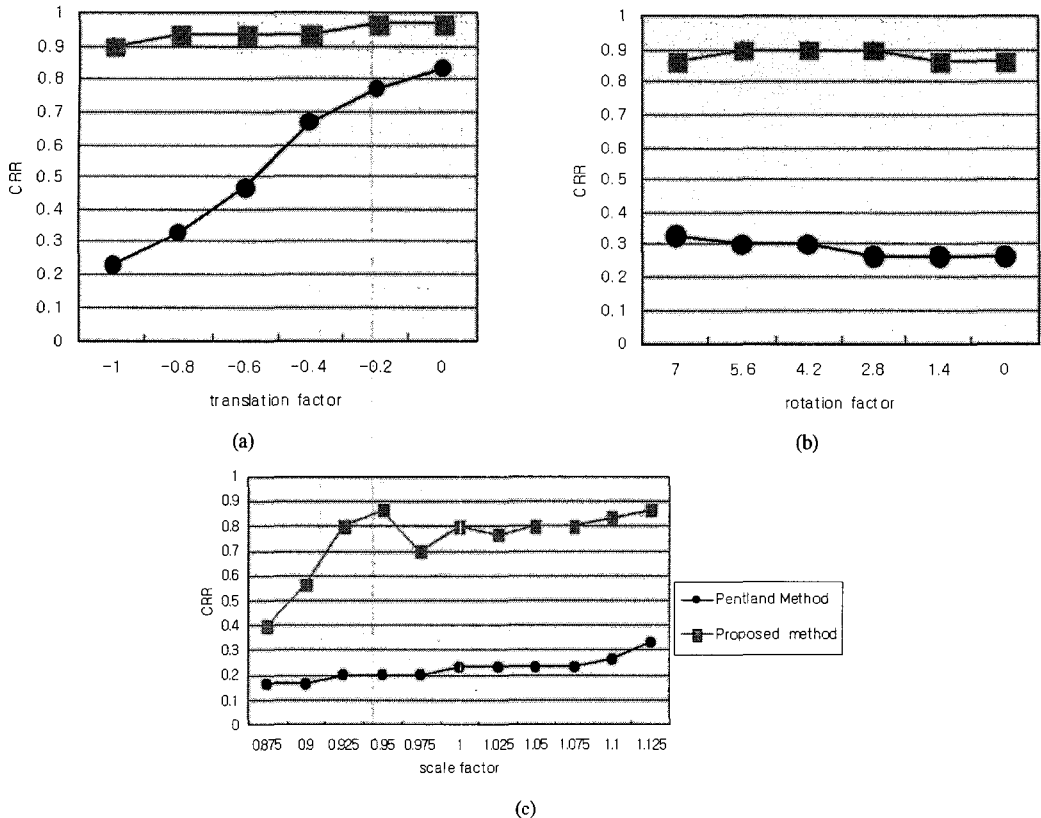


그림 2. 오차보정 알고리즘을 적용하였을 때와 적용하지 않았을 경우의 얼굴인식률 (CRR) 비교 (a) 수평이동 정도는 1, 회전각은 7, 크기비율은 1.125일 때 수직이동 정도를 달리하였을 경우 (b) 수직 수평방향으로 이동 정도는 각각 1, 크기 비는 1.125일 때 회전각을 달리하였을 경우 (c) 수직 수평방향 이동 정도는 각각 1, 회전각은 7일 때 크기변환 비를 달리하였을 경우.

이 모두 0.4미만이지만 오차를 보정 한 후 CRR 은 모두 0.85이상으로 향상된 것을 볼 수 있다. 실험 (c)는 영상을 7도 회전시킨 후 크기 변환을 달리 한 영상을 수직, 수평방향으로 1씩 이동하였을 경우 CRR 을 나타낸 것이다. 이 경우에도 오차 보정전 과 오차보정 후의 CRR이 두 배 이상 향상된 것을 볼 수 있다.

[그림 2]의 실험결과를 보면 제안한 방법으로 영상등록 오차를 완벽하게 수정하지는 못한다. 이러한 이유 중 하나는 영상의 이동, 회전, 크기에 대해 상당한 변환을 하였기 때문에 모든 변환을 거친 영상은 결국 그 오차가 한 화소를 넘어가게 되어 앞에서 가장한 subpixel 범위를 넘어가기 때문이다. 한 예로 (16,16)에 위치한 화소를 scale factor 0.9로 변환시킨 후에 7 도 회전 변환하고 나서 수직, 수평 방향으로 모두 -1 화소씩 이동한 경우를 생각해 보자. 크기변환으로 인해 (15.2, 15.2)의 위치로 이동하게 되며, 회전변환으로 인해 (16, 14)로 다시 수직, 수평 이동변환으로 인해 (15, 13)의 위치로 이

동하게 된다. 결국 영상의 이동범위는 앞에서 가장한 범위를 넘어 제안한 방법으로 영상등록오차를 보정할 수 없게 된다. 이러한 이유로 [그림 2]의 (c)에서 scale factor가 1보다 작을 때 오차보정 후 CRR이 다른 것에 비해 현저히 낮은 것을 볼 수 있다. 영상등록 오차를 완벽하게 보정하지 못한 또 다른 이유는 w_v, w_h, w_z, w_r 등의 벡터 성분이 서로 수직이 아니기 때문이다. 제안한 방법에서는 수직, 수평 방향의 오차를 먼저 보정하므로, 수직, 수평이동 변환과 회전, 크기변환이 동시에 일어난 경우, 회전과 크기변환에 의한 오차가 수평이나 수직이동 성분과 수직이 아니어서 그 성분에 영향을 준다. 따라서 회전과 크기변환에 관한 오차를 완벽하게 수정할 수 없게 된다.

본 논문에서 제안한 방법으로 실제 영상에 적용하여 실험해 보았다. 실험에 사용한 얼굴영상은 MIT database에서 제공하는 얼굴과 카메라의 zoom 을 사용하여 얼굴영상의 크기가 달라진 data set와 머리를 옆으로 돌려서 찍은 data set을 사용하였다.

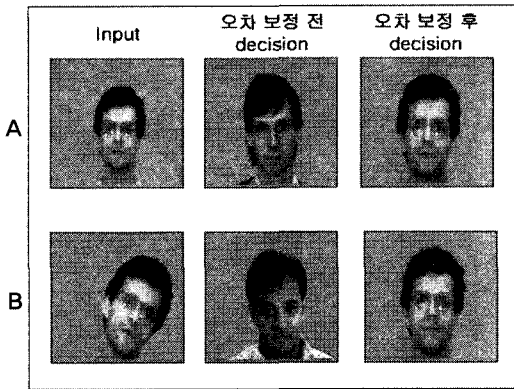


그림 3. 실제 영상을 사용하여 실험한 결과(A: scale 변화, B: 각도 변화)

[그림 3]에서 확인할 수 있듯이 실제 영상을 가지고 실험하였을 경우에도 얼굴의 크기와 회전에 의한 차이를 보정하여 인식 능력을 개선시킬 수 있다.

제안한 방법의 가장 큰 단점은 training 영상과 새로 들어온 시험 영상간의 차이가 모두 등록 과정 중에 생기는 영상의 변형으로 생긴다고 가정하는 것이다. 만약 시험 영상으로 위치나 크기는 같지만 조명의 밝기가 다른 사진이 들어오게 되더라도 제안한 방법에 의해 오차를 고치게 된다. 이런 경우 오차를 보정하는 것이 오히려 얼굴 인식에 방해가 될 수도 있다.

V. 결론

본 논문에서는 상관관계가 높은 training set를 영상영역이 아니고 eigenspace 영역에서 효율적으로 얼굴 인식하는 방법을 사용하였다.

기존의 많은 얼굴 인식 방법은 얼굴 영상의 위치, 크기 등이 database와 완벽하게 맞추어 졌다는 전제 하에 이루어지고 있다. 본 논문에서는 영상의 등록 과정 중 영상의 배치가 잘못 이루어진 경우에 대해 eigenspace에서 그 오차를 보정하는 알고리즘을 제안하였다. IV장의 실험을 통해 알 수 있듯이 영상이 이동, 회전, 크기 변환이 발생한 경우 대부분의 경우에 잘못된 인식을 한다. 하지만 eigenspace에서 가중치 벡터의 차이를 선형 근사화 하여 영상의 변환으로 인한 오차를 수정한 후 인식률이 크게 향상

된 것을 확인 할 수 있다. Eigenspace에서의 오차 수정 방법은 영상영역에서 이루어지는 것보다 계산량을 많이 감소시킬 수 있어서 더욱 의미가 있다.

하지만 지금까지 연구의 단점은 training 영상과 새로 들어온 영상간의 차이가 모두 등록 과정 중에 생기는 영상의 변형으로 생긴다고 가정하는 것이다. 향후 연구 과제로 조명의 변화 등을 고려한 얼굴인식 방법을 함께 적용한다면 본 논문에서 제안하는 알고리즘을 보다 더 안정적으로 활용할 수 있을 것이다.

참고 문헌

- [1] W. Zhao, R. Chellappa, A. Rosenfeld, and J. Phillips, "Face Recognition: A Literature Survey," Technical Report, CS-TR4167, Univ. of Maryland, 2000.
- [2] M.-H. Yang, D.J. Kriegman, and N. Ahuja, "Detecting Faces in Images: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 1, pp. 34-58, Jan 2002.
- [3] M. Kirby, and L. Sirovich, "Application of the Karhunen-Love Procedure for the Characterization of Human Faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 1, pp. 103-108, Jan 1990.
- [4] B.K. Gunturk, A.U. Batur, Y. Altunbasak, M.H. Hayes, III, and R.M. Mersereau, "Eigenface-domain super-resolution for face recognition," *IEEE Transactions on Image Processing*, Vol. 12, No. 5, pp. 597-606, May 2003.
- [5] M. Turk and A. Pentland, "Face Recognition using Eigenface," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 586-591, Maui, Hawaii, May 1991.
- [6] ftp://whitechapel.media.mit.edu/pub/images/Media_laboratory, Massachusetts Institute of Technology.

문 지 혜 (Ji-Hye Moon)

정회원

2002년 2월 이화여자대학교 정보통신학과 학사

2004년 2월 이화여자대학교 정보통신학과 석사

2004년 3월~현재 MagnaChip 반도체 CIS개발팀
연구원

<관심분야> 영상처리, 패턴인식

이 병 옥 (Byung-Uk Lee)

정회원

이화여자대학교 정보통신학과 교수

한국통신학회 논문지 제 22권 6호 참조

<관심분야> 영상인식, 영상처리