

J. Natural Sci., Vol. 9. 2002
Sangmyung University

얼굴인식을 통한 웹기반 인증 시스템 설계

Web based Authentication System Design
of Face Recognition

배 경 율

Kyoung Yul Bae

2002

상명대학교 자연과학연구소

The Institute of Natural Science
SangMyung University

얼굴 인식을 통한 웹 기반 인증 시스템설계

배 경 율

상명대학교 자연과학대학 소프트웨어학부

Web Based Authentication System Design of Face Recognition

Julius Kyoung Yul Bae

Division of Computer Software, Sangmyung University, 7 Hongji-Dong,
Chongro-Ku, Seoul, 100-743, Korea

초 록

본 논문의 얼굴 인식 기술은 'Eigenfaces' 알고리즘을 기초로 하며, 다차원 얼굴 공간에 얼굴의 고유 특징을 매핑한 'Eigenfaces'를 생성하기 위해 주성분분석법(PCA, Principal Component Analysis)을 사용하였다. 현재까지 얼굴인식의 알고리즘으로는 PCA나 독립 성분분석법(ICA; Independent Component Analysis), 선형/피셔 판별 분석법(LDA; Linear/Fisher Discriminant Analysis) 방법 등이 소개되어 있으나, 어느 한 가지만으로는 만족할 만한 인식률을 얻기 힘들므로 본 논문에서는 PCA 방법과 'Eigenfaces' 알고리즘을 활용한 웹 기반의 개인 인증 모델을 설계하고, 얼굴 인식을 웹에 적용하기 위해서 고려해야 할 사항들을 기술하였다.

제 1 장. 서 론

제 1 절. 연구 배경

최근 세계 전역에서 발생하고 있는 테러사건을 계기로 보안에 대한 관심이 급증함과 더불어 얼굴 인식에 대한 개발이 활기를 띠고 있는 가운데 공항이나 인구 밀집 지역에서의 얼굴 인식 시스템은 테러범이나 수배자의 색출해줌으로서 보안 감시 능력을 향상시킬 수 있는 새로운 보안 대책으로 활용되고 있다. 그러나 얼굴 인식은 복잡한 알고리즘과 구현상의 어려움으로 인해 현재 극히 일부 업체에서만 얼굴 인식 시스템을 상용화하고 있다. 본 논문에서는 얼굴 인식 시스템을 활용하여 웹 기반에서 개인 인증에 대한 여러 제약조건을 극복할 수 있는 시스템 설계에 주안점을 두고 있다.

제 2 절. 얼굴 인식 관련 이슈

얼굴 인식의 핵심은 인식하려는 인물 이미지나 비디오 세그먼트를 데이터베이스에 이미 저장되어있는 이미지와 비교해서 근접한 이미지가 존재하느냐에 달려있다. 이러한 이미지 비교를 위해서 현재 다양한 연구가 이루어지고 있으며, 얼굴 인식의 효율성을 높이기 위한 이슈들은 다음과 같다.

- 얼굴 인지(face perception)가 전체론적 해석 또는 특징 해석의 결과인가?
- 안면 특징들의 중요성에 순서를 매기기
(*Ranking of significance of facial features*)
- 캐리커처(*Caricatures*)
- 차별성(*Distinctiveness*)
- 공간주파수 해석의 역할(*The role of spatial frequency analysis*)
- 시점불변 인식(*Viewpoint-invariant recognition*)
- 조명변화의 효과(*Effect of lighting change*)
- 움직임 및 얼굴 인식(*Movement and face recognition*)
- 표정(*Facial expression*)

본 연구에서 설계하고자 하는 웹 기반 얼굴 인식 인증 시스템에는 위의 이슈들 중 얼굴의 특징점을 분석하고, 어느 정도의 시점 이동 및 조명 변화, 움직임을 허락한다는 가정하에 이루어졌다.

제 2 장. 얼굴 인식 시스템

제 1 절. 단일 이미지 상의 얼굴 인식

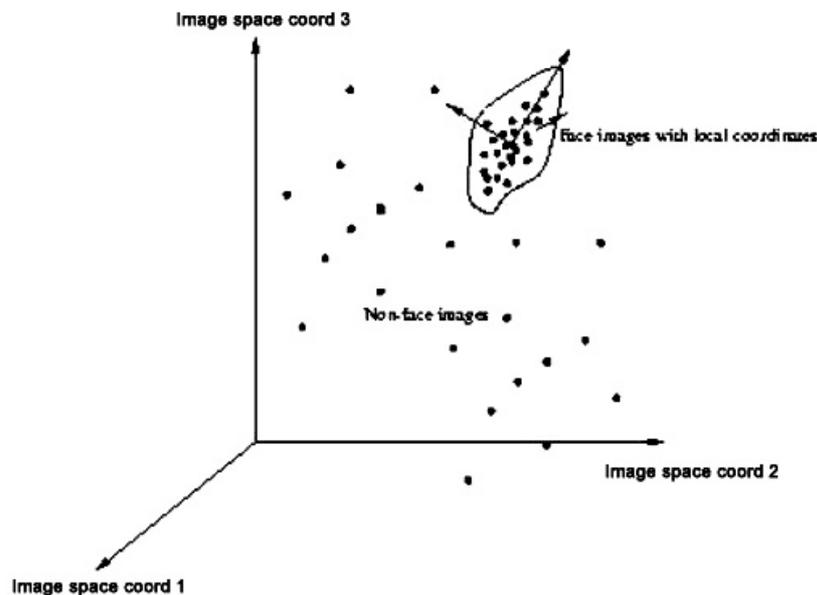
1. 분할 및 추출

복잡한 배경 속에서 사람의 얼굴 위치를 결정하기 위해서는 '얼굴(Face)' 영역과 '비얼굴(Non-face)' 영역으로 나눌 수 있는 사람 얼굴 패턴 분포를 모델링해야 한다. 각 이미지에 위치해서, 특징 벡터들과 국소 이미지 패턴과 분포-기반 모델(Distribution-based Model) 사이에서 계산된다. 그리고 나서 이 차이 벡터(Difference Vector)는 현재 이미지 위치에 사람의 얼굴이 존재하는가의 여부를 결정하기 위해 훈련된 분류기(Classifier)로 입력된다. 시스템은 모든 가능한 스케일에서 가장 유사한 국소 이미지 패턴을 위해 이미지를 낱알이 스캐닝함으로써 다른 크기의 얼굴도 검출하게 된다.

1.1 특징 추출(Feature Extraction)

특징추출은 그것이 어떤 패턴분류 작업에 대해 중요한 것처럼 얼굴분할과 인식 모두에 핵심적이다. 이러한 특징을 추출해 내기 위한 방법들과 추출한 특징들을 적용하기 위해서는 'Eigenfaces' 알고리즘 원리의 이해를 필요로 한다.

Eigen Face는 $256 * 256$ 이미지를 65536차원공간으로 만드는데 이것은 잡음과 빛의 조건에 민감하여 원래의 이미지를 가지고 얼굴인식을 하기에는 너무 복잡하고 시간 또한 오래 걸린다. 그러므로 가장 중요한 몇 개의 고유 벡터(Eigen-vector)만으로 얼굴을 표현하는데 사용할 수 있다. Eigen Face 기본 개념은 고차원으로 표현된 이미지를 저차원으로 표현해서 얼굴을 쉽게 인식하게 하는 것이다. 이렇게 차원을 줄이면 이미지의 모든 픽셀을 비교하지 않고도 Testing Face를 Training face들과 비교할 수 있다.



1.2 Eigen Face Training method

얼굴 이미지는 $N^2 \times 1$ vectors로 표현되며 X_1, X_2, \dots, X_M 의 M 장의 이미지가 있다.

$$\text{Step1) } \bar{X} = \frac{1}{M} \sum_{i=1}^M X_i \quad M\text{장의 이미지의 평균을 구한다.}$$

$$\text{Step2) } \Phi_i = X_i - \bar{X}$$

계산된 평균 이미지에서 각각 이미지의 차를 구하여 편차인 Φ_i 를 구한다. 이 과정을 통해서 vector는 normalize 된다.

$$\text{Step3) } C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (N^2 \times N^2 \text{ matrix})$$

where $A = [\Phi_1 \Phi_2 \dots \Phi_M]$ ($N \times M$ matrix)

이 이미지들의 편차로 이루어진 matrix A 와 A^T 의 곱으로 공분산 행렬(covariance matrix)을 구할 수 있다. 이 공분산 행렬을 이용하여 Eigen-value와 Eigen-vector를 구하여 이미지를 선형결합(linear combination)의 형태로 만들 수 있다.

$$\text{Step4) } A^T A v_i = \mu_i v_i \quad (\text{Eigen-value 와 Eigen-vector 구하기})$$

공분산 행렬은 다음과 같이 나타낼 수 있다.

$$C = A \Delta A^T \quad \text{where } A = [\Phi_1 \Phi_2 \dots \Phi_M]$$

그러나 이미지 vector의 공분산 행렬 크기가 $N^2 \times N^2$ 이기 때문에 만약 256 by 256의 이미지라면 65536 by 65536의 행렬이고 여기서 Eigen-value를 구한다는 것은 엄청난 비용을 초래하게 된다.

$A^T A$ 는 $M \times M$ size로 공분산 행렬보다는 작다.

$A^T A$ 을 이용하여 Eigen-value를 구하면,

$$A^T A v_i = \mu_i v_i \quad \Rightarrow \quad A A^T A v_i = A \mu_i v_i \quad \Rightarrow \quad C A v_i = A \mu_i v_i$$

$$C A v_i = A \mu_i v_i \quad \text{or} \quad A^T C A v_i = \mu_i A v_i \quad \Rightarrow \quad C u_i = \mu_i u_i \quad (u_i = A v_i)$$

위의 식을 보면 $A^T A$ 와 $A A^T$ 를 이용하여 각각 Eigen-value를 구할 수 있다. 그 두 Eigen-value를 비교하면 C를 이용하여 구한 상위 M 개의 Eigen-value와 $A^T A$ 를 이용하여 구한 Eigen-value가 같다는 것을 알 수 있다. 이 Eigen-value를 이용하여 Eigen-vector를 구하면 $A^T A$ 는 v_i 가 되며, $A A^T$ 는 u_i 가 된다.

즉, $A A^T$ 로 계산을 하면 계산량을 줄이면서 같은 Eigen-value를 얻을 수 있으며 이에 따른 Eigen-vector는 u_i 를 사용한다. 상위 몇 개의 Eigen-value만으로도 본래의 이미지를 표현할 수 있다. 상위 몇 개의 Eigen-value가 이미지의 주요 성분을 거의 포함하고 있으므로 예를 들어 100개의 Eigen-value 중 10%정도의 큰 값을 제외하고는 나머지는 0에 가까운 값을 갖는다. 0에 가까운 Eigen-value는 원래 이미지를 복원하는데 많은 영향을 끼치지 못하기 때문에 이 정도의 오류는 감안하고 처리 시간을 줄이는 것이 더 효과적이다.

$$\text{Step 5) } \hat{X} = b_1 u_1 + b_2 u_2 + \dots + b_N u_N = \sum_{i=1}^N b_i u_i \quad (b_i = u_i^T X)$$

각 이미지의 계수(Coefficient)를 위와 같은 식으로 구해서 Training image set의 각 계수들을 데이터베이스에 저장하는 방법으로 얼굴 데이터베이스가 만들어진다.



1.3 Eigen Face Recognition method

$$w_i = A u_j \quad (A = \bar{X}_{image} - X_{input})$$

- Step 1) $\Gamma = \bar{X} - X$ 번째 이미지가 들어오면 이미지를 normalize한다.
- Step 2) Training 과정을 거쳐 입력된 이미지 M개의 계수를 구한다.
- Step 3) 데이터베이스에 저장된 계수와 거리가 가장 작은 이미지를 찾는다.
- Step 4) 일정 거리 안에 들어오면 같은 사람의 이미지로 판단한다.

위와 같은 방법으로 Eigen Face를 인식할 수 있으나 얼굴영상을 Eigen Face에 적용하기 위해서는 통계학적 특성을 바탕으로 한 주성분분석법(PCA)을 적용함으로써 인식률을 좀 더 높일 수 있다.

1.4 PCA(Principal Component Analysis) 알고리즘의 활용

PCA기법은 벡터표현의 통계학적 특성을 바탕으로 한 알고리즘으로서 Karhunen-Loeve 근사 방법이라고도 한다. 통계적으로 변화가 있는 N차원의 M개의 벡터를 공분산 행렬의 고유 벡터로 표현하는 것이다.

하나의 얼굴은 영상 공간에서 하나의 벡터로 표현될 수 있고, 각 픽셀을 벡터성분으로 나타낸다. 그러나 이러한 영상 공간은 얼굴을 표현하기에 적절치 못하며, 따라서 얼굴을 묘사할 수 있는 얼굴 공간(Face space)으로 만들어야 한다. 이러한 얼굴공간의 기본 벡터를 주성분(Principal Component)이라고 한다. 하나의 얼굴을 표현하는 픽셀 모두가 관련하는 것이 아니라 이웃하는 픽셀들만 연관되어 있다. 즉, PCA 기법의 궁극적인 목표는 공간의 차원을 줄이는 것이다.

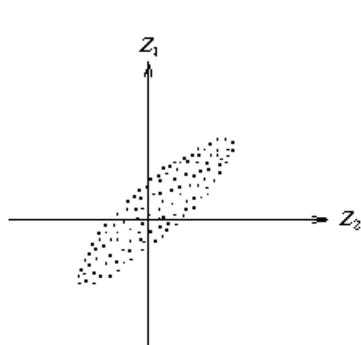


fig. 3(a) A data set and its principal axis

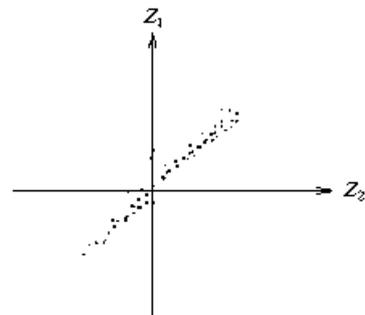


fig. 3(b) Reduction of the data on dimension

위 그림 3(a)에서 각 성분들 사이에는 긴밀한 상관관계를 갖는 2차원의 영상 데이터들이다. 여기서 데이터를 직선에 의해 표시된 부수 공간으로 투사하게 되면 숫자 데이터로 표현이 가능하며, 그 결과는 그림 3(b)에서 보여주고 있다.

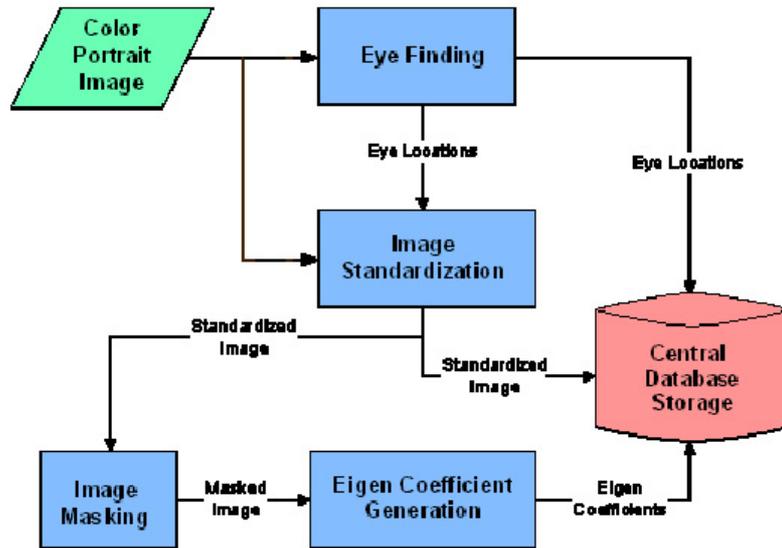
즉, PCA의 기본 아이디어는 전체 영상공간에서 얼굴을 가장 잘 표현할 수 있는 벡터를 찾는 것이다. 원래의 얼굴 이미지에서 일치하는 공분산 행렬의 Eigen-vector를 찾는 것이다. 이러한 Eigen-vector가 얼굴처럼 보이기 때문에 Eigen Face라고 하는 것이다.

2. 얼굴 등록 (Facial Enrollment)

얼굴 등록과정은 Eigen vector에 부합하고 유일하도록 얼굴 디지털 이미지를 개조(convert)함으로써 얼굴을 검색하고 얼굴인지를 인식하기 위해 인물 이미지나 비디오 세그먼트의 입수 과정으로 정의할 수 있다. 이 과정은 표현 콤포넌트(Representation Component)에 의해 수행되는 다음의 네 가지 단계를 포함하고 있다.

- *Eye Finding* (눈의 위치 탐색)
- *Image Standardization* (이미지 표준화)
- *Image Masking* (이미지 마스킹)
- *Eigen Coefficient Generation* (Eigen 계수 생성)

얼굴 등록 과정에 포함되는 주요 연산은 'Eigenfaces' 알고리즘을 기초로 하며, 다음의 Flow Diagram은 실험자에 대한 Eigen vector와 눈의 위치를 저장하는 과정을 보여주고 있다.



Face Recognition Enrollment Process Flow

2.1 눈의 위치 탐색 (Eye Finding)

눈의 위치 탐색 클래스는 표현 컴포넌트(Representation Component)와 획득 서버(Acquisition Server)의 일부분으로서, 얼굴을 인식하고 등록하는 첫 번째 단계이다. 각 눈의 센터에 대한 고유 위치가 최적화된 얼굴 인식을 수행하는데 가장 주안점이 된다.

자동적으로 Eye Finding을 하기 위해서는 비디오 프레임이나 디지털 이미지 내에서 사람의 머리 부분을 파악하는 것부터 시작한다. 머리 부분이 파악된 다음에는 눈 위치 판단 기술을 이용하여 전체 눈의 위치를 찾아낸다. 마지막으로 각 눈의 기하학적 중심좌표를 계산해낸다.

눈의 위치가 결정되었으면 확정값(confidence value)을 생성하고 낮은 확정값은 등록에 실패한 경우이므로 위의 과정이 재실행된다.

각 이미지에 대한 눈의 위치 좌표는 데이터베이스에 저장되며, 중요한 요소로 이용된다.

2.2 이미지 표준화 (Image Standardization)

이미지 표준화는 얼굴을 인식 등록하는 두 번째 단계로, 인물 이미지의 표준화 과정을 수행하며, 이 과정에서 눈의 위치를 탐색하는 초기 단계에 생성된 눈의 위치 좌표를 활용한다. 표준화를 위해서 원본 컬러 이미지가 정 사각의 그레이 스케일 이미지로 줄어들게 되며, 원본 이미지 중에 머리 주변의 영역만을 포함하게 된다.

눈의 위치는 이미지의 크기를 조정하고, 비뚤어진 이미지의 회전율을 계산하는데 필수적이며, 이미지 보정이 끝나게 되면 표준화 과정이 종료된다. 다음의 그림은 표준화 과정을 거친 결과의 예이다.



2.3 이미지 마스킹 (Image Masking)

표준화 과정의 다음 단계로 각 이미지에 일관된 마스크를 입히는 이미지 마스킹 과정을 거치게 된다. 마스크는 인물 이미지 중 배경이나 옷, 목, 머리카락과 같은 불필요한 부분을 제거한다. 다음의 그림은 이미지 마스킹 이전과 이후를 보여주고 있다.



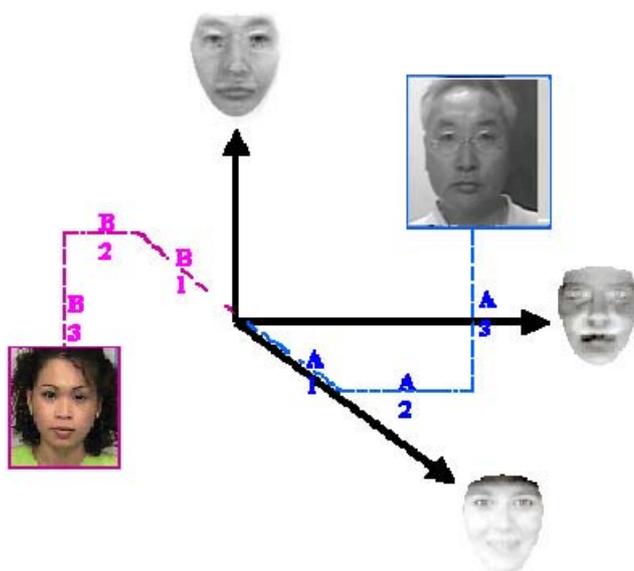
마스크는 표준화를 통해 계산된 눈의 위치와 관련해서 동일한 방식으로 등록된 모든 이

이미지에 적용이 되며, 마스크된 영역은 개인의 얼굴 특징이 적용된 Eigen Face 이미지와 일치한다.

2.4 Eigen 계수 생성 (*Eigen Coefficient Generation*)

이미지 표준화와 마스크가 끝나게 되면 다차원의 Eigen 공간으로 투사된다. 이러한 얼굴 이미지의 투사 결과는 유일한 Eigen vector를 형상화한 Eigen 계수의 집합이다.

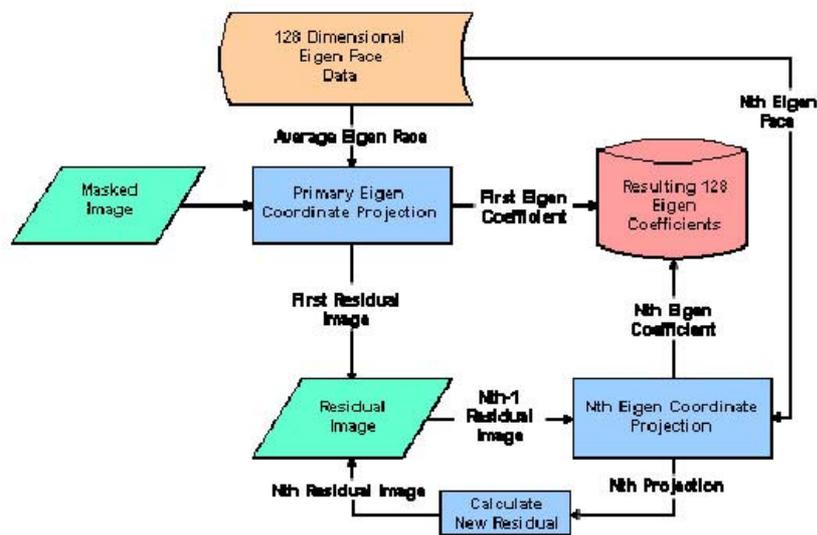
다차원의 Eigen 공간은 128개의 직교 좌표(Eigen faces)로 그려지며, 각 좌표는 단일 특징의 Eigen face 이미지로 표현된다. 다음의 그림은 Eigen 공간에서 128개의 vector좌표에 대한 3차원적 표현이다. 등록된 이미지의 첫 계수는 평균 Eigen face로 지정되는 초기 Eigen 공간좌표 위의 이미지를 투사시켜 계산한다. 계산이 끝난 첫 번째 좌표 투사는 잔여(residual) 이미지를 만들어내기 위해서 원본 이미지로부터 빠져 나온다.



3-D Representation of Eigen Space

원본 이미지에서 빠져 나온 잔여 이미지는 두 번째로 지정된 Eigen 공간좌표 위에 투사됨으로써 두 번째 계수를 얻게 된다. 두 번째 투사는 다시 새로운 잔여 이미지를 만들어내려고 이전의 잔여 이미지에서 빠져 나온다.

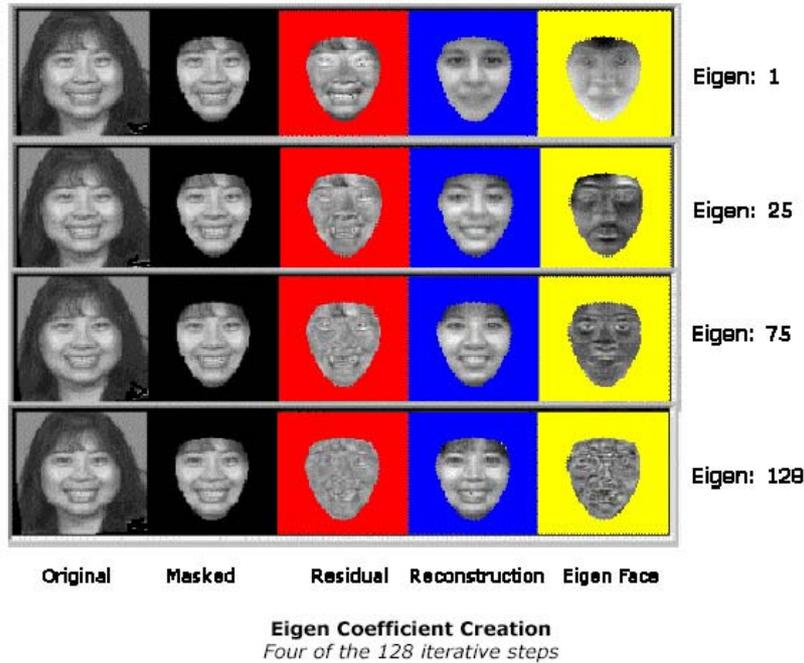
각각의 새로운 잔여 이미지가 성공적으로 다음 좌표 위에 투사된다. 더 나아가서는 구성되지 않은 잔여 이미지를 만들어내려고 빠져 나온다. 다음 그림에서는 계수 생성 알고리즘의 플로우 다이어그램을 보여주고 있다.



Eigen Coefficient Creation Flow Diagram

생성된 잔여 이미지를 각 좌표 위에 투사하는 과정에서 총 128개의 특성 Eigen 계수가 생성되고, 이렇게 생성된 특성 계수들은 얼굴 인식 Eigen 공간 내에서 완전한 vector 투사를 나타낸다. 부합하는 Eigen face 이미지에 관한 이들 128개 계수의 결합은 원본 마스크 이미지와 거의 비슷하게 증명되거나 거의 똑같아 보일 수 있는 재구성 마스크 이미지를 만들어낸다.

다음 그림은 128개의 특성 Eigen 계수의 생성 과정을 묘사하고 있다.



제 3 장. 웹 기반 얼굴 인식 구현을 위한 시스템 모델링

제 1 절. 얼굴 인식 모듈의 구성

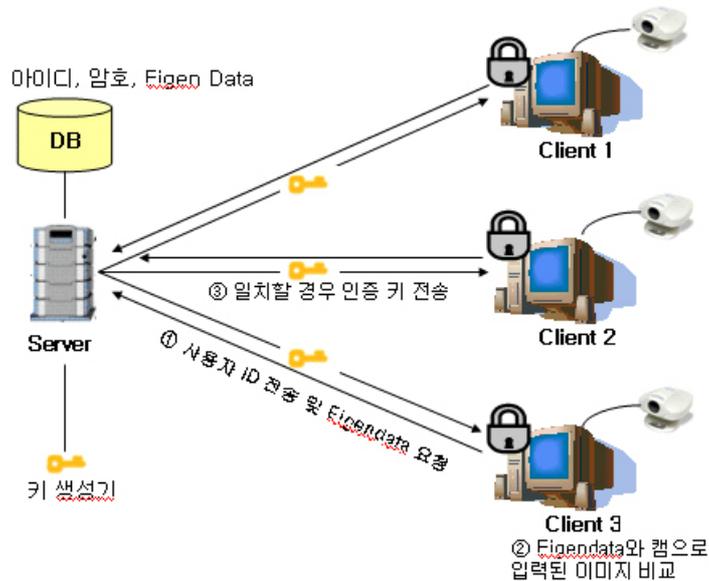
1. 얼굴 등록 모듈

3장에서는 이전 장에서 논의했던 얼굴 인식 시스템의 얼굴 등록 과정을 웹에 적용하여 서버에서 제공하는 웹 어플리케이션을 클라이언트(Client)가 직접 다운로드 받아 설치함으로써 얼굴 등록 및 인증이 클라이언트 사이드에서 이루어지는 모듈을 구성하게 된다.

얼굴 등록 및 인증이 서버 사이드(Server-side)에서 이루어질 수도 있으나 서버 사이드의 부하를 방지하며 다중 사용자 접속시 원활한 서비스 제공을 위해서 클라이언트 사이드 인증 방식을 채택하였다. 다만 서버 사이드에서는 Eigen Face에 대한 데이터 값과 아이디, 암호를 데이터베이스로 저장하고 클라이언트의 신분 확인에 이용될 키

인증 처리를 다루게 되며, 클라이언트 사이드의 인증 결과에 따라서 다양한 서비스를 제공하도록 설계하였다.

다음 그림은 클라이언트의 인터페이스 구성을 나타낸 플로우(Flow) 이다.



클라이언트는 사용자의 ID를 서버로 전송하면서 입력 이미지와 비교할 Eigen Face 관련 데이터 값을 요청하게 된다. 서버측에서는 클라이언트의 ID와 데이터베이스에 저장된 아이디를 비교하고 아이디에 맞는 Eigen Data 값을 클라이언트에 전송해주고, 클라이언트의 얼굴 등록 및 신원 확인 시스템에 의해 비교(Compare)가 이루어진다.

Eigen Face와 원본 이미지의 데이터 값이 일치하는 경우에는 서버측으로 True 값과 아이디를 전송하게 되며, 서버측은 비교값이 'True'인 경우에만 인증 키를 전송해서 서비스를 이용할 수 있는 권한을 주며, 얼굴 인식이 제대로 이루어지지 않은 경우가 발생하게 되므로 이를 위한 방안으로 직접 패스워드를 입력하여 비교값이 'False'이지만 본인인 경우를 'True'로 바꿔서 Eigen Data를 업데이트 시켜줌으로써 유효 범위를 확장시켜준다.

키 전송 및 아이디, 암호와 관련된 부분은 현재 보안 분야에서 다양한 연구와 발표가 이루어진 상태이며, SSL(Security Socket Layer)이나 PKI(Public Key Infrastructure) 방식 등을 이용함으로써 안전한 키 전송이 이루어지도록 하는 방안을 본 시스템 구현 단계에서 다루어지게 될 것이다.

제 4 장. 결론 및 개선 방안

본 연구에서는 얼굴 인식 시스템의 알고리즘과 웹 기반에서 얼굴 인식 기술이 응용되는 방법을 간단하게 다루었으며, 앞으로 이루어질 구현 단계에 필요한 일반적인 요구사항에 대하여 정리해 보았다. 본 논문의 근간이 되는 얼굴 인식 기술은 최근 각광받는 생체 인식의 한 분야로서 꾸준히 새로운 연구가 진행되고 있으며, 얼굴 인식 기술을 이용한 새로운 어플리케이션이 상용화되고 있다. 얼굴 인식은 타 생체 인식에 비하여 사용의 편의성이나 비 접촉성 등의 이점이 있으나, 얼굴 인식 기술의 구현 자체가 어렵고 배경 및 조명 등 시스템 외적인 면의 간섭을 완전히 배제할 수 없다는 단점이 남아있다. 또한, 얼굴 인식에 대한 절대적인 이론이 정립되어 있지 않은 바 아직도 많은 연구 개발의 여지가 있다고 볼 수 있다.

그럼에도 얼굴 인식기술이 넓은 응용 분야를 가지는데 관공서, 사업체 등의 출입 통제 및 과거의 사진 데이터만 남아있다면 미아 찾기에도 활용할 수 있으며, 공항이나 인구 밀집 지역에서의 수배자를 검색하는 시스템 등이 그 예라 할 수 있다.

본 연구에서는 이런 응용 분야 중에서도 웹 서비스를 제공할 때 사용자 신원 인증을 얼굴 인식 기술로 처리를 한다면 사용자의 실수로 인한 비밀번호 분실 방지 및 금융 관련 분야에서 자동화된 고객 인식, 수표 및 카드 범죄자의 색출 등의 강력한 보안대책으로 충분히 이용될 수 있음을 시사하고 있다.

앞으로 이루어질 웹 기반의 얼굴 인식 인증 처리 시스템 구현 단계에서는 인증 시간의 최적화와 조명, 화질 및 데이터베이스 구성에 대한 문제점을 해결하고, 얼굴 검색 엔진의 인식률을 높이기 위한 대안적인 알고리즘의 구현이 이루어질 것이며, 키 인증 전송에 대한 보안 대책을 수립될 계획이다.

참고문헌

- [1]Matthew A. Turk and Alex P. Pentland, "Face Recognition Using Eigenfaces" *Vision and Modeling Group, The Media Laboratory, MIT, 1991, pp 586-591*
- [2]W. W. Bledsoe, "The model method in facial recognition," *Panoramic Research Inc., Palo Alto, CA, Rep. PRI:15, Aug. 1996*
- [3]Viisage Technology, Inc. "FaceTOOLS™ Software Development Kit Reference" P9~15