

스마트폰을 이용한 얼굴인식 출입관리 시스템

권기현*, 이형봉**

Gate Management System by Face Recognition using Smart Phone

Kihyeon Kwon *, Hyung Bong Lee **

요약

본 논문에서는 스마트폰 얼굴인식을 통해 출입을 관리하는 시스템을 설계하고 구현한다. 이를 위해 스마트폰에서 얼굴인식을 위한 사용가능한 다양한 알고리즘을 조사하였다. 얼굴 인식의 첫 단계는 얼굴검출이며 다음 단계는 얼굴인식이다. 얼굴 검출을 위해서는 컬러 세그멘테이션, 템플릿매칭 등의 알고리즘을 적용하였으며, 얼굴 인식을 위해서는 PCA(Principal Component Analysis)에 기반을 둔 Eigenface와 LDA(Linear Discriminant Analysis)에 기반을 둔 Fisherface를 비교하여 구현하고 적용하였다. 스마트 폰의 제한된 하드웨어에서 얼굴인식 시스템을 구현하는 관계로 알고리즘의 정확도와 알고리즘의 계산 복잡도 사이에서 적절한 조절이 필요하였다.

▶ Keyword : 얼굴인식, 얼굴검출, 스마트폰, 출입관리

Abstract

In this paper, we design and implement of gate management system by face recognition using smart phone. We investigate various algorithms for face recognition on smart phones. First step in any face recognition system is face detection. We investigated algorithms like color segmentation, template matching etc. for face detection, and Eigen & Fisher face for face recognition. The algorithms have been first profiled in MATLAB and then implemented on the Android phone. While implementing the algorithms, we made a tradeoff between accuracy and computational complexity of the algorithm mainly because we are implementing the face recognition system on a smart phone with limited hardware capabilities.

▶ Keyword : Face Recognition; Face Detection; Mobile Phones, Gate Management

• 제1저자 : 권기현 • 교신저자 : 이형봉

• 투고일 : 2011. 06. 15, 심사일 : 2011. 08. 03, 게재확정일 : 2011. 09. 19.

* 강원대학교 전자정보통신공학부(Dept. of Information & Communication Engineering, Kangwon National University)

* 강릉원주대학교 컴퓨터공학과(Dept. of Computer Engineering, Gangneung-Wonju National University)

I. 서론

스마트 폰의 성능이 점차 강력해짐에 따라, 스마트 폰과 연동하여 일정한 서비스를 수행하는 분야에 관심이 증대되고 있다[1]. 현재 대부분의 출입관리 시스템은 보안을 위해 패스워드를 사용하고 있으나, 얼굴인식을 스마트 폰에 적용하면 훨씬 더 안전하고 유연한 접근이 가능하며 패스워드를 기억할 필요가 없게 된다[2].

이 연구의 목표는 안드로이드 폰과 지정된 임베디드 컨트롤러 간에 얼굴인식을 통해 출입관리 시스템을 구현하는 것으로, 출입문에는 카메라 등의 영상 인식장치가 없이 스마트폰의 카메라와 무선으로 연결된 임베디드 컨트롤러, 임베디드 컨트롤러와 블루투스로 연결된 출입문 사이의 연동을 통해 출입관리를 하는 것이다.

얼굴 인식은 지난 10년간 매우 활발한 연구 분야중의 하나이다. 다양한 기술이 얼굴 검출과 인식을 위해 연구되어왔다 [3]. 얼굴 인식의 첫 단계는 얼굴검출이며 다음 단계는 얼굴 인식이다. 얼굴 검출을 위해서는 주로 컬러 세그멘테이션(color segmentation), 형태학상 처리(morphological processing) 그리고 템플릿 매칭 알고리즘(template matching algorithm)을 사용하여 얼굴검출을 수행할 수 있다. 컬러 세그멘테이션(color segmentation)에 대한 상당히 폭넓은 연구가 진행되었으며 스किन모델링 방법이 제한된 훈련세트에서 더 잘 동작한다[4]. RGB, YCbCr 또는 HSV(hue, saturation, value) 같은 다양한 컬러스페이스가 사용될 수 있으며[4], Chai는 피부색을 검출하기 위해 색조(Hue)와 Cr을 사용하는 방법을 제안하였다 [5]. 본 연구에서는 얼굴 인식 기법으로 널리 사용되는 두 기법인 Eigenface와 Fisherface 알고리즘[6, 7]을 사용하여 인식을 수행한다.

스마트 폰의 제한된 하드웨어에서 얼굴인식 시스템을 구현하는 관례로 알고리즘의 정확도와 알고리즘의 계산 복잡도 사이에서 적절한 조절이 필요하다. 본 논문의 구성은 2장에서 얼굴인식 기반 출입관리 시스템에 대한 구성도를 살펴보고, 3장, 4장에서는 얼굴검출 및 얼굴인식을 위해 사용되는 알고리즘과 얼굴 검출과정과 얼굴 인식과정에 대해 기술하였으며, 5장에서는 시스템구현과 성능평가를 설명하고, 6장에서 결론을 맺는다.

II. 시스템 구성

2.1 시스템 구성도

구현한 출입관리 시스템의 구성은 세 부분으로 나뉘는데, 먼저 얼굴검출 단계에서는 스마트폰에서 컬러 세그멘테이션(color segmentation), 형태학상 처리(morphological processing) 그리고 템플릿 매칭 알고리즘(template matching algorithm)을 적용하고, 임베디드 컨트롤러에서는 얼굴 인식을 위해 PCA(Principal Component Analysis)기반 Eigenface, LDA(Linear Discriminant Analysis)기반 Fisherface 얼굴 인식 알고리즘을 사용하며, 출입문 개폐를 위해서는 블루투스를 사용한다. 모든 알고리즘은 MATLAB에서 구현한 다음 스마트 폰 및 임베디드 컨트롤러로 이식하여 사용하였다. 구현된 전체적인 시스템의 구성도는 그림 1과 같이 지정된 무선랜을 검색하여(①), 스마트폰에서 촬영된 얼굴이미지를 지정된 무선랜(WIFI) 모듈을 경유, 임베디드 컨트롤러로 전송하면(②) 컨트롤러에서 얼굴인식을 처리하여, 블루투스로 구동되는 출입문의 개폐를 수행(③)하는 구조이다.

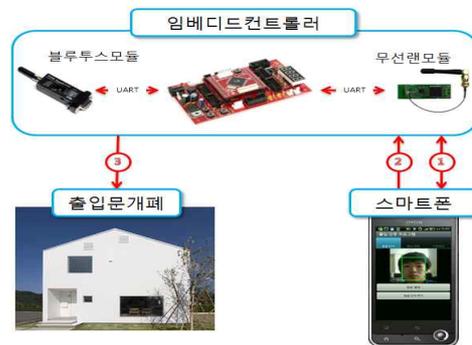


그림 1. 시스템 구성도
Fig. 1. System Architecture

2.2 얼굴검출 및 인식 절차

그림 2는 얼굴인식절차의 주요 단계를 나타낸 것이다.

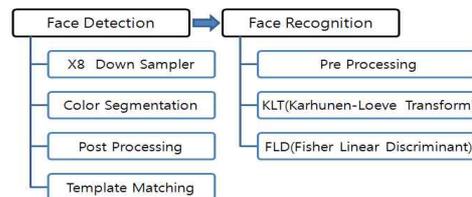


그림 2. 얼굴 인식 절차
Fig. 2. Process of Face Recognition

III. 얼굴검출

3.1 얼굴검출 단계

얼굴인식단계의 첫 단계는 얼굴 검출이다. 컬러 세그멘테이션(color segmentation), 형태학상 처리(morphological processing) 그리고 템플릿 매칭 알고리즘(template matching algorithm)을 사용하여 얼굴검출을 수행 한다. 얼굴검출 처리 알고리즘을 모두 사용할 필요는 없으며, 주어진 조건하에서 빠르고 적절하게 수행되는 알고리즘을 찾는 것이 필요하다.

다음 기법을 사용하여 얼굴을 검출한다.

- 1) 컬러 세그멘테이션을 사용하여 스킨 픽셀을 찾는다.
- 2) 형태학상 처리를 사용하여 고립 픽셀을 제거한다.
- 3) 템플릿 매칭을 사용하여 얼굴을 추출한다.

추출된 얼굴이미지는 얼굴인식에서 사용하게 된다.

3.2 컬러 세그멘테이션

컬러 이미지에서 피부색의 검출은 얼굴인식에서 매우 잘 알려지고 강력한 기법이다. 피부색 검출 절차에서, 각 픽셀은 컬러 컴포넌트 값에 따라서 피부색인지 아닌지 분류된다.

피부색을 찾는 빠른 방법은 두 가지가 있다. 첫 번째는 계산을 피하기 위해 RGB 공간에서 처리하는 것이다. 컬러값 (R, G, B)을 가지는 픽셀은 다음과 같을 때 피부색으로 분류된다.

- $R > 95$ and $G > 40$ and $B > 20$ and
- $R > G$ and $R > B$ and
- $R - G > 15$

이 알고리즘은 일반적으로 잘 수행되나 RGB 분류기의 성능이 느리다. 또 다른 방법으로 Cr or Hue 분류기에 기반을 둔 것이 있으나 분류가 너무 엄격하거나 어떤 조명상태에서는 너무 불분명한 단점이 있다.

끝으로, Cr 분류기는 한 픽셀이 $Cr \in [136 \ 173]$ 이면 피부로 간주한다[5]. Cr 컴포넌트가 RGB(affine transformation)로부터 계산하기 편하고 수행할 테스트가 2개만 있으므로, 매우 빠르게 처리 할 수 있고 매우 좋은 결과를 구할 수 있다. 따라서 이 분류기를 사용하도록 하였다.

3.3 형태학상 처리(morphological processing)

컬러 세그멘테이션 후, 피부색이 아닌 픽셀의 마스크가 얻

어진다. 그러나 이 마스크는 완전하지 않다. 얼굴의 일부가 마스크 되어 있으나 피부색이 아닌 부분이 여전히 나타날 수 있다. 형태학상 처리를 통해 피부색이 아닌 픽셀을 제거할 수 있다.

3.4 템플릿 매칭(template matching)

컬러 세그멘테이션 다음에는, 얼굴 검출의 마지막 단계로 템플릿 매칭을 사용한다. 템플릿 매칭은 입력에 대한 상호상관과 템플릿으로, 입력 이미지 $I(x, y)$ 에서 템플릿 $T(x, y)$ 를 사용하여 오브젝트를 찾는 절차이다. 상호상관은 계산이 용이하도록 FFT와 IFFT를 사용한 주파수 영역에서 구현된다.

사용한 표준 템플릿은 그레이 이미지로 입력 이미지는 그레이 이미지로 변환되고 템플릿 이용 상호상관을 계산하였다. 그림 3과 같이 표준 템플릿은 어떤 회전(찌그러짐)도 없는 똑 바른 얼굴이므로, 입력 이미지가 이와 같이 바로고 왜곡이 없을 때 템플릿 매칭이 잘 될 수 있다.



그림 3. 표준얼굴
Fig. 3. Standard average face template

컬러 세그멘테이션 및 형태학상 처리 다음에 이상적인 얼굴 부분이 남는 경우도 있을 수 있다. 이런 경우에는 템플릿 매칭이 필요하지 않을 수도 있다. 그러나 손이나 목 같은 피부가 노출된 경우나 배경색이 피부색과 비슷한 경우에는, 템플릿 매칭이 얼굴을 검출하는데 도움이 될 수 있다.

IV. 얼굴인식

앞의 단계에 의해 얼굴 부분이 검출된 다음, 훈련세트와 인식 작업이 이루어진다.

4.1 얼굴인식 알고리즘

얼굴 인식 기법으로 널리 사용되는 두 기법인 Eigenface와 Fisherface 알고리즘에 대한 간단히 소개하면 다음과 같다.

Eigenface 방법은 차원감축기법을 사용하며, PCA(Principal Component Analysis)로 알려져 있다. 이들 방법은

모든 투영 이미지의 산포도를 최대화하는 선형 프로젝션을 줄여 차원을 감축한다.

N이미지 훈련 집합 $\Gamma(i = 1, 2, \dots, N)$ 가 주어진 경우, 각 크기는 $m \times n$ 이고, 이 집합을 큰 매트릭스로 변환한다.

$$A = [\Phi_1, \Phi_2 \dots \Phi_n]$$

여기에서 Φ_i 는 컬럼벡터이고 이미지 하나를 의미하며

$$\begin{aligned} \Phi_i &= \phi_i - \mu \\ \phi_i &= \text{reshape}(\Gamma_i', [mn, 1]) \\ \mu &= \text{mean}(\phi_i) \end{aligned}$$

전체 이미지 산포 행렬(scatter matrix)은 다음과 같이 정의된다.

$$S_T = AA^T$$

선형 변환 W로 이미지 공간을 p-차원 특징공간으로 매핑을 고려하면, $p \leq N \ll nm$ 이다. PCA는 투영 이미지의 전체 산포 행렬의 판별식(determinant)을 최대화하도록 W_{opt} 을 선택하는 문제이다.

$$W_{opt} = \text{argmax} |W^T S_T W| = [w_1 w_2 \dots w_p]$$

w_i 는 p-largest 고유값(eigenvalues)에 대한 S_T 의 고유벡터이다. 이들 각각은 고유얼굴(eigenface)을 구성하게 된다. 특징 공간의 차원은 p로 줄어들게 된다. 훈련 집합 이미지와 테스트 이미지의 가중치가 구해지고 훈련세트이미지와 비교하여 유클리디언 거리가 가장 가까운 것이 인식된다.

Eigenface 방법은 전체 훈련 집합에서 산포도를 최대화하므로, 같은 클래스에 속한 점들이 투영된 공간으로 클러스터링이 잘되지 않거나 서로 서로 간섭될 수도 있다. 빛에 따른 편차(variation)가 클러스터간의 편차보다 한 클러스터내의 클래스 간의 편차를 더 크게 만들기도 한다. 이 문제를 극복하는 방법은 이런 편차를 줄이는 것이며, Fisherface 알고리즘이 이에 해당한다. 이 알고리즘은 클래스 내부의 산포도를 최소화하고 클러스터와 클러스터 사이를 최대화하도록 고안되었다. Between-class scatter matrix를 S_B , with n-class scatter를 S_W 라고 하면 최적 투영 공간은 다음 방법으로 구해진다.

$$W_{opt} = \frac{\text{argmax} |W^T S_B W|}{\text{argmax} |W^T S_W W|} = [w_1 w_2 \dots w_p]$$

S_W 의 rank(계수)는 커 봐야 N-c이다. 여기에서 c는 훈련 집합 클래스의 수(파일 수)이고, PCA를 사용하여 차원을 줄이는 첫 단계에서 사용된다. 여기에는 많아봐야 c-1개의 고유벡터가 있으므로, 최대 c-1개의 Fisherface가 있게 된다.

4.2 얼굴인식 훈련 및 테스트 세트

총 105개의 이미지를 사용하였고, 21개의 클래스(사람), 각 클래스 당 5개의 이미지를 사용하였다.



그림 4. 전 처리한 훈련 이미지 집합
Fig. 4. Pre-processed training set

얼굴 검출기에 의해 얻어진 부분 얼굴 이미지는 인식의 입력으로 사용된다. 얼굴은 같은 크기로 재조정되고 중심부로 이동 처리된다. 얼굴 이미지는 색상, 빛의 방향 등에 따라 상당히 달라진다. 얼굴 표정이나 각도에 대한 것 보다 색상, 빛의 방향 등이 같은 클래스(사람)에 대해 더 많은 편차를 유발한다. 배경색도 변화하여, 같은 클래스의 이미지들 간에 차이를 가져오게 된다. 이런 효과를 완화시키기 위해서는, 몇몇 전처리 단계가 수행되어야 한다. 여기에는 정규화작업, 색을 그레이로 변환, 히스토그램 평탄화(histogram equalization) 등이 있다. 결과 이미지 중 일부가 그림 4에 제시되어 있다.

4.3 Eigenface

동일한 훈련 세트를 사용하여 eigenface와 Fisherface에서 사용하였다. Eigenface에서는 계산된 평균 이미지가 그림 5에 있다. 우리의 경우 105개의 이미지를 사용하여, 104개의 고유벡터, 즉, 104 개의 고유얼굴이 구해진다. 고유값의 크기의 내림차순으로 정렬하였다. 첫 번째 25개의 고유얼굴을 그림 6에 나타내었다. 훈련 집합의 얼굴이미지는 고유얼굴로 만들어지고, 특징공간의 값으로 가중치가 구해진다.



그림 5. 평균이미지
Fig. 5. Average face of the training set

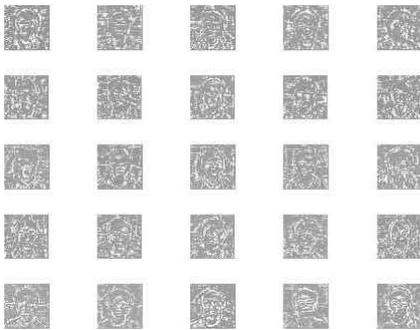


그림 6. Eigenface
Fig. 6. The eigenface

4.4 Fisherface

Fisherface 에서는 위에서 구한 eigenface가 첫 단계로 사용된다. 그러나 단지 $N-c = 105-25 = 80$ 개의 고유 벡터가 차원감축 되어 사용되어, Fisher 선형 판별도가 낮은 차원 공간에서 구해졌다(그림 7).

Eigenface와 유사하게, 훈련 집합의 이미지는 Fisherface 로 만들어지고, 가중치가 구해져 저장된다.

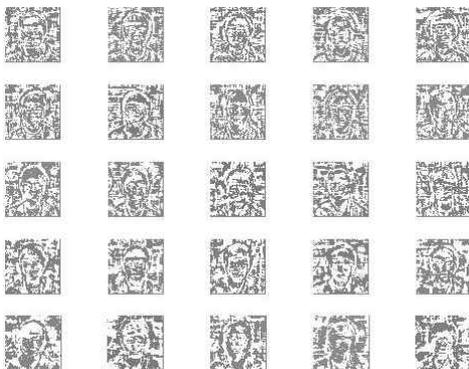


그림 7. Fisherface
Fig. 7. The Fisherface

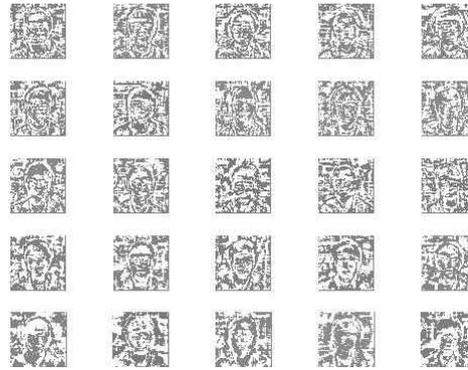


그림 7. Fisherface
Fig. 7. The Fisherface

V. 구현 및 성능평가

5.1 실험 환경

스마트폰을 이용한 얼굴인식 출입관리 시스템을 위해 스마트폰, 무선랜모듈, 임베디드 컨트롤러, 블루투스 모듈이 사용 되었으며 각 사양은 표 1과 같다.

표 1. 하드웨어 스펙
Table 1. Hardware Specification

항 목	주요 스펙
스마트폰	· 안드로이드 v22
WiFi-RS232 변환모듈	· SOC칩, WiFi 프로토콜 스택 탑재 · 802.11b 모드 지원 무선랜 최대 11Mbps
임베디드 컨트롤러	· AVR Atmega128
블루투스 내장형 RS232 이답터	· 921.6kbps · Bluetooth v2.0 + EDR, Class 1, Profiles - SPP

5.2 구현

MATLAB에서 얼굴인식 알고리즘의 프로파일링 및 테스트를 수행한 다음, 안드로이드에서 알고리즘을 구현하였다.

안드로이드 플랫폼에서 Fisherface, Eigenface 얼굴 인식 알고리즘을 개발하였으며 Fisherface가 Eigenface보다 더 효율적인 것으로 평가되어 Fisherface를 선택하였다. 가중치 매트릭스를 임베디드 컨트롤러에 저장하여, 고정 훈련 세트로 사용하였다.

다른 중요한 문제는 계산 시간을 가능한 작게 하는 것이었다. 이리하여, 사용자 사용성이 최대가 되게 하는 것이다. 이

목표를 얻기 위해서, 고성능 해상도 카메라의 사진을 8의 배수로 나누어 해상도를 줄였다. 계산 시간은 전체 이미지에 비해 64배 빨라졌다. 두 번째로, I/O 문제도 중요하다. MATLAB으로 작성된 KLT(Karhunen-Loeve Transform), FLD(Fisher Linear Discriminant) 그리고 평균 얼굴 매트릭스 등의 텍스트 파일을 바이너리 파일로 작성하였다. DataInputStream을 사용하는 것이 텍스트에서 라인단위로 읽어 들이는 것보다 훨씬 빨랐다. 최종적으로, 우리 알고리즘은 사용자 검출 및 식별에 1.4 sec 이하 걸렸다. 그림 8에 사용자 인터페이스를 묘사하였다.

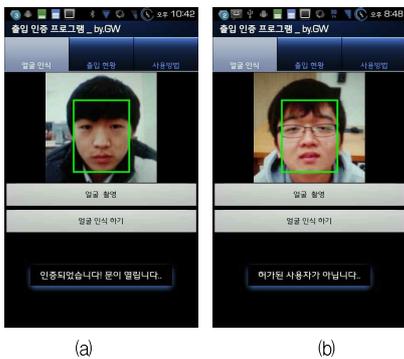


그림 8. 얼굴 인식 화면
Fig. 8. Result of the Face Recognition application for Android. (a) user recognized, (b) user rejected

5.3 성능 평가

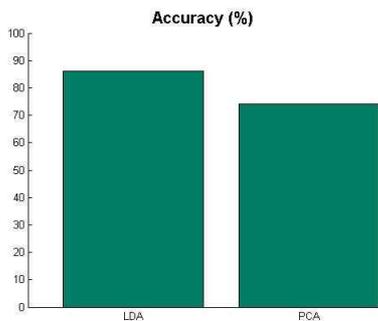


그림 9. 알고리즘 정확도
Fig. 9. Accuracy between LDA & PCA

LDA(Fisherface)와 PCA(Eigenface)의 정확도는 각각 87.15%, 74.5%로 LDA가 우수하다(그림 9). LDA와 PCA의 훈련시간(a)과 전체실행시간(b)은 LDA가 PCA에 비해 상대적으로 크지만 훈련세트를 미리 준비하여 임베디드 컨트롤러에 위치시키기 때문에 훈련시간에 문제가 되지 않는다(그림 10).

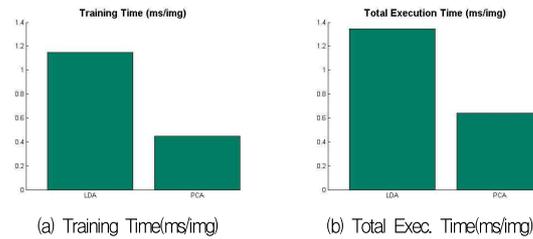


그림 10. 훈련시간과 전체실행시간(ms/img)
Fig. 10. Training Time & Total Exec. Time(ms/img)

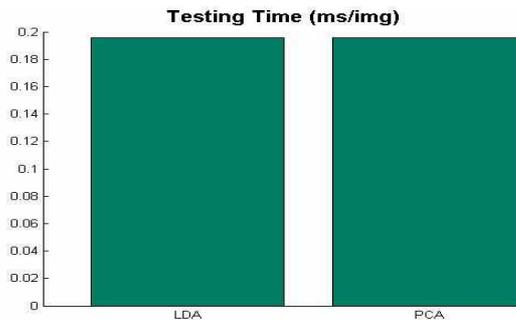


그림 11. 테스트 시간(ms/img)
Fig. 11. Testing Time(ms/img)

인식에 걸리는 시간은 LDA(Fisherface)와 PCA(Eigenface) 모두 차이가 없기 때문에 인식에 사용하는데 문제가 되지 않았다(그림 11). 모바일 폰에서 인식은 보안을 위해 사용하므로 약간 높은 인식 거절율을 유지하도록 하여 잘못된 승인 비율을 낮게 하는 것이 필요하다.

VI. 결론

이 프로젝트에서, 모바일 폰에서 얼굴 검출과 인식을 구현하기 위한 다양한 알고리즘을 조사하여 적용하였다. 얼굴 검출을 위해 컬러 세그멘테이션, 템플릿매칭을 사용했다. 얼굴 검출 알고리즘에는 몇 가지 제한사항이 있는데, 배경색이 피부색과 유사한 경우 컬러 세그멘테이션 방법은 배경색과 피부색을 구분하기 어렵다. 템플릿매칭 알고리즘의 결과는 선택된 표준 템플릿(방향 등), 조명조건 등에 의존적이게 된다. 얼굴 인식을 위해 Eigenface와 Fisherface 알고리즘을 사용하여 얼굴인식에 사용하였다. 우리는 Eigenface를 사용했을 때 74.5%, Fisherface를 사용 시 87.15%의 정확도를 보였으며, 후자가 더 나은 성능을 보였다. 끝으로, 얼굴인식알고리

즘을 안드로이드에서 구현하여 표준 얼굴 검출 기능과 연동하였고 전체 얼굴 인식 시스템을 완성하였다.

향후연구로, IPCA (Incremental PCA for online learning)과 ILDA (Incremental LDA for online learning)을 적용하여 동적인 사용자 관리가 가능한 출입 관리 시스템으로 발전시키고자 한다.

참고문헌

- [1] Brian C. Becker, "Evaluation of Face Recognition Techniques for Application to Facebook," IEEE Xplore, 2009.
- [2] Kwon, Man-Jun, "Real-Time Face Recognition System using PDA," Journal of fuzzy logic and intelligent systems, Vol.15 No.5, pp.649-654, 2005
- [3] V Vezhnevets, V Sazonov, A Andreeva, "A survey on pixel-based skin color detection techniques," Proc. Graphicon, 2003 - Citeseer.
- [4] P. Peer, J. Kovac, F. Solina, "Human skin colour clustering for face detection," in submitted to EUROCON 2003 - International Conference on Computer as a Tool.
- [5] D. Chai and K. N. Ngan, "Face segmentation using skin-color map in videophone applications," IEEE Trans. on Circuits and System for video Technology, Vol.9, No.4, pp. 551-564, 1999.
- [6] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," IEEE transactions on pattern analysis and machine intelligence, Vol. 19, No. 7, 1997.
- [7] Thomas Heseltine, Nick Pears, Jim Austin, Zezhi Chen, "Face Recognition: a comparison of appearance-based approaches," Proc. VIIth Digital image computing: Techniques and applications, 10-12, 2003.
- [8] Jang, Seok-Woo, Choi, Hyung-II, Kim, Gye-Young, "Adult Image Detection Using Skin Color and Multiple

Features," Journal of the Korea society of computer and information, v.15, no.12, pp.27-35, 2010.

저자 소개



권기현

1993년 : 강원대학교 전자계산학과
이학사.
1995년 : 강원대학교 전자계산학과
이학석사
2000년 : 강원대학교 컴퓨터과학과
이학박사
2002년~현재 : 강원대학교 전자정보
통신공학부교수
관심분야 : 임베디드시스템, 패턴인식
Email: kweon@kangwon.ac.kr



이형봉

1984년 : 서울대학교 계산통계학과
학사.
1986년 : 서울대학교 계산통계학과
석사.
2002년 : 강원대학교 컴퓨터과학과
박사
2004년~현재 : 강릉원주대학교 컴퓨
터공학과 부교수
관심분야 : 임베디드 시스템, 센서네트
워크, 데이터마이닝 알고리즘
Email: hblee@gwnu.ac.kr