

논문 2010-47CI-3-2

# 다중 얼굴 태깅 자동화

( Automatic Tagging Scheme for Plural Faces )

이 충 연\*, 이 재 동\*\*, 진 성 아\*

( Chung-yeon Lee, and Jae-dong Lee, and Seongah Chin )

## 요 약

최근 웹페이지의 생성 및 웹이 가진 정보량이 기하급수적으로 늘면서 사용자의 검색 목적을 파악하여 효율을 높이기 위한 다양한 방법이 연구되고 있으며, 태깅 시스템이 하나의 대안으로 떠오르고 있다. 태깅 시스템은 인터넷 사용자로 하여금 태그라고 불리는 메타데이터를 글, 사진, 동영상 등에 부여하도록 함으로써 콘텐츠의 검색 및 브라우저를 편리하게 하는 시스템이다. 이처럼 태그는 해당 페이지의 대표 키워드를 의미하므로 콘텐츠 분류의 기준을 마련할 수 있으나, 사용자에게 의해 직접 입력되어야 하는 수고가 필요하고, 또한 무분별한 태깅으로 인해 오히려 분류에 방해가 되는 등의 문제점들이 있다. 본 논문에서는 이러한 태깅의 문제를 해결하기 위한 방법으로 얼굴인식 알고리즘을 활용한 영상콘텐츠 내에서의 다중 얼굴 태깅 자동화 방법을 제시한다. 이를 위해 먼저 여러 얼굴검출 방법 중 Haar-like features와 AdaBoost 알고리즘을 이용하여 빠른 속도와 높은 정확도로 영상콘텐츠 내에서 얼굴 영역을 검출한다. 이후 PCA와 고유얼굴을 이용하여, 검출해 낸 얼굴을 데이터베이스에 미리 저장해 놓은 프로필 사진과 비교, 인식해냄으로써 해당 인물에 대한 정보를 불러와서 자동으로 태깅하는 시스템을 구현하였다. 이러한 새로운 방식의 태깅 기술은 현존하는 사진공유, 쇼핑, 검색 등의 수많은 웹서비스에 적용이 가능하며, 특히 소셜네트워크서비스에서의 사진 관리나 인물검색 등에서 활용할 때 큰 효과를 보일 것으로 기대된다.

## Abstract

To aim at improving performance and reflecting user's needs of retrieval, the number of researches has been actively conducted in recent year as the quantity of information and generation of the web pages exceedingly increase. One of alternative approaches can be a tagging system. It makes users be able to provide a representation of metadata including writings, pictures, and movies etc. called tag and be convenient in use of retrieval of internet resources. Tags similar to keywords play a critical role in maintaining target pages. However, they still needs time consuming labors to annotate tags, which sometimes are found to be a hinderance caused by overuse of tagging. In this paper, we present an automatic tagging scheme for a solution of current tagging system conveying drawbacks and inconveniences. To realize the approach, face recognition-based tagging system on SNS is proposed by building a face area detection procedure, linear-based classification and boosting algorithm. The proposed novel approach of tagging service can increase possibilities that utilized SNS more efficiently. Experimental results and performance analysis are shown as well.

**Keywords :** 자동태깅, 태깅시스템, 소셜네트워크, 웹2.0, 얼굴인식

## I. 서 론

전 세계의 인터넷 사용자들은 자신의 미니홈피나 블로그(blog), 또는 카페나 클럽과 같은 참여형 대형 콘텐

츠 공간을 통해 자신의 생각을 담은 글이나 직접 제작 또는 촬영한 영상 콘텐츠 등을 올리고 있으며, 다수의 기업들이 이를 활용한 다양한 비즈니스 모델들을 통해 막대한 규모의 수익을 창출해내고 있다. 이처럼 사용자 제작콘텐츠(UCC, User Created Contents)는 현재 인터넷에서 가장 각광받는 분야 중 하나이며, 앞으로도 계속 성장할 잠재성을 지닌 비즈니스 아이템이다<sup>[1]</sup>.

한편 이러한 글이나 사진, 동영상 등의 UCC에 사용자가 직접 키워드를 다는 태그(tag)는 웹 2.0 시대의 새

\* 정희원, 성결대학교 멀티미디어공학부  
(Dept. of Multimedia Engineering, Sungkyul Univ.)

\*\* 학생회원, 고려대학교 컴퓨터공학과  
(Dept. of Computer Science & Engineering, Korea Univ.)

접수일자: 2010년4월5일, 수정완료일: 2010년4월30일

로운 인터넷 정보 유통 채널로 급부상하였으며, 현재 싸이월드, 네이버, 구글(Google), 플리커(Flickr) 등 국내외 주요 인터넷 기업들이 이를 시행하고 있다<sup>[2]</sup>.

하지만 태그의 이렇게 화려한 장점의 이면에는, 스팸 태그나 태깅 오남용으로 인한 쓰레기 정보 홍수(deluge of spam data) 등의 다양한 부작용이 있으며, 콘텐츠마다 직접 일일이 태그를 달아야 하는 불편함으로 인하여 일반적으로 사용자들이 잘 사용하지 않아 현재 태그 서비스의 유용성 논란이 발생하고 있는 실정이다<sup>[5]</sup>.

만약 사용자가 올린 콘텐츠를 분석하여 이에 맞는 태그를 자동으로 입력해준다면 부적합하거나 불필요한 태그의 입력을 막을 수 있으며, 사용자는 직접 태그를 입력하지 않고도 태그의 정보 분류 효과를 누릴 수 있게 될 것이다. 이를 간파한 국내외의 몇몇 소셜네트워크서비스(Social Network Service, SNS) 업체에서는 얼굴 영역과 해당 인물에 관한 정보(이름, 홈페이지 주소)를 연결하여 주는 태깅 서비스를 선보였다. 하지만 이 방법 또한 사용자가 모든 태그를 직접 입력해야 하는 불편함으로 인해 많은 호응을 얻지 못하고 있다.

최근에는 태그를 자동으로 추출하는 서비스들이 많이 등장하고 있으며, 이러한 자동화된 태깅 서비스들은 텍스트 기반의 문자 패턴을 분석하여 빈도수가 많은 검색어를 태그로 추출하거나 텍스트 자체에 대한 연관된 정보들을 메뉴화 하여 자동으로 관련이 있는 정보를 출력하여 주는 등 더욱 지능화되어지고 있는 추세이다<sup>[6]</sup>. 또한 태그구름(tag cloud)과 같은 개념을 도입하여 태그들의 집합을 빈도수에 따라 색상이나 굵기, 크기 등과

같은 요소들을 각기 달리 나타냄으로써 보다 시각적인 서비스를 제공하기도 한다<sup>[7]</sup>.

이밖에도 웹 자원의 지리적인 메타데이터를 태그형태로 추가하는 기법인 위치태그(geometric tag)<sup>[8]</sup>, 공유된 콘텐츠에 다수의 사용자들이 키워드의 형태로 메타데이터를 추가하는 협력태깅(collaborative tagging)<sup>[9]</sup>, 텍스트 분석을 통해 자동으로 태그를 생성하는 자동태깅(automatic tagging)<sup>[10]</sup> 등 태깅 기술을 응용하고 확장하기 위해 다양한 기법들이 시도되고 있다.

본 논문에서는 이처럼 태그가 가지고 있는 문제들을 해결하고, SNS 사용자들이 부담 없이 태그를 활용할 수 있도록 하기 위해 얼굴인식 기반의 태깅 자동화 시스템을 제안한다. 제안하는 시스템은 사진 속 다수의 얼굴을 자동으로 검출한 후 인식하고, 각각의 얼굴에 해당하는 태그를 입력해주는 시스템으로, 기존 태그의 단점을 보완할 뿐 아니라 얼굴인식이라는 재미 요소를 가미함으로써 SNS 이용자들의 적극적인 참여를 유도할 수 있다는 장점이 있다. 또한 엔터테인먼트<sup>[3]</sup>나 마케팅 산업에서 사용할 경우 추출된 태그 정보를 가공하여 소비자와의 자연스러운 접점을 제공해주는 새로운 비즈니스 모델을 창출해 낼 수도 있을 것이다.

이 시스템의 핵심 기술인 얼굴인식 기술은 사진이나 동영상으로부터 실시간으로 얼굴을 검출하고 인식하는 기술로 사진에서의 얼굴 유무 판별 및 동영상 내에서의 실시간 자동 얼굴 검출 등에 활용된다<sup>[11]</sup>. 또한 얼굴인식 기술은 생체정보 획득 방법에 대한 강제성이 적어 다른 생체인식 기술들이 사용자에게 줄 수 있는 거부감

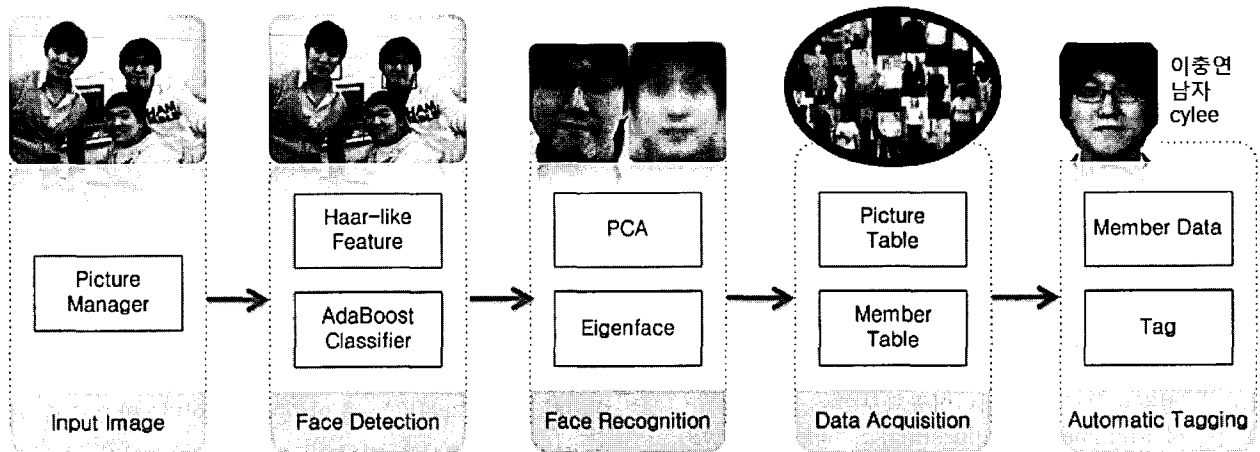


그림 1. 시스템 개괄도  
Fig. 1. System overview.

이나 불편함이 존재하지 않는다는 특성이 있다<sup>[12]</sup>.

본 연구에서는 입력된 이미지로부터 실시간으로 얼굴을 검출하고 이를 인식하여 자동으로 태그를 입력하는 시스템을 그림 1과 같이 개발하였다. 이 시스템은 사진첩 모듈에 얼굴인식 기능을 탑재하여 이미지에 나타난 얼굴 영역을 검출하여 표시해주고, 자동으로 또는 사용자가 이를 클릭할 경우 데이터베이스에 저장된 프로필 사진과 대조하여 인식함으로써 프로필 출력 및 태그 입력이 이루어지도록 구현하였다.

이를 위해 Haar-like feature와 AdaBoost 알고리즘을 이용하여 얼굴영상을 학습하고, 주성분분석 알고리즘을 통해 추출한 고유얼굴(Eigenface)을 얼굴인식에 이용한다. 또한 구현하는 시스템은 플랫폼의 제약을 최소화하고, 데이터베이스를 사용하도록 함으로써 향후 웹이나 모바일과의 호환이 용이하도록 한다.

## II. 본 론

### 1. 얼굴 검출

입력된 영상 콘텐츠에는 사람의 얼굴과 얼굴이 아닌 배경 및 객체 영상이 포함되어 있다. 따라서 태깅을 위한 얼굴인식 이전에 이미지에서 얼굴 영역만을 검출해내는 작업이 선행되어야 한다. 본 논문에서는 아다부스트(Adaptive Boosting, 이하 AdaBoost) 알고리즘과 Haar-like 특징(Haar-like features)을 이용하여 입력된 이미지를 빠르고 정확하게 처리할 수 있도록 하며, 시스템에서 다중 얼굴 검출을 실시간으로 가능하게 하였다.

Viola와 Jones<sup>[13]</sup>가 제안한 AdaBoost를 이용한 얼굴 검출 방법은 Haar-like 특징값을 적분이미지를 통해 빠른 속도로 계산하여 얼굴 영역을 검출한다. 여기서는 학습과정에서 얼굴영상과 비얼굴영상을 입력받아 둘의 차이를 가장 잘 나타내주는 특징들을 선택하고, AdaBoost를 사용하여 이러한 특징들을 선형적으로 결합함으로써 최종적인 분류기를 생성해낸다. 이 방법은 적은 양의 훈련 데이터(training data)로도 학습이 가능하며, 적분이미지를 활용하여 실행 효율이 증대된다.

#### 가. Haar-like Features

얼굴 영상은 조명이나 포즈, 배경 등의 다양한 외부 환경 요소에 영향을 받기 때문에 이와 같은 환경 요소에 강인하면서도 빠른 연산이 가능한 얼굴 검출 방법이 요구된다. 이러한 문제를 해결하기 위해 그림 2와 같이

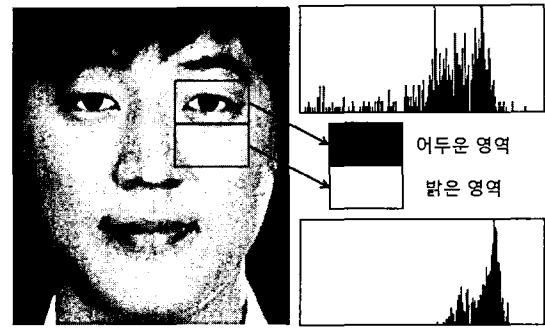


그림 2. Haar-like feature 적용 예  
Fig. 2. Example of Haar-like feature.

간단하고 객체 간의 구별 능력이 있는 Haar-like features를 사용한다.

Haar 웨이블릿(wavelet)은 1910년 Haar<sup>[14]</sup>가 제안한 간단한 직교 웨이블릿으로, 영상에서의 수평, 수직, 대각 방향에 대한 에지(edge)를 각 스케일별로 분해하는 기법이다. Papageorgiou<sup>[15]</sup>는 Haar 웨이블릿을 이용하여 물체를 검출하는 알고리즘을 제안하였으며, Viola와 Jones<sup>[13]</sup>는 이것을 확장한 Haar-like features를 이용하여 얼굴검출에 활용하였다.

본 논문에서는 기울어진 얼굴에 대한 검출 효율을 높이기 위해 Viola와 Jones의 Haar-like features를 확장하여 45도 기울어진 특징들을 추가한 Lienhart<sup>[16]</sup>의 Haar-like features를 사용하였다. 그림 3에서 볼 수 있는 것처럼 Lienhart가 제안한 확장된 Haar-like features는 Viola의 기존 대각선 feature가 제거되고, 대신 에지와 라인, 중심 특성에 각각 대각선 특성이 추가되었다.

Haar-like features는 그림 3과 같이 서로 인접한 사각형으로 구성되어 있으며, 검은색과 흰색으로 표시된

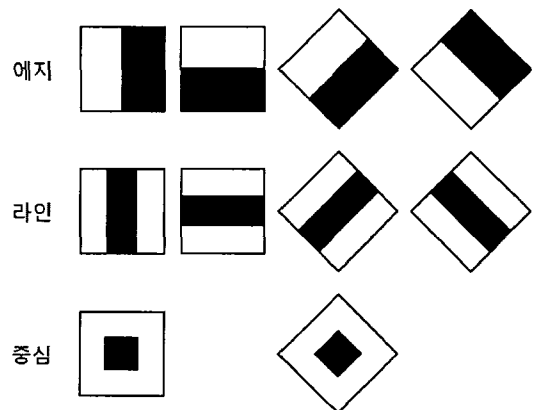


그림 3. Lienhart가 확장한 Haar-like features  
Fig. 3. The extended Haar-like features of Lienhart.

각 영역 안에 위치한 픽셀들의 값을 더하여 영역의 합을 구한 뒤, 그 값들에 가중치를 곱한 값의 합을 구하여 특징값을 계산할 수 있다. 즉, 흰색 영역과 검은색 영역의 명암 차다. 이때 Haar-like features는 직사각형 형태를 하고 있기 때문에, 적분이미지를 이용하여 빠른 속도로 계산이 가능하다.

나. 적분이미지(Integral Image) 적용

입력된 영상에서 Haar-like features 영역에 대한 픽셀 값의 합을 계산할 때 중복 연산을 최소화하기 위해 적분이미지를 사용한다. 계산된 각 영역의 픽셀값 합은 영역합테이블(Summed Area Table, SAT)<sup>[17]</sup>에 저장하여 참조된다.

이미지 상의 한 픽셀  $P(x,y)$ 까지의 영역에 대한 적분이미지값  $i$ 는 그림 4(a)에서 표현된 것과 같이 좌측 상단의 시작점으로부터  $P(x,y)$  위치까지의 모든 픽셀 값들의 합으로, 다음 식 (1)에 의해 계산되고, 식 (2)에 의해 갱신된다.

$$i(x, y) = \sum_{x' \leq x, y' \leq y} P(x', y') \quad (1)$$

$$i(x, y) = i(x, y) + i(x-1, y) + i(x, y-1) - i(x-1, y-1) \quad (2)$$

적분이미지를 이용하면 그림 4(b)의 D 영역과 같이 이미지의 중간에 놓여있는 임의의 영역에 포함된 픽셀 값의 합을 빠르게 구할 수 있다. 점  $\alpha$ 에서의 적분이미지값을 A영역의 각 픽셀값을 모두 합한 값이라고 하고, 점  $\beta$ 에서의 적분이미지값은 A영역과 B영역의 합, 마찬가지로 점  $\gamma$ 에서의 적분이미지값은 A영역과 C영역의 합, 점  $\delta$ 에서의 적분이미지값은 A, B, C, D 영역 모두의 픽셀값의 합이라고 할 때, D영역의 픽셀값의 합,  $S(D)$ 는 다음 식 (3)과 같이 적분이미지를 이용하여 구할 수 있다.

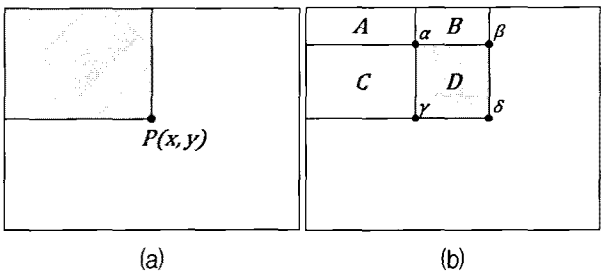


그림 4. 적분이미지를 이용한 계산  
Fig. 4. Examples of integral image.

$$S(D) = i(x_\delta, y_\delta) - i(x_\beta, y_\beta) - i(x_\gamma, y_\gamma) + i(x_\alpha, y_\alpha) \quad (3)$$

이와 마찬가지로 Haar-like features는 이웃한 두 개 이상의 직사각형으로 이미지 상에 놓이기 때문에, 적분 이미지를 이용하면 빠른 속도로 계산이 가능하다. 두 개의 직사각형으로 이루어진 특징은 6개 점의 적분이미지로 구할 수 있으며, 세 개의 직사각형으로 이루어진 특징은 8개 점, 네 개의 직사각형으로 이루어진 특징은 9개 점의 적분이미지를 이용하여 구할 수 있다.

다. 얼굴검출 특징값 계산

얼굴검출을 위한 Haar-like 특징값은 입력 영상에서 Haar-like feature 마스크를 이동시키며 구한다. 마스크를 구성하는 사각형 성분들을  $r_m$ 이라고 하고, 사각형에 포함된 모든 픽셀값의 합을  $S(r_m)$ 이라 할 때,  $M$ 번째 마스크에 대한 특징값  $f_M$ 을 식(4)와 같이 구할 수 있다.

$$f_M = \sum_{m=1}^N [\omega_m \times S(r_m)] \quad (4)$$

단,  $M$ 은 마스크 인덱스이고,  $m$ 은 마스크 성분 인덱스

이때 두 성분의 면적비가 다를 경우 이를 보상해주기 위한 가중치를 유도해야 한다.  $r_m$ 의 가중치  $\omega_m$ 은 이웃한 마스크 성분의 가중치  $\omega_{m'}$ 와 반대되는 부호를 갖게 되며, 식 (5)와 같이 나타낼 수 있다.

$$-\omega_m \times A(r_m) = \omega_{m'} \times A(r_{m'}) \quad (5)$$

단,  $A(r_m)$ 은 마스크 성분에 대한 면적임.

예를 들면 그림 5와 같은 라인(line) 특성을 가지는 Haar-like feature는 성분  $r_2$ 와 성분  $r_1$ ( $r_2$ 를 포함한 전체영역)으로부터 특징값을 구할 수 있다.  $r_1$ 의 좌상단

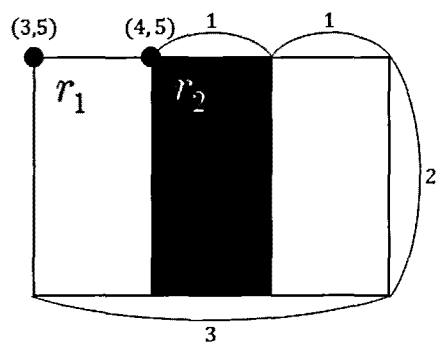


그림 5. Line 특성 feature에서의 Haar-like 특징값 계산  
Fig. 5. Calculation of Haar-like feature value.

좌표가 (3, 5),  $r_2$ 의 좌상단 좌표가 (4, 5)이고,  $r_1$ 과  $r_2$ 의 너비는 각각 3과 1이며, 높이는 둘 다 2라고 할 때, 각 영역의 넓이비는 6:2, 즉 가중치는 3과 -1이 되며, 이후  $S(r_1)$ ,  $S(r_2)$ 를 이용하여 특징값  $f$ 를 계산할 수 있다.

라. 얼굴영역 분류기(classifier) 학습

Haar-like 특징 추출 단계에서 생성된 얼굴검출에 대한 특징값을 이용하여 얼굴 영역을 분류해내기 위해 본 논문에서는 AdaBoost 알고리즘과 캐스케이드(Cascade) 구조를 이용하였다. AdaBoost 알고리즘은 약분류기(weak classifier)들의 선형 결합을 통하여 최종적으로 높은 검출 성능을 가지는 강분류기(strong classifier)를 생성하는 기법으로, Freund와 Schapire<sup>[18]</sup>에 의해 소개되었으며, 약분류기 수가 증가할수록 에러율이 지수적으로 영점에 근접함이 증명되었다.

이전 단계에서 구한 얼굴검출 특징계수를 이용한 약분류기는 식(6)과 같이 나타낼 수 있다.

$$h_M = \begin{cases} 1 & \text{if } R_M f_M(P) < R_M \theta_M \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

단,  $f_M$ 은 M번째 마스크에서의 특징값이고,  $\theta_M$ 은 문턱값이며,  $R_i$ 은 패리티(parity)임.

약분류기는 마스크 내의 Haar-like 특징(feature)들 중 얼굴과 배경을 가장 잘 분류하는 하나의 특징을 선별하며, 이를 위해 적분이미지를 통해 구한 픽셀값의 합을 이용하여 에러가 최소화되는 문턱값(threshold)을 찾는다. 이렇게 구한 약분류기들은 AdaBoost 학습 알고리즘에 의해 선형적으로 결합되어 최종적인 강분류기를 구성하게 된다. 얼굴 검출을 위한 AdaBoost 알고리즘은 간단하게 표 1과 같이 요약된다.

학습 영상은 얼굴 영상과 얼굴이 아닌 영상으로 구분되어 입력된다.  $P_i$ 는 각각의 훈련 영상이며,  $Y_i$ 는 그 영상이 얼굴 영상인지 아닌지를 나타내주는 인덱스값이다. 알고리즘 시작 단계에서는 우선 가중치를 초기화한 후, 약분류기의 개수만큼 반복하면서 에러율이 가장 작은 가중치를 찾는다. 학습 초기에는 큰 특징들을 통해 얼굴 영상의 에러를 줄이고, 뒤로 갈수록 세세한 특징들을 통해 얼굴과 비얼굴을 구별시키는 특징을 판별한다. 따라서 계층의 앞부분은 크기가 크고 특징의 수가 적은 반면, 뒷부분은 크기가 작고 특징의 수는 많게 된다. 이렇게 가중치가 조정된 약분류기들이 선형적으로

표 1. 얼굴 검출을 위한 AdaBoost 알고리즘<sup>[13]</sup>  
Table 1. Algorithm of AdaBoost for face detection.

<p>① 학습이미지 입력: <math>(P_1, Y_1), \dots, (P_N, Y_N)</math>                  단, <math>P_i</math>: 입력영상,  <math>Y_i</math>: 입력영상이 얼굴일 때 1, 얼굴이 아닐 때 0</p> <p>② <math>Y_i</math> 각각에 부여할 가중치를 다음과 같이 초기화  <math>w_{t,i} = \left\{ \begin{matrix} 1 &amp; 1 \\ 2m, &amp; 2l \end{matrix} \right\}</math>                  단, m: 얼굴이 아닌 이미지 수, n: 얼굴 이미지 수</p> <p>③ 결합할 약분류기의 수만큼 다음 작업을 반복</p> <p>▶ 가중치 정규화: <math>w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}</math>                  단, <math>w_{t,i}</math>는 확률분포</p> <p>▶ 특징점 <math>j</math>에 대한 약분류기 <math>h_j</math>를 학습시키며, 에러값을 계산  <math display="block">\epsilon_j = \sum_i w_i  h_j(P_i) - Y_i </math></p> <p>▶ 가장 작은 에러값(<math>\epsilon_t</math>)을 가지는 분류기 <math>h_t</math>를 선택</p> <p>▶ 가중치를 다음과 같이 갱신  <math display="block">w_{t+1,i} = w_{t,i} \beta_t^{1-e_i} \quad \left( \beta_t = \frac{\epsilon_t}{1-\epsilon_t} \right)</math></p> <p>※ 학습영상 <math>P_i</math>가 잘 분류되었다면 <math>e_i = 0</math>이고, 그 이외에는 <math>e_i = 1</math></p> <p>④ 최종 결합된 강분류기는 다음과 같다.  <math display="block">h(P) = \begin{cases} 1 &amp; \sum_{t=1}^T \alpha_t h_t(P) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ \left( \alpha_t = \log \frac{1-\epsilon_t}{\epsilon_t} \right) \\ 0 &amp; \text{otherwise} \end{cases}</math></p>
--

결합되면서 최종적으로 강분류기를 생성하게 된다.

마. Cascade 구조를 이용한 분류 속도 향상

AdaBoost 알고리즘이 Neural Network나 SVM과 같은 기존의 분류 알고리즘보다 좋은 성능을 보이기 위해서는 많은 수의 훈련 영상과 약분류기들이 필요하며, 이 경우 훈련시간과 검출시간이 늘어나게 된다. Viola는 이러한 계산시간을 줄이기 위해 Cascade 구조를 이용하여 분류기를 구성하였다<sup>[13]</sup>.

이 방법의 원리는 초기 단계에서 얼굴영상을 유지하면서 비얼굴영상들을 걸러냄으로써 다음 단계로 갈수록 좀 더 복잡하고 구별하기 힘든 특징들을 분류하는데 집중하는 것이다. 이를 위해 단계별로 50% 정도의 오판율(false positive rate, 이하 FPR)을 감수하며, 100%에 가까운 얼굴 검출률(detection rate, 이하 DR)을 갖도록 한다. 이 정도의 높은 FPR 자체는 분류기로서 사용하기에 적합하지 않지만, 많은 수의 후속 단계를 거침으로써 판별력과 효율이 증대되어, 최종적으로는 분류기로 사용하기 적합한 성능을 갖게 된다.

## 2. 얼굴 인식

본 논문에서는 주성분분석(Principal Component Analysis, PCA)<sup>[19]</sup>을 이용하여 얼굴인식을 수행하였다. PCA는 얼굴에서 구성요소들을 일일이 분리하지 않고 하나의 주성분으로 생각하여 처리하는 기법으로, 각 입력영상을 1차원 벡터로 변환한 후 전체 입력영상들의 직교적인 공분산 행렬(orthogonal covariance matrix)을 계산한다. 이후 공분산행렬의 고유값(eigen-value)에 따라 고유벡터(eigen-vector)를 구하고, 입력된 얼굴과 기준 얼굴의 고유벡터를 벡터 공간상에서 비교하여 가장 가까운 거리의 벡터를 지닌 얼굴이 인식되도록 한다.

### 가. 얼굴인식 과정

얼굴영상의 다차원의 정보는 얼굴을 표현하기에 적절하지 못하며, 얼굴을 묘사할 수 있는 얼굴 공간(face space)으로 표현 되어야 한다. 따라서 얼굴 인식에서의 PCA 기법은 고차원의 얼굴 정보들을 얼굴 공간의 저차원으로 축소시킴으로써 데이터를 다루기 쉽게 하기 위한 방법이라 할 수 있다.

PCA를 이용한 얼굴 인식은 크게 학습, 입력, 인식의 세 단계로 구분된다. 학습단계는 입력 영상과 비교할 수 있는 학습영상 집합으로부터 평균 얼굴 벡터를 계산한 후 각 학습 얼굴 벡터와의 차를 구하여 공분산 행렬을 통해 고유값과 고유벡터를 계산한다. 이 값이 구해지면 고유벡터를 크기순으로 재배열하여 얼굴 모양과 비슷한 고유얼굴을 생성하며 학습영상을 아주 작은 고유값에 대응하는 고유얼굴을 제외한 학습영상의 특징을 나타내는 특징 벡터를 생성한다.

입력 단계에서는 학습 단계에서 생성된 평균 얼굴 벡터와 고유얼굴을 사용한다. 학습 단계와 마찬가지로 평균얼굴의 차영상을 고유얼굴 공간에 투사시켜 입력 영

상의 특징 벡터의 값을 추출한다.

마지막으로 인식 단계에서는 학습 영상의 특징 벡터와의 유클리디안 거리(euclidean distance)를 비교하여 그 거리가 최소가 되는 얼굴이 입력 얼굴과 유사한 특징을 가진 얼굴이므로 이 영상을 입력 얼굴과 동일한 인물로 결정한다.

### 나. 고유얼굴 생성

학습할 얼굴영상들을 하나의 벡터로 표현하기 위해 얼굴영상의 각 픽셀을 스캔 방식으로 정의한다. 각각의 얼굴 영상의 크기가 W×H일 때 각 픽셀 정보를 하나의 행으로 연결하여 벡터성분으로 표현하면 그림 6과 같이 생성된다.

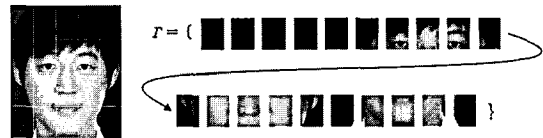


그림 6. 얼굴영상 벡터  
Fig. 6. Face image vector.

이렇게 생성된 각 얼굴영상 벡터  $\Gamma$ 가 N개의 성분을 가지고, 학습 얼굴 영상의 수가 M개 일 때, 학습에 필요한 모든 얼굴영상들을  $M \times N$ 의 크기를 가지는 하나의 행렬 S로 만들 수 있다.

얼굴영상 행렬 S로부터 주성분을 이용하여 고차원의 자료를 저차원으로 변환함으로써 원 자료의 복잡성을 제거할 수 있게 하는 하나의 절차이며 공분산 행렬 및 주성분 분석을 위한 과정은 다음과 같이 수행된다.

- ① 이미지 정규화 과정을 거친 후 얼굴 벡터의 집합으로부터 평균 얼굴 벡터(그림 7)를 계산한다.

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \tag{7}$$

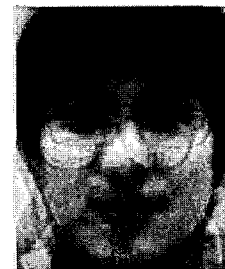


그림 7. 평균 얼굴 벡터  
Fig. 7. Mean vector of faces.

② 각 학습 얼굴 벡터와 평균 얼굴 벡터의 차를 계산한다.

$$\Phi_i = \Gamma_i - \Psi \quad (8)$$

③ 차벡터  $\phi_i$ 로부터 공분산 행렬을 계산한다.

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (9)$$

④ 공분산 행렬로부터  $M$ 개의 고유값  $\lambda_i$ 와 이에 대응하는 고유벡터  $u_i$ 를 계산한다.

$$Cu_i = \lambda_i u_i \quad (\text{단 } u_i \neq 0) \quad (10)$$

공분산 행렬에서 구한 고유값은 분산의 정도를 나타내며, 이에 대응하는 고유벡터를  $W \times H$ 로 재배열하면 얼굴의 모양과 닮아 보이므로 이를 고유얼굴(Eigenface)이라고 한다<sup>[20]</sup>. 고유얼굴의 개수는 학습영상 개수와 동일하며 고유얼굴을 생성하는 식은 다음과 같다.

$$u_i = \frac{1}{M} \sum_{k=1}^M \lambda_{ik} \Phi_k \quad l = 1, 2, \dots, M \quad (11)$$

고유얼굴(그림 8)은 얼굴 공간을 구성하는 벡터에서 학습 영상의 공통적인 특징을 가장 잘 나타내는 기저 벡터를 의미하며 각 얼굴 벡터와 평균 얼굴 벡터의 차 벡터로 공분산 행렬의 고유벡터에 해당한다.



그림 8. 고유얼굴  
Fig. 8. Eigenfaces.

#### 다. 고유성분벡터 추출

공분산 행렬의 고유치는 평균얼굴 영상에 대한 분산의 정도를 나타내기 때문에 가장 큰 고유치에 대응하는

고유벡터로 이루어진 고유얼굴이 가장 비슷한 얼굴이 되며, 고유치가 작을수록 얼굴의 특징이 작아진다. 본 연구에서는 큰 고유치에 해당하는  $M$ 개의 학습 얼굴영상에 대해서만 고유벡터로 사용하며 입력영상의 성분벡터 비교를 위해 각각의 고유얼굴 성분 벡터<sup>[21]</sup>를 생성한다.

$$w_k = u_k^T (\Gamma - \Psi) \quad k = 1, 2, \dots, M \quad (12)$$

$$\Omega^T = [w_1, w_2, \dots, w_M] \quad (13)$$

#### 라. 얼굴 인식

$\Omega_k$  값이 구해지면 학습 얼굴영상의 고유얼굴상에서의 성분벡터와 입력영상의 성분벡터와의 유클리디언 거리를 비교하여, 그 거리가 최소가 되는 얼굴이 입력된 얼굴과 가장 유사한 얼굴이라고 판단하여, 최종적으로 이 영상 후보를 인식 결과로 결정한다.

### III. 연구결과 및 토의

#### 1. 실험 환경

제안한 다중 얼굴 태깅 자동화 시스템은 AMD Athlon X2 (2.70GHz CPU), 2GB RAM, ATI Radeon HD3400 (VGA) 사양의 환경에서 실험하였다.

실험에 사용된 코드는 Microsoft Windows XP Professional 운영체제에서 C# 언어와 OpenCV 라이브러리를 사용하여 컴파일(compile)하였다. 입력영상으로 사용된 인물 사진들은 디지털카메라(Nikon D80)를 사용하여 얼굴 기울어짐에 상관없이 1936×1296의 해상도로 실내에서 촬영되었으며, 이후 프로필 사진은 240×200, 그룹 사진은 800×600으로 크기를 조정하였다.

또한, 향후 웹서비스로의 전환이 용이하도록 Microsoft의 닷넷 프레임워크를 플랫폼으로 사용하였으며, SNS 환경과 최대한 유사하게 제작하기 위해 모든 프로필 정보와 영상 콘텐츠들을 각각 독립적으로 구축된 데이터베이스에 저장하여 사용하였다.

#### 2. 자동 태깅 시스템

제안한 다중 얼굴 태깅 자동화 시스템을 구현하기 위해 먼저 20대 남녀 10명의 프로필 사진과 각각의 이름, ID, 이메일 주소를 기록하는 사용자 정보 테이블(member-table), 그리고 사용자들이 촬영된 단체사진들



그림 9. 시스템 인터페이스 및 결과 화면  
 Fig. 9. System Interface and experimental result.

과 각 사진의 제목, 태그를 기록하는 사진 테이블 (picture table)로 구성된 데이터베이스를 구축하였다. 이때 두 테이블은 서로 독립적으로 어떠한 매개체도 가지고 있지 않다.

이후 자체 제작한 SNS의 사진 관리 시스템 모형을 개발하였다 (그림 9). 기 구축한 데이터베이스와 연동하게 하여 사진 테이블에 기록된 사진 정보에 자동으로 태깅할 수 있도록 하였다. 사용자는 시스템을 통해 새로운 프로필이나 사진을 데이터베이스에 등록 또는 수정하거나 삭제하는 것이 가능하며, 아이디를 입력하여 사용자 정보를 검색할 수 있다.

사용자가 활성화버튼(Activate)을 눌러 얼굴 검출 및 인식을 활성화한 후, 마우스 커서를 사진 속 인물의 얼굴에 올리면 검출된 얼굴 영역이 사각형 박스로 표시된다. 이후 얼굴을 클릭하면 해당 인물의 프로필 사진 및 정보가 출력되고, 태그 목록에는 해당 인물의 이름이 태깅된다. 이때 입력되는 태그는 사용자가 입력하는 것이 아니라, 시스템이 얼굴을 인식하여 자동으로 출력하는 것이다.

3. 성능 평가

제작된 시스템의 성능을 평가하기 위해 얼굴 검출시 소요되는 시간과 검출률, 얼굴영상 학습시간, 그리고 얼굴 인식시 소요되는 시간과 인식률을 측정하였다.

가. 얼굴 검출시간

먼저 실험에 사용된 30장의 단체사진을 이용하여 총 95명의 얼굴을 검출한 시간은 그림 10과 같다. 평균 검출시간은 366.69ms로 매우 양호한 결과를 보였다.

average detection time : 366.96 ms

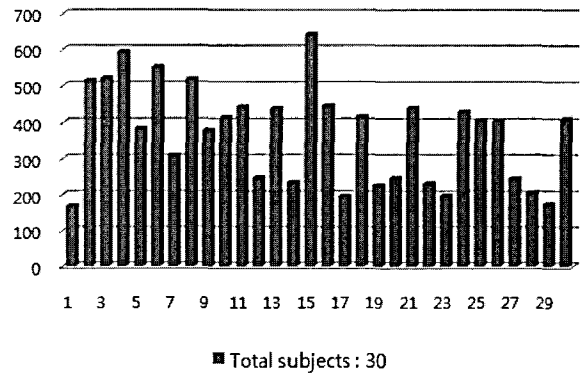


그림 10. 얼굴 검출시간  
 Fig. 10. Average detection time.

나. 얼굴 검출률

30장의 단체사진에서의 95명의 얼굴에 대한 검출률은 그림 11과 같은 결과를 보였다. 조명변화와 다양한 촬영 각도·얼굴 크기·피부색, 그리고 안경착용·얼굴가림 등과 같은 환경적 요인에도 불구하고 평균 90.8%의 높은 검출률을 보였다.

average detection rate : 90.8%

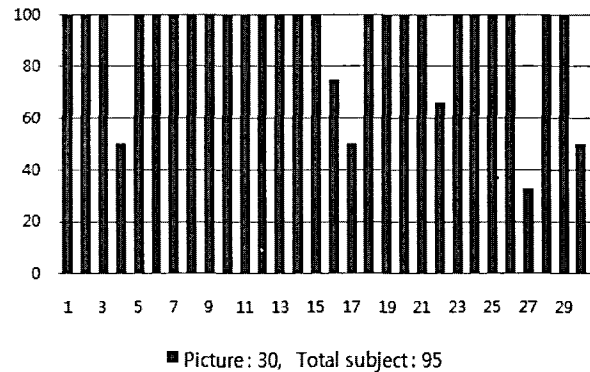


그림 11. 얼굴 검출률  
 Fig. 11. Average detection rate.

다. 얼굴 학습시간

얼굴 인식 단계 이전에 입력이미지와 비교를 위한 학습 데이터가 필요하므로 10장의 얼굴사진을 학습하였다. 3~10명에 대한 각각의 학습시간은 그림 12와 같으며 학습시킬 이미지의 양이 늘어날수록 평균적으로 시간이 증가하는 것을 확인할 수 있다.



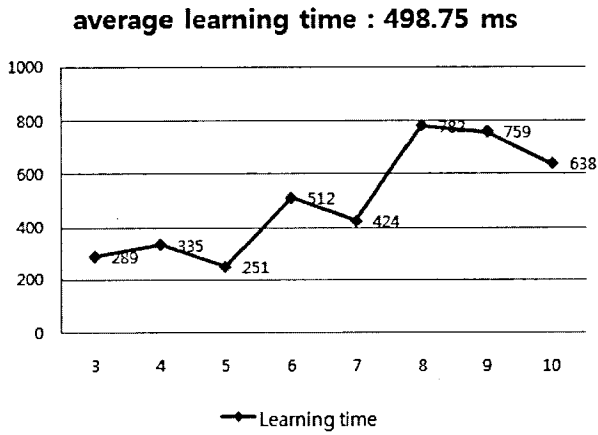


그림 12. 얼굴 학습시간  
Fig. 12. Average learning time.

라. 얼굴 인식시간 및 인식률

학습된 데이터를 대상으로 한 입력영상에 대한 인식 시간과 인식률은 각각 그림 13과 그림 14에 나타난 바와 같다. 8명의 얼굴 학습 데이터를 이용하여 10장의 단체사진 이미지에서 각각 2~4명씩, 총 21명의 얼굴을

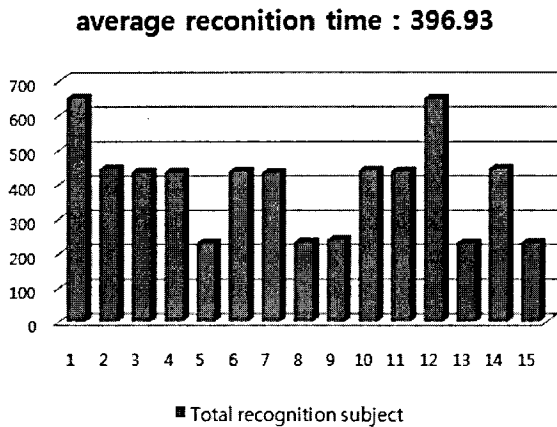


그림 13. 얼굴 인식시간  
Fig. 13. Average recognition time.

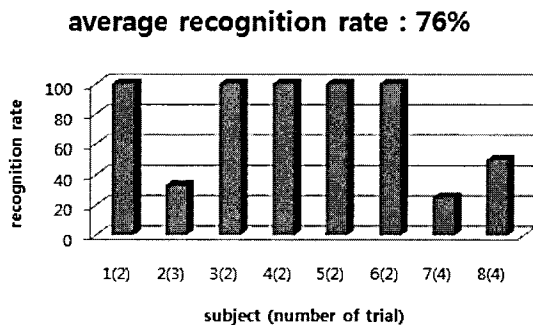


그림 14. 얼굴 인식률  
Fig. 14. Average recognition rate.

인식 테스트한 결과 평균 396.93ms의 인식시간과 평균 76%의 인식률을 보였다. 30~50%의 인식률을 보이는 이미지에 대해서는 조명변화나 얼굴의 기울어짐 정도가 심한 것으로 확인되었고, 학습 데이터 수를 늘릴 때마다 얼굴 인식률이 저하되는 현상을 발견할 수 있었다. 향후 이러한 문제에 대한 정량적인 분석 및 추가 실험을 통해 성능 저하의 정확한 원인을 파악하고, 이러한 환경적인 변화에 강인한 알고리즘에 대해 지속적인 연구가 필요할 것이다<sup>[4]</sup>.

IV. 결 론

본 논문에서는 텍스트 기반의 수동적인 기존 태깅 시스템과 차별되는 태깅 자동화 방법에 대해 제안하였다.

얼굴검출 과정에서는 Haar-like 특징과 AdaBoost를 이용하여 높은 검출률과 빠른 속도를 가진 분류기를 생성했으며, 인식 과정에서는 PCA를 이용하여 고유얼굴을 생성한 후, 여기서 추출된 고유벡터 성분을 비교하여 오차가 가장 적은 얼굴을 선택하여 인식하였다.

이는 웹2.0을 지나 웹3.0으로 도약하는 현 인터넷 시장에 인공지능 알고리즘을 적용하여 새로운 형태의 웹 서비스를 제공하고, 자동화된 얼굴인식을 통해 사용자로 하여금 흥미롭고 편리한 인터넷 콘텐츠 이용을 가능하게 하는데 큰 의의가 있으며, 또한 인맥 네트워크를 중시하는 SNS에서, 사용자가 더욱 넓은 인맥을 구축하거나, 잃어버린 인맥을 복구할 수 있는 환경을 제공해 줄 것이다. 2006년 마이스페이스에 이어, 2007년 페이스북, 2008년 트위터(Twitter)와 믹시(Mixi), 그리고 지금도 끊임없이 생겨나고 있는 다양한 SNS 및 웹서비스들에 대해 세계의 관심이 증가하는 가운데, 이러한 사용자 중심의 독창적인 서비스는 기술적으로 정제되어 있는 국내 인터넷 시장에 활기를 불어넣어주고, 유망 기술로 떠오르고 있는 패턴인식 분야의 연구 수준을 향상시킬 것으로 기대된다.

※ 본고는 2009년도 대한전자공학회 추계종합학술대회 발표 논문의 확장판임.

참 고 문 헌

[1] 박종진, “인터넷 영상 소비 패턴과 업계 주도권 전망: SBSi의 온라인 디지털 콘텐츠 사업전략,” 삼성

- 경제연구소 디지털사례세미나, 2006년 5월
- [2] 한국소프트웨어진흥원, “포털사이트를 중심으로 한 국내 웹2.0 서비스 현황과 전망,” *SW산업동향*, 2006년
- [3] 이혜규, 박성한, “개인화된 방송 콘텐츠의 효율적 검색을 위한 메타데이터 검색 구조 설계,” *전자공학회논문지*, 제46권 CI편, 제2호, 100-105쪽, 2009년 3월
- [4] 조현중, 강민구, 문승빈, “조명 변화 환경에서 PCA 기반 얼굴인식 알고리즘의 신뢰도에 대한 연구,” *전자공학회논문지*, 제46권 CI편, 제2호, 19-26쪽, 2009년 3월
- [5] S. Bateman, C. Brooks and G. McCalla, “Collaborative Tagging Approaches for Ontological Metadata in Adaptive E-Learning Systems,” in *Proc. of 4th Int'l Workshop on Applications of Semantic Web Technologies for e-Learning*, pp.3-12, June 2006.
- [6] T. Rattenbury, N. Good and M. Naaman, “Towards Extracting Flickr Tag Semantics,” in *Proc. of 16th Int'l Conf. of World Wide Web*, pp.1287-1288, 2007.
- [7] M. J. Halvey and M. T. Keane, “An Assessment of Tag Presentation Techniques,” in *Proc. of 16th Int'l Conf. on World Wide Web*, pp.1313-1314, 2007.
- [8] A. Pyalling, M. Maslov and P. Branslavski, “Automatic Geotagging of Russian Web Sites,” in *Proc. of 15th Int'l Conf. on World Wide Web*, pp.965-966, 2006.
- [9] S. A. Golder and B. A. Huberman, “Usage Patterns of Collaborative Tagging Systems,” *Journal of Information Science*, Vol. 32, No. 2, pp. 198-208, 2006.
- [10] P. A. Chirita, S. Costache, W. Nejdl and S. Handschuh, “P-TAG: large scale automatic generation of personalized annotation tags for the web,” in *Proc. of 16th Int'l Conf. on World Wide Web*, 845-854, 2007.
- [11] R. Chellappa, C. L. Wilson, and S. Sirohey, “Human and Machine Recognition of Faces: A Survey,” *Proceedings of IEEE*, Vol. 83, No. 5, pp. 705-741, May 1995.
- [12] S. Jahanbin, H. Choi, A. C. Bovik and K. R. Castleman, “Three Dimensional Face Recognition using Wavelet Decomposition of Range Images,” in *Proc. of IEEE Conf. on Image Processing*, Vol. 1, pp. 145-148, September 2007.
- [13] P. Viola and M. Jones, “Robust Real-Time Face Detection,” *International Journal of Computer Vision*, Vol. 57, No. 2, pp. 137-154, May 2004.
- [14] A. Haar, “Zur Theorie der Orthogonalen Funktionen-Systeme,” *Mathematische Annalen*, Vol. 69, pp. 331-371, 1910.
- [15] C. P. Papageorgiou, M. Oren and T. Poggio, “A General Framework for Object Detection,” in *Proc. of IEEE Conf. on Computer Vision*, pp. 555-562, 1998.
- [16] R. Lienhart, A. Kuranov and V. Pisarevsky, “Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection,” *MRL Technical Report*, May 2002.
- [17] F. C. Crow, “Summed-area Tables for Texture Mapping,” in *Proc. of 11th Annual Conference on Computer Graphics and Interactive Techniques*, Vol. 18, No. 3, pp. 207-212, 1984.
- [18] Y. Freund and R. E. Schapire, “A Decision-Theoretic Generalization of On-line Learning and An Application to Boosting,” *Journal of Computer and System Sciences*, Vol. 55, No. 1, pp. 119-139, August 1997.
- [19] M. Turk and A. Pentland, “Eigenfaces for Recognition,” *Journal of Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71-86, 1991.
- [20] L. Sirovich and M. Kirby, “Low-dimensional procedure for the characterization of human faces”, *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, Vol. 4, No. 3, pp. 519-524, March 1987.
- [21] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 17, pp. 711-720, July 1997.

저 자 소 개



이 충 연(정회원)  
 2010년 성결대학교 멀티미디어  
 공학부 학사 졸업  
 2010년~현재 성결대학교  
 멀티미디어공학부  
 Vi2com Lab 연구원

<주관심분야 : 인공지능, BCI, 가상현실>



이 재 동(학생회원)  
 2010년 성결대학교 멀티미디어  
 공학과 학사 졸업  
 2010년~현재 고려대학교 컴퓨터  
 공학과 석사 과정  
 <주관심분야: 가상현실, HCI, HRI>



진 성 아(정회원)-교신저자  
 1999년 미국 Stevens Institute of  
 Technology, 컴퓨터과학과  
 박사 졸업  
 2000년~2001년 서강대학교 영상  
 대학원 미디어공학과 연구  
 교수

2007년~2008년 미국 Wayne State University  
 방문연구교수

2001년~현재 성결대학교 멀티미디어공학부  
 부교수, 산학협력단장

<주관심분야 : 얼굴모델링, 시각정보처리, BCI, 가상현실>