

에이다부스트와 신경망 조합을 이용한 표정인식

Facial Expression Recognition by Combining Adaboost and Neural Network Algorithms

홍용희 · 한영준 · 한현수*

Yonghee Hong, Youngjoon Han and Hernsoo Hahn*

* 숭실대학교 전자공학과

요 약

표정은 사람의 감정을 표현하는 대표적인 수단이다. 이러한 이유로 표정은 사람의 의도를 컴퓨터에 전하는데 효과적인 방법으로 사용될 수 있다. 본 논문에서는 2D 영상에서 사람의 표정을 보다 빠르고 정확하게 인식하기 위해 Discrete Adaboost 알고리즘과 신경망 알고리즘을 통합하는 방법을 제안한다. 1차로 Adaboost 알고리즘으로 영상에서 얼굴의 위치와 크기를 찾고, 2차로 표정별로 학습된 Adaboost 강분류기를 이용하여 표정별 출력 값을 얻으며, 이를 마지막으로 Adaboost 강분류기 값으로 학습된 신경망 알고리즘의 입력으로 이용하여 최종 표정을 인식한다. 제안하는 방법은 실시간이 보장된 Adaboost 알고리즘의 특성과 정확성을 개선하는 신경망 기반 인식기의 신뢰성을 적절히 활용함으로써 전체 인식기의 실시간성을 확보하면서도 정확성을 향상시킨다. 본 논문에서 구현된 알고리즘은 평온, 행복, 슬픔, 화남, 놀람의 5가지 표정에 대해 평균 86~95%의 정확도로 실시간 인식이 가능하다.

키워드 : 에이다부스트, 신경망, 표정 인식, 유사 하르 특징, 패턴 인식

Abstract

Human facial expression shows human's emotion most exactly, so it can be used as the most efficient tool for delivering human's intention to computer. For fast and exact recognition of human's facial expression on a 2D image, this paper proposes a new method which integrates an Discrete Adaboost classification algorithm and a neural network based recognition algorithm. In the first step, Adaboost algorithm finds the position and size of a face in the input image. Second, input detected face image into 5 Adaboost strong classifiers which have been trained for each facial expressions. Finally, neural network based recognition algorithm which has been trained with the outputs of Adaboost strong classifiers determines final facial expression result. The proposed algorithm guarantees the realtime and enhanced accuracy by utilizing fastness and accuracy of Adaboost classification algorithm and reliability of neural network based recognition algorithm. In this paper, the proposed algorithm recognizes five facial expressions such as neutral, happiness, sadness, anger and surprise and achieves 86~95% of accuracy depending on the expression types in real time.

Key Words : Adaboost, Neural Network, Facial Expression Recognition, Haar-like Feature, Pattern Classification

1. 서 론

사람과 컴퓨터 간의 의사소통을 사람과 사람 사이의 수준으로 개선하기 위해서 많은 연구들이 이루어져 왔다. 의사소통의 방법으로는 크게 몸짓, 생체 신호, 음성, 표정 등이 있다.

몸짓을 인식하여 감정을 인식하는 방법으로는 움직이는 신체 부위에 나타나는 부분특징, 전체적인 움직임에 나타나는 전체 특징을 다중 혼합 특징 정보로 얻어 저차원의 심볼로 구성하여 인식하는 방법이 있다[1].

생체 신호는 얼굴에 EMG(Electromyography) 센서를 부착하고 얼굴 근육의 움직임을 이용하여 표정을 인식하였다[2]. 하지만 얼굴에 직접 센서를 부착해야 하므로 불편하

며 자연스러운 표정에 방해가 되는 단점이 있다.

음성은 음성의 톤(Tone), 피치(Pitch) 등을 이용하는 방법이 연구되고 있다[3]. 음성 인식을 통한 사용자의 감정 파악은 지역마다 다른 어투, 개인마다 다른 목소리 톤과 피치와 같은 변수들 때문에 정확한 인식의 어려움이 있다[4].

표정은 사람의 감정에 따라 시각적으로 미묘하며 빠르게 여러 얼굴 근육의 움직임에 따라 상대방에게 시각적으로 표출되는 감정 표현의 일부분으로 이를 광학 센서를 이용하여 인식 하게 되면 다른 방법들에 비해 비접촉 방식으로 불편함이 없고, 몸짓과 같이 몸의 큰 움직임을 이용할 필요가 없어 다른 방법들에 비해 이용자 측면에서 편리한 측면이 있다. 하지만 광학 센서를 이용한 인식 방법은 조명 환경에 취약한 단점이 있다.

영상을 이용하여 표정을 인식하기 위해 많은 방법이 연구되어져 왔다. 여러 가지 방법들을 크게 모델 기반(Model Based)와 영상 기반(Image Based) 방법으로 나눌 수 있다

접수일자 : 2010년 6월 22일

완료일자 : 2010년 11월 6일

[5].

모델 기반 방법은 얼굴에서 특징점을 검출하고 얼굴 근육의 움직임을 파악하여 표정을 인식하는 방법으로 Lanitis[6]는 AAM(Active Appearance Models)를 이용하여 얼굴 근육 움직임을 변화와 조명 변화 정보를 이용하여 표정을 인식하였다. Cheon[7]는 AAM으로 얻어진 특징점들의 움직임을 분석하여 표정을 인식하였다. Bouchraf[8]는 특징점들 간의 거리를 인식하여 표정 인식을 수행하였다. Gao[9]는 Line-based Caricatures를 생성하여 LEM(Line Edge Map)과 비교하여 표정을 인식하였다.

영상 기반 방법은 얼굴 전체 영상의 밝기 값을 이용하여 표정을 인식하는 방법으로 Lisetti[10]는 얼굴 이미지 전체를 가버 웨이블릿(Gabor Wavelet)으로 구분하였다. Padgett[11]은 얼굴 이미지를 눈, 코, 입 영역을 나누어 각각 영역에 PCA(Principal Component Analysis)를 적용하여 얻은 특징 값을 이용하였다. Kim[12]은 각 표정별로 특화된 3x3 Haar-like 특징을 후보로 찾아 추가하고 Adaboost 알고리즘을 적용하여 표정을 인식했다. Yang[13]은 피라미드 가버특징(Pyramid Gabor feature)을 이용하여 기존 가버특징에 비해 빠르고 정확하게 특징을 얻어 CKFD(Complete Fisher Kernel Linear Discriminant) 분석을 통하여 표정을 인식하였다.

모델과 영상 기반 방법을 동시에 이용한 방법으로 Kotsia[14]가 DNMF(Discriminant Non-negative Matrix Factorization)을 이용하여 영상 기반 정보를 얻은 후 Grid tracking을 이용하여 SVM(Support Vector Machine)을 통해 모델 기반 정보를 얻어 표정을 구분하는 방법이 있다.

서로 다른 인식기를 조합한 하이브리드 방법으로 Padgett[15]가 PCA 프로젝션을 이용하여 얼굴 영상에서 주요 데이터를 얻은 후 신경망을 이용하여 표정을 구분하였으며, Lien[16]은 Dense Optical Flow를 이용해 얻은 정보를 PCA 투영하여 주요 데이터를 추출한 후 이를 HMM(Hidden Markov Model)에 입력하여 표정을 인식하였다. Fasel[17]은 기본 얼굴 영상과의 차영상을 PCA나 ICA에 투영한 데이터를 최근린 분류(Nearest Neighbor Classification)를 사용하여 표정을 구분하였다.

모델 기반 방법은 얼굴의 특징점의 위치나 움직임 등을 기반으로 하는 방법으로 가려짐 현상에 강한 모습을 보이지만 다양한 표정 구분하기 위해서는 정확한 특징점들의 위치를 찾아야 하며 높은 해상도 입력을 필요로 한다. 이러한 요구 사항들은 긴 처리 시간을 야기하며 실시간 표정 인식에 부적합하게 된다.

영상 기반 방법은 본래 영상의 화소 값을 이용하므로 고 해상도 이미지에서 특징점을 찾고 시작해야 하는 모델 기반 방법에 비해 빠른 장점이 있으나 얼굴 검출이 정확하게 이루어지지 않았을 경우 모델 기반의 방법에 비해 오인식이 늘어나는 단점이 있다.

본 논문에서는 영상 기반의 방법을 이용하였다. 보통 영상 기반 방법에 이용되는 가버필터(Gabor filter), PCA, SVM(Support Vector Machine), 신경망(Neural Network) 등과 같은 특징들과 인식기들은 정확도가 높으나 느린 단점이 있다. 하지만 최근 실시간이 보장되며 정확도가 높은 Viola와 Jones[18]가 사용한 Haar-like 특징 기반 Discrete Adaboost를 이용한 알고리즘이 등장함으로써 느린 문제가 해소되었다. Adaboost 알고리즘은 예측 성능이 조금 떨어지는 여러 규칙들을 순차적 효과적으로 조합하여 높은 성능의 규칙을 생성하는 방법으로 최근 높은 정확성과 빠른 성

능으로 인해 많이 이용되고 있다.

본 논문의 알고리즘에서는 빠르고 정확하나 출력의 기복이 심한 Haar-like 특징 기반 Discrete Adaboost를 최대한 이용하고 모든 Adaboost 강분류기의 출력 값을 신경망을 이용하여 고려함으로써 출력 값의 기복이 심한 Adaboost의 단점을 수용하면서 실시간의 속도와 높은 정확도를 유지한 표정인식기를 구현하였다.

본 논문은 더욱 높은 정확도의 강분류기를 얻기 위해 Viola의 4가지 Haar-like 특징을 확장 시켜 13가지 특징을 이용하였고 인식하기 위한 24x24 윈도우 크기를 2배 확장 시켜 48x48 윈도우를 사용하여 Adaboost 강분류기가 표정을 충분히 학습할 수 있는 정보를 제공하였다. 본 논문의 표정인식 알고리즘은 Adaboost 기반 얼굴 검출 알고리즘을 통해 얻은 얼굴 영상을 각 표정별로 학습된 Adaboost 강분류기에 입력하여 각 표정별로 강분류기 출력 값을 얻고 이 출력 값들을 정규화 하여 학습된 신경망에 입력하여 최종적으로 표정 인식을 수행한다. 이 때 상대적으로 느린 신경망의 비중을 줄이고 출력 값의 기복이 심한 Adaboost의 단점을 보완하여 실시간 실행 속도와 높은 인식률을 얻는 방법을 제안한다.

논문의 구성은 2장에서 표정 인식 시스템을 서술하였고, 3장에서는 확장된 Haar-like 특징에 대해 다루고, 4장에서는 Discrete Adaboost를 소개하며, 5장에서는 사용된 신경망의 구성과 Adaboost와 조합하였을 경우 장점을 설명하였다. 6장에서는 표정 데이터베이스를 이용하여 본 논문 알고리즘의 실시간성과 정확성을 검증하였고, 7장에서는 결론 및 향후 연구 방향을 서술하였다.

2. 표정 인식 시스템

본 논문에서 제안하는 표정 인식 시스템의 전체 구조는 그림 1과 같다.

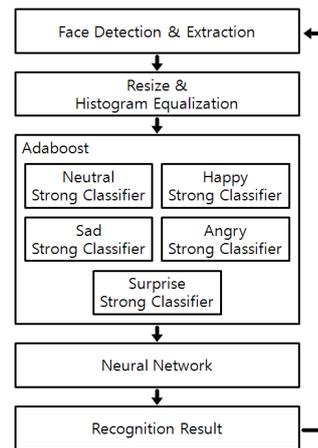


그림 1. 표정 인식 시스템

Fig. 1. Facial Expression Recognition System

얼굴이 포함된 영상을 얻은 후 영상에서 Viola[18]의 Haar-like 특징 기반 Discrete Adaboost를 이용하여 얼굴의 위치와 크기를 파악한다. 그 후 얼굴을 영상에서 추출하여 48x48 크기로 영상을 변환한다. 얼굴 영상을 조명에 영향을 비교적 덜 받고 일정한 입력을 위해 히스토그램 평활

화를 한다. 이렇게 얻어진 얼굴 영상을 5가지 표정별로 학습된 Adaboost 강분류기(Strong Classifier)에 입력한다. 각 표정별로 학습된 Adaboost 강분류기는 각각의 학습된 표정이 지어진 얼굴 영상을 입력 받았을 때 높은 값을 출력하는 경향을 가지고 있다. 하지만 각 표정별로 학습된 강분류기 출력 값이 각 표정별로 최대 값 및 최소 값이 같지 않으며 표정별로 서로 다른 최대 최소 값을 가진다. 이러한 문제로 각 표정별 강분류기의 출력 값을 수용할 수 있는 충분한 범위를 반영하며 이를 신경망의 입력을 위해 -1~1 사이로 정규화하여 사용한다. 정규화된 표정 값은 신경망의 5개 입력 노드(Input Node)에 입력된다. 5개의 입력 노드에 입력된 값은 4개의 숨은 노드(Hidden Node)를 거쳐 5개의 출력 노드(Output Node)로 출력된다. 신경망의 출력 값 중 가장 높은 값에 해당 되는 표정을 최종으로 인식된 표정으로 출력한다.

3. 확장된 Haar-like 특징

본 논문에서 표정을 인식하기 위해 Viola[18]의 4종류의 Haar-like 특징을 확장하여 13종류의 Haar-like 특징을 이용하였다. 그림 2는 확장된 Haar-like 특징의 종류를 보여주고 있다.

그림 2의 특징들은 각 특징마다 각각 마스크 모양에 해당하는 특징들을 이미지에서 추출해 낸다. 이 중에 일부를 보자면 1×2 특징은 세로 성분, 2×1 특징은 가로 성분, 1×3 특징 중 2번째 특징은 세로선 성분, 3×1 특징 중 2번째 특징은 가로선 성분, 2×2 특징 중 5번째 특징은 대각선 성분을 추출한다.

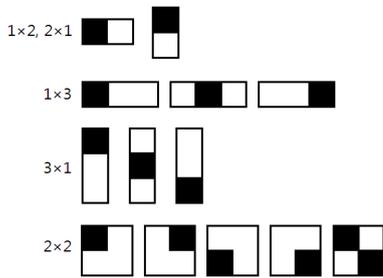


그림 2. 확장된 Haar-like 특징
Fig. 2. Extended Haar-like Feature



그림 3. Haar-like 특징이 적용된 예
Fig. 3. Example of adapted Haar-like feature

Kim[12]의 표정인식 실험에서 4종류 이상으로 확장된 Haar-like 특징을 기반으로 표정 인식을 수행했을 때 4종류의 Haar-like 특징을 이용하였을 경우에 비해 높아진 표정 인식률을 얻었으며 적은 수의 약분류기를 이용하여 표정인식에 충분한 강분류기를 얻었다. Haar-like 특징의 종류를 확장함으로써 Adaboost 학습 알고리즘에 더욱 다양한 약분류기 선택권을 제공하여 더욱 높아진 정확도의 강분류기를 얻기 때문인 것을 알 수 있다. 이와 같은 이유로 본 논문

에서는 더욱 정확하고 효율적인 강분류기를 얻기 위해 13종류로 Haar-like 특징의 종류를 확장하였다.

위와 같은 Haar-like 특징들은 그림 3과 같이 48×48 크기 영상에 적용된다. 그림 2와 같은 Haar-like 특징들이 그림 3과 같이 48×48 크기의 영상에 적용되었을 때 1×2, 2×1 특징은 각각 677,376개를 가지며, 1×3, 3×1 특징은 각각 442,176개를 가지며, 2×2 특징은 각각 331,776개의 특징을 가져 총 5,666,688개의 특징을 가지며 이를 이용하여 이미지에서 특징을 추출한다.

$$f(x) = |W_{sum} - B_{sum}|$$

$$W_{sum} = \sum_{n=1}^M \sum_{x=W_{n,x}}^{W_{n,x}+W_{n,w}-1} \sum_{y=W_{n,y}}^{W_{n,y}+W_{n,h}-1} i(x,y) \quad (1)$$

$$B_{sum} = \sum_{n=1}^N \sum_{x=B_{n,x}}^{B_{n,x}+B_{n,w}-1} \sum_{y=B_{n,y}}^{B_{n,y}+B_{n,h}-1} i(x,y)$$

식 (1)은 Haar-like 특징의 값을 나타낸다. $i(x,y)$ 는 (x,y) 좌표의 화소 밝기를 의미하며 M 과 N 은 마스크의 각각 흰 사각형 개수와 검은 사각형 개수이다. W_x, W_y, W_w, W_h 는 각각 흰색 사각형의 x 축 및 y 축 좌표, 가로 및 세로 길이를 나타내고 B_x, B_y, B_w, B_h 는 각각 검은색 사각형의 x 축 및 y 축 좌표, 가로 및 세로 길이를 의미한다. 식 (1)은 Haar-like 특징의 검은 사각 영역의 화소 값의 합과 흰 사각영역 화소 값의 합의 차이 값을 나타낸다.

Haar-like 특징은 화소별로 일일이 더해져 합을 구한다면 많은 수행 시간이 걸린다. 이러한 문제를 해결하기 위해서 특징 값은 적분 영상(Integral Image)을 이용하여 구한다. 적분 영상은 식 (2)를 통하여 생성할 수 있다.

$$ii(x,y) = \sum_{x'=0}^{x-1} \sum_{y'=0}^{y-1} i(x',y') \quad (2)$$

3	3	2	1	3
1	4	5	2	4
1	2	6	3	5
6	2	1	3	1
5	2	5	1	4

3	6	8	9	12
4	11	18	21	28
5	14	27	33	45
11	22	36	45	58
16	29	48	58	75

그림 4. 일반 영상(왼쪽)과 적분 영상(오른쪽)
Fig. 4. Example of image(left) and integral image(right)

그림 4는 5×5의 일반 영상(왼쪽)을 적분 영상(오른쪽)으로 변환한 예이다. 적분 영상에서 그림 5와 같이 회색 사각 영역의 화소 값들의 합을 구할 때 식 (3)을 이용하여 회색 사각 영역 크기에 관계없이 단순한 4번의 합차로 계산 가능하다. 초기에 적분 영상을 만들 때 기회비용이 들어가지만 이후 빠른 연산으로 전체 시스템의 속도를 높인다.

$$\sum_{x=x_1}^{x_4} \sum_{y=y_1}^{y_4} i(x,y) = ii(x_4,y_4) + ii(x_1-1,y_1-1) - ii(x_2,y_2-1) - ii(x_3-1,y_3) \quad (3)$$

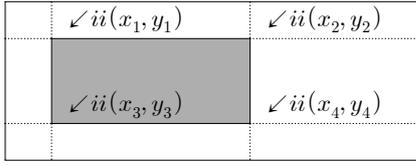


그림 5. 적분 영상의 회색 사각영역 화소 값들의 합 $ii(x_4, y_4) + ii(x_1 - 1, y_1 - 1) - ii(x_2, y_2 - 1) - ii(x_3 - 1, y_3)$

Fig. 5. Sum of pixels of gray rectangle in integral image is calculated by $ii(x_4, y_4) + ii(x_1 - 1, y_1 - 1) - ii(x_2, y_2 - 1) - ii(x_3 - 1, y_3)$

4. Discrete Adaboost 알고리즘

4.1 약분류기(Weak Classifier)

본 논문에서는 Discrete Adaboost 학습 알고리즘을 이용하여 평온, 행복, 슬픔, 화남, 놀람 5가지 표정을 학습하여 각 표정에 잘 반응하는 강분류기를 생성하여 이용하였다. 강분류기를 생성하기 위해서는 Adaboost 학습 알고리즘에 약분류기를 제공해주어야 한다.

$$h(x) = \begin{cases} 1 & \text{if } f(x) > \theta \\ -1 & \text{otherwise} \end{cases} \quad (4)$$

식 (4)은 약분류기를 나타내고 있다. $f(x)$ 는 영상에서 Haar-like 특징 값을 의미하며, $h(x)$ 는 약분류기를 표현한다. θ 는 사전 정의해둔 문턱 값(Threshold)이다. 약분류기의 값은 Haar-like 특징 값이 사전 정의된 θ 값보다 크면 1을 가지며 그 외에는 -1 값을 가진다.

그림 6은 식 (4)에서 사전 정의된 문턱 값 θ 의 사전 정의 방법을 보여준다. True Image는 학습시킬 표정 영상 데이터베이스, False Image는 학습 시키고자 하는 표정 이외의 데이터베이스를 의미한다. 학습시킬 표정과 그 이외의 표정 데이터베이스의 평균과 분산을 ML(Maximum Likelihood)를 이용하여 구한다. 학습시킬 표정 데이터베이스의 확률 분포와 그 이외 표정 데이터베이스의 확률 분포가 만나는 지점을 베이저안 정리(Bayes' Theorem)에 따라 문턱 값 θ 로 정하여 이용한다.

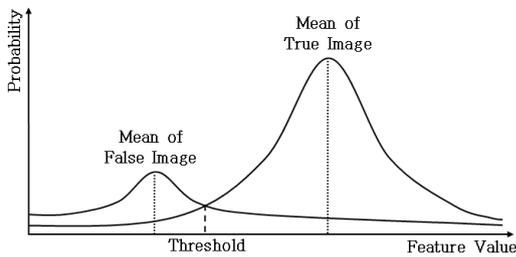


그림 6. 베이저안 정리를 이용한 문턱 값
Fig. 6. Determined optimal threshold by Bayes' Theorem

4.2 Discrete Adaboost 학습

본 논문에서는 각각 표정에 따라 잘 반응하는 분류기를 얻기 위해 Freund[19]의 Discrete Adaboost 알고리즘을 이용하였다. Discrete Adaboost는 약 60~80%의 정확도를 가지며 상호보완 할 수 약분류기를 찾아 순차적 선형적으로

조합시켜 높은 정확도를 가지는 강분류기를 생성하는 알고리즘으로 표 1과 같은 과정을 따른다.

표 1의 x_N 은 학습 데이터, t_N 은 학습 데이터의 참 거짓 여부, m 은 거짓 데이터 개수, l 은 참 데이터 개수, $w_n^{(1)}$ 은 첫 번째 학습 데이터의 무게값, e_m 은 약분류기의 오류율, α_m 은 학습 단계에서 선택된 약분류기의 무게값, $y_m(x)$ 은 약분류기, $Y_m(x)$ 은 강분류기를 나타낸다.

표 1. Discrete Adaboost 학습 알고리즘

Table 1. Discrete Adaboost training algorithm

Training set: $(x_1, t_1), (x_2, t_2), \dots, (x_N, t_N)$
x_N : data t_N : True(1) or False(-1)
1. Initialize image data weighting w_n by setting $w_n^{(1)} = \frac{1}{2m}, \frac{1}{2l}$ m = number of negative image l = number of positive image $w_n^{(1)}$: first data weighting coefficient
2. For $m = 1, 2, \dots, M$: (a) Find weak classifier $y_m(x)$ which have minimum error rate. $J_m = \sum_{n=1}^N w_n^{(m)} I(y_m(x_n) \neq t_n)$ $I(y_m(x_n) \neq t_n)$: if $y_m(x_n) = t_n$ return 0, if $y_m(x_n) \neq t_n$ return 1
(b) Compute error ratio $e_m = \frac{\sum_{n=1}^N w_n^{(m)} I(y_m(x_n) \neq t_n)}{\sum_{n=1}^N w_n^{(m)}}$ Set current weighting of weak classifier. $\alpha_m = \ln \left\{ \frac{1 - e_m}{e_m} \right\}$ α_m : weak classifier weighting coefficient e_m : error rate
(c) For next weak classifier, update image data weighting. $w_n^{(m+1)} = w_n^{(m)} \exp\{\alpha_m I(y_m(x_n) \neq t_n)\}$
3. Finally, make a strong classifier using linear combination. $Y_m(x) = \text{sign} \left(\sum_{m=1}^M \alpha_m y_m(x) \right)$

학습과정의 초기에는 표 1의 1번과 같이 학습 데이터에 똑같은 가중치를 부여하고 시작한다. 2번 (a)과정에서 학습 중에 오류율이 가장 작은 약분류기를 선택한다. 그리고, (b)과정에서 그 약분류기의 오류율을 계산한 후에 이 오류율을

이용하여 약분류기의 가중치를 계산한다. 다음으로 (c)과정에서 선택된 약분류기가 잘못 분류한 학습 데이터의 가중치를 증가시킨다.

학습 데이터의 가중치가 변경된 상태에서 오류율이 가장 작은 약분류기를 다시 찾는다. 이러한 과정을 반복하며 최종 강분류기에 쓰일 약분류기들을 순차적으로 선택한다. 이러한 방법을 통해 선택된 약분류기는 표 1의 3번과 같이 선택형 조합을 통해 정확도가 높은 강분류기로 이용된다.

5. 신경망 알고리즘

본 논문에서 이용된 신경망 알고리즘은 표정마다 일정하지 않는 Discrete Adaboost 강분류기의 출력 값의 단점을 보완하여 단순히 표정별로 문턱 값을 정해서 표정 인식을 수행할 경우에 비해 많은 이점을 얻었다.

그림 7은 본 논문에서 사용된 신경망의 구조를 보여주고 있다. 각 표정별 Adaboost 강분류기 출력 값을 -1~1의 범위로 정규화한 후 이를 대칭활성함수(Symmetric Sigmoid Function) 기반의 입력 노드 5개, 숨은 노드 4개, 출력 노드 5개로 이루어진 학습된 신경망에 입력하여 출력 값이 가장 높은 표정을 선택한다.

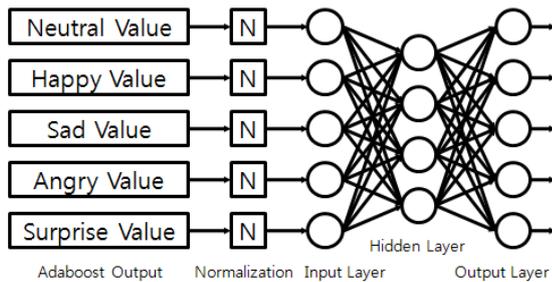


그림 7. 신경망의 구조
Fig. 7. Structure of Neural network

각 표정별로 학습된 Adaboost 강분류기의 출력을 입력 값으로 이용하는데 그 값이 표정별로 최대 최소 출력이 다르다. 비교적 인식하기 쉬운 행복, 놀람은 -30~20 정도 출력 값을 가지며 비교적 인식하기 어려운 평상, 슬픔, 화남은 -20~10 정도 출력 범위를 가진다. 일정하지 않은 강분류기의 출력을 최소 -20에서 최대 20 범위까지만 반영하고 이를 -1~1로 정규화 시킨다. 이를 학습된 신경망의 입력으로 이용하여 표정을 인식한다.

강분류기의 출력을 입력으로 이용하는 방법은 보통 수십 수백 개의 특징 값을 바로 신경망에 입력하여 수십 수백 개의 노드 간의 무게 값을 계산해야 하는 부담스러운 기존 방식에 비해 입력 5개만으로 신경망의 부담을 크게 줄일 수 있으며, 속도가 빠르며 정확성이 보장되어 있는 Adaboost 강분류기의 출력 값을 이용하므로 표정 인식 시스템의 전체적인 정확도를 높여주어 실시간을 보장할 수 있다.

또한 신경망을 최종 표정 인식 선택에 이용하게 되면 일반적으로 각 표정별 Adaboost 강분류기 출력 값에 문턱 값을 이용하여 표정 인식하는 것에 비해 여러 이점이 있다. 예를 들어 행복한 표정을 입력 받았을 때 행복 강분류기가 가장 큰 값을 내며 다른 표정 강분류기들은 낮은 값을 보일 것이다. 이 때 단순히 행복 강분류기의 최대 값만 고려하는

게 아니라 다른 표정 강분류기의 낮아진 값들도 신경망이 표정 인식 시 참고하게 되므로 단순한 문턱 값을 이용하는 방법에 비해 인식률이 높아진다. 또한 입력되는 표정의 강도가 높지 않아 표정별 강분류기의 정의된 문턱 값을 넘을 수 없어 인식 못하게 되는 경우도 신경망을 이용하여 표정 인식이 가능하다.

6. 실험 및 분석

본 논문에서 제안하는 표정 인식 시스템의 정확도와 실행 시간을 확인하기 위해 Adaboost와 문턱 값을 조합한 경우, Adaboost와 신경망을 조합한 경우, 가버필터(Gabor filter)와 신경망을 조합한 경우 3가지 인식 환경을 이용하여 실험을 진행하였다.

비교를 위한 가버필터와 신경망의 조합한 알고리즘은 3가지 주파수와 0°, 45°, 90°, 135° 4가지 방향의 필터를 48x48 윈도우에 12x12의 일정한 격자점 형성 및 적용하여 특징 값을 1728개 얻고 이를 1728개의 입력 노드, 1036개의 숨은 노드, 5개의 출력 노드로 구성된 학습되어진 신경망에 입력하여 출력 노드 중 가장 큰 출력 노드를 선택하여 인식 하였다.

6.1 표정 데이터베이스 및 Adaboost 학습 결과



그림 8. 표정 데이터베이스의 예
Fig. 8. Sample of facial expression database

실험을 위해 JAFFE(Japanese Female Facial Expression)[20]과 자체적으로 모은 SSUFE(Soongsil University Facial Expression)과 인터넷에서 표정관련 검색어를 입력하여 얻은 표정 영상들 등 3가지 데이터베이스를 구성하여 실험을 진행하였다. 일반적으로 특정 목적을 가지고 모아진 데이터베이스는 제한된 환경 하에서 만들어졌기 때문에 자연스러운 표정을 얻지 못하며 다양하지 못한 환경은 표정 인식기의 성능이 떨어지는 원인이 된다. 제한된 데이터베이스의 문제를 예방하고자 인터넷에서 검색어를 통해 더욱 자연스러운 표정과 다양한 인종, 연령층, 환경의 표정 데이터베이스를 모아 사용하였다. 평은 259개, 행복 255개, 슬픔 120개, 화남 150개, 놀람 191개로 총 975개 표정 영상을 모았으며 그림 8은 위와 같이 모은 데이터베이스

의 예를 보여주고 있다.

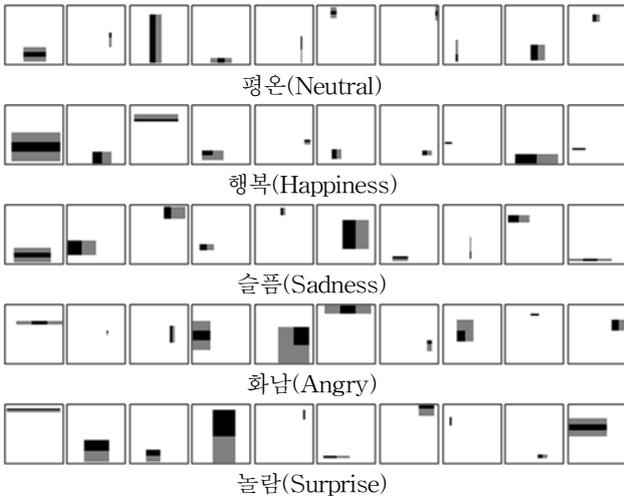


그림 9. 표정별로 선택된 초기 10개의 Haar-like 특징
 Fig. 9. Initial 10 Haar-like features on each facial expression

위와 같은 표정 데이터베이스를 이용하여 각 표정별로 Discrete Adaboost를 이용하여 표정별로 잘 반응할 수 있는 강분류기를 생성하였다. 그림 9는 Discrete Adaboost 학습 알고리즘이 표정별로 선택한 초기 10개의 Haar-like 특징의 모양을 각각 왼쪽부터 차례대로 보여주고 있다. 평온은 일반적인 얼굴 특징을 고루 선택 하고 있으며, 행복은 웃는 입과 눈을, 슬픔은 처진 눈과 입을, 화남은 주름진 미간을, 놀람은 치켜 놀라간 눈썹과 벌어진 입을 중점적 특징으로 선택하여 강분류기를 구성하고 있다. 이렇게 선택된 Haar-like 특징 기반의 약분류기는 학습된 표정을 입력 받았을 때 높은 값을 출력하는 특징을 가진다.

6.2 표정 인식률 비교

Adaboost와 신경망의 조합의 표정 인식 정확도를 확인하기 위해 다음과 같은 비교 실험을 진행하였다.

각 표정마다 정규화된 강분류기 출력에 문턱 값을 두어 강분류기의 출력이 문턱 값이 넘었을 경우 인식하는 방법, 정규화된 강분류기 출력을 신경망에 입력하여 신경망의 출력이 가장 높은 값을 가진 표정을 인식하는 방법, 가버필터 값을 신경망에 입력하여 가장 높은 출력 값의 표정을 인식하는 방법을 비교 하였다. 문턱 값은 베이지안 정리에 따라 정하였다. 공통 사항으로 각 표정마다 100개의 약분류기를 조합한 강분류기를 이용하였다.

그림 10은 첫 번째 실험 결과를 나타내고 있다. 흰색 바는 Adaboost와 문턱 값을 이용하였을 때, 검은색 바는 Adaboost와 신경망을 이용하였을 때, 회색바는 가버필터와 신경망을 이용하였을 때의 정확도를 나타내고 있다. 실험 결과에서 Adaboost와 문턱 값을 이용했을 때 0.74~0.9의 정확도, Adaboost와 신경망을 이용했을 때 0.86~0.95의 정확도, 가버필터와 신경망을 이용하였을 때 0.84~0.93의 정확도를 보였다. 전반적으로 Adaboost와 신경망을 이용했을 경우에 정확도가 높음을 알 수 있으며 Adaboost와 신경망을 이용한 방법이 문턱 값을 이용한 방법에 비해 안정적인 결과를 보이고 있음을 알 수 있다. Adaboost 강분류기의 낮

은 출력을 보이는 평온, 슬픔, 화남 표정을 보았을 때 낮은 출력으로 인해 문턱 값을 못 넘는 경우가 발생하기 때문에 오인식되는 경우가 신경망을 이용했을 때에 비해 많았다. 반면 강분류기가 잘 반응하는 표정인 행복, 놀람의 경우에는 신경망을 이용하였을 경우와 비교해 인식률이 조금 떨어지긴 했지만 전반적으로 잘 인식하고 있음을 알 수 있다.

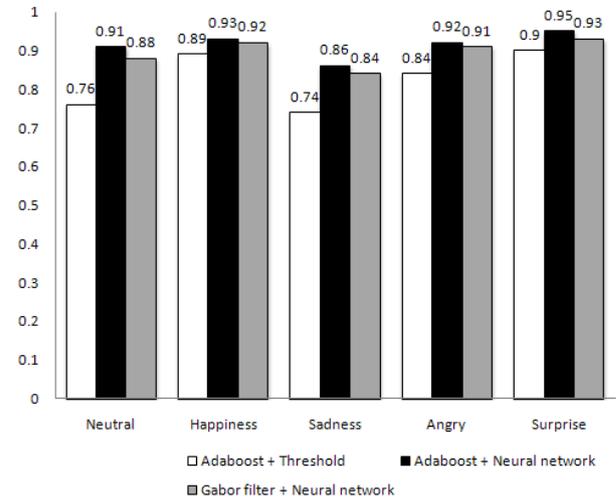


그림 10. Adaboost와 문턱 값, Adaboost와 신경망, 가버필터와 신경망을 이용하였을 경우의 표정별 정확도
 Fig. 10. Accuracy rate of each facial expressions which are use Adaboost and threshold, Adaboost and Neural network, Gabor filter and Neural network each

Adaboost와 신경망을 이용한 최종 표정 인식은 전체적 인식률을 높여주긴 하지만 기본적으로 표정 인식하는 Adaboost 강분류기 성능이 뒷받침이 되어야 높은 인식률을 얻을 수 있음을 알 수 있다.

6.3 표정 인식 수행 시간

Adaboost와 신경망을 조합한 표정 인식 시스템의 수행 시간을 알아보기 위해 2.4Ghz의 CPU의 컴퓨터와 320x240의 카메라 입력 영상을 이용하여 실험하였다.

Adaboost를 이용한 얼굴 검출 시간과 5개의 표정별로 100개의 Adaboost 강분류기와 신경망을 이용하여 표정을 인식하는 시간을 측정해 보았다. 또한 비교를 위해 가버필터와 신경망 조합을 이용하였을 때의 시간도 측정 하였다.

표 2는 표정 인식 시스템의 수행 시간을 나타내고 있다. Adaboost 기반의 얼굴 검출 시간이 가장 오래 걸려 52ms가 걸렸다. Adaboost와 신경망 조합을 이용한 경우 표정을 구분하기 위한 Adaboost 강분류기의 출력 값을 계산하는데 6ms, 신경망을 이용하여 최종 표정 인식 하는데 0.015ms가 걸려 표정 인식에는 6.015ms가 필요했으며 전체적으로 한 프레임 당 평균적으로 58.015ms의 수행 시간이 필요로 하였다. 이는 초당 17 프레임 정도 처리할 수 있어 실시간 처리가 가능함을 알 수 있다. 또한 가버필터와 신경망을 조합하였을 경우에는 가버필터를 이용해 특징 값을 얻는데 4.172ms, 이를 신경망을 이용해 인식하는데 573.271ms가 걸려 전체적으로 629.443ms가 한 프레임 당 필요함을 알 수 있다. Adaboost와 신경망을 조합한 경우에 비해 가버필터와 신경망의 조합은 특징 값을 구하는 경우는

시간이 적게 걸렸지만 상대적으로 커져버린 신경망으로 인해 전체적으로 느려진 것을 알 수 있다.

표 2. 표정 인식 시스템 수행 시간

Table 2. Process time of facial expression recognition system

Process		Time (ms)
Face detection		52.000
Adaboost + Neural Network	Discrete Adaboost strong classifier	6.000
	Neural network	0.015
Total		58.015
Gabor filter + Neural Network	Gabor filter feature extraction	4.172
	Neural network	573.271
Total		577.443

그림 11은 표정 인식 시스템을 카메라에 적용하여 실행한 데모를 보여주고 있다. 평온, 행복, 슬픔, 화남, 놀람 5가지 표정을 인식하는 것을 볼 수 있다. 실제로 카메라에 적용해본 결과 평온, 행복, 화남, 놀람 4가지 표정은 잘 인식이 되었으나 슬픔과 같이 특징이 잘 드러나지 않는 표정의 경우 원활히 인식하지 못하는 모습을 보였다.



그림 11. 표정 인식 데모

Fig. 11. Demo of facial expression recognition

7. 결론 및 향후 연구 방향

본 논문은 실시간과 높은 정확성이 보장된 Adaboost 알고리즘과 신경망을 조합하여 빠르면서도 정확도가 높은 표정 인식 시스템 구현을 목표로 연구를 진행하였다. 가버필터와 신경망 조합과 비교하였을 때 특징 값들을 바로 신경망에 입력하여 사용하는 일반 영상 기반 방법에 비해

Adaboost와 신경망의 조합은 신경망의 비중이 줄어 빨라짐을 알 수 있었다. Adaboost와 신경망 조합은 빠르고 정확도 높은 Adaboost 출력 값을 입력 값으로 하여 신경망의 비중을 줄여주어 전체 시스템 속도 향상과 정확도 향상을 가져 왔다. 실험을 통해 약 86~95%의 정확도와 6.015ms의 빠른 수행속도를 얻어 Adaboost와 신경망의 조합의 우수성을 보였다.

하지만 본 논문에서 사용된 알고리즘은 얼굴 추출이 정확히 이루어지지 못하였을 경우 오인식률이 높아지는 단점이 있다. 향후 이런 문제를 해결하기 위해 얼굴 추출 부분을 보완할 계획이다.

Acknowledgment

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2010-(C1090-1021-0010)). 본 논문은 2010년 정부 교육과학기술부(한국학술진흥재단)의 2단계 BK21사업의 재원으로 지원을 받아 수행된 연구임.

참고 문헌

- [1] 이용재, 이철우, "전역 및 부분 특징 정보를 이용한 제스처 인식," *정보과학회논문지: 소프트웨어 및 응용*, 32권, 8호, pp. 759-768, 2005.
- [2] Andreas Haag, Silke Goronzy, Peter Schaich and Jason Williams, "Emotion Recognition Using Bio-sensors: First Steps towards an Automatic System," *In Proceedings of LNCS*, vol.3068, pp.36-48, 2004.
- [3] 심귀보, 박창현, "음성인식으로 감정인식 요소 분석," *퍼지 및 지능 시스템학회 논문지*, 11권, 11호, pp. 510-515, 2001.
- [4] 고현주, 이대중, 박장환, 전명근, "웨이블릿 변환을 이용한 음성에서의 감정 추출 및 인식 기법," *퍼지 및 지능 시스템학회 논문지*, 14권, 2호, pp. 150-155, 2004.
- [5] B. Fasel, Juergen Luetin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, pp. 259-275, 2003.
- [6] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Automatic interpretation and coding of face images using flexible models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 743-756, 1997.
- [7] Yeongjae Cheon, Daijin Kim, "A Natural Facial Expression Recognition Using Differential-AAM and k-NNS," *Pattern Recognition*, vol.42, no.7, pp.1340-1350, 2008.
- [8] Bouchra Abboud, Franck Davoine, Mo Dang, "Facial expression recognition and synthesis based on an appearance model," *Signal Processing: Image Communication*, vol.19, no.8, pp.723-740, 2004.
- [9] Yongsheng Gao, Maylor K. H. Leung, Siu Cheung Hui, Mario W. Tananda, "Facial Expression Recognition From Line-Based

저자 소개

Caricatures," *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, vol. 33, no. 3, May. 2003.

[10] C. L. Lisetti and D. E. Rumelhart, "Facial expression recognition using a neural network," *The 11th International Flairs Conference. AAAI Press*, 1998.

[11] C. Padgett, G.W. Cottrell, "Representing face images for emotion classification," *Proc. Conf. Advances in Neural Information Processing Systems*, vol. 9, pp. 894-900, 1997.

[12] Do Hyoung Kim, Sung Uk Jung, Myung Jin Chung, "Extension of cascaded simple feature based face detection to facial expression recognition," *Pattern Recognition Letters*, vol. 29, pp. 1621-1631, 2008.

[13] DuanDuan Yang, LianWen Jin, JunXun Yin, LiXin Zhen, JianCheng Huang, "Facial Expression Recognition with Pyramid Gabor Features and Complete Kernel Fisher Linear Discriminant Analysis", *International Journal of Information Technology*, vol. 11, no. 9, pp 91-100, 2005.

[14] Irene Kotsia, Stefanos Zafeiriou, Ioannis Pitas, "Texture and shape information fusion for facial expression and facial action unit recognition," *Pattern Recognition*, vol. 41, pp. 833-851, 2008.

[15] C. Padgett, G. Cottrell, "Representing Face Image for Emotion Classification," *Advances in Neural Information Processing Systems*, vol. 9, pp. 894-900, 1996.

[16] J. Lien, T. Kanade, J. Cohn, C. Li, "Automated facial expression recognition based on FACS action units," *IEEE Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, 1998.

[17] B. Fasel, J. Luetttin, "Recognition of a symmetric facial action unit activities and intensities," *Proceedings of the International Conference on Pattern Recognition*, pp. 1100-1103, 2000.

[18] Paul Viola, Michael J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.

[19] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting", *Journal of Japanese Society for Artificial Intelligence*, vol. 14, no. 2, Sep. 1999.

[20] Japanese Female Facial Expression JAFFE <http://kasrl.org/jaffe.html>



홍용희 (Yonghee Hong)

2009년 : 숭실대 정보통신전자공학부 학사
2009년~현재 : 숭실대 대학원 전자공학과 석사과정

관심분야 : 얼굴검출, 얼굴표정인식, 영상처리
Phone : 02-821-2050
Fax : 02-826-8937
E-mail : kolakola@ssu.ac.kr



한영준 (Youngjoon Han)

1996년 : 숭실대 전자공학과 학사
1998년 : 숭실대 전자공학과 석사
2003년 : 숭실대 전자공학과 박사
2009년~현재 : 숭실대 정보통신전자 공학부 부교수

관심분야 : 로봇비전, 영상처리, 비주얼 서보잉
Phone : 02-821-2050
Fax : 02-826-8937
E-mail : young@ssu.ac.kr



한헌수 (Hernsoo Hahn)

1991년 : University of Southern California (공학박사)
1992년~현재 : 숭실대 정보통신전자 공학부 교수
1994년 : 일본기계기술 연구소 객원연구원
1998년 : 숭실대 어학원장
1999년 : 숭실대 정보통신전자공학부 학부장

관심분야 : 자동화 시스템, 자료융합, 물체인식
Phone : 02-821-2050
Fax : 02-826-8937
E-mail : hahn@ssu.ac.kr