
SVM을 이용한 얼굴 검출 성능 향상에 대한 연구

이지근* · 정성태**

A Study on the Performance Enhancement of Face Detection using SVM

Chi-Geun Lee* · Sung-Tae Jung**

이 논문은 2004년도 원광대학교의 교비 지원에 의해서 수행됨

요 약

본 논문에서는 SVM(Support Vector Machine)을 이용하여 얼굴 검출 성능을 향상시키는 방법을 제안한다. 본 논문에서는 먼저 영상내의 거대한 특징 집합으로부터 중요한 작은 특징 집합을 선택하는 AdaBoost 기반 객체 검출 방법을 사용하여 얼굴 후보 영역을 검출한다. 그 다음에는 특징 벡터에 대해 SVM 기반 이진분류를 수행하여 후보 영역의 영상이 얼굴인지 아닌지를 판별한다. 실험 결과 본 논문에서 제안한 방법은 기존의 방법에 비하여 얼굴 검출의 정확도를 향상시켰다.

ABSTRACT

This paper proposes a method which improves the performance of face detection by using SVM(Support Vector Machine). First, it finds face region candidates by using AdaBoost based object detection method which selects a small number of critical features from a larger set. Next it classifies if the candidate is a face or non-face by using SVM(Support Vector Machine). Experimental results shows that the proposed method improve accuracy of face detection in comparison with existing method.

키워드

Face Detection, AdaBoost, SVM

1. 서 론

최근 들어 컴퓨터 비전은 다양한 분야에서 활발하게 연구가 진행되어 왔다. 특히, 얼굴 검출은 영

상 감시 시스템, 원격회의 시스템, 휴먼-컴퓨터 인터페이스, 얼굴 인식 시스템과 같은 응용을 위한 핵심 요소 기술로서 널리 연구되고 있다. 얼굴 검출은 얼굴의 움직임, 크기 변화, 얼굴 표정, 인종,

*원광대학교 컴퓨터공학과

**원광대학교 전기전자 및 정보공학부

나이, 성별, 헤어스타일, 타 객체와의 겹침, 조명, 카메라의 기계적인 특성 등에 따라 다양하게 나타날 수 있기 때문에, 얼굴 검출 기술에 대한 상당한 연구가 진행되었지만 아직도 많은 문제점을 가지고 있다. 기존의 얼굴 검출 방법은 지식 기반(knowledge-base) 방법[1], 특징 기반(feature-based) 방법[2], 형판 정합(template matching) 방법[3], 외형 기반(appearance-base) 방법[4] 등이 있다.

기존의 얼굴 검출 시스템에서 얼굴 스킨 색을 이용한 방법과 모션 정보를 이용하여 배경을 제거하는 방법은 조명과 카메라 등 외부환경에 따라 매우 제한적 시스템이고, 신경망 등은 많은 연산 작업을 필요로 한다. 대부분의 스틸 이미지에 대한 얼굴 검출 시스템은 높은 해상도의 이미지를 이용하여 처리하며, 결과에 대한 빠른 응답을 요구하지 않고 정확한 검출 성능에 초점이 맞추어져 있다. 그러나 비디오 영상 또는 카메라를 이용한 시스템은 낮은 해상도에서 빠른 시간 내에 변하는 이미지의 정보를 이용하여 얼굴 검출 정확성과 함께 빠른 응답을 요구하게 된다. 따라서 본 논문에서는 낮은 해상도의 저가형 웹카메라에서 실시간 다중 얼굴 검출을 위해 이미지내의 픽셀들의 정보에서 특징(feature) 정보를 생성하고 이를 이용하는 방법을 사용하였다. 이러한 특징 정보를 이용하여 얼굴 영역을 검출하는 과정의 가장 큰 문제점은 주어진 이미지에 존재하는 다양한 크기의 얼굴을 찾기 위해 멀티스케일의 검색 윈도우를 이용하여 주어진 이미지 상에서 쉬프트 하면서 스캔 과정을 수행하는데 이는 많은 연산을 필요로 한다. 이를 해결하기 위해 빠르게 객체를 검출하기 위한 AdaBoost 객체 검출 방법이 제안되었다[5,6,7]. AdaBoost 방법은 영상내의 거대한 특징 집합으로부터 중요한 작은 특징 집합을 선택하여 연산량을 줄이는 방법으로 Harr-like 특징과 인테그랄 이미지를 적용하여 빠르고 정확한 얼굴 후보 영역을 검출한다. 본 논문에서는 AdaBoost에 의해 검출된 얼굴 영역의 정확성을 향상시키기 위해 SVM을 이용한 얼굴 검증과정을 추가하였다. AdaBoost에 의해 검출된 얼굴영역의 특징 벡터에 대해 SVM 기반 이진분류[8,9]를 수행하여 얼굴 영역의 영상이 얼굴인지 아닌지를 판별하였다. 본 논문에서 제안하는 다중 얼굴 검출 시스템의 실험은 AdaBoost 방법만을 사용하여 얼굴 후보 영역을 검출한 경우에 발생할 수 있는 오인식된 얼굴 영역을 검증하기 위해 AdaBoost 방법과 SVM 방법을 함께 적용하여 얼굴 검출을 하였다. 이 경우, AdaBoost 방법만을 사용했을 때의 얼굴 영역 검출시간은 극히 적은 시간

이 증가하였지만 얼굴 검출 인식률에 있어서 AdaBoost기법만을 사용했을 경우의 오검출률 3.80%에서 SVM 기법을 병행했을 경우의 0.22%로 정확도가 향상되었다.

II. 간단한 특징을 이용한 빠른 얼굴 후보영역 검출

2.1 전처리 단계

본 논문에서는 얼굴 후보 영역을 검출하기 위해 입력 영상에 대한 전처리 단계로 조명 보정, 히스토그램 평활화, 모션 추적 과정을 적용하였다. 조명 보정은 극단적인 빛(light)의 각도에 따른 격심한 그림자를 줄이기 위해 사용하였다. 히스토그램 평활화는 이미지내의 명암 값의 분포가 특정 값으로 집중되거나 균일하지 못한 영상에 대하여 일정한 명암값의 분포를 갖도록 하여 이미지내의 조명 밝기의 차이와 카메라의 노출에 따른 응답곡선을 보정하기 위해 사용하였다. 모션 추적은 동영상 내에서 얼굴의 움직임이 발생한 경우에 얼굴 검출을 수행하기 위해 사용하였다. 모션 추적을 계산하기 위해서는 이전 프레임과 현재 프레임의 차이를 이용하였고, 움직임이 발생한 경우에 대하여 얼굴 후보 영역 검출을 수행하였다.

2.2 Harr-like 특징과 인테그랄 이미지

카메라로부터 입력되는 얼굴 영상은 다양한 외부환경 요소로부터 많은 영향을 받는다. 정확한 얼굴 검출을 위해서는 얼굴검출에 영향을 미치는 환경 요소에 강인하면서 빠른 연산이 가능한 얼굴 검출 방법이 요구된다. 이를 위해서는 그림 1과 같이 얼굴 영역 내에서 간단하면서도 객체 간의 구별에 효과적인 특징을 사용하여 해결해야 한다.

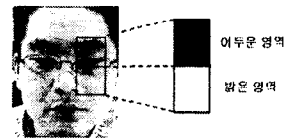


그림 1. 얼굴 영역 내의 특징 예
Fig. 1 An example of feature within face region

본 논문에서 얼굴 검출을 위한 특징으로 Papa georinou et al[5,6]에 의해 제안된 간단하면서도 연산이 빠른 Harr-like 특징을 얼굴 검출을 위한 특징 집합으로 사용하였다. Harr-like 특징은 인테그랄 이미지를 이용하여 빠르게 연산 할 수 있다.

이러한 Harr-like 특징과 인테그랄 이미지의 사용은 복잡한 연산을 줄이고, 높은 검출 성능이 요구되는 실시간 얼굴 검출에 효과적으로 적용된다.

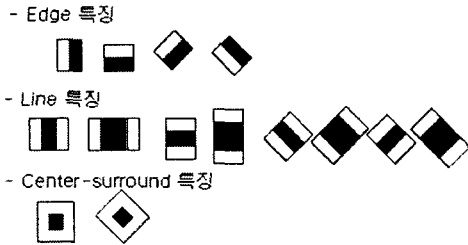


그림 2. Harr-like 특징 프로토타입
Fig. 2 Harr-like feature prototype

그림 2에서의 Harr-like 특징 프로토타입들은 이미지내의 얼굴을 검출하기 위한 효과적인 특징으로 사용된다. Harr-like 특징계수를 이용해서 얼굴 영역을 검출하면, 유동적인 얼굴의 특성에도 강한 검출 성능을 보인다. 이미지내에서 Harr-like 사각형의 픽셀 합계 계산을 빠르게 수행하기 위해 인테그랄 이미지를 사용하였다.

2.3 AdaBoost 알고리즘을 이용한 얼굴 분류기 생성

Harr-like 특징 추출 단계에서 생성된 얼굴 특징 계수를 이용하여 얼굴을 분류하기 위해서 본 논문에서는 Adaptive Boosting(AdaBoost) 기법과 캐스케이드(cascade) 구조를 이용하였다

AdaBoost 알고리즘은 Freund와 Schapire에 의해 소개 되었으며 약한 분류기의 선형적인 결합을 통하여 최종적으로 높은 검출 성능을 가지는 강한 분류기를 생성하여 분류하는 기법으로서 어려움은 약한 분류기의 수가 증가할수록 영점에 지수적으로 근접함이 증명되었다[5,6,7]. AdaBoost 알고리즘은 기본적으로 데이터의 확률 분포를 가지고 학습이 진행된다. 약한 분류기를 학습 시킨 후 올바르게 분류된 데이터에 대해서는 확률 분포를 낮추고 올바르게 분류되지 않은 데이터에 대해서는 확률 분포를 높인다. 그러면 다음 번 약한 분류기를 만들어 낼 경우에는 잘못 분류된 데이터에 집중해서 학습이 된 약한 분류기가 탄생하게 된다.

AdaBoost 알고리즘을 이용한 얼굴 검출 방법은 얼굴과 비얼굴 패턴으로 구성된 훈련영상 집합에서 추출된 사각형 특징 집합을 이용하여 학습된다. 오분류된 훈련 패턴은 가중치를 증가시키고, 바로

게 분류된 훈련 패턴은 가중치를 감소시키는 과정을 반복하면서 최소의 어려움을 나타내는 Harr-like 특징만을 선택한다. 즉, 분류기의 가중치를 수정해 가면서 단계적으로 반복하여 각 단계마다 특징의 선형 집합 중에서 얼굴을 검출하는데 결정적인 역할을 하는 특징만을 남기고 나머지는 제거하는 방식이다.

2.4 캐스케이드(cascade) 구조를 이용한 빠른 배경 제거

본 논문에서는 얼굴 검출의 연산 속도를 향상시키기 위해 그림 3과 같이 AdaBoost 분류를 캐스케이드 구조로 만들어 적용하였다. 캐스케이드 구조는 여러 개의 스테이지를 나누고 스테이지별 약한 분류기의 수를 다르게 하여 수행하는 방법이다. 처음 스테이지 분류에서 대부분의 배경과 같은 비얼굴이 제거되는 기능을 가지고 있어 단일 스테이지를 적용하는 방법보다 훌륭한 성능을 보였다.

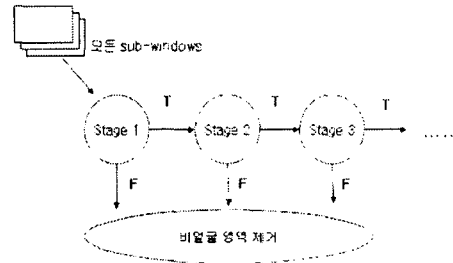


그림 3. 얼굴 검출을 위한 캐스케이드 구조
Fig. 3 Structure of cascade for face detection

본 논문에서 AdaBoost 학습을 위해 총 24개의 스테이지를 이용하였고, AdaBoost 알고리즘을 이용한 얼굴 검출 학습의 결과로 그림 4의 첫 번째 특징(약한 분류기)과 두 번째 특징(약한 분류기)의 예가 표시되어 있다. 첫 번째 특징은 눈 영역과 뺨 영역의 다른 명도가 측정되어 선택 되었고, 두 번째 특징은 두 눈 사이의 다른 명도로 측정되어 선택되었다.

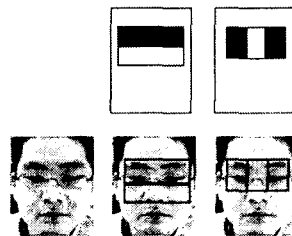
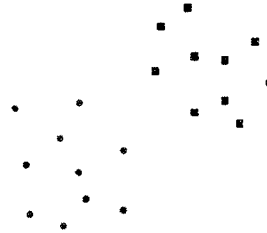


그림 4. AdaBoost의 학습에 의해 선택된 특징
Fig. 4 Feature which selected by AdaBoost learning

얼굴 검출 단계에서 입력 이미지에 대하여 얼굴 검색 윈도우를 일정 크기로 멀티스케일 하면서 일정 간격으로 입력 영상 위에서 쉬프트하면서 스캔을 수행하였고, 모든 서브 윈도우에 대해 스테이지 분류기를 통해 얼굴 영역을 분류하였다. 초기 스테이지에서 많은 수의 배경영상이 제거되었고, 다음 스테이지는 이전 스테이지에서 제거되지 않은 배경영상을 제거하여 고려해야하는 영상의 개수를 감소시켰다. 이와 같이 각 스테이지에서 배경영상을 제거하여 연산량을 대폭 감소시킬 수 있었다.



III. SVM을 이용한 얼굴 영역 검증

3.1 서포트벡터머신

(Support Vector Machine : SVM)

SVM은 1995 Vapnik[9]에 의해 이진패턴 분류 문제를 해결하기 위해 제안된 학습방법으로 각 범주의 대표적인 특성을 갖는 벡터를 이용하여 최적의 하이퍼분리면(Hyperplane)을 찾아서 두 범주를 분류하는 기법이다. SVM은 인식 성능과 처리 속도 성능이 매우 뛰어나기 때문에 최근에 얼굴인식, 문자인식, 문서 범주화 등 다양한 분야에 적용되고 있다[8,9,10].

그림 5(a)에서와 같이 유클리드 공간에서 두 개의 범주를 갖는 N개의 객체가 p차원에 존재할 때, 두 범주를 분류하는 하이퍼분리면이 무수히 많이 존재하게 되고, 두 범주를 가장 효과적으로 나눌 수 있는 최적의 하이퍼분리면이 존재하게 된다.

그림 5(b)의 X_1, X_2 와 같이 하이퍼분리면으로부터 가장 가까이에 위치한 데이터를 서포트벡터(Support Vector:SV)라 하고 하이퍼분리면에서 SV의 위치까지의 최소 거리를 마진(Margin)이라한다. 최적의 하이퍼분리면(Optimal Separating Hyperplane:OSH)은 마진을 가능한 최대로 하여 두 범주간의 구분을 최적화한다.

그림 5(b)에서 하이퍼분리면들을 벡터의 내적으로 표현하면 식(1)과 같고 W 는 하이퍼분리면과 직교하는 가중치 벡터이고, b 는 기준값이다.

$$(W \cdot X) + b = 0 \quad W \in R^n, \quad b \in R \quad (1)$$

(a) 하이퍼분리면

(b) 최적 하이퍼분리면

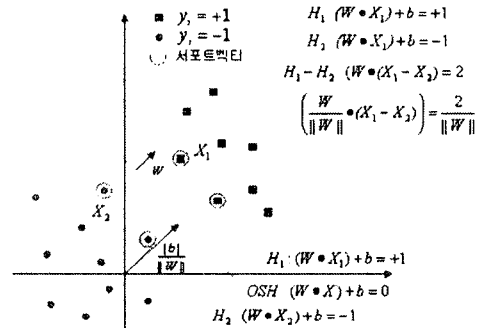


그림 5. SVM의 하이퍼분리면
Fig. 5 Hyperplane of SVM

SVM의 학습 단계는 하이퍼분리면들 중에서 최적의 하이퍼분리면을 찾는 과정이고, 최적의 하이퍼분리면은 마진 $2 / ||W||$ 가 최대가 되도록 하며, $||W|| / 2$ 을 최소화하는 문제가 된다. SVM의 테스트 단계는 임의의 n차원 벡터의 입력 객체 $X_i \in R^n$ 을 두 개의 범주 $y \in \{+1, -1\}$ 로 예측하는 과정이고, 학습을 통해 계산된 최적 하이퍼분리면에 응답하는 분류 결정 함수 $f(X)$ 는 식(2)와 같다.

$$f(X) = \text{sign}((W \cdot X) + b) \quad (2)$$

선형 분리가 가능한 경우에는 식(3)과 같은 최적화 문제가 된다.

$$\max \frac{2}{||W||} \Rightarrow \min \frac{||W||}{2}$$

subject to $y_i(X_i \cdot W + b) - 1 \geq 0 \quad \forall i$ (3)

최적화 문제에서 함수의 최소 또는 최대 문제를 라그랑지(Lagrange) 함수로 유도하면 식(4)와 같은 원문제(Primal)가 된다.

min $L_P = \frac{1}{2} \|W\|^2 - \sum_{i=1}^N a_i [y_i(W \cdot X_i + b) - 1]$
 subject to $a_i \geq 0, i = 1, \dots, N$ (4)

식(4)에 대해서 울프 쌍대문제(Wolfe Dual Problem)로 유도하고 KKT(Karush-Kuhn-Tucker) 조건을 적용하면, 식(6)을 제약조건으로 하고 식(5)를 최대화하는 최적화 문제가 된다[9,10].

max $L_D = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j X_i \cdot X_j$ (5)

subject to $\sum_{i=1}^N a_i y_i = 0, 0 \leq a_i \leq C$ (6)

식(5),(6)에서 라그랑지 계수 a_i 는 2차 프로그래밍 문제(Quadratic Programming:QP)의 해[11]이며, $W = \sum_{i=1}^N a_i y_i X_i$ 이고, C 는 마진 내의 객체를 허용하기 위한 페널티 값이다. $0 < a_i < C$ 이면 SV가 되고, $a_i = C$ 이면 마진에 위치하고, $a_i = 0$ 이면 그밖에 위치한다.

비선형 SVM은 선형 분류가 불가능한 경우에 입력 공간을 분리하는 비선형 최적 결정면을 이용하여 두 벡터의 내적을 나타내기 위한 커널 함수로 입력 공간의 내적을 특징 공간의 내적으로 변형시켜 고차원으로 투영하여 계산을 용이하게 한다.

비선형 SVM 문제는 입력 데이터 공간을 커널 함수를 이용하여 보다 높은 특징 공간으로 맵핑하여 보다 높은 차원의 유클리드 공간으로 학습 데이터를 투영한다.

$f(X_i) = \Phi(X_i) \cdot W + b$ (7)

식(7)은 하이퍼분리면의 일반식이고 $\Phi(X)$ 는 X 의 기저함수로서 보다 높은 차원의 맵핑을 수행하는 함수이다. 식(8)과 같이 두 벡터의 내적을 나타내기 위한 커널 함수 $K(X, X_i)$ 을 이용하여 투영시켜서 선형 분류문제로 해결한다.

$K(X, X_i) = \Phi(X) \cdot \Phi(X_i)$ (8)

맵핑함수를 사용한 라그랑지 듀얼 문제는 식(9),(10)과 같다.

max $L_D = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j \Phi(X_i) \cdot \Phi(X_j)$ (9)

subject to $\sum_{i=1}^N a_i y_i = 0, 0 \leq a_i \leq C$ (10)

일반적으로 사용되는 커널 함수는 식(11)과 같다.

Polynomial : $K(X, X_i) = (X \cdot X_i + 1)^d$

Gaussian : $K(X, X_i) = \exp\left(-\frac{\|X - X_i\|^2}{2\sigma^2}\right)$

Sigmoidal : $K(X, X_i) = \tanh(kX \cdot X_i - \delta)$ (11)

3.2 SVM을 이용한 얼굴 검증

AdaBoost 기반 객체 검출 방법을 이용하여 검출된 얼굴 후보 영역의 특징벡터에 대해 SVM을 이용하여 검출된 후보영역이 얼굴인지 아닌지를 검증 하였다. SVM은 이진 분류기로서 얼굴과 비얼굴을 각각 얼굴(+1)클래스와 비얼굴(-1)클래스로 분류하여 학습을 수행하였다.

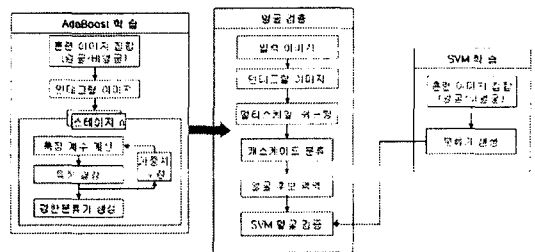


그림 6. 얼굴 영역 검증 흐름도
 Fig. 6 Flowchart for face region verification

본 논문에서는 SVM 검증을 이용하여 빠른 실시간 얼굴 검증 시스템을 구현 하였다. 얼굴 검출 과정에서 입력 영상이 들어오면, SVM의 특징계수를 이용하여 얼굴 영상과 비얼굴 영상의 판별을 거친다.

본 논문에서 제안한 AdaBoost와 SVM을 이용한 얼굴 검증을 실험하기 위해 순수 AdaBoost 방

법과 AdaBoost-SVM 복합 알고리즘 방법을 구축하였다. 첫 번째, 식(12)와 같이 SVM의 학습을 위해 훈련 이미지를 19×19=361 크기로 정규화하고 마스크를 적용하여 훈련데이터를 만들었다.

$$X_i=(d_1, \dots, d_{361}, y), y_i \in \{face, non-face\} \quad (12)$$

그림 7에서 정규화된 얼굴 이미지는 약간의 배경과 헤어가 차지하고 있으므로 마스크를 적용하여 비얼굴 영역을 제거하고 훈련데이터 벡터를 301로 축소하였다. 이와 같이 생성된 훈련 데이터는 식(13)과 같이 나타낼 수 있다. 이러한 훈련 데이터에 비선형 SVM을 적용하였고, 오류 허용 매개변수 C는 200으로 설정하여 학습을 수행하였다.

$$X_i=(d_1, \dots, d_{301}, y_i), y_i \in \{face, non-face\} \quad (13)$$

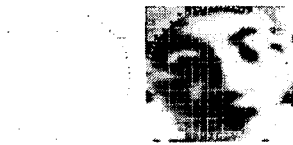


그림 7. SVM의 특징계수에 대한 마스크
Fig. 7 Mask which for coefficient of feature

IV. 실험 및 결과

그림 8과 표 1은 본 논문에서 구현된 얼굴 검출 시스템의 실행 화면과 실험 환경이다.

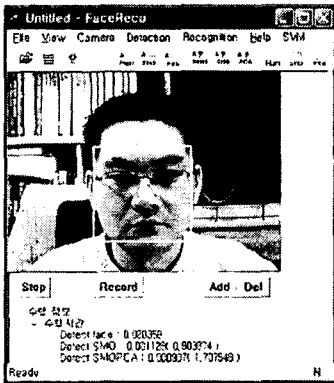


그림 8. 실행 화면
Fig. 8 Scene of face detection

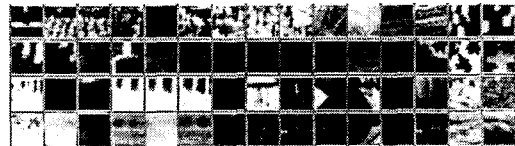
표 1. 실험 환경
Table. 1 Environment of experimentation

구 분	환경	비 고
운영체제	Windows XP	
CPU	Intel Pentium IV 3.0GHz	
프로그램 언어	MS Visual C++ 6.0	
입력 장치	웹카메라 4종류	팬-틸트 기능(1개)

AdaBoost얼굴 검출에 필요한 학습을 위해 MIT 얼굴 데이터베이스[12]를 이용하였다. 그림 9의 MIT 얼굴 DB는 훈련용 데이터로 총 6,977개 이미지(얼굴 : 2,429개 비얼굴 : 4,548개)로 구성되고, 시험용 데이터로 총 24,045개 이미지(얼굴 : 472개, 비얼굴 : 23,573개)로 구성되어 있다.



(a) MIT 훈련용 얼굴 DB 샘플



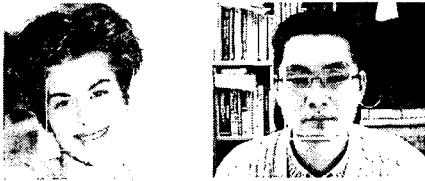
(b) MIT 시험용 비얼굴 DB 샘플

그림 9. MIT 얼굴 DB
Fig. 9 Face DB of MIT

학습 이미지의 크기는 24×24로 정규화 하였고 24개의 스테이지 캐스케이드 구조로서 처음 5개의 스테이지의 특징계수의 수는 1, 10, 25, 25, 50로 각각 설정하였고 나머지 스테이지는 점점 증가시켜 최종 전체 특징계수는 6061개를 사용하였다.

AdaBoost 테스트 단계에서는 웹카메라의 영상 320×240 크기의 이미지를 입력으로 하여 멀티스케일의 탐색윈도우를 이용하여 주어진 이미지 상에서 쉬프트하면서 얼굴 후보 영역을 검출하였다. 멀티스케일은 입력이미지를 멀티스케일하지 않고 검색 윈도우에 대해서 1.25 스케일 값을 설정하여 수행하였다. 쉬프트 단계에서 쉬프트 간격은 2픽셀로 설정하여 수행하였고, 검출된 후보영역들 중에 각

영역 내부에 존재하는 배경 영역은 제거하였다. 그림 10은 실시간 얼굴 영역 검출 예이다.



(a)스틸이미지 (b) 웹카메라

그림 10. 실시간 얼굴 영역 검출
Fig. 10 Real-time face detection

SVM 학습 단계에서는 BioID 얼굴 정규화 데이터와 MIT CBCL의 훈련용 얼굴, 비얼굴 데이터베이스를 이용하였다. 각 얼굴, 비얼굴 이미지들은 SVM의 학습용 훈련데이터로 이용하였다. 얼굴 후보 영역이 얼굴 인지 아닌지를 판단하기 위해서 비선형 SVM 분류기를 이용하였고, 2차 다항식 커널과 페널티를 $C=200$ 으로 설정하였다. 본 논문에서 적용한 AdaBoost기법과 SVM의 복합 알고리즘의 성능을 테스트하기 위해 순수 SVM의 학습과 검증도 함께 실험하였다. 그림 11은 AdaBoost 알고리즘을 단독으로 사용하여 얼굴영역을 검출할 경우의 오검출된 얼굴 영역의 예이다.

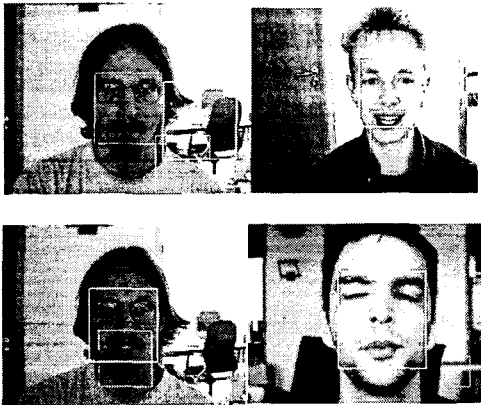


그림 11. AdaBoost의 얼굴 영역 오인식 예
Fig. 11 An example of false detection by using AdaBoost Algorithm

본 논문에서 제안하는 SVM 얼굴 검출 방법은 표 2의 실험 결과와 같이 AdaBoost 기법만을 사

용하여 오인식된 얼굴 영역을 SVM과 AdaBoost 기법을 함께 적용하여 얼굴 영역을 검증함으로써 검출률과 검출시간의 큰 변동 없이 오검출률을 낮출 수 있었다.

표 2. 실시간 얼굴 검출 실험 결과
Table. 1 Experimental result of real-time face detection

구 분	AdaBoost	AdaBoost + SVM
검출률(%)	96.1	95.9
오검출률(%)	3.80	0.52
검출시간(ms)	20.0	22.2

V. 결 론

본 논문에서는 SVM을 이용하여 AdaBoost 기법을 이용한 얼굴검출에서 발생할 수 있는 오검출률을 낮추었다. 본 논문에서 사용된 AdaBoost와 SVM은 모두 통계적 학습 기반 알고리즘이다. 학습 기반 알고리즘은 학습 과정에서 사용자가 직접 학습 샘플데이터, 매개변수(Parameter) 값, 커널 함수를 미리 결정해야 한다. 그러므로 최적의 성능을 구현하기 위해 다양한 데이터를 학습 알고리즘에 적용하여 실험에 의한 최적의 값을 선택하였다. AdaBoost 분류 방법은 간단한 Harr-like 특징들을 인테그랄 이미지를 이용하여 연산을 빠르게 하였고, 캐스케이드 구조는 스테이지별로 특징 수를 다르게 하여 초기 스테이지에서 많은 배경을 제거하여 계산 속도를 빠르게 하였다. AdaBoost 기법에 의해 검출된 얼굴 영역의 특징계수에 대하여 SVM 기반 이진분류를 수행하여 얼굴 검출의 정확도를 향상시켰다. 기존 AdaBoost기법만을 이용한 얼굴 검출 방법에서 발생할수 있는 오검출률을 SVM을 적용함으로써 큰 폭으로 낮출 수 있었다.

참고문헌

- [1] Ming-Hsuan Yang, Kriegman, D.J., Ahuja N., "Detecting face in images : a survey", Pattern Analysis and Machine Intelligence, IEEE Transactions on, Volume: 24 Issue:1, pp.34-58, 2002
- [2] G.Yang, T.S. Huang, "Human Face Detection

- in a Complex Background", Pattern Recognition, Vol. 27, No1, pp.53-63, 1994
- [3] Haiyuan WU, Qian CHEN, "Detecting Human Face in Color Images", Proc of IEEE, pp.2232-2236, 1996
- [4] R.Brunelli, T.Poggio, "Face Recognition: Features versus Templates", IEEE Trans. PAMI., Vol.15 pp.1042-1052, 1993
- [5] Viola, P, Jones, M, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on , Volume: 1 , 8-14, pp.I-511 - I-518 vol.1, Dec. 2001
- [6] Lienhart, R, Maydt, J, "An extended set of Harr-like features for rapid object detection", Image Processing. 2002. Proceedings. 2002 International Conference on , Volume: 1 , 22-25, pp.I-900 - I-903 vol.1, Sept. 2002
- [7] Rainer Lienhart, Alexander Kuranov, Vadim Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection", DAGM'03 25th Pattern Recognition Symposium, Madgeburg, Germany, pp.297-304, Sep. 2003
- [8] V. Vapnik, "The Nature of Statistical Learning Theory", Springer-verlag, New York, 1995.
- [9] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition" in Data Mining and Knowledge Discovery, v.2 n.2, pp.121-167, 1998.
- [10] E. Osuna, R. Freund, F. Girosi, "Training Support Vector Machines: An application to face detection", Proceedings IEEE. CVPR, pp.130-136, 1997
- [11] Platt, J.C., "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines", Microsoft Research Technical Report MSR-TR-98-14, 1998.
- [12] Center for Biological and Computational Learning at MIT and MIT, "CBCL DATASETS", <http://cbcl.mit.edu/cbcl/software-datasets>, 2004.

저자 소개



이지근(Chi-Geun Lee)

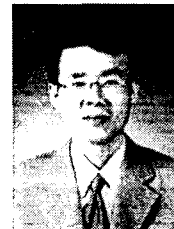
2000년 2월 원광대학교 컴퓨터공학과 학사

2002년 2월 원광대학교 컴퓨터공학과 석사

2002년 3월 ~ 현재 원광대학교

컴퓨터공학과 박사과정

※ 관심분야: 영상처리, 영상인식, 컴퓨터 그래픽스



정성태(Sung-Tae Jung)

1989년 2월 서울대학교 공학 석사

1994년 8월 서울대학교 공학 박사

1995년 3월 ~ 현재 원광대학교

전기전자 및 정보공학부 교수

※ 관심분야: 영상처리, 컴퓨터 그래픽스