

# 특징들의 공유에 의한 기울어진 얼굴 검출

## Rotated face detection based on sharing features

송영모\*, 고윤호\*\*

Young-mo Song, Yun-ho Ko

**Abstract** - Face detection using AdaBoost algorithm is capable of processing images rapidly while having high detection rates. It seemed to be the fastest and the most robust and it is still today. Many improvements or extensions of this method have been proposed. However, previous approaches only deal with upright faces. They suffer from limited discriminant capability for rotated faces as these methods apply the same features for both upright and rotated faces. To solve this problem, it is necessary that we rotate input images or make independently trained detectors. However, this can be slow and can require a lot of training data, since each classifier requires the computation of many different image features. This paper proposes a robust algorithm for finding rotated faces within an image. It reduces the computational and sample complexity, by finding common features that can be shared across the classes. And it will be able to apply with multi-class object detection.

**Key Words** : face detection, boosting, sharing features, multi-class

### 1. 서론

AdaBoost는 Boosting 알고리즘 중의 하나로 약한 분류기(weak classifier)의 선형 결합에 의해 강한 분류기(strong classifier)를 생성하여 2-클래스 분류가 가능한 알고리즘이다. P. Viola와 M. Jones의 AdaBoost를 이용한 얼굴 검출 알고리즘은 haar-like feature와 적분 영상을 사용하여 얼굴 영상과 배경 영상을 입력 받아 둘의 차이를 잘 나타내는 약한 분류기들의 선형 결합으로 강한 분류기를 생성한다[1]. 이 방법은 빠른 얼굴 검출 속도와 뛰어난 성능으로 인해 여러 분야에서 널리 사용되고 있다.

#### 1.1 AdaBoost

AdaBoost 알고리즘의 흐름은 다음과 같다.

Input:  $(x_i, y_i), x_i \in X, y_i \in \{-1, 1\}$

- 1). Start with weights  $w_i = 1/N, i = 1, \dots, N$ .
- 2). Repeat for  $m = 1, \dots, M$ :
  - a) Fit the classifier  $h_m(x)$  on the training data.
  - b) Compute  $e_m = E_w[1_{(y \neq h_m(x))}]$ ,  $c_m = \log((1 - e_m)/e_m)$ .
  - c) Set  $w_i := w_i \exp[c_m \cdot 1_{(y \neq h_m(x))}]$ , and renormalize.
- 3). Output the classifier

$$H(x) = \text{sign} \left[ \sum_{m=1}^M c_m h_m(x) \right]$$

저자 소개

\* 准會員 : 忠南大學 메카트로닉스工學科 碩士課程

\*\* 正會員 : 忠南大學 메카트로닉스工學科 助教授 · 工博

#### 1.2 Adaboost를 이용한 기울어진 얼굴 검출

가장 간단한 방법으로 입력 영상이나 검출기를 여러 각도로 회전하여 검출하면 다른 각도의 얼굴도 검출 가능하겠지만, 모든 각도별로 검출을 해야 하므로 회전된 횟수만큼의 중복 연산이 필요하게 되어 매우 많은 연산량으로 인해 검출 속도의 저하를 가져오게 된다.

다른 방법으로는 다분류가 가능한 multi-class AdaBoost 알고리즘을 이용하는 방법이 있다[2]. 이 알고리즘은 각각의 분류에 대하여 독립된 검출기를 학습시킨 후 이들에 대한 검출기를 생성하는 것이다. 결과적으로 검출물에 있어서는 좋은 결과를 보일 수 있겠지만, 얼굴 각도에 따라 다른 검출기를 만들어 검출하므로 입력 영상을 회전하여 검출하는 것과 연산량의 차이가 없다.

### 2. 특징들의 공유에 의한 기울어진 얼굴 검출

본 논문에서는 정면 얼굴과 각도별로 기울어진 얼굴들이 다른 대상체가 아닌 얼굴이라는 공통된 대상체에 대한 영상이므로 이들이 지닌 유사한 특징을 이용하여 기울어진 얼굴 검출에 적용하고자 한다. 제안하는 방법은 AdaBoost의 기본적인 개념을 응용하여 다중 분류가 가능한 JointBoost를 이용하였으며, 기울어진 얼굴을 검출할 때 입력 영상이나 검출기의 회전 없이 단일의 검출로 이들을 분류할 수 있으며 같은 학습량에서 더 높은 성능을 나타낸다.

#### 2.1 JointBoost

JointBoost는 다분류, 다각도(multi-class and multi-view)의 대상에 대한 분류기를 만들기 위해 고안되었다[3]. 여러 대상체를 분류할 때, 각각의 독립된 특징으로 검출기를 만드

는 것이 아니라 서로 공통된 특징으로부터 대상체를 분류하는 것이 가장 핵심이 되는 개념이다. 따라서 서브셋(subset,  $S(n)$ )이라는 개념을 두어 검출하고자 하는 것이 결합된 서브셋에 포함이 되면, 즉 특징이 공유되면 다음의 a값이나 b값을 선택하게 된다. 서브셋에 포함이 되지 않으면 상수값으로 k값을 선택한다.

$$a_S(f, \theta) = \frac{\sum_{c \in S(n)} \sum_i w_i^c z_i^c \delta(v_i^f > \theta)}{\sum_{c \in S(n)} \sum_i w_i^c \delta(v_i^f > \theta)}$$

$$b_S(f, \theta) = \frac{\sum_{c \in S(n)} \sum_i w_i^c z_i^c \delta(v_i^f \leq \theta)}{\sum_{c \in S(n)} \sum_i w_i^c \delta(v_i^f \leq \theta)}$$

$$k^c = \frac{\sum_i w_i^c z_i^c}{\sum_i w_i^c} \quad c \notin S(n)$$

전체적인 구성을 보면 다음의 알고리즘과 같다.

- 1) Initialize the weights  $w_i^c = 1$  and set  $H(v_i, c) = 0, i = 1..N, c = 1..C$ .
- 2) Repeat for  $m = 1, 2, \dots, 2^c - 1$ 
  - a) Repeat for  $n = 1, 2, \dots, 2^c - 1$ 
    - i) Fit shared stump:
 
$$h_m(v, c) = \begin{cases} a_S & \text{if } v_i^f > \theta \text{ and } c \in S(n) \\ b_S & \text{if } v_i^f \leq \theta \text{ and } c \in S(n) \\ k^c & \text{if } c \notin S(n) \end{cases}$$
    - ii) Evaluate error
 
$$J_{\text{wse}}(n) = \sum_{c=1}^C \sum_{i=1}^N w_i^c (z_i^c - h_m^n(v_i, c))^2$$
  - b) Find best subset:  $n^* = \arg \min_n J_{\text{wse}}(n)$ .
  - c) Update the class estimates
 
$$H(v_i, c) := H(v_i, c) + h_m^{n^*}(v_i, c)$$
  - d) Update the weights
 
$$w_i^c := w_i^c e^{-z_i^c h_m^{n^*}(v_i, c)}$$

## 2.2 클래스 분류

우선적으로 실제 사진을 촬영할 때 검출 가능한 얼굴의 각도를 구분지어 3-클래스로 분류하였다. 주로 많이 유지하는 자세로  $-30^\circ$ 부터  $30^\circ$ 까지의 얼굴 각도가 목격이 되었기에  $-10^\circ \sim 10^\circ$ 를 클래스 1,  $10^\circ \sim 30^\circ$ 를 클래스 2,  $-30^\circ \sim -10^\circ$ 를 클래스 3으로 분류하였으며 배경은 클래스 4로 분류하였다. 사용된 영상은  $32 \times 32$  크기의 얼굴 영상을 각도별로 분류하여 사용하였으며, 클래스 4는 배경 영상을 사용하였다.



클래스 1    클래스 2    클래스 3    클래스 4  
그림 1. 클래스 분류

## 2.3 Haar-like feature

Haar-like feature는 흰색 영역의 밝기 값과 검은색 영역의 밝기 값의 차이로 정의되어 얼굴의 특징을 검출할 때 사용된다. 정면 얼굴과 기울어진 얼굴들의 공유할 수 있는 특징을 찾기 위해서는 정면 얼굴 검출에 사용된 Haar-like feature의 변형이 필요하다. 따라서 그림 2와 같이 공유할 수 있는 특징을 잘 찾을 수 있는 14가지 유형의 특징을 만들고, 이들의 크기에 변화를 주며 검출 위치를 제한하여 특징 집합을 만들었다.

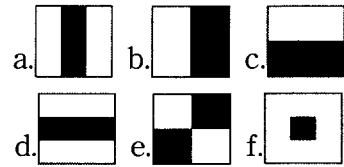


그림 2.1 기존의 유형

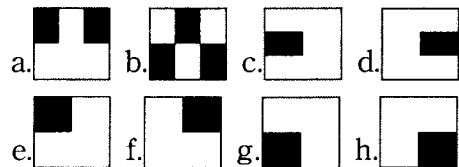


그림 2.2 변형된 유형

이들은 모든 영역에서 크기를 변화시키며 특징을 얻기에는 특징 집합이 너무 커지므로 위치에 제한을 두어 특징의 개수를 줄였다. 그림 2.1의 a와 b는 눈과 코의 위치, c와 d는 눈과 입, e는 입, f는 코의 위치로 제한하였다. 그리고 그림 2.2의 a, b, c, d는 눈과 코의 위치, e와 f는 눈의 위치, g와 h는 입의 위치에서 검출하도록 제한하여 전체 2,215개의 특징 집합을 구성하였다.

## 2.4 약한 분류기

훈련에 참여하는 각 클래스에 대하여 동일한 위치에서 동일한 모양의 Haar-like feature값의 평균과 클래스 4(비얼굴)의 평균의 1/2 값으로  $\theta$ 를 설정하였다. 따라서 2,215개의 Haar-like feature 집합에 대한 3개의 약한 분류기가 생성되므로 총 6,645개의 약한 분류기 집합을 얻을 수 있다.

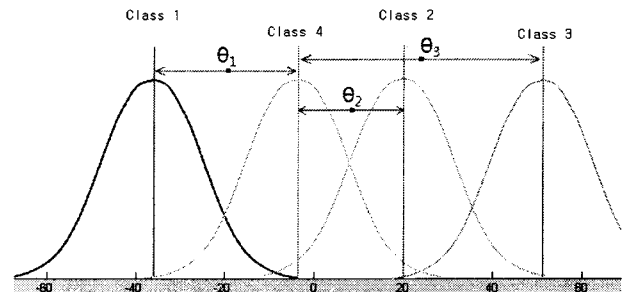


그림 3. 약한 분류기의 예

## 2.5 강한 분류기

매 훈련 반복마다 선택된 약한 분류기의 선형 결합으로 강한 분류기가 생성되면 새로운 입력 영상에서  $32 \times 32$  크기의 서브윈도우(sub-window)가 이동하며 강한 분류기로 검출을 한다. 강한 분류기의 파라미터 값으로는 선택된 약한 분류기들의  $\theta, a, b, k$  그리고 선택된 서브셋 종류를 지니고 있으며, 서브윈도우 안에서 약한 분류기들의 응답값을 반복 수

만큼 누적하여 클래스 1, 2, 3에 대한 강한 분류기 값이 생성된다. 따라서 이들에 대한 임계치(threshold)를 설정하여 이 값보다 작은 것들은 클래스 4로 분류하고, 이 값보다 큰 값들 중에서 가장 큰 값을 검출 결과로 나타낸다. 즉, 클래스 4는 따로 검출하지 않고, 각 클래스에 대한 강한 분류기 응답값에서 임계치보다 작은 값은 클래스 4로 분류한다.

### 3. 실험 결과

훈련에 사용된 데이터는 32x32 크기의 영상으로 클래스 1, 2, 3는 각각 200장씩 훈련에 참여시켰고, 클래스 4는 400장으로 총 1000장의 영상을 훈련에 참여시켰다. 그리고 훈련 반복은 150번을 수행하였다. 그 결과 40개의 특징들이 공유되었으며 110개의 특징들은 독립적으로 사용되었다. 훈련에 참여한 데이터 1000개와 참여하지 않은 데이터 1000개에 대한 훈련 반복수와 임계치에 따른 검출률은 그림 4와 같다. 훈련에 참여한 데이터의 경우 검출률은 97% 정도로 나왔으며, 오검출은 3% 정도로 나왔다. 훈련에 참여하지 않은 데이터에 대해서는 94%의 검출률을 보였으며, 오검출은 6% 정도로 나왔다. 그리고 그림 4.3은 AdaBoost를 이용하여 얼굴 700장과 비얼굴 1200장을 이용한 단일 스테이지의 실험 결과이다. 훈련셋과 비훈련셋에 대한 검출률은 각각 97%와 94% 정도로 위의 결과와 비슷하게 나왔다. 훈련에 참여한 데이터와 사용한 약한 분류기의 종류가 다르지만, 유사한 반복수와 성능에서 AdaBoost는 정면 얼굴만을 검출하지만 본 논문의 방법에서는 기울어진 얼굴도 검출할 수 있다는데 가장 큰 의의가 있다. 즉, AdaBoost로 기울어진 얼굴을 검출하려면 450번의 반복 수행을 해야 한다는 것이다.

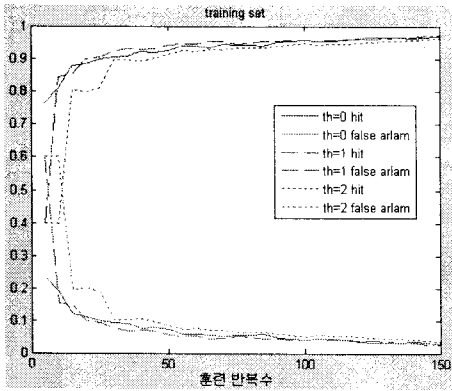


그림 4.1 훈련셋에 대한 검출률

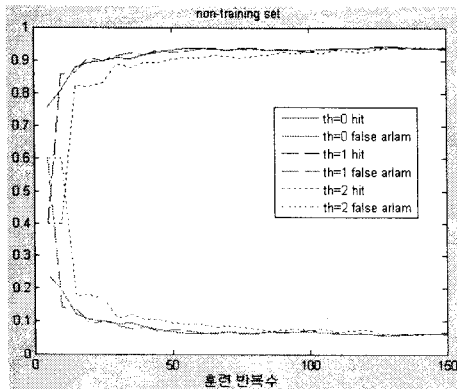


그림 4.2 비훈련셋에 대한 검출률

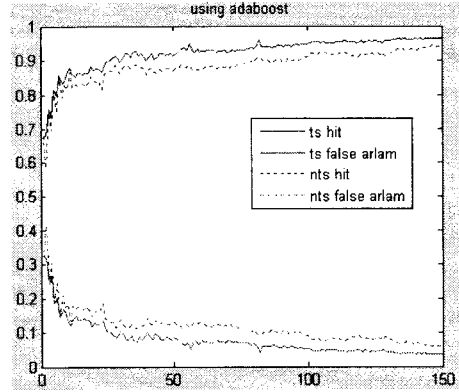


그림 4.3 AdaBoost의 검출률

그림 5의 경우에서처럼 학습에 참여하지 않은 새로운 영상에 대하여 본 논문의 방법에서는 기울어진 얼굴을 검출하였지만, AdaBoost의 경우 정면 얼굴만 검출하였다.



그림 5. 기울어진 얼굴 검출 예

### 4. 결론

본 논문에서는 AdaBoost를 응용한 JointBoost를 사용하여 특징들의 공유에 의한 기울어진 얼굴 검출 방법을 제안하였다. 제안된 방법은 AdaBoost를 이용한 얼굴 검출 방법에서 검출하지 못하는 기울어진 얼굴을 검출하는 방법을 제안하였으며, 특징들을 공유하는 방법을 사용하여 독립된 검출기로 기울어진 얼굴을 검출하는 방법보다 높은 성능과 적은 연산을 나타내었다. 하지만 훈련 시간은 AdaBoost를 이용한 방법보다 수배의 시간이 소요되는 단점을 지니고 있다. 따라서 훈련 시간을 효과적으로 줄이도록 개선해야 하는 추후 연구 과제가 남아 있다. 그리고 본 연구를 응용한다면, 얼굴이 아닌 다른 사물의 다각도, 다분류에 대한 검출기를 설계할 수 있을 것이다.

### 참고 문헌

- [1] P. Viola and M. Jones, "Robust real-time face detection", Computer Vision, 2001. ICCV 2001. Proc. Eighth IEEE International Conference on, vol. 20, pp. 1254 - 1259, July 2001.
- [2] R. Schapire and Y. Singer. "BoosTexter: A boosting-based system for text categorization". Machine Learning, vol. 39, pp. 135-168, 2000.
- [3] A. Torralba, K. P. Murphy and W. T. Freeman, "Sharing visual features for multiclass and multiview object detection", PAMI 2007. vol. 29(5), pp. 854-869, 2007.