

실시간 객체 검출을 위한 개선된 Haar-like Feature 정규화 방법

정회원 박기영*, 종신회원 황선영*

An Improved Normalization Method for Haar-like Features for Real-time Object Detection

Ki-Yeong Park* *Regular Member*, Sun-Young Hwang* *Lifelong Member*

요 약

본 논문에서는 객체 검출에 사용되는 Haar-like feature의 정규화 방법에 대해 다룬다. 기존의 Haar-like feature의 분산 정규화는 후보 윈도우 픽셀들에 대한 표준편차 계산에 사용되는 별도의 적분 영상 생성을 위해 많은 연산을 필요로 했으며 밝기 변화가 작은 영역에서 오검출이 증가하는 문제를 가지고 있으나, 제안하는 정규화 방법은 별도의 적분 영상을 사용하지 않아 처리 속도가 빠르며, 제안하는 방법을 사용하여 학습시킨 분류기는 밝기 변화에 대해 강건한 성능을 보인다. 실험 결과 제안한 방법을 사용했을 때 객체 검출기의 처리 속도는 26% 향상되었으며, 제안한 방법을 사용하여 학습시킨 분류기들은 5% 이상 향상된 검출률을 보였으며, 밝기 변화가 심한 경우는 45% 향상된 검출률을 보였다.

Key Words : Haar-like feature, Normalization, Classifier, Object detection, Real-time system

ABSTRACT

This paper describes a normalization method of Haar-like features used for object detection. Previous method which performs variance normalization on Haar-like features requires a lot of calculations, since it uses an additional integral image for calculating the standard deviation of intensities of pixels in a candidate window and increases possibility of false detection in the area where variance of brightness is small. The proposed normalization method can be performed much faster than the previous method by not using additional integral image and classifiers which are trained with the proposed normalization method show robust performance in various lighting conditions. Experimental result shows that the object detector which uses the proposed method is 26% faster than the one which uses the previous method. Detection rate is also improved by 5% without increasing false alarm rate and 45% for the samples whose brightness varies significantly.

I. 서 론

객체 검출기는 영상으로부터 얼굴, 보행자, 차량과 같은 특정한 종류의 객체를 찾아낸다. 이와 같은 객체는 형태, 색상 등이 다양하여 동일 분류 안에서의 변

동성(in-class variability)이 크며, 크기나 개수를 미리 알 수 없고, 배경이 특정되지 않아 검출이 어렵다^[1]. 초기에는 교통 표지판이나 얼굴과 같이 형태 변화가 적은 객체의 검출을 위해 사전 정보를 이용하여 객체의 템플릿을 만들고 이 템플릿과 비교하여 객체를 찾

※ 본 연구는 부품소재기술개발사업(10032575) 지원 및 한국산업기술평가관리원 관리로 수행되었습니다.

* 서강대학교 전자공학과 CAD & ES 연구실 (hwang@sogang.ac.kr)

논문번호 : KICS2011-03-145, 접수일자 : 2011년 3월 12일, 최종논문접수일자 : 2011년 8월 1일

는 방법이 사용되었으니^{2,3}, 형태 변화가 큰 객체의 검출에는 적절하지 않으므로 최근에는 객체 샘플 데이터로 객체 검출기를 학습시키는 연구가 많이 이루어지고 있다⁴⁻¹⁰. 객체 검출기는 디지털 카메라나 감시 카메라 등에 응용되며, 최근에는 차량에 장착된 카메라의 영상으로부터 보행자나 차량을 검출하는 연구가 활발히 이루어지고 있다¹¹⁻¹⁵.

객체 검출기가 실용적으로 활용되기 위해서는 처리 속도가 중요하다. Viola와 Jones⁹는 Haar-like feature와 다단계 분류기를 이용하는 객체 검출기를 제안하였다. Haar-like feature는 적분 영상(Integral image)을 이용하여 객체의 크기와 상관없이 일정한 연산량으로 빠르게 계산할 수 있으며, 단계마다 연산이 점점 복잡해지도록 구성된 다단계 분류기는 객체가 아닌 대부분의 영역을 초기 단계에서 제외시키고 객체일 가능성이 높은 일부 영역들에 대해서만 여러 단계의 연산을 수행하므로 처리 속도가 빠르다. 이렇게 구성된 객체 검출기는 700MHz PC에서 384×288 픽셀 크기의 영상에 대해 윈도우 이동(Sliding window) 기법만으로 얼굴을 초당 15 프레임의 속도로 검출하여 실시간으로 객체를 검출할 수 있는 길을 열었으며¹⁶, 스크립트 언어를 사용하는 웹브라우저나 휴대폰과 같은 저사양의 환경에 적용된 사례도 있다^{17,18}.

영상의 영역별로 발생하는 밝기 변화는 객체 검출기의 성능에 영향을 미치므로 객체 검출기의 성능을 높이기 위해서는 영역별 밝기 차이에 대한 보정이 필요하다. 특히 Haar-like feature는 사각형 영역들의 밝기 차를 이용하므로 밝기 변화에 많은 영향을 받는다. Viola와 Jones는 이 영향을 줄이기 위해 학습과 검출 과정에서 Haar-like feature를 분산 정규화(Variation normalization)하여야 한다고 제시했다⁹. 그러나 분산 정규화는 표준편차 계산을 위한 별도의 적분 영상을 필요로 하여 객체 검출기의 처리 속도에 많은 영향을 미친다. TI사의 600MHz DSP TMS320DM6437에서 720×480 픽셀 크기의 적분 영상 생성에 11.2 ms가 소요되었다는 결과가 발표되었으며¹⁹, 이는 초당 30 프레임의 동영상에서 한 프레임의 1/3에 해당하는 시간이다.

본 연구에서는 실시간 객체 검출기에서 사용하기 위하여 기존에 비해 연산량을 줄인 Haar-like feature 정규화 방법을 제안한다. 제안하는 방법은 별도의 적분 영상을 필요로 하지 않고 기존보다 계산이 간단하며 영상의 밝기 변화에 대해서 강건한 성능을 보인다.

본 논문의 구성은 다음과 같다. 2절에서는 객체 검출기의 기본적인 구성과 Viola와 Jones가 제안한 객

체 검출 방법 및 Haar-like feature 정규화 방법에 대해 설명한다. 3절에서는 기존 정규화 방법이 처리 속도에 미치는 영향과 밝기 변화가 적은 영역에서 오검출이 증가하는 문제를 분석하고 새로운 정규화 방법을 제안한다. 4절에서는 기존의 정규화 방법과 제안한 방법을 사용한 객체 검출기의 처리 속도를 비교한 실험 결과를 제시하고 각각의 정규화 방법으로 학습시킨 분류기들의 성능을 비교한 실험 결과를 제시한다. 마지막으로 5절에서는 결론과 추후 과제에 대해서 기술한다.

II. Background 및 관련 연구

본 절에서는 객체 검출기의 구조에 대해서 설명하고, Viola와 Jones의 객체 검출기를 구성하는 Haar-like feature, 적분 영상, 그리고 다단계 분류기에 대해 설명한다. 또한 이들이 제안한 Haar-like feature의 정규화 방법에 대해 설명한다.

2.1 연구 배경

객체 검출기는 영상에서 객체가 위치할 수 있는 후보 영역을 선택하는 후보 선정 (Hypothesis generation) 단계와 분류기를 사용하여 후보 선정 단계에서 선택한 영역이 객체인지 배경인지를 결정하는 후보 검증 (Hypothesis verification) 단계로 구성된다¹³. Viola와 Jones의 객체 검출기는 객체의 크기에 해당하는 여러 크기의 윈도우들을 영상 전 영역에 대해 이동시키며 모든 윈도우 영역을 후보로 사용하는 윈도우 이동 기법을 사용하여 후보를 선정하며, Haar-like feature를 사용하는 다단계 분류기로 선정된 후보가 객체인지를 결정한다.

2.1.1 Haar-like feature

사람이 쉽게 같은 종류라고 인식하는 객체들도 각각의 크기와 모양 등이 다양하기 때문에 이들의 공통적인 특징을 기술하는 것은 어려운 과제이다. Papageorgiou et al.^{11,51}은 객체의 공통적인 특징의 기술을 위해 Haar wavelet의 사용을 제안하였고, Viola와 Jones⁹는 이를 확장시킨 Haar-like feature를 제안하였다. 특히, 이들은 AdaBoost 알고리즘²⁰⁻²²을 사용하여 과완전(over-complete) Haar-like feature 집합⁵¹으로부터 분별력이 높은 일부를 선택해 다단계 분류기를 구성하는 방법과 적분 영상을 사용하여 Haar-like feature를 계산하는 방법을 제시하였다. Haar-like feature는 그림 1-(a)와 같이 2개 이상의 인

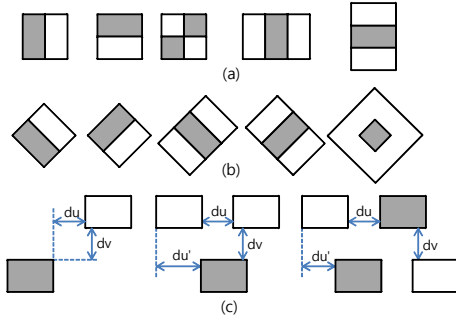


그림 1. Haar-like feature의 예. (a) Papageorgiou et al.의 Haar wavelet과 Viola and Jones의 Haar-like feature, (b) Lienhart et. al의 45° 회전시킨 Haar-like feature, (c) Li et. al의 disjoint Haar-like feature.

접한 사각형 영역들로 구성되며, 회색과 흰색으로 표시된 영역들 간의 밝기 차로 그 값이 정의된다⁹⁾.

여러 연구에서 다양한 형태의 Haar-like feature들이 제안되었다. Lienhart et al.^[23]은 그림 1-(b)의 45° 회전시킨 Haar-like feature를 제안하였고, Li et al.^[24]은 그림 1-(c)의 인접하지 않은 사각형들로 구성된 disjoint Haar-like feature를 제안하였다. Mita et al.^[25]은 여러 개의 Haar-like feature들을 결합한 joint Haar-like feature를 제안하였고, Zheng et al.^[15]은 edge나 ridge 형태의 패턴을 나타내기 위하여 Haar-like feature를 응용한 Image strip feature의 사용을 제안하였다.

2.1.2 적분 영상

적분 영상을 사용하면 Haar-like feature를 구성하는 사각형 영역의 픽셀들의 합을 빠르게 계산할 수 있다⁹⁾. 적분 영상은 식 (1)과 같이 (x,y) 위치의 픽셀 $ii(x,y)$ 가 입력 영상의 원점 $i(0,0)$ 으로부터 해당 위치의 픽셀 $i(x,y)$ 까지의 사각형 영역에 있는 모든 픽셀들의 합을 갖는다²⁶⁾.

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

적분 영상을 사용하면 입력 영상의 사각형 영역의 픽셀들의 합을 적분 영상의 4개 픽셀만으로 계산할 수 있다. 그림 2는 적분 영상을 사용하여 입력 영상의 사각형 영역의 픽셀 합을 계산하는 방법을 보인다. 적분 영상의 픽셀 ii_4 는 입력 영상 A+B+C+D 영역의 합을 가지고 있고, 픽셀 ii_2 는 A+B 영역의 합을, 픽셀 ii_3 는 A+C 영역의 합을, 픽셀 ii_1 은 A 영역의 합을 가지고 있으므로, 입력 영상의 D 영역은 적분 영상에서

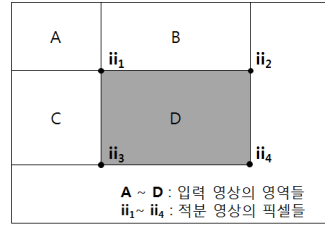


그림 2. 적분 영상을 이용한 사각형 영역 픽셀들의 합 계산. D 영역의 합은 $ii_4 - ii_2 - ii_3 + ii_1$.

$ii_4 - ii_2 - ii_3 + ii_1$ 를 계산하여 구할 수 있다⁹⁾.

다양한 크기의 객체를 검출하기 위하여 영상을 여러 크기로 축소시킨 영상 피라미드^[11]를 사용하는 경우가 많지만, Haar-like feature는 크기에 따른 연산량의 차이가 없으므로 Viola와 Jones의 객체 검출기는 영상을 축소하는 대신에 Haar-like feature를 확대하여 효율적으로 다양한 크기의 객체를 검출할 수 있다.

2.1.3 다단계 분류기

Viola와 Jones는 객체 분류기를 그림 3과 같이 단계로 구성하였다⁹⁾. 각 단계는 AdaBoost 알고리즘을 이용하여 과완전 Haar-like feature 집합으로부터 객체와 배경을 가장 잘 구별하는 소수의 Haar-like feature들을 선택하여 객체의 대부분을 통과시키고 배경의 절반 정도를 걸러내도록 학습되며, 이런 단계들을 여러 개 연결하여 다단계 분류기를 구성한다. 다단계 분류기는 영상의 많은 부분을 차지하는 배경 영역을 초기 단계들에서 제외시키고 객체일 가능성이 높은 일부 영역에 대해서만 여러 단계를 거치도록 하여 효율적으로 객체를 검출할 수 있다.

그림 4는 다단계 분류기의 각 단계의 구성을 보인다²⁷⁾. 각 단계는 Haar-like feature HF를 사용하는 여러 개의 판정 트리로 구성된다. 각 판정 트리는 HF와 경계치 t 를 비교하여 좌 L 또는 우 R 값을 결정하며, 결정된 좌, 우 값들을 합한 후, 단계의 경계치 T와 비교하여 후보 윈도우가 객체인지 배경인지를 판정한다. 각 단계에서 후보 윈도우가 배경으로 판정되면 판

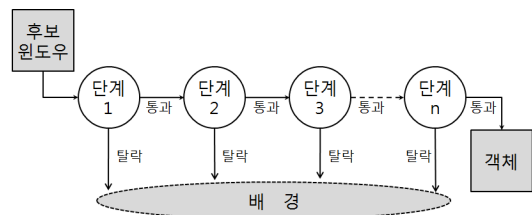


그림 3. 다단계 객체 분류기의 구성.

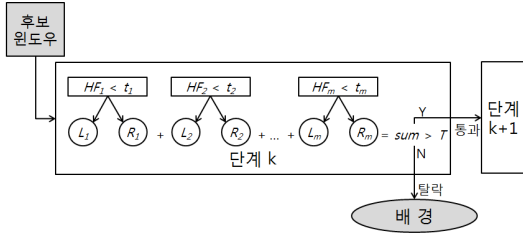


그림 4. 다단계 분류기의 각 단계의 구성.

정은 그 단계에서 종료되고, 객체로 판정되면 객체 여부에 대한 판정이 다음 단계에서 반복된다. 이렇게 마지막 단계까지 모두 통과한 후보 윈도우만이 최종적으로 객체로 결정된다.

2.2 Haar-like feature 정규화

조명에 의해 영상의 영역별로 밝기의 차이가 생길 수 있으므로 Haar-like feature를 정규화하여 밝기를 보정해야 한다. Viola와 Jones는 분류기 학습과 객체 검출 과정에서 윈도우의 표준편차를 구하여 Haar-like feature들을 분산 정규화 해야 한다고 제시했으며, 표준편차의 빠른 계산을 위해 픽셀 제공에 대한 별도의 적분 영상을 사용하는 방법을 제시하였다⁹⁾. Lienhart et al.도 표준편차와 평균을 사용하여 Haar-like feature의 밝기를 보정하는 방법을 제시하였다²³⁾.

윈도우의 표준편차를 이용한 Haar-like feature의 정규화는 식 (2)와 같이 계산된다²³⁾. μ 는 윈도우의 평균, σ 는 윈도우의 표준편차, $i(x,y)$ 는 정규화하지 않은 픽셀, $\bar{i}_\sigma(x,y)$ 는 표준편차를 이용하여 정규화한 픽셀이다.

$$\bar{i}_\sigma(x,y) = \frac{i(x,y) - \mu}{\sigma}, \quad c \in R^+ \quad (2)$$

표준편차는 픽셀 $i(x,y)$ 에 대한 적분 영상 I 외에 픽셀 제공 $i^2(x,y)$ 에 대한 적분 영상 I_{sq} 를 사용하여 빠르게 계산할 수 있다⁹⁾. 윈도우의 픽셀 수가 N 일 때 윈도우의 평균 μ 는 식 (3)과 같이 계산되며 표준편차 σ 는 식 (4)와 같이 계산된다.

$$\mu = \frac{1}{N} \sum_{x,y \in window} i(x,y) \quad (3)$$

$$\sigma^2 = \mu^2 - \frac{1}{N} \sum_{x,y \in window} i^2(x,y) \quad (4)$$

윈도우의 평균 μ 의 계산에는 적분 영상 I 가 사용

되며, 픽셀 제공의 합 $\sum i^2(x,y)$ 의 계산에는 적분 영상 I_{sq} 가 사용된다. Haar-like feature의 정규화는 식 (2)와 같이 윈도우의 각 픽셀들에 대해서 적용하지 않고, 식 (6)과 같이 Haar-like feature 값에 대해 직접 적용해도 동일한 결과를 얻는다. Haar-like feature HF 는 식 (5)와 같이 계산되며, 식 (6)에서 표준편차를 사용하여 정규화한 Haar-like feature HF_σ 는 각 픽셀을 정규화한 것과 동일한 결과를 얻는다. N 은 윈도우의 픽셀 수이고, r_i 와 w_i 는 각각 Haar-like feature를 구성하는 사각형 영역과 그 가중치이다.

$$HF = \frac{1}{N} \sum_i r_i w_i = \frac{1}{N} \sum_i w_i \sum_{x,y \in r_i} i(x,y) \quad (5)$$

$$\begin{aligned} HF_\sigma &= \frac{1}{\sigma} HF = \frac{1}{N} \sum_i \frac{1}{\sigma} r_i w_i \\ &= \frac{1}{N} \sum_i w_i \sum_{x,y \in r_i} \frac{1}{\sigma} i(x,y) \\ &= \frac{1}{N} \sum_i w_i \sum_{x,y \in r_i} \bar{i}_\sigma(x,y) \end{aligned} \quad (6)$$

III. 제안하는 Haar-like feature 정규화

본 절에서는 기존 정규화 방법에서 사용되는 적분 영상의 연산량과 메모리 사용량에 대해 분석하고, 기존 정규화 방법을 사용했을 때 오검출이 증가하게 되는 경우에 대해 분석한다. 그리고 별도의 적분 영상을 사용하지 않는 개선된 정규화 방법을 제안한다.

3.1 기존의 정규화 방법

기존의 분산 정규화는 표준편차 계산을 위한 별도의 적분 영상을 필요로 한다. 적분 영상은 생성에 많은 연산이 필요하고 메모리를 많이 사용하므로 객체 검출기의 처리 속도에 적지 않은 영향을 미친다. 또한 분산 정규화는 밝기 변화가 적은 영역에서 오검출이 증가하게 되는 문제점을 가지고 있다.

3.1.1 적분 영상의 연산량과 메모리 사용량

표준편차를 이용하는 분산 정규화는 표준편차 계산을 위해 적분 영상 I 와 적분 영상 I_{sq} 를 사용한다. 적분 영상을 식 (1)과 같이 픽셀 별로 계산하면 너무 많은 연산이 필요하므로, 식 (7)과 식 (8)의 재귀적 방법을 사용하여⁹⁾ $M \times N$ 픽셀 크기의 영상에 대해 $2MN$ 번의 덧셈 연산으로 적분 영상을 생성할 수 있다²⁸⁾. $S(x,y)$ 는 누적된 행의 합이고 $S(x,-1)=0$,

$ii(-1,y)=0$ 이다.

$$S(x,y) = S(x,y-1) + i(x,y) \quad (7)$$

$$ii(x,y) = ii(x-1,y) + S(x,y) \quad (8)$$

TI사의 600MHz DSP TMS320DM6437에서 더블 버퍼링을 사용하지 않을 때 720×480 픽셀 크기의 적분 영상 I 의 계산에 재귀적 방법을 사용하여도 11.2ms가 소요되었다는 결과가 발표되었는데^[9], 이는 초당 30 프레임의 동영상에서 한 프레임 처리 시간의 1/3에 해당한다. 정규화를 위해 추가적으로 필요한 적분 영상 I_{sq} 의 경우는 2MN 번의 덧셈 연산 외에 각 픽셀을 제공하는 MN 번의 곱셈 연산이 더 필요하므로 별도의 적분 영상을 만드는 것은 실시간 객체 검출기의 처리 속도에 적지 않은 영향을 미치게 된다.

적분 영상의 픽셀은 입력 영상 한 프레임의 합을 저장할 수 있는 크기여야 한다. 적분 영상 I 는 수메가 픽셀 크기의 8비트 영상에 대해서도 픽셀 당 4바이트의 메모리로 충분하지만, 기존 정규화 방법에서 사용하는 적분 영상 I_{sq} 는 픽셀 제공들의 합이므로 320×240 픽셀 크기의 영상에 대해서도 픽셀 당 4바이트로 충분하지 않으므로 대부분의 경우 픽셀 당 8바이트를 사용해야 한다. 적분 영상을 이용하여 사각형 영역의 합을 계산하기 위해 네 개의 픽셀을 읽을 때에 메모리에서 네 픽셀들이 너무 떨어져 있으면 메모리를 랜덤으로 읽는 모습이 되므로 캐시를 효율적으로 사용할 수 없다. 적분 영상의 픽셀 크기가 커지며 이런 가능성은 더욱 늘어난다.

3.1.2 분산 정규화의 문제점

Haar-like feature는 영역들 간의 밝기 차를 사용하므로 후보 윈도우의 표준편차가 클수록 Haar-like feature의 절대값이 커지는 경향을 보인다. 그러나 표준편차가 작은 경우에 표준편차로 나누어 정규화하면 밝기 차가 적어도 Haar-like feature의 절대값이 커지

게 되므로 오검출이 증가하게 된다. 그림 5는 기존의 정규화 방법을 사용한 분류기로 차량을 검출한 경우를 보인다. 하늘이나 도로면 같이 밝기 변화가 작은 영역에서 오검출이 많이 발생하는 것을 볼 수 있다. 밝기 변화가 작은 영역에서 오검출이 증가하는 문제는 4절의 테스트 샘플 밝기별 분류기 성능비교 실험에서도 확인되었다.

3.2 제안하는 정규화 방법

기존의 정규화 방법은 표준편차 계산을 위한 별도의 적분 영상을 사용하여 처리 속도에 적지 않은 영향을 미칠 수 있고, 표준편차가 작은 영역에서 오검출이 증가하는 문제가 있으므로, 본 연구에서는 Haar-like feature를 윈도우의 평균으로 나누는 정규화 방법을 제안한다.

영역들 간의 밝기 차는 어두운 영역에서 작고 밝은 영역에서 큰 경향이 있어 Haar-like feature의 절대값도 어두운 영역에서 작아지고 밝은 영역에서 커지게 되므로 Haar-like feature를 윈도우의 평균으로 나누어 밝기 변화에 대한 보정을 할 수 있다. 또한, 표준편차로 정규화하는 경우는 윈도우의 표준편차가 작으면 윈도우 영역의 밝기와 상관없이 정규화한 Haar-like feature 값이 커지는 문제가 있었지만, 평균으로 정규화하면 윈도우의 영역이 어두운 경우만 Haar-like feature 값이 커지므로 정규화로 인해 오검출이 증가하게 될 가능성이 줄어든다.

윈도우의 평균 μ 는 적분 영상 I 만을 사용하여 계산할 수 있으며 정규화한 픽셀 값 $\bar{i}_\mu(x,y)$ 는 식 (9)와 같이 계산할 수 있다. 또한, 제안한 정규화는 기존 방법과 마찬가지로 윈도우의 각 픽셀에 적용하지 않고 Haar-like feature에 직접 적용할 수 있으며, 식 (3)과 식 (5)에서 N 이 소거되므로 정규화한 값 HF_μ 는 μ 를 계산하지 않고 식 (10)과 같이 윈도우의 합으로부터 직접 계산할 수 있다.

$$\bar{i}_\mu(x,y) = \frac{i(x,y)}{\mu} \quad (9)$$

$$HF_\mu = \frac{1}{\mu} HF = \frac{\sum_i r_i w_i}{\sum_{x,y \in window} i(x,y)} \quad (10)$$

IV. 실험 결과

본 절에서는 정규화하지 않은 경우와 기존의 방법

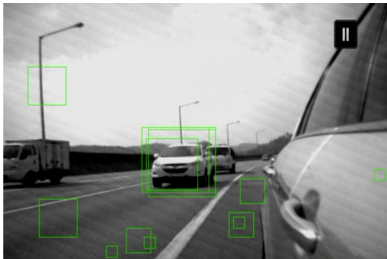


그림 5. 표준편차가 작은 영역에서 오검출이 발생한 예.

과 제안한 방법으로 정규화한 경우의 Haar-like feature 값의 분포를 비교하고, 기존의 방법과 제안한 방법을 사용할 때의 객체 검출기의 처리 속도와 각각의 방법으로 학습시킨 분류기들의 성능을 비교한 실험 결과를 제시한다. 또한 영상의 밝기 변화에 대한 분류기의 성능 변화를 확인하기 위하여 테스트 샘플들의 밝기를 여러 단계로 변경한 후에 분류기의 성능을 측정하는 결과를 제시한다.

4.1 정규화한 Haar-like feature 값의 분포 비교

밝기가 다양한 샘플들의 Haar-like feature 값의 분포를 확인하기 위하여 차량의 전면을 학습시킨 분류기에서 첫 번째와 두 번째의 Haar-like feature를 선택하고, 학습에 사용된 포지티브 샘플 중 500개에 대해서 Haar-like feature의 값을 계산하였다.

그림 6은 정규화하지 않은 Haar-like feature 값의 분포를 보인다. 포지티브 샘플들의 밝기는 평균 36~172이며, Haar-like feature들은 샘플들의 밝기가 밝아질수록 절대값이 커지는 분포를 보인다. 그림 7은 표준편차로 정규화한 Haar-like feature HF_{σ} 와 평균으로 정규화한 HF_{μ} 를 각각의 표준편차로 나누고 평균이 0이 되도록 변환한 후에 비교한 결과를 보인다. 밝기가 다양한 샘플들에 대해 HF_{μ} 값들이 HF_{σ} 와 마찬가지로 일정하게 유지되는 분포를 보여, 제안한 방법이 Haar-like feature의 정규화에 사용될 수 있음을 보인다.

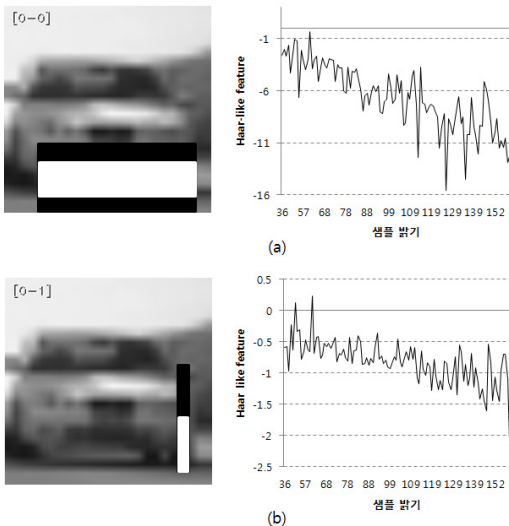


그림 6. 정규화하지 않은 Haar-like feature 값의 분포. (a) 분류기의 첫 번째 Haar-like feature 값의 분포, (b) 분류기의 두 번째 Haar-like feature 값의 분포.

4.2 정규화 처리 속도 비교

TI사의 600MHz DSP TMS320DM6437에서 객체 검출기를 구현하여 정규화 처리 속도를 비교하였다. 720x480 픽셀 크기의 영상에 대해 객체 검출기의 프레임당 평균 처리 시간, 적분 영상 생성 소요 시간, 표준편차 또는 픽셀 합계 계산 소요 시간, 그리고 Haar-like feature 계산 및 판정 소요 시간을 각각 측정하였다. 프레임 당 평균 처리 시간은 60초 분량의 영상을 처리한 후 평균을 계산하였으며, 적분 영상 생성 시간은 300회 생성 후에, 표준편차 또는 픽셀 합계 계산은 20만회 이상 실행 후에, Haar-like feature 계산 및 판정은 2백만 회 이상 실행 후에 평균 시간을 계산하였다.

한 프레임 당 평균 처리 시간은 기존 방법을 사용한 경우에 117ms가 소요되었고 제안한 방법을 사용한 경우에 87ms가 소요되어 26% 개선되었다. 표 1은 적분 영상 생성, 정규화 값 계산, 그리고 Haar-like feature 계산 및 판정에 소요된 시간을 보인다. 적분 영상 생성 소요 시간 T_{ii} 는 기존 방법의 경우는 세계의 적분 영상 I , I_{rot} , I_{sq} 를 생성하는데 소요된 시간이며, 제안한 방법의 경우는 I_{sq} 를 제외한 I , I_{rot} 의 생성에 소요된 시간이다. T_{norm} 은 후보 윈도우의 표준편차 또는 픽셀 합계 계산에 소요된 시간이다. T_{haar} 는 Haar-like feature를 계산하여 정규화하고 경계치와 비교하는데 소요된 시간이다. 제안한 방법을

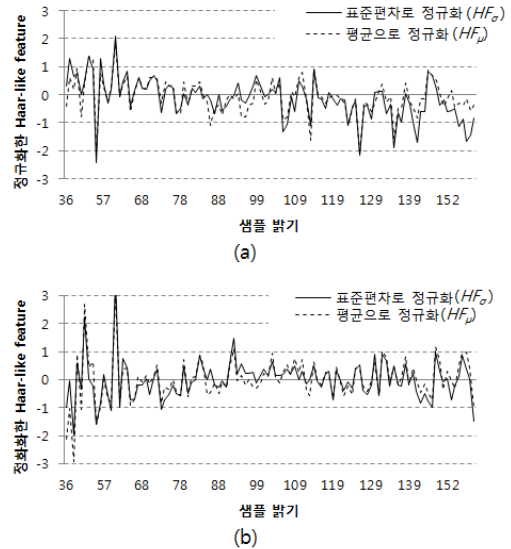


그림 7. 기존 방법과 제안한 방법으로 정규화한 Haar-like feature 값의 분포. (a) 첫 번째 Haar-like feature 값의 분포, (b) 두 번째 Haar-like feature 값의 분포.

사용했을 때 T_{ii} 는 17 ms가 줄어들어 38% 개선되었고, T_{norm} 은 5.7배 빨라졌으며 T_{haar} 는 9% 개선되었다.

4.3 정규화 방법에 따른 분류기 성능 비교

다양한 환경의 영상에서 정규화 방법에 따른 분류기의 성능 차이를 비교하기 위해서 얼굴 정면, 차량 전면, 차량 후면, 야간 보행자에 대한 분류기를 학습시켜 성능을 평가하였다.

4.3.1 샘플 준비 및 분류기 학습

학습에 필요한 얼굴 정면 샘플은 MIT CBCL Face Database #1^[29]의 학습 샘플 2,429개 전체를 사용하였으며, 차량 전면 및 야간 보행자 샘플은 직접 촬영한 동영상으로부터 각각 2,583개와 1,000개를 준비하였고, 차량 후면 샘플은 직접 촬영한 동영상과 정지 영상으로부터 2,119개를 준비하였다. 샘플의 크기는 얼굴 정면의 경우는 19×19 픽셀, 차량 전면은 24×24 픽셀, 차량 후면은 20×20 픽셀, 보행자는 14×28 픽셀을 사용하였다.

성능 평가용 얼굴 정면 샘플은 Caltech Faces 1999 (Front)^[30]중에서 얼굴 그림과 얼굴의 일부가 가려진 4개를 제외한 446개와, MIT CBCL Face Recognition Database^[31]의 얼굴 정면과 12도 이내로 회전한 얼굴 샘플 1,440개를 사용하였다. Caltech 샘플은 일반적인 실내외 환경에서 촬영된 사진들로 실내에서는 주로 플래시를 사용하여 얼굴이 밝게 촬영된 경우가 많다. MIT CBCL 샘플들은 조명의 방향과 시점(view point)을 3D 합성한 것으로 조명 방향이 다양하며, 배경 부분이 검게 처리되어 오검출이 거의 발생하지 않는다. 차량 전면과 야간 보행자는 학습 샘플과 동일한 카메라로 촬영한 동영상으로부터 2,620개와 1,071개를 준비하였으며, 차량 후면은 학습 샘플과 다른 카메라로 촬영한 동영상과 인터넷에서 수집한 영상들로부터 1,565개를 준비하였다.

분류기의 학습과 평가를 위해서 OpenCV2.2^[32]에 포함된 학습 도구인 opencv_haartraining과 성능 평가 도구인 opencv_performance를 수정하여 사용하였다. 분류기는 각 단계의 검출률과 오검출률을 각각 99.7%

와 50%로 설정하여 20단계씩 학습시켰으며 학습 알고리즘으로는 Gentle AdaBoost^[33]를 사용하였다.

4.3.2 정규화 방법에 따른 분류기 성능 비교

테스트 샘플을 사용하여 분류기들의 검출수와 오검출수를 측정하고 식 (11), 식 (12)와 같이 검출률과 오검출률을 계산하였다.

$$\text{검출률} = \frac{\text{검출 회수}}{\text{테스트 샘플 개수}} \quad (11)$$

$$\text{오검출률} = \frac{\text{오검출 회수}}{\text{테스트 샘플 개수}} \quad (12)$$

그림 8은 테스트 샘플들을 사용하여 평가한 분류기들의 ROC (Receiver Operating Characteristic) 커브로 제안한 정규화를 사용한 분류기들이 5% 이상 높은 검출률을 보인다. (a)와 (b)는 동일한 얼굴 정면 분류기를 다른 테스트 샘플들을 사용하여 평가한 것으로, 기존 정규화를 사용한 분류기는 조명의 방향이 다양하게 합성된 MIT CBCL 샘플에서 큰 성능 저하를 보이지만, 제안한 정규화를 사용한 분류기는 큰 차이를 보이지 않고 조명 변화에 대해 강건한 성능을 보인다. 동일한 카메라를 사용하여 촬영한 동영상으로부터 학습 샘플과 테스트 샘플을 제작한 (c)와 (e)의 경우는 분류기의 성능 차이가 근소하나, 서로 다른 카메라를 사용한 (d)의 경우는 동일한 오검출률에서 검출률이 12% 정도 차이를 보여, (a)와 (b)의 경우와 함께 학습

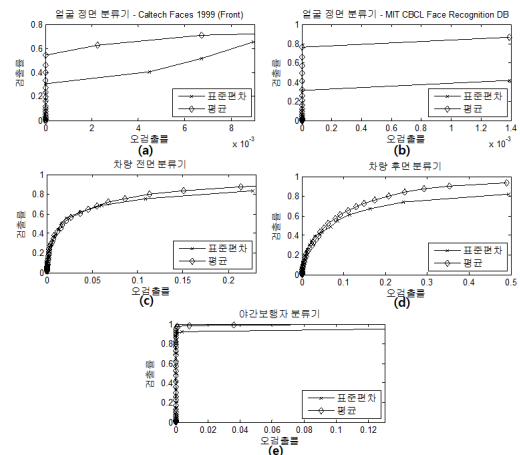


그림 8. 기존 방법과 제안한 방법을 사용하여 학습시킨 분류기들의 ROC 커브. (a) Caltech Faces 1999 (Front) 샘플을 사용한 얼굴 정면 분류기 평가 결과, (b) MIT CBCL Face Recognition Database 샘플을 사용한 얼굴 정면 분류기 평가 결과, (c) 차량 전면 분류기 평가 결과, (d) 차량 후면 분류기 평가 결과, (e) 야간 보행자 분류기 평가 결과.

표 1. 정규화 처리 속도 비교

	기존 정규화	제안한 정규화
T_{ii}	45 ms	28 ms
T_{norm}	13.0 μ s	2.3 μ s
T_{haar}	5.8 μ s	5.3 μ s

샘플과 테스트 샘플이 전혀 상관없는 경우에 제안한 정규화 방법이 좀 더 우수한 성능을 보인다.

4.4 영상의 밝기 변화에 대한 분류기 성능 변화 비교

영상 밝기 변화에 따른 분류기의 성능 변화를 확인하기 위하여 테스트 샘플들의 밝기를 그림 9와 같이 +3EV, +2EV, +1EV, -1EV, -2EV, -3EV씩 변경한 후에 분류기의 성능을 평가하여 원본 테스트 샘플에 대한 평가 결과와 비교하였다.

그림 10은 여러 단계로 밝기를 변경한 테스트 샘플들에 대한 분류기들의 성능 변화를 보인다. (a)의 경우 기존 방법과 제안한 방법을 사용한 분류기 모두 -2EV 또는 -3EV에서 최고의 검출률을 보이며 밝은 샘플에서는 급격한 성능 저하를 보인다. 이는 Caltech Faces 1999 (Front)가 플래시를 사용하여 밝게 촬영한 샘플



그림 9. 여러 단계로 밝기를 변경한 테스트 샘플의 예.

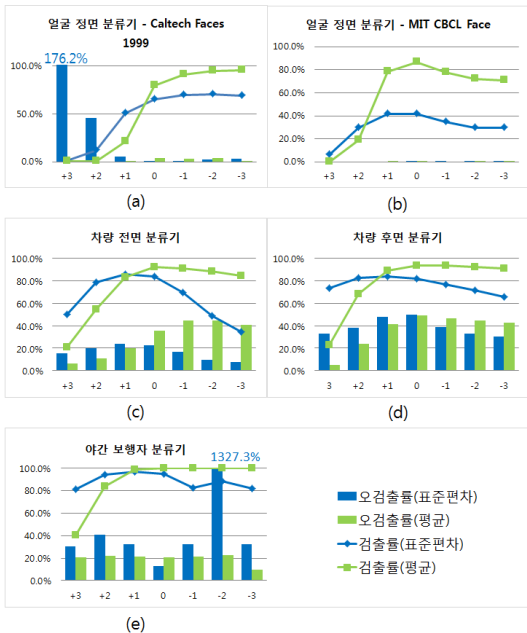


그림 10. 테스트 샘플의 밝기를 +3EV, +2EV, +1EV, 0, -1EV, -2EV, -3EV 변경하였을 때 분류기의 성능 변화. (a) Caltech Faces 1999 (Front) 사용 얼굴 정면 분류기 평가 결과, (b) MIT CBCL Face Recognition Database 사용 얼굴 정면 분류기 평가 결과, (c) 차량 전면 분류기 평가 결과, (d) 차량 후면 분류기 평가 결과, (e) 야간 보행자 분류기 평가 결과.

들을 많이 포함하고 있기 때문인 것으로 보인다. (a)를 제외하면 +1EV에서 -3EV 사이의 밝기에서 제안한 방법을 사용한 분류기가 기존 방법을 사용한 분류기보다 좋은 성능을 보이며, 특히 어렵게 조정된 샘플들에서 성능 변화가 적은 강건한 모습을 보인다. 밝게 조정된 샘플들에서는 제안한 방법을 사용한 분류기들이 큰 폭의 성능 저하를 보이지만, 기존 방법을 사용한 경우도 (a), (b), (c)에서 큰 폭의 성능 저하를 보이며, (d), (e)에서도 분명한 성능 저하를 보이므로 기존 방법이 밝은 샘플들에 대해 우수하다고 할 수는 없다.

그림 10-(a)의 +3EV, +2EV의 경우와 (e)의 -2EV의 경우에 기존 방법을 사용한 분류기들의 오검출률이 큰 폭으로 증가하는 것을 보인다. 이는 표준편차로 나누어 정규화할 때 밝기 변화가 작은 영역들에서 오검출이 증가하는 현상으로 (a)의 경우는 배경이 밝아지면서 표준편차가 작아져서 오검출이 증가한 예이며, (e)의 경우는 배경이 어두워지면서 오검출이 증가한 경우이다. (e)의 경우는 특히 고정된 카메라에서 움직이는 보행자들을 촬영하여 샘플들의 배경이 동일하여 오검출률의 증가폭이 더 커졌다.

V. 결론

Viola와 Jones의 객체 검출기는 처리 속도가 빠르기 때문에 많은 분야에서 실용적으로 활용될 수 있으나, 객체의 특징을 기술하기 위한 Haar-like feature는 조명의 영향을 많이 받으므로 밝기 변화의 영향을 줄이기 위해 정규화한 후에 사용해야 한다. 기존의 윈도우의 표준편차를 이용한 분산 정규화는 많은 연산을 필요로 하여 객체 검출기의 처리 속도에 적지 않은 영향을 주며 밝기 변화가 작은 영역에서 오검출이 늘어나는 문제가 있으므로, 이를 개선하기 위해 본 논문에서는 윈도우의 평균을 이용한 정규화 방법을 제안하였다.

실험 결과 제안한 방법을 사용하였을 때 적분 영상 생성 소요 시간이 38% 줄었으며, 정규화 값 계산과 Haar-like feature 계산에 소요되는 시간이 각각 5.7배와 9% 개선되어 객체 검출기의 전체 처리 속도가 26% 개선되었다. 기존의 정규화 방법과 제안한 정규화 방법을 사용하여 학습시킨 얼굴 정면, 차량 전면 및 후면, 야간 보행자 분류기의 성능을 비교한 결과 제안한 방법을 사용한 분류기들이 5%이상 높은 검출률을 보였으며, 조명 변화가 많은 MIT CBCL Face Recognition Database의 샘플들로 테스트 한 경우는 기존 방법에 비해 45% 높은 검출률을 보였다. 테스트

샘플들의 밝기를 여러 단계로 변경하여 성능을 비교한 경우에도 +1EV ~ -3EV의 구간에서 제안한 방법을 사용한 검출기가 우수한 성능을 보였다.

본 연구를 통하여 영상의 밝기 변화가 분류기의 성능에 영향을 미치며, Haar-like feature의 정규화 방법도 분류기의 성능에 많은 영향을 미치는 것이 확인되었다. 제안한 정규화 방법을 사용하여 학습시킨 분류기가 어두운 영상에 대해서는 견고한 성능을 보였지만 영상이 밝아지는 경우에는 급격한 성능 저하를 보였으므로 영상의 밝기에 적응적으로 정규화를 적용하여 분류기가 밝기 변화에 대해 좀 더 강건한 성능을 갖도록 하는 방안에 대한 추가 연구가 필요하다.

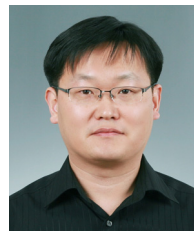
참 고 문 헌

- [1] C. Papageorgiou, M. Oren, and T. Poggio, "A General Framework for Object Detection," in *Proc. Sixth Int. Conf. Computer Vision*, Bombay, India, pp.555-562, Jan. 1998.
- [2] M. Betke and N. Makris, "Fast Object Recognition in Noisy Images using Simulated Annealing," in *Proc. Fifth Int. Conf. Computer Vision*, Boston, MA, USA, pp.523-530, Jun. 1995.
- [3] A. Yuille, P. Hallinan, and D. Cohen, "Feature Extraction from Faces using Deformable Templates," *Int. Journal of Computer Vision*, Vol.8, No.2, pp.99-111, Aug. 1992.
- [4] H. Rowley, S. Baluja, and T. Kanade, *Human Face Detection in Visual Scenes*, Carnegie Mellon Univ. Computer Science Technical Report CMU-CS-95-158R, 1995.
- [5] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian Detection Using Wavelet Templates," in *IEEE Conf. Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, pp.193-199, Jun. 1997.
- [6] E. Osuna, R. Freund, and F. Girosi, *Support Vector Machines: Training and Applications*, A. I. Memo 1602, MIT, 1997.
- [7] K. Sung and T. Poggio, "Example-based Learning for View-based Human Face Detection," *IEEE Transactions Pattern Analysis and Machine Intelligence*, Vol.20, No.1, Jan. 1998.
- [8] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *Int. Journal of Computer Vision*, Vol.38, No.1, pp.15-33, Jun. 2000.
- [9] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Kauai, HI, USA, pp.511-518, Dec. 2001.
- [10] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Vol.1, San Diego, CA, USA, pp.886-893, Jun. 2005.
- [11] D. Gavrila and S. Munder, "Multi-Cue Pedestrian Detection and Tracking from a Moving Vehicle," *Int. Journal of Computer Vision*, Vol.73, No.1, pp.41-59, Jun. 2007.
- [12] S. Bota and S. Nedesvchi, "Multi-feature Walking Pedestrians Detection for Driving Assistance Systems," *IET Intelligent Transport Systems*, Vol.2, Issue. 2, pp.92-104, Jun. 2008.
- [13] M. Enzweiler and D. Gavrila, "Monocular Pedestrian Detection: Survey and Experiments," *IEEE Transactions Pattern Analysis and Machine Intelligence*, Vol.31, No.12, pp.2179-2195, Dec. 2009.
- [14] S. Alvarez, M. Sotelo, I. Parra, D. Llorca, and M. Gavilan, "Vehicle and Pedestrian Detection in eSafety Applications," in *Proc. World Congress Engineering and Computer Science*, Vol.II, San Francisco, USA, pp.662-667, Oct. 2009.
- [15] W. Zheng and L. Liang, "Fast Car Detection Using Image Strip Features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Miami, Florida, USA, pp.2703-2710, Jun. 2009.
- [16] P. Viola and M. Jones, "Robust Real-time Face Detection," *Int. Journal of Computer Vision*, Vol.57, No.2, pp.137-154, May 2004.
- [17] T. Ephraim, T. Himmelman, and K. Siddiqi, "Real-Time Viola-Jones Face Detection in a Web Browser," in *Proc. Canadian Conf. Computer and Robot Vision*, Kelowna, British

- Columbia, Canada, pp.321-328, May 2009.
- [18] J. Ren, N. Kehtarnavaz, and L. Estevez, "Real-Time Optimization of Viola-Jones Face Detection for Mobile Platforms," in *Proc. IEEE Dallas Circuits and Systems Workshop SoC: Design, Applications, Integration, and Software*, Dallas, TX, USA, pp.1-4, Oct. 2008.
- [19] B. Kisacanin, "Integral Image Optimizations for Embedded Vision Applications," in *Proc. IEEE Southwest Symp. Image Analysis and Interpretation*, Santa Fe, New Mexico, USA, pp.181-184, Mar. 2008.
- [20] Y. Freund and R. Schapire, "A Short Introduction to Boosting," *Journal of Japanese Society for Artificial Intelligence*, pp.771-780, Sep. 1990.
- [21] Y. Freund and R. Schapire, "Experiments with a New Boosting Algorithm," in *Proc. Int. Conf. Machine Learning*, Bari, Italy, pp.148-156, Jul. 1996.
- [22] R. Schapire, Y. Freund, P. Bartlett, and W. Lee, "Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods," in *Proc. Int. Conf. Machine Learning*, Nashville, TN, USA, pp.322-330, Jul. 1997.
- [23] R. Lienhart and J. Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection," in *Proc. Int. Conf. Image Processing*, Rochester, New York, USA, pp.900-903, Sep. 2002.
- [24] S. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, and H. Shum, "Statistical Learning of Multi-View Face Detection," in *Proc. European Conf. Computer Vision*, Vol.4, Copenhagen, Denmark, pp.67-81, May 2006.
- [25] T. Mita, T. Kaneko, and O. Hori, "Joint Haar-like Features for Face Detection," in *Proc. Int. Conf. Computer Vision*, Beijing, China, pp.1619-1626, Oct. 2005.
- [26] F. Crow, "Summed-Area Tables for Texture Mapping," in *Proc. Conf. Computer Graphics and Interactive Techniques*, Vol.18, Minneapolis, MN, USA, pp.207-212, Jul. 1984.
- [27] B. Benson, J. Cho, D. Goshorn, and R. Kastner, "Field Programmable Gate Array (FPGA) Based Fish Detection Using Haar Classifiers," in *Proc. American Academy of Underwater Sciences Symp.*, Atlanta, GA, USA, pp.160-167, Mar. 2009.
- [28] S. Ehsan, A. Clark, and K. McDonald-Maier, "Novel Hardware Algorithms for Row-Parallel Integral Calculation," *Digital Image Computing: Techniques and Applications*, Melbourne, Australia, pp. 61-65, Dec. 2009.
- [29] CBCL Face Database #1, MIT Center For Biological and Computation Learning, <http://www.ai.mit.edu/projects/cbcl>
- [30] Caltech Faces 1999(Front), Computational Vision at Caltech, <http://www.vision.caltech.edu/html-files/archive.html>
- [31] B. Weyrauch, J. Huang, B. Heisele, and V. Blanz, "Component-based Face Recognition with 3D Morphable Models," in *Proc. Conf. Computer Vision and Pattern Recognition Workshop*, Washington D.C., USA, pp.85, Jun. 2004.
- [32] OpenCV, Open source computer vision library, <http://opencv.willowgarage.com/wiki/>
- [33] J. Friedman, T. Hastie, and R. Tibshirani, "Additive Logistic Regression: a Statistical View of Boosting," *Annals of Statistics* 28, pp.337-407, Aug. 1998.

박기영 (Ki-Yeong Park)

정회원



1991년 2월 한양대학교 물리학
과 학사

2006년 8월 서강대학교 정보통
신대학원 석사

2007년~현재 (주)엠아이웨어 부장

2008년~현재 서강대학교 전자
공학과 CAD & ES 연구실
박사 과정

<관심분야> 패턴 인식, 실시간 시스템, 병렬처리 시
스템

황 선 영 (Sun-Young Hwang)

중신회원



1976년 2월 서울대학교 전자공학과 학사

1978년 2월 한국 과학원 전기 및 전자공학과 공학석사

1986년 10월 미국 Stanford대학 전자공학 박사

1976년~1981년 삼성반도체(주) 연구원, 팀장

1986년~1989년 Stanford 대학 Center for Integrated System 연구소 책임연구원 및 Fairchild Semiconductor Palo Alto Research Center 기술 자문

1989년~1992년 삼성전자(주) 반도체 기술 자문

1989년 3월~현재 서강대학교 전자공학과 교수

<관심분야> SoC 설계 및 framework 구성, CAD시스템, Computer Architecture 및 DSP System Design 등