

머신비전을 이용한 도로상의 보행자 검출에 관한 연구

A Study on the Pedestrian Detection on the Road Using Machine Vision

이 병 룡*, Quoc Bao Truong, 김 형 석, 배 용 환
(Byung-Ryong Lee¹, Quoc Bao Truong¹, Hyoung-Seok Kim¹, and Yong-Hwan Bae²)

¹University of Ulsan

²Andong National University

Abstract: In this paper, we present a two-stage vision-based approach to detect multi views of pedestrian in road scene images. The first stage is HG (Hypothesis Generation), in which potential pedestrian are hypothesized. During the hypothesis generation step, we use a vertical, horizontal edge map, and different colors between road background and pedestrian's clothes to determine the leg position of pedestrian, then a novel symmetry peaks processing is performed to define how many pedestrians is covered in one potential candidate region. Finally, the real candidate region where pedestrian exists will be constructed. The second stage is HV (Hypothesis Verification). In this stage, all hypotheses are verified by Support Vector Machine for classification, which is robust for multi views of pedestrian detection and recognition problems.

Keywords: pedestrian detection, vision-based, HG (Hypothesis Generation), HV (Hypothesis Verification), different color method, symmetry peak processing, SVM (Support Vector Machine)

1. 서론

자동차가 달리는 도로에서 보행자는 많은 위험에 노출되어 있다. 유럽에서는 매년 15만 명 정도의 보행자가 자동차로 인하여 부상을 당하고 있으며, 6000명 이상이 목숨을 잃는다고 보고되고 있다. 미국의 경우 자동차 사고의 약 12%가 보행자 관련 사고이다[1]. 따라서 주행중인 자동차가 보행자를 실시간으로 정확하게 인식하는 것은 운전자보조시스템(driver assistance system)의 중요한 기능이 될 수 있다. 하지만, 자동적으로 보행자를 인식하는 것은 매우 어려운 작업이다. 그 이유는 보행자는 다양한 얼굴색과 다양한 외모를 갖고 있으며, 보행자의 모습이 정면, 측면 혹은 뒷모습일 수도 있으며, 입은 옷이나 날씨 혹은 시간에 따른 명암의 차이, 그리고 복잡한 배경으로부터 보행자를 정확하게 추출해 내어야 하기 때문이다. 현재 움직이는 차량으로부터 보행자를 인식하기 위해 주로 비전센서를 사용하는데, 비전센서는 가시광선 비전센서와 적외선 비전센서로 구분할 수 있다. 대부분의 경우 가시광선 비전센서를 사용하고 있지만, 특별한 경우 적외선 비전센서를 사용하는 연구도 보고되고 있다[2-4]. 머신비전(machine vision) 기반의 보행자 인식에서 사용되는 CCD 카메라의 숫자에 의해 모노비전(monocular vision)과 스테레오 비전(stereo vision)으로 구분할 수 있다. 스테레오 비전의 경우 모노비전보다 CCD 카메라를 1개 더 사용하지만 영상정보로부터 깊이정보(depth information)를 얻을 수 있는 장점이 있다[5-7].

참고문헌[8-9]에서 저자들은 획득된 비전 영상에서 보행자

를 검출하기 위해 다양한 스케일을 이용해서 이미지를 스캔하는 방법을 적용하였는데, 이러한 방법들은 이미지를 처리하는데 시간이 많이 걸리는 단점이 있으므로 운전자 보조시스템이나 무인주행을 위한 시스템에는 적합하지 않다고 볼 수 있다. 참고문헌[10-13]의 저자들은 주행차량의 차선에 있는 보행자를 검출하거나 이미지 상의 제한된 범위 내에서 보행자를 검출하였다. 이 방법 또한 이미지 내의 모든 보행자를 검출할 수 없다는 점에서 제한적이라고 볼 수 있다. 보행자를 검출하기 위한 매우 우수한 방법중의 하나는 그림 1과 같이 보행자가 있을 것으로 판단되는 후보지역생성(HG: Hypothesis Generation) 단계와 HG지역 중에서 보행자가 확실히 있는 지역 즉 후보지역확정(HV: Hypothesis Verification)이라는 2단계를 거치는 방법이다.

후보지역생성을 위한 접근방법은 크게 3가지로 분류되는데 배경분리(background subtraction) 방법, 스테레오 기반(stereo-based) 방법, 그리고 형상정보 획득(shape information) 방법이다[1]. 이 중에서 배경분리방법이 목표물을 분리하는데 있어서 가장 신뢰할 만한 방법이라고 할 수 있다.

참고문헌[14]에서 저자들은 차량 위에서의 카메라 움직임을 평가하기 위해 능동적 배경분리(active background subtraction) 방법을 제안하였다. 차량의 속도로부터 위치벡터가 계산되고, 이 정보가 각 이미지 프레임에 변환 적용된다. 이렇게 해서



그림 1. 보행자를 검출하기 위한 2단계 처리: 보행자 추정 단계와 확정단계.

Fig. 1. Two-step pedestrian detection strategy: HG and HV.

* 책임저자(Corresponding Author)

논문접수: 2010. 12. 6., 수정: 2011. 3. 10., 채택확정: 2011. 3. 17.

이병룡, 김형석: 울산대학교 기계공학부

(brlee@ulsan.ac.kr/khseog@mail.ulsan.ac.kr)

Quoc Bao Truong: 울산대학교 기계공학부(bao_truongquoc@yahoo.com)

배용환: 안동대학교 기계교육과(yhbae@andong.ac.kr)

보행자 영역을 이미지의 배경으로부터 분리하였다. 스테레오 기반 방법은 이미지 상에서 보행자의 위치를 쉽게 획득할 수 있다는 이점이 있다[5,6]. 형상정보획득 방법은 시간과 연계된 정보(temporal information)가 필요가 없다. 즉, 시간에 따라 변하는 배경과 결부된 문제점을 극복할 수 있다. 하지만 이 방법은 보행자의 다양한 형태를 고려해야 하므로 계산시간이 많이 든다는 단점이 있다[15].

후보지역확정 단계에서도 다양한 방법들이 있는데, 대부분이 학습(learning)에 기반을 둔 알고리즘을 사용하고 있다. 즉, 학습을 위한 다양한 학습영상(training image)으로부터 목표물의 특징 클래스를 수집하게 되며, 검출성능을 향상하기 위해 다양한 비 목표물(non-object)들의 특징 클래스도 수집된다. 먼저, 각 학습용 이미지는 일련의 로컬 및 글로벌 특징 (예를 들면 Haar wavelet, SIFT, Shape Context, PCA, Gabor 등)에 의해 구현되고, 그 다음 물체와 비 물체간의 선택경계(decision boundary)를 구한다. 여기에 사용되는 분류기 학습 도구들로는 Adaboost, Support Vector Machine, 유전자 알고리즘 등이 있다[6,9,16-22]. 이러한 도구들의 성능은 사용하는 특징이나 선택함수(decision function)등에서 차이를 보이지만, 궁극적으로 목표물 클래스를 구성하는 부분의 기하학적인 특성에 영향을 받는다고 할 수 있다.

본 연구에서는 후보지역생성(HG: Hypothesis Generation) 단계와 후보지역확정(HV: Hypothesis Verification) 단계를 거쳐서 도로 위의 보행자를 검출할 수 있는 알고리즘을 제안하였다. 후보지역 생성을 효과적으로 처리하기 위해 보행자 부근에 생성되는 에지 정보의 기하학적인 특성을 이용하였다. 그리고, 후보지역 내에 보행자가 몇 명이 포함되어 있는지를 판단할 수 있도록 대칭 히스토그램 방법(symmetry histogram method)을 처음으로 제안하였다. 후보지역 확정단계에서는 수정된 GA (Genetic Algorithm)에 기반한 SVM 분류기를 사용하여 보행자 영역을 확정하도록 하였다. 제안된 알고리즘은 다양한 배경과 다양한 모습, 다양한 위치에 서 있는 보행자들을 성공적으로 검출할 수 있었다. II 장에서는 수평 에지 보완(horizontal-edge repair), 보행자와 도로의 색상 차이, 대칭 히스토그램 방법(symmetry histogram method)등을 이용하여 후보지역을 생성하는 단계를 설명하였고, III 장에서는 Haar-Like 특징, 의사결정 가지(decision tree), 수정된 GA 알고리즘 및 SVM 분류기를 이용한 후보지역확정 단계를 설명하였다. IV 장에서는 제안된 알고리즘의 성능을 보이기 위한 다양한 영상을 이용한 실험을 수행하였다. 끝으로, V 장에서는 결론 및 본 논문의 기여도를 정리하였다.

II. 보행자 추정지역의 검출

영상에서 보행자가 있을 가능성이 있는 영역을 추정하기 위해서, 보행자의 외관에 대한 사전 정보들이 사용될 수 있다. 영상에서 보행자의 경계선을 분석해 보면 매우 강한 수직 에지(vertical edge)를 갖게 되고 좌우의 경계선은 대칭을 이루는 구조를 갖추고 있다. 또한 보행자와 도로의 경계선은 다수의 특징적인 수평 에지(horizontal edge)들로 구성되어 있다. 이러한 특징을 바탕으로 보행자의 추정지역을 탐색하는 알고리즘을 제안하고자 한다.

1. 원 영상으로부터의 에지 추출



그림 2. 원 영상과 수직 및 수평방향 Sobel 에지 검출 후의 영상.

Fig. 2. The original image and the images after Sobel edge detection in vertical and horizontal direction (from left to right).



(a) Before

(b) After

그림 3. 작은 수직 및 수평 에지의 제거 전과 제거 후 영상.

Fig. 3. Images before and after rejecting small vertical and horizontal edges.

본 연구에서는 Sobel 에지 검출법을 수평방향과 수직방향으로 적용하여 원 영상의 에지를 추출하였다. 그림 2는 Sobel 에지 검출법을 사용하여 에지를 검출한 예를 보여주고 있다.

그림 2에서 중요하지 않은 수평 에지와 수직 에지들을 제거하기 위하여 비교적 큰 에지들만을 남겨두고 작은 에지들은 제거하는 과정을 거친다. 그림 3은 작은 에지들을 제거한 후 원 영상에 에지들을 중첩한 영상을 보여주고 있다.

그림 3의 보행자 영역에서 수직 및 수평 에지 정보들을 분석해 보면 수직 에지가 수평 에지보다 길다는 것을 알 수 있다. 따라서 식 (1)과 같은 에지 비를 도입한다.

$$\frac{\text{수평에지의 길이}}{\text{수직에지의 길이}} = \frac{1}{3} \tag{1}$$

그리고 본 연구에서 사용된 640 x 480 픽셀의 영상에서, 보행자의 영역을 추정하기 위해 수직 및 수평 에지에 대한 임계값(threshold)을 적용하였다. 이 값은 여러 번의 실험을 통하여 구한 값으로 수직 에지는 15 픽셀에서 45 픽셀 사이, 수평 에지는 5 픽셀에서 15 픽셀 사이의 값으로 정하였다. 이 범위를 벗어나는 에지 정보들은 보행자를 위한 후보영역



그림 4. 수평 에지 보정 후의 영상.

Fig. 4. Image result after repairing horizontal edges.

이 아니라고 가정하고 제거된다. 보행자의 발 영역에는 다수의 수평 에지들이 존재하므로 정확한 발 위치를 찾는 것이 어렵다. 이러한 문제점을 해결하기 위하여 Truong과 Lee [23]가 제안한 수평 에지 보정방법을 적용하였다. 이 방법은 해당 영역의 작은 여러 개의 수평 에지들을 긴 하나의 수평 에지로 변환하게 해 준다. 그림 4는 이 방법을 적용한 후의 에지영상을 보여주고 있다. 그림 4의 예에서 영상 내의 수평 에지의 수가 491개에서 112개로 감소하였다.

2. 배경영역에 위치한 수평 에지의 제거

수평 에지들을 보정한 후에도 보행자의 영역 외부, 즉 도로나 주변의 바닥 등에 수평 에지들이 많이 존재하게 된다. 일반적으로 보행자 영역 내부의 옷의 색상과 보행자 주변의 바탕색은 색상적으로 차이가 있으므로 ‘색상차이법’을 이용하여 쓸모없는 수평 에지들을 제거하고 보행자 영역으로 추정되는 경우의 수평 에지들만 남기게 한다. 먼저, 영상의 밑부분에 위치하는 도로영역의 평균 색상값을 구한다. 그런 다음 보행자의 발 부근의 수평 에지들을 검출하기 위하여 다음과 같은 규칙을 적용한다.

- (1) 각 수평 에지의 위와 아래 영역에서 평균 색상값을 계산한다.
- (2) HSI색상값이 색상차이를 구별하는 데에는 효과적이므로 계산된 RGB 색상값을 HIS 색상값으로 변환한다.
- (3) 에지 위와 아래 영역의 색상평균값이 같으면 도로나 땅과 같은 배경 영역에 있는 에지 이므로 삭제하고, 색상평균값이 차이가 나는 에지들은 남겨둔다.

위 규칙을 적용한 영상의 예를 그림 5에 나타내었다.

3. 보행자 탐색을 위한 ROI(Region of Interest)영역 정의

보행자의 영역은 가로와 세로의 비에 의해 규정할 수 있다. 많은 보행자 영상들을 분석한 결과 평균적인 세로-가로 비는 약 0.33의 값을 가진다고 밝혀져 있다[11]. 보행자의 발 위치에서 영상의 위 끝까지의 거리를 y 라고 놓고 보행자의 발 위치에서 수평 길이 $w=0.33y$ 를 구한다. 발 부근의 특징을 알아보기 위한 관심영역(ROI: Region-of-Interest)의 폭은 w 로 하고 높이는 $h=0.15y$ 로 정의한다. 이 ROI를 이용하여 대칭 히스토그램(symmetry histogram)을 수행하고, ROI 내에 있는 보행자 정보를 추출하게 된다. ROI 영역을 발부분과 다리의 하부로 정하는 이유는 몸체의 상부에 비해 배경이 균질하고 단순한 경우가 많기 때문이다. 보통 ROI 내부에 1명의 보행



그림 5. (a) 색상차이법 적용전 에지 (b) 적용후의 에지.
Fig. 5. Edge distribution before and after applying different color method: before (a) and after (b).

자가 검색되지만, 2명의 보행자가 매우 밀착되어 있는 경우는 한 ROI 내에 2명의 보행자가 검출될 수도 있다.

4. 대칭 히스토그램법(symmetry histogram method)을 이용한 보행자 위치 검출

4.1 ROI 영역에서 대칭 히스토그램 값 계산

대칭 히스토그램을 생성하기 위해 먼저 Sobel 에지 검출법을 수직방향으로 적용하여 에지 정보를 구한다. 그림 2의 경우 가운데 영상이 이에 해당된다. 그런 다음 ROI 영역 내의 임의의 위치 x 에서 만든 수직선을 따라 대칭되는 픽셀의 개수를 계산하여 히스토그램으로 나타낸 것을 대칭히스토그램이라고 정의한다. ROI 내의 x 방향으로 k 번째 수직선에 대한 대칭 히스토그램 값을 구하는 식은 다음과 같다.

$$If \begin{cases} |x - x_1| = |x - x_2| \\ y_1 = y_2 \\ I(x_1, y_1) = 255 \\ I(x_2, y_2) = 255 \\ d(I(x_1, y_1), (x_2, y_2)) \leq 0.2w \\ 0 \leq y_1, y_2 \leq h \end{cases} \quad then \quad S_k = S_k + 1 \quad (2)$$

위 식에서 정의된 변수는 다음과 같이 정의된다.

S_k : k 번째 수직선에 대한 대칭 히스토그램 값

w, h : ROI 영역의 폭과 높이

$d(I(x_1, y_1), (x_2, y_2))$: 두 에지 픽셀 $I(x_1, y_1)$ 와 $I(x_2, y_2)$ 사이의 거리

대부분의 연구자들은 영상내의 물체에 대한 대칭성을 구하기 위해 명암 히스토그램(intensity histogram)을 많이 사용하게 되는데 이러한 접근 방법은 2가지 면에서 문제를 가지고 있다. 첫째, 영상 픽셀들의 명암값은 환경의 변화에 매우 민감하다. 조명의 변화에 의해 명암값은 쉽게 변하게 되며, 명암에 의존하는 대칭성 파악에 영향을 줄 수 있다. 특히, 물체에 그림자가 부분적으로 존재하게 되면 대칭성 판별에 크다란 오차를 주게 된다. 둘째, 영상에 의존하여 보행자를 탐색하는 경우 도로는 주요한 배경이 된다. 명암에 의한 대칭성 판별을 할 경우 도로는 매우 높은 대칭성 값을 나타내므로 보행자 라는 물체의 대칭성을 구하는데 방해가 된다. 하지만 본 연구에서 제안한 대칭 히스토그램 방법은 위에 언급한 문제점들을 극복할 수 있고, 조명이나 날씨의 변화에 매우 강인한 특성을 가지고 있다. 대칭 히스토그램을 작성한 후 최



그림 6. (a) 선정된 ROI와 (b) 대칭 히스토그램 계산 결과.
Fig. 6. Regions of interest (ROI) used for symmetry searching (a) and result of symmetry histogram (b).

댓값을 갖는 위치는 다음과 같은 방법으로 구할 수 있다.

- 1) HCA (Hill Climbing Algorithm)를 이용하여 대칭 히스토그램의 지역 최댓값(local maximum)를 구한다.
- 2) 지역최댓값 중에서 가장 큰 값(MaxVal)을 구한다.
- 3) $0.2 * MaxVal$ 값보다 적은 지역최솟값은 무시한다.

그림 6은 주어진 ROI 들에 대한 영상 히스토그램을 적용한 그림이다. 6(a)는 보행자의 발 영역에 2개의 ROI가 생성되어 있음을 보여주고 있고, 6(b)는 해당 ROI에 대칭 히스토그램 방법을 적용하여 대칭 히스토그램과 히스토그램의 지역 최댓값을 표시하고 있다.

4.2 ROI 내에 위치하는 보행자의 수 계산

위 3절에서 언급한 바와 같이 보통 ROI 내부에 1명의 보행자가 검색되지만, 2명의 보행자가 매우 밀착되어 있는 경우는 한 ROI 내에 2명의 보행자가 검출될 수도 있다. 본 연구에서는 1개의 ROI 내의 보행자의 수를 판단하는 알고리즘을 다음과 같이 제안하였다.

- 1) 대칭 히스토그램으로 구한 지역최댓값을 이용하여 다음과

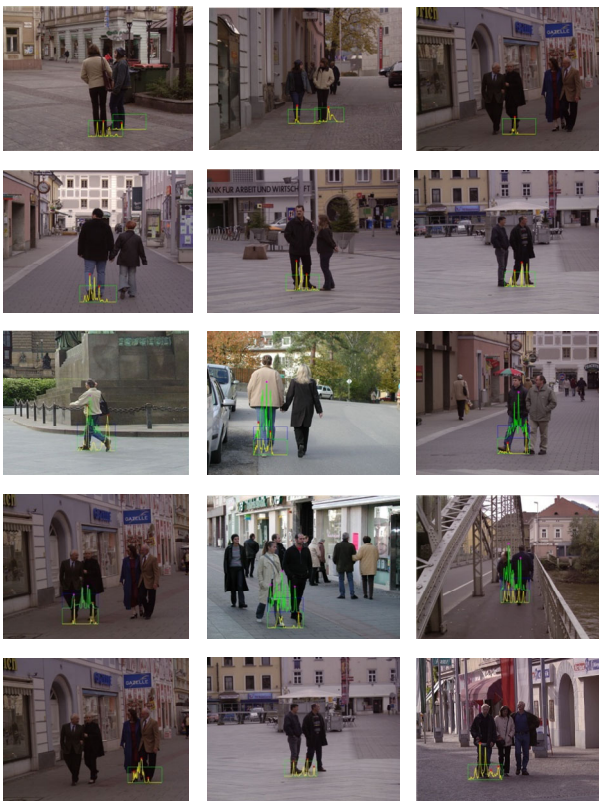


그림 7. 대칭 히스토그램 알고리즘 적용한 후의 결과영상. 1번 행: One peak으로 1명, 2번 행: two near peaks으로 1명, 3번 행: two far peaks으로 1명, 4번 행: two far peaks으로 2명, 5번 행: Three peaks으로 2명.

Fig. 7. Result images after applying symmetry histogram algorithm. First row: One peak and one person, Second row: two near peaks and one person, Third row: two far peaks and one person, Fourth row: two far peaks and two persons, Fifth row: Three peaks and two persons.

같이 결정한다.

- 1.1) 지역최댓값이 1개 존재하면 보행자는 1명이다.
- 1.2) 지역최댓값이 3개 존재하면 보행자는 2명이다.
- 1.3) 지역최댓값이 2개 존재하면 보행자는 1명 혹은 2명이다. 이 경우 다음 단계 2로 진행한다.

- 2) 두 지역최댓값 사이의 거리를 d 라고 할 때

2.1) 만약 $d \leq \frac{1}{3}w$ 이면 보행자는 한 명이다.

2.2) 그렇지 않으면 단계 3으로 진행한다.

- 3) ROI내에 보행자가 1명 혹은 두 명인 경우이며 다음과 같은 과정을 수행한다.

3.1) 현재 ROI 위치 위에 크기가 같은 부가적인 ROI를 생성한다

3.2) 새로 생성된 ROI 내의 대칭 히스토그램을 적용하여 지역최댓값들을 구한다.

3.3) 지역최댓값이 1개이면 보행자는 1명이다. 그렇지 않으면 보행자는 2명이다.

그림 7은 위 알고리즘을 적용한 결과를 보여주고 있다.

4.3 보행자 추정 영역의 생성

대칭 히스토그램 알고리즘을 이용하여 주어진 ROI 내에 몇 명의 보행자가 포함되는지를 알 수 있다. 이 절에서는 보행자로 추정되는 지역의 경계선을 정하는 알고리즘을 다음과 같이 제안하였다.

- 1) 보행자 후보지역에 대한 좌/우 경계선을 결정한다.

1.1) 대칭 히스토그램 적용 후 지역최댓값이 1개 이면, 즉 ROI 내에 보행자가 1명인 경우 보행자에 대한 대칭축은 지역최댓값이 있는 위치(Pos)로 정한다. 그런 다음 보행자의 좌/우 경계선은 식 (3)과 같이 정한다.

$$\begin{aligned} \text{LeftBorder} &= \text{Pos} - 0.35 * w \\ \text{RightBorder} &= \text{Pos} + 0.35 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (3)$$

1.2) ROI내에 지역최댓값이 2개이고 두 값 사이 거리가 $d \leq \frac{1}{3}w$ 인 경우 (즉, 보행자가 1명인 경우):

1.2.1) 2개의 지역최댓값이 크기 차이가 많은 경우 식 (3)을 이용한다. 이 때 Pos값은 지역최댓값이 큰 위치로 한다.

1.2.2) 2개의 지역최댓값이 크기가 비슷하고 두 위치 사이의 거리가 $0.2w$ 이내인 경우 Pos는 식 (4)와 같이 계산한다.

$$\text{Pos} = \frac{P_{\text{Left}} + P_{\text{Right}}}{2} \quad (4)$$

여기서 P_{Left} 와 P_{Right} 은 좌/우 지역최댓값의 위치를 말한다.

식 (4)를 이용해서 Pos를 구한 뒤 식 (3)을 이용하여 좌/우 경계선을 구한다.

1.3) ROI 내에 지역최댓값이 2개이고 두 값 사이 거리가 $d > \frac{1}{3}w$ 가 인 경우:

1.3.1) 보행자가 1명인 경우,

P_{Left} 와 P_{Right} 의 거리가 $0.5w$ 보다 적은 경우, 식 (5)를 이용해서 좌/우 경계선을 구한다.

$$\begin{aligned} \text{LeftBorder} &= P_{\text{Left}} - 0.25 * w \\ \text{RightBorder} &= P_{\text{Right}} + 0.25 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (5)$$

그 외 경우는 식 (6)을 이용해서 좌/우 경계선을 구한다.

$$\begin{aligned} \text{LeftBorder} &= P_{\text{Left}} - 0.1 * w \\ \text{RightBorder} &= P_{\text{Right}} + 0.1 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (6)$$

1.3.2) 보행자가 2명인 경우,

1.3.2.1) 왼쪽 영역,

$$\begin{aligned} \text{LeftBorder} &= \text{Pos} - 0.4 * w \\ \text{RightBorder} &= \text{Pos}, \end{aligned} \quad w = \text{ROI의 폭} \quad (7)$$

1.3.2.2) 오른쪽 영역,

$$\begin{aligned} \text{LeftBorder} &= \text{Pos} \\ \text{RightBorder} &= \text{Pos} + 0.4 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (8)$$

1.4) ROI내에 지역최댓값이 3개인 경우(보행자가 2명):

1.4.1) 왼쪽 영역,

$$\begin{aligned} \text{LeftBorder} &= P_1 - 0.4 * w \\ \text{RightBorder} &= P_2 + 0.1 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (9)$$

1.4.2) 오른쪽 영역,

$$\begin{aligned} \text{LeftBorder} &= P_2 - 0.1 * w \\ \text{RightBorder} &= P_3 + 0.4 * w, \end{aligned} \quad w = \text{ROI의 폭} \quad (10)$$

여기서 P_1, P_2 그리고 P_3 는 3개의 지역최댓값 위치를 말한다.

2) 보행자 후보지역에 대한 상부 경계선을 결정한다.

보행자 후보지역에 대한 상부 경계선은 보행자의 발이 위치하는 하부위치와 위에서 구한 좌/우 경계선을 이용하여 구한다. 보행자 후보지역에 대한 폭은 식 (11)과 같이 구할 수 있다.

$$\text{Width} = \text{RightBorder} - \text{LeftBorder} \quad (11)$$

그리고 상부 경계선은 식 (12)로 구할 수 있다.

$$\text{TopBorder} = \text{BottomBorder} - (\text{Width} * \text{Ratio}) \quad (12)$$

여기서 Ratio 는 1.3절에서 정의된 보행자 영역의 폭-높이 비를 말한다. 이 비는 $\text{Ratio} = [0.28, 0.5]$ 사이의 값으로 설정되며, BottomBorder 의 위치에 따라 변하는 값이다. 위 보행자 추정 영역 생성 알고리즘을 적용한 영상 결과들을 그림 8에 나타내었다

III. 보행자 영역 확정 단계

2종류의 패턴을 분류하는 방법(two-class pattern classification problem)에서 목표영역을 확정하는 단계는 필수적이고 매우 중요한 요소이다. 즉, 보행자 후보영역 중에서 최종적으로 보행자 영역과 보행자가 아닌 영역으로 구분을 확정하는 것이다. 현재 개발되어 있는 대부분의 패턴분류 시스템들은 감독 학습(supervised learning)을 사용하고 있다. 이 방법은 먼저, 몇 개의 패턴 특징을 발췌하고, 다음에 이 패턴의 차이점을 학습시켜 다른 부류(class)와 구분하게 한다. 하지만 이러한 방법에서 분류성능은 패턴의 특징과 패턴을 분류하는 분류기(classifier)에 매우 민감한 편이다. 특히 SVM (Support Vector Machine)분류기의 분류 성능을 향상하기 위해서는 다음의 2가지 문제가 해결되어야 한다.

- (1) 분류를 효과적으로 수행하기 위한 최적의 특징패턴 수 (feature set)를 어떻게 정하는가 하는 문제이다. 일반적으로 특징패턴수가 많아질수록 분류속도는 매우 둔화되기 때문이다.
- (2) 분류 성공률을 높이기 위해 SVM분류기의 변수들을 어떻게 설정하는가 하는 문제이다.

현재 대부분의 연구에서 수행된 패턴분류에서는 실험적인 방법을 통해 분류기 파라미터를 설정하고 있기 때문에 최적의 분류성능을 기대하기 어렵다. 일반적으로 특징패턴(feature set)이 달라지면 분류기 파라미터의 최적값도 변화하기 때문이다.

본 연구에서는 Haar-like 특징을 이용하여 보행자 이미지의 특징을 추출하고, 의사결정 가지 (decision tree) 학습을 사용하여 이미 형성된 특징 수를 먼저 적절한 수준까지 줄이고, 그 후에 수정된 유전자알고리즘 (modified genetic algorithm) [22]을 이용하여 SVM 학습 파라미터를 최적화하여 보행자 추출을



그림 8. 보행자 추정 영역 생성 알고리즘을 적용한 영상결과. 1번 행: One peak으로 1명, 2번 행: two near peaks으로 1명, 3번 행: two far peaks으로 1명, 4번 행: two far peaks으로 2명, 5번 행: Three peaks으로 2명.

Fig. 8. Result images after applying the algorithm of pedestrian candidate regions. First row: One peak and one person, Second row: two near peaks and one person, Third row: two far peaks and one person, Fourth row: two far peaks and two persons, Fifth row: Three peaks and two persons.

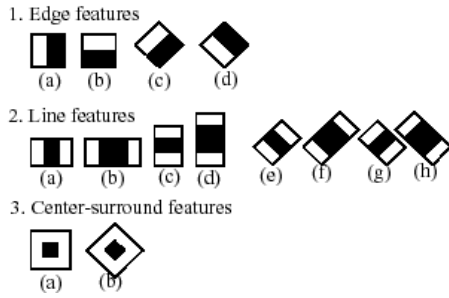


그림 9. Haar-like 특징추출을 위한 단위특징.

Fig. 9. Basic features for Haar-Like feature extraction.

위한 특징 수를 더욱 줄임과 동시에 분류성능을 높이는 방법을 제안하였다.

1. Haar-Like 특징

참고문헌[24,25]에 언급되어 있는 바와 같이 각 Haar-like 특징은 탐색창에 대한 상대좌표 및 특징의 크기 등에 의해 표현할 수 있으며, Haar-like 특징을 구성하는 특징 요소들은 그림 9와 같다. 각각의 특징은 흑색과 백색의 사각형의 결합으로 구성되어 있으며, 45도 정도 기울어진 확장요소들도 포함할 수 있다. Haar-like 특징 값은 두 성분의 가중치 값의 합으로 계산된다.

일반적으로 분류기에 입력되는 특징들의 개수가 굉장히 많기 때문에 계산시간이 많이 소요되며 실시간(real-time) 적용에 제약이 따르게 된다. Viola 등[24]은 픽셀의 합산 계산시간을 효과적으로 줄이기 위해 SAT (Summed Area Table) 방법을 제안하였다. SAT는 식 (13)과 같이 정의된다.

$$SAT(x, y) = \sum_{i < x, j < y} I(i, j) \tag{13}$$

사각형영역 $r = \{(x, y), x_0 \leq x < x_0 + w, y_0 \leq y < y_0 + h\}$ 에서의 픽셀의 합은 SAT를 이용하여 식 (14)와 같이 계산할 수 있다. 사각형 영역이 일정한 각도로 기울어져 있는 경우는 Lienhart등[25]이 제안한 방법으로 계산할 수 있다.

$$S(r) = SAT(x_0 + w, y_0 + h) - SAT(x_0 + w, y_0) - SAT(x_0, y_0 + h) + SAT(x_0, y_0) \tag{14}$$

2. 의사결정 가지(decision tree) 학습

의사결정 가지는 일종의 이진 가지(binary tree)로서 각 노드(node)는 2개의 자녀 노드를 가진다. 의사결정 가지는 분류나 혹은 회귀(regression)에도 사용될 수 있다. 가지는 시작마디(root node)에서 시작하여 재귀적으로 생성이 된다. 모든 학습 데이터(특징 벡터와 응답)는 각 마디를 분화하는데 사용된다. 각 마디에서 최소자승법 등의 방식으로 최적결정규칙(optimal decision rule)이 정해진다. 가지들이 생성될 때, 상호검증과정(cross-validation)을 이용해서 모델의 over-fitting을 초래하는 일부 가지들을 제거할 수도 있다. 본 연구에서는 의사결정 가지를 학습하기 위하여 Open CV에서 제공되는 툴을 사용하였다[26]. 본 연구에서 다루는 보행자 탐색시스템은 특징요소의 개수가 너무 많으므로 바로 적용하기가 어렵다. 따라서 학습 알고리즘을 이용하여 각 특성의 분류능력을 평가하여 단순 특징집합(coarse feature set)을 구하고자 한다. 대부분의 기존

연구에서는 이러한 목적을 위해 Adaboost 알고리즘을 사용하고 있지만, 본 논문에서는 의사결정 가지(decision tree) 학습을 이용하였다. 그 이유는 의사결정 가지 학습이 Adaboost 알고리즘보다 간단하여 학습시간이 빠르며, 또한 메모리 사용도 적어서 대형의 데이터 요소를 학습하는데 적합하기 때문이다.

3. SVM (Support Vector Machine) 학습

SVM [27]은 two-class 분류기로 사용된다. 선형 또는 비선형 영역을 학습하는 방법이다. Two-class 의 한 집단에 속하는 점들에 대해 SVM은 two-class 사이의 경계선의 거리를 최대화 할 수 있는 초 평면(hyper-plane)을 찾아낸다. 두 개의 클래스에서 다음과 같이 l 개의 점을 고려하면

$$(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l), x_i \in R^N, y_i \in \{-1, +1\} \tag{15}$$

두 개의 클래스를 구분할 초 평면 식은 다음과 같다.

$$f(x) = \sum_{i=1}^l y_i a_i k(x, x_i) + b \tag{16}$$

여기서 $k(x, x_i)$ 는 커널 함수(kerner function)이고, $f(x)$ 는 x 의 소속 값을 나타낸다. 최적의 초 평면을 구하는 문제는 식 (16)에서 0이 아닌 모든 상수 a_i 를 찾는 것이다. a_i 에 대응하는 데이터 포인트 x_i 가 최적 초 평면의 서포트 벡터(support vector)가 된다. 본 논문에서는 다음과 같은 Gaussian 커널 함수를 사용하였다.

$$K(x_i, x_j) = C \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \tag{17}$$

식 (17)을 SVM 분류기에 적용하는 경우 커널 함수 파라미터인 γ 와 C 의 값의 선정이 분류기 성능에 많은 영향을 미치게 되지만, γ 와 C 의 값은 미리 알기 어렵다. 따라서 본 논문에서는 수정된 유전자 알고리즘(GA: Genetic Algorithm)을 적용하여 SVM 학습 파라미터를 최적화하였다.

4. 수정된 유전자 알고리즘(genetic algorithm)

유전자 알고리즘은 자연도태의 메커니즘의 원리에 바탕을 두고 있으며 주어진 함수에 대한 최적화 방법으로 많이 활용하고 있다. 유전자 알고리즘은 선택(selection), 교배(crossover) 및 변이(mutation) 과정을 겪는다.

i) Coding and encoding individual(각 개체의 형성)

본 연구의 경우, 각 개체는 특징부분집합(feature subset)과 각 개체내의 파라미터로 형성되어 있다.

$$01100 \dots \underbrace{110110}_{P_1} \underbrace{0110001}_{P_2} \underbrace{110101101}_{P_3}$$

각 개체는 3개의 부분요소로 구성되어 있다. 왼쪽에 위치한 P_1 벡터는 298비트로 Haar-like 특징부분집합을 나타내며 의사결정 가지 학습과정에 사용된다. P_1 벡터내의 숫자 1은 대응되는 위치에서의 특징이 선택되었음을 의미한다. P_2 는 9 비트로서 γ 를 나타내며, P_3 는 12 비트로서 SVM 학습모델의 커널 식에 있는 C 를 나타내는데 사용된다. γ 와 C 는 다음 식으로 표현된다: $\gamma = 0.01 * P_2, C = 0.01 * P_3$. 여기서 P_2 와 P_3 는 두 2진 코드에 상응하는 숫자이다. 따라서, γ 와 C 의 범위는 $\gamma = [0.01, 5.12]$ 과 $C = [0.01, 40.96]$ 로 제한됨을 알 수 있다.

ii) 적합도 평가(Fitness evaluation)

특징 부분집합의 추출, SVM 학습모델에 대한 최적 파라미터를 찾는 목적은 좀더 적은 특징을 이용하여 분류성능을 향상시키는 것이다. 본 연구에서는 [22]에서 사용한 적합도함수(fitness function)를 사용하였다. 분류성능을 위해 3종류의 평가항목을 사용하였다. 오차율 $Err(f(x))$ 는 정상적으로 분류되지 않을 확률로 정의된 값이며, 정밀도 $Rec(f(x))$ 는 $f(x)=1$ 을 갖는 샘플이 올바르게 분류되는 확률로 정의된다. 여기서, $f(x)$ 는 식 (12)에서 정의되는 의사결정함수(decision function)이다. SVM 분류기의 평가함수는 다음과 같다.

$$G(f(x)) = (1 - \lambda) * (1 - Err(f(x))) + \lambda * Rec(f(x)) \quad (18)$$

여기서 $\lambda \in [0.1, 0.5]$ 범위로 $Err(f(x))$ 와 $Rec(f(x))$ 사이의 trade-off으로 결정이 되는데, 본 논문의 경우 $\lambda = 0.3$ 의 값을 사용하였다.

iii) GA 연산자(GA operators)

선택연산자는 룰렛 휠(Roulette wheel) 방법을 사용하 $Rec(f(x))$ 였고, 교배와 변이를 위해 일반적으로 사용하는 1점 교배/변이 방법(one point cross-over and mutation operators)을 사용하였다.

iv) GA 프로세스(GA processing)

본 연구의 경우 학습을 위한 데이터 셋이 매우 많으므로 좋은 분류성능을 얻기가 어렵고, 학습시간도 많이 소요되기 때문에 일반적으로 사용하는 GA이 적합하지 않다. 따라서, 본 논문에서는 수정된 GA 알고리즘을 다음과 같이 제안하였다.

- 1) 2진 코딩방법으로 M개의 염색체를 가지는 개체들을 난수적으로 생성한다.
- 2) 각 개체들에 대한 적합도(fitness)를 계산한다.
- 3) GA 연산자를 이용하여 다음 세대의 개체들을 아래와 같이 생성한다.
 - 3.1) 이전 세대의 개체들로부터 N개의 염색체를 추출하도록 선택연산자(selection operator)를 적용한다.
 - 3.2) 추출된 염색체의 처음 N/4 개를 현재세대의 개체에 복제한다.
 - 3.3) 현재세대의 개체의 마지막 N/4 개를 제거한다.
 - 3.4) 현재세대의 개체에서 난수적으로 N/4 개의 염색체를 새로 생성한다.
 - 3.5) 변이 확률(mutation probability)에 의거하여 변이연산자를 적용한다.
 - 3.6) 현재세대의 각 개체들을 적합도 값의 순서로 나열하고 새 개체들을 다음세대로 편입시킨다.
- 4) 세대값을 $Gen=Gen+1$ 로 증가시킨다. 적합도 값이 포화되면 종료하고, 아니면 2번으로 이동한다.

위 수정된 GA 알고리즘에서 단계 3.2와 3.3은 다음 세대의 적합성을 향상시킨다. 그리고 단계 3.4는 다음세대의 다양성을 높이는 동시에 지역최솟값에 빠지는 것을 방지하는데 도움을 준다. 실험적인 결과에 의하면 본 연구에서 제안한 수정된 GA 알고리즘이 기존방식의 GA보다 우수한 학습결과를 주는 것을 확인하였다.

IV. 실험결과

1. 데이터 세트의 구성

본 연구에 사용한 학습 데이터로 6000개의 보행자 이미지와 6000개의 비보행자(non-pedestrian) 이미지를 사용하였다. 이 이미지들의 일부 부분은 NICTA 보행자 데이터 셋[28]에서 수집하였고, 나머지는 별도로 수집한 자료들이다. Open CV 함수 툴을 사용하여 모든 이미지를 16*40 픽셀의 JPG 포맷으로 변환하였다.

1.1 분류기 시스템의 성능비교

본 연구에서 제안한 수정된 GA-SVM 분류시스템과 기존의 GA-SVM 분류시스템 간의 분류성능을 표 1에 비교하였다.

표 1에서 알 수 있듯이 수정된 GA-SVM 알고리즘이 보다 적은 145개의 특징을 이용하면서도 높은 검출율(98.5%)을 보여주고 있다. 또한 수정된 GA-SVM 알고리즘은 2가지 측면에서 장점을 가진다: 1) 잡음을 내포하는 특징들을 배제함으로써 분류기의 검출능력이 향상시킨다. 2) 특징 셋의 수를 줄임으로써 분류 속도를 증대한다.

수정된 GA 알고리즘을 이용하여 구한 SVM 분류기 커널식의 최적 파라미터값 γ 와 C 는 표 2와 같이 구해졌다.

2. 보행자 탐색 실험

시뮬레이션 프로그램은 Visual C++을 기반으로 코딩하였으며, Open CV 컴퓨터 비전 툴과 LIBSVM Version 2.88[29]를 사

표 1. 제안한 수정된 GA-SVM 분류시스템과 기존의 GA-SVM 분류시스템 간의 분류성능 비교.

Table1. Performance comparison between the proposed GA-SVM classifier and traditional GA-SVM classifier.

Method	Number of feature	Error rate	Recall rate	Precision rate
Traditional GA-SVM	153	0.02933	0.975902	0.965167
Modified GA-SVM	145	0.008167	0.998648	0.98500

표 2. 수정된 GA 알고리즘을 이용하여 구한 SVM 분류기의 최적 파라미터값.

Table2. Optimal parameters of SVM solved from modified GA algorithm.

No. of feature	Fitness of modified GA training	C	γ
145	0.922550	2.68	1.27

실험1: 다양한 각도로 서있는 보행자 검출

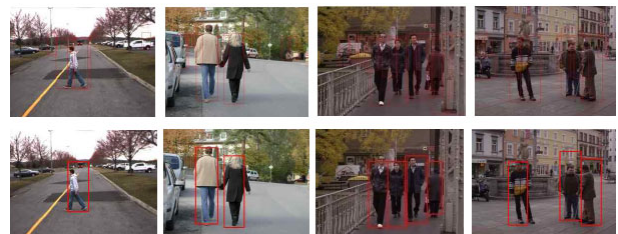


그림 10. 다양한 각도로 서있는 보행자 검출실험결과: 위) 보행자 후보지역 생성, 아래) 보행자후보지역 확정.

Fig. 10. Result of pedestrian detection test with difference views of pedestrian - HG stage (first row of images) and HV stage (second row of images).

실험2: 근거리 혹은 원거리에 있는 보행자 검출

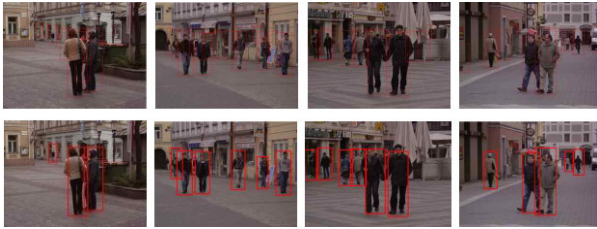


그림 11. 근거리 혹은 원거리에 있는 보행자 검출실험결과: 위) 보행자 후보지역 생성, 아래) 보행자후보지역 확정.

Fig. 11. Result of pedestrian detection test with near and far of pedestrian - HG stage (first row of images) and HV stage (second row of images).

실험3: 매우 밀착해 있는 보행자 검출

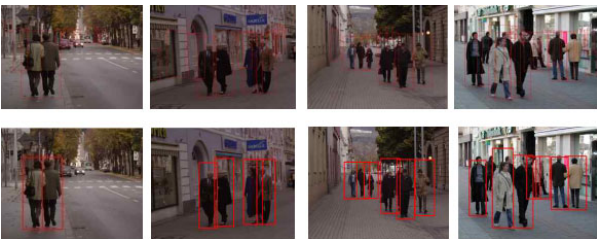


그림 12. 매우 밀착해 있는 보행자 검출실험결과: 위) 보행자 후보지역 생성, 아래) 보행자후보지역 확정.

Fig. 12. Result of pedestrian detection test with pedestrians staying very close to each other - HG stage (first row of images) and HV stage (second row of images).

용하였다. 본 연구에서 제안한 보행자 검출시스템의 우수한 성능을 입증하기 위하여 다양한 환경에서 보행자를 성공적으로 검출할 수 있음을 위의 실험과 같이 보였다.

실험 1~3의 결과에서 알 수 있듯이 제안한 보행자검출 알고리즘이 HG 단계와 HV 단계에서 우수한 결과를 보여주고 있음을 알 수 있다. 얼굴인식을 이용해서 보행자를 인식하는 경우 보행자가 뒤돌아 서 있으면 검출하기 어렵지만 우리의 경우는 보행자의 전면, 측면, 후면의 자세까지 검출이 가능하고, 보행자가 가까이 있거나 멀리 떨어져 있는 경우나 두 사람이 매우 붙어 있는 경우도 정확하게 검출하였다.

V. 결론

본 연구에서는 수정된 GA (Genetic Algorithm)에 기반한 SVM 특징 부분요소선택(features subset selection) 및 분류법을 사용한 보행자검출 알고리즘을 제안하였다. 제안된 알고리즘을 구현하기 위하여 Visual C++, Open CV 컴퓨터비전 툴 computer-vision tool, 그리고 LibSVM 라이브러리를 사용하였다. 제안된 알고리즘은 다양한 배경과 다양한 모습, 다양한 위치에 서 있는 보행자들을 성공적으로 검출할 수 있었다. 본 연구의 학문적 기여도는 다음과 같이 요약할 수 있다:

첫째, 영상의 수평 및 수직 에지와 보행자와 배경간의 색상차이를 적용하여 배경화면과 보행자 영역을 추정하였고 추정지역 내의 보행자의 숫자를 계산하기 위해 새로운 개념인 대칭 히스토그램을 제안하였다. 대칭 히스토그램을 이용

한 보행자 영역을 추출은 영상잡음이나 음영 등에 매우 안정적이다. 그리고 보행자의 자세와 위치 등이 다양해도 검출이 가능하다는 점이다. 특히, 본 연구에서 제안한 알고리즘은 차량에서 보행자까지의 거리정보가 없더라도 보행자 추출이 가능하며 카메라를 보정할 필요도 없다.

둘째, Haar-like 특징 세트, 의사결정 가지 학습 및 수정된 GA에 기반을 둔 SVM 분류기의 최적 파라미터를 구할 수 있는 방법을 제시하여 분류성능을 높였다. 이 방법은 영상의 특징분류뿐만 아니라 다양한 정보나 데이터의 학습을 필요로 하는 분야에도 적용할 수 있다. 특히 의사결정 가지 학습을 이용하여 굉장히 많은 특징세트로부터 상대적으로 적은 특징세트를 추출할 수 있으므로 학습에 소요되는 계산시간과 메모리의 사용량을 줄일 수 있는 장점도 부수적으로 얻을 수 있다.

참고문헌

- [1] Z. Li, K. Wang, L. Li, and F.-Y. Wang, "A review on vision-based pedestrian detection for intelligent vehicles," *Vehicular Electronics and Safety, IEEE International*, pp. 57-62, 2006.
- [2] M. Bertozzi, A. Broggi, M. Felisa, G. Vezzi, and M. Del Rose, "Low-level pedestrian detection by means of visible and infrared tetra-vision," *Proc. IEEE Intelligent Vehicles Symposium*, Tokyo, Japan, pp. 231-236, June 2006.
- [3] M. Bertozzi, A. Broggi, M. Del Rose, and M. Felisa, "A symmetry-based validator and refinement system for pedestrian detection in far infrared images," *Proc. IEEE Intelligent Transportation Systems*, Seattle, USA, pp. 155-160, Sep. 2007.
- [4] M. Bertozzi, A. Broggi, C. Caraffi, M. Del Rose, M. Felisa, and G. Vezzi, "Pedestrian detection by means of far-infrared stereo vision," *International Journal of Computer Vision and Image Understanding*, vol. 106, no. 2-3, pp. 194-204, May 2007.
- [5] M. Bertozzi, E. Binelli, A. Broggi, and M. Del Rose, "Stereo vision-based approaches for pedestrian detection," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 16-22, 2005.
- [6] F. Han, Y. Shan, R. Cekander, H. S. Sawhney, and R. Kumar, "A two-stage approach to people and vehicle detection with HOG-base SVM," *Proc. PerMIS*, pp. 133-140, 2006.
- [7] D. Fernández, I. Parra, M. Á. Sotelo, and P. A. Revenga, "Bounding box accuracy in pedestrian detection for intelligent transportation systems," *Proc. IEEE Industrial Electronics IECON*, pp. 3486-3491, 2006.
- [8] L. Xin, D. Bin, and H. Hagen, "Vision-based real-time pedestrian detection for autonomus vehicle," *Proc. IEEE Vehicular Electronics and Safety*, pp. 1-5, Dec. 2007.
- [9] L. Yu, W. Yao, H. Liu, and F. Liu, "A monocular vision based pedestrian detection system for intelligent vehicles," *Proc. IEEE Intelligent Vehicles Symposium*, pp. 524-529, Eindhoven, The Netherlands, June 2008.
- [10] M. Bertozzi, A. Broggi, Fascioli and P. Lombardi, "Vision-based Pedestrian Detection: will Ants Help?," *Proc. IEEE Intelligent Vehicles Symposium*, vol. 1, pp. 1-7, 2002.
- [11] Guo Lie, Wang Rong-ben, Jin Li-sheng, Li Lin-hui, and Yang Lu, "Algorithm study for pedestrian detection based on monocular vision," *Proc. IEEE Vehicular Electronics and Safety*, pp. 83-87, Dec. 2006.

- [12] G. Ma, S.-B. Park, S. Muller-Schneiders, A. Ioffe, and Kummert, "Vision-based Pedestrian Detection - Reliable Pedestrian Candidate Detection by Combining IPM and 1D Profile," *Proc. IEEE Intelligent Transportation Systems*, Seattle, USA, pp. 137-142, Sep. 2007.
- [13] G. Ma, A. Kummert, S.-B. Park, S. Muller-Schneiders, and A. Ioffe, "A symmetry search and filtering algorithm for vision based pedestrian detection system," *SAE Technical Paper series*, World Congress Detroit, Michigan, April 2008.
- [14] T. Hashiyama, D. Mochizuki, Y. Yano, and S. Okuma, "Active frame subtraction for pedestrian detection from images of moving camera," *Proc. IEEE International Conference on Systems, Man and Cybernetics*, pp. 480-485, 2003.
- [15] C. Curio, J. Edelbrunner, T. Kalinke, C. Tzomakas, and W. von Seelen, "Walking pedestrian recognition," *Proc. IEEE International Transactions on Transportation and System*, vol. 1, no. 3, pp. 155-163, 2000.
- [16] S. Munder and D. M. Gavrila, "An experimental study on pedestrian classification," *IEEE International Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1863-1868, 2006.
- [17] M. Bertozzi, A. Broggi, M. Del Rose, M. Felisa, A. Rakotomamonjy, and F. Suard, "A pedestrian detector using histogram of oriented gradients and a support vector machine classifier," *Proc. IEEE Intelligent Transportation Systems*, Seattle, USA, pp. 143-148, Sep. 2007.
- [18] J. B. Song, D.-W. Lee, and J.-T. Park, "Door detection with door handle recognition based on contour image and support vector machine," *Journal of Institute of Control, Robotics and Systems(in Korean)*, vol. 16, no. 12, pp. 1226-1232, 2010.
- [19] L. Jing, K.-H. Woo, and W.-H. Choi, "Gaze direction estimation method using support vector machines (SVMs)," *Journal of Institute of Control, Robotics and Systems(in Korean)*, vol. 15, no. 4, pp. 379-384, 2009.
- [20] D. Chen, X. B. Cao, Y. W. Xu, and H. Qiao, "An evolutionary support vector machines classifier for pedestrian detection," *Proc. IEEE Intelligent Robots and Systems*, Beijing, China, pp. 4223-4227, 2006.
- [21] L. Malago-Borja and O. Fuentes, "Object detection using image reconstruction with PCA," *International Journal on Image and Vision Computing*, doi: 10.1016/j.imavis.2007.03.004, 2007.
- [22] X. B. Cao, Y. W. Xu, D. Chen, and H. Qiao, "Associated evolution of support vector machine-based classifier for pedestrian detection," *International Journal on Information Science 179* (2009), pp. 1070-1077, 2009.
- [23] Q. B. Truong and B. R. Lee, "Vehicle detection algorithm using hypothesis generation and verification," *Lecture Notes in Computer Science, Book Emerging Intelligent Computing Technology and Application*, vol. 5754/2009, pp. 534-543, 2009.
- [24] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features," *Proc. IEEE International Conference on Computer Vision and Parttern Recognition*, vol. 1, pp. 1-511-I-518, 2001.
- [25] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," *Proc. IEEE International Conference on In Image Processing*, vol. 1, pp. I-900-I-903, 2002.
- [26] L. Breiman, J. Friedman, R. A. Olshen, R. and C. J. Stone, *Classification and Regression Trees*, Wadsworth, pp. 203-215, 1984.
- [27] S. R. Gunn, "Support vector machine for classification and regression," *Technical report of Faculty of Engineering, Science and Mathematics School of Electronics and Computer Science, University of Southampton*, USA, 1998, <http://users.ecs.soton.ac.uk>.
- [28] http://www.nicta.com.au/research/projects/AutoMap/computer_vision_datasets.
- [29] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm11111>.



이 병 통

1983년 부산대학교 기계공학과 졸업(학사). 1988년 동 대학원 기계공학과 졸업(석사). 1994년 North Carolina 주립대 대학원 기계공학과 졸업(공학박사). 1995년~현재 울산대학교 기계공학부 교수. 관심분야는 로봇 및 머신비전.



Quoc Bao Truong

1998년 Cantho 대학교(베트남) 컴퓨터학과 졸업(학사). 2003년 Hochiminh 국립대 대학원 컴퓨터학과 졸업(석사). 2007년~현재 울산대학교 기계공학부 박사과정. 관심분야는 머신비전 응용.



김 형 석

2009년 울산대학교 대학원 기계자동차공학과 박사. 2009년~2010년 지능형자동차부품기술개발팀 선임연구원. 2010년~현재 울산대학교 기계기술연구소 위촉연구원. 관심분야는 물고기 로봇, 머신비전.



배 용 환

1984년 부산대학교 기계설계학과 졸업(학사). 1989년 동 대학원 기계공학과 졸업(석사). 1996년 동 대학원 생산기계공학과 졸업(공학박사). 1997년~현재 안동대학교 기계교육과 교수. 관심분야는 생산자동화 및 E-maintenance 분야.