

# 외곽선 영상과 Support Vector Machine 기반의 문고리 인식을 이용한 문 탐지

## Door Detection with Door Handle Recognition based on Contour Image and Support Vector Machine

이 동 욱, 박 중 태, 송 재 복\*  
(Dong-Wook Lee<sup>1</sup>, Joong-Tae Park<sup>1</sup>, and Jae-Bok Song<sup>1</sup>)  
<sup>1</sup>Korea University

**Abstract:** A door can serve as a feature for place classification and localization for navigation of a mobile robot in indoor environments. This paper proposes a door detection method based on the recognition of various door handles using the general Hough transform (GHT) and support vector machine (SVM). The contour and color histogram of a door handle extracted from the database are used in GHT and SVM, respectively. The door recognition scheme consists of four steps. The first step determines the region of interest (ROI) images defined by the color information and the environment around the door handle for stable recognition. In the second step, the door handle is recognized using the GHT method from the ROI image and the image patches are extracted from the position of the recognized door handle. In the third step, the extracted patch is classified whether it is the image patch of a door handle or not using the SVM classifier. The door position is probabilistically determined by the recognized door handle. Experimental results show that the proposed method can recognize various door handles and detect doors in a robust manner.

**Keywords:** door detection, door handle recognition, general hough transform, support vector machine

### I. 서론

실내환경에서 동작하는 이동로봇은 안전한 자율주행을 위해 환경에 대한 다양한 정보를 이용한다. 지도작성 및 위치추정을 위해 환경 내에 존재하는 코너, 직선과 같은 기하학적인 특징 정보들을 사용하는 것이 그 예이다. 그러나 직선 및 코너와 같은 특징들은 단순하고 정형화된 환경 이외의 환경에서는 추출하기 어려운 단점이 있다. 또한, 각각의 특징들을 확실하게 구분할 수 있는 대표성이 부족하여, 추출된 특징들간의 오정합이 발생할 가능성이 높다. 따라서 복잡한 환경에서도 위치추정이나 지도작성에 강인하게 사용될 수 있는 특징을 추출할 수 있어야 하는데, 이러한 특징 중 하나로 단순하거나 복잡한 모든 실내 환경에 공통적으로 존재하는 문을 고려할 수 있다. 만약, 로봇이 환경 내에 존재하는 문을 강인하게 추출하여 사용할 수 있다면 보다 신뢰성 있는 주행을 수행할 수 있다. 예를 들어, 복도와 같이 기하학적인 특징이 비슷하여 위치 추정 성공률이 떨어지는 환경에서 탐지된 문의 위치를 통해 강인하게 위치추정을 수행할 수 있다[1]. 또한, 탐지된 문을 통해 공간을 분류하고, 분류된 공간에 의미를 부여한 시맨틱(semantic) 지도를 작성할 수 있다[2,3].

이를 위해 환경 내에 존재하는 문을 탐지하여 주행에 활용하는 다양한 연구가 수행되었다. 대표적으로, 카메라로부터 획득된 영상에서 수직선 및 수평선을 추출하여, 추출된 직선

들간의 교점을 이용한 문 탐지 방법이 제안되었다[4]. 이 방법은 비교적 간단한 방법을 통해 문을 탐지할 수 있는 장점을 가지고 있으나, 추출된 직선 간의 교점의 개수가 많아질수록 연산량 및 문 탐지 오정합률이 증가하는 단점을 가지고 있다. 또한, 직선 추출을 위한 파라미터 값이 문 탐지율에 큰 영향을 주는 단점도 가지고 있다.

또한, 상대적으로 직선보다 추출되기 쉬우며 대표성이 강한 코너점을 통해 문을 탐지하는 연구가 수행되었다[5]. 이 방법은 카메라 입력영상에서 코너점을 추출하여, 추출된 점의 상대적인 위치와 실제 환경에서 문의 크기를 비교하여 문을 탐지하는 방식이다. 이를 통하여 좀더 강인하게 문을 탐지할 수 있었으나, 추출된 코너점의 오정합으로 인해 문이 아닌 영역을 문이라고 판단하는 문제점을 가지고 있다. 이외에도, 허프(hough) 변환 알고리즘을 이용하여 원형의 문고리를 인식하고 이를 이용해 문을 탐지하는 방법이 제안되었다[6]. 그러나 제안된 방법은 원형의 문고리만을 인식할 수 있고, 문고리가 아닌 원형인 부분을 문고리로 인식하는 문제점을 가지고 있다.

본 논문에서는 다양한 형태의 문고리를 인식하여 실제 환경에서 안정적으로 문을 탐지하는 기법을 제시하였다. 허프 변환 알고리즘의 확장된 형태인 GHT (General Hough Transform) [7] 알고리즘을 개선하여 다양한 형태의 문고리 인식에 적용하였다. 또한, 외곽선 정보를 이용해 문고리를 인식하는 GHT 알고리즘의 단점을 보완하고, 인식률을 향상시키기 위해 기계학습의 일종인 SVM (Support Vector Machine) [8] 을 통해 생성된 분류기를 이용하였다. 마지막으로, 인식된 문고리의 위치에 확률 기법을 적용하여 문고리의 오인식에도 강인하게 문을 탐지할 수 있도록 하였다.

\* 책임저자(Corresponding Author)

논문접수: 2010. 6. 23., 수정: 2010. 8. 27., 채택확정: 2010. 8. 31.

이동욱, 박중태: 고려대학교 메카트로닉스 협동과정 대학원

(dlehddnr86@korea.ac.kr/geullu@korea.ac.kr)

송재복: 고려대학교 기계공학부(jbsong@korea.ac.kr)

※ 본 연구는 지식경제부 지원으로 수행하는 21C 프론티어 연구개발 사업 및 융복합형 로봇전문인력 양성사업의 지원으로 수행되었음.

본 논문의 구성은 다음과 같다. II 장에서는 전체 문 탐지 과정에 대해 서술하고, III 장에서는 문고리에 대한 DB 생성 방법과 생성된 DB로부터 GHT 알고리즘과 SVM을 이용하여 문고리를 인식하는 방법에 대해 설명한다. IV 장에서는 인식된 문고리의 위치에 대한 확률을 이용하여 문을 탐지하는 방법에 대해 설명하며, V 장에서는 실제 문고리 인식과 인식된 문고리를 이용한 문을 탐지하는 실험 및 결과에 대해 설명한다. 마지막으로 VI 장은 본 논문의 결론에 대해 서술한다.

II. 문 탐지 개요

본 논문에서 제안하는 문 탐지 방법은 그림 1과 같이 크게 세 부분으로 구성된다. 첫 번째 단계는 다양한 형태의 문고리 인식을 위해 문고리에 대한 DB를 생성하는 단계이다. 로봇이 주행하면서 획득되는 영상에서 나타나는 문고리의 기하학적인 변화를 고려하여 영상 DB를 생성하고, GHT 알고리즘을 적용하기 위해 R-table이라는 표를 작성한다. 또한, 인식률을 향상시키기 위해 앞서 획득한 영상 DB와 SVM 기법을 통해 문고리 분류기를 생성한다. R-table과 SVM 분류기에 관한 내용은 3절에서 자세히 설명한다. 두 번째 단계는 앞서 생성한 R-table과 입력영상을 GHT 알고리즘에 적용하여 문고리를 인식하는 과정이다. 안정적인 문고리 인식을 위해 입력영상에서 문고리에 대한 관심 영역(ROI: Region of Interest)을 설정하고, 설정된 관심 영역에서 외곽선을 추출한다. 추출된 외곽선과 R-table을 비교하여 다수의 문고리 후보 영역을 입력영상에서 추출한다. 이 후, 추출된 문고리 후보 영역에서 문고리가 존재하는 영역과 그렇지 않은 영역을 SVM 분류기를 통해 분류한다. 세 번째 단계는, 인식된 문고리의 위치에 확률 기법을 적용하여 안정적으로 문을 탐지하는 과정이다. 입력영상에서 인식한 문고리의 위치의 확률값이 일정값 이상일 경우 오인식 없이 문고리를 인식하였다고 판단하고, 그 영상을 획득한 지점에서 문이 탐지되었다고 판단한다.

III. 문고리 인식

1. General Hough transform

GHT 알고리즘은 영상 내에서 임의의 형태의 도형을 추출할 수 있다. 추출할 도형에 대한 정보는 외곽선과 기준점을 이용하여 나타내는데, 기준점은 보통 도형의 무게중심으로 정한다. GHT를 통해 도형을 추출하는 과정을 그림 2와 3을 통해 자세히 설명한다. 그림 2(a)와 같이 도형의 외곽선 위의

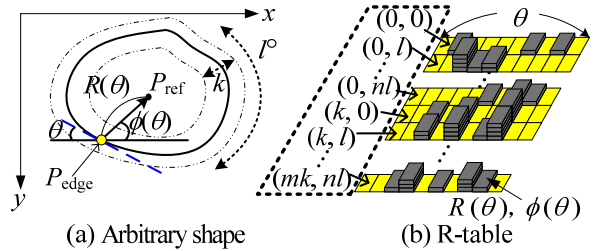


그림 2. GHT 알고리즘. Fig. 2. GHT algorithm.

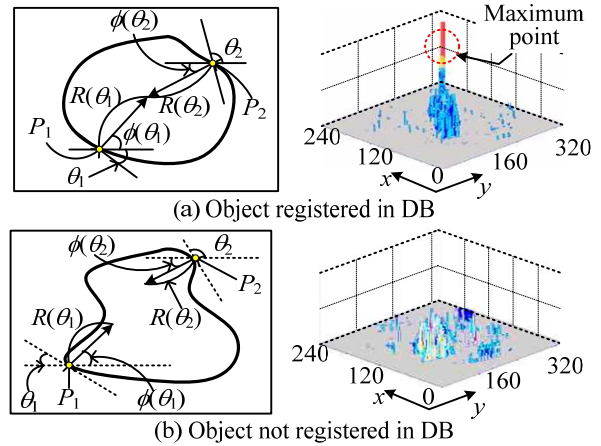


그림 3. GHT 알고리즘을 통한 물체 검출. Fig. 3. Object detection using GHT algorithm.

임의의 점  $P_{edge}$ 에서 그은 접선의 기울기  $\theta$ (일반적으로  $\theta$ 의 범위는  $[0^\circ \sim 180^\circ]$ , 단위는  $1^\circ$ )를 인덱스로 하고, 기준점  $P_{ref}$ 까지 거리  $R(\theta)$ 와 각도  $\phi(\theta)$ 를 이용하여 그림 2(b)과 같이 R-table를 작성한다. 이때  $\theta$ 의 간격에 따라 도형의 추출 성능과 계산량의 차이가 있다. 예를 들어,  $\theta$ 의 간격을  $5^\circ$ 로 설정하면  $1^\circ$  단위로 설정하는 경우에 비하여, 계산량이 적어 도형의 추출 속도가 빠르지만, 도형의 왜곡에 민감하여 기하학적인 변화가 발생하면 도형을 추출하는 데 어려움이 발생한다.

다음 과정은 크기와 회전 변화가 발생하여도 안정적으로 도형을 추출하기 위하여, 그림 2(a)의 점선으로 표시된 화살표와 같이 추출할 도형을  $k$ 배만큼 확대 또는 축소하거나,  $P$ 만큼 회전하여 이와 같은 과정을 반복한다. 크기 또는 회전 변화를 주어 R-table을 작성할 때 그림 2(b)의 점선으로 표시한 부분과 같이 각각의 변화를 순서쌍 형태로 같이 저장한다. 이 과정은 크기 및 회전 변화의 순서쌍이  $(mk, nl)$ 이 될 때까지 반복한다. 이때,  $m, n$ 은 각각 크기 또는 회전 변화의 양을 나타내는 변수(정수)이다. 이 값이 클수록 찾고자 하는 도형의 기하학적인 변화에 대해 강한 R-table을 작성할 수 있지만, 입력영상에서 도형을 추출할 때 계산량이 많아진다는 단점이 있다. 반면에,  $m, n$ 이 작을수록 도형을 추출할 때 계산량이 적어지지만, 다양한 양의 기하학적인 변화가 발생하는 도형에 대해서는 정보를 저장하지 않고 있으므로 이러한 도형을 추출하는 데 어려움이 발생하게 된다. 그러므로 입력영상에서 찾고자 하는 도형에 대한 기하학적인 변화의 양을 고려하여  $m, n$ 을 설정해야 한다.

도형의 검출 과정은 도형에 대한 기준점을 찾는 방식으로

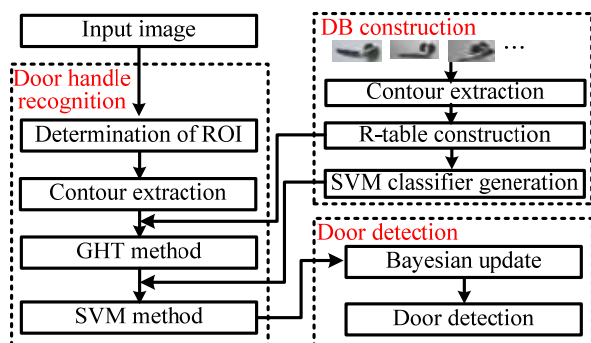


그림 1. 문 탐지 개요도. Fig. 1. Schemes of door detection.

수행된다. 기준점을 찾는 과정은 앞서 생성한 R-table과 실제 입력영상에서의 추출된 외곽선을 이용하여 다음과 같이 수행된다. 그림 3(a)와 같이, 입력영상에서 추출된 외곽선의 임의의 점  $P_n(x, y)$ 에서 계산한 접선의 기울기가  $\theta_n$ 일 때 R-table에서  $\theta_n$ 에 해당하는 모든  $R(\theta_n), \phi(\theta_n)$ 을 다음 식에 대입하여 계산된 좌표( $x_{est}, y_{est}$ )에 보팅(voting)한다.

$$\begin{aligned} x_{est} &= x + R(\theta_n) \cos(\phi(\theta_n)) \\ y_{est} &= y + R(\theta_n) \sin(\phi(\theta_n)) \end{aligned} \quad (1)$$

예를 들어,  $\theta$ 에 대하여 R-table에  $R(\theta), \phi(\theta)$ 가 5개가 저장되어 있다면 허프공간에는 5곳의 ( $x_{est}, y_{est}$ ) 좌표가 보팅된다. 모든 외곽선에 대해 보팅이 완료되면, 찾고자 하는 도형의 기준점은 가장 많이 보팅된 지점이다. 그림 2와 비교했을 때, 그림 3(a)와 같이 찾고자 하는 도형이 영상 내에 존재하면 보팅 결과가 한 점으로 수렴하고, 수렴된 점이 찾고자 하는 도형의 기준점이 된다. 반면에, 그림 3(b)와 같이 다른 도형이 존재하면 보팅 결과가 발산하게 되어 도형의 기준점을 찾을 수 없다. 이와 같이, GHT 알고리즘을 이용하면 회전 변화나 크기 변화에 강인하게 도형을 추출할 수 있고, 다소의 왜곡이나 일부 가려진 영역에 대해서도 추출이 가능하다.

2. GHT 알고리즘과 SVM 분류기를 이용한 문고리 인식

2.1 문고리에 대한 R-Table 생성

문고리에 대한 R-table을 작성하기 위해서는 문고리의 영상 DB가 필요하다. 영상 DB는 그림 4와 같이 획득한다. 문고리는 문에 고정되어 있으므로, 로봇이 벽으로부터 일정 거리만큼 떨어져서 주행하면서 획득되는 영상에서 그림 4와 같이 회전 변화가 없고, 크기 변화와 어파인(affine) 변화만 발생한다. 이러한 기하학적 변화를 고려하여 45°, 0°, -45° 세 방향에

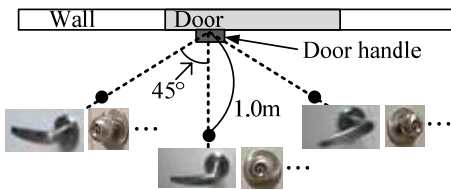


그림 4. 문고리의 영상 DB 획득.  
Fig. 4. Acquisition of image database of door handle.

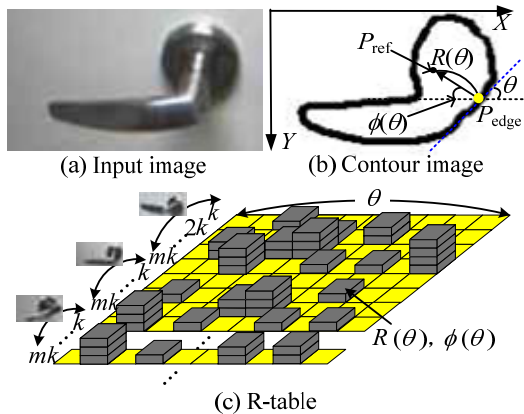


그림 5. R-table 작성.  
Fig. 5. Making the R-table.

서 일정 거리(카메라의 인식 성능을 고려한 거리)만큼 떨어진 지점에서 영상 DB를 획득한다.

문고리에 대한 R-table은 각 영상 DB의 외곽선과 기준점을 이용해서 작성한다. 그림 5(a)와 같이 문고리의 영상 DB에서 그림 5(b)처럼 잡음에 강인한 외곽선을 캐니 경계검출기(canny edge detector)를 이용하여 추출하며, 기준점은 문고리의 무게 중심으로 한다. 그리고 2.1절에서 언급한 대로 문고리 외곽선 상의 임의의 점에서 그은 접선의 기울기와 그 점에서 기준점까지의 거리와 각도를 이용하여 R-table을 작성한다. 기존의 GHT 알고리즘은 도형의 크기 변화와 회전 변화를 고려하여 R-table을 작성하였지만, 앞서 언급한 대로 문고리는 회전 변화가 나타나지 않으므로, 그림 5(c)와 같이 어파인 변화와 크기 변화에 대한 R-table을 작성한다.

2.2 관심 영역의 설정

GHT 알고리즘을 이용하여 문고리를 인식하기 위해서 입력영상의 외곽선도 필요하다. 입력영상에서 전처리 과정 없이 외곽선을 추출하면, 다수의 외곽선이 추출된다. 이렇게 추출된 외곽선을 R-table과 비교하여 문고리를 찾는 과정은 많은 계산량을 요구하며, 또한 오인식이 발생할 가능성이 크게 된다. 따라서 이러한 문제점을 개선하기 위해 문고리에 대한 관심 영역을 설정하고, 그 영역에서 추출된 외곽선만을 R-table과 비교하여 문고리를 찾는다. 문고리에 대한 관심 영역은 문고리의 색 정보, 주변 환경을 고려하여 설정한다.

입력영상에서 문고리의 색 정보와 비슷한 영역은 컬러 필터를 이용하여 추출하고, 그 영역을 관심 영역으로 설정한다. 이 때 사용되는 컬러 필터의 임계치는 영상 DB에서 문고리 영역의 각 색 공간에 대한 히스토그램으로부터 결정된다. 그림 6(a)와 같이 영상 DB에서 마스크 이미지를 이용하여 문고리와 배경을 분리하고, 분리된 문고리 영역의 컬러 히스토그램을 구한다. 컬러 필터의 임계치는 그림 6(b)와 같이 앞서 구한 히스토그램에서 최대값을 가지는 픽셀값  $X_{max}$ 을 기준으로, 분리한 문고리 영역의 전체 픽셀의 개수의 80%가 되는 범위 [ $X_{low}, X_{high}$ ]로 정한다.

실제 환경에서는 그림 7(a)와 같은 입력영상에서 문고리 색 정보가 조명 변화나 노이즈로 인해 DB에서의 문고리 색 정보와 달라질 수 있다. 따라서 그림 7(b)같이 컬러 필터링된 영역(빨간색 영역)을 바로 사용하지 않고, 그림 7(c)와 같이 필터링된 영역을 모폴로지 연산을 통해 확장한다. 그림 7(d)

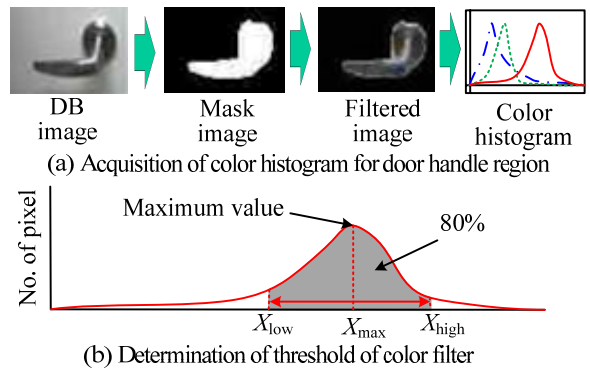


그림 6. 문고리에 대한 컬러 필터의 설정.  
Fig. 6. Determination of ROI for color information of door handle.

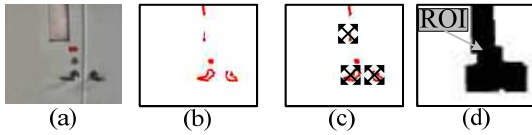


그림 7. 색 정보를 이용한 관심 영역 설정: (a) 입력영상, (b) 컬러 필터링된 영상, (c) 영역 확장 (d) 관심 영역.

Fig. 7. Determination of ROI using color information of door handle: (a) input image, (b) color filtered image (c) expansion of color filtered region, and (d) ROI.

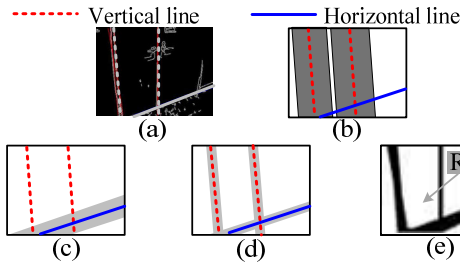


그림 8. 직선정보를 이용한 관심영역 설정: (a) 수평선 및 수직선 추출, (b) 수직선에 대한 포함영역, (c) 수평선에 대한 제외영역, (d) 추출된 직선들에 대한 제외영역 (e) 직선정보에 기반한 관심영역.

Fig. 8. Determination of ROI using line information: (a) acquisition of vertical and horizontal lines, (b) included region for vertical lines, (c) excluded region for horizontal lines, (d) excluded region for extracted lines, and (e) ROI.

는 입력영상에서 최종적으로 선택된 관심 영역을 나타낸다.

다음으로 문고리의 주변 환경을 고려하여 관심 영역을 설정한다. 일반적으로 문고리는 문의 경계 근처에 위치해 있고, 벽과 바닥 또는 천장과 의 경계에는 존재하지 않으므로 이러한 주변 환경 정보를 이용하여 관심 영역을 추출할 수 있다. 문의 경계는 영상에서 수직선으로 나타나고, 벽과 바닥 또는 천장과 의 경계는 수평선으로 나타난다. 그림 8(a)와 같이 허프 변환을 이용하여 입력영상에서 수평선 및 수직선을 추출한다. 수직선 근처에는 문고리에 대한 외곽선이 존재할 수 있으므로 그림 8(b)와 같이 수직선으로부터 일정 영역을 포함 영역으로 등록한다. 수평선 근처와 추출된 직선들(수평선, 수직선) 바로 위에서는 문고리가 존재하지 않으므로 그림 8(c)와 (d)와 같이 제외 영역으로 등록한다. 문고리의 주변 환경에 대한 관심 영역은 그림 8(e)와 같이 전체 영상에서 포함 영역인 부분과 제외 영역이 아닌 부분으로 설정한다.

정리하자면, 앞서 구한 컬러 필터를 이용한 관심 영역, 수직선 또는 수평선 추출을 이용한 관심 영역중 공통된 부분만을 문고리 추출을 위한 관심 영역으로 설정한다.

### 2.3 클러스터링

실제 입력영상에서 관심 영역을 설정하고, 그 영역에서 추출된 외곽선과 R-table을 비교하여 허프 공간에 문고리의 기준점을 보팅하게 되면 한 지점으로 보팅되지 않고, 그 주위로 분산되어 보팅되거나, 문고리의 기준점이 아닌 곳에 보팅되는 경우가 발생한다. 그 이유는 조명 변화, 노이즈, 문고리의 기하학적인 변화로 인해 추출되는 외곽선의 왜곡되거나 관심 영역에서 문고리가 아닌 다른 외곽선들도 같이 추출되

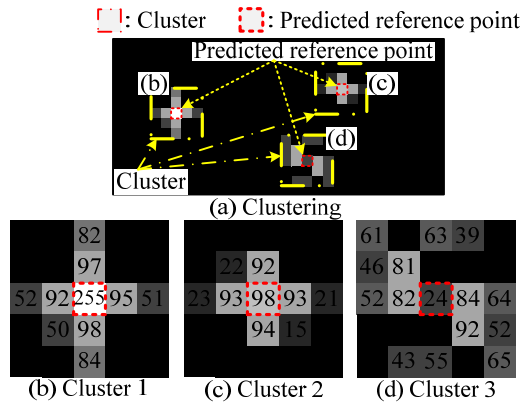


그림 9. 보팅 결과의 클러스터링.

Fig. 9. Clustering of result of GHT.

기 때문이다. 이러한 문제점을 개선하기 위해 허프 공간에 보팅된 값들을 클러스터링(clustering)하여 클러스터의 대표점을 문고리의 기준점으로 판단한다.

만약 그림 9(a)와 같이 여러 개의 클러스터가 생성될 경우, 클러스터를 구성하는 원소들의 총 보팅된 횟수, 위치 분산을 이용하여 문고리가 존재하는 클러스터를 찾는다. 이상적인 경우 문고리의 모든 외곽점에서 추정된 기준점이 한 지점에 보팅되고, 그 횟수는 외곽점의 개수와 같다. 하지만 실제 환경에서는 앞서 언급한 문제들로 인해 클러스터의 원소들이 나누어 보팅된다. 그러므로 클러스터 내의 원소의 총합은 모든 외곽점의 개수에 근접하여야 한다. 만약 외곽점의 개수보다 너무 적거나 많을 경우 기준점을 포함하지 않는 클러스터라고 가정하고 제외한다. 또한, 클러스터 원소의 총 합이 외곽점의 개수와 비슷하더라도 기준점에 모여있지 않고 그 주위로 퍼져서 위치 분산이 큰 경우에도 제외한다. 클러스터의 원소들에 대한 위치 분산은 보팅 횟수를 고려하여 다음 식과 같다.

$$(\sigma_x^2, \sigma_y^2) = \left( \sum_{k=1}^N \frac{P(x_k, y_k)}{\sum_{k=1}^N P(x_k, y_k)} (x_c - x_k)^2, \sum_{k=1}^N \frac{P(x_k, y_k)}{\sum_{k=1}^N P(x_k, y_k)} (y_c - y_k)^2 \right) \quad (2)$$

여기서  $(\sigma_x^2, \sigma_y^2)$ 은 각각의 클러스터의 x축, y축 분산이고,  $(x_c, y_c)$ 은 클러스터의 대표점의 위치이고,  $P(x_k, y_k)$ 는 클러스터의 k 번째 원소의 보팅된 횟수이며, N은 클러스터의 총 원소의 개수이다. 예를 들어, 문고리 외곽선의 총 픽셀수가 900개이고, 문고리가 존재하는 클러스터를 찾는 기준이 클러스터의 원소의 합은  $900 \pm 10\%$ , x축과 y축 분산의 임계치는 각각 1이라고 가정한다. 그림 9(b)의 클러스터는 각각 원소의 합이 956이고, 분산은 (0.6, 0.95)이므로 문고리가 존재하는 클러스터라고 가정한다. 하지만 그림 9(c)와 같은 경우 클러스터의 원소의 합이 551이므로 설정한 범위 내에 없고, 그림 9(d)와 같은 경우 원소의 합은 903으로 범위 내에 속하지만 분산이 (1.97, 1.74)로 분산의 범위에서 벗어나므로 제외된다. 이와 같이 클러스터링 과정을 통해 문고리의 기준점을 안정적으로 추정할 수 있다.

2.4 SVM 분류기를 이용한 문고리 인식 성능 향상

GHT 알고리즘만을 이용하여 문고리를 인식할 경우 그림 10(a)와 같이 인식이 잘되는 경우도 있지만, 그림 10(b)와 같이 영상에서 문고리와 비슷한 외곽선을 가진 부분을 문고리로 오인식하는 단점이 있다. 이러한 오인식을 줄이기 위해 SVM 분류기를 사용하여 GHT 알고리즘으로 추정된 문고리 위치의 영상을 문고리가 존재하는 영상과 그렇지 않은 영상으로 분류한다.

분류기 생성에 사용되는 데이터 집합은 문고리에 대한 영상조각(positive image patch)과 그 외의 영상조각(negative image patch) 두 종류로 구성되어 있다. 각각의 영상조각에서 문고리에 해당하는 영역의 컬러 히스토그램을 추출하고, 이를 특징벡터로 변환한다. 그림 11와 같이 변환된 특징벡터를 벡터 공간에 매핑하여 분류기를 생성한다. 분류기 생성에 사용되는 문고리의 영상조각은 그림 12와 같은 과정을 거쳐 획득된다. 실제 환경에서 획득된 영상은 크기 변화, 어파인 변화만 발생된다. SVM에 학습되는 데이터 집합은 영상 DB로부터 생성하므로, 문고리의 어파인 변화만을 고려한다. 그러므로 데이터 집합은 크기 변화를 고려하여 그림 12와 같이 각각의 영상 DB를 크기를 변화시켜 획득한 영상 조각들로 구성된다. 생성된 분류기는 GHT 알고리즘을 이용하여 추정된 문고리의 위치의 일정 영역을 문고리가 존재하는 영상과 그렇지 않은 영상으로 분류한다. 예를 들어, GHT 알고리즘을 통해 추출된 문고리의 후보 영역이 그림 13(a)와 같은 경우, 세 영

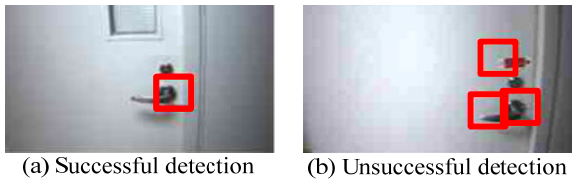


그림 10. GHT 알고리즘을 이용한 문고리 인식.  
Fig. 10. Door handle recognition with GHT.

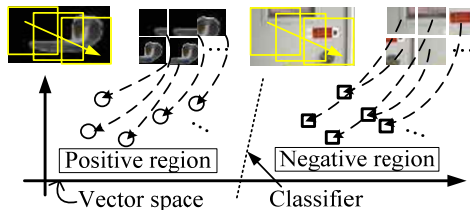


그림 11. Positive 및 negative 영상 조각 추출 및 벡터 공간 매핑.  
Fig. 11. Positive and negative image patch and mapping at vector space.

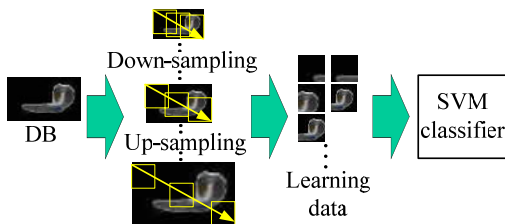


그림 12. 학습 데이터 생성.  
Fig. 12. Construction of learning data.

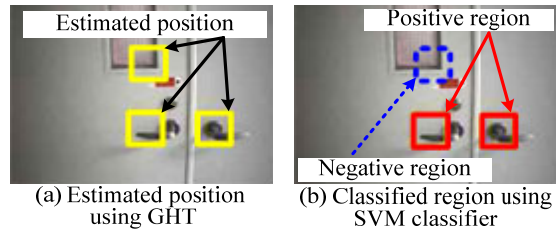


그림 13. SVM을 이용한 문고리 인식.  
Fig. 13. Door handle recognition with SVM.

역에 대해 분류기를 이용하여 분류하면 그림 13(b)와 같이 문고리가 존재하는 영역(positive region)과 존재하지 않는 영역(negative region)으로 구분된다.

IV. 확률 기반의 문 탐지

인식한 문고리의 위치를 이용하여 바로 문을 탐지하면 인식하는 과정에서 잘못된 인식(False-positive recognition)으로 인해 문 탐지에 실패하는 경우가 발생한다. 따라서 안정적이고 정확한 문 탐지를 위해 인식한 문고리 위치에 확률 기법을 적용한다.

그림 14(a)와 같이 주행 중 획득한 영상에서 문고리를 인식한 경우, 영상에서 그 위치에 해당하는 영역의 확률이 증가된다. 그 다음 획득되는 영상에서도 계속 같은 지점에 있는 문고리가 인식되는 경우에는 확률은 높아지고, 그렇지 않으면 낮아지게 된다. 이와 같은 과정을 통해 현재 영상에서 문고리를 인식한 경우, 그 위치의 현재 확률이 일정한 값(0.8) 이상일 때 오인식 없이 문고리를 인식하였고, 그 지점에 문이 존재한다고 판단한다. 인식한 문고리 위치에 대한 확률 갱신은 베이시안 갱신방법에 적용하여 수행하였다[9]. 영상  $I_0$ 에서  $I_n$ 까지 인식한 문고리의 위치 정보  $z_{0:n}$ 가 주어지는 경우, 현재 영상  $I_n$ 에서  $I_n(x, y)$ 에 문고리가 존재하는 확률  $p(I_n(x, y)|z_{0:n})$ 은 다음과 같다.

$$p(I_n(x, y)|z_{0:n}) = \frac{p(I_{n-1}(x, y)|z_{0:n-1})p(I_n(x, y)|z_n)}{[1 - p(I_{n-1}(x, y)|z_{0:n-1})][1 - p(I_n(x, y)|z_n)]} \quad (9)$$

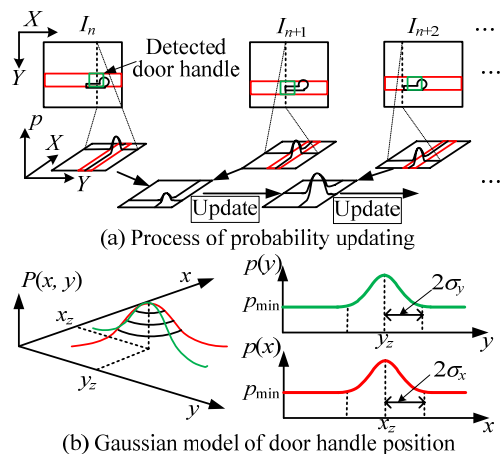


그림 14. 인식된 문고리 위치에 대한 확률을 통한 문 탐지.  
Fig. 14. Door detection with probability for recognized door handle position.

여기서  $p(I_n(x, y)|z_{0:n-1})$ 는 이전 영상의  $I_{n-1}(x, y)$ 에 문고리가 존재할 확률,  $p(I_n(x, y)|z_n)$ 는 현재 영상에서 인식을 통해  $I_n(x, y)$ 에 문고리가 존재할 확률이다.  $p(I_n(x, y)|z_n)$ 은 그림 15(b)와 같이 결합 가우시안 모델을 이용하여 계산하며 다음과 같다.

$$\begin{cases} p_{\min} & (0 \leq x < x_z - \sigma_x) \\ p_{\min} + \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left\{-\frac{(x-x_z)^2}{2\sigma_y^2}\right\} & (x_z - \sigma_x \leq x < x_z + \sigma_x) \\ p_{\min} & (0 \leq y < y_z - \sigma_y) \\ p_{\min} + \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left\{-\frac{(y-y_z)^2}{2\sigma_x^2}\right\} & (y_z - \sigma_y \leq y < y_z + \sigma_y) \end{cases} \quad (4)$$

여기서  $p_{\min}$ 은 최소 확률(0.2)이고,  $(\sigma_x, \sigma_y)$ 는 인식된 문고리 위치에 대한 분산이다.

**V. 실험 환경 및 결과**

**1. 다양한 형태의 문고리 인식 실험 및 결과**

그림 15(a)와 같이 다양한 문고리에 대한 인식 성능을 알아보기 위해, 그림 15(b)과 같이 문고리로부터 거리(1m, 1.5m, 2m)와 각도(-45°, 0°, 45°)를 달리하여 인식 실험을 수행하였다. 문고리 인식은 Point Grey사의 Bumblebee2를 이용하여 획득한 영상을 사용하였다. 문고리에 대한 R-table 작성시  $\theta$ 는 2° 간격,  $k=0.5$ ,  $m=3$ 으로 설정하였다. 영상의 크기는 320×240으로 획득한 영상에서 문고리를 인식하는데 걸리는 시간은 평균 350 ms-450 ms 이다. 문고리 인식의 결과는 표 1과 같고, 인식률  $P_{\text{recog}}$ 은 다음과 같이 정의하였다.

$$P_{\text{recog}} = \frac{I_{\text{recog}}}{I_{\text{total}}} \quad (5)$$

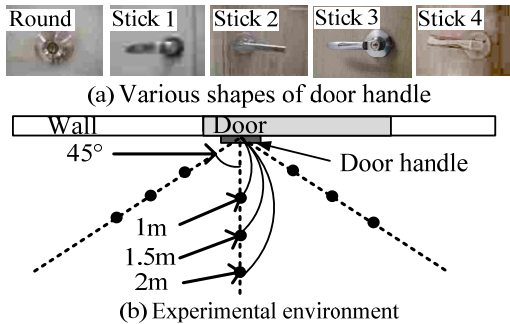


그림 15. 다양한 형태의 문고리 및 인식 성능 실험 환경.

Fig. 15. Experimental environment and performance of door handle recognition.

표 1. 다양한 형태의 문고리 인식률.

Table 1. Recognition rate of various shape of door handles.

	1m			1.5m			2m		
	45°	0°	-45°	45°	0°	-45°	45°	0°	-45°
Round	88%	92%	86%	62%	82%	51%	0%	0%	0%
Stick1	91%	98%	92%	89%	97%	87%	0%	0%	0%
Stick2	52%	59%	51%	9%	10%	7%	0%	0%	0%
Stick3	68%	78%	69%	67%	70%	63%	0%	0%	0%
Stick4	51%	64%	55%	46%	42%	39%	0%	0%	0%

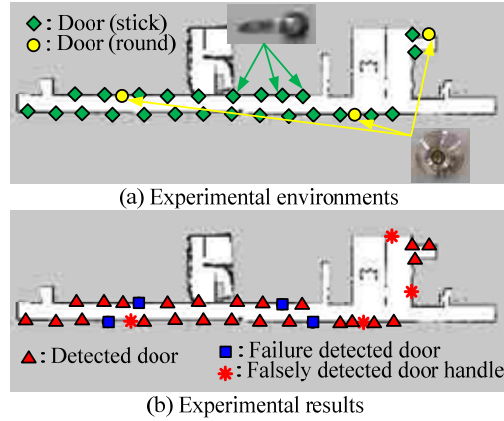


그림 16. 실험 환경 및 문 탐지 결과.

Fig. 16. Experimental environments and result of door detection.

여기서  $I_{\text{total}}$ 은 총 획득한 영상의 개수이고,  $I_{\text{recog}}$ 는 문고리를 인식한 영상의 개수이다.

표 1에서 보면, 본 논문에서 제안된 문고리 인식 방법은 1m 거리에서 단순한 형태의 문고리인 round나 stick1을 80% 이상으로 인식할 수 있었다. 그러나 크기가 작은 문고리(stick 2)의 경우 추출되는 외곽선이 너무 작고, 복잡한 무늬가 있는 문고리(stick 3)를 인식할 경우 추출된 외곽선에서 문고리 모양을 찾을 수 없어 낮은 인식 성능을 보였다. 거리가 멀어질수록 가까운 거리에서 인식하였을 때 보다 낮은 인식률을 보였는데, 이는 입력영상에서 문고리의 크기는 거리가 멀어질수록 작아지고, 따라서 추출되는 외곽선도 작아지기 때문이다. 특히 2m에서 문고리를 인식한 경우, 획득되는 영상에서 문고리에 대한 픽셀들이 단순한 직선의 형태로 나타나기 때문에 GHT 알고리즘만으로는 인식이 어렵다. 또한, 측면에서 문고리를 인식한 경우(45°, -45°), 어파인 변화로 인해 정면에서 인식하는 경우보다 인식률이 저하된다. R-table 생성시 다양한 어파인 변화를 고려하여 생성하지 않았기 때문에 어파인 변화가 조금만 발생하면 R-table에 저장되어 있는 문고리의 외곽선 정보와 많이 달라져 인식률이 낮아진다.

**2. 문 탐지 실험 및 결과**

실제 문 탐지는 벽으로부터 일정한 거리(1.0 m)만큼 떨어진 지점을 추종하고, 벽을 바라보며 획득된 영상을 이용하여 수행하였다. 문 탐지율  $p_{\text{door}}$ 는 문고리의 오인식으로 인한 잘못된 지점을 문이라고 탐지하는 경우를 고려하여 다음과 같이 정의한다.

$$p_{\text{door}} = \frac{N_{\text{door}}}{N_{\text{total}}} \times \frac{N_{\text{door}}}{N_{\text{dd}}} \quad (6)$$

여기서  $N_{\text{door}}$ 은 정확하게 문을 탐지한 횟수,  $N_{\text{total}}$ 은 현재 환경에 존재하는 총 문의 개수,  $N_{\text{dd}}$ 는 오 탐지를 포함하여 문을 탐지한 총 횟수이다. 예를 들어, 총 10개의 문이 존재하는 복도환경에서 문을 8번 탐지하고 이 중 7번을 정확히 탐지한 경우  $N_{\text{total}}=10$ ,  $N_{\text{dd}}=8$ ,  $N_{\text{door}}=7$ 이므로 탐지율은 61%이다. 실제 문 탐지 실험에서는 그림 16과 같이 문이 28개 존재하는 복도환경을 주행하였다. 한 사무실이나 가정환경에 존재하는 문고리의 종류는 다양하지 않기 때문에 그림 16(a)와 같이 원형 문고리를 추가하여 실험하였다. 실험 결과 28번 문을

탐지였으며(문고리 오인식 포함), 이 중 24번을 정확히 탐지하였다. 최종적인 문 탐지율은 73%이었다.

## VI. 결론

본 논문에서는 실내 환경에서 존재하는 문을 안정적으로 탐지하기 위하여 문고리 인식을 활용하는 방법에 대해 연구하였다. GHT 알고리즘과 SVM을 통해 다양한 형태의 문고리를 인식하고, 인식된 문고리 위치의 확률 분포를 이용하여 안정적인 문 탐지를 수행하였다. 제안된 방법은 단순한 형태의 문고리의 경우 추출되는 외곽선이 뚜렷하여 높은 인식 성능을 보였다. 이와 같은 방법으로 탐지된 문은 지도상에 공간을 분류하는 용도로 사용될 수 있다. 또한, 위치 추정에 강인한 특징으로도 사용될 수 있다. 그러나 본 방법은 영상에서 추출되는 외곽선이 양이 적거나 복잡한 모양의 문고리는 인식하지 못하는 한계를 보였다. 따라서 다수의 외곽선이 추출되는 복잡한 형태의 문고리를 인식할 수 있도록 알고리즘 개선하는 것이 추후 연구과제이다.

## 참고문헌

- [1] I. Monasterio and W. Burgard, "Mobile robot mapping and localization in non-static environments," *Proc. of the National Conf. on Artificial Intelligence*, July 2005.
- [2] O. M. Mozos, P. Jensfelt, H. Zender, G.-J. M. Kruijff, and W. Burgard, "From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots," *Proc. of the IEEE/RSJ Intelligent Robots and Systems*, Apr. 2007.
- [3] C. Stachniss and E. Lázcano, "Learning to traverse doors using visual information," *Mathematics and Computers in Simulation*, vol. 60, no. 3, pp. 347-356, 2002.
- [4] J. J. Guerrero, A. C. Murillo, J. Kosecka, and C. Sagues, "Visual door detection integrating appearance and shape cues," *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 512-521, 2008.
- [5] S. Stoeter and L. Papanikolopoulos, "Real-time door detection in cluttered environments," *Proc. of the IEEE International Symposium on Intelligent Control*, July 2000.
- [6] J. Lee, N. L. Doh, W. K. Chung, B. You, and Y. I. Youm, "Door detection algorithm of mobile robot in hallway using pc-camera," *International Symposium on Automation and Robotics in Construction*, 2004.
- [7] D. H. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, pp. 111-122, 1980.
- [8] V. Vapnik, *Statistical Learning Theory*, John Wiley and Sons, 1998.
- [9] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, The MIT Press, 2005.



### 이 동 욱

2009년 광운대학교 전자공학과(공학사). 2009년~현재 고려대학교 대학원 메카트로닉스와 석사과정 재학중. 관심분야는 이동로봇의 위치추정.



### 박 중 태

2005년 고려대학교 전산학과(이학사). 2007년 고려대학교 메카트로닉스학과(공학석사). 현재 메카트로닉스학과 박사과정 재학중. 관심분야는 이동로봇의 탐사 기법 및 시스템 아키텍처.



### 송 재 복

1983년 서울대학교 기계공학과(공학사). 1986년 서울대학교 기계공설계학과(공학석사). 1992년 MIT 기계공학과(공학박사). 1993년~현재 고려대학교 기계공학부 교수. 관심분야는 안전 머니플레이터, 이동로봇의 주행, 지능로봇 시스템

의 설계 및 제어.