

論文96-33B-1-8

# Generalized Hough Transform을 이용한 이차원 물체인식 비전 시스템 구현에 대한 연구

(A study on vision system based on Generalized Hough Transform 2-D object recognition)

具本哲\*, 朴鎮秀\*\*, 秦成一\*\*\*

(Bon Cheol Koo, Jin Soo Park, and Sung Il Chien)

## 요 약

본 논문은 GHT(Generalized Hough Transform)을 이용하여 물체의 위치, 회전각 및 크기변화를 인식하게 하는 것을 목표로 한다. 특히 회전 및 가려진 물체를 잘 인식함을 보이고자 한다. GHT는 일종의 모델베이스 물체인식 알고리즘으로써 먼저 물체의 정보를 R-table(Reference table) 형태로 저장한 뒤, 그 R-table을 사용하여 물체를 인식한다. 본 논문에서는 GHT 알고리즘을 실제적인 비전 시스템에 적용하기 위하여 GHT 알고리즘을 개선하였다. 첫째, R-table 작성시 물체의 부분적인 경계선으로부터 R-table을 작성하였으며 영상을 디지털화 하였을 경우 발생하는 에러를 보상하기 위하여 클러스터링(clustering) 알고리즘을 사용하였다. 둘째, 기존의 Ballard의 GHT 알고리즘은 물체의 위치, 회전각 및 크기변화를 인식하기 위하여 4차원의 배열이 필요하지만 단지 2차원의 배열만으로 물체인식이 가능하도록 하였다. 특히 크기변화를 인식하는 간단한 방법을 제안하였다. 테스트 결과 제안된 GHT 알고리즘이 실제적인 비전 시스템에 있어서 비교적 잘 동작함을 알 수 있었다. 특히 겹쳐진 물체를 잘 인식함을 알 수 있었다.

## Abstract

The purpose of this paper is object recognition even in the presence of occlusion by using generalized Hough transform(GHT). The GHT can be considered as a kind of model based object recognition algorithm and is executed in the following two stages. The first stage is to store the information of the model in the form of R-table(Reference table). The next stage is to identify the existence of the objects in the image by using the R-table. The improved GHT method is proposed for the practical vision system. First, in constructing the R-table, we extracted the partial arc from the portion of the whole object boundary, and this partial arc can be used for constructing the R-table. Also, clustering algorithm is employed for compensating an error arised by digitizing an object image. Second, an efficient method is introduced to avoid Ballard's use of 4-D array which is necessary for estimating position, orientation and scale change of an object. Only 2-D array is enough for recognizing an object. Especially, scale token method is introduced for calculating the scale change which is easily affected by camera zoom. The results of our test show that the improved hierarchical GHT method operates stably in the realistic vision situation, even in the case of object occlusion.

\* 正會員, LG 電子 미디어 通信 研究所 CS Gr.  
(CS Group, Media Communication R&D Lab.,  
LG Electronics)

\*\* 正會員, 韓國通信 研究開發團 情報處理 研究室  
(Information Processing Technique Section,  
Korea Telecom Research Center)

\*\*\* 正會員, 慶北大學校 電子工學科  
(Dept. of Elec., Eng., Kyungpook Univ.)

※ 본 연구는 1992년 한국과학재단 연구비 지원에 의한 결과임(과제번호: 921-1100-009-2)

接受日字: 1995年3月4日, 수정완료일: 1995年12月16日

## I. 서론

Hough transform<sup>[1-2]</sup>은 영상내의 직선성분의 검출 및 위치탐색에 유용한 방법으로 잘 알려져 있다. 그 후 이 개념이 보다 확장되어 원이나 타원, 포물선 같은 곡선형태의 물체를 인식하는데에도 적용되게 되었다. 그러나 이들 알고리즘들은 함수 형태로 표현 가능한 물체들에 대해서만 제한되어 사용되어 왔다. 1981년에 Ballard<sup>[4]</sup>는 Merlin과 Farber의 알고리즘<sup>[3]</sup>을 발전시켜 명암변화방향정보를 사용하여 비함수 형태의 물체에도 적용할 수 있는 보다 일반화된 Hough transform(generalized Hough transform : GHT)을 제안했다. GHT는 일종의 템플릿 매칭으로서 모델 물체의 정보를 R-table(Reference table)에 기록한 후에 테스트 영상에서 이 R-table을 이용하여 찾고자 하는 물체를 인식한다.

지금까지 GHT의 연구는 원리적이고 주로 artificial 영상에 대하여 진행되어왔으나 실제적인 응용에 대한 연구는 거의 이루어지지 않고 있다. 이에 본 논문은 GHT를 실제적인 비전 시스템에 실용화시켜 동작시키는데 중점을 두고, 이를 위한 몇 가지 방법을 제안했다.

먼저 GHT<sup>[5-9]</sup>의 특징을 살펴보면 여러 물체가 동시에 존재하는 경우에도 각각의 물체를 분할(segmentation)하지 않고 인식할 수 있다. 이는 물체의 분할과정이 일반적인 영상해석에서 상당히 어렵고 계산시간을 많이 요구하는 과정임을 고려할 때 대단한 장점이 될 수 있다. 또한 GHT는 노이즈에 강한 속성이 있어서, 영상인식에 있어서의 어려움 중에 하나인 겹쳐진 물체에 대한 인식<sup>[10-11]</sup>에도 좋은 성능을 보이고 있다. 그리고 GHT 알고리즘은 본질적으로 병렬처리가 가능하므로 하드웨어 구현이 용이하다<sup>[12]</sup>. 일반적인 partial shape를 이용한 알고리즘이 병렬처리가 아니라는 점을 감안할 때 실용화적인 측면에서 커다란 장점이 된다. 또한 R-table을 자동화시킬 수 있는 장점이<sup>[13]</sup> 있어 인식할 물체의 자동피쳐추출방법이 가능하다. 이러한 GHT의 특징으로 볼 때 실제적인 비전 시스템으로의 응용에 대한 큰 장점을 가지고 있으며 robust한 비전 시스템 구축에 용이하다고 판단된다. 이에 본 논문은 GHT의 실제 물체인식에 관련된 여러 문제점들을 해결하고 실제 비전 시스템<sup>[14]</sup>으로 동작시키고자 하는데 있다.

첫째, 물체의 정보를 가지고 있는 R-table 작성시 물체의 모든 경계점으로부터 작성하는 기존의 방식을 사용하지 않고 물체의 특징을 잘 나타낼 수 있는 부분에 대한 정보를 중점적으로 추출하는 방법을 사용하였다. 특히 물체의 경계부분중 곡선부분을 가장 많이 포함하고 있는 부분을 추출하였다. 이렇게 할 경우 GHT의 단점으로 지적되는 직선부분의 비효율성도 개선시킬 수 있으며 GHT의 수행속도 또한 크게 향상시킬 수 있다.

둘째, GHT 수행시 가장 핵심적인 요소가 명암변화 방향정보이다. 명암변화방향정보를 얼마나 정확하게 계산하는가는 GHT를 이용하여 물체를 인식할 때 필수적이다. 그러나 물체영상은 디지털화되어 모니터에 화소로서 표현되며 우리가 이용하는 영상데이터도 이렇게 디지털화된 영상이므로 실제영상과는 오차가 있게 된다. 또한 실제 비전용 카메라 입력에서 물체가 이차원 내에서 회전이 되면 카메라 센서와 프레임 그래버의 디지털 특성으로 인하여 물체의 종횡비(aspect ratio)가 상당히 바뀐다<sup>[15]</sup>. 그래서 같은 명암변화각(gradient direction)을 가진 두화소가 다른 명암변화각으로 계산될 수 있으므로 차이가 작은 명암변화각들은 군집화(clustering)알고리즘으로 같은 군으로 합쳐주었다.

셋째, 로봇트가 물체를 인식하고 물체를 집어 이동시키는 동작을 하기 위해서는 물체의 위치뿐만 아니라 회전 및 크기변화(scale variation)에 대한 정보도 필수적이다. 기존의 GHT 방식을 사용할 경우 사차원의 파라미터 영역을 사용하여야 된다. 본 논문은 이차원 배열만으로도 물체의 위치, 회전각 및 크기변화를 단계적으로 정보를 추출하여 인식하는 계층적 방법을 제안한다. 특히 scale token방법을 사용하여 카메라 줌의 변화에 의한 크기변화를 보정하는 방법을 제안한다.

II장에서는 전체적인 시스템 구성도에 대해서 기술하고, III장과 IV에서는 개선된 GHT 알고리즘에 대해서 기술하였으며, V장에서는 실험결과를 기술하였다.

## II. GHT를 이용한 비전 시스템의 구성

본 연구에서 제시한 시스템의 개략도는 그림 1과 같다. 우선 인식 및 위치 측정을 하고 싶은 물체들을 각각 입력시켜서 데이터베이스를 R-table 형태로 작성한다. 그리고 카메라를 통하여 테스트 할 물체영상이 입

력되던 그 영상에 포함된 물체 중 인식하고자 하는 물체의 R-table에 의해서 GHT를 수행하여 물체의 위치, 회전각 및 크기변화를 추출하여 물체를 인식한다.

위와 같은 GHT 비전 시스템은 크게 두 가지 과정으로 나뉘어진다. 첫 번째 과정은 모델물체영상에 대한 R-table을 작성하는 과정이다. 두 번째 과정은 작성된 R-table을 이용하여 카메라로부터 받아들이는 입력 영상에 포함된 물체를 인식하여 그 물체의 크기변화, 위치 및 회전각을 추정하는 과정이다. 여기서 첫 번째 과정을 학습모드(learning mode), 두 번째 과정을 운용모드(operating mode)라고 정의하자.

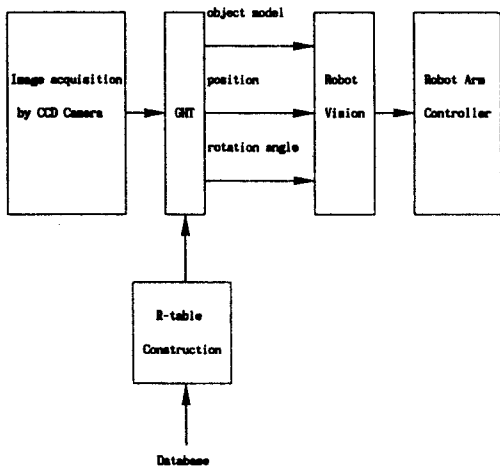


그림 1. GHT에 기반한 비전 시스템 개략도  
Fig. 1. Diagram of Vision System based on GHT.

### III. 학습모드

#### 1. Global R-table 과 Local R-table

학습모드는 인식하고자 하는 물체의 영상을 카메라로부터 받아들여 그 물체의 정보를 가지고 있는 R-table을 작성하는 과정이다. 이러한 R-table은 GHT 알고리즘 수행시 사용되어지며, GHT 알고리즘으로 물체를 인식하기 용이한 형태로 구성되어진다.

그림 2와 같이 디지털 영상에 곡선형태의 경계를 가지는 물체가 입력되었을 때, 먼저 적당한 위치에 참고점( $X_c, Y_c$ )을 택한다. 일반적으로 참고점은 그 물체의 무게중심을 선택한다. 한편 물체를 이루는 경계부분의 한 점( $X, Y$ )를 통과하는 수평선을  $a$ , 그 점에서의 접선의 법선성분 중 물체의 안쪽방향으로 향하는 성분을  $b$ 라할 때,  $a$ 와  $b$ 가 이루는 각을  $\phi$ (gradient direc-

tion)라 하고, 그 경계점에서 중심점에 이르는 벡터를  $\gamma$ 이라 정의하자. 벡터  $\gamma$ 은 점( $X, Y$ )와 중심점( $X_c, Y_c$ )와의 거리인  $r$ 과, 수평축  $a$ 와 벡터  $\gamma$ 이 이루는 각인  $\alpha$ 로 구성되어 진다. 이와 같은 방법으로 물체를 이루는 모든 경계점에 대해  $\phi, \gamma(r, \alpha)$ 값을 구한 후  $\gamma(r, \alpha)$ 값이  $\phi$ 값의 함수가 되도록  $\phi$ 값을 기준으로 분류하여 표 1과 같은 R-table을 작성한다. 이때 같은  $\phi$ 값을 가지는 경계점들의 벡터  $\gamma(r, \alpha)$ 은 같은  $\phi$ 군으로 모은다.

인식하고자 하는 물체의 모든 경계점들을 이용하여 R-table을 작성할 경우 두 가지의 중요한 문제점을 발생시킬 수 있다. 첫째, 물체의 직선부분은 허프영역에 작은 값들을 축적시키므로 물체인식에 유용한 정보가 되지 못한다. 둘째, 수평과 수직해상도의 차이는 물체의 크기변화를 유발하는데 이로 인해 물체를 오인식할 가능성이 커진다. 이러한 물체의 중형비를 보정해 주기 위하여 부가적인 계산이 필요하게 되고 이것은 물체인식 시간을 증가시키는 요인이 된다. 그러나 물체의 모든 경계점들을 이용하여 물체의 R-table을 작성하지 않고, 물체인식에 중요하다고 생각되는 부분적인 경계점들만으로 R-table을 작성할 경우, 부가적인 계산을 하지 않아도 위의 문제점들을 해결할 수 있다. 여기서 부분적인 경계점들을 물체의 경계선 중 곡선부분을 많이 포함하고 있는 부분으로 정했다. 기존의 방식대로 물체의 모든 경계점에서 R-table을 작성할 경우 Global R-table 이라고 부르고, 물체의 부분적인 경계점들만으로 R-table을 작성할 경우 Local R-table 이라고 부르기로 한다.

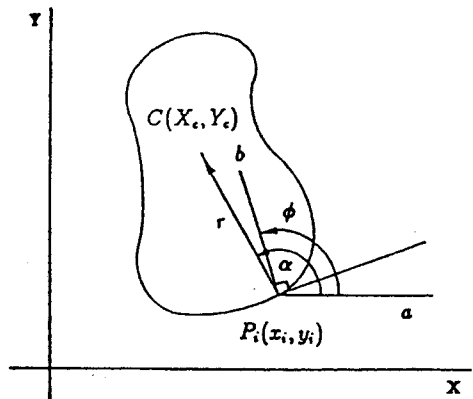


그림 2. R-table 작성을 위한 임의의 형태  
Fig. 2. Geometry used to form the R-table.

표 1. R-table  
Table 1. Formation of R-table.

$\phi$	$r = r \angle \alpha$
$\phi_1$	$r_1^1 \angle \alpha_1^1, r_2^1 \angle \alpha_2^1, \dots, r_{n_1}^1 \angle \alpha_{n_1}^1$
$\phi_2$	$r_1^2 \angle \alpha_1^2, r_2^2 \angle \alpha_2^2, \dots, r_{n_2}^2 \angle \alpha_{n_2}^2$
$\vdots$	$\vdots$
$\phi_m$	$r_1^m \angle \alpha_1^m, r_2^m \angle \alpha_2^m, \dots, r_{n_m}^m \angle \alpha_{n_m}^m$

where  $\phi$  : gradient direction angle.  
 $\gamma(r, \alpha)$  : vector from edge point to reference point.

2. Local R-table의 자동작성

Local R-table의 자동작성은 데이터베이스가 바뀔 경우 새로운 작성이 요구되므로 시스템의 융통성면에서 보아도 상당히 중요한 부분이라고 할 수 있다. 물체인식에 중요하다고 생각되는 부분적인 경계점들은 물체의 경계부분 중 곡선부분을 가장 많이 포함하고 있는 부분으로 정한다. 이렇게 하는 이유는 GHT 수행시 가장 중요한 요소는 명암변화방향정보인데, GHT 알고리즘 특성상 직선부분에서는 이 명암변화방향정보가 유용하지 못하다. 그러므로 곡선부분을 가장 많이 포함하는 부분이 GHT 수행시 유용한 정보를 가지고 있는 경우가 많다. 이러한 물체의 곡선부분은 곡률정보(curvature information)<sup>[16]</sup>를 이용하면 추출이 가능하다. 곡률은 물체의 위치변화, 회전, 크기변화에 무관하게 물체를 기술하는데 많이 사용되어져 왔다. 모든 물체의 경계점에서 곡률값을 계산하여 곡률의 변화가 가장 큰 부분을 추출해 낸다. 이렇게 할 경우 물체의 경계선 중에서 곡선부분을 가장 많이 포함하고 있는 부분이 된다. 그 과정은 다음과 같다.

- 과정 1. Local R-table을 작성할 영상을 카메라로부터 받아들여 이진화한다.
- 과정 2. 이진화된 영상에 Martelli가 사용한 경계추적 알고리즘<sup>[17]</sup>을 적용하여 물체의 경계부분을 추출한다.
- 과정 3. 다음의 수식을 물체의 경계선 좌표에 적용하여 곡률값을 계산한다.

$$K = \frac{x\ddot{y} - \dot{x}\ddot{x}}{(x^2 + y^2)^{3/2}} \quad (1)$$

과정 4. 위의 수식으로부터 계산된 물체의 경계선의 곡률값에 다음의 하이패스 필터 마스크를 씌워 하이패스 필터링을 한다.

-1	-1	0	1	1
----	----	---	---	---

- 과정 5. 하이패스 필터링된 값에 대하여 물체의 경계선부분의 40%씩을 순환적으로 더한다.
- 과정 6. 위의 과정을 거친 물체의 부분 중 가장 큰 값을 갖는 부분이 우리가 원하는 물체의 경계선의 일부가 된다.
- 과정 7. 이렇게 추출된 물체의 부분적인 경계점에서 물체의 무게중심을 구하여 참고점으로 정하고,  $\phi$ 와 벡터 $\gamma$ 의 값을 구하여  $\phi$ 를 군집화하여 Local R-table을 작성한다. 여기서  $\phi$ 값을 군집화 하는 이유는 영상이 디지털화 되었기 때문에 같은  $\phi$ 값을 가진 두 화소가 다른  $\phi$ 값으로 계산될 수 있으므로 차이가 작은  $\phi$ 들은 같은 군으로 합쳐준다. Agglomerative clustering 알고리즘<sup>[18]</sup>을 사용하여 차이가 작은  $\phi$ 들은 같은 군으로 뭉쳐 주었다. Agglomerative clustering 알고리즘은 다음과 같다. c를 우리가 원하는 군집의 갯수이고,  $\hat{C}$ 은 알고리즘 수행도중 변하는 군집의 갯수라 하자. 그리고  $X_i = \{ x_i \}$ ,  $i=1, \dots, n$ 는 n개의 샘플의 집합이라 하자.

- 과정 1.  $\hat{C} = n$ ,  $X_i = \{ x_i \}$ ,  $i=1, \dots, n$ . 라고 둔다.
- 과정 2.  $\hat{C} \leq c$  이면 끝낸다. 그렇지 않으면 아래의 과정들을 수행한다.
- 과정 3. 가장 가까운 두개의 군집  $x_i, x_j$ 를 찾는다.
- 과정 4.  $x_i$ 와  $x_j$ 를 하나의 군집으로 합치고  $x_j$ 를 삭제한다.  
 $\hat{C}$ 을 하나 감소시킨다.
- 과정 5. 과정 2.로 돌아간다.

여기서 가장 가까운 두개의 군집  $x_i, x_j$ 를 구하기 위하여 다음의 식을 사용하였다. 즉 두개의 군집의 거리 값이 가장 작은 군집을 하나의 군집으로 묶어준다.

$$dmin(X_i, X_j) = \underset{X \in X_i, X' \in X_j}{Min} \|X - X'\| \quad (2)$$

실제 우리가 사용한 Local R-table에는 물체의 경계 부분 중 정보를 많이 포함하고 있는 물체의 부분으로부터 작성된 표 1과 같은 형태의 테이블과 함께 크기 보정용 데이터가 포함된 형태로 이루어진다. 크기를 보정하는 방법에 대해서는 IV장에서 기술하겠다.

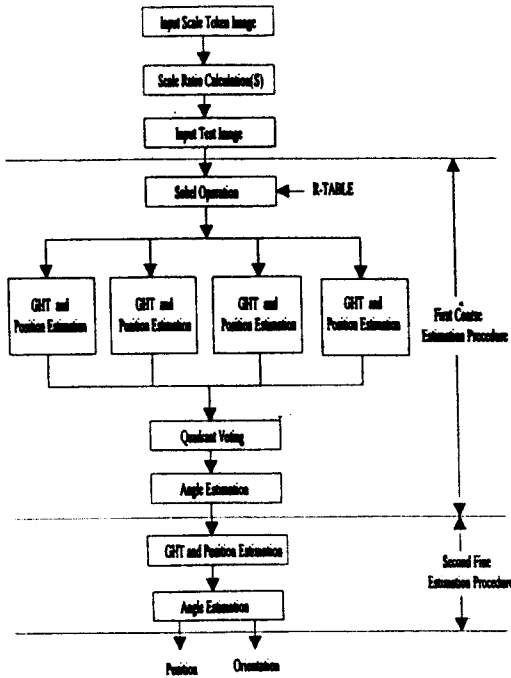


그림 3. GHT의 수행과정  
Fig. 3. Strategy for GHT execution.

#### IV. 운용모드

어떤 임의의 형태에 대해서 표 1과 같은 R-table이 만들어지면, GHT 알고리즘은 이 R-table을 사용하여 어떤 입력 영상에서 이 R-table과 일치되는 물체를 찾아내게 된다. 기존의 GHT 알고리즘으로 물체의 위치, 회전각 및 크기변화까지 인식하기 위해서는 사차원의 파라미터 영역이 필요하게 된다. 물체의 위치에 해당하는 이차원의 배열과 회전각과 크기변화에 해당되는 각각의 배열이 필요하게 된다. 이렇게 사차원의 배열을 사용할 경우 엄청난 메모리가 필요하게 되고 허프영역의 검색시간 또한 많이 걸리게 되므로 실제에 있어서는 거의 사용하기 어렵다. 하지만 본 논문은 크기(scale)가 변화되고, 회전된 물체에 대해서도 단지 이차원의 누적배열만으로 인식이 가능하게 했다. 먼저 크기를 보정하고 보정된 크기변화의 비를 이용하여 GHT

를 수행, 물체의 위치를 알아낸 뒤 물체의 위치를 이용하여 회전각을 인식한다. 이와 같이 계층적으로 정보를 추출하여 물체를 인식하는 개선된 GHT 알고리즘을 그림 3에 나타내었으며 그 과정을 하나씩 기술하겠다.

##### 1. 크기변화 보정

카메라로 직접 입력 받은 영상데이터는 카메라의 줌에 의해 쉽게 크기변화를 받기 쉽다. 이를 보정하기 위하여 학습이나 운영을 하기 전에 크기변화를 보정해 줄 물체의 영상을 받아들인다. 이때 입력되는 물체를 scale token이라 부르기로 하자. 이등변 삼각형형태의 물체를 scale token으로 사용했다. 사용된 scale token의 영상을 그림 4에 나타내었다. scale token은 학습이나 운영모드에서 한번씩만 입력하면 된다. 즉 학습모드에서는 두번의 영상입력을 필요로 하게 된다. 먼저 R-table을 작성하고자 하는 물체의 R-table 작성 당시의 크기정보를 구한 뒤 R-table에 기록해 둔다. 그 다음 학습하고자 하는 물체의 영상을 받아들여 R-table을 작성한다. 또한 운용모드에서도 두 번의 영상입력을 필요로 한다. 먼저 scale token의 영상을 입력하여 크기정보를 계산한 뒤 R-table에 기록해 둔 크기정보와의 비로서 간단하게 크기변화의 비를 계산한다. 그 다음 테스트할 영상을 받아들인다.

크기 정보는 입력된 scale token 영상에서 scale token이 차지하는 화소의 수로서 사용했다. 학습모드에서 구한 화소의 수를 R-table에 기록해 두고 운용모드를 시작할 때 구한 화소의 수와 비교하여 입력영상의 크기변화의 비를 간단하게 구할 수 있다. 학습모드를 시작할 때 구한 scale token의 화소의 수를 S1이라 하고, 운용모드를 시작할 때 구한 scale token의 화소의 수를 S2라 하면, 영상의 크기변환 비 S는 아래의 식과 같다.

$$S = \sqrt{\frac{S1}{S2}} \quad (3)$$

즉, 운용모드시 scale token의 영상을 받아들여 학습모드시의 scale token과의 비를 구하여 변형된 크기의 변화를 구하는 것이다. 이렇게 scale token으로서 사전에 필요한 크기정보를 추출하여 사용함으로써 허프 영역을 하나 줄일 수 있으며 어려운 스케일 문제를 간단하게 처리 할 수 있는 장점이 있다..

##### 2. GHT를 이용한 물체의 위치 인식

어떤 임의의 형태에 대해서 R-table이 만들어지면,

GHT 알고리즘은 이 R-table을 사용하여 어떤 입력 영상에서 이 R-table과 일치되는 물체를 찾아내게 된다. 만약에 물체가 회전하게 되면 물체가 회전한 만큼 R-table에 있는  $\phi$ ,  $\alpha$ 값도 회전되어야 영상에서의 그 물체가 R-table과 일치하게 된다. 즉 회전된 물체를 인식하고자 할 경우 가능한 회전각에 대해 R-table을 회전시켜 GHT 알고리즘을 수행시켜야 한다. 만약 가능한 모든 회전각에 대해 R-table을 수행할 경우 수행 시간이 굉장히 길어지게 된다. 이에 가능한 회전각인  $360^\circ$ 를 네개의 영역으로 나누어 각각의 영역에서 GHT를 수행하여 구한 물체의 위치 중 가장 큰값을 물체의 위치로 정한다. 그리고 GHT가 수행되는 각각의 영역에서는 R-table의  $\phi$ 와  $\alpha$ 를  $10^\circ$ 씩 각도를 증가시키며 GHT를 수행하고 각각의 허프평면을 겹치도록 하였다.

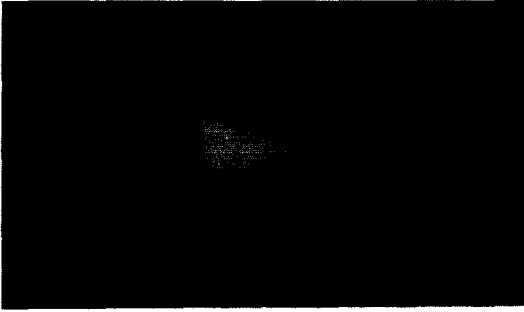


그림 4. scale token의 영상  
Fig. 4. Image of scale token.

가령 물체의 회전된 각이  $0^\circ \leq \theta < 90^\circ$  범위 안에 존재한다고 가정하면, R-table의  $\phi$ 와  $\alpha$ 를  $10^\circ$ 씩 각도를 증가시키면서 각각의 허프평면을 겹치면, 아홉개의 허프평면이 하나의 허프평면으로 겹쳐지게 된다. 여기서 누적배열의 최고값이 물체의 위치( $X_c, Y_c$ )가 된다. 물체의 회전각의 범위에 제한이 없는  $0^\circ$ 에서  $360^\circ$ 인 경우  $90^\circ$ 씩 네개의 영역으로 나누어 위의 과정을 되풀이 한다. 즉  $\theta$ 를  $0^\circ \leq \theta < 90^\circ$ ,  $90^\circ \leq \theta < 180^\circ$ ,  $180^\circ \leq \theta < 270^\circ$ ,  $270^\circ \leq \theta < 360^\circ$ 의 네개의 영역으로 나누어 GHT 알고리즘을 수행한다. 각각의 영역에서 최고값을 찾고, 그 중 가장 큰 값을 갖는 곳이 물체의 위치가 된다. 이렇게 네개의 영역으로 나누어 GHT를 수행하는 이유는 노이즈가 누적배열에 누적되어 오인식되는 것을 막기 위함이다.

사용되는 GHT 수식은 다음과 같다. ( $X_c, Y_c$ )는 찾은 자 하는 물체의 위치이고  $P_i(X_i, Y_i)$ 는 경계점이다

$$X_c = X_i + r \times S \times \cos(\alpha + \theta) \quad (4)$$

$$Y_c = Y_i + r \times S \times \sin(\alpha + \theta) \quad (5)$$

즉 입력영상에서의 모든 경계점  $P_i(X_i, Y_i)$ 에 대해서  $\theta$ 만큼 회전된 R-table의 해당  $\phi$ 란의 ( $r, \alpha$ )값을 다음의 두 수식에 대입하여 가능한 중심점을 찾는다. 여기서 S는 크기변화의 비를 나타낸다. 크기변화의 비는 물체의 위치를 구할때 단순히 곱해주면 된다. 즉 카메라줌에 의해 R-table을 작성할 때와 물체의 크기가 변화된 경우 크기변화의 비 만큼을 물체의 위치 추정시 보정해 주면 된다.

### 3. 회전각의 인식

이렇게 구한 물체의 위치로부터 물체의 회전각을 구하기 위해 사용되는 수식은 다음과 같다. 이 식은 식 (5)을 식 (4)로 나누고 정리하면 구할 수 있다.

$$\theta = \text{atan2} \left[ \frac{Y_c - Y_i}{X_c - X_i} \right] - \alpha \quad (6)$$

먼저 물체의 가능한 회전각에 해당되는 누적배열을 만든다. 이 누적배열의 크기는 물체의 회전각에 해당되는 0에서 360의 크기를 가지게 된다. 앞에서 구한 물체의 위치( $X_c, Y_c$ )를 식 (6)에 대입하고 입력영상에서의 각각의 경계점  $P_i(X_i, Y_i)$ 에 대해 R-table의 모든  $\alpha$ 값을 대입하면  $\theta$ 값을 구할 수 있는데 이때 구한  $\theta$ 에 해당되는 누적배열의 값을 "1" 증가시킨다. 위의 과정을 입력영상에서의 모든 경계점  $P_i(X_i, Y_i)$ 들에 대해서 반복한다. 이렇게 구한 누적배열의 최고값에서의  $\theta$ 값이 물체의 회전각  $\theta$ 가 된다. 이렇게 할 경우 GHT 수행시 회전각에 대한 누적배열을 사용하지 않아도 되므로 누적배열을 한차원 줄일 수 있게 된다.

### 4. 개선된 GHT 알고리즘

위에서 설명한 방법을 사용하게 되면 크기변화(S)와 회전각( $\theta$ )에 대한 누적배열을 사용하지 않아도 되므로 결국 전체적인 누적배열은 사차원에서 이차원으로 줄어들게 된다. 즉 누적배열은 기존의  $\text{Acc}(X_c, Y_c, S, \theta)$ 에서  $\text{Acc}(X_c, Y_c)$ 가 되게 된다. 그럼 GHT가 수행되는 전체적인 과정을 요약해 보면 다음과 같다. 여기서  $\phi$ 값은 Sobel 에지연산자로 구했으며 S는 크기변화의 비이다.

과정 1. scale token 영상으로부터 크기변화의 비(S)를 계산한다.

과정 2. 입력영상에서의 각 경계점( $P_i$ )에 대해서  $\phi(P_i)$  값을 계산한다.

과정 3.  $\theta$ 를  $0^\circ \leq \theta < 90^\circ$ ,  $90^\circ \leq \theta < 180^\circ$ ,  $180^\circ \leq \theta < 270^\circ$ ,  $270^\circ \leq \theta < 360^\circ$ 의 네개의 영역으로 나누어 각각의 영역에 대해서 다음의 과정을 수행하여 각각의 영역에서의 누적배열의 최고값의 위치를 찾는다.

과정 3.1 누적배열  $Acc(X_c, Y_c)$ 를 "0"으로 초기화 한다.

과정 3.2  $\theta$ 를  $10^\circ$ 씩 증가시키면서  $\theta$ 만큼 회전된 R-table을 구한다.

과정 3.3 입력영상에서의 모든 경계점  $P_i(X_i, Y_i)$ 에 대해서  $\theta$ 만큼 회전된 R-table의 해당  $\phi$ 란의  $(r, \alpha)$ 값을 다음의 두 수식에 대입하여 가능한 중심점을 찾는다.

$$X_c = X_i + r \times S \times \cos(\alpha + \theta) \quad (4)$$

$$Y_c = Y_i + r \times S \times \sin(\alpha + \theta) \quad (5)$$

과정 3.4 해당되는 누적배열의 값을 "1" 증가시킨다.

$$Acc(X_c, Y_c) = Acc(X_c, Y_c) + 1$$

과정 4. 네개의 각각의 영역에서의 누적배열의 최고값 중 가장 큰 값의 위치가 최종적인 물체의 위치가 된다.

과정 5. 위에서 구한 물체의 위치( $X_c, Y_c$ )로부터 물체의 회전각을 계산한다.

과정 5.1 물체의 회전각에 대한 누적배열을 "0"으로 초기화한다.

$$Acc[\theta] = 0, 0 \leq \theta \leq 360.$$

과정 5.2 입력영상에서의 각각의 경계점  $P_i(X_i, Y_i)$ 에 대해서 R-table의 모든  $\alpha$ 값을 다음의 수식에 대입하여 구한  $\theta$ 에 해당되는 누적배열을 "1"씩 증가시킨다.

$$\theta = \text{atan2} \left[ \frac{Y_c - Y_i}{X_c - X_i} \right] - \alpha \quad (6)$$

$$Acc[\theta] = Acc[\theta] + 1$$

여기서  $(X_i, Y_i)$ 는  $\phi$ 값을 구하기 위하여 Sobel 연산을 취한 테스트 영상에서 문턱치값 이상이 되는 점들이고  $\alpha$ 는 R-table에 있는 값이다.

과정 5.3 회전각에 대한 누적배열의 최고값에서의  $\theta$ 가 물체의 회전각이 된다.

그리고 이렇게 구한 물체의 위치와 회전각을 좀 더 정확하게 구하기 위하여 계산된 회전각을 중심으로  $+5^\circ$ 에서  $-5^\circ$ 의 범위에서  $1^\circ$ 씩 R-table을 회전시켜가며 GHT를 다시 수행하여 위치 및 회전각을 정밀하게 찾도록 하였다.

## V. 실험결과

### 1. 학습모드

학습모드에서는 두 번의 카메라 입력을 필요로 한다. 한 번은 카메라 줌에 따른 크기 변화를 보정해 주기 위해 사용되는 scale token 영상을 입력받아 면적당 화소의 수를 R-table에 저장한다. 즉 기존의 R-table에 그 물체의 R-table을 작성할 당시의 크기 값을 저장해 두는데 이를 위해서는 먼저 scale token 영상으로부터 크기값을 구해야 한다. 그 다음 R-table을 작성할 물체를 카메라로부터 입력 받아 Local R-table을 자동적으로 작성한다.



(a)



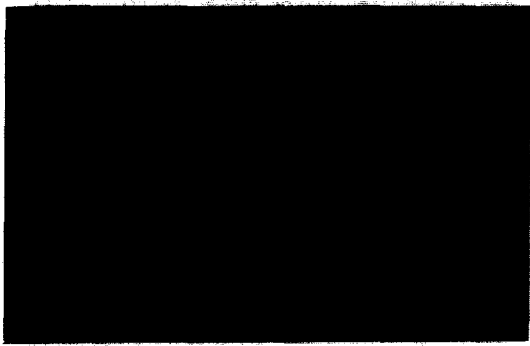
(b)

그림 5. R-table 작성을 위한 데이터베이스

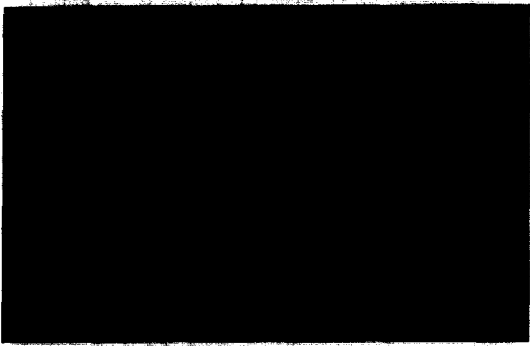
(a) object 1 (b) object 2

Fig. 5. Database for R-table construction.

(a) object 1 (b) object 2



(a)

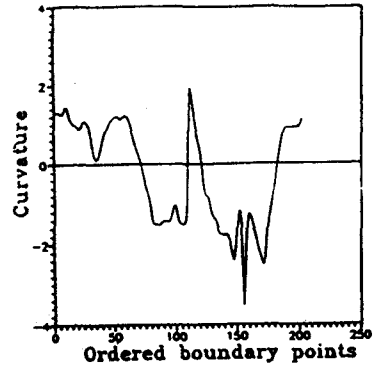


(b)

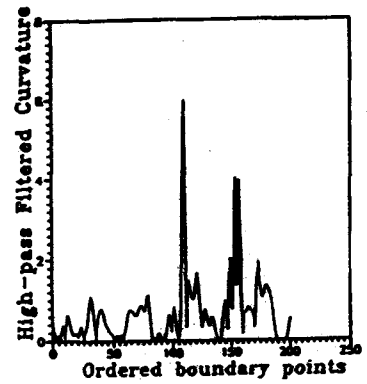
그림 6. 데이터베이스의 경계선을 추출한 결과  
(a) object 1의 경계선  
(b) object 2의 경계선

Fig. 6. Contour image of database.  
(a) Contour image of object 1  
(b) Contour image of object 2

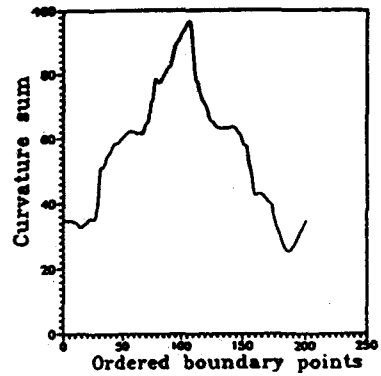
실험의 데이터베이스로는 기계부품을 사용했는데 이를 그림 5에 나타내었다. 그림 5에서 각각의 물체를 object 1 과 object 2 라고 하자. 각각의 물체의 그레이 영상을 이진화하여 경계추적 알고리즘으로 경계부분을 추출한다. 그림 6에 이렇게 구한 물체의 경계선 영상을 나타내었다. 이렇게 구한 물체의 경계선 부분 중에서 특징적인 부분을 추출해낸다. 그렇게 하기 위하여 물체의 각각의 경계선 좌표점에서 곡률값을 구한다. 이때 사용한 곡률 수식은 식 (1)에 나타내었다. 곡률값 계산시 사용한  $\sigma$ 값은 실험적으로 2 로 정하였다. 우리의 목적은 이 곡률값으로부터 곡률값의 변화가 가장 심한 부분을 추출하는 것인데 단순히 이 곡률값으로부터 그러한 부분을 추출하기 어려우므로 몇가지 전처리 과정을 이 곡률값에 적용한다. 먼저 하이패스 필터를 통과시킨다. 이렇게 하는 이유는 곡률값의 변화의 세기만을 찾기 위해서이다.



(a)



(b)



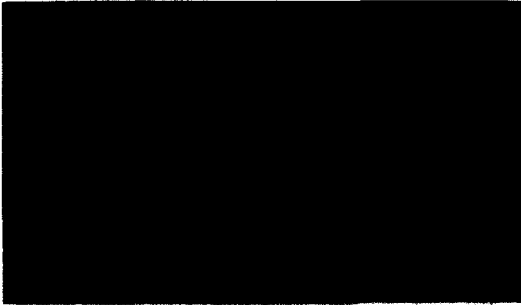
(c)

그림 7. object 2의 경계선에 여러가지 처리를 한 결과  
(a) object 2의 경계점에서의 curvature 값  
(b) curvature값에 High pass filter를 통과시킨 값  
(c) High pass filtering 된 값에 물체의 경계선 길이의 40%씩을 순환적으로 더한 값

Fig. 7. The result of various processing.  
(a) Curvature value of object 2  
(b) High passed curvature value of object 2  
(c) Curvature sum of object 2



그 다음 순환적으로 하이패스 필터링된 값을 더한다. 이 값 중 가장 큰 값을 갖는 영역이 우리가 찾고자 하는 부분이 된다. 그림 7에 object 2의 경계선 좌표 점에서 구한 곡률값과, 곡률값에 하이패스 필터를 통과한 값, 물체의 경계선 길이의 40%씩을 순환적으로 더해 준 값을 나타내었다. 최종적으로 추출된 Object 1, 2 각각의 물체에 대한 부분적인 모양을 그림 8에 나타내었다. 이렇게 구한 물체의 부분의 좌표와 무게중심으로부터 표 1과 같은 형태로 Local R-table을 작성한다. 여기에는 물체의 R-table 작성 당시의 크기값이 포함되게 된다. 이런 과정을 거쳐 찾고자 하는 물체의 정보를 Local R-table에 저장하는 학습모드가 이루어지게 된다.



(a)



(b)

그림 8. 최종적인 object 1,2의 부분적인 경계선

(a) object 1의 부분적인 경계선

(b) object 2의 부분적인 경계선

Fig. 8. Partial arc of database.

(a) Partial arc of object 1

(b) Partial arc of object 2

2. 운용모드

운용모드에서는 카메라로부터 받아들인 테스트 영상으로부터 원하는 물체를 GHT 알고리즘으로 인식하는 과정이다. 운용모드에서도 두 번의 카메라 입력을 필요로 한다. 먼저 scale token 영상을 받아들여 변환된

크기의 비를 구한다. 그러나 테스트 할 때마다 scale token의 영상을 받아들여 크기 변화의 비를 계산할 필요는 없다. 일단 한번만 scale token의 영상으로부터 테스트 할 당시의 크기값만 구한다면 그 다음부터는 그 값으로써 찾고자 하는 물체의 R-table에 있는 scale값과 비교하여 찾고자 하는 물체의 크기 변화비를 구하면 된다. 즉 테스트할 당시 처음에 한번만 하면 된다.

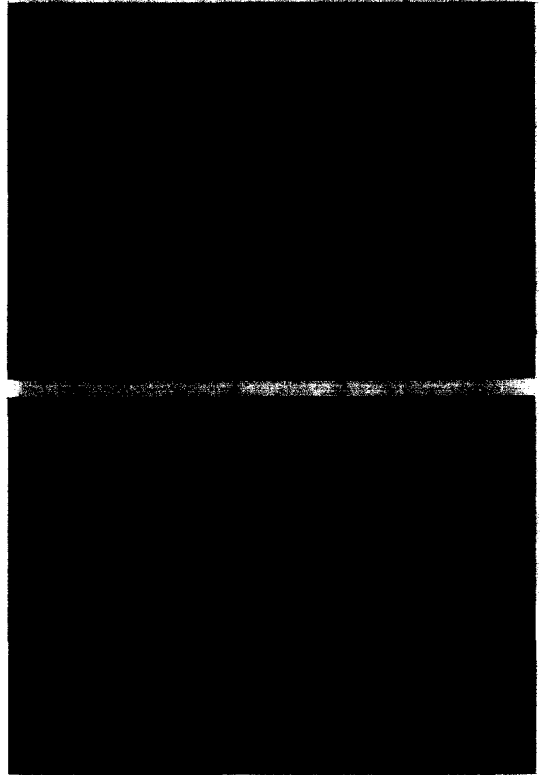


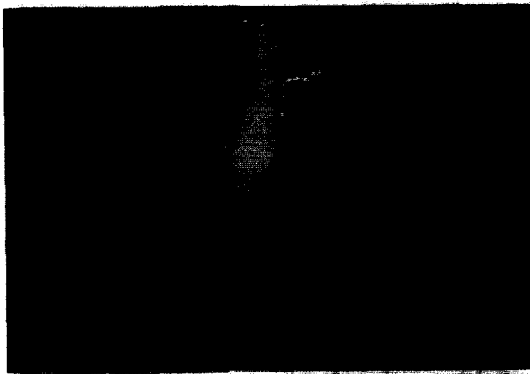
그림 9. Sebel 에지연산자로부터 구한 테스트영상의 경계선

Fig. 9. Edge image obtained from Sobel operation.

찾고자 하는 물체가 다른 물체와 섞여 있는 경우, 다른 물체와 겹쳐 있는 경우에 대해서 조사해 보았다. 본 실험에 사용된 데이터는 R-table 작성시 화소의 수를 1로 보았을때, 변환된 비율이 1.25 배인 경우이다.

GHT 알고리즘에서 물체의 위치와 회전각을 찾아내기 위해서는 명암변화방향정보인  $\phi$ 가 필요한데 이  $\phi$ 값은 Sobel 에지연산자를 사용해서 구한다. 이렇게 구한 경계선 영상을 몇개의 테스트 영상으로부터 구한 결과를 그림 9에 나타내었다. 그 다음 각각의 경계선에서의  $\phi$ 값과 식 (3)로부터 계산된 크기변화비로부터

GHT 알고리즘을 수행한다. GHT는 다른 물체인식 알고리즘과는 달리 영상분할이 필요 없고, 위와 같이  $\phi$ 값만 구하면 물체를 인식할 수 있는 장점이 있다.



(a)



(b)



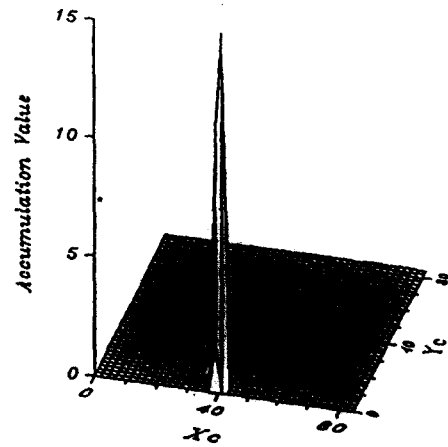
(c)

그림 10. 테스트 영상

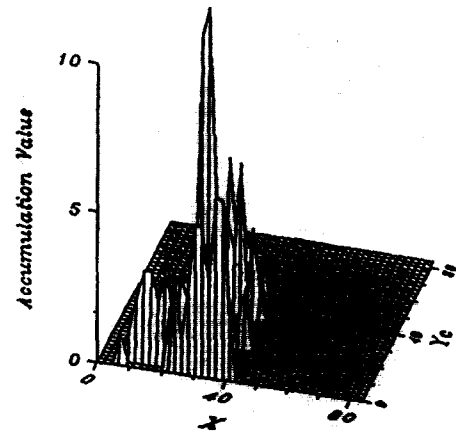
- (a), (b) 찾고자 하는 물체외에 다른 두개의 물체가 있는 경우
- (c) 찾고자 하는 물체가 겹쳐 있고 다른 물체도 있는 경우

Fig. 10. Test images.

- (a), (b) Complex images not occluded
- (c) Complex image with occlusion



(a)



(b)

그림 11. 누적배열의 삼차원 그림

- (a) 그림 10. (b)의 영상에 대한 누적배열의 삼차원 그림
- (b) 그림 10. (c)의 영상에 대한 누적배열의 삼차원 그림

Fig. 11. 3-D plot of Hough space.

- (a) Hough space of Fig. 10. (b)
- (b) Hough space of Fig. 10. (c)

그림 10에 테스트한 결과를 나타내었다. 인식결과를 원래의 영상에 겹쳐 흰 점선으로 표시했다. 그림 10의 (a)와 (b)는 object 1과 object 2가 두개의 다른 물체와 함께 있는 경우인데 쉽게 물체가 인식됨을 볼수 있다. 즉 GHT는 노이즈에 아주 강한 장점을 가지고 있음을 알수 있다. 이 경우에 다른 물체는 노이즈로 작용하는데 신호처리 관점에서 보면 상당히 낮은 SNR에

서도 우리가 사용한 GHT의 성능이 상당히 우수함을 알 수 있다. (c)는 object 2가 다른 물체와 겹쳐 있고 또 다른 두개의 물체가 있는 경우이다. 부분적으로 가려진 물체도 잘 인식할 수 있음을 볼 수 있는데, 이 경우 기존의 영상분할 방법으로는 원하는 물체를 분할하기가 상당히 어려운데 본 연구의 GHT는 영상분할 과정 없이도 복잡한 영상에서의 물체인식을 잘 행할 수 있음을 보이고 있다. 또한 Local R-table을 이용하여 GHT를 수행할 경우 계산시간도 상당히 줄어들음을 볼 수 있었다.

Hough 공간상에서 누적배열에 대한 삼차원 그림을 그림 11에 나타내었다. GHT 수행시에 얼마나 안정된 최고값을 찾는가는 중요한 문제중의 하나이다. 그림 11에서 (a)는 그림 10의 (b)에 대한 누적배열의 결과이고, 그림 11의 (b)는 그림 10의 (c)에 대한 결과이다. 그림 10의 (b)와 같이 여러 물체가 섞여 있고 찾고자 하는 물체가 이동되고 회전되어도 최고값이 아주 안정되어 있음을 볼 수 있다. 그림 10의 (c)에서는 찾고자 하는 물체가 다른 물체와 겹쳐 있고 여러 물체가 있는 경우이다. 이 경우 최고값 주위에 비교적 큰 값이 많이 존재하는 것을 볼 수 있다. 그러나 이러한 복잡한 경우에도 최대 피크가 비교적 안정하게 나타남을 볼 수 있다.

## VI. 결론 및 검토

전술한 계층적 방식에 의한 개선된 GHT에 대한 내용을 요약하면 다음과 같이 크게 두 부분으로 나눌 수 있다. 첫째, 모델물체의 정보를 추출하는 과정 즉 R-table을 작성하는 과정이다. GHT는 모델베이스 물체인식 알고리즘으로서, 모델물체의 정보를 R-table 형태로 저장한다. R-table 작성시 경계선 부분 중 곡선부분을 가장 많이 포함하는 부분을 추출한 Local R-table 방식을 사용하였고, 카메라 센서와 프레임 그래머의 디지털 특성으로 인한 명암변화방향성분값의 에러를 줄이기 위하여 차이가 작은 값들은 군집화하였다. 둘째는 작성된 R-table을 이용하여 GHT 알고리즘을 수행, 물체의 위치와 회전각을 인식하는 과정이다. 기존의 사차원의 배열을 사용하는 대신 이차원의 배열만으로 물체의 위치, 회전각 및 크기변화를 인식하는 과정을 보였다. 특히, scale token의 정보로부터 크기변화의 비를 구하는 방법을 제안하였다. 실험결과

계층적 방식에 의한 개선된 GHT는 실제 영상분석 시스템으로서 비교적 양호한 동작을 보임을 알 수 있었다. 실험에서 보인바와 같이 백그라운드 의 갯수가 2~3개 정도는 안정하게 동작하며 인식물체가 어느정도 유사하지 않는 경우 4~5개까지는 안정하게 동작한다. 특히 기존의 비전 알고리즘으로는 상당히 어려운 문제로 알려진 여러 물체가 동시에 존재하는 경우에도 각각의 물체를 분할하지 않고서도 인식이 가능하고, 물체가 겹쳐진 경우에도 좋은 성능을 보임을 알 수 있었다.

결론적으로 본 논문은 GHT를 실제적인 비전 시스템으로 동작시키기 위한 계층적 방식에 의한 개선된 GHT를 제안하였으며 실험결과 어느 정도의 복잡한 영상에서도 안정되게 동작함을 알 수 있었다. 또한 GHT는 본질적으로 병렬처리가 가능하므로 하드웨어의 구현이 용이하여 robust한 비전 시스템 구현에 대단한 이점이 될 수 있을 것이다. 한편 R-table작성시 곡률이 가장 큰부분을 가장 중요한 피쳐로 추출하였는데 데이터베이스가 결정되었을 경우 중요한 피쳐를 추출하는 방법에 대한 연구는 앞으로 해볼만한 가치가 있다고 본다.

## 참 고 문 헌

- [1] P. V. C. Hough, "Method and means for recognizing complex patterns", *U.S. Patent* 3,069,654, 1962.
- [2] R. O. Duda and P. E. Hart, "Use of Hough transformation to detect line and curves in pictures", *Commun. Ass. Comput. March.*, Vol. 15, Jan., 1972.
- [3] P. M. Merlin and D. J. Farber, "A parallel mechanism for detecting curves in pictures", *IEEE Trans. Comput.* Vol. C24, 96-98(1975).
- [4] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes", *Pattern Recognition*, Vol. 13, No. 2, pp.111-122, 1981.
- [5] D. H. Ballard and C. M. Brown, *Computer Vision*, Prentice-Hall., Englewood Cliffs, New Jersey, 1982.
- [6] E. Davies, "A new framework for an-

- alysing the properties of the generalized Hough transform", *Pattern Recognition Letter* vol. 6, June, 1987.
- [7] E. Davies, "Application of the generalized Hough transform to corner detection", *IEEE Proceedings* vol. 135, Pt. E., No. 1, Jan., 1988.
- [8] E. R. Davies, "A new parametrisation of the straight line and its application for the optimal detection of objects with straight edges", *Pattern Recogn. Lett.*, pp. 9-14, June, 1987.
- [9] 김수항, 임승민, 이규태, "효율적 패턴인식을 위한 순차적 GHT", 대한전자공학회, 제 28 권 5호, pp 10-17, 1991년 5월
- [10] M. Koch and R. Kasap, "Using polygons to recognize and locate partially occluded objects", *IEEE Trans. on PAMI*, Vol. 9, No.4, July, 1987.
- [11] N. Ansari and E. Delp, "Partial shape recognition : A landmark-based approach", *IEEE Trans. on PAMI*, Vol. 12, No. 5, May, 1990.
- [12] O. S. Chae, "Specialized parallel structures for VLSI implementation of the Hough transform for arbitrary shape detection", *Ph.D. Dissertation*, Oklahoma State University, July, 1986.
- [13] 박진수, 김종우, 김대영, 진성일, "R-table 자동인식 및 물체인식을 위한 GHT," 추계종합 학술대회 논문집 제13권 제2호 pp 549-553, 1990년 11월
- [14] B. Klaus and P. Horn, *Robot Vision*, the MIT press, Cambridge, 1986.
- [15] R. Lenz and R. Tsai, "Techniques for calibration of scale factor and image center for high accuracy 3-D machine vision metrology", *IEEE Trans. on PAMI*, Vol. 10, No. 5, September, 1989.
- [16] Farzin Mokhtarian and Alan Mackworth, "Scale-based description and recognition of planar curves and two-dimensional shapes", *IEEE Trans. on PAMI*, Vol. 8, No. 1, January, 1986.
- [17] A. Martelli, "An application of heuristic search methods to edge and contour detection", *Communication of the ACM*, Vol. 19, No 2, pp.73-83, February, 1976.
- [18] R. O. Duda & P. E. Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, 1973.
- [19] R. J. Schilling, *Fundamentals of Robotics*, Prentice-Hall., Englewood Cliffs, New Jersey, 1990.

저 자 소 개



具本哲(正會員)  
1968년 4월 21일생. 1991년 2월 경북대학교 전자공학과 학사 졸업. 1994년 2월 경북대학교 전자공학과 석사 졸업. 1994년 1월 ~ 현재 LG전자(주) 연구원. 주관심분야는 패턴인식, 로

봇비전, 이동통신

朴鎭秀(正會員)  
1990년 2월 경북대학교 전자공학과 학사 졸업. 1992년 2월 경북대학교 전자공학과 석사 졸업. 1992년 1월 ~ 현재 한국통신 연구개발단 연구원

秦成一(正會員) 第32卷 第B編 第9號 參照  
현재 경북대학교 전자공학과 부교수