

# 투사영상 불변량을 이용한 장애물 검지 및 자기 위치 인식

## Obstacle Detection and Self-Localization without Camera Calibration using Projective Invariants

노경식, 이왕현, 이준웅, 권인소

(Kyoung-Sig Roh, Wang-Heun Lee, Jun-Woong Lee, and In-So Kweon)

**Abstract :** In this paper, we propose visual-based self-localization and obstacle detection algorithms for indoor mobile robots. The algorithms do not require calibration, and can be worked with only single image by using the projective invariant relationship between natural landmarks. We predefine a risk zone without obstacles for a robot, and update the image of the risk zone, which will be used to detect obstacles inside the zone by comparing the averaging image with the current image of a new risk zone. The positions of the robot and the obstacles are determined by relative positioning. The method does not require the prior information for positioning robot. The robustness and feasibility of our algorithms have been demonstrated through experiments in hallway environments.

**Keyword :** self-localization, projective invariant, obstacle detection

### I. 서론

실내 주행 로봇이 주어진 작업 처리를 자동화하는데 있어서 자기 위치 인식과 장애물 검지는 매우 중요한 능력이다. 이를 위한 다양한 연구가 진행되어 왔는데, 그 중 비전을 이용한 방법이 그 타당성과 단순성 때문에 여러 장점을 보여주고 있으며 많은 성공 사례가 보고되고 있다[6][8]. 이 방법 중 우리는 한 장의 영상을 이용한 것에 관심을 두고, 실내 주행 로봇의 기하학적 모델은 있다고 가정하였다.

Dulimata[8]는 천장 조명과 문의 번호 판을 이용한 자기 위치 인식 능력을 갖춘 실내 주행 로봇 시스템을 제안하였다. 그리고 향후 연구로써 비전을 이용하는 방법과 효율적 표지판 이용을 제안하였다. Matsumoto[7]는 장애물 검지 뿐 아니라 자기 위치 인식과 조정각(steering angle) 결정을 동시에 할 수 있는 VSRR(View-Sequenced Route Representation)이라는 통로의 새로운 시각적 표현 방법을 제안하였다. Kosaka와 Kak[6]은 모델 베이스 추론법(model-based reasoning)과 불확실성의 예측을 이용한 비전에 바탕을 둔 방법을 제안하였다. 주위 환경의 3차원 CAD모델과 운동 불확실성 예측을 이용하여 자기 위치 인식을 수행하였는데 칼만 필터의 개선을 통해 정확성을 기했다. 대부분의 비전에 기반한 연구들은 내부 센서나 기존의 위치 정보와 로봇의 운동을 통해 로봇의 위치는 미리 알려져 있다고 가정한다. 또한 정밀한 위치 인식과 현재 자세와 예측된 자세와의 에러를 최소화 하는데 비전이 사용되고 있다. 하지만, 종종 필요한 정보를 구할 수 없는 경우가 발생하는데 이 경우는 기존의 방법은 비효율적이고 적용하기가 힘들다. 게다가 대

부분 작업 이전에 카메라의 캘리브레이션과 고정된 카메라 설치를 필요로 한다. 예를 들어, Matsumoto는 통로 모델로써 VSRR을 구축하고, 이를 템플릿 매칭을 하여 인식하는데 이용하였다. 하지만, 만약 카메라의 설치가 바뀌면 새로운 VSRR을 필요로 한다. Kosaka와 Kak의 방법은 CAD모델로부터 기대치 지도(expectation map)을 생성하기 위해 카메라의 매개 변수를 필요로 한다.

이상과 같은 여러 문제점을 해결하고자 본 논문에서는 투사 영상 불변량을 이용하여 어떠한 카메라 정보나 사전 지식도 필요치 않는 효율적 자기 위치 인식 알고리즘을 제안하였다. 장애물 인식은 변화 검지법을 이용하고 검지된 장애물의 위치는 상대적 위치 선정법(relative positioning)에 의해 결정된다. 조명과 노이즈 효과를 줄이기 위해 항상 생신되는 기준 위험 지역을 미리 정의하였다. 통로 혹은 비슷한 실내 환경에서의 제안된 주행 알고리즘은 두 가지 가정을 하고 있다. 바닥은 평평하고 바닥과 옆의 벽이 만나는 두개의 평행한 직선이 발생한다는 것이다. 또한 영상과 모델간의 매칭을 위한 주위 환경 지도 데이터베이스는 주어져 있다고 가정한다. 바닥과 문의 수직선들의 교점이 투사 영상 불변량을 계산하는데 특징점으로 쓰인다. 오프라인으로 각 교점들로 구성된 불변량을 모델 베이스로 미리 구축하는데 이것은 모델과 영상간의 대응점을 찾는데 사용되고, 이 대응점이 주행 로봇과 위험 지역 안에 들어오는 장애물의 위치를 계산하는데 사용된다. 실내 통로에서의 실험을 통해 제안된 알고리즘의 강인성과 타당성을 증명하였다.

### II. 이론

#### 1. 투사 불변량과 상대적 위치 선정법

투사 영상 불변량은 컴퓨터 비전에서 중요한 역할을 해 왔으며, 많은 적용 사례가 제안되었다. 본 연구에서는 자기 위치 인식을 위해 일반적 복도 환경하에서 보이는 지형물간의 투사 영상 불변량을 이용하였다. 본 장에서는 기본적 불변량 이론에 대해 설명하겠다. 자세한 것은

접수일자 : 1998. 6. 19., 수정완료 : 1998. 12. 24.

노경식 : 삼성종합기술원

이왕현 : 한국과학기술원

이준웅 : 기아자동차(주)

권인소 : 한국과학기술원 전기 및 전자공학과

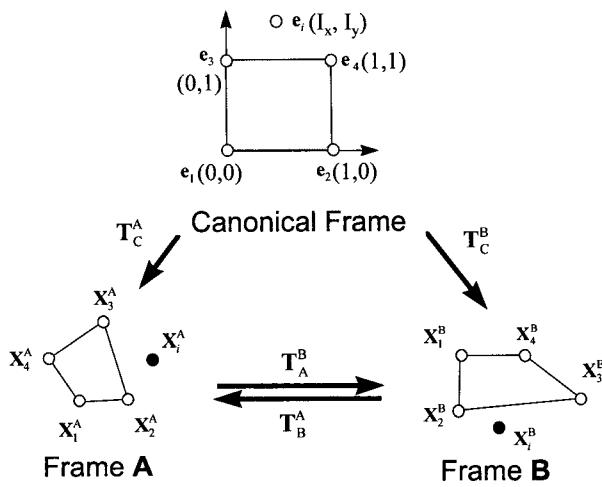


그림 1. 투사 변환 관계.

Fig. 1. Projective invariant relationship.

[1][2][4][5][10]에 잘 나타나 있다. 그림 1은 표준 좌표계(canonical coordinate)에서의 투사 영상 불변량을 보여주고 있다.

이 그림에서  $\mathbf{X}_A^i$ 와  $\mathbf{X}_B^i$ ,  $i = 1 - 4$ 는 A, B 두 좌표계에서의 4쌍의 대응점을 나타내고 있으며,  $\mathbf{e}_i$ ,  $i = 1 - 4$ 는 표준 좌표계에서의 대응점을 나타내고 있다. 4점들에 대한 다른 점들의 표준 좌표값은 다음과 같은 투사 영상 불변량으로 계산되어진다[10].

$$\begin{aligned} I_x &= \frac{\det(\mathbf{X}_1^A \mathbf{X}_2^A \mathbf{X}_3^A) \det(\mathbf{X}_1^A \mathbf{X}_4^A \mathbf{X}_2^A)}{\det(\mathbf{X}_1^A \mathbf{X}_1^A \mathbf{X}_2^A) \det(\mathbf{X}_4^A \mathbf{X}_2^A \mathbf{X}_3^A)} = \frac{\det(\mathbf{X}_1^B \mathbf{X}_2^B \mathbf{X}_3^B) \det(\mathbf{X}_1^B \mathbf{X}_4^B \mathbf{X}_2^B)}{\det(\mathbf{X}_1^B \mathbf{X}_1^B \mathbf{X}_2^B) \det(\mathbf{X}_4^B \mathbf{X}_2^B \mathbf{X}_3^B)} \\ I_y &= \frac{\det(\mathbf{X}_1^A \mathbf{X}_2^A \mathbf{X}_4^A) \det(\mathbf{X}_1^A \mathbf{X}_3^A \mathbf{X}_2^A)}{\det(\mathbf{X}_1^A \mathbf{X}_1^A \mathbf{X}_2^A) \det(\mathbf{X}_4^A \mathbf{X}_3^A \mathbf{X}_1^A)} = \frac{\det(\mathbf{X}_1^B \mathbf{X}_2^B \mathbf{X}_4^B) \det(\mathbf{X}_1^B \mathbf{X}_3^B \mathbf{X}_2^B)}{\det(\mathbf{X}_1^B \mathbf{X}_1^B \mathbf{X}_2^B) \det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_1^B)} \end{aligned}$$

$$\det(\mathbf{X}_\alpha \mathbf{X}_\beta \mathbf{X}_\gamma) = \begin{vmatrix} x_\alpha & x_\beta & x_\gamma \\ y_\alpha & y_\beta & y_\gamma \\ 1 & 1 & 1 \end{vmatrix} \quad (1)$$

여기서,

파셀 오차에 의한 (1)의 불변량은 측정 오차를 갖는다. 이 불변량의 오차를 계산하기 위하여 핵심 오차를 다음 식과 같이 표현한다.

$$\tilde{x}_i = x_i + \xi_i, \quad \tilde{y}_i = y_i + \eta_i \quad (2)$$

여기서,  $\xi_i, \eta_i$ 은 노이즈로써, 독립적으로 분포된 평균이 "0"이고 분산이  $\sigma_i^2$ 인 가우시안 노이즈로 가정할 때, 다음과 같은 통계적 성질을 갖는다.

$$\begin{aligned} E[\xi_i] &= E[\eta_i] = 0 \\ V[\xi_i] &= V[\eta_i] = \sigma_i^2 \\ E[\xi_i \xi_j] &= \begin{cases} \sigma_0^2 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \\ E[\eta_i \eta_j] &= \begin{cases} \sigma_0^2 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \\ E[\xi_i \eta_j] &= 0. \end{aligned} \quad (3)$$

핵심 오차에 의한 불변량은 다음 식과 같이 표현된다.

$$\tilde{\mathbf{I}} = \tilde{\mathbf{I}}(\tilde{x}_1, \tilde{y}_1, \tilde{x}_2, \tilde{y}_2, \tilde{x}_3, \tilde{y}_3, \tilde{x}_4, \tilde{y}_4, \tilde{x}_5, \tilde{y}_5) \quad (4)$$

이 불변량의 분산을 구하기 위하여, 각 점에 대하여 테일러 전개 하면 다음과 같이 된다.

$$\tilde{\mathbf{I}}_i \approx \mathbf{I}_i + \sum_{i=1}^5 [(\tilde{x}_i - x_i) \frac{\partial \tilde{\mathbf{I}}}{\partial \tilde{x}_i} + (\tilde{y}_i - y_i) \frac{\partial \tilde{\mathbf{I}}}{\partial \tilde{y}_i}] \quad (5)$$

(3)과 (5)로부터, 불변량에 대한 분산은 다음식과 같이 표현된다.

$$E[(\tilde{\mathbf{I}}_i - \mathbf{I}_i)^2] = \sigma_0^2 \sum_{i=1}^5 \left[ \left( \frac{\partial \tilde{\mathbf{I}}}{\partial \tilde{x}_i} \right)^2 + \left( \frac{\partial \tilde{\mathbf{I}}}{\partial \tilde{y}_i} \right)^2 \right] \equiv (\Delta \mathbf{I})^2. \quad (6)$$

A, B 두 좌표계에서의 4개의 대응점이 주어져 있고, 만약 A 좌표계에서의 i번째 점의 좌표값을 안다면, B 좌표계에서의 좌표값은 상대적 위치 선정법에 의해 결정된다. (1)을 바꿔 써보면 다음과 같다.

$$\begin{aligned} I_x &= \frac{\det(\mathbf{X}_4^B \mathbf{X}_2^B \mathbf{X}_3^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_2^B)} = \frac{\det(\mathbf{X}_i^B \mathbf{X}_2^B \mathbf{X}_3^B)}{\det(\mathbf{X}_i^B \mathbf{X}_1^B \mathbf{X}_2^B)} \\ &= \frac{X_i^B(Y_2^B - Y_3^B) - Y_i^B(X_2^B - X_3^B) + (X_2^B Y_3^B - X_3^B Y_2^B)}{X_i^B(Y_1^B - Y_2^B) - Y_i^B(X_1^B - X_2^B) + (X_1^B Y_2^B - X_2^B Y_1^B)} \\ I_y &= \frac{\det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_1^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_3^B)} = \frac{\det(\mathbf{X}_i^B \mathbf{X}_3^B \mathbf{X}_1^B)}{\det(\mathbf{X}_i^B \mathbf{X}_1^B \mathbf{X}_3^B)} \\ &= \frac{X_i^B(Y_3^B - Y_1^B) - Y_i^B(X_3^B - X_1^B) + (X_3^B Y_1^B - X_1^B Y_3^B)}{X_i^B(Y_1^B - Y_2^B) - Y_i^B(X_1^B - X_2^B) + (X_1^B Y_2^B - X_2^B Y_1^B)} \end{aligned} \quad (7)$$

또는,

$$\begin{aligned} aX_i^B + bY_i^B &= c \\ dX_i^B + eY_i^B &= f \end{aligned} \quad (8)$$

여기서, a, b, c, d, e, f는  $\mathbf{X}_k^A, \mathbf{X}_k^B, k = 1 - 4$ 과  $\mathbf{X}_i^A$ 의 함수이며, 다음과 같다.

$$\begin{aligned} a &= I_x \frac{\det(\mathbf{X}_4^B \mathbf{X}_2^B \mathbf{X}_3^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_2^B)} (Y_1^B - Y_2^B) - (Y_2^B - Y_3^B), \\ b &= -I_x \frac{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_2^B)}{\det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_2^B)} (X_1^B - X_2^B) + (X_2^B - X_3^B), \\ d &= I_y \frac{\det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_1^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_3^B)} (Y_1^B - Y_2^B) - (Y_3^B - Y_1^B), \\ e &= -I_y \frac{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_2^B)}{\det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_1^B)} (X_1^B - X_2^B) + (X_3^B - X_1^B), \\ c &= -I_x \frac{\det(\mathbf{X}_4^B \mathbf{X}_2^B \mathbf{X}_3^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_2^B)} (X_1^B Y_2^B - X_2^B Y_1^B) + (X_2^B Y_3^B - X_3^B Y_2^B), \\ f &= -I_y \frac{\det(\mathbf{X}_4^B \mathbf{X}_3^B \mathbf{X}_1^B)}{\det(\mathbf{X}_4^B \mathbf{X}_1^B \mathbf{X}_3^B)} (X_1^B Y_2^B - X_2^B Y_1^B) + (X_3^B Y_1^B - X_1^B Y_3^B). \end{aligned}$$

따라서, B 좌표계에서의 대응되는 i번째 좌표값은 다음과 같다.

$$\begin{aligned} X_i^B &= \frac{(ce - bf)}{(ae - bd)} = \mathbf{F}_x(\mathbf{X}_1^A, \mathbf{X}_2^A, \mathbf{X}_3^A, \mathbf{X}_4^A, \mathbf{X}_1^B, \mathbf{X}_2^B, \mathbf{X}_3^B, \mathbf{X}_4^B, \mathbf{X}_i^A) \\ Y_i^B &= \frac{(af - cd)}{(ae - bd)} = \mathbf{F}_y(\mathbf{X}_1^A, \mathbf{X}_2^A, \mathbf{X}_3^A, \mathbf{X}_4^A, \mathbf{X}_1^B, \mathbf{X}_2^B, \mathbf{X}_3^B, \mathbf{X}_4^B, \mathbf{X}_i^A) \end{aligned} \quad (9)$$

상대적 위치 관계식은 다음과 같이 표현된다.

$$\mathbf{F}_{xi} \equiv {}^A \mathbf{F}_{xi}^B, \mathbf{F}_{yi} \equiv {}^A \mathbf{F}_{yi}^B \text{ or } {}^A \mathbf{F}^B(\mathbf{X}^A) \equiv ({}^A \mathbf{F}_{xi}^B, {}^A \mathbf{F}_{yi}^B) \quad (10)$$

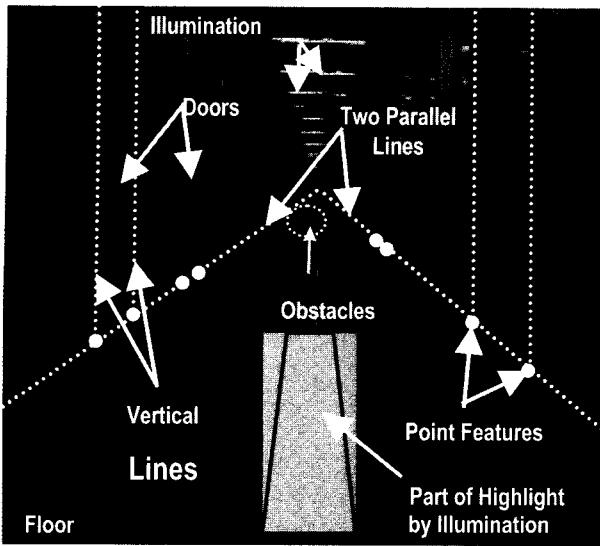


그림 2. 일반적인 복도 영상.

Fig. 2. Scene of the general hallway.

## 2. 자기 위치 인식 및 장애물 검지

본 장에서는 자기 위치 인식 및 장애물 검지를 자세히 설명하겠다. 그림 2는 일반적 실내 환경하에서 널리 관측되는 복도 영상을 보여주고 있다. 조명과 바닥에서의 반사는 복도를 따라 균일하게 분포되어 있다. 두 평행선의 교점은 소실점이라 불리는데 이는 투사 기하학(projective geometry)의 전형적인 특징 중의 하나이다. 그리고 문에 의한 수직선이 나타난다. 본 연구에서는, 이 두 평행선과 수직선의 교점을 특징점(landmarks)으로 사용하였다(그림 2에서 원으로 표시된 점들).

그림 3은 실내 주행 로봇의 주행을 위한 비전 시스템의 순서도를 보여주고 있다. 크게 모델 베이스 구축, 선처리 과정, 매칭, 자기 위치 인식 그리고 장애물 검지 등 5가지 단계가 있다. 모델 베이스 구축 단계에서는 불변량인 주변 환경 정보를 저장한다. 선처리 단계에서는 주어진 영상에서 소실점과 특징점, 문틀을 추출한다. 매칭 단계에서는 이 문틀에 대응하는 문틀을 찾아내고, 이 대응 관계와 상대적 위치 선정법에 의하여 자기 위치 인식을 한다. 마지막 단계에선 장애물의 검지와 자세를 결정한다. 강인한 장애물 검지를 위해선 미리 고정된 위험 지역을 설정하였다. 주로 에러는 선처리과정과 매칭 단계에서 일어난다.

### 1) 선처리 과정

복도 영상에서, 특징점은 두 평행선과 수직선들의 교점들이 된다. 두 평행선은 소실점에서 만나게 되는데, 본 연구에서 Hough 변환을 이용하여 두 평행선과 수직선을 추출한다. 점 특징점은 추출된 직선들의 교점을 구해 얻는데, 이들 중 문틀에 해당하는 기본 점들을 구하여 투사 영상 불변량을 계산한다. 일반적으로, 문은 거의 균일한 밝기값을 가진다. 즉, 수직 방향으로는 그리 변화가 없다. 따라서, 첫째 문틀을 기본 점으로 하여 불변량 계산시 에러를 줄였다.

### 2) 모델 베이스 구축과 매칭 과정

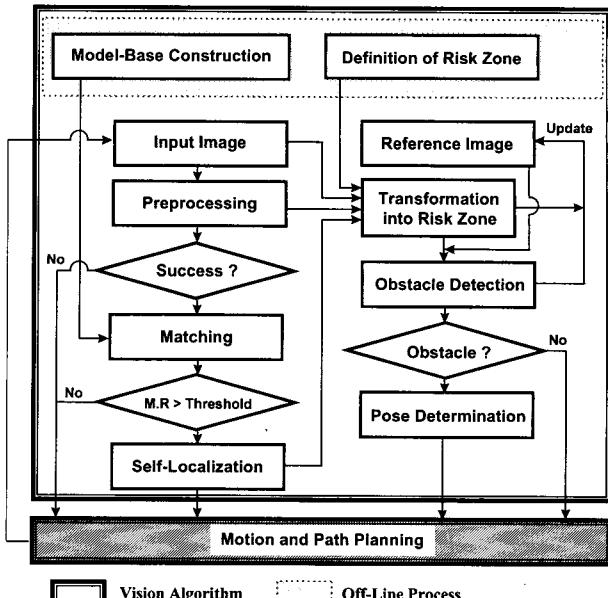


그림 3. 비전 알고리즘의 순서도.

Fig. 3. Flowchart algorithm for self-localization and obstacle detection.

본 절에서는 불변량을 이용한 데이터베이스 구축에 대해 설명하겠다. 효율적 모델 베이스 구축을 위하여 영상 불변량에 의해 indexing된 hash 테이블을 사용하였다. 불변량은 이차원 불변량으로써 한 평면위의 5점으로 계산된다. 매칭은 가정 발생과 검증 과정으로 구성되어 있다. 가정은 불변량이 가리키는 해쉬(hash) 테이블을 지적(indexing)함으로써 발생되며, 발생된 가정은 일치(alignment) 방법에 의해 검증된다.

그림 4는 전형적인 복도의 퍼염도를 보여주고 있다. 좌우 벽의 특징점은 마루와 문틀들 사이의 교점들로써,  $L_k$ 와  $R_k$ 로써 각각 표현된다.

모델 베이스 구축을 위한 pseudo 코드는 표 1과 같다. 표 1에서  $H(\cdot)$ 는 해쉬 함수로써 다음과 같다.

$$H(I) = ABC \quad (11)$$

여기서,

$$A = \begin{cases} l & : \text{the left feature} \\ r & : \text{the right feature} \end{cases},$$

$$B = \begin{cases} p & : \text{if } I > 0 \\ m & : \text{if } I < 0 \end{cases}, \quad \text{and}$$

$$C = \text{integer}(|I| \times 100) \quad (12)$$

또한, weight는 다음과 같이 정의된다.

$$\text{weight} = \frac{1}{\Delta I \sqrt{2\pi}} \exp\left(-\frac{(j - I)^2}{\Delta I^2}\right), \quad (I - 3\Delta I) \leq j \leq (I + 3\Delta I) \quad (13)$$

여기서,  $\Delta I$ 는 (6)으로부터 계산되며, 이 weight는 대응 점을 찾을 때, 일치 정도를 나타낸다. 모델에서의 대응 점을 찾기 위한 매칭 알고리즘은 표 2와 같다.

발생된 가정은 일치방법에 의하여 검증된다. 일치 방법은 널리 잘 알려진 검증 방법으로써, 발생된 가정에

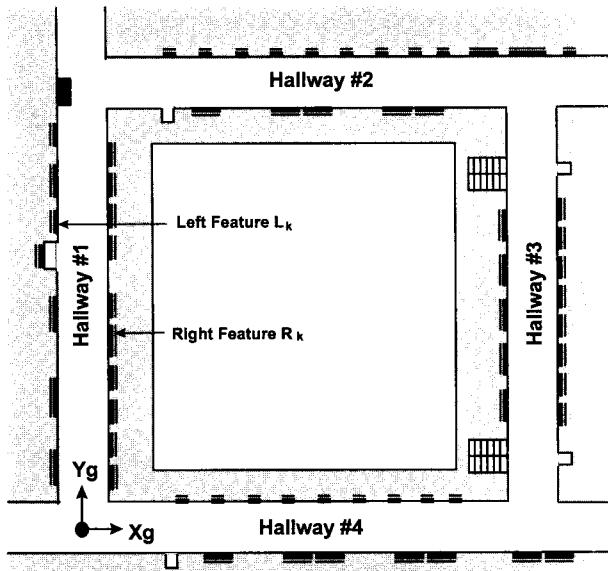


그림 4. 데이터베이스 구축을 위한 복도 영상 모델.  
Fig. 4 Top view of a hallway.

표 1. 모델 베이스 구축을 위한 Pseudo code.  
Table 1. Pseudo code for the model-base construction.

```

for Hallway Hi
for left feature Lk & Lk+1
  for right feature Rl & Rl+1
    ASSIGN Basis Points [ Lk : Rl : Lk+1 : Rl+1 ]
      for left feature m (except k & k+1)
        COMPUTE Invariant & Error Bound (I, DI)
        For j=I-3DI ~ I+3DI
        STORE (i, k, l, weight) at H(j)
      End for
    end for
    for right feature n (except l & l+1)
      COMPUTE Invariant & Error Bound (I, DI)
      For j=I-3DI ~ I+3DI
      STORE (i, k, l, weight) at H(j)
    End for
  end for end for end for

```

표 2. 대응점 탐색 알고리듬.  
Table 2. Pseudo code for the matching.

```

EXTRACT Features & Basis Features in an input image
ASSIGN Basis Points [ bl1 : br1 : bl2 : br2 ]
For left feature lfm (except Basis point))
COMPUTE Invariant
INDEXING & VOTING
end for
for right feature rfn (except Basis point)
COMPUTE Invariant
INDEXING & VOTING
end for
If Voting # > Threshold
Hypotheses Generation & Verification

```

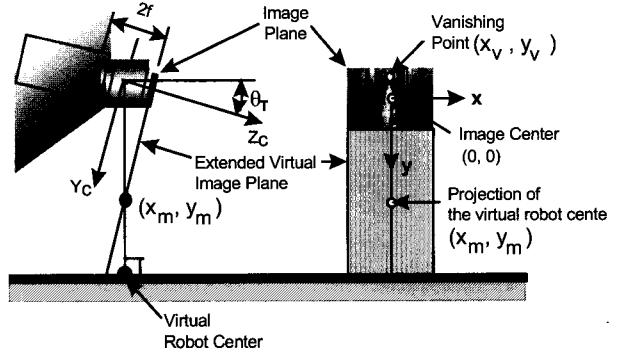


그림 5. 가상 로봇 중심의 정의.

Fig. 5. The virtual robot center and its projection onto the image plane.

대해 투사 변환을 계산하고, 이를 이용하여 모델의 다른 점들이 영상면으로 투사되고, 이것이 추출된 특징점과 비교되고, 마지막으로 최소 에러를 가지는 가정을 최적의 매칭으로 간주한다. 에러는 다음과 같이 정의된다.

$$\mathbf{M} \cdot \mathbf{R}(\tilde{\mathbf{x}}, \mathbf{x}) = \frac{1}{\sigma_p \sqrt{2\pi}} \exp\left(-\frac{-\|\tilde{\mathbf{x}} - \mathbf{x}\|^2}{2\sigma_p^2}\right) \quad (14)$$

여기서  $\tilde{\mathbf{x}}$ 와  $\mathbf{x}$ 는 투사된 점과  $\sigma_p$ 안에 들어오는 영상에서 추출된 점을 나타낸다. 그리고  $\sigma_p = 3\sqrt{\Delta I_x^2 + \Delta I_y^2}$  와  $\Delta I_x, \Delta I_y$ 은 (1)에서 정의되었으며, (6)에 의해 계산된다.

### 3) 자기 위치 인식

카메라의 중심(optical center)을 수직 방향으로 바닥면에 투사시켜 얻은 점을 가상의 로봇의 중심으로 가정한다. 예를 들어, 그림 5에서 가상의 로봇 중심을 영상면에 투사시켜 얻어진 영상점을 상대적 위치 선정법에 의해 자기 위치 인식에 사용된다.

그림 5에서, 가상 로봇 중심의 투사된 점  $(x_m, y_m)$  과 소실점  $(x_v, y_v)$  사이의 관계식을 다음과 같이 유도 할 수 있다.

$$f : s_y y_v = s_y y_m : f \quad (15)$$

여기서  $s_y$ 는 영상 좌표계를 metric 좌표계로 변환시키는 scale 인자이다. 따라서, 가상 로봇 중심의 영상면 위로의 투사점의 좌표는 다음과 같다.

$$y_m = \left( \frac{f}{s_y} \right)^2 \frac{1}{y_v} = f^2 \frac{1}{y_v} \quad (16)$$

추출된 영상면의 4점과 가상 로봇 중심에 대응하는 영상 점,  $(x_m^{lm}, y_m^{lm})$ 을 이용하여 로봇의 위치,  $(X_m^g, Y_m^g)$ 를 상대적 위치 선정법에 의하여 구할 수 있다. (10)에서, 만약 카메라의 광축이 바닥과 평행하다면,  $y_v^{lm}=0$ 이 된다. 실제 적용에 있어서는 fy가 아주 크기 때문에  $y_v^{lm}=0$ 이라고 가정한다. 이 경우에 불변량, Ix과 Iy 다음과 같다.

$$\begin{aligned} I_x &= \frac{\det(x_4 x_1 x_2) 0(y_2 - y_3) - \infty(x_2 - x_3) + (x_2 y_3 - x_3 y_2)}{\det(x_4 x_2 x_3) 0(y_1 - y_2) - \infty(x_1 - x_2) + (x_1 y_2 - x_2 y_1)} \\ I_y &= \frac{\det(x_4 x_1 x_2) 0(y_3 - y_1) - \infty(x_3 - x_1) + (x_3 y_1 - x_1 y_3)}{\det(x_4 x_3 x_1) 0(y_1 - y_2) - \infty(x_1 - x_2) + (x_1 y_2 - x_2 y_1)} \quad (17) \end{aligned}$$

$$\begin{aligned} I_x &\approx \frac{\det(x_4 x_1 x_2)(x_2 - x_3)}{\det(x_4 x_2 x_3)(x_1 - x_2)} \\ \text{혹은 } I_y &\approx \frac{\det(x_4 x_1 x_2)(x_3 - x_1)}{\det(x_4 x_3 x_1)(x_1 - x_2)} \quad (18) \end{aligned}$$

상대적 위치 결정에 관한 (5)에서, 만일  $\mathbf{A}$ 를 영상면에서 정의된 좌표계라 하고,  $\mathbf{B}$ 를 물체면에서 정의된 좌표계라 하면, 로봇의 위치는 다음식과 같다.

$$\begin{aligned} X_m &= \frac{(ce - bf)}{(ae - bd)} = F_x(x_1, x_2, x_3, x_4, X_1, X_2, X_3, X_4) \\ Y_m &= \frac{(af - cd)}{(ae - bd)} = F_y(x_1, x_2, x_3, x_4, X_1, X_2, X_3, X_4) \quad (19) \end{aligned}$$

(15)는 가상의 로봇 중심에 대응하는 영상 좌표값,  $(x_m^{\text{lm}}, y_m^{\text{lm}})$ 에 무관하게 계산할 수 있기 때문에, 케리브레이션 변수 없이도 로봇의 위치를 계산할 수 있다.

### 3. 장애물 검지 및 자세 결정

장애물을 검출하기 위하여 먼저 로봇의 주행 방향의 위험 지역(risk zone)을 구축한다. 그러나, 이것은 주행 로봇의 자세에 따라 변하므로, 고정된 기준 위험 지역의 설정이 필요하다. 그림 6은 이 기준 위험 지역을 보여주고 있다. 여기서 위험 지역 좌표계의 중심은 북도의 중앙에 위치한다.

그림 7은 각 좌표계 사이의 관계를 보여주고 있다. 여기서,  $\mathbf{x}^{\text{RIm}}$ 과  $\mathbf{X}^{\text{RW}}$ 은 각각 영상과 기준 좌표계에서의 위험 지역을 나타낸다. 그리고  $\mathbf{x}^{\text{lm}}$ 과  $\mathbf{X}^{\text{w}}$ 은 각각 현재 로

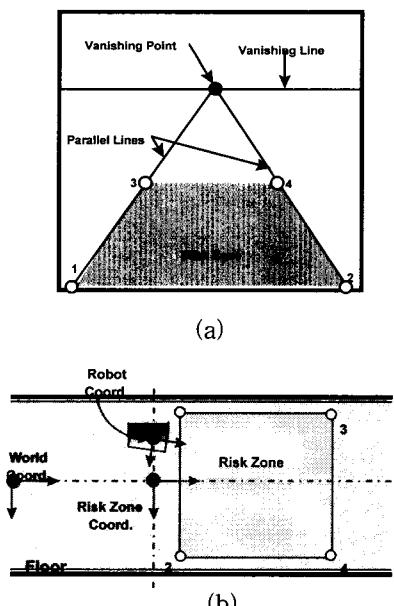


그림 6. 위험지역의 정의 (a) 영상에서의 위험지역, (b) 기준 좌표계에서의 위험지역.

Fig. 6. Definition of risk zone (a) risk zone in image, (b) risk zone in world.

봇의 위치를 나타낸다. (6)에서 정의된 함수는 각 좌표계에서의 상대적 위치 관계를 나타낸다.

함수  $({}^{\text{RIm}}\mathbf{F}_x^{\text{RW}}, {}^{\text{RIm}}\mathbf{F}_y^{\text{RW}})$ 은 혹은  $({}^{\text{RW}}\mathbf{F}_x^{\text{RIm}}, {}^{\text{RW}}\mathbf{F}_y^{\text{RIm}})$ 은 미리 정의된 위험 지역으로부터 얻어질 수 있다. 게다가,  $({}^{\text{lm}}\mathbf{F}_x^{\text{w}}, {}^{\text{lm}}\mathbf{F}_y^{\text{w}})$  또는  $({}^{\text{w}}\mathbf{F}_x^{\text{lm}}, {}^{\text{w}}\mathbf{F}_y^{\text{lm}})$ 은 매칭 알고리즘으로부터 얻어진다. 마지막으로, 관계식  $\mathbf{T}_{\text{RW}}^{\text{w}}$ 자기 위치 인식의 결과에 의해 얻어진다. 기준 위험 지역으로의 변환은 다음과 같다.

$$\mathbf{x}^{\text{lm}} = {}^{\text{w}}\mathbf{F}^{\text{lm}}(\mathbf{X}^{\text{w}}) = {}^{\text{w}}\mathbf{F}^{\text{lm}}(\mathbf{T}(\mathbf{X}^{\text{RW}})) \quad (20)$$

$$\mathbf{x}^{\text{RIm}} = {}^{\text{RW}}\mathbf{F}^{\text{RIm}}(\mathbf{X}^{\text{RW}}) \quad (21)$$

$$\mathbf{x}^{\text{lm}} \rightarrow \mathbf{x}^{\text{RIm}} \text{ or } {}^{\text{w}}\mathbf{F}^{\text{lm}}(\mathbf{T}(\mathbf{X}^{\text{RW}})) \rightarrow {}^{\text{RW}}\mathbf{F}^{\text{RIm}}(\mathbf{X}^{\text{RW}}) \quad (22)$$

장애물 검지는 변화 검지법에 기인한다.  $E_n$ 과  $E'_n$ 을 각각 현재 영상과 위험 지역으로의 변환된 영상의 밝기라고 하자. 그리고,  $E_n^{\text{RIm}}$ 은 n번째 기준 영상,  $\Delta E_n$ 은 다음과 같이 정의된 변화된 영상이라고 하자.

$$\Delta E_n = \begin{cases} HIGH & \text{if } |E_n^{\text{RIm}} - E_{n+1}'| > Threshold \\ LOW & \text{otherwise} \end{cases} \quad (23)$$

그러면  $(n+1)$ 번째 기준 영상은 다음식으로 얻어질 수 있다.

$$E_{n+1}^{\text{RIm}} = \frac{nE_n^{\text{RIm}} + M'_{n+1}}{n+1} \quad (24)$$

$$\text{여기서, } M'_{n+1} = \begin{cases} E_{n+1}' & \text{if } \Delta E_n = LOW \\ E_n^{\text{RIm}} & \text{if } \Delta E_n = HIGH \end{cases}$$

변화된 영상  $\Delta E'_n$ 에 대하여, 코너 검지법을 이용하여 장애물을 검지한다. 코너의 위치는 그림 7에서 나타난 각 좌표계 간의 관계식을 이용하여 구해진다.

$$\mathbf{X}^{\text{w}} = {}^{\text{RW}}\mathbf{F}^{\text{w}}(\mathbf{X}^{\text{RW}}) = {}^{\text{RW}}\mathbf{F}^{\text{w}}({}^{\text{RIm}}\mathbf{F}^{\text{w}}(\mathbf{x}^{\text{RIm}})) \quad (25)$$

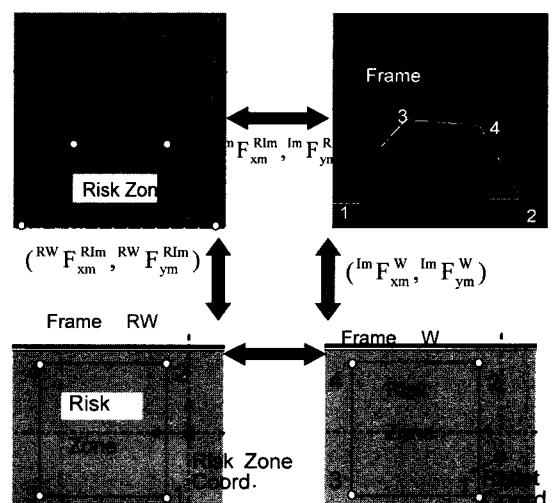


그림 7. 각 좌표계 간의 관계.

Fig. 7. The relationship between each coordinate system.

### III. 실험

실험은 복도 환경에서 행해졌으며, 586 컴퓨터를 이용하였고 입력 영상 크기는  $256 \times 240$ 이다.

#### 1. 선처리 과정

그림 8은 선처리한 결과를 보여주고 있다. 에지 검지는 Sobel 연산자를 이용하였고, 세선화 작업은 Gonzalez [9]를 이용하였다. 그 다음으로 Hough 변환에 의해, 두 평행선과 수직선들을 추출한다. 특징점들은 그림 8(d)에서 보는 바와 같이 이들 직선들의 교점이 된다. 마지막으로, 각 벽에 있는 문틀은 인접한 두 특징점 들간의 밝기값 분포로부터 결정된다. 문에 대응하는 영상영역들은 주변벽에 비해 어두운 밝기값과 균일한 밝기(낮은 분산)를 갖는다. 그림 8(e)는 위에서 기술한 관찰에 근거하여 얻어진 결과를 보여주고 있다.

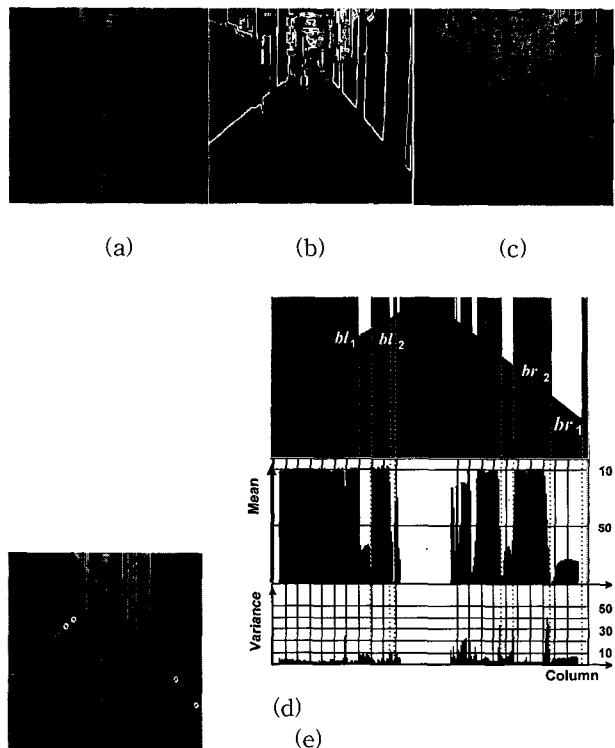


그림 8. 선처리 결과 (a) 입력 영상, (b) 에지 검지, (c) 세선화, (d) 특징점 추출, (e) 수직선(혹은 행)방향의 평균 및 분산.

Fig. 8. Results of Procedures of the image processing (a) input image, (b) edge detection, (c) thinning, (d) feature extraction, (e) mean and variance along the vertical.

#### 2. 매칭

추출된 특징점을 중 첫번째 문틀의 양쪽의 점들이 기준 특징점으로 선택된다(그림 8(e)의 br1, br2, bl1, bl2로 표시된 점들). 표 3은 추출된 특징점들의 영상 좌표값을 보여주고 있다. 여기서, bl과 br은 각각 왼쪽과 오른쪽의 기준점들이고, lf과 rf은 각각 왼쪽과 오른쪽의 특징 점들이다.

표 4는 4개의 기준점들에 대하여 각 특징점들의 영상

표 3. 추출된 특징점.

Table 3. Extracted features.

No	Image Coord.	No	Image Coord.	No	Image Coord.
1	$bl(76,101)$	6	$lf(103, 82)$	11	$rf(164,100)$
2	$br(247,183)$	7	$lf(105, 80)$	12	$rf(157, 93)$
3	$bl(86, 94)$	8	$rf(195,131)$	13	$rf(153, 89)$
4	$br(220,156)$	9	$rf(186,122)$	14	$rf(150, 86)$
5	$lf(100, 84)$	10	$rf(168,104)$	15	

표 4. 계산된 불변량.

Table 4. Computed invariant.

No.	Invariant	Index Value	No.	Invariant	Index Value
5	(0,0.72)	lp72	10	(0.15,1)	rp15
6	(0,0.78)	lp78	11	(0.13,1)	rp13
7	(0,0.82)	lp82	12	(0.09,1)	rp9
8	(0.38,1)	rp38	13	(0.08,1)	rp8
9	(0.28,1)	rp28	14	(0.07,1)	rp7

표 5. 가정 발생과 매칭률.

Table 5. Hypotheses and matching ratio of each hypothesis.

No.	Hypotheses	M.R
1	(1,0,2)	0.73
2	(1,2,2)	0.94

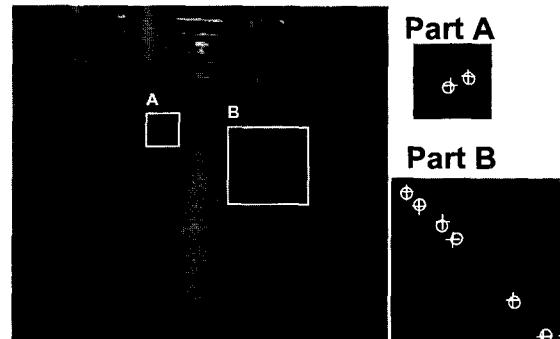


그림 9. 추출된 특징점과 역 투사로 얻어진 모델링 (○ : 특징점 + : 투사된 모델점).

Fig. 9. Results of verification (○ : point features + : projected model points).

불변량과 선처리과정의 결과로 얻어지는 대응하는 색인값들을 나타내고 있다.

표 5는 발생한 가정과 각 가정의 검증 결과를 보여주고 있다. 검증은 일치 방법과 (10)에 의해 매칭의 정도를 계산함으로써, 얻을 수 있다. 매칭의 결과 두 번째 가정이 첫 번째에 비하여 높은 값을 주었으며, 따라서 이 가정을 가장 근사한 가정으로 선택한다.

그림 9는 두번재 가정에 대하여 투사되고 추출된 특징점을 보여준다. 여기서 원은 주어진 영상으로부터

표 6. 자기 위치 인식 결과.

Table 6. Result of matching.

No	Image Coordinates ( $\mathbf{x}^{lm}$ )	World Coordinates ( $\mathbf{X}^w$ )
1	$Bl(76,101)$	(-103.5, 1151) cm
2	$Br(247,183)$	(103.5, 598) cm
3	$Bl(86, 94)$	(-103.5, 1318) cm
4	$Br(220,156)$	(103.5, 688) cm
i	Robot Position : $\mathbf{X}_m = {}^{lm}\mathbf{F}^w(\mathbf{x}_m)$	

표 7. 자기 위치 인식 결과.

Table 7. Result of a self-localization with the exact  $y_m^{lm}$ .

Computed Value	True Value
(-2.03, 367.50) cm	(0, 360) cm
$(\mathbf{x}_m^{lm}, y_m^{lm}) = (0, 1070950.469)$	

표 8.  $y_m^{lm} \approx \infty$  일 때의 자기 위치 인식 결과.Table 8. Result of a self-localization when  $y_m^{lm} \approx \infty$ .

Computed Value	True Value
(-2.14, 367.0) cm	(0, 360) cm
$(\mathbf{x}_m^{lm}, y_m^{lm}) = (0, \infty)$	

추출된 특징점이고, 십자는 매칭에 의해서 얻어진 투사된 모델점이다.

### 3. 자기 위치 인식

그림 8에 주어진 이미지를 이용하여 자기 위치 인식 알고리즘을 테스트하였다. 표6은 매칭된 결과를 나타내고 있다.

실험에서  $f = 16mm$ ,  $sy = 0.0013mm$ 인 카메라를 사용하였고 (12)를 이용하여  $(\mathbf{x}_m^{lm}, y_m^{lm}) = (0, 1070950.47)$ 을 얻었다. 추출된 영상면의 4점과 가상 로봇 중심의 영상 점,  $(\mathbf{x}_m^{lm}, y_m^{lm})$ 을 이용하여 상대적 위치 선정법에 의한 기준 좌표계에서의 로봇의 위치를 예측할 수 있다.

표 7은 제안된 방법에 의해서 얻어진 예상 결과이다.

표 8은  $y_m^{lm} \approx \infty$  일 때의 자기 위치 인식 결과를 보여준다.

### 4. 장애물 검지 및 자세 결정

표 9는 영상과 로봇 좌표계에서의 기준 위험 지역을 나타내는 점들을 보여준다. 여기서, H와 V는 영상의 수평 수직의 픽셀 수를 나타낸다.

그림 10은 장애물 검지 결과이다. 그림 10(a)와 (b)는 각각 입력 영상과 그 선처리 결과이다. 그림 10(c)와 (d)는 각각 원래 입력 영상의 위험 지역과, 매칭과 자기 위치 인식으로부터 얻어진 기준 위험 지역을 보여준다. 그림 13(e)는 현재 기준 위험 지역을 나타낸다. 그림 10(f)와 (g)에는 (17)에 의해 검지된 장애물을 보여주고 있다.

표 9. 영상과 로봇 좌표계에서의 기준점들의 좌표.

Table 9. The risk zone in the image and the global coordinates.

No.	Image Coordinates ( $\mathbf{x}_i^{Rlm}$ )	Robot Coordinates ( $\mathbf{X}_i^{RW}$ )
1	(0, V) pixel	(-70, 250) cm
2	(H, V) pixel	(70, 250) cm
3	(H/4, (V+ $y_v$ )/2) pixel	(-70, 640) cm
4	(3H/4, (V+ $y_v$ )/2) pixel	(70, 640) cm
i	$\mathbf{x}_i^{Rlm} = {}^{RW}\mathbf{F}^{Rlm}(\mathbf{X}_i^{RW})$ or $\mathbf{X}_i^{RW} = {}^{Rlm}\mathbf{F}^{RW}(\mathbf{x}_i^{Rlm})$	

표 10. 영상과 로봇 좌표값.

Table 10. Result of pose determination for the detected obstacle in an image sequence for hallway I.

True obstacle position A:(-40,1610), B:(-15,1610) unit : cm

	Robot 위치	True 위치	Robot 위치 Error	Point A	Point B	Error of Point A	Error of Point B
a)	-8,1030	0,1020	-8, 10	-37,1603	-17,1603	-3,7	2,7
b)	-10,1068	0,1060	-10, 8	-34,1587	-13,1587	-6,13	-2,13
c)	-3,1130	0,1140	-3,-10	-39,1615	-17,1615	-1,-5	2,-5
d)	0,1229	0,1220	0, 9	-42,1620	-20,1620	2,-10	5,-10
e)	9,1293	0,1300	-9, -7	-37,1618	-13,1618	-3,-8	-2,-8
f)	-7,1369	0,1380	-7,-11	-39,1630	-14,1630	-1,-20	-1,-20
Average Error	6,1,9	Average Error	2,7,10,5	2,3,10,5			

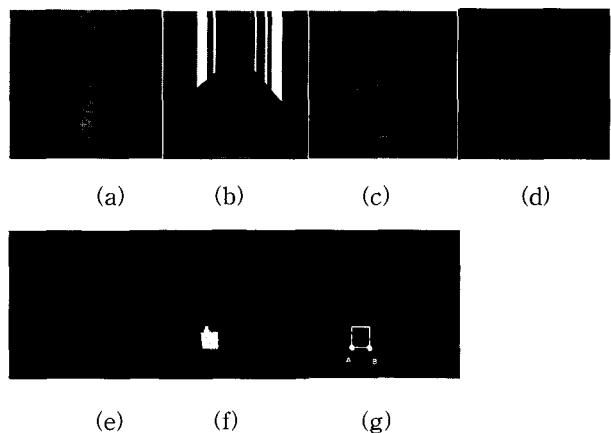


그림 10. 장애물 검지. (a) 입력영상, (b) 선처리과정, (c) 자기위치인식, (d) 영상 변환, (e) 기준 위험 지역, (f) 변화 검지, (g) 장애물 검지.

Fig. 10. The obstacle detection. (a) input, (b) image processing, (c) self-localization, (d) transformation, (e) reference risk zone, (f) change detection, (g) obstacle detection.

표 10은 그림 11의 각 영상 입력에 대하여, 자기 위치 인식의 결과들과 추출된 장애물 위치의 계산 결과를 표시하고 있다.

그림 11은 hallway I에서의 연속되는 영상에서의 장애물 검지 결과를 보여주고 있다.

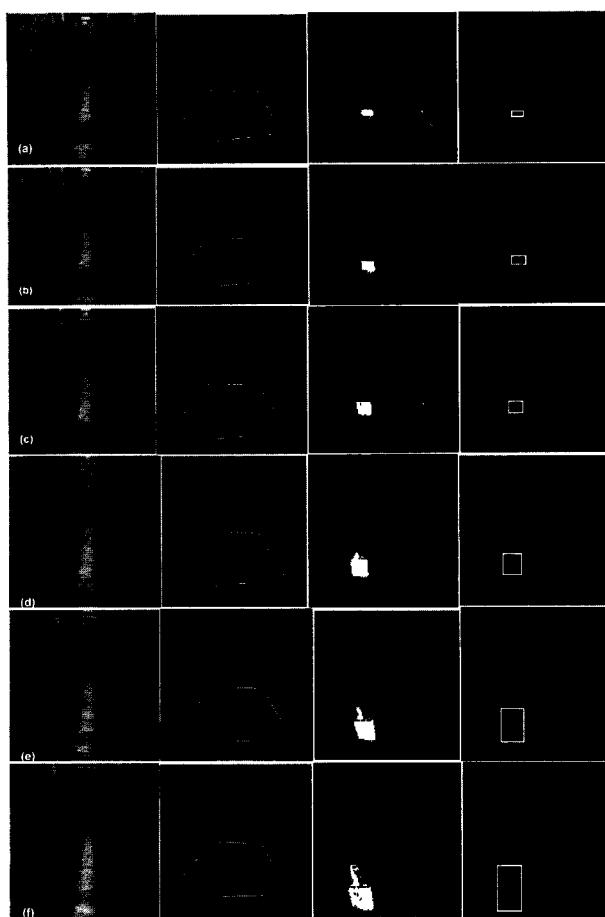


그림 11. 복도에서의 연속영상에서의 장애물 검지 결과.

Fig. 11. The result of obstacle detection for a sequence of images from Hallway I.

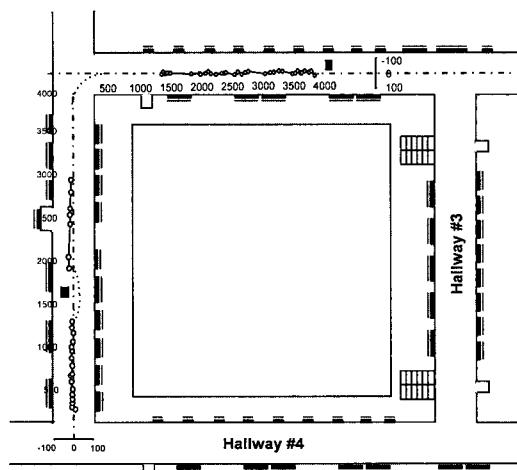
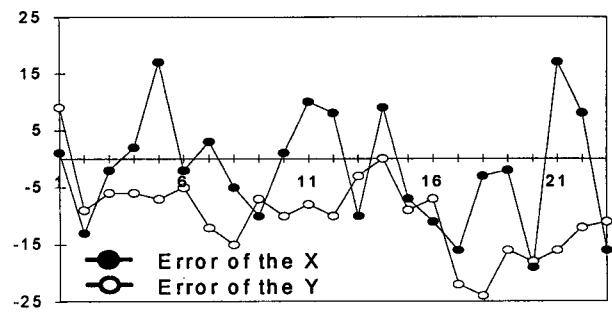


그림 12. 계산된 로봇의 위치.

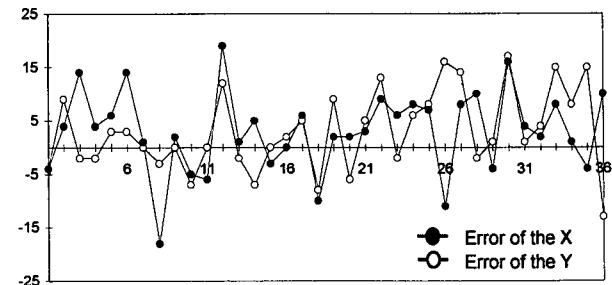
Fig. 12. The computed position of robot.

## 5. 주행 실험

그림 12은 자기 위치 인식과 장애물 검지 알고리즘의 정확도를 테스트한 실험 결과이다. 장애물은 Hallway I



(a) Hallway I



(b) Hallway II

그림 13. 자기 위치 인식 에러.

Fig. 13. The error of self-localization.

과 II에 놓여 있다. 로봇은 복도의 중앙을 따라 정해진 목표점까지 주행하도록 명령 되었다.

그림 13은 로봇의 위치의 참값과 추출값 사이의 에러를 보여주고 있다.

성공률은 80%이고 잘못 매칭한 경우와 false alarm은 0%였고 나머지 20%는 실패했다. 실패 원인은 매칭 비(matching ratio)가 너무 낮았기 때문이다. 자기 위치 인식의 평균 에러는 (9.7, 7.9)cm였다. 최대 에러는 복도에 있는 장애물 근처에서 나타났다. 장애물 위치의 평균 에러는 (2.5, 10.5)cm였다.

## IV. 결론

본 논문에서는, 실내 주행 로봇의 주행을 위한 비전을 이용한 새로운 방법을 제시하였다. 제안된 방법은 초점 거리, scale factor등과 같은 카메라의 매개변수들이 필요치 않으며 카메라의 자세와도 무관하다. 또한 초기 로봇의 위치에 관한 정보도 필요없다. 따라서, 비록 주행 중 실패할 경우도 발생하긴 하지만, 로봇이 미지의 위치에서 출발하더라도 제안된 알고리즘은 강인함을 알 수 있다. 자기 위치 인식과 관련하여서는, 대응점을 찾는데 있어서 투사 영상 불변량을 이용한 방법을 제시하였다. 또한 장애물 검지와 상대적 위치 선정법에 의한 검지된 장애물의 자세(위치)를 결정할 수 있는 방법을 제시하였다. 따라서 본 논문에서는 케리브레이션이 안된 한장의 영상만으로도 주행 로봇의 기본적 문제들을 해결할 수 있었다.

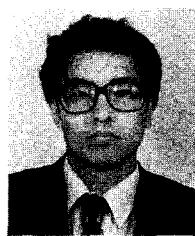
## 참고문헌

- [1] J. G. Semple, *Algebraic Projective Geometry*, Oxford Press, 1952.

- [2] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, 1973.
- [3] K. Sugihara, "Some location problems for robot navigation using a single camera," *CVGIP : Image Understanding*, vol. 42, pp. 112-129, 1988.
- [4] J. L. Mundy and A. Zisserman, *Geometric Invariance in Computer Vision*, MIT Press, 1992.
- [5] J. R. Mohr, L. Morin and E. Gross, "Relative positioning with uncalibrated cameras," *Geometric Invariance in Computer Vision*, edited by L. Mundy and A. Zisserman, MIT Press, pp. 440-460, 1992.
- [6] A. Kosaka and A. C. Kak, "Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties," *CVGIP : Image Under.* vol. 56, no. 3, November, pp. 271-329, 1992.
- [7] Y. Matsumoto, M. Inaba and H. Inoue, "Visual navigation using view-sequenced route representation," *Proc. of ICRA96*, Minneapolis, Minnesota, April pp. 83-88, 1996.
- [8] S. H. Dulimarta and K. A. Jain, "Mobile robot localization in indoor environment," *Pattern Recognition*, vol. 30, no. 1, pp. 99-111, 1997.
- [9] R. C. Gonzalez, P. Wintz, *Digital Image Processing*, Addison-Wesley, 1987.
- [10] K. S. Roh and In So Kweon, "2D object recognition using invariant contour descriptor and projective refinement," *Pattern Recognition*, May, 1998.

#### 노경식

제어·자동화·시스템공학 논문지 제4권, 제3호, 참조.



#### 이왕현

1985년 서울대학교 제어계측학과 졸업. 1992년 한국과학 기술원 자동화 및 설계공학과 졸업. 1992년-현재 한국과학기술원 박사과정 재학중. 1988년-현재 한국생산기술연구원 근무. 주관심 분야는 이동로봇, 센서퓨전.

#### 이준웅

제어·자동화·시스템공학 논문지 제4권, 제3호, 참조.

#### 권인소

제어·자동화·시스템공학 논문지 제4권, 제3호, 참조.