

전방향 능동거리 센서를 이용한 이동로봇의 자기위치 추정

Mobile Robot Localization using an Active Omni-directional Range Sensor

°정인수*, 조형석**

*한국 과학기술원 자동화 및 설계공학과(Tel:+82-2-958-3461; Fax: +82-2-968-1638; E-mail: jis@lca.kaist.ac.kr)

**한국 과학기술원 기계공학과(Tel: +82-42-968-3213; Fax: +82-42-968-3210; E-mail: hscho@lca.kaist.ac.kr)

Abstract Most autonomous mobile robots view things only in front of them. As a result, they may collide against objects moving from the side or behind. To overcome the problem we have built an Active Omni-directional Range Sensor that can obtain omni-directional depth data by a laser conic plane and a conic mirror. Also we proposed a self-localization algorithm of mobile robot in unknown environment by fusion of Odometer and Active Omni-directional Range Sensor.

Keyword Self-localization, Conic mirror, Laser conic plane, Active Omni-directional Range Sensor, Odometer

1. 서론

현재 여러 센서들을 이용하여 이동로봇을 자율 주행시키는 많은 연구들이 진행되고 있다. 그중 분해능이 비교적 높은 시각센서를 사용하는 연구가 많이 이루어지고 있는데, 대부분 이동로봇이 주행하는 전(前)면방향의 장애물을 측정하는데만 국한되어 있다. 따라서, 주행시 좌우 측면 또는 뒤에서 접근하는 장애물에 대해서는 충돌을 회피하기가 어렵다. 이에, 전(前)방향을 측정할 수 있는 센서 시스템에 대한 연구가 활발히 진행되고 있다. 본 연구에서는 원뿔 거울과 카메라 및 원뿔형 레이저면을 이용한 이동로봇용 전방향 능동 거리 센서시스템[1]을 개발하였다. 제안된 센서 시스템은 삼각 측정법에 의해 한번에 전(前)방향에 산재되어 있는 장애물간의 거리를 정확히 구하고, 그 측정된 거리정보는 이동 로봇의 자율주행뿐만 아니라 이동 로봇의 절대적인 위치와 자세를 알아내는 자기 위치보정 작업에 있어서도 매우 중요한 정보로 이용된다. 또한, 원뿔형 레이저면을 위아래로 이동시켜 측정된 이차원 거리정보들을 조합하여 주변환경에 대한 삼차원 거리정보를 만들 수 있으므로 물체 인식 작업도 수행할 수 있다. 이와 더불어 본 센서 시스템에 적합한 새로운 이동로봇의 자기위치 추정 알고리즘을 제안하고자 한다. 이동로봇이 주어진 작업영역내에서 자율 주행을 하기 위해서는 자신의 현재 위치와 자세를 추정할 수 있어야 하고, 이를 위해 사용되는 자기위치 추정방법으로는 데드렉콘(dead reckoning), 관성 주행, CCD 카메라와 표식(landmark)를 사용하는 방법, 표지탑(beacon)을 사용하는 방법등이 있다. 데드렉콘[2]이나 관성 주행 시스템은 매 시각마다 이동로봇의 속도나 가속도를 측정함으로써 이동로봇의 위치를 연속적으로 계산될 수 있다는 장점이 있으나, 주행거리가 길어지거나 휠의 미끄러짐등이 발생하면 치명적인 누적오차가 발생된다. 이러한 문제점을 해결하기 위해 CCD 카메라와 랜드마크를 사용하는 연구[3][4]가 진행되었으나, 카메라가 랜드마크로부터 멀어질수록 위치 오차가 커지고, 어둡거나 밝은 환경 또는 연기가 자욱한 환경에서는 사용할 수가 없다. 이에, 개선된 형태의 자기위치 추정 방법으로 주변환경의 여러곳에 기준이 되는 위치 표지탑(beacon)을 설치하고 레이저빔이나 초음파에 의해 이동로봇의 위치를 추정하는 연구[5]-[7]가 진행되었으나 표지탑이 설치된 영역이 넓어질수록 측정 오차는 커지고, 심지어 이동로봇이 그

영역을 벗어나면 측정 오차가 너무 커져서 위치 측정값을 신뢰할 수 없다. 이와 더불어 사전에 얻어진 주변환경 모델과 초음파 센서들로 부터 측정된 데이터를 비교하므로써 이동로봇의 위치를 추정하려는 모델기준 방법(model based method)[8]-[10]들도 행해졌으나 사전에 주변환경을 알고 있어야 한다는 제약 때문에 복잡한 형태로 구성되거나 시시각각 변화되는 실제 환경에 대해서는 적합하지 못하다. 따라서 사전의 정보없이 복잡하게 변화되는 주변환경하에서도 이동로봇의 위치와 자세를 정확히 추정할 수 있는 자기위치 추정 방법이 절실히 요구된다. 이에 본 연구에서는 레이저에 의한 레인지 거리 정보와 거리주행계 정보를 동시에 얻고, 그 정보들을 적절히 결합(fusion)하므로써 좀더 정확한 이동로봇의 위치 및 자세를 구하고 시시각각 변화하는 주변환경에 적응력을 갖는 자기위치 추정 알고리즘을 제안하고자 한다.

2. 전방향 능동거리 센서 시스템

CCD 카메라와 원뿔거울을 적절히 배치하면 원뿔 거울의 고유한 특성에 의해 전(前)방향의 주변환경을 한번에 볼 수 있게 된다. 그 특성을 이용하기 위해 본 연구에서는 원뿔 거울과 CCD 카메라를 수직으로 배치하고, 전(前)방향의 거리 정보를 구하기 위해 그 수직축 선상에 레이저 포인트 소스를 두어 수직축 중심으로 회전시키므로써 원뿔형의 레이저면을 형성시켰다. 즉, 레이저 포인트 소스를 고속으로 회전시켜 주변환경에 레이저 띠가 형성되도록 하고, 원뿔 거울을 통해 전(前)방향에 맺힌 레이저 띠 영상을 카메라로 읽어들이는 것이다. 이때 레이저 띠는 방위각에 따라 유일하게 결정되는 각 레이저 포인트들의 집합체이므로 삼각 측정법을 이용하여 각 레이저 포인트들에 대한 거리를 측정하면 한번에 주변환경에 대한 이차원 거리정보를 구하는 결과가 된다. 또한 원뿔형의 레이저 면을 위,아래 각도로 조절하면서 이차원 거리정보를 조합해 나가면 전(前)방향의 주변환경에 대한 삼차원 거리 정보도 얻을 수 있다. 본 센서 시스템의 구성 및 원리에 대한 개략도는 그림 1 과 같다.

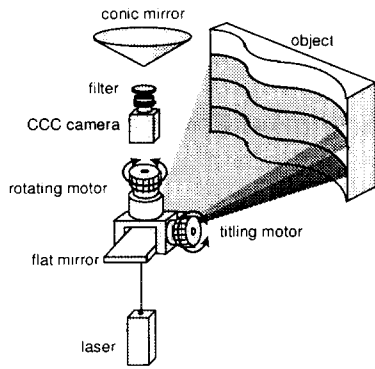


그림 1 전방향 능동거리센서 시스템
Fig. 1 The active omni-directional range sensor system

이에 실제환경에서 이차원 거리정보를 구한 실험 결과는 그림 2 와 같다.

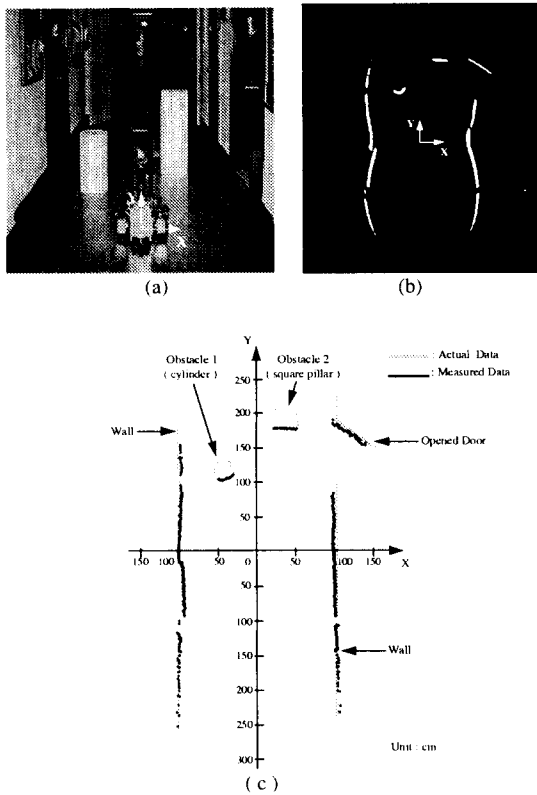


그림 2 (a) 실험 환경
그림 2 (b) 원뿔거울을 통해 투영된 영상
그림 2 (c) 측정된 이차원 거리정보
Fig. 2 (a) The experimental environment
Fig. 2 (b) Input image reflected by conic mirror
Fig. 2 (c) Measured two-dimensional depth data

3. 제안된 자기위치 추정 알고리즘

또한 본 센서 시스템을 이동로봇의 자율주행에 적용하기 위해서는 이동로봇의 자기위치 추정 알고리즘이 필요하였고, 이에 레이저에 의한 레인지 거리 정보와 거리주행계 정보를 결합(fusion)하여 이동로봇의 자기위치를 추정하는 새로운 알고리

즘을 제안하고자 한다. 본 알고리즘은 사전에 주변환경에 대한 정보가 필요없으며, 이동로봇의 자기위치 추정뿐만 아니라 주변에 산재되어 있는 장애물들의 특성도 유추해 낼 수 있다는 장점을 가지고 있다.

3.1 제안된 알고리즘의 원리

거리 주행계에 의한 테드랙콘 정보는 주변환경 조건에 따라 심각한 누적오차를 유발시킨다. 이에 본 연구에서는 서로 다른 두 위치에서 얻어진 레인지 정보들을 정합함으로써 상대적인 위치와 자세의 변환 정보를 추정하고 이 추정값에 의해 테드랙콘 방법에 의해 구한 이동로봇의 현 위치 추정값을 보정해 준다. 이를 위해서는 모든 물체가 강체이며, 각 물체들은 연속된 점으로 표현 가능하다는 가정이 필요하다. 이에 레인지 정보에 나타난 주변 물체들을 표현하기 위해 어떠한 물체도 표현 가능한 자유곡선(free-form curve) 표현 방법을 채택하였다. 따라서 이동로봇이 자율주행시 전, 후 위치에서 얻은 두 레인지 정보들을 자유곡선으로 표현하고 그 두 자유곡선의 직교거리를 최소화(minimum normal distance)하는 최적 기법을 이용하여 두 자유곡선을 정합한다. 사용된 최적화 방법은 최소 자승법(least square method)와 엠에스티메이터(M-estimator)이었으며, 시뮬레이션에 의해 두 기법의 성능을 평가해 보았다.

3.2 자유 곡선 정합 (free-form curve registration)

본 연구에서는 레이저에 의해 구해진 이차원 레인지 거리 정보를 자유곡선으로 표현하였으므로 정합코자 하는 두 곡선은 유클리디언 변환(euclidian transformation)관계에 있다고 간주할 수 있다. 이에 두 곡선 사이의 변환 관계는

$$T = T(a, b, c, d) = \begin{bmatrix} a & -b & c \\ b & a & d \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

또한 자유 곡선 정합을 위해 이동로봇이 주행시 전, 후 위치에서 구한 두 곡선 사이의 직교 거리를 구해야 한다. 이를 위해, 그림 3에서 알 수 있듯이, 전 위치에서 구해진 곡선상의 한 점 $P_i(x_i, y_i)$ 에서 법선을 구한후 그 법선을 따라 후 위치에서 구해진 곡선과 만나는 점을 구하고 다시 그 점에 대한 접선을 구한다. 이에 그 접선과 점 $P_i(x_i, y_i)$ 사이의 거리, 즉 두 곡선 사이의 직교 거리 d_i 를 구할 수 있다.

여기서 임의의 직선 $Ax + By + C = 0$ 와 임의의 한 점인 $P(x_o, y_o)$ 사이의 거리는

$$d = \frac{Ax_o + By_o + C}{\sqrt{A^2 + B^2}} \quad (2)$$

따라서 이동 전 위치에서 구해진 곡선상의 한 점 $P_i(x_i, y_i)$ 에서의 d_i 는

$$d_i = \frac{[A_i \ B_i \ C_i]}{\sqrt{A_i^2 + B_i^2}} [x_i \ y_i \ 1]^T \quad (3)$$

이며, 이는 곡선상에 존재하는 모든 점에 적용된다.

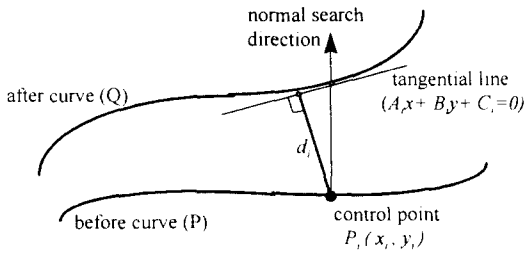


그림 3 두 자유곡선사이의 거리오차
Fig. 3 Error distance between two free-form curves

이에 오차 함수를 다음과 같이 정의한다.

$$e_k = \sum_i d_i^2 = \sum_i \frac{1}{\sqrt{A_i^2 + B_i^2}} (A_i(ax_i^{k-1} - by_i^{k-1} + c) + B_i(bx_i^{k-1} + ay_i^{k-1} + d) + C_i)^2 \quad (4)$$

그러므로 이 오차함수를 최소화 하는 모션 파라메타(motion parameter)들을 여러 최적화 기법에 의해 구할 수 있다. 이에 모션 파라메타에 대한 오차함수의 편미분은 다음과 같은 최소자승 형태의 식을 유도한다.

$$\begin{bmatrix} \sum (Ax + By)^2 & \sum (Bx - Ay)(Ax + By) & \sum A(Ax + By) & \sum B(Ax + By) \\ \sum (Bx - Ay)(Ax + By) & \sum (Bx - Ay)^2 & \sum A(Bx - Ay) & \sum B(Bx - Ay) \\ \sum A(Ax + By) & \sum A(Bx - Ay) & \sum A^2 & \sum AB \\ \sum B(Ax + By) & \sum B(Bx - Ay) & \sum AB & \sum B^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} -\sum C(Ax + By) \\ -\sum C(Bx - Ay) \\ -\sum CA \\ -\sum CB \end{bmatrix} \quad (5)$$

즉, 식(3)과 식(4)를 반복적으로 수행하므로써 두 자유곡선을 정합하고, 최종적으로 두 곡선사이의 모션 파라메타를 구해낸다. 그러나, 이동로봇이 실제 환경에서 장애물을 피해야하며 자율 주행을 할때에는 정합에 오차를 유발시키는 많은 노이즈, 즉 장애물의 이동, 사리짐, 동장 등이 발생하므로 일반적인 최소자승법은 정확한 모션 파라메타를 구하기 어렵다. 다시말하면 이동로봇 주행과 같은 작업하에서의 정합문제에는 더욱 강건한 최적화 기법이 요구된다. 이를 위한 많은 강건 기법(robust technique)이 있으나, 주변환경에 대한 사전 정보가 없는 상태에서의 레인지 데이터 정합 문제에 있어서는 파라메타 개수나 일치점을 알 필요가 없는 엠에스티메이터(M-estimator) 기법이 가장 적합하다고 사려되어 이를 적용코자 한다. 이에 오차 함수를 다시 정의하면

$$e_k = \sum_i \omega(d_i^{k-1}) d_i^2 \quad (6)$$

$$\omega(d_i) = \begin{cases} 1 & d_i < \sigma \\ \frac{\sigma}{|d_i|} & \sigma < d_i < 3\sigma \\ 0 & d_i > 3\sigma \end{cases} \quad (7)$$

여기서 ω 는 weight 함수이며, 식(7)은 Huber[11]에 의해 제안되어 널리 사용되는 형태이다. 그러므로 이 오차함수를 최소화 하는 모션 파라메타(motion parameter)들을 구해야 하고, 이에

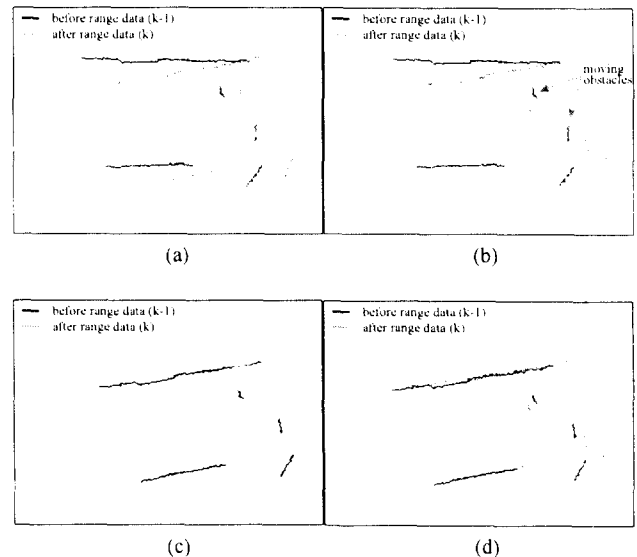
모션 파라메타에 대한 오차함수의 편미분에 의해 다음과 같은 식을 유도할 수 있다.

$$\begin{bmatrix} \sum \omega(d_i^{k-1}) A_i(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) \\ \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay)^2 & \sum \omega(d_i^{k-1}) A_i(Bx - Ay) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) \\ \sum \omega(d_i^{k-1}) A_i(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay) & \sum \omega(d_i^{k-1}) A_i^2 & \sum \omega(d_i^{k-1}) A_i AB \\ \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) & \sum \omega(d_i^{k-1}) A_i(Bx - Ay)(Ax + By) & \sum \omega(d_i^{k-1}) A_i AB & \sum \omega(d_i^{k-1}) A_i B^2 \\ -\sum \omega(d_i^{k-1}) C(Ax + By) & -\sum \omega(d_i^{k-1}) C(Bx - Ay) & -\sum \omega(d_i^{k-1}) CA & -\sum \omega(d_i^{k-1}) CB \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} -\sum \omega(d_i^{k-1}) C(Ax + By) \\ -\sum \omega(d_i^{k-1}) C(Bx - Ay) \\ -\sum \omega(d_i^{k-1}) CA \\ -\sum \omega(d_i^{k-1}) CB \end{bmatrix} \quad (8)$$

이에 식(5) 대신 식(8)을 이용하여 반복적으로 수행하므로써 두 자유곡선을 정합하고, 최종적으로 두 곡선사이의 모션 파라메타를 구해낸다.

3.3 시뮬레이션

이에 시뮬레이션을 통해 두 최적기법을 이용한 이동로봇의 자기위치 추정 알고리즘을 실현해 보았다. 시뮬레이션의 정확도를 높이기 위해 실제 환경에서 본 센서 시스템에 의해 구한 이차원 거리 정보를 데이터로 사용한다. 그 이차원 거리 정보를 이동로봇이 X축으로 500 mm, Y축으로 -200 mm, 방향은 10.0°로 움직였다는 가정하에 정합코자하는 새로운 이차원 거리정보를 만들었고, 이에 제안된 알고리즘을 적용하여 추정된 모션 파라메타를 통해 오차를 비교하므로써 그 성능을 평가하여 보았다. 그림 4 (a)는 움직이는 장애물이 없는 경우 이동로봇의 이동전후 위치에서의 이차원 거리 정보이다. 그림 4 (b)는 두개의 움직이는 장애물이 있다고 가정한 경우이다. 이에 최소자승법에 의해 행해진 자기위치 추정 결과는 그림 4 (c), (d)와 같다. 그림 4 (c)에서 추정된 값은 X값은 499.7 mm, Y값은 -201.7 mm, 방향은 9.72°이며, 반복횟수는 5회이었다. 그림 4 (d)에서 추정된 값은 X값은 123.1 mm, Y값은 -286.7 mm, 방향은 9.34°이며, 반복횟수는 15회이었다. 또한 엠에스티메이터에 의해 행해진 자기위치 추정 결과는 그림 4 (e), (f)와 같다. 그림 4 (e)에서 추정된 값은 X값은 500.8 mm, Y값은 -201.7 mm, 방향은 9.70°이며, 반복횟수는 6회이었다. 그림 4 (f)에서 추정된 값은 X값은 508.5 mm, Y값은 -199.3 mm, 방향은 9.74°이며, 반복횟수는 13회이었다. 이상의 결과에서 알수있듯이 움직이는 장애물이 없는 경우에는 두 최적기법 모두 목표값을 훌륭히 추정하는 반면, 움직이는 장애물이 있는 경우에는 엠에스티메이터가 탁월한 능력을 발휘한다는 것을 알수있다.



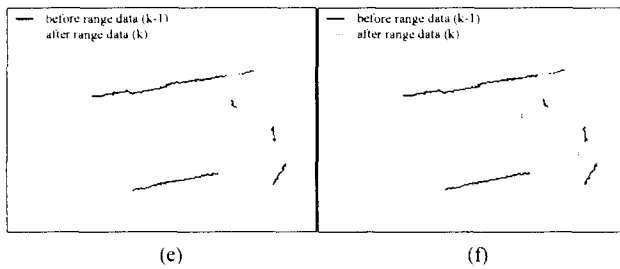


그림 4 (a) 정적 환경에서의 이차원 거리정보
 그림 4 (b) 동적 환경에서의 이차원 거리정보
 그림 4 (c) 정적환경에서의 최소자승법에 의한 위치 추정
 그림 4 (d) 동적환경에서의 최소자승법에 의한 위치 추정
 그림 4 (e) 정적환경에서의 엠에스티메이타에 의한 위치 추정
 그림 4 (f) 동적환경에서의 엠에스티메이타에 의한 위치 추정
 Fig. 4 (a) Two-dimensional depth data in static environment
 Fig. 4 (b) Two-dimensional depth data in moving environment
 Fig. 4 (c) Self-localization by Least Square method (static environment)
 Fig. 4 (d) Self-localization by Least Square method (moving environment)
 Fig. 4 (e) Self-localization by M-estimator method (static environment)
 Fig. 4 (f) Self-localization by M-estimator method (moving environment)

4. 결론

본연구에서는 새로운 형태의 이동로봇용 센서 시스템을 개발하였고, 실험을 통해 얻어진 최대 측정 오차는 유효 측정 영역에서 7.8 cm 이었다. 이 정도의 거리측정오차는 이동로봇에 적용된다는 관점에서 허용 가능한 거리 측정 오차라 사려된다. 앞으로 제안된 센서시스템의 측정 정밀도를 높이기 위한 더욱 개선된 형태의 시스템 설계를 고려해 보고자 한다. 또한, 본 센서시스템에 적합한 이동로봇의 자기위치 추정방법을 제안하였고, 시뮬레이션을 통해 그 가능성을 검증하였다. 이동로봇은 현실적으로 움직이는 장애물이 산재한 환경에서 주어진 작업을 수행해야 하는 경우가 대부분이며, 그러한 환경속에서의 자기 위치 추정을 위해서는 엠에스티메이타와 같은 강건한 최적기법을 필요로 한다. 또한 시뮬레이션에서 알수 있듯이 움직이는 장애물은 강건한 최적 기법에 의해 정지된 물체들과 구별될 수 있다. 따라서 따로 분류된 움직이는 장애물에 대한 특성 정보도 동시에 얻을 수 있는 알고리즘 개발이 진행중이다. 더불어 실제 실험을 통해 그 성능을 평가해 보고자 한다.

참고문헌

[1] I.S. Joung and H.S. Cho, " An Active Omni-directional Range Sensor for Mobile Robot Navigation", IFAC-IFIP-IMACS Conf. on Control of Industrial System, Belfort, France, vol. 2, pp. 431-436, 1997.
 [2] S. Yuta *et al.*, " An implementation of MICHII-A locomotion command system for intelligent mobile robot ", in Proc. Int. Conf. Advanced Robotics, pp. 127-134, 1985.
 [3] J. H. Kim and H. S. Cho, " Real time determination of a mobile robot's position by linear scanning of a landmark ", Robotica, vol. 10, pp. 309-319, 1992.
 [4] I. Fukui, " TV image processing to determine the position of a robot vehicle ", Pattern Recognition, vol. 24, no. 1-6, pp. 101-109, 1981.

[5] K. Nishide, M. Hanawa and T. Kondo, " Automatic position findings of vehicle by means of laser ", in Proc. IEEE Conf. Robotics and Automation, pp. 1343-1348, 1986.
 [6] T. Arai and E. Nakano, " Development of measuring equipment for location and direction(MELODI) using ultrasonic waves ", Trans. ASME, J. of DSMC, vol. 105, pp. 152-156, 1982.
 [7] L. Kleeman, " Optimal estimation of position and heading for mobile robots using ultrasonic beacons and dead reckoning ", in Proc. IEEE Int. Conf. Robotics and Automation, pp. 2582-2587, 1992.
 [8] M. Drumheller, " Mobile robot localization using sonar ", IEEE trans. PAMI, vol. 9, no. 3, pp. 376-382, 1991.
 [9] J. J. Leonard and H. F. Durrant-Whyte, " Mobile robot localization by tracking geometric beacons ", IEEE trans. on Robotics & Automation, vol. 7, no. 3, pp. 376-382, 1991.
 [10] A. A. Holenstein, M. A. Muller and E. Badreddin, " Mobile robot localization in structured environment cluttered with obstacles ", in Proc. IEEE Int. Conf. Robotics & Automation, pp. 2576-2581, 1992.
 [11] P. J. Huber. Robust Statistics. John Willey and Sons, 1981.