

알려지지 않은 장애물 존재시 자율주행 운반체의 장애물 회피

원 지옥, 이 기성
홍익대학교 전기제어공학과

Collision Avoidances of an Autonomous Mobile Robot With Unknown Obstacles

Jiuk Won, Keeseong Lee
Department of Electrical & Control Engineering,
Hong-Ik University

Abstract - A real-time collision avoidance algorithm for an AMR (Autonomous Mobile Robot) using the Distance Transform and Vector Field Histogram is studied. This approach enables AMR to find a collision-free path with the unknown obstacles. All system parameters including positions of the obstacles can be accepted using a mouse icon and all the obtained trajectories can be displayed on a computer monitor in graphics.

1. 서론

현재 제조업 등 산업현장에서는 공장자동화를 위해 로봇에 대한 연구가 급속히 증대되고 있다. 최근의 로봇은 기존의 고정식 로봇보다는 작업환경이 넓고, 경로를 지정없이 Station과 Station 사이의 최적의 경로를 찾아 원하는 작업을 효율적으로 처리할 수 있는 자율주행 로봇에 대한 연구가 활발히 진행 중이다. 자율주행 로봇의 원형이라 할 수 있는 AGV(Automated Guided Vehicle)의 경우에는 작업환경의 변화에 능동적으로 대처하기 어렵고, 주행경로를 변화시키기 위해서는 매우 많은 경제적 비용이 필수적이며 주행 경로상에 사전에 알 수 없었던 장애물이 존재하는 경우에는 정지를 하거나 오동작이 일어나게되어 무인화된 작업영역에서 작업목적을 달성하기 어려운운 것이 사실이었다. 자율주행 로봇은 최적 경로를 찾기위해 Global Map을 작성하고 장애물이 존재할 경우에는 Local Map을 작성하여 장애물을 피해야하는 효율적인 장애물 회피 알고리즘을 필요로 한다.

본 논문에서는 주어진 환경하에서 시작점과 목표지점으로 환경을 인지하고, 주변환경을 인식해서 시점으로부터 장애물들을 회피하며 실시간 목표지점에 도달하도록 하였다. 로봇이 이를 수행하기 위해서 주변환경을 각종 센서 (초음파 센서, 적외선 센서, 접촉 센서, Vision 센서)를 통해서 인지할 수 있어야 하며, 이를 이용해서 Global Map을 작성하고 경로를 따라 이동할 수 있도록 구성하였다. 이때 예상치 못한 장애물이나 통과하기 힘든 경로가 존재할때는 가상력장인 벡터장 히스토그램(VFH: Vector Field Histogram)[5,6]을 이용해서 경로를 재설정하여 Local Map을 작성하고 장애물을 회피함으로써 목표지점까지 주행할 수 있는 알고리즘을 설계하였다.

2. 경로계획을 위한 알고리즘

자율주행 로봇이 임의의 실내 환경하에서 시점으로부터 종점에 이르는 최적경로를 통해 충돌없이 이동을 하기 위해서는 Global Map과 Local Map을 작성해야 한다. 벽면추종법(Wall Following)[1]은 벽면을 따라 주행하는 기존의 마이크로 마우스의 경로계획 기법을 그대로 사용하였으므로 자유공간과 장애물이 불규칙한 크기로 존재하는 공간에서는 실제로 적용하기 어렵다. 접선법(Tangent Graph)[3,4]의 경우 Global Map 작성에 있어서 장애물들의 Edge를 추출하여 그 Edge사이에 존재하는 접선들을 모두 찾아내고 그 접선들 중에서 통과 가능한 경로들을 찾아낸다. 그리고 그 경로들 중에서 최적경로를 선택해서 Global Map을 작성한다. 따라서 복잡한 자연환경에서 구현할 수 있는 특징이 있으나 전처리에 시간이 너무 소요되므로 실시간 응답을 하기가 어려우며 환경의 변화에 능동적으로 동작하기 어려운 문제가 있다. Edge 판별법[1]은 알려지지 않은 장애물을 만났을때 일시 정지하여 그 장애물의 Edge들을 판별하고 장애물의 Edge사이에 가상선을 긋는다. 그리고 Edge와 임의의 가상선 영역을 장애물로 인식하여 그 장애물을 회피하여 시점에서 종점에 이르는 경로를 따라 이동하게 된다. 그러나 이 알고리즘은 실시간에 센서 데이터 정보만으로 정확히 Edge를 판별하기 어렵고, 판별을 하더라도 장애물이 사각형이 아닌 다각형이거나 원형인 경우에는 센서 데이터의 신뢰성을 보장하기 어렵다. 그리고 Network/Graph Models (일명 Tangent Graph법)[3]과 Grid-based Model[2]이 있는데 Network/Graph Models은 복잡한 자연환경에 적합한 방법이지만 복잡한 접선들 중에서 실제로 유효한 데이터를 얻는데는 자연시간이 많기 때문에 실시간 처리가 어려운 문제가 있을 뿐 아니라 센서정보를 정확하게 요구하며 실제 적용하기가 어려운 단점을 가지고 있다. 거리변환법(DT: Distance Transform)[1]과 같은 Grid-based Model은 Network/Graph Models보다 적용하기에는 간단한 방법인데 DT는 데이터의 처리문제에 있어서 특정의 Grid만을 추출하여 분석하기 때문에 Global Map 작성에 실시간 응답을 보인다. 그러나 낮은 해상도의 데이터로 인하여 경로계획에 부정확한 특성을 보인다. Certainty Grids 방법[7]의 경우에는 초음파센서의 다중반사경로 효과등으로 불완전한 센서 정보를 확률적으로 처리하여 센서 앞에 존재하는 물체의 위치와 거리를 알려주며 이것으로부터 각 Cell에 주어진 접령 또는 비접령 확률을 계산하는 방법을 사용하여 알려지지 않은 환경의 지도작성에 사용하였다. Local Map을 작성하는 방법중 Borenstein과 Koren[5,6]은 로봇의 운동면상에 목적지와 장애물을 힘의 크기와 방향을 가진 가상력장으로 가정하고, 장애물을 피하며 목적지까지 빠른 시간내에 도달하

는 방법을 제안하였다. 그러나 이방법은 좁은 주행 공간 내에서는 로봇의 주행이 불안정하였기 때문에 다시 각 셀들의 확실성 값들을 이용하여 로봇의 주위를 장애물의 밀도에 대하여 표현하고 그 사이에서 경로를 찾아 주행하는 베티장 히스토그램 방법을 제안하고 실시간으로 장애물을 회피하며 주행하도록 하였다[8].

3. 경로계획

로봇의 주행 경로에서 만날 수 있는 장애물이 알려진 경 우에는 거리 변환법(DT)을 사용하였다. 장애물과 같은 물체의 인식이 Grid의 특징만을 추출하여 사용하므로써 Global Map작성을 작성하였는데 물체를 실제의 크기보다 크게 인식하는 방법을 사용하지는 않았다. 알려지지 않은 장애물의 경우에는 장애물과의 충돌이 우려되기 되기 때문에 매우 신중히 다루어야 하므로 Local Map을 VFH를 응용하여 작성하였다. 그러나 Global Map작성에는 적합하지 않았다. 따라서 Global Map 작성은 DT를 이용하고 Local Map 작성은 초음파센서를 이용한 VFH를 응용해서 이 두가지 Map작성법을 유기적으로 연계시켰다.

3.1 Global Map 작성

Global Map을 작성하기 위해서는 DT를 적용하고 환경을 인식해서 그 최적경로를 찾게된다. DT는 목표점으로부터 시작점에 이르는 통과가능한 경로를 검색함으로써 경로계획을 하는데, 경로계획은 자율주행 운반체가 도착해야 할 목표점을 초기위치로 설정하고 초기위치에서 시점에 이르는 구간을 장애물이 있는 구간과 없는 구간으로 분류해서 처리한다.

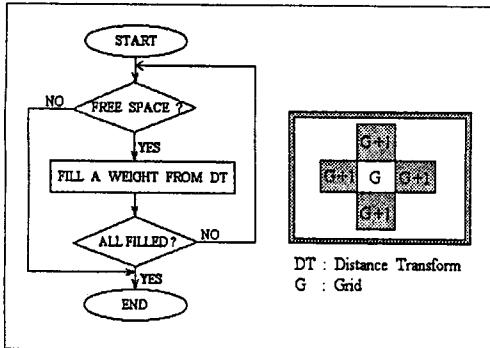


그림 1. DT에 의한 Weight값 설정방법

분류를 위해 정사각형 형태의 Cell로 세분화하는데 본 논문에서는 그림 1과 같이 Cell의 각각에 목표점으로부터 거리를 고려하여 인접한 4개의 Cell만을 고려하며 가중치를 인가한다. 그 후 설정된 가중치부터 시작하여 자유공간이 없을 때까지 모두 가중치를 인가한다. 그리고 장애물은 데이터형의 가장 큰 값을 인가하는데 시점은 장애물보다 한 단계 작은값을 인가하고 목표점은 제일 작은값으로 인가하게 된다. 가중치를 모두 인가한 후 목표점에 이르는 가장 경사가 가파른 경로(가중치의 변화가 가장 큰 경로)를 탐색하는 형식으로 진행하게 된다. 그림 2는 환경을 인식하기 전의 초기 값과 환경의 인식을 마친 후의 가중치변화에 대한 예를 보여주고 있다.

70	70	70	70	70						
70	70	70	70	70						
70	70	70	70	70						

3	2	1	0	1						
4	3	2	1	2						
5	4	3	2	3						

(a) Original Block of Grid

(b) After DT

그림 2. 초기치와 DT후의 가중치 비교의 예

만일 이 가파른 경사의 페스가 존재하지 않으면 Start Cell에서 Goal Cell에 이르는 페스가 없다고 결론짓는다. 앞에서 경사면의 해석을 하게되는 주된 이유는 적각으로 이동하는 경우보다도 로봇이 회전해야 할 필요를 줄이어 상대적으로 종점에 이르는 시간을 절약할 수 있기 때문이다. 자유공간(Free Space)에서 넓은 영역을 Global Map 작성을 위해 모두 해석하게 된다면 방대한 계산량과 이에 따른 계산시간의 지연이 생기게 되므로 경사면을 해석하면 수행 속도의 향상을 볼 수 있을 것이다.

9	8	7	6	5	4	3	2	3	4
10	9	8	9	9	3	2	1	2	3
11	10	9	9	9	2	1	G	1	2
12	11	10	9	9	9	2	1	2	3
13	12	11	9	9	9	9	3	4	
14	13	12	9	9	9	9	9	4	5
15	14	S	10	9	8	7	6	5	6

그림 3. 사각형 장애물의 인식과 Weight값 설정

9	8	7	6	5	4	3	2	3	4
10	9	8	9	9	3	2	1	2	3
11	10	9	9	9	2	1	G	1	2
12	11	10	9	9	9	2	1	2	3
13	12	11	9	9	9	9	2	3	4
14	13	12	9	9	9	9	9	4	5
15	14	S	8	7	6	5	4	5	6

그림 4. 다각형 장애물의 인식과 Weight값 설정

그림 3과 그림 4는 중앙의 다각형 형태로 존재하는 장애물의 해석 예를 보여주고 있다. 다각형의 장애물의 정확한 해석을 위해서는 많은 노력이 필요하나 본 논문에서는 사용될 로봇의 크기를 고려하여 80cm간격으로 Grid를 지정해서 그 Grid에서의 특징만을 추출하여 Global Map 작성에 이용하였다. 따라서 기존의 사용하면 DT방식과는 달리 장애물의 Edge 부분이 Grid의 중앙에 위치하지 않기 때문에 장애물이 작게 인식되는 경향이 있다. 이 경우 낮은 해상도로 인한 오차가 존재하기 때문에 다음절에서 언급할 VFH를 이용해서 해상도를 높여 처리하게된다. 위와 같은 과정을 수행하기 위해서는 실행되어야 할 것이 있는데 사용자는 환경을 정확하게 입력해야 하고, 자율주행 운반체는 분류학적인 방법으로 센서 입력데이터를 이용해 환경을 인지하고 처리해야한다. 그후 그 데이터에 근거해서 환경을 학습(Learning)하고, 경로계획에 따른 수행과정을 거치게 되어 실제 환경을 Update하게 된다.

3.2 최단경로 탐색 알고리즘

대부분의 최적화 문제에서처럼 최단경로의 평가는 계산 시간을 고려해야 한다. 앞에서 언급한 것과 같이 거리변환법은 Global Map에서 최단경로를 발견하는데 효과적인 알고리즘으로 디자인할 수 있음을 알 수 있었다. 즉 이를 구현하기 위해 Grid-based Model 방법을 기초로 하여 거리변환법을 사용하였다. 또한 시스템의 환경을 최적화하기 위해서 재귀호출(Recursive Programming)법을 사용함으로써 메모리의 효율을 극대화하고 수행결과를 그래픽으로 처리하여 한눈에 알아볼 수 있도록 하였다.

3.3 장애물 회피 알고리즘(Local Path)

장애물이 있는 경우 장애물의 위치와 크기에 따라 결정되는 가상의 척력(斥力)으로 만들어진 벡터장 히스토그램(VFH)을 사용한다. 벡터장 히스토그램은 VFF(Virtual Force Field)를 보완한 것으로서 Certainty Grid를 확률적으로 해석하여 통과 가능한 넓이를 항상시킨다. 그러나 이 알고리즘은 효율적인 Global Map 작성을 해주지 않으면, 장애물이 폐포면 형태를 가지며 시점과 종점을 가로막는 경우에 종점을 찾을수가 못하는 경우가 있다. 따라서 Global Map 작성에는 바탕직하지 않으므로 본 논문에서는 Local Map 작성에 VFH의 기본 알고리즘을 이용하였다. 로봇은 Global Map 작성에서 얻은 경로를 추종하는 전방향 진행 성분과 장애물에 반대방향으로 작용하는 성분을 벡터합성을 하고, 이 선분들로 이루어지는 히스토그램을 이용하여 Threshold 영역에서 Global Map에서 얻어진 경로를 찾아가는 경로를 발생한다. 위의 방법이 원래의 VFH와의 차이점은 Global Map을 이용해서 현재위치 전방향 한 그리드 지점을 임시종점(Sub-Goal)으로 설정하고 VFH를 통해 Threshold를 찾아 로봇이 이동을 해서 효과적으로 그 장애물을 피해 규정된 경로를 추종하는 것이다. 같은점은 Steering Control을 조향장치에 대한 고려를 하여 Threshold 부분으로 Heading Angle, θ 를 조정하여 모터구동을 위한 계산량을 줄였다는 것이다. 따라서 빠져나가기 힘든 경로를 만났을 경우에도 자율주행 로봇은 정지시키지 않고도 빠르게 주행시킬 수 있는 장애물 회피 알고리즘을 구현하였다.

3.4 Global Map과 Local Map과의 결합

Global Map을 작성하여 로봇이 그 경로를 추종하며 동작하면 중 장애물을 만났을 경우에는 우선 VFH에 의해서 피해갈 수 있는 Path를 발견하여 동작한다. 그러나 Global Map을 따

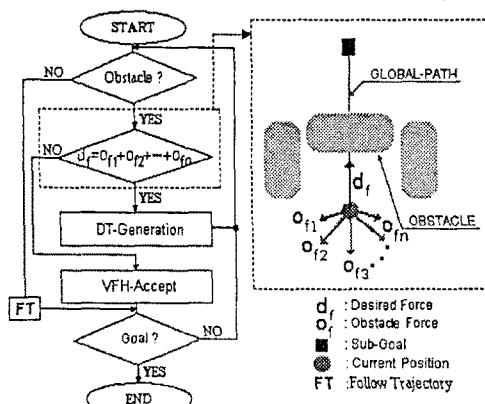


그림 5. 장애물 회피 알고리즘

라갈 수 없을 정도의 큰 알려지지 않은 장애물이 존재할 경우에는 진행방향의 초음파정보를 이용해서 Global Map작성함수에 돌려주게 되며 이의 수행 개념도는 그림 5와 같다. 만일 Global Map 작성함수가 이 값을 받게되면 진행하기 어려운 지점의 절대좌표와 장애물의 위치함수를 통해 환경을 Update하고 그 지점을 시점으로해서 다시 Global Map 작성을 하며 그 결과를 Local Map 함수에게 돌려준다. 따라서 그 지점부터 경로계획을 다시 시작하게 된다.

4. 시뮬레이션 방법 및 결과

위의 벡터장 기법을 이용한 경로계획의 실용성을 증명하기 위해 다음과 같은 시뮬레이션 과정을 거쳤다. 그림 6과 같이 가로 20m 세로 19.2m의 직사각형 형태의 실내에서 이동로봇이 동작한다고 초기설정을 하였고, Grid-based Model을 구체화 하기 위해서 사용예정인 HMR-1(Hong-Ik Mobile Robot-1)로봇의 크기를 고려하여 80cm 간격의 Grid를 설치하였다. 그리고 그 Grid에 장애물의 존재 유무를 판단하여 거리변환법을 사용하여 로봇이 동작할 환경을 인지하도록 하였다. 알려진 장애물을 알려지지 않은 장애물을 설치하기 위해 사각형 형태의 임의의 장애물을 공간상에 배치하였으며 알려진 장애물과 알려지지 않은 장애물을 구별하기 위해서 색을 다르게 입력 받을 수 있도록 오른쪽 메뉴에 Icon형태로 배치하였다. 정확한 입력을 위해 마우스의 현재좌표를 화면 우측, 상단에 디스플레이 함으로써 사용자관리에 편의성을 도모하였고, 시작점과 종점을 다른 색으로 원형태의 로봇을 근사화 해서 배치하도록 하였다. 또한 거리변환법의 결과를 우선 살펴보기 위한 메뉴를 배치하였고, 이 거리변환법과 VFH를 결합한 결과를 명시적으로 보기 위한 메뉴를 따로 구성해서 사용하였다. 또한 어느 특정의 실내공간을 매번 입력시키는 번거로움을 줄이기 위한 기능을 추가하여 배치하였다.

시뮬레이션의 과정을 가시적으로 보이기 위해 해상도 640*480, 16컬러를 지원하는 모니터를 사용하였고 이는 실제 상황에서 한 Pixel당 4cm의 해상도를 갖게 된다. 그리고 로봇의 진행과정을 살펴볼 수 있도록 Global Map은 실선으로, 실제 시뮬레이션에 의해 발생된 로봇의 경로는 절선의 다른 색상으로 나타내었다. 시뮬레이션 진행과정은 그림 6과 같이 우선 입력장치인 마우스를 통해서 장애물, 시점, 종점등을 입력한다. 그후 입력된 환경정보를 이용해서 그림 7과 같이 DT법에 의해 경로를 발생하고, 그림 8은 그림7에서의 경로를 이용해서 초음파 센서 정보를 이용하여 시점에서 종점에 이르는 경로를 나타내며, 그림 9와 10에서는 Global Map을 작성한 후 알려지지 않은 장애물이 경로상에 발생될때 그 장애물을 피해 종점에 이르는 수행과정을 보여준다. 그림 9는 알려지지 않은 장애물이 경로를 가로막지 않아 그 알려지지 않은 장애물을 피해가는 경우이고, 그림 10은 장애물이 발생된 경로를 가로막아 자율주행 로봇이 경로를 추종하기 힘든 경우이다. 따라서 이 경우 로봇은 경로를 재 발생시키고 그 경로를 추종하는 모습을 보여준다.

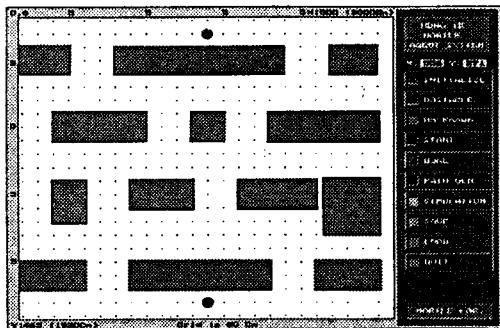


그림 6. 경로계획을 위해 환경설정한 모습

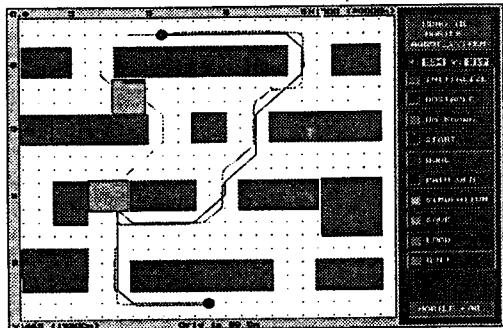


그림 10. 경로를 재설정한 장애물의 회피

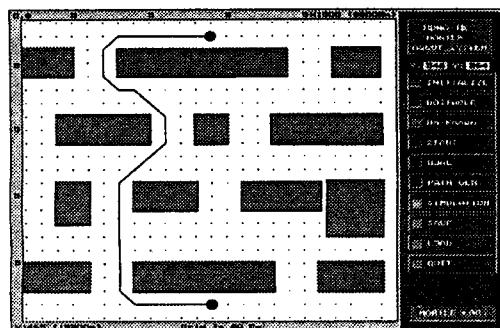


그림 7. 전체지도 작성을 마치고 경로발생한 화면

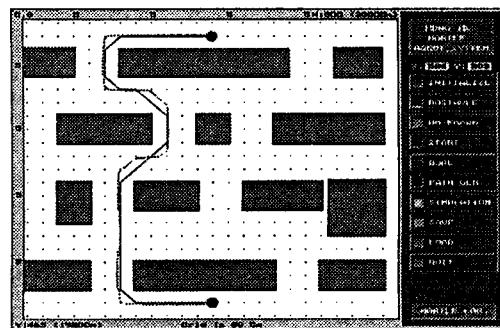


그림 8. 입력환경 변수들을 인식과 장애물회피

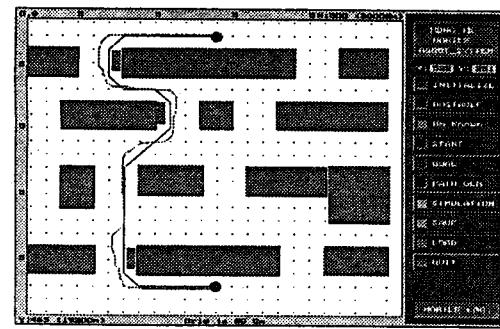


그림 9. 알려지지 않은 장애물의 회피

5. 결론

본 논문에서는 자율주행 로봇의 경로계획을 Global Map 작성과 Local Map작성을 유기적으로 결합하는 새로운 방식을 도입하였다. Global Map작성을 위한 대표적인 알고리즘의 하나인 거리변환법의 경우에는 이미 설정된 Cell들의 특징만을 추출하기 때문에 입력된 장애물의 인식에 있어서 낮은 해상도로 장애물을 정확하게 인식하지 못하는 단점이 있었다. 또한 Local Map작성을 위한 VFN의 경우에도 알려진 장애물임에도 불구하고 “그” 형태의 장애물에서 Deadlock현상으로 시스템의 성능을 현저히 저하시키는 결과를 초래하였다. 따라서 본 논문에서는 위의 두가지 방법의 장점만을 도입하여 효과적으로 장애물을 회피하여 경로계획을 할 수 있는 알고리즘을 구현하였다. 결과를 입증하기 위해 시뮬레이션 과정을 통해 효용성을 보였으며, 실 시스템에 적용하여 경로계획과 장애물 회피를 할 수 있는 효과적인 알고리즘임을 알 수 있다.

6. 참고문헌

- [1] Mckerrow Philip, *Introduction to Robotics*, Addison Wesley, pp. 431-476, 1993.
- [2] Alexander Zelinsky, "A Mobile Robot Exploration Algorithm," *IEEE J. Robotics and Automation*, Vol. 8, No. 6, pp. 707-717. 1992.
- [3] Yun-Hui Liu and Suguru Arimoto, "Path Planning Using a Tangent Graph for Mobile Robots Among Polygonal and Curved Obstacles," *Int. J. Robotics Research*, Vol. 11, No. 4, pp. 376-382. August 1992.
- [4] V.J. Lumelsky and A.A. Stepanov, "Path Planning Strategies for a Point a Mobile Automation Amidst Unknown obstacle of Arbitrary Shape," *Algorithmica*, Vol. 2, No. 4, pp. 403-430, 1987.
- [5] Johann Borenstein and Yoram Koren, "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robot," *IEEE Trans. on Robotics and Automation*, Vol. 7, No. 3, pp. 278-288, June 1991.
- [6] Johann Borenstein and Yoram Koren, "Real-Time Obstacle Avoidance for Fast Mobile Robot," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 19, No. 5, pp. 1179-1187, Sept./Oct. 1989.
- [7] 임 종환, 조동우, "초음파 센서를 이용한 자율 이동 로봇의 써론티 그리드 형성", 전기학회 논문지, 39권 4호, 1990년 4월.
- [8] Johann Borenstein and Yoram Koren, "Histogramic In-Motion Mapping for Mobile Robot Obstacle Avoidance," *IEEE Trans. on Robotics & Automation*, Vol. 7, No. 4, pp. 535-539, August 1991.