

## 광류 정보를 이용한 이동 로봇의 장애물 회피 항법

이한식<sup>1</sup> · 백준걸<sup>2\*</sup> · 장동식<sup>3</sup>

<sup>1</sup>(주)오픈비주얼 / <sup>2</sup>인덕대학 산업시스템경영과 / <sup>3</sup>고려대학교 산업시스템정보공학과

### Obstacle Avoidance for a Mobile Robot Using Optical Flow

Han-Sik Lee<sup>1</sup> · Jun-Geol Baek<sup>2</sup> · Dong-Sik Jang<sup>3</sup>

<sup>1</sup>OpenVisual, Inc., Seoul, 136-701

<sup>2</sup>Department of Industrial Systems Engineering, Induk Institute of Technology, Seoul, 139-749

<sup>3</sup>Department of Industrial Systems and Information Engineering, Korea University, Seoul, 136-701

This paper presents a heuristic algorithm that a mobile robot avoids obstacles using optical flow. Using optical flow, the mobile robot can easily avoid static obstacles without a prior position information as well as moving obstacles with unknown trajectories. The mobile robot in this paper is able to recognize the locations or routes of obstacles, which can be detected by obtaining 2-dimensional optical flow information from a CCD camera. It predicts the possibilities of crash with obstacles based on the comparison between planned routes and the obstacle routes. Then it modifies its driving route if necessary. Driving acceleration and angular velocity of mobile robot are applied as controlling variables of avoidance. The corresponding simulation test is performed to verify the effectiveness of these factors. The results of simulation show that the mobile robot can reach the goal with avoiding obstacles which have variable routes and speed.

**Keywords:** obstacle avoidance, optical flow, mobile robot, fuzzy logic

#### 1. 서론

최근 들어 사람이 작업하기 힘든 작업 공간에서 사람을 대신 하여 작업을 수행할 수 있는 무인화에 대한 연구가 활발히 진행되고 있으며 무인화 설비의 핵심으로 사용되는 로봇에 대한 많은 연구가 진행되고 있다. 특히 작업환경이 열악하여 사람이 작업을 수행하기 어려운 작업장을 이동하며 작업을 수행할 수 있는 무인 이동 로봇(autonomous mobile robot)에 대한 연구의 필요성이 증가하고 있다.

이동 로봇의 종류는 생산 현장에서 많이 볼 수 있는 자재 운반용 무인운반차(AGV), 위험성이 높은 지역에서 사용되는 작업 로봇, 사람이 작업할 수 없는 환경에서 사용되는 작업 로봇, 사무용 빌딩에서 문서 운반용으로 사용되는 로봇 등 매우 다양하다. 현재 가장 많이 사용되고 있는 산업용 이동 로봇은 대부분이 유선 안내 시스템(wire-guided system)으로서 작업장 바

다에 묻혀 있는 유도 전선을 따라서 이동 로봇이 이동하는 형태를 갖는다(Larcombe, 1981; Sonka *et al.*, 1993). 그러나 이러한 유선 안내 시스템은 고정된 경로를 갖기 때문에 작업의 범위가 한정되어 있고 경로가 바뀔 때마다 안내선을 다시 만들어야 하는 단점이 있다. 따라서 진정한 의미의 무인화를 위해서는 안내선의 필요없이 자유로운 경로를 갖는 이동 로봇의 개발이 필수적이다.

자유 경로를 갖는 이동 로봇의 개발을 위해서는 목적지까지의 최적 경로를 결정하는 문제와 장애물이 나타났을 때 이를 회피하는 문제 등 해결해야 할 많은 문제가 발생한다. 최적 경로 결정 문제의 경우 장애물이 고정되어 있는 경우에 대해 인공지능을 이용한 많은 알고리즘이 개발되었다(Beom and Cho, 1992; Brooks, 1986; Moravec and Elfes, 1985). 그러나 장애물이 이동하는 경우에 대한 연구는 부족한 상태이며, 장애물이 이동하는 경우 이동 로봇의 최적 경로 결정 문제는 결국 이동하는 장애물의 회피(obstacle avoidance) 문제와 밀접한 관계를 갖

\* 연락저자: 백준걸 교수, 139-749 서울특별시 노원구 월계동 산76 인덕대학 산업시스템경영과, Fax: 901-7660, e-mail: jungeol@hanmail.net  
2001년 8월 접수, 1회 수정(1일 소요) 후, 2001년 11월 게재 확정.

는다. 특히 이미 계획된 경로상에 미지의 장애물이 갑자기 나타나는 경우와 같이 복잡하고 빠르게 변하는 환경에서 장애물을 회피하기 위해서는 장애물 탐지(obstacle detection)가 무엇보다도 중요한 문제가 된다.

이동 로봇이 움직이는 장애물을 효율적으로 회피하기 위해서는 장애물 탐지에 소요되는 시간을 단축하는 것이 중요하기 때문에 장애물 탐지를 위해 사용되는 센서는 이동 로봇이 장애물을 회피하는 데 있어서 충분한 시간을 가질 수 있도록 충분히 넓게 탐지할 수 있어야 한다. 장애물 탐지에 사용되는 센서는 초음파 센서, 레이저 센서, 시각 센서 등이 가장 광범위하게 사용된다.

이 중 초음파 센서와 레이저 센서는 짧은 시간 안에 장애물의 출현 여부를 판단할 수 있는 장점을 지니고 있지만 장애물의 이동방향이나 속도, 크기 등에 대한 해석력이 낮기 때문에 효과적인 장애물 회피를 수행할 수 없는 단점을 지닌다(Beom and Cho, 1992). 이에 반해 시각 센서는 초음파 센서나 레이저 센서에 비해 보다 넓은 범위에 대한 정보를 제공할 수 있고, 장애물의 위치와 크기 등을 탐지할 수 있다는 장점을 지닌다(Borenstein and Koren, 1989; Enkelmann, 1991).

그러나 카메라를 이용하여 장애물의 위치와 크기 등을 탐지하기 위해서는 두 대 이상의 카메라를 이용하여 얻어지는 스테레오 영상 정보를 분석해야 하기 때문에 많은 계산 시간을 필요로 하는 문제점이 발생한다. 특히 장애물이 고정되어 있지 않고 이동하는 경우 실시간 처리의 한계로 인해 실질적인 장애물 회피 방법으로 적용되기 어렵다(Beaufre and Zeghloul, 1995). 하지만 광류(optical flow) 정보를 이용하는 경우 한 대의 카메라에서 얻어진 연속된 영상 정보를 이용하여 장애물의 위치, 크기 및 이동방향을 탐지할 수 있기 때문에 장애물 탐지 및 회피를 위한 현실성 있는 알고리즘을 개발할 수 있다(Camus, 1994).

본 연구에서는 한 대의 카메라에서 얻어진 연속된 영상 정보를 토대로 광류를 계산하고, 이로부터 장애물의 위치와 이동에 관한 정보를 도출하는 방법을 제시하고자 한다. 또한 도출된 장애물의 위치와 이동 정보, 그리고 이동 로봇의 운행 정보를 기반으로 퍼지논리 제어기(fuzzy logic controller)를 이용하여 움직이는 장애물이 존재하는 환경에서 효율적인 장애물 회피를 위한 이동 로봇의 운행 제어 방법을 제시하고자 한다.

## 2. 광류를 이용한 장애물 회피 시스템

본 연구에서는 움직이는 장애물이 존재하는 환경에서 이동 로봇이 장애물을 효율적으로 회피할 수 있도록 하기 위한 방법으로 광류 정보와 퍼지논리 제어기를 이용한 이동 로봇의 운행 제어 방법을 제시한다. 본 연구에서 제시하는 이동 로봇의 운행 제어 시스템은 <그림 1>과 같은 구조를 지니고 있으며 구성 요소별 특징은 다음과 같다.

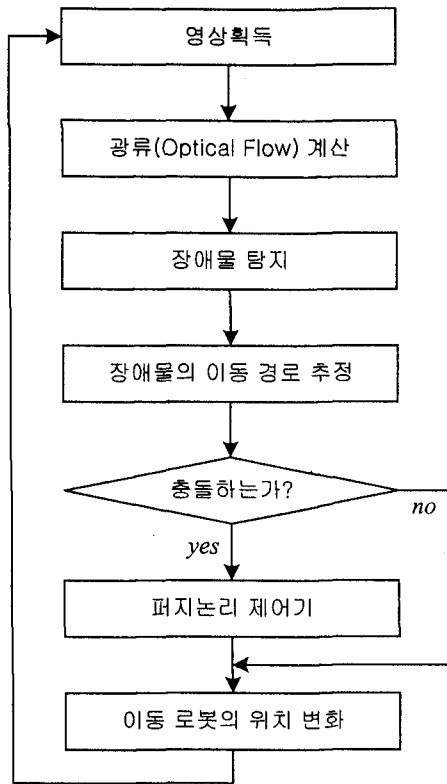


그림 1. 이동 로봇의 운행 제어 시스템.

### • 영상 획득

한 대의 카메라에 의해 아주 짧은 시간 간격 사이에 잡힌 연속된 두 개의 정지 영상으로부터 광류를 검출한다. 본 연구에서는 여기서 얻어지는 광류 정보를 바탕으로 장애물을 탐지하고 장애물의 이동 경로를 추정한다.

### • 광류 계산

광류 계산은 두 개의 연속된 정지 영상을 토대로 각 화소(pixel)의 흐름을 계산하여 광류 정보를 도출한다. 광류 계산을 통해서 얻어지는 정보는 장애물의 크기, 위치, 이동 속도 및 이동 방향 등이다. 광류 계산은 한 대의 카메라를 이용한다는 장점을 지니지만 모든 화소에 대한 흐름을 계산해야 하기 때문에 실시간 처리에 어려움이 있었다. 그러나 Camus(1994)가 제안한 광류 계산 알고리즘을 이용하는 경우 실시간으로 장애물을 탐지할 수 있다.

### • 충돌여부 판단

광류 계산에 의해 도출된 움직이는 장애물의 크기, 위치, 이동 속도 및 이동 방향 정보를 이용하여 주행중인 이동 로봇과 장애물의 충돌 여부를 예측할 수 있다. 만약 탐지된 장애물이 이동 로봇과 충돌할 가능성이 있는 경우 퍼지논리 제어기를 이용하여 이동 로봇의 장애물 회피를 수행하게 된다.

### • 퍼지논리 제어기를 이용한 장애물 회피

본 연구에서는 광류 계산에 의해 도출된 광류 정보와 장애

물의 충돌여부 정보를 이용하여 이동 로봇이 움직이는 장애물을 회피할 수 있는 퍼지논리 제어를 제시한다. 퍼지 제어를 위해 사용되는 입력변수는 광류의 크기, 화면상의 높이, 입사각, 장애물의 충돌 위치 등이며, 퍼지 제어 결과 출력되는 값은 이동 로봇이 장애물을 회피하기 위한 회피 각도와 속도이다.

## 2.1 광류를 이용한 장애물 탐지

### 2.1.1 영상 획득

본 연구에서는 움직이는 장애물에 대한 정보를 얻기 위해 CCD 카메라를 사용한다. 카메라는 이동 로봇의 상단 중앙부에 부착되며, 카메라의 높이와 카메라가 지면과 이루는 각도 등이 중요한 고려 사항이 된다. 카메라의 높이가 높으면 지면과의 거리가 멀어지게 되므로 넓은 영역에 대한 영상을 얻을 수 있지만, 장애물에 대한 상세한 정보는 얻을 수 없게 된다. 반면에 카메라의 높이가 낮으면 지면과의 거리가 가까워지므로 장애물에 대한 상세한 정보를 얻을 수 있지만, 한정된 좁은 영역에 대한 정보만을 얻게 된다는 단점이 있다. 또한 카메라가 지면과 이루는 각도가 크면 전체적인 구조를 한꺼번에 볼 수 있지만 좁은 영역만을 볼 수 있다는 단점을 지니며, 카메라가 지면과 이루는 각도가 작으면 넓은 영역을 볼 수 있지만 앞에서 본 영상이므로 부분적으로 가려진 물체 등이 있을 수 있어 제한된 영역밖에 볼 수 없다는 단점을 지닌다. 따라서 카메라의 적절한 높이와 각도를 결정하기 위한 연구가 필요하며, 본 연구에서는 모의 실험을 통하여 적절한 값을 결정한다.

### 2.1.2 광류 정보의 추출 및 충돌 예측

생물의 시각 시스템은 이산적이긴 하지만, 광강도에 대한 정량화(quantization)는 매우 세밀하기 때문에 거의 연속적인 광강도를 인식할 수 있다. 세밀한 광강도를 가진 두 개의 연속된 영상의 변화를 연속적 흐름으로 나타낸 정보를 광류라고 한다(Horn and Schunck, 1983).

광류, 즉 순간적인 속도장(velocity field)은 영상면의 모든 화소에서의 2차원 속도 벡터를 결정하며, 속도 벡터는 시각장에서 각 점의 움직이는 속도와 방향을 나타낸다.

광류를 계산하기 위한 방법의 하나로써 움직이는 물체 표면의 각 점에 대한 광강도를 위치와 시간의 함수인  $f(x, y, t)$ 로 표시한다. 짧은 시간 간격  $dt$  후의 영상에서 움직이는 물체의 한 점이  $x$  축,  $y$  축으로  $dx, dy$  만큼 이동하였다고 가정하면, 이동된 점의 광강도는 이동하기 전의 광강도와 근사하다는 사실을 가정할 수 있고 이는 식 (1)과 같이 표현할 수 있다.

$$f(x+dx, y+dy, t+dt) \cong f(x, y, t) \quad (1)$$

식 (1)에서 좌변을 Taylor 시리즈로 전개하면 식 (2)와 같이 표현할 수 있다.

$$f(x+dx, y+dy, t+dt) =$$

$$f(x, y, t) + \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial t} dt + \text{고차항들} \quad (2)$$

식 (2)에서 고차항들은 매우 작은 값이므로 무시한다면, 식 (1)과 식 (2)를 통해 식 (3)을 도출할 수 있다.

$$\frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial t} dt = 0 \quad (3)$$

식 (3)은 식 (4)와 같이 변환될 수 있다.

$$-\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x} \frac{dx}{dt} + \frac{\partial f}{\partial y} \frac{dy}{dt} \quad (4)$$

식 (4)에서  $\frac{\partial f}{\partial t}$ ,  $\frac{\partial f}{\partial x}$ ,  $\frac{\partial f}{\partial y}$ 는 모두 측정 가능한 값들이고,  $\frac{dx}{dt}$ ,  $\frac{dy}{dt}$ 는  $x$  축,  $y$  축 방향으로의 운동 속도를 말한다. 따라서  $\frac{dx}{dt} = u$ ,  $\frac{dy}{dt} = v$ 와 같이 표기하면 식 (4)는 식 (5)와 같이 변환될 수 있다.

$$-\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x} u + \frac{\partial f}{\partial y} v \quad (5)$$

식 (5)는 식 (6)과 같이 간단하게 정리될 수 있으며, 식 (6)을 광류 공식이라 부른다.

$$-\frac{\partial f}{\partial t} = \nabla f \cdot \nu \quad (6)$$

식 (6)에서  $\nabla f$ 는 영상의 공간적 변화율을 나타내고,  $\nu$ 는  $\nu = (u, v)$ 로서 속도를 나타낸다. 식 (6)이 의미하는 것은 영상에 나타난 물체의 한 점에 있어서 광강도의 변화는 그 영상의 공간적 변화율과 속도를 곱한 것과 같다는 것이다.

일반적으로 광류를 계산하기 위해서는 연속적인 영상에 존재하는 모든 화소에 대한 흐름을 계산해야 하기 때문에 실시간으로 정확한 광류를 계산한다는 것은 매우 어렵다. 그러나 어떤 가정이 주어진다면 광류 계산은 실시간에 효율적으로 계산될 수 있다. 예를 들어, Camus (1994)의 실시간 광류 계산 알고리즘은 식 (7)과 같은 속도( $\nu$ ), 거리( $d$ ), 시간( $t$ ) 사이의 간단한 관계를 이용한다.

$$\Delta \nu = \frac{\Delta d}{\Delta t} \quad (7)$$

일반적으로 속도 변수를 찾기 위해, 프레임 간의 간격을  $\delta t$  라는 상수로 놓고, 거리 변수(픽셀 이동)에 대한 탐색을 실시한다면 속도 변수는 식 (8)과 같이 정의될 수 있다.

$$\Delta \nu = \frac{\Delta d}{\delta t} \quad (8)$$

또한 이동 거리를  $\delta d$  라는 상수로 놓고 시간 간격 변수에 대한 탐색을 실시하면 속도 변수는 식 (9)와 같이 정의될 수 있다.

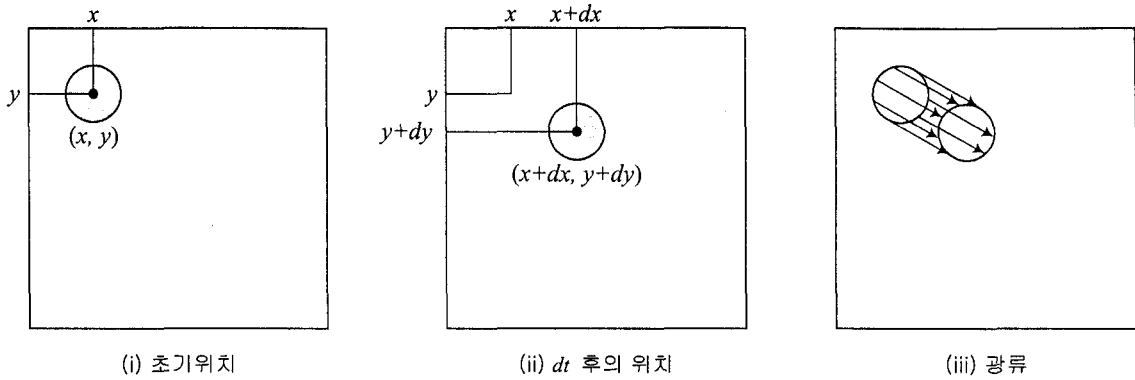


그림 2. 광류 계산.

$$\Delta v = \frac{\delta d}{\Delta t} \quad (9)$$

이러한 시간-공간의 trade-off는 공간적인 2차원 탐색을 시간적인 선형 탐색으로 줄어들게 함으로써 매우 빠른 실시간 구현을 가능하게 한다.

두 개의 연속된 영상을 예로 들면 해당 광류는 <그림 2>와 같이 표현될 수 있다(Khatib, 1986). 광류는 상대적인 움직임에 관한 정보를 나타내는 것으로써 움직이는 장애물의 속도와 방향 정보, 그리고 이동 로봇 자신의 속도와 방향 정보가 결합된 유용한 정보이다. 장애물이 움직이고 카메라를 장착한 이동 로봇도 움직이는 경우, 화면에 나타난 장애물의 광류는 장애물의 운동에 의한 광류 성분과 이동 로봇의 운동에 의한 광류 성분의 벡터 합으로 나타난다.

또한 광류는 이동 로봇과 장애물의 이동 외에 이동 로봇에 장착된 카메라의 높이와 카메라가 지면과 이루는 각도에 대한 정보도 포함한다. 따라서 충돌 여부를 검사하기 위해서는 움직이는 장애물과 이동 로봇의 현재 속도와 운동 방향을 각각 계산하여 예측하는 것보다 광류의 성질을 이용하는 것이 충돌 여부를 쉽게 예측할 수 있다.

<그림 3>은 실세계 좌표계상에서 장애물과 이동 로봇이 각각 주행중일 때, 이동로봇에 부착된 카메라의 화면상에 장애물의 운동이 어떻게 나타나는가를 설명한다. 즉, 광류의 연장선을 그어 충돌 영역 내로 들어오는 경우, 현재의 주행 상태를 유지했을 때 장애물이 이동 로봇과 충돌할 것이라고 판단한다.

### 2.1.3 장애물 형태 보정

본 연구에서는 장애물의 형태를 <그림 4>와 같이 가로, 세로 비율을 고려하여 긴 쪽을 지름으로 하는 구의 형태로 단순화 하였다. 단순화된 구는 장애물을 내부에 포함하며 어느 정도의 안전지역까지 확보할 수 있는 장점을 지닌다. 또한 구의 중심을 장애물의 대표값으로 설정하여 중심의 이동만을 고려하면 되므로 계산량을 줄일 수 있다. 그러나 이와 같은 형태의 단순화는 장애물이 이동 로봇과 같이 단순한 형태를 갖는 경우에는 효과적이지만, 어느 일정 부위에 심한 돌출이 존재하는 것과 같은 특이한 형태를 가진 장애물에는 적용할 수 없다는 단점을 지닌다. 본 연구에서는 움직이는 장애물을 다른 이동 로봇으로 가정하기 때문에 장애물을 <그림 4>와 같은 형태로 단순화할 수 있다.

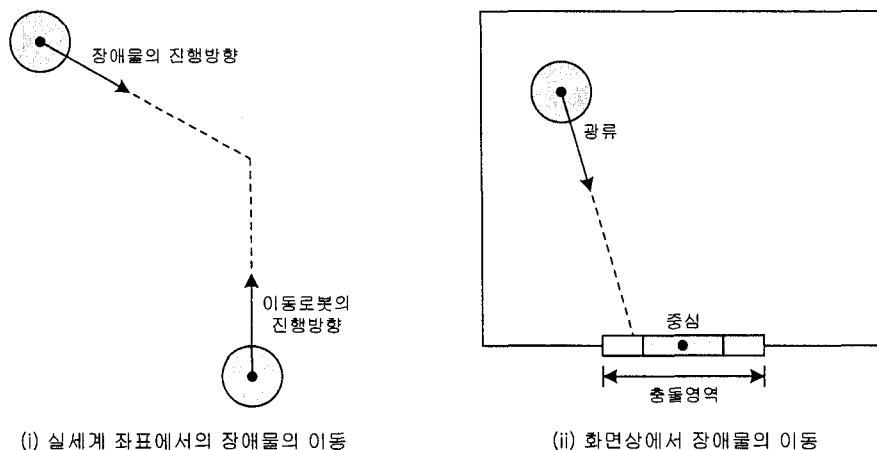


그림 3. 광류의 특성.

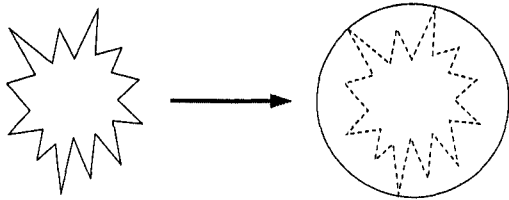


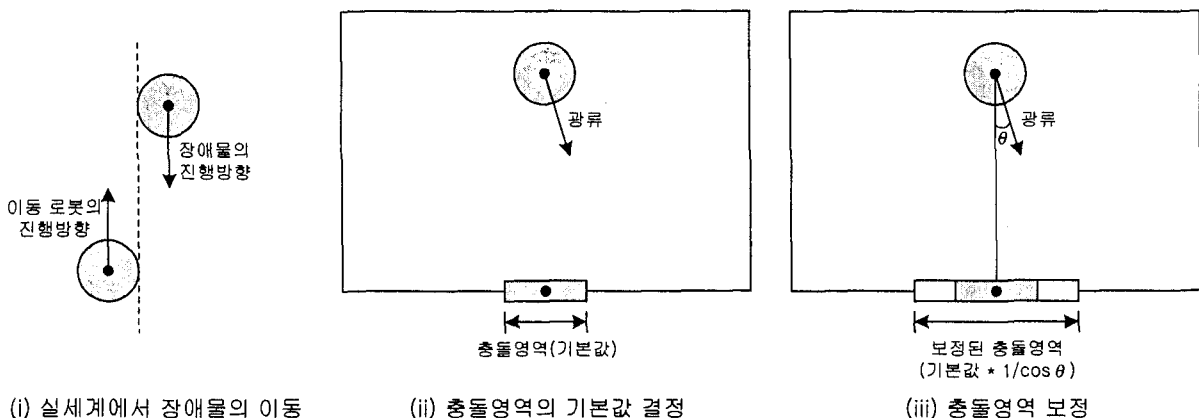
그림 4. 장애물 형태 보정.

2.1.4 충돌 영역 설정

장애물과 이동 로봇의 충돌여부를 판단하기 위해 사용되는 충돌 영역은 이동 로봇에 장착된 카메라의 높이, 카메라가 지면과 이루는 각도, 장애물의 광류가 영상의 수직선과 이루는 각도, 장애물의 크기 등에 의해 영향을 받는다.

본 연구에서는 이동 로봇의 주행시, 카메라의 높이와 각도가 미리 설정된 값으로 고정되어 있다고 가정하고, 장애물의 종류를 이동 로봇으로 가정하기 때문에 충돌 영역의 크기는 장애물의 광류가 화면상의 수직선과 이루는 각도에만 영향을 받는다. <그림 5>의 (i)에서와 같이 장애물이 정면에서 이동 로봇에 평행하게 접근할 경우 화면상에 나타난 광류는 (ii)와 같다. 충돌 영역을 설정하기 위해서는 먼저 <그림 5>의 (ii)에서와 같이 화면상에 표현되는 이동 로봇의 크기를 충돌 영역의 기본값으로 설정하고, (iii)에서와 같이 광류가 화면상의 수직선과 이루는 각도( $\theta$ )에 따라 기본값에  $1/\cos\theta$ 를 곱하여 충돌 영역의 크기를 보정한다. 예를 들어,  $\theta$ 가  $60^\circ$  라면 충돌 영역은 기본값의 2배( $1/\cos 60^\circ = 2$ )가 된다.

광류 계산을 통해 알 수 있는 정보는 화면상에 나타난 장애물의 크기와 위치, 그리고 광류 정보뿐이다. 따라서 이 세 가지 정보만을 가지고 장애물의 운동을 정확히 예측하여 충돌을 회피할 수 있는 이동 로봇의 방향과 속도를 결정하는 데는 여러 가지 제약이 따른다. 따라서 본 연구에서는 입력값의 불확실성이 존재하는 경우에도 적절한 출력값을 제공할 수 있는 퍼지논리 제어기를 사용하여 충돌회피를 위한 이동 로봇의 운동(방향, 속도)을 결정할 수 있는 방법을 제시한다.



(i) 실세계에서 장애물의 이동

(ii) 충돌영역의 기본값 결정

(iii) 충돌영역 보정

그림 5. 충돌영역 설정.

2.2 퍼지논리 제어기를 이용한 장애물 회피

본 연구에서는 움직이는 장애물의 2차원 영상에서 추출된 광류 정보를 이용하여 이동 로봇이 장애물을 회피하면서 원하는 목적지를 찾아갈 수 있도록 제어할 수 있는 퍼지논리 제어기를 제시한다. 퍼지논리 제어기는 부정확하고 애매한 데이터를 처리할 수 있고, 기존의 수학적 모델에 기초한 제어기가 해결하기 어려운 복잡한 문제를 해결할 수 있는 실시간 결정 능력(real-time decision) 능력을 지니고 있기 때문에 실시간 자율 행동 결정 능력(real-time autonomous behavior decision ability)이 필수적인 움직이는 장애물이 존재하는 환경에서 이동 로봇의 경로계획 문제를 풀기 위한 방법으로 다양하게 적용되어 왔다 (Song and Tai, 1992).

퍼지논리 제어기는 입력변수 공간(input space)과 출력변수 공간(output space) 사이의 사상을 표현하는 데 사용되며, 퍼지 규칙 베이스는 입력 변수 공간과 출력변수 공간을 각각 양자화하는 입·출력 퍼지집합에 의해 만들어진다.

본 연구에서 제안하는 퍼지논리 제어기는 <그림 6>과 같은 구조를 지닌다.

2.2.1 퍼지논리 제어기의 구조

불확실한 환경에서 이동 로봇의 운행 목적은 이동 로봇이

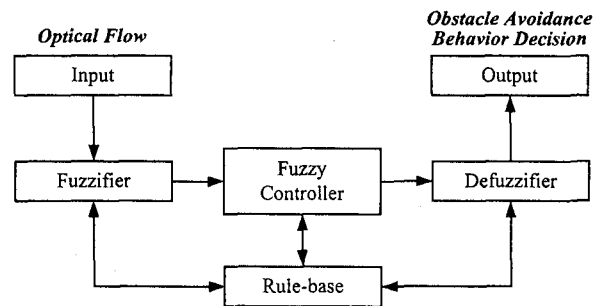


그림 6. 퍼지논리 제어기.

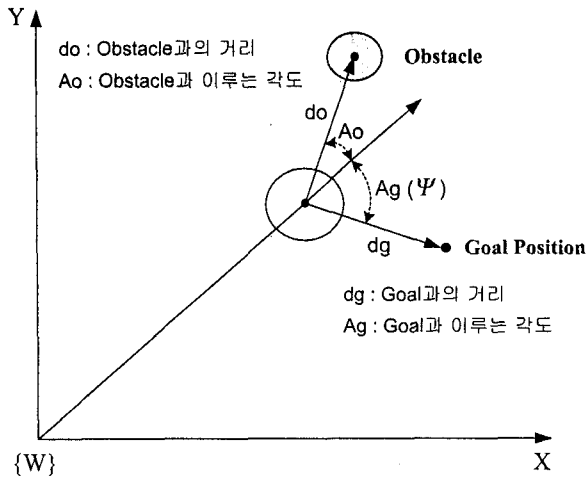


그림 7. 이동 로봇과 장애물과의 관계.

장애물과 충돌하지 않고 원하는 목적지에 도달할 수 있도록 하는 것이다. 이러한 목적을 효과적으로 달성하기 위해서는 <그림 7>에 나타난 실세계 좌표계  $\{W\}$ 에서 장애물의 위치와 이동 로봇의 위치에 따라 이동 로봇의 운행 방향과 속도를 조절해야 한다.

장애물을 회피하기 위한 이동 로봇의 움직임은 회피 각도 ( $\Delta\phi$ )와 속도 ( $\Delta v$ )의 조절에 의해 제어될 수 있다. 따라서 본 연구에서는 이동 로봇이 장애물을 회피하면서 목적지에 도달할 수 있도록 제어하는 퍼지논리 제어기의 입력변수로서 광류의 크기( $s$ ), 광류의 입사각( $\theta$ ), 장애물이 화면상에 위치하는 높이( $d$ ), 충돌 위치( $p$ ), 목적지와의 각도( $\psi$ )를 설정하였고, 출력변수로는 이동로봇이 장애물을 회피하기 위한 회피 각도와 속도를 설정하였다. 퍼지논리 제어기를 구성하는 구성요소에 대한 자세한 설명은 다음과 같다.

(1) 입력 변수

① 광류의 크기( $s$ )

광류의 크기는 이동 로봇과 장애물 사이의 상대적인 접근 속도에 대한 2차원 이미지상의 표현이며, 크기가 클수록 이동 로봇과 장애물은 빠른 속도로 가까워지고 있음을 나타낸다. 그러나 동일한 크기의 광류라도 화면상의 높이에 따라 접근속도가 달라지기 때문에 광류의 크기만으로는 접근속도를 완벽하게 표현할 수 없다. 즉, 광류가 동일한 크기로 나타날 때, 화면상의 위쪽에 위치할수록(장애물과의 거리가 멀수록) 더 빠른 접근속도를 나타내며, 화면상의 아래쪽에 위치할수록(장애물과의 거리가 가까울수록) 더 느린 접근 속도를 나타낸다. 다시 말하면 장애물과 이동 로봇이 일정한 속도로 가까워지고 있을 때, 화면 위쪽에서는 광류가 작게 표시되고 화면 아래쪽에서는 크게 표시된다. 따라서 이러한 현상을 고려하기 위하여 광류가 화면상에 위치하는 높이를 하나의 입력변수로 선정해야 한다.

② 화면상의 높이( $d$ )

이동 로봇과 장애물 사이의 거리를 정확하게 계산하기 위해서는 거리에 대한 지표로써 광류가 화면상에 위치하는 높이를 입력변수로 사용해야 한다. 즉, 장애물의 광류가 화면상에서 위쪽에 위치할수록 이동 로봇과의 거리는 멀다고 판단할 수 있다. 광류의 크기와 화면상의 높이는 이동 로봇과 장애물의 충돌 여부를 나타내는 중요한 변수로써 이동 로봇의 행동을 결정하는 중요한 기준이 된다.

③ 광류의 입사각( $\theta$ )

광류가 영상의 수직선과 이루는 각도를 나타내는 값으로써 이동 로봇이 장애물을 어느 방향으로 회피할 것인가를 결정하는 데 필요한 변수이다.

④ 충돌 위치( $p$ )

이동 로봇과 장애물이 현재의 속도와 방향을 그대로 유지하면서 진행할 경우, 장애물이 이동 로봇상에 부딪히게 되는 위치를 추정할 값으로써 이동 로봇의 앞면 중심을 기준으로 계산한다. 충돌 위치는 입사각과 함께 이동 로봇이 장애물을 어느 방향으로 회피할 것인가를 결정하는 데 필요한 변수이다.

⑤ 목적지와의 각도( $\psi$ )

현재 이동 로봇이 진행하는 방향과 목적지 방향이 이루는 각도를 말한다. 이동 로봇이 장애물과 충돌할 가능성이 없다고 판단되었을 때는 목적지로 빠르게 이동하기 위해서 목적지와의 각도에 따라 회피각도가 결정된다.

(2) 출력 변수

① 회피 각도( $\Delta\phi$ )

이동 로봇이 장애물과 충돌하는 것을 방지하기 위해 현재의 진행 방향에서 어느 방향으로 얼마만큼의 각도를 틀어야 하는지를 나타내는 변수로써 주로 입사각과 충돌 위치에 따라 회피 각도가 결정된다.

② 속도( $\Delta v$ )

이동 로봇이 장애물과 충돌하는 것을 방지하기 위해 속도를 줄여야 할 필요가 있을 때 속도를 얼마만큼 줄여야 하는지를 나타내는 변수로써, 장애물과 충돌할 가능성이 매우 적거나 장애물을 회피할 충분한 시간적 여유가 있을 때에는 이동 로봇의 속도를 원래의 평균 속도만큼 향상시킬 수 있어야 한다.

(3) 퍼지 연산자

퍼지 연산자는 입력 변수의 값( $Z$ )을 언어 변수( $\hat{Z}$ )로 변환하는 역할을 수행하며, 퍼지 집합의 이름표(label)로 사용되는  $\hat{Z}$ 는  $fuzzifier(Z)$ 로 정의될 수 있다. 본 연구에서는 각각의 입력변수에 대한 언어 변수를 <표 1>과 같이 정의하였고, 소속함수를 <그림 8>과 같이 정의하였다.

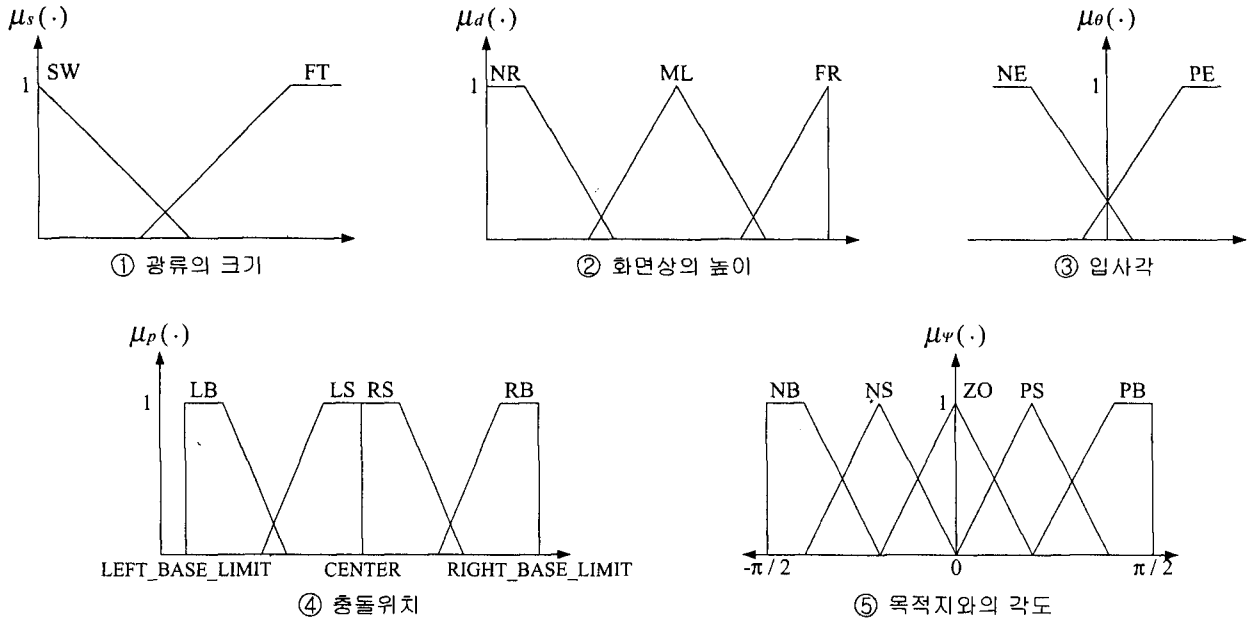


그림 8. 퍼지 집합의 소속 함수( $\mu_z(\cdot)$ ).

표 1. 입력 변수들의 언어 변수

크기( $s$ )	화면상의 높이( $d$ )		입사각 ( $\theta$ )		충돌위치 ( $p$ )		목적지와의 각도( $\psi$ )	
	빠름	멀다	NE	음	LB	좌측	NB	큰음수
FT	FR	ML	PE	양	LS	약간 좌측	NS	작은 음수
SW	NR	가깝다			RS	약간 우측	ZO	일치
					RB	우측	PS	작은 양수
							PB	큰양수

표 2. 출력 변수들의 언어 변수

회피 각도( $\Delta\phi$ )		속도( $\Delta v$ )	
LB	좌측	NB	작게
LS	약간 좌측	NE	조금 작게
CT	유지	ZO	등속
RS	약간 우측	PE	조금 크게
RB	우측		

2.2.2 퍼지제어 규칙 구축

본 연구에서는 입력 변수값을 기반으로 효율적인 장애물 회피를 위한 이동 로봇의 제어를 위해 IF-THEN 표현 형태를 갖는 퍼지 규칙(fuzzy rule)을 다음과 같이 정의하였다.

첫 번째 부류의 규칙은 장애물이 먼 거리에서 느린 속도로

접근하는 경우에 대한 퍼지 규칙을 나타낸 것으로써, 이때는 장애물의 회피보다는 목적지를 찾아가는 행동이 더 중요하다고 판단되기 때문에 광류의 크기( $s$ ), 화면상의 위치( $d$ ), 그리고 목적지와의 각도( $\psi$ )에 따라 이동 로봇의 회피 각도( $\Delta\phi$ )를 결정하였다.

IF ( $s=SW$ ) AND( $d=FR$ ) AND( $\psi=NB$ )  
THEN( $\Delta\phi=LB$ )

IF ( $s=SW$ ) AND( $d=FR$ ) AND( $\psi=NS$ )  
THEN( $\Delta\phi=LS$ )

IF ( $s=SW$ ) AND( $d=FR$ ) AND( $\psi=ZO$ )  
THEN( $\Delta\phi=CT$ )

IF ( $s=SW$ ) AND( $d=FR$ ) AND( $\psi=PS$ )  
THEN( $\Delta\phi=RS$ )

IF ( $s=SW$ ) AND( $d=FR$ ) AND( $\psi=PB$ )  
THEN( $\Delta\phi=RB$ )

두 번째 부류의 규칙은 장애물을 회피하기 위한 충분한 시간적 여유를 지닌 경우에 대한 퍼지 규칙을 나타낸 것으로써, 이동 로봇의 회피각도와 속도를 조금만 변화시키더라도 여유 있게 충돌을 회피할 수 있다.

IF ( $s=FT$ ) AND( $d=FR$ ) AND( $\theta=NE$ ) AND( $p=RB$ )  
THEN( $\Delta\phi=LS$ ) AND( $\Delta v=NE$ )

IF ( $s=FT$ ) AND( $d=FR$ ) AND( $\theta=NE$ ) AND( $p=LB$ )  
THEN( $\Delta\phi=RS$ ) AND( $\Delta v=ZO$ )

IF ( $s=FT$ ) AND( $d=FR$ ) AND( $\theta=NE$ ) AND( $p=RS$ )

THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{FR}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{FR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{ZO}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{FR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{FR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{FR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{ZO}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{ZO}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NE}$ )

세 번째 부류의 규칙은 장애물이 가까운 거리에 위치하거나 거리는 어느 정도 여유가 있지만 접근 속도가 빠른 경우에 대한 퍼지 규칙을 나타낸 것으로서, 이동 로봇의 속도를 줄여 안전거리를 확보하면서 방향을 전환한다.

IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{NE}$ )

IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{SW}$ ) AND( $d = \text{NR}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{NE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RB}$ )  
 THEN( $\Delta\varphi = \text{LS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LB}$ )  
 THEN( $\Delta\varphi = \text{RS}$ ) AND( $\Delta v = \text{NE}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{RS}$ )  
 THEN( $\Delta\varphi = \text{LB}$ ) AND( $\Delta v = \text{NB}$ )  
 IF( $s = \text{FT}$ ) AND( $d = \text{ML}$ ) AND( $\theta = \text{PE}$ ) AND( $p = \text{LS}$ )  
 THEN( $\Delta\varphi = \text{RB}$ ) AND( $\Delta v = \text{NB}$ )

마지막 부류의 규칙은 장애물이 아주 가까운 거리에서 빠르게 접근하는 경우에 대한 퍼지 규칙을 나타낸 것으로서, 이때는 매우 급박한 상황을 나타내기 때문에 속도를 가능한 크게 줄여서 안전 거리를 확보할 수 있도록 한다.

IF( $s = \text{FT}$ ) AND( $d = \text{NR}$ )  
 THEN( $\Delta v = \text{NB}$ )

### 2.2.3 출력 변수의 비퍼지화

퍼지논리 제어기에 의해 출력된 출력 변수의 값은 언어 변수 형태의 값을 갖기 때문에 이동 로봇의 제어를 위해서는 출력 변수의 전체 집합에서 정의된 퍼지 제어 조작량을 명확한 비퍼지 제어 조작량으로 변환시켜 주는 비퍼지화 작업이 필요하다.

비퍼지화 방법으로는 합성된 출력 변수의 퍼지집합에서 소속 함수값이 가장 큰 부분에 해당되는 명백한 제어값을 제어 입력으로 사용하는 최대값 방법, 소속 함수값이 최대값을 가지는 곳의 값들의 평균을 제어 입력으로 사용하는 최대 평균법, 출력 변수의 퍼지 집합에 대한 무게 중심을 구하여 해당 제어값을 제어 입력으로 사용하는 무게 중심법 등이 있다(Lee and Oh, 1991). 일반적으로 무게 중심법이 다른 방법들에 비해 우월한 성능을 보이기 때문에 본 연구에서는 무게 중심법을



사용하여 비퍼지화를 실시한다. 본 연구에서 사용하는 출력 변수에 대한 비퍼지화 식은 다음과 같다.

$$\frac{\Delta \varphi}{\Delta v} = \frac{\sum_{j=1}^k (\mu(\Delta \varphi_j) \cdot \Delta \varphi_j)}{\sum_{j=1}^k \mu(\Delta \varphi_j)}$$

$$\frac{\Delta v}{\Delta v} = \frac{\sum_{j=1}^k (\mu(\Delta v_j) \cdot \Delta v_j)}{\sum_{j=1}^k \mu(\Delta v_j)}$$

### 3. 모의 실험 및 결과

#### 3.1 모의 실험 모형 및 평가 방법

본 연구에서는 앞에서 제시한 광류 정보와 퍼지논리 제어기를 이용한 장애물 회피 방법에 대한 효율성을 검증하기 위해 <그림 9>와 같은 모의 실험기를 통해 이동 로봇의 장애물 회피에 대한 모의 실험을 실시하였다.

본 연구에서 제안하는 모의 실험기는 <그림 9>에서 제시한 바와 같이 장애물이 이동 로봇의 카메라에 포착되는 영상을 획득하는 투사영상 생성기, 연속된 영상으로부터 장애물의 광류를 계산하는 광류 계산기, 추출된 광류로부터 퍼지논리 제어기의 입력 자료를 추출하는 장애물 이동 정보 추출기, 장애물의 이동정보와 이동 로봇의 이동정보로부터 이동 로봇의 항로변경 정보를 추출하는 항로 제어기, 그리고 시간의 흐름에 따라 변화하는 장애물과 이동 로봇의 위치 및 카메라의 좌표계를 계산하는 좌표 생성기로 구성되어 있다.

모의 실험기를 이용하여 이동 로봇의 장애물 회피에 대한 모의 실험을 실시하기 위해서는 장애물의 항로와 이동 로봇의 시작지점 및 목표지점을 지정하고, 단위 시간의 흐름에 따라 변화하는 상황을 기반으로 제안된 장애물 회피 알고리즘에 따라 결정된 이동 로봇의 항로를 분석하여 여러 가지 상황에 대한 이동 로봇의 장애물 회피 성능을 검증한다. 모의 실험의 환경 변인은 장애물의 수와 이동 경로, 이동 로봇의 시작지점과

목표지점, 이동 로봇의 속도, 가속도, 각속도의 최대 한계값이다. 또한 모의 실험의 제어 변인으로 고려되는 것은 퍼지논리 제어기의 입력 변수에 대한 소속 함수, 이동 로봇의 단위 속도, 단위 가속도, 단위 각속도, 그리고 로봇에 부착된 카메라의 좌표계이다.

모의 실험을 통한 장애물 회피 알고리즘의 성능 평가는 이동 로봇의 시작지점에서 목적지점까지의 항로 길이와 이동시간으로 평가한다. 본 연구에서 제시하는 모의 실험기는 단위 시간별 변화된 이동 로봇의 위치를 저장할 수 있기 때문에 이를 이용하여 식 (10)과 같이 이동 로봇의 항로 길이를 계산할 수 있다.

식 (10)에서  $T$ 는 이동 로봇의 이동시간을 의미하고 식 (11)에서  $(x_t, y_t)$ 는  $t$ 시점에서의 이동 로봇의 위치를 의미한다.

$$D = \sum_{i=1}^T D_i \tag{10}$$

$$D_i = \sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2} \tag{11}$$

장애물이 없는 상황에서 이동 로봇의 시작지점에서 목적지점까지의 항로 길이를  $D_0$  라하고, 장애물이 있는 경우의 항로 길이를  $D_N$  이라 하면, 장애물로 인해 야기된 항로의 변경 길이는  $D_N - D_0$  가 된다. 이 길이는 절대 수치로 시간이 고려되지 않고 있으나 장애물에 의한 항로의 지연 길이를 평가하는 좋은 척도가 된다. 장애물 회피 알고리즘에 대한 성능 평가 척도로서 이동시간을 고려할 경우,  $T_0$ ,  $T_N$ 을 각각 장애물이 없는 경우와 있는 경우의 이동 로봇이 시작지점에서 목적지점까지 운행하는 데 소요되는 시간이라 할 때 장애물 회피에 소요되는 시간은  $T_N - T_0$  로 계산될 수 있다.

#### 3.2 실험 계획

본 실험에서는 먼저 퍼지논리 제어기에서 입력 변수에 대한 소속 함수의 변화가 이동 로봇의 이동 경로에 미치는 영향을 분석하고자 한다. 이 결과를 바탕으로 입력 변수의 소속 함수를 결정한 후, 환경 변인의 변화가 이동 로봇의 주행에 미치는 영향을 분석하기 위해 이동 로봇에 장착된 카메라의 높이와 지면과 이루는 각도, 그리고 이동 로봇의 성능(최대 속도, 최대 가속도, 최대 각속도)을 변화시켜 이에 따른 이동 로봇의 경로 및 기타 특성을 분석한다. 모의 실험을 위해 사용된 이동 로봇의 기본적인 환경 변인은 <표 3>과 같다.

표 3. 이동 로봇의 환경 변인

최대 속도	: 3000 / 30(mm/frame)
최대 가속도	: 최대 속도 / 25(mm <sup>2</sup> /frame)
최대 각속도	: 3° (degree/frame)
카메라의 각도	: 25° (degree)
카메라의 높이	: 1200(mm)
출발점	: (10, 10)(mm)
목적지	: (20000, 20000)(mm)

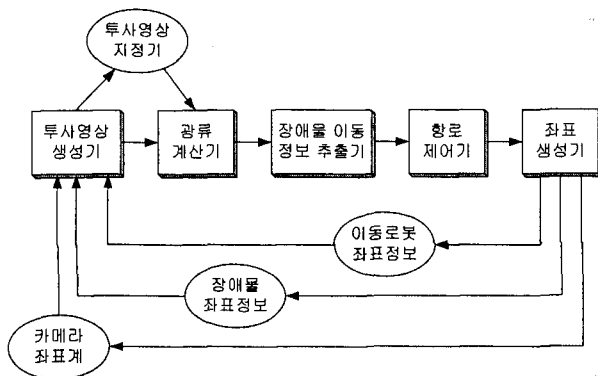


그림 9. 모의 실험기 모형.

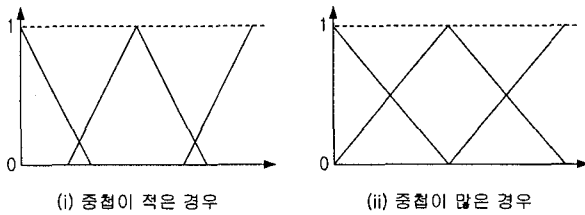


그림 10. 소속 함수의 중첩도 수준.

첫 번째 실험은 입력 변수에 대한 소속 함수의 변화가 이동 로봇의 운행에 미치는 영향을 분석하기 위한 실험으로써 <그림 10>과 같이 소속 함수의 기울기를 조절하여 중첩되는 부분을 적게 하는 경우와 많게 하는 경우에 대해 이동 로봇의 장애물 회피 성능을 평가한다.

소속 함수의 중첩도에 따른 영향을 분석하기 위한 장애물의 유형은 이동 방향에 따라 9가지 유형을 선택하고, 각 유형별로 장애물의 이동 속도를 2가지씩 설정하여, 총 18가지 유형에 대한 실험을 실시한다.

두 번째 실험은 여러 가지 환경 변인들 중에서 최대 속도, 최대 가속도, 최대 각속도, 카메라가 지면과 이루는 각도에 대하여 각각 3가지 수준을 설정하고, 각각의 수준에 대하여 50가지의 장애물 유형을 설정하여 다양한 장애물의 유형에 대한 장애물 회피 실험을 실시한다. 단 이 실험에서 사용되는 소속 함수의 중첩도는 첫 번째 실험의 결과를 바탕으로 설정한다. 실험에 사용되는 50가지 유형의 장애물은 정지 장애물, 등가속도 직선 운동을 하는 장애물, 곡선 운동을 하는 장애물, 등속 직선 운동을 하는 장애물 등 다양한 이동 유형을 갖는다.

표 4. 모의실험 계획표

환경 변인	수준		
	수준 1	수준 2	수준 3
최대 속도	2000/30	3000/30	4000/30
최대 가속도	최대속도/15	최대속도/25	최대속도/35
최대 각속도	1	3	5
카메라의 각도	5	25	45

3.3 실험 결과 및 분석

첫 번째 실험에서는 결과에 큰 차이가 나지는 않았지만, 소속 함수의 중첩이 많은 경우가 적은 경우에 비해 이동 로봇의 항로 변경 길이와 장애물 회피에 소요되는 시간이 작게 나타났다. 따라서 두 번째 실험을 위한 소속 함수의 중첩도 수준은 중첩이 많은 것을 선택하였다. 두 번째 실험에서는 총 4,050가지의 경우에 대해 장애물 회피 실험을 실시한 결과 4,027가지 경우에 대해서는 장애물을 성공적으로 회피하였고, 나머지 23가지 경우에 충돌이 발생하였다. 각 환경 변인별 실험 결과는

표 5. 최대 속도에 대한 실험 결과

	수준 1	수준 2	수준 3
충돌 횟수	2	2	2
평균이동시간	438.3243	297.6790	228.2367
평균이동거리	28153.5781	28156.9185	28192.3045

표 6. 최대 각속도에 대한 실험 결과

	수준 1	수준 2	수준 3
충돌 횟수	3	2	1
평균이동시간	301.7289	297.6790	299.8581
평균이동거리	28158.4512	28156.9185	28154.8489

표 7. 최대 가속도에 대한 실험 결과

	수준 1	수준 2	수준 3
충돌 횟수	2	2	4
평균이동시간	298.0865	297.6790	291.4099
평균이동거리	28154.8933	28156.9185	28154.2175

표 8. 카메라의 각도에 대한 실험 결과

	수준 1	수준 2	수준 3
충돌 횟수	3	2	4
평균이동시간	296.2743	297.6790	300.4609
평균이동거리	28155.6273	28156.9185	28156.6143

- 1) 이동거리는  $\sum \sqrt{\Delta x^2 + \Delta y^2}$  로 계산된 것임.
- 2) 이동시간은 1/30초단위임.

다음과 같다.

<표 5>에서 나타난 바와 같이 최대 속도에 대한 실험에서는 수준별 이동시간의 변화가 뚜렷이 나타났으며, 속도가 커질수록 이동 거리가 증가하는 경향을 보인다. 최대 각속도에 대한 실험은 <표 6>에서와 같이 각속도가 증가할수록 이동 거리가 감소하는 경향을 보인다. 또한 최대 각속도가 클수록 신속하게 경로를 바꿀 수 있기 때문에 충돌 횟수가 줄어들을 알 수 있다. 최대 가속도에 대한 실험 결과는 <표 7>에서 보듯이 최대 가속도가 클수록 이동시간은 점차 줄어들었으며 이동거리에는 큰 영향이 없었다. 카메라가 지면과 이루는 각도와 장애물 회피 능력의 관계는 카메라의 각도가 커지면 획득되는 광류 정보는 이동 로봇과 가까운 영역에 국한되기 때문에 빠르게 이동해 오는 장애물의 경우 이동 로봇에 근접한 후에야 탐지가 가능하므로 회피에 어려움이 발생한다. 또한, 카메라의 각도를 작게 하면 장애물을 멀리서부터 관찰할 수 있지만, 이동

로봇과 근접한 영역에 대해서는 탐지할 수 없으므로 장애물이 이동 로봇의 근접지역에서 갑자기 나타나는 경우나 장애물이 등가속 운동 또는 곡선 운동을 하는 경우 회피에 어려움이 따른다.

#### 4. 결론

본 연구에서는 자유 이동 경로를 갖는 이동 로봇이 움직이는 장애물을 효율적으로 회피할 수 있도록 하기 위한 방법으로 광류 정보와 퍼지논리 제어기를 이용한 이동 로봇의 운행 제어 방법을 제시하였다. 본 연구에서 제시한 이동 로봇의 장애물 회피 항법은 이동 로봇에 부착된 카메라를 통해 획득된 2차원 영상으로부터 광류 정보를 도출하고, 도출된 광류 정보를 이용하여 이동 로봇과 장애물과의 충돌 여부를 예측한 후, 퍼지 논리 제어기를 이용하여 장애물과의 충돌을 회피할 수 있도록 이동 로봇의 운행을 제어하였다. 또한 본 연구에서 제시한 이동 로봇의 장애물 회피 항법은 장애물 회피와 목적지점까지의 도달시간 단축이라는 두 가지 항법 목표를 가지고 이동 로봇의 운행 방향과 속도를 결정할 수 있도록 하였다. 본 연구에서 제시한 장애물 회피 항법의 타당성을 검증하기 위한 모의 실험에서는 이동 로봇이 다양한 경로를 갖는 여러 가지 유형의 장애물을 성공적으로 회피하며 목적지에 도달할 수 있음을 보여 주었다. 본 연구에서 장애물을 탐지하기 위해 사용한 광류 정보는 한 대의 카메라만을 이용하여 도출할 수 있기 때문에 정보 처리량을 줄일 수 있고 장비의 비용을 절감할 수 있다는 장점을 제공한다. 또한 본 연구에서 제시한 모의 실험기는 이동 로봇을 직접 제작하여 운행하지 않더라도 실제 상황에 근접한 다양한 상황에서 장애물과의 충돌 여부를 실험할 수 있다는 장점을 제공한다.

#### 참고문헌

Beaufreire, B. and Zeghloul, S. (1995), Navigation Method for a Mobile Robot Using Fuzzy-based Method, *International Journal of Robotics and Automation*, 10(3).

Beom, H. R. and Cho, H. S. (1992), A sensor- based obstacle avoidance controller for a mobile robot using fuzzy logic and neural network, *IEEE International Conference on Intelligent Robots and Systems*, 1470-1475.

Borenstein, J. and Koren, Y. (1989), Real-time obstacle avoidance for fast mobile robot, *IEEE Transactions on Systems, Man, Cybernetics*, 19(5), 1179-1187.

Brooks, R. A. (1986), A robust layered control system for a mobile robot, *IEEE Transactions on Robotics and Automation*, 2(1), 14-23.

Camus, T. (1994), Real-time Optical Flow, *MVA/SME Applied Machine Vision 94*.

Enkelmann, W. (1991), Obstacle detection by evaluation of optical flow fields from image sequences, *Image and Vision Computing*, 9(3), 160-168.

Horn, B. and Schunck, B. (1983), Determining optical flow, *Artificial Intelligence*, 17, 185-203.

Khatib, O. (1986), Real-time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotics Research*, 5(1), 90-98.

Larcommbe, M. H. E. (1981), Tracking stability of wire guided vehicles, *Proceedings of International Conference on Automated Guided Vehicle System*, 137-144.

Lee, K-H. and Oh, K-R. (1991), Fuzzy Logics and Applications, Hongneung Publishing Co.

Moravec, H. P. and Elfes, A. (1985), High Resolution Maps from Wide Angle Sonar, *Proceedings of IEEE Conference on Robotics and Automation*, 116-121.

Song, K. and Tai, J. (1992), Fuzzy Navigation of a Mobile Robot, *Proceeding of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 621-627.

Sonka, M., Hlavac, V. and Boyle, R. (1993), *Image Processing, Analysis and Machine Vision*, Chapman & Hall Computing.