IR 센서와 영상정보를 이용한 다 개체 로봇의 장애물 회피 방법

Obstacle Avoidance Method for Multi-Agent Robots Using IR Sensor and Image Information

전 병 승, 이 도 영, 최 인 환, 모 영 학, 박 정 민, 임 묘 택*

(Byung-Seung Jeon¹, Do-Young Lee², In-Hwan Choi³, Young-Hak Mo³, Jung-Min Park³, and Myo-Taeg Lim³)

¹Samsung Electronics

²HYUNDAI KEFICO

³School of Electrical Engineering

Abstract: This paper presents obstacle avoidance method for scout robot or industrial robot in unknown environment by using IR sensor and vision system. In the proposed method, robots share the information where the obstacles are located in real-time, thus the robots can choose the best path for obstacle avoidance. Using IR sensor and vision system, multiple robots efficiently evade the obstacles by the proposed cooperation method. No landmark is used at wall or floor in experiment environment. The obstacles don't have specific color or shape. To get the information of the obstacle, vision system extracts the obstacle coordinate by using an image labeling method. The information obtained by IR sensor is about the obstacle range and the locomotion direction to decide the optimal path for avoiding obstacle. The experiment was conducted in 7m x 7m indoor environment with two-wheeled mobile robots. It is shown that multiple robots efficiently move along the optimal path in cooperation with each other in the space where obstacles are located.

Keywords: swarm robot, cooperative robot, obstacle avoidance

1. 서론

모바일 로봇을 활용할 때 주변 환경을 인식하여 장애물을 회피하거나 경로를 설정하는 것은 중요한 문제이다. 어떠한 환경에서도 장애물을 효과적으로 회피하여 목적지까지 도달하는 시간을 최소로 하기위해서 어떤 장소 혹은 경로로 이동해야 하는지가 중요한 문제이다[1-5].

이러한 문제를 해결하기 위해서 단일 로봇으로는 많은 연구가 진행되고 있지만 단일 로봇시스템에는 근본적인 한계가 있다. 예를 들면, 단일 로봇만으로 주변 환경을 탐사한다면 너무 많은 시간이 걸릴 것이다. 또한 단일 로봇이 할 수 없는 일을 하기 위해서는 서로 다른 기능을 가진 로봇들의 협업은 필수적이다. 특히, 고성능의 단일 로봇은 로봇 상업화의 가장 큰 장애물인 비용증가와 관리의 어려움을 야기하기 때문에 최근에는 다 개체의 로봇을 운용해서 문제를 해결하는 연구가 진행되고 있다[6-9].

다 개체 로봇 또는 군집 로봇 시스템이란 다수의 로봇이 협동제어를 통해 단일 로봇 이상의 성능을 내거나 단일 로봇은 할 수 없는 작업을 수행하는 것을 목표로 하는 시스템이다. 이 시스템은 동시에 여러 장소에서 정보를 획득하고 공유하며 여러 작업을 진행한다[10-12]. 최근의 다 개체

로봇들은 제어를 위하여 천장에 카메라를 설치하여 로봇의 움직임을 관리하는 방식 또는 스타게이지등의 마커를 통해 절대 좌표를 이용하여 로봇의 위치를 인식하고 로봇을 제어한다[13-15]. 본 논문에서는 오직 로봇의 센서와 통신에 의해서 로봇간의 상대위치와 장애물의 위치를 파악하고 주행을 수행하는 방법에 대하여 연구한다.

현재 산업현장에서 대부분 사용되는 경로유도 방식의 이동로봇 시스템의 단점을 보완하기 위한 자율 이동로봇의 연구는 계속되고 있으며, 산업현장, 국방 분야, 서비스, 교육로봇분야에도 점차 적용되고 있는 실정이다. 협동작업 로봇에 관한 연구는 여러 대의 로봇이 서로 방해하지 않고 목표위치에 잘 도달하는 방법, 각각의 로봇들이 역할을 나누어 물건을 옮길 때의 운송방법, 여러 대의 로봇 중 한 대가 고장이 생겨도 복구되어 물건을 운반하는 방법 등이 연구되고 있다[16-18].

본 논문에서는 모바일 로봇이 비교적 저렴한 적외선 센서와 비전센서를 통해 얻은 영상정보를 바탕으로 초기위치를 알 수 없는 상태에서 주행하며 여러 개의 장애물의 정보를 획득하여 환경을 인식한다. 획득한 정보를 바탕으로 단일 로봇이 아닌 다수의 로봇들이 협업을 통하여 최적의 경로를 찾아 빠른 시간 내에 장애물을 회피하여 목적지까지 도착한다.

본 연구는 그림 1과 같은 시스템으로 구성되며 여러 대의 자율 이동로봇이 한정된 공간에서 함께 협동 작업을 수행하는 경우에 적용될 수 있다. 정해진 조건에서 각각의 로봇들은 임무를 수행하면서 서로 간의 작업을 방해하지 않

* 책임저자(Corresponding Author)

논문접수: 2011. 11. 20., 수정: 2012. 8. 1., 채택확정: 2012. 11. 5.

전병승: 삼성전자(best.jeon@samsung.com)

이도영: 현대케피코(dozero4@korea.ac.kr)

최인환, 모영학, 박정민, 임묘택: 고려대학교 전기전자전공공학부

{metty, darios8686, destin11, mlim}@korea.ac.kr

그림 1. 시스템 구성.

Fig. 1. System structure.

고 보다 정확하게 장애물의 정보를 획득하여 공유하고 장애물 사이에서 최적의 경로를 선택하여 이동하기 위하여 경로계획, 센서데이터 융합과 공유, 장애물 인식 알고리즘 등의 기능이 요구된다. 이 가운데 경로계획과 장애물 인식은 카메라와 센서의 정보를 바탕으로 한다.

논문의 순서는 다음과 같다. II 장에서는 이동로봇의 운동 모델에 대하여 소개하며, III 장에서는 로봇들 간의 거리, 장애물과의 거리 등과 같은 주변 환경으로부터의 정보를 얻기 위한 영상시스템과 센서시스템의 구성에 대해 소개한다. IV 장에서는 앞서 소개된 시스템으로부터 얻은 정보를 바탕으로 이동로봇의 경로 설정 방법을 소개한다. V 장에서는 제안된 시스템의 성능 검증을 위한 실험 환경을 소개하고, 단일 로봇으로 이동할 때와 다 개체의 로봇이 이동할 때의 성능을 실험적으로 검증한다. 마지막으로 결론과 차후 연구 진행 방향에 대해서 기술한다.

II. 이동로봇의 운동모델

이동 로봇이란 자신의 힘으로 위치를 변화시킬 수 있는 모든 로봇을 말한다.

바퀴 이동 로봇을 분류하기 위해서는 로봇의 움직임에 대한 분석이 먼저 이루어져야 한다. 그림 2는 로봇이 위치한 공간을 나타내기 위한 좌표 시스템이다.

그림 2. 이동로봇의 좌표 시스템.

Fig. 2. Coordinate of mobile robot.

위의 그림에서 X_1, Y_1 좌표계는 로봇 외부에 기준점을 둔 절대 좌표계이다. 그리고 X_2, Y_2 좌표계는 로봇의 중심점을 원점으로 하고 로봇이 향한 방향을 X축으로 하는 좌표계이다. 로봇의 자세를 분석하기 위해 다음과 같이 자세 (pose) P 를 정의할 수 있다.

$$P = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad (1)$$

여기서 x, y 는 로봇의 중심점의 좌표이고, θ 는 로봇의 앞부분이 향하는 방향각이다. 위의 정의에서 알 수 있듯이 로봇의 자세는 절대 위치 (x, y)와 로봇이 향한 방향각 (θ)으로 이루어지며, 이 세 가지를 이용하여 로봇의 자세를 표현하며 제어한다. 로봇의 자세를 분석하는 것은 궁극적으로 로봇의 구동부에 적절한 출력을 가함으로써 원하는 자세를 취하게 하기 위함이다.

주로 이용되는 로봇의 모델은 그림 3과 같은 양 바퀴 구동로봇으로, 구조가 간단하다. 양 바퀴 구동 로봇의 기구학적인 모델은 그림 3으로 부터 얻을 수 있다. 오른쪽 바퀴와 왼쪽 바퀴의 각속도를 ω_R, ω_L 이라고 하면 non-slipping 조건에 의해 오른쪽 바퀴와 왼쪽 바퀴의 지면과의 접촉점의 속도는 다음과 같다.

$$V_R = r \omega_R, \quad V_L = r \omega_L \quad (2)$$

단, 로봇 중심의 선속도는 v 이고 V_R, V_L 은 각각 오른쪽, 왼쪽바퀴의 선속도이다. 반시계 방향으로의 로봇의 각속도를 ω 라고 하면 v, ω 와 ω_R, ω_L (오른쪽, 왼쪽바퀴의 각속도)은 다음과 같은 관계를 가진다.

$$\begin{aligned} \omega &= \frac{V_R - V_L}{L} = r \frac{\omega_R - \omega_L}{L} \\ v &= \frac{V_R + V_L}{2} = r \frac{\omega_R + \omega_L}{2} \end{aligned} \quad (3)$$

여기서 L 은 양 바퀴 사이의 거리이다. $(x \ y \ \theta)^T$ 와 $(v \ \omega)^T$ 은 다음과 같은 기구학 식에 의해 연관 관계를 가지게 된다.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} \quad (4)$$

그림 3. 양 바퀴 구동 로봇.

Fig. 3. Two-wheel mobile robot.

III. 시스템구성

1. 영상시스템

로봇은 이동하면서 센서를 이용하여 장애물을 탐지하고, 센서에서 얻은 정보를 바탕으로 회피해야 할 방향을 설정하고 최적의 경로를 탐색하여 주행한다. 각각의 로봇들은 몸체에 장착되어있는 센서로 장애물을 인식하여 거리판단 및 위치를 확인한다. 웹캠영상에서 얻어진 영상을 그림 4와 같은 순서를 거쳐서 장애물 인식 및 방향 설정을 위해 필요한 정보를 구한다.

이진화 처리에서는 임계값 이상의 값은 255로 변환하고 임계값보다 작은 값은 0으로 변환한다. 이를 식으로 표현하면 식 (5)와 같다.

$$P'(x, y) = \begin{cases} 255 & P(x, y) \geq T \\ 0 & P(x, y) < T \end{cases} \quad (5)$$

장애물을 인식하기 위해 가장 먼저 영상을 이진화[그림 5]하여 픽셀 데이터를 검정(0)과 흰색(255)으로 구성한다. 영상 이진화를 통해서 배경을 검정색으로 설정하고, 여러 종류의 장애물들을 흰색으로 설정한다. 레이블링 기법[그림 6]은 이러한 영상에서 도형을 분리하고 흰색 도형이 차지하고 있는 영역의 좌표 정보를 알고자 할 때 사용된다. 간단한 처리를 위해서 2가지의 데이터만 존재하고 서로 인접한 데이터의 영역을 계산하고 영역의 정보를 알아내는 방법이다. 이 방법은 현재 자기 자신의 픽셀과 붙어있어서 연결된 영역이거나 또는 같은 색상 범위를 갖는 영역들을 하나의 물체로 간주한다[그림 7].

레이블링을 통해 영역이 구분되어지면 영상에서 차지하는 영역이 5퍼센트 이하이거나 80이상인 부분은 예외로 처

그림 4. 장애물 인식 순서.

Fig. 4. Obstacle detection sequence.

그림 5. 원본영상(좌)과 이진화 이미지(우).

Fig. 5. Original image(left) and binary image(right).

그림 6. 레이블링 탐색.

Fig. 6. Labelling search.

그림 7. 레이블링 결과.

Fig. 7. Labelling result.

리한다. 최소최대 영역이 제거되면 영역을 계산하여 장애물을 판별한 후 장애물을 회피한다.

2. 센서시스템

적외선 센서는 수광부와 발광부 한 쌍의 소자로 구성된다. 발광부는 일정 스펙트럼의 적외선을 계속 방출하며, 수광부는 발광부에서 방출되어 반사되어오는 적외선을 수광한다. 적외선 센서는 수광부의 빛의 양에 따라서 소자의 통과 전류를 변화시켜, 컬렉터와 에미터 사이에 흐르는 전류의 크기를 저항을 통해서 빛의 세기에 비례하게 전압을 출력한다. 이 전압을 ADC를 통해서 디지털 신호로 변환하여 장애물의 유무, 장애물과의 거리 등의 정보를 얻을 수 있다. 그러나 빛의 반사는 반사면의 상태에 따라 돌아오는 광량이 차이가 나기 때문에 정확한 거리 측정은 불가능하다.

본 실험에서 사용한 센서는 빛을 보내고, 그 빛이 반사되어 돌아오는 각도를 측정한다. 삼각측량법과 비슷한 방법으로, 반사면의 상태에 상관없이 매우 정확한 거리 측정이 가능하다[19].

GP2Y0D21YK

그림 8. 측정 거리 특성.

Fig. 8. Measurement distance characteristics.

그림 9. 장애물 인식.

Fig. 9. Obstacle detection.

그림 10. 장애물 인식 그래프.

Fig. 10. Obstacle detection graph.

그림 8과 같이 로봇에 장착된 거리측정센서는 장애물이 없는 곳에서는 일정한 전압 값이 측정되다가 감지거리 안에 장애물이 놓여있으면 전압 값은 High(1) 레벨로 상승한다. 장애물의 감지거리에 따라 전압 값은 바뀌게 되며 거리가 가까워질수록 전압 값은 상승한다.

그림 9와 같이 로봇이 1개의 장애물을 만났을 때 센서의 반응은 그림 10과 같다.

IV. 회피공간 결정

1. 이동로봇 방향 결정

여러 개의 장애물이 있는 환경에서 현재 위치에서 주어진 목적지까지 로봇을 이동시키는 경로 계획에 대한 개념은 봄비는 거리를 걸어가는 인간이 취할 수 있는 행동의 결과와 같다. 시각을 통해 얻어지는 정보를 이용해 다른 사람의 현재 위치와 움직이고 있는 방향을 인지한다. 이 정보만 알고 있는 상황에서 다른 사람들이 있는 위치를 고려하여 현재 위치에서 목적지까지 다른 사람과의 충돌 없이 움직일 수 있는 최적의 경로를 추정한다. 로봇으로 바뀌 말하면 설정한 경로로 이동하기 시작한 후 시간이 흐르면 또

그림 11. 센서 모델링.

Fig. 11. Sensor modelling.

그림 12. 회피 공간 표시.

Fig. 12. Avoidance space mark.

다른 장애물로 인해 이전에 생각했던 경로는 더 이상 최적의 경로가 아니며 장애물로 인해 경로가 방해받을 수 있다. 이때 다시 영상 정보와 센서 정보를 활용하여 최적의 경로를 생성하고, 그 방향으로 움직이게 된다. 본 연구에서는 적외선 거리 센서와 카메라를 통해 얻은 각도와 거리 등의 정보를 활용한 방법을 사용한다. 그림 11은 실시간 처리를 위해 센서가 장애물을 감지하는 센서 모델링 기법을 제시하였다.

여기서 $n = 0, 1, \dots, \text{max_number}$ 은 적외선 거리 센서의 배치수를 의미한다. 그리고 $\text{Sector}(n)$ 은 로봇과 감지된 장애물사이의 거리를 의미한다.

그림 11에서 보면 $\text{Sector}(0)$ 에서부터 $\text{Sector}(n)$ 까지 감지거리의 변화가 생긴 센서를 제외한 나머지 섹터들이 이동로봇이 가야할 방향의 후보군들이다. 후보군의 방향 결정 면적은 그림 12에서 나타난다.

$\text{Sector}(n)$ 의 방향 벡터에서 왼쪽과 오른쪽의 각 15° 씩의 경계를 두어 벡터라인이 격자를 60 % 이상 차지하게 되면

그림 13. 회피 공간.

Fig. 13. Avoidance space.

후보군의 면적으로 인정한다. 후보군의 Sector(0)과 Sector(n), Sector(n+1)과의 면적 및 거리를 환산하여 최종 후보군을 결정하고 로봇이 최종후보군 방향으로 주행한다. 카메라를 통해 영상정보를 얻게 되면 영상처리를 통하여 주변 환경을 인식하여 오브젝트를 파악하고, 회피 공간[그림13]을 결정한다. 추출된 정보는 640x480개의 픽셀데이터의 거리 값과 감지된 오브젝트의 x 좌표와 y 좌표 값이다. 로봇 중심에서의 상대 각도는 화각 60°대 전체픽셀비로 환산한다.

Robot Number X = 1, 2, 3

n = 0, 1, 2, 3, ... 전체픽셀

DistanceToObject(n) : n번째 오브젝트와의 거리 값

(n = 0, 1, ... i : 오브젝트 개수)

Object_X(n) : n번째 픽셀의 물체 x좌표값

Object_Y(n) : n번째 픽셀의 물체 y좌표값

InfoData(n) = {DistanceToObject(n), Object_X(n),

Object_Y(n)}

2. 회피방향 설정 및 장애물 이동방향 정보

로봇이 장애물을 회피하여 이동하기 위해서 장애물 인식에 필요한 최소한의 정보를 수집하여 방향을 설정하여야 한다. 장애물의 이동방향정보는 연속적인 영상에서 얻어진 좌표를 통하여 생성한다.

첫 번째 프레임 영상에서 얻은 로봇의 위치 First_Point와, 두 번째 프레임 영상에서 얻은 위치 Second_Point를 알고 있고, 예상 위치를 Predict_Point라고 한다면, 식 (9)와 같이 Predict_Point를 구할 수 있다. 각 점들을 First_Point(x_1, y_1), Second_Point(x_2, y_2), Predict_Point(x_3, y_3)로 표현하고, 로봇과 장애물 사이의 거리를 식 (6)과 같이 벡터 성분으로 나타낸다.

$$\overrightarrow{Distance} = (x_2 - x_1, y_2 - y_1) \quad (6)$$

벡터의 크기와 방향각은 식 (7), 식 (8)과 같다.

$$|\overrightarrow{Distance}| = Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (7)$$

그림 14. 오브젝트 이동 예측 좌표.

Fig. 14. Estimation coordinates of obstacle.

$$\theta = \tan^{-1} \left(\frac{(y_2 - y_1)}{(x_2 - x_1)} \right) \quad (8)$$

$$\begin{aligned} x_3 &= Distance \times \cos a + x_2 \\ y_3 &= Distance \times \sin a + y_2 \end{aligned} \quad (9)$$

장애물의 이동방향정보도 역시 First_Point와 Second_Point의 정보로부터 알 수 있다.

$$\begin{cases} x_2 - x_1 < 0 \\ x_2 - x_1 > 0 \\ x_2 - x_1 = 0 \end{cases} \quad (10)$$

위의 식에서 $x_2 - x_1 < 0$ 조건을 만족하면 장애물은 오른쪽에서 왼쪽 방향으로 이동하는 경우가 되고, $x_2 - x_1 > 0$ 조건을 만족하면 장애물은 왼쪽에서 오른쪽 방향으로 이동하고 있는 경우이다. $x_2 - x_1 = 0$ 이면 장애물은 x 좌표에서 정지가 된 것처럼 보이지만 y 좌표의 계산에 의해서 다가오는 물체인지 멀어지는 물체인지를 판단 할 수 있다.

$$\begin{cases} y_2 - y_1 < 0 \\ y_2 - y_1 > 0 \\ y_2 - y_1 = 0 \end{cases} \quad (11)$$

다음 식에서는 y의 좌표 값에 의해 $y_2 - y_1 < 0$ 이면 장애물은 로봇과의 거리에서 멀어지고, $y_2 - y_1 > 0$ 이면 장애물은 로봇과 가까워진다. $y_2 - y_1 = 0$ 을 만족하면 장애물은 가까워지지도 않고 멀어지지도 않은 상태에서 x 좌표의 계산에 의해 오른쪽 방향인지 왼쪽방향인지 알 수 있다.

그림 15. 회피공간 결정.

Fig. 15. Decision of obstacle space.

그림 16. 이동 방향 결정.

Fig. 16. Decision of moving direction.

Robot의 반경이 R이라고 했을 때,

$$\begin{aligned} 1 : \text{Object1_Width} & \quad 2 : \text{Avoid_area1} = |a - b| \\ 3 : \text{Object2_Width} & \quad 4 : \text{Avoid_area2} = |c - d| \\ 5 : \text{Object3_Width} & \\ |a - b| > R & = \text{True} \\ |a - b| < R & = \text{False} \\ |c - d| > R & = \text{True} \\ |c - d| < R & = \text{False} \end{aligned}$$

True 값을 반환하게 되면 회피후보공간이 되고, False값을 반환하게 되면 회피후보공간이 될 수 없다.

각 로봇이 이동할 때 리더로봇과의 거리 R_{dist1} , R_{dist2} 를 구하고 회전할 각도 θ_b , θ_d 을 계산 한 후에 이동 할 위치를 정한다. 정해진 위치에 도달하기 위해 각 로봇의 센서가 획득한 정보를 토대로 방향을 결정한다. Obs_{dir} 는 장애물을 감지한 센서이고, R_{dir} 는 로봇의 직진 방향이다. 여기서 R_{dir} 방향과 장애물의 가로축 방향의 간격이 로봇의 크기보다 작게 되면 공간을 통과하지 못하기 때문에 새로운 방향을 제시해야 한다. 그래서 $R_{dir} + Obs_{dir}$ 합성 벡터를 이용하여 방향을 설정하고, θ_{des} 만큼 회전하여 새로운 방향으로 이동한다.

V. 실험 결과

1. 실험환경

실험을 위하여 그림 17과 같이 장애물을 배치한다. 바닥에 있는 격자무늬는 실험에 관계가 없다. 실험은 건물 내부의 세미나실 환경에서 수행하였으며 바닥이 평평하고, 주행 중 이동로봇에는 어떠한 외란도 가해지지 않는다고 가정한다. 실험에 사용된 소프트웨어는 Microsoft Visual Studio 2005 환경에서 제작되었고, Windows 7 (32bit), Windows XP (32bit) 운영체제하에서 실행하였다.

실험에 사용된 로봇은 양 바퀴 구동형 타입(2-wheel drive type)이며, 14.7 x 13 x 13.5 [cm]의 크기를 가지는 이동로봇이다. 시스템의 AT91SAM7X256(48 MHz) 마이크로컨트롤러를 사용하여 제어하며, 2개의 step motor와 6개의 적외선 센서를 제어한다. 카메라는 USB케이블을 PC에 연결하여 PC

그림 17. 실험환경.

Fig. 17. Environment of experiment.

에서 제어한다. MCU는 PC와 무선 통신을 100 ms 마다 패킷(packet)방식으로 통신을 수행하여 컴퓨터로부터 들어오는 알고리즘을 이용하여 각각의 모터를 독립적으로 제어한다. 총 6개의 적외선 센서가 전면과 후면에 부착되어있다. 물체 추적 등 영상 정보를 획득하기 위한 카메라 역시 로봇 전면 하단부분에 장착되어있다.

2. 실험결과 및 분석

2.1 한 대의 로봇이 이동할 때

환경과 장애물에 대한 정보를 미리 입력하지 않고 센서 데이터와 카메라를 이용한 영상 정보만을 가지고 장애물의 정보를 실시간으로 판단하여 주행하는 실험이다. 목표점은 맨 끝 장애물이 끝나는 시점으로 정하고, 로봇에게 인위적으로 주어지는 외란은 없다고 판단하고 실험한다.

로봇은 영상을 통해 주변 환경에 장애물이 존재하는지 인식한다. 카메라에서 장애물의 상대적 거리와 센서에서 획득한 장애물과의 거리를 비교 후 로봇에서부터 장애물까지의 거리를 계산한다. 영상에서는 장애물의 영역을 계산한

그림 18. 센서 배치.

Fig. 18. Sensor arrangement.

표 1. 이동로봇의 재원.

Table 1. Experiment result of one robot.

재원	특징	
크기	L: 130 mm W: 147 mm H: 135 mm	
Main Control Processor	AT91SAM7X256(48MHz)	
Motor	모터 드라이버	SLA7026M Stepping
	모터	SST42D2120 Stepping
PC	CPU : Core2Quad 2.50GHz	
	RAM : 4GB VGA : Geforce 9600GT	
시각 센서	1.3-MP WebCam C500(Logitech)	
IR 센서부	Sharp GP2Y0A21 거리측정센서	

다음에 밑변의 중심점이 로봇의 왼쪽 위치 또는 오른쪽 위치에 있는지 판별 하고, 로봇은 획득한 정보를 바탕으로 장애물을 피해 주행한다.

그림 19. 한 대의 로봇주행.

Fig. 19. Navigation of one robot.

그림 20. 회피 공간 비교.

Fig. 20. Avoidance space comparison.

표 2. 한 대 로봇의 주행 결과.

Table 2. Experiment result of one robot.

로봇	차수	장애물 통과횟수	주행 시간	주행 성공 여부
1대	1	4	1:25:47	성공: 목표지점 도착
	2	4	1:14:00	성공: 목표지점 도착
	3	0.5	x	실패: 초기위치 복귀
	4	4	2:01:23	성공: 경로 탐색 실패
	5	4	1:14:57	성공: 목표지점 도착
	6	2	x	실패: 경로 벗어남
	7	2	x	실패: 경로 벗어남
	8	3	1:30:11	성공: 목표지점 도착
Avg		2.937	1:42:43	62.5%성공(5:3)

그림 20에서 보면 회피공간의 후보군이 세 곳이 있다. 이 중 회피 공간 ②는 로봇이 진입하기에는 너무 좁은 공간으로서 다른 공간을 후보군으로 선택해야 한다. 하지만 위의 주행결과로 보면 로봇은 회피 공간 ②를 선택하여 주행 한 후 그림 19와 같이 두 번째 장애물을 인식하지만 오른쪽과 왼쪽 모두 장애물이 있기에 로봇은 다른 회피공간을 찾아 주행하게 된다. 로봇은 장애물이 없을 경우 자신이 회전한 만큼의 각도만큼 복귀하도록 알고리즘이 구성되어 있다. 위의 결과를 보면 로봇은 자신의 전면에 있는 정보만을 가지고 판별하기 때문에 다른 지역의 회피공간을 판단하지 못하여 오히려 시간을 낭비하고 돌아가는 경우가 발생한다. 몇 차례의 실험결과 목표지점까지 오지 못하는 경우도 발생하고, 다시 초기위치로 되돌아오는 현상도 발생한다. 이를 해결하기 위해 다음 절에서는 로봇의 수를 증가시키어 서로의 장애물 정보를 공유하여 최적의 회피공간으로 유도하는 실험을 진행한다. 표 2는 한 대 로봇의 주행결과를 나타내며, 0.5는 장애물을 한 개로 봤을 때 반 정도 회피하다가 돌아왔다는 의미이다.

2.2 세 대의 로봇이 이동할 때

일하게 하여 실험한다.

그림 22에서 보면 회피공간의 후보군이 3군데가 있다. 이 중 회피 공간 ②의 후보군은 로봇이 진입하기에는 너무 좁은 공간으로서 다른 공간을 후보군으로 선택해야 한다. 한 대의 로봇으로 실험할 때는 회피 공간 ②를 선택하여 주행 한 후 그림 19와 같이 두 번째 장애물을 인식하여 오른쪽과 왼쪽 모두 장애물이 있기에 로봇은 그림 19와 같이 다른 회피공간을 찾아 주행하게 되었지만 세 대의 로봇이 주행할 때는 리더로봇 앞에 있는 회피공간과 나머지 서브로봇들이 인식한 회피공간을 비교하여 로봇이 이동할 방향을 설정한다. 그림 21에서 보면 리더로봇은 회피 공간 ②를 선택하지 않고 회피 공간 ①을 선택하고, 로봇은 여유롭게 장애물을 회피하여 이동한다. 세 대의 로봇들은 동시에 주행하면서 실시간으로 리더로봇에게 장애물에 대한 정보를 제공한다. 리더로봇은 획득한 정보를 바탕으로 장애물을 회피하여 그림 21과 같이 목표지점으로 안전하게 주행한다. 표 3은 세 개의 로봇의 주행 결과이며, 1.5는 장애물을 한 번 통과하고 그 다음에 반 정도 회피하다가 돌아왔다는 의미이다.

그림 21. 세 대의 로봇주행.

Fig. 21. Navigation of three robots.

그림 22. 회피 공간 비교.

Fig. 22. Avoidance space comparison.

표 3. 세 대 로봇의 주행 결과.

Table 3. Experiment result of three robots.

로봇	차수	장애물 통과횟수	주행 시간	주행 성공 여부
3대	1	1	x	실패: 리더로봇 이탈
	2	1.5	x	실패: 로봇 뒤영킴
	3	4	1:07:13	성공: 목표지점 도착
	4	5	00:53:09	성공: 목표지점 도착
	5	5	00:55:82	성공: 목표지점 도착
	6	4	00:54:92	성공: 목표지점 도착
	7	4	00:52:01	성공: 목표지점 도착
	8	3	00:49:87	성공: 목표지점 도착
Avg		3.438	00:59:97	75% 성공(6:2)

IV. 결론

본 연구에서 제안한 방법은 세 대의 로봇이 동시에 영상 정보를 획득하기 때문에 각 위치에서 장애물의 위치와 방향정보를 PC에 전송하고, PC에서는 장애물의 거리, 위치, 회피 공간 면적을 계산하여 최적의 방향을 리더로봇에게 전송한다. 이때 각 장애물의 인식률을 높이고 영상처리 시간을 줄이기 위하여 Land Region을 설정하고 픽셀 검색도 확률 값을 추가하여 이웃데이터가 많은 것을 장애물로 인식하고 밀변의 좌표만을 추적하여 장애물을 판단할 수 있는 장점을 지녔다. 서버 로봇은 앞에 펼쳐진 장애물과의 거리와 위치정보를 계산하여 로봇이 어느 쪽 방향으로 주행해야 효과적인지 알려주면 리더로봇은 효과적인 공간으로 주행하여 회피한다. 이러한 실험결과를 바탕으로 1대의 로봇으로 여러 개의 장애물이 있는 공간을 주행하는 것보다 여러 대의 로봇으로 서로 정보를 공유하여 주행하는 것이 시간적인 면이나 최적의 공간 선택에서도 효과적이라고 볼 수 있다.

본 연구에서는 실시간으로 데이터를 획득하고 과거의 데이터는 사용되지 않는다. 하지만 과거의 데이터를 활용하여 3D MAP을 작성하고, 이 정보를 바탕으로 자율 주행하여 최적의 회피 공간 선택과 장애물 인식의 정확도를 높이는 연구와 리더로봇과 서버로봇의 개념이 아닌 모든 로봇이 동시에 주변 환경의 정보를 교환하여 보다 강력하고 효율적인 협업을 수행하는 연구가 진행되고 있다.

참고문헌

- [1] L. Jain, E. Aidman, C. Abeynayake, and P. Dasgupta, "Multi-robot task allocation for performing cooperative foraging tasks in an initially unknown environment," *Innovations in Defence Support Systems-2*, Springer-Verlag, vol. 338, pp. 5-20, 2011.
- [2] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," *Proc. of the IEEE International Conference on Robotics & Automation*, San Francisco, vol. 1, pp. 476-481, Apr. 2000.
- [3] T. H. Kang and B. K. Kim, "Efficient online path planning algorithm for mobile robots in dynamics indoor environments," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 7, pp. 651-658, July 2011.
- [4] Y. H. Kim, U. K. Song, and B. K. Kim, "Development of precise localization system for autonomous mobile robots using multiple ultrasonic transmitters and receivers in indoor environments," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 4, pp. 353-361, Apr. 2011.
- [5] T. T. Yang, Z. Y. Liu, and H. Chen, "Formation control and obstacle avoidance for multiple mobile robots," *Acta Automatica Sinica*, vol. 34, no. 5, pp. 588-593, May 2008.
- [6] A. I. Mourikis and S. I. Roumeliotis, "Performance analysis of multirobot cooperative localization," *IEEE Trans. on Robotics*, vol. 22, no. 4, Aug. 2006.
- [7] S. Liu, Y. Tian, and J. Liu, "Multi mobile robot path planning based on genetic algorithm," *Proc. of the 5th World Congress on Intelligent Control and Automation*, vol. 5, pp. 15-19, Jun. 2004.
- [8] H. Lee, E. J. Jung, B. J. Yi, and Y. Choi, "Navigation strategy of multiple mobile robot systems based on the null-space projection method," *International Journal of Control, Automation, and Systems*, vol. 9, no. 2, pp. 384-390, 2011.
- [9] F. Arrichiello, J. Das, H. Heidarrson, A. Pereira, S. Chiaverini, and G. S. Sukhatme, "Multi-robot collaboration with range-limited communication: experiments with two underactuated ASVs," *Field and Service Robotic*, vol. 62, pp. 443-453. 2010.
- [10] Y. U. Cao, A. Fukunaga, and A. Kahng, "Cooperative mobile robotics: antecedents and directions," *International Conference on Intelligent Robots and Systems*, Pittsburgh, vol. 4, pp. 1-23, Aug. 1997.
- [11] L. E. Parker, "Current state of the art in multi-robot teams, in distributed autonomous robotic systems," New York: Springer, pp. 3-12, 2000.
- [12] M. Dorigo, "SWARM-BOT: an experiment in swarm robotics," *Swarm Intelligence Symposium, Proceedings IEEE*, pp. 192-200, June 2005.
- [13] A. Farinelli, L. Iocchi, and D. Nardi, "Multi robot systems: a classification based on coordination," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2015-2028, Oct. 2004.
- [14] D. H. Yang, "Collision avoidance algorithm for multiple mobile robots using roadmaps," Ajou University, Doctoral Dissertation, 2006.
- [15] J. H. Kim, "Robot soccer engineering," KAIST Press, 2002.
- [16] E. Pagello, A. D'Angelo, and E. menegatti, "Cooperation issues and distributed sensing for multirobot systems," *Proceedings IEEE*, vol. 94, no. 7, pp. 1370-1383 Jul. 2006.
- [17] E. Semsar-Kazerooni and K. Khorasani, "Multi-agent team cooperation: a game theory approach," *Automatica*, vol. 45, no. 10, pp. 2205-2213, 2009.
- [18] W. B. Dunbar and R. M. Murray, "Distributed receding horizon control for multi-vehicle formation stabilization," *Automatica*, vol. 42, no. 4, pp. 549-558, 2006.
- [19] H. Park, S. Baek, and S. Lee, "IR sensor array for a mobile robot," *Proc. of the 2005 IEEE/ASME, International Conference on Advanced Intelligent Mechatronics Monterey*, California, USA, 24-28 Jul. 2005.

전 병 승

2008년 경희대 전자공학과 졸업. 2010년 고려대 전기전자전파공학부 석사. 2010년~현재 삼성전자 근무. 관심분야는 영상 처리 및 자율 이동 로봇, 최적 제어.

이 도 영

2010년 광운대 전자공학과 졸업. 2012년 고려대 전기전자전파공학부 석사. 2012년~현재 현대케피코 근무. 관심분야는 영상 처리 및 이족 보행 로봇, 최적 제어.

최 인 환

2010년 고려대 전기전자전파공학부 졸업. 2010년~현재 동 대학원 석/박사 통합과정 재학. 관심분야는 패턴인식 및 영상처리, 자율 이동 로봇, 최적 제어.

모 영 학

2009년 고려대 전기전자전파공학부 졸업. 2009년~현재 고려대학교 메카트로닉스협동과정 석/박사 통합과정 재학. 관심분야는 패턴인식 및 영상처리, 자율 이동 로봇, 최적 제어.

박 정 민

2006년 고려대 전기전자전파공학부 졸업. 2008년 동 대학원 석사. 2008년~현재 동 대학원 박사 과정 재학. 관심분야는 최적 제어 및 강인 제어, 자율 이동 로봇.

임 묘 택

1985년 고려대 전기공학과 졸업. 1987년 동 대학원 석사. 1994년 Rutgers University 박사. 1996년~현재 고려대학교 전기전자전파공학부 교수. 관심분야는 최적 제어 및 강인 제어, 영상 기반 로봇 제어, 자율 이동 로봇.