2D 카메라를 이용한 A* 경로 계획 기법

신승엽*, 조성재, 김예지, 심소현, 엄기현, 조경은
동국대학교 일반대학원 멀티미디어공학과
*동국대학교 영상대학원 멀티미디어학과
교신저자 e-mail : cke@dongguk.edu

A* Path Planning using 2D Camera

Seungyoub Ssin*, Seoungjae Cho, YeJi Kim, Sohyun Sim, Kyhyun Um, Kyungeun Cho
Dept. of Multimedia Engineering, Graduate School of Dongguk University
*Dept. of Multimedia, Graduate School Of Digital Image & Contents, Dongguk University

요 약

본 논문에서는 로봇이 정해진 폐구간을 이동하기 위해서 위에서 아래로 촬영한 카메라 정보를 활용한다. 로봇을 특정위치로 이동시키기 위해서는 카메라를 제어하는 서버 시스템과 로봇의 위치를 인식하기 위한 마크가 필요하다. 서버는 로봇의 위치를 로봇으로 인식하는 마크의 색 값으로 카메라로부터 인지하고 로봇에 위치 이동 명령을 수행할 서버와 로봇이 네트워크를 통해 Planning 을 수행한다. 본 연구에서 휴머노이드 로봇인 나오와 로봇에 위치를 촬영할 카메라 그리고 이미지 처리를 하기 위해 OpenCV 와 이동 알고리즘으로 A*를 활용하여 Planning 을 구현한다.

1. 서론

이미 로봇 산업은 제조업에서 사람에 역할을 대체하는 것뿐만 아니라 가정에서 여러 서비스를 제공하는 방향으로 발전하고 있다. 이러한 로봇은 정해진 공간에서만 작업을 하는 것이 아니라 자율적인 이동을 통하여 사람들에게 편의를 줄 수 있게 기본 인공지능을 장착할 필요를 가지게 되었다. 본 논문에서는 홈 서비스 로봇이 폐공간 상에서 어떤 식으로 이동하는 지에 대한 연구를 한다. [1,4,5]

우리는 엘더버른(Aldbaran)사의 나오(Nao)를 사용하여 실험하고 나오(Nao)는 총 25 개의 상체와 하체의 관절을 가지고 임베디드 리눅스를 탑재한 개방형시스템으로 2 족 보행을 하는 휴머노이드 로봇이다.[5] 그리고 개발 소프트웨어로 콜레그래프(Choregraphe)를 사용하고, 파이썬(Python) 언어를 사용하여 제작된다.

전체적으로 Online 개발방법을 지향하고 Online 개발 방법이란 핵심 처리 부를 서버(중앙시스템)로 구성하여 통신을 통하여 필요한 정보만을 주고 받는 구조로 과거 개발방법론에서 메모리 문제와 개발의 편의성 문제를 극복 할 수 있게 한다.[3,4] 서버 시스템은 마이크로 소프트 MFC(Microsoft MFC)를 사용하여 제작되었고 서버는 나오(Nao)와 이동정보를 실시간으로 주고 받으면서 위치이동을 수행하게 한다.

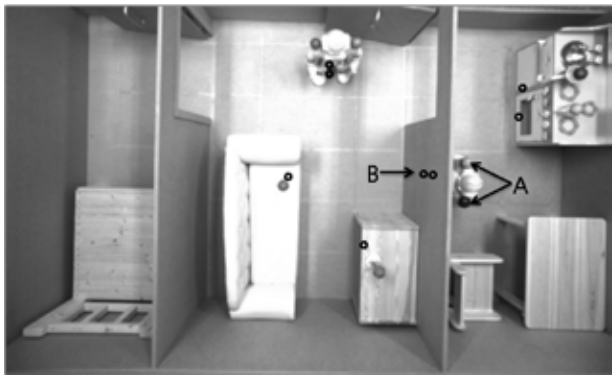
서버는 로봇의 위치를 카메라를 통하여 얻어 오는데 이는 나오(Nao)어깨에 붙은 색 감지 상징물(Iconographic)을 사용하여 인식한다.[2] 이번 연구에서 우리는 OpenCV 를 사용하여 색 감지 기능을 구성하고 감지된 색은 2 가지로 위치와 벡터를 생성한다. 위치와 벡터를 얻은 나오(Nao)는 다시 카메라에서 지정하는 목표점을 향하여 A*알고리즘을 통하여 이동을 수행한다.

본 연구에서 로봇의 자율적인 이동을 카메라를 이용하여 구성하고 실제 서비스 환경에서 적용할 수 있는 가능성을 보여준다.

2. OpenCV 를 사용한 위치인식

본 연구에서는 로봇의 자율적인 이동을 위해 나오 및 객체의 위치를 실시간으로 인식해야 한다. 인식을 위한 방법으로, 카메라를 통해 받은 영상에서 특정 색을 식별하여 물체의 위치를 검출한다. 각 객체 고유의 색은 서로 중복이 되거나 빛에 의한 색의 왜곡이 생겨 색 검출에 어려움이 따른다. 따라서 본 연구에서는 각 객체를 대표하는 색을 선정하여 객체마다 색 감지 상징물을 부착한다. 부착한 색 감지 상징물의 색을 구별하기 위한 방법으로는 OpenCV 의 색 검출 기능을 활용한다.

그림 1 은 OpenCV 를 통해 나오 및 각 객체의 위치를 검출한 그림이다. 나오 및 객체의 위치를 좌표로 나타내기 위해 나오의 방을 9x6 의 셀로 분할하였다. 1 차적으로 영상에서 색 감지 상징물(A)을 통해 각 객체의 영상에서의 위치를 검출하고, 2 차적으로 각 객체가 바닥에 수직으로 투영된 위치(B)를 계산하여 원으로 표시한다. 마지막으로 B 를 나오 방의 셀에 맞는 좌표로 변환하여 서버 프로그램으로 전송한다.



(그림 1) OpenCV 를 사용한 객체의 위치 검출

3. 위치이동을 위한 A* 구현

나오의 위치이동을 위해 먼저 나오의 현재 위치와 방향을 파악해야 한다. 이를 위해 먼저 천장에 2D 카메라를 설치하여 나오의 이동 가능 범위의 영상을 실시간으로 촬영해야 한다. 본 연구에서는 촬영한 영상에서 나오의 위치 및 방향을 파악하기 위하여 나오의 양 어깨에 서로 다른 색의 공을 부착하였다. 그림 2 는 나오의 위치 및 방향을 계산하는 알고리즘을 나타낸 것이다.

```
function NaoCalcPositionDirection(image, lsc, rsc)
    lsCenter ß CalcColorRegionCenter(image, lsc)
    rsCenter ß CalcColorRegionCenter(image, rsc)
    position ß GetMeanPoint(lsCenter, rsCenter)
    direction ß RotateVector(rsCenter - lsCenter, π/2)
    return position, direction
end
```

(그림 2) 영상에서 나오의 위치 및 방향을 계산하는 알고리즘

그림 2 에서 lsc 와 rsc 는 각각 왼쪽/오른쪽 어깨에 부착된 공의 R,G,B 색상 값이다. CalcColorRegionCenter 함수는 image 의 픽셀 중 lsc 또는 rsc 에 해당하는 픽셀의 집합의 중심 좌표를 구하는 함수이다. GetMeanPoint 함수를 통해 lsCenter 와 rsCenter 의 평균을 구함으로써 나오의 중심 위치를 계산한다. rsCenter - lsCenter 는 왼쪽 어깨의 중심 좌표에서 오른쪽 어깨의 중심 좌표를 향하는 벡터이며 이를 RotateVector 함수를 통해 시계 반대 방향으로 π/2(90°) 만큼 회전시킴으로써 나오의 방향 벡터를 계산한다.

나오의 현재 위치 및 방향을 파악한 후에는 나오를 목적지까지 이동시키기 위해 A* 알고리즘을 사용하여 이동 경로를 계산한다. A* 알고리즘을 적용하기 위해 나오가 활동하는 홈 공간을 격자 형태로 관리한다. 격자 공간에서 일반적인 A* 알고리즘을 수행함으로써 나오의 이동 시작 위치로부터 목표 위치까지 순서대로 거쳐야 할 격자 좌표의 큐(queue)를 계산한다.

그림 3 은 A* 알고리즘을 통해 얻은 이동 경로를 따라 나오를 이동시키는 알고리즘을 나타낸 것이다.

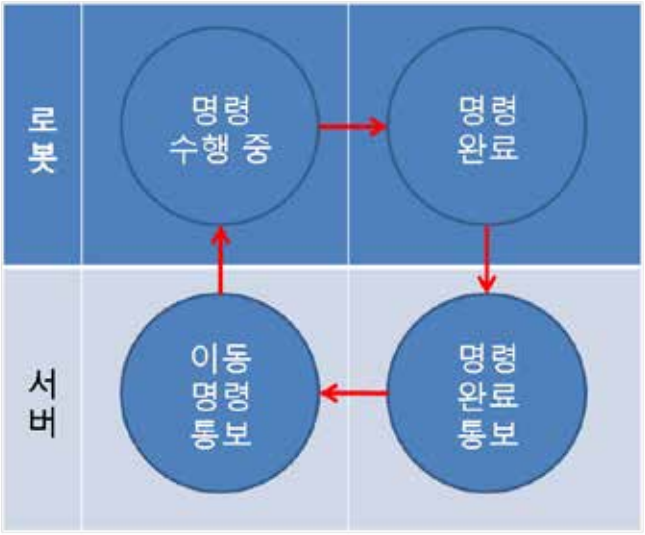
```
function NaoMove(image, lsc, rsc, target)
    position, direction ß
        NaoCalcPositionDirection(image, lsc, rsc)
    Q ß CalcAstarPath(position, target)
    while Q is not empty do
        next ß Q.Pop()
        length ß GetLength(next - position)
        angle ß GetAngle(direction, next - position)
        socket.Send(length, angle)
        position, direction ß
            NaoCalcPositionDirection(image, lsc, rsc)
    end
end
```

(그림 3) A* 알고리즘을 통해 얻은 이동 경로를 따라 나오를 이동시키는 알고리즘

그림 3 에서 target 은 목표 격자 좌표이다. 우선 NaoCalcPositionDirection 함수를 통해 나오의 초기 위치 및 방향을 계산한다. 그 후 CalcAstarPath 함수를 통해 현재 위치와 목표 위치를 파라미터로 입력받아, 순서대로 거쳐야 할 좌표의 큐(queue)를 반환한다. 큐에서 순서대로 다음으로 이동해야 할 좌표를 꺼내어 해당 좌표까지의 거리와 회전해야 할 각도를 계산한다. 계산한 거리 및 각도는 서버로 전송하여 실제 나오를 이동시킨다. 나오가 이동을 마친 후 현재 위치 및 방향을 다시 계산하여 현재 상태를 갱신한다. 큐에 저장된 좌표로 이동하기를 반복하면 최종 목표 위치에 도착하게 된다.

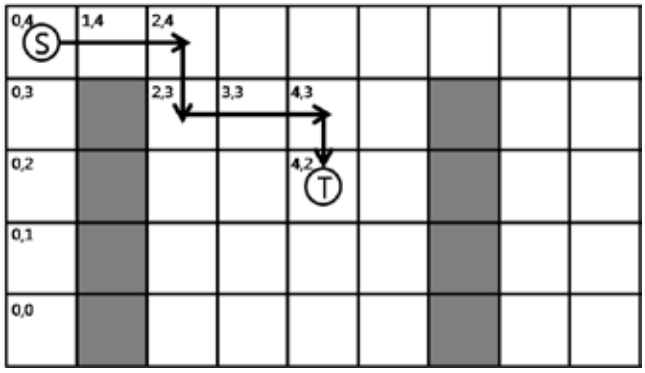
4. 서버를 통한 이동 명령 수행

서버는 Microsoft MFC 로 작성되었고 나오(Nao)와 TCP/IP 통신을 통하여 명령을 수행한다. 나오(Nao)의 명령 체계는 기본적으로 비동기적(Asyncronous)인 동작들로 구성되어 있어서 동시에 여러 동작을 수행할 수 있으나 이동은 목표지점까지 도달 하였는지에 결과를 가지고 다음 작업으로 수행을 결정하므로 기본적으로 동기적(Syncronous)인 처리로 수행된다.



(그림 4) 서버 시스템과 나오(Nao)의 통신과정

기본적인 동작은 전진, 회전으로 각각 파라미터 값을 받아서 수행되는 값으로 구성되어 있다. 그림 5 는 본 연구에서 구현된 A* Planning 의 예를 보여주고 있다.



(그림 5) Planning 을 통한 이동 실험 예

<표 1> 실제 Planning 을 통한 데이터 생성

position	direction	command
0,4	동	Go
1,4	동	Go
2,4	동	Right
2,4	남	Go
2,3	남	Left
2,3	동	Go
3,3	동	Go
4,3	동	Right
4,3	남	Go
4,2	남	move succeeded

5. 결론 및 향후 연구

본 논문을 통해서 로봇이 자율적으로 planning 을 수행하여 원하는 목표지점까지 이동을 수행하는 것을 구현하였고, 현재 보편적인 장비인 카메라를 통해 수행되는 과정을 제작하여 실제 생활에 적용 가능성을 보여 주었다.

이번 논문에서는 폐공간 상에서 이동을 중심으로 구성하였고 향후 다른 공간으로 이동 할 시 다른 카메라로 이동을 어떻게 구현 할지 그리고 어깨 상징물(다른 색깔 공)없이 오브젝트를 어떻게 인식 시킬지에 대한 연구를 하게 될 것이다.

(그림 6) 서버 시스템과 나오(Nao)

감사의 글

이 논문은 2011 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2011-0011266). 또한 이 논문은 2012 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (2012R1A1A2009148).

참고문헌

[1] Anna Hristoskova[1],*, Carlos E. Agüero[2], Manuela Veloso[3] and Filip De Turck[1](2012), 1 Department of Information Technology, Ghent University - IBBT, Ghent, Belgium, 2 Robotics Group, Universidad Rey Juan Carlos, Madrid, Spain, 3 Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA, Heterogeneous Context-Aware Robots Providing a Personalized Building Tour.

[2] M. Peña-Cabrera[1], I. Lopez-Juarez[2], R. Ríos-Cabrera[2]M. Castelán[2] and K. Ordaz-Hernandez[2] (2011), [1]Universidad Nacional Autonoma de México [2]Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV) Mexico, Object Location in Closed Environments for Robots Using an Iconographic Base, pp. 201 ~ 214

[3] Nathan Koenig, Maja J Matari ´c (2006), Interaction Laboratory. University of Southern California. University Park, Los Angeles, California, USA ,Behavior-Based Segmentation of Demonstrated Tasks.

[4] Monica N. Nicolescu* and Maja J. Mataric ´ (2003), Computer Science Department, University of Southern California 941 West 37th Place, Mailcode 0781 Los Angeles, CA 90089-0781, Natural Methods for Robot Task Learning: Instructive Demonstrations, Generalization and Practice.

[5] 서기성(2011), 기본편 Nao 를 이용한 인터렉티브 휴머노이드 입문. 서울 : ㈜NT 리서치.