휴머노이드 로봇의 지능적 행위 구현에 관한 연구

서주희*, 장인우*, 우종우*
*국민대학교 컴퓨터공학부
email: crazyDMP@gmail.com

A Study on the Development of Intelligent Behavior of Humanoid Robot

Suh , Joohee*[*], Jang, Inwoo*, and Woo, Chongwoo*
*School of Computer Science, Kookmin University

요 약

본 논문에서는 로봇의 지능적 행위를 구현하기 위하여 인공지능의 몇 가지 기법을 휴머노이드 로봇에 적용하고 이를 테스트 도메인에서 실험하는 연구결과를 기술하였다. 본 연구에서 적용한 기법들은, 인공지능의 계획기법에 기반한 로봇의 계획생성, A*알고리즘을 적용한 길 찾기, 외부 센서 값에 기반한 장애물회피 및 로봇의 자기 위치인식, 그리고 원하는 물체를 파악하기 위해 템플릿 매칭을 이용한 영상인식 등 네 가지 방향으로 접근하였다. 전반적으로 로봇의 실험은, 웹 페이지로부터 사용자의 쇼핑 목록을 입력 받아, 인공지능의 계획기법에 기반하여 서버에서 이에 대한 실행계획을 만들고 난 후, 로봇이 서버로부터 TCP/IP 기반의 소켓 통신을 통하여 세부 실행계획을 전달받아 임무를 수행하게 된다. 또한 이러한 임무를 수행하기 위해서는 로봇자신의 현재위치에 대한 정보 및 목표물에 대한 위치인식이 요구되며, 이를 위해서 사전에 주어진 맵의 좌표를 찾아가는 방법을 사용하였다.

1. 서론

최근 지능형 로봇은 국내외에서 활발한 연구가 진행되고 있으며, 국내에서도 다양한 기능을 가진 지능형 로봇에 관한 연구가 활성화 되고 있다 [1][2]. 특히 산업을 위한 서비스 로봇은 이미 상당히 많이 연구 발전되고 있으며, 최근에는 사용자 개인을 위한 서비스로봇도 연구 개발되고 있다[3]. 이러한 지능형 로봇이 주변환경을 인지하고 사람과 상호작용을 통한 서비스를 제공하기 위해서는 몇 가지 하드웨어와 인공지능관련 기법들이 필요하게 된다. 대표적으로 사용되는 하드웨어는 다양한 센서와 카메라가 주로 사용되고 있고, 로봇이 가질 수 있는 지능적 요소들은 위치 인식(localization), 목표 지점의 최단 거리 찾기, 그리고 영상인식 등이 활발히 연구되고 있다 [4].

본 논문에서는 이렇게 개인에게 적절한 서비스를 제공할 수 있는 지능형 로봇의 구현을 목적으로 하여 몇 가지 지능적인 기능을 접목한 지능형 휴머노이드 로봇 시스템을 설계 및 개발하고, 이를 간단한 테스트 도메인에 적용시켜보았다. 본 연구의 시스템에서 적용한 지능적 기능은 다음과 같이 몇 가지로 요약할 수 있다. 우선, 본 연구의 로봇은 주어진 시나리오에 따라 실행계획을 생성하고, 생성된 계획에 따라 하나씩 계획을 수행 함으로서 임무를 완성한다 [5].

둘째, 기본적으로 로봇의 이동은 주어진 맵의 좌표를 기반으로 목표의 물체를 A* 알고리즘을 이용하여 찾아가는 방법으로 수행하였다. 이때 수행되는 임무는 무선통신을 통하여 서버에서 로봇에게 주어진다. 셋째, 목적물체의 인식은 템플릿 매칭을 사용하여 인식한다. 이와 같은 기본적인 몇 가지 접근으로 로봇 시스템의 지능적 행위를 실험하였다. 실험을 위한 테스트 도메인은 사람의 요구사항을 대신하여 서비스 할 수 있는 쇼핑환경을 축소하여 만들고 로봇이 쇼핑 서비스를 수행하는 상황을 실험하였다.

2. 관련연구

2.1 지능형 로봇

현재 휴머노이드 로봇에 대한 연구는 유럽과 미국 등에서 활발히 연구되고 있으나, 최근에는 일본에서 주도적으로 연구가 진행되고 있다. 국외에서 연구되는 대표적인 지능형 로봇으로는 혼다의 ‘아시모(ASIMO)’와 미국 매사추세츠공과대학(MIT) 미디어랩에서 개발한 ‘키스멧(Kismet)’이 있다. 아시모는 휴머노이드 로봇으로 자이로 센서를 통해 몸의 균형을 유지하고 CCD 카메라로 사람을 인식하여 악수하거나 손을 흔드는 동작이 가능하다. 미국 매사추세츠공과대학(MIT) 미디어랩에서 개발한 키스멧(Kismet)은 아

[*] 본 논문은 서울시 산학연 협력사업의 지원을 받아 연구 수행된 논문입니다

동형 로봇으로 함께 있는 사람의 말과 행동에 따라 표정과 움직임이 달라지는 등 인간과 상호작용을 하며 이를 위해서 안구에 들어있는 카메라와 소형 무선 마이크 등으로 주위의 상황을 인식한다 [5][6][7].

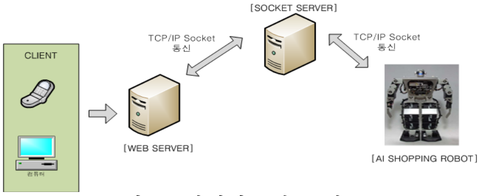
국내에서는 카이스트의 '휴보(hubo)'와 테크노비전의 '보노보(bonobo)'가 있다. 휴보는 아시모와 마찬가지로 휴머노이드 로봇이며 아시모와는 달리 눈에 부착된 CCD 카메라가 개별적으로 움직이고 손가락 관절을 추가한 면에서 성능이 우수하다고 볼 수 있다. 보노보 역시 휴머노이드 로봇이며 양 발바닥에 4 개의 입력센서를 통해서 몸의 기울어진 정도를 측정해 균형을 잡고 걸을 수 있는 특징이 있다.

이와 같은 로봇 시스템들은 대부분 영상 및 각종 하드웨어를 통한 인식으로 인간과의 상호작용 기능을 기본적으로 제공하고 있다.

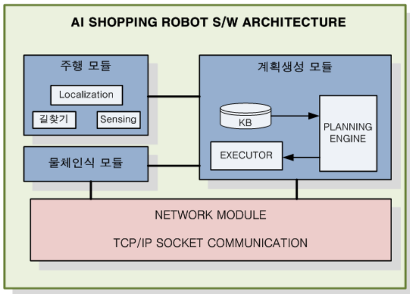
3. 시스템 설계

3.1 시스템 구성 및 개요

본 연구의 전체 시스템은 아래 그림 1 과 같이 사용자들이 웹을 통해 접속 할 수 있는 웹 서버, 로봇과의 소켓 통신을 담당하는 소켓 서버, 그리고 서비스를 실행하는 지능형 휴머노이드 로봇으로 구성된다.



(그림 1) 전체시스템 구성도



(그림 2) 로봇시스템 구성

본 연구의 로봇시스템은 그림 2 와 같이 크게 4 개의 모듈로 구성하였다. 우선, 서버와 TCP/IP 통신을 하는 네트워크 모듈, 원하는 물체를 가져오기 위해서 물체를 인식할 수 있는 물체인식모듈, 로봇의 전반적인 계획을 생성하는 계획생성모듈, 그리고 생성된 계획에 따라 장애물 회피 및 위치인식 후 이동 하는 주행모듈이 있다.

3.2 계획생성 모듈

로봇시스템의 지능적인 이동 행위를 자동으로 생성하기 위해서 본 연구에서는 인공지능의 계획기법으로 접근하였다. 다양한 계획기법이 존재하지만 본 연구에서는 우선적으로 가장 기본적인 STRIPS 기반의 비계층적 계획기법을 기반으로 하여 계획모듈을 구성하였다 [8].

계획모듈의 핵심부분인 엔진은 계획을 생성하기 위한 입력 데이타로, Initial_state, Goal_state 그리고 계획 생성 Rule 을 가지고 있는 Knowledge_Base 를 입력받아서 계획(Plan)을 생성한뒤 생성된 계획들을 Executor 에서 수행하게 된다. 생성된 계획(plan)은 로봇 움직임들로 구성되며, 아래 그림 3 은 계획모듈에서 수행되는 Rule 의 일부이다 [9][10].

	Action	Precondition	Negative Effect	Positive Effect
Op1	move(pos_product)	product_stay(product_state) robot_stay(table_state) armempty	robot_stay(table_state)	robot_stay(product_state)
Description	로봇이 빈 손으로 테이블 위치에서 물품 위치로 이동하는 액션			
Op2	grep(obj_product)	product_stay(product_state) robot_stay(product_state) armempty	armempty product_stay(product_state)	has(obj_product)
Description	로봇이 빈손일 때 물품 위치에서 물품을 잡는 액션			
Op3	move(pos_table)	robot_stay(product_state) has(obj_product)	robot_stay(product_state)	robot_stay(table_state)
Description	로봇이 물품 위치에서 물품을 가지고 테이블 위치로 이동하는 액션			
Op4	release(obj_product)	robot_stay(table_state) has(obj_product)	has(obj_product)	armempty product_stay(table_state)
Description	로봇이 테이블 위치에서 물품을 가지고 있는 상태에서 물품을 테이블 위에 놓는 액션			

(그림 3) 계획생성 Rule

[생성된 계획(Plan)]

1) Move_to_object: 물체가 있는 곳으로 이동한다
2) Grasp_object: 물체를 집는다
3) Move_to_Table: 테이블로 이동한다
4) Release_object: 테이블위로 물체를 올려 놓는다

3.3 주행 모듈

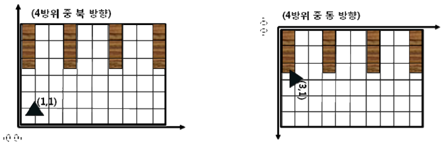
로봇의 주행 모듈은 크게 3 개의 모듈로 나누어 지는데 위치 인식을 담당하는 위치 인식(Localization), 모듈, 맵(Map) 정보를 담당하고 가장 빠른길을 찾는 길찾기 모듈 그리고 장애물을 감지하는 센싱 모듈이 있다.

3.3.1 위치인식(Localization) 모듈

로봇의 위치 인식은 로봇이 미리 가지고 있는 맵 좌표와 4 방위표를 이용하였다. 즉, 맵 안에서의 로봇의 Step 이동과 로봇의 방위에 따라 좌표 계산을 하여 위치를 인식하는 방법이며 이것을 'Step In Map' 이라 하였다. 로봇은 동, 서, 남, 북의 4 개의 방위를 가지고 각 방향에 맞는 좌표를 가지고 자신의 위치를 기억하며 알고리즘은 아래의 순서를 따른다. 좌표계를 이용하여 로봇이 위치 인식을 하는 간단한 예를 들어보면 다음과 같다.

1. 로봇이 북쪽을 바라보고 (1,1) 위치에 있다.
   ➔ (NORTH, 1, 1)
2. 로봇이 앞으로 2 Step 이동한다.
   ➔ (NORTH, 1, 3)
3. 로봇이 오른쪽으로 90 도 회전한다.

➔ (EAST, 3, 1)



(그림 4) 로봇의 좌표 이동 결과

위의 예처럼, 로봇은 자신의 Step 과 방향전환에 따라 위치변화를 파악한다.

[Step In Map 알고리즘]

1. 로봇이 북쪽을 향하고 있을 때 로봇의 초기 위치 (NORTH, x, y) 및 X 좌표와 Y 좌표의 최대값인 MaxofX, MaxofY 를 세팅한다.
2. 로봇이 m Step 전진 할 경우의 위치
   ➔ (NORTH, x, y+m)
3. 로봇이 시계방향으로 90 도 회전할 경우의 위치
   ➔ (EAST, MaxofY – y, x)
4. 로봇이 반시계방향으로 90 도 회전할 경우의 위치
   ➔ (WEST, y, MaxofX - x)

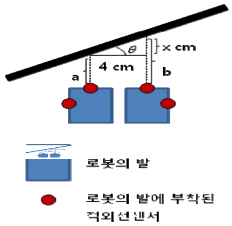
3.3.2 길찾기 모듈

본 논문에서는 주행을 위한 맵 정보와 물체의 위치를 로봇이 가지고 있으며 현재 로봇 좌표와 타겟 좌표의 위치를 가지고 로봇이 가야 할 최적의 경로를 생성하기 위해서 A* 알고리즘을 사용하였다. A*는 처음 OpenQ 에 들어가는 현 재 로봇의 위치를 기반으로 타켓 좌표까지의 경로를 계산하게 되고, 휴리스틱은 Manhattan Distance 를 사용하였다. 이렇게 연산된 최적의 주행 경로는 위의 위치인식 모듈의 "Step In Map" 과 같이 작용하여 로봇의 움직임을 제어하게 된다..

3.3.3 센싱(Sensing) 모듈

센싱 모듈은 로봇이 주행 도중 센서로부터 입력받은 데이터를 기반으로 장애물을 인식하고, 이를 회피하는 알고리즘이다. 본 연구의 로봇에서는 로봇의 발에 총 4 개의 센서가 부착되어 있으며 (SHARP 전자의 GP2YOA41SKOF 모델), 이는 거리 측정을 할 수 있는 적외선 센서이다. 적외선 센서의 값을 고가의 하드웨어가 아닌 이상 매우 불안정 하기 때문에 센서값을 필터링 한 후에 시스템에 적용해야 한다. 본 연구에서는 불안정한 센서 값을 보정하기 위하여 Median 필터링을 사용하였으며 입력 센서값의 개수는 10 개로 설정하였다[11].

필터링된 센서값에 의해 로봇이 장애물을 회피하게 되는데 기본적으로 앞, 뒤, 좌, 우에 장애물에 있으면 뒤, 앞, 우, 좌의 방향으로 Step 이동을 하지만 그림 5 와 같은 경우에는 다음 수식을 적용하여 계산된 각도만큼 로봇이 움직임을 보정하여 보다 안정적으로 주행 하도록 하였다.



$if\ (a < b)$

$$\cos\theta = \frac{x^2 + (\sqrt{4^2 + x^2} + 4^2)}{2\sqrt{4^2 + x^2} \cdot 4^2}$$

$$\theta = \cos^{-1}\left(\frac{x^2 + (\sqrt{4^2 + x^2} + 4^2)}{2\sqrt{4^2 + x^2} \cdot 4^2}\right)$$

$turnRight\ (\theta)$

(그림 5) 장애물을 만났을 경우의 수식

3.4 물체 인식 모듈

로봇이 수행해야 할 행동으로, 로봇이 물체를 집으려고 할 때 물체가 어디에 있는지, 또 물체와의 거리는 얼마나 되는지 정확히 측정해야 한다. 본 시스템에서는 물체 인식에는 템플릿 매칭을 사용하였으며 물체와 로봇간의 거리 측정을 위해서는 3.3.3 절의 Sensing Module 을 사용하였다. 템플릿 매칭이란 영상인식 기법 중 하나로 소스 이미지에서 찾고자 하는 물체 이미지(템플릿)가 소스 이미지의 어느 부분에 해당하는지를 찾아내는 것이 목적이다. 이 기법은 물체 이미지와 소스 이미지와의 유사성 비교를 통해 최대의 유사성을 가지는 부분을 물체로 인식하도록 한다[12]. 유사성을 구하기 위한 여러 방법이 있지만 본 논문에서는 상관계수(Coefficient of Correlation)법을 이용하였고 유사성을 구한 결과로 얻을 수 있는 소스이미지와 물체이미지와의 상관계수 맵을 통해 최대의 유사성 부분을 찾을 수 있다.

$$R(x,y) = \frac{sum_{x',y'}[T(x',y') \bullet I'(x+x',y+y')]}{sqrt[sum_{x',y'}T'(x',y')^2 \bullet sum_{x',y'}I'(x+x',y+y')^2]}$$

실제로 템플릿 매칭 기법으로 로봇이 물체를 인식할 때는 인식하고자 하는 물체의 정보를 데이터베이스 안에 미리 가지고 있게 하였다.

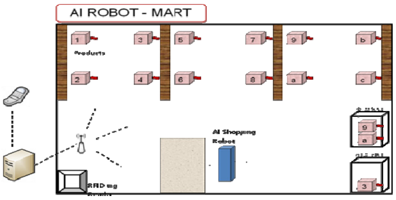
4. 실험 및 결과

4.1 개발 환경 및 맵 구성

본 연구의 로봇 시스템에서는 실제 휴머노이드 로봇이 활동할 수 있는 쇼핑 장소를 휴머노이드 로봇 크기에 맞게 그림 6 와 같이 제작한 후 웹 서버와 소켓서버 그리고 로봇과의 연동을 테스트 하였다. 서버와 로봇의 하드웨어 사양 및 개발 소프트웨어는 아래의 표 1 과 같다.

<표 1> 개발환경

	Web Server	Socket Server	Humanoid Robot
Language	php 5.0	JAVA sdk1.6	C
OS	Redhat Enterprise	Windows XP	Qlus 1.4
Server S/W	Apache 2.2 Eclipse	Apache 2.2 Eclipse	
H/W	P4 1.5Ghz	Quad 2.6Ghz	EV2440 (arm-linux) MR-C3024F(motor control)



(그림 6) 실험 환경

4.2 실험 시나리오

본 연구의 실험에서는 로봇이 사용자로부터 쇼핑 정보를 입력 받아 쇼핑계획을 수립하여, 쇼핑 서비스를 실행할 수 있도록 다음과 같이 시나리오를 구성하였다.

- 사용자가 그림 7 의 인터페이스로 구성된 웹 페이지에 접속하여 쇼핑할 것을 체크한다.
- 전송 버튼을 누르면 웹 서버에서 소켓 서버로 다시 소켓 서버에서 로봇으로 쇼핑 리스트를 전송된다.
- 로봇이 쇼핑을 하기 위한 계획을 생성한다.
- 로봇이 쇼핑 서비스를 실행한다

4.3 시스템 구현

4.3.1 사용자 인터페이스

본 실험에서는 사용자가 핸드폰, 노트북 혹은 PC 를 통해 시스템의 웹 페이지로 접속하여 로봇에게 명령을 할 수 있도록 그림 7 과 같이 구성하였다.

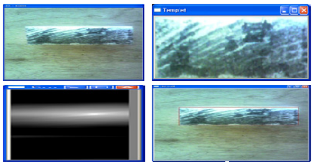
(그림 7) 사용자 인터페이스

4.3.2 장애물 회피 실험

실제 로봇이 생성한 계획대로 움직이면서 장애물을 만났을 때 시스템 설계에서 설명한 위치인식, 맵핑, 센싱 모듈들을 이용해서 장애물을 피할 수 있는지에 대하여 실험하였다. 장애물을 회피하면서 완벽하게 임무를 수행하는 것은 아직 성공률이 높지 않은데 그 이유는 로봇의 배터리 양에 움직임 변화 그리고 로봇 위치인식의 오차 등이다. 본 연구의 장애물 회피 실험에서는 약 70%의 성공률을 보였다.

4.3.3 물체 인식 실험

로봇에 부착된 카메라로 템플릿 매칭을 이용하여 물체 인식을 실험하였다.



(그림 8) 템플릿 매칭을 이용한 영상처리 결과

실험결과를 나타내는 그림 8 은 좌측 상단부터 시계방향 순서로 로봇이 촬영한 실제 영상, 인식하고자 하는 물체의 템플릿 영상, 상관계수 Map 을 나타내는 영상, 그리고 적색 선을 통해 물체 인식을 한 모습을 볼 수 있다.

5. 결론 및 향후 과제

본 논문에서는 휴머노이드 로봇에 관한 다양한 연구를 목적으로 인공지능의 계획기법의 적용, A*알고리즘을 이용한 효율적인 길찾기, 센서값에 기반한 위치인식, 그리고 목적물체를 파악하기 위한 영상인식 등의 알고리즘을 적용시켜 보는 연구를 시도하였다. 이러한 연구를 위한 테스트 영역으로는 사용자를 대신하여 쇼핑을 대행해 주는 것으로 설정하고 이에 따른 실험을 성공적으로 수행하였다.

실험 결과를 분석해 보면, 현재 대부분의 이족로봇의 문제점인 위치인식이 본 연구에서 제안한 "Step in Map" 알고리즘으로 어느 정도 접근이 가능하지만 상황에 따라 여전히 많은 오차를 보이고 있다. 즉, 로봇의 Step 방향이 가야 할 방향에 비해 조금만 차이가 나도, 로봇의 Step 수가 증가할수록 오차의 차이가 커지게 된다. 따라서 단순한 알고리즘의 적용보다는 위치 센서 등과 같은 하드웨어 적인 실험 도구와 병행하는 것도 좋은 접근이라고 생각된다. 또한, 현재의 로봇은 초기 상태에서 주어진 Map 을 미리 가지고 있고, 이에 따라 목표물에 접근하지만, 추후에는 맵을 생성해 가면서 목표물에 접근하는 방법으로 연구가 진행되어야 할 것이다.

참고문헌

[1] 문승빈 외 7 인, "지능형로봇 국내외 표준 동향", 정보과학회지 제 26 권 제 4 호, pp. 79-88, 2008.

[2] "로봇 표준 동향", www.standard.co.kr

[3] World Robotics 2006, International Federation of Robotics, 2006.

[4] 정 슬, "지능로봇시스템 기술", 전자공학회지 제 32 권 제 1 호, pp. 13, 2005.

[5] 유범재, 오용환, 최영진, "휴머노이드 연구동향", 한국정밀공학회지 제 21 권 제 7 호, pp. 15-21, 2004.

[6] 정명진, "해외 로봇 연구 동향", 제어 자동화 시스템공학회지 제 11 권 제 2 호, pp. 28-33, 2005.

[7] 이동욱, 김홍석, 이호길, "감성교감형 로봇 연구동향", 정보과학회지 제 26 권 제 4 호, pp. 65-71, 2008.

[8] Robert Lundh, Lars Karlsson and Alessandro Saffiotti, "Plan-Based Configuration of a Group of Robots", in Proceeding of the 17th European Conference on Artificial Intelligence (ECAI), pp. 1-5, 2006.

[9]Yolanda Gil, "Plan Representation and Reasoning with Description Logics", http://www.isi.edu/~blythe/cs541/Readings/PlanRep.pdf.

[10] 임기현, 서일홍, "이동로봇의 물체인식과 주행을 위한 로봇 지식체계", 전자 공학회 논문지 제 44 권 제 6 호, pp. 435-445, 2007.

[11] Tao Chen and Hong Ren Wu, "Adaptive Impulse Detection Using Center-Weighted Median Filters", IEEE Signal Processing Letters, Volume 8, NO.1, pp. 1-3, 2001.

[12] Clark F. Olson, "Maximum-Likelihood Template Matching", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 2, pp. 2052-2057, 2000.