

다수 이동로봇의 원격제어를 위한 Graphic Man-Machine Interface의 구현

김한영(金翰映), 한헌수(韓獻洙)
승실대학교 공과대학 정보통신전자공학부
전화 : (02) 821-2050 / 팩스 : (02) 821-2050

Implementation of a Graphic Man-Machine Interface for a Teleoperation of Multiple Mobile Robots

Hanyoung Kim, Hernsoo Hahn
School of Electronics Engineering Soongsil University
E-mail : hahn@logos.soongsil.ac.kr

Abstract

The goal of this paper is to provide a Graphic man-machine interface that can be used to control multiple robots simultaneously. The proposed GUI scheme gave emphasis on making multiple robots perform the cooperative works, maintaining a given formation. It controls multiple robots in two different modes. : a group mode and a individual mode. In the group mode, a common goal position and formation are delivered to individual robots at the same time, and in the individual mode one robot is selected. o increase the efficiency of the interface, a time scheduler is provided. The experimental results are included.

I. 서론

Teleoperation이란 작업공간과 멀리 떨어진 곳에서 원하는 작업을 처리하기 위해 로봇을 제어하여 수행하는 것을 의미하며, 최근에는 공장과 같은 산업현장이나 사람이 직접 작업하기에 위험한 핵처리 관련 시설에서 다수의 이동로봇들을 이용한 작업수행이 많이 이루어지고 있다. 이런 과정에서 [8]과 같이 인공지능이 높은 자율이동로봇을 이용할 경우에는 사용자가 로봇의 최종목표만을 설정해 주고 중간과정은 관여하지 않아도 되기 때문에 작업을 간편화할 수 있지만, 높은 안정성과 정확성을 요구하는 환경에서는 오히려 사용자의 감시와 명령에 따라 진행되는 teleoperation의 형태가 긍정적으로 받아들여지고 있다. Teleoperation을 위해서는 사용자와 로봇간의 원활한 정보교환과 명령 전달을 위한 인터페이스가 필요하며, 현재 이 분야에 대한 연구가 활발히 진행되고 있다.

기존에 연구되어진 내용을 개괄적으로 살펴보면, 하나의 이동로봇에 대해 다양한 명령어를 아이콘으로 만들어 로봇의 진행방향과 속도를 조절 할 수 있도록 하는 방법[1], 사용자가 직접 로봇의 모터구동 값을 제어하는 것이 아니라 상위레벨 명령어를 사용하여 중간과

정은 vision system이 장착된 로봇 스스로 해결하도록 하는 방법[2], 터치스크린을 이용해 로봇에 장착되어 있는 카메라로부터 전송되어진 영상을 보여주고 사용자는 이 터치스크린을 통해 로봇이 이용할 목적지를 입력하는 방법[3], 인터넷을 통한 원격 제어 방법[4,5] 등이 제안되었다. 최근에는 다수의 이동로봇이 동일한 공간에서 사용되는 현실적인 요구에 의해 이를 제어하는 방법도 제안되었는데, Suzuki[6]는 카메라로부터 얻어진 영상과 센서측정값을 이용해 각 로봇들의 상태를 확인한 후, 화면에 나타난 로봇의 리스트에서 로봇을 선택하여 명령을 내릴 수 있도록 했다. Khoshnevis[7]는 환경인식, 작업분배, 로봇제어가 가능한 서버컴퓨터와 다수 이동로봇을 이용하여 silicon chip을 원하는 위치로 옮기는 시스템을 제안하였는데, 각 로봇을 제어할 수 있는 아이콘을 만들고 레이저포인터를 이용해 선택하여 로봇의 방향, 속도를 제어할 수 있도록 했다.

그러나 기존의 연구결과를 이용해 다수 로봇을 제어하기에는 여러 문제점이 있는데, [1,3,4,5]와 같은 경우 로봇의 행동을 하나하나 직접 제어해야 하기 때문에 사용자의 부담이 너무 커서 다수의 이동로봇을 제어하는 데에는 적용하기 어렵다. [6,7]에서는 다수의 이동로봇을 제어하기 위한 사용자 인터페이스를 제안하기는 하였으나 실제로 명령을 동시에 내리는 것이 아니라 로봇 하나를 선택한 후, 명령을 각각 구별해서 내려야 하기 때문에 명령전달시간이 오래 걸려 다수의 이동로봇을 이용한 공동작업이 효율적으로 수행되기 어렵다.

본 논문에서는 효과적으로 다수의 이동로봇을 동시에 제어할 수 있는 Graphic Man-Machine Interface를 제안한다. 로봇 제어는 개별 제어뿐만 아니라 실시간으로 다수의 이동로봇을 제어하기 위해 그룹으로 묶어 이동을 명령하는 것도 가능하며, 목적지까지의 이동경로 산출과 장애물 회피를 자동으로 수행하고, 그룹으로 동작하는 로봇들이 이동하는 동안에 각 로봇의 상대적인 위치를 지정할 수 있어 구조적인 제어가 가능하도록 했다.

II. 시스템 구성

전체적인 시스템 구조는 그림 1과 같이 다수의 이동

로봇과 이를 제어할 수 있는 사용자 인터페이스, CCD 카메라를 이용해 작업환경을 보여주는 영상시스템, 명령을 포함한 데이터의 전송을 위한 양방향 통신 시스템, 로봇구동모듈 등으로 나누어 질 수 있다.

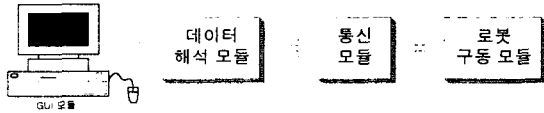


그림 1. 시스템 모듈

GUI 모듈은 로봇으로부터 얻어진 작업환경에 대한 정보를 사용자에게 일목요연하게 보여주고, 사용자로부터 로봇의 속도, 위치, 작업수행 등 제어에 관련된 명령을 입력받는 부분이다. 데이터 해석 모듈은 사용자가 입력한 명령과 변수 값을 해석하여 각 로봇에게 전달되는 패킷형태로 바꾸어주는 역할과 로봇으로부터 전달되어진 정보들을 분석하는 역할을 하고, 통신모듈은 송신부와 수신부가 명시된 패킷형태의 데이터를 전송하기 위한 양방향 채널을 설정하여 전송하는 일과 통신 중에 발생할 수 있는 여러 가지 문제들을 해결하는 역할을 한다. 그리고 로봇구동모듈은 로봇의 행동을 직접 제어하는 부분으로 사용자로부터 전달되어진 명령을 로봇이 이해할 수 있는 모터구동 값으로 변환하는 역할을 한다.

로봇의 종류는 작업환경을 감시하기 위해 CCD카메라를 장착한 로봇과 작업수행을 위한 일반 로봇으로 나누어지며, 본 논문에서 사용하는 이동로봇은 2개의 모터를 차륜구동방식으로 제어하여 이동한다.

III. 사용자 인터페이스 구성

3.1 로봇 제어를 위한 인터페이스

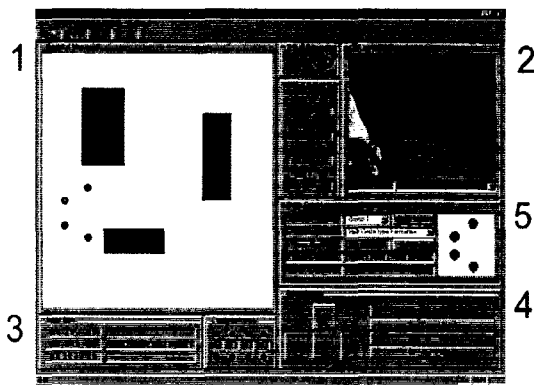


그림 2. GUI화면 구성

Windows O/S에서 실행되는 사용자 인터페이스의 전체적인 모습은 그림 2와 같이 실제 작업환경을 축소시켜놓은 graphical map(그림 2의 1), 작업의 수행과정을 감시할 수 있는 CCD영상 창(그림 2의 2), 선택된 로봇의 상태를 보여주는 창(그림 2의 3), 명령 아이콘의 집합(그림 2의 4), 그리고 그룹제어를 위한 창(그림 2의 5)으로 구성된다.

3.1.1 Graphical Map

Graphical map은 실제 작업이 이루어지는 환경을 일정비율로 축소시켜 놓은 지도로서 고정장애물과 작업을 수행하고 있는 로봇들을 관찰할 수 있는 창이다. 이러한 graphical map은 작업환경이 그려진 일반 비트맵 파일을 입력으로 받아 자동으로 고정장애물의 위치를 파악한 후, 데이터 베이스에 저장하여 로봇의 이동경로 산출 및 장애물 회피에 이용한다.

graphical map은 현재의 작업환경을 가장 빠르게 인식할 수 있는 도구로서 실시간으로 현재 로봇의 위치와 이동방향 등을 보여주며, 로봇들이 작업중인가의 여부와 그룹에 속해 있는가의 여부를 사용자가 쉽게 알아볼 수 있도록 구별하여 표시해 주기 때문에 사용자는 명령을 내릴 로봇의 현재 상태를 직관적으로 알 수 있다. graphical map을 통해 로봇은 개별 또는 그룹으로 제어가 가능하며 그룹제어의 경우, 각 로봇의 상대적인 위치지정을 포함하여 공통의 목적지를 지정할 수 있어 로봇 모두에게 이동 목적지를 각각 지정해 주어야 하는 번거로운 작업을 생략할 수 있다. 그리고 다수 이동로봇의 그룹 지정은 마우스를 이용하여 로봇을 그룹에 포함시키거나 그룹에서 제거시킬 있으며, 마우스의 오른쪽 버튼을 누르면 나타나는 동적 메뉴로 로봇이 해야 할 행동을 직접 명령할 수 있도록 하였다.

3.1.2 CCD영상 출력 창

로봇에 장착되어져 있는 CCD카메라로부터 얻어진 영상을 보여주는 창으로 사용자는 실시간 영상 정보를 이용하여 작업환경의 상태를 확인할 수 있으며, 여러대의 카메라를 요소 요소에 배치하고 선택적으로 출력하는 것이 가능하기 때문에 넓은 작업공간을 효율적으로 관리할 수 있다.

3.1.3 Robot 상태 창

작업을 수행중인 로봇에 관한 정보를 보여주는 창으로 현재 선택되어 있는 로봇, 작업중인 로봇, 고장으로 더 이상 작업수행이 곤란한 로봇의 리스트들을 보여주고, 서버 컴퓨터와 로봇 사이의 데이터 통신상태를 표시해 주는 부분으로 구성되어 있다. 그리고 정상적으로 통신이 이루어지지 않을 경우 고장난 로봇으로 지정한다.

3.1.4 명령 아이콘

명령 아이콘은 표 1과 같이 graphical map을 통한 명령입력을 보완하는 방법으로 로봇을 제어하는 데 사용되는 명령을 버튼형태의 아이콘으로 만들어 놓은 것이다. 조이스틱과 같이 로봇의 모터구동 값을 직접 제어할 수 있는 화살표 모양의 아이콘들과 위험한 상황이 갑자기 발생했을 경우, 이에 대처하기 위해 일시정지 같은 명령 아이콘 등으로 구성되어져 있다.

3.1.5 그룹제어 창

그룹제어 창은 다수의 로봇을 하나의 로봇처럼 그룹으로 묶어 제어하기 위한 인터페이스를 제공한다. 예를 들어 그룹의 구성원을 선택하여 저장하거나 미리 저장된 그룹 정보를 불러올 수 있고, 그룹으로 지정된 로봇간의 상대적인 위치를 조절하기 위해 대형

(formation)을 지정할 수 있다. 특히 현재 설정되어 있는 대형의 종류와 현재 선택된 로봇의 수에 따라 달라지는 위치와 모양을 미리 볼 수 있어 그룹을 충돌 없이 안전하게 제어할 수 있도록 했다.

3.1.6 상세 정보 창

사용자가 graphical map에 나타난 로봇을 선택하면 그 로봇의 자세한 정보를 보여주는 창이 활성화되는데, 로봇의 특성과 현재 수행하고 있는 명령, 위치, 방향을 기본적으로 보여주며, 로봇에 장착된 센서로부터 얻어진 정보를 표시해 준다. 초음파 센서의 경우 측정값을 분석하여 장애물의 위치와 방향에 관한 정보를 얻어 주변의 장애물 분포를 보여주며, 로봇 둘레에 장착되어 있는 접촉센서는 장애물과 로봇의 어느 위치가 충돌했는가에 대한 정보를 제공한다.

표 1 사용되는 기본 명령어의 종류

명령어	행동 내용
go forward	로봇을 현재 이동방향으로 전진
go backward	로봇을 후진
turn left	로봇의 이동방향을 왼쪽으로 전환
turn right	로봇의 이동방향을 오른쪽으로 전환
goal point	목적지까지의 경로를 산출하여 이동
go init pos	로봇을 초기 위치로 이동
stop	해당 로봇의 작업을 일시 중지
all stop	모든 로봇의 작업을 일시 중지

3.2 로봇 배치(Formation)

로봇을 그룹으로 묶어 제어하는 경우, 각 로봇의 배치와 관련하여 상대적인 위치를 조절할 수 있는 기능이 필요하다. 예를 들어 그룹으로 지정된 로봇들이 이동한 후 목표 지점을 중심으로 횡으로 정렬된 형태를 유지하도록 할 경우, 사용자는 로봇의 목적지를 각각 지정해 주어 수동으로 횡정렬 대형을 형성하도록 하는 것이 아니라 로봇그룹에게는 공통적인 목적지를 지정하면서 그룹 대형만 선택해 주면 자동으로 각 로봇들간의 상대적인 위치를 계산하여 원하고자 하는 로봇배치 형태(formation)의 종류는 그림 3과 같다.

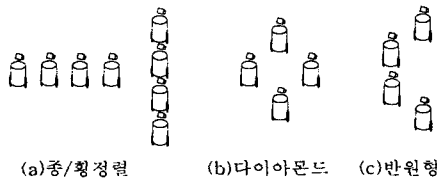


그림 3. 로봇 그룹의 대형

우선 종/횡 정렬형은 목적지 앞에서 로봇을 일렬로 늘어선 형태로 배치하고, 다이아몬드형은 목적지를 중심으로 둘러싸는 모양을 만든다. 한편 반원형은 벽에 붙어있는 목적지를 둘러싸는 경우에 적합하다. 대형을 사용하는 주된 이유는 각 로봇들이 충돌 없이 지정된 목적지로 접근이 용이하도록 함으로써 작업을 순조롭게 처리하기 위해서이다. 이러한 그룹 대형을 지정함으로써 다수의 이동로봇을 더욱 효율적으로 제어할 수 있다.

3.3 작업 계획 및 관리(Task Scheduler)

다수의 이동 로봇들을 이용해 작업을 명령하고 수행할 경우, 각 로봇간에 시간적인 행동순서가 중요하게 작용할 때가 있는데, 이런 경우를 위해 Task scheduler를 제공한다. 만약 1번 로봇이 A 작업을 수행한 후에 2번 로봇이 B 작업을 수행하고, 마지막으로 3번 로봇이 C 작업을 수행한다고 했을 때, 명령을 내리는 사용자는 각 로봇이 해야할 행동과 시간적 순서를 미리 지정하여 하나의 작업단위로 저장한 후, 명령을 내릴 때는 이를 아이콘화 하여 쉽게 불러 쓸 수 있도록 하는 기능이다. Task scheduler는 사용자 인터페이스와 같이 GUI형태로 제공되며, 작업할 로봇, 작업 종류, 작업의 순서를 지정할 수 있다. Task Scheduler의 이용으로 사용자는 긴 시간을 필요로 하는 작업을 명령할 경우 작업을 간편하게 명령할 수 있어 불필요한 시간 낭비를 줄일 수 있다. 그리고 로봇들이 수행할 작업에 시간적인 순서를 부여함으로써 구조화된 작업을 명령할 수 있기 때문에 시스템의 응용범위를 확대시킬 수 있다.

IV. 시스템 구현

4.1 로봇 제어

로봇 제어를 위한 방법은 크게 개별 제어와 그룹 제어로 나누어지는데, 기본이 되는 개별 제어는 마우스를 이용해 graphical map에서 명령을 내릴 로봇을 선택하고 해당 로봇에 대한 정보가 화면에 나타나면, 사용자는 현재 상황을 파악한 후 로봇이 이동할 목적지를 방향, 이동거리 등의 값으로 지정한다. 물론 이런 과정은 마우스를 이용해 graphical map에 최종의 목적지만을 지정하면 자동으로 수행된다. 로봇을 그룹으로 묶어 제어하는 경우, 다수의 이동로봇을 그룹으로 지정하기 위해 마우스를 graphical map상에서 움직여 생기는 영역 안에 위치하고 있는 모든 로봇들을 선택할 수도 있고, 하나 하나 마우스로 선택해서 그룹에 포함시키거나 제거시킬 수 있다. 그리고 미리 준비된 로봇의 배치형태를 지정하여 로봇들간의 상대적 위치를 쉽게 조절할 수 있고, 선택된 배치형태를 유지하면서 이동하는 것도 가능하다.

4.2 이동경로 산출

로봇의 목적지 위치가 입력으로 들어오면 현재 위치에서 목적지까지의 이동경로를 산출해 내어야 하는데, 이 논문에서는 문제를 해결하기 위해 영상처리에서 Line following을 위해 주로 사용되는 A* 알고리즘을 로봇들과 장애물 정보를 가지고 있는 graphical map image에 응용하였다. A* 알고리즘은 현재 위치에서 다음 이동방향을 찾기 위해 근접한 8픽셀(Pixel) 중에 장애물이 존재하는 방향과 이전 위치를 제외한 나머지 방향을 검사하여 임시버퍼에 저장한 후 여러 이동경로 후보 중에 최소경로비용을 가지는 것을 선택하여 목적지까지의 이동경로를 산출하는 방법으로 다음 식에 의해 결정된다.

$$비용 = 이동거리 * \kappa_1 + 남은거리 * \kappa_2 \quad (1)$$

(κ_1, κ_2 는 각각에 대한 가중치)

4.3 장애물 회피

로봇이 이동할 때, 장애물과의 충돌회피 기능은 teleoperation에서 기본적인 기능으로 여겨질 만큼 중요하다. 우선 고정된 장애물 회피는 A* 알고리즘을 통해 산출된 이동경로에서 이미 고려되어졌기 때문에 여기서는 이동장애물, 즉 자신 이외의 로봇과의 충돌 회피에 대해서만 다루었다. 장애물 회피 알고리즘은 다음과 같이 두 부분으로 나누어져 있다. 1차 장애물 회피에서는 이동하고 있는 로봇에 대해 적용하고, 2차 장애물 회피는 정지해 있는 로봇에 적용한다.

▶ 1차 장애물 회피 알고리즘

- step 1. 상대 로봇이 미리 정의된 충돌 위험거리 안에 들어와 있는가?
- step 2. 이동 방향 앞쪽에 있는가?
- step 3. 이동하고 있는 로봇인가?
- step 4. 상대 로봇 이동방향이 충돌 가능한가?
- step 5. 충돌 가능하다고 판단되면 우선 순위가 높은 로봇이 다른 로봇을 정지시킨 후, 이동한다
- step 6. step 5에서 정지시켰던 로봇과 안전거리 이상 벌어지면 정지 신호를 해제해준다

▶ 2차 장애물 회피 알고리즘

- step 1. 정지해 있는 로봇을 만났을 경우, 현재 이동 방향을 가로막고 있는가?
- step 2. 가로막고 있다면 로봇을 고정장애물로 인식하고 현재 위치에서 목적지까지 다시 이동경로를 산출한다.

2차 장애물 회피의 step 1에서 정지해 있는 로봇이란 실제 모든 명령을 수행하고 정지해 있는 로봇과 1차 장애물 회피 과정에서 정지신호를 받은 로봇 두 가지 모두 포함된다.

4.4 무선 통신 기능

로봇이 이해하고 수행할 수 있는 형태로 변환된 명령어와 로봇으로부터 얻은 정보는 RF모듈을 통해 양방향으로 전달된다. 통신상에서 사용되는 데이터의 형식은 송신부와 수신부의 ID, 전송되어질 데이터의 크기 및 종류, 실제 데이터, 그리고 checksum 같은 코드를 하나의 패킷으로 만들어 전송한다. 그리고 통신 모듈은 전송에 대한 응답이 계속 없을 경우 인터럽트를 통해 현재 로봇의 상태를 파악하여 사용자에게 알린 후, 문제 해결을 위한 조치를 스스로 내리거나 작업수행을 정지시킨 후, 사용자의 명령을 기다리도록 한다.

V. 실험 및 고찰

모의 실험은 가로 10m, 세로 10m의 가상 공간에서 실행되었으며, 그림 4는 네 대의 이동로봇을 그룹으로 선택하여 다이아몬드형의 대형을 형성하고 유지하며 이동하는 작업을 수행한 결과를 보여준 것이다.

실제 실험은 제안된 사용자 인터페이스를 로봇측구에 적용해 보았다. 사용자는 CCD카메라를 통해 전송되어진 영상을 이용하여 작업환경을 인식하고 제안된 인터페이스를 이용해 다수의 이동로봇들을 빠르고 간편하게 제어할 수 있었다.

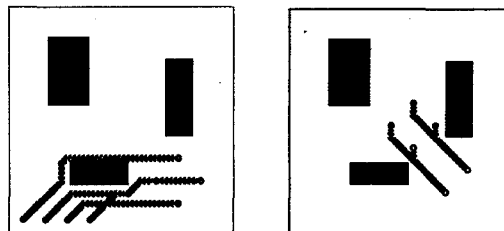


그림 4. formation 형성과 유지

VI. 결론

본 논문에서는 기존 연구에서 사용되었던 사용자 인터페이스가 다수의 로봇으로 공동작업을 수행하는데 가지는 한계를 해소하기 위한 Graphical man-machine interface 기법을 제안한다. 로봇은 개별 제어뿐만 아니라 실시간으로 다수 이동로봇을 이용하기 위해 그룹으로 묶어 제어할 수 있다. 그리고 이동경로 산출 및 장애물 회피, 로봇그룹이 대형을 형성한 후 유지하며 이동하는 것이 가능하고, 수행할 작업들의 순서를 지정하여 일괄적으로 명령, 처리할 수 있도록 했다.

실험결과, 제안한 방법이 다수의 이동로봇을 실시간으로 제어할 수 있는 장점을 갖음을 확인하였으나, 서버컴퓨터의 부하를 분산시키는 문제와 통신모듈의 속도와 신뢰성을 향상시키는 일이 해결해야 할 과제로 남는다.

참고문헌

- [1] Piotr Skrzypczyński, "Supervision and Teleoperation System for an Autonomous Mobile Robot", pp. 1177-1181, IEEE IROS, 1997.
- [2] Tomohiro Shibata, Yoshio Matsumoto, Taichi Kuwahara, Masayuki Inaba, and Hirochika Inoue, "Hyper Scooter : a Mobile Robot Sharing Visual Information with a Human", pp. 1074-1079, IEEE Robotics and Automation, 1995.
- [3] Takeshi Sekimoto, Takashi Tsubouchi and Shin'ichi Yuta, "A Simple Driving Device for a Vehicle", pp. 147-154, IEEE IROS, 1997.
- [4] Tse Min Chen and Ren C. Luo, "Multisensor Based Autonomous Mobile Robot Through Internet Control", pp. 1248-1253, IECON, 1997.
- [5] Tse Min Chen and Ren C. Luo, "Remote Supervisory Control of an Autonomous Mobile Robot via World Wide Web", pp. ss60-ss64, IEEE ISIE, 1997.
- [6] Tsuyoshi Suzuki, Teruo Fujii, Hajime Asama, Kazutaka Yokota, Hayato Kaetsu, Nobuo Mitomo and Isao Endo, "Cooperation between a Human Operator and Multiple Robots for Maintenance Tasks at a Distance", Distributed Autonomous Robotic Systems, vol. II, pp. 50-58, Springer, 1996.
- [7] Behrokh Khoshnevis, "Centralized Sensing and Control of Multiple Mobile Robots", pp. 543-546, Proceeding on Computers and Industrial Engineering, 1997.
- [8] Ken Sugawara and Masaki Sano, "Cooperative Acceleration for Task Performance : Foraging Behavior of Interacting Multi-robots System", pp. 343-354, Elsevier Science B.V, 1997.