

모바일 센서차량을 위한 내장형 소프트웨어 설계

이주현[○] 이현승 송하운 박준
홍익대학교 컴퓨터공학과

jooheony[○]@hanmail.net lhseung@oodb.cs.hongik.ac.kr, hayoon@wow.hongik.ac.kr, jpark@hongik.ac.kr

Embedded Software Architecture For Mobile Sensor Vehicle

Joo Heon Lee[○], Hyun Seung Lee, Ha Yoon Song, Joon Park
Department of Computer Engineering, Hongik University

요 약

본 논문에서는 모바일 센서 차량의 다양한 기능을 구현한 소프트웨어를 설계하였다. 구현한 모바일 센서 차량은 주행하며 카메라로 장애물을 인식하고 지도를 그리며 다른 모바일 센서 차량과 정보를 교환한다. 따라서 구현을 위해서 다차원적인 시스템을 고려해야 하며 각 구현 부분들이 유기적으로 데이터 교환 및 처리가 이루어 져야 한다. 이러한 시스템을 구현하기 위하여 클래스 기반으로 각 구현 부분을 나누고 통합적으로 운용되는 소프트웨어 시스템을 구현하였다.

1. 서 론

모바일 센서 차량이 주행하는 데에는 각도와 이동거리의 2가지 정보를 이용한다. 즉 목표로 하는 지점으로 방위각을 잡고 이동하면 이동한 만큼의 정보를 가지고 스스로 지도를 그린다. 하지만 주행 중에 나타나는 각종 장애물 및 다른 차량과의 정보 교환을 통해서 좀 더 다차원적인 지도를 정확하게 그릴 수 있다. 본 논문에서는 소프트웨어적인 관점에서 모바일 센서 차량의 시스템을 설계 및 구현하였다.

본 논문의 구성은 2장에서 차량의 소프트웨어 시스템 설계에 관한 관련 연구를 설명하고 3장에서는 각 차량의 하드웨어 및 소프트웨어를 설명한다. 4장에서는 각각의 차량이 얻은 정보를 가지고 지도로 만드는 시스템을 설명하고 5장에서는 향후 연구방향을 제안하고 결론을 맺는다.

2. 관련 연구

센서 차량은 스스로 환경의 변화를 감지하고 변화에 적절히 대응해야 하며 따라서 센서 차량이 같은 목적을 수행 하더라도 그 상황과 감지된 정보에 따라서 다른 행동

을 할 수 있어야 한다. 이러한 지능형 모바일 센서 차량의 설계를 위해서는 각각의 센서들이 얻은 정보를 가공하고 통합 관리가 가능한 체계적인 소프트웨어 시스템이 필요하게 된다. 모바일 센서 차량의 각 기능을 효과적으로 통합하고 관리하기 위해서 고려해야 할 사항[1]은 3가지로 나뉘볼 수 있다.

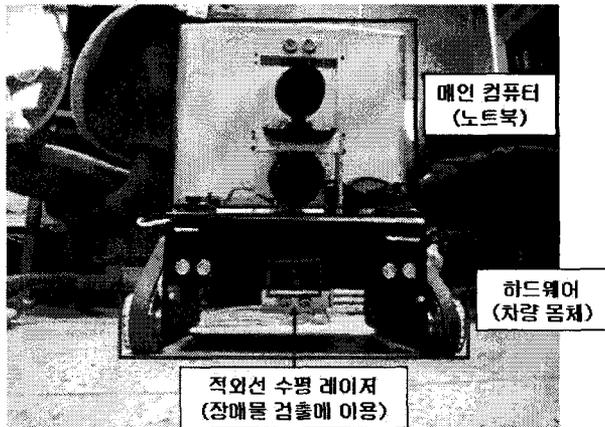
첫째로 개방성이다. 표준적이며 개방적 인터페이스를 통하여 다양한 기술과 접목 될 수 있어야 한다. 모바일 센서 차량의 실현을 위한 각 요소들이 표준적이고 개방적인 인터페이스를 가져야만 다양한 기술 개발자들이 참여하여 효과적인 시스템 통합을 이룰 수 있다.

둘째는 확장성으로 새로운 기능을 추가하거나 개선하는 것이 용이하고 기존의 시스템에 미치는 영향이 최소화 되어야 한다는 점이다. 새로운 기능이 지속적으로 추가되거나 개선될 수 있기 때문에 중요한 요건이 된다.

셋째로 호환성이다. 센서 차량의 기능을 요소 단위로 교체하거나 다른 센서 차량과 공유될 수 있어야 한다. 모바일 센서 차량에 적용되거나 습득된 기술을 다른 센서 차량과 공유하기 위해서는 기능 요소들 간에 호환성이 요구된다.

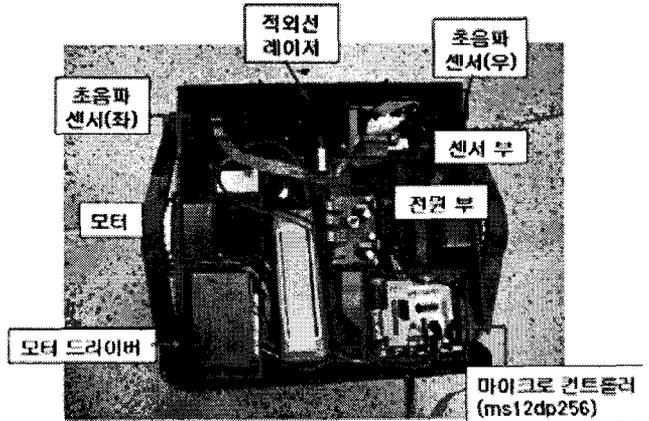
3. 모바일 센서 차량의 구현

모바일 센서 차량의 구현은 하드웨어 부분과 소프트웨어 부분으로 나누어진다. 하드웨어부분은 소형 마이크로 컨트롤러에서 컨트롤이 이루어지며 메인 컴퓨터의 소프트웨어 시스템과 제어 데이터를 주고받는다. 데이터는 RS-232 시리얼 통신 규격을 이용하여 정해진 패킷 규격을 통해서 데이터의 송수신이 이루어진다.



[그림 1. 구현한 모바일 센서 차량]

컴퓨터의 소프트웨어는 마이크로컨트롤러에서 보내준 정보를 이용하여 주행에 필요한 정보로 가공한다. 크게 5부분으로 나누어진다. 우선 하드웨어를 제어하는 T-board control 클래스는 하드웨어의 마이크로컨트롤러와 시리얼 통신을 담당하며 주행에 관한 정보와 전자 나침반의 정보를 얻는다. 현 방위각에서 얼마만큼 이동했는가에 대한 정보를 가지고 차량의 위치를 갱신하게 된

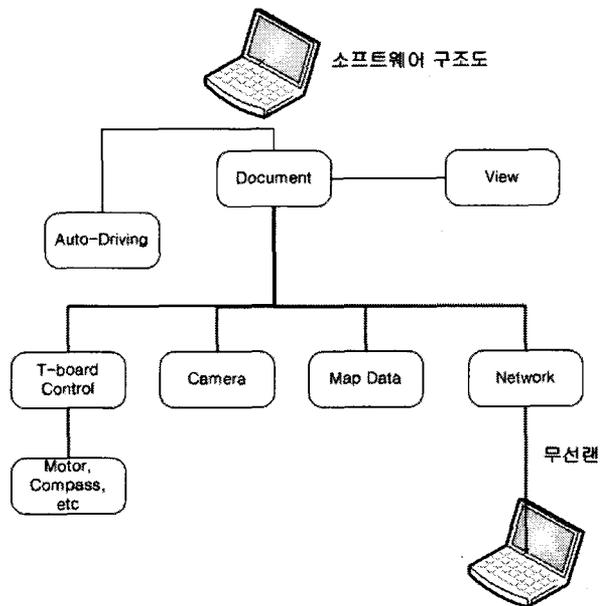


[그림 2. 센서 차량을 구성하는 하드웨어]

3.1 하드웨어

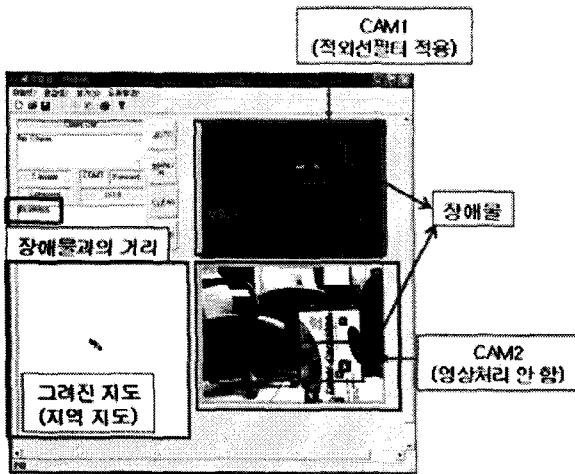
모바일 센서 차량을 구성하는 하드웨어의 마이크로프로세서는 주행에 필요한 각 부분을 제어한다. 각 부분은 마이크로프로세서가 제어하며 크게 바퀴를 구동하는 모터부분과 적외선 레이저, 초음파 센서, 각 센서와 카메라의 회전을 제어하는 센서부, 마이크로컨트롤러와 메인 컴퓨터의 소프트웨어와의 통신을 할 RS-232시리얼 포트가 있게 된다. 즉 메인 컴퓨터에서 요청한 신호로 각 부분을 제어하고 필요한 정보를 시리얼 통신(RS-232)을 통해서 전달한다. 차량을 2계층으로 나누어서 제어하는 것은 소프트웨어의 개방성, 호환성, 확장성을 하드웨어와 별개로 보장하기 위함이다. 따라서 패킷 양식을 설계하고 하위레벨에서는 마이크로 컨트롤러가 직접 제어하고 상위레벨에서는 소프트웨어가 복잡하고 다차원적인 데이터를 처리하게 된다.

3.2 소프트웨어



[그림 3. 소프트웨어 구조도]

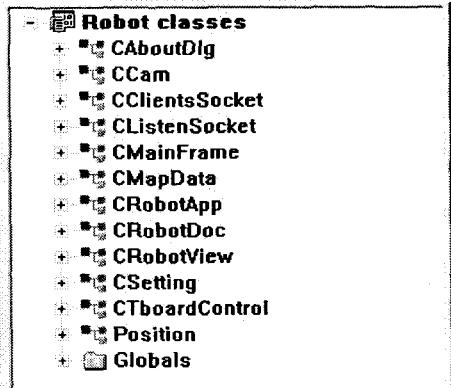
다. Camera 클래스는 카메라에서 얻은 정보(p3p알고리즘)[2]를 가지고 장애물의 크기와 거리를 측정하여 구조체 형식으로 저장하여 차후에 지도 작성 시 이용할 수 있도록 하며, Map Data 클래스는 T-board클래스에서 얻은 위치 정보와 Camera 클래스에서 얻은 장애물 정보를 가지고 지도를 만든다. 이 지도 정보를 가지고 자율 주행을 하며 Network 클래스는 현재 가진 정보를 다른 차량과 교환을 한다. 센서 차량을 구성하는 소프트웨어의 전체적인 구조도는 [그림 3]과 같다.



[그림 4. 센서차량 소프트웨어 프로그램]

시스템은 각 부분을 담당하는 클래스들과 전체적인 주행을 담당하는 Auto-driving 쓰레드 함수로 구성되는데 Document 클래스를 통해서 정보를 교환하게 되며 이 때 주행을 담당하는 Auto-driving 쓰레드는 주기적으로 이동한 위치의 갱신, 맵의 갱신, 주행의 방향 결정 등을 하게 된다. View 클래스를 통해서는 지도를 화면에 표시하게 되어 사용자가 확인 할 수 있도록 한다.

소프트웨어를 설계하고 구현함에 있어서 가장 중요한 부분은 각 데이터의 유기적인 흐름이다. 즉 클래스로 독립적으로 구성하면서 데이터는 공유해야 하는 부분이 많으므로 그 통로 역할을 하는 Document 클래스를 활용하여 데이터의 교환이 유기적으로 되도록 구현하였다.



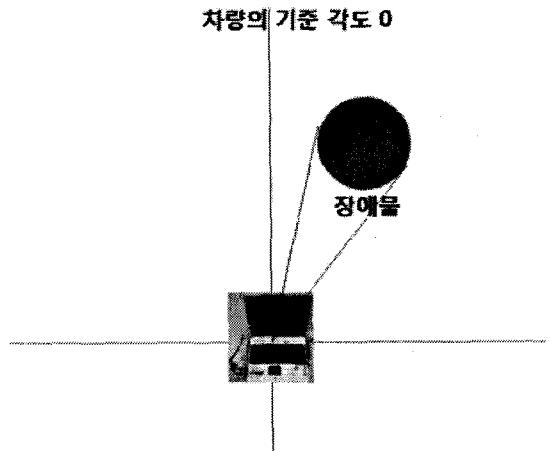
[그림 5. 클래스 리스트]

4. 지도 정보의 구현

각 차량이 얻은 정보를 가지고 자기 스스로의 지도를 작성하여 다른 차량과의 정보를 교환하여 종합적인 맵을 만들게 된다. 이때 맵에 표시하는 정보는 다음과 같다.

- 1) 아직 탐사되지 않은 지역(흰색)
- 2) 장애물이 있는 지역(붉은색)
- 3) 다른 로봇이 있는 지역(푸른색)
- 4) 이미 탐사한 지역(초록색)
- 5) 도달할 수 없는 지역(붉은색)
- 6) 기타영역(미정)

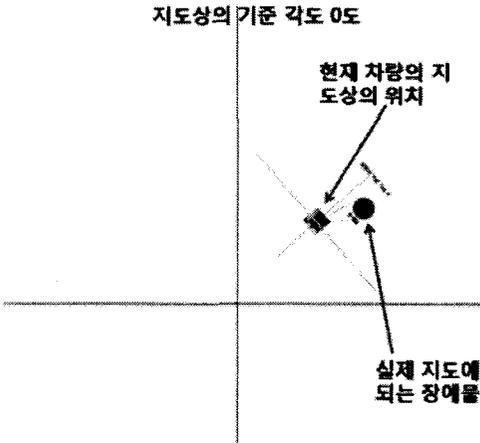
카메라에서 파악한 장애물 정보는 차량 기준으로 만들어진 정보이다. 즉 차량의 정면을 0도로 생각했을 때 그것을 기준으로 장애물의 위치를 파악한 것이다.



[그림 5. 차량 기준 좌표계]

이 장애물 정보를 가지고 차량의 위치정보에 좌표변환을 시켜서 지역 지도를 만든다. 지도 기준의 좌표는 방위각을 기준으로 차량이 방향을 바꾸어도 항상 일정하게 각도를 유지해 준다.

지도는 1차적으로 지역 지도를 만들고 이후에 이 지역 지도로 전역 지도를 만든다.



[그림 6. 지도 기준 좌표계]

지역 지도는 [그림 4]의 지도 부분처럼 Grid Map으로 구성을 하게 된다.[3] 즉 주변 환경을 등간격의 격자로 표현하며 센서에 표시된 정보를 이용하여 각 격자에 물체 존재 유무를 표시하게 된다. 각 센서 차량에서 구성한 지역 지도를 통합하여 전역 지도를 표현하게 되는데 Grid Map 기반의 전역 지도는 큰 용량으로 인해 메모리를 많이 차지하고 주변 환경의 상세한 표현을 위해서는 세밀한 격자가 요구되므로 어떤 기능을 수행하는데 많은 시간이 소요되는 단점을 갖는다. 따라서 구성된 Grid Map 정보를 이용하여 Topology Map[4]을 구성함으로써 센서 차량의 신속한 경로 계획과 주변 환경 정보를 나타낼 수 있다.

5. 결 론

본 논문에서 구현한 모바일 주행 차량은 각종 센서를 탑재하고 하위레벨의 마이크로프로세서와 상위레벨의 컴퓨터 시스템으로 분리하여 확장성 및 호환성을 유지하기 위해 설계되었다. 센서 차량이 주행하면서 지역 지도를 구성함으로써 주변 환경 정보를 지도에 표현하게 된다.

향후 연구과제로 여러 센서차량의 유기적인 통신을 위한 다른 센서 차량과의 지도 정보 교환을 연구할 계획이

며 이를 위한 지역화 방법의 절차는 먼저 무선 랜의 RSSI신호로 대략적인 위치를 추정하고, 카메라로 다른 차량을 식별한 후 서로의 통신절차를 통해서 정확하게 된다.[5] 일단 차량의 위치를 결정하게 되면 자기 자신의 지도를 송신하게 된다. 그러면 상대 차량은 수신된 지도를 가지고 각 차량의 상대 위치를 적용하여 자신의 지도를 확장할 수 있다. 본 논문에서 구현한 센서 차량 프로그램을 이용하여 넓은 전역 지도 구성과 여러 센서 차량들이 특정 패턴이나 그룹을 지어 이동하는 기법 등의 연구가 진행될 예정이다.

6. Acknowledgement

본 연구는 『서울시 산학연 협력사업』(Seoul R&BD Program)의 일환으로 수행하였다.

7. 참고문헌

- [1] 광별샘, 이재호. 지능형 로봇 제어를 위한 시스템 통합 방법론, 한국정보과학회지(Communications of the Korea Information Science Society) 제24권3호, 2006
- [2] David Nist'er, Sarnoff Corporation CN5300, Princeton, NJ-08530, USA. A Minimal Solution to the Generalized 3-Point Pose Problem. Journal of Mathematical Imaging and Vision archive Volume 27, 2007
- [3] Benjamin Kuipers, Joseph Modayil. Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy. IEEE International Conference Robotics and Automation(ICRA) Proceedings, 2004
- [4] 권태범, 송재복, 지역 및 전역 환경에 대한 세션화 기반 위상지도의 작성. 제어자동화 시스템공학 논문지, 2006
- [5] Aram Galstyan, Bhaskar Krishnamachari. Distributed Online Localization in Sensor Networks Using a Moving Target. Information Processing In Sensor Networks, Proceedings of the third international symposium on Information processing in sensor networks, 2004