

크리켓 실내 위치 시스템에서 실시간 무인 로봇 위치 추정

유재봉^o 서정택 최현식 박찬영
한림대학교

yoojaebong, gdknight, ullezi, cypark@hallym.ac.kr

Real-Time Unmanned Robot Tracking in the Cricket Indoor Location System

Jaebong Yoo^o Jungtaek Seo Heonsik Choi Chanyoung Park
Hallym University

요 약

유비쿼터스 컴퓨팅 애플리케이션은 빈번히 위치 정보를 이용하기 때문에 모바일 노드들의 위치를 파악하여 서비스를 제공해 주는 것이 점차 중요하게 인식되고 있다. 여기서 모바일 노드들의 위치 추정은 센서 네트워크에서 노드들에게 좌표를 할당하는 것과 같은 의미이다. 센서 네트워크 환경에서 모바일 노드 위치 추정을 다루는 연구들이 늘어나고 있으나, 현재 제안된 거의 모든 연구들의 위치 추정 실험에서는 엔코더, 자이로, 가속도 센서 등을 이용해서 실제 이동량을 측정해야만 한다.

이 점에 착안하여, 본 연구에서는 어떤 추가적인 센서를 추가하지 않고 실험에서 얻은 모바일 노드의 좌표들을 카메라로 획득한 실제 영상에 맵핑시킴으로써 위치 추정 실험을 검증하기 위한 방법론을 제시하고, 제안한 방법론을 이용하여 크리켓 실내 위치 시스템(Cricket Indoor Location System)의 대표적인 두 가지 구조인 Active and Passive Infrastructure에서 무인 로봇의 위치 추정 실험 결과를 보여 준다.

1. 서론

모바일 노드의 위치를 결정하는 것은 모바일 컴퓨팅에서 매우 중요한 문제이다. 그래서 GPS(Global Positioning System)가 잘 동작되지 않는 실내 환경에서는 위치 정보를 제공하기 위한 특별한 시스템을 설계하고 구현해야만 한다[1, 2]. 현재까지 제안된 대표적인 실내 위치 추정 시스템으로 적외선 신호를 이용하는 Active Badge System[3], 초음파 펄스를 이용하는 Active Bats System[4], RF와 초음파의 속도 차이를 이용한 Cricket Location System[5], UWB(Ultra Wide Band)를 이용한 위치 추정 시스템[6], 영상 인식을 이용한 위치 추정 시스템인 Easy Living Project[7], IEEE 802.11 Wireless LAN 환경 기반인 Microsoft사의 RADAR[8], 그리고 RF 신호의 도달시간을 측정하여 위치를 파악하는 PinPoint사의 3D-iD[9] 등이 있다.

일반적으로, 위치 추정 알고리즘(Localization Algorithm)이 이러한 실내 위치 추정 시스템에서 적용되어 실제 환경에서 구현되어질 때, 가장 심각한 문제는 LOS(Line-of-sight) 연결을 방해하는 장애물을 고려하여 모바일 노드의 위치를 획득하는 것이 어렵다는 점이다[10]. 센서 네트워크 환경에서 모바일 노드 위치 추정을 주제로 많은 연구가 진행되고 있으며, 좋은 결과들이 계속해서 나오고 있다.

위치 추정 알고리즘을 검증하기 위해서 실제 이동한 좌표와 실험에서 얻은 좌표를 비교분석해야 한다. 하지만, 실제 이동한 좌표를 얻기 위해서는 엔코더, 자이로 센서, 가속도 센서 등을 이용해서 실제 이동량을 측정해야 한다.

본 연구에서는 이런 점을 착안하여 실험에서 얻은 모바일 노드의 위치 좌표들을 카메라로 획득한 실제 이미지에 맵핑시킴으로써 위치 추정 실험을 검증하기 위한 방법론을 제시하고 그 방법론을 바탕으로 Cricket Location System에서 무인 로봇의 위치 추정을 실험하였다.

2. 관련연구

Active Badge는 각 방에 적외선 수신기를 설치하고 적외선을 송신하는 badge 형태의 단말기를 이용하여 위치를 측정하는 시스템이다. 시스템의 구성이 비교적 간단하고 저렴한 비용으로 시스템을 구성할 수 있지만, 제한된 거리 내에서만 가능하고, 햇빛, 형광 등에 노출되면 성능이 저하된다는 단점이 있다.

Active Bats는 건물에 초음파 수신기를 그리드 형태로 설치하고 Bat라는 초음파 송신기를 이용한다. 서버에서 특정 Bat를 찾을 경우, 호출된 Bat가 초음파를 발생시켜 위치를 추적하는 중앙 집중형 시스템이다. 중앙 집중형 네트워크 기반의 위치인식 시스템이기 때문에, 시스템을 구축할 때 비용이 많이 들며 별도의 네트워크 관리가 필요하다는 단점이 있다.

RADAR와 3D-iD와 같은 시스템들은 RF 신호를 이용하여 AP에서 무선 LAN 기기들이 장치가 전송하는 신호의 세기와 신호 대 잡음비를 측정하고 이를 이용해 무선 LAN 장치들의 2차원적 위치를 계산하는 방법이다. 모든 장치가 무선 LAN을 지원하여야 하기 때문에 무인 이동 로봇처럼 제한적인 전원 장치를 가진 기기들에는 적용되기 힘들다는 단점이 있다.

고성능 3차원 카메라인 Digiclip으로 구축한 Easy Living 시스템은 컴퓨터 비전 기술을 이용하여 물체의 위치를 찾는 데 정확성을 향상시키지만, 비전 프레임워크를 분석하기 위해 많은 프로세싱 시간이 소요된다는 것이 단점이다.

MIT에서 개발된 Cricket은 천장에 부착된 초음파 송신기(Beacon)들로부터 수신된 신호를 이동체에 부착된 초음파 수신기(Listener)에서 계산하여 위치를 측정하는 시스템이다. 천장의 beacon들은 초음파 신호와 RF 신호를 동시에 발생한다. 초음파와 RF신호는 전파속도가 서로 다르기 때문에 listener에도착한 신호의 시간차를 이용하여 beacon과 listener 사이의 거리를 측정한다. 아래 [표 1]에 나타난 바와 같이 비교적 설치가

간단하고, 저렴한 단말기만 설계된 위치에 부착시키면 간단한 계 시스템을 구축할 수 있는 핸드셋 기반의 Cricket Location System을 구축하여, 위치 추정 실험을 검증하기 위해 본 논문에서 제안하는 방법론에 대해서 살펴보겠다.

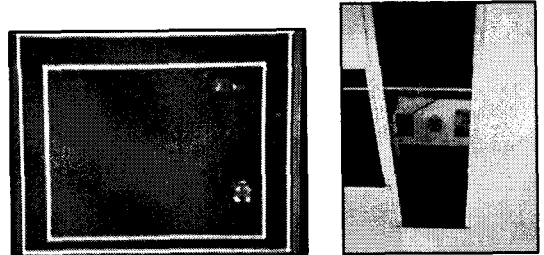
[표 1] 위치 추적 기술 비교

| 방식 | 정밀도 | 설치규모 | 가격 | 제한사항 |
|----------------|---------------|--------------------------|-------------------------|-----------------|
| Active Badge | Room size | 방마다 센서 1개 | 망 관리 비용, 센서와 단말기는 가격 저렴 | 해별, 형광등 |
| Active Bats | 95 % | 10m ² 당 센서 1개 | 망 관리 비용, 센서와 단말기 가격 저렴 | 최저에 경자역의 센서 설치 |
| MS-RADAR | 3~3.4m(50%) | 한 층에 3개 이상의 Base station | 802.11 망 설치, 무선 NIC 가격 | 무선 I.A.N NIC필요 |
| PinPoint 3D-iD | 1~3m | 건물 내 여러 개의 Base station | 망 설치 하드웨어 가격이 비쌌 | 802.11 망에 간섭 발생 |
| Easy Living | 가변 | 방에 3개의 카메라 필요 | 처리능력이 좋은 컴퓨터와 카메라 | 카메라 |
| Cricket | 4×4ft. (100%) | 16 sq ft당 1개의 비콘 설치 | 약 10달러 | 단말기에서 계산 |

[표 2] 무인 로봇 (ESTK 2440)

| 운영체제 | Linux Kernel 2.6.X |
|------|--|
| 사양 | S3C2440A MCU SDRAM 64MB AMD Flash 1MB(NOR Type), NAND Flash 32MB USB Device Connector UART 1 Port for Debug Ethernet 1 Port JTAG Port for Multi-ICE and RealView ICE |

마지막으로, WebCam의 동영상 캡처 능력은 130만 화소 (640×480)이고 프레임 속도는 최대 30 frames/second 이다. [그림3]의 왼쪽은 WebCam이 획득한 이미지이며, 오른쪽은 WebCam을 천장에 설치한 모습을 보여 준다.



[그림 3] WebCam이 촬영한 이미지와 천장에 설치된 모습

3. 시스템

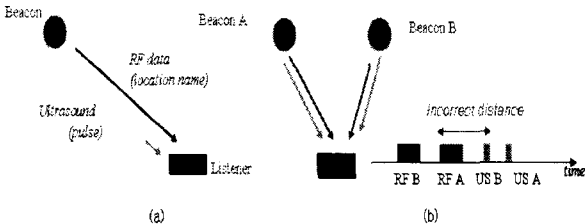
본 시스템은 이동 로봇의 위치 추정을 위한 Cricket, Cricket과 모터 제어부를 연결해 주는 임베디드 보드 그리고 실제 실험 장소를 동영상 촬영하여 실시간으로 스트리밍 해 주는 WebCam으로 구성된다.

첫 번째, Cricket은 [그림 1]과 같이 ATmega128L 마이크로 프로세서와 RF/초음파 송·수신부로 구성되어 있다.



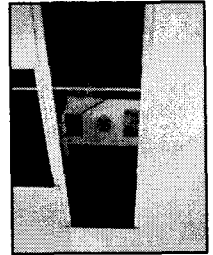
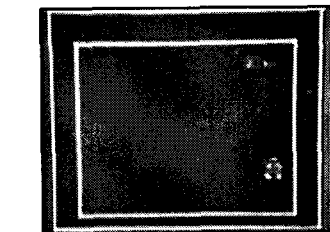
[그림 1] Cricket

이와 같은 cricket들로 구성된 Cricket Location System은 실내 환경에서 거리를 측정하기 위해서 [그림 2]와 같이 초음파의 속도 차이를 이용하여 모바일 노드의 위치를 제공한다. (RF : 433Mhz, 초음파 : 40Khz)



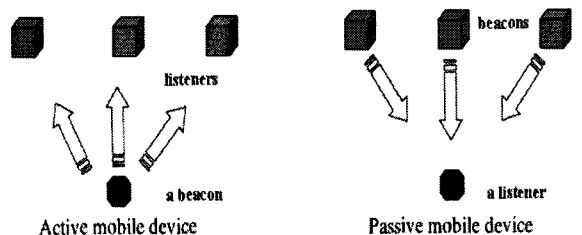
[그림 2] Beacon과 listener를 이용한 거리 계산

두 번째, 임베디드 보드는 이동 로봇의 두뇌와 같은 부분으로써 Cricket으로부터 수신 받은 거리 정보를 바탕으로 자신의 위치를 계산하며, 맵 정보를 이용하여 장애물을 피해 목적지까지



3.1 시스템 구조

위에서 언급한 세 가지 구성 요소를 바탕으로 본 시스템은 Active Badge, Active Bat, Ubisense[11]의 구조인 Active Mobile Infrastructure(AMI)와 Cricket의 구조인 Passive Mobile Infrastructure(PMI)를 제공한다.

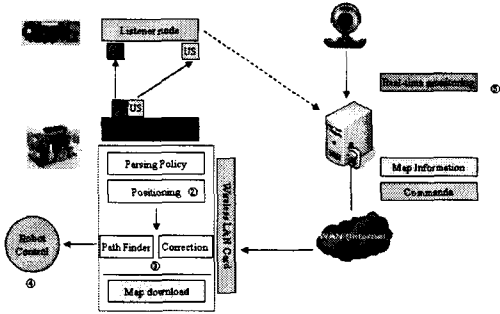


[그림 4] Two architectures of a location system

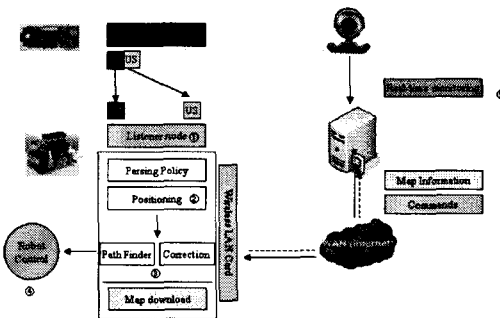
AMI에서는 beacon으로부터 여러 개의 listener로 동시에 거리 정보를 전송하기 때문에 추적이 용이하고, PMI에서는 listener가 한 번에 한 개의 beacon으로부터 거리 정보를 받기 때문에 추적에 비교적 적합하지 않으나, 확장성이나 개인의 프라이버시 보호 측면에서는 탁월하다[12].

[그림 5]는 AMI의 개념을 적용하여 제안된 시스템의 구조를

나타내며, [그림 6]은 PMI의 개념을 적용하여 제안된 시스템의 구조를 보여주고 있다.



[그림 5] AMI를 적용한 시스템



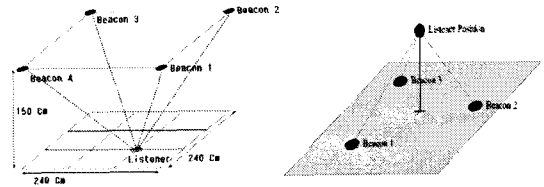
[그림 6] PMI를 적용한 시스템

기본적으로 두 시스템은 클라이언트 서버 구조로 이루어져 있다. 가장 큰 차이점은 무인 로봇의 위치를 어디서 계산하는가에 따라서 AMI와 PMI를 구분 지을 수 있기 때문에 [그림 6]의 PMI가 적용된 시스템을 예를 들어 설명한다. 클라이언트인 무인 로봇은 초음파 신호를 전송하는 송신부(Tx, Beacon)와 초음파신호를 감지하여 거리를 측정하는 수신부(Rx, Listener)로 구성된 센싱 모듈(①), listener로부터 얻어진 거리 값들을 이용해 위치를 추적하는 모듈(②), 현재 로봇의 위치와 서버로부터 수신된 맵 정보를 가지고 평면상에서 충돌을 회피하여 목적지까지 이동할 수 있도록 최적의 경로를 찾아주는 모듈(③), 그리고 계산된 경로를 따라 로봇에 실제 명령어를 전달하는 모듈(④)로 이루어져 있다. 서버(⑤)는 로봇의 이동을 감시하고 맵 정보와 목적지를 전달하며 응급사태 시 알람 기능과 실시간 제어 등을 하며, WebCam과 무인로봇으로부터 전송받은 위치 좌표를 한 영상위에 맵핑시켜서 실시간 위치 추정을 제공하도록 설계하였다.

3.2 로봇 위치 추정

Cricket의 기본적인 거리 측정 방법은 초음파를 이용한 거리 측정 방식이다. [그림 2]와 같이 초음파의 비행시간 측정에 의한 방식을 이용한다. RF의 속도는 초음파에 비해 매우 빠르기 때문에 RF 신호 지연을 0으로 간주한다. 왜냐하면 Cricket이 담당할 범위가 수 미터 이내이기 때문에 RF 신호의 지연은 무시할 수 있기 때문이다. beacon이 동시에 RF 신호와 초음파를 브로드캐스트하면 listener가 RF와 초음파의 속도 차이를 이용하여 초음파의 측정된 비행시간을 구하여 거리를 계산한다.

본 연구에서는 2차원의 평면상에서 이동하는 무인 로봇의 위치 측정을 필요로 하기 때문에 동일 직선상에 위치하지 않는 세 개의 beacon을 이용하여 2차원 좌표를 얻어온다. 최소 세 개 이상의 beacon이 있을 경우, listener는 측정된 beacon과의 거리들로부터 삼각측량법을 사용하여 위치를 계산할 수 있다.



[그림 7] 삼각측량을 통한 Listener의 위치 파악

listener와 beacon 간의 거리를 d_i 로 놓고 listener의 위치를 (X, Y, Z) , beacon의 위치를 (x_i, y_i, z_i) 로 놓으면 다음 식을 얻을 수 있다.

$$d_i^2 = (x_i - X)^2 + (y_i - Y)^2 + (z_i - Z)^2 \quad (1)$$

beacon들은 건물의 천장에 설치되기 때문에 z_i 의 값들은 같은 높이의 건물 내에서 모두 동일한 위치이다.

$$d_0^2 = (x_0 - X)^2 + (y_0 - Y)^2 + (z_0 - Z)^2 \quad (2)$$

$$d_1^2 = (x_1 - X)^2 + (y_1 - Y)^2 + (z_1 - Z)^2 \quad (3)$$

식(1)에 beacon (x_0, y_0, z_0) 을 대입한 식(2)과 식(1)에 beacon (x_1, y_1, z_1) 을 대입한 식(3)을 양변에 더할 경우 z 값들은 동일한 값($z_0 = z_1 =$ 천장 높이)이므로 소거되므로 (4)와 같은 식을 만들 수 있다.

$$2(x_1 - x_0)X + 2(y_1 - y_0)Y = x_1^2 - x_0^2 + y_1^2 - y_0^2 - d_1^2 + d_0^2 \quad (4)$$

따라서 3개의 beacon으로 두 개의 1차 방정식을 만들 수 있고 (5)와 같은 행렬식으로 연립방정식의 해를 구할 수 있다.

$$\vec{Ax} = \vec{b} \quad (5)$$

식(5)의 행렬 A 와 벡터 \vec{x}, \vec{b} 는 다음과 같다.

$$A = \begin{bmatrix} 2(x_1 - x_0) & 2(y_1 - y_0) \\ 2(x_2 - x_0) & 2(y_2 - y_0) \\ \dots & \dots \\ 2(x_i - x_0) & 2(y_i - y_0) \end{bmatrix}, \vec{x} = \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$\vec{b} = \begin{bmatrix} x_1^2 - x_0^2 + y_1^2 - y_0^2 - d_1^2 + d_0^2 \\ x_2^2 - x_0^2 + y_2^2 - y_0^2 - d_2^2 + d_0^2 \\ \dots \\ x_i^2 - x_0^2 + y_i^2 - y_0^2 - d_i^2 + d_0^2 \end{bmatrix}, i \geq 2, \det(A) \neq 0 \quad (6)$$

식(6)에서 $i \geq 2$ 이고 행렬 A 의 determinant가 0이 아닐 경우, 연립방정식의 해인 벡터 \vec{x} 는 다음 식(7)로부터 구할 수 있다.

$$\vec{x} = A^{-1}\vec{b} \quad (7)$$

벡터 \vec{x} 의 성분인 X, Y 는 현재 무인 로봇이 있는 좌표가 되므로 X, Y 값에 의해 무인 로봇의 현재 위치를 2차원 좌표평면에 나타낼 수 있다.

3.3 이동 경로 탐색

서버는 실제 건물의 2차원 평면도를 무인 로봇이 인식할 수 있는 그리드 형태의 2차원 배열로 맵핑하여 유지한다. 로봇은 서버로부터 목적지와 맵 정보를 수신한 후, 3.2절에서 언급한 방법대로 로봇의 현재 위치를 추정하여 출발지로 결정한다. 일단 출발지와 목적지가 설정이 된다면 수신받은 맵 정보를 바탕으로 A^* 알고리즘을 이용하여 로봇의 최적 이동 경로를 계산한다[13].

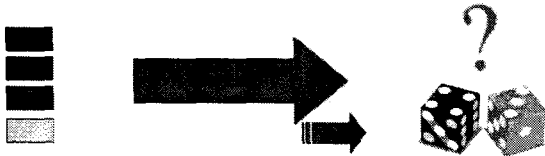
3.4 Parsing policies

본 시스템은 무인 로봇의 위치를 추정하기 위해서 Cricket의 아이디어인 SP(Space ID), beacon과 listener 사이의 거리인 DB(Distance to Beacon), 그리고 Cricket의 시스템 시간인 TS(System Time) 세 가지 파라미터를 이용한다[14].

먼저, Parsing policy란 beacon들로부터 온 SP, DB 그리고 TS을 담은 정보를 받은 listener가 무인 이동 로봇의 위치 좌표를 계산하기 위해 네 개의 beacon 중에서 서로 다른 세 개의 beacon들의 정보를 취합하여 위치 추정 오차를 줄이기 위해 설계된 것이다.

Active Mobile Infrastructure(AMI)와 Passive Mobile Infrastructure(PMI)에서의 parsing policy에 대해서 알아보자.

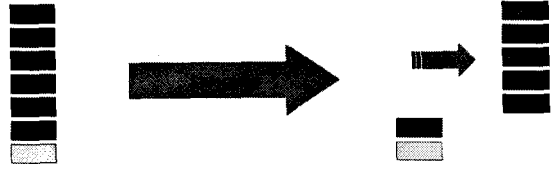
AMI에서의 parsing policy에 대해서 살펴보면, 무인 로봇에 부착된 beacon이 listener들에게 자신의 ID와 TS값을 브로드캐스트하면, 각 listener들은 beacon의 ID, TS 그리고 계산된 거리(DB)를 서버로 전달하여 무인 로봇의 위치를 계산하게 된다. 이 때 beacon에 대한 정보를 받은 listener들의 수가 많을수록 무인 로봇 위치를 정확하게 계산할 수 있으나, 실시간 위치 추정에 걸리는 시간은 증가한다. [그림 8]에서 사용되는 블록은 색깔에 따라 다른 아이디를 가진 listener로부터 온 데이터를 의미한다.



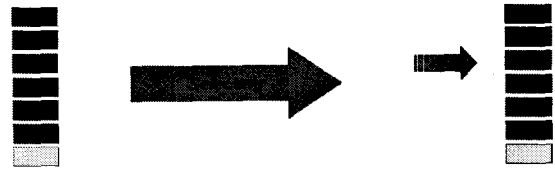
[그림 8] Parsing Policies for AMI

반면에, PMI에서는 listener가 동시에 다수의 beacon의 정보를 수신한다는 것은 불가능하다. 왜냐하면 beacon들이 서로 간의 충돌을 피해서 listener로 정보를 송신하기 때문이다. 따라서 여기서 parsing policy는 beacon들로부터 수신받은 일련의 연속된 정보들을 과싱하고, 각각 다른 SP를 가진 세 beacon들로부터 온 데이터를 선택해야 위치 계산이 가능하다. 하지만 서로 다른 세 beacon들의 정보를 이용하여 이동하는 무인 로봇의 위치 추정할 때 오차가 커지는 현상이 자주 발생된다. 왜냐하면 기본적으로 Cricket은 PMI에서 beacon과 listener를 동기화하지

않기 때문이다. 이처럼 동기화되지 않은 PMI에서 parsing policy는 서로 다른 beacon들의 TS 정보를 이용하여 무인 로봇의 위치 추정 오차를 줄이기 위해 제안되었다. 아래 [그림 9]와 [그림 10]에서 사용되는 블록은 색깔에 따라 각각 다른 아이디를 가진 beacon으로부터 온 데이터를 의미한다.



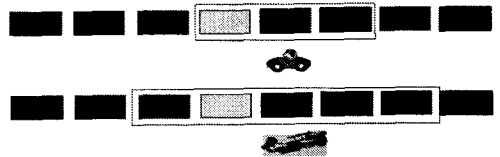
[그림 9] Parsing Policy for AMI (high mobility)



[그림 10] Parsing Policy for AMI (high accuracy)

이 parsing policy는 위치를 계산하는 시점에서 선택된 데이터들이 유효한지 즉, 처음 선택된 beacon의 TS와 세 번째 선택된 beacon의 TS의 차이가 유효한지를 검사한다. 따라서 TS의 유효한 범위를 결정할 필요가 있다. 위의 [그림 9]와 같이 TS의 유효 범위를 크게 설정할 경우에는 무인 로봇의 현재 위치를 계산하고 추적하는데 low accuracy, high mobility를 제공할 것이다. 하지만 위의 [그림 10]과 같이 TS의 유효한 범위를 작게 설정한다면 최악의 경우 너무 많은 데이터를 버리기 때문에 high accuracy, low mobility를 제공하므로 위치 추적에 적합하지 않다.

이런 문제점을 해결하기 위해서 선택된 노드의 정보를 항상 최신의 정보로 업데이트하고 처음 선택된 노드와 마지막에 선택된 노드가 유효한지 체크하는데, 무인 로봇의 이동 속도를 고려하여 adaptive sliding window 메커니즘을 사용하여 mobility와 accuracy의 조질하여 위치 추적을 최적화할 수 있다. 예를 들면, 아래 [그림 11]과 같이 고속인 경우에는 window 크기를 크게 하고 저속인 경우에는 window 크기를 작게 하여 무인 로봇의 위치 추정의 정확성과 효율성을 향상시킬 수 있다.



[그림 11] An Example of Adaptive sliding window

3.5 Real-time view for tracking

무인 로봇의 위치 추정에 있어서 실제 이동 거리와 측정된 이동 거리를 비교하는 Real-time view를 제공할 목적으로 천장에 WebCam을 설치하였다. 실험 장소를 실시간으로 촬영하

여 서버로 스트리밍 전송을 하고 획득한 영상 위에 로봇으로부터 받은 위치 좌표를 맵핑시킴으로써 어떤 추가적인 센서를 사용하지 않고 무인 로봇의 위치 추정 실험을 검증하는데 의미가 있다.

무인 로봇으로부터 받은 좌표 $Robot(x, y)$ 을 WebCam에서 획득한 영상 위에 맵핑시키는 위해서는 단위 픽셀 당 실제거리를 구해야 한다. 이 때 실험장소의 크기(real_size)는 240×240 cm이고, 획득한 이미지 크기(image_size)는 640×480 pixel이다. 단위 픽셀당 실제 거리를 L 이라고 하면

$$L = \frac{real\ size}{image\ size} \quad (8)$$

즉, $L = \begin{bmatrix} 240 \\ 640 \\ 240 \\ 480 \end{bmatrix}$ 인 2×1 행렬로 표현할 수 있다.

$Robot(x, y)$ 좌표를 획득한 WebCam 영상 위에 맵핑시킨 좌표를 $WebCam(x, y)$ 라고 한다면,

$$WebCam(x, y) = \frac{Robot(x, y)}{L} \quad (9)$$

라고 정의할 수 있다. 위 식(8)과 식(9)을 통해서 무인 로봇의 좌표들을 맵핑시켜 실시간으로 위치를 추정할 수 있다.

4. 실험

4.1 실험 환경

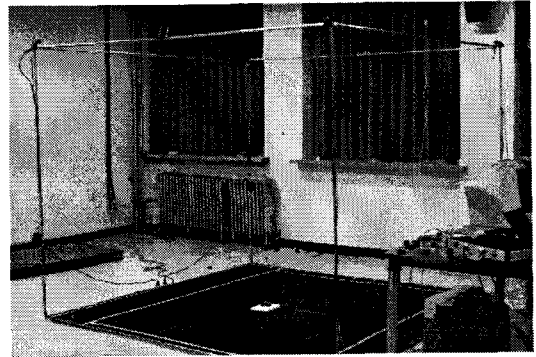
본 시스템에서 로봇은 자신의 위치를 스스로 계산하여 서버로 전송하고 본 논문에서 개발된 Robot Monitoring Center는 이 정보를 이용해 로봇의 위치를 추적할 수 GUI를 제공한다. 개발 환경은 다음 [표 3]과 같다.

[표 3] 개발 환경

| | | 개발 환경 |
|-------|------------|---|
| Notes | | Hardware - Atmega128L Software - Cricket 2.3.2 |
| Robot | | ESTB2440(ARM S3C2440) Mini-CAR MR01 |
| OS | Linux | Qplus + Esto Fedora core 6 |
| | Windows XP | Robot Monitoring Program |

아래 [그림 12]를 살펴보면 AMI(Active Mobile Infrastructure)인 경우에는 Depth가 200cm 되는 곳에 4개의 listener를 설치한 후, RS-232C 케이블을 서버에 연결을 한다. beacon은 무인 로봇에 장착된다. 하지만, PMI(Passive Mobile Infrastructure)인 경우에는 Depth가 200cm 되는 곳에 4개의 beacon을 설치한 후, listener를 무인로봇에 장착한다.

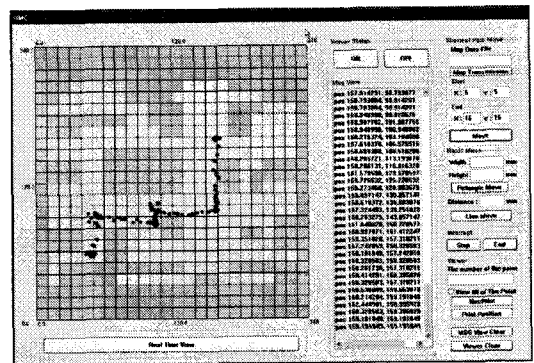
아래 [그림 12]에 보이는 실험 환경은 $240\text{cm}(\text{Width}) \times 200\text{cm}(\text{Depth}) \times 240\text{cm}(\text{Height})$ 이다.



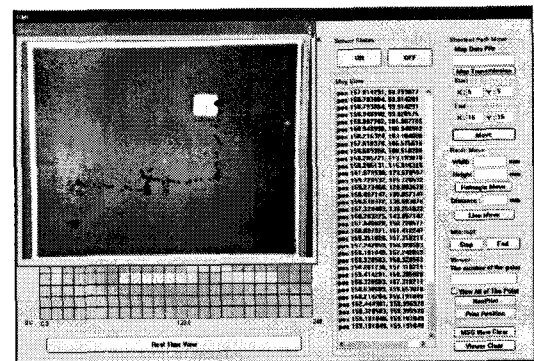
[그림 12] Experimental setup for AMI and PMI

4.2 실험 결과

먼저, AMI의 실험 과정과 결과에 대해서 살펴보자.



[그림 13] Map view for AMI



[그림 14] Real-time view for AMI

무인로봇이 처음 부팅되면 [그림 13]에 나타난 서버의 맵 정보를 무인로봇에 전송한 후, 무인로봇이 이동할 때 [그림 13]와 같은 건물도 평면도 상에 전송된 무인로봇의 현재 위치가 작은 점으로 나타난다. 오른쪽 메시지 창(Msg View)은 서버에서 계산된 무인로봇의 좌표를 출력하는 창이다. 무인로봇의 위치는 이동 중에도 경로를 따라 실시간으로 파악이 가능하며, 궤적

경로 계산과 실제 이동 경로가 차이가 있을 시 무인로봇의 현재 좌표를 기준으로 다음 목적지까지의 위치를 보정하면서 최초로 수신 받은 목적지까지 이동한다. [그림 14]과 같이 실시간으로 이동 로봇 좌표를 Map view와 Real-time view에 좌표를 동시에 맵핑시킴으로써 크리켓 센서 정보들의 거리 정보를 바탕으로 계산된 이동 경로와 무인로봇이 실제로 이동한 경로를 확인할 수 있다.

다음으로, PMI 실험 과정과 결과를 살펴보면, 무인이동 로봇이 처음 부팅되면 [그림 15]에 나타난 서버의 맵 정보를 이동 로봇에 전송한 후, 로봇이 이동할 때 [그림 15]와 같은 건물의 평면도 상에 전송된 로봇의 현재의 위치가 작은 점으로 나타난다. 왼쪽 메시지 창(Msg View)은 로봇에서 계산되어 전송된 좌표들을 출력하는 창이다. 최초로 수신 받은 목적지까지 이동하기 위해서는 AMI와 동일한 과정으로 최적 경로 계산과 실제 이동 경로가 차이가 있을 시 로봇의 현재 좌표를 기준으로 다음 목적지까지 위치를 보정한다.

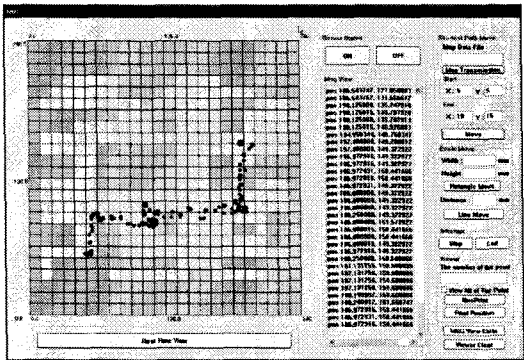
스텝을 구현하였다. AMI가 PMI보다 무인로봇을 더 정확히 추적할 수 있으나, 추적할 무인로봇이 많은 경우에는 중앙 집중형 시스템이기 때문에 확장성이 힘들고 서버에 과부하가 많이 걸리는 단점이 있다. 반면에 PMI는 AMI보다 이동로봇의 좌표를 스스로 계산하기 때문에 자신의 위치를 노출시키지 않아 개인 프라이버시 보호 측면에서는 우수하다는 것을 확인할 수 있다. 향후 연구 과제로는 실시간 잡음 운동방정식을 가진 시간에 따른 방향을 추적하는 효율적인 칼만 필터를 적용할 예정이다.

Acknowledgements

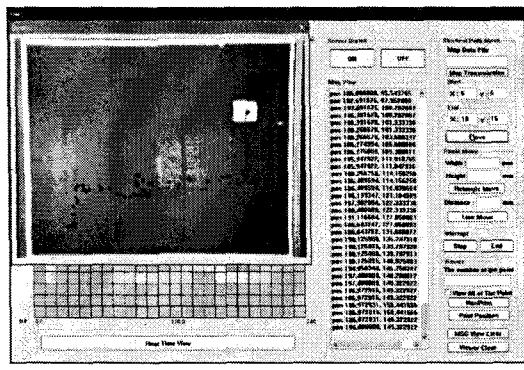
본 연구는 산업자원의 지역혁신 인력양성사업(KOTEF)의 지원으로 이루어졌습니다.

참고문헌

- [1] Per Enge, Todd Walter, Sam Pullen, Changdon Kee, Yi-Chung Chao, and Yeou-Jyh Tsai, "Wide Area Augmentation of the Global Positioning System," In Proceedings of the IEEE, number 8 in 84, August 1996
- [2] Baul, P., Padmanabhan, V. Radar, "An In-Building RF-based User Location and Tracking System," In Proc. IEEE INFOCOM, Mar. 2000
- [3] R. Want et al, "The Active Badge Location System," ACM Trans, Information Systems, pp. 91-102. Jan. 1992
- [4] Harter, A., Hopper, A., Steggles, P., Ward, A., Webster, P., "The Anatomy of a Context-Aware Application," In Proc. 5th ACM MOBICOM Conf. Aug. 1999
- [5] Nissanka B. Priyantha, Anit Chakraborty, Hari Balakrishnan, "The Cricket Location-Support system," Proc. 6th ACM MOBICOM, Boston, MA, Aug. 2000
- [6] Joon-Yong Lee and Robert A. Scholtz, "Ranging in a Dense Multipath Environment Using an UWB Radio Link," IEEE JSAC, Vol.20, No.9, Dec. 2002
- [7] Microsoft Research, EasyLiving Website, <http://www.research.microsoft.com/easyliving>
- [8] P. Bahl and V. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," Proc. IEEE Infocom 2000, IEEE CS Press, Los Alamitos, Calif. 2000, pp. 775-784
- [9] PinPoint to introduce its new 3D-ID locator system for tracking both people and equipment, BBI NEWSLETTER 1998; VOL 21; NUMBER 10 221-224
- [10] Priyantha, N.B. Balakrishnan, H. Demaine, E.D. Teller, S., "Mobile-assisted localization in wireless sensor networks," IEEE Infocom, March 2005
- [11] Cadman, J. "Deploying Commercial Location-Aware Systems," In Proc. Fifth International Conference on Ubiquitous Computing, October 2003
- [12] Adam Smith, Hari Balakrishnan, Michel Goraczko, Nissanka Priyantha, "Tracking Moving Devices with the Cricket Location System," Proc. 2nd USENIX/ACM MOBISYS Conf., Boston, MA, June 2004
- [13] Jaebong Yoo, Hyeonjun Shin, JungTaek Seo, Sangyun Kim, Chanyoung Park, "Auto-driving System using RF Navigator in Indoor Environment", KCC2007, June 2007
- [14] The Cricket Indoor Location System(Cricket v2 Manual), <http://cricket.csail.mit.edu/v2man.html>



[그림 15] Map view for PMI



[그림 16] Real-time view for PMI

5. 결론

본 시스템은 Cricket Indoor Location System을 바탕으로 AMI(Active Mobile Infrastructure)와 PMI(Passive Mobile Infrastructure)를 구축한 후, parsing policy를 적용하여 오차가 심한 거리 정보를 사진에 제거하여 위치 추정의 오차를 줄였다. 또한 실험에서 얻은 모바일 노드의 위치 좌표들을 카메라로 획득한 실제 이미지에 맵핑시킴으로써 위치 추정 실험을 검증하기 위한 방법론을 제시하여 실시간 무인로봇 위치 추적 시