

실내 환경에서 효율적인 위치 추적을 위한 알고리즘에 관한 연구

전 현 식* · 우 성 현* · 이 호 응* · 류 인 선* · 윤 성 근* · 박 현 주**

A Study on Algorithm for Efficient Location Tracking in Indoor Environment

Jeon Hyeon-Sig* · Woo Sung-Hyun* · Lee Ho-Eung*
Ryu In-Seon* · Yoon Sung-Kun* · Park Hyun-Ju**

Abstract

According to developing Wireless Communication, not only a location based service at the outside but also a location based service at the inside was more increased socially. This paper proposes the efficient algorithm to locate a transfer object in frequent change of indoor environment using Indoor location tracking system we develop ourself. Proposing algorithm in this paper can locate a transfer object using the Fingerprint, one of the Location Tracking techniques which are used in general to minimize error data between Location Tracking System and Fingerprint, using this way that corrects location data as KF apply to result data can improve accuracy of a transfer object. At last we are going to compare and analyze existing typical triangulation with proposed Indoor location tracking system to demonstrate algorithm efficiency for proposed Indoor location tracking system.

Keywords : Positioning, Ranging, Fingerprint, Indoor Environment

논문접수일 : 2006년 5월 19일

논문게재확정일 : 2006년 7월 26일

* 본 논문은 정보통신부 정보통신연구진흥원에서 지원하고 있는 정보통신 기초기술 연구지원 사업과 2단계 BK21사업의 지원에 의하여 연구되었음.

* 한밭대학교 정보통신전문대학원, 대전광역시 유성구 덕명동 산 16-1 한밭대학교 4동 413호 데이터베이스연구실, Tel : 042-821-1220, Fax : 042-821-1220, E-mail : hsjeon@hanbat.ac.kr

** 한밭대학교 정보통신컴퓨터공학부, 부교수

1. 서론

최근 정보통신 기술의 급격한 발전으로 인해 네트워크 인프라가 광범위하게 보급되고 있고, 첨단 디지털 장비가 일상생활에 보편화 되어감에 따라 이를 바탕으로 한 유비쿼터스(Ubiquitous) 시대가 도래하고 있다. 이러한 유비쿼터스 시대에 다양한 고객의 요구를 만족하기 위해서는 객체의 인식과 위치 추적에 관련된 연구가 필요하다.

기존 위치 추적 서비스는 GPS를 활용한 실외 환경 중심으로 개발되어졌으나, 현재에는 다양한 기술의 보급과 사용자의 요구로 인하여 실내 환경에서 위치 추적에 대한 관심과 필요성이 증가되고 있다. 실내 환경에서의 위치 추적 시스템은 다양한 분야에 활용될 수 있는데, 대표적인 예로 물류나 유통 분야에서의 사용자의 구매 프로세스를 추적하여 분석함으로써 다양한 서비스 제공 및 부가가치를 창출할 수 있다. 유비쿼터스가 제시하는 많은 사용자 위주의 서비스들은 사용자의 주 생활 공간인 실내에서 제공된다. 그러나 GPS와 같은 기존의 위치 추적 서비스는 실내 환경에 부적합한 특성을 보이므로 실내 환경에 적합한 위치 추적 시스템에 대한 연구들이 진행되고 있다[1]. 실내 위치 추적을 구현한 대표적인 시스템은 적외선을 활용하여 위치 추적 시스템을 구현한 AT&T社의 Active Badge와 초음파와 RF를 활용한 MIT의 Cricket 등이 있다.

IEEE 802.15.3 표준그룹에서 제안된 UWB(Ultra WideBand)는 실내용 거리 측정에 적합하다. 이를 바탕으로 기존에 개발된 Zigbee(IEEE 802.15.4)의 PHY 부분을 개선함으로써 위치 인식을 가능하게 하는 IEEE 802.15.4a의 표준그룹이 생성되었다. IEEE 802.15.4a 표준그룹은 임펄스, 첵(Chirp), 카오스 신호를 활용한 위치 인식 시스템을 제안하고 있다[2-5]. 이러한 실내 위치 인식 시스템의 개발과 함께 실내 환경에 최적화된

위치 검색 기법들이 점점 중요해지고 있다.

현재 위치 시스템에서 가장 널리 사용하는 위치 척도로 TOA(Time of Arrival)가 있다. 그러나 실제 실내 위치 시스템은 전파의 반사와 회절, 분산 등의 NLOS(Non Line of Sight) 환경을 감내해야 하기 때문에 기존 TOA를 보완하는 연구가 진행되고 있다. 이러한 NLOS 에러를 완화하기 위한 대표적인 방법으로 KF(Kalman Filtering)을 이용하는 방법, Scattering Model을 이용한 접근 방법 등이 있다.

본 논문에서는 실내 환경에 효율적인 실내 위치 추적 알고리즘을 제안한다. TOA 기반의 위치 추적 시스템은 실내 환경에서의 NLOS 특성으로 인해 위치 정확도가 부정확하다. 따라서 본 논문에서는 NLOS 현상을 반영하기 위해 실내 환경에서 위치 인식 기술로 주로 사용되는 Fingerprint 기법을 위치 추정 알고리즘으로 활용하여 위치 정확도를 보정한다. 그리고 전파의 도달거리가 환경에 의해 변화되어 길어지는 위치의 JUMP 현상을 완화하기 위해 KF를 적용한 알고리즘을 제안한다[6, 7].

본 논문의 구성은 다음과 같다. 2장에서는 대표적인 위치 척도 기법인 삼각측량 기법과 본 연구에서 사용되는 Fingerprint와 KF에 대한 기존의 연구를 기술한다. 3장에서는 본 논문에서 사용하는 실내 위치 추적 시스템에 대한 설명과 제안한 알고리즘에 대해 기술한다. 4장에서는 제안한 알고리즘의 성능 평가를 실시하고 5장에서 결론을 맺고 향후 연구 과제를 제시한다.

2. 관련연구

2.1 삼각측량기법(Triangulation)

대표적인 위치 척도 기법인 삼각측량 기법은 신호원(Base Station, 본 논문에서는 Server라 명칭)과 수신기(Mobile Station, 본 논문에서는

MU(Mobile Unit)이라 명칭) 사이의 전파 도달 시간(TOA)을 측정하여 MU와 Server 간의 거리를 알게 되는 측위 방식이다. 일반적으로 분산된 많은 MU들은 Server에서 전송된 신호를 받는데 사용되고, TOA(Time of Arrival)나 TDOA(Time Difference of Arrival)로 정확한 위치 측정을 한다. TOA나 TDOA 데이터를 갖는 위치 결정 알고리즘은 위치 서비스 센터에서 MU의 위치를 추적하는데 사용된다. TOA는 MU로부터 최소 3개의 Server까지의 절대적인 시간을 계산하여 위치를 추적하는 알고리즘으로 동기식, 또는 비동기식으로 계산될 수 있다. 동기식 방법은 MU와 Server가 서로 동기화되어 있는 시스템이다. 이 방법에서 MU는 Server로 절대적인 현재 시간을 기록하여 신호를 보낸다. 이 신호는 MU의 주변에 있는 각각의 Server로 도달하게 된다. Server와 MU는 현재 시간을 동기적으로 맞추고 있기 때문에 Server는 받은 신호에 대한 절대적인 도달 시간을 계산할 수 있으며, 이 시간에 대한 거리를 이용하여 MU에 대한 위치를 계산하게 된다. 비동기식 방식에서는 모든 Server와 MU 간에 동기를 맞출 필요가 없다. Server는 현재 시간을 기록하고 MU에 신호를 보낸다.

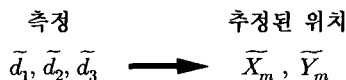
최소 3개의 Server로부터 수신된 TOA를 통해 Server와 MU의 거리 계산 공식

$$d_1 = \sqrt{(x_1 - X_m)^2 + (y_1 - Y_m)^2}$$

$$d_2 = \sqrt{(x_2 - X_m)^2 + (y_2 - Y_m)^2}$$

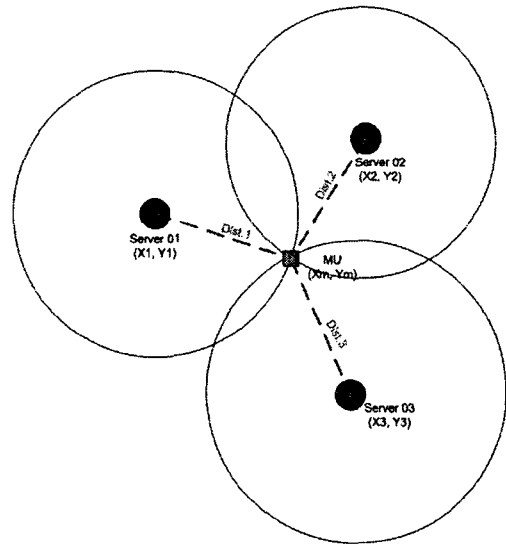
$$d_3 = \sqrt{(x_3 - X_m)^2 + (y_3 - Y_m)^2}$$

측정된 거리 정보 활용하여 MU의 위치 좌표 생성



<그림 1> TOA 신호를 이용한 위치 추정

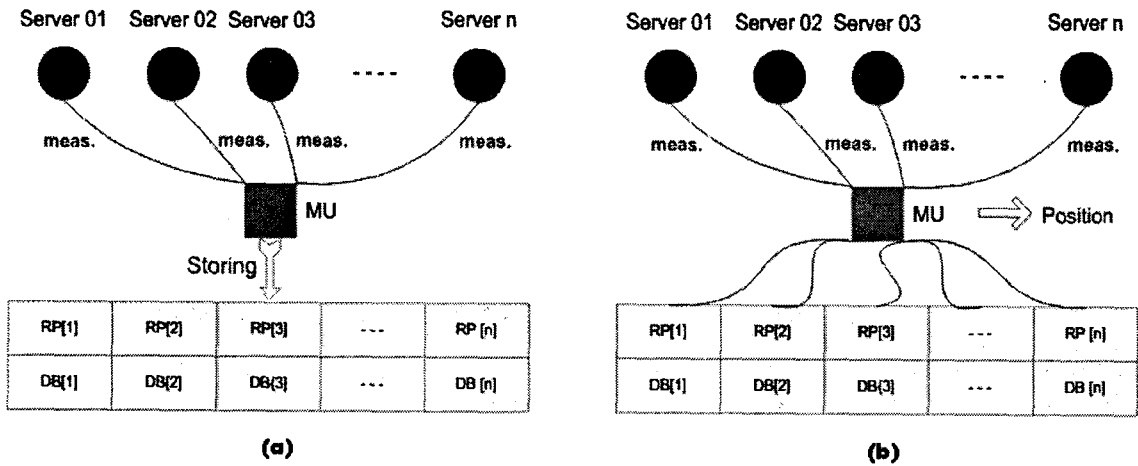
MU는 신호를 받는 즉시 응답 신호를 보내어 Server에 도달하도록 한다. Server는 MU에서 보내온 신호의 도착 시간과 기존에 저장해 놓은 신호의 시간차를 이용하여 <그림 1>과 같이 거리를 계산하게 된다[8]. 이를 통해 생성된 거리를 활용하여 3개의 Server로부터 MU에 대한 거리를 반지름으로 갖는 3개의 원을 그리고 <그림 2>와 같이 3개의 원이 공통으로 만나는 교차 영역을 MU의 위치로 선택하는 방법이다. 이러한 방법은 MU의 위치를 결과로 나타내기 위해 원이 교차하는 전체 영역을 MU의 위치로 판단하게 된다.



<그림 2> 삼각측량 기법

2.2 핑거프린트(Fingerprint)

본 절에서는 제안하는 알고리즘에서 활용되는 Fingerprint를 소개한다. Fingerprint는 휴리스틱 정보를 데이터베이스에 저장한 후 MU에서 입력된 위치 데이터 값과 비교하여 현재 위치를 리턴하는 기법이다. 이해를 돕기 위해 WLAN(Wireless Local Area Network) 환경에서 구축된 Fingerprint 시스템을 예를 들어 설명한다.



〈그림 3〉 Fingerprint의 Training 단계(a)와 Positioning 단계(b)

WLAN 환경에서는 MU의 위치를 측정하기 위해 각 Server로부터의 위치 정보(SS : Signal Strength)로 MU의 위치를 측정한다[9].

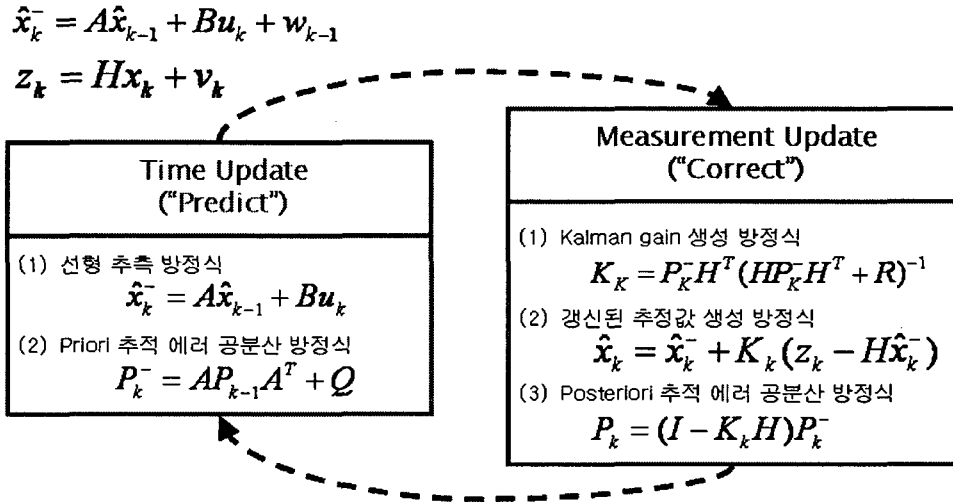
〈그림 3〉에서 보는 것과 같이 Fingerprint는 데이터베이스를 생성하는 “Training” 단계와 MU가 요구하는 위치 데이터 값을 측정하는 “Positioning” 단계로 나누어진다.

“Training” 단계는 객체의 위치 측정을 요구하는 전체 영역을 대상으로, RP를 일정한 간격으로 배치한다. RP를 전체 영역에 고르게 배치한 후 RP에 MU를 위치시킨 후 모든 Server로부터의 SS를 측정한다. 그리고 각 Server로부터의 위치 데이터를 Database에 기록한다. 이 프로세스는 초기에 선택한 모든 RP를 방문할 때까지 반복 수행하여 유지한다. “Positioning” 단계는 MU의 위치를 측정하기 위해 위치를 요구하는 장소에서 SS를 측정한다. 그 후 “Training” 단계에서 유지된 Fingerprint 데이터베이스와 비교한 후 요구한 MU의 위치와 가장 적합한 위치 데이터의 정보를 MU로 리턴한다[10].

2.3 칼만필터(Kalman Filter)

본 장에서는 제안하는 위치 추적 시스템에서

위치 보정을 위해 사용되는 Kalman Filter를 소개한다. Kalman Filter는 프로세스의 상태를 추적하기 위해 오차를 최소화 하는 방법으로 효율적인 계산 수단을 제공하는 수학 방정식의 집합이다[11]. 이 필터는 위성정보와 같이 코드화된 통계적인 정보 분석을 수행하는 고급 컴퓨터용 필터로서 장애물에 의해 약간의 수신 장애가 발생할 경우와 같은 약한 신호에도 좋은 결과를 검출해 낸다. 또한 신호와 잡음의 분리도가 우수하여 순간적인 위치의 점프(JUMP) 현상을 방지할 수도 있으며, 정확한 위치, 속도 및 시간을 안정적으로 검출하는데 크게 기여한다. 또한 과거와 현재뿐만 아니라 미래 상태까지도 지원을 하며, 모델시스템의 특성이 자세히 알려지지 않은 경우에도 추적이 가능하다. 이는 피드백 제어(Feedback Control) 형식을 사용하여 추적한다. 필터는 임의 시간에 프로세스 상태를 추적하고 노이즈 측정 형식으로 피드백을 얻는다[12]. Kalman Filter 알고리즘에 사용되는 방정식은 예측을 위한 시간 갱신 방정식(Time Update Equation)과 측정 갱신 방정식(Measurement Update Equations)으로 구성된다. Kalman Filter의 평가 알고리즘은 수치 문제를 풀기 위한 예측



〈그림 4〉 Kalman Filter Cycle Process

자-교정자(predictor-corrector) 알고리즘으로, 프로세스는 <그림 4>와 같이 진행된다[11, 13].

<그림 4>에서 살펴 본 바와 같이 Kalman Filter Cycle 프로세스의 'Time Update' 과정은 시간에 앞서 미리 현재 상태를 예측하는 단계로, 순방향으로 현재 상태 추정결과를 전달한다. 'Measurement Update' 과정은 해당 시간에 실제 측정에 의해 전달된 추정 상태 값들을 조정한다. <그림 4>의 'Time Update'의 단계에서 'x'는 최적화를 하고자 하는 상태 변수를 의미하며, 계수 'A'는 한 단계에서의 상태변수와 다음 단계의 상태변수를 연결하는 변환 계수를 표현한다. 'B'와 'u'는 하나의 값으로 인식할 수 있으며 이들은 시스템에 무관한 추가 입력 값이다. 그리고 'Q'는 k단계에서 상태변수 'x'의 참값과의 차이 값, 즉 시스템 오차(system noise)이다. 선형 추측 방정식을 통해 Priori 추정 값을 생성한 후, 'Measurement Update' 과정을 수행한다.

'Measurement Update'의 단계에서 'z'는 관측값이고, 이는 상태변수 'x'와 변환계수 'H'에 의해 표현되며, 'R'는 관측값 'z'와 관측 참값과의 오차(measurement noise)이다. 즉, 시간 갱

신 방정식에서의 오차 'Q'와 측정 갱신 방정식에서의 'R'은 각각의 참값에 대해 정규분포하며 그 평균은 0이다.

<그림 4>에서 나와 있듯이 측정 갱신 방정식에서 첫 번째 단계는 Kalman gain 'K_k'를 계산한다. 다음 'z_k'를 얻기 위해 실제 프로세스를 측정 후 측정을 통합하고, 마지막 과정은 이를 활용하여 posteriori 에러 공분산을 얻기 위한 단계이다. 각 시간과 측정갱신 쌍들 이후의 프로세스는 새로운 priori 추정을 투입 또는 예측을 이용한 이전단계 posteriori 추정들로 반복된다. 이 반복 특성은 Kalman Filter의 매력적인 특성 중의 하나로 각 추정에 대한 모든 데이터를 직접 연산하도록 설계된 Wiener 필터보다 구현을 훨씬 쉽게 한다. 하지만 앞의 Kalman Filter의 원리에서 살펴보면 알 수 있듯이 보정된 정보가 안정화 되는 시간까지 불확실한 위치 데이터를 감수해야 한다는 문제점이 있다.

2.4 LOS/NLOS 특성

무선 통신환경에서 기지국(Server)과 MU는

서로 간에 방해물이 전혀 없는 LOS 환경과 방해물이 존재하는 NLOS 환경으로 나눌 수 있다. 도심과 같은 고밀도 지역이나 실내에서는 NLOS 환경으로 이동국과 기지국 간의 전파는 직선거리를 가지지 못하고, 반사나 굴절로 인하여 다중 경로(multipath)가 발생하여 전파하는 거리가 직선거리보다 더욱 멀어지게 된다. 정확한 SOI의 위치를 찾기 위한 중요 과제 중의 하나는 장애물에 의하여 발생하는 NLOS 문제의 해결이다. 전형적인 위치 알고리즘은 NLOS 환경에서 불완전하게 수행되므로 NLOS를 식별할 수 있는 기법 개발과 적절한 에러 완화 기법 채택이 필요하다. 일반적으로 전형적인 위치 알고리즘에서 사용하는 데이터에 해당하는 범위 측정은 식 (2-1)과 같이 모델화 할 수 있다.

$$r_m(t_i) = l_m(t_i) + n_m(t_i) \quad (2-1)$$

$$r_m(t_i) = l_m(t_i) + n_m(t_i) + NLOS_m(t_i), \quad m = 1, \dots, M \quad (2-2)$$

식 (2-1)은 LOS를 모델화하고, 식 (2-2)는 NLOS를 모델화하였다. 여기서 $r_m(t_i)$ 는 거리 측정, $l_m(t_i)$ 는 실제 거리, $n_m(t_i)$ 는 측정 노이즈, $NLOS_m(t_i)$ 는 시간(t_i)에서의 NLOS 에러이다. 하지만 NLOS 에러는 시간에 따라 변하는 위치 특성이므로 특징을 기술하기 어려운 문제점이 있다.

정확한 NLOS 식별이 가능하다면, NLOS 에러를 완화하기 위한 방법을 사용하여 더 나은 위치 추적이 가능하다. 만약 셋 이상의 Anchor에서 측정이 가능하고 적어도 세 개의 LOS Anchor를 찾게 되면, NLOS Anchor에서 영역 측정 데이터는 무시되고, 위치 추정 알고리즘에서 SOI의 위치 측정에 사용될 수 있다. 다음으로 측정된 Anchor 중 NLOS 데이터가 있다면 RSA(Range Scaling

Algorithm)와 같은 에러를 완화하는 방법을 사용하여 SOI의 위치를 측정할 수 있다. RSA에서 모든 Anchor에 대한 실제 영역 추적은 제한된 최소화 문제로 형식화할 수 있다. 영역추정의 LOS/NLOS 상태를 알 때 적용되는 NLOS 에러를 완화하기 위한 다른 알고리즘은 RWA(Recursive Weighting Algorithm) 등이 있다[14].

3. 실내 위치 추적 알고리즘

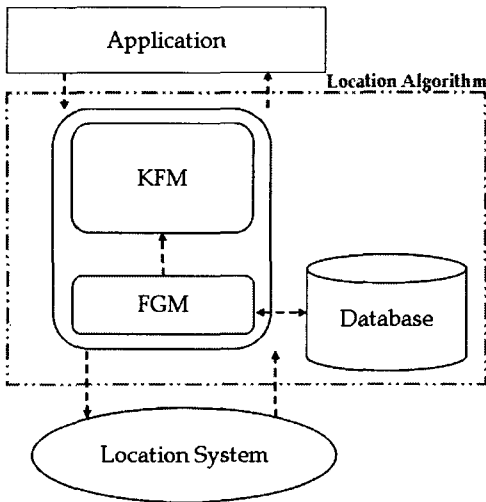
본 장에서는 제안한 실내 위치 추적 알고리즘에 대해 살펴본다.

3.1 개요

본 논문에서 제안하는 Location Algorithm의 구조는 <그림 5>와 같다. Location System의 하위 계층은 Server와 MU로 구성되어 Server와 MU간의 TOA를 측정하는 실내 위치 추적 시스템이 있다. <그림 5>에서 보는 것과 같이 제안하는 알고리즘은 크게 2개의 모듈과 하나의 데이터베이스로 구성된다. 첫째, Fingerprint 역할을 담당하는 FGM(Fingerprint Module)이 있다. 둘째, FGM에서 입력되는 위치 데이터의 보정을 담당하는 KFM(KF Module)이 있다. 마지막으로 본 시스템에서 사용하는 데이터를 저장하고 관리하기 위한 데이터베이스가 있다. 데이터 베이스는 FGM에서 사용하는 데이터를 저장하는 FDB(FGM DB)와 시스템의 전체적인 데이터를 관리하기 위해 사용되는 SDB(System DB)로 구성된다.

Location Algorithm의 흐름은 다음과 같다. 먼저 MU의 위치 추적 어플리케이션에서 이동 객체의 현재 위치에 대한 정보를 요청한다. 그러면 Location Algorithm에서는 하위 위치 추적 시스템에 MU의 현재 위치를 요청하게 되고,

위치 추적 시스템으로부터 요청한 MU의 위치를 리턴 받는다. 이 때 제공받는 위치 데이터는 각 Server와 MU 사이의 거리 정보이다. Location Algorithm은 이 거리 정보를 이용하여 FGM에 위치 데이터를 요청한다. FGM은 3.1절에서 살펴본 위치 추적 시스템에서 얻은 데이터를 활용하여 FDB에서 가장 적합한 RP(Reference Point)를 선별한 후 요청한 이동 객체의 위치를 계산한다. KFM에서는 FGM으로부터 넘겨받은 데이터에 KF를 적용하여 위치 데이터를 보정하게 된다. Location Algorithm은 KFM에서 보정된 위치 데이터를 요청한 MU의 위치 추적 어플리케이션에 응답하는 구조로 구성되어 있다.



〈그림 5〉 제안하는 위치 추정 알고리즘의 구조

3.2 FGM(Fingerprint Modules)

기존 Fingerprint 기법에서는 신호 강도세기

(RSSI : Received Signal Strength Indication)를 활용하여 데이터베이스에 저장한 후 위치를 요청하는 MU에 대응하는 위치를 리턴하는 방법으로 이동 객체의 위치를 측정하였다. 하지만 제안하는 알고리즘에서는 Server와 MU의 거리 데이터를 데이터베이스에 저장하여 이동 객체의 위치를 측정하는 방법으로 구현하였다. 제안하는 FGM의 튜플(tuple) 구조는 <그림 6>과 같다.

RP_No	Bitmap_Structure	Server_Structure	Coord	Option
-------	------------------	------------------	-------	--------

〈그림 6〉 FGM의 튜플구조

<그림 6>에서 RP_No는 참조지점의 식별자를 나타낸다. Server_Structure는 Bitmap Structure와 Server, Received Distance로 구성된다. Bitmap Structure는 사용되는 Server의 사용유무를 쉽게 파악하기 위해 비트맵 인덱스로 구성하여 유지한다. Bitmap은 해당 데이터 값이 기준에 만족하면 1로 설정되고 그렇지 않으면 0으로 설정되는 구조이다. <그림 7>은 Bitmap_Structure에 대해 도시한다.

<그림 7>은 해당 Bitmap의 서버가 사용되면 bit 값이 1로 설정되고, 그렇지 않으면 bit 값이 0으로 설정된다. 따라서 <그림 7>의 RP 1에서 사용되는 Server는 1, 2, 3, 5, 7이 사용됨을 확인할 수 있고, 입력된 데이터는 Server 1, 2, 3, 4, 5에서 입력된 것을 확인할 수 있다. 기본적으로 위치 추적을 하기 위해서는 최소 3개의 Server를 사용해야 하기 때문에 만약 3개 이상이 되지 않는 경우 RP의 정보는 사용하지 않고

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
RPI	1	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Input Data	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

〈그림 7〉 Bitmap_Structure의 구조

다른 RP를 탐색하여 탐색시 오버헤드를 줄일 수 있는 장점이 있다.

Server_Structure의 Server_No는 사용되는 Server의 식별자를 유지하는 부분이고, Dist.는 Server와 MU간의 거리차에 대한 데이터를 유지하는 필드이다. 다음으로 Coord는 Training 단계시에 측정된 RP의 위치 데이터를 저장하는 필드이다. 마지막으로 Option은 향후 확장성을 고려하여 추가시킨 필드이다.

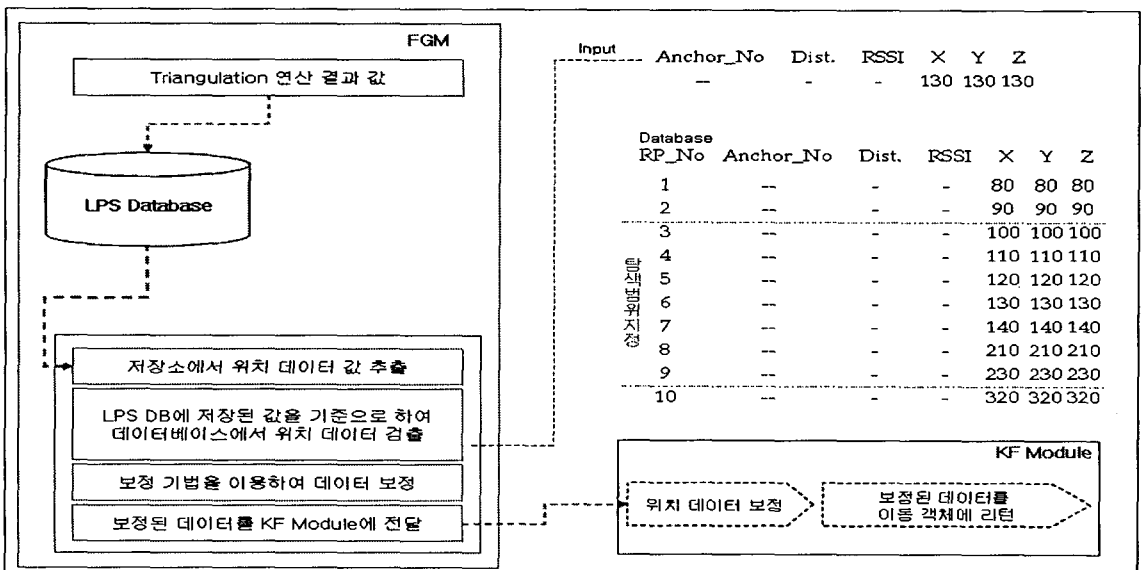
초기 Training 단계는 기존의 2.2절에서 살펴본 Fingerprint와 같이 전체 영역을 대상으로 하여 RP를 일정한 간격으로 배치한 후 RP와 Server간 거리를 측정하여 FDB에 저장한다. <그림 8>은 Training 수행 후 FGM에 저장된 정보의 예시를 보여준다. <그림 8>에서 사용된 Server의 수는 20개로 가정한다. 또한 이해를 돕기 위한 그림이기 때문에 Dist.와 X, Y, Z의 데이터는 상관관계가 없음을 밝힌다. <그림 8>에서 보는 것과 같이 bitmap_ST는 총 20개의 Bitmap으로 구성되어 있다. 여기서 사용된

Server는 1로 사용되지 않은 서버는 0으로 설정하여 유지한다. bitmap_ST에서 보는 것과 같이 총 5개의 Server로부터 수신되었음을 확인할 수 있으며, 사용된 Server는 1, 2, 4, 5, 7번 Server로부터 데이터가 수신되었음을 알 수 있다. 그리고 각 Server로부터의 Dist. 값을 저장하고 있으며, X, Y, Z은 RP의 계산된 좌표 값이다.

RP_No	bitmap_ST	Server_No	Dist.	X	Y	Z
1	1101	1	10	400	400	400
	1010	2	20			
	0000	4	40			
	0000	5	50			
	0000	7	70			
2	0000	7	70	1500	1500	1500
	0010	13	130			
	0000	15	150			
	1011	16	160			
	1000	17	170			

<그림 8> Training 단계 이후 생성된 튜플구조의 예

다음으로 이동 객체의 위치 데이터 요청이 있을 경우 실내 위치 추적 알고리즘은 빠른 응답성을 보장해야 한다. 따라서 제안하는 알고리즘에서



<그림 9> 요청한 이동 객체의 위치와 관련된 RP를 탐색하는 루틴

는 기존 삼각 측량 기법의 데이터를 활용하여 요청한 이동 객체의 위치와 가장 관련이 있는 RP를 찾는다. 이와 관련된 흐름은 <그림 9>에서 확인할 수 있다. <그림 9>에서 보는 것과 같이 TOA 연산 결과값을 활용하여 FGMDB에서 요청한 위치 데이터와 관련된 RP를 발견한다. TOA 값을 활용하면 전체 RP를 탐색하지 않고 수신되는 데이터와 관련있는 RP를 탐색할 수 있기 때문에 탐색범위를 줄일 수 있는 장점이 있다. 요청한 데이터와 관련되는 RP인지 발견하는 기준은 수신된 Server의 Dist를 활용하여 관련된 RP의 위치를 파악한다. 관련된 RP의 위치를 탐색한 후 MU의 위치를 측정하기 위해 보정 기법(Interpolation)을 적용하여 요청한 이동 객체의 위치 데이터를 계산한다. 보정 기법의 결과 데이터를 KFM에 전달하여 위치 데이터를 보정

한 후 MU에 최종 위치 데이터를 전송한다.

<그림 10>은 FGMDB에서 가장 근접한 RP를 탐색하는 알고리즘이다. 먼저 제어할 데이터베이스를 오픈한다(단계 1). 그 후 모든 RP를 대상으로 하여 삼각측량의 결과데이터의 X 좌표와 데이터베이스에 저장되어있는 RP의 X 좌표와 비교를 하여 큰 경우, 선택된 RP로부터 상·하로 3개씩 선택한다. 이 때 FGMDB는 X 축의 내림차순으로 정렬되어 있다(단계 2). 그 후 선별된 7개의 RP를 대상으로 하여 사용된 Server의 정보를 비교한다. Request_Bitmap()은 두 정보를 비교하여 두 정보 모두 '1'을 가지는 개수를 리턴한다(단계 4). Request_Bitmap()의 결과값이 3개 이상이면 해당 RP 후보군으로 선택한다(단계 3, 4). 다음 TRI와 후보군 RP의 데이터의 차이가 시스템에서 설정한 기준값보다 작으면

Algorithm [Searching Algorithm]

```
[INPUT] Coord          TRI          // 삼각측량의 결과 데이터
        Server_Str      Input_Data    // 입력된 데이터의 서버 정보

1) Request_DBOpen();          // 제어할 데이터베이스를 오픈한다.
2) For Each (모든 RP)
    IF (TRIX > RP에 저장되어 있는 X)
        RP를 선별한다. 선택된 RP로부터 상·하 3개씩

3) For Each (선별된 RP)
4)   Request_Bitmap(Server_Str, Input_Data);
    IF (4)과정의 결과값이 3개 이상이면
        해당 RP를 후보군으로 선택한다.

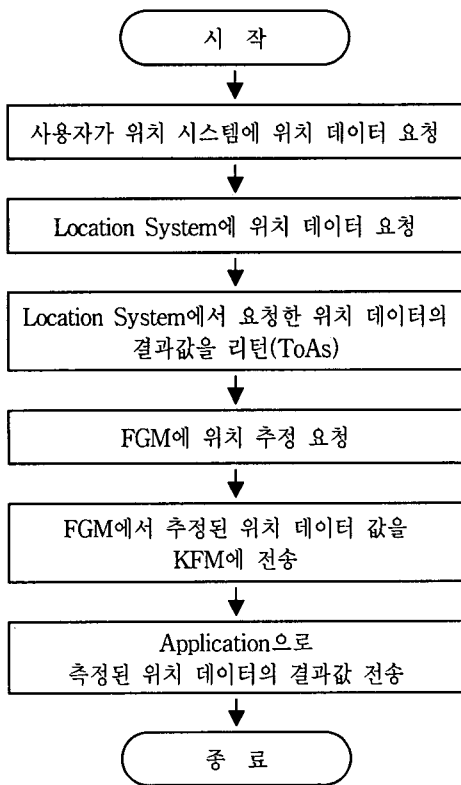
5) IF (TRI와 선택된 RP의 데이터의 차이가 시스템에서 설정된 기준값보다 작으면)
    최종적으로 해당 RP를 선택한다.
    ELSE
        RP를 선별할 수 있도록 확장한다.

6) Return Result_Data;          // 결과 데이터를 리턴한다.
```

<그림 10> 탐색 알고리즘

해당 RP를 선택한다. 만약 기준값보다 작지 않으면 RP를 선별할 수 있도록(단계 2)로 돌아가 RP를 확장하여 선별할 RP의 수를 증가시킨다. 여기서 시스템에서 설정된 기준값은 환경 특성에 종속적이기 때문에 해당 환경에 맞추어 시스템에 설정해야 한다(단계 5). 최종적으로 RP가 선택되면 결과 데이터를 리턴한다(단계 6).

<그림 11>은 제안하는 알고리즘의 전체 프로세스를 보여준다.



<그림 11> 제안하는 알고리즘의 전체 프로세스

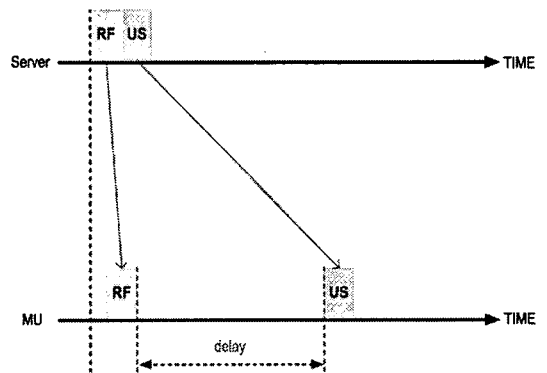
4. 성능평가

4.1 실내 위치 추적 시스템

본 절에서는 제안한 알고리즘의 성능을 평가하기 위해 사용된 실내 위치 추적 시스템에 관해 기술한다.

(1) 개요

본 연구에 사용된 위치 추적 시스템은 LOS 환경일 때, 약 ±30cm의 오차로 거리를 측정할 수 있다. 기본적인 위치 측정 방법은 고정 디바이스(Server)와 이동 디바이스(MU)간의 신호 지연 시간을 계산하여 지연 시간에 대한 거리를 계산한 후 이것을 최소 3개의 MU로부터 신호 지연 시간을 받음으로써 삼각측량에 의해 MU의 거리를 측정할 수 있다.



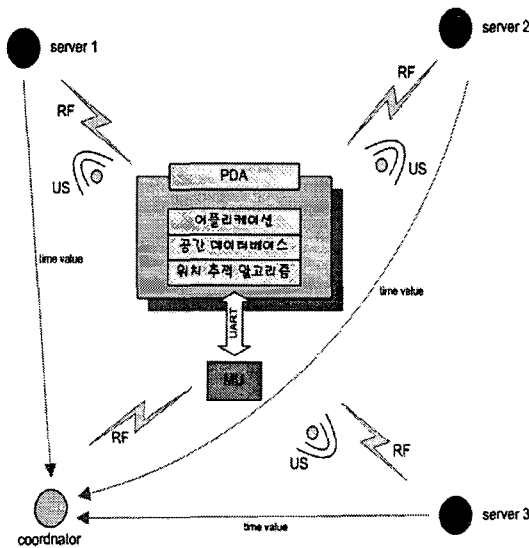
<그림 12> 사용된 시스템에서의 US 지연시간을 계산하는 방법

위치 추적 시스템에서 삼각측량에 필요한 데이터를 얻는 방법은 <그림 12>와 같다. 이동 디바이스는 RF 신호를 보내고 난 뒤 US(Ultrasonic) 신호를 보내게 된다. 이동 디바이스가 송신한 RF 신호를 수신한 고정 디바이스는 MCU의 내부 타이머를 동작 시킨다. <그림 12>에서 RF에 의한 지연이 표시되어 있지만, RF신호는 빛의 속도를 가지며 실내 환경이 넓은 지역을 가지는 경우가 적으므로 이 차이는 무시할 만큼의 작은 값을 가진다. RF 신호에 의해 각 고정 디바이스는 타이머를 동작시킬 정확한 순간을 가지게 된다. US 신호는 초당 380m/s의 속력을 가지므로 RF 신호에 비해 매우 느림을 알 수 있다. Server는 US 신호가 도착할 때 동작 중인 내부 타이머를 중단하고 이때의 타이머 값을 전송한다. 타이머

값은 데이터 수집을 하는 디바이스에 의해 수집되어 호스트(PDA 또는 데스크탑 PC)에 전달하게 된다. 전달된 데이터는 본 논문에서 제시하는 알고리즘에 의해 처리 되어 결과 값을 MU에 전달한다.

(2) 시스템 아키텍처

<그림 13>은 본 논문에서 사용하는 위치 추적 시스템의 개념도를 나타낸다. 고정 디바이스는 이동 디바이스로부터 RF 신호와 US 신호를 사용하여 시간 데이터를 얻게 된다. 이렇게 얻은 데이터는 데이터를 수집하는 디바이스에게 전달된다. 데이터를 수집한 디바이스는 고정 디바이스로부터 받은 각각의 시간 데이터를 거리 정보로 환산하여 RF통신을 통해 MU로 수집된 데이터를 전송한다. MU는 시리얼 라인을 통해 호스트(PDA 또는 데스크탑 PC)에 거리 데이터를 전달한다. 삼각 측량과 어플리케이션은 호스트에서 담당하게 된다. 호스트에서는 거리를 보정하기 위한 여러 가지 통계처리 알고리즘과 필터링 알고리즘을 적용하여 위치의 정확도를 높이고 있다.



<그림 13> 위치 추적 시스템의 개념도

4.2 실험환경

본 실험에서 사용된 환경 조건은 <표 1>과 같다. Location Algorithm은 MU의 이동시 위치 데이터를 측정후 결과값을 MU로 전송하여 이동 객체의 위치를 측정하였다. 본 테스트 환경에서 MU로 전송된 위치 데이터 정보는 PC로 입력되고, PC에서 이동 객체의 위치 정보를 누적하여 유지하였다. 총 10만회 테스트 수행 후 Window Size를 50으로 설정하여 샘플링한 결과를 도시하였다.

<표 1> 실험에서 사용된 환경 조건(파라미터)

MU의 수	1개
Server의 수	5개
RP의 수	9개, 18개, 36개
MU의 시작위치	[X] 3,500 [Y] 3,500 [Z] 1,000 (mm)
측정 면적	7 X 7m ²
객체 이동속도	300cm/sec

실험을 수행한 컴퓨팅 환경은 <표 2>와 같다.

<표 2> 실험 컴퓨팅 환경

CPU	AMD 3000+
RAM	DDR3200 1024MB
OS	Microsoft Windows XP sp2

4.3 RP 수에 따른 성능 분석

본 절에서는 RP의 수에 따른 이동 객체의 위치 정확도를 측정하였다. Fingerprint의 특성상 RP가 많을수록 위치 정확도가 높기 때문에 본 실험에서는 RP의 수를 9개, 18개, 36개로 설정하여 이동 객체의 위치 정확도를 측정하였다. 본 실험에서 이동 객체는 약 1초에 3m를 이동하는 것으로 가정하여 테스트를 수행하였다.

<표 3>과 <그림 14>는 RP를 9개로 설정했을 때의 실험 결과를 보여준다.

<표 3> RP가 9개 일 때의 실제와 측정 값 차이의 평균 값
(단위 : mm)

	측정값
삼각측량	352.6
Fingerprint + KF	271.5

<그림 14>의 X축은 샘플링된 윈도우 크기를 나타내고 Y축은 측정 윈도우에서의 'X' 좌표 값을 나타낸다. 그림에서 보듯이 기존 삼각측량(TRI)의 결과보다 Fingerprint+KF의 조합이 실제 값(Real)과 가까운 것을 확인할 수 있다. <표 3>의 결과 데이터는 총 10만개 데이터의 실제 값과 측정 값 차이의 평균값이다. 약 8cm 정도 정확도가 증가되었음을 확인할 수 있다.

<그림 14>의 A지점과 B지점과 같은 경우 TRI는 입력 데이터의 JUMP 현상으로 인하여 위치

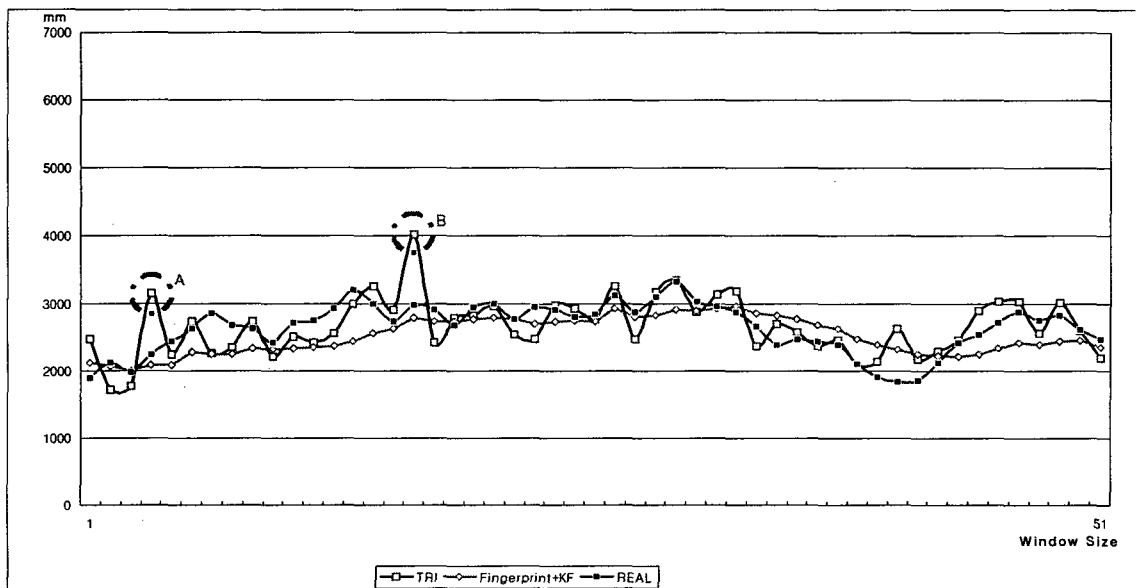
데이터의 오류값이 커지는 것을 확인할 수 있다. 제안하는 알고리즘에서는 Fingerprint와 KF를 결합한 이점으로 A와 B지점같이 데이터가 크게 벗어나는 경우가 현저히 줄어드는 것을 확인할 수 있다.

<표 4>와 <그림 15>는 RP를 18개로 설정했을 때의 실험 결과를 보여준다.

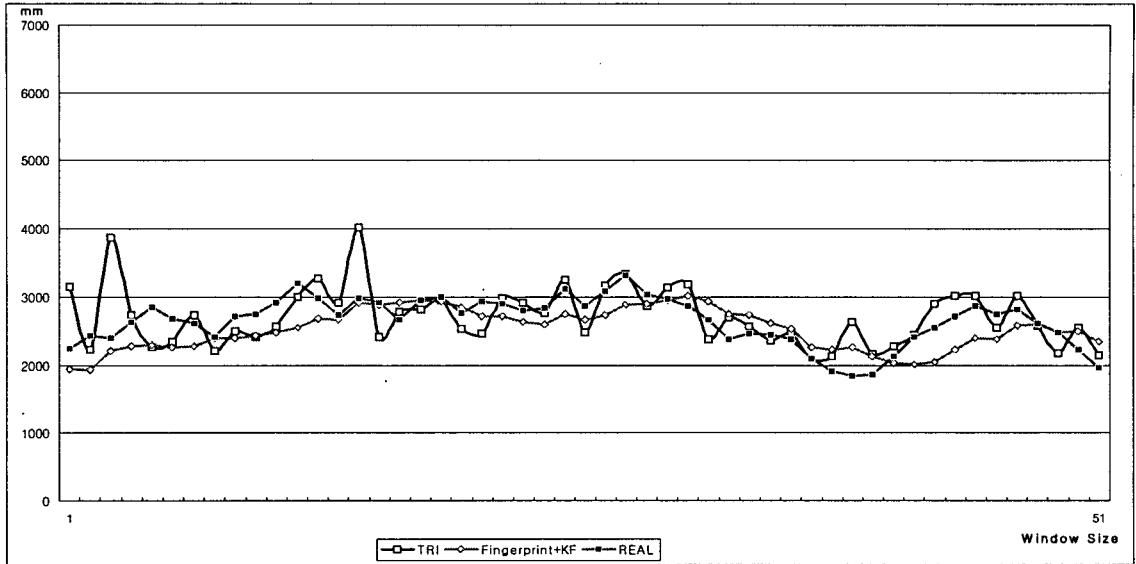
<표 4> RP가 18개 일 때의 실제와 측정 값 차이의 평균 값
(단위 : mm)

	측정값
삼각측량	352.6
Fingerprint + KF	265.5

<그림 15>는 <그림 14>와 같은 범례를 나타낸다. 앞에서 살펴본 것과 마찬가지로 기존 삼각측량(TRI)의 결과보다 Fingerprint+KF의 조합이 실제 값(Real)과 가까운 것을 확인할 수 있다. <표 4>에서 확인할 수 있듯이 기존 삼각측량 기법



<그림 14> RP 9개일 때의 결과 값



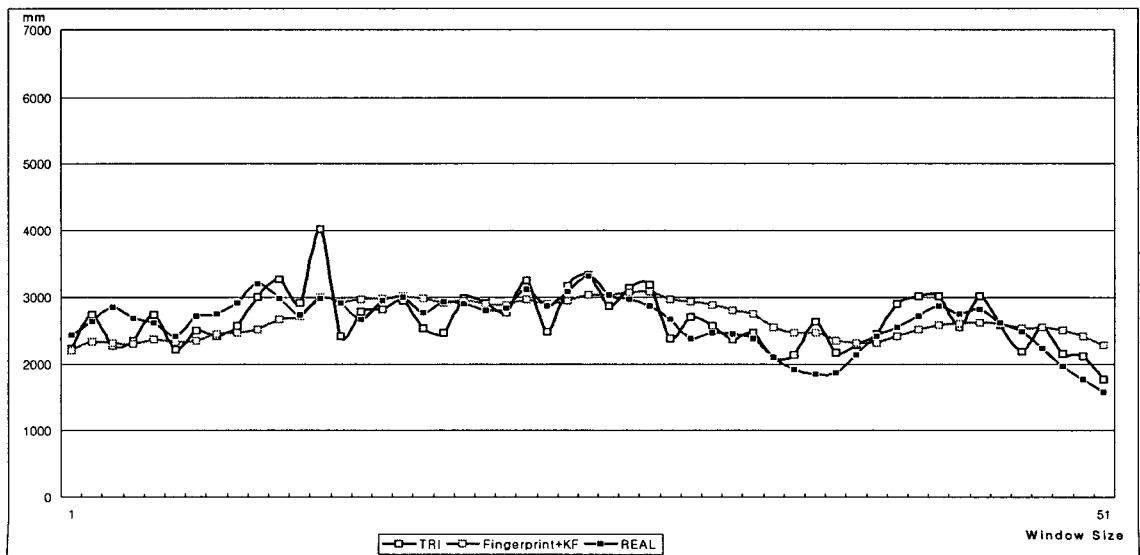
〈그림 15〉 RP 18개일 때의 결과 값

보다 약 8.7cm 정도 정확도가 향상되었음을 살펴볼 수 있으며, RP가 9개일 경우보다 약 0.5~0.6cm 정도 정확도가 향상됨을 살펴볼 수 있다.

〈표 5〉와 〈그림 16〉은 RP를 36개로 설정했을 때의 실험 결과를 보여준다.

〈표 5〉 RP가 36개 일 때의 실제와 측정 값 차이의 평균 값 (단위 : mm)

	측정값
삼각측량	352.6
Fingerprint + KF	258.7



〈그림 16〉 RP 36개일 때의 결과 값

<그림 16>에서 살펴볼 수 있듯이 기존 삼각측량(TRI)의 결과보다 Fingerprint+KF의 조합이 실제 값(Real)과 가까운 것을 확인할 수 있다. <표 5>에서 확인할 수 있듯이 기존 삼각측량 기법보다 약 10%의 결과 데이터는 총 10만개 데이터의 실제 값과 측정 값 차이의 평균값이다. X좌표는 약 9.4cm 정도 정확도가 향상되었음을 살펴 볼 수 있으며, RP가 9개일 경우보다 약 1.2~1.5cm 정도 정확도가 향상됨을 살펴볼 수 있다. 이 실험 결과에서 볼 수 있듯이 Fingerprint는 RP가 많을수록 정확도가 높아진다는 것을 알 수 있다.

5. 결론 및 향후 연구

유비쿼터스 시대가 도래함에 따라 사회적으로 실내 위치 추적 시스템에 대한 관심이 증가되었다. 하지만 기존 실내 위치 추적 시스템은 실내 환경의 빈번한 변화에 능동적으로 대응하지 못하고, 실내 환경의 NLOS 특성으로 인하여 이동 객체의 정확한 위치 측정이 어려운 문제가 있다.

본 연구에서는 유비쿼터스 환경에서 서비스를 제공하는데 필수 요소인 실내 이동 객체의 위치를 효과적으로 파악하기 위한 실내 위치 추적 시스템과 알고리즘을 제안했다. 기존 실내 위치 추적 시스템은 삼각측량을 위치 척도로 사용하였으나, 제안한 시스템에서는 Fingerprint와 KF의 조합을 사용하였다. Fingerprint를 이용하여 NLOS가 빈번한 실내 환경에서의 환경적 요인을 반영할 수 있어, 이동 객체의 위치 정확도를 향상시킬 수 있었음을 실험을 통해 확인할 수 있었다. 그리고 이동 객체의 경우 전파의 JUMP 현상이 발생하여 위치 정확도가 많이 떨어지는 문제점이 발생하는데, KF를 활용하여 이 문제점을 해결하였다.

향후 연구과제로는 제안한 FGMDB의 효과적

인 저장 기법에 관한 연구가 필요하고, 최적화된 탐색기법과 새로운 색인구조에 관한 연구가 필요하다. 또한 실내 환경의 특성을 반영하기 위하여 LOS(Line-of-Sight)와 NLOS(Non-LOS)의 구별하여 시스템에 반영할 수 있는 알고리즘에 관한 연구가 필요하다.

참고 문헌

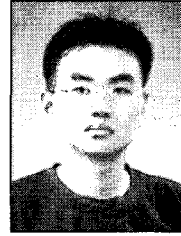
- [1] J. Hightower, G. Borriello, "Location systems for ubiquitous computing", *IEEE Computer*, Vol. 34, No. 8, August 2001, pp. 57-66.
- [2] "Merged Proposal of DS-UWB with Optional CS-UWB on UWB Band for IEEE802.15.4a", <http://www.ieee802.org/15/pub/TG4a.html>, IEEE 15-05-0127-02-004a, 2005.
- [3] J. Y. Lee, R. A. Scholtz, "Ranging in a Dense Multipath Environment Using an UWB Radio Link", *IEEE Journal on Selected Areas in Comm.*, Vol. 20, No. 9, 2002.
- [4] R. J. Fontana, "Experimental Results from an Ultra Wideband Precision Geolocation System", *Short-Pulse Electromagnetics*, 2000.
- [5] R. J. Fontana, S. J. Gunderson, "Ultra-Wideband Precision Asset Location System", *Proc. IEEE Conf. on Ultra Wideband Systems*, 2002.
- [6] P. Krishnamurthy, P. Chrysanthis, P. Personal, "On Indoor Position Location with Wireless LANs Prasithsangaree", *Indoor and Mobile Radio Comm.*, 2002, pp. 720-724.

- [7] E. Haldat, "Fingerprinting Based-Technique for Positioning", <http://www.telecomlab.oulu.fi>, 2002.
- [8] Ding-Bing Lin, Rong-Teng Juang, Hsin-Piao Lin, Cheng-Yi Ke, "Mobile location estimation based on differences of signal attenuations for GSM systems", *IEEE*, Vol. 1, pp. 77-80, June 2003.
- [9] M. A. Youssef et al., "A Probabilistic Clustering-Based Indoor Location Determination System", CS-TR-4350, University of Maryland, 2002.
- [10] A. Smailagic et al., "Location Sensing and Privacy in a Context Aware Computing Environment", Proc. Pervasive Computing, 2001.
- [11] G. Welch and G. Bishop, "An Introduction to the Kalman Filter", UNC-Chapel Hill TR 95-041, 2004.
- [12] N. J. Thomas et al., "A Robust Location Estimator Architecture with Biased Kalman Filtering of TOA Data for Wireless Systems", Spread Spectrum Techniques and Applications, 2000.
- [13] K. K. C. Yu, et al., "An Adaptive Kalman Filter for Dynamic Harmonic State Estimation and Harmonic Injection Tracking", *IEEE Transactions on Comm.* Vol. 20, No. 2, 2005, pp. 1577-1584.
- [14] Bao Long Le, Ahmed K, Tsuji H. "Mobile location estimator with NLOS mitigation using Kalman filtering", *IEEE Wireless Communications and Networking*, Vol. 3, pp. 1969-1973, 2003.

□ 저자소개

**전 현 식**

한밭대학교 정보통신공학과 석사학위를 취득했으며, 현재 한밭대학교 전파공학과 박사과정에 재학중이다. 주요 관심분야로 실내 위치 추적 시스템과 공간 데이터베이스 등이다.

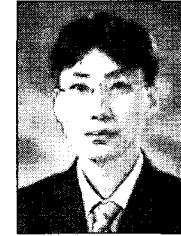
**류 인 선**

한밭대학교 정보통신공학과 학사학위를 취득 후 현재 한밭대학교 전파공학과 석사과정에 재학중이다. 주요 관심분야로는 무선 통신 프로토콜과 블루투스, 임베디드 시스템 등이다.

**우 성 현**

한밭대학교 정보통신공학과 석사학위를 취득했으며, 현재 (주)고산전기엔지니어링에 재직중이다. 주요 관심분야로 LBS, 공간 데이터베이스 Kalman

Filter 등이다.

**윤 성 근**

한밭대학교 정보통신공학과 학사학위를 취득 후 현재 한밭대학교 전파공학과 석사과정에 재학중이다. 주요 관심분야로 무선 통신 프로토콜과 Zigbee,

임베디드 시스템 등이다.

**이 호 응**

한밭대학교 정보통신공학과 석사학위를 취득 후 현재 한밭대학교 전파공학과 박사과정에 재학중이다. 주요 관심분야로는 무선 통신 소프트웨어,

Zigbee, 블루투스, 저전력/실시간 운영체제 등이다.

**박 현 주**

서울대학교 전산학과와 이학박사를 취득하였으며, 현재 국립 한밭대학교 정보통신컴퓨터공학부 부교수로 재직중이다. 주요 관심분야로 임베

디드 소프트웨어와 공간 데이터베이스 등이다.