

# RSSI 판독 라이브러리 함수 및 옥내 측위 모듈 구현

임재걸<sup>†</sup>, 심규박<sup>\*\*</sup>, 정승환<sup>\*\*\*</sup>

## 요 약

IEEE 802.11 기술 덕분에 학교와 대형 쇼핑몰을 비롯한 사무실, 병원, 역 등지에서도 무선 LAN을 통한 인터넷 접속이 가능하다. 본 논문은 무선 LAN에 현재 가장 많이 사용되는 2.4GHz 대역의 802.11b와 802.11g 프로토콜이 탑재된 액세스포인트(AP: Access Point)로부터 수신한 신호의 세기(RSSI: Received Signal Strength Indicator)를 판독할 수 있는 C# 라이브러리 함수를 제안한다. 위치기반서비스는 사용자의 현재 위치를 실시간으로 측정하여 현재 위치를 기반으로 길을 안내하거나, 현재 위치와 관련한 콘텐츠를 제공하는 등의 유용한 서비스를 제공한다. 옥내에서 위치기반서비스를 제공하려면 옥내에 있는 사용자의 위치를 판정하는 옥내측위가 반드시 선행되어야 한다. 옥내측위 기술로 적외선, 초음파, UDP 패킷의 신호세기 등을 이용하는 방법들이 소개된 바 있다. 이러한 방법들은 측위를 위한 특수 장비를 설비해야만 한다는 단점이 있다. 본 논문은 RSSI를 판독하는 라이브러리 함수를 제공할 뿐만 아니라 제공하는 함수를 이용한 옥내 측위 구현 예도 소개한다. 구현에 적용된 방법들은 이미 널리 알려진 K-NN (K Nearest Neighbors), 베이시안 방법 그리고 삼각측량법이다. K-NN 방법과 베이시안 방법은 일종의 지문방식인데, 지문방식은 준비단계와 실시간단계로 구성되며, 실시간 단계의 처리 과정은 처리속도가 빨라야만 한다. 본 논문은 실시간 단계의 속도를 개선하는 방법으로 판단나무 방법(Decision Tree Method)을 제안하고, 이러한 방법들의 성능을 실험적으로 평가한 결과를 소개한다.

## Implementation of a Library Function of Scanning RSSI and Indoor Positioning Modules

Jae-Geol Yim<sup>†</sup>, Kyu-Bark Shim<sup>\*\*</sup>, Seunghwan Jeong<sup>\*\*\*</sup>

## ABSTRACT

Thanks to IEEE 802.11 technique, accessing Internet through a wireless LAN (Local Area Network) is possible in the most of the places including university campuses, shopping malls, offices, hospitals, stations, and so on. Most of the APs (access points) for wireless LAN are supporting 2.4 GHz band 802.11b and 802.11g protocols. This paper is introducing a C# library function which can be used to read RSSIs (Received Signal Strength Indicator) from APs. An LBS (Location Based Service) estimates the current location of the user and provides useful user's location-based services such as navigation, points of interest, and so on. Therefore, indoor LBS is very desirable. However, an indoor LBS cannot be realized unless indoor positioning is possible. For indoor positioning, techniques of using infrared, ultrasound, signal strength of UDP packet have been proposed. One of the disadvantages of these techniques is that they require special equipments dedicated for positioning. On the other hand, wireless LAN-based indoor positioning does not require any special equipments and more economical. A wireless LAN-based positioning cannot be realized without reading RSSIs from APs. Therefore, our C# library function will be widely used in the field of indoor positioning. In addition to providing a C# library function of reading RSSI, this paper introduces implementation of indoor positioning modules making use of the library function. The methods used in the implementation are K-NN (K Nearest Neighbors), Bayesian and trilateration. K-NN and Bayesian are kind of fingerprinting method. A fingerprint method consists of off-line phase and realtime phase. The process time of realtime phase must be fast. This paper proposes a decision tree method in order to improve the process time of realtime phase. Experimental results of comparing performances of these methods are also discussed.

**Key words:** Wireless LAN(무선근거리통신망), Indoor positioning(옥내측위), RSSI(수신신호세기지시자), K-NN(K-최근방), Decision tree(의사결정나무)

## 1. 서 론

위치기반서비스(Location Based Service: LBS)란 사용자가 지정한 지리적 위치나 혹은 현재 사용자가 있는 지리적 위치를 고려하여 사용자에게 위치를 고려한 유용한 정보를 제공하는 서비스로 [1], 선단 관리, 육상 교통 제어, 항공기 통제 등의 목적으로 무수히 많은 실용 시스템이 개발되었다 [2-6]. 실용적인 위치기반서비스는 대부분 GPS(Global Positioning System)나 GLONASS(GLObal Navigation Satellite System)로 사용자의 위치를 판단한다[7]. 이러한 GPS 기반 위치기반서비스는 이동물체가 모두 실외에 있다는 특징이 있다. 그 이유는 GPS 신호는 옥내에서 수신되지 않기 때문이다.

위치기반서비스는 실외에서만 유용한 것이 아니라 실내에서도 역시 매우 유용하다. 예를 들어 박물관 관람객의 위치를 자동으로 판정할 수 있으면 관람객이 관람하고 있는 전시물에 대한 콘텐츠를 맞춤형으로 제공할 수 있으며, 관람객이 원하는 장소를 현재 위치를 고려하여 안내하여 줄 수도 있다.

옥내 위치기반서비스를 위한 다양한 옥내 측위 시스템이 발표되었다. 예를 들면, Active Badgel[8] 시스템에서는 이동물체들에 적외선을 발사하는 송신기가 부착되어 있고, 방마다 여러 개의 수신기가 고정 지점에 설치되어 있다. 적외선은 벽을 통과하지 못하기 때문에 이동물체가 어느 방에 들어오면 그 방에 설치된 고정 수신기들만 그 이동물체의 적외선을 감지할 수 있다. 수신기는 중앙컴퓨터에 연결되어 있어서 수신기에 감지된 적외선을 바탕으로 이동물체가 어느 방 안에 있다는 것을 판정한다. 이상에서 본 바와 같이, Active Badge 시스템에서는 측위를 위한 특수 장비가 필요하고, 측정된 위치의 단위가 방이라는 것을 알 수 있다.

BAT[9,10] 시스템에서는 이동물체에 부착된 송신기가 고주파(RF: Radio frequency) 신호와 초음파(Ultrasound)를 송신한다. 수신기들은 천정에 행렬 모양으로 고정되어 있다. 송신기가 고주파 신호와 초

음파 신호를 동시에 송신하면, 수신기는 이 두 가지 신호가 도착한 시각의 차이를 바탕으로 송신기와 수신기 사이의 거리를 구한다. 번개가 보인 시각과 천둥이 들린 시각의 차이로 번개가 친 장소와 나와의 거리를 구할 수 있는 것과 동일한 원리를 적용하는 것이다. 수신기들은 모두 중앙컴퓨터에 연결되어 있어서 중앙컴퓨터는 수신기들의 좌표와 이동물체와의 거리를 바탕으로 이동물체의 정확한 위치를 판정한다. 이상에서 본 바와 같이, BAT 시스템도 Active Badge와 마찬가지로 측위를 위한 특수 장비를 사용함으로 설치 경비가 많이 든다는 것을 알 수 있다.

Cricket[11] 시스템에서는 Active Badge나 BAT와 반대로 이동물체 자신이 자신의 위치를 계산한다. Cricket도 BAT와 마찬가지로 고주파 신호와 초음파 신호를 함께 사용하는데, BAT에서는 이동물체에 부착된 송신기가 신호들을 송신한 반면, Cricket에서는 고정 지점에 부착된 비컨(beacon)들이 고주파 신호와 초음파 신호들을 동시에 송신하면 이동물체에 부착된 수신기가 이 신호들을 수신하고, 신호들의 도착 시각의 차이를 바탕으로 거리들을 구하고, 비컨들의 좌표와 거리를 바탕으로 자신의 좌표를 구한다. 이상에서 본 바와 같이, Cricket 시스템도 역시 비컨과 수신기와 같은 측위를 위한 특수 장비를 사용한다.

RADAR[12] 시스템에서는 이동물체가 고정 지점에 위치한 세 개의 기지 스테이션(base station: 탁상 컴퓨터를 사용함)에 UDP 패킷을 송신하면 기지 스테이션이 패킷의 신호 세기를 바탕으로 이동물체의 위치를 판독한다. 판독을 위하여 사용되는 기술은 K-NN(K nearest neighbor)이라는 일종의 지문(fingerprint) 방식이다. 지문 방식은 준비단계(Offline phase)와 실시간 측정 단계(On-line phase)로 구성된다. 준비단계에서는 측위 소프트웨어가 적용되는 범위내의 각 지점(후보지점이라고 함)에서 특징 값(RADAR에서는 UDP packet의 신호의 세기)을 측정하여 후보지점과 측정값으로 구성된 찾아보기표를 작성한다. 실시간 측정 단계에서는 사용자의 현재 위치에서 측정된 특징 값과 찾아보기 표의 내용과

※ 교신저자(Corresponding Author) : 심규박, 주소 : 경북 경주시 석장동 707(780-714), 전화 : 054)770-2245, FAX : 054)770-2520, E-mail : gpshim@dongguk.ac.kr  
 접수일 : 2007년 7월 31일, 완료일 : 2007년 11월 14일

† 중신회원, 동국대학교 과학기술대학 컴퓨터멀티미디어학과 교수

(E-mail : yim@dongguk.ac.kr)

\*\* 정회원, 동국대학교 과학기술대학 정보통계학과 교수

\*\*\* 동국대학교 과학기술대학 컴퓨터멀티미디어학과 석사생  
(E-mail : honour@dongguk.ac.kr)

※ 이 논문은 2007년도 정부재원(산학연 공동기술개발컨소시엄사업비)으로 중소기업청의 지원을 받아 연구되었음.

비교하여 사용자의 위치를 판정한다. 이때 K 개의 가장 가까운 지점을 찾아, 이들의 평균을 사용자의 현재 위치로 판정하는 방법을 K-NN이라 한다.

이상에서 살펴본 옥내 측위 방법들은 측위를 위한 특수 장비와 환경을 갖추어야 한다. 특수 장비를 갖추려면 상응하는 시간과 비용을 투자해야만 하는 단점이 있다. 그래서 측위를 위한 특수 장비가 전혀 필요 없는, 무선 랜을 이용하는 방식이 이미 여러 곳에서 발표된 바 있다[13,14]. 이러한 방법들은 무선 랜 장비인 AP(Access Point)가 전송하는 비컨의 신호 세기(RSSI: Received Signal Strength Index)를 이용하여 사용자의 위치를 판정한다.

현재 디지털 홈 네트워크의 응용이 활발히 연구되고 실현됨에 따라 가정은 물론 사무실과 학교, 병원, 역 등지에서 무선 LAN을 이용한 인터넷 사용이 가능하게 되었다. 이것은 유선 LAN 형태인 인터넷의 단점을 보완하기 위해 고안된 IEEE 802.11 기술[15] 덕분으로, 인터넷 네트워크의 말단에 위치해 불필요한 배선작업 및 유지관리 비용을 최소화하는 효과를 준다.

무선 랜을 위하여 이미 설비된 AP의 RSSI를 판독하면 특수 장비의 필요 없이 사용자의 위치를 판정할 수 있다. RSSI를 판독하는 기존의 프로그램은 상용화 서비스로 개발된 프로그램과 C++ 언어로 작성된 제한적인 데모 프로그램이 있으나, 판독된 RSSI 데이터를 바로 사용하기에는 불편함이 있다. 본 논문에서는 흔히 무선 LAN, Wi-Fi(와이파이)라고 부르는 컴퓨터 무선 네트워크에 사용되는 IEEE 802.11 기술에 대하여 알아보고, 현재 많이 사용하는 802.11b, 802.11g 프로토콜에 대하여 C# 프로그래밍 언어에서 RSSI를 판독하는 방법과 그 구현 모듈을 소개한다. 여기서 C# 언어는 C++언어에 비해 프로그래밍 구현 시간이 빠르고 많은 라이브러리를 제공하는 반면, 일부 IEEE 802.11 기술 같은 라이브러리가 아직 제공되지 않았기 때문에 C# 언어에서 자유롭게 사용할 수 있도록 RSSI 판독 모듈을 구현하였다. 또한 지문방식의 다양한 측위 전략을 이 함수를 이용하여 구현하고, 실험을 통하여 성능을 비교한다.

## 2. 관련 연구

옥내 측위 방법에 대한 다양한 연구 결과가 발표

되었다. AP(Access Point)와 이동물체와의 거리를 D, 이동물체에서 측정한 AP의 신호세기를 S라 하고, D와 S 사이의 상관관계를 구하여 사용자의 위치를 판정하는 방법의 실험 결과가 [13]에 발표되었다.

무선 LAN용 AP의 신호 세기를 특정 값으로 하는 지문방식에서 사용하는 기술에는 확률적 방법, K-NN, 그리고 신경망이 있다. 이들 세 가지 기술들의 성능 평가 결과가 [14]에 발표되었다.

[16]은 TOA와 TDOA와 같은 시간 기반 기술의 정확도를 제고하는 방안을 논의하고, 시뮬레이션으로 제안한 방법의 효율성을 검증하였다. TOA는 이동물체가 신호를 송신한 시각으로부터 기지스테이션에 도달한 시각의 차이로부터 거리를 계산하는데 이동물체의 시계와 기지스테이션의 시계가 동기화되기 어렵기 때문에 정확한 거리를 구하는데 어려움이 있다. TDOA는 여러 개의 기지 스테이션들이 각각 도달시각을 측정하여 이들 간의 차이를 바탕으로 거리를 계산한다. 거리를 구한 다음에는 삼각측량 방법으로 이동물체의 위치를 계산한다. 제안하는 방법들은 특수 장치가 필요하며 아직 구현이 불가능함으로 시뮬레이션으로 제안하는 방법을 검증하였다.

[17]은 네트워크 기반 무선 측위 기술들로 TOA와 TDOA와 같은 시간 기반 기술들을 설명하고 간섭이나 감쇄와 같은 이 분야에서 극복해야 할 과제들을 개괄적으로 살펴본다. [18]은 사용자의 위치를 이산적인 정보가 아니라 연속적인 정보로 보고 판정하는 Center-of-Mass라는 방법과 Time-Averaging이라는 두 가지 방법을 소개한다. Center-of-Mass는 모든 후보지점들에 대하여 사용자가 현재 그 곳에 위치할 확률을 구하고 확률 값이 가장 큰 N 개의 지점들의 평균을 사용자의 위치로 판정한다. Time-Averaging은 Center-of-Mass 방법으로 구한 최근 W 개의 추정 값의 평균을 현재 위치로 판정한다.

[19]는 후보지점들을 신호가 잡히는 AP들을 고려하여 clustering함으로써 실시간 측정 단계에서 비교할 후보지점들의 대상을 축소하는 방법을 소개한다. [20]은 무선 LAN의 AP의 신호세기를 바탕으로 위치를 판정하는 방법의 특성으로 자신의 위치를 노출하지 않는다는 것, AP 중 몇 개가 작동을 하지 않더라도 측위가 가능하다는 것, 등을 들고, 찾아보기 표를 작성할 때 사용한 샘플들의 수와 샘플들의 표준편차를 가중치로 사용하는 K-NN 알고리즘을 사용하여

후보지점들 간의 거리와 오차의 관계를 살펴보는 실험을 수행하였다. [21]은 계층적 베이지안 그래프 모형을 측위에 최초로 적용하였으며 여러 사용자의 위치를 동시에 찾을 수 있는 방법을 제안한다.

한편, IEEE 802.11은 IEEE의 LAN/MAN 표준 위원회의 11번째 워킹 그룹에서 개발된 표준 기술을 의미한다. 보통 외부 WAN과 백본 스위치, 그리고 사무실 같은 공간에 설치된 AP 까지 이더넷 네트워크로 연결되면, AP 주변의 컴퓨터들은 무선으로 네트워크를 형성할 수 있다. 여기서 AP는 이더넷 허브와 비슷한 역할을 하며, 고유의 SSID(Service Set Identifier)와 BSSID (Basic SSID(MAC))를 가지고 있어 무선 클라이언트의 연결을 도와준다.

IEEE 802.11 네트워크 환경은 AP에 다수의 클라이언트가 연결되어 네트워크를 구성하는 Infrastructure 방식과 AP 없이 클라이언트끼리 네트워크를 구성하는 AD-HOC 방식이 있다. 그리고 이 무선 LAN의 표준 전송 방식에는 프로토콜의 초기버전에 이어 (b), (a), (g), (n) 순으로 등장하였다. 802.11 초기버전의 프로토콜은 2.4GHz 대역 전파에서 최고 전송 속도 2Mbps로 데이터를 송수신 하며, 이보다 기술을 더욱 발전시킨 802.11b는 최고 전송 속도가 11Mbps이다. 다음으로 등장한 802.11a 프로토콜은 5GHz 대역에서 최고 54Mbps 까지 전송 속도를 지원하며, 802.11g 프로토콜은 2.4GHz 대역에서 (a) 프로토콜과 동일한 전송 속도를 제공한다. 특히 (b) 프로토콜과 쉽게 호환되기 때문에 현재는 (b)와 더불어 (g) 프로토콜이 널리 쓰이고 있다. 마지막으로 등장한 802.11n 프로토콜은 현재 개발 중이며, 최고 540Mbps 까지 속도를 지원할 예정이다. RSSI의 값은 -30dBm 대에서 -90dBm 대까지 이뤄지며, AP로부터 통신 가능한 거리는 실내 45m, 실외 90m 정도로 제한된다.

위에서 나열한 기존의 연구들이 제시하는 측위 방법을 구현하려면, 응용프로그램에서 802.11 네트워크 카드를 제어하고 수신한 내용을 읽을 수 있어야 한다. 본 논문은 이러한 기능을 제공하는 C# 동적링크 라이브러리 함수를 제공한다. 그리고 제공하는 라이브러리를 이용하여 기존의 지문 측위 방법을 구현함으로써 라이브러리의 효율성을 입증한다. 나아가서 기존의 지문 측위 방법의 단점을 보완하는 판단나무 방법을 제안하고 제안하는 방법의 효율성을 실험

적으로 입증한다.

### 3. RSSI 판독 라이브러리 함수 구현

#### 3.1 802.11 NIC과의 통신과정

802.11 NIC(Network Interface Card)는 네트워크 OSI(Open Systems Interconnection) 7계층 중 물리 계층에 위치한 무선 LAN Card를 말한다.

AP(Access Point)들이 SSID, BSSID, RSSI, Network Type등의 정보를 브로드캐스트하면 클라이언트는 802.11 NIC를 통해 정보들을 캐치하고 RSSI가 가장 강한 AP를 선택해 통신을 하게 된다.

#### 3.1.1 NDIS

그림 1에는 윈도우즈의 OSI 7계층의 구조를 보이는데, 이 중 NDIS(Network Driver Interface Specification)는 Microsoft와 3Com이 공동으로 개발한 데이터링크 계층과 네트워크 계층 사이의 부계층으로 데이터링크 계층의 API를 제공한다. 물리 계층에 위치한 802.3 NIC, 802.5 NIC, 802.11 NIC 등과 응용 계층에 위치한 응용프로그램의 통신은 NDIS를 통해 이뤄진다. 응용 계층이 802.11 NIC로부터 정보를 가져온다는 것은 무선 LAN Card와 통신한다는 것을 의미하며, NDIS 드라이버를 Access 할 수 있는 NDIS IOCTL Interface를 구현함으로써 통신 문제를 해결할 수 있다. 그림 1의 NDISUIO 드라이버는 Public Code로 작성되었고 응용프로그램이 NDIS 드라이버와 통신할 수 있도록 인터페이스를 제공하기 때문에 NDIS IOCTL Interface 구현에 큰 도움을 준다.

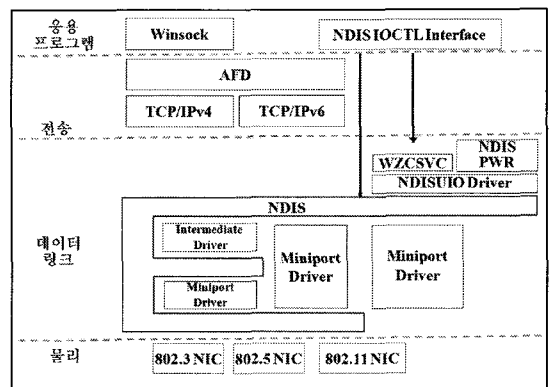


그림 1. 윈도우즈 네트워킹 구조

3.1.2 802.11 자동 구성

그림 1에 보이는 Wireless Zero Config 서비스 (WZCSVC)는 802.11 특정 OID와 드라이버 통신을 하며 Device 상태를 모니터링하고 연결을 초기화 하는 서비스를 제공한다. 기본적으로 Windows XP가 WZCSVC를 이용하고 있기 때문에 이 WZCSVC를 사용하기 위해서는 먼저 Windows XP의 무선 LAN 설정 기능을 비활성화 해야 한다.

3.1.3 802.11 NIC과의 통신 과정

그림 1처럼 응용프로그램은 WZCSVC 서비스를 이용하여 NDISUIO 드라이버를 통해 802.11 NIC와 통신을 할 수 있다. 표 1은 응용프로그램과 무선 LAN Card가 통신하는 과정을 정리한 것이다.

3.2 RSSI 판독 모듈 구현

RSSI를 판독하는 모듈을 구현하기 위하여 그림 2에 보이는 것처럼 자연과학관에 설치된 AP들 중 CISCO AIRONET 1131G Series Model인 AP4를 이용한다. 노트북은 삼성 SENS M40 Model이며, 무선 LAN Card는 Intel(R) PRO/Wireless 2200BG Network Connection이다. 프로그램 개발 도구로는 Microsoft Visual .NET 2005 통합 툴의 C# 언어로 WirelessManager Project를 생성하여 동적 라이브러리인 WirelessManager.dll 파일을 생성한다.

3.2.1 C#에서 802.11 NIC과 통신하기 위한 준비

kernel32.dll에 있는 CreateFile(), DeviceIoControl(), CloseHandle(), FormatMessage() 등의 DLL 익스포트 함수들을 C# 코드에서 호출하기 위해서는 해당 함수에 대응되는 static, extern 메소드를 선언하고 이 메서드에 DllImport 애트리뷰트를 지정해야 한다

표 1. 응용프로그램과 무선 LAN Card의 통신과정

- ① Windows XP의 WZCSVC를 비활성화 시킨다.
- ② kernel32.dll의 CreateFile() 함수로 NDISUIO 드라이버에 접근. CreateFile() 함수는 핸들을 반환한다.
- ③ kernel32.dll의 DeviceIoControl() 함수를 통해 Device들과 통신한다.
- ④ Device와 통신이 끝나면 CloseHandle() 함수로 Device와의 접근을 해제하고 Windows XP의 무선랜 설정 기능(WZCSVC)을 활성화 시킨다.

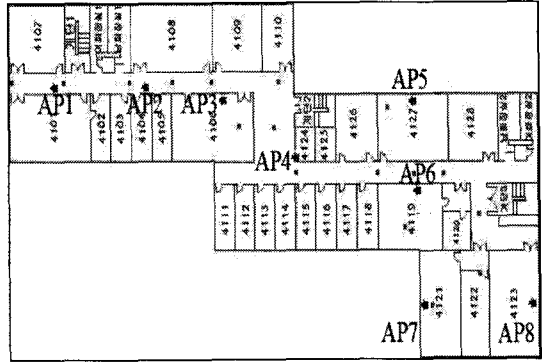


그림 2. 자연과학관에 설치된 AP들

표 2. DeviceIoControl() 함수의 PInvoke 정의

```
[DllImport("kernel32", SetLastError = true)]
private static extern bool DeviceIoControl(
    void* hDevice, IO_CTL_CODE dwIoCtlCode,
    void* lpInBuf, int nInBufSize,
    void* lpOutBuf, int nOutBufSize,
    out int lpBytesRet, void* lpOverlapped
);
```

다. 즉 표 2와 같이 PInvoke(플래폼 호출)를 정의해야 한다. 여기서 표 2는 여러개의 함수들 중 DeviceIoControl() 함수에 대하여 PInvoke를 정의한 것이다.

CreateFile() 함수는 NDIS\_FILE\_NAME(@"\\.\Ndisuio")으로 NDISUIO 드라이버에 연결을 시도하고 연결이 성공되면 Handle을 반환한다. DeviceIoControl() 함수의 첫 번째 매개변수는 앞에서 반환된 Handle이며, 두 번째 매개변수인 dwIoCtlCode는 수행하고자 하는 IO Control Code를 말한다. Network Device와 통신할 경우 DeviceType(0x12), Access(0x1f0x2), Function, Method(0x0)의 속성 값에 따라 IO Control Code를 다음과 같이

$$((DeviceType) << 16) | ((Access) << 14) | ((Function) << 2) | (Method)$$

DDK(Winioctl.h, nuiouser.h)에 정의된 규칙으로 생성한다. DeviceType은 16비트 Left Shift, Access는 14비트 Left Shift, Function은 2비트 Left Shift하고 Method는 Shift하지 않은 상태에서 비트 OR를 함으로써 IO Control Code가 생성된다. 그림 3은 생성된 값이 차지하는 비트의 자리를 보이고 있다.

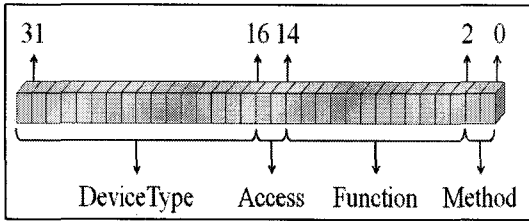


그림 3. IO Control Code 값의 비트의 자리

여기서 Function 속성의 값에 따라 IO Control Code의 수행하는 역할이 달라지는데, Function 속성 값에는 여러 가지가 있기 때문에 본 논문에서는 표 3에 보이는 5개 요소를 사용한다.

만약 통신 작업이 끝나면 CloseHandle() 함수를 통해 연결을 끝내며, 통신 작업 중 발생할 수 있는 Error에 대하여 FormatMessage() 함수를 통해 원인을 메시지로 알려준다.

### 3.2.2 DeviceIoControl() 함수

두 번째 매개변수인 IO Control Code에 따라서 DeviceIoControl() 함수의 사용 용도가 달라진다. 또한 이 용도에 따라 함수 호출에 요구되는 데이터와 호출 후 출력되는 데이터의 존재 여부가 표 3과 같이 각각 다르다.

Bind Wait의 목적은 CreateFile() 함수를 통해 생성된 핸들을 가지고 Network Device로의 접근을 시도하는 것으로, 입력 버퍼와 출력버퍼는 사용되지 않는다.

Query Binding은 Bind Wait가 성공할 경우 모든 유,무선랜카드를 대상으로 각 장치에 대하여 BindingIndex, DeviceName-Offset, DeviceName-Length, DeviceDesc-rOffset, DeviceDiscrLength, DeviceName, 그리고 DeviceDescription을 그림 4에 보이는 바와 같이 Buffer에 출력한다. Binding-Index는 장치가 검색된 순서, DeviceNameOffset은

표 3. IO Control Code에 따른 버퍼의 사용 여부

| IO Control Code         | InBuffer | OutBuffer |
|-------------------------|----------|-----------|
| Bind Wait (0x204)       | 사용 안함    | 사용 안함     |
| Query Binding (0x203)   | 사용       | 사용        |
| Open Device (0x200)     | 사용       | 사용 안함     |
| Set Object ID (0x205)   | 사용       | 사용 안함     |
| Query Object ID (0x201) | 사용       | 사용        |

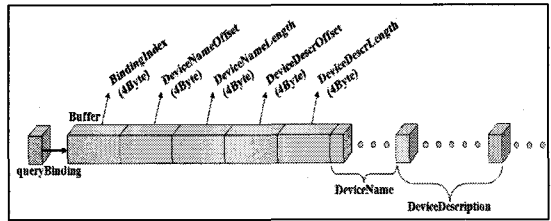


그림 4. Query Binding에서의 Buffer 상태

DeviceName field의 offset, DeviceNameLength는 Device-Name이 차지하는 공간의 크기, Device-DescrOffset은 DeviceDescription field의 offset, 그리고 DeviceDiscrLength는 DeviceDescription field가 차지하는 공간의 크기를 나타낸다. Device-Description을 참조하여 버퍼의 현재 내용이 유선랜카드에 관한 것인지 아니면 무선랜카드에 관한 것인지를 알 수 있다.

RSSI 판독 함수는 DeviceDescription을 참조하여 Buffer에 출력된 정보가 유선랜카드 정보이면 BindingIndex를 1 증가하고 DeviceIoControl()을 호출하여 다음 장치에 대하여 다시 조사하며, 무선랜카드이면 DeviceName과 DeviceDescription을 다른 변수에 저장하여 Open Device에서 사용될 수 있도록 한다.

Open Device의 목적은 Query Binding에서 Wireless LAN Card로 판단한 DeviceName을 입력 버퍼에 저장하여 함수를 호출함으로써, 해당 Device로 열기 작업을 수행하는 것이다. 이때 출력 버퍼는 사용되지 않는다.

Set Object ID의 목적은 Wireless LAN Card로 명령을 내리는 것으로, 입력 버퍼에 DDK의 ntddndis.h에 정의된 OID 중 Bssid\_List\_Scan(0x0D01011A) 값을 저장한 후 함수를 호출하면 Wireless LAN Card가 주변의 AP 단말기들을 대상으로 Bssid List를 수집한다. 여기서 Bssid는 주변 AP 단말기의 브로드캐스팅 정보로 Mac Address, SSID, RSSI, Infrastructure Mode등을 말한다. 그리고 이 함수를 호출하고 대략 6초가 경과해야 Wireless LAN Card가 주변의 Bssid List를 충분히 수집할 수 있다.

Query Object ID의 목적은 Wireless LAN Card로부터 데이터를 가져오는 것으로, 입력 버퍼에 ntddndis.h에 정의된 OID 중 Bssid\_List (0x0D010217) 값을 저장

한 후 함수를 호출하면 출력 버퍼에 Bssid List들이 저장된다. 여기서 그림 5의 아래에 보이는 버퍼 중 맨 앞의 레이블명 Oid와 Data는 입력 버퍼이고 그 뒤에 있는 공간들은 모두 출력버퍼로써, 레이블명 Oid는 함수 호출전에 Bssid\_List값으로 저장되고, 레이블명 Data는 함수가 호출되고 나서 출력버퍼에 저장된 Bssid의 개수로 갱신된다.

### 3.2.3 WirelessManager Project

그림 6의 WirelessManager는 그림 1의 NDIS IOCTL Interface 역할을 수행할 수 있는 Wireless-Manager.dll을 생성한 Project이다. WirelessManager Project는 C# 언어로써 포인터 연산자(\*), 주소 연산자(&), 역 참조 멤버 연산자(->)를 사용하기 때문에 Project 속성에서 unsafe code 요소를 활성화 시켜야 한다.

표 4는 그림 6에 보이는 각 파일들이 하는 일을 정리한 것으로, 크게 데이터 타입을 정의한 파일들과 Wireless LAN Card와의 통신 및 제어를 구현한 파일이다. 그리고 이 Project의 네임스페이스는 "Dongguk3CSLab.Wireless"이다.

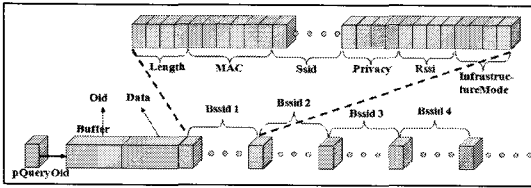


그림 5. Query Object ID에서의 버퍼 상태

표 4. WirelessManager Project의 각 파일

| 파일명                  | 내용                                   |
|----------------------|--------------------------------------|
| ManagerDefinition.cs | ◎ IO Control 타입 정의<br>◎ 802.11 타입 정의 |
| AccessPoint.cs       | ◎ AccessPoint 타입 정의                  |
| MacAddress.cs        | ◎ AccessPoint의<br>MacAddress 구조체 정의  |
| NdisDevice.cs        | ◎ Device 타입 정의                       |
| WirelessManager.cs   | ◎ 무선 LAN과의 통신 및 제어                   |

### 3.3 구현 결과

WirelessManager Project가 생성한 Wireless-Manager.dll을 이용하여 RSSI를 판독할 수 있다.

#### 3.3.1 WirelessManager.dll을 사용하여 RSSI 판독

NDISUIO 드라이버를 사용하려는 C# Project에서 그림 7처럼 WirelessManager.dll을 References에 추가하고, Dongguk3CSLab.Wireless를 using하면 C# 코드 어디에서나 이 모듈을 사용할 수 있다. 표 5와 같이 초기화 하고, 다음과 같이 ScanAPList() 함수를 호출하면

```
AccessPoint[] aplist;
```

```
aplist = WirelessManager.ScanAPList(dev);
```

주변 AP들이 브로드캐스트하는 정보를 수신하여 RSSI를 판독할 수 있다. 그림 8은 Wireless-Manager.dll을 이용하여 임의의 지점에서 RSSI 판독 프로그램을 수행한 실행화면이며, 1초 단위로

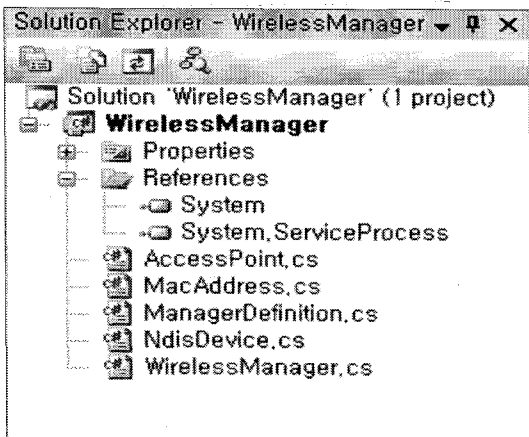


그림 6. WirelessManager Project

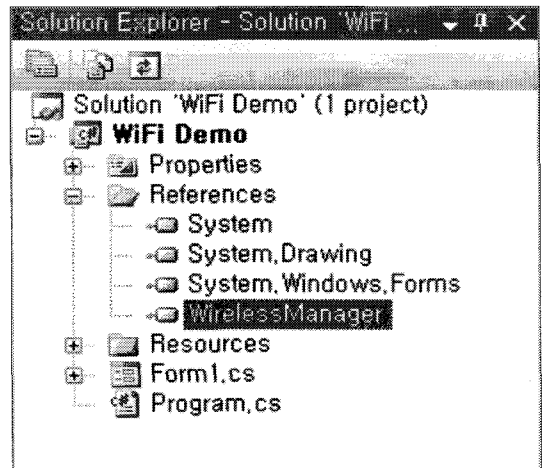


그림 7. Reference 추가

표 5. WirelessManager.dll 사용을 위한 초기화

```

NdisDevice dev;

// 기존에 WirelessManager가 사용되고 있었다면,
// 기존 사용을 종료하고 다시 초기화 한다.
if(WirelessManager.CurrentDevice != null)
    WirelessManager.Finish();
WirelessManager.Start();

// Wireless manager의 Query Binding까지
WirelessManager.OpenWirelessDevice();

// 무선랜카드의 devicename과 devicedescription을
// (그림 4 참조) dev에 배정함.
dev = WirelessManager.CurrentDevice;
    
```

RSSI 값이 갱신됨을 볼 수 있다.

#### 4. 옥내 측위 모듈 구현

본 장에서는 3장에 소개된 RSSI 판독 함수를 이용하여 구현한 옥내 측위 모듈들을 소개한다.

##### 4.1 삼각측량법

현재 지점에서 위치가 이미 알려진 세 지점 각각에 대한 거리를 정확히 알면 현재 지점의 좌표를 정확히 구할 수 있다. 이동단말기를 소지한 사용자의 위치를 측정하는 측위의 경우에는 세 개의 AP가 위치가 알려진 세 지점의 역할을 할 수 있다. AP와 사용자 사이의 거리는 이동단말기에서 판독한 RSSI로 추정할 수 있다. 이 목적을 달성하기 위하여 RSSI와 거리와의 관계식을 작성한다. RSSI와 거리와의 관계식을 구하기 위하여 액세스 포인트에서 1m 단위로 RSSI를 실측한 결과가 그림 9에 보인다. 실측 데이터를 회귀분석한 결과 다음과 같은 관계식을 구하였다.

$$RSSI = -11.61 \ln(\text{Distance}) - 35.333 \quad (1)$$

그림 2에 보이는 테스트베드상의 30개 지점 각각에서 RSSI를 측정하고, 식 1을 적용하여 거리를 구한 다음, 삼각측량법을 이용하여 사용자의 위치를 판정하는 실험을 300회씩 실시하는 실험을 실시한 결과, 본 측위 모듈의 평균 오차는 7.15m라는 결과를 얻었다.

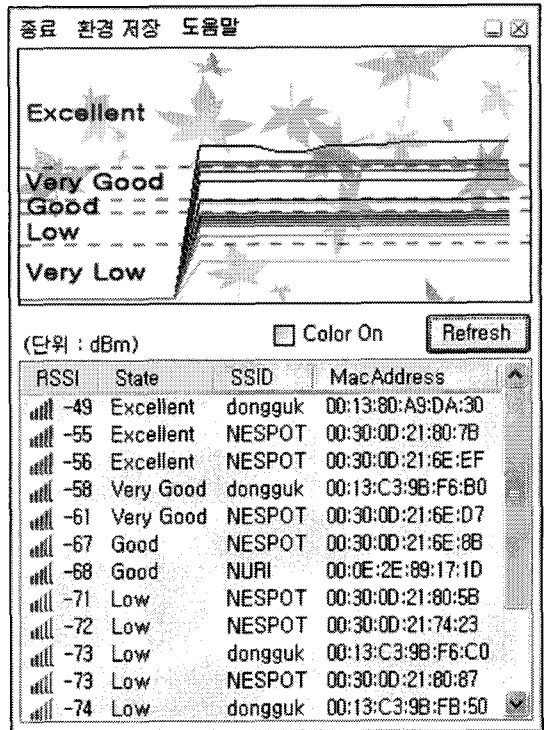


그림 8. RSSI 판독 프로그램의 실행화면

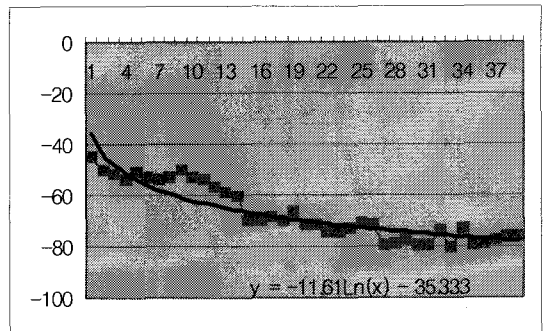


그림 9. RSSI와 거리와의 관계식

##### 4.2 지문방식 알고리즘

지금까지 발표된 지문방식 알고리즘 중 K-NN 방법과 베이지안(Bayesian) 방법을 AP의 RSSI를 읽는 C# 함수를 이용하여 구현할 수 있다.

K-NN 방법은 준비단계(off-line phase)와 실시간 측정 단계(realtime phase)로 구성된다. 준비단계에서는 측위 소프트웨어가 적용되는 범위내의 각 후보 지점에서 RSSI를 측정하여 훈련데이터라 한다. 각 후보지점에서 일반적으로 3 개 이상의 AP들의 RSSI



가 읽힘으로 한 훈련데이터는 3개 이상의 RSSI로 구성된다. 각 후보지점에서 100개 이상의 훈련데이터를 수집하여, 각 AP마다 훈련데이터의 RSSI들의 평균을 구한다. K-NN 방법에서는 이와 같이 구한 훈련데이터의 평균을 지문이라 한다. K-NN 방법은 준비단계에서 후보지점 이름과 지문으로 구성된 찾아보기 표를 작성한다. 실시간 측정 단계에서는 사용자의 현재 위치에서 측정한 RSSI 값과 찾아보기 표의 내용을 비교하여 사용자의 위치를 판정한다. 이때 K개의 가장 비슷한 항을 찾아, 이 후보지점들의 좌표의 평균을 사용자의 현재 위치로 판정하는 방법을 K-NN이라 한다.

준비단계에서 작성한 찾아보기 표의 예로 표 6을 들 수 있다. 표에서 RSSI는 APi가 발신한 신호의 세기이다.

실시간 단계에서 사용자의 현재 위치에서 RSSI 값을 실측하고, 실측한 값과 가장 가까운 항을 찾아보기 표에서 K 개 찾는다. 예를 들어 실시간 측위 단계에서 실측한 값이 X=(38, 46, 53)이라 하면, X와 A의 거리는  $\sqrt{(38-31)^2+(46-48)^2+(53-55)^2}$  이 된다. 비슷한 방법으로 다른 후보지점들에 대한 거리도 구한다. K가 1이라면 X와 가장 가까운 후보지점 하나를 찾아 현재 위치가 그곳이라고 판정한다. K=2라면 가장 가까운 두 후보지점을 찾아 이들의 평균(지점이 좌표임으로 좌표의 평균)을 현재 지점으로 판정한다.

베이지안 방법은 실시간 측정 단계에서 채취한 지문(일련의 AP의 신호 세기들로 구성된 배열)이 주어진 조건에서 각 후보지점이 발생할 조건부 확률이 가장 큰 후보지점을 현재 사용자의 위치로 판정하는 측위 방법이다. 예를 들어, 실시간 측정 단계에서 실측하여 얻은 AP들의 신호 세기들로 구성된 배열이  $X=(x_1, x_2, \dots, x_n)$ 라 하고, 후보지점의 수를 m이라 하자. 다음 식을 만족하는 후보지점  $CP_j$ 를 사용자의 현재 지점이라고 베이지안 알고리즘은 결정한다:

표 6. 찾아보기 표의 예

| 후보지점 | RSSI1 | RSSI2 | RSSI3 |
|------|-------|-------|-------|
| A    | 31    | 48    | 55    |
| B    | 37    | 47    | 54    |
| C    | 40    | 45    | 52    |
| D    | 45    | 45    | 50    |
| ...  | ...   | ...   | ...   |

$$P(CP_i|X) > P(CP_j|X) \text{ for } 1 \leq j \leq m, j \neq i. \quad (2)$$

베이의 정리에 의하여 다음이 참이다:

$$P(CP_i|X) = \frac{P(X|CP_i)P(CP_i)}{P(X)}$$

모든 후보지점에 대하여 P(X)는 상수임으로 P(X|CP<sub>i</sub>)P(CP<sub>i</sub>)를 최대로 하여주는 CP<sub>i</sub>가 곧 식 (2)를 만족하는 후보지점이 된다.

따라서 베이지안 알고리즘은 P(X|CP<sub>i</sub>)P(CP<sub>i</sub>)를 극대화하는 CP<sub>i</sub>를 찾아 준다.

### 4.3. 판단나무 방법

기존의 지문방식은 실시간 단계의 처리 속도를 개선하는 노력이 부족하다. 본 절에서는 실시간 단계의 처리 속도를 개선한 판단나무 방법을 제안한다. 판단나무 알고리즘은 준비단계에서 훈련 데이터(training data)로 그림 10과 같은 판단나무를 건축한다. 훈련데이터는 준비단계 동안 각 후보지점에서 실측한 신호세기 배열들로 구성된다. 판단나무의 내부노드는 레이블로 AP의 이름을 갖고 있다. 레이블이 AP1인 노드의 가지는 AP1에서 읽은 RSSI 값을 레이블로 갖는다. 따라서 가지의 수를 한정하기 위하여 (예를 들면 6개로) 훈련데이터의 RSSI 값들을 급간격으로 이산화한다.

주어진 훈련데이터로 판단나무를 구축하는 알고리즘, Construct\_DT가 표 7에 보인다. 알고리즘의 단계 D. i 은 다음과 같은 식으로 훈련데이터를 후보지점별로 나누는데 필요한 예상정보량인 I를 구한다.

$$I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m p_i \log_2(p_i)$$

단, m은 후보지점의 수, S는 훈련데이터의 수(표 2의 행 수), s<sub>i</sub>는 후보지점 CP<sub>i</sub>의 훈련데이터의 수,

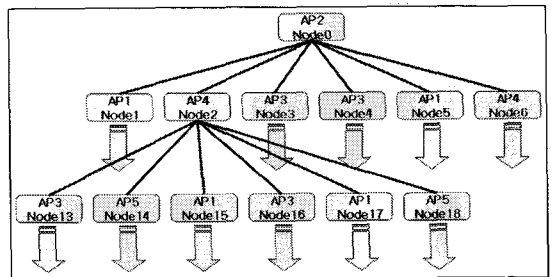


그림 10. 판단나무의 예

표 7. 판단나무를 구축하는 알고리즘

```

algorithm Construct_DT(int[][] Table, ListType MacList, int Index)
{
  A. if (all the CPs of Table are the same, CPi) // Table with only one row satisfies this condition
    i. Tree[Index] = CPi
  B. else if (number of rows in Table == 0)
    i. Tree[Index] = NULL;
  C. else if (number of columns in Table == 1)
    i. Tree[Index] = for each CPi in Table, probability of CPi
  D. else { // Number of rows !=1 and !A and !B and !C
    i. Compute I;
    ii. Compute Entropies for each AP;
    iii. Tree[Index] = MacAddress of the AP with maximum Gain and array P, where P[i] is the probability
        of CPi in Table;
    iv. Construct subMacList;
    v. loop (i=1; i<=number of Intervals; i++) {
      1. generate subTable;
      2. subIndex = Index * number of Interval + i;
      3. Construct_DT (subTable, subMacList, subIndex);
    vi. end loop )
    E. end if }
}
End Construct_DT
    
```

그리고  $p_i = \frac{s_i}{s}$ 이다.

알고리즘의 단계 D.ii는 엔트로피, 즉 훈련데이터를 k번째 AP의 신호세기를 기준으로 구분한 결과를 분류하는데 필요한 예상 정보량을 계산한다.  $AP_k$ 가 다음과 같이 v(급간격의 수) 가지의 값을 갖는다고 하자.

$$\{a_1, a_2, \dots, a_v\}$$

$S_j$ 는 S를 구성하는 행 중,  $AP_k$ 의 값이  $a_j$ 인 행들의 집합이다. 부분집합  $S_j$ 에서 후보지점  $CP_i$ 의 행의 수를  $s_{ij}$ 라 하자. 그러면, 엔트로피  $E(AP_k)$ 는 다음 식으로 구한다.

$$E(AP_k) = \sum_{j=1}^v \frac{s_{ij} + \dots + s_{mj}}{s} I(s_{ij}, \dots, s_{mj})$$

알고리즘의 단계 D.iii은 다음 식으로 정보이득  $G(AP_k)$ 를 구한다.

$$Gain(AP_k) = I(s_1, s_2, \dots, s_m) - E(AP_k)$$

알고리즘 Construct\_DT는 정보이득이 가장 큰  $AP_k$ 를 선택하여 훈련데이터를 분류하는 작업을 더 이상 분류가 되지 않을 때까지 반복한다.

#### 4.4 실험 결과

그림 2에 보이는 테스트베드상의 4119 실험실예 점과 점 사이의 거리가 1m인 바둑판 모양의 후보지점을 정하여, 구현한 K-NN, 베이시안 및 판단나무 방법의 정확성을 비교하는 실험을 실시한 결과가 그림 11에 보인다. X축은 실시간 단계에 이동단말기에서 RSSI를 측정하는 횟수로, 100회 측정의 평균치를 사용하면 측위 결과의 오차가 세 가지 경우 모두 약 2.4m가 되는 것을 보인다. 준비단계는 시간에 구애받지 않는 처리과정으로, 훈련데이터의 수는 일반적인

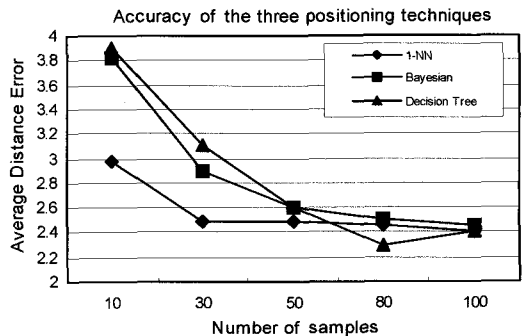


그림 11. 샘플의 수와 세 가지 방법의 정확도의 관계

로 100 개 이상 여러 개임으로 세 가지 방법의 정확도에 차이가 없다고 볼 수 있다.

판단나무를 구축하는 과정의 처리시간은 찾아보기 표를 구축하는 시간보다 더 걸린다. 그러나 주어진 판단나무를 이용하여 사용자의 위치를 판정하는 실시간 단계의 처리 시간은 기존의 방법들보다 빠르다. 판단나무의 깊이가 훈련데이터를 구성하는 AP들의 수보다 크지 않고, 각 노드의 가지의 수가 급간격의 수보다 크지 않음으로, AP들의 수를  $M$ 이라 하고 급간격의 수를  $V$ 라 한다면, 실시간 단계의 처리 시간은  $O(M*V)$ 이다. 후보지점의 수를  $C$ 로 표기하면,  $K$ -NN 방법의 실시간 단계 처리 시간은  $O(C*M)$ 이고 베이시안 방법은  $O(M*V*C)$ 이다. 일반적으로 측위에 사용되는 AP의 수는  $M=5$ 개 이하이고 급간격의 수는  $V=7$ 개 이하인데 반하여 후보지점의 수  $C$ 는 백여 개 이상으로 훨씬 더 큰 수임으로 판단나무 방법의 실시간 단계 처리 시간이 훨씬 빠른 것을 알 수 있다.

## 5. 결 론

본 논문에서는 IEEE 802.11 프로토콜 중 2.4GHz 대역을 사용하는 802.11b와 802.11g를 대상으로 OSI 7계층의 응용프로그램 계층에서 물리계층간의 통신 절차를 알아보고, C#언어에서 WirelessManager Project를 통해 RSSI 판독 모듈을 구현하였다. 생성된 WirelessManager.dll만 있으면 C# 언어로 구성된 프로그램 어디에서나 AP의 정보를 Access할 수 있기 때문에 AP의 정보를 요구하는 관련 연구에서 많은 도움이 될 것으로 예상된다.

또한 구현한 RSSI 판독 모듈을 이용하여, 삼각측량법,  $K$ -NN, 베이시안, 판단나무 방법 옥내 측위 모듈을 구현하고, 각 방법의 정확도를 비교하는 실험 결과를 보였다. 특히 본 논문이 제안하는 판단나무를 이용한 옥내 측위는 정확도 면에서 다른 지문 방식과 대등하면서 실시간 단계의 실행시간이 빠르다는 장점을 보인다.

## 참 고 문 헌

- [1] ETSI GSM 03.71, "Digital cellular telecommunications system (Phase 2+); Location Services (LCS)," 2000.
- [2] Batool, F., Raheed, H., Azim, M. and Khan, S.A., "An Optimal Route & Vehicle Assignment Fleet Management System with Real-Time Monitoring," *First International Conference on Information and Communication Technologies*, ICICT 2005. 27-28, pp. 344-344, Aug. 2005.
- [3] Gerland, H.E., "ITS Intelligent Transportation System: fleet management with GPS dead reckoning, advanced displays, smartcards, etc," *Proceedings of the IEEE-IEE Vehicle Navigation and Information Systems Conference*, pp. 606-611, 12-15 Oct. 1993.
- [4] Liao, Z., "Taxi dispatching via Global Positioning Systems," *IEEE Transactions on Engineering Management*, Vol.48, Issue 3, pp. 342-347, Aug. 2001.
- [5] Lowenstein, G., Phanos, J., and Rish, E., "Sole means navigation in US Navy aircraft," *IEEE Aerospace and Electronic Systems Magazine*, Vol.3, Issue 8, pp. 16-22, Aug. 1988.
- [6] Wang, K., Chuang, C.P., Li, H.W. and Lu, Y.C., "Internet marine transport container global positioning service (IMTCGPS)," *IEEE 5th International Workshop on Networked Appliances, 2002. Liverpool. Proceedings. 2002 30-31* pp. 127-130, Oct. 2002.
- [7] 주재순, 염경환, 이상정, "GPS/GLONASS 통합 수신용 RF 전단부의 설계 및 제작," 정보과학회논문지, 제12권 4호, pp. 1226-2285, 2001.
- [8] Want, R., Hopper, A., Falcao, V., and Gibbons, J., "The Active Badge Location System," *ACM Transactions on Information Systems* 10, 1, pp. 91-102, 1992.
- [9] Harter, A. and Hopper, A., "A New Location Technique for the Active Office," *IEEE Personal Communications* 4, 5, pp. 43-47, 1997.
- [10] Harter, A., Hopper, A., Steggles, P., Ward, A., and Webster, P., "The Anatomy of a Context-Aware Application," *Proc. 5th ACM*

[1] ETSI GSM 03.71, "Digital cellular telecommunications system (Phase 2+); Location

*MOBICOM Conf*, Seattle, WA, Aug. 1999.

[11] Priyanth, N., Chakraborty, A. and Balakrishnan, H., "The Cricket Location-Support System," *Proc. of 6th ACM International Conference on Mobile Computing and Networking*, Boston, MA, Aug. 2000.

[12] Bahl, P. and Padmanabhan, V., "RADAR: An in-building RF-based user location and tracking system," *INFOCOM 2000*, pp. 775-784, Mar. 2000.

[13] Lassabe, F., Canalda, P., Chatonnay, P. and Spies, F., "A Friis-based calibrated model for WiFi terminals positioning," *Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*, 2005. WoWMoM 2005. 13-16 June 2005, pp. 382-387, 2005.

[14] Lin, T.N. and Lin, P.C., "Performance Comparison of Indoor Positioning Techniques based on Location Fingerprinting in Wireless Networks," *Proceedings of the 2005 International Conference on Wireless Networks, Communications and Mobile Computing*, Vol. 2, 13-16, pp. 1569-1574, June 2005.

[15] [http://en.wikipedia.org/wiki/IEEE\\_802.11](http://en.wikipedia.org/wiki/IEEE_802.11).

[16] Yousef, N.R., Sayed, A.H. and Jalloul, L.M.A., "Robust wireless location over fading channels," *IEEE Transactions on Vehicular Technology*, Vol.52, Issue 1, pp. 117-126, Jan. 2003.

[17] Sayed, A.H., Tarighat, A. and Khajehnouri, N., "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, Vol. 22, Issue 4, pp. 24-40, July 2005.

[18] Youssef, M. and Agrawala, A., "Continuous Space Estimation for WLAN Location Determination Systems," *Proceedings of 13th International Conference on Computer Communications and Networks*, (ICCCN 2004) pp. 161-166, 2004.

[19] Youssef, M., Agrawala, A. and Shankar, A.U., "WLAN Location Determination via Clustering and Probability Distributions," *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2003, pp. 143-150, 23-26 March 2003.

[20] Prasithsangaree, P., Krishnamurthy, P. and Chrysanthi, P., "On indoor position location with wireless LANs," *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2002. Vol.2, pp. 720-724, 15-18 Sept. 2002.

[21] Madigan, D., Einahrawy, E., Martin, R.P., Ju, W., Krishnan, P. and Krishnakumar, A.S., "Bayesian indoor positioning systems," *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005)* Vol.2, pp. 1217-1227, 13-17 March 2005.



임재걸

1981년 동국대학교 전자계산학과 졸업  
 1987년 일리노이대학교 시카고 캠퍼스 컴퓨터과학 석사  
 1990년 일리노이대학교 시카고 캠퍼스 컴퓨터과학 박사  
 1992년~현재 동국대학교 과학

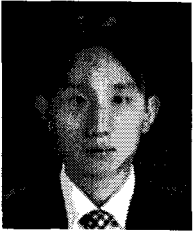
기술대학 컴퓨터멀티미디어 학과 교수  
 관심분야 : 시스템 설계 및 분석, 인공지능, 패트리 넷 이론 및 응용.



심규박

1986년 동국대학교 대학원 통계학과 이학석사  
 1993년 동국대학교 대학원 통계학과 이학박사  
 1994년~현재 동국대학교 과학기술대학 정보통계학과 교수

관심분야 : 전산통계, 신뢰도검정, 통계 자료분.



정 승 환

2007년 동국대학교 컴퓨터학과  
졸업

2007년~현재 동국대학교 대학원  
전자계산학과 석사과정  
재학중

관심분야 : LBS, GIS, 802.11 표준