고급 컴퓨터 네트워크 Term Project

**WiFi Signal Strength 유지를 통한 Gesture Detection의 정확성 향상**

**15조: 최찬희 (2015-21273), 이지화 (2015-31051)**

# Abstraction

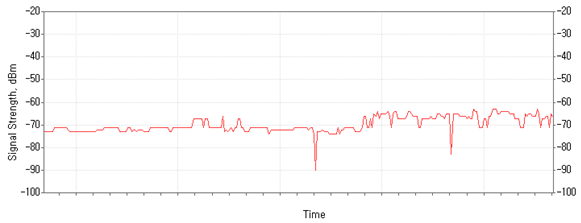
WiGest는 WiFi 신호를 트래킹하여 그 신호가 변화하는 패턴을 인식하는 방법으로 사용자의 제스처를 탐지해내는 프레임워크이다. WiGest는 RSSI 신호의 edge와 pause를 파악하여 제스쳐의 primitives로 사용하기 때문에, WiFi 신호의 세기에 많은 영향을 받는다. 그러므로 WiGest의 정확성을 향상시키기 위해서는 RSSI를 노이즈가 없이 강한 세기로 수신되는 상태가 유지되어야 한다. 이 보고서에서는 주변에 있는 여러 개의 AP들로부터 받는 신호들을 활용하여 제스쳐가 인식되는 수신부의 RSSI가 항상 높은 품질로 유지하는 방법을 소개한다. 이러한 방법을 통해 WiGest의 제스처 인식의 정확도를 높일 수 있다.

# Introduction and Problem Description

최근 모바일 기술의 급격한 성장과 더불어 사용자가 어플리케이션을 컨트롤 하는 다양한 방법들도 개발되고 있다. 그 중 제스처 인식은 다양한 디바이스에서 채용되고 있는 입력 시스템으로서 스마트폰, 노트북, 게임기, 자동차 등 많은 분야에 적용되고 있다. 특히 터치를 기반으로 하는 디바이스에서 사용자의 손이 젖어 있거나, 더러워져있을 때 또는 장갑을 끼고 있거나 다른 행동을 하는 등 터치 입력이 어려운 상황에서 제스처 인식은 특히 유용하다.

기존의 제스처 인식 시스템은 일반적으로 비전 프로세싱이나 자이로센서, 초음파나 적외선 등을 사용하는데 이러한 기존 시스템들은 특정 어플리케이션 전용이거나 빛에 민감하거나, 고가의 장비나 기구가 필요하고 모바일 디바이스를 붙잡고 있어야 하거나 센서를 몸에 부착하거나 입어야 하는 등의 다양한 제약 사항을 가지고 있다. 이러한 제약사항을 극복하고 모바일 디바이스에서 실행되는 다양한 어플리케이션에 핸즈 프리 입력이 가능하도록하는 와이파이 기반 제스처 인식 시스템이 최근 제안되었다. 이 시스템은 와이파이 연결이 가능한 디바이스와 주변에서 쉽게 찾을 수 있는 사회 기반 시설을 이용하는 것으로 사람의 움직임으로 인해 변화되는 무선 신호의 특성을 분석하는 RSSI나 CSI와 같은 것에 기반을 두고 있다. 하지만 무선 채널의 잡음과 무선 전파의 복잡성 등으로 인해 관심 영역에 대한 칼리브레이션이 필요하고 특정 신호 특성을 추출하기 위해 표준 와이파이 하드웨어의 수정이 요구되기도 한다.

WiGest는 앞서 설명한 바와 같이 WiFi 신호를 트래킹하여 사용자의 제스쳐를 인식하는 최신의 연구이다. WiGest는 보정 과정과 특수한 하드웨어를 배제하여 기존의 WiFi device를 그대로 활용하는 것이 장점이다. WiGest는 primitives로 RSSI 신호의 rising edge, falling edge, pause 3가지를 primitives로, gesture 인식의 기본 재료로 사용한다. RSSI 신호를 분석해보면, 손이 멀어지는 것이 rising edge를 만들어내고, 손이 가까워지면 falling edge, 손을 가만히 두면 pause 신호가 나타난다. 이렇게 얻어지는 세 종류의 primitives를 재료로, 속도, 크기, 신호의 순서들을 잘 조합한 패턴을 gesture family로 정의한다. 즉, 손을 특정한 모양으로 움직이면 그 모양에 걸맞는 primitives의 패턴이 나올텐데, 그런 독특한 패턴들을 gesture family로 만들어 둔다. 그러면, application layer에서는 특정한 gesture가 반복되는 횟수, 주기 등을 기준으로 어떤 기능을 수행할 지 미리 mapping을 해두고 후에 사용자가 그 gesture 명령을 내리면 그게 연결된 기능이 수행된다.



WiGest와 같이 WiFi 신호를 기반으로 한 제스처 인식 시스템에서는 노이즈를 처리하는 방법이 중요하다. 다양한 패스로 전파되는 무선 신호의 간섭으로 인해 신호에 노이즈가 끼게 되고, 공기라는 전파 매체의 특성상 전자기장에 의한 노이즈도 쉽게 발생하게 된다. 그런데, WiGest의 경우, RSSI 가 일정수준 이상이면 95%의 높은 인식 정확도를 보여주지만, 이러한 RSSI 수준이 유지되지 않으면 제스쳐 인식의 정확도가 급격히 떨어지는 문제점이 있다. 기존의 WiGest의 경우, AP가 많아지거나 interference로 인해 무선 신호의 품질이 떨어지는 경우에 대해 대비하는 부분이 없다. 또한, 모바일 환경에서 이동하는 경우에 AP가 멀어질수록 RSSI 값이 거리에 따라 급격하게 작아지므로, 그에 따라 제스쳐의 인식률 또한 급격하게 하락한다. RSSI 값이 낮아지는 요인으로 거리에 의한 신호의 감쇄, 주변의 AP 개수가 늘어남에 따른 interference가 있다. 다음의 그림[?]은 거리에 따른 RSSI 값이 낮아지는 것을 확인할 수 가 있다. 또한, 그림[?]은 AP로부터의 거리가 멀어짐에 따라서 RSSI 값이 낮아짐을 보여준다.

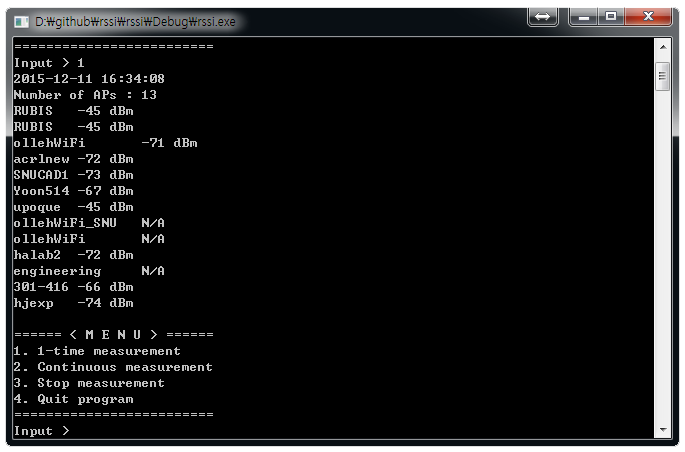
우리는 이 보고서에서 이렇게 RSSI값이 낮아지는 문제를 여러 AP에서 보내는 신호 중 가장 최적의 신호를 골라서 제스처 인식에 사용하는 방법으로 해결하고자 한다.

# Method

기존 논문에서는 제스처 인식을 위해 사용하는 WiFi 신호를 특정 AP 한 개로 한정하여 인식의 정도를 측정하고 무선 환경에서의 일반적 제약 사항을 적용한 실험 사례를 소개하였다. 제약 사항에는 거리에 따른 WiFi 신호 세기의 감쇄와 주변 장애물에 의한 Interference 등이 있었다.

본 보고서에서는 WiFi 모듈이 탑재된 기기 주변의 여러 AP를 효과적으로 활용하는 것을 기초로 하여 위와 같은 제약 사항을 보다 적극적으로 극복해 나가는 방법을 제안한다. 먼저 제스처 인식이 원활하게 이루어질 수 있는 RSSI의 Threshold를 지정하고, 거리의 변화 혹은 주변 장애물에 의해 RSSI가 Threshold 보다 낮아지면 수신 감도가 더 좋은 AP를 선택하는 방식으로 RSSI의 변화폭을 최소화하여 제스처 인식률을 특정 수준 이상으로 유지할 수 있도록 한다.

이에 대한 실험을 위해 WiFi 모듈이 장착된 노트북을 수신 기기로 하고, 실험노트북에서 RSSI를 실시간 측정할 수 있도록 Windows 기반 툴을 제작하였다. 툴은 wlanapi 를 기반으로 구현하였으며 WlanOpenHandle, WlanEnumInterfaces, WlanQueryInterface, WlanScan, WlanGetAvailableNetworkList, WlanGetNetworkBssList 등을 활용하여 WiFi 신호의 RSSI를 측정할 수 있도록 하였다. 프로그램이 실행되면 주변에 탐색되는 모든 AP의 SSID와 각각의 RSSI를 화면에 출력하고, 동시에 텍스트 파일로 저장하여 신호 세기의 분석이 용이하도록 구현하였다.

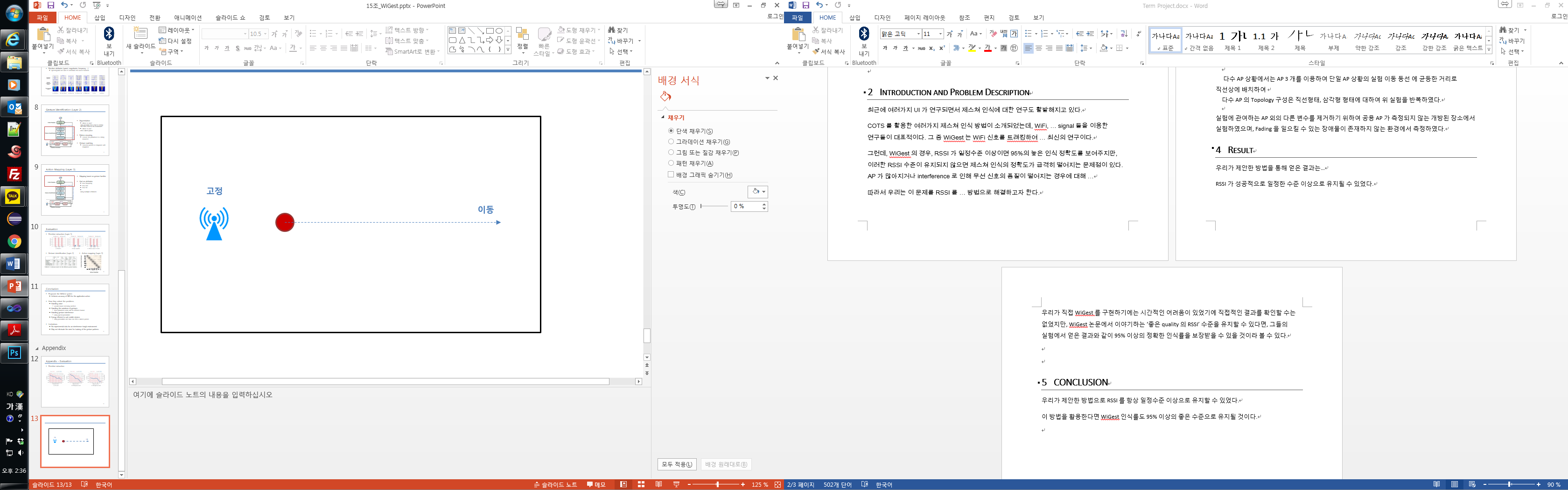


먼저 실험 툴이 WiGest의 기본 기능을 할 수 있는지 판단하기 위하여 단일 AP 상황에서 AP와 노트북간의 거리를 약 1m로 유지하고 손을 아래에서 위로 흔드는 제스처로 인해 RSSI가 변화되는 정도를 측정하였다.

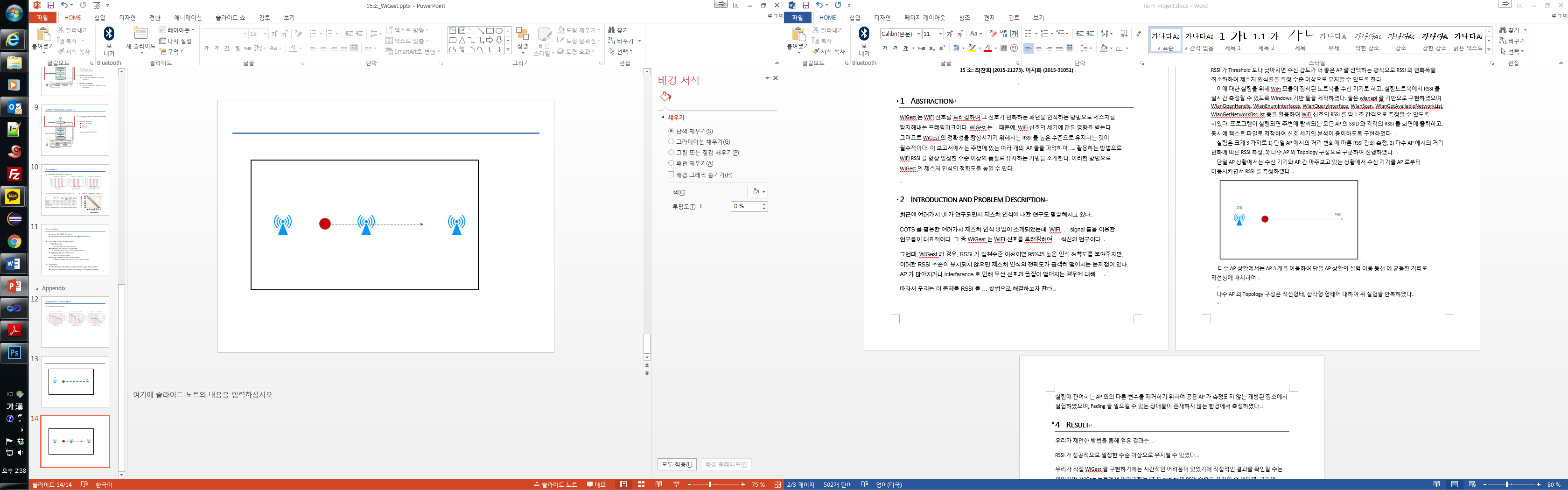
위 그래프와 같이 손의 움직임에 따라 RSSI가 변화되는 것을 확인하였고, 해당 툴을 활용하여 WiGest 인식기의 성능 확인 실험을 진행하였다.

실험은 크게 3가지로 1) 단일 AP에서의 거리 변화에 따른RSSI 감쇄 측정, 2) 다수 AP에서의 거리 변화에 따른 RSSI 측정, 3) AP간 Interference에 의한 RSSI 감쇄로 구분하여 실험하였다. 거리 변화에 따른 측정은 AP와 수신기기 사이의 거리를 0~5m까지 변화 시키면서 0.5m 간격으로 RSSI를 측정하였다.

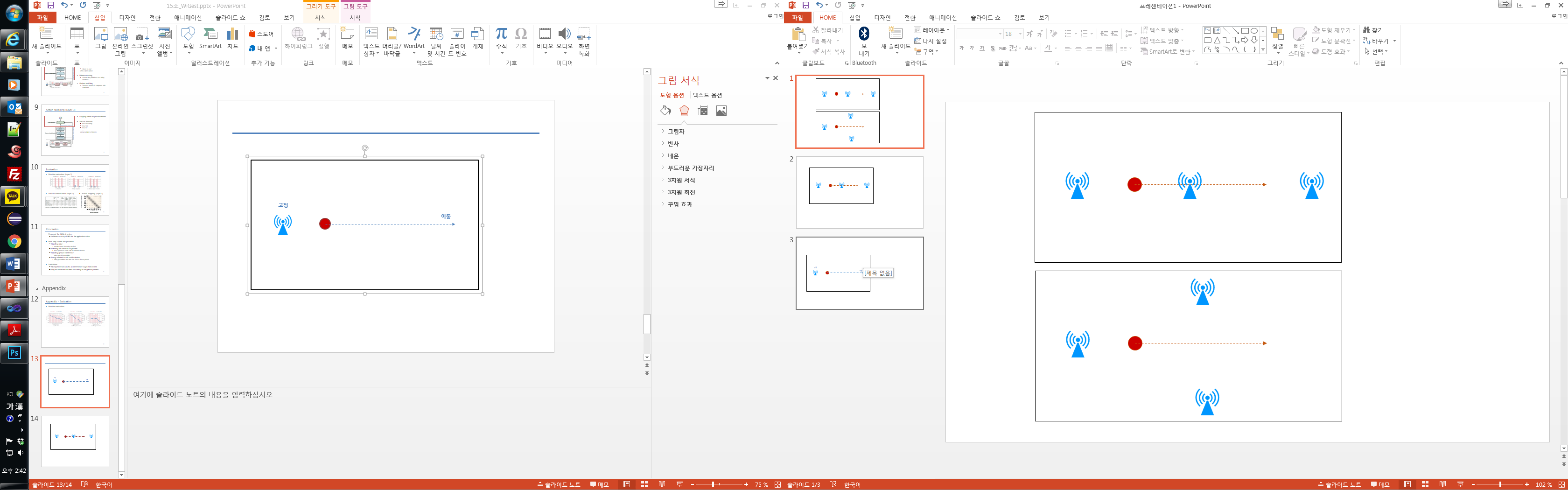
단일 AP 상황에서는 수신 기기와 AP간 마주보고 있는 상황에서 수신 기기를 AP로부터 이동시키면서 RSSI를 측정하였다.



다수 AP 상황에서는 AP 3개를 이용하여 단일 AP 상황의 실험 이동 동선 에 균등한 거리로 직선상에 배치하여 수신 기기 이동 시 각 RSSI를 측정하여 최대값을 선정하였다.



다수 AP의 Topology 구성은 직선형태, 삼각형 형태에 대하여 위 실험을 반복하였다.



실험에 관여하는 AP외의 다른 변수를 제거하기 위하여 공용 AP가 측정되지 않는 개방된 장소에서 실험하였으며, Fading을 일으킬 수 있는 장애물이 존재하지 않는 환경에서 측정하였다.

# Result

우리가 제안한 방법을 통해 얻은 결과는…

RSSI가 성공적으로 일정한 수준 이상으로 유지될 수 있었다.

우리가 직접 WiGest를 구현하기에는 시간적인 어려움이 있었기에 직접적인 결과를 확인할 수는 없었지만, WiGest 논문에서 이야기하는 ‘좋은 quality의 RSSI’ 수준을 유지할 수 있다면, 그들의 실험에서 얻은 결과와 같이 95% 이상의 정확한 인식률을 보장받을 수 있을 것이라 볼 수 있다.

# CONCLUSION

우리가 제안한 방법으로 RSSI를 항상 일정수준 이상으로 유지할 수 있었다.

이 방법을 활용한다면 WiGest 인식률도 95% 이상의 좋은 수준으로 유지될 것이다.

# Appendix

■ 프로그램 수행 환경

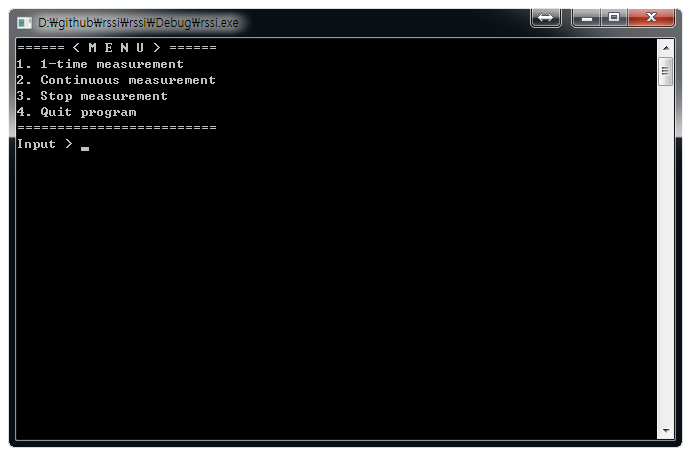
|  |  |
| --- | --- |
| Lan Model | ipTime N100mini |
| Wireless Standard | IEEE 802.11n/b/g |
| Power | USB Power |
| OS | Windows 7 |
| Interface | USB 2.0, WPS button |
| Wireless RF | 2.4 GHz 13Channels |
| RX Sensitivity | -65dBm at 150Mbps  -74dBm at 54Mbps  -84dBm at 11Mbps |
| Transmission Power | 16dBm ±2dB |
| Modulation | OFDM, CCK, BPSK, QPSK |
| ANT | Internal ANT |

■ 프로그램 사용 방법

1. 무선네트워크 활성화 및 AP 연결

2. 프로그램 실행: rssi.exe 실행

3. 모드 선택: 1~4 번호 선택



- 1-time measurement: RSSI 1회 측정

- Continuous measurement: RSSI 연속 측정

- Stop measurement: RSSI 연속 측정 종료

- Quit program: 프로그램 종료

4. 결과 보기: result.txt 생성

- 매 측정마다 측정 시간 기록

- 검색되는 모든 AP의 SSID와 RSSI를 기록