

센서 이용 안드로이드 앱 소개

임재걸[○], 박병훈^{*}

^{○*}동국대학교 컴퓨터공학과

e-mail: {yim[○], ppark^{*}}dongguk.ac.kr

Introducing an Android App Utilizing Sensors

Jaegool Yim[○], Byunghun Park^{*}

^{○*}Dept. of Computer Engineering, Dongguk University at Gyeongju

● 요약 ●

스마트폰의 처리장치 능력이 점점 향상되고, 스마트폰에 장착된 센서들이 점점 더 소형화되면서 동시에 정확해짐에 따라 지능적인 스마트폰 애플리케이션이 무수히 출시되고 있다. 본 논문은 안드로이드 스마트폰의 GPS 센서 값을 이용하여 차량이 정지 상태인지 아닌지를 판정하는 앱을 소개한다.

키워드: 안드로이드(Android), 센서(Sensor), 글로벌위치시스템(GPS)

I. 서론

차량이 정지 상태인지 여부를 판정하는 기술은 지능형 애플리케이션 개발에 널리 사용될 수 있다. 기존의 연구에서는 가속센서 값을 속도로 환산하여 정지 상태를 판정하였으나 실험결과 곡선 길에서 차량이 속도를 늦추는 경우에도 정지라고 판단하는 경우가 자주 발생한다.

안드로이드 스마트폰의 LocationManager[1]는 speed도 반환함에 착안하여, 본 논문은 이를 이용한 차량정지 상태 여부를 판정하는 애플리케이션을 개발한다. 본 애플리케이션이 정확한지 시험하기 위하여 실제 주행상태를 실험자가 버튼을 클릭하여 입력할 수 있도록 한다.

실험 결과, 차량이 0.5초 이상 정지 상태를 유지하면, 제안하는 방법은 100% 정확하게 정지 상태를 인지한다. 또한 차량이 주행을 시작하면 0.5초 이후에 100% 정확하게 주행 중임을 인지한다.

II. 설계

1. 요구사항

본 애플리케이션은 200ms 마다 GPS [2] 신호를 수신하여, speed가 Threshold 보다 작으면 정지 상태라고 판정한다. 판정결과 분석 자료로 쓰이기 위하여 현재시각, 위도, 경도, 속도, 정확도, 현재 Threshold, 사용자가 입력한 실제 상태, 그리고 판정 결과를 200ms 마다 기록한다. 사용자가 실제 상태(정지, 천천히 이동, 빠르게 이동, 곡선 이동)를 쉽고 빠르게 입력할 수 있도록 사용자 인터페이스에 버튼들을 제공한다.

2. 프로그램 구성

본 프로그램은 크게 6가지 부분으로 구성된다. 첫 번째는 앱을 구동하여 초기화를 담당하고 GPS 사용을 요청하는 부분이다. 두 번째는 GPS 신호를 갱신하는 부분이다. 세 번째는 GPS 신호 수신 여부를 보고 기록 또는 기록 정지하는 부분이다. 네 번째는 사용자의 현재 상태를 변경하는 부분이다. 다섯 번째는 Threshold 값을 변경하는 부분이다. 여섯 번째는 Threshold 값과 비교하여 상태를 판정하고 필요한 정보를 기록하며 이를 200ms 마다 반복하는 부분이다.

3. 흐름도

앱 내에서 데이터 기록에 대한 전체적인 흐름도는 그림 1과 같다. 앱을 시작하면 GPS 업데이트를 요청하고 필요한 정보를 저장하기 위한 폴더가 존재하는지 확인하여 존재하지 않는다면 폴더를 생성한다. 다음으로 기록 파일을 생성하고 GPS 신호가 수신되었는지 확인한다. GPS 신호가 수신되면 GPS에 대한 정보를 갱신하고 현재 속도가 Threshold보다 작거나 같은지 확인한다. 만약 현재속도가 Threshold보다 작거나 같다면 Stop을 true로 하고 그렇지 않다면 false로 한다. 기록 파일에 기록할 때는 Stop을 보고 단말기에서 판정한 상태를 기록한다. 다음으로 200ms 대기한 뒤 종료 여부를 보고 종료하지 않는다면 GPS 정보 갱신부터 다시 반복한다.

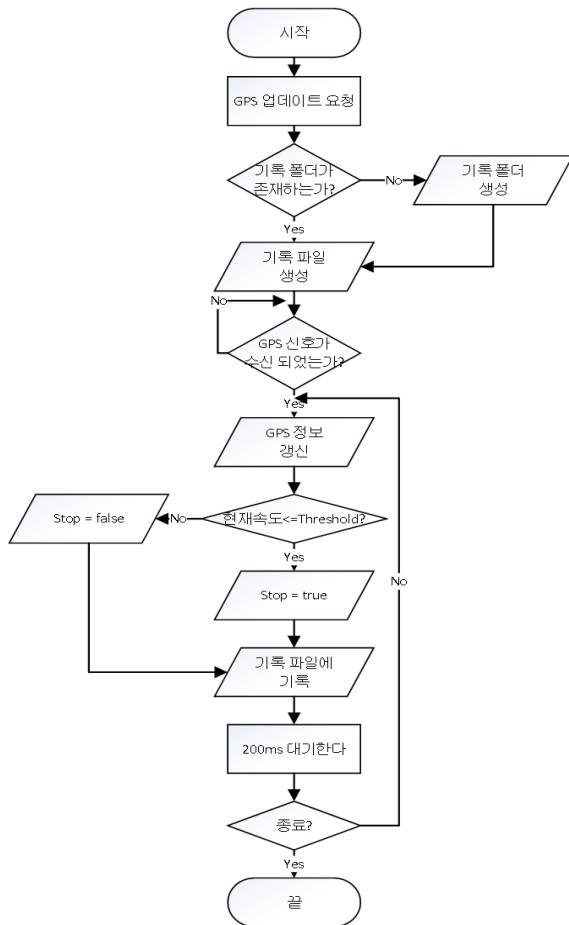


그림 120. 프로그램 흐름도
Fig. 1. A flowchart for the program

III. 구현

GPS 사용을 요청하는 부분과, 데이터를 기록할 파일이 들어 있을 폴더가 존재하는지 확인하고 없으면 생성하고, 데이터를 기록할 파일을 생성하는 코드는 그림 2와 같다. 데이터를 기록할 엑셀파일의 이름은 파일을 생성하는 순간의 현재시각이고, 이 파일이 있는 폴더의 이름은 SensorTest1이다. 그림에서 requestLocationUpdates는 GPS 값으로 위치를 update하고, locationManager에 정의된 onLocationChanged를 실행하라고 명시한다.

```
File file = new File(Environment.getExternalStorageDirectory() + "/SensorTest1");
locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, locationManager);
if (!file.exists()) {
    file.mkdir();
}
file = new File(Environment.getExternalStorageDirectory() + "/SensorTest1/" + cal.get(Calendar.YEAR) + "-" + ... "-" + cal.get(Calendar.MILLISECOND) + ".csv");
```

그림 121. GPS 등록 및 파일 생성 부분
Fig. 2. GPS registration and file creation part

기록시작을 지시하는 버튼과 기록종료를 지시하는 버튼을 사용자 인터페이스에 제공한다. 그리고 레이블이 각각 “정지”, “천천히 이동”, “빠르게 이동”, “곡선이동”인 버튼들을 인터페이스에 제공하여 사용자가 차량의 현재 상태를 손쉽게 빠르게 입력할 수 있도록 하였다.

IV. 결론

정지(이동)상태가 500ms 이상이면 100% 정확하게 정지(이동)임을 판정한다.

사사

본 논문은 한국과학재단의 기초과학연구 프로그램(NRF-2011-0006942)과 지정부의 "IPTV 기반 글로벌 문화관광 방송 기반 구축" (10037393)의 지원을 받음.

참고문헌

[1] <http://developer.android.com/reference/android/>
[2] Jaegeol Yim, Ilseok Ko, Jaesu Do, Jaehun Joo, Seunghwan Jeong, "Implementation of a Prototype Positioning System for LBS on U-campus" Journal of Universal Computer Science, vol.14, no. 14, pp. 2381-2399