

WIFI RSSI ELŐFELDOLGOZÓ KÖNYVTÁR ANDROIDRA

Bogdándy Bence

hallgató, Miskolci Egyetem, Gépészmérnöki Kar
3515 Miskolc, Miskolc-Egyetemváros, e-mail: bogdandy@iit.uni-miskolc.hu

Absztrakt

Jelen cikk keretében a szerző több WiFi RSSI szűrő technika, egy kliens- oldali szűrő alkalmazás, és egy RSSI előfeldolgozó Android szoftver könyvtár fejlesztésének bemutatásával foglalkozik. Három implementált szűrő kerül bemutatásra. A fejlesztés R nyelven folyt.

Kulcsszavak: RSSI szűrő technika, Android, R nyelv, Horus szűrő

Abstract

Within the frame of this paper, the author discusses the development of several WiFi RSSI filtering techniques, a client-side filtering application, and an RSSI pre-processing Android software library. Three implemented filters are presented. The implementation was in R language.

Keywords: RSSI filtering techniques, Android, R language, Horus filter

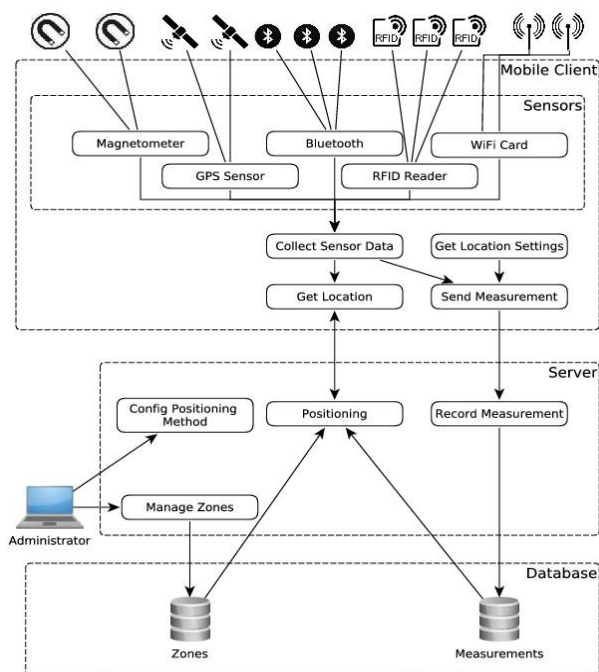
1. Bevezetés

A pozicionálást és navigációt segítő szolgáltatások alapvető feladatot látnak el az élet több területén. A logisztika, a katonaság, és az átlagember közlekedése mind elképzelhetetlen lenne pontos, és naprakész alkalmazások és eszközök nélkül. A globális helymeghatározó rendszerek tervezés alapján csak olyan kültéri környezetben megbízhatóak, ahol a műholdas GPS jelek elérhetőek a kliens eszközökön. Ilyen globális műholdas helyzetmeghatározó (Global Navigation Satellite Systems, GNSS) rendszerekre jó példa a GPS, Galileo és a GLONASS rendszerek. A GNSS rendszerek alapfeltételei közé tartozik a műholdas lefedettség, és az elektromágneses jel akadálytalan útja a vevőkészülékhez, amelyek hiányában a rendszer pontatlan és megbízhatatlan. A beltéri helymeghatározó rendszerek az elmúlt évtizedekben aktív kutatási témává váltak, mivel nincs olyan általánosan elfogadott szabvány, technológia vagy rendszer, amely teljesíti az ilyen rendszernél elvárt követelményeket. A beltéri helymeghatározó rendszereket a felhasználók által használt készülékek pozíciójának megbecslésére tervezték olyan zárt terekben, ahol a GNSS jelek torzulnak vagy elérhetetlenek. Mind a fingerprinting, mind a háromszögelés módszere széles körben elfogadott módszerek, melyek a beltéri pozicionálás kihívásait egyedien oldják meg. Ezeket a módszereket gyakran használják különböző tudományokban, valamint beltéri helyzetmeghatározó rendszerek fejlesztésénél. A háromszögelés pontatlansága a jelcsillapítás hatására keletkezik. Ez a jelenség korlátozza a háromszögelés használhatóságát beltéri helymeghatározó rendszerekben. A fingerprinting módszer képes adatbányászati és heurisztikus technikákat használni a helymeghatározáshoz. A fingerprinting működése a következő: A rendszer felállít egy adatbázist a méréseknél feltérképezett pozíciók koordinátái alapján. Ez után a rendszer különböző algoritmusok segítségével próbálja meghatározni a felhasználó helyzetét az ismeretlen pozíciók esetén. Az ujjlenyomat-alapú beltéri pozicionálási

módszerek hatékonysága növelhető valós idejű kliens oldali jelszűrők alkalmazásával, amelyek növelik a rendszer teljesítményét, és megbízhatóságát. Számos, különböző hardver megoldás létezik a piacon a feladat megoldására, azonban általánosabb, könnyebben elérhető megoldás, az okostelefonok használata.

2. ILONA Rendszer

Az ILONA egy hibrid beltéri helymeghatározó rendszer, amely a különböző szenzorok méréseit használva pontos pozicionálást biztosít. Ezek a technológiák több mérést és szenzort biztosítanak, amely nagy méretű adathalmazt biztosít, amely segít pontosabb helyzetmeghatározási adatok kiszámításában. A rendszer architektúrája az kliens-szerver modellen alapul. Ebben az esetben, a kliens oldali alkalmazás Android eszközön fut, míg a szerver különböző heurisztikus algoritmusokkal számítja ki a felhasználó helyzetét. A rendszer nem egyetlen heurisztikus módszertől függ, hanem kiterjeszhető újszerű pozicionáló algoritmusokkal.



1. ábra. Az ILONA rendszer [7].

Idősor szűrés

A távolságbecslés az AP-k feldolgozatlan RSSI adatainak felhasználásával megbízhatatlan lehet, mivel az RSSI adatok gyakran zajosak. A mérések előfeldolgozása, és a zaj csökkentése nagy mértékben javíthatja a helymeghatározási folyamatot. Az ilyesfajta átalakításokat, és hiba eltávolítást az adathalmazból hívjuk idősor-szűrésnek. A WLAN RSSI megfigyeléseinek szűrése használható arra, hogy csökkentse a hatását a rádióhullámoknál természetesen előforduló inkonzisztenciának. Az API-ban három különböző szűrési módszer lett implementálva: a Horus, Statikus Időablakoló és Dinamikus Időablakoló szűrők.

Horus szűrő

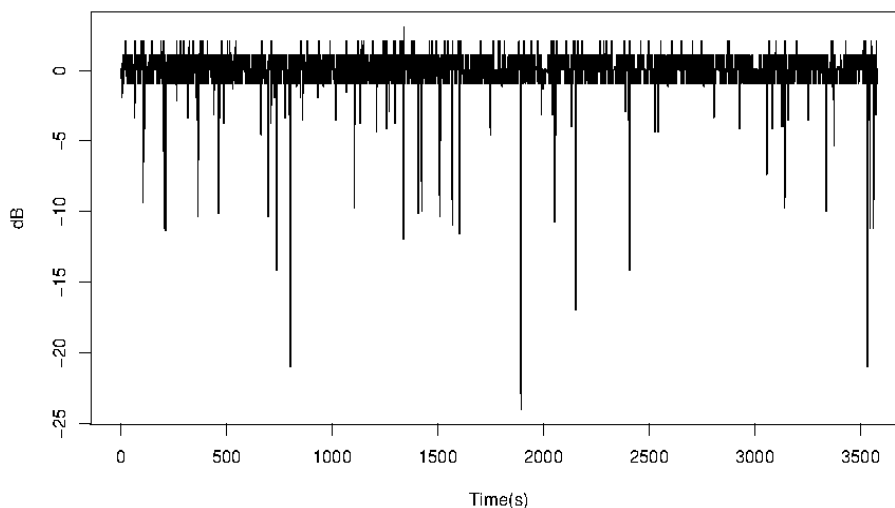
A Horus rendszernek egy saját szűrőmechanizmusa volt, amire ebben a munkában Horus szűrőként hivatkozunk. Ez a szűrő a sorozat értékeinek átlagát veszi, nyers adatok helyett. Ez a módszer csökkenti a mérési hibák súlyosságát. A Horus szűrő implementációja az utolsó M (User specified Memory Size) értékek átlagát veszi figyelembe, és kiszámítja az átlagot, amely csökkenti az váratlan ugrások hatását az RSSI értékekben. A szimulációk azt mutatják, hogy a szűrő kevésbé zajossá teszi az adathalmazt, azonban a szűrő használatából adódik néhány probléma. Ha a jelenlegi mérés kiugró értéket tartalmaz, az átlagolás hatására közelebb kerül majd a zajmentes adathalmazhoz. A következő lépésben az algoritmus figyelembe veszi a korrigált értéket, amelyet a következő érték átlagolásához használ fel.

Időablakoló szűrők küszöbértékekkel

Az időablak-szűrők küszöbértékekkel a Horus szűrő egy továbbfejlesztett, kiegészített változatai. Mindkét módszer időablakoló mechanizmust és küszöbértéket használ arra, hogy meghatározza, hogy a mért adat kiugró-e, és szükség van-e a mérés szűrésére. A küszöbérték mindkét módszerben eltérő módon van kiszámítva.

3. Szimulációk

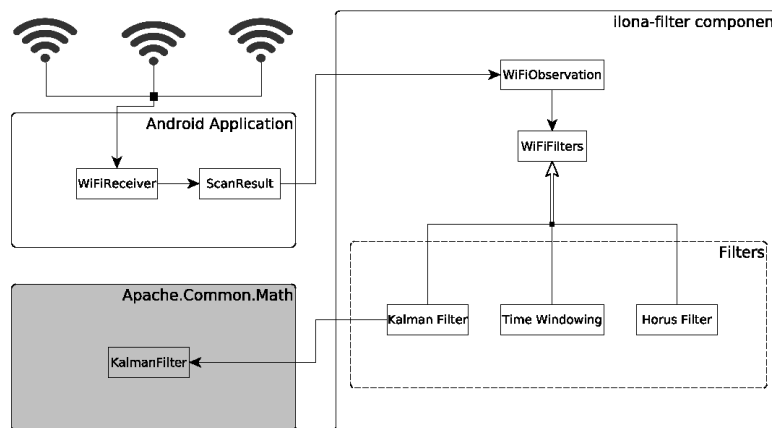
A szűrők szimulációinak létrehozása a szűrőalgoritmusok összehasonlítása és kiértékelése különböző adathalmazokon motiválta. Ezek a mérések három felé csoportosíthatók. Az első egy forgalmas hétköznap alatt, számos járókelővel a rögzítő eszköz környezetében. Ez sok zajt okozott az adathalmazban, ami nagy kiugrásokat jelentett az RSSI értékekben. A második eset egy délutáni munkanapon készült, visszafogott emberi forgalommal a környezetben. Ez átlagos zajszintet eredményezett az adathalmazban. Az utolsó adathalmazt egy csendes pénteken lett rögzítve, kevés emberek által okozott zajjal. A szimulációk R nyelven lettek implementálva. A szűrők előre rögzített adathalmazokon lettek szimulálva. Ezek a szimulációk egyetlen WiFi hozzáférési pontot használtak a valós élethelyzet szimulálása érdekében.



2. ábra. Zajos WiFi RSSI adathalmaz szűrés előtt.

4. Tervezés

A kibővíthetőség és a rugalmasság két kulcsfontosságú szempont volt az API tervezésénél. Az alapfeltételezés szerint kompakt és független komponensnek kell lennie, amelyet bármely Android-projektben könnyedén lehet integrálni. Az Android platformra való fejlesztés biztosítja az ILONA mobil kliens, és szűrő API széles körű elérhetőségét.



3. ábra. A szűrő API felépítése.

Az API képes feldolgozni az Android eszközön felvett RSSI méréseket. Ezeket a méréseket az Android eszköz ScanResults formátumban tárolja, ami egy rendszerspecifikus adat.

Többféle szűrési technika létezik az API-ban, nevezetesen a Horus szűrő és a Időablakoló szűrők. A szűrő komponens olyan Android, vagy más Java alapú alkalmazásokra van tervezve, amelyek igénylik a zajmentes RSSI értékek használatát. Az Android kliens feljegyzi és tárolja a WiFi RSSI értékeket egy ScanResults nevű struktúrában. A szoftverkönyvtár ezen adat feldolgozására lett létrehozva. Az API a ScanResults adathalmazt átalakítja RSSI Observation adattípussá, amely intuitívan manipulálható és feldolgozható az Android API mélyebb ismerete nélkül. Minden szűrési módszer különböző algoritmusokat használ. A szűrő könyvtár általánosított struktúrája lehetővé teszi a másik alkalmazásokkal való egyszerű integrálhatóságot.

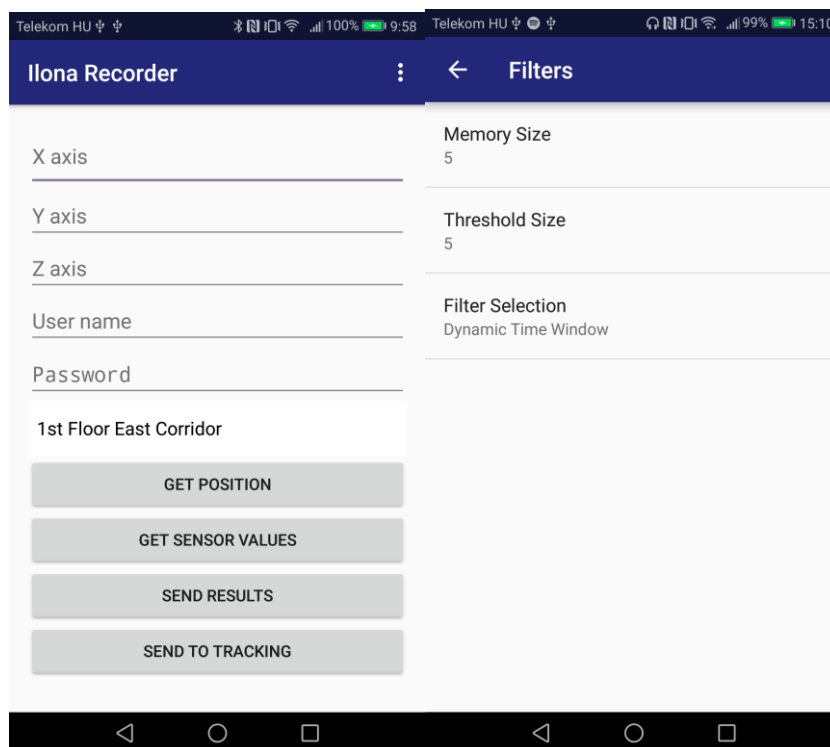
A szűrési szolgáltatások a Strategy (Stratégia) tervezési mintát használják, amely dinamikus viselkedést biztosít az alkalmazásban lévő szűrők működésének. Az alkalmazásban a stratégia megvalósítását a WiFiRSSIFilter absztrakt osztályban láthatjuk. Ennek a komponensnek saját Maven Projectje van, elszeparálva magát az API többi részétől. Minden WiFiRSSIFilter-t öröklő szűrőalgoritmusok egyedi Maven Projectbe lettek szervezve. Az API mindig a WiFiRSSIFilter osztályt használja, de a viselkedés és a kontextus megváltoztatható futási időben, a Strategy tervezési mintának köszönhetően. Ez azt eredményezi, hogy az alkalmazás futás közben tudja dinamikusan változtatni a felhasznált szűrőalgoritmust.

5. Implementáció

A kifejlesztett Android kliens egy fontos lépés volt a fejlesztés során. Ez az alkalmazás volt használva a szűrő API fejlesztése és tesztelése során.

Android Kliens

A kliens eszköz képes észlelni a WiFi RSSI jeleket, és rögzíteni őket aszinkron adatgyűjtési módszereken keresztül. A rögzített adatok szerializálva vannak a WiFi szűrő API segítségével, hogy hatékonyan szűrhető objektumokat létrehozva, a szerver könnyedén feldolgozható objektumokkal dolgozhasson. A mért adatokat egy előre meghatározott szűrő algoritmus alapján dolgozza fel az algoritmus, amelyet az alkalmazás beállításai között finom hangolhat. Ezek a beállítások leírják, hogy mely szűrési algoritmust és paramétereket használja az alkalmazás az adatfeldolgozás során. A WiFi adaton kívül más méréseket is továbbít a szerver felé, mind például a bluetooth eszközök RSSI értékét, magnetométer értékeket, és más, engedélyezett szenzorok által felvett adatokat. A mérések JSON formátumban továbbítódnak a szerver felé. Ez a formátum egyszerűen továbbítható és feldolgozható szerver oldalon. A szerver az adatfeldolgozás után visszaküld egy szintén JSON formátumba szerializált választ, amely egyértelműen leírja a felhasználó szerver által megbecsült pozícióját.



4. ábra. Android alkalmazás.

6. Összefoglalás

Ez a munka több WiFi RSSI szűrő technika, egy kliens- oldali szűrő alkalmazás, és egy RSSI előfeldolgozó Android szoftver könyvtár fejlesztésének bemutatásával foglalkozik. A szűrő technikák betekintést nyújtanak az előfeldolgozó szűrők működésébe, és hatásukra egy RSSI adathalmazon. A három implementált szűrő: a Horus rendszerben megismert szűrő, és két idő és küszöbértéket használó szűrőalgoritmus. Ezen algoritmusok fejlesztése az R nyelven folyt. Az algoritmusok több adathalmaz

segítségével lettek szimulálva, melyek segítettek a teljeskörű szimulációt és tesztelést. A kiterjeszhetőség és rugalmasság fontos szempontok voltak a szűrő könyvtár fejlesztése során. Az interfész alapú fejlesztés és a stratégia tervezési minta használata nagy mértékben megkönnyítette a könyvtár kiterjeszhetőségét különböző szűrő algoritmusokkal. A komponens alapú fejlesztés és a függőség kezelőszoftverek használata lehetővé tették az Android integrációt.

A kifejlesztett API tartalmazza a Horus Szűrőt, és két változatát az időablakos, küszöbértéket használó szűrőknek. A kifejlesztett Android alkalmazás a szűrő könyvtár, és a benne lévő algoritmusok használhatóságát demonstrálja. Az alkalmazás képes a navigációra és a pozícionálásra. Az alkalmazás több gyűjtött információt is továbbít a szerverhez, ami alapján történik a felhasználó pontos helymeghatározása.

A bemutatott szűrő könyvtár használata az ILONA rendszer jövőbeli kliens oldali alkalmazásaiban használva lesz. Más szűrőalgoritmusok implementálásával és használatával jobb betekintést nyerhetünk azok hatásaira a pozícionálásban.

Köszönetnyilvánítás

A cikkben ismertetett kutató munka az EFOP-3.6.1-16-2016-00011 jelű „Fiatalodó és Megújuló Egyetem – Innovatív Tudásváros – a Miskolci Egyetem intelligens szakosodást szolgáló intézményi fejlesztése” projekt részeként – a Széchenyi 2020 keretében – az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

7. Felhasznált irodalom

1. B. Bogdándy and Z. Tóth, “*Comparison of wifi rssi filtering methods*,” in The 10th Jubilee Conference of PhD Students in Computer Science, p. 11, 2016.
2. Z. Weissman, “*Indoor location*,” White paper, Tadlys Ltd, 2004.
3. M. Youssef and A. Agrawala, “*The horus wlan location determination system*, in „Proceedings of the 3rd international conference on Mobile systems, applications, and services, pp. 205–218, ACM, 2005.
4. Z. Tóth, P. Magnucz, R. Németh, and J. Tamás, “*Data model for hybrid indoor positioning systems*,” Production Systems and Information Engineering, vol. 7, no. 1, pp. 67–80, 2015.
5. Z. Tóth and J. Tamás, “*Miskolc iis hybrid ips: Dataset for hybrid indoor positioning*,” in Radioelektronika (RADIOELEKTRONIKA), 2016 26th International Conference, pp. 408–412, IEEE, 2016.
6. O. Aubert and A. Beugnard, “*Adaptive strategy design pattern*,” 2001.
7. Zsolt Tóth (2017) *ILONA: indoor localization and navigation system*, Journal of Location Based Services, 10:4, 285-302.

Jelen cikk a szerző engedélyével jelent meg másodközlésben. Az első megjelenés bibliográfiai adatai: Bogdándy Bence: *WIFI RSSI előfeldolgozó könyvtár Androidra*. Diáktudomány: A Miskolci Egyetem Tudományos Diákköri Munkáiból 11. pp. 59-64. (2018)