

Okostelefon mozgásérzékelő szenzorai pontosságának javítása neurális hálózat segítségével, kalibrálás NFC-vel lekérdezhető hiteles pozíciók felhasználásával

Correction of smartphone motion sensor data using neural net algorithm, calibration with reference positions using NFC

**TAJTI Tibor, PERJÉSI András, GEDA Gábor, MAGYAR Péter,
GREGUS Tamás, BIRÓ Csaba**

**Eszterházy Károly Főiskola, Eger, Eszterházy tér 1. +3636520400,
tajti@aries.ektf.hu, <http://matinf.ektf.hu>**

ABSTRACT

At the Eszterházy Károly College in Eger we participate in a project on the field of Internet of things. In one group of the project we are researching the RFID/NFC technology combined with other possibilities available in mobile phones. Mobile phones are a good target for such researches and developments. Now we are examining the possibility to use mobile sensors and NFC together for indoor positioning with the help of neural network.

ÖSSZEFOGLALÓ

Az egri Eszterházy Károly Főiskolán kutatást folytatunk az Internet of things témakörben. Az egyik csoportban az RFID/NFC technológia és a mobiltelefonokban elérhető egyéb képességek felhasználásával foglalkozunk. A mobiltelefonok jó célpontjai az ilyen irányú kutatásoknak, fejlesztéseknek. Most a mozgásérzékelő szenzorok és az NFC együttes felhasználásának lehetőségeit vizsgáljuk beltéri pozicionálás érdekében neurális hálózat támogatásával.

KULCSSZAVAK

Internet of things, mobiltelefon szenzorok, NFC, neural network, beltéri pozicionálás

1 BEVEZETÉS

Az Eszterházy Károly Főiskola és a Bay Zoltán Alkalmazott Kutatási Nonprofit Kft a TÁMOP-4.2.2.C-11/1/KONV FutureRFID - Az RFID/NFC technológia továbbfejlesztési lehetőségei az „Internet of Things koncepciói mentén című projekt keretén belül több kutatócsoportban kutatja a terület műszaki, informatikai, jogi és társadalmi kérdéseit.

A hibrid megoldások, az egyes eszközök kombinálhatósága számos esetben nyit új perspektívákat, jelenleg a mobiltelefonok NFC és egyéb képességei együttes alkalmazásával foglalkozunk.

Az okostelefonok többsége rendelkezik GPS képességgel, ami a szabadtéri pozicionálást lehetővé teszi néhány méteres pontossággal. A GPS használhatósága a műholdak jeleinek vételétől függ. Ezt a környezetben lévő objektumok (hegy, fák, épületek) gátolhatják. Zavart vétel esetén a pozicionálást egyéb technikákkal lehet pontosítani, pl. a hiányzó jelek ideje alatt az előzőleg számított irány és sebesség feltételezése, a GSM, WIFI, Bluetooth hálózat jeleinek erősségéből az adótól való távolság számítása (RSSI – Received Signal Strength Indication), esetleg a mobil készülékben elérhető

mozgásérzékelő szenzoroktól származó információk, amelyekből az irányt, lassulást, gyorsulást lehet kiszámítani. Ezen lehetőségek segítséget adhatnak addig, amíg a GPS által a pozíció újra nem számítható.

Zárt térben (épületben, barlangban, pincegarázsban) a GPS jelek általában nem használhatóak. Számos ilyen esetben is hasznos lenne viszont a pozicionálás lehetősége, pl. metró aluljáróban, bevásárló központban, vagy annak garázsában. A feladat fontosságát jelzi, hogy a megoldásra a gyártók egy szervezetet hoztak létre In-Location Alliance néven.[1]

2 POZICIONÁLÁS LEHETŐSÉGEI

Ennek a feladatnak egy megoldási lehetőségével foglalkozunk mi is. Ha a GPS jel nem áll rendelkezésre, a kezdő pozíció meghatározása is megoldandó feladat. Egyszerű és hatékony megoldás lehet erre a megfelelő sűrűséggel kihelyezett tábla, amelyhez odalépve egy mozdulattal lekérdezheti a mobiltelefonon futó alkalmazás a pontos pozíciót. Erre több megoldás is lehetséges:

- A pozicionálást segítő tábla tartalmazhat egy NFC tag-et. A mobiltelefonok egyre nagyobb arányban rendelkeznek NFC képességgel, amely segítségével pl. a mobil pénztárca is működhet. Az NFC tag-től az alkalmazás a telefon odaérintésével lekérdezheti az épületen belüli pontos pozíciót, és egyéb információkat, akár az épület térképét is.
- Azon telefonok számára, amelyek még nem rendelkeznek NFC csippel, de kamerával igen, ilyen információkat QR kódként lehet ugyanott elhelyezni, amely tartalmazhatja a pozíciót, és esetleg WI-FI vagy Bluetooth kapcsolat felvételéhez szükséges információt, amely térkép vagy egyéb szükséges térkép adatok letöltését teheti lehetővé.
- URL-t vagy más azonosító kódot, amely segítségével az adatokat az alkalmazás hálózaton keresztül le tudja tölteni. Ebben az esetben a felhasználó érdeke, hogy a telefon ténylegesen ott legyen, amely pozíció adatait letölti az alkalmazás.

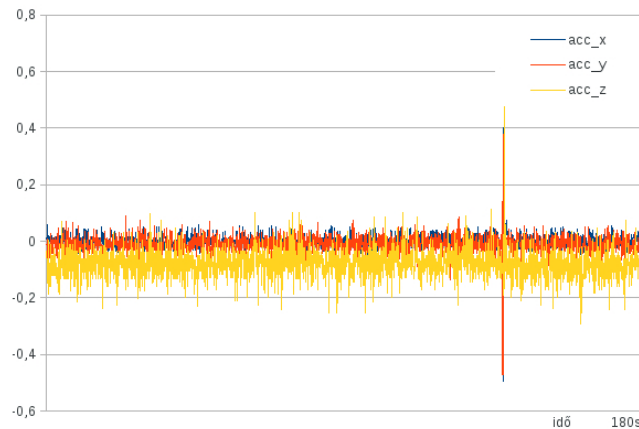
Az NFC csip alkalmazása a QR kódhoz képest plusz lehetőségeket ad, mivel hálózatra kapcsolása esetén online frissíthető információkat is tartalmazhat.

A pozíció beállítása után a telefon a beépített érzékelők által szolgáltatott adatokra hagyatkozhat. Elsősorban a következő szenzorokra lehet építeni:

- magnetométer (iránytű)
- gyorsulásmérő
- giroszkóp
- gravitáció szenzor

Ezen szenzorok méréseinek összegzésével elméletben a kiindulópont ismeretében a pozíció kiszámítható. A mérési tapasztalatok azonban azt mutatják, hogy a szenzorok által szolgáltatott adatok pontatlanok, és nem folytonosak, így nem folytonosan integrálhatóak, ezért a pozíció számításához javítás nélkül felhasználva őket nagyon pontatlan eredményt kapunk.

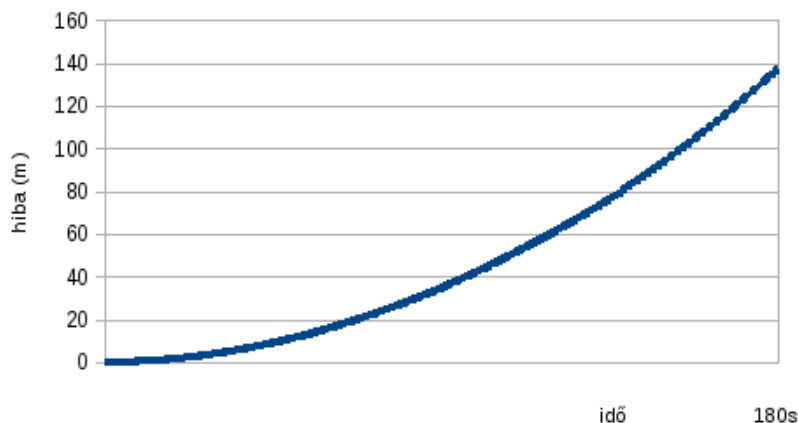
3 SENZORADATOK PONTOSSÁGA, ZAJ



1. ábra
Gyorsulásmérő adatok mozdulatlan helyzetben

A szenzorokból érkező adatok pontosságának és a javítási lehetőségek kutatásához mobil telefon alkalmazást fejlesztettünk, amely képes a lekérdezett adatokat lementeni a vizsgálatok számára.

Az 1. ábrán példát láthatunk gyorsulásmérő szenzor adataira. Jól látható, hogy mozdulatlan helyzetben is kétféle hiba is terheli az adatokat. Az egyik hiba a „z” irányú gyorsulás adatok átlagos negatív irányú távolsága az X tengelytől. Ezt a hibát kalibrálással lehet javítani. A másik hiba a zaj, amit zajszűrő algoritmus alkalmazásával lehet csökkenteni. A kalibrálás és a zajszűrő algoritmus megfelelő kiválasztása és paraméterezése is sok kísérletezést igényel.



2. ábra
A pozíciósámítás hibája az idő függvényében

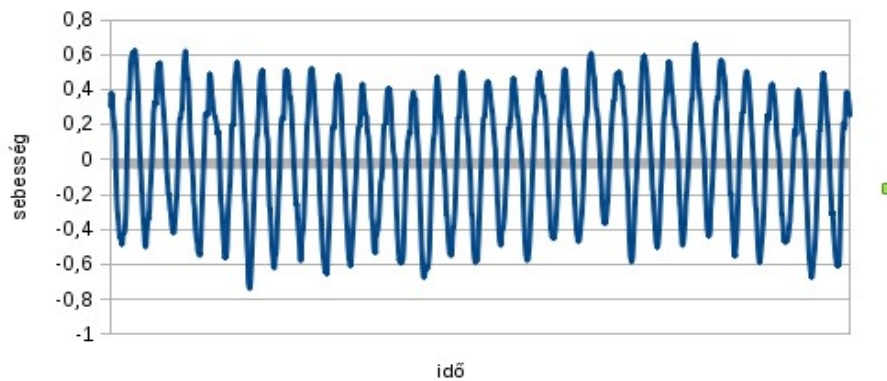
Ugyan a hiba mértéke nem tűnik nagyknak, a pozíciósámítás hibája az összegzés miatt az idő függvényében exponenciálisan növekedhet.

A 2. ábrán a grafikon 3 perces időtartamot ábrázol. Ezen idő alatt már több mint 100 méterre nőtt a számított pozíció hibája. Nagyon fontos tehát a hibát csökkentő lehetőségek kihasználása.

4 MOZGÁS KÖVETÉSE GYORSULÁS ADATOK ALAPJÁN

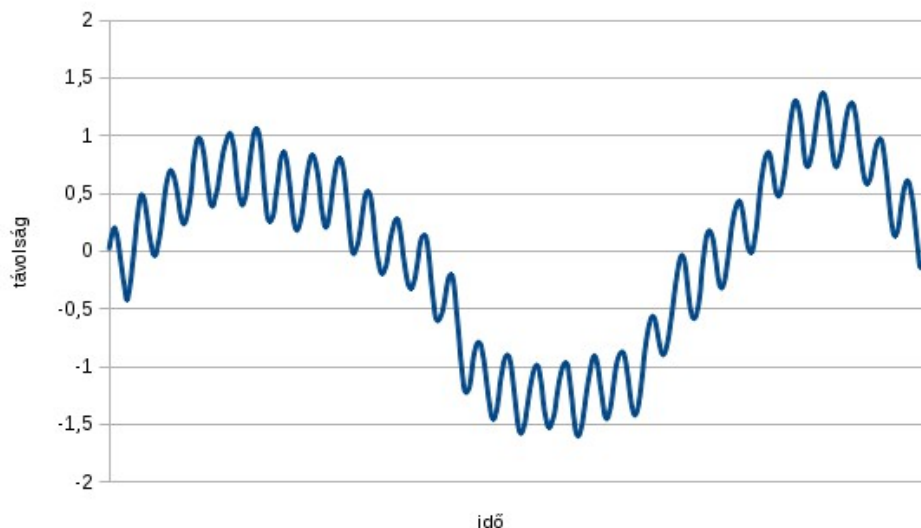
Az előző fejezetben csak álló eszköz szenzoradataiból számoltunk, és tapasztaltuk, hogy kell a hibára számítanunk. Kérdés, hogy mozgó eszközknél van-e esélyünk nyomonkövetni a pozíciót.

Az egyszerűség kedvéért az első teszteket 1 dimenziós mozgással végeztük. Ez könnyebben ellenőrizhető, grafikonok alapján jobban átlátható mintha 2 vagy 3 dimenzióban vizsgálnánk.



3. ábra
Sebesség az idő függvényében

Két pont közötti periodikus mozgás esetén elég a két pont távolságát ismerni, ezekhez tudjuk viszonyítani a számított pontok eltérését a valósakhoz képest.



4. ábra
Távolsága az idő függvényében

Az eszköz sebessége nagyjából jónak tűnik. Ezt sebességmérő eszköz hiányában pontosan nem tudtuk ellenőrizni, a nagyságrend és a periodicitás rendben van. Pontosabban ellenőrizni a pozíciók alapján tudunk.

A 4. ábrán az elmozdulás látható, amely az egy vonalban előre-hátra mozgó eszköz középponttól való távolságát mutatja. A teszt érdekessége, hogy bár a 3 perc elteltével meglehetősen pontos pozíciót eredményezett a számítás, közben előfordultak 1 méter körüli eltérések is a valóshoz képest.

Látjuk tehát, hogy próbálkozásunk nem esélytelen, viszont a pontosság javítása fontos.

5 HIBAJAVÍTÁS NEURÁLIS HÁLÓZAT ALGORITMUS ALKALMAZÁSÁVAL

Mivel a különböző készülékek eltérő típusú szenzorokkal lehetnek felszerelve, amelyek tulajdonságai is különbözhetnek, ezért felmerül annak lehetősége, hogy a kapott adatokat olyan algoritmussal dolgozzuk fel, amely rendelkezik tanuló és hibatűrő képességgel. Ilyen tulajdonságokkal rendelkeznek a mesterséges neurális hálózatok. Vizsgáljuk meg tehát a mesterséges neurális hálózat

alkalmazhatóságát a mobil telefonok szenzoraitól érkező adatokból való pozíciószámítás pontosítására.

Vegyünk egy egyszerű neurális hálózat algoritmust, amely megfelel a célra. Ilyen lehet például a hiba-visszaterjesztéses (error backpropagation) neurális hálózat.

Ezen algoritmus a következőképpen használható:

1. Minta adatokat kell előállítani, amelyek tartalmazzák a bemenő adatokat és a kimenő értékeket.
2. A minta adatokat három részre osztjuk, az első rész a tanuló adatokat tartalmazza, a második rész a validációs adatokat, a harmadik pedig a teszt adatokat. A tanuló adatokon tanítjuk a neurális hálózatot, a validációs adatokon ellenőrizzük, hogy nehogy túltanulás következzen be, a teszt adatokon pedig végül majd mérjük az alkalmazás hasznosságát.
3. A kezdeti tanulás után használatba lehet venni az alkalmazást. A felhasználás során keletkező adatokat természetesen szintén felhasználhatjuk majd az algoritmus tanításához (lásd 2. pont).

A bemenő adatok a mi feladatunkban a szenzorok által adott értékek, a kimenő adatok pedig az elmozdulás értékek. Az elmozdulás értékeket a tanításhoz a referencia pozíciók fogják szolgáltatni, amelyeket NFC tag vagy QR kód leolvasásával kap meg az alkalmazás. A szenzoroktól sok adatot kapunk, az elegendően sok referencia adat biztosítása a tanuló algoritmus számára a következő megoldandó feladat.

6 ADATOK ELŐÁLLÍTÁSA A NEURÁLIS HÁLÓZAT SZÁMÁRA

Induljunk ki abból, hogy a szenzoradatok elegendően sűrűn állnak rendelkezésre. Ez telefononként eltérő és a kiolvasási módszertől is függ, tipikusan 10Hz és 100Hz között változhat. A mi tesztjeinknél 15-20 Hz a tipikus adatgyakoriság. A referencia pontoktól viszont csak sokkal ritkább időközönként kapunk valós pozíció adatot. A köztes időpillanatokra valamilyen módon a referencia pontoktól kapott adatokból képezhetünk interpolációt. Ezt a legegyszerűbben úgy tehetjük meg, hogy a két referencia pont közötti adatsorral egy olyan transzformációt végzünk, amely az eredetileg számított végpontot a referencia pontba viszi lineáris leképezéssel, forgatással és szorzással.

Legyen a P_i pont 2 dimenzióban (X_i, Y_i) koordinátákkal adott.

P_i a kiinduló pont, vagy az eddigi utolsó referencia pont, P_j a következő referencia pont, P_j' pedig a P_j pontban számított utolsó pozíció. $i < k \leq j$ esetén P_k' a két referencia pont között egy számított pozíció. Az egyszerűbb formulák érdekében a P_j és P_k' pontokat a P_i ponthoz viszonyítva vesszük. Így a szükséges transzformáció lépései:

Kiszámítjuk a P_j' pont P_j pontba való transzformációjához szükséges forgatási szöget és szorzó tényezőt.

Ehhez először a forgatási szög:

$$\varphi = \arctan(Y_j/X_j) - \arctan(Y_j'/X_j')$$

A P_j' pontot elforgatva ezzel a szöggel kapjuk a P_j'' pontot:

$$\begin{bmatrix} X_j'' \\ Y_j'' \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} * \begin{bmatrix} X_j' \\ Y_j' \end{bmatrix}$$

Innen a skalár szorzó tényező kiszámításához mindegy, hogy az x vagy az y koordinátákat vesszük alapul:

$$l = X_j/X_j'' = Y_j/Y_j''$$

Ezzel szöggel forgatjuk és a szorzó tényezővel korrigálva kapjuk $i < k \leq j$ -re minden P_k pont koordinátáit a P_k' pontból.

Az eljárást minden referencia pont pontos pozíció adatainak beolvasása után végrehajtjuk.

A fentieket végrehajtva minden pillanathoz rendelkezésre áll egy javított pozíció. Ezután a

javított pozíció adatoknak megfelelően visszamódosítjuk a sebesség adatokat is, majd miután ezek megvannak, a gyorsulás adatokat is korrigáljuk.

Így a neurális hálózat algoritmus tanításához elegendő adatunk lesz, mégpedig a korrigált gyorsulás adatok, amelyek annál pontosabbak lesznek, minél gyakrabban olvastunk be referencia pont koordinátákat.

Az input adataink természetesen az eredeti gyorsulás adatok, kiegészítve tetszés szerint egyéb adatokkal, pl. orientációval.

A megoldás tervezett működése a számítások és a neurális hálózat algoritmus együttes felhasználásával:

1. Adatgyűjtés a szenzorokról. A megfelelő szenzorok kiválasztásával, a gyorsulás, gravitáció, elfordulás, orientáció adatok párhuzamosan egy rekordba rögzítése.
2. A szenzorokról gyűjtött adatokból a sebesség és a pozíció adatok kiszámítása. Ezen számított adatokkal bővítjük az adatrekordokat.
3. Referencia pont rögzítése esetén a fentebb leírt algoritmus szerint az előző referencia pont óta keletkezett adatok korrigálása. Ezekkel a korrigált adatokkal kiegészítjük az adatrekordokat.
4. Az előző pontokban leírt módon előállított adatsorokra neurális hálózat algoritmus tanítása.

7 NEURÁLIS HÁLÓZAT EREDMÉNYEK

Különböző neurális hálózat algoritmusok különböző paraméterekkel futtatva eltérő eredményeket adnak. Egy adott algoritmus paramétereinek beállításától függ az eredmény. A tanulást sok más paraméter mellett befolyásolja a tanuláshoz felhasznált adatmennyiség, a rejtett rétegbeli neuronok száma, valamint a tanulási ciklusok száma. Ezek függvényében vizsgáltuk az átlagos hibanégyzet (MSE) alakulását, amit már a teszt adathalmazon számolunk.

A neurális hálózat célja a gyorsulásmérő, a giroszkóp és az irányszenzor adataiból a gyorsulásmérő számára a referenciaadatok segítségével számított elvárt gyorsulás adatok megtanulása volt. A következő táblázatban néhány példát mutatunk be a teszt eredményekből.

Megjegyezzük, hogy az első eredményben szereplő MSE érték közel van ahhoz, amennyit a gyorsulásadatok és az átlaguk közti különbségből kapunk.

Neurális hálózat teszt eredmények

1. táblázat

Tanulási adat	Rejtett neuronok	Max epoch	MSE
100	2	100	0,02736
200	5	100	0,00556
200	5	500	0,00478
500	10	500	0,00513
500	10	1000	0,00495
500	20	100	0,00620
500	20	500	0,00535
500	20	1000	0,00495
1000	10	100	0,00447
1000	10	500	0,00453
1000	10	1000	0,00468
1000	20	100	0,00455
1000	20	500	0,00444
1000	20	1000	0,00451

Mint láthatjuk, a tanulási adatok nagyobb mennyisége hozzájárul a jobb eredményhez, azaz a kisebb hibához a teszt adathalmazon. Ugyanígy a köztes rétegbeli neuronok számától és a tanulás hosszától is függ az algoritmus hatékonysága. Ezek növelése viszont egy szint után már nem okoz javulást, ha nem tudunk több minta adatot biztosítani a tanuláshoz.

8 TOVÁBBI LEHETŐSÉGEK

Jelen munkánkban eddig a technikai lehetőségekkel és a neurális hálózat alkalmazásával foglalkoztunk, de mint említettük, más zajcsökkentő eljárások is léteznek, lásd [2] , melyek akár a neurális hálózattal kombinálva is használhatóak [3]. A következőkben célunk ezen megoldásokat együtt alkalmazó programok fejlesztése.

Programozható robotra helyezve a mobiltelefont nagyobb adatmennyiséget tudunk gyűjteni a szenzorokról úgy, hogy a referencia pontok adatai is rendelkezésünkre álljanak.

A mobil telefonok egyre gyakrabban tartalmazznak OpenCL interfészen keresztül programozható GPU csipeket. Ezek nagyfokú párhuzamosítást tesznek lehetővé, miáltal a telefon akár nagy sűrűséggel előálló szenzor adatok folyamatos feldolgozására is képes lehet.

9 HIVATKOZÁSOK

- [1] <http://www.in-location-alliance.com/>
- [2] Shay Reuveny , Maayan Zadik:3D Motion tracking with Gyroscope and Accelerometer
- [3] Mohammad Gorji Sefidmazgi1, Mohammad Farrokhi:Online Calibration of Inertial Sensors Using Kalman Filters and Artificial Neural Networks