|  |  |
| --- | --- |
|  |  |
|  | |

**Technická zpráva**

**Analýza a interpretace biologických dat [MABD]**

**Projekt č. 10: Počítání kroků pomocí přizpůsobené filtrace**

**Zadání:**   
Pomocí akcelerometru chytrého telefonu s aplikací *Sense-it*nasnímejte 5 signálů chůze a zaznamenejte si referenci (ručně spočítané kroky). Detekujte kroky s využitím *přizpůsobené filtrace*a spočítejte je (automaticky). Porovnejte přesnost vašeho stanovení počtu kroků s referencí.

**Bc. Veronika Pošustová, Bc. Jana Schwarzerová**

**Brno 2. 12. 2018**

# Teoretický rozbor

# Počítání kroků se již dlouho používá jako metoda měřící vzdálenost. Kroky mají několik výhod v rámci posuzování fyzických činností, například jsou intuitivní, jsou snadno měřitelné atd. Počínaje polovinou devadesátých let, vědci se začali zajímat o kvantifikaci kroků za den z hlediska ambulantní fyzické aktivity. [1] Studie [2] pojednává o detekci chodidel, protože procházka jako aktivita je jedním z parametrů pro hodnocení zdraví pacienta. Například celková doba procházky během dne umožňuje rychlé posouzení, zda je pacient mobilní, či nikoliv. [2] Od roku 2010 se využívá aktivita založená na akcelerometru. [1] Akcelerometr je zařízení, které měří zrychlení. Akcelerometry integrované v dnešních chytrých mobilních telefonech bývají použity k odhadu vzdálenosti ušlých kroků pomocí zrychlení podél os x, y a z. [3]

# Pro naměření dat pomocí akcelerometru byla v této práci použita aplikace *Sense-it*. Tato aplikace umožňuje přístup ke všem senzorům v rámci chytrých telefonů. Dále umožňuje nastavit parametry potřebné pro různé snímání jako je například vzorkovací frekvence. Aplikace *Sense-it* vytvoří protokol dat, vizualizuje je v grafech a exportuje do souborů \*.csv, což umožňuje následně naměřená data dále zpracovat v jiných programech. Čehož jsme v této práci využili a naměřená data pomocí akcelerometru byla dále zpracována v programovacím prostředí Matlab2018a, kde jsme pomocí přizpůsobené filtrace provedli detekci jednotlivých kroků.

# Přizpůsobená filtrace (také označovaná jako adaptivní filtrace) spočívá v kontinuální změně parametrů podle předem stanoveného algoritmu s využitím jak informace ze vstupu tak i pomocné informace přicházející většinou z výstupu celého systému. [4] Tato základní myšlenka byla aplikována v této práci za účelem detekce jednotlivých kroků. Pomocí akcelerometru a s použitím aplikace *Sense-it* jsme získali signál, kde je informace o daném zrychlení [ms*-2*] v závislosti na čase [*s*]. Podle teoretického předpokladu, že zrychlení bude největší právě při vykonávaném kroku, jsme aplikovali okno, díky kterému jsme zprůměrovali všechny naměřené kmity a tím jsme dostali kmit, který detekuje jeden krok. Při následném použití korelace mezi vlnou detekující jeden krok a celým naměřeným signálem, jsme získali informaci, která udává výskyt jednotlivých kroků v měřeném signálu.

# Popis metody řešení

# Pomocí akcelerometru chytrého telefonu jsme změřili celkem 10 signálů, z toho byly 4 změřeny s mobilním telefonem v ruce, 4 s mobilním telefonem v kapse a nakonec 2 signály pro interpretaci algoritmu na rozsáhlejší data – zvolili jsme 50 kroků pro jedno měření a 50 kroků s delšími pauzami pro druhé měření. Jako referenci jsme vždy spočítali kroky manuálně.

# Vytvořili jsme algoritmus pro automatické počítání kroků. Skript *Projekt.m* obsahuje algoritmus pro výpočet aplikovaný na dvě měření, konkrétně měření, když objekt držel mobil v ruce a šel a poté měření, když objekt měl mobil v kapse a šel. Nejprve jsou načtena data (x, y, z osy – směry pohybu) příkazem *load*. Poté je spočítán ze všech těchto směrů celkový pohyb pomocí Euklidovské vzdálenosti a následně je odečteno tíhové zrychlení. Pro ilustraci jsme nejdříve detekovali kroky jako píky v signálu pomocí funkce *findpeaks* bez použití filtrace, s nastavenými parametry minimální vzdálenosti a minimální výšky píků. Tyto hodnoty byly nastaveny empiricky.

# Poté jsme vytvořili referenční vlnku pro aplikaci přizpůsobené filtrace, viz Obr. 1, konkrétně jde o funkci *jeden\_krok.m*. Vlnku jsme spočítali jako průměr jednotlivých úseků (píků) v signálu. Tyto dva signály jsme následně korelovali. Korelaci je možné provést jako obrácenou konvoluci, takže jsme získanou vlnku otočili v čase a aplikovali konvoluci na původní signál a referenční vlnku pomocí příkazu *conv*. Ve vyfiltrovaném signálu jsme opět pomocí funkce *findpeaks* našli píky pomocí obdobně nastavených parametrů, jako v předchozím signálu. Korelaci a následnou detekci píku, provádí funkce *korelace\_detekce.m.* Závěrem jsme statisticky vyhodnotili úspěšnost nalezených kroků jednoduchým podílem počtu kroků, vypočítaných manuálně, a počtu kroků, nalezených automaticky. Výsledky jsme převedli na procenta a zapsali do předem vytvořené tabulky (výsledky viz kapitola Dosažené výsledky a jejich interpretace).

C:\Users\Jana\Desktop\Projekt\Jeden_krok.emf  
**Obr. 1: Příklad referenční vlnky**

# Celý algoritmus pro počítání kroků jsme ještě aplikovali na delší signály, abychom dokázali zhodnotit správnost řešení pro obecné využití. Na konci se ještě ve skriptu *Projekt.m* nachází vykreslení a tedy porovnání již zmiňovaných dvou měření a tedy s mobilním telefonem v ruce a s mobilním telefonem v kapse.I tyto výsledky jsou diskutovány v následující kapitole.

# Dosažené výsledky a jejich interpretace

# Vyhodnocení všech měření s mobilním telefonem v ruce a s mobilním telefonem v kapse jsou v následujících tabulkách. Běžně se setkáváme s krokoměry, které detekují více kroků, než člověk skutečně ujde. Takové výsledky vidíme, viz Tabulka 1, při měření v ruce. Naopak při měření v kapse jsme detekovali méně kroků, než byla referenční hodnota. Srovnání je zde i graficky znázorněné, viz Obr. 2.

# Vidíme rozdíly v obou signálech – při měření s mobilním telefonem v ruce je po přizpůsobené filtraci signál zbaven rušivých elementů a výška píků je přibližně stejně velká. Díky tomu můžeme lépe posoudit, jestli se jedná o krok nebo ne. Při měření s mobilem v kapse nedetekujeme jen samostatné kroky, ale i jiné pohyby. Navíc signály kroků mohou být výrazně slabší, a proto nejsme schopni detekovat je všechny.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | bez filtrace | 1. měření | 2. měření | 3. měření | 4. měření |
| manuálně spočítané kroky | 20 | 20 | 10 | 10 | 10 |
| korky spočítané automaticky | 21 | 22 | 12 | 11 | 10 |
| **vyhodnocení [%]** | **105** | **110** | **120** | **110** | **100** |

Tabulka 1: Vyhodnocení měření s mobilním telefonem v ruce

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 5. měření | 6. měření | 7. měření | 8. měření |
| manuálně spočítané kroky | 10 | 10 | 10 | 10 |
| korky spočítané automaticky | 6 | 9 | 7 | 10 |
| **vyhodnocení [%]** | **60** | **90** | **70** | **100** |

Tabulka 2: Vyhodnocení měření s mobilním telefonem v kapse

# Abychom mohli říct, že algoritmus je funkční, aplikovali jsme ho na rozsáhlejší data, viz skript *Aplikace\_rozsahlejsich\_dat.m*. Měřili jsme s mobilním telefonem v ruce. Nejdříve jsme naměřili signál 50 kroků, následně 50 kroků s delšími pauzami a sledovali jsme, jak program vyhodnotí zastavení vždy po 10 krocích. Na obrázku, viz Obr. 4, jsou výsledná data. Výsledky jsou pozitivní, protože z 50 skutečně naměřených kroků, bylo automaticky spočítáno 51 kroků bez zastavování a 54 kroků se zastavováním.

C:\Users\Jana\Desktop\Projekt\Srovnani_mobil_v_ruce_mobil_v_kapse.emf  
**Obr. 3: Srovnání dvou různých měření**

C:\Users\Jana\Desktop\Projekt\Aplikace_rozsahlejsich_dat.emf

**Obr. 4: Měření s mobilním telefonem v ruce - 50 kroků**

# Závěr

# Pomocí akcelerometru chytrého telefonu jsme změřili celkem 10 signálů, z toho byly 4 změřeny s telefonem v ruce, 4 s telefonem v kapse a 2 signály pro interpretaci algoritmu na rozsáhlejší data (50 kroků). Jako referenci jsme vždy spočítali kroky ručně. Následně jsme vytvořili algoritmus pro automatické počítání, statisticky jsme vyhodnotili úspěšnost tohoto počítání a srovnali jsme úspěšnost v závislosti na umístění mobilního telefonu při měření. Nakonec jsme algoritmus aplikovali na signál získaný z měření 50 kroků.

# Zhodnotili jsme výsledky, které byly relativně přesné, a proto můžeme konstatovat, že jsme kroky detekovali úspěšně. Relativně přesné myslíme tím, že se výsledky liší většinou o jeden krok. Pokud se zamyslíme, jak měření probíhá tak osoba, která jde konkrétní počet například deseti kroků, může započítat poslední krok jako přiložení nohy k noze, či jako nový nášlap. Dokonce stejná osoba v různých měřeních za sebou může počítat jinak. V tomto subjektivním názoru počítání kroků je zřejmé, že se tedy budou detekce lišit s chybou přičtení či odečtení jednoho kroku.

# Zjistili jsme, že detekce je nejpřesnější při měření s mobilem v ruce, protože tak získáváme ‘čistý’ signál – píky jsou jen na pozicích skutečných kroků. Při dodatečné aplikace algoritmu na rozsáhlejší signály, konkrétně z měření 50 kroků se zastavením a bez zastavení jsme zjistili, že algoritmus díky detekci pomocí přizpůsobené filtrace nevadí, že kroky nejsou kontinuálně za sebou a pauzy způsobené zastavením nedetekuje, což je velkou výhodou detekce pomocí přizpůsobené filtrace.

# Zdroje:

1. BASSETT, David R., Lindsay P. TOTH, Samuel R. LAMUNION a Scott E. CROUTER. Step Counting: A Review of Measurement Considerations and Health-Related Applications. *Sports Med*. 2017, , 47: 1303–1315. DOI: 10.1007/s40279-016-0663-1.
2. BARRALON, Pierre, Nicolas VUILLERME a Norbert NOURY. *Walk Detection With a Kinematic Sensor: Frequency and Wavelet Comparison*. Faculte de Medecine 38706 LA TRONCHE, France, 2006, 4.
3. BYLEMANS, Inge, Maarten WEYN a Martin KLEPAL. Mobile Phone-based Displacement Estimation for Opportunistic Localisation Systems. *Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*.
4. *Elektrorevue: Adaptivní filtrace zašuměných řečových signálů* [online]. Ústav Telekomunikací, VUT v Brně, Fakulta elektrotechniky a informatiky, 2002 [cit. 2018-11-27]. Dostupné z: <http://www.elektrorevue.cz/clanky/02063/index.html>