

**Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky a kvantitativních metod**

Senzory mobilních zařízení

Využití senzorů mobilního zařízení v dopravě

Diplomová práce

Autor: Bc. Tomáš Pochobradský
Studijní obor: Aplikovaná informatika, prezenční forma

Vedoucí práce: doc. Mgr. Tomáš Kozel, Ph.D.

Hradec Králové

26. dubna 2015

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 26.4.2015

Bc. Tomáš Pochobradský

Poděkování

Děkuji mému vedoucímu doc. Mgr. Tomáši Kozlovi, Ph.D. za metodickou, pedagogickou a odbornou pomoc. Jeho cenné rady mi velmi pomohly při zpracování této diplomové práce. Poděkování patří i mým rodičům a známým, kteří mě podporovali po celou dobu studia.

Anotace

Cílem práce je analyzovat možnosti využití senzorů mobilního zařízení v dopravě. Je zde navrženo a následně implementováno řešení založené na nalezených možnostech. Aplikace, která je součástí této práce, analyzuje za pomoci senzorů povrch a kvalitu vozovky. Začátek diplomové práce je zaměřen na typy senzorů využívaných v dnešních mobilních zařízeních. Dále je provedena analýza využití těchto senzorů v dopravě. V následujících kapitolách je podrobněji rozebrán Android Sensor Framework. Po seznámení s tímto rozhraním práce rozebírá možnosti kalibrací, které jsou v aplikaci přítomny. Na základě předchozího poznání je proveden samotný návrh a implementace aplikace s názvem „Road Checker“. Hlavním cílem této práce je proniknutí do základních principů analýzy a vývoje aplikace pro mobilní zařízení. Velký důraz je kladen především na práci se senzory. Práce se snaží obsáhnout hlavní aspekty, které se této problematiky týkají.

Annotation

Sensors of mobile devices

The objective of the diploma thesis is to analyze the possibilities of using mobile equipment sensors in transport. The solution based on the possibilities found is designed and subsequently implemented. The application, which is a part of the thesis, analyses the surface and quality of the road with the help of sensors. The beginning of the thesis is focused on types of sensors used in today's mobile devices. Further, an analysis of the use of these sensors is performed. In the following chapters, the Android Sensor Framework is examined in detail. Having been acquainted with this interface, the thesis analyses the calibration options, which are present in the application. On the basis of previous knowledge, the design and implementation of the application called "RoadChecker" is performed. The main objective of this work is to penetrate the basic principles of the analysis and development of applications for mobile devices. The greatest emphasis is given to work with sensors. The thesis aims to cover the main aspects relating to the issue.

Obsah

| | |
|---|----|
| 1. Úvod | 1 |
| 2. Sensory mobilních zařízení | 3 |
| 2.1. Aktuální možnosti..... | 4 |
| Akcelerometr | 4 |
| Barometr | 6 |
| Gyroskop | 7 |
| Proximity senzor | 8 |
| Magnetometr | 9 |
| Senzor okolního osvětlení | 11 |
| Biometrické senzory | 12 |
| 2.2. Trendy..... | 14 |
| 3. Analýza využitelnosti senzorů v dopravě | 17 |
| 3.1. Asistenční systémy v automobilech | 17 |
| Systémy pro sledování a řízení dopravní situace..... | 18 |
| Systémy pro sledování a řízení stavu vozidla | 21 |
| Systémy pro sledování stavu řidiče | 22 |
| 3.2. Využitelnost senzoru mobilního zařízení..... | 23 |
| Využití akcelerometru..... | 23 |
| Využití ostatních senzorů | 24 |
| 3.3. Možnost propojení přenosného zařízení s vozidlem..... | 25 |
| 4. Softwarová rozhraní pro přístup k senzorům v Androidu..... | 26 |
| 4.1. Sensor manager | 26 |
| Konstanty | 27 |
| Pomocné metody..... | 29 |

| | | |
|------|---|----|
| 4.2. | Senzory a jejich dostupnost..... | 31 |
| 4.3. | Souřadnicový systém senzorů | 32 |
| 4.4. | Senzory a vývoj aplikace | 33 |
| | Identifikování senzorů | 33 |
| | Monitorování sensorových událostí..... | 34 |
| 5. | Přesnost a možnosti kalibrace senzorů | 36 |
| 5.1. | Zjištění aktuální přesnosti..... | 36 |
| 5.2. | Kalibrace v Android Sensor Framework | 37 |
| | Sensor magnetického pole | 37 |
| | Gyroskop | 37 |
| | Vektor rotace | 38 |
| | Manuální kalibrace..... | 39 |
| 5.3. | Kalibrace dat senzorů při vývoji aplikace | 40 |
| | Kalibrace pozice zařízení | 40 |
| | Low-pass filtr..... | 40 |
| 6. | Návrh a implementace aplikace | 42 |
| 6.1. | Funkční požadavky | 42 |
| 6.2. | Non-funkční požadavky | 44 |
| 6.3. | Návrhové modely..... | 45 |
| | Use case..... | 45 |
| | Model tříd..... | 46 |
| 6.4. | Získání dat | 47 |
| | Získání dat ze senzorů..... | 47 |
| | Získání dat z GPS | 48 |
| 6.5. | Výpočty v aplikaci..... | 48 |
| | Vyrovnání náklonu zařízení..... | 48 |

| | |
|--|----|
| Low-pass filtr | 51 |
| Výpočet průměrných hodnot otřesů | 52 |
| Výpočet výchylek | 55 |
| 6.6. Zobrazení výsledů | 58 |
| Zobrazení dat ze senzorů..... | 58 |
| Grafické znázornění výsledků výpočtů | 59 |
| Zobrazení výsledku na mapě..... | 61 |
| 7. Shrnutí výsledků a závěr | 63 |
| Seznam použité literatury a zdrojů..... | 65 |
| Seznam obrázků | 69 |
| Seznam tabulek..... | 70 |
| Seznam kódů..... | 71 |
| Příloha A – DVD..... | 72 |

1. Úvod

Operační systém android je platformou určenou převážně pro použití na mobilních zařízeních. Poskytuje vývojářům velkou míru svobody, která je doplněna o výbornou podporu ze strany společnosti Google. Tato podpora zahrnuje veliké množství dokumentace a přehledných tutoriálů, které byly využity pro tvorbu této práce.

Tato diplomová práce se ze začátku zabývá aktuálními možnostmi, které jsou při současné technologii dostupné. Jsou zde popsány různé skupiny a typy senzorů. Protože je senzorů velké množství, je výběr zúžen na sensory, které jsou využity v mobilních zařízeních. Následně je v této kapitole uvedena úvaha o vývoji, kterým se toto odvětví technologii bude odebírat.

V další části je provedena analýza využitelnosti senzorů v dopravě. Moderní auta jsou doslova prošpikována sensory. Většina jich je na začátku kapitoly představena na příkladu adaptivních tempomatů a principů, na kterých fungují. Z této problematiky se posléze vychází při analýze využití mobilních senzorů v automobilové dopravě, přičemž je kladen důraz na akcelerometr. Tato analýza je doplněna o několik příkladů.

Následující kapitola diplomové práce se zabývá softwarovým rozhraním pro přístup k senzorům v operačním systému Android. Podrobně jsou zde popsány konstanty a pomocné metody, které jsou přístupné přes `SensorManager`. Po té je zde uvedeno, jak identifikovat sensory, které jsou dostupné na daném zařízení, a jak z nich získat data. Důležitou částí této kapitoly je popis souřadnicových systémů, které jsou následně využity při implementaci kalibrace zařízení.

Kapitola zabývající se přesností a kalibrací senzorů mobilního zařízení vychází z předešlé kapitoly rozebírající `Android Sensor Framework`. Některé prvky tohoto rozhraní jsou zde využity k představení různých kalibrací, které jsou v něm již implementovány. Po té jsou kalibrace teoreticky rozebrány a prakticky využity v následující kapitole.

V pořadí pátá kapitola pojednává o návrhu a implementaci praktické části diplomové práce. Tato část je logicky rozdělena do několika celků obsahujících důležité

části návrhu a implementace aplikace. Na začátku jsou popsány jak funkční, tak nefunkční typy požadavků. Objektový návrh je doplněn o několik návrhových vzorů, které jsou reprezentovány diagramy. Podkapitola týkající se implementace obsahuje popis řešení funkcionalit, které byly řešeny při implementaci aplikace. Není zde popsán podrobný vývoj aplikace pro mobilní zařízení s OS Android krok po kroku. Podrobná implementace následujících problémů se nachází v příloze A.

Cílem této práce by mělo být seznámení se s prvky vývoje aplikace využívající senzory pro operační systém Android. Tato aplikace by měla analyzovat pomocí akcelerometru nerovnosti na silnicích. Tyto výsledky by bylo vhodné posléze reprezentovat.

2. Senzory mobilních zařízení

„Moderní telefony jsou doslova prošpikovány nejrůznějšími senzory. Bez některých se v moderních telefonech neobejdeme. Jiné senzory nabízí něco navíc, co sice není úplně nezbytné, ale nabízí nám to vyšší komfort při používání telefonu.“ (1)

„Signál ze sensorů je filtrovaný a zesílený. Analog-digitálním konvertorem (ADC – analogue to digital converter) je převeden do digitální podoby. Některé jednoduché zpracování digitálního signálu je provedeno senzorem dříve, než je signál modulován a připraven k přenosu.“ – přeloženo z (2)

Výpočetní technologie v dnešní době prochází obrovským vývojem. To platí především u technologií mobilních. Dochází k velkému rozšíření přenosných zařízení. Tato zařízení se mohou lišit jak po stránce hardwarové, konstrukční, tak po stránce softwarové. Z pohledu hardwaru jsou zařízení čím dál více výkonnější a mají čím dál větší výdrž. Co se týče konstrukce, jsou zde zastoupeny především tři typy zařízení, tedy smartphone, tablet nebo notebook. V současné době se však tyto typy začínají prolínat. Není těžké narazit na smartphone o velikosti tabletu, popřípadě na notebook, který se po odejmutí klávesnice stává tabletem. I přes rozmanitost dnešního trhu mají tato zařízení něco společného. Tím, co je může spojovat, jsou právě senzory.

U levnějších modelů přenosných zařízení jsou zastoupeny pouze senzory základní. Ve vyšších třídách jich je k nalezení nepřeberné množství a stávají se prostředkem, kterým výrobci dokáží svá zařízení odlišit od ostatních, tedy od konkurence. Díky tomu jde vývoj sensorů neustále dopředu a stále více sensorů se dostává i do levnějších zařízení.

Senzory slouží primárně ke zvýšení komfortu při využívání zařízení, které je jimi osazeno. Toto zařízení tak díky nim dokáže automaticky reagovat na prostředí, ve kterém se právě nachází. V základu sensorů využívá operační systém. Ten pomocí nich provádí akce, které se pro mnoho uživatelů již staly běžnými. Lze mezi ně zařadit například změna nasvícení displeje nebo změna jeho orientace. Data ze sensorů jsou dále zprostředkována aplikacím, které je dokáží využít ke svým specifickým účelům.

Senzory můžeme rozdělit celkem do tří hlavních kategorií. Pohybové senzory dokáží měřit zrychlení nebo pohyb kolem tří os. Senzory pro určení polohy dokáží určit fyzickou polohu zařízení. Dále existují senzory, které dokáží zjišťovat informace o prostředí. Celé rozdělení je znázorněno níže, přičemž v následujících kapitolách budou rozebírány typy senzorů podrobněji (3).

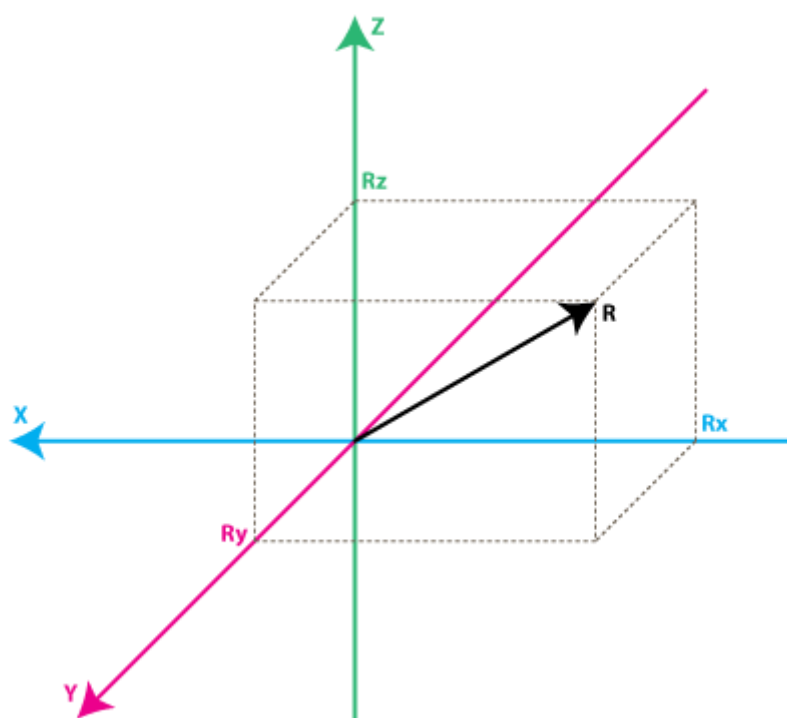
- Pohybové senzory
 - Akcelerometr
 - Gyroskop
 - Senzory rotace
- Senzory pro určování polohy
 - Sensory orientace
 - Magnetometry
- Senzory pro získání informací o prostředí
 - Barometry
 - Fotometry
 - Teploměry

2.1. Aktuální možnosti

Akcelerometr

Akcelerometr je jeden ze základních pohybových senzorů, který měří velikost zrychlení, které na něj působí. V akcelerometru je velice důležitým faktorem gravitační síla Země. Tato síla působí na každý předmět na povrchu planety. Při využití dat z tohoto senzoru se tedy musí brát toto v potaz. Používá se ke změření zrychlení, popřípadě ke změření síly vibrací zařízení, které senzor obsahuje.

„Akcelometry v mobilních telefonech fungují na principu piezoelektrického jevu, což je schopnost krystalu generovat elektrické napětí při jeho deformování. Toto elektrické napětí je přitom dobře měřitelné, protože hmota krystalu zůstává stejná a vytvořený elektrický náboje je tedy úměrný síle, která ho vygenerovala a také zrychlení.“ (1)



Obr. 1: Model souřadnicového systému akcelerometru (18)

Jak je uvedeno na obrázku (Obr. 1) akcelerometr využívá tříosý souřadnicový systém. Data, která se z něj dají získat, mají tedy tvar tříložkového vektoru, přičemž každá složka obsahuje velikost zrychlení, popřípadě gravitační, které v daném směru působí. Pro výpočet celkového zrychlení v určitém směru je tedy využít následující výpočet.

$$R_x^2 + R_y^2 + R_z^2 = R^2$$

Akcelerometr je velmi citlivý senzor. Využívá se především v automobilovém průmyslu, kde je součástí různých bezpečnostních systémů. Když se tento senzor dostatečně zmenšil a vyvinuly se nové technologie, začal se využívat v mobilních zařízeních. Zde byl původně využit jako složka varování zařízení, ve kterém byl využit, před pádem. Jako příklad je na serveru mobilnet.cz (1) uveden pád notebooku, který ho je díky akcelerometru schopen zaznamenat, a zaparkováním hlav předejít zničení harddisku. Z rozvojem vývoje aplikací na mobilních zařízeních, se začaly akcelerometry rozšiřovat. Pomocí něj jsou ovládány některé hry. Výborným příkladem jeho využití je

uživatelské prostředí v operačním systému iOS 8, kde je celý interface natáčen podle naklonění zařízení.

Barometr

Barometr je dalším typem senzoru pomáhající zařízení zaměřit jeho nadmořskou výšku. Barometr je vlastně měřič tlaku, který v případě přenosných zařízení, měří tlak atmosférický. Tento senzor má obecně mnoho využití. Uplatnění nachází například v letectví nebo geodézii. V současné době však svou hlavní úlohu sehrává při předpovídání počasí.

Barometr byl podle (1) vynalezen v 17. století italským fyzikem a matematikem Torricellim. Dalším vynálezcem, který se zabýval barometrem, byl Francouz Lucien Vidi. Jeho barometr neobsahuje kapalinu a je označován jako aneroid. Tento barometr již neobsahoval jedovatou rtuť. Další jeho výhodou je bezesporu menší velikost. Nejmodernější barometr je barometr elektronický. Tento barometr lze označit za tlakový elektronický senzor. Podle serveru mobilnet.cz tento senzor obsahuje analogově-digitální převodník. Je zřejmé již podle názvu, že se stará o převod analogových dat senzoru na data digitální. Postupem času se stal senzor dostatečně malý na to, aby našel své místo v mobilních zařízeních.

„Podle tehdejšího vyjádření Googlu byl jeho účel zcela jiný, ale také zajímavý. Pomáhal totiž systému GPS upřesnit nadmořskou výšku, kde se právě nacházíte, a tím i zrychlit zaměření a zafixování polohy.“ (1)

Systém GPS jako takový, dokáže zaměřit polohu zařízení. Ke zjištění nadmořské šířky a délky jsou zapotřebí tři satelity. Aby bylo možné zjistit i nadmořskou výšku, je nutné, aby zařízení vyhledalo i čtvrtý satelit. To může dobu určení polohy značně prodloužit. Z toho důvodu je právě použit v mobilních zařízeních barometr. Barometr tedy pomáhá k lepší navigaci.

Podle serveru wired.com (4) však mají vědci s tímto senzorem větší plány. Jak již bylo zmíněno výše, barometr dokáže reagovat na změny atmosférického tlaku. Mimo informací o nadmořské výšce to může ale také znamenat změny počasí. Pomocí

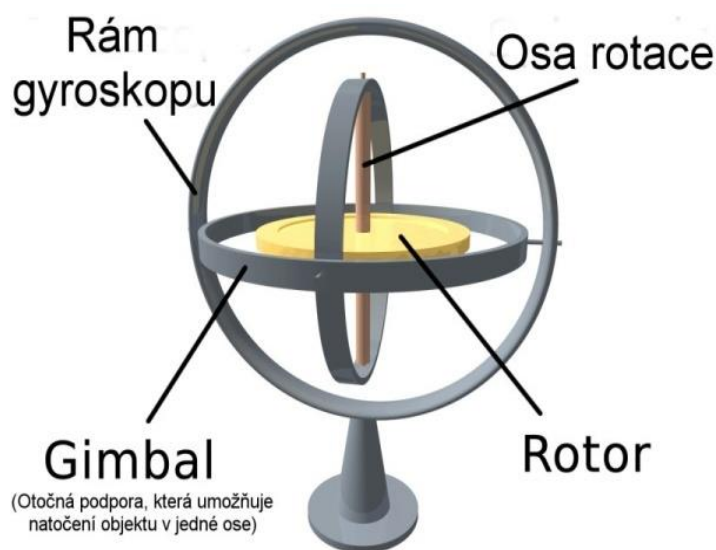
přenosných zařízení by byla získávána data o aktuálním tlaku a ta by byla zasílána na server přímo ke zpracování. Potenciál využití mobilních zařízení je v tomto případě veliký.

Gyroskop

Gyroskop je další ze senzorů, které dnes smartphony, tablety a podobná zařízení obsahují. Tato součástka umožňuje převracet obraz displeje, což je dnes velice populární a oblíbenou vlastností přenosných zařízení. Při potřebě určit natočení zařízení, velmi často pomáhá gyroskopu akcelerometr, který byl popsán výše. V praxi se právě tyto dva senzory navzájem doplňují, a díky tomu lze získat velice přesná data o aktuální poloze zařízení.

Princip

Gyroskop funguje podobně jako setrvačnick. Stabilitu v jakékoliv poloze gyroskopu zajišťuje gyroskopická akcelerace. Gyroskop v přenosném zařízení neustále rotuje. Zároveň zaznamenává změnu kapacity kondenzátorů a na základě toho pak určí současnou polohu, jak je uvedeno na serveru letemsvetemapplem.eu (5). Je důležité také zmínit, že se gyroskop otáčí v ložiscích. To je znázorněno na obrázku pod textem (Obr. 2). Díky tomu u něj lze pozorovat nepatrné tření.



Obr. 2: Znázornění gyroskopu (1)

„Většinou je setrvačnick u ložený v tzv. Cardanových závěsech, které mu umožňují otáčet se všech třech rovinách. Přitom platí, že čím větší je kolo rotoru nebo otáčky vyšší, tím víc má tendenci zachovat svou osu rotace díky momentu své setrvačnosti (gyroskopický efekt).“ (1)

Historie gyroskopu

Počátky gyroskopu sahají do 19. století. Principy fungování gyroskopu objevil počátkem 19. století německý astronom Johann Bohnenberger. Samotný gyroskop byl sestaven až daleko později. První mobilní telefon s gyroskopem byl Apple iPhone 4. (1)

„Vysoce kvalitní optické gyroskopy se přitom nespolehají pouze na samotnou setrvačnost, ale k regulaci otáček si pomáhají i optickými snímači, které pracují na principu Sagnacově jevu v kruhovém laseru nebo v cívce optického vlákna.“ (1)

Gyroskopy mají široké uplatnění, vedle přenosných zařízení se využívají také u letadel a u lodí pro navigaci.

Proximity senzor

„Proximity senzor je schopen detekovat přítomnost blízkých objektů aniž by došlo k fyzickému kontaktu.“ – přeloženo z (6)

Proximity senzor je dnes již standardní součástí mobilních telefonů, které mají dotykový displej. Proximity senzor je senzor přiblížení. Zaznamenává, zda se k mobilnímu telefonu přibližuje nějaký objekt. Využívá se zejména při hovorech. Základem moderních smartphonů je kapacitní display. Ten jak je známo, reaguje při styku s lidskou pokožkou. Když se telefon přiloží k uchu, hrozí, že omylem tváří provedeme nějakou akci (např. vypnutí hovoru). Zde přichází na řadu proximity senzor. Pokud je zaznamenáno přiblížení objektu, tváře, je displej vypnut.

Proximity senzor lze využít k více účelům nežli je jen samotné vypínání displeje mobilních telefonů. Například výrobce Samsung použil tento senzor k tomu, aby mobilní telefon vytočil telefonní číslo, kterému právě píšeme sms zprávu, pokud přiložíme ucho k telefonu během psaní této sms zprávy.

Fungování proximity senzoru

Proximity senzor se nachází nad displejem mobilního zařízení.

„Senzor vysílá elektromagnetické pole či svazek elektromagnetického záření (nejčastěji infračervené). Pomocí tohoto svazku zjišťuje změny v odraženém signálu, který se mu vrací. Pokud se objekt přiblíží do dostatečné blízkosti, odražený signál se změní a senzor vyhodnotí, že je objekt přítomen.“ (7)

V mobilních telefonech se používají senzory, které jsou v této oblasti nejlevnější, nazývají se optické senzory. Optické senzory používané u mobilních zařízení umí rozpoznat přítomnost lidského těla na vzdálenost kolem pěti centimetrů.

Proximity senzor má stejně jako ostatní senzory mimo mobilní zařízení daleko více využití. Senzory přiblížení se používají i pro zaznamenávání daleko větších objektů, například pro rozpoznání lodí či ponorek. Tyto senzory se objevily již před první světovou válkou a právě kvůli světovým válkám bylo rozšířeno a urychleno jejich uplatnění. (1) Senzory zachycující přiblížení dnes nalezne v celé škále objektů. Například oblíbený je senzor přibližování u automobilů, kde pomáhá určit vzdálenost od objektu a pomáhá tak předcházet nehodám. Dále je senzor používán například u oken a dveří, kde hlídá, jestli jsou zavřené.

Magnetometr

Magnetometr se dá nalézt u většiny smartphonů. Pomáhá určit přesnější informace o daném kurzu, podobně jako to dělá kompas. Magnetometr ale neslouží jen k přesnějšímu určení směru, ale také ho lze využít k dálkovému řízení, či ke snímání gest. Když je tento senzor doplněn ještě o akcelerometr, dokáže pomocí dostupných dat zařízení určit své natočení oproti zemskému povrchu a to ve všech třech osách.

„Magnetometr měří ve třech osách složku lokálního magnetického pole, které je zde tvořeno součtem geomagnetického pole a také polem, které je tvořeno okolím a jeho parametry.“ (8)

Fungování magnetometru

Jak je uvedeno v předchozí citaci, magnetometr měří ve třech osách lokální magnetické pole. Toto magnetické pole je pak dáno součtem magnetického pole a geomagnetického pole. Magnetometr k tomuto měření používá princip magnetorezistence¹ (1). Ta funguje tak, že u zmagnetizovaného plíšku je měřen odpor, který za daného proudu má při nulovém vlivu externího magnetického pole. Magnetometry rozdělujeme podle (1) do dvou základních skupin.

- Magnetometry skalární
 - Měřena intenzita magnetického pole
- Magnetometry vektorové
 - Měřena intenzita a směr magnetického pole

Historie

První magnetometr byl vyroben německým matematikem a fyzikem Carl Friedrich Gauss (1777 - 1855) v roce 1833. Dnes jsou magnetometry široce využívány pro měření zemského magnetického pole a různých magnetických anomálií. Používají se například pro výzkum polární záře nebo při geologickém výzkumu a archeologii, protože dokáží odhalit i skryté neobvyklé útvary v hloubce desítek metrů.

V kosmickém výzkumu se trojosý magnetometr používá v umělých družicích na výzkum magnetosféry nebo tam, kde se vyžaduje, aby byly ostatní přístroje orientovány podle magnetického pole. Magnetometry se však používají i pro vojenské účely. Dokáží například detekovat ponorky spolehlivěji než sonar. To je také důvod, proč některé země jako například USA, Kanada nebo Austrálie považuje výkonné magnetometry za citlivý vojenský materiál a omezuje jejich distribuci. Teprve posledních letech došlo k dostatečné miniaturizaci a zlevnění magnetometrů, že mohly být začleněny do integrovaných obvodů. Stále častěji se tak používají jako elektronické kompasů nejen v mobilních telefonech, ale i v tabletech, ručních navigacích a další elektronice. Například, podle (9), magnetometr LIS3MDL od firmy STMicroelectronics měří jen 2 x 2 x 1mm.

¹ Schopnost materiálu změnit svou hodnotu elektrického odporu vlivem vnějšího magnetického pole (odkaz)

„Tyto malé kompas, které mají rozměry několik milimetrů, jsou obvykle založeny na dvou nebo třech polovodičových snímačích magnetického pole, které poskytují data mikroprocesoru. Ten pak z těchto dat vypočítá správnou polohu s pomocí trigonometrie. Díky tomu telefon vždy ví, kterým směrem je sever a může tak například automaticky natočit mapu, aby odpovídala vaší orientaci.“ (1)

Senzor okolního osvětlení

Hlavní úlohou light senzoru neboli senzoru okolního osvětlení je automatické nastavování úrovně jasu displeje. Light senzor vyhodnotí aktuální osvětlení okolí a dle toho pak displej přisvítí nebo naopak ztmaví.

Light senzor se snaží zajistit optimální nasvícení displeje tak, aby byl pro uživatele co nejčitelnější a svícení displeje bylo co nejméně energeticky náročné, jelikož čím více displej svítí, tím je samozřejmě spotřeba energie větší. Je zcela logické, že mobilní zařízení svůj jas přizpůsobují nejen stavu okolního světlení, ale i stavu baterie. Je-li tedy baterie slabá, displej se sám automaticky ztmaví, stane se úspornějším. Jak je uvedeno na webu beryko.cz, tak jako světelný senzor bývá nejčastěji použit fotorezistor, což je elektrická součástka pracující na základě fotoelektrického jevu. Princip tohoto senzoru je podrobněji popsán v následujících citacích.

„Světlo, které dopadá na horní vrstvu součástky, narazí na elektrony, kterým předává svou energii. Čím více světla dopadne, tím více vznikne volných elektronů a součástka bude více vodivá (bude klesat její elektrický odpor). Řídící obvod podle naměřené vodivosti určí aktuální světelné podmínky a nastaví jas.“ (7)

„Fotovodivé vrstvy mohou vznikat buď metodou tlustých vrstev a sintrací neboli spékáním práškových částic, s použitím monokrystalu nebo tenkých vrstev. Právě tenké vrstvy jsou dnes nejčastější metodou, jak vyrábět fotorezistory. Na křemíkovou nebo germaniovou destičku se napaří tenká vrstvička kovu ve tvaru hrabiček. Světlo, pak mění vodivost mezi kovovými vložkami. Použitý kov přitom ovlivňuje, jaké světlo fotorezistor detekuje. To nemusí být pouze ve viditelném spektru, ale i ultrafialové nebo infračervené. Často se při výrobě používá Sulfid kademnatý (CdS) a Sulfid olovnatý (PbS). Většina současných senzorů okolního osvětlení používá kombinaci

několika fotodiod, z nichž každá z nich je citlivá na jinou část světelného spektra a výsledek je dopočítáván. To umožňuje poměrně přesné měření úrovně světla.“ (1)

Senzor okolního osvětlení se také používá i v jiných elektronických zařízeních, například v přenosných počítačích, tabletech nebo televizorech.

Biometrické senzory

V dnešní je nutné neustále přijímat nová bezpečnostní opatření. Mobilní zařízení obsahují velké množství citlivých dat a údajů. K zabránění jejich zneužití, je třeba, aby toto zařízení obsahovalo určité bezpečnostní prvky. Jedním z těchto bezpečnostních prvků jsou biometrické senzory. Tyto senzory můžeme zaznamenat například v podobě čteček otisků prstů, v podobě zařízení, která umí rozpoznávat hlas či aktivovat činnost na základě snímání očí a mnoho další. Tyto senzory nacházejí stále větší pole uplatňování. Lze je nalézt ve velkých firmách, v bankomatech a mobilních zařízeních. Zkrátka biometrické senzory se stávají součástí moderních technologických zařízení. O tomto nárůstu svědčí i výdaje, které byly vynaloženy do těchto biometrických technologických zařízení. Podle serveru scienceworld.cz (10) v roce 2006 přesáhly 2 miliardy dolarů. Vývoj biometrických senzorů jde neustále kupředu, ať už se to týká materiálů či velikosti senzorů.

Biometrické senzory se dostaly do popředí teprve na konci dvacátého století. Během posledních let však zaznamenaly velký vzestup. Podařilo se vytvořit čipové sady, které při používání v biometrických systémech spotřebovávají méně energie a jsou méně ekonomicky nákladné.

Biometrie otisku prstu

Biometrické senzory jsou různého druhu (11).

- Snímače kapacitní
- Optoelektronické
- Teplotní
- Elektroluminiscenční
- Radiofrekvenční

- Multispektrální.

Optoelektronické biometrické snímače

Tyto snímače pracují na rozdílném odrazu světla.

„Optický snímač zachycuje digitální zobrazení otisku pomocí viditelného světla (na rozhraní plochy hranolu a přiloženého prstu). Obraz otisku se přenesse na maticový CCD detektor, je následně digitalizován a dále předán pro zpracování. Pod vrstvou, kam se přikládá prst (dotykový povrch), je vrstva fosforu, která osvětluje celou plochu prstu. Odražené světlo od povrchu prstu prochází luminoformní vrstvou k CCD maticovému detektoru, a tam se vytvoří obraz otisku (z papilárních linií se světlo odráží, z rýhy nikoliv).“ (11)

Předností optoelektronického snímače je vysoká kvalita a minimální vliv okolí. Nevýhodou je pak možné špatné vykreslení prstu, pokud je prst například znečištěn. Také může dojít k tomu, že snímaný prst se zcela nevykreslí a přimíchá se do něj předchozí, cizí, otisk prstu.

Kapacitní biometrické snímače

Tento snímač je založen na digitálním snímání. Využívá rozdílu v kapacitě desky snímače, který obsahuje velké množství elektrod, a povrchu prstu.

Výhodou kapacitního biometrického snímače je dozajista malý rozměr a také vysoká kvalita. Bohužel tento typ snímače nemá dlouhou trvanlivost. Většinou vydrží kapacitní snímač tři roky.

Teplotní biometrické snímače

Teplotní snímače jsou založeny na citlivém čipu, pyrodetektoru, který snímá rozdíl mezi teplotami papilárních linií a jejich meziprostorem. Abychom získali otisk prstu, musíme u tohoto senzoru přejíždět prstem přes citlivou plochu. Otisk prstu pak získáme v podobě digitálních pásů. Tyto pásy pak poskládají konečný obraz daného otisku. Tento snímač ale nemá příliš valnou kvalitu.

Elektroluminiscenční biometrické snímače

„Princip činnosti je založený na využití speciální vrstvy, která reaguje na tlak způsobený luminiscenčním efektem. Důležité z hlediska funkčnosti je světlo – eliminující vrstva, která filtruje světlo z míst, kde na ni tlačí papilární linie.“ (11)

Obraz prstu je i v tomto případě získán v digitální podobě. Nevýhodou těchto senzorů je pak snadnější poškození zařízení díky velké náchylnosti na prach a vodu.

Radiofrekvenční biometrické snímače

Princip fungování radiofrekvenčního biometrického snímače je velmi dobře popsán na serveru biometricke-ctecky.cz.

„Princip činnosti spočívá v připojení generátoru střídavého signálu na 2 rovnoběžné desky (jedna deska je plocha snímače a druhá plocha otisku prstu). Jelikož je vlnová délka mnohem větší než délka desek, vyskytuje se pouze složka elektrického pole bez pole magnetického. Pokud tedy jedna z desek bude náš otisk prstu, tvar pole se změní a bude kopírovat tvar papilárních linií, tzn. výběžky a prohlubně otisku prstu. Vodivé prostředí mezi prstem a plochou je docíleno pomocí vodivé plochy kolem každého snímače, a proto i suché prsty nejsou problémem, jelikož se pracuje s živou tkání těsně pod povrchem pokožky. Zvlněním pole, které je způsobené přiloženým otiskem prstu, dopadá na senzory signál s rozdílnou velikostí signálu. Výběžky mají větší signál a tzv. údolí nižší signál. Kapacitní senzory tak měří rozdílnou permitivitu mezi výběžky a údolími.“ (11)

Z předchozí citace vyplývá, že tato technologie je velmi odolná vůči nečistotám. To zejména z toho důvodu, že snímače reagují s vodivostí. Další výhody jsou například i výše zmíněná akceptace suché pokožky, nebo dokonce i částečně poškozená kůže na prstech.

2.2. Trendy

Sensory jsou stále přesnější a levnější na výrobu. Jak bylo již řečeno, stávají se prostředkem konkurenceschopnosti jednotlivých výrobců. Z toho důvodu se

pravděpodobně v budoucnu budou objevovat i nejlevnější přenosná zařízení s většinou senzorů, zmíněných v předchozích částech této kapitoly. Kromě toho výrobci pracují na zdokonalování starých a vývoji nových senzorů.

Budoucnost senzorů se díky rychlému vývoji nedá přesně odhadnout. Lze se však zaměřit na novinky, které obsahují aktuální vlajkové lodě různých výrobců a tím si vytvořit nějakou představu o tom, kam bude toto odvětví průmyslu nadále směřovat.

Nové sensory jsou například využívány pod záštitou programu Samsung S-Health. Za zmínku zde určitě stojí měřič tepu, který je přítomen přímo na zařízení a není tak nutné dokupovat žádné nové příslušenství. Tento senzor se nachází na zadní straně zařízení pod fotoaparátem. Princip, jakým funguje je následující. Uživatel přiloží prst na senzor. Ten pomocí světla prst prosvítí, analyzuje pohyby cév nacházejících se v prstu, a díky tomu zjistí aktuální tep. Recenzent David Mysliveček však v jeho článku uvádí, že se u tohoto sensoru na Samsung Galaxy S5 setkal s několika chybami měření. (12) Uvádí zde například, že tento senzor vyžaduje velmi specifické přiložení prstu, uživatel se nesmí hýbat, nesmí mluvit, zrychleně dýchat nebo přehnaně tlačit na senzor. Z toho se dá odvodit, že tento senzor sice vydává správnou cestou, ale není ještě zdaleka v cíli.

Jak bylo již řečeno, pracuje se na vylepšování senzorů stávajících. Za zmínku zde stojí například RGB senzor. Jedná se o vylepšený senzor okolního osvětlení. Díky tomu, že mimo velikost osvětlení vnímá také jeho odstíny, dokáže upravovat barvy na displeji tak, aby působily co nejkomfortněji. (13) Dalším méně běžným senzorem je takzvaný senzor gest. Tento senzor dokáže pomocí infračervených paprsků snímat gesta rukou, aniž by se uživatel dotkl obrazovky. (13)

„Mnoho toho může nabídnout také nanotechnologie. První vlaštovkou je mobilní senzor, který vyvinul Jing Li, který pracuje v pro americkou NASA. Původně se snažil vyrobit senzor, který by sledoval úniky z paliva nosných raket. Jeho senzor je však tak malý, že je ho možné zabudovat do mobilního telefonu a vystačí si s pouhými pěti miliwatty. Detektor je natolik citlivý, že dokáže analyzovat toxiny ze vzduchu. Může vás varovat, před nebezpečnou koncentrací nebezpečných látek, jako jsou oxid uhelnatý, metan, chlor, amoniak a další.“ (1)

Jak je uvedeno v předchozí citaci, směrem, kterým by se mohly senzory vydat, je nanotechnologie. Tím se otevírá pro mobilní zařízení zcela nový obzor obsahující nespočet možností.

„Dal by se však využít i pro neinvazivní lékařské vyšetření. Stačí, aby na něj uživatel dýchl, a on určí, jaké toxiny se nachází v jeho dechu. Různé toxiny přitom signalizují různé nemoci.“ (1)

3. Analýza využitelnosti senzorů v dopravě

Od dob, kdy automobily byly jen čistě surové stroje, uplynulo již mnoho času. Rozvoj počítačové techniky zasáhl mnoho odvětví průmyslu. Proč tomu tak je? Jedná se především o rychlost a přesnost, s jakou dokáží moderní počítače zpracovávat informace. V dopravě jsou takové systémy, které slouží jako rozhraní mezi člověkem a strojem. Jak již tomu tak bývá, impuls vývoji těchto systémů dala armáda. Konkrétně obor vývoje a výzkumu letectví v minulém století přinesl technologie, které se začínají hojně využívat i v automobilovém průmyslu. Za zmínku stojí systém Fly-by-wire. Jedná se o systém nepřímého řízení letadla. Ten přinesl značnou revoluci tím, že pilot již neřídí letadlo sám, nýbrž přes ovládací prvky zasílá signály do počítače, který je zpracovává, vyhodnocuje a reprodukuje. Mimo informace od pilota má počítač informace z mnoha senzorů nacházejících se v letadle a dokáže díky tomu i minimalizovat chyby, kterých se může pilot dopustit. Po vzoru Fly-by-wire se začaly podobné systémy objevovat i v automobilech. Nazývají se asistenční systémy a právě zde hrají senzory velmi důležitou roli.

3.1. Asistenční systémy v automobilech

Základní funkcí asistenčních systému v automobilu je asistence řidiči vozidla. Asistenční systémy, jak vyplývá z názvu, by měly řidiči usnadnit ovládání vozidla. Díky nim lze, podobně jako v leteckém průmyslu, nehodám předejít nebo alespoň minimalizovat jejich dopad. Asistenční systémy umožňují řidiči také rychleji zareagovat. Často se stává, že řidič něco přehlédne a díky tomu dojde ke krizové situaci. Aby se dalo těmto situacím předejít, jsou zde právě tyto systémy, které slouží jako druhý pár očí. V některých případech mohou dokonce tyto systém úplně převzít kontrolu nad vozidlem, aby zabránily kolizi.

Asistenční senzory lze podle časopisu AUTOMA (14) rozdělit dle účelovosti na tyto skupiny:

- Sledování a řízení dopravní situace
- Sledování a řízení stavu vozidla

- Sledování stavu řidiče
- Komunikaci s řidičem
- Řízení vozidla (ve smyslu řízení všech příslušných systémů vozidla)

Systemy pro sledování a řízení dopravní situace

V rámci této kategorie, podle (14), jsou nejdůležitější systémy, které slouží pro zlepšení viditelnosti. Jsou užitečné zejména při nevhodných podmínkách, jako je déšť nebo mlha. Jsou zde využity především infračervené kamery a mikrovlnné radary. Objekt, který je díky špatné viditelnosti pro řidiče neviditelný, se pak promítne na poloprůhledném displeji nacházejícím se před čelním sklem. Tento display bývá označován jako HUD². Levnější automobily toto promítají skrz klasický display palubního počítače. I tak se tato technologie vyskytuje především v luxusním segmentu.

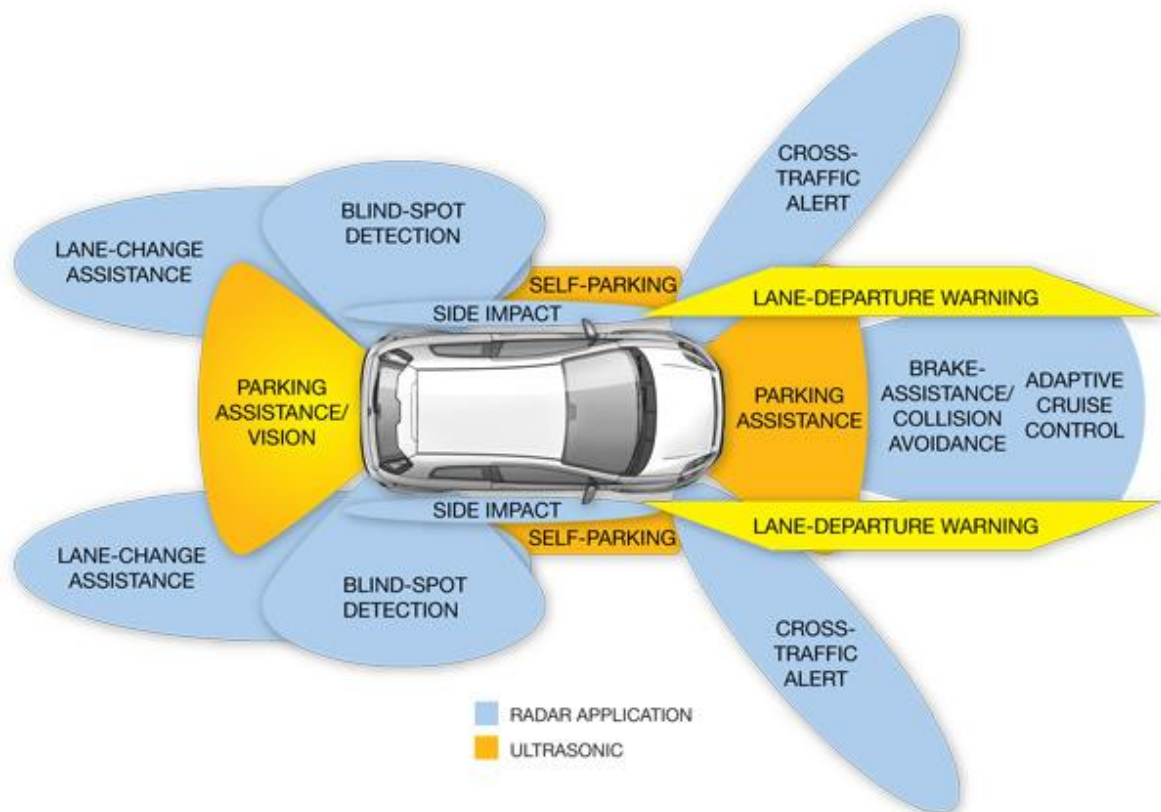
Vozidla mohou obsahovat dva typy tempomatů. Jedná se o klasický tempomat a tempomat adaptivní. Systémy pro zlepšení viditelnosti dokáží s adaptivním tempomatem komunikovat a v případě nutnosti regulovat rychlost vozidla. Pokud je dostatek času, jsou tyto systémy schopny vozidlo i zastavit.

Adaptivní tempomat

Senzory, které využívá adaptivní tempomat, jsou orientované nejen ve směru jízdy. Mohou být také orientovány směrem dozadu a do stran, jak je uvedeno na obrázku (Obr. 3). To zejména v úhlech, ve kterých má řidič omezený výhled.

Optické senzory mohou být využity i k rozpoznávání dopravních značek a včasnému upozornění řidiče. Tomuto přístupu však, alespoň na dálnicích, konkuruje spíše telematický přístup, využívající tzv. mluvící značky (značky se ozývají kódovým slovem neseným vysokofrekvenčním signálem) (14).

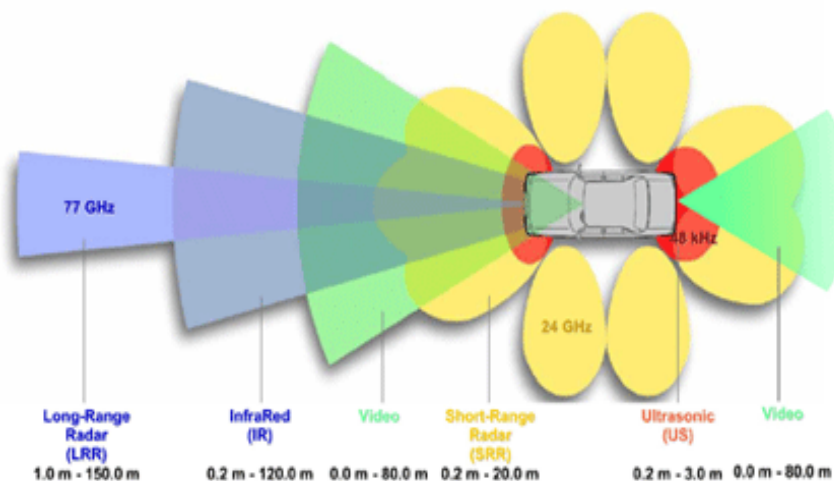
² Head-Up Display – tato technologie, jako řada dalších, pochází z vojenského prostředí. Je hojně využívána ve stíhacích letadlech.



Obr. 3: Přehled senzorů a jejich zón a využití (21)

V systémech adaptivních tempomatů různých verzí se používá množství senzorů. Jejich dosahy a vyzářovací diagramy jsou schematicky znázorněny obrázkem (Obr. 4). Podle něj má největší dosah mikrovlnný radar. Ten pracuje na frekvenci 77 GHz a je schopen detekovat překážku ve vzdálenosti asi 120 m před vozidlem. Tento radar v dosti velké vzdálenosti zaznamená překážku a umožní určit její rychlost. Radar by jako snímač postačoval pro jízdu na přímé silnici. Podle (15) musí být pro jízdu v zatáčkách nebo v hustém provozu doplněn infračervenou kamerou, popřípadě videokamerou pracující ve viditelné oblasti. Pomocí nich dokáže počítač pro zpracování obrazu analyzovat situaci před vozidlem. Radar i kamery umí zachytit větší počet vozidel před sebou podobně, jako dokáží některé fotoaparáty analyzovat několik lidských tváří. Po zpracování tohoto obrazu je pak vybráno nejbližší vozidlo. Sensory, radary orientované svým vyzářovacím diagramem do stran a dozadu a sonar s krátkým dosahem, slouží k varování před kolizí a jako součást protikolizních systémů. Pro krizové situace, kdy je již kolize nevyhnutelná, jsou vpředu a vzadu ultrazvukové senzory. Jinými slovy se

jedná o sonar, jehož dosah je jen okolo jednoho metru. Využívá se k aktivaci prvků pasivní bezpečnosti, jako je například dotažení pásů nebo aktivace airbagů. (viz. (16))



Obr. 4: Dosah a vyzařovací diagramy jednotlivých senzorů (11)

Senzory s příslušným počítačovým vybavením pro analýzu situace jsou základem řídicí jednotky systému ACC. Na volné silnici, kde není v dohledu žádné jiné vozidlo, udržuje konstantní rychlost. To znamená, že pokud má automobil automatickou převodovku, zadává řídicí jednotce motoru žádanou hodnotu otáček motoru. Jestliže se vpředu objeví pohybující se překážka, kterou snímače zaregistrují a rozpoznají jako automobil jedoucí menší rychlostí v tomtéž pruhu, vozidlo díky systému ACC dojede na předvolenou vzdálenost, přizpůsobí rychlost automobilu jedoucímu vpředu a sleduje jej. Jde-li o nepohybující se (pevnou) překážku, snaží se optimálně brzdit, aby před ní zastavil. Když se překážka objeví příliš pozdě, než aby bylo možné zastavit, maximálně zmenší rychlost a aktivuje protikolizní systémy. Při zmenšování rychlosti musí řídicí jednotka systému ACC spolupracovat s řídicími jednotkami motoru a brzd. Řídicí jednotka motoru a řídicí jednotka brzd, které byly původně nezávislé systémy, se nyní v rámci činnosti systému ACC chovají jako embedded systémy. (viz. (16))

Velmi užitečný je systém ACC při velmi malých rychlostech. Těchto rychlostí automobil dosahuje například při popojíždění v dopravních zácpách. Tento režim se nazývá stop and go. (14) Podle Petra Vysokého se v tomto režimu stává nejvíce

drobných nehod zejména z toho důvodu, že řidič neudrží po dlouhou dobu zvýšenou pozornost.

„Systém ACC umožní jízdu v koloně rychlostí řádově kilometry za hodinu, vždy s přesným zastavením ve vzdálenosti řádově metrů od automobilu jedoucího vpředu, jenž právě zastavil.“ (14)

Podobně jako tomu bylo u senzorů v mobilních zařízeních, i zde se tyto systémy dostávají postupně z luxusních automobilů čím dál více do automobilů střední třídy.

Systémy pro sledování a řízení stavu vozidla

Jako součást systémů pro sledování a řízení stavu vozidla je jeden z nejznámějších a nejrozšířenějších systémů ABS. ABS (zkratka převzata z anglického spojení Anti-lock Braking System) zabraňuje zablokování kola a následnému smyku při brzdění. Princip, jakým tento systém funguje, je popsán v následujících citacích.

„Systém zabraňuje zablokování kol při brzdění tím, že automaticky reguluje brzdnou sílu v třmenech tak, aby nedošlo k zablokování kol. Při zablokování kola by totiž došlo ke ztrátě adheze mezi pneumatikou a vozovkou, tím by se stalo vozidlo neřiditelné. Vozidlo by nebylo možné řídit otáčením volantu.“ (17)

„U ABS se při brzdění porovnává postupná rychlost vozu stanovená z údajů akcelerometru měřícího jeho podélné zrychlení (resp. zpomalení), s postupnou rychlostí stanovenou na základě měření úhlové rychlosti otáčení kol. Jestliže obě rychlosti nesouhlasí, dochází k prokluzu kol, a ABS upraví brzdné síly tak, aby k prokluzu nevznikal. Uvedený systém může také fungovat jako elektronická uzávěrka diferenciálu při jízdě na kluzkém povrchu.“ (14)

Zdroj (14) dále uvádí, že další součástí systémů pro sledování a řízení stavu vozidla jsou systémy automatické stabilizace. Mezi ně patří například ASC (Automated Stability Control) a ESS (Electronic Stabilization Systems). Jejich hlavní uplatnění je zejména při zatáčení ve vysokých rychlostech. V takovém případě vlivem odstředivé síly, pokud je příliš velká, může dojít ke vzniku smyku. Při extrémních hodnotách může dokonce nastat převržení vozu.

„Kombinací selektivního brzdění jednotlivých kol, selektivního hnacího momentu přenášeného na jednotlivá kola a změnou tuhosti pérování jednotlivých kol je možné omezit vliv příčného zrychlení tak, že popisované nežádoucí jevy nemohou nastat. Vhodným řízením tohoto systému lze upravit sklon vozidla k nedotáčivému nebo přetáčivému chování.“ (12)

Jako hlavní prvek tohoto systému je právě akcelerometr.

Systémy pro sledování stavu řidiče

Auto samo o sobě je jedním z nejspolehlivějších zařízení. Stojí za ním dlouhý vývoj a testování. Jak Petr Vysoký píše (14), výsledkem je, že nejméně spolehlivou součástí celého systému řízení vozidla se stává samotný řidič. Výhodou člověka je schopnost improvizovat ve složitých dopravních situacích, pro které neexistují předem daná pravidla. Naopak nevýhodou je, že řidič může například podlehnout únavě. Z toho důvodu je nutné sledovat stav řidiče. Jakmile je zaznamenána menší pozornost řidiče, může ho automobil určitým způsobem varovat a vyzvat, aby jízdu přerušil a odpočinul si.

Příklad systémů kontrolující stav řidiče (18):

- DAM (Driver Alertness Monitoring)
 - Monitorovací systém pohotovosti řidiče
 - Za pomoci infračervené kamery se snímají pohyby očí řidiče
 - Sleduje vybočení vozidla z jízdního pruhu
- DAS (Driver Attention Monitoring Systém)
 - Monitorovací systém pozornosti řidiče
 - Systém firmy Mitsubishi pro sledování stupně pozornosti, kterou řidič užitkového vozidla věnuje jízdě
 - Senzory sledují styl jízdy
 - Systém zvukem upozorňuje na nebezpečí opuštění jízdního pruhu
- DSM (Driver Status Monitoring)
 - Monitorování stavu řidiče

3.2. Využitelnost senzoru mobilního zařízení

Z předchozích částí této kapitoly vyplývá, že to, co mají současná vozidla a telefony, popřípadě jiná přenosná zařízení společně, jsou právě senzory. V určitém slova smyslu jsou i vozidla mobilními zařízeními. Obsahují i dokonce i podobné senzory. Ty jsou v každém z těchto typů zařízení používány svým specifickým způsobem. Lze však skloubit senzory přenosného zařízení s chováním automobilů?

Vozidla mají své senzory přímo přizpůsobené k tomu, co mají dělat. Senzory přiblížení, akcelerometry a jiné jsou přímo kalibrované a nastavené, aby odváděly co nejlépe svoji práci. Naproti tomu senzory dostupné v telefonech, jsou primárně vytvořené k tomu, aby zaznamenávaly prostředí a kontext, ve kterém se nacházejí.

Využití akcelerometru

Aby mohlo přenosné zařízení pomocí akcelerometru zaznamenávat stejné, nebo alespoň podobné údaje, jako senzory v automobilu, je nutné ho pevně propojit s konstrukcí vozidla. Pokud by řidič nebo nějaká jiná osoba v automobilu měla své zařízení například v kapse, zařízení by pomocí akcelerometru bylo schopné zaznamenat údaje o síle zrychlení automobilu. Byly by však značně zkresleny chováním lidského těla při působení těchto sil. Nemluvě o tom, že lidé v automobilu nesedí neustále v jedné poloze, ale naopak se pohybují.

Další velice oblíbenou možností, jak převážet mobilní zařízení v automobilu, je ho odložit do nějaké přihrádky. Tyto přihrádky, až na pár výjimek v luxusních automobilech, nejsou navrženy tak, aby telefon pevně držely na svém místě. Z toho důvodu při jízdě dochází k nechtěnému pohybu zařízení ve směru odstředivé síly a to vede k dalším, tentokrát již nechtěným otřesům zaznamenaných akcelerometrem.

Lepší volbou je využití držáků do auta. I zde však existují jistá omezení. Některé držáky mají slabší konstrukci. Z toho důvodu může při jízdě docházet ke značným vibracím zařízení a to zvláště pokud má zařízení vyšší hmotnost a držák je připevněn k čelnímu sklu. Tyto vibrace lze omezit, když se zařízení svojí spodní hranou dotýká palubní desky vozidla.

Co se využitelnosti senzorů přenosného zařízení týče, nejlepší volbou by bylo pevné připojení k samotné konstrukci vozidla. Zde by docházelo k čistému přenosu sil, které působí na automobil. U moderních aut se toto místo v interiéru pravděpodobně nenachází. Lze však využít pevného připojení zařízení k palubní, přístrojové desce nebo ke středovému panelu. Měření zde nedosahuje takové přesnosti, jako kdyby bylo zařízení přímo přiděláno ke konstrukci. Výchyly oproti této metodě jsou však zanedbatelné a lze je řešit programově pomocí low-pass filtrů.

Využití ostatních senzorů

V automobilové dopravě se velmi rozšířilo využívání navigací. Navigace využívá nejčastěji systému GPS k rozeznání polohy, kde se aktuálně vozidlo nachází. Jak bylo již zmíněno v první kapitole této práce, k urychlení vyhledávání aktuální polohy se využívá celá řada přístupů. Jedním z nich je využití barometru k upřesnění nadmořské výšky. Po akcelerometru je to tedy další senzor, který se vyskytuje v přenosném zařízení a lze ho velice dobře využít i v automobilové dopravě.

Adaptivní tempomaty, kterými jsou vybavena některá vozidla, obsahují senzory přiblížení. I když mobilní telefony také obsahují senzory přiblížení, tak pro využití v dopravě jsou využitelné jen velmi málo. Proč tomu tak je? Hlavním důvodem je použitá technologie a určení, k čemu mají tyto senzory sloužit. U mobilního zařízení je dosah tohoto senzoru jen pár centimetrů, protože byl navržen převážně pro identifikaci vzdálenosti od lidské tváře, aby mohl posléze telefon vypnout display (viz. 2.1). Senzory přiblížení, které jsou třeba v automobilech, potřebují dosah daleko větší.

Dalším senzorem využívaným v souvislosti s akcelerometrem je magnetometr. Jeho využití v dopravě má svá jistá úskalí. Tento senzor pracuje na podobném principu jako kompas. Stejně jako kompas je velmi náchylný na zdroje magnetické síly ve svém okolí. Pokud se tedy nachází mobilní zařízení ve vozidle, je velmi pravděpodobné, že data z něj budou zkreslena vnějšími vlivy.

Ostatní senzory jsou velmi užitečné, co se mobilního zařízení týče. Lze však jejich užitečnost aplikovat na automobilový průmysl? Například senzor okolního osvětlení v mobilním zařízení by mohl být využit pro upozornění řidiče, že venku je již nízké

osvětlení a že je třeba zapnout světla. Biometrické senzory by zase mohly sloužit ke zvýšení bezpečnosti vozidel. Využití senzorů mobilního zařízení je nespočetně mnoho. Stačí nalézt odpověď na dvě otázky: Co a jak?

3.3. Možnost propojení přenosného zařízení s vozidlem

Moderní automobily obsahují určitý stupeň konektivity s mobilním zařízením. Jedná se především o konektivitu s multimediálním systémem. Ta je řešena buď přes kabel nebo bezdrátově za použití technologie bluetooth či Wi-Fi. Dražší automobily dokáží dokonce komunikovat přes internet. Toto je využíváno například v obytných vozech, kde si může řidič přes mobilní telefon na dálku nastavit vytápění vozu.

Tato konektivita je využívána hlavně k zvýšení komfortu při užívání automobilu. Bylo by však velice zajímavé, kdyby se přes tento informační most dala zpřístupnit data ze senzorů nainstalovaných přímo v zařízení. To by otevřelo zcela nové možnosti při vývoji mobilních aplikací. Tato možnost se však zdá být nepravděpodobná především z toho důvodu, že vozidla jako taková obsahují dostatečně výkonné výpočetní systémy, aby operace nad senzory prováděly samy. Informace, které jsou poskytovány přes palubní počítač řidiči, se mohou lišit na stupni výbavy automobilu. Pokud by byla tato data zpřístupněna přes nějaké rozhraní, automobilky by mohly přicházet o část svých zisků tím, že informace by neposkytovala jejich vozidla, nýbrž aplikace třetích stran.

Mnoho automobilek tuto problematiku řeší přes své vlastní aplikace. Zde se však mobilní zařízení nevyužívá k výpočtům. Slouží především jako front-end k zobrazení informací o vozidle nebo k zadávání jednoduchých instrukcí. Na druhou stranu může smartphone sloužit jako zdroj informací pro automobil. Ten pak může skrz něj kontrolovat nové události, jako jsou například příchozí emaily a poskytovat je řidiči přímo skrz palubní počítač. Více například na oficiálních stránkách Audi. (19)

Z předchozího odstavce vyplývá, že automobily a přenosná zařízení jsou stále propojenější. Navzájem spolu dokáží komunikovat a využívat informací získaných z této komunikace. Tyto informace se však týkají pouze informativního a multimediálního obsahu.

4. Softwarová rozhraní pro přístup k sensorům v Androidu

Android Sensor Framework je rozhraní, které dovoluje vývojáři přistupovat k sensorům a k datům, která tyto senzory poskytují. S tím souvisí i schopnost určit, jaké senzory se v zařízení vyskytují. Lze pomocí něj řídit, jak mají být senzory přesné, tedy, jak rychle se mají obnovovat. Toto rozhraní je poskytováno jako součást balíčku `android.hardware`, který mimo jiné obsahuje také nástroje pro obsluhu kamery. Patří mezi základní balíčky dostupné pro vývoj aplikace.

Sensor Framework obsahuje následující komponenty (3):

- Sensor manager
- Sensor
- Sensor event
- Sensor event listener

4.1. Sensor manager

```
SensorManager sensorManager =  
    (SensorManager) getSystemService(Context.SENSOR_SERVICE);
```

Kód 1: Získání instance SensorManager

Sensor manager je pro sensorové rozhraní stěžejní třídou. V předchozím kódu (Kód 1) je znázorněno, že jeho instance je získána ze služeb. Tyto služby se nachází na úrovni systému. Přes své statické metody dokáže poskytnout všechny senzory, které jsou na zařízení k dispozici. Pro získání surových dat ze sensorů lze pomocí něj registrovat listenery. Ty jsou registrovány pomocí konstant, které také poskytuje třída sensor manager.

Konstanty

Sensor manager poskytuje celou řadu konstant, pomocí kterých lze upravit data, která jsou přes listenery poskytována. Jedním typem konstant je například definování rychlosti, jakou mají být data poskytována. Ty jsou využívány při vytváření listeneru.

- **SENSOR_DELAY_FASTEST**
 - Získávání dat ze senzorů nejrychleji jak je to možné
- **SENSOR_DELAY_GAME**
 - Druhé nejrychlejší nastavení, které se využívá především pro hraní her
- **SENSOR_DELAY_UI**
 - Používá se pro změny uživatelského rozhraní (UI – User Interface)
- **SENSOR_DELAY_NORMAL**
 - Základní nastavení
 - Používá se například pro změny orientace obrazovky

S listenery dále souvisí zjištění, s jakou přesností jsou dodávána data ze senzorů. Přesnost nabývá celkem tří hodnot. Tyto hodnoty jsou definovány dalšími konstantami.

- **SENSOR_STATUS_ACCURACY_HIGH**
 - Vysoká přesnost
- **SENSOR_STATUS_ACCURACY_MEDIUM**
 - Střední přesnost
- **SENSOR_STATUS_ACCURACY_LOW**
 - Nízká přesnost
 - V oficiálních zdrojích je uvedeno, že pokud senzor podává data s touto přesností, je nutné ho kalibrovat
- **SENSOR_STATUS_ACCURACY_UNRELIABLE**
 - Data vracená s touto přesností jsou nepoužitelná
 - Je nutná kalibrace, nebo se zařízení nenachází v prostředí, ze kterého by mohla získávat data

- `SENSOR_STATUS_ACCURACY_NO_CONTACT`
 - Data vracená s touto přesností jsou nepoužitelná
 - Senzor nemá žádný kontakt s tím, co má měřit (V Android API je uveden jako příklad měřič tepu)

Dalšími důležitými typy konstant jsou ty, které upravují data, která jsou přes listenery vracena. Každé z nich mají své specifické využití. Za zmínění stojí konstanty, které dovolují simulovat gravitaci na různých vesmírných objektech. Některé z nich jsou uvedeny níže. Všechny tyto konstanty vrací floatové hodnoty v jednotkách SI³ (m/s²)

- `GRAVITY_EARTH`
 - Základní nastavení
 - Hodnota gravitace, která je na Zemi (cca 9.81 m/s²)
- `GRAVITY_MOON`
 - Gravitace měsíce
- `GRAVITY_SUN`
 - Gravitace slunce
- `GRAVITY_DEATH_STAR_I`
 - „Gravitace (odhad) na první Hvězdě smrti v jednotkách Impéria⁴ (m/s²)“ – přeloženo z Android API (20)
- ...

Existují také konstanty ovlivňující data, která se týkají osvětlení. Tyto konstanty jsou uvedeny v jednotkách lux. Jedná se o jednotky, které určují intenzitu osvětlení. Některé z těchto konstant jsou uvedeny níže.

- `LIGHT_SUNLIGHT`
 - Osvětlení za slunného počasí
- `LIGHT_NO_MOON`
 - Osvětlení v noci, pokud nesvítí měsíc

³ Soustava SI (zkratka z francouzského Le Système International d'Unités) je mezinárodně domluvená soustava jednotek fyzikálních veličin, která se skládá ze základních jednotek, odvozených jednotek a násobků a dílů jednotek. (22)

⁴ Odkaz na filmy Star Wars. Podobné odkazy jsou u vývojářů poměrně běžné.

- LIGHT_CLOUDY
 - Osvětlení, když je oblačno
- ...

Sensor manager také dovoluje změnu svého souřadnicového systému. Každá osa souřadnic má svůj vlastní identifikátor, přičemž je zde brán ohled i na záporné osy. Těmito identifikátory jsou právě konstanty uvedené v sensor manageru. Příklad identifikace os pomocí konstant je uveden níže.

- AXIS_X
 - Identifikace osy X
- AXIS_MINUS_X
 - Identifikace záporné hodnoty osy X

Pomocné metody

Jak bylo již zmíněno, sensory poskytují vývojáři surová data, která může díky listenerům libovolně využívat. Použití těchto dat je ovšem poměrně složité. Aby byl vývoj aplikace co nejvíce zjednodušen, jsou ve třídě Sensor manager implementovány pomocné metody, které dovedou velmi zjednodušit práci se sensorovými daty. Například bez těchto metod by musel vývojář jen pouze s využitím tří souřadnic vypočítat velikost úhlu náklonu zařízení nebo sám zkonstruovat matici rotace.

Matice rotace a matice náklonu

Matice rotace v sobě obsahuje hodnoty rotace okolo všech os, tedy okolo x, y, z. Tato matice velikosti 4x4 slouží jako základ pro většinu výpočtů. Slouží jako nositel informací o poloze zařízení. Lze jí získat pomocí dvou metod, tedy `getRotationMatrix` a `getRotationMatrixFromVector`.

Pro použití první metody je nutné znát data jak z akcelerometru, tak data z magnetometru. Důvod použití těchto dvou senzorů je ten, že pokud leží zařízení displejem nahoru, tak pomocí akcelerometru se získávají úhly rotace kolem os X a Y. Díky použití magnetometru lze získat úhel rotace okolo osy Z. Pomocí této metody lze získat nejen matici rotace, ale také matici náklonu. Rotační matice lze dále také získat

pomocí rotačního vektoru. Tato možnost je dostupná pro zařízení, která obsahují potřebný senzor, tedy TYPE_ROTATION_VECTOR.

Zatímco matice náklonu je využívána převážně pro získání úhlů náklonu zařízení pomocí metody `getInclination`, tam matice rotace má daleko širší využití. Lze použít pro zjištění orientace zařízení. Porovnáním dvou rotačních matic lze pomocí metody `getAngleChange` zjistit, o jaký úhel jsou obě matice oproti sobě natočené. Díky své velikosti, 4x4, a jejímu tvaru je ji možné rovnou použít pro vykreslování v OpenGL ES při použití `glLoadMatrixf (float[], int)` (20).

Změna souřadnicového systému

Matice rotace je závislá na určitém systému souřadnic. Často se ale stává, že je zapotřebí tento systém souřadnic změnit. Zařízení není pevně připevněné k zemi, je mobilní. Lze s ním libovolně pohybovat a natáčet ho. Změna souřadnicového systému je tedy výhodná pro zjednodušení výpočtů. To je prováděno pomocí metody `remapCoordinateSystem`. Tato metoda má celkem čtyři parametry. První slouží pro zadání stávající rotační matice. Další je naopak použit pro získání rotační matice, která vznikla po transformaci souřadnicového systému. Další dva jsou použity pro informace, jak má být souřadnicový systém transformován. Nastavuje se zde, na jaké osy souřadnicového systému se mají promítnout stávající osy X a Y. Pro tyto účely jsou použity konstanty zmíněné v kapitole 4.1 – konstanty

Listenery senzorových událostí

Pro získání aktuálních dat ze senzorů slouží listenery. Jejich registrace je zprostředkována třídou `SensorManager`. Při registraci listenerů lze zadat i několik dalších parametrů, které dokáží ovlivnit, jak mají být daná data poskytována. Důležitým parametrem je například definování rychlosti, jakou lze data získávat. Nastavení rychlosti poskytování dat senzory je velice důležité. Musí se zde například brát v úvahu i náročnost na spotřebu energie zařízení. Jak již bylo zmíněno, pro získání dat se musí přes `SensorManager` registrovat listenery. Je však nezbytně nutné tyto listenery také odregistrovat. Pokud by se na to zapomnělo, zůstaly by zapnuté, i kdyby aplikace již nebyla aktivně využívána a hrozilo by vybití zařízení za několik málo hodin. Samotné listenery poskytují přímo surová data ze senzorů. Tato data jsou vrácena ve

floatovém poli, které většinou obsahuje hodnoty X, Y, Z. To platí zvláště při využití akcelerometru.

4.2. Senzory a jejich dostupnost

| Sensor | API 14 | API 9 | API 8 | API 3 |
|--------------------------|------------------|------------------|------------------|-------|
| TYPE_ACCELEROMETER | Ano | Ano | Ano | Ano |
| TYPE_AMBIENT_TEMPERATURE | Ano | | | |
| TYPE_GRAVITY | Ano | Ano | | |
| TYPE_GYROSCOPE | Ano | Ano | | |
| TYPE_LIGHT | Ano | Ano | Ano | Ano |
| TYPE_LINEAR_ACCELERATION | Ano | Ano | | |
| TYPE_MAGNETIC_FIELD | Ano | Ano | Ano | Ano |
| TYPE_ORIENTATION | Ano ⁵ | Ano ⁵ | Ano ⁵ | Ano |
| TYPE_PRESSURE | Ano | Ano | | |
| TYPE_PROXIMITY | Ano | Ano | Ano | Ano |
| TYPE_RELATIVE_HUMIDITY | Ano | | | |
| TYPE_ROTATION_VECTOR | Ano | Ano | | |
| TYPE_TEMPERATURE | Ano ⁵ | Ano | Ano | Ano |

Tab. 1: Dostupnost senzorů podle platformy. (3)

Jak je uvedeno v předchozí tabulce (Tab. 1) dostupnost senzorů je dána platformou, která je na zařízení dostupná. Toto však platí převážně pro takzvané softwarové senzory. Jak jsou však zařízení skutečně vybavena, ve smyslu

⁵ Označen jako zastaralý

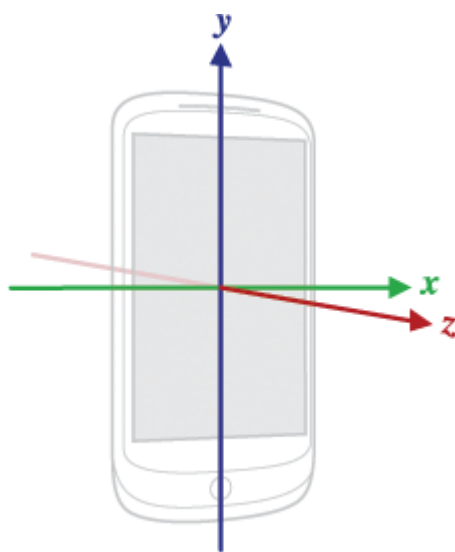
hardwarových senzorů, záleží převážně na výrobci zařízení. Z toho důvodu je nutné při tvorbě aplikace testovat, jestli senzor, který chce vývojář použít, je na zařízení skutečně dostupný.

4.3. Souřadnicový systém senzorů

Matice rotace, která byla podrobněji rozebírána v předchozí kapitole, je vlastně matematickým znázorněním rotace souřadnicového systému zařízení vzhledem k souřadnicovému systému světa.

Zařízení má svůj souřadnicový prostor pevně daný (viz. Obr. 5):

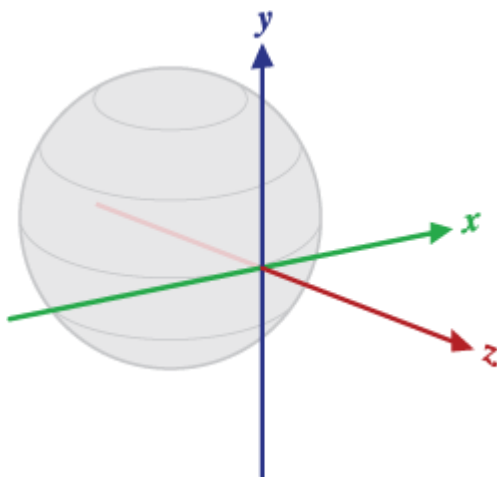
- Osa Y směřuje od spodu zařízení k reproduktoru.
- Osa X z levé strany zařízení k pravé
- Osa Z směřuje z displeje směrem k pozorovateli.



Obr. 5: Souřadnicový systém zařízení (2)

Dle (21) je souřadnicový systém světa následující (viz. Obr. 6):

- Osa Y směřuje k severu
- Osa X na východ
- Osa Z od středu Země k nebi.



Obr. 6: Souřadnicový systém světa (14)

Z toho vyplývá, že pokud by bylo zařízení položeno na zádech a jeho horní část by směřovala přímo k severu, výsledná matice rotace by byla nulová.

4.4. Sensory a vývoj aplikace

Typická aplikace využívá sensorově orientovanou API pro vykonávání dvou základních úloh. První úlohou je identifikování sensorů, které jsou na zařízení přítomny. S tím souvisí i zjištění, jakou funkci sensory plní a jaké mají schopnosti. Druhou úlohou je monitorování sensorových událostí. Díky tomu lze reagovat na různé změny chování v senzorech. Tímto chováním se rozumí reakce na změnu hodnot dat, které sensor poskytuje, a reakce na změnu přesnosti sensorů.

Identifikování sensorů

```
List<Sensor> sensors = sensorManager.getSensorList(Sensor.TYPE_ALL);
```

Kód 2: Získání všech dostupných sensorů

Informace o tom, jaké sensory má aktuálně používané zařízení k dispozici, se dá získat skrz metodu `getSensorList`, zprostředkovanou přes instanci třídy

SensorManager. To je ukázáno ve zdrojovém kódu výše (Kód 2). Poté lze jednoduše přístupem do vzniklého listu zjistit kompletní informace senzorech.

Monitorování senzorových událostí

```
SensorEventListener mSensorEventListener =
    new MySensorEventListenerImpl();

if(sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)!=null){

    sensorManager.registerListener(mSensorEventListener,
    sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER),
    SensorManager.SENSOR_DELAY_UI);
}
}
```

Kód 3: Implementace a registrace senzorových událostí

V předchozí ukázce zdrojového kódu (Kód 3) je znázorněno, jak by měla vypadat implementace a registrace senzorových událostí. Je zde využito vlastního interface `MySensorEventListener`, který rozšiřuje původní `SensorEventListener` jen o několik menších funkcionalit. Jeho hlavní metody však zůstávají stejné. Podstatné jsou zde však následující řádky. Protože každé zařízení, na kterém může být aplikace nainstalována, může mít k dispozici jiné senzory, je nutné ověřit, zdali je potřebný senzor zařízením podporován. Pokud ano, lze pomocí třídy `SensorManager` registrovat nový listener. Je zde také dobře vidět, jak se dá provést nastavení přesnosti senzoru. Zde se opět využije třídy `SensorManager`, kde jsou pro tento účel deklarovány statické atributy. Po registraci nového listeneru je nutné také tento listener odregistrovat, pokud již nemá být používán. To je doporučeno obzvláště z důvodu šetření energie zařízení. Vhodné je volat metodu pro odregistrování například v metodě `onPause` aktivity, jak je znázorněno v následujícím zdrojovém kódu (Kód 5).

```
@Override
public void onPause() {
    sensorManager.unregisterListener(mSensorEventListener);
    super.onPause();
}
```

Kód 5: Volání metody onPause

Jak bylo popsáno výše, aplikace musí být schopna reagovat na změnu hodnot dat a změnu přesnosti senzorů. Jako u většiny hardwaru, se kterým zařízení nějak komunikují, se toto děje pomocí listenerů. Je zde využito již zmíněného rozhraní `SensorEventListener`. Jeho základní implementace je ukázána v následujícím kódu (Kód 4).

```
SensorEventListener mSensorEventListener = new SensorEventListener(){
@Override
public void onSensorChanged(SensorEvent event) {}
@Override
public void onAccuracyChanged(Sensor sensor, int accuracy) {}
};
```

Kód 4: Implementace rozhraní `SensorEventListener`

5. Přesnost a možnosti kalibrace senzorů

U polohových a pohybových senzorů, konkrétně například u akcelerometru, gyroskopu nebo magnetometru, se nejčastěji lze setkat s nutností kalibrace. Sensory poskytují velké množství dat. Data mohou obsahovat určitou míru zkreslení nebo v sobě mohou obsahovat i nadbytečné externí vlivy. Postupem času také může docházet ke ztrátě přesnosti. Z toho důvodu je nutné sensory kalibrovat. Avšak i kalibrace samotná v sobě může určitý stupeň chyby. Při vývoji aplikace se tak musí zvolit správný postup, jak data využívat. Je důležité rozhodnout, jestli využívat data surová nebo kalibrovaná. V případě nutnosti kalibrace je pak možné si u některých senzorů vybrat kalibrace předdefinované, anebo si vytvořit vlastní.

5.1. Zjištění aktuální přesnosti

Android Sensor Framework obsahuje možnost, jak zjistit přesnost daného senzoru. V aplikaci při implementaci listeneru na konkrétní senzor lze pomocí metody `onAccuracyChanged` zaznamenat událost, kdy došlo ke změně přesnosti, jak je zobrazeno v následujícím kódu (Kód 6). Na tuto událost lze pak libovolně reagovat.

Více o jednotlivých přesnostech, které jsou zde uvedeny, bylo popsáno dříve (4.1 – podkapitola Konstanty).

```
@Override
public void onAccuracyChanged(Sensor sensor, int accuracy) {
    switch (accuracy) {
        case SensorManager.SENSOR_STATUS_ACCURACY_HIGH:
            break;
        case SensorManager.SENSOR_STATUS_ACCURACY_MEDIUM:
            break;
        case SensorManager.SENSOR_STATUS_ACCURACY_LOW:
            break;
        case SensorManager.SENSOR_STATUS_UNRELIABLE:
            break;
        case SensorManager.SENSOR_STATUS_NO_CONTACT:
            break;
    }
}
```

Kód 6: Získání přesnosti senzoru

5.2. Kalibrace v Android Sensor Framework

Jak již bylo dříve napsáno, vývojáři je přes Android Sensor Framework poskytováno několik typů senzorů. Jedním typem jsou senzory, které vrací surová, nijak nezpracovaná data. Druhým typem jsou senzory poskytující již nějakým způsobem data předzpracovaná. Data z jednoho hardwarového senzoru mohou být tedy vývojáři poskytnuta přes toto rozhraní jako několik softwarových senzorů. V některých se lze setkat právě s kalibrací.

Sensor magnetického pole

Sensor magnetického pole je vývojáři nabízen jak v kalibrované, tak nekalibrované verzi. Tím, co se na magnetickém senzoru kalibruje, je zkreslení, které může nastat, pokud se mobilní zařízení vyskytne blízko nějakého předmětu vydávajícího magnetické záření. Jak bylo již řečeno v předchozích kapitolách, magnetometr funguje podobně jako kompas. Pokud je kompas přiblížen k nějakému magnetu, jeho ručička se od severu vychýlí. Podobným vlivům má právě kalibrace zabránit. Jak je ale zmíněno na (22), pokud je magnetické záření příliš silné, tak i tato kalibrace nestačí, aby tento sensor poskytoval správné hodnoty.

Gyroskop

Podobně jako u senzoru magnetického pole i gyroskop je nabízen v kalibrované a nekalibrované verzi. To, co se však musí u gyroskopu kalibrovat, je jeho posun. V první kapitole této diplomové práce bylo zmíněno, že gyroskop je zavěšen na ložiscích a že díky tomu v něm dochází k určitému tření, tedy určitému stupni zkreslení. Toto má právě kalibrace redukovat. Vztah mezi kalibrovaným a nekalibrovaným typem gyroskopu je na (13) popsán následovně:

$$kalibrované_x \sim = nekalibrované_x - odhad_zkreslení_x$$

Dále je uvedeno, že tato kalibrace může vytvářet v proudu dat určité skoky a dokonce může i některé výsledky v aplikacích určitou měrou zkreslit, jak vyplývá z citace níže.

Z toho důvodu se nekalibrované hodnoty senzoru používají pro aplikace, které potřebují pro svoji práci tento proud dat plynulejší.

„Nekalibrované senzory poskytují hrubější výsledky. Mohou zahrnovat určité zkreslení, ale jejich měření obsahují menší skoky z opravy aplikované prostřednictvím kalibrace. Některé aplikace mohou preferovat tyto nekalibrované výsledky, které jsou hladší a spolehlivější.“ – přeloženo z (21)

Vektor rotace

Vektor rotace je poskytován přes Android Sensor Framework pomocí tří typů senzorů:

1. Rotation Vector Sensor
2. Geomagnetic Rotation Vector Sensor
3. Game Rotation Vector Sensor

Rotation Vector Sensor je prvním ze tří typů softwarového sensoru. Poskytuje informace o kolik stupňů je natočeno zařízení podle os X, Y, Z. Obsahuje tedy celkem tři údaje, které jsou reprezentovány následovně (21):

- $x * \sin(\theta / 2)$
- $y * \sin(\theta / 2)$
- $z * \sin(\theta / 2)$

Výsledné údaje, které tento typ softwarového sensoru poskytuje, jsou získány zpracováním údajů z gyroskopu. Geomagnetic Rotation Vector Sensor poskytuje podobná data. K jejich výpočtu nevyužívá gyroskop, ale magnetometr. To mu přináší jak určité výhody, tak nevýhody. Na serveru developers.android.com (21) je jako výhoda zmíněna nízká energetická náročnost. Naopak jako nevýhoda je uváděno, že data poskytovaná tímto senzorem poskytují značné výchylky. Oby tyto dva sensory jsou kalibrovány tak, aby poskytovaly údaje, které jsou vztaženy k souřadnicovému systému světa (viz. 4.3 Souřadnicový systém senzorů). Nekalibrovaný typ sensoru je poskytnut skrz Game Rotation Vector Sensor. Tento sensor není vázán geomagnetickým polem Země. K identifikaci otočení zařízení se využívá gyroskop. Již

z názvu vyplývá, že je využíván především v aplikacích, kde není podstatná poloha vůči Zemi, tedy především v mobilních hrách.

Manuální kalibrace



Obr. 7: Ukázka kalibrace G-Sensoru

Manuální kalibraci senzorů je nejčastěji zprostředkována přes ROM, neboli softwarová nadstavba operačního systému Android, která je obvykle dodávána výrobcem. Na obrázku (Obr. 7) je vidět, jak je řešena kalibrace G-senzoru v prostředí HTC Sense 3.5. Kalibrace zde probíhá tak, že uživatel musí položit zařízení na rovný povrch a následně je G-Senzor vycentrován na všech osách. To je znázorněno graficky jako vodováhy. Samsung kalibraci gyroskopu ve své nadstavbě, TouchWiz, řeší podobně. Je zde však místo vodováhy znázorněn terč, který je zaměřován stále přesněji. Výrobci však v této době nepředpokládají, že by bylo nutné senzory, které poskytují ve svých zařízeních, kalibrovat. Proto se lze s touto možností manuální

kalibrace setkat stále vzácněji. Pokud však má uživatel pocit, že je nutné zařízení kalibrovat a v jeho nastavení tyto možnosti kalibrace chybí, je možné si stáhnout aplikace třetích stran, které se na tuto činnost specializují. Velice často však tyto případy končí reklamací produktu.

5.3. Kalibrace dat senzorů při vývoji aplikace

Kalibrace senzorů jako takových je jedna věc, tou druhou je však kalibrovat proud dat, které která senzor poskytuje. Některé aplikace potřebují pro svoje potřeby tato data kalibrovat specificky. Například potřebuje znát svůj normálový stav a oproti němu jsou pak zaznamenávány výchyly.

Kalibrace pozice zařízení

V aplikaci, která je vyvíjena jako součást této diplomové práce je důležité kalibrovat zařízení pro získání jeho ustálené polohy. Aplikace si poté zapamatuje, jakou polohu, na základě dat z akcelerometru, má zařízení vůči povrchu Země. Z těchto údajů jsou posléze prováděny potřebné výpočty.

V praxi to probíhá tak, že je uložena matice M_0 , která obsahuje aktuální hodnoty akcelerace ve všech osách. Pokaždé, když sensory poskytnou nová data, je sestrojena nová matice M_1 . Následně jsou matice M_1 a M_0 porovnány. Na základě toho je získáno aktuální zrychlení, které na zařízení působí, oproti kalibrovaným hodnotám uloženým v matici M_0 . Díky tomu se lze například efektivně zbavit gravitačního zrychlení Země. Dále je v aplikaci při kalibraci ukládána rotační matice. Ta je zase důležitá pro zjištění aktuálního náklonu zařízení oproti kalibrovanému stavu. Tento problém je více rozebrán v kapitole 6.5.

Low-pass filtr

Bylo zmíněno, že kalibrace může sloužit v Android Sensor Framework k minimalizování nechtěných vlivů z okolního prostředí do výsledných dat. Z toho vyplývá, že i low-pass, v překladu nízkofrekvenční, filtr lze považovat za určitý stupeň

kalibrace. Tento filtr je používán v aplikaci, která je vyvíjena jako součást této diplomové práce. Jeho implementace bude v následujících kapitolách dále rozvedena.

„Low-pass filtr je filtr, který umožňuje signály pod mezním kmitočtem (známý jako pásmo propustnosti) a zeslabuje signály nad mezní frekvence (známé jako kmitočtové pásmo). Tím, že odstraní některé frekvence, filtr vytváří vyhlazovací účinek.“ – přeloženo z (23)

V aplikaci je tento filtr použit jako prostředek k minimalizování šumu, který senzory, konkrétně akcelerometr, produkují. Pokud je například mobilní zařízení položeno na pevné podložce, je možné stále si všimnout drobného kmitání. Pokud by byla použita tyto surová data k pohybu nějakého digitálního objektu po obrazovce zařízení, by bylo zřetelné, že by tento pohyb byl velmi trhavý. Použitím právě tohoto filtru lze výsledná data kalibrovat tak, že výsledný pohyb bude plynulý. Tento filtr obsahuje určitou konstantu, kterou lze tuto plynulost ovlivňovat. Více o použití tohoto filtru v aplikaci bude rozebíráno v následujících kapitolách.

6. Návrh a implementace aplikace

Jakou součástí této diplomové práce vzniká aplikace s názvem RoadChecker. Jedná se o aplikaci, která využívá přítomnosti mobilního zařízení v dopravním prostředku. Díky svým sensorům dokáže nejen analyzovat kvalitu silnice, ale i reagovat na extrémní situace, které při řízení dopravního prostředku nastávají.

Vývoj každé aplikace by se měl řídit určitou metodikou. Obecně metodika popisuje postup vývoje aplikace od začátku až po konečnou implementaci. Tato práce částečně vychází z metodiky zvané unifikovaný proces (24). V ní je jako nástroj využíván jazyk UML. Dále budou z této metodiky podrobněji rozebrány následující části:

- Funkční požadavky
- Non-funkční požadavky
- Návrhové modely

Stejně jako většina aplikací má i tato jistou implementační logiku, kterou se řídí. Tato logika je dána ve třech bodech, které budou v dalších kapitolách rozvedeny.

- Získání dat
- Zpracování dat
- Zobrazení výsledku

Tato kapitola obsahuje ukázky kódu, které se přímo vztahují k implementaci. Použití těchto ukázek v kontextu je přiloženo jako součást příloh této diplomové práce.

6.1. Funkční požadavky

Aplikace RoadChecker by měla sloužit jako nástroj pro analýzu kvality silnic. Jejím primárním cílem by mělo být identifikování nerovností na silnici a zjišťování kvality zvolených úseků pozemních komunikací. Pro tuto činnost je třeba, aby aplikace znala následující informace:

1. Velikost sil, které na vozidlo působí
2. Rychlost vozidla

3. Poloha

4. Čas

Velikost sil bude získávána pomocí akcelerometru. Díky tomuto senzoru lze získat velikost sil ve všech třech osách. Síly, které působí horizontálně, se budou analyzovat v osách X, Y. Osa Z bude sloužit primárně pro analýzu otřesů, které nastanou, pokud vozidlo přešlo nějakou nerovnost. Na horizontální úrovni tak bude díky akcelerometru pozorovatelné zrychlení a brzdné účinky automobilu. V zatáčkách se pak bude dávat sledovat odstředivá síla.

Aby mohla být data v těchto osách sbírána, bude muset být provedena kalibrace zařízení pro ustálení aktuální polohy, ve které se bude nacházet. Oproti této kalibrované poloze se budou provádět následující výpočty průměrů a výchylek zaznamenaných akcelerometrem.

Rychlost vozidla se nedá ze senzoru vypočítat, ale pouze odhadnout (25). Z toho důvodu bude nutné, aby zařízení využívalo systém GPS. Rychlost je důležitá pro analýzu výchylek. Předpokládá se, že velikost výchylek se bude měnit spolu s rychlostí vozidla. To znamená, že mezi těmito veličinami bude přímá závislost. Ze systému GPS lze posléze zjistit mimo rychlosti i polohu vozidla. Poloha bude důležitá zejména z důvodu získání informace o tom, kde byla daná výchylka, nerovnost vozovky, zaznamenána.

Veškerá data, která budou aplikací zaznamenána a následně i zpracována budou nahrávána do několika textových souborů v daném formátu. Jako vhodný formát tvaru textu se uvažuje oddělení jednotlivých dat středníkem. Toto je užitečné zejména pro následující zpracování v tabulkovém procesoru, který bude podporovat import dat.

Mimo již zmíněného exportu dat by měla aplikace umět své výsledky i nějakým způsobem reprezentovat. Uvažuje se o využití Google Maps API. To je vhodné zejména pro vytvoření vrstvy nad mapovými podklady, kam se budou vypočítané výchylky graficky zobrazovat. Mimo zobrazení výsledků na mapě by bylo také vhodné zobrazovat aktuální výsledky výpočtů a posléze je i vhodným způsobem graficky znázornit. Toto lze znázornit například využitím grafů, popřípadě využitím knihovny OpenGL ES, která

je součástí Android SDK. Použitím OpenGL by však přineslo větší míru svobody při reprezentování výsledků, než kdyby se využily knihovny pro reprezentaci grafů. Předpokládá se i zvýšení výkonu. Operace zabývající se zobrazováním informací se částečně rozloží mezi grafickou kartou a procesorem zařízení.

6.2. Non-funkční požadavky

Celková implementace aplikace je závislá a tím pádem i omezena platformou, pro kterou je vyvíjena. V tomto případě je aplikace přizpůsobena na použití v mobilních zařízeních se systémem Android. Zprostředkovaně je zde využíváno zdrojů poskytovaných tímto operačním systémem. Celá aplikace je vyvíjena v objektovém jazyce Java, který je doplněn o Android SDK zpřístupňujícím již zmíněné zdroje.

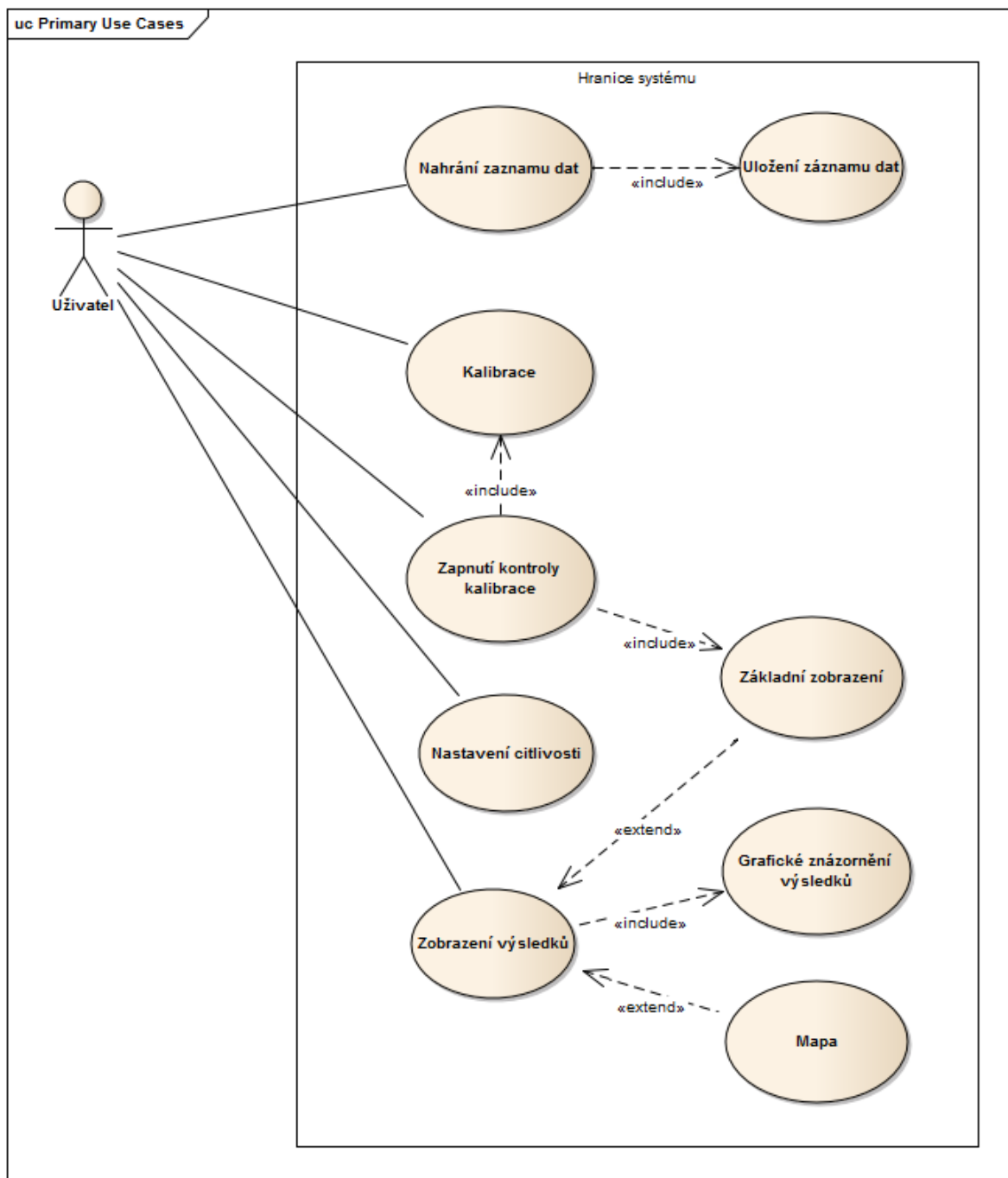
Vzhledem k tomu, že je aplikace vyvíjena pro mobilní zařízení, je vývojář omezen jak výkonem a kapacitou paměti, tak výdrží na baterii. I když výkon v přenosných zařízeních stále roste, je důležité, aby aplikace využívala co nejméně výpočetních prostředků a tím šetřila baterii. Nejvíce bývá vytížena paměť RAM. To je způsobeno knihovnou Google Maps API v2, která si do mezipaměti ukládá mapové podklady. Vzhledem k tomu, že je díky rychlosti získávání dat ze senzoru provedeno mnoho výpočtů v krátký čas, je pravděpodobné, že bude také vytížen procesor. Dá se tedy předpokládat, že aplikace bude vyžadovat rychlejší zařízení.

Aplikace bude vyvíjena jako lokačně závislá, z toho důvodu bude nutné zjišťovat polohu zařízení. Zjištění polohy bude implementováno pomocí GPS systému. Aplikace bude muset vědět, jakou rychlostí se vozidlo pohybuje. K zajištění co nejpřesnějších výsledků je vhodné, aby zařízení podporovalo A-GPS, který dokáže zlepšit zaměření vozidla i v hustě zastavěných oblastech. Zařízení s touto aplikací se bude pohybovat primárně na silnicích, tak se nemusí řešit omezený přístup k signálu v budovách.

Vzhledem k tomu, že bude aplikace zaznamenávat otřesy vozidla, bude nutné, aby mobilní zařízení bylo nějakým vhodným způsobem připevněno ke konstrukci vozidla. Kdyby se tak neučinilo, mohlo by docházet k pohybu zařízení ve vozidle a to by mohlo přinášet zkreslené výsledky.

6.3. Návrhové modely

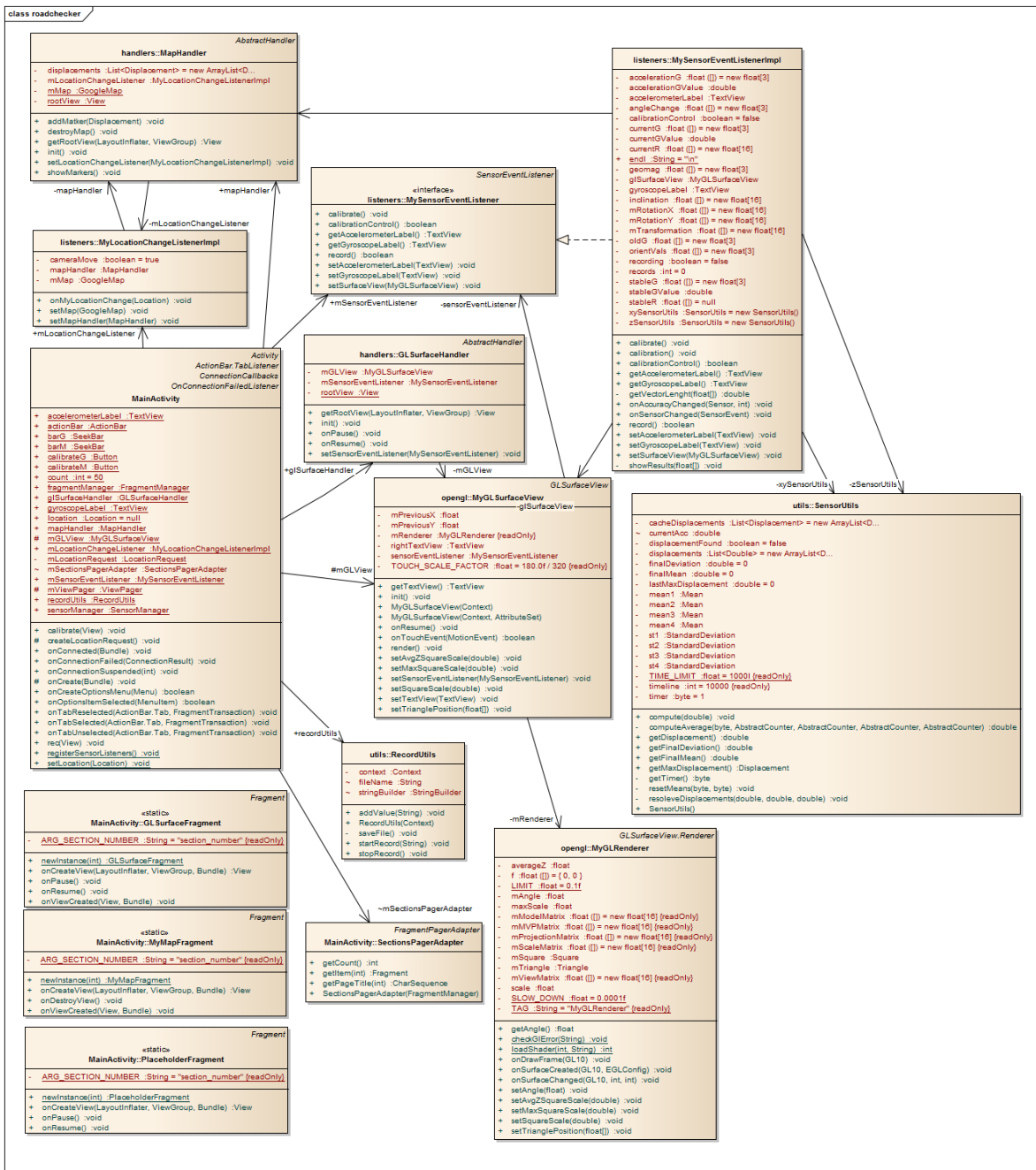
Use case



Obr. 8: Use-case diagram aplikace

Use-case diagram zobrazuje chování systému, tedy aplikace RoadChecker, z pohledu uživatele. Je znázorněn na předchozím obrázku (Obr. 8).

Model tříd



Obr. 9: Class diagram aplikace

Class diagram, neboli diagram tříd (Obr. 9), který je zde zobrazený, zobrazuje jen hlavní třídy, které se v aplikaci nachází. Protože na zobrazení tohoto diagramu velikost stránky nedostačuje, bude přiložen jakou součástí digitální přílohy A, kde bude dostupný v plné kvalitě. Fragmenty, které jsou využity v aplikaci, jsou deklarované jako statické vnitřní třídy v jedné základní aktivitě s názvem MainActivity. Pro přehlednost byly v diagramu vyčleněny.

6.4. Získání dat

Získání dat je první věcí, kterou musí vývojář při implementaci své aplikace provést. Kdyby tak neučinil, neměl by s čím provádět výpočty a posléze ani nic, co vy mohl ve svém uživatelském prostředí zobrazit.

Aplikace RoadChecker je primárně pracuje s těmito daty:

- Data ze sensorových událostí
 - Akcelerometr
 - Magnetometr
- GPS data
 - Aktuální poloha
 - Rychlost

Získání dat ze senzorů

Protože se aplikace převážně soustředí na práci se sensorovými daty, bylo pro její potřeby nutné vytvořit nový interface `MySensorListener`, který rozšiřuje ten původní o několik nových metod. Za zmínku slouží dvě nejdůležitější:

- `public void calibrate ()`
- `public boolean record ()`

Jak již z jejich názvu vyplývá, dovolují tomuto listeneru data, která poskytuje kalibrovat a také nahrávat.

Tento listener je implementován tak, aby zvládl pracovat jak s daty z magnetometru, tak s daty akcelerometru. Jeho použití je znázorněno v předchozí ukázce (Kód 7). To poskytuje vývojáři určité výhody. Pokud je například třeba provést zrušení registrace listeneru sensoru, volá se tato metoda pouze jednou. Dále si lze v kódu všimnout podmínky, která ověřuje přítomnost daného typu sensoru na zařízení. Data z obou senzorů jsou pak nadále v tomto listeneru zpracovávána a poskytována dále k zobrazení.

```

MySensorEventListener mSensorEventListener =
    new MySensorEventListenerImpl();

public static void registerSensorListeners() {
    if(sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)!=null){

        sensorManager.registerListener(mSensorEventListener,
            sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER),
            SensorManager.SENSOR_DELAY_UI);
    }
    if(sensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD)!=null){

        sensorManager.registerListener(mSensorEventListener,
            sensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD),
            SensorManager.SENSOR_DELAY_UI);
    }
}
}

```

Kód 7: Použití vlastního sensor event listeneru

Získání dat z GPS

Data z GPS jsou zprostředkována prostřednictvím knihovny Google Maps API v2. Ta nejen že zpřístupňuje mapové podklady, ale vytváří také vrstvu nad službou poskytující GPS. Zde je využita implementace vlastního listeneru pro naslouchání změny lokace, v níž se aktuálně zařízení nachází. Aktuální data, která jsou přes službu GPS poskytnuta, jsou obalena třídou Location. Skrze ní lze přistupovat k údajům, jako jsou například aktuální poloha zařízení, nebo aktuální rychlost, kterou se zařízení pohybuje. Poloha je poskytována v zeměpisných souřadnicích a rychlost v jednotkách metr za sekundu (m/s).

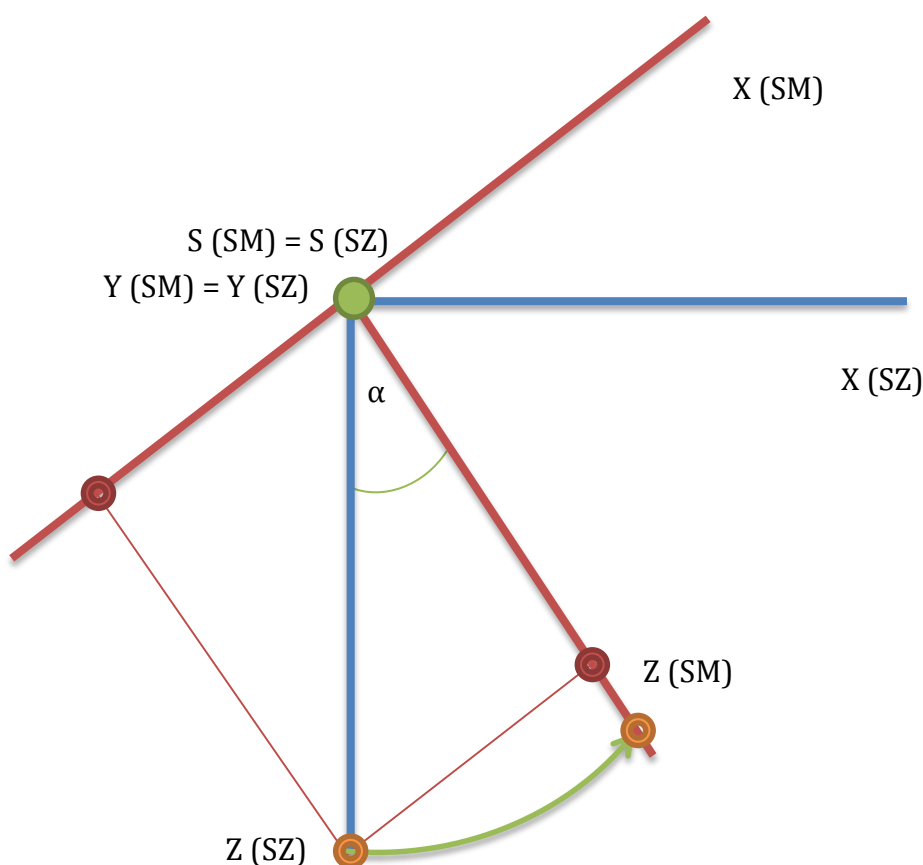
6.5. Výpočty v aplikaci

Vyrovnání náklonu zařízení

Jsou dány dva souřadnicové systémy. Souřadnicový systém Země (dále jen SZ) a souřadnicový systém mobilního zařízení (dále jen SM). Aby se daly zjišťovat velikosti výchylek akcelerometru při detekci výmolu na vozovce, je nutné zaznamenávat data na ose Z v soustavě SZ. Data jsou však poskytována v soustavě SM, která je oproti té druhé

natočená o určitý prostorový úhel. K získání potřebných dat je tedy nutné provést jejich transformaci z jedné soustavy souřadnic do druhé.

Tento převod je dán celkem dvěma transformacemi. Je vlastně nutné SM horizontálně vycentrovat. To je provedeno natočením vektoru akcelerace skládajícího se ze složek X, Y, Z získaného ze soustavy souřadnic SM o zápornou hodnotu úhlu, o který je soustava souřadnic SM natočena proti SZ. Předpokládá se zde, že obě tyto osy mají stejný počátek souřadnic. Pro účely aplikace stačí provést natočení podél os X a Y. Tím se dojde k vycentrování horizontu a osa Z pak ukazuje ve stejném směru, jako osa Z ze soustavy SZ. Otočení okolo osy Z není potřebné. Kdyby bylo provedeno, došlo by k synchronizování vertikálního natočení podle geomagnetického pole země. Osa Y zařízení by tedy směřovala k severu.



Obr. 10: Znázornění výpočtu

Obrázek výše znázorňuje výpočet (Obr. 10). Pro zjednodušení je ukázáno jen otočení kolem osy Y, která směřuje k pozorovateli. Červená barva je zvolena pro

soustavu souřadnic zařízení. Modrá barva naopak pro soustavu souřadnic reálného světa. Jsou oproti sobě navzájem natočeny o úhel alfa.

Příklad:

Na zařízení působí gravitační síla Země o velikosti zhruba 9.81. Zařízení je natočeno kolem osy Y o úhel 30 stupňů. Z toho vyplývá, že na zařízení je zobrazen vektor, který má souřadnice (X; Y; Z) = (4,9; 0; 8,5). Pokud je tento vektor vynásoben maticí rotace o úhel mínus alfa, tak výsledný vektor bude mít hodnoty b (0; 0; 9,81). Osa X zařízení se kalibrovala do horizontální polohy a výsledné gravitační zrychlení Země je získáno na ose Z. Výpočet znázorněn v následující rovnici.

$$\begin{pmatrix} 4,9 \\ 0 \\ 8,5 \\ 1 \end{pmatrix} * \begin{pmatrix} \cos(-30^\circ) & 0 & \sin(-30^\circ) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-30^\circ) & 0 & \cos(-30^\circ) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 9,81 \\ 1 \end{pmatrix}$$

Podobně, jako v předchozí ukázce příkladu, je tento problém řešen v aplikaci. V ní však se provádějí transformace podle os X a Y, přičemž je nutné dbát na pořadí těchto operací. Ukázka implementace je znázorněna níže (Kód 8). V ní je jako podpora při skládání transformací použita třída Matrix.

```
SensorManager.getRotationMatrix(currentR, inclination, currentG, geomag);
SensorManager.getOrientation(currentR, orientVals);

Matrix.setIdentityM(mTransformation, 0);
Matrix.setIdentityM(mRotationX, 0);
Matrix.setIdentityM(mRotationY, 0);

Matrix.setRotateM(mRotationX, 0,
    Math.round(-Math.toDegrees(orientVals[1])), 1f, 0f, 0f);
Matrix.setRotateM(mRotationY, 0,
    Math.round(Math.toDegrees(orientVals[2])), 0f, 1f, 0f);

Matrix.multiplyMM(mTransformation, 0, mRotationX, 0, mRotationY, 0);
Matrix.multiplyMV(result, 0, mTransformation, 0,
    new float[]{accelerationG[0], accelerationG[1], accelerationG[2], 1}, 0);
```

Kód 8: Vyrovnání náklonu zařízení

Low-pass filtr

Jak bylo již zmíněno, aplikace RoadChecker obsahuje low-pass filtr. Využívá ho primárně k redukci šumu a k zvýšení plynulosti při reprezentaci zaznamenaných dat. Díky něho je také docíleno ustálení výchylek, které se vyskytují v klidovém stavu.

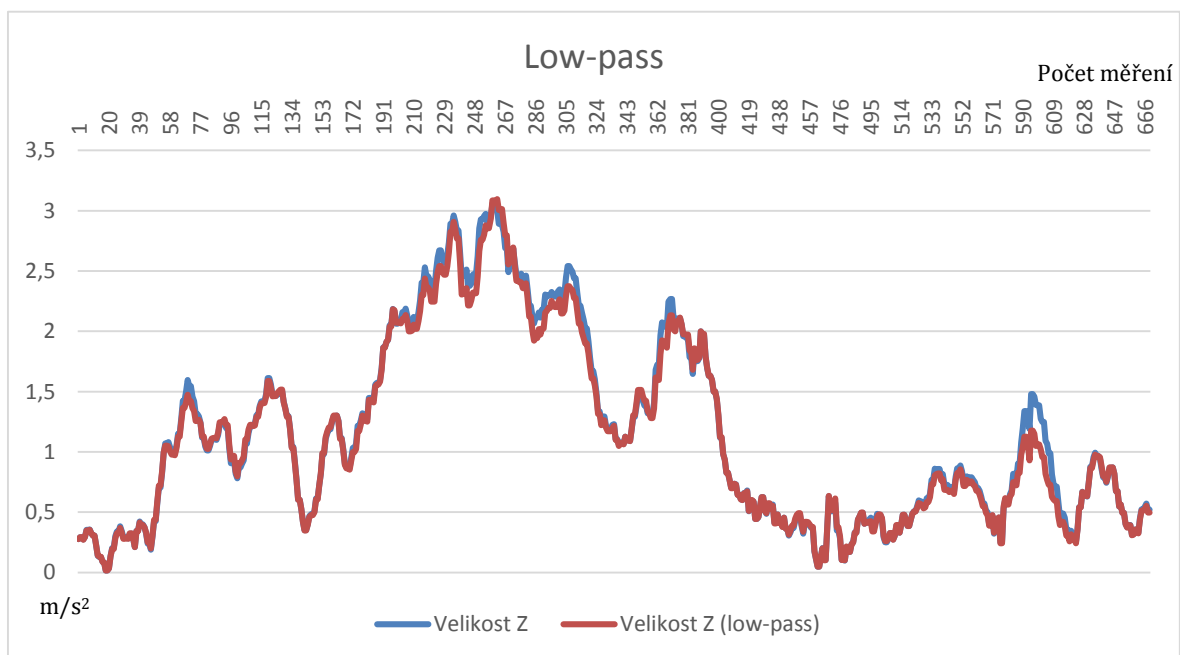
```
public static float[] filter(float[] input, float[] old) {
    if (old == null) return input;

    for (int i = 0; i < input.length; i++) {
        old[i] = old[i] + ALPHA * (input[i] - old[i]);
    }
    return old;
}
```

Kód 9: Low-pass filtr (21)

Zdrojový kód (Kód 9), který je zobrazen výše, zobrazuje podobu, jak je algoritmus implementován v aplikaci. Ke své činnosti potřebuje tři informace:

- Nová data ze senzoru
- Velikost výsledku, který byl vypočítán při minulé sensorové události
- Parametr alfa

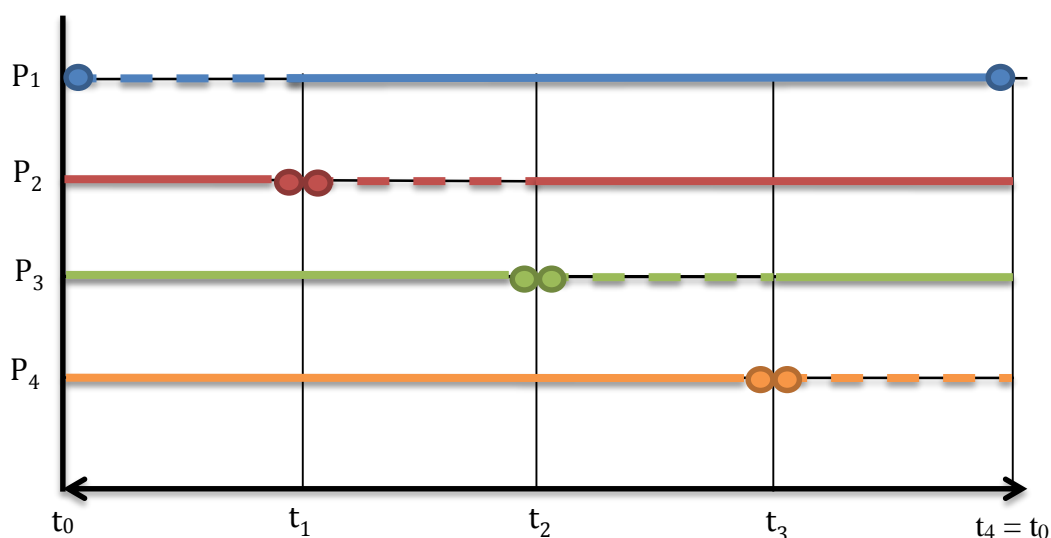


Obr. 11: Graf znázorňující zpracování low-pass filtrem (alfa = 0.181)

Parametr ALFA znázorňuje jakýsi vztah mezi oběma čísly. Udává, jak hodně jsou vyhlazená data ovlivněna novou hodnotou. Čím je tato konstanta menší, tím jsou data více vyhlazená. Pro využití tohoto filtru v aplikaci byla zjištěna kritická hodnota 0,1. Když Parametr ALFA klesne pod tuto hodnotu, výrazně klesá pravděpodobnost, že aplikace detekuje nějakou výchytku. Jak vypadají data, která prošla tímto filtrem oproti normálním, je znázorněno na obrázku níže.

Výpočet průměrných hodnot otřesů

V aplikaci RoadChecker není možné použít základní průměr ze všech dat, která jsou zaznamenána. Kdyby se toto použilo, po nějaké době by i sebevětší výchytky v datech měly na průměr mizivý dopad. Sensory produkují velké množství dat v krátké době. Z toho důvodu by tedy reálně hrozilo přetečení zásobníku, který má aplikace k dispozici. Z toho důvodu byla v aplikaci použita modifikovaná podoba plovoucího průměru. Jeho hlavní výhodou je rychlé přizpůsobení datům, která jsou mu poskytována.



Obr. 12: Graf využívaných průměrů v čase t

Plovoucí průměr funguje tak, že každý určitý časový okamžik se začne průměr počítat znovu. Průměr implementovaný v této aplikaci funguje podobně. Muselo se však dbát na to, že v první chvíli, kdy je průměr počítán, je velice citlivý na výchytky a

to by mohlo negativně ovlivnit konečnou analýzu. Aby byly po celou dobu sledování získávány rovnoměrné výsledky, jsou v jednu dobu vypočítávány čtyři průměry – na obrázku (Obr. 12) $P_{1...4}$. K jejich vynulování dochází ve vyznačených bodech. Pro každý z průměrů tato operace nastává v jiný časový okamžik. K získání výsledné hodnoty průměru je vypočítán průměr ze tří dílčích průměrů. Ten čtvrtý má čas, aby stihl posbírat od svého vynulování dostatek dat, a díky tomu mohl podávat vyvážené výsledky. Na obrázku je tento čas znázorněn přerušovanou čarou u každého průměru. Výpočty výsledného průměru jsou uvedeny v následující tabulce (Tab. 2).

| Časový interval | Výpočet průměru P |
|-----------------|-----------------------------|
| $(t_0; t_1)$ | $P = (P_2 + P_3 + P_4) / 3$ |
| $(t_1; t_2)$ | $P = (P_1 + P_3 + P_4) / 3$ |
| $(t_2; t_3)$ | $P = (P_1 + P_2 + P_4) / 3$ |
| $(t_3; t_0)$ | $P = (P_1 + P_2 + P_3) / 3$ |

Tab. 2: Výpočet průměru

Pro implementaci tohoto algoritmu je důležitý časovač. Reálný čas musí být rozdělen do sekvencí po čtyřech stejně dlouhých intervalech. Délka časového intervalu je dána konstantou. V budoucích verzích aplikace by však šla tato konstanta nastavovat podle rychlosti vozidla. Pro účely této diplomové práce však byl zvolen čas deset sekund, který se pro výsledky testů ukázal jako dostačující. Jeho implementace je ukázána v následujícím kódu (Kód 10). Výsledek tohoto algoritmu je zobrazená formou grafu (Obr. 13).

```

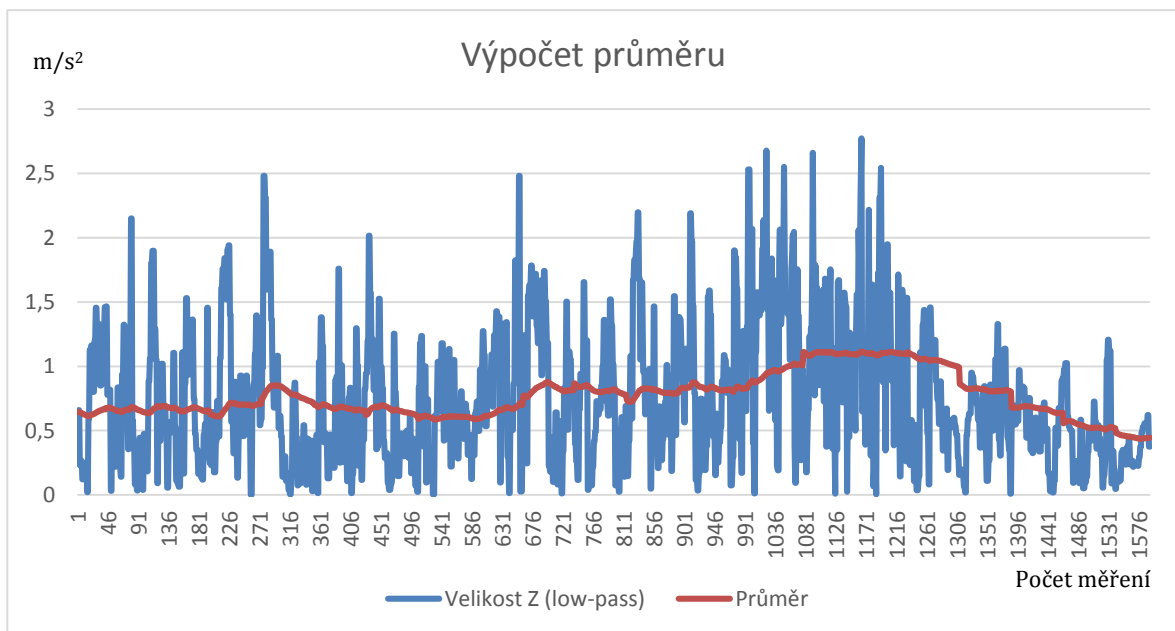
private final int timeline = 10000;

private byte getTimer() {
    long currentTime = System.currentTimeMillis() % timeline;
    byte timer = 1;

    if (currentTime > 0) {
        timer = 1;
    }
    if (currentTime > timeline / 4) {
        timer = 2;
    }
    if (currentTime > timeline / 2) {
        timer = 3;
    }
    if (currentTime > timeline / 4 * 3) {
        timer = 4;
    }
    return timer;
}

```

Kód 10: Rozdělení sekvencí



Obr. 13: Reprezentace výpočtu průměru

Výpočet výchylek

Analýza výchylek je další ze stěžejních částí aplikace RoadChecker. Bylo zjištěno, že pro spolehlivou detekci výmolů postačuje rychlost obnovy senzoru daná hodnotou `SensorManager.SENSOR_DELAY_UI`. Jako první věcí, kterou je potřeba provést, je kalibrace získaných dat. Následně je z nich vytvořen plovoucí průměr (přesněji popsán v předchozí kapitole). Tento průměr je pro výpočet výchylek důležitý. Po několika sériích pokusů bylo zjištěno, že tento průměr se dokáže automaticky přizpůsobit velikosti dat, která pochází ze senzoru. Pokud se automobil pohybuje rychleji, jsou data, která senzor poskytuje více výrazná a i průměr je díky tomu větší. Z toho vyplývá, že pomocí tohoto průměru se dokáže aplikace při určování výchylek přizpůsobit rychlosti vozidla. Dále bylo z pokusů zjištěno, že pokud je aktuální akcelerace větší než průměr zvětšený o konstantu velikosti 1 m/s^2 , jedná se o výchylku. Díky tomu, že se rychlosti přizpůsobuje průměr, je tato konstanta neměnná. To bylo ověřeno zhruba třiceti pokusy pro rychlosti 10 – 110 km/h. Výsledek je znázorněn na následujícím grafu (Obr. 16). Výchylky jsou zde reprezentovány fialovou barvou. Z předchozího poznání vyplývá, že čím je automobil rychlejší, tím jsou výchylky výraznější. Tomu se



Obr. 14: Oblast, kde byla zaznamenána data pro následující ukázkou analýzy (27)

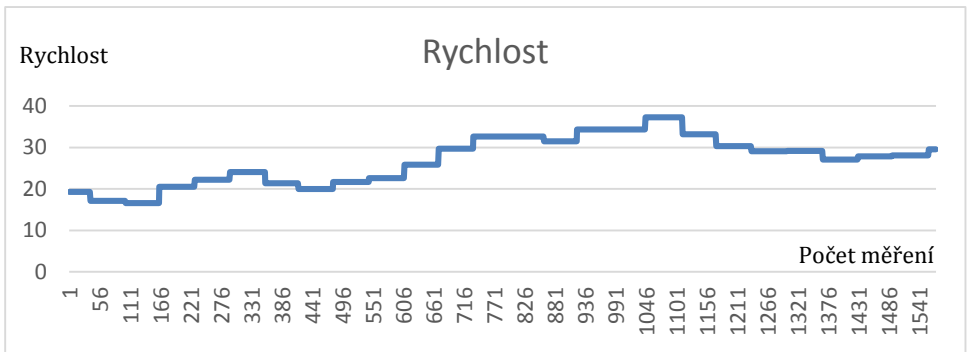
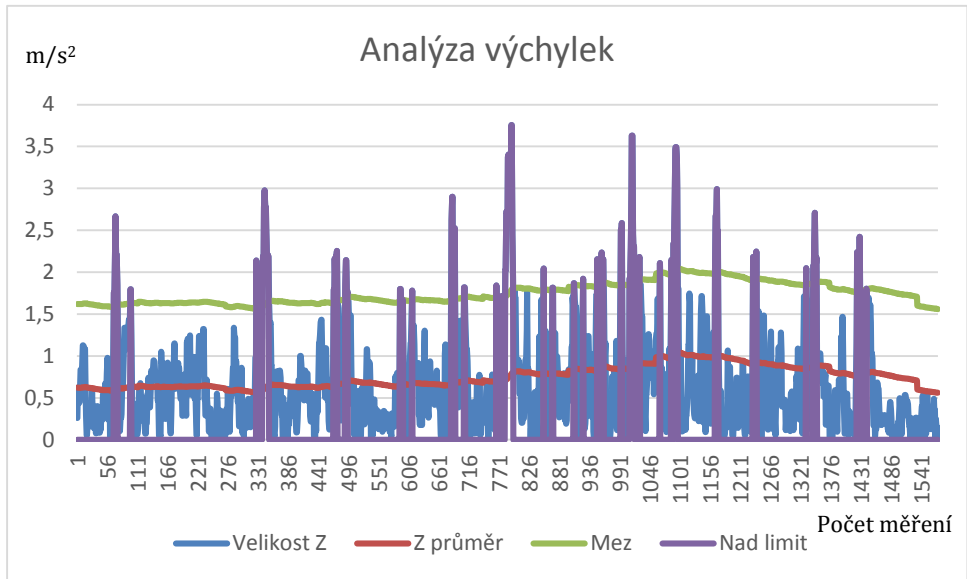
následně přizpůsobuje průměr. Pokud je rychlost nižší, než 10 km/h. Jede automobil tak pomalu, že výchyly splynou s šumem, který je následně utlumen low-pass filtrem.

Každý automobil má ale dvě nápravy. To znamená, že pokud vozidlo vjede na nějaký výmol, jsou zaznamenány dvě výchyly, každá na jednu nápravu. To je řešeno následovně. Je vytvořena časová konstanta, závislá na rychlosti určující, kolik času je třeba, aby došlo k zobrazení dvou po sobě jdoucích výchylek.

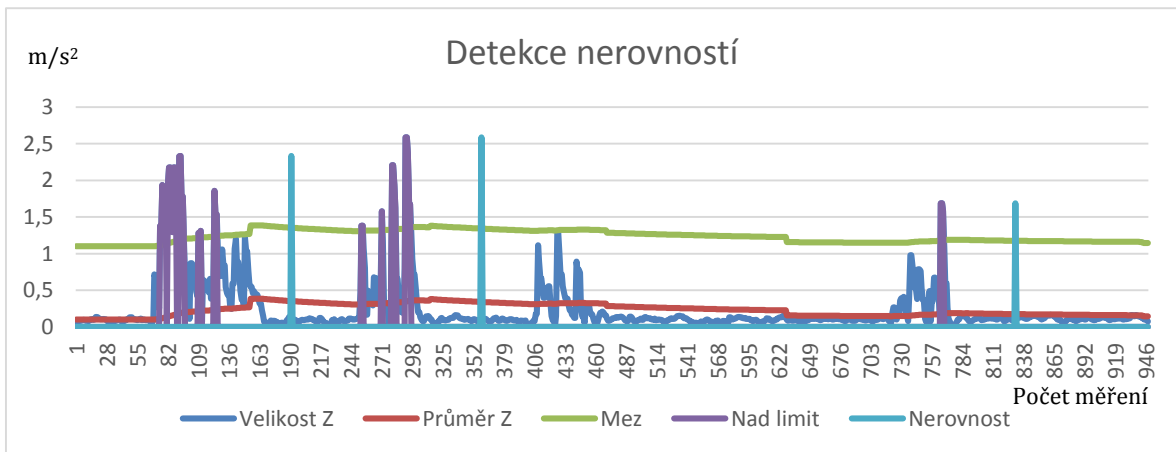
Příklad:

Je zaznamenána výchyly a je uložena do cash paměti aplikace. Pokud dojde do doby, která je dána konstantou, k zaznamenání další výchyly, je opět uložena. Když do doby delší, než hodnota reprezentovaná konstantou, žádná další výchyly nenastane, je velikost výchylek v cash porovnána a následně je ta největší předána dále ke zpracování. Jejich detekce je znázorněna v následujícím grafu modrou barvou (Obr. 16). Doba mezi skupinou fialových a jednou modrou výchyly je čas, který aplikace čekala na další nerovnosti. Jak bylo zmíněno dříve, jedná se o konstantu ovlivněnou rychlostí. Tato modrá výchyly je posléze zobrazena na mapě jako nerovnost. Pro výchyly byla vytvořena vlastní třída, která obsahuje podstatné údaje pro výpočty.

- Velikost
- GPS Location
 - Rychlost
 - Poloha
- Čas v milisekundách



Obr. 16: Vztah rychlosti na analýzu výchylek



Obr. 16: Grafické znázornění detekce výmolů

6.6. Zobrazení výsledů

Celý uživatelský interface je postaven na fragmentovém modelu využívajícím swipe view. To znamená, že aplikace obsahuje jen jednu aktivitu, MainActivity, která je dále rozdělena na tři fragmenty. Každý z těchto fragmentů dokáže pracovat nezávisle na jiných a představuje jednu obrazovku z aplikace. Swipe view znamená v překladu posuvný pohled. Jedná se převážně o to, že uživatel může mezi jednotlivými fragmenty přecházet pouhým přejetím prstu.

Zobrazení dat ze senzorů

Jedná se o první z fragmentů. Tento pohled slouží k zobrazení výsledků výpočtů a může sloužit pro kontrolu vypočítávaných dat. Obsahuje jak hrubá, tak kalibrovaná data ze senzorů. Je ukázán na následujícím obrázku (Obr. 17).

Lze z něj vyčíst tyto informace:

- Data z akcelerometru
- Kalibrovaná data z akcelerometru
- Kalibrované zrychlení
- Náklon zařízení ve stupních oproti souřadnicovému systému Země uložený při kalibraci
- Náklon zařízení ve stupních oproti kalibrované hodnotě
- Aktuální rychlost
- Velikost alfa konstanty

Na obrázku si lze také všimnout kalibrované hodnoty akcelerometru, kde se gravitační zrychlení země rozloženého do všech os soustředilo na osu Z.

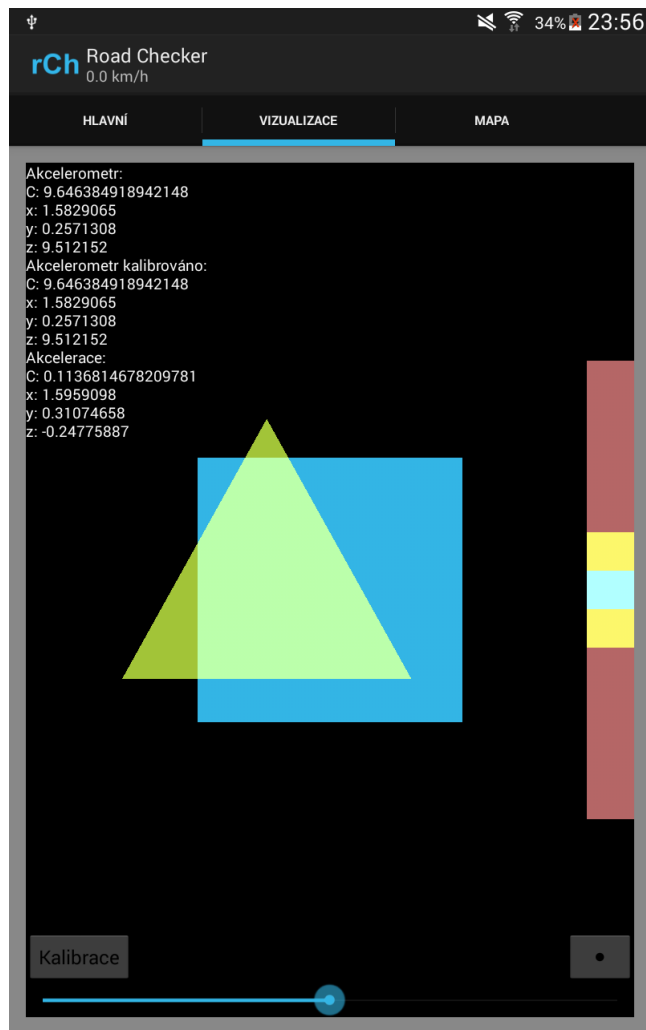


Obr. 17: Ukázka prvního fragmentu

Grafické znázornění výsledků výpočtů

Druhý fragment slouží ke grafické reprezentaci výsledů. Byl vytvořen převážně z důvodu ověření správnosti výsledků a k usnadnění představy výpočtů, které jsou na pozadí prováděny. Grafické rozhraní je zprostředkováno knihovnou OpenGL. Konkrétně se jedná o verzi GL ES 2.0, která je dodávána jako součást Android SDK.

Pro inicializaci tohoto rozhraní byl vytvořen handler (`GLSurfaceHandler`), který dokáže, při spuštění konkrétního fragmentu, vytvořit instanci `GLSurfaceView`. Nebylo však využito defaultního view. Byla vytvořena třída `MyGLSurfaceView`, která toto view rozšiřuje a přidává některé prvky, které jsou v aplikaci využívány. Na obrázku (Obr. 18) je toto view reprezentováno černou barvou. Stručněji řečeno slouží v aplikaci jako kreslicí plátno, na které jsou renderovány objekty.



Obr. 18: Ukázka zobrazení skrz MyGLSurfaceView na druhém fragmentu

K renderování objektů je využita třída MyGLRenderer, jejíž instance je vytvořena obsazena ve třídě MyGLSurfaceView. Ta zároveň slouží i jako vrstva, přes kterou lze k tomuto rendereru přistupovat a předávat parametry objektů, které jsou vykreslovány. Renderer umí znázornit dva typy objektů:

- Trojúhelník
- Čtverec

Tyto objekty lze libovolně transformovat pomocí transformačních matic. Konkrétně je využity následující transformační matice:

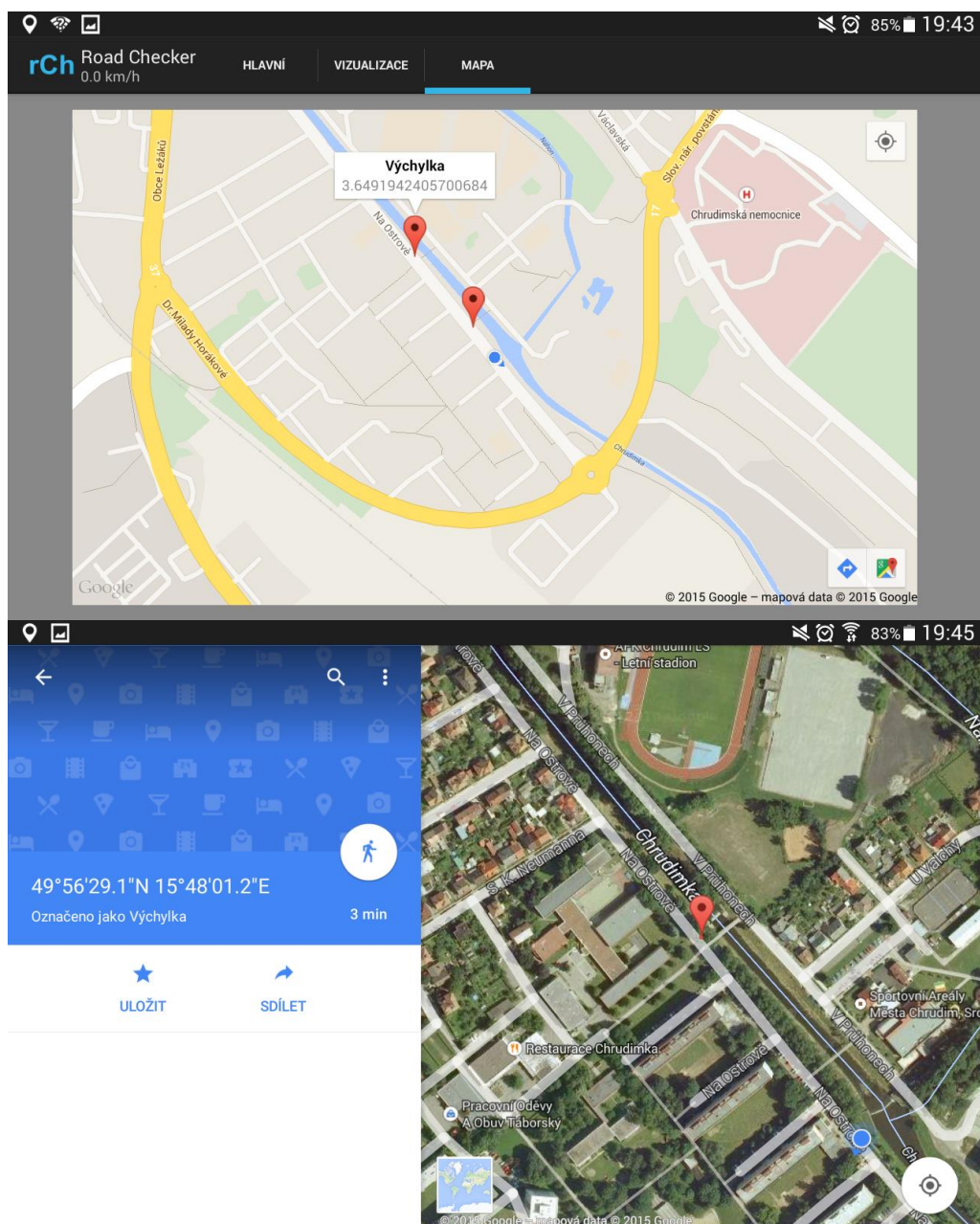
- Posunutí
- Změna měřítka

Posunutí je použito při znázornění horizontální (osy X, Y) odstředivé síly, která může působit na zařízení při pohybu vozidla. Na obrázku je toto znázorněno zeleným trojúhelníkem. Změna měřítka je využita u čtverců. Červený obdélník, transformovaný čtverec, na kraji znázorňuje velikost poslední výchylky, která byla zaznamenána. Aktuální síly působící vertikálně (osa Z) jsou reprezentovány žlutým obdélníkem na kraji obrazovky. Modré objekty reprezentují velikost plovoucího průměru v horizontální i vertikální úrovni, jehož výpočet byl podrobněji vysvětlen v předchozích kapitolách.

Zobrazení výsledku na mapě

Výsledky měření jsou zobrazovány pomocí Google Maps API v2 na mapách. Pokud dojde v aplikaci k detekci výmolu nebo nějaké jiné nerovnosti, je na mapu umístěna značka. Ta je vytvořena pomocí třídy Marker. To s sebou nese několik výhod. Například, pokud na něj uživatel klikne, je mu zobrazena nabídka, jestli do tohoto místa chce navigovat, nebo jestli chce toto místo zobrazit přes Google Maps (Obr. 19). Díky tomu si ho lze prohlédnout například přes StreetView.

View pro tento fragment je poskytováno přes třídu MapHandler. Mimo to, se v této třídě provádí inicializace mapy a nasektování pro ní příslušných listenerů. Za zmínku stojí například vlastní implementace LocationListeneru, který poskytuje údaje o současné poloze zařízení. Dále jsou v této přítomny metody na přidávání a zobrazování značek. Hned za začátku běhu aplikace lze přidávat detekované výchytky. Pokud je zrovna není možné zobrazit, jsou uloženy do mezipaměti a jejich vykreslení proběhne, až to bude možné.



Obr. 19: Zobrazení třetího fragmentu a jeho návaznosti na Google Maps

7. Shrnutí výsledků a závěr

Hlavním cílem této práce bylo analyzovat možnosti využití senzorů mobilního zařízení v dopravě a na základě zjištěných poznatků implementovat aplikaci. Byly zde využity zkušenosti z vývoje předchozích aplikací. Tyto znalosti byly v průběhu vývoje aplikace aplikovány k řešení několika problémů, které tento vývoj doprovázely.

Na začátku této práce bylo obecné seznámení s problematikou senzorů a to nejen mobilních zařízení. Byly zde primárně popsány druhy senzorů, které jsou v dnešní době nejvíce používané. Následně bylo uvedeno několik dalších senzorů, a to buď více ojedinělých, nebo nacházejících se v určitém stádiu vývoje. Následně byla provedena analýza jejich využití. Na základě využívání senzorů pro vozidla, bylo uvažováno nad možným využitím mobilních senzorů. Bylo zjištěno, že mobilní zařízení v automobilovém odvětví použít lze. K tomu, aby podávalo věrohodné výsledky, však musí být pevně připevněno ke konstrukci vozidla.

Dalším cílem bylo seznámení a pochopení toho, jak lze vyvíjet aplikace na zařízení vybaveném operačním systémem Android. K tomu slouží konkrétně Android Sensor Framework, který byl dopodrobna popsán. Tento popis byl doplněn o několik názorných ukázek kódu. Od této kapitoly se pomalu přešlo k rozboru toho, jak funguje kalibrace a jaká je přesnost senzorů v mobilním zařízení. Z těchto poznatků se vycházelo právě při implementaci aplikace.

Jak bylo již řečeno, v rámci této diplomové práce byla vytvořena aplikace pro mobilní zařízení s operačním systémem Android s názvem RoadChecker. Tato aplikace využívá senzory mobilního zařízení k analyzování nerovnosti na silnicích. K tomuto účelu slouží především akcelerometr, který je velmi citlivý a při správné kalibraci dokáže zaznamenat velikost otřesů.

Do aplikace byla implementována kalibrace do ustálené polohy pomocí transformací soustavy souřadnic. Pro vyhlazení dat, která jsou později zpracovávána, byl vytvořen low-pass filtr. Pro běh aplikace byl důležitý výpočet modifikovaného plovoucího průměru. Ten je schopen plynulého přizpůsobení se rychlosti vozidla a díky tomu reagovat i na změnu povrchu vozovky. Z průměru bylo vycházeno při detekci

výchylek. Po jejich zaznamenání byl implementován mechanismus, který z nich vybral ty podstatné. Posléze jsou zobrazeny na mapě. Zobrazení aplikace RoadChecker obsahuje také grafické vyjádření výpočtů pomocí knihovny OpenGL ES. Tyto výpočty lze samozřejmě zobrazit i formou čísel. To slouží převážně pro kontrolu správnosti výsledků.

Mimo podstatných funkcí poskytuje aplikace nástroje pro provedení analýzy dané komunikace. Lze si například nastavit velikost, jak má být aplikace citlivá. Vypočtené výsledky si může uživatel nahrát a uložit ve formě textového souboru, který je dále zpracovatelný v tabulkových procesorech. Implementace aplikace obsahuje názorné ukázky zdrojových kódů a jejich využití přímo v aplikaci. Tyto ukázky jsou také doplněny mnoha grafy vzniklými analyzováním již zmíněných textových souborů.

Bylo provedeno mnoho testů ověřujících funkčnost aplikace. Zjistilo se, že je schopna spolehlivě detekovat nerovnosti, které se na pozemních komunikacích vyskytují. Tato aplikace by se dala dále rozvíjet a rozšiřovat o nové funkce. Dat, která jsou jí poskytována a která zpracovává, je mnoho. Lze například uvést, že je schopna detekovat odstředivé síly působící na řidiče při průjezdech zatáčkou. V aplikaci je tato detekce zobrazena pouze jako grafický doplněk při zobrazování výsledků. Tato funkcionality však není nijak dále zpracovávána. Další možností, jak vylepšit aplikaci, by bylo zvýšení její přesnosti. Ta by šla zvýšit přidáním možnosti volby a ukládání profilů pro různé automobily. Zde by šlo uvažovat i o tom, že by se aplikace mohla automaticky učit, jak se dané vozidlo pohybuje a jakou citlivost má pro vozidlo použít.

Aplikace by se dala využít jako prostředek pro detekci špatných úseků silnic. Pokud by nějaká organizace měla systém pro sledování kvality vozovky, pak by tato aplikace mohla vhodně sloužit pro sběr údajů. Po přidání podpory pro komunikaci s tímto systémem by pak každý motorista mohl pomocí svého telefonu pomoci při analýze výmolů. Pokud by systém zjistil, že je v jednom místě neustále hlášena nějaká výchylka, mohl by poskytnout podnět pro opravu daného úseku správou silnic.

Seznam použité literatury a zdrojů

1. **Škopek, Pavel.** Techbox: váš telefon je prošpikovaný senzory. *mobilenet.cz*. [Online] 17. 7 2013. [Citace: 15. 4 2015.] <http://mobilenet.cz/clanky/techbox-vas-telefon-je-prospikovany-senzory-12496>.
2. **Poslad, Stefan.** *Ubiquitous Computing: smart device, enviroment, and interactions*. Chichester : John Wiley & Sons Ltd, 2009. ISBN 978-0-470-03560-3.
3. **Google Inc a Open Handset Alliance.** Sensors Overview. *Android Developers*. [Online] 2015. [Citace: 16. 4 2015.] http://developer.android.com/guide/topics/sensors/sensors_overview.html.
4. **Gohring, Nancy.** Your Android Phone Could Help Scientists Predict Your Weather. *WIRED*. [Online] 7. 1 2013. [Citace: 15. 4 2015.] <http://www.wired.com/2013/01/android-weather-prediction>.
5. **Hecl, David.** Jak funguje digitální gyroskop? Princip toho, proč iPhone dokáže určit, v jaké je poloze. *Letem světem Applem*. [Online] 23. 1 2014. [Citace: 15. 4 2015.] <http://www.letemsvetemapplem.eu/2014/01/23/jak-funguje-digitalni-gyroskop-princip-toho-proc-iphone-dokaze-urcit-v-jake-je-poloze/>.
6. **Tennoe, Mariam T., Surhone, Lambert M. a Henssonow, Susan F.** *Webots*. Bean Bassin, Mauritius : VDM Publishing House Ltd., 2010. ISBN: 978-613-0-54624-3.
7. **Šmíd, Josef.** Senzory v mobilních telefonech od A do Z. *Beryko.cz*. [Online] Beryko.cz, 29. 11 2014. [Citace: 15. 4 2015.] <http://www.berkyo.cz/blog/recenze/senzory-v-mobilnich-telefonech-od-a-do-z.html>.
8. **Srb, Jindřich.** Magnetometr – technologie pro mobilní telefony. *Mobilizujeme.cz*. [Online] 22. 1 2012. [Citace: 2015. 4 2014.] <http://mobilizujeme.cz/clanky/magnetometr-nova-technologie-pro-mobilni-telefony/>.

9. **STMicroelectronics.** LIS3MDL ultra low-power, high performance 3-axis magnetometer. *STMicroelectronics*. [Online] 2015. [Citace: 22. 4 2015.] http://www.st.com/web/catalog/sense_power/FM89/SC1449/PF255198.
10. **Scienceworld.** Biometrické senzory se zmenšují a zlevňují. *Scienceworld*. [Online] 16. 7 2006. [Citace: 15. 4 2015.] <http://www.scienceworld.cz/clovek/biometricke-senzory-se-zmensuji-a-zlevnuji-1372/>.
11. **ABBAS, a.s.** Biometrie otisku prstu. *biometricke-ctecky.cz*. [Online] ABBAS, a.s., 2015. [Citace: 15. 4 2015.] <http://www.biometricke-ctecky.cz/biometriky/otisk-prstu/>.
12. **Myslivoček, David.** Telefon Samsung Galaxy S5 – evoluce, nikoli revoluce (recenze). *svetandroida.cz*. [Online] 1. 4 2014. [Citace: 22. 4 2015.] <http://www.svetandroida.cz/samsung-galaxy-s5-recenze-201404>.
13. **SamsungTomorrow.** What You May Not Know About GALAXY S4 Innovative Technology. *samsungtomorrow.com*. [Online] 10. 4 2013. [Citace: 23. 4 2014.] <https://www.mojandroid.sk/samsung-galaxy-s4-blizsi-pohlad-na-9-senzorov/>.
14. **Vysoký, Petr.** Asistenční systémy v automobilech. *Časopis Automa*. [Online] MPresent, 2007. [Citace: 15. 4 2015.] http://automa.cz/index.php?id_document=30855.
15. **Conner, Margery.** Automobile sensors may usher in self-driving cars. *EDN*. [Online] 26. 5 2011. [Citace: 15. 4 2015.] <http://www.edn.com/design/automotive/4368069/Automobile-sensors-may-usher-in-self-driving-cars>.
16. **Troppmann, Rainer, a další.** Tech Tutorial: Driver Assistance Systems, an introduction to Adaptive Cruise Control. *EE Times*. [Online] 23. 6 2006. [Citace: 4. 16 2015.] http://www.eetimes.com/document.asp?doc_id=1272754.
17. **Sajdl, Jan.** ABS (Anti-lock Braking System). *autolexicon.net*. [Online] 2015. [Citace: 23. 4 2015.] <http://cs.autolexicon.net/articles/abs-anti-lock-braking-system/>.

18. **Dekra CZ a.s.** Stručný přehled zkratk používaných v automobilovém PRŮMYSLU. *dekra-automobil.cz*. [Online] 2009. [Citace: 23. 4 2015.] <http://www.dekra-automobil.cz/index.php?file=zkratky.php>.
19. **Audi, AG.** Audi connect. *Audi Deutschland*. [Online] 2015. [Citace: 16. 4 2015.] <http://www.audi.de/de/brand/de/neuwagen/layer/connect/audi-connect-en.html>.
20. **Google Inc a Open Handset Alliance.** SensorManager. *Android Developers*. [Online] 2015. [Citace: 16. 4 2015.] http://developer.android.com/guide/topics/sensors/sensors_overview.html.
21. —. Motion Sensors. *Android Developers*. [Online] 2015. [Citace: 16. 4 2015.] http://developer.android.com/guide/topics/sensors/sensors_motion.html.
22. —. Position Sensors. *Android Developers*. [Online] 2015. [Citace: 16. 4 2015.] http://developer.android.com/guide/topics/sensors/sensors_position.html.
23. **MathWorks.** Low-Pass Filter. *MATLAB*. [Online] MathWorks, 2015 . [Citace: 16. 4 2015.] <http://www.mathworks.com/discovery/low-pass-filter.html>.
24. **Arlow, Jim a Neustadt, Ila.** *UML2 a unifikovaný proces vývoje aplikací*. Brno : Computer Press, a.s., 2011. ISBN 978-80-251-1503-9.
25. **Meier, Reto.** *Professional Android 2 Application Development*. Indianapolis, Indiana : Wiley Publishing, Inc., 2010. ISBN: 978-0-470-56552.
26. **Gangster, Gadget.** Accelerometer & Gyro Tutorial. *Instructables.com*. [Online] Autodesk, Inc, 2015. [Citace: 15. 4 2015.] <http://www.instructables.com/id/Accelerometer-Gyro-Tutorial/step1/The-Accelerometer/>.
27. **Kistner, Gavin.** Frame Rate-Independent Low-Pass Filter. <http://phrogz.net/>. [Online] 2011. [Citace: 22. 4 2015.] <http://phrogz.net/js/framerate-independent-low-pass-filter.html>.

28. **Bureau international des poids et mesures.** Système International d'Unités. *BIPM.* [Online] 1960. [Citace: 16. 4 2015.] <http://www.bipm.org/en/CGPM/db/11/12/>.

29. **Google Inc.** Mapy Google. *maps.google.com.* [Online] 7 2011. [Citace: 23. 4 2015.] <https://www.google.cz/maps/@49.952501,15.79174,3a,75y,140.53h,88.18t/data=!3m5!1e1!3m3!1sYBr9GSd2T5FLdpOoxfi54w!2e0!3e5?hl=cs>.

Seznam obrázků

| | |
|---|----|
| Obr. 1: Model souřadnicového systému akcelerometru (18) | 5 |
| Obr. 2: Znárodnění gyroskopu (1) | 7 |
| Obr. 3: Přehled senzorů a jejich zón a využití (21) | 19 |
| Obr. 4: Dosah a vyzářovací diagramy jednotlivých senzorů (11) | 20 |
| Obr. 5: Souřadnicový systém zařízení (2) | 32 |
| Obr. 6: Souřadnicový systém světa (14) | 33 |
| Obr. 7: Ukázka kalibrace G-Sensoru | 39 |
| Obr. 8: Use-case diagram aplikace | 45 |
| Obr. 9: Class diagram aplikace | 46 |
| Obr. 10: Znárodnění výpočtu | 49 |
| Obr. 11: Graf znázorňující zpracování low-pass filtrem (alfa = 0.181) | 51 |
| Obr. 12: Graf využívaných průměrů v čase t | 52 |
| Obr. 14: Reprezentace výpočtu průměru | 54 |
| Obr. 13: Oblast, kde byla zaznamenána data pro následující ukázkou analýzy (27) | 55 |
| Obr. 15: Vztah rychlosti na analýzu výchylek | 57 |
| Obr. 16: Grafické znázornění detekce výmolů | 57 |
| Obr. 17: Ukázka prvního fragmentu | 59 |
| Obr. 18: Ukázka zobrazení skrz MyGLSurfaceView na druhém fragmentu | 60 |
| Obr. 19: Zobrazení třetího fragmentu a jeho návaznosti na Google Maps | 62 |

Seznam tabulek

| | |
|---|----|
| Tab. 1: Dostupnost senzorů podle platformy. (2) | 31 |
| Tab. 2: Výpočet průměru | 53 |

Seznam kódů

| | |
|--|----|
| Kód 1: Získání instance SensorManager | 26 |
| Kód 2: Získání všech dostupných senzorů | 33 |
| Kód 3: Implementace a registrace senzorových událostí..... | 34 |
| Kód 5: Implementace rozhraní SensorEventListener | 35 |
| Kód 4: Volání metody onPause..... | 35 |
| Kód 6: Získání přesnosti senzoru | 36 |
| Kód 7: Použití vlastního sensor event listeneru | 48 |
| Kód 8: Vyrovnání náklonu zařízení..... | 50 |
| Kód 9: Low-pass filtr (21)..... | 51 |
| Kód 10: Rozdělení sekvencí..... | 54 |

Příloha A – DVD

Součástí diplomové práce je DVD s tímto obsahem:

- Zdrojové kódy aplikace
- Diplomová práce v digitální formě
- UML diagramy v plné kvalitě
- Spustitelný instalační balíček aplikace



UNIVERZITA HRADEC KRÁLOVÉ
Fakulta informatiky a managementu
Rokitanského 62, 500 03 Hradec Králové, tel: 493 331 111, fax: 493 332 235

Zadání k závěrečné práci

Jméno a příjmení studenta:

Tomáš Pochobradský

Obor studia:

Aplikovaná informatika (2)

Jméno a příjmení vedoucího práce:

Tomáš Kozel

Název práce:

Senzory mobilních zařízení

Název práce v AJ:

Sensors of mobile devices

Podtitul práce:

Využití senzorů mobilního zařízení v dopravě

Podtitul práce v AJ:

The use of mobile device sensors in traffic

Cíl práce: Cílem práce je analyzovat možnosti využití senzorů mobilního zařízení v dopravě a navrhnout řešení založené na nalezených možnostech.

Osnova práce:

1. Úvod
2. Senzory mobilních zařízení - aktuální možnosti, principy a trendy
3. Analýza využitelnosti senzorů v dopravě
4. Softwarová rozhraní pro přístup k senzorům v Androidu
5. Přesnost a možnosti kalibrace senzorů
6. Návrh a implementace aplikace
7. Shrnutí výsledků a závěr

Projednáno dne: *13.10.14*

Podpis studenta

Podpis vedoucího práce