

# VYUŽITÍ TECHNOLOGIE KINECT V OBLASTI NAVIGACE TĚLESNĚ POSTIŽENÝCH OSOB

Juraj Kisztner<sup>1</sup>

Institut Geoinformatiky, Hornícko-geologická fakulta, VYSOKÁ ŠKOLA BÁŇSKÁ -  
TECHNICKÁ UNIVERZITA OSTRAVA,  
juro.kisztner@gmail.com

**Abstrakt.** V priloženej práci je spracovaná problematika využitia technológie Kinect ako zariadenia, pomocou ktorého je možné identifikovať prekážky v trase človeka so zrakovým postihnutím a navigovať ho v obchádzke tejto prekážky. V práci je vytvorená aplikácia na identifikáciu objektov z mračna bodov z prístroja Kinect a následné predanie týchto dát navigačnému softvéru, ktorý pomáha v obchádzke prekážok.

V prvej časti sa práca zaoberá možnosťami identifikácie objektov z mračna bodov, ich vymedzením a určením lokálnych súradníc prekážky. V druhej časti sa práca zaoberá prepojením identifikovaných prekážok s navigáciou.

Pretože v práci sa snažíme riešenie prispôbiť potrebám ľudí so zrakovým postihnutím, všetky komunikačné prostriedky medzi naším riešením a užívateľom je vytvorené pomocou zvukových upozornení.

**Kľov slova:** Kinect, navigácia, spracovanie mračna bodov, GIS

**Abstract.** The thesis deals with the use of Kinect technology as a device enabling to identify the obstacles in the way of visually impaired persons and navigate them to bypass these obstacles. The thesis includes an application for identifying objects from the point cloud from Kinect device and subsequent transfer of these data into the navigation software which helps to bypass the obstacles.

The first part of the thesis deals with the possibilities of identifying objects from the point cloud, their specification and determination of the local coordinates of the obstacle. The second part of the thesis is dedicated to the connection of the identified obstacles with the navigation.

Since our aim was to adjust the outcomes to the needs of visually impaired persons, all communication devices between the outcome and the users are created with the use of audible signals.

**Keywords:** Kinect, navigation, point cloud, GIS

# 1 Úvod

Zrak je jedným zo zmyslov človeka. Patrí medzi tie najdôležitejšie. Pomocou zraku vnímame svet okolo seba a prijímame ním väčšinu informácií z okolia. V prípade že o tento ľudský vnem prideme alebo bude značne obmedzený, stáva sa život výrazne obtiažnejším. Každodenné schopnosti, ktoré sú pre nás vidiacich bežné, sa stávajú obmedzenejšie a náročnejšie. Či už ide o prácu, záľuby, starostlivosť o svojich blízkych, domácnosť, komunikácia s okolím a v neposlednom rade orientáciu a pohyb. V tejto práci sa budeme zaoberať práve kompenzáciou obmedzenia zrakovu postihnutých v orientácii a pohybe. V tejto práci sa snažíme k dnes bežne dostupným navigačným pomôckam - bielej palici pridať nový systém založený na hernom bezkontaktnom ovládači Kinect.

V súčasnosti je vo väčšine prípadov technológia Kinect využívaná v oblasti herného priemyslu, pre ktorý bola primárne vytvorená. Vďaka nízkej cene a pomerne zaujímavým technologickým možnostiam je ale táto technológia využívaná aj inak ako na poli herného priemyslu. Často ide o oblasť robotiky, ovládania počítačových systémov gestami a podobne.

Vďaka možnostiam tejto technológie sme sa rozhodli využiť Kinect spolu z GIS na identifikovanie prekážok a navigáciu zrakovu postihnutých ľudí. Takéto využitie Kinectu je dnešnej dobe nepreskúmané a prináša možné zlepšenie kvality pohybu v teréne pre nevidiacich a slabozrakých. Spojením oblasti geoinformatiky, diaľkového prieskumu a počítačového videnia je možné identifikovať objekty z mračna bodov meraného Kinectom a následne tieto objekty spracovať.

## 1.1 Zrakové postihnutie

Zrakové postihnutie je poškodenie zrakového analyzátora alebo porucha jeho funkcií. Zrakovým analyzátorom rozumieme ľudské oko, ktoré je schopné predať pomocou nervovej sústavy informácie do kôrového zrakového centra v mozgu. Zrakove postihnutie je podľa WHO [11] spôsobované:

- neliečenými chybami refrakcie oka (krátkozrakosť, ďalekozrakosť alebo astigmatizmus), 43 %
- neliečeným šedým zákalom, 33 %,
- zeleným zákalom, 2 %.

Osoby so zrakovým postihnutím sú ľuďia s rôznymi druhmi a stupňami znížených zrakových schopností. Užšie sa týmto termínom rozumejú tí, u ktorých poškodenie zraku nejako ovplyvňuje činnosť v bežnom živote a pri ktorých bežná optická korekcia nepostačuje. Zrakové postihnutie sa klasifikuje podľa WHO [11] do piatich kategórií, a to:

- stredná slabozrakosť,
- silná slabozrakosť,
- ťažko slabý zrak,
- praktická nevidomosť,
- úplná nevidomosť.

Zaujímavými faktami si môžeme lepšie utvoriť obraz o zrakovom postihnutí. Podľa WHO [11] viac ako 90 % zrakovo postihnutých sa nachádza v rozvojových krajinách a viac ako 65 % zrakovo postihnutých je starších ako 50 rokov.

## 2 Analýza a návrh

### 2.1 Špecifikácia funkcií

Pre nás vidiacich je ťažké si predstaviť svet a získavanie informácií bez zraku. Vidiaci človek si môže predstaviť potreby nevidiaceho len veľmi ťažko, a preto sme sa snažili veľkú časť SW a HW riešenia konzultovať priamo s nevidiacimi alebo s ich blízkym okolím, a preto si uvedomujeme, že priame nasadenie vyžaduje úzku a dlhodobú spoluprácu na vývoji takéhoto zariadenia. Návrh funkcií vychádza z preštudovaných materiálov, z konzultácií s organizáciou SONS a priamou diskusiou so zrakovo postihnutými a ich asistentmi.

Požiadavky na systém sú:

- systém postavený na otvorenej platforme,
- dôraz na minimalizovanie rozmerov HW častí systému,
- čo najdlhšia prevádzková doba v teréne,
- zvukové upozornenia,
- minimalizácia hmotnosti systému,
- dôraz na rýchlosť zisťovania prekážok.

Na základe požiadaviek bude systém prevádzkovaný na mini počítači *Raspberry Pi*, kapacitnej externej batérii s možnosťou napájania 19 V a 5 V zároveň. Softvér bude postavený na operačnom systéme Linux Debian a skriptovacím jazyku Python s knižnicami *OpenCV* a *Libfreenect*.

## 3 Kinect

Kinect je primárne herné zariadenie. Založené na ovládaní hier pomocou pohybov a zvuku. Vyrobený bol spoločnosťou Microsoft pre hernú konzolu Xbox 360 a dnes novú generáciu Xbox One. Užívateľ nepotrebuje na ovládanie hry žiadny iný herný ovládač pretože Kinect umožňuje hru ovládať pohybmi a gestami. Kinect je dnes možné využiť okrem herných konzol aj na PC. Spoločnosť Microsoft v roku 2011 vydala softvérový balíček na prácu s Kinectom, vďaka čomu je možné využiť Kinect na ovládanie PC gestami a pohybmi.

Pre účely tejto práce však nebol využitý softvérový balíček od spoločnosti Microsoft z dôvodov viazanosti na operačný systém Microsoft Windows, ktorý nie je prakticky možné využiť pre menšie HW zariadenia ako je Raspberry Pi. Jedným z ďalších dôvodov je close-source politika spoločnosti Microsoft, ktorá by nepomáhala ďalšiemu slobodnému rozvoju celku alebo častí tejto práce. Výbornou alternatívou k softvérovému riešeniu od spoločnosti Microsoft je softvér *libfreenect* od otvorenej komunity *OpenKinect*, ktorý umožňuje využiť Kinect na dnes najviac rozšírených platformách, akými sú Windows, Linux a OSX.

Kinect sa skladá z motorizovaného podstavca, mikrofónov, RGB kamery a hĺbkovej kamery. Je napájaný na 19 V a pomocou USB portu je ho možné pripojiť k PC.

## 4 Segmentácia obrazu

Spracovanie obrazu je jedna z najdôležitejších častí tejto práce. Správne využitie segmentácie obrazu umožňuje identifikovať objekty a pomocou takto získaných informácií navigovať nevidiaceho užívateľa.

Segmentácia obrazu je metóda digitálneho spracovania obrazu, pomocou ktorej identifikujeme v obraze segmenty so spoločnými vlastnosťami. Tieto segmenty by pri dokonalej segmentácii mali predstavovať objekty. Dokonalá segmentácia je prakticky nemožná, pretože snímky sú zaťažené šumom a informačným šumom. Snahou je sa čo najviac priblížiť dokonalej segmentácii pomocou odstránenia šumu a správneho výberu segmentačného algoritmu. Výber algoritmu závisí prevažne od účelu, na ktorý využívame spracovanie obrazu. Segmentačné algoritmy sa podľa [4] dajú rozdeliť do troch skupín, ktoré si ďalej trochu predstavíme.

### Prahovanie

Prahovanie je najjednoduchšou a najstaršou metódou. Často sa využíva v kombinácii so sofistikovanejšími metódami. Jeho výhoda je malá implementačná náročnosť a rýchlosť.

Princíp fungovania prahovania spočíva na hodnotení jasů. Všetky hodnoty jasů nižšie než prah sú považované za pozadie a hodnoty vyššie ako prah sú považované za objekt.

### Regionálne metódy

Algoritmy spadajúce do tejto skupiny vytvárajú segmenty na základe homogenity pixelov. Kritérium homogenity môže byť založené na základe rôznych vlastností, napríklad jas alebo okolie pixelu.

Algoritmus inicializuje počiatočný pixel, takzvané semienko. Rozmiestnenie počiatočných pixelov môže byť náhodné, pravidelné alebo algoritmom. Z tohto pixelu sa následne iteratívne rozširuje segment. Veľkou výhodou takýchto algoritmov je odolnosť voči šumu. Ako uvádza Straka [4], medzi najznámejšie regionálne algoritmy patrí *Split and Merge* a *Watershed*.

### Detekcia hrán



**Obr. 1.** Ukážka globálneho prahovania v prostredí OpenCV.  
Zdroj: Sukromný archív autora.

Detekcia hrán spočíva v detekcii lokálneho rozdielu vo vlastnostiach obrazu. Špecifikovanie všeobecného procesu segmentácie na základe detekcie hrán sa dá podľa Straku [4] zhrnúť do dvoch podprocesov. V prvej fáze sa hrana detekuje a v druhej fáze sa z detekovanej hrany vytvorí samotná segmentácia. V procese detekcie hrán je dôležité zaviesť pojem hrana.

Hrana je zjednodušene povedané miesto, v ktorom dochádza k výraznej zmene intenzity. Ak si predstavíme jas ako funkciu, môžeme na detekciu hrán využiť deriváciu. Maximálna hodnota derivácie bude v kolmom smere voči hrane.

Na základe derivácii sa rozdeľujú algoritmy detekcie hrán na algoritmy využívajúce prvú deriváciu a algoritmy, ktoré využívajú druhú deriváciu. Medzi algoritmy využívajúce prvú deriváciu patrí Canny filter, ktorý bol využitý aj v tejto práci. Informácie o aplikácii Canny filtra sú popísané v kapitole 5.2. Filtrami, ktoré využívajú druhú deriváciu, sú napríklad Laplacian filter a Marr-Hildreth filter.

## 5 Vlastné riešenie

Komplexné riešenie, ktoré predstavujeme v tejto práci, využíva skupinu rozličných SW a HW prostriedkov na vytvorenie prototypového riešenia (obrázok 2). Prototypové riešenie pomáha zlepšovať pohyb zrakovo postihnutého v teréne na základe pokročilých metód detekcie obrazu z Kinectu, vývoja mobilných aplikácií a v neposlednom rade využitím poznatkov zo serverových a databázových riešení.



Obr. 2. Prototypové riešenie

### 5.1 HW úprava Kinectu

Kinect je zariadenie primárne určené ako ovládač pre herné zariadenia XBOX, preto nie je tvar a HW prispôsobený pre pohyb v teréne na tele zrakovo postihnutej osoby. Najdôležitejšou úpravou bolo pripojenie Kinectu k externému zdroju energie. Kinect je napájaný z elektrickej siete napätím 220 V a transformátorom je napätie prevedené na 19 V. Z týchto dôvodov bolo nutné pripraviť napájanie Kinectu na napájanie z externého zdroja. Odstránením transformátoru a pripojením AC/DC Male konektoru sme dosiahli možnosť univerzálneho napojenia na externý zdroj energie.

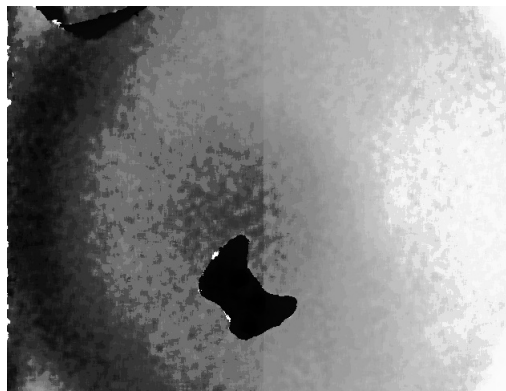
### 5.2 Modul na detekciu objektov

Ako sme uviedli v predchádzajúcich kapitolách, bolo potrebné vytvoriť modul, ktorý umožní spracovať obrazové dáta z Kinectu. Z našich predchádzajúcich testov sme zistili, že možnosti Kinectu, čo sa týka kvalitného zisťovania prekážok sú v intervale 0.5 - 4.0 m. Tento interval je v našich aplikáciách použitý na detekciu prekážok. Na základe rôznych testov sme dospeli k výbornej kombinácii Canny filtra a regionálneho filtra pre potreby tejto práce. Jedným z hlavných obmedzení bola požiadavka na rýchlosť analyzovaného obrazu. Dôvodom je hlavne praktické využitie, kde v teréne nemôže užívateľ čakať na odpoveď systému viac ako pár sekúnd.

Kinect ukladá informácie v 11-bit škále, čo znamená, že informácie v obraze môžu nadobúdať hodnoty od 0-2047. Pretože dnes väčšina algoritmov implementovaných v knižnici OpenCV umožňuje spracovať hodnoty v rozsahu 0-255, teda 8-bit. Pri prvotnom spracovaní treba upraviť RAW dáta z Kinectu na 8bit obraz. Po spracovaní RAW dát je zjavné, že sa časť informácií stratí.

Predspracovaním hĺbkových obrazových dát môžeme získať lepšie hodnoty na zisťovanie prekážok. Samotné predspracovanie v našom riešení spočíva v dvoch krokoch.

Prvý krok odstraňuje šum. Pomocou jednoduchého vyrovnania histogramu upravujeme kontrast hĺbkových dát. V následnom kroku pomocou funkcie gaussovského rozostrenia obrazu odstránime šum. V tomto kroku dochádza k strate detailov, preto treba správne nastaviť veľkosť kernelu, ktorým sa bude obraz rozmazávať.

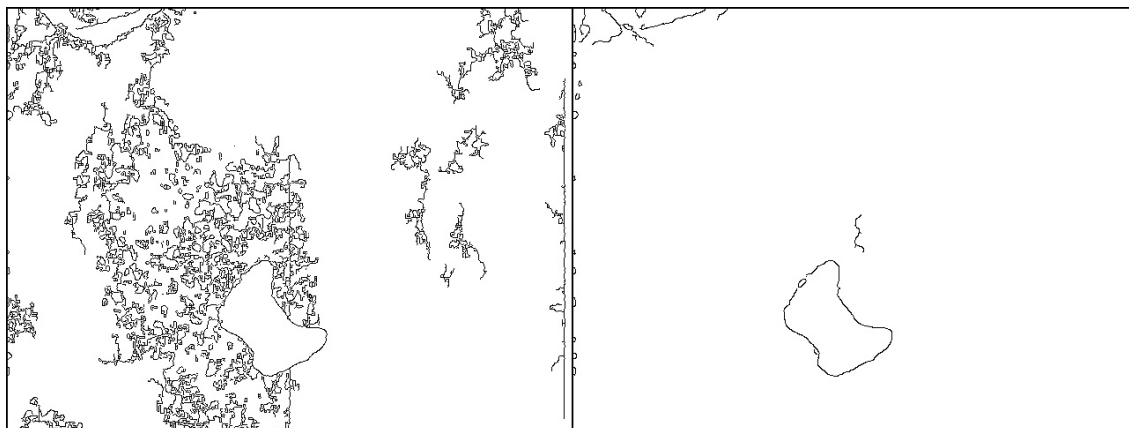


**Obr. 3.** Obraz po úprave histogramu.



**Obr. 4.** Obraz po použití gaussovského rozostrenia.

Na obrázkoch 3 a 4 je vidieť že po vyrovnaní histogramu dochádza k vzniku vysokokontrastných oblastí na miestach, kde by sa nemali objaviť, preto po použití rozostrenia Canny filter nedetekuje hranu na takýchto miestach. Na obrázku 5 je možné vidieť práve využitie gaussovskeho rozostrenia pri identifikácii hrán.



**Obr. 5.** Detekcia hrán pred (vľavo) a po (vpravo) použití rozostrenia.

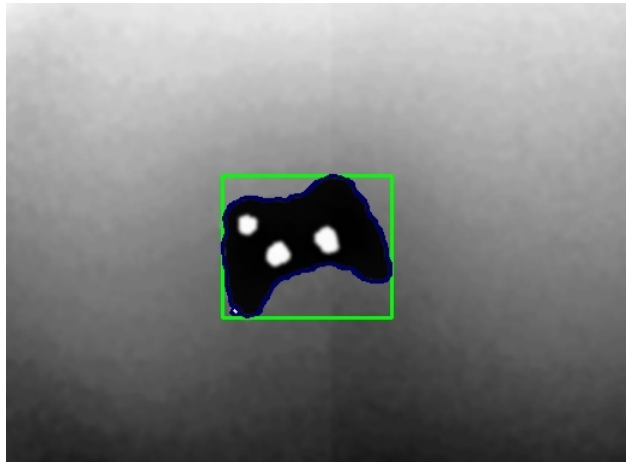
V druhom kroku sa z detekovaných hrán vytvorí dočasne segmenty. Segmentácia prebieha na základe preloženia obdĺžnika nad hranou. Tieto obdĺžniky obsahujú požadovaný objekt a jeho okolie. Popis odstránenia okolia a zachovania informácií o požadovanom objekte je popísaný ďalej v tejto kapitole.

Mnohé zo segmentov môžu predstavovať šum. Tento šum je často tvorený malými alebo výrazne tvarovo atypickými (úzkymi) segmentami. V našom systéme je každý segment podrobený testovaniu na veľkosť a tvar a v prípade nesplnenia podmienok je tento segment považovaný za šum a je odstránený.

Po odstránení šumu je potrebné spracovať požadovaný objekt a odstrániť informácie o okolí. Odstránenie okolia prebieha na základe štatistického spracovania polohy pixelov v trojrozmernom priestore. Z dôvodov požiadavky na rýchlosť detekcie sme vyhodnocovali každý desiaty pixel v smere osi X a desiaty pixel v smere osi Y. Klastrovou analýzou sme tieto pixely roztriedili do dvoch skupín. Jedna skupina predstavuje pozadie a druhá predstavuje požadovaný objekt. Realizáciu klastrovej analýzy v našom riešení je vidieť na obrázku 7. Červené body predstavujú pixely, ktoré patria pozadiu a fialové pixely predstavujú objekt.

Následne sú poslané tieto informácie do databázy. V prípade, že objekt je tvorený viacerými hranami a je na jednom objekte vygenerovaných viacej segmentov je potrebné spojiť správne segmenty do jedného, ktoré predstavujú aj v





**Obr. 6.** Segmentácia obrazu založená na štatistike a detekcií hrán v obraze.

skutočnosti jeden objekt. Vlastnosťou, ktorú sme využili, je priestorová blízkosť alebo prienik. Z obrázka 8 je zrejmé, ako jeden objekt môže obsahovať viac segmentov, je potrebné ich správne zlúčiť do jedného objektu. Avšak často sa môže vyskytnúť situácia, kde sa dva objekty na získanom obraze budú prekryvať, ale nebudú totožné, a preto je potrebné tieto dva objekty zachovať od seba odlíšené a takto ich interpretovať užívateľovi. Posudzovaním vzájomnej polohy v lokálnych súradniciach obrazu a vzdialenosti môžeme rozhodnúť, ktoré segmenty tvoria jeden objekt a ktoré tvoria dva separátne objekty. Na toto riešenie sme využili rozšírenie databázy PostgreSQL - PostGIS, ktorým sme zisťovali priestorovú blízkosť objektov.

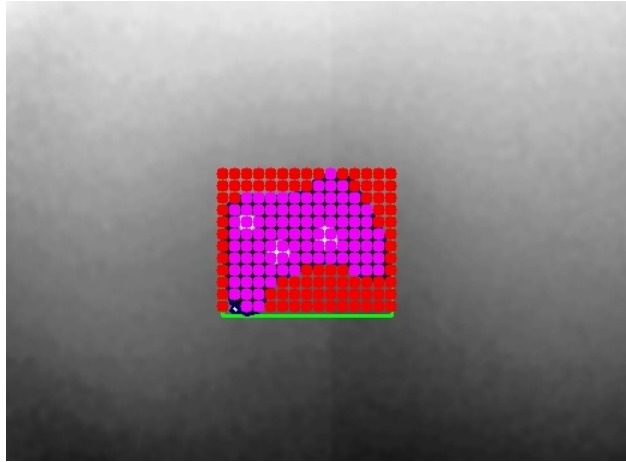
### 5.3 Server-side aplikácia

Serverová aplikácia je sada nástrojov, ktoré umožňujú spúšťať zariadenie Kinect a všetky časti softvérového riešenia na strane serveru. Volanie týchto aplikácií je umožnené pomocou ovládača, ktorý je bližšie popísaný v kapitole 5.4. Na servery je v podstate uložené celé aplikačné riešenie:

- Pripojenie a spúšťanie Kinectu,
- detekcia objektov,
- navigácia,
- zvukové navádzanie.

#### Navigácia

Hlavnou myšlienkou pri vytváraní tohto systému bolo umožnenie zrakovo postihnutým zlepšiť pohyb v teréne a navigovať ich. Navigácia pre zrakovo postihnutých je veľmi náročnou časťou, čo sa týka technickej realizácie. Navigovanie



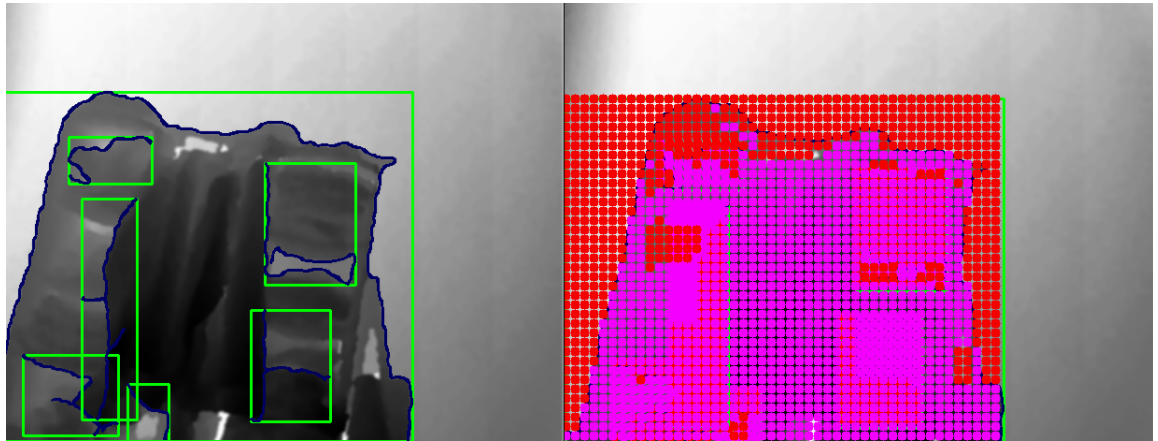
**Obr. 7.** Klastrová analýza segmentu.

zrakovo postihnutých nemôže byť realizované podobne, ako sme zvyknutí v turistickej alebo dopravnej navigácii, kde užívateľ dostáva svoje navigačné dáta a zároveň vníma zrakom svoje okolie. Takáto orientácia je značne jednoduchá a poznáme ju už od čias máp. V prípade navigácie zrakovo postihnutého je potrebné navigačné údaje predávať pomocou zvuku (kap. 5.3), a to v čo najjednoduchšej forme.

Riešenie navigácie bolo realizované pomocou rozšírenia PostGISu o pgRouting. PgRouting obsahuje mnoho algoritmov z oblasti sieťových analýz a navigácie, medzi najzaujímavejšie patrí implementovaná heuristika pri sieťových analýzach a problém obchodného cestujúceho.

V našom prípade sme použili Dijkstrov algoritmus pre nájdenie najkratšej cesty. Navigovanie zrakovo postihnutých v našom systéme je realizované dvomi spôsobmi. Prvý spôsob odosiela do aplikácie parametre súčasného miesta a cieľ. Aplikácia vyhledá v pripravenej sieti najbližší uzlový bod, ktorý sa nastaví ako východiskový pre navigáciu.

Do tabuľky s vypočítanými navigačnými údajmi pridávame stĺpec *angle* obsahuje vypočítaný uhol na nasledujúcu hranu v najkratšej ceste. Pomocou nižšie uvedeného SQL dotazu sú získané dve hodnoty azimutu na líniu 1 a líniu 2. Odčítaním týchto dvoch azimutov získame uhol medzi líniou 1 a 2. Princíp ukázaný v tomto príklade je použitý na výpočet všetkých uhlov v najkratšej ceste.



Obr. 8. Ukážka objektu tvoreného viacerými segmentami.

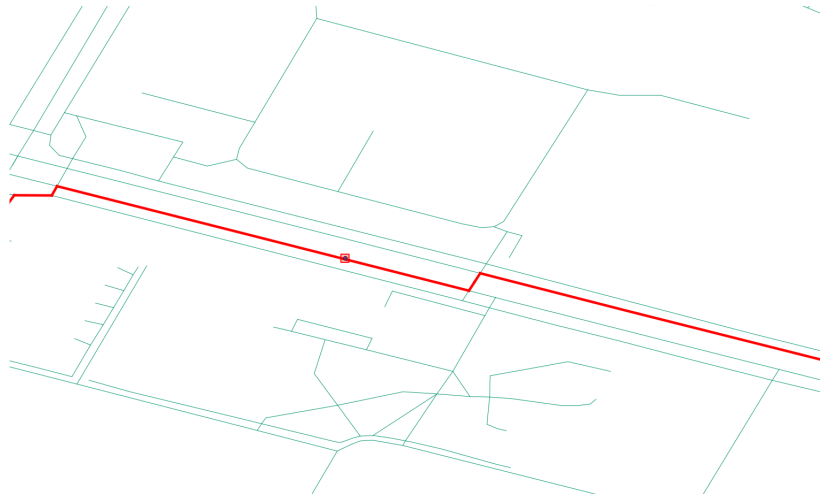
```
SELECT ST_Azimuth(ST_StartPoint(ways.the_geom), ST_EndPoint(ways.the_geom))
/(2*pi())*360 as deg FROM ways WHERE gid= 1;
SELECT ST_Azimuth(ST_StartPoint(ways.the_geom), ST_EndPoint(ways.the_geom))
/(2*pi())*360 as deg FROM ways WHERE gid= 2;
```

Stĺpec *direction* je odvodením stĺpca *angle*. Prepočítaním hodnôt sme schopní uložiť 5 smerov navigovania. Right, left, straight, slight right a slight left umožňujú navigovať zrakovo postihnutého podľa uhlu a smeru medzi hranami.

Druhý spôsob navigovania zrakovo postihnutého sa využíva v prípade, že zrakovo postihnutý dostane informácie o prekážke, ktorú podľa vlastného úsudku nie je schopní bezpečne prekonať. Systém požiada o prepočet trasy. Systém vypočíta alternatívnu trasu z aktuálnej polohy. Alternatívna trasa však nemôže obsahovať hranu, na ktorej sa prekážka nachádza.

Prepočet trasy prebieha použitím rovnakej časti systému ako v navigácii, ktorú sme popísali vyššie. Pred samotným vypočítaním najkratšej cesty je však potrebné upraviť sieť, v ktorej sa naviguje. Aktuálna poloha je pripojená k najbližšej hrane. V tomto mieste je potrebné upraviť túto hranu, tak aby končila v tomto mieste a na ďalší uzol v sieti už nepokračovala. Jediná cesta z tohto miesta je vrátiť sa naspäť a pokračovať po novovypočítanej trase. Výpočet trasy pred a po vložení je vidieť na obrázkoch 9 a 10. Posledným spôsobom navigácie, ktorá je implementovaná v našom systéme, je navigovanie opisu prekážky. Prekážky, ktoré sú systémom detekované, sú užívateľovi predané zvukovou formou, a to ich vzdialenosť a veľkosť. Užívateľ dostáva informácie, tak ako pri predchádzajúcich formách navigácie prostredníctvom zvukového modulu.

## Zvukový modul



**Obr. 9.** Trasa pred vložením prekážky.

Zvukový modul v našom systéme je rozhraním medzi výstupom všetkých častí systému a zrakovo postihnutým. Pomocou zvukovej karty a open-source softvéru *espeak* sú užívateľovi odovzdávané navigačné dáta. Vypočítané navigačné dáta sú predané pomocou parametrov zvukovému modulu. Parametre, ktoré sú potrebné pre zvukovú navigáciu, sú uložené v databáze.

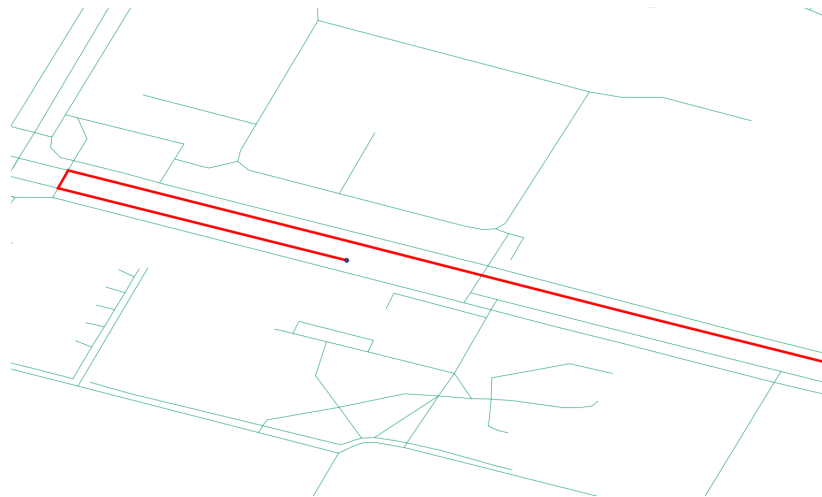
Espeak je optimalizovaný pre anglický jazyk. Z toho dôvodu sú zvukovému modulu odovzdávané parametre v anglickom jazyku.

#### 5.4 Dialkové ovládanie systému

Dialkové ovládanie slúži k interakcii zrakovo postihnutého so systémom. Dialkové ovládanie je aplikácia, ktorá je vytvorená pre mobilné telefóny s operačným systémom Android. Aplikácia bola vytvorená v jazyku Java s použitím Android SDK. Aplikáciu sme nazvali *Batmobile* a slúži ako rozhranie medzi serverom a klientom. Užívateľ týmto rozhraním odosiela požiadavky na server. Tieto požiadavky sú následne spracované a vykonané. Z dôvodov jednoduchosti bolo potrebné rozdeliť obrazovku na štyri časti (obrázok 11). Zrakovo postihnutý je schopný celkom prirodzene rozlíšiť tieto časti.

Prvá časť sa nazýva *Nastav cieľ*. Ako názov napovedá, ide o výber cieľového bodu. Po kliknutí na toto tlačidlo aplikácia ponúkne výber zo štyroch cieľov. V našom základnom nastavení je v aplikácii možné využiť tieto cieľové body:

- Domov
- Obchod
- Banka
- Úrad



Obr. 10. Trasa po vložení prekážky.

Každá z týchto častí umožňuje nastaviť iný cieľ. Tieto ciele majú predstavovať ukážku najčastejších miest, ktoré zrakovo postihnutý navštevuje. Po stlačení jedného z týchto cieľov je na server odoslaná informácia o ciele a následne je využitý v navigácii.

Z tejto ponuky sa užívateľ môže vrátiť do hlavného okna stlačením HW tlačidla späť. Vrátenie do hlavnej ponuky je dôležitá súčasť aplikácie, pretože v prípade, že sa užívateľ rozhodne, že nezmení cieľ alebo vyberie cieľ, nemusí opustiť celú aplikáciu.

Nasledujúcou časťou je položka *Naviguj*. Stlačením tejto položky sa inicializuje HTTP POST požiadavkou navigačný SW na strane serveru, ktorý sme si popísali v kapitole 5.3. Cieľ je nastavený z predchádzajúcej kontextovej ponuky a počiatočným bodom sú súradnice získané z GPS zariadenia, ktoré obsahuje mobil. Tieto súradnice sú odoslané navigačnému softvéru.

Odpoveďou užívateľovi je prečítaná najkratšia trasa z počiatočného bodu do cieľového bodu pomocou softvéru espeak.

Treťou časťou je položka *Zisti prekážku*. Po kliknutí na túto časť z úvodnej obrazovky je inicializovaný HTTP POST, ktorý odošle požiadavku na štart Kinectu a zmerania okolia. Odpoveďou na túto požiadavku je vykonaná detekcia prekážok a odpoveďou je pomocou zvukového modulu opis prekážok. Na základe odpovedí môže užívateľ rozhodovať o obchádzke prekážky alebo o prepočítaní trasy.

Poslednou, štvrtou časťou je položka *Obíď prekážku*. Po kliknutí na toto tlačidlo je systému odoslaná požiadavka na prepočítanie trasy. Súradnice aktuálneho



**Obr. 11.** Domovská menu aplikácie Batmobile.

miesta sú odoslané a v systéme sa vykoná zmena topológie siete. Tento proces je bližšie popísaný v kapitole 5.3. Odpoveďou užívateľovi je prepočítaná trasa, ktorá nezahŕňa úsek s prekážkou. Pomocou zvukového modulu je užívateľovi táto trasa interpretovaná.

## 6 Testovanie a práca so systémom

Systém je postavený na získavaní dát zo zariadenia Kinect, ich interpretácii s ďalšími dátami, ako sú GPS súradnice aktuálneho miesta, cieľový bod navigácie a navigačná sieť.

Väčšina z týchto údajov sa získava až priamo v teréne. Testovanie v teréne otestovalo možnosti reálneho využitia v budúcnosti.

Figurantovi bolo počas testovania znemožnené využívať jeho zrakový zmysel a celý pohyb mohol vykonávať iba pomocou nášho systému.

Testovanie systému sme vykonali v areáli VŠB-TUO. Figurantovi, ktorý testoval systém, boli počas testovania simulované podmienky podobné reálnym situáciám, s ktorými sa môže zrakovo postihnutý stretnúť. Snažili sme sa, aby figurant bol donútený využiť všetky časti nášho systému.

## 7 Záver

Kinect je úžasným kusom HW, ktorý okrem herného priemyslu ma výborné uplatnenie v širokom spektre technických riešení. Jedno z takýchto riešení sme ukázali v tejto práci v ktorej sme úspešne vytvorili prototypovú aplikáciu, ktorá využíva Kinect na detekciu prekážok pre zrakovo postihnutých užívateľom. Spolu s GIS je Kinect silným nástrojom, ktorý môže úspešne pomôcť zlepšiť život zrakovo postihnutým osobám. Zrýchliť ich pohyb v neznámom prostredí a zlepšiť ochranu ich zdravia pred nárazmi do prekážok.

Navigačný SW, ktorý sme vytvorili, využíva dáta z GPS, databázový systém PostgreSQL s rozšírením PostGIS a pgRouting, ktoré umožňujú navigovať užívateľa v teréne a obchádzať prekážky.

V reálnom nasadení potrebuje naše riešenie značné úpravy, pretože potreby a obmedzenie každého zrakového postihnutého je individuálne, potrebuje aj takéto riešenie individuálny prístup. Určitým nedostatkom je použitie mobilného telefónu na diaľkové ovládanie, ktoré nám bolo bližšie pri vývoji ako vytvorenie HW komponenty. HW komponenta by bola značne citlivejším a použiteľnejším riešením, ale z nedostatku elektrotechnického vzdelania sme zvolili cestu vývoja mobilnej aplikácie, ktorá môže byť na druhej strane prívetivejšia pre mladšiu generáciu užívateľov.

V budúcnosti predpokladáme rozvoj aj tohto smeru nášho riešenia a postupné vylepšenie algoritmov a postupov pre reálne nasadenie.

## References

1. *KAHLER A., BRADSKI G. : Learning OpenCV.* O'Reilly Media, 2013. ISBN 978-1-4493-1465-1.
2. *KONECNY G. : Geoinformation.* Taylor & Francis, 2003. ISBN 0-415-23795-5.
3. *MOMJIAN, B.: PostgreSQL : praktický průvodce.* Computer Press, Brno, 2003. ISBN 80-7226-954-2.
4. *STRAKA, Stanislav. Segmentace obrazu.* Brno, 2009. Diplomová práce. Masaryková univerzita, Fakulta informatiky.
5. *RŮŽIČKA J., PEŇÁZ T., HORÁK J., STANKOVIČ J. : Publikování prostorových dat na internetu.* Distanční text. VŠB-TU Ostrava, 2003. ISBN 80 – 248 - 0416 – 6
6. *PAVLÍČEK, Pavel. Pomůcky pro usnadnění — Kompenzační pomůcky pro uživatele se zrakovým postižením[online][cit. 12. marec 2014]. Dostupný na World Wide Web: <http://pomucky.blindfriendly.cz/pomucky-pro-usnadneni-mobility.html>*
7. *PHP: Hypertext Preprocessor[online][cit. 1. marec 2014]. Dostupný na World Wide Web: <http://php.net/>*
8. *PostGIS[online][cit. 1. marec 2014]. Dostupný na World Wide Web: <http://postgis.refractions.net>*
9. *PostgreSQL 9.1.13 Documentation[online][cit. 1. marec 2014]. Dostupný na World Wide Web: <http://www.postgresql.org/docs/9.1/interactive/index.html>*
10. *Python[online][cit. 1. marec 2014]. Dostupný na World Wide Web: <http://python.org/>*
11. *WHO. WHO — Visual impairment and blindness[online]c2013[cit. 20. marec 2014]. Dostupný na World Wide Web: <http://www.who.int/mediacentre/factsheets/fs282/en/>*



## Seznam obrázků