Využitie rozšírenej reality v indoorovej navigácii

Bianka Šimková

3AIb, 2017 - 2018

Abstrakt. V našej práci je spomínaný úvod do problematiky indoorovej navigácie, ktorá je potrebná v rámci budovy vzhľadom nato, že GPS signály sú obmedzené prevažne na vonkajšie účely. Opisuje prehľad súčasného stavu v danej oblasti zo skúmaných vedeckých článkov. Cieľom je analyzovať a preskúmať existujúce riešenia využívajúce rozšírenú realitu pre určovanie polohy v interiéri prostredníctvom spracovania obrazu. Tak isto preskúmať použiteľnosť a presnosť senzorov smartfónov pre zobrazovanie navigačných značiek v režime rozšírenej reality.

Kľúčové slová: rozšírená realita, indoorová navigácia, senzory smartfónov, analýza obrazu

1 Úvod

Keďže navigácia pre chodcov sa stala všadeprítomnou prostredníctvom smartfónov s technológiou GPS, dopyt po spoľahlivom lokalizovaní v interiéri sa výrazne zvyšoval. Lokalizácia a navigácia v interiéroch sa považujú za príležitosť pre rôzne aplikácie, ako napríklad usmerňovanie cestujúcich na letiskách, účastníkov konferencie, návštevníkov nákupných centier, nemocníc alebo budov a mnohých iných služieb. Tradične používaná vonkajšia metóda lokalizácie GPS nie je dostupná vo vnútorných prostrediach. Je to spôsobené tým, že tento signál nie je dostatočne silný na to, aby prenikol cez steny budovy. Taktiež štruktúra vonkajšieho prostredia je odlišná od štruktúry toho vnútorného, kde sa môžu nachádzať eskalátory, výťah, poschodia alebo schody.

Polohu používateľa môžeme získavať prostredníctvom vopred definovaných prvkov, ktoré aplikácia rozpoznáva, ako sú QR kódy, ale aj chodby. Prístupy založené na analýze obrazu z kamery pre mobilnú lokalizáciu sa nespoliehajú na infraštruktúru budovy, a preto sú škálovateľné a lacné.

Pre spoľahlivejšiu lokalizáciu je možné využiť navigačné šípky, ktoré by boli dokresľované do obrazu z kamery, ako súčasť rozšírenej reality. Toto riešenie je optimálnejšie a prijateľnejšie pre používateľa než väčšina súčasných aplikácií, ktoré využívajú pre indoorovú navigáciu 2D mapy. Daný spôsob sa opiera o charakteristické referenčné obrazy a iné faktory, ako napríklad poloha smartfónu. Tú vieme získať pomocou zabudovaných senzorov ako akcelerometer, magnetometer či gyroskop. Tieto senzory sa zaradzujú k indoorovej lokalizácii, IPS.

1.1 Rozšírená realita

Rozšírená realita je spojenie digitálnych informácií s prostredím používateľa v reálnom čase. Na rozdiel od virtuálnej reality, ktorá vytvára úplne umelé prostredie, rozšírená realita využíva existujúce prostredie a navyše prináša nové informácie, ktoré sú získavané z rôznych zdrojov. Pomocou pokročilých technológií (napríklad pridanie počítačového videnia a rozpoznávania objektov) sa informácie v okolitom reálnom svete používateľa stávajú interaktívnymi a digitálne manipulovateľnými.

1.2 Indoorový pozičný systém

Indoorový pozičný systém, alebo aj IPS, môžeme opísať ako GPS pre indoorové prostredia. Využíva sa napríklad na vyhľadávanie ľudí alebo objektov vo vnútri budov práve prostredníctvom mobilných zariadení alebo tabletov. Môže sa ukázať nepostrádateľnou na miestach ako sú obchodné centrá, letiská alebo nemocnice.

1.3 Indoorová navigácia

Indoorová navigácia sa zaoberá navigáciou vo vnútri budov, kde GPS signály nemajú dosah. Využívajú sa tu teda iné technológie určovania polohy, ak je požadované automatické polohovanie. Na rozdiel od GPS je v tomto systéme možné vo väčšine prípadov detegovať aj podlažie, kde sa používateľ nachádza.

1.4 Prehľad súčasného stavu

V súčasnosti existuje mnoho riešení, či aplikácií, ktoré pomáhajú pri orientácii a lokalizácií v indoorovom prostredí. Avšak vedecké články, ktoré sa danou problematikou zaoberajú a využívajú pri tom analýzu obrazu z kamery, pracujú vo väčšine prípadov s mobilným robotom [3]. V takomto prípade je spracovávanie obrazu z kamery jednoduchšie a prináša optimálnejšie výsledky. Mobilný robot má kameru v statickej polohe, nenakláňa sa do strán. Tak je možné získať určité informácie z obrazu kamery, potrebné napríklad pri určovaní smerovania chodby, v nemennej podobe. Mobilný robot má taktiež konštantný pohyb, ktorý vieme s určitou pravdepodobnosťou predpovedať a statické senzory.

V našej práci však pracujeme s človekom, čo znamená, že pohyb predvídať nedokážeme a dáta zo senzorov sa neustále menia, keďže poloha kamery smartfónu nie je statická. Väčšina prác, ktoré sme čítali, opisovala hlavne optimálnejšie situácie, v ktorých svoje riešenia testovali. Išlo o prostredie chodieb, na ktorých sa nenachádzali skoro žiadne ďalšie objekty, ako ľudia, ktorí by prechádzali popred kameru alebo nábytok popri stenách. Ďalším možným problémom, ktorý sa vo

vedeckých prácach riešil taktiež len minimálne, sú odlesky svetiel na podlahe alebo samotné svetlá.

2 Návrh riešenia

V našej práci budeme pre navigáciu používateľa využívať obraz z kamery smartfónu doplnený o dáta zo senzorov. Samotná implementácia vlastného riešenia pozostáva z 3 hlavných krokov:

- Detekcia smerovania chodby
- Detekcia podlahy
- Vykreslenie navigačného prvku na detegovanú podlahu

2.1 Detekcia smerovania chodby

Určenie smerovania chodby je dôležité kvôli natočeniu navigačnej šípky. Tá by vo väčšine prípadov rovných chodieb mala ukazovať na jej koniec, pokiaľ by navigácia nenaznačila používateľovi zmenu smeru ešte skôr. Tento spôsob je možné kombinovať s dátami zo zabudovaných mobilných senzorov. Konkrétne s údajom, ktorý určuje natočenie na svetové strany, s azimutom. Ich kombinácia by mohla zaručiť zníženie výskytu chyby, a teda zlú navigáciu používateľa. Chyba v údajoch zo senzorov môže nastať vplyvom magnetického poľa.

2.1.1 Detekcia hrán

Najskôr je potrebné obraz z videa predspracovať snímku po snímke, než budú naň aplikované algoritmy. Ide o jeho prevedenie do čiernobieleho obrazu a následnú detekciu hrán. Každému pixelu v obraze je priradená hodnota, v našom prípade jas, ktorá je výstupom obrazovej funkcie f(x, y), kde x a y sú súradnice v rovine. Danú obrazovú funkciu môžeme brať ako maticu s počtom stĺpcov x a počtom riadkov y. Účelom detekcie hrán je zvýrazniť tie oblasti obrazu, kde dochádza k prudším zmenám jasu. Neželaným javom, ktorý zvyčajne môže nastať pri detegovaní hrán, je zvýraznenie šumu. Pomocou aplikovania algoritmu na detekciu hrán získame obraz zbavený nadbytočných informácií, pričom sa ale zachová pôvodná štruktúra obrazu. Hrany sa zvyčajne nachádza na hranici medzi dvomi regiónmi. V našom prípade je najdôležitejšie detegovať hrany, ktoré tvoria hranicu medzi podlahou a stenami chodby.

Existuje viacero algoritmov, prostredníctvom ktorých vieme nájsť hrany v obraze. Sú to detektory založené na gradiente, detektory založené na Laplace-ovi alebo Cannyho hranový detektor. Pri prvom spomínanom detektore sa využívajú derivácie prvého rádu. Lokálna veľkosť gradientu sa používa na výpočet mieri sily hrZnámymi

sú filtre ako Sobel, Prewitt alebo Roberts. Detektory založené na Laplace-ovi pre zmenu využívajú derivácie druhého rádu.

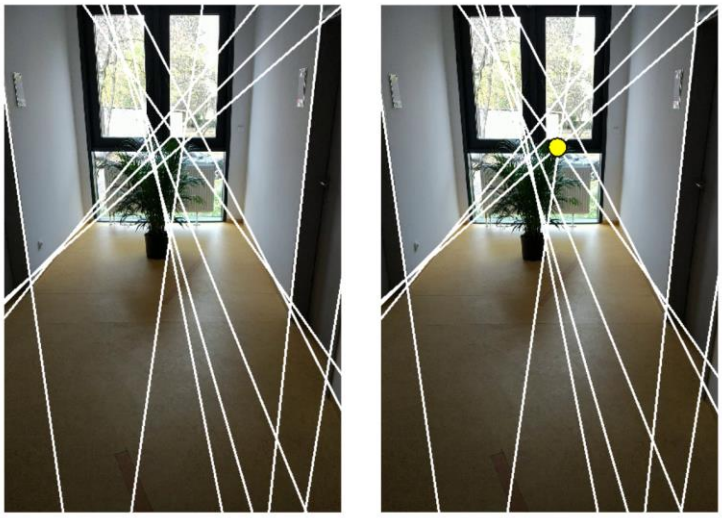
My sme na detekciu hrán vyskúšali dva spôsoby. Prvým bol Sobelov operátor. Ten počíta osobitne prvé derivácie pre osi X a Y. Sú to však len aproximácie, keďže obrazy nie sú kontinuálne. Druhým skúšaným postupom bol Cannyho hranový detektor. Filozofia tohto detektora je, že berie do úvahy nielen hodnotu, alebo teda gradient hrany na aktuálnom bode obrazu ale aj spoľahlivosť jeho susedov. Z týchto dvoch spomínaných postupov sme si zvolili Sobelov operátor. Tento filter prinášal optimálnejšie výsledky v ďalších krokoch implementácie.

Obrázok 1. Výsledok po aplikovaní Sobelovho operátora na snímku chodby

Obrázok 2. Výsledok po aplikovaní Cannyho hranového detektora na snímku chodby

2.1.2 Detekcia priamok v obraze

Po zredukovaní počtu hrán v obraze môžeme prejsť k samotnej detekcií výrazných čiar, alebo teda priamok. Na to sme využili Houghovú transformáciu, ktorá okrem čiar dokáže v obraze nájsť aj iné tvary, ktoré je možné zadať parametricky, ako kruh alebo elipsa.



Obrázok 3. Výsledok po aplikovaní Houghovej transformácie na snímku chodby

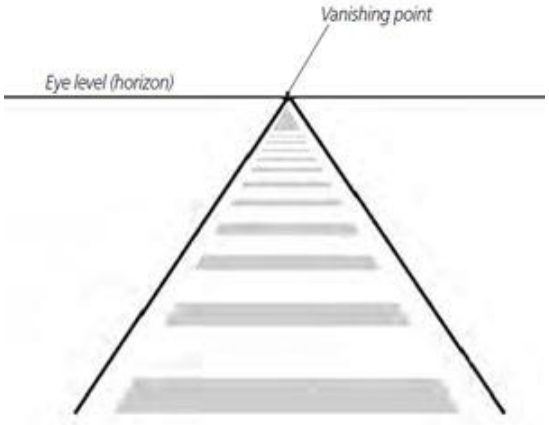
Na obrázku č.3 je vidieť, že bola nájdená aj potrebná hranica medzi podlahou a stenou chodby. Takáto detekcia priamok v obraze avšak nedáva optimálne výsledky v každom prípade. Nie stále je možné pomocou vopred určeného prahu získať dostatočný počet správnych priamok. Ako nesprávne priamky môže byť detegované napríklad svetlá alebo ich odlesky na podlahe. Vopred určiť optimálny počet priamok, ktoré by mali byť nájdené je taktiež nie veľmi ľahká úloha vzhľadom na meniace sa prostredie, svetelné podmienky, ľudí prechádzajúcich po chodbách alebo objekty popri stenách.

Čo je to vlastne prah a prahovanie? Prostredníctvom prahovania vieme rozdeliť objekty v obraze na popredie a pozadie, a to len na základe hodnoty ich jasu, alebo teda gradientu. Prah je vlastne hranica, ktorá určí či daný pixel patrí k popredíu, k objektom, teda hodnota jasu je vyššia než prah, alebo či patrí k pozadiu. Samozrejme existuje viacero spôsobov prahovania, ktoré môžu rozdeliť obraz do viacerých skupín, nie len na popredie a pozadie.

2.1.3 Vanishing point

Na obrázku č.3 vpravo je možné vidieť nájdený vanishing point, alebo teda bod prieniku v nekonečne. Je to bod v ktorom sa v diaľke stretávajú paralelné čiary, ak sú reprezentované v lineárnej perspektíve. Je definovaný aj ako bod, v ktorom niečo zmizne alebo prestane existovať. Pre nás je ale dôležitý fakt, že je to bod, ktorý určuje smerovanie chodby. Vypočítali sme ho ako miesto, kde sa pretína najviac detegovaných priamok. Takých vanishing pointov je ale na obraze zvyčajne viacero, závisí to od toho, kde bol tento obraz zaznamenaný alebo v koľkých smeroch sa nachádzajú objekty v obraze. Ak je 10 objektov v 10 rôznych smeroch, máme najmenej 20 bodov prieniku v nekonečne (jeden pre ľavú stranu, jednu pre pravú stranu a to desaťkrát). Avšak len jeden takýto bod určuje smerovanie chodby.

Pomôcť určiť ten správny by mohol fakt, že tento bod by mal mať prienik s pomyselnou čiarou horizontu. Táto čiara oddeľuje zem a oblohu, rozdeľuje všetky viditeľné smery na dve kategórie. Tie, ktoré pretínajú zemský povrch a tie, ktorí nie. Horizont sa nachádza približne vo výške očí. Tam, kde je táto čiara rovnobežná s líniou zorného poľa (kolmá na rovinu obrazu), je dokonale horizontálna.



Obrázok 4. Vanishing point ležiaci na pomyselnej čiare horizontu

Vzhľadom na to, že používateľ nedrží mobilné zariadenie neustále v jednej polohe, horizont sa v priebehu snímania obrazu mení. Jeho pozíciu je potrebné vypočítať. Využili sme na to zabudované senzory, ktoré sú už bežnou súčasťou väčšiny inteligentných mobilných zariadení.

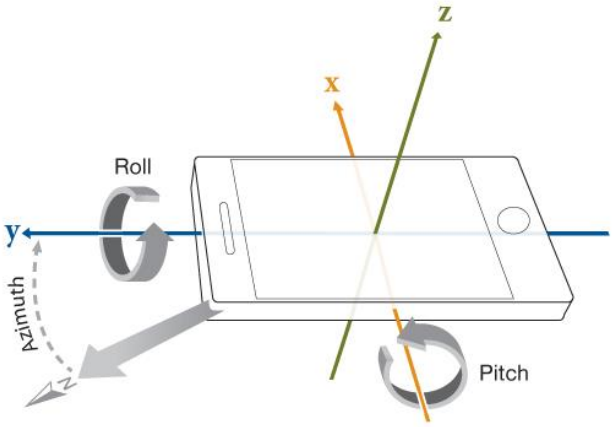
Vytvorili sme si pomocnú Android aplikáciu, ktorá počas zaznamenávania obrazu zapisuje jednotlivé hodnoty natočenia okolo osí X, Y a Z k príslušným snímkam a sekundám videa.

0	0.0	0.0	0.0	0
1	-66.1444	-52.25403	-0.388997	0
2	-65.02614	-52.29309	-0.589804	0
3	-65.02614	-52.29309	-0.589804	0
4	-62.79351	-52.92560	-1.471905	0
5	-61.85059	-55.12654	-2.474625	0
6	-61.85059	-55.12654	-2.474625	0
7	-61.79086	-56.36475	-2.698684	0
8	-61.79086	-56.36475	-2.698684	0
9	-61.32776	-56.99924	-2.804010	0
10	-61.32776	-56.99924	-2.804010	0
11	-61.46304	-57.18742	-2.611374	0
12	-61.46304	-57.18742	-2.611374	0
13	-60.97911	-57.38417	-2.385512	0
14	-61.59285	-57.43671	-1.815469	0
15	-61.59285	-57.43671	-1.815469	0
16	-61.1385	-57.76282	-1.898003	0
17	-61.1385	-57.76282	-1.898003	0
18	-60.98672	-58.00158	-1.789779	0
19	-60.98672	-58.00158	-1.789779	0

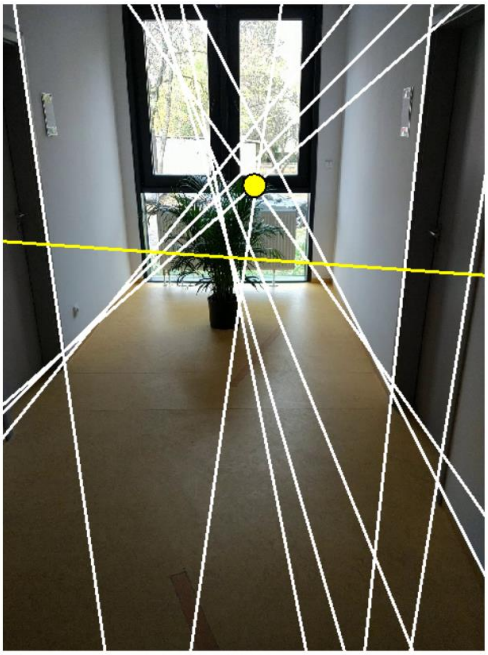
Obrázok 5. Vzorka údajov zo zabudovaných mobilných senzorov

Prostredníctvom týchto dát je následne možné vykonať operácie translácie a rotácie obrazu. Výsledný horizont je vykreslený do pôvodného obrazu. Analógiou sú napríklad indikátory polohy využívané v lietadlách.

Týmto postupom sme ale nezískali očakávané výsledky v prevažnej väčšine snímok. Vykreslená čiara horizontu síce bola natočená o správny uhol, no jej posunutie nevyhovovalo reálnemu horizontu. Preto nebolo možné využiť prienik detegovaného bodu prieniku priamok v nekonečne a čiary horizontu. Pre smerovanie chodieb sme využili len dáta zo zabudovaných senzorov, konkrétne údaj o azimute. Tieto dáta udávajú natočenie na svetové strany, teda vieme kam je používateľ natočený a kam ho treba nasmerovať. Aj keď pôvodný cieľ bolo tieto dva výstupy kombinovať pre dosiahnutie lepších výsledkov.



Obrázok 6. Natočenie okolo osí X, Y a Z

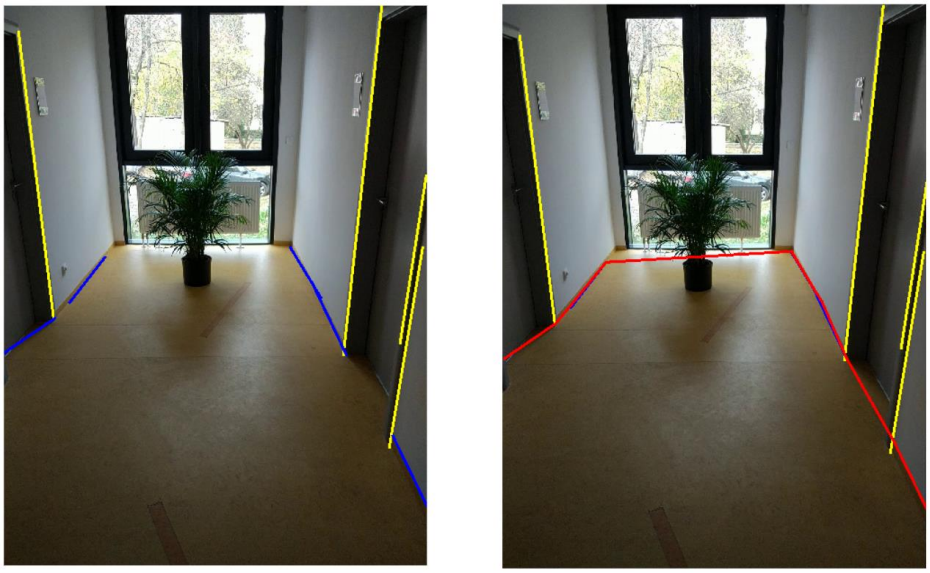


Obrázok 7. Detegovaný vanishing point a vykreslený horizont

2.2 Detekcia podlahy

Detekcia, alebo teda segmentácia podlahy je dôležitá z hľadiska následného vykresľovania navigačného prvku na túto plochu. Ide o ohraničenie oblasti, ktorá obsahuje pixely patriace segmentu podlahy. Tento problém nie je triviálny z hľadiska rôznorodosti podlahy. Každá podlaha môže byť v budove iná, môže mať rôznu farbu, textúru. Možnosť výskytu objektov, či už statických alebo dynamických môže taktiež urobiť detekciu ťažšou. V našom prípade nie je potrebné detegovať aj prekážky, vzhľadom na to, že navigačný prvok môže byť vykresľovaný aj cez nich. Prostredníctvom správnej detekcie podlahy by bolo možné jednoduchšie určiť presnejšiu polohu vanishing pointu. Ak by sme detegovanými bočnými obrysmi podlahy viedli priamky, tie by mali prienik v nekonečne.

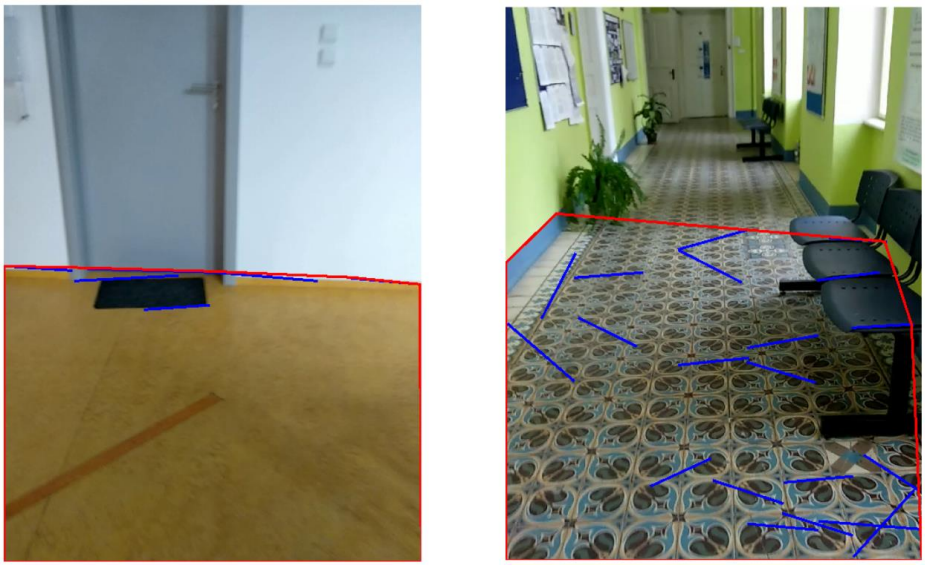
Ako algoritmus sme použili obmenenú Houghovu transformáciu. Tá využíva pre výpočet len určitú podmnožinu náhodne vybraných bodov. Výsledkom sú nájdené segmenty čiar.



Obrázok 8. Segmenty čiar získané obmenenou Houghovou transformáciou spolu s vyznačenou oblasťou podlahy

Na obrázku č.8 sú vyznačené detegované segmenty čiar. Každý segment je definovaný dvomi koncovými bodmi na obraze. Tieto segmenty sme rozdelili na dve skupiny. Horizontálne, na modro vyznačené segmenty a vertikálne, žlté segmenty. Segment bol charakterizovaný ako horizontálny ak jeho uhol natočenia bol väčší než 110° a menší než 70°, vertikálny ak v rozmedzí 70° až 110°. Všetky ostatné segmenty boli označené ako nadbytočné a vo výslednom obraze zobrazené neboli. Vzhľadom na rušivé podmienky v reálnom svete dáva predchádzajúci postup často aj nesprávne segmenty čiar, ktoré nepatria k oblasti podlahy. Bolo potrebné odstrániť ich. Čiary, ktoré mali dĺžku kratšiu ako 60 pixelov sme teda nepovažovali za potrebné a do výsledného obrazu pridané neboli. Mohlo totiž ísť o odlesky svetiel na podlahe alebo o samotné svetlá. Počiatočné a koncové body modrých horizontálnych segmentov sme pospájali, aby sme dostali približný obrys podlahy. Tento postup bol inšpirovaný prácou [3] od Yinxiao Li a Stanley T. Birchfield. Problém nastáva ak je podlaha nehomogénna, napríklad ak obsahuje dlaždice. To znamená veľké množstvo menších detegovaných segmentov čiar v oblasti podlahy, nie len tie, ktoré tvoria hranicu. Pospájaním ich počiatočných a koncových bodov získame nepravidelný mnohostranný útvar, ktorý absolútne neodpovedá tvaru podlahy.

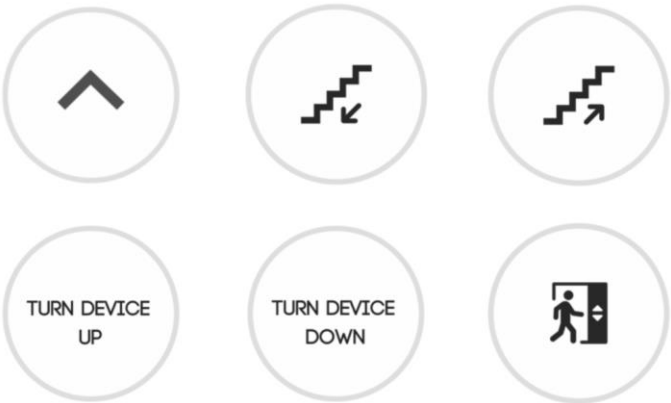
Pre získanie optimálnych výsledkov v takýchto prípadoch sme využili konvexný obal. Ide o minimálny konvexný polygón obsahujúci všetky body množiny, v našom prípade podlahu. Pre ľubovoľné dva body platí, že celá ich spojnica patrí do tejto množiny. K ich vytvoreniu sa využívajú algoritmy ako Gift wrapping, prírastkový algoritmus a niekoľko ďalších. V našom prípade bol využitý algoritmus Gift wrapping. Do úvahy boli brané aj spodné body obrazu.



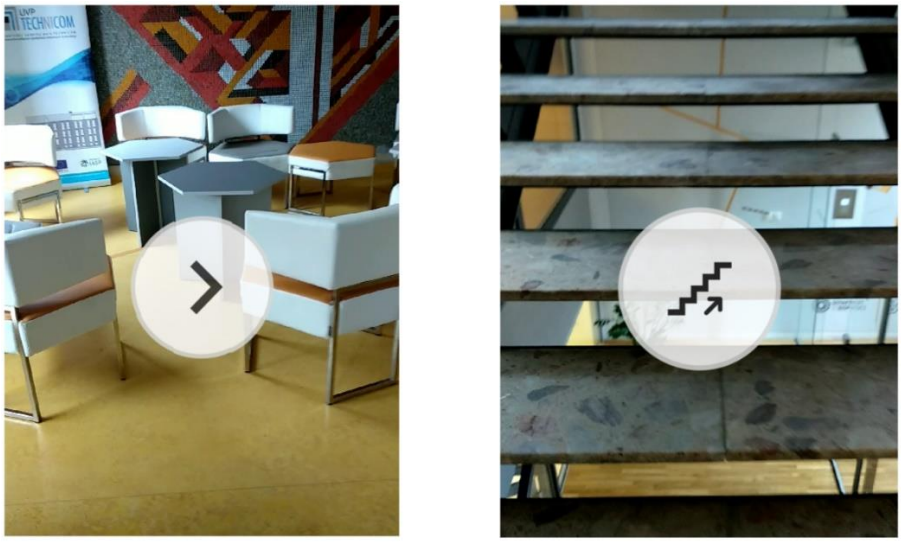
Obrázok 9. Vytvorenie konvexného obalu zo segmentov čiar

2.3 Vykreslenie navigačného prvku na detegovanú podlahu

Tretím krokom je vykreslenie navigačnej šípky, alebo teda navigačného prvku na detegovanú oblasť podlahy. Tento prvok je rozdelený na niekoľko typov podľa aktuálnej situácie, v ktorej sa používateľ nachádza. Ide o klasickú navigačnú šípku situovanú v krúžku, prvky zobrazujúce pohyb po schodoch a prvok pre polohu vo výťahu. Vytvorili sme aj prvky, ktoré upozorňujú používateľa, aby zmenil natočenie mobilného zariadenia. Tento krok bol zavedený kvôli lepšiemu spracovávaniu obrazu. Ak mal používateľ mobilné zariadenie naklonené priveľmi nadol alebo nahor, spracovanie obrazu nedávalo optimálne výsledky. Znižovalo to pravdepodobnosť, že vanishing point a segmenty čiar sú správne detegované.



Obrázok 10. Jednotlivé typy navigačného prvku pre rôzne situácie



Obrázok 11. Navigačný prvok v 2D verzii

Obrázok 12. Navigačný prvok v 3D verzii

Práve pre nestabilnú detekciu podlahy sme sa rozhodli vytvoriť dve verzie navigačného prvku. Verziu 2D a 3D zobrazenia. 2D verzia zobrazuje navigačný prvok v strede obrazu bez akéhokoľvek skreslenia, viď obrázok č.11. V 3D verzii sa navigačný prvok, konkrétne navigačná šípka, vykresľuje na detegovanú podlahu, obrázok č.12. Využíva sa pri tom perspektívna transformácia. K jej uskutočneniu využívame údaj zo zabudovaných senzorov mobilného zariadenia, konkrétne natočenie okolo osi X, dopredu a dozadu. Čím viac má používateľ naklonený mobil smerom nadol, tým viac je navigačná šípka sploštená a opačne. Samozrejme to platí

len do určitého uhla, po jeho prekročení sa zobrazí prvok, ktorý upozorňuje na zmenu polohy zariadenia. Pri verzii 3D sa aplikuje perspektívna transformácie len v prípade navigačnej šípky, ostatné prvky ostávajú nemenné kvôli čitateľnosti ich obsahu.

Členenie navigácie na viacero prvkov, hlavne oblasť schodov a výťahu, sa použilo z dôvodu komplikovanej detekcie hrán a sklonu schodiska. Pôvodne navrhovaným navigačným prvkom bola klasická navigačná šípka bez kruhu. Pri jej vykresľovaní na obraz bol pridávaný tieň, čo dodávalo efekt, že je šípka je situovaná nad podlahou. Riešenie sa stávalo komplikovanejším až pri perspektívnej transformácií, keď navigačná šípka mala ukazovať iným smerom ako vpred. Pri vykresľovaní šípky v kruhu je to jednoduchšie z hľadiska toho, že aj keď je šípka natočená do strany, transformuje sa stále len jej obal, teda kruh. Tým pádom neboli potrebné ďalšie výpočty. Pri obyčajnej šípke bolo potrebné pre každý napr. piaty stupeň prepočítať hodnoty transformácie, aby jej vzhľad vyzeral reálne. Preto sme zvolili navigačnú šípku v kruhu. Navyše pri tomto type je možné dopĺňanie o spomínané prvky ako sú ukazovatele schodov, text a pod pre jednotný vzhľad.

3 Záver

V našej práci bolo cieľom analyzovať a porovnať známe prístupy a existujúce riešenia, ktoré využívajú obraz z kamery mobilného zariadenia a rozšírenú realitu pre navigáciu používateľa v indoorovom prostredí. Tie sú rozdelené na vedecké články a komerčné aplikácie. Väčšina existujúcich riešení bola zameraná na mobilného robota s predvídateľným pohybom a statickými senzormi. My sme ale pracovali so smartfónom a človekom. Preto sme museli naše riešenia upraviť, aby brali do úvahy aj nepriaznivé situácie, ktoré pri navigácií robota nie je potrebné riešiť.

Implementáciu sme rozdelili na tri kroky. Prvým krokom bola detekcia smerovania chodby, aby sme vedeli učiť smer navigačnej šípky. Nasledovala detekcia podlahy, kam v poslednom kroku bola vykreslená navigačná šípka. Zistili sme, že detekcia podlahy nie je triviálnym problémom. Osobitne tejto oblasti je venované určité množstvo vedeckých prác s odlišnými riešeniami. Pri implementácií sme využívali aj zabudované mobilné senzory, ktoré značne pomohli pri zisťovaní orientácie zariadenia. Preskúmali sme a oboznámili sme sa s metódami z oblasti analýzy obrazu, ktoré nám dopomohli k splneniu implementačných krokov.

Rozhodli sme sa vytvoriť komplexný navigačný prvok pozostávajúci z viacerých podôb. Tie záviseli od situácie, alebo oblasti, v ktorej sa používateľ nachádzal. Pre tento krok sme sa rozhodli preto, lebo detekcia podlahy neprinášala vždy optimálne výsledky.

Výsledkom práce je prehľad existujúcich riešení a aplikácií v oblasti indoorovej navigácie ako aj implementácia vlastných riešení a postupov. Zistili sme v akých situáciách je možné získať optimálne výsledky a kedy naopak nie je možné prostredníctvom daných metód očakávať korektný výstup.

Literatúra

1. MULLONI, Alessandro; SEICHTER, Hartmut; SCHMALSTIEG, Dieter. Handheld augmented reality indoor navigation with activity-based instructions. In: Proceedings of the 13th international conference on human computer interaction with mobile devices and services. ACM, 2011. p. 211-220.
2. MÖLLER, Andreas, et al. A mobile indoor navigation system interface adapted to vision-based localization. In: Proceedings of the 11th international conference on mobile and ubiquitous multimedia. ACM, 2012. p. 4.
3. LI, Yinxiao; BIRCHFIELD, Stanley T. Image-based segmentation of indoor corridor floors for a mobile robot. In: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on. IEEE, 2010. p. 837-843.