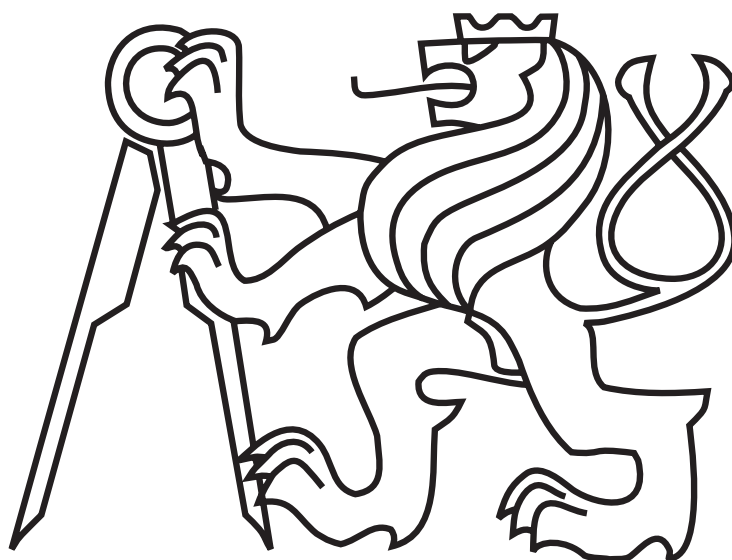


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

BAKALÁŘSKÁ PRÁCE



Lucie Halodová

**Údržba map pro dlouhodobou navigaci
mobilních robotů**

Katedra kybernetiky

Vedoucí práce: **Ing. Tomáš Krajník, Ph.D.**

Květen, 2018

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Halodová** Jméno: **Lucie** Osobní číslo: **457087**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra kybernetiky**
Studijní program: **Kybernetika a robotika**
Studijní obor: **Robotika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Údržba map pro dlouhodobou navigaci mobilních robotů

Název bakalářské práce anglicky:

Continuous Map Update for Long-Term Mobile Robot Visual Navigation

Pokyny pro vypracování:

Tato práce se bude zabývat metodami, které umožňují mobilním robotům vytvářet, aktualizovat a vylepšovat své modely prostředí tak, aby se udržely s přirozeně se měnícím prostředím. Tyto modely budou obsahovat významné elementy prostředí (orientační body) použité v metodě teach-and-repeat pro vizuální navigaci mobilních robotů.

Navržená metoda by měla být schopná efektivně integrovat dlouhodobé pozorování více orientačních bodů do reprezentace, která obsahuje dostatečnou informaci k vykonání vizuální teach-and-repeat navigace. Tato metoda bude zaintegrovaná do Robotického Operačního Systému (ROS) a experimentálně ověřena na reálné robotické platformě.

- 1) Seznamte se s principy metody teach-and-repeat pro vizuální navigaci mobilních robotů [1,2].
- 2) Seznamte se s metodami, které umožňují držet krok se změnami prostředí [1,3,4,5,6].
- 3) Seznamte se s dostupnými rámci vizuální navigace mobilních robotů.
- 4) Vyberte sadu perspektivních navigačních metod a metod aktualizujících mapu, kombinujte je a proveďte jejich srovnání na veřejně dostupných datasetech.
- 5) Zaintegrujte nejlepší metody do Robotického Operačního Systému a experimentálně je vyhodnoťte na reálném robotovi.

Seznam doporučené literatury:

- [1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard. Past, present, and future of simultaneous localization and mapping: Towards the robust perception age," IEEE Transactions on Robotics, 2016.
- [2] T. Krajník, J. Faigl, V. Vonasek et al., Simple, yet Stable Bearing-only Navigation," Journal of Field Robotics , 2010.
- [3] P. Furgale and T. D. Barfoot, Visual teach and repeat for long-range rover autonomy," Journal of Field Robotics , 2010.
- [4] S. Lowry, N. Sunderhauf, P. Newman, J. Leonard, D. Cox, P. Corke, and M. Milford. Visual place recognition: A survey," IEEE Transactions on Robotics, 2015.
- [5] P. Muhlfellner, M. Bürki, M. Bosse, W. Derendarz, R. Philippsen, and P. Furgale. Summary maps for lifelong visual localization. Journal of Field Robotics , 2015.
- [6] F. Dayoub and T. Duckett, An adaptive appearance-based map for longterm topological localization of mobile robots," in Proc. of Int. Conference on Intelligent Robots and Systems (IROS) , 2008.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Tomáš Krajník, Ph.D., centrum umělé inteligence FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **11.01.2018**

Termín odevzdání bakalářské práce: _____

Platnost zadání bakalářské práce: **30.09.2019**

Ing. Tomáš Krajník, Ph.D.
podpis vedoucí(ho) práce

doc. Ing. Tomáš Svoboda, Ph.D.
podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studentky

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne.....

Poděkování

Ráda bych poděkovala všem lidem, kteří mě během mé bakalářské práce podporovali a motivovali pro její dokončení. Jmenovitě mému vedoucímu Tomášovi Krajníkovi za individuální přístup a množství získaných dovedností, Jitce Hodné za morální podporu a Tomášovi Roučkovi za neobvyklou trpělivost.

Abstrakt

Tato práce se zabývá metodami, které umožňují vytvářet a udržovat mapy pro dlouhodobou navigaci mobilních robotů v proměnlivém prostředí, které podléhá změnám světelných podmínek, počasí i procesům spojeným se střídáním ročních období. Hlavním problémem je práce s neurčitostí, jejíž akumulaci v mapě je nutné zabránit. Tato neurčitost vzniká z nepřesností senzorických dat nebo v důsledku přirozených procesů dynamického prostředí. V této práci je prezentována metoda adaptivního mapování, která je schopna integrovat do mapy nové informace, díky kterým se může přizpůsobovat okolí. Zároveň z ní také odebírá zastaralé či již nerelevantní informace, které kontaminují mapu a zhoršují výslednou spolehlivost lokalizace a robustnost navigace. Tato metoda je porovnána s dalšími typy map a na získaných datasetech je experimentálně ověřena a vyhodnocena v závislosti na kritériích souvisejících se změnou prostředí a chybou lokalizace.

Abstract

The goal of this thesis is to maintain methods for creating and updating maps for long-term mobile robot navigation in the changing environment, which is affected by illumination, weather conditions and seasonal changes. The main problem is to cooperate with uncertainty, whose accumulation in the map is necessary to be prevented. This uncertainty arises in the inaccuracy of sensory data or as a result of natural processes of a dynamic environment. In this work, a method of adaptive mapping is presented, which is able to integrate new information into the map, thanks to which it is able to fit its surroundings. At the same time, it also removes obsolete or no more relevant information from the map, which contaminates and worsens the final reliability of localization and robustness of navigation. This method is compared with other types of map and it is experimentally verified on the gathered dataset and evaluated according to the criteria related to environmental changes and the error of localization.

Obsah

1	Úvod	1
2	Přehled principů užívaných v navigaci mobilních robotů	3
2.1	Lokalizace	3
2.1.1	Relativní lokalizace	3
2.1.2	Absolutní lokalizace	5
2.2	Mapování	9
2.2.1	Metrické mapy	9
2.2.2	Topologické mapy	11
2.2.3	Hybridní mapy	12
2.2.4	Tvorba mapy	14
2.3	Současná lokalizace a mapování - SLAM	14
2.3.1	Rozšířený Kalmanův filtr (EKF)	15
2.3.2	SLAM založený na grafu s optimalizačními technikami	16
2.3.3	Částicové filtry	16
2.4	Plánování pohybu	17
2.5	Navigace	18
2.5.1	Navigace bez mapy	19
2.5.2	Navigace na základě mapy	19
2.5.3	Dlouhodobá navigace v dynamickém prostředí	20
2.6	Zpracování obrazu pro vizuální navigaci	22
3	Správa map	26
3.1	Navigace podle statické mapy	27
3.2	Navigace podle plastické mapy (Remapping)	27
3.3	Adaptivní mapa	28
3.3.1	Odebírání elementů z mapy	29
3.3.2	Přidávání prvků do mapy	31

4	Popis navigačního systému	32
4.1	Zpracování obrazu	33
4.2	Fáze učení - mapování	33
4.3	Autonomní navigace	33
4.4	Implementace systému	34
4.5	Rozšíření systému	36
5	Datasety	39
5.1	První dataset - navigace podle statické mapy	39
5.2	Druhý dataset - navigace podle statické, plastické a adaptivní mapy	40
5.3	Třetí dataset - navigace podle adaptivní mapy	41
6	Experimenty	43
6.1	Vytvoření dat pro analýzu	43
6.2	Statistické testování dat	44
6.3	Analýza a testování kritérií	45
6.3.1	Relevance mapy	45
6.3.2	Přesnost mapy	46
6.3.3	Chyba lokalizace	47
6.3.4	Přesnost navigace	49
6.4	Shrnutí a diskuze experimentálních výsledků	50
7	Závěr	53

Seznam obrázků

1	Rotační enkodér	4
2	Senzory inerciální navigace	5
3	Kompasy	6
4	Senzory aktivních majáků	7
5	Orientační body	8
6	Metrické mapy	10
7	Reprezentace Oktomapy [1]	10
8	Porovnání mapy tvořené metodou mřížky obsazenosti a geometrické mapy [2]	11
9	Topologická mapa [3]	12
10	Hybridní mapa [4]	13
11	Autonomní průzkum a vytváření mapy [5]	15
12	Reprezentace pozice robota pomocí SLAM filtrů [6]	16
13	Výsledná cesta z RRT metody [7]	18
14	SIFT: porovnávání pixelů se sousedy z různých škál [8]	23
15	FAST detektor: zvýrazněné pixely představují okolí od klíčového bodu označeného písmenem C. Čárkovaná čára zobrazuje dostatečný počet dvanácti pixelů, které jsou světlejší než pixel ve středu kružnice [9].	24
16	MSER detektor: hledání stabilních oblastí [10]	25
17	Snímek obrazovky z navigačního systému. Horní obrázek pochází z mapy a dolní z aktuálního průjezdu. Zelené spojnice znázorňují správné shodné dvojice a červené nesprávné.	30
18	Hybridní mapa v používaném navigačním systému	34
19	Zjednodušené schéma při mapování	35
20	Zjednodušené schéma při navigování	36
21	Schéma rozšířeného systému	37
22	Jednotlivé pohledy z průjezdu první trasou	40
23	Jednotlivé pohledy z průjezdu druhou trasou	41
24	Pohled na místo s třetí trasou	42

SEZNAM OBRÁZKŮ

25	Pravděpodobnost, že počet správných shodných dvojic je vyšší než daná hodnota	45
26	Pravděpodobnost, že poměr správných shodných dvojic vůči celkovému počtu vytvořených dvojic je větší než daná hodnota	47
27	Pravděpodobnost, že chyba posunutí bude menší než daný počet pixelů	48
28	Chyba pozice robota v závislosti na počtu průjezdů po naučené trase podle externího lokalizačního systému	50
29	Měření přesnosti navigace pomocí externího lokalizačního systému [11]	51

1 Úvod

Pro dnešní svět je typické, že manuální práci vykonávají roboti. V továrnách u běžících pásů můžeme vidět nejrůznější roboty, které mohou mít za úkol například jenom přenášet balík, skládat výrobky do krabice či svářet těžké plechy pro výrobu automobilů. Manuální práce je typicky spojena s nějakou opakující se činností, kterou snadno můžeme naučit právě robota. Navíc po zavedení této automatizace pak není vyžadovaná kontrola člověkem. V dnešní době samozřejmě existují i roboti, kteří nevykonávají žádnou manuální činnost, ale mohou pouze sloužit lidem, jako například systém Alexa, který umí na základě hlasové interakce přehrávat hudbu či třeba poskytovat informace o počasí. Dalším příkladem mohou být roboti, kteří se snaží navázat sociální kontakt s člověkem, jako například robot NAO, který se používá i pro edukativní účely.

V dnešní době je snaha vytvářet stroje, které jsou inteligentní a autonomní. Doposud se však řeší pouze problém částečně inteligentních nebo částečně autonomních robotů. Tito roboti typicky operují v jednodušších strukturovaných prostředích s nízkou mírou neurčitosti, anebo vyžadují kontrolu člověkem.

Jednou z oblastí robotiky je i mobilní robotika zabývající se roboty, kteří se umí pohybovat. Aby jejich chování bylo autonomní, často je tato oblast spojena s navigací. Pro inteligentní navigaci je typicky nutné mít mapu, díky které může robot určit svou pozici v prostoru a dále podle ní plánovat pohyb tak, aby dosáhl požadovaného cíle.

Jedním ze způsobů navigace je i metoda *teach-and-repeat*, na které je založený navigační systém v této práci. V této metodě nejdříve naučíme robota, jaký proces má vykonávat, a ten je pak schopen ho autonomně opakovat. Nejdříve si robot během manuálně naučené trasy vytvoří mapu, podle které pak umí určovat svoji pozici a plánovat pohyb tak, aby byl schopen se navigovat po naučené trase.

Tento proces může být využitý v několika různých odvětvích. Od zemědělství, ve kterém lze použít pro traktor, který orá pole, až po drony, které mají za úkol například monitorovat skladiště. Ve všech těchto procesech stačí robota naučit trasu, kterou chceme, aby pak autonomně opakoval.

Nicméně pokud chceme takového robota dlouhodobě autonomně navigovat, je nutné uvažovat i změny prostředí. Pokud se robot naviguje pouze podle staré naučené trasy, pak nevyhnutelně musí brzy dojít k nějaké nehodě v důsledku vývoje vzhledu okolí. Proto je nanejvýš vhodné řešit, jak aktualizovat, udržovat a spravovat mapy, podle kterých se robot lokalizuje

a pohybuje. Tento problém, i přestože je velmi diskutovaný, zůstává stále nevyřešen, a je tak hlavní motivací méj práce.

Cílem této práce je vylepšit již používaný navigační systém tak, aby byl schopen dlouhodobé robustní navigace i v proměnlivých podmínkách. Robot by měl být schopen se podle jedné mapy navigovat i v exteriéru, ve kterém jsou přirozenými procesy změna světelných podmínek, počasí, změny ročního období i změny vyvolané například pohybem člověka. Všechny tyto změny prostředí můžeme spolu s nepřesnostmi měření ze senzorů považovat za zdroje neurčitosti, které negativně ovlivňují kvalitu navigace, a proto je žádoucí je potlačit. Aby to bylo možné, je nutné postupně integrovat do mapy nové informace, které robot obdrží při opakovaných průjezdech. Nicméně je nutné uvažovat, aby nově přidané informace mapu nekontaminovaly a nezhoršily tak celkovou schopnost a kvalitu navigace. Zároveň je vhodné z mapy odebírat ty informace, které přestávají být relevantní k naučené trase, a tedy mohou také zhoršovat výslednou navigaci.

Struktura práce je rozdělena do pěti hlavních částí. Následující sekce uvádí čtenáře do problematiky navigace v mobilní robotice. Zájem je zde soustředěn hlavně na dlouhodobou vizuální navigaci v proměnlivém prostředí, pro níž je vhodné čtenářovi představit různé druhy map a metod lokalizace. Druhá kapitola se zabývá problémem správy map. V této kapitole je představena nově navrhnutá metoda tvorby mapy vycházející ze dvou již známých metod, které jsou nedostačující pro dlouhodobou navigaci v měnícím se prostředí. Její zaintegrování do již existujícího navigačního systému je popsáno ve třetí kapitole. Další sekce obsahuje představení tří vytvořených datasetů při provedených experimentech. Na těchto datasetech je v poslední části provedeno zhodnocení a porovnání jednotlivých map vzhledem k navigaci ve vyvíjejícím se okolí.

2 Přehled principů užívaných v navigaci mobilních robotů

Cílem této kapitoly je shrnout principy používané v navigaci mobilních robotů a vymezit tak konkrétní úlohu této bakalářské práce.

Jedním z klíčových problémů mobilní robotiky je autonomní navigace. K navigaci po předem určené trase potřebuje robot znát svou relativní pozici vůči trase a požadovanou polohu cílové destinace. Požadavek navigace tedy souvisí se dvěma problémy řešenými v robotice, a to s odhadem vlastní pozice v prostoru (tj. lokalizace) a s plánováním pohybu. Pro řešení těchto problémů potřebuje robot mapu, ve které následně může odhadovat svoji pozici a plánovat další pohyb. Porovnáním mapy se sensorickou informací je možné robota lokalizovat. Integrace sensorické informace do mapy se potom nazývá mapování. Je obtížné pracovat se sensorickými informacemi kvůli neurčitosti a nepřesnosti senzorů. Tuto neurčitost je potřeba uvažovat při zpracování sensorických dat, jinak dochází k vytváření nekvalitních map nepoužitelných k lokalizaci a plánování pohybu robota. Při integraci sensorické informace je tedy nutné řešit potlačení šumu, a to ať už jde o integraci v matematickém smyslu, kdy chyba postupně narůstá, či pouze jako zaintegrovaní nesprávné informace ze senzoru do mapy. Lokalizace a mapování jsou velmi úzce propojeny a pokud jsou studovány dohromady, pak se jedná o SLAM (Současná lokalizace a mapování).

2.1 Lokalizace

Lokalizace je proces určení aktuální polohy robota. Lokalizační techniky lze rozdělit do dvou oblastí dle způsobu určení pozice robota na relativní a absolutní lokalizaci. Též se toto rozdělení někdy označuje jako kontinuální a globální lokalizace [3].

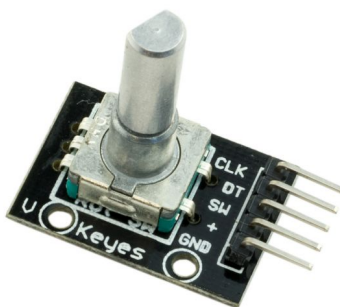
2.1.1 Relativní lokalizace

Relativní lokalizace je určení pozice robota vzhledem k jeho počáteční poloze. Závisí na předchozím odhadu polohy robota, chybu odhadu je nutné v průběhu jízdy opravovat. Tento problém se také nazývá sledování pozice (position tracking). Relativní lokalizace je snazší než absolutní díky menší paměťové a časové náročnosti.

Z hlediska relativního měření pozice (též zvané dead-reckoning) můžeme rozlišit dvě základní metody sensorického přístupu [12].

Odometrie

Odometrie je metoda, která využívá data ze senzorů pohybu (např. optické, magnetické enkodéry) k určení pozice robota. Abychom dostali pozici robota, je nutné integrovat získanou rychlost, čímž vzniká lineární chyba. Tento princip integrace inkrementálního pohybu přes čas vede k akumulaci chyby a jejímu narůstání s větší ujetou vzdáleností. Chyba je také způsobena vlivem prokluzování kol robota, která se projevuje především v exteriéru. Není to tedy vhodná metoda z hlediska dlouhodobé navigace. Přesto je její použití v této oblasti velmi rozšířené díky své jednoduchosti a relativně nízké ceně jako např. v [13]. V [14] se používá odometrie na rozdíl od jiných senzorů pro lokalizaci, protože robot se využívá v radioaktivním prostředí, kde není možné používat jiný senzor pro lokalizaci (např. předem rozmístěné aktivní majáky). Pro krátkodobou navigaci má odometrie navíc velmi dobrou přesnost.



Obrázek 1: Rotační enkodér

Inerciální navigace

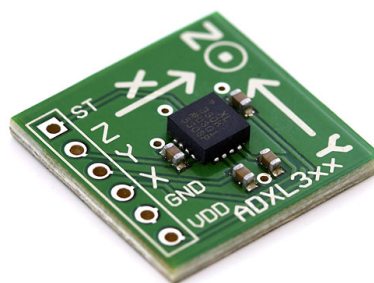
Senzory používané v inerciální navigaci (akcelerometr, gyroskop) měří zrychlení. Pro získání pozice musíme zrychlení dvakrát zintegrovat, což způsobuje kvadratickou chybu a hlavní nevýhodu těchto senzorů.

Akcelerometry jsou výhodné svou nízkou cenou, schopností fungovat jako samostatné zařízení a oproti odometrii umí měřit ve více směrech. Jsou však citlivé na nerovnosti povrchu a mají značný šum při nízkých zrychleních. Při vyšším zrychlení je tento šum méně významný [15]. Proto jsou pro účely dlouhodobé lokalizace vhodné pouze jako doplňkový senzor.

Gyroskopy měří úhlové zrychlení a díky zjištěné orientaci v prostoru jsou schopny vyvážit orientační chybu odometrie. Jsou však nevýhodné svou velikostí, cenou nebo váhou a jejich nepřesnost zatěžuje Coriolisova síla, která urychluje či zpomaluje zrychlení. Velmi přesné a vhodné pro mobilní robotiku jsou laserové či optické gyroskopy, které překonávají některé problémy gyroskopů. Jejich nevýhodou je zejména ne příliš nízká cena. Využití je např. v [16].



(a) Remote-control gyroskop



(b) Modul akcelerometru

Obrázek 2: Senzory inerciální navigace

2.1.2 Absolutní lokalizace

Absolutní lokalizace je určení pozice bez žádné znalosti o předchozí pozici robota. Robot musí určit svou polohu ve světovém souřadném systému. Oproti relativní lokalizace je výhodné, že se zde neakumuluje chyba z předchozího odhadu. Tento problém určení pozice se také nazývá problém "uneseného" robota ("kidnapped-robot problem"). Měření absolutní pozice lze rozdělit do tří základních oblastí - kompas, lokalizace dle majáků a lokalizace dle orientačních bodů.

Kompas

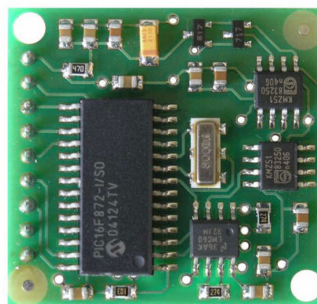
Kompas je zařízení, které umožňuje určit směr, typicky na základě magnetického pole. Pro navigaci je směr robota nejdůležitějším parametrem, proto

2.1 Lokalizace

je magnetický senzor obvykle důležitý pro většinu autonomních vozidel řešících úlohu navigace. Magnetický kompas umožňuje vlivem magnetizace ukazovat světový sever, jeho nevýhodou je však zkreslení např. vlivem ocelových materiálů či lokálních magnetických anomálií. Také nejsou příliš vhodné pro použití ve vnitřních prostorech. Pro mobilní robotiku je z této kategorie nejvhodnější fluxgate kompas, který má relativně nízkou cenu, nízkou spotřebu energie a zároveň je schopný odolávat otřesům a vibracím, tedy lze ho za určitých podmínek (např. tlumení) používat i v kopcovitém terénu [17]. Jiným příkladem kompasu může být senzor založený na slunečním záření, který snímá pozici slunce. Takovýto senzor byl použitý na Marsu pro navigaci robota, protože je zde velmi slabé magnetické pole, podle kterého se nelze navigovat [18].



(a) První kompas objevený v Čínské dynastii



(b) Modul magnetického kompasu

Obrázek 3: Kompas

Lokalizace podle majáků

Rozlišujeme dva typy majáků - aktivní a pasivní. *Pasivní majáky* nevyšílají žádné signály, příkladem může být čárový kód, který je detekovaný laserovým skenerem [19]. *Aktivní majáky* vysílají či přijímají signály a můžeme je z hlediska techniky rozdělit na trilateraci a triangulaci.

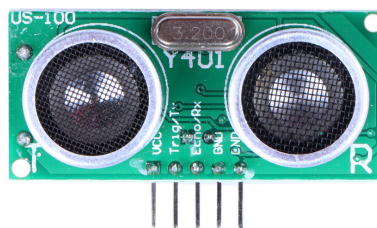
Pokud se měří vzdálenost robota od známých pozic majáků, jedná se o **trilateraci**. Na trase je umístěno několik majáků (alespoň tři) na předem známých pozicích a přijímač signálu z majáku je potom přímo na robotovi projíždějícím trasou. Z doby letu signálu se vypočte vzdálenost vysílače a přijímače. Příkladem trilaterace může být globální systém pro určování polohy (GPS).

Triangulace je naopak založena na rotačním sensoru umístěném typicky na robotovi, který přijímá signály z vysílačů (typicky aktivních majáků) na trase pod různými úhly. Na základě alespoň tří úhlů je potom možné vypočítat souřadnice robota a jeho směr. Velmi častý problém je, že aktivní majáky na trase nemusí být vidět, a tedy podmínka tří úhlů (ze tří viditelných majáků) nemusí být splněna. Optické triangulační systémy jsou široce rozšířené v mobilní robotice. Několik příkladů je uvedených v přehledu optických systémů užívaných k určování pozice [20].

Globální systém určování polohy (GPS) se skládá z několika satelitů vysílajících vysokofrekvenční signál se zakódovanou informací o jejich aktuální poloze. Využitím trilaterální metody stačí, aby přijímač umístěný na Zemi přijmul signál ze tří satelitů. Poté lze z jejich přesné vzdálenosti dopočítat zeměpisnou šířku, délku a nadmořskou výšku. GPS má dnes přesnost 10-20 metrů. Přesnější metodou je např. diferenciální GPS (DGPS) s lokální korekcí zjišťující odchylky GPS od skutečného stavu, která má při nejlepší implementaci přesnost pod 10 cm. GPS je velmi používaná pro řešení úloh navigace v exteriérních podmínkách [21], [22]. Nevýhodou při použití v mobilní robotice je hlavně příjem signálu ze satelitů v kopcovitém terénu, mezi budovami či interference signálů.



(a) GPS modul



(b) Ultrazvukový senzor

Obrázek 4: Sensory aktivních majáků

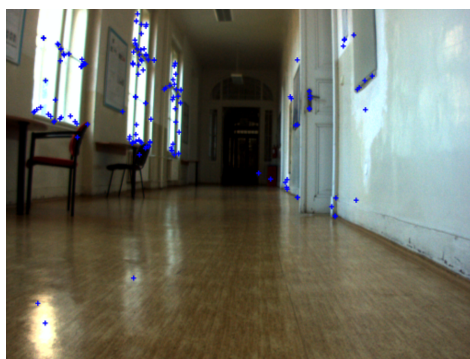
Dalším přístupem může být lokalizace pomocí akustických majáků. Např. v [23] užívají *sonar* jako **ultrazvukový senzor** pro lokalizaci. Sonar je výhodný zejména tím, že je levný. Pro určení pozice jsou dostatečné dva vysílače a dva přijímače. Z doby letu signálu je následně možné vypočítat vzdálenost

přijímače od vysílače. Sonar se využívá hlavně ve vnitřních prostorech, protože není vhodný pro dlouhé vzdálenosti a je ovlivňovaný deštěm a větrem.

Lokalizace dle orientačních bodů (Landmark Navigation)

Orientační bod je významný bod v okolí, který lze snadno rozlišit od jiných bodů. Typicky jde o nějaký geometrický objekt s vysokým kontrastem vzhledem k pozadí. Tyto body (resp. jejich relativní pozice vůči robotovi) získané ze senzoru (typicky kamera umístěná na platformě) jsou porovnány se známou mapou. Lokalizace robota potom funguje na principu vypočtení pozice robota na základě porovnání pozice orientačních bodů z mapy a z pozice aktuálně viděných orientačních bodů kamerou. Často se tato metoda kombinuje s lokalizací pomocí odometrie, která určí přibližnou polohu a směr robota. Orientační body mohou být dvou typů - přirozené a umělé.

Přirozené orientační body jsou takové geometrické objekty, jejichž primárním účelem není lokalizace. Příkladem mohou být dveře, jejichž hlavní funkcí je průchod skrz ně do budovy. Typicky se z prostředí získávají hrany či rohy dveří, oken, zdí a podobných objektů díky vyššímu kontrastu vzhledem k pozadí. Ovšem na budovách je několik oken vedle sebe v podobné výšce a s podobným okolím a tak snadno může dojít k popletení získaných orientačních bodů a v důsledku toho také ke zkreslení lokalizace. Čím přesnější bude výběr orientačních bodů, tím jednoznačnější bude jejich detekce, popis jejich okolí a jejich porovnávání s mapou ke zjištění jejich shody.



(a) Přirozené orientační body v místnosti [6] (b) Čárový kód jako orientační bod [19]

Obrázek 5: Orientační body

Umělé orientační body jsou oproti tomu speciální značky dodané do daného prostředí pro zvýšení spolehlivosti lokalizace. Snadněji se detekují

díky optimálnímu kontrastu a známému tvaru. Z tohoto důvodu je lokalizace podle těchto bodů velmi přesná. Tvary se mohou lišit svým uspořádáním i tvarem, používají se kruhy [24], obdélníky či různé kontrastní barvy [25]. Umělým orientačním bodem může být i třeba čárový kód detekovaný laserovým skenerem [19].

Přesnost lokalizace podle orientačních bodů klesá s rostoucí vzdáleností. Tento problém se obvykle řeší vytvářením lokálních podmap, díky kterým je potom možné provádět lokalizaci s bližšími orientačními body. Detekce přirozených orientačních bodů bývá výpočetně náročná, což omezuje nasazení robotů s nízkým výpočetním výkonem do dynamických prostředí. Také je závislá na počasí, a tedy orientační body nemusí být dobře detekovány a mohou tak způsobit chybu lokalizace při porovnávání orientačních bodů s mapou, která není aktualizována. Použití aktivních majáků oproti této metodě tedy může být výhodné v lokalizaci na delší vzdálenost či rychlejším zpracováním dat. Lokalizace dle orientačních bodů je tedy vhodné doplnit ještě jinou technikou pro určování pozice, typicky se doplňuje odometrií.

Specifickou oblastí je lokalizace podle orientačních bodů získaných z kamery - neboli význačných bodů (features). V této metodě se porovnávají význačné body detekované z obrazu kamery na robotovi s význačnými body v již vytvořené globální mapě. Pokud se najde shoda mezi těmito význačnými body, potom je možné vypočítat polohu robota. Výhodou této metody je, že prostředí nemusí být předem nijak upravováno. Nevýhodou je požadavek na přesnost mapy včetně požadavku na stabilní a snadno rozeznatelné význačné body pro hledání jejich shody (matching) s význačnými body aktuálně získanými z kamery umístěné na robotovi. .

2.2 Mapování

Mapování je proces, při němž dochází k vytváření map. Při tomto procesu jsou senzorické informace typicky po nějakém dalším zpracování zaneseny do mapy. Mapy můžeme rozdělit do dvou základních oblastí na topologické a metrické mapy. [3]

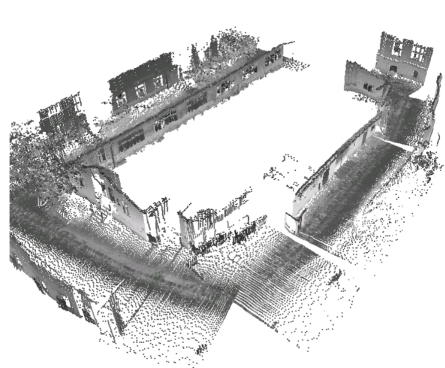
2.2.1 Metrické mapy

Metrické mapy jsou reprezentovány jako souřadnice ve 2D nebo 3D prostoru. Pomocí metrických transformací jsou potom ze souřadnicových dat získány pozice překážek a objektů. Výhodou metrických map je jejich přehlednost a využitelnost i pro jiné roboty. Svým vzhledem také připomínají

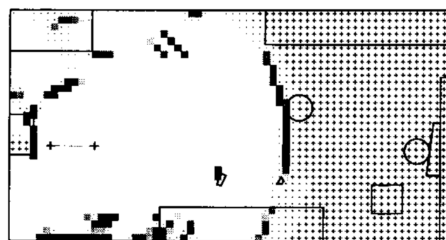
2.2 Mapování

architektonické plány a jsou tedy snadno pochopitelné i pro člověka. Je také snadné do takovýchto map integrovat data z více senzorů. Mezi hlavní nevýhody patří zejména nepřesnosti vlivem šumu sensorických dat.

Jeden ze způsobů reprezentace metrických map může být *3D bodové úložiště*. V [26] využívají 3D bodové úložiště pro vytvoření mapy domácího prostředí. Data z laserového skenu transformují do 3D modelu prostředí.



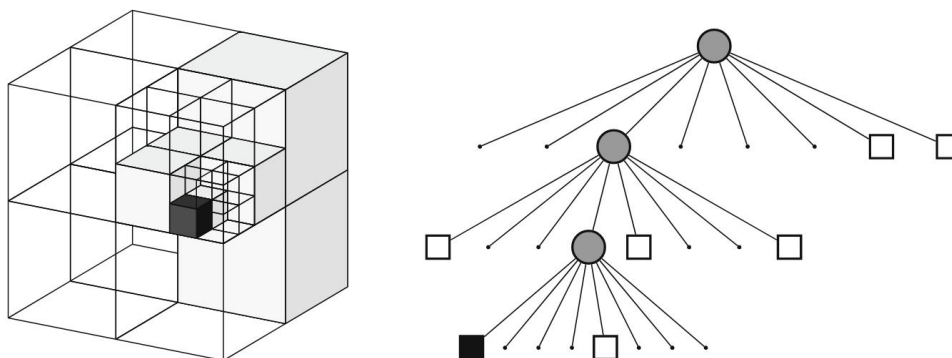
(a) 3D bodové úložiště [6]



(b) Dvoudimenzionální mřížka obsazenosti vytvořená ze sensorických dat sonaru. Bílá barva reprezentuje prázdné místo, černá obsazené místo a symbol "+" nezmapované místo [27].

Obrázek 6: Metrické mapy

Dalším způsobem ukládání znalostí o prostředí do mapy je *mřížka obsazenosti* [27]. Prostředí je diskretizováno a rozděleno do několika políček. Každé políčko potom obsahuje informaci o pravděpodobnosti výskytu objektu na této pozici. Tato metoda sice vyžaduje hodně paměti, ale na druhou stranu je výpočetně nenáročná např. oproti extrahování význačných bodů z obrázků.

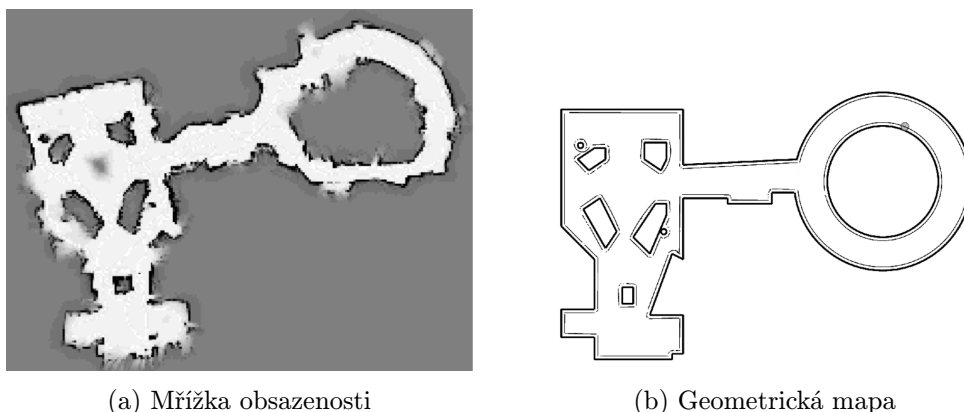


Obrázek 7: Reprezentace Oktomapy [1]

Dalším zajímavým přístupem je vytváření takzvaných "Oktomap", které

se snaží řešit problém paměťové náročnosti efektivním algoritmem. V [1] vytváří 3D model prostředí založený na mřížce obsazenosti. Prostor rozdělují na oktant, který zároveň reprezentují pomocí stromové struktury. Pokud jsou všichni potomci jednoho uzlu stejného stavu (obsazený nebo prázdný), potom je takto označen celý uzel reprezentující jeden oktant, jinak jsou potomci dále větveni. Tímto způsobem je zajištěno významné snížení počtu uzlů, které musí být vyhodnoceny, a také paměťové náročnosti celé struktury.

Další možností je vytváření *geometrické mapy*. V této metodě je prostředí modelováno sadou geometrických primitiv jako např. přímka, oblouk či polygon. Tyto mapy jsou více abstraktní, neobsahují tolik detailů jako mřížka obsazenosti a reprezentují pouze významné oblasti. Oproti mřížce obsazenosti jsou méně paměťově náročné a více kompaktní, ale je složitější je vytvořit, protože vyžadují interpretaci sensorických dat. Je také výpočetně méně náročné v takové mapě provádět plánování pohybu [2].



Obrázek 8: Porovnání mapy tvořené metodou mřížky obsazenosti a geometrické mapy [2]

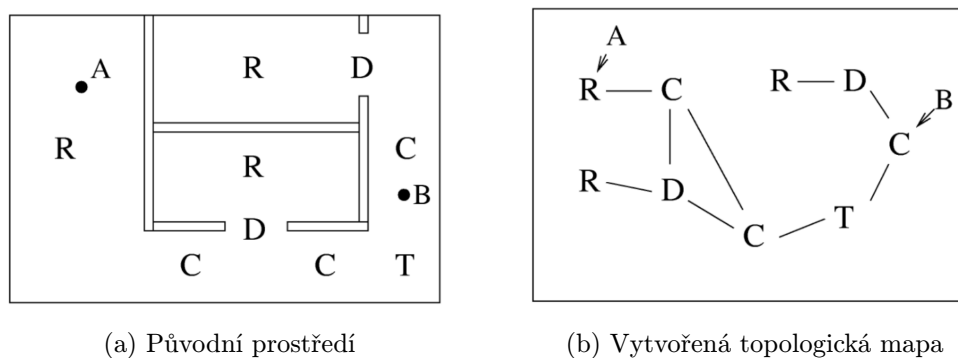
Mezi metrické mapy také patří mapy tvořené z uložených orientačních bodů, které slouží zejména pro lokalizaci nebo navigaci. Tato oblast byla více diskutována v sekci 2.1.2.

2.2.2 Topologické mapy

V topologických mapách je model prostředí reprezentován jako graf, ve kterém uzly představují rozlišitelná místa a hrany představují spojnice mezi těmito místy. Tyto mapy jsou méně detailní a jsou tedy velmi abstraktní. Způsob vytváření mapy vede na diskretizaci prostředí a je také vhodný pro plánování pohybu díky malé výpočetní náročnosti v porovnání s metrickými

mapami. Pojmenovaná místa mají při plánování též snazší význam pro člověka, ale kvůli vysoké míře abstraktnosti jsou pro člověka hůře čitelné. Nevýhodou tohoto typu mapy je její malá objektivnost a těžší rozpoznávání míst, kde typicky záleží na úhlu pohledu. Mapa proto nemůže být použita jiným robotem, model prostředí je potřeba vytvořit daným robotem.

Příkladem topologické mapy může být mapa metra, dopravní síť či průchod místnostmi v domě. V obrázku 9 je příklad reprezentace topologické mapy části budovy. V obrázku 9a je skutečné prostředí, ve kterém jsou vyznačeny dvě pozice A a B. Dále místnosti jsou reprezentovány písmenem R, chodby jsou označeny jako C, rohy chodby jako T a dveře jako D. Topologická mapa takového prostředí je potom na obrázku 9b.



Obrázek 9: Topologická mapa [3]

Topologické mapy bývají většinou kombinovány s metrickými mapami, které jsou použity pro vytvoření rozlišitelných míst. V článku [28] se autoři ale zabývají zejména vytvářením mapy tras pomocí takzvaných *bran*. Jako brány označují místa přechodu z jednoho místa v prostředí do jiného a detekují je pomocí sonaru. Tyto brány se od sebe navzájem nemusí vůbec odlišovat, pro navigaci je důležité jen rozpoznat, že jde o bránu. Další možností je dát robotovi větší volnost v rozpoznávání dosažených míst. Místa jsou v tomto případě definována jako situace popsaná senzory a dosažení nového místa je poté zaznamenáno změnou této situace.

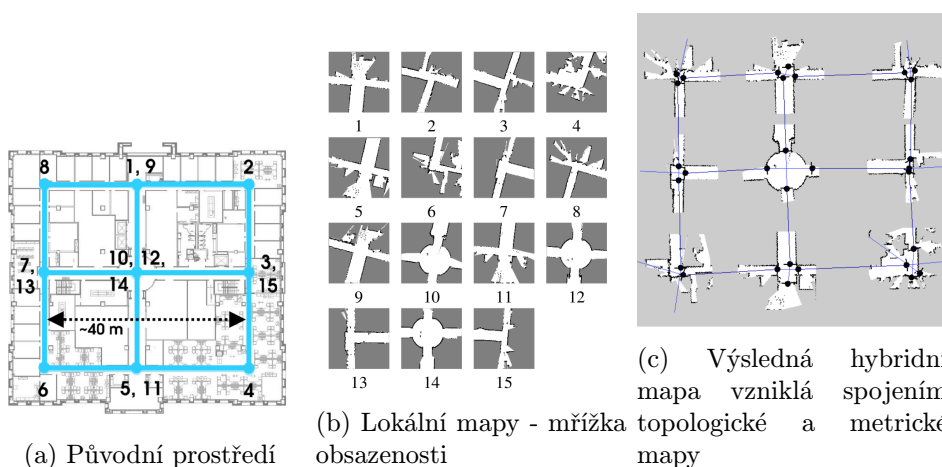
2.2.3 Hybridní mapy

Speciální oblastí jsou hybridní mapy, které jsou kombinací metrických a topologických map. Spojení těchto map je v mobilní robotice populární díky snazšímu zpracování map.

2.2 Mapování

V [29] používají kombinaci metrické mřížky obsazenosti s topologickou mapou. Nejdříve z dat sonarového senzoru vytvoří mřížku obsazenosti. Následně teprve vytváří topologickou mapu na základě již vytvořené mřížky obsazenosti, která je rozdělena do několika regionů. Na rozhraní těchto regionů je poté snadné určit např. průchody chodbou. Jiný přístup [4] kombinuje lokální metrické mapy a globální topologické mapy pro vytváření map pro metodu SLAM (viz sekce 2.3). Topologická mapa je tvořena pomocí bran a organizována stromovou strukturou, metrické mapy podávají detailnější informace o lokálním místě.

Zajímavým konceptem je *Prostorově sémantická hierarchie (SSH)* modelující znalosti o rozsáhlých prostředích ve více úrovních [30]. Úroveň řízení reprezentuje vzory interakce sensorimotoru se světem pro řízení pohybu a tvoření mapy. V této úrovni jsou stavěny lokální geometrické mapy. Kauzální úroveň zajišťuje kontinuální chování mezi různými stavy tohoto diskrétního modelu. Na topologické úrovni jsou pak místa a cesty mezi nimi. Vzory chování na této úrovni jsou propagovány do kauzální úrovně. Uzly SSH reprezentace obsahují tedy množinu objektů a vztahů mezi nimi a celkově tak tvoří kognitivní model.



Obrázek 10: Hybridní mapa [4]

V [31] používají metodu *Atlas* podobnou konceptu SSH, ve které jsou uzly tvořeny souřadnicovým "rámem". Tento rám obsahuje lokální mapu prostředí, pozici robota a s nimi spojené neurčitosti. Hrany představují transformace mezi přilehlými rámy, které obsahují některé stejné informace jako například stejné rysy v mapě. V konstantním čase se poté hledá v grafové struktuře algoritmem Dijkstra nejlepší rám, který odpovídá lokálnímu prostředí.

Dalším zajímavým nástrojem pro kombinaci topologických a metrických map je *LaMa* [32], který je založený na konceptu SSH. Uzly představují místa a hrany reprezentují cesty pro navigaci mezi nimi. Uzly i hrany mohou obsahovat další informace jako například metrickou mapu, relativní pozici atd.

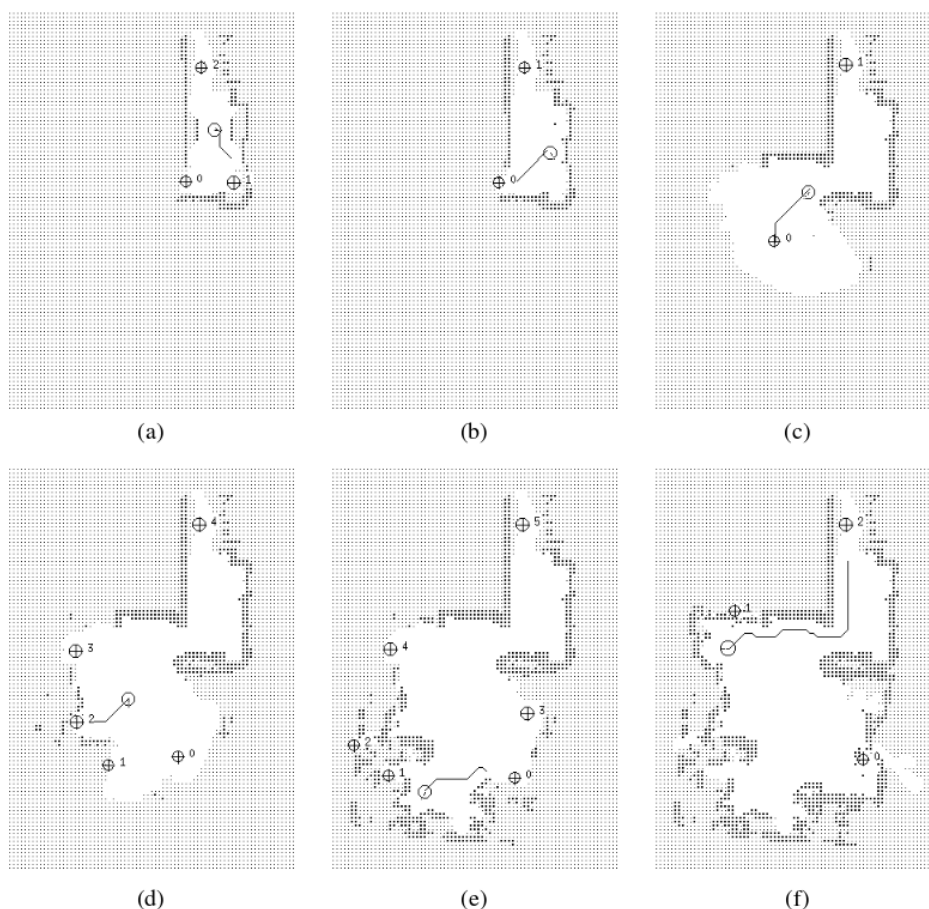
2.2.4 Tvorba mapy

Z hlediska vytváření mapy existují dva přístupy. Buď máme již existující mapu a nebo musí robot postavit svou vlastní mapu prostředí, což se děje během (tele)operované jízdy nebo při autonomním průzkumu. Autonomní průzkum je vytváření nové mapy, zatímco robot projíždí neznámým prostředím.

Příkladem může být zajímavá metoda průzkumu založeném na hranicích [5]. Hranice jsou oblasti mezi otevřeným prostorem a ještě nezmapovaným prostorem. Cílem robota je vydat se za těmito hranicemi, respektive naviguje se za nejbližší neprozkoumanou hranicí podle plánování založeném na DFS (prohledávání do hloubky) v mřížce obsazenosti. Důležitým prvkem v této metodě je také reakční vyhýbání se překážkám, když se robot snaží dosáhnout požadované hranice. Jakmile robot dorazí k hranici, může vidět do neprozkoumaného prostoru a zanechat si tak další informaci do vytvářené mapy. Autoři používají data ze sonaru v kombinaci s daty z laseru, aby minimalizovali chybu vlivem nepřesnosti senzorů, která by mohla významně ovlivnit nejen mapu, ale i další průzkum prostředí. Reprezentace tohoto průzkumu je na obrázku 11.

2.3 Současná lokalizace a mapování - SLAM

Současná lokalizace a mapování (SLAM) je označení problému, kdy je potřeba, aby si robot průběžně vytvářel mapu prostředí a zároveň se podle této mapy lokalizoval. Klíčovým problémem je správná správa neurčitosti pozice robota a mapy, která je založena na senzorických informacích, které bývají zašuměné. Typicky se proto používají filtry či jiné pravděpodobnostní metody, které mají za úkol snížit vliv senzorického šumu hypotézami o skutečné situaci. SLAM může být dle způsobu řešení algoritmy rozdělen do tří oblastí - rozšířené Kalmanovy filtry, částicové filtry a grafové SLAMy. [33]



Obrázek 11: Autonomní průzkum a vytváření mapy [5]

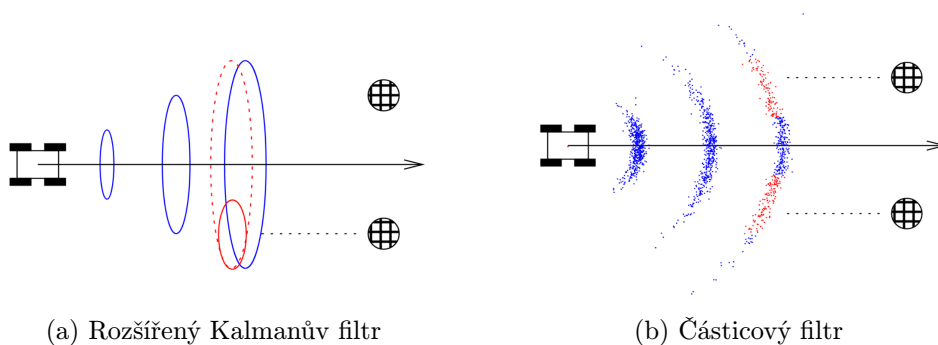
2.3.1 Rozšířený Kalmanův filtr (EKF)

Rozšířený Kalmanův filtr modeluje neurčitost pozice robota a elementů mapy pomocí Gaussovo normálního rozdělení. Protože odhad pozice je nejistý, potom je nejistý i odhad sensorických informací v mapě. Typickým příkladem může být situace, ve které se z předchozí odhadnuté pozice robota v čase a aktuální odometrie určí aktuální pozice robota, která v kombinaci s elementy uloženými v mapě (např. pozice význačných bodů) je použita pro odhad aktuální skutečné pozice. Tyto algoritmy nejsou dobře využitelné, pokud neznáme přesně elementy uložené v mapě. Klíčovým problémem EKF algoritmů je jejich kvadratická složitost, která je omezena, pokud se mapy rozdělí do několika podmap, které jsou zpracovány samostatně. Dalším významným problémem je vytváření jen jedné hypotézy o skutečné pozici, což může vlivem Gaussova rozdělení způsobit podstatnou chybu. EKF SLAM je

často používaný pro SLAM založený na význačných bodech [34], [35].

2.3.2 SLAM založený na grafu s optimalizačními technikami

Tato skupina algoritmů řeší problém nelineární optimalizací. V tomto SLAM systému se vytváří graf, jehož uzly představují pozice robota a elementy mapy. Hrany potom reprezentují pohyb robota či nové měření. Do takového grafu je velmi snadné přidávat další uzly a hrany. Příkladem může být graf, ve kterém uzly představují orientační body, podle kterých se robot naviguje, a pozice robota v čase, které jsou navzájem propojené hranou s hodnotou odpovídající datům z odometrie. Hrana mezi orientačním bodem a pozicí robota v čase vyjadřuje vztah, že v daném čase viděl robot v mapě určitý orientační bod. Hrany představují v podstatě slabé podmínky, které mohou být relaxovány. Postupem času tyto podmínky narůstají, avšak jsou nejhůře lineární, což je hlavní výhodou oproti EKF SLAM algoritmům. Každá spojnice v grafu je poté popsána logaritmickou pravděpodobností, jejíž maximum se hledá pro optimalizaci. Hlavní nevýhodou této skupiny algoritmů je, že při dlouhé trase může být optimalizace velmi náročná. Podobně jako u EKF SLAM algoritmů lze tento problém řešit rozdělením map do více podmap. Příkladem uvedené techniky je metoda zvaná GraphSLAM, která využívá síť grafů pro reprezentaci dat [36].



Obrázek 12: Reprezentace pozice robota pomocí SLAM filtrů [6]

2.3.3 Částicové filtry

Neurčitost pozice robota je v tomto případě modelována množinou částic, kde každá částice odpovídá jedné pozici na trase a jí odpovídající množině elementů mapy. Hustota částic koresponduje s hustotou pravděpodobnosti

pozice robota. Množství odhadnutých stavů částic roste exponenciálně s dimenzí prostoru, tedy klíčovým problémem je příliš velký objem dat v paměti. Pro řešení problému SLAM jsou částicové filtry upraveny pro menší náročnost a označovány jako FastSLAM.

Příkladem může být situace, ve které máme pro každou pozici robota na trase množinu orientačních bodů. Pro každý orientační bod potom máme jejich průměr a rozptyl dle Gaussova rozdělení. Pozice na trase je určována vzorkovacím krokem na základě pravděpodobnostního modelu pohybu robota. Částice je označení pro jednu pozici na trase a jí odpovídající množinu orientačních bodů. S každým následujícím měřením se vypočítává váha na základě normálního rozdělení. Dle této váhy je následně určováno, zda je orientační bod pro danou částici přijatelný či nikoliv. Poté se opět přepočtou průměry a rozptyly Gaussova rozdělení.

Výhodou toho algoritmu je, že je velmi snadný na implementaci, a díky propojeným vztahům jednotlivých částic může pracovat vhodně pro SLAM problémy s nepropojenými daty. Významnou výhodou je modelování více hypotéz zároveň. V [37] používají Rao-Blackwellised částicový filtr při vytváření 3D metrických map orientačních bodů. V [38] používají FastSLAM s využitím mřížky obsazenosti a metody scan-matching pro snížení náročnosti rostoucího počtu částic.

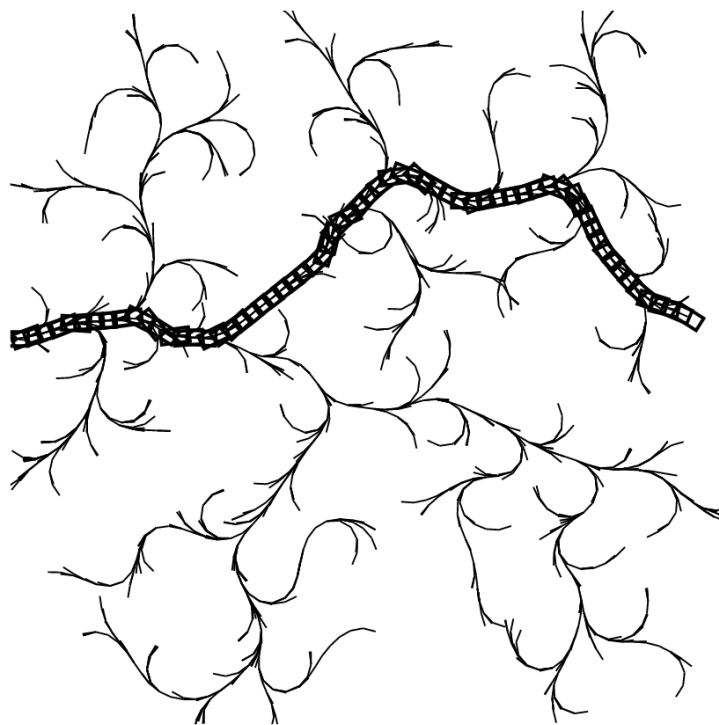
2.4 Plánování pohybu

Pokud už je známá pozice robota a mapa prostředí, pak následuje stanovení způsobu dosažení cíle, kam požadujeme robota dopravit. Úlohou dopravení robota z počáteční pozice do cílové pozice za předpokladu vyhnutí se překážkám a objektům v cestě se zabývá plánování pohybu. Pro plánování pohybu je klíčovým problémem úprava signálů aktuátorů robota pro jeho řízení. Problém plánování pohybu můžeme rozdělit na dvě úlohy - plánování cesty a řízení pohybu [6].

Plánování cesty je požadavek na průjezd robota ze startovní pozice do cílové pozice přes několik známých pozic v mapě. Aby se plánovací algoritmus mohl vyhnout překážkám v prostředí, využívá se nejčastěji v plánování mřížka obsazenosti nebo geometrické mapy [39] [40]. Z plánovacích algoritmů se pak užívá A* nebo Dijkstra vhodné pro topologické mapy [41]. Problémem těchto algoritmů je, že jsou založeny na diskretizaci prostředí a neuvažují omezení pohybu robota jako například nemožnost otočit se na místě. Proto je lepší využívat metody, které uvažují pohyb robota jako například *RRT*.

Rapidní průzkum náhodného stromu (Rapidly-exploring Random Tree (RRT)) je plánovací technika založená na stromové struktuře, jejíž vrcholy představují stav (místo v mapě) a hrany poté reprezentují cesty do stavů (míst), které nejsou obsazeny překážkou [7]. Algoritmus pak vybírá náhodný stav a podle něj nejbližší aktuální vrchol RRT stromové struktury. Pohybem do nového vrcholu se vytvoří nové stavy, které se přidávají do stromu. Ke stavu se také ukládá vstup do aktuátorů, který souvisí s tím, jak je možné se z nejbližšího stavu dostat do nového stavu. Reprezentace tohoto algoritmu je na obrázku 13.

Řízení pohybu se poté zabývá nastavením vstupů do aktuátorů robota tak, aby vykonal požadovaný pohyb po trase a projel všemi pozicemi z plánování cesty až do cílové destinace.



Obrázek 13: Výsledná cesta z RRT metody [7]

2.5 Navigace

Úloha navigace spočívá v dopravení robota ze startovní pozice do cílové pozice za předpokladu vyhnutí se překážkám. Můžeme ji rozdělit do dvou základních oblastí - navigace podle mapy a navigace bez mapy [42].

2.5.1 Navigace bez mapy

Jedná se o reaktivní metody navigace využívající např. optický tok, vizuální odometrii či segmentaci. Segmentaci obrazu využívají v [43], kde pomocí klasifikace rozlišují v obrázku z monokulární kamery, zda je určitý segment průjezdný či neprůjezdný a sestaví si tak obrys cesty, po které se může robot navigovat. Robot je pak v reálném čase schopen následovat cestu a zároveň se vyhýbat překážkám na ní. Navigovat robota bez mapy je možné i sledováním význačných bodů. V této metodě se objevený význačný bod z předchozího obrázku hledá na podobném místě v následujícím příchozím obrázku z kamery na robotovi. Příklad užití je v tomto článku [44].

2.5.2 Navigace na základě mapy

Tuto oblast je možné rozdělit ještě do dvou dalších skupin - navigace podle apriorně známých map a navigace podle map vytvářených v průběhu jízdy robotem. Druhá zmíněná oblast souvisí zejména s již zmíněnými SLAM algoritmy (viz sekce 2.3). Navigace na základě již hotové mapy může být dále rozdělena na navigaci podle metrických, topologických či hybridních map.

Speciální oblastí jsou metody *teach-and-repeat*, které jsou založené na dvou fázích - mapování a poté samotná navigace podle naučené mapy. V první fázi se robot naučí trasu a vytvoří si mapu během (tele)operované jízdy operátorem. V druhé fázi už provádí robot autonomní navigaci podle naučené trasy a vytvořené mapy. Během této autonomní navigace robot přijímá obrázky z kamery umístěné na platformě. V reálném čase z tohoto obrázku extrahuje význačné body, které následně porovnává s význačnými body uloženými v mapě vytvořené při (tele)operované jízdě. Během těchto porovnání se hledají shody mezi význačnými body z aktuálního pohledu z kamery a těmi uloženými v mapě. Díky této metodě je možné například korigovat kumulativní chybu odometrie. Směr robota je pak opravován na základě porovnávání význačných bodů aktuálně viděných kamerou umístěnou na robotovi a význačných bodů uložených v mapě [45]. Tyto metody také mohou souviset se sledováním význačných bodů, jako např. v [46].

Navigace lze dále rozdělit dle prostředí na statickou a dynamickou. Statické a strukturované prostředí se na rozdíl od dynamického nemusí vyrovnávat se změnou prostředí (např. světelných podmínek, změna ročních období či průchod člověka) a má snadno rozpoznatelné přirozené význačné body. V této práci se budeme dále zabývat exteriérem, které spadá do kategorie dynamického a nestrukturovaného prostředí. Používaný systém v této práci

je založen na teach-and-repeat vizuální metodě [45], která je blíže popsána v sekci 4.

2.5.3 Dlouhodobá navigace v dynamickém prostředí

Pokud chceme robota po naučené trase navigovat dlouhodobě, je nutné se vypořádat se změnami prostředí. Takové změny prostředí působí jako další zdroj neurčitosti, který je potřeba zohlednit při integraci informací do mapy. Jedním zdrojem neurčitosti je již zmíněný šum sensorických informací, který se dá omezit dalším zpracováním, jako je například použití filtrů (rozšířený Kalmanův filtr, částicový filtr a další) či použitím více sensorů a provedením sensorické fúze jejich dat, která by měla lépe odpovídat skutečné situaci. Typickým příkladem může být ověření dat ze sonaru pomocí laserového skeneru či korekce kumulativní chyby odometrie pomocí kamery a metod počítačového vidění. Druhým zdrojem neurčitosti jsou změny prostředí, které můžeme rozdělit do tří kategorií podle jejich typického charakteru.

První kategorií jsou změny související s náhodnou dočasnou proměnou okolí (neboli *fluktuace*) jako je například pohybující se člověk či zvíře, projíždějící auto nebo přesunutá popelnice u domu. Kvůli krátkodobosti a náhodnosti těchto procesů se informace o tomto druhu změn do mapy typicky neukládá a naopak je žádoucí je z mapy odebírat. *Variace* jsou druhým typem změn, které se projevují změnou počasí (déšť, mlha apod.), denním světlem (typicky den a noc) nebo změnou ročního období, a je možné očekávat jejich opětovný výskyt po nějaké době. Posledním druhem jsou *trvalé změny* jako je například pokácení stromu, demolice budov za účelem vytvoření parku či výstavba nových domů místo polí, na které je žádoucí se adaptovat a jejichž výskyt nelze predikovat. Z těchto důvodů je vhodné aktualizovat mapu, podle které se robot naviguje a která bere v úvahu všechny tři druhy změn.

Jedním z nejpobulárnějších přístupů je "navigace založená na zkušenosti" [47], kde je ke každému uzlu v topologické mapě, reprezentujícímu dané místo v prostředí, možnost přiřadit více map. Tyto představují různé podoby daného místa v závislosti na okolních podmínkách (den a noc, počasí atd.). Tato metoda byla úspěšně zintegrována do teach-and-repeat systému, který tak získal schopnost dlouhodobě navigovat mobilní roboty [48]. Výše uvedená metoda je schopna si postupně zapamatovat všechny možné podoby daného prostředí a později je využít pro lokalizaci. Během vytváření těchto nových map je nutná asistence operátora.

Aktualizací map se zabývá např. článek [49], ve kterém autoři využívají konceptu krátkodobé a dlouhodobé paměti inspirované lidským mozkem. Vy-

užívají topologickou mapu, ve které je každý uzel asociován s množinou význačných bodů. K jejich detekci používají algoritmus SURF [50]. První vytvořená mapa je pak následně aktualizována přidáváním nových význačných bodů a odebráním již zastaralých význačných bodů, u kterých už dlouhodobě nedochází ke shodě s původním (referenčním) obrázkem. Na základě inspirace paměti člověka rozlišují dva procesy - "recall" (vybavování) a "rehearsal" (nácvik). Během nácviku musí být nové význačné body viděny opakovaně v některém uzlu topologické mapy. Pokud projdou několika stavy, potom se dostanou do dlouhodobé paměti. Naopak význačné body, které se odlišují od aktuálního obrázku zase propadají ve stavech níže, až jsou úplně zapomenuty. Tím je zaručeno, že nekvalitní význačné body budou zapomenuty a naopak ty kvalitní se zase dostanou do dlouhodobé paměti. Autoři ověřili navrženou metodu pomocí dat nasbíraných robotem, který dlouhodobě sledoval určité oblasti (vnitřní prostory) s měnícím se vzhledem (pohybující se nábytek či lidé) ze statických pozic. Na rozdíl od tohoto článku se naše práce zabývá autonomní navigací v exteriérních oblastech s měnícím se vzhledem, což je náročnější kvůli pohybujičím se pozicím v prostředí.

Podobně tomuto způsobu to řeší i [51], kteří vytváří takzvanou "Souhrnnou mapu" ("Summary Map"). Při každém průjezdu ukládají nové získané orientační body z kamery do databáze, ze které následně na základě ohodnocovací funkce vybírají menší množinu robustních a relevantních orientačních bodů. Tato ohodnocovací funkce je založena na počtu průjezdů, při kterých se význačný bod objevil znovu. Z dlouhodobého hlediska tak tato funkce podporuje význačné body, které byly vidět při pár průjezdech a mohly by být vidět znovu při dalších průjezdech, a omezuje ty význačné body, které se objevily jen jednou. Autoři navíc potvrdili silnou korelaci mezi počtem úspěšných detekcí význačných bodů během mapovací fáze a šancí na úspěšné detekce těchto bodů během autonomní navigace. Nové význačné body se přidávají do databáze a při překročení prahu chyby reprojekce [51] se význačné body z databáze zase odebírají.

Jiní autoři [52] používají také koncept dlouhodobé a krátkodobé paměti. Krátkodobou paměť aktualizují pokaždé, když obdrží nová data ze sensorů. Dlouhodobou paměť pak aktualizují po průjezdu danou trasou nebo na konci dne. Podobně jako v článku [51] se snaží vytvářet databázi s omezenou velikostí, ale protože není možné uchovávat všechna naměřená data, tak ukládají náhodné vzorky z podmap jednotlivých průjezdů. Na konci dne pak integrují nové informace z databáze do dlouhodobé paměti na základě pravděpodobnostních a statistických metod.

V [53] používají databázi sensorických dat k predikci, zda elementy prostředí budou opět viděny při dalším průjezdu. Autoři zde vytváří tempo-

rální modely viditelnosti význačných bodů obrazu a na základě opětovného pozorování stejných či podobných míst získávají informace pro dlouhodobé predikce viditelnosti těchto význačných bodů v čase. Autoři ověřili, že tato metoda vylepšuje vizuální lokalizaci v měnícím se prostředí. Velmi podobnou metodu využívají i [54], kteří se soustředí hlavně na odebírání význačných bodů, jejichž výskyt v prostředí již není pravděpodobný. Jejich model prostředí reprezentuje vývoj prostředí a zakládá se na časové funkci přežití a Bayesovské pravděpodobnosti. V [55] prezentují metodu, která je schopná předpovědět, jak se změní vzhled prostředí zahrnující změny ročního období i počasí. Na rozdíl od běžného přístupu porovnávajícího zimní obrázky s letními se zde autoři snaží předpovědět na základě zkušeností z databáze, jak obrázek pořízený v zimě bude vypadat v létě a naopak. V [56] jsou schopni izolovat v čase proměnlivé a neměnné komponenty obrazu. Přehled používaných metod může být nalezen v sekci "Related work" článku [51] nebo ve vyčerpávajícím přehledu [57].

Většina těchto přístupů se soustředí zejména na samotnou lokalizaci, kromě [51], kteří se zabývají i dlouhodobou navigací v proměnlivých podmínkách, což vede typicky na složitější práci s mapami s požadavkem na vysokou efektivitu, spolehlivost a schopnost práce v reálném čase. Moje práce je nejbližší tomuto článku, ale oproti přístupu [51] nevytvářím žádnou podmnožinu orientačních bodů pro konkrétní průjezd daným prostředím a snažím se problém řešit úpravou jedné mapy.

2.6 Zpracování obrazu pro vizuální navigaci

V mobilní robotice se jako jeden ze senzorů často používá kamera, z jejíž dat je pak možné vytvářet mapy, lokalizovat se podle nich (viz 2.1.2) a dále plánovat svůj pohyb. Ve vizuálních metodách navigace navíc hraje klíčovou roli zpracování obrazu, které se používá k odhadu směru robota. V této části budou představeny pojmy používané při zpracování obrazu pro vizuální navigaci a nejčastější metody získávání význačných bodů z obrazu. Z dostupných metod představím pouze nejpoužívanější a nejvíce relevantní k této bakalářské práci. Podrobný přehled používaných metod pro zpracování obrazu může být nalezen v [58].

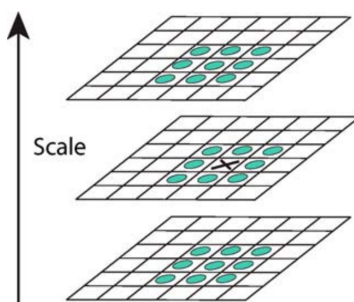
Tyto metody jsou založené na extrakci význačných bodů z obrazu a popisu jejich lokálního okolí. Takto získané rysy obrazu jsou pak schopné do určité míry odolávat různým úhlům pohledu a osvětlení při vytváření shodných dvojic rysů z obrazu kamery a z mapy. Také jsou neměnné vůči škálování obrazu, jeho natočení či různému nastavení kamery.

Extrakce rysů z obrazu se skládá z detekce význačných bodů a z jejich popisu. Význačným bodem je nějaký geometrický objekt (např. to může být roh dveří), který nazýváme klíčovým bodem. Deskriptor pak vytváří popis jeho okolí, který je typicky robustní vůči škálování a obsahuje informace o dané oblasti jako například intenzita gradientu či textura.

Rys obrazu by se měl několik následujících vlastností. Měl by být opakovaně extrahovaný, a tedy robustní vůči změnám úhlů pohledu a osvětlení. Dále by měl být snadno rozlišitelný, aby bylo snazší význačné body párovat. Dalšími důležitými vlastnostmi jsou také výpočetní náročnost jeho získání či kvantita, která by měla dostatečně reprezentovat obrázek.

Scale Invariant Feature Transform (SIFT)

SIFT [8] (v překladu transformace rysů neměnných vůči škálování) je metoda extrakce, ve které detektor hledá klíčové body neměnné vůči škálovatelnosti pomocí sady rozdílů gausiánů. Ve vytvořeném prostoru různých škál se hledá lokální extrém, který odpovídá oblasti připomínající skvrnu. Rysy typu SIFT jsou také neměnné vůči natočení, protože jim je přiřazen nejčastější směr gradientu okolních pixelů. Díky tomu jsou tyto rysy velmi přesné, rozlišitelné, opakovaně extrahovatelné a robustní světelným odchylkám a odchylkám úhlu pohledu, avšak je velmi výpočetně náročné je získat.



Obrázek 14: SIFT: porovnávání pixelů se sousedy z různých škál [8]

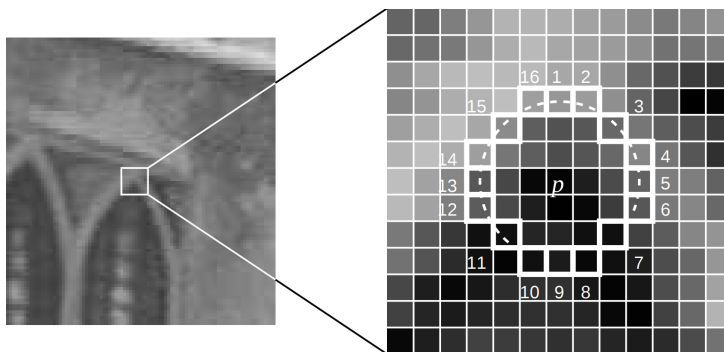
Speeded-Up Robust Features (SURF)

SURF [50] znamená v překladu zrychlené robustní rysy. Detektor této metody je založen na hesiánu hledající rysy, které se liší ve dvou kolmých směrech. Pomocí hesiánu se nachází body na různých škálách, na kterých se

pak pomocí laplacianu gausiánů hledá oblast zájmu. Její struktura připomíná rohy. Tato metoda urychluje proces hledání, který se provádí v metodě SIFT. Tedy hlavní výhodou metody SURF je její rychlost, aniž by však byla na úkor pozitivních vlastností metody SIFT. Tato metoda je nejvíce používaná v navigačním systému používaném v této práci, zejména díky její robustnosti.

Upright SURF

Tato metoda vychází z metody SURF a vynechává při vypočítávání krok, ve kterém se vypočítává natočení. Tedy tato metoda není invariantní vůči natočení. Tato metoda může být výhodnější než SURF například při dlouhodobé lokalizaci v exteriéru.



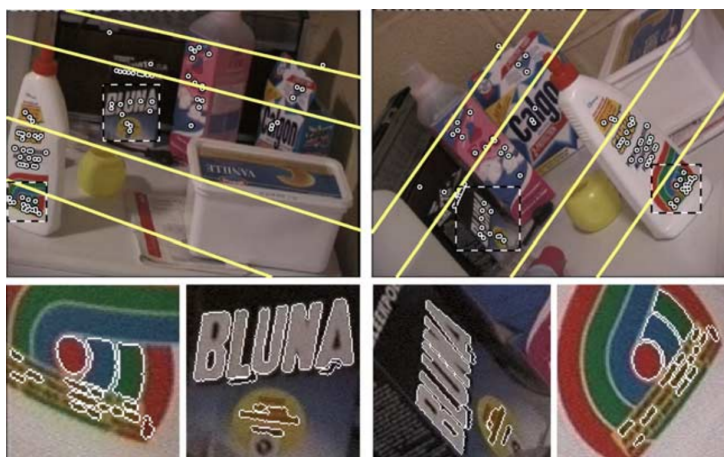
Obrázek 15: FAST detektor: zvýrazněné pixely představují okolí od klíčového bodu označeného písmenem C. Čárkovaná čára zobrazuje dostatečný počet dvanácti pixelů, které jsou světlejší než pixel ve středu kružnice [9].

Features From Accelerated Segment Test (FAST)

FAST [9] (rysy ze zrychleného segmentačního testu) je extraktor, jehož detektor porovnává intenzity pixelů, které leží na kružnici v určité vzdálenosti od pixelu ve středu kružnice (viz obrázek 15). Pokud tato kružnice obsahuje určitý souvislý počet stejně jasných pixelů, potom je střed této kružnice označen za klíčový bod. Intenzita jednotlivých pixelů se pak vyhodnocuje vzhledem k intenzitě pixelu ve středu kružnice. Na základě požadavku souvislého počtu je pak možné snadno zamítnat kandidáty. Díky tomu je tato metoda výpočetně velmi efektivní. Struktura hledaných oblastí též připomíná rohy. Zrychlená verze metody FAST se nazývá AGAST [59], která je adaptivní a generická.

Maximally stable extremal regions (MSER)

Detektor metody MSER [10] (maximálně stabilní extrémní oblasti) hledá stabilní oblasti, které zůstávají neměnné i při proměnlivých podmínkách při transformaci obrazu. Oblasti se hledají na základě segmentace s velkým počtem prahů a slučují se dohromady. Tato metoda je odolná i velkým odchylkám úhlu pohledu nebo afinním transformacím.



Obrázek 16: MSER detektor: hledání stabilních oblastí [10]

Binary Robust Independent Elementary Features (BRIEF)

BRIEF [60] (binární robustní nezávislé prvky) je paměťově velmi efektivní metoda, protože její deskriptory mají formu binárního stringu. Ten je vypočítáván na základě porovnávání intenzit dvojic pixelů z obrázku, který je nejdříve zjemněn.

3 Správa map

Tato kapitola se zabývá představením metod použitých v této práci pro vytváření a údržbu mapy.

Při správě map pro autonomní navigaci se často řeší dilema plasticity. *Plastická mapa* vystavená deformaci vlivem změny okolí se tomuto procesu podřídí a přijme formu danou poslední pozorovanou podobou prostředí, kdežto mapa, která není plastická (nazvěme ji *statickou mapou*), těmto změnám odolává a drží si svůj původní tvar. Odlišnost těchto dvou forem krásně ilustruje příklad modelíny reprezentující plasticitu, která si po deformaci udržuje nový tvar, a tenisového míčku, který se typicky nedeformuje.

Plastická mapa je robustní vůči zdrojům neurčitosti vyvolaným změnami prostředí a umí tak držet krok s vyvíjejícím se okolím, ale nedokáže se vyrovnat se sensorickým šumem. Ten způsobuje, že se v mapě postupně kumulují nepřesnosti měření, které postupem času zhoršují přesnost lokalizace a spolehlivost navigace. Statická mapa je na druhou stranu typicky více robustní vůči sensorickému šumu, ale nedokáže držet krok se změnami prostředí, protože se v čase vůbec nemění. Na základě této kategorizace jsou v této práci uvedeny tři typy map pro autonomní navigaci. Prvním typem je *statická mapa*, která není vůbec plastická a která je vytvořena během (tele)operované jízdy robotem a poté používána při autonomní navigaci. Statická mapa nedrží krok se změnami prostředí a časem tak přestane být relevantní, což má opět negativní dopad na lokalizaci a navigaci. Druhým způsobem navigace je jízda podle *plastické mapy*, která se vytváří během každého průjezdu po naučené trase. Poslední metoda navigace podle *adaptivní mapy* se snaží vytvořit kompromis mezi výše diskutovanými vlastnostmi a zkombinovat tak prvky statické mapy, která se vůbec nemění, a plastické mapy, která se mění celá při každém průjezdu.

Mapa používaná v navigačním systému popsáném v kapitole 4 je hybridní mapa skládající se z množiny lokálních map, které dohromady utváří mapu globální. Lokální mapa obsahuje extrahované význačné body z kamery umístěné na robotovi. V této mapě se ukládají v samostatném poli klíčové body (keypoints) a deskriptory popisující jejich okolí. Jeden klíčový bod má rozměr 7 jednorozměrného pole typu `KeyPoint` a jeden deskriptor má rozměr 64 jednorozměrného pole typu `float`. Sada klíčových bodů se poté ukládá jako jednorozměrné pole a sada deskriptorů jako matice typu `float` s počtem řádků odpovídajícím počtu klíčových bodů a počtem sloupců reprezentující rozměr 64 čísel. Každá tato lokální mapa je pojmenována podle vzdálenosti, při které byla vytvořena. Jedná se tak o lokální metrické mapy, neboť se zde ukládají přesné pozice klíčových bodů a některé další informace o nich jako

například velikost okolí nebo kontrast vůči pozadí. Celkový soubor těchto indexovaných lokálních map poté tvoří globální mapu reprezentující topologickou mapu. Uzly grafu topologické mapy představují jednotlivé lokální metrické mapy a hrany grafu pak reprezentují informace z odometrie.

Hrany topologické mapy jsou asociovány se senzorickou mapou obsahující profil cesty. V této mapě se ukládá ujetá vzdálenost, dopředná a úhlová rychlost hnacích kol pásů robota a doplňkové informace jako například rychlost pásových kol robota určených k jeho zvedání (flippers) tak, aby byl robot schopen jízdy i po schodišti nebo v jiných podobně náročných terénech. Tyto informace jsou ukládány jako jednorozměrné pole typu float.

3.1 Navigace podle statické mapy

Statická mapa je mapa, která se vytvoří při první (tele)operované jízdě robotem. Tato mapa obsahuje pouze výše zmíněné klíčové body, deskriptory a profil cesty. Robot se pak podle této mapy naviguje během všech následujících průjezdů naučenou trasou. Výhoda statické mapy spočívá v její robustnosti vůči neurčitosti senzorické informace, tedy když se robot naviguje podle této mapy a dokáže rozpoznat alespoň jeden význačný bod, potom chyba pozice robota nediverguje. Tato skutečnost je podložena i matematickým důkazem [6], [45]. Dokonce bylo i experimentálně ukázáno, že robot s posunutou startovní pozicí je schopen zkonvergovat na naučenou trasu a nadále po ní jezdit [61]. Významnou nevýhodou této mapy je však její omezenost při měnících se vizuálních podmínkách. Tato mapa nezahrnuje žádné změny prostředí včetně změny světelných podmínek, počasí či ročního období. Kvůli tomuto nedostatku pak není možné robota navigovat v měnícím se prostředí a po určité době je vyžadováno vytvoření nové mapy operátorem, podobně jako v [47]. Tento požadavek pak omezuje schopnost autonomní navigace.

3.2 Navigace podle plastické mapy (Remapping)

Jedná se o mapu podobné struktury jako je statická mapa, ukládá totiž také klíčové body, deskriptory a profil cesty. Nejdříve se vytvoří první mapa při (tele)operované jízdě operátorem. Poté během autonomní navigace robot vytváří další mapu, podle které se naviguje v dalším průjezdu. Jedná se v podstatě o navigaci podle poslední mapy (remapping). Myšlenka této metody se zakládá na schopnosti přizpůsobovat se měnícímu se prostředí. Pokud si robot vytvoří mapu při posledním průjezdu, pak bude spíše schopný

se podle ní navigovat i při zhoršených světelných podmínkách než podle statické mapy, která se může už od aktuální podoby prostředí významně lišit. Možným úskalím této metody je akumulace chyb v důsledku neurčitosti lokalizace, která postupně vede ke zhoršení kvality mapy, a tedy po určité době je nevyhnutelné, že se robot vzdálí od původní trajektorie. Nicméně stále by měl být schopný jezdit podle podobné trasy.

3.3 Adaptivní mapa

Navrhnutý přístup snažící se řešit uvedenou problematiku v této bakalářské práci je takzvaná *adaptivní mapa*. Tato mapa se snaží zaintergovat nové vizuální informace do mapy při zachování kvalitních informací již v mapě uložených. V mapě je potřeba uchovávat jen určité množství význačných bodů kvůli paměťové i výpočetní náročnosti, a tedy není možné si ukládat veškeré nově extrahované význačné body. Je potřeba vybírat, které význačné body uložit a stejně tak které význačné body ze staré mapy odebrat, protože při porovnávání význačných bodů aktuálně viděných kamerou a těch uložených v lokální mapě často dochází k vytvoření chybné dvojice označující shodu význačných bodů. V mapě se obměňuje určitý počet elementů vždy tak, aby celkový počet význačných bodů zůstal stejný jako v původní (první) mapě.

Výběr význačných bodů, které se mají zachovat či odebrat, se provádí na základě ohodnocovací funkce, tedy podobně jako v článku [51]. Na rozdíl od jejich metody ale neukládám každý nově viděný význačný bod do databáze. V mojí navržené metodě se nové význačné body přidávají do mapy na základě nejmenší podobnosti s význačnými body již uloženými v mapě, aby případné přidání nových elementů do mapy ji nekontaminovalo a nekazilo tak schopnost navigace. Je nutné zaručit, aby nově přidaná informace nezhorsila kvalitu mapy. Oproti [51] se úprava mapy provádí okamžitě a v reálném čase, tedy podobně jako v [49], ale pouze s jednou mapou.

V této mapě se k význačným bodům skládajícím se z klíčového bodu a jemu odpovídajícímu deskriptoru ukládá ještě odpovídající hodnocení tohoto význačného bodu. Hodnocení se do mapy pak ukládá jako samostatné pole o velikosti odpovídající počtu uložených klíčových bodů. Adaptivní mapa kombinuje obě předchozí mapy dohromady a snaží se vytvořit takovou mapu, která umí držet krok se změnami prostředí a zároveň je robustní vůči senzorickému šumu.

Abychom mohli takovouto mapu vytvořit, je nutné postupně nejdříve odebrat nekvalitní význačné body na základě ohodnocovací funkce a poté přidávat nové význačné body na základě nejmenší podobnosti s význačnými body

již uloženými v mapě. Počet těchto obměňovaných elementů v mapě pak formuje míru její plasticity. Pokud budeme vyměňovat všechny elementy mapy, pak dostaneme právě plastickou mapu. Naopak pokud nebudeme vyměňovat žádné prvky v mapě, pak dostaneme statickou mapu. V následujících sekcích bude podrobně rozebrán způsob odebírání význačných bodů z mapy a jejich přidávání do mapy.

3.3.1 Odebírání elementů z mapy

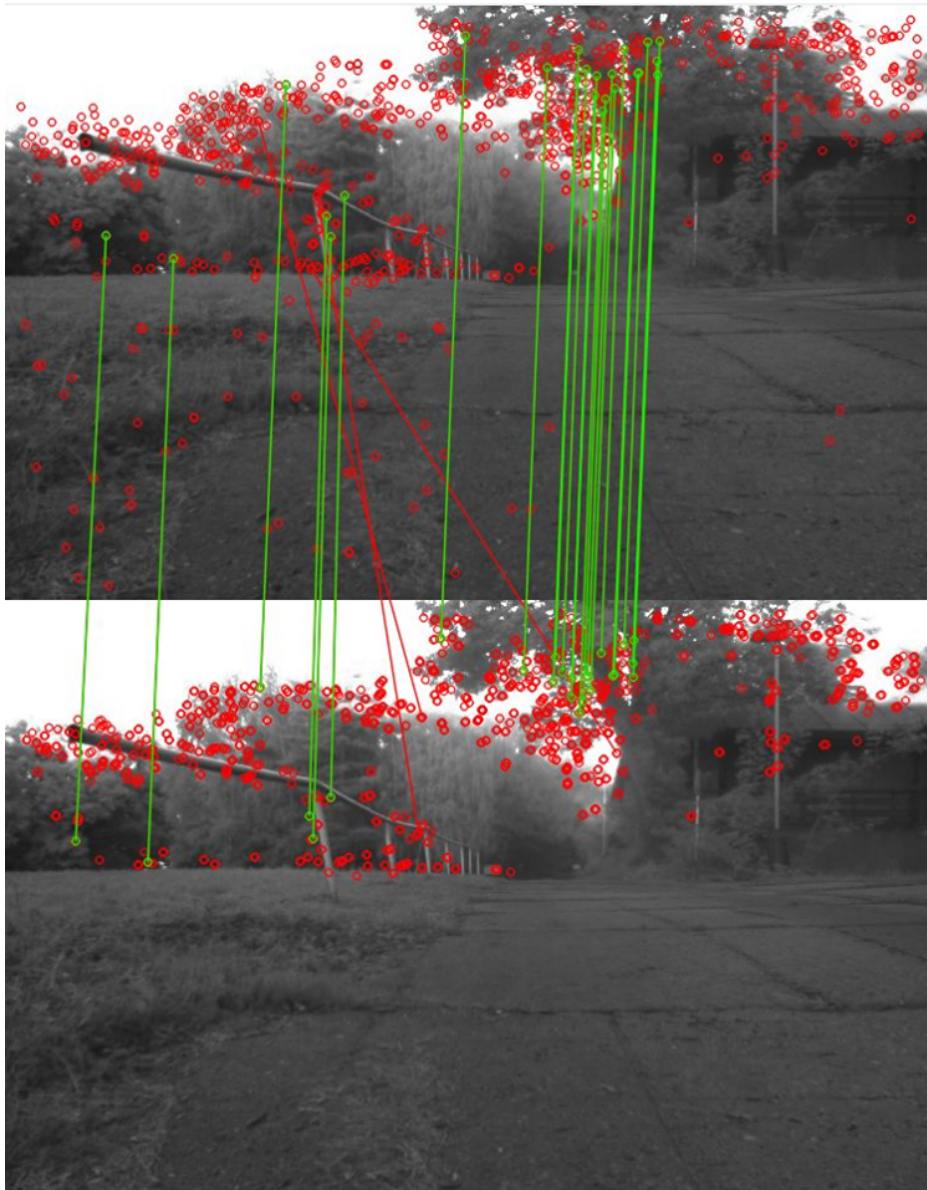
Odebírání význačných bodů z mapy je založeno čistě na výsledném hodnocení jednotlivých elementů v mapě na základě ohodnocovací funkce. Určitý počet těchto prvků s nejhorším hodnocením je z mapy odebrán.

Nejdůležitějším požadavkem je odebírat z mapy zejména ty význačné body, které snižují její kvalitu. Tyto jsou při párování ohodnoceny hodnotou -1 typicky při každém průjezdu, a tedy jejich hodnocení se rychle snižuje a tyto elementy jsou tedy brzo odebrány z mapy. Dlouhodobé hodnocení také chrání před rychlými změnami. Například pokud některý význačný bod již přestal být vidět, a tedy v důsledku toho je jeho hodnocení nerostoucí, tak je nutné, aby jeho hodnocení kleslo až na hranici, při které už bude z mapy odebrán. Zabránění rychlým změnám také omezuje vliv neurčitosti způsobené senzorickým šumem, která by pak následně vedla k akumulaci chyb lokalizace. Navíc pokud byly do mapy chybně přidány nekvalitní význačné body, které se následně budou nesprávně párovat s těmi z aktuálního pohledu kamery, budou pak hned v následujícím průjezdu z mapy odebrány, a tedy tato navržená metoda je odolná i vůči zanášení chybných informací do mapy.

Ohodnocovací funkce

Každý algoritmus, který má za úkol najít shodné význačné body ve dvou obrázcích, vrací nejen správné, ale i nesprávné korespondence těchto dvojic. Je vhodné tuto získanou množinu detekovaných dvojic dále zpracovat a rozdělit ji na správné shodné dvojice a špatné. Na základě histogramové metody popsané v sekci 4.3 se určí převažující směr spojnic detekovaných dvojic, které jsou následně označeny jako správné shodné dvojice (viz 17). Zbylé dvojice odchylující se od převažujícího směru reprezentují špatné shodné dvojice. Při ohodnocování se následně postupně zlepšuje hodnocení těch význačných bodů v mapě, které vytvořily správné shodné dvojice. Naopak špatně vytvořeným shodným dvojicím se toto hodnocení postupně zhoršuje. Nespárovaným význačným bodům v mapě, kterým algoritmus nepřihradil žádnou

3.3 Adaptivní mapa



Obrázek 17: Snímek obrazovky z navigačního systému. Horní obrázek pochází z mapy a dolní z aktuálního průjezdu. Zelené spojnice znázorňují správné shodné dvojice a červené nesprávné.

odpovídající dvojici z extrahovaných význačných bodů obrázku kamery, se hodnocení nemění.

Tím je zaručeno sledování dlouhodobého vývoje význačných bodů uložených v mapě, možnost uchovávání robustních elementů, které jsou vidět při

většinu autonomních průjezdů, a odebrání nekvalitních význačných bodů, které buď mapu kazí nebo dlouhodobé navigaci nenapomáhají.

3.3.2 Přidávání prvků do mapy

Jak již bylo zmíněno výše, není možné přidávat do mapy všechny nově detekované význačné body. Navíc je nutné z těchto nových význačných bodů vybírat ty, které nezhorší kvalitu mapy. Může se totiž snadno stát, že nový význačný bod bude velmi podobný již uloženému význačnému bodu v mapě, a tak může dojít k jejich nesprávnému párování. Když se potom na tuto množinu aplikuje algoritmus, který hledá shody význačných bodů, může dojít k vytvoření špatné shodné dvojice díky nově přidanému elementu do mapy, což zhorší hodnocení původního dobrého význačného bodu, kvalitu celé mapy a robustnost výsledné navigace. Na základě těchto úvah se snažím vybrat z nových význačných bodů extrahovaných z aktuálního obrazu kamery ty, které jsou nejméně podobné již uloženým prvkům v mapě - hledají se jejich maximální nejkratší vzdálenosti od význačných bodů uložených v mapě.

Maximální nejkratší vzdálenost

Pro vytvoření hodnocení nově přidaných bodů je nutné každý význačný bod ohodnotit vzdáleností k význačným bodům uloženým v mapě. Tyto vzdálenosti můžeme získat z algoritmu, který hledá odpovídající shodné dvojice extrahovaných význačných bodů z pohledu kamery a těch uložených v mapě.

Vytváření shodných dvojic je v tomto navigačním systému založeno na algoritmu hledání k nejbližších sousedů (kNN) z extrahovaných význačných bodů aktuálně příchodního pohledu kamery k danému korespondujícímu význačnému bodu v mapě. Pro každého tohoto souseda je následně nalezen nejbližší odpovídající význačný bod v mapě. Pokud se některý aktuálně detekovaný význačný bod nespároval s žádným elementem mapy, pak zůstává jeho vzdálenost dostatečně velká. Vzdálenost jednotlivých význačných bodů reprezentuje jejich vzájemnou podobnost číslem z intervalu $(0, 1)$. Nakonec se z výsledné množiny vybírá určitý počet význačných bodů z aktuálního pohledu kamery, které mají tuto nejkratší vzdálenost nejdelší, a jsou tedy nejméně podobné všem již uloženým prvkům v mapě. Po přidání těchto nových význačných bodů do mapy je jejich ohodnocení nastaveno na hodnotu 0.

4 Popis navigačního systému

Navigační systém použitý v této práci se nazývá BearNav a je detailněji popsán v článku [61], který navazuje na předchozí systém popsáný v [45], [6]. Tato kapitola se soustředí na stručný popis této navigační metody, její implementaci a rozšíření, které je předmětem této bakalářské práce.

BearNav je navigační metoda založená na technice *teach-and-repeat*, při které se robot nepotřebuje explicitně lokalizovat a je schopen se navigovat v reálném čase. Hlavním cílem popsané metody je požadavek na robustní metodu, která je jednoduchá na implementaci a výpočetně nezávislá na velikosti nebo typu prostředí. Tato metoda nevyžaduje kalibraci senzorů a je možné ji použít i na jednoduchých robotech s levnými senzory. Díky těmto vlastnostem je potom robot schopný navigovat se i podle malého množství přirozených orientačních bodů ve vnitřních i venkovních prostorech.

V základní konfiguraci tento systém využívá pouze odometrii a monokulární kameru pro počítačové vidění. Robot na základě odometrie měří svou ujetou vzdálenost v prostoru, zatímco pomocí metod počítačového vidění je řízené otáčení robota. Díky tomu je možné korigovat kumulativní chybu odometrie, a tedy chyba pozice v tomto navigačním systému nediverguje [6], [45], [61]. Systém je snadno rozšířitelný dalšími senzory (např. kompas, gyroskop, sonar), ale pro jeho dostatečně spolehlivou funkci stačí použití pouze odometrie a kamery.

Metoda *teach-and-repeat* se někdy označuje také jako *map-and-replay* a reprezentuje proces naučení se nějakému chování a následně schopnost jeho zopakování. Můžeme si tento proces snadno představit na příkladu dítěte, které se učí házet míčem do basketbalového koše. Nejdříve rodič hází míčem spolu s dítětem, tedy vlastně vede jeho pohyby a dítě se jim tak může naučit spolu s potřebným vynaložením intenzity sil. Poté, co se takto naučí házet míčem, je schopné se v tom dále zdokonalovat. Analogicky to funguje i v robotice, například při svařování. Operátor vede robota po nějaké cestě (či vykonává nějaký pohyb) a robot si zaznamenává potřebné sensorické informace. Poté ve fázi přehrávání je schopný vykonat velmi podobný, nikoliv zcela totožný, pohyb. Navíc může být schopen tento pohyb přizpůsobit nové sensorické informaci na základě podnětů z okolí.

V tomto navigačním systému si robot nejdříve vytváří mapu během fáze učení, při které je operátorem řízen prostřednictvím (tele)operace. Poté robot používá vytvořenou mapu pro autonomní navigaci ve fázi opakování. Protože prostředí se v důsledku přirozených procesů mění, může si robot mapu v průběhu autonomní navigace upravovat. Způsob této úpravy je hlavním

předmětem této práce.

4.1 Zpracování obrazu

Pro extrahování význačných bodů je využita převážně metoda SURF [50] pro detekci orientačních bodů i formování deskriptorů popisujících jejich okolí díky její robustnosti vůči větším změnám úhlu pohledu. Tento systém však umožňuje i použití algoritmů AGAST [59] nebo upright-SURF pro detekci a algoritmu BRIEF [60] pro vytvoření deskriptorů, které jsou výpočetně méně náročné a robustnější vůči změnám prostředí. Detailnější popis těchto metod extrakce byl diskutován v sekci 2.6. Pro získávání kvalitního obrazu z kamery je nutné správně řídit expozici, v našem případě metodou prezentovanou v článku [62], která navrhuje, že je lepší se navigovat podle obrázků spíše podexponovaných než přeexponovaných.

4.2 Fáze učení - mapování

Během této fáze je robot řízen operátorem a vytváří si mapu prostředí. V průběhu (tele)operované jízdy robot měří ujetou vzdálenost a zaznamenává si změny dopředné i úhlové rychlosti vzhledem k aktuální vzdálenosti. Tuto sekvenci nazýváme *profil cesty*. Robot také současně extrahuje význačné body z příchozích obrázků z kamery a každých 0.2 m ujeté vzdálenosti si tyto extrahované význačné body ukládá do *lokální mapy*, která je pojmenována podle aktuální vzdálenosti na trase (viz 18). Jde v podstatě o vytváření hybridní mapy. Model prostředí je reprezentován grafem topologické mapy, ve které je každý uzel asociován s metrickou mapou význačných bodů a hrany odpovídají sensorické mapě zaznamenaných rychlostí robota. Jednotlivé uzly pak představují lokální mapy, jejichž poskládání dohromady tvoří jednu globální mapu průjezdu po požadované trase a jejichž přepínání je řízeno pomocí ujeté vzdálenosti změřené na základě odometrie.

4.3 Autonomní navigace

Na začátku této fáze je robot umístěn blízko startovní pozice, vytvoří si seznam lokálních map s extrahovanými význačnými body a nahraje si uložený profil cesty, aby nastavil požadovanou rychlost a dále ji podle tohoto profilu měnil. Následně si robot nahraje první lokální mapu s indexem vzdálenosti startovní pozice. Zatímco se robot pohybuje, tak mu přichází význačné body

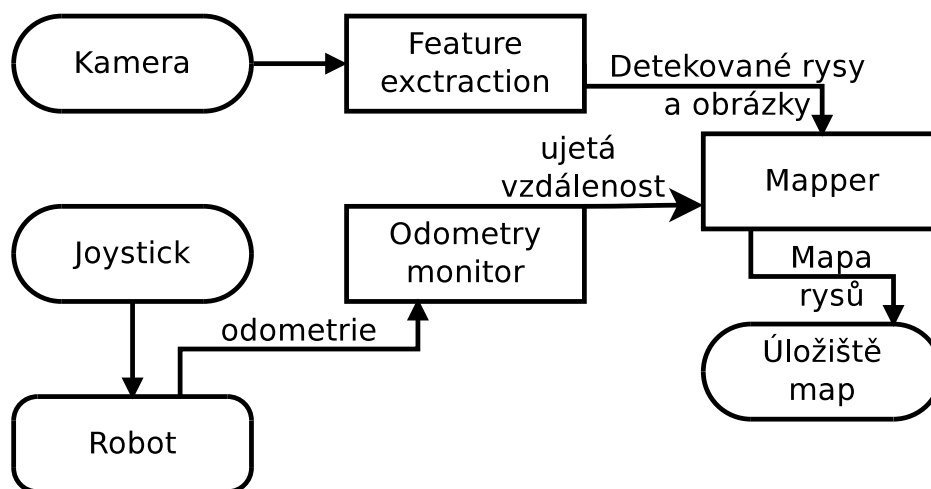


Obrázek 18: Hybridní mapa v používaném navigačním systému

extrahované z aktuálního pohledu z kamery umístěné na platformě. Tyto význačné body porovnává s těmi uloženými v aktuální nahrané lokální mapě. Podle rozdílu horizontálních pozic těchto význačných bodů se vytváří histogram, jehož maximum indikuje nejčastější vzdálenost těchto význačných bodů, což odpovídá posunu aktuálně příchozího obrázku vůči obrázku z mapovací fáze. Tento rozdíl koresponduje s odchylkou robota od naučené trasy, a proto je poté přepočítán na úhlovou rychlost, která je připočtena k uloženému profilu cesty. Tímto se opravuje nepřesnost sensorických měření a zejména kumulativní chyba odometrie. Pokud maximum v histogramu nelze jednoznačně určit například vlivem málo extrahovaných význačných bodů či špatným nastavením expozice a jasu kamery, potom se tato nekvalitní vizuální informace převedená na rychlost otáčení robota nepřidává k profilu cesty a robot pokračuje podle původních uložených hodnot. Během jízdy robot porovnává aktuální obrázky vždy s nejbližší uloženou lokální mapou a s delší ujetou vzdáleností se přepíná na další lokální mapu.

4.4 Implementace systému

Navigační systém je implementovaný v *Robotickém operačním systému* (ROS) ve verzi Kinetic a je schematicky znázorněn na obrázku 19 a 20.

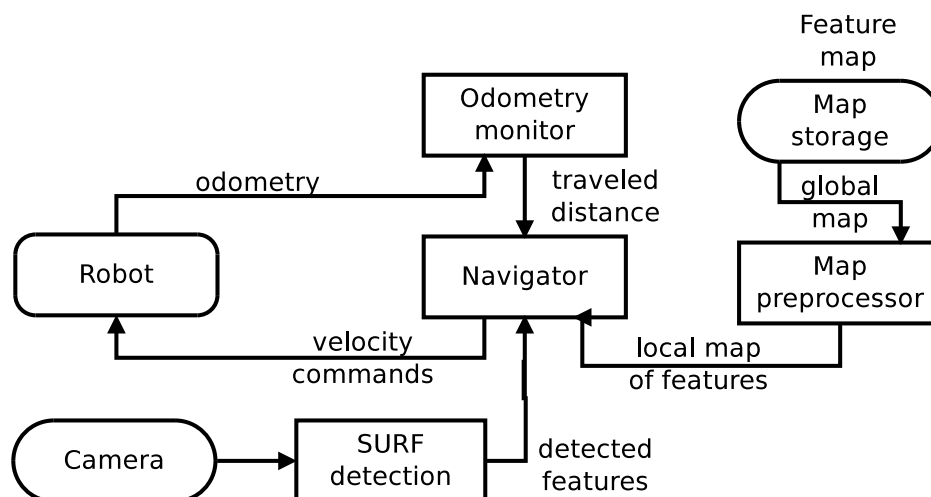


Obrázek 19: Zjednodušené schéma při mapování

Nejdříve jsou v uzlu *Feature extraction* extrahovány význačné body z přichozích obrázků z kamery na robotovi. Tyto extrahované význačné body jsou potom skrz komunikační kanál (topic) publikovány do uzlů s názvem *Mapper* a *Navigator*. Uzel *Odometry monitor* dostává data z odometrie robota a měří ujetou vzdálenost, kterou posílá dalším uzlům. *Mapper* ukládá příchozí extrahované význačné body do lokální mapy podle ujeté vzdálenosti, kterou dostává z uzlu *Odometry monitor*. Toto ukládání se děje jen po uplynutí určité vzdálenosti (0.2 m), což je oznámeno speciální zprávou publikovanou z uzlu *Odometry monitor*. *Mapper* také ukládá *profil cesty* se změnou dopředných a úhlových rychlostí robota měněných pomocí joysticku ovládaného operátorem. Při autonomní jízdě nejdříve uzel *Map preprocessor* nahraje všechny vytvořené uložené lokální mapy a profil cesty a posílá je uzlu *Navigator* podle ujeté vzdálenosti přicházející z uzlu *Odometry monitor*. V uzlu *Navigator* jsou tedy k dispozici extrahované význačné body z aktuálního obrazu kamery přicházející z uzlu *Feature extraction* a také význačné body uložené v lokální mapě z uzlu *Map preprocessor*. V uzlu *Navigator* pak probíhá porovnávání těchto význačných bodů a výpočet jejich vzájemného posunutí. Podle histogramové metody (viz 4.3) se nejčastější rozdíl tohoto posunutí přepočítaný na rychlost připočte k rychlosti otáčení robota z příchozího profilu cesty a odešle se příkaz robotovi s informacemi o požadované dopředné a úhlové rychlosti.

Všechny moduly jsou implementovány v jazyku C++ a zároveň jsou akčními servery v systému ROS. Jejich parametry je možné dynamicky měnit skrz dynamický rekonfigurátor. Operátor také může snadno pozorovat aktuální stav systému díky informacím ze zpětné vazby těchto akčních serverů.

V závislosti na dané fázi se používají různé moduly s rozdílnou interakcí. Zároveň je velmi snadné systém rozšířit o další moduly či vynechat použití některých modulů pro jiné zamýšlené chování robota.



Obrázek 20: Zjednodušené schéma při navigování

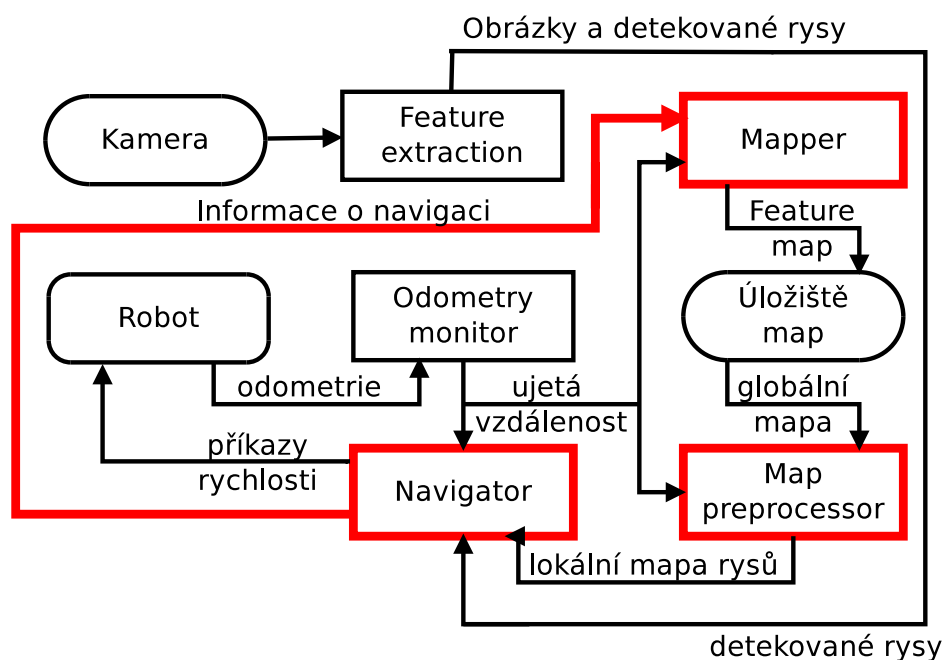
4.5 Rozšíření systému

Výše popsaný stav systému neumožňuje vytvářet mapu během autonomní navigace. Aby to bylo možné, je potřeba vyřešit nezávislost mapování na joysticku (a tedy i na operátorovi) a zároveň uvažovat posunutí obrázků z histogramové metody 4.3. Pro adaptivní mapu je potřeba ohodnocovat jednotlivé význačné body v mapě metodou popsanou v 3.3. Nejvhodnější místo pro ohodnocovací funkci je přímo v uzlu *Navigator*, ve kterém se provádí porovnávání význačných bodů extrahovaných z obrázku a význačných bodů uložených v mapě.

Nejdříve jsem rozšířila zprávu *Feature* o informaci o hodnocení daného význačného bodu. V uzlu *Navigator* jsem potom implementovala ohodnocování význačných bodů extrahovaných z obrázku kamery na základě vzdálenosti popsané v 3.3.2. Do tohoto uzlu bylo také potřeba přidat ohodnocování význačných bodů ze správných a špatných shodných dvojic. V systému bylo již implementováno vybírání správných shodných dvojic ze všech dvojic utvořených algoritmem k-nn pro jejich vykreslování do obrazu, tedy bylo velmi snadné rozlišit špatné shodné dvojice pro ohodnocování. Vybírání správných shodných dvojic je provedeno na základě histogramové metody 4.3. Pro vytvořené shodné dvojice jsem poté implementovala přičítání jejich aktuálních

hodnot k jejich celkovému hodnocení, které si tyto význačné body s sebou nesou v příchozí zprávě z uzlu *Map preprocessor*. Význačné body v mapě jsou totiž hodnoceny podle toho, jakou utvořily dvojici s význačnými body extrahovanými z obrázků kamery (3.3.1). Pro význačné body z mapy, které se nespárovaly s žádnými význačnými body extrahovanými z aktuálního pohledu kamery, se hodnocení nemění. Dále jsem zde implementovala přidávání určitého počtu aktuálně viděných a právě ohodnocených význačných bodů. Do mapy se přidávají význačné body, které jsou nejméně podobné těm již uloženým v mapě, jak je více vysvětleno v 3.3.1. Následně jsem implementovala odebrání stejného počtu význačných bodů s nejmenším hodnocením.

V uzlu *Navigator* se pak formuje nově vytvořená zpráva *Navigation info*, kterou pak poslouchá skrz komunikační kanál uzel *Mapper*. Tato zpráva obsahuje význačné body z aktuálního obrazu kamery a význačné body z odpovídající obdržené lokální mapy, vypočítané posunutí mezi nimi, informaci, která indikuje, zda byla mapa upravena, a některé další informace vhodné zejména pro statistické vyhodnocování.



Obrázek 21: Schéma rozšířeného systému

Poté jsem upravila ještě uzel *Mapper* tak, aby bylo možné ukládat hodnocení do mapových souborů. Aktuálně umí tento uzel přijímat extrahované význačné body přímo z uzlu *Feature extraction*, což je využito pouze v mapovací fázi navigační metody *teach-and-repeat*, ale také význačné body z

mapy, která přichází z uzlu *Navigator*. Tato mapa se může měnit adaptivně podle výše uvedeného způsobu, ve kterém se obměňuje určitý počet elementů mapy, a tedy formuje se zde *adaptivní mapa*. Pak je možné zde vytvářet *statickou mapu* tak, že se nemění vůbec žádné elementy v mapě (nevyměňuje se žádný počet význačných bodů v mapě). *Plastická mapa* se pak vytváří tak, že se obměňují všechny elementy v mapě bez ohledu na jejich hodnocení a upravuje se pouze jejich posunutí podle histogramové metody 4.3 tak, aby nedocházelo k velké chybě pozice robota. Pro správnou funkci uzlu *Mapper* jsem také musela upravit stavový automat zajišťující již implementovanou vnitřní logiku uzlu.

Posledním uzlem, který ještě potřeboval upravit, byl *Map preprocessor*. V tomto uzlu jsem upravila načítání map ze souborů tak, aby se načítalo i jednotlivé hodnocení význačných bodů a ukládalo se do struktury potřebné pro posílání zpráv po komunikačním kanále. Aby byl systém kompatibilní i s mapami, které neukládají žádné hodnocení o uložených význačných bodech, tak se těmto při načítání přiřadí hodnota 0.

5 Datasets

Naměřené datasety jsou vytvořeny v rámci projektu GAČR 17-27006Y, jehož součástí je ověření schopnosti robota jezdit po celý den a udržovat tak krok s měnícími se venkovními podmínkami. Získané datasety obsahují senzorické údaje robota obdržené při několika desítkách autonomních průjezdů různých tras. První trasa a datasety s ní spojené vznikly během několika dní až týdnů v létě 2017, kdy se vytvářela pouze statická mapa a byly sbírány data z autonomních průjezdů. Druhá a třetí trasa byla vytvořena pak na jaře 2018, kdy byl už robot navigován podle plastické a adaptivní mapy. Všechny tři datasety obsahují různé světelné i povětrnostní podmínky.

Všechny tyto experimenty byly provedeny na vrchu Hostibejk v Kralupech nad Vltavou. Na tomto vrchu je budova, ke které vede příjezdová cesta s parkovištěm. V okolí je zde pak malá louka a les obklopující budovu a cestu ze všech stran. Při navigaci se zde vyskytují objekty jak ve velké, tak i v blízké vzdálenosti. Toto prostředí je tedy dostatečně dynamické a vhodné pro ověření našeho navigačního systému.

5.1 První dataset - navigace podle statické mapy

V létě 2017 se robot navigoval podle trasy dlouhé cca 60 metrů, na které vykonával různé prudké zatáčky. Tento dataset byl pořízen za různých světelných podmínek. Na trase se také občas vyskytují náhodně procházející lidé představující *fluktuace*.

První den jsme vytvořili při prvním průjezdu statickou denní mapu, podle které se robot následně navigoval při všech dalších průjezdech. Při několika dalších průjezdech, při kterých jsme sbírali potřebná data, se robot držel naučené trasy bez výrazné chyby pozice. Navíc pokud jsme robota umístili dál od startovní pozice (např. 0,5 metru), byl potom schopný po několika dalších průjezdech zkonvergovat zpět na naučenou trasu.

Nicméně jak již bylo zmíněno v sekci 3.1, tato mapa neumí držet krok s měnícími se podmínkami. Tedy robot nebyl schopen se po setmění navigovat po naučené trase. Rozhodli jsme se proto vytvořit další mapu, podobně jako v [47], pro noční prostředí (tele)operovanou jízdou robotem po velmi podobné (nikoliv však zcela stejné) trase. Pro vytvoření mapy je potřeba přítomnost operátora, který narušuje koncept dlouhodobé autonomní navigace v proměnlivých podmínkách. Z této nevýhody tedy vyplývá potřeba vytváření mapy, která umí držet krok se změnami prostředí.

5.2 Druhý dataset - navigace podle statické, plastické a adaptivní mapy

Tento dataset se v dalším vyhodnocování nevyužívá. Slouží pouze k ověření, že podle statické mapy se nelze navigovat v měnících se podmínkách, a uvádí nás tak do problematiky, v níž je potřeba řešit, jak vytvářet mapu pro dlouhodobou navigaci v proměnlivém prostředí.



Obrázek 22: Jednotlivé pohledy z průjezdu první trasou

5.2 Druhý dataset - navigace podle statické, plastické a adaptivní mapy

Na jaře 2018 bylo nutné navigovat se podle trochu jiné cesty z následujících důvodů. Zaprvé, zjistili jsme, že mapa z roku 2017 je příliš dlouhá, a tedy ke změnám světelných podmínek dochází rychleji, než se robot stihne vrátit zpět na startovní pozici. Zadruhé, abychom mohli měřit chybu navigace pomocí externího kamerového zařízení, je nutné, aby se celá trasa vešla do záběru kamery externího lokalizačního systému. Vzhledem k uspořádání prostředí a dostupným prostředkům jsme omezili trasu robota na cca 16 metrů. Použitý externí lokalizační systém whycon [11] zde slouží k ověření hypotézy o robustnosti plastické, statické a adaptivní mapy vůči změnám prostředí a senzorickému šumu.

Cílem těchto experimentů bylo ukázat, že se robot zvládne navigovat i při stmívání, od úplného světla až po úplnou tmu, kdy je operační prostor slabě osvětlen lampami a robot musí být vybaven vlastním reflektorem, podobně jako automobil jedoucí v noci a osvětluje si cestu dálkovými světly. Navíc se v prostředí projevovaly změny vlivem *fluktuace*, podobně jako v 5.1. Na začátku každého testovacího dne jsme vytvořili novou mapu, aby výsledky úspěšnosti jednotlivých metod byly navzájem snadno porovnatelné.

5.3 Třetí dataset - navigace podle adaptivní mapy



Obrázek 23: Jednotlivé pohledy z průjezdu druhou trasou

5.3 Třetí dataset - navigace podle adaptivní mapy

O dva týdny později, než byl vytvořen druhý dataset, jsme zjistili, že na keřích, které byly při navigaci v blízké vzdálenosti, vyrostly květy, a tedy pro robota se toto místo výrazně změnilo. Rozhodli jsme se proto změnit trasu na lepší operační prostor, ve kterém nebylo příliš mnoho takto proměnlivých objektů v bezprostřední blízkosti před robotem.

Dataset získaný z tohoto experimentu se snaží nejlépe zachytit proces stmívání a ověřit při tom schopnost a spolehlivost navigace podle adaptivní mapy. Tento experiment byl zahájen zhruba půl hodiny před stmíváním a probíhal až do úplného vybití robota a notebooku s navigačním softwarem, což nastalo zhruba půl hodiny po úplném setmění, kdy bylo již nutné, aby robot používal vlastní reflektor.

5.3 Třetí dataset - navigace podle adaptivní mapy



Obrázek 24: Pohled na místo s třetí trasou

6 Experimenty

V této kapitole je vyhodnocena schopnost a kvalita navigace a chyba lokalizace vznikající vlivem neurčitosti sensorických informací v závislosti na měnících se venkovních podmínkách. Během dlouhodobé navigace se mohou postupně kumulovat chyby, které jsou způsobeny neurčitostí dat ze sensorů a změnami prostředí. V tomto navigačním systému se vlivem změny prostředí projevuje neurčitost sensorické informace nepřesnostmi vzniklými při párování význačných bodů z map spolu s chybou odometrie. To následně vede k neurčitosti lokalizace i celé navigace. Všechny tyto uvedené důsledky souvisí s kvalitou mapy.

Aby bylo možné ohodnotit kvalitu navigace podle daného typu mapy a porovnat tyto metody navzájem, je nutno stanovit následující kritéria. Nejdříve vyhodnotím *relevanci* jednotlivých map vzhledem k navigaci v závislosti na počtu správně vytvořených shodných dvojic. Dále porovnáám jejich *přesnost* podle poměru správně vytvořených shodných dvojic vůči celkovému počtu vytvořených dvojic. *Chyba lokalizace* podle význačných bodů bude vyhodnocena porovnáním rozdílu posunutí obrázků vypočítaného z histogramové metody 4.3 a určené člověkem. Posledním kritériem je *přesnost navigace*, které sleduje chybu pozice robota pomocí externího lokalizačního systému. První dvě kritéria souvisí hlavně se změnou prostředí, kdežto druhá dvě kritéria vypovídají o pozici robota. Porovnání jednotlivých metod však nemusí být ze získaných výsledků patrné na první pohled, proto následně ještě ověříme výsledky statistickým testováním.

6.1 Vytvoření dat pro analýzu

Pro dosažení opakovatelnosti vyhodnocení jednotlivých kritérií se budou při analýze datasetu používat pouze obrázky se stejnou vzdáleností z odometrie, tedy obrázky z jednotlivých map při jízdě robotem. V tomto navigačním systému je však komplikované vytvořit více druhů map ve stejný čas a uložit si statistiky navigace podle nich při následujících opakovaných průjezdech. Proto je nutné nasimulovat mapování podle obrázků uložených v adaptivní mapě, která se vytvářela při každém průjezdu naučenou trasou při zhoršujících se světelných podmínkách. Podle statické mapy vytvořené při (tele)operované jízdě a z obrázků z jednotlivých adaptivních map jsem vytvořila plastickou mapu a adaptivní mapu s různým nastavením obměňovaných elementů v mapě (pro plastickou mapu se obměňují všechny význačné body v mapě, pro adaptivní pouze určitý počet (5-10%), jak je uvedeno v sekci 3.3).

Následně je možné podle takto utvořených map odsimulovat navigaci, ve které místo obrázků z kamery budou přicházet obrázky z adaptivní mapy vytvářené při jednotlivých průjezdech. Tak je možné získat jednotlivé statistiky navigace a data pro další zpracování a vyhodnocení. Tyto statistiky obsahují počet vytvořených shodných dvojic a kolik z nich bylo správně a nesprávně vytvořených, vzdálenost, při jaké byly obrázky zachyceny kamerou, posunutí hodnocených obrázků vůči sobě na základě histogramové metody a podle referenčních hodnot získaných od člověka.

6.2 Statistické testování dat

Pro testování je zde použitý studentův t-test, který je schopen určit, zda se výsledky měření při jedné mapě významně liší od výsledků měření při druhé mapě. T-test vyžaduje, aby testovací vzorek byl z normálního rozdělení. Pokud je náhodná veličina neznámého rozdělení, potom je možné studentův t-test použít dle centrální limitní věty pouze za předpokladu, že testovací vzorek je dostatečně velký. V následujících testováních má testovací vzorek vždy velikost okolo 1000 testovaných hodnot. Podle centrální limitní věty se totiž s narůstající velikostí testovaného vzorku blíží toto rozdělení normálnímu.

V použitém párovém t-testu náhodný výběr představuje dvojice hodnot z jednotlivých průjezdů podle daných typů map. Tyto hodnoty odpovídají testovaným kritériím. Párový t-test pak v tomto případě následně testuje, zda rozdíl středních hodnot rozdělení z těchto dvou skupin dat je roven nule.

Párový t-test lze snadno převést na jednovýběrový t-test. Pokud označíme první skupinu dat jako a_i a druhou skupinu dat jako b_i , potom můžeme vytvořit náhodnou veličinu $x_i = a_i - b_i$, jejíž střední hodnotu označíme jako $\mu = 0$. Testovou statistiku pak vypočítáme jako

$$T = \frac{\bar{X} - \mu_0}{S} \sqrt{n},$$

kde \bar{X} je výběrový průměr náhodné veličiny X , S^2 je její výběrový rozptyl, μ_0 je střední hodnota normálního rozdělení a n je velikost testovaného vzorku. Nulová hypotéza, že se střední hodnota normálního rozdělení této náhodné veličiny rovná nule, se zamítá, pokud se její výběrový průměr příliš liší od 0. Pak jsou požadované průběhy považovány za podobné. Všechny testy byly provedeny na hladině významnosti 5%.

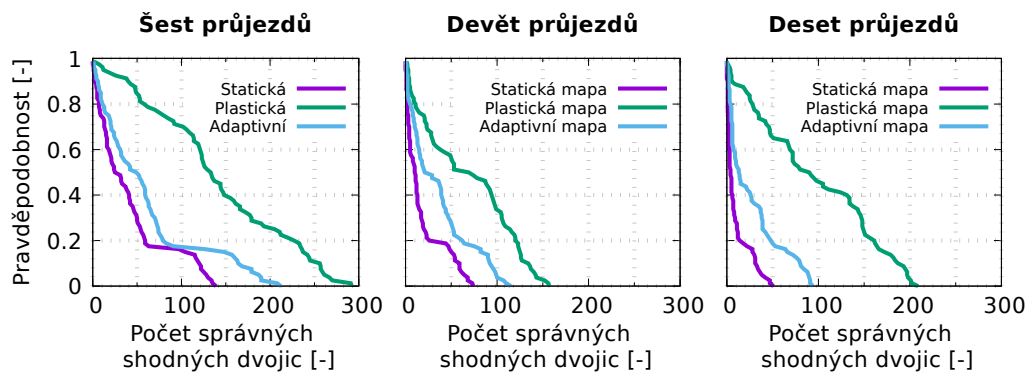
6.3 Analýza a testování kritérií

V této sekci postupně vyhodnotím výsledky experimentů v závislosti na uvedených kritériích. Grafy jsou vykresleny vždy pro šestý, devátý a desátý průběh, neboť ukazují nejzajímavější změny z celkového počtu deseti průjezdů. Mezi devátým a desátým průjezdem se výrazně setmělo. Nejdříve analyzuji jednotlivé průběhy v grafech, poté je statisticky ověřím párovým t-testem a na závěr shrnu a porovnam jednotlivé výsledky dílčích analýz experimentálních dat.

6.3.1 Relevance mapy

Toto kritérium hodnotí, jak moc je daná mapa relevantní k navigaci a tedy jak je robot schopný se podle ní navigovat v měnícím se prostředí. Relevance mapy lze snadno porovnávat podle počtu význačných bodů v ní uložených, které vytvořily správné shodné dvojice s význačnými body extrahovanými z příchozího obrázku.

V grafu 25 jsou uvedeny výsledky vyhodnocení tohoto kritéria pro všechny tři mapy z šestého, devátého a desátého průjezdu. Jednotlivé grafy průjezdů udávají pravděpodobnost, že počet správných shodných dvojic je vyšší než daná hodnota. Nejlepší křivka pro toto kritérium je ta, která má pro danou pravděpodobnost největší počet správných shodných dvojic.



Obrázek 25: Pravděpodobnost, že počet správných shodných dvojic je vyšší než daná hodnota

Jak se dalo předpokládat, nejrelevantnější mapou pro navigaci je plastická mapa, neboť tato mapa si při každém průjezdu vytvoří zcela novou

mapu význačných bodů a v důsledku toho se rapidně zvyšuje pravděpodobnost, že tyto uložené význačné body budou viditelné a tedy i dobře párovatelné při dalším průjezdu. Pro statickou mapu se postupně při každém dalším průjezdu snižuje počet správných dvojic, což naznačuje, že postupem času není možné se podle statické mapy při zhoršujících se světelných podmínkách navigovat. Adaptivní mapa pak zde vystupuje jako kompromis mezi oběma metodami. Protože se v ní obměňuje jen určitý počet význačných bodů, nevytvoří tolik správných shodných dvojic při dalším průjezdu. Nicméně na prvním datasetu bylo experimentálně ověřeno, že se nelze při zhoršujících se světelných podmínkách navigovat podle statické mapy a je potřeba vytvořit novou mapu manuální jízdou operátorem. Naopak na druhém a třetím datasetu bylo ukázáno, že robot dokáže držet krok s těmito zhoršujícími se podmínkami a zvládne se navigovat podle adaptivní mapy i s nižším počtem správných shodných dvojic.

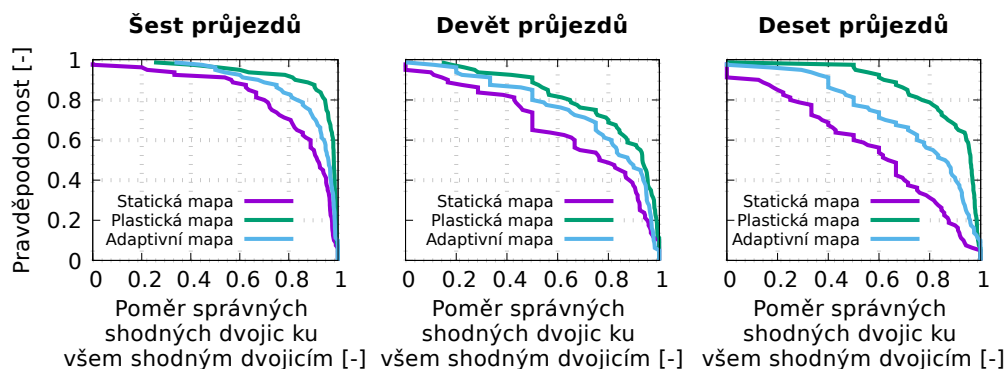
Statistické testování potvrzuje, že nejrelevantnější mapou pro navigaci je plastická, méně relevantní je adaptivní a nejméně relevantní je statická mapa. To znamená, že skutečně plastická mapa umí nejlépe držet krok s měnícím se prostředím a naopak statická mapa se změnám prostředí přizpůsobit neumí. Adaptivní mapa se částečně přizpůsobuje vyvíjejícímu se okolí díky obměně určitého počtu elementů mapy.

6.3.2 Přesnost mapy

Přesnosti jednotlivých map pro navigaci lze porovnat poměrem správně vytvořených shodných dvojic ku celkovému počtu vytvořených shodných dvojic, neboť ten je v důsledku měnících se venkovních podmínek pro dané typy map různý. Je totiž nutné také uvažovat počet špatně vytvořených shodných dvojic, aby bylo možné posoudit kvalitu a přesnost mapy při navigaci.

Počet správných shodných dvojic nám dobře vypovídá o schopnosti navigace, avšak nepodává nám žádnou informaci o kvalitě dané mapy. Například pro statickou mapu se počet správných shodných dvojic snižuje vlivem zhoršujícího se osvětlení, kdežto pro plastickou mapu je tento počet stále skoro stejný. To nám ovšem nic neříká celkovém počtu vytvořených shodných dvojic a o počtu, kolik z nich bylo nesprávných, které v podstatě kazí kvalitu mapy, neboť v mapě bychom chtěli uchovávat co nejvíce správných shodných dvojic a ideálně žádné špatné shodné dvojice.

Následující graf 26 popisuje pravděpodobnost, že poměr správných shodných dvojic vůči celkovému počtu vytvořených dvojic je větší než daná hodnota. Nejlepším poměrem je v tomto případě 1, neboť se snažíme dosáhnout



Obrázek 26: Pravděpodobnost, že poměr správných shodných dvojic vůči celkovému počtu vytvořených dvojic je větší než daná hodnota

situace, kdy všechny vytvořené shodné dvojice budou správné. Tedy nejlepší křivkou je ta, která má pro danou pravděpodobnost poměr co nejbližší číslu 1.

Z grafu 26 můžeme vidět, že tento poměr opět nejlépe vychází pro plastickou mapu, neboť umí držet krok s měnícími se podmínkami a vytváří si mapu při každém průjezdu, takže se díky tomu při následujícím průjezdu vytvoří velký počet správných shodných dvojic s mapou z minulého průjezdu. Naopak pro statickou mapu se tento poměr kazí v důsledku rezistence těmto změnám, kvůli kterým se postupně snižuje počet shodných význačných bodů v mapě s těmi extrahovanými z příchozího obrázku, a tedy i poměr těch správných dvojic vůči všem vytvořeným dvojicím. Oproti předchozímu kritériu se zde ale pravděpodobnostní průběh adaptivní mapy podobá více plastické mapě, což potvrzuje experiment, při kterém bylo možné se navigovat podle adaptivní mapy i při zhoršujících se světelných podmínkách, ale podle statické nikoliv.

Stejně jako v předchozí sekci, statistické testování potvrzuje, že nejpřesnější mapou je plastická, méně přesná je adaptivní a nejméně přesná je statická mapa, což opět odpovídá schopnosti přizpůsobit se měnícímu se prostředí.

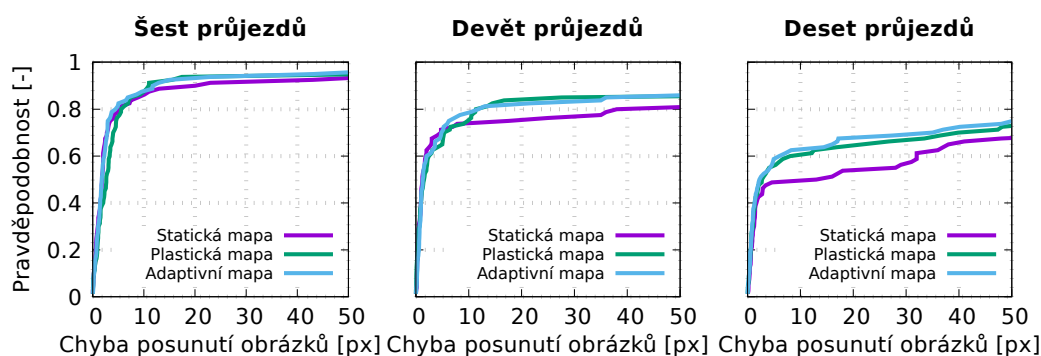
6.3.3 Chyba lokalizace

Předchozí dvě kritéria se zabývaly tím, jak se navzájem párují význačné body. Nicméně pro spolehlivou navigaci je nutné zohledňovat i lokalizaci. Její chyba zde vzniká zejména z nežádoucího posunutí obrázků, které vzniká v

6.3 Analýza a testování kritérií

důsledku toho, že se robot odchyluje od původní trajektorie. K tomu dochází z důvodu nepřesného vyhodnocení směru robota.

Toto kritérium tedy popisuje chybu určení směru robota podle význačných bodů. Absolutní chybu této nepřesnosti vypočítáme na základě rozdílu posunutí jednotlivých obrázků vůči sobě z histogramové metody našeho navigačního systému a podle referenčních hodnot získaných manuální anotací. Histogram se vytváří z jednotlivých vypočítaných rozdílů horizontálních pozic význačných bodů z mapy a extrahovaných bodů z obrázku. Maximum tohoto histogramu pak indikuje nejčastější velikost spojnice shodných dvojic. Referenční hodnoty se vypočítávají z posunutí obrázků vůči sobě podle toho, jak se je člověk snaží manuálně překrýt tak, aby se co nejvíce shodovaly. Aby nevznikala chyba lokalizace, měla by být tato dvě čísla stejná, respektive jejich rozdíl by měl být nulový.



Obrázek 27: Pravděpodobnost, že chyba posunutí bude menší než daný počet pixelů

Graf 27 znázorňuje pravděpodobnost, že chyba posunutí bude menší než daný počet pixelů. Ideální křivka je taková, která má pro danou pravděpodobnost chybu pozice co nejmenší, nejlépe rovnou 0 pixelům.

Po šesti průjezdech je chyba posunutí obrázků pro všechny typy map skoro stejná. Rozdíl se začne projevovat až později, kdy je zřejmé, že statická mapa má pro danou pravděpodobnost mnohem větší chybu posunutí obrázků než ostatní porovnávané mapy. To odpovídá tomu, že neurčitost lokalizace podle význačných bodů uložených ve statické mapě narůstá v čase v důsledku toho, že se během jednotlivých průjezdů zcela setmí. Při devátém průjezdu je chyba posunutí obrázků pro plastickou a adaptivní mapu také velmi podobná, ale při desátém průjezdu, při kterém došlo k výraznému setmění, je již jasněji vidět, že adaptivní mapa má pro danou pravděpodobnost menší

chybu posunutí obrázků než plastická mapa. Tento výsledek lze také interpretovat tak, že se robot lokalizuje podle pozice význačných bodů lépe podle adaptivní mapy než podle plastické mapy.

Tato série grafů 27 pochází z prostředí, ve kterém se projevují (velké) změny prostředí. Pro statistické testování však bylo zkoumáno jak prostředí, které se mění, tak i to, které se nemění. Neměnicí se prostředí pak bylo vytvořeno tak, že se jedna mapa (představující jeden průjezd) opakovaně přehrávala tak, aby simulovala prostředí, které se téměř nemění.

V grafu 27 si můžeme všimnout, že pro devátý průjezd není na první pohled zcela jasné, zda má adaptivní mapa menší chybu pozice obrázků než plastická. Proto je zde velmi vhodné provést statistické testování, a to pro obě výše zmíněná prostředí. Pro proměnlivé prostředí (shodující se s průběhy v grafu) má největší chybu posunutí statická mapa, ale pro plastickou a adaptivní mapu vychází chyby podobně.

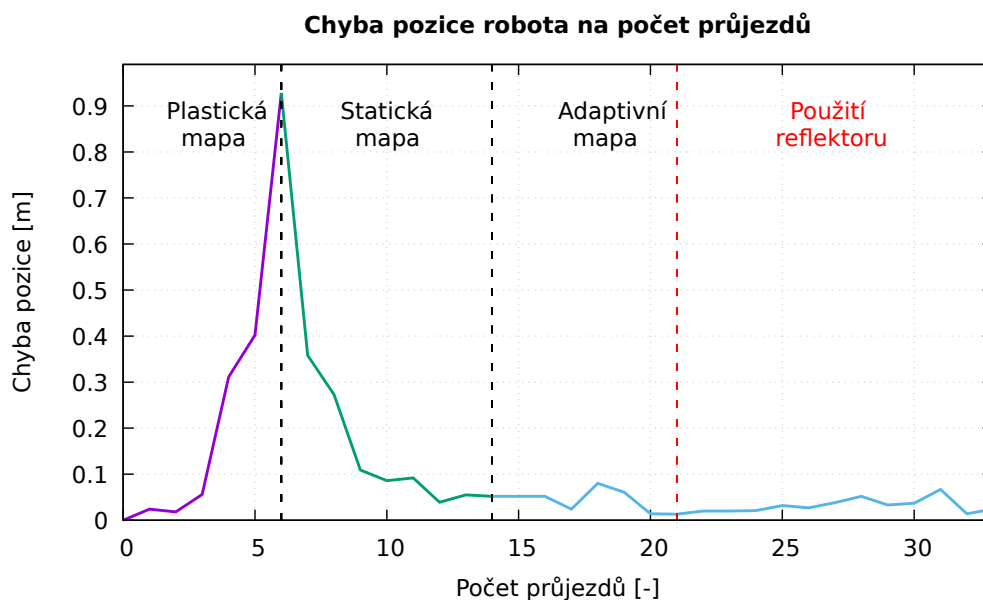
Udělalí jsme tedy ještě jeden experiment na datech z prostředí, které se téměř nemění. Po jedenácti průjezdech statistický test potvrdil, že nejmenší chybu posunutí obrázků má adaptivní mapa, potom statická a nejhorší chybu posunutí obrázků má plastická mapa. V té se totiž při téměř neměnicím se okolí projevuje postupně kumulovaný šum pocházející z nepřesnosti senzorů. Význačné body v této mapě nemají vztah k žádným jiným význačným bodům, protože se pokaždé vytváří zcela nová mapa. V důsledku toho se pak postupně posouvá obrázek, který se ukládá do mapy.

Poslední test tedy ukázal, že při navigaci podle plastické mapy pozice robota pomalu diverguje, což se neprojeví při zkoumání relevance či přesnosti mapy pro navigaci, protože po sobě vytvářené mapy jsou vhodné pro hledání shodných bodů. Ty se však, jak se ukázalo, hledají v navzájem posunutých obrázcích. Toto statistické testování tedy potvrzuje, že plastická mapa je významně zatížena chybou lokalizace podle význačných bodů.

6.3.4 Přesnost navigace

Nepřesnost lokalizace je také zatížena chybou pozice robota z odometrického systému. Tato neurčitost spolu s nepřesností lokalizace podle význačných bodů byla zjišťována pomocí externího lokalizačního systému whycon [], který může nejlépe potvrdit schopnost i kvalitu navigace a lokalizace robota tak, že bude sledovat odchylku pozice robota od požadované trasy.

V grafu 28 je vykreslen vývoj chyby pozice robota v závislosti na počtu opakovaných průjezdů. Při experimentálním testování navržených metod správy map se nejdříve robot navigoval podle plastické mapy. Jak můžeme



Obrázek 28: Chyba pozice robota v závislosti na počtu průjezdů po naučené trase podle externího lokalizačního systému

vidět v prvních šesti průjezdech, chyba pozice robota narůstala vlivem kumulativní neurčitosti odometrie a chyby lokalizace podle význačných bodů. Při takovýchto průjezdech zatížených chybou pozice se vytvářela nová mapa pro další průjezd, což postupně vede k odchylování se od původní trajektorie. Během osmi následujících jízd po naučené trase podle statické mapy se robot dokázal vrátit zpět na původní trajektorii, a tak se experimentálně potvrdila platnost konvergence této mapy matematicky dokázaná v [6], [45] a experimentálně ověřená v [61]. V následujících devatenácti průjezdech pak graf potvrzuje hypotézu, že adaptivní mapa není zatížena chybou lokalizace i přes změnu prostředí, a tedy se neodchýlí od původní trajektorie. V grafu je ještě znázorněn 21. průjezd, od kterého se úplně setmělo a bylo nutné, aby si robot trasu při opakovaných průjezdech osvětloval vlastním reflektorem.

6.4 Shrnutí a diskuze experimentálních výsledků

Abychom mohli vyhodnotit, která mapa, a tedy i metoda navigace podle této mapy, je nejlepší, musíme posoudit všechna výše uvedená kritéria dohromady.

Plastická mapa je nejrelevantnější mapou k aktuálnímu průjezdu díky velkému počtu správných shodných dvojic a také má největší přesnost mapy, jak



Obrázek 29: Měření přesnosti navigace pomocí externího lokalizačního systému [11]

bylo ověřeno poměrem těchto správných shodných dvojic vůči všem vytvořeným shodným dvojicím. Akumuluje se u ní chyba lokalizace podle význačných bodů, což jsem ověřila rozdílem posunutí obrázků podle histogramové metody našeho navigačního systému a podle referenčních hodnot určených člověkem. Statistické testování ukázalo, že plastická mapa má nejhorší chybu lokalizace podle význačných bodů, což se ukázalo i při vyhodnocování přesnosti navigace, ve které spolu s kumulující se chybou odometrie měla největší a dále rostoucí odchylku od původní trasy. Kvůli kumulujícímu se posouvání pozice se postupně snižuje schopnost navigace. Tyto aspekty tedy znemožňují dlouhodobou navigaci podle plastické mapy.

Statická mapa není schopná držet krok s proměnlivým prostředím, což bylo ukázáno jednak nejméně relevantní mapou k aktuálnímu průjezdu, ale i podle nejmenší přesnosti mapy. Statická mapa tyto aspekty v čase postupně ztrácí. Důvodem je ubývající počet vytvořených shodných dvojic v důsledku změny prostředí, což ovlivňuje i snižování správných shodných dvojic. Tento efekt je také spojený s tím, že při zhoršujících se světelných podmínkách dochází k velké chybě lokalizace podle význačných bodů, což odpovídá velké chybě posunutí obrázků. Lokalizace podle externího lokalizačního systému prokázala, že pokud se prostředí nemění, robot jezdí podle stále stejné trajektorie

vytvořené v mapovací fázi s minimálními odchylkami. Zároveň provedený experiment potvrdil, že podle této mapy je robot schopný zkonvergovat zpět na původní naučenou trajektorii při malých změnách vzhledu prostředí. Toto je nejdůležitější a nejlepší vlastnost této mapy.

Adaptivní mapa kombinuje pozitivní vlastnosti obou map a tím se jeví jako nejvhodnější mapa pro dlouhodobou navigaci v proměnlivých podmínkách. Tato mapa umí držet krok se změnami prostředí, v čemž vyniká oproti statické mapě, a zároveň se při autonomní navigaci neakumuluje chyba pozice robota, čímž je výhodnější oproti plastické mapě. Adaptivní mapa sice není tak relevantní k aktuálnímu průjezdu jako plastická mapa a ani nemá takovou přesnost, ale na druhou stranu vyniká v lokalizaci a přesnosti navigace. Tato mapa má nejmenší chybu posunutí obrázků a díky tomu má také nejmenší odchylku od původní trajektorie. V tomto kritériu je nejlepší ze všech uvedených druhů map zejména proto, že si udržuje kvalitní význačné body v mapě, které jsou stále vidět již od doby mapování, a následně přidává nové vhodné elementy do mapy. Tedy díky tomu je možné omezovat jev, který vzniká u plastické mapy - vlivem vytváření mapy při každém průjezdu se plastické mapě zhoršuje schopnost lokalizace, což má za následek i nekvalitní navigaci. Na základě těchto vlastností je robot navigující se podle adaptivní mapy schopný při zhoršujícím se osvětlení vykonávat stále stejnou či velmi podobnou trasu, kterou se naučil v mapovací fázi při dobrém osvětlení.

Z výše uvedených porovnání je zřejmé, že se podařilo navrhnout i ověřit takovou metodu správy map, která se umí adaptovat na měnící se prostředí pomocí obměňování určitého počtu elementů v mapě a podle které je možné se dlouhodobě navigovat, protože zde nevzniká chyba lokalizace, která by závažně kazila kvalitu navigace. Tato mapa se tedy umí vyrovnat s oběma typy neurčitostí - umí omezit integrování neurčitých sensorických informací do mapy a zároveň je schopna si ukládat nové informace související se změnou prostředí.

7 Závěr

Hlavním cílem této práce bylo rozšířit navigační systém BearNav [61] pro schopnost dlouhodobé navigace v proměnlivém prostředí. Tento systém je založený na metodě *teach-and-repeat*, ve které si robot nejdříve vytváří mapu během (tele)operované jízdy operátorem a následně se podle této mapy nadále autonomně naviguje. Avšak jakmile se v důsledku přirozených procesů prostředí naučené trasy změní, robot ztrácí svou schopnost autonomní navigace a vyžaduje vytvoření nové mapy.

Pro řešení tohoto problému jsem zhodnotila stávající metody navigace a údržby map a navrhla novou metodu, která je kombinuje a vytváří tak nejlepší model prostředí vhodný pro dlouhodobou navigaci. Nová metoda správy map, *adaptivní mapa*, vychází ze dvou již známých druhů map, které však nedokážou danou problematiku úspěšně řešit. Prvním typem je *statická mapa*, která je vytvořena při prvním průjezdu robotem a dále se již nemění. Je vhodná ale pouze pro robustní navigaci v prostředí, které nepodléhá změnám. Druhým způsobem je navigace podle *plastické mapy*, která se vytváří při každém průjezdu trasou. Tato mapa sice umí držet krok s proměnlivým okolím, ale není schopná robustní dlouhodobé navigace. Vytvořená adaptivní mapa kombinuje pozitivní vlastnosti výše zmíněných druhů. Díky těmto vlastnostem je pak schopna robustní dlouhodobé navigace v proměnlivém prostředí.

Novou metodu jsem zaintegrovala do operačního systému ROS a ověřila její schopnost dlouhodobé navigace na vytvořených datasetech. Adaptivní mapa vytvoří při prvním průjezdu mapu, kterou při dalších průjezdech nadále upravuje. Aby se vyrovnala s neurčitostmi vzniklými buď integrováním senzorických dat se šumem či neurčitostmi souvisejícími se změnou prostředí, je nutné vybírat, jakou informaci do mapy přidat a jakou odebrat. Tato metoda přidává na základě ohodnocování do mapy nové informace tak, aby byla schopna držet krok s vyvíjejícím se okolím, a odebírá z mapy ty elementy, které kazí její spolehlivou navigaci.

Na základě čtyř kritérií souvisejících se změnou prostředí porovnatelnou podle viditelnosti jednotlivých význačných bodů a s chybou lokalizace a schopností přesné navigace jsem porovnála všechny tři implementované metody. Z experimentálních výsledků se potvrdilo, že adaptivní mapa je schopna díky údržbě mapy držet krok s měnícím se prostředím, ve kterém je robot schopen se dlouhodobě navigovat podle této mapy.

V následující práci by bylo vhodné vymyslet optimální algoritmus, který by adaptivně měnil počet obměňovaných elementů v mapě. Návrhem pro

další práci může být i vytvoření adaptivní mapy pro rychlosti robota tak, aby uměl zpomalovat v náročnějších částech trasy a mohl zrychlovat v těch méně obtížných.

Reference

- [1] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Autonomous Robots*, 34(3):189–206, 2013.
- [2] David J Austin and Brenan J McCarragher. Geometric constraint identification and mapping for mobile robots. *Robotics and Autonomous Systems*, 35(2):59–76, 2001.
- [3] David Filliat and Jean-Arcady Meyer. Map-based navigation in mobile robots:: I. a review of localization strategies. *Cognitive Systems Research*, 4(4):243–282, 2003.
- [4] Benjamin Kuipers, Joseph Modayil, Patrick Beeson, Matt MacMahon, and Francesco Savelli. Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 5, pages 4845–4851. IEEE, 2004.
- [5] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*, pages 146–151. IEEE, 1997.
- [6] Tomáš Krajník. *Large-scale mobile robot navigation and map building*. PhD thesis, Ph. D. thesis, Czech Technical University in Prague, 1999, Draft, 2011.
- [7] Steven M LaValle. Rapidly-exploring random trees: A new tool for path planning. 1998.
- [8] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [9] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006.
- [10] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767, 2004.

REFERENCE

- [11] Whycon localization system - non-ros version used in the experiments for the jint paper. <https://github.com/gestom/whycon-orig>. Accessed: 2018-05-20.
- [12] Johann Borenstein, Hobart R Everett, Liqiang Feng, and David Wehe. Mobile robot positioning-sensors and techniques. Technical report, NAVAL COMMAND CONTROL AND OCEAN SURVEILLANCE CENTER RDT AND E DIV SAN DIEGO CA, 1997.
- [13] P Goel, SI Roumeliotis, and GS Sukhatme. Robot localization using relative and absolute position estimates in proc ieee int. In *Conference on Robots and Systems, October, Kyongju, Korea, 1999*.
- [14] Frédéric Chenavier and James L Crowley. Position estimation for a mobile robot using vision and odometry. In *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pages 2588–2593. IEEE, 1992.
- [15] Hugh HS Liu and Grantham KH Pang. Accelerometer for mobile robot positioning. *IEEE Transactions on Industry Applications*, 37(3):812–819, 2001.
- [16] Kiyoshi Komoriya and Eiinei Oyama. Position estimation of a mobile robot using optical fiber gyroscope (ofg). In *Intelligent Robots and Systems' 94. 'Advanced Robotic Systems and the Real World', IROS'94. Proceedings of the IEEE/RSJ/GI International Conference on*, volume 1, pages 143–149. IEEE, 1994.
- [17] Lauro Ojeda, Johann Borenstein, and Ann Arbor. Experimental results with the kvh c-100 fluxgate compass in mobile robots. *Ann Arbor*, 1001(48109):42110, 2000.
- [18] Richard Volpe. Mars rover navigation results using sun sensor heading determination. In *Intelligent Robots and Systems, 1999. IROS'99. Proceedings. 1999 IEEE/RSJ International Conference on*, volume 1, pages 460–467. IEEE, 1999.
- [19] Amy J Briggs, Daniel Scharstein, Darius Braziunas, Cristian Dima, and Peter Wall. Mobile robot navigation using self-similar landmarks. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 2, pages 1428–1434. IEEE, 2000.

REFERENCE

- [20] Rainer Mautz and Sebastian Tilch. Survey of optical indoor positioning systems. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–7. IEEE, 2011.
- [21] Stefano Panzieri, Federica Pascucci, and Giovanni Ulivi. An outdoor navigation system using gps and inertial platform. *IEEE/ASME transactions on Mechatronics*, 7(2):134–142, 2002.
- [22] Kazunori Ohno, Takashi Tsubouchi, Bunji Shigematsu, Shoichi Maeyama, and Shin'ichi Yuta. Outdoor navigation of a mobile robot between buildings based on dgps and odometry data fusion. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 2, pages 1978–1984. IEEE, 2003.
- [23] Lindsay Kleeman and Roman Kuc. Mobile robot sonar for target localization and classification. *The International Journal of Robotics Research*, 14(4):295–318, 1995.
- [24] Cheng-Chih Lin and R Lal Tummala. Mobile robot navigation using artificial landmarks. *Journal of Robotic Systems*, 14(2):93–106, 1997.
- [25] Gijeong Jang, Sungho Lee, and Inso Kweon. Color landmark based self-localization for indoor mobile robots. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 1, pages 1037–1042. IEEE, 2002.
- [26] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, Mihai Dolha, and Michael Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927–941, 2008.
- [27] Alberto Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.
- [28] David Kortenkamp, L Douglas Baker, and Terry Weymouth. Using gateways to build a route map. In *Intelligent Robots and Systems, 1992., Proceedings of the 1992 IEEE/RSJ International Conference on*, volume 3, pages 2209–2214. IEEE, 1992.
- [29] Sebastian Thrun and Arno Bücken. Integrating grid-based and topological maps for mobile robot navigation. In *Proceedings of the National Conference on Artificial Intelligence*, pages 944–951, 1996.

REFERENCE

- [30] Benjamin Kuipers. The spatial semantic hierarchy. *Artificial intelligence*, 119(1-2):191–233, 2000.
- [31] Michael Bosse, Paul Newman, John Leonard, Martin Soika, Wendelin Feiten, and Seth Teller. An atlas framework for scalable mapping. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pages 1899–1906. IEEE, 2003.
- [32] Karel Košnar, Tomáš Krajník, Vojtech Vonásek, and Libor Preucil. Lama-large maps framework. In *Proceedings of Workshop on Field Robotics, Civilian-European Robot Trial*, pages 9–16, 2009.
- [33] Sebastian Thrun and John J Leonard. Simultaneous localization and mapping. In *Springer handbook of robotics*, pages 871–889. Springer, 2008.
- [34] Andrew J Davison and David W Murray. Simultaneous localization and map-building using active vision. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):865–880, 2002.
- [35] Stephen Se, David Lowe, and Jim Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *The international Journal of robotics Research*, 21(8):735–758, 2002.
- [36] Sebastian Thrun and Michael Montemerlo. The graph slam algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006.
- [37] Robert Sim, Pantelis Elinas, Matt Griffin, James J Little, et al. Vision-based slam using the rao-blackwellised particle filter. In *IJCAI Workshop on Reasoning with Uncertainty in Robotics*, volume 14, pages 9–16, 2005.
- [38] Dirk Hahnel, Wolfram Burgard, Dieter Fox, and Sebastian Thrun. An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 1, pages 206–211. IEEE, 2003.
- [39] O Hachour. Path planning of autonomous mobile robot. *International journal of systems applications, engineering & development*, 2(4):178–190, 2008.

REFERENCE

- [40] Gene Eu Jan, Ki Yin Chang, and Ian Parberry. Optimal path planning for mobile robot navigation. *IEEE/ASME Transactions on mechatronics*, 13(4):451–460, 2008.
- [41] Zhanying Zhang and Ziping Zhao. A multiple mobile robots path planning algorithm based on a-star and dijkstra algorithm. *International Journal of Smart Home*, 8(3):75–86, 2014.
- [42] Francisco Bonin-Font, Alberto Ortiz, and Gabriel Oliver. Visual navigation for mobile robots: A survey. *Journal of intelligent and robotic systems*, 53(3):263–296, 2008.
- [43] Pablo De Cristóforis, Matías A Nitsche, Tomáš Krajník, and Marta Mejail. Real-time monocular image-based path detection. *Journal of Real-Time Image Processing*, 11(2):335–348, 2016.
- [44] Clark F Olson, Larry H Matthies, Marcel Schoppers, and Mark W Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215–229, 2003.
- [45] Tomáš Krajník, Jan Faigl, Vojtěch Vonásek, Karel Košnar, Miroslav Kulich, and Libor Přeučil. Simple yet stable bearing-only navigation. *Journal of Field Robotics*, 27(5):511–533, 2010.
- [46] Zhichao Chen and Stanley T Birchfield. Qualitative vision-based mobile robot navigation. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2686–2692. IEEE, 2006.
- [47] Winston Churchill and Paul Newman. Experience-based navigation for long-term localisation. *The International Journal of Robotics Research*, 32(14):1645–1661, 2013.
- [48] Michael Paton, Kirk MacTavish, Laszlo-Peter Berczi, Sebastian Kai van Es, and Timothy D Barfoot. I can see for miles and miles: An extended field test of visual teach and repeat 2.0. In *Field and Service Robotics*, pages 415–431. Springer, 2018.
- [49] Feras Dayoub and Tom Duckett. An adaptive appearance-based map for long-term topological localization of mobile robots. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3364–3369. IEEE, 2008.

REFERENCE

- [50] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.
- [51] Peter Mühlfellner, Mathias Bürki, Michael Bosse, Wojciech Derendarz, Roland Philippsen, and Paul Furgale. Summary maps for lifelong visual localization. *Journal of Field Robotics*, 33(5):561–590, 2016.
- [52] Peter Biber, Tom Duckett, et al. Dynamic maps for long-term operation of mobile service robots. In *Robotics: science and systems*, pages 17–24, 2005.
- [53] Tomáš Krajník, Jaime P Fentanes, Joao M Santos, and Tom Duckett. Fremen: Frequency map enhancement for long-term mobile robot autonomy in changing environments. *IEEE Transactions on Robotics*, 33(4):964–977, 2017.
- [54] David M Rosen, Julian Mason, and John J Leonard. Towards lifelong feature-based mapping in semi-static environments. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 1063–1070. IEEE, 2016.
- [55] Peer Neubert, Niko Sunderhauf, and Peter Protzel. Appearance change prediction for long-term navigation across seasons. In *Mobile Robots (ECMR), 2013 European Conference on*, pages 198–203. IEEE, 2013.
- [56] Stephanie Lowry and Michael J Milford. Supervised and unsupervised linear learning techniques for visual place recognition in changing environments. *IEEE Transactions on Robotics*, 32(3):600–613, 2016.
- [57] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2016.
- [58] Tomáš Krajník, Pablo Cristóforis, Keerthy Kusumam, Peer Neubert, and Tom Duckett. Image features for visual teach-and-repeat navigation in changing environments. *Robotics and Autonomous Systems*, 88:127–141, 2017.
- [59] Elmar Mair, Gregory D Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In *European conference on Computer vision*, pages 183–196. Springer, 2010.

REFERENCE

- [60] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.
- [61] Tomas Krajník, Filip Majer, Lucie Halodová, Jan Bayer, Tomas Vintř, and Jan Faigl. Navigation without localisation: reliable teach and repeat based on the convergence theorem. *arXiv preprint arXiv:1711.05348*, 2017.
- [62] Lucie Halodová and Tomáš Krajník. Exposure setting for visual navigation of mobile robots, 2017.

Příloha

Obsah přiloženého CD

V tabulce 1 jsou uvedena jména všech kořenových adresářů přiloženého CD s popisem obsahu.

Jméno adresáře	Popis obsahu
text	bakalářská práce ve formátu pdf
sources	zdrojové kódy
datasets	mapy pro navigaci

Tabulka 1: Obsah přiloženého CD