



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

STEREO REKONSTRUKCE PŘÍČNÉHO PROFILU VOZIDLA

STEREO RECONSTRUCTION OF VEHICLES' CROSS SECTION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jan Boch

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Peter Honec, Ph.D.

BRNO 2024

Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Jan Boch

ID: 220964

Ročník: 2

Akademický rok: 2023/24

NÁZEV TÉMATU:

Stereo rekonstrukce příčného profilu vozidla

POKYNY PRO VYPRACOVÁNÍ:

Moderní dopravní infrastruktura je konstruována s přihlédnutím k obvyklým profilům vozidel, které ji využívají. Existují ale případy, kdy řidiči vozidla nerespektují dopravní značení, a vjedou do míst, která jsou pro danou třídu vozidla neprůjezdná - úzké průjezdy, nízké mosty. Někdy také dochází k posunutí nákladu vozidla, který potom přesahuje obrys vozidla, nebo celkově nevhodnému naložení vozidla. Z těchto a dalších praktických důvodů je výhodné měřit příčné profily vozidel. Cílem této práce je navrhnout systém pro rekonstrukci příčného profilu vozidla za použití rekonstrukce z více kamer (případně v kombinaci s dalšími 2D a 3D senzory).

1. Vypracujte rešerši podobných řešení a dostupných metod a prostředků
2. Navrhněte systém zpracování snímků a sestavte požadavky na HW systému
3. Navrhněte konfiguraci HW systému (počet a typ kamer/senzorů, parametry kamer etc.)
4. Nasnímejte minimální dataset v navrhované konfiguraci a ověřte její vhodnost
5. Implementujte navrhovaný HW
6. Nasnímejte dataset pro vyhodnocení
7. Implementujte navrhovaný systém zpracování snímků a prezentace výsledků
8. Kvantitativně a kvalitativně vyhodnoťte úspěšnost měření

DOPORUČENÁ LITERATURA:

[1] HARTLEY, Richard a Andrew ZISSERMAN. Multiple view geometry in computer vision. 2nd ed. Cambridge: Cambridge University Press, 2003. ISBN 978-0521540513.

[2] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. Image processing, analysis, and machine vision. Fourth edition. Stamford, CT: Cengage Learning, 2015. ISBN 978-1133593607.

Termín zadání: 5.2.2024

Termín odevzdání: 15.5.2024

Vedoucí práce: Ing. Peter Honec, Ph.D.

Konzultant: Ing. Tomáš Zemčík

doc. Ing. Petr Fiedler, Ph.D.

předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato diplomová práce se zabývá návrhem stereo systému zpracování snímků vozidel a následné rekonstrukci jejich 3D modelu. Cílem práce je rekonstruovat model vozidla pro kontrolu jeho rozměrů a nákladu na které jsou kladeny restriktce v případě využití kritické dopravní infrastruktury (tunely, mosty). Práce začíná matematickým popisem snímání scény a pokračuje řešením teoretických možností konfigurace systému. Jako nejvhodnější konfigurace byla zvolena jedna 3D kamera od společnosti Intel RealSense a jedna RGB kamera. Softwarové řešení pracuje s návrhem několika různých algoritmů z nichž stojí za zmínku rekonstrukce pomocí hloubkových map nebo například triangulace. Výsledné modely v poslední kapitole obsahují data pouze z výše zmíněné 3D kamery. Ze snímků získaných RGB kamerou se model z 3D kamery nepodařilo nijak vylepšit a proto ji lze prohlásit za redundantní. Výsledný 3D model je i tak dostačující pro účely kontroly vozidla a v případě pokračující práce je třeba navrhnout jinou, optimálnější konfiguraci.

KLÍČOVÁ SLOVA

Počítačové vidění, Systém zpracování snímků, Rekonstrukce 3D modelu, Stereovize, 3D kamera, Vozidla, Kalibrace, Rektifikace, Hloubková mapa, Triangulace, Mračno bodů

ABSTRACT

This master thesis deals with the design of a stereo system for processing vehicle images and subsequent reconstruction of their 3D model. The aim of the thesis is to reconstruct the vehicle model to control its dimensions and load on which restrictions are imposed in case of the use of critical transport infrastructure (tunnels, bridges). The thesis begins with a mathematical description of scene capture and continues with a research of theoretical possibilities of system configuration. One Intel RealSense 3D camera and one RGB camera were chosen as the most suitable configuration. The software solution works with the design of several different algorithms of which it is worth mentioning reconstruction using depth maps or triangulation. The resulting models in the last chapter contain data only from the aforementioned 3D camera. The model from the 3D camera could not be improved with the images from the RGB camera and camera can therefore be declared redundant. Resulting 3D model is still sufficient for the purpose of vehicle inspection and in case of continued work, another, more optimal configuration should be designed.

KEYWORDS

Computer vision, Image processing system, Reconstruction of 3D model, Stereovision, 3D camera, Vehicles, Calibration, Rectification, Depth map, Triangulation, Point cloud

BOCH, Jan. *Stereo rekonstrukce příčného profilu vozidla*. Diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2024. Vedoucí práce: Ing. Peter Honec, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Bc. Jan Boch
VUT ID autora: 220964
Typ práce: Diplomová práce
Akademický rok: 2023/24
Téma závěrečné práce: Stereo rekonstrukce příčného profilu vozidla

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

* Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Ing. Tomáši Zemčíkovi a Skupině počítačového vzdělání za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

Brno

.....

podpis autora

Obsah

Úvod	12
1 Metody snímání 3D objektů	13
1.1 Triangulace	13
1.1.1 Pasivní triangulace	13
1.1.2 Aktivní triangulace	14
1.2 Měření doby letu	16
1.3 Ostatní metody	17
2 Matematický popis snímání scény	18
2.1 Zobrazovací projekce	18
2.1.1 Ortografická projekce	19
2.1.2 Perspektivní projekce	19
2.2 Model dírkové kamery	20
2.3 Vnitřní parametry kamery	21
2.3.1 Zkreslení snímků	22
2.4 Vnější parametry kamery	23
2.5 Stereo systém	24
2.6 Epipolární geometrie	25
2.7 Způsoby reprezentace 3D modelu	28
2.7.1 Mračno bodů	28
2.7.2 Polygonová síť	28
2.7.3 Hloubková mapa	28
2.7.4 Shrnutí	28
3 Návrh systému zpracování stereo snímků	29
3.1 Systém obsahující stereo pár kamer	29
3.2 Systém obsahující 3D kameru	30
3.2.1 Hloubkové kamery založené na principu doby letu	31
3.2.2 Hloubkové kamery založené na principu aktivní triangulace	31
3.2.3 Zhodnocení systému s jednou hloubkovou kamerou	31
3.3 Systém obsahující 3D a RGB kameru	31
3.4 Systém s 3D kamerou a stereo párem RGB kamer	32
4 Volba hardwarových prostředků	33
4.1 Výběr RGB-D kamery	33
4.1.1 Kinect	33
4.1.2 Asus Xtion	34

4.1.3	Další výrobci	35
4.1.4	Intel RealSense	35
4.2	Výběr RGB kamery	36
4.3	Návrh systému uchycení kamer	38
5	Systém zpracování snímků	41
5.1	Softwarové prostředky	41
5.2	Kalibrace stereo páru	42
5.3	Kalibrace 3D kamery	45
5.4	Rektifikace stereo snímků	46
5.5	Tvorba mračna bodů z 3D kamery	48
5.6	Tvorba mračna bodů z barevných snímků	50
5.6.1	Tvorba mračna bodů z hloubkové mapy	50
5.6.2	Tvorba mračna bodů triangulací	54
5.6.3	Tvorba mračna bodů pomocí bundle adjustmentu	58
5.6.4	Tvorba mračna bodů pomocí dalších metod	60
6	Zhodnocení nasnímaného datasetu	62
6.1	Modely z 3D kamery umístěné na spodní pozici	62
6.2	Modely z 3D kamery umístěné na vrchní pozici	63
6.3	Porovnání kvality hloubky modelu z 3D kamery s modelem z neuro- nové sítě	64
	Závěr	66
	Literatura	68
	Seznam symbolů a zkratk	73
A	Obsah přiložené SD karty	74

Seznam obrázků

1.1	Schéma systému využívající pasivní triangulaci	14
1.2	Schéma systému využívající aktivní triangulaci	15
1.3	Schéma systému měření doby letu	17
2.1	Transformace souřadnic při snímání obrazu kamerou	18
2.2	Ortografická projekce	19
2.3	Perspektivní projekce	20
2.4	Model dírkové kamery	20
2.5	Promítnutí pozorovaného bodu scény	21
2.6	Vztah mezi ohniskovou vzdáleností a horizontálním zorným úhlem	22
2.7	Různé typy zkreslení	23
2.8	Transformace souřadnic při snímání obrazu kamerou	24
2.9	Epipolární geometrie	25
2.10	Projekce na souřadnicový systém levé kamery	26
3.1	Barevný a hloubkový snímek scény	29
3.2	Stereo systém dvou kamer	30
3.3	Systém s 3D kamerou a stereo párem RGB kamer	32
4.1	Kinect V1 a Kinect V2	33
4.2	RGB-D kamera Intel RealSense D455	36
4.3	Vztah mezi ohniskovou vzdáleností a zorným polem	37
4.4	Pohyb objektu vůči kamerám	38
4.5	Návrh horizontálního uchycení kamer	39
4.6	Návrh horizontálního uchycení kamer	40
5.1	Návrh systému zpracování snímků	42
5.2	Nalezení rohů šachovnice	44
5.3	Textura sloužící k auto-kalibraci 3D kamery	46
5.4	Rektifikovaný pár	47
5.5	Rektifikované snímky	48
5.6	Hloubkový snímek v barevném podání	49
5.7	Jednotlivá mračna bodů z 3D kamery	50
5.8	Block Matching algoritmus	51
5.9	Systém zpracování snímků pomocí hloubkové mapy	52
5.10	Hloubková mapa získaná pomocí stereoSGBM algoritmu	53
5.11	Mračno bodů vytvořené pomocí hloubkové mapy	54
5.12	Systém zpracování snímků pomocí triangulace	55
5.13	Stereo snímky s nalezenými body zájmu a epipolárními přímkami	56
5.14	Mračno bodů vytvořené pomocí triangulace	58
5.15	Systém zpracování snímků pomocí bundle adjustmentu	60

5.16	Odhad hloubkové mapy pomocí nástroje MiDaS	61
6.1	Mračno bodů vytvořené 3D kamerou (Ford)	62
6.2	Mračno bodů vytvořené 3D kamerou (Škoda)	62
6.3	Mračno bodů vytvořené 3D kamerou (Volkswagen)	63
6.4	Mračno bodů vytvořené 3D kamerou (Volvo)	64
6.5	Mračno bodů vytvořené pomocí neuronové sítě	65
6.6	Porovnání kvality hloubky mračen bodů	65

Seznam tabulek

4.1	Porovnání parametrů obou verzí Kinectu	34
4.2	Parametry zvolené hloubkové kamery	36
4.3	Požadované parametry RGB kamery	38
5.1	Porovnání počtu nalezených bodů zájmu různými algoritmy	57
6.1	Porovnání hustoty bodů výsledných modelů vozidel	64

Úvod

Tato diplomová práce pojednává o návrhu systému pro rekonstrukci příčného profilu vozidla za použití stereo rekonstrukce z více kamer. Cílem práce je navrhnout stereo systém, který bude snímat vozidla v dopravní infrastruktuře a rekonstruovat 3D modely vozidel na základě získaných snímků.

Důvodem k vytvoření tohoto systému je kontrola požadavků, které kladou určité standardní prvky moderní dopravní infrastruktury na vozidla, která je využívají kupříkladu mosty nebo tunely. Na základě rekonstruovaného 3D modelu je možné tyto parametry změřit popřípadě vypočítat. Typickými parametry mohou být například maximální možná výška nebo šířka vozidla. Dále je také možné kontrolovat zda-li není náklad vozidla špatně naložen a zda-li nepřesahuje obrys vozidla.

První část této práce se zabývá metodami pomocí kterých je možné rekonstruovat 3D model z více 2D snímků.

Ve druhé kapitole se poté hovoří o matematickém popisu snímání scény. Je zde popsán postup přepočtu bodu reálného objektu v 3D souřadnicích na pixel tohoto bodu promítnutého na obrazovou rovinu.

Třetí kapitola pojednává o návrhu systému pro zpracování stereo snímků. Jsou zde rozebrány jednotlivé konfigurace s různým počtem kamer v různých kombinacích. U každé možnosti jsou uvedeny výhody i nevýhody takového řešení.

Ve čtvrté kapitole je poté vytvořena rešerše dostupných hardwarových prostředků na současném trhu a dle výsledků srovnání jsou vybrány nejvhodnější prostředky. Dále jsou zde popsány návrhy uchycení kamer na stativ.

Pátá kapitola obsahuje návrh softwarového řešení systému zpracování získaných snímků. Hovoří se zde o několika přístupech, které jsou následně popsány v praktické aplikaci. Jsou zde také prezentovány jejich výsledky společně s jejich vhodností. Na konci kapitoly jsou zmíněny další přístupy, které se netýkají přímo zadání práce, ale slouží pro srovnání funkčnosti jednotlivých metod.

Na závěr jsou v poslední kapitole prezentovány vytvořené modely. Každý z těchto modelů slouží k demonstraci různých jevů, které vznikají při snímání datasetu a vytváření modelů.

1 Metody snímání 3D objektů

Důvodů proč rekonstruovat 3D objekty z 2D snímků je v dnešní době mnoho. Mezi nejběžnější aplikace patří kontrola kvality výrobků na páse, inspekce silničních povrchů, navigace mobilních robotů v prostoru nebo například mapování objektů k vytváření přesných modelů.

Dnešní běžně dostupné kamery většinou používají CCD nebo CMOS snímače pomocí kterých převádějí obraz 3D objektu na 2D snímek. Tímto způsobem ovšem dochází ke ztrátě informace o 3. rozměru snímaného objektu (z-osová souřadnice). Níže popsané metody tedy řeší zpětnou úlohu jak získat 3. souřadnici objektu z 2D snímku.[1]

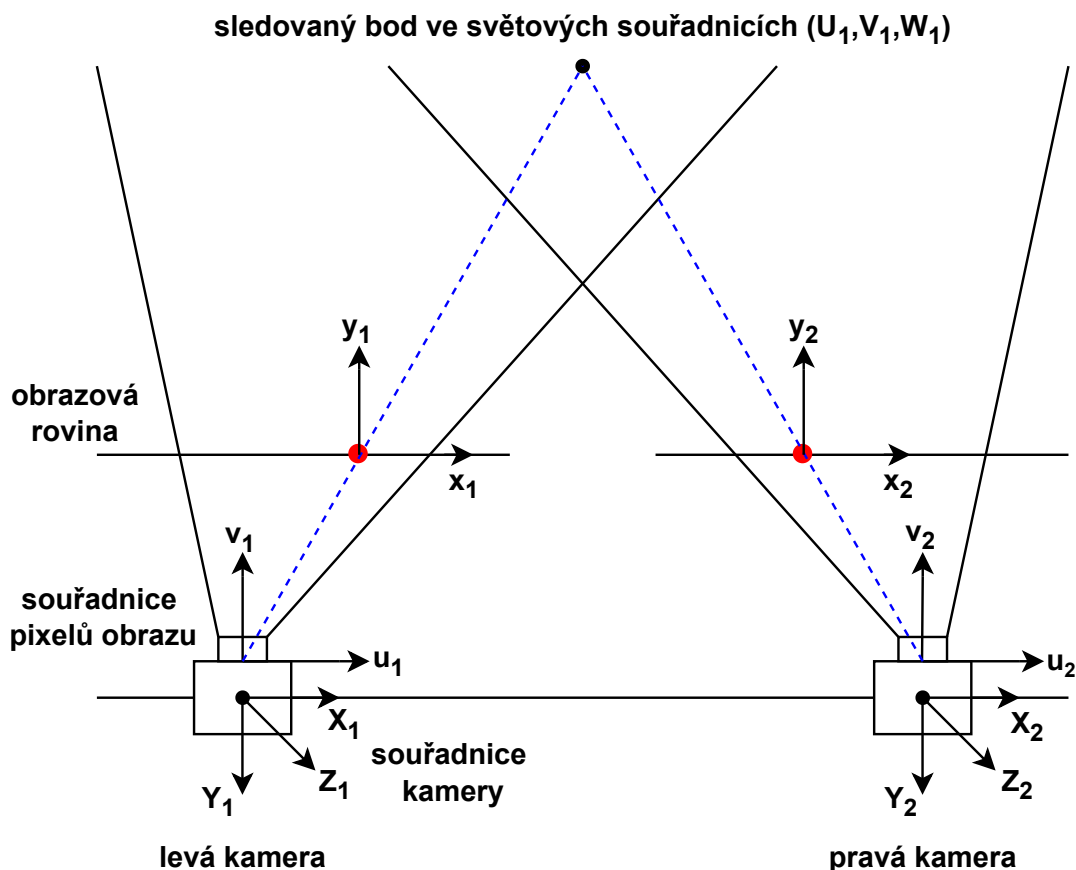
V této práci se dále budeme zabývat optickými metodami zpracování obrazu. Tyto metody se obecně dají dělit na aktivní a pasivní. U pasivních metod je zapotřebí pouze snímač, který zpracovává světelnou informaci sledovaného objektu. V případě aktivních metod je kromě snímače ještě zapotřebí externí zdroj světla.[1]

1.1 Triangulace

Jednou z nejběžněji používaných metod je metoda triangulace. Jak již bylo zmíněno v kapitole výše, při snímání objektu 2D kamerou dochází ke ztrátě hloubkové informace. Proto je pro 3D rekonstrukci zapotřebí získat ještě jednu dodatečnou informaci. V případě triangulační metody můžeme tuto hloubkovou informaci vypočítat ze známé geometrie systému. Metoda triangulace se dělí na pasivní, kdy systém tvoří pouze snímače a sledovaný bod a aktivní, kdy je systém tvořen snímačem, sledovaným objektem a zdrojem světla.[1]

1.1.1 Pasivní triangulace

V případě pasivní triangulace je nejčastěji systém tvořen dvěma kamerami a sledovaným bodem. Ze známé orientace těchto kamer jsme poté schopni dopočítat vzdálenost pozorovaného bodu. Základním předpokladem je, že se pohledy těchto dvou kamer překrývají. Pokud neznáme vzájemnou polohu těchto kamer, můžeme využít předmět o známých rozměrech k jejich kalibraci. Ze znalosti těchto rozměrů jsme poté schopni určit polohu kamer ze snímků ve kterých se kalibrační předmět nachází. Tento typ systému je většinou vhodný pro snímání dynamické scény, kde potřebujeme synchronizovaně získat více snímků z různých pohledů. V případě statického systému lze využít pouze jednu pohyblivou kameru, která nasnímá objekt z různých úhlů.[1]



Obr. 1.1: Schéma systému využívající pasivní triangulaci

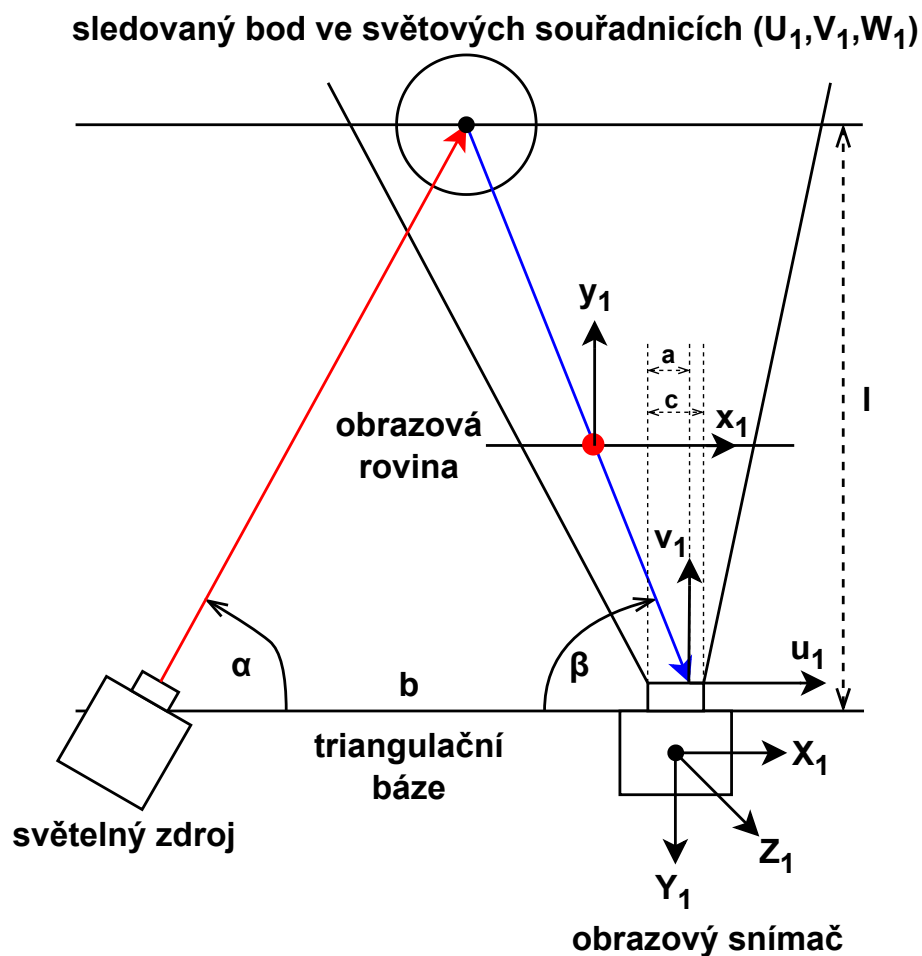
Na obrázku 1.1 je znázorněno promítnutí pozorovaného bodu objektu na obrazové roviny obou kamer (červené body). Tento světelný paprsek se poté promítne na některém z pixelů obrazových snímačů. Při snímání bodu dochází k transformaci mezi jednotlivými souřadnicovými systémy znázorněnými na obrázku. Sledovaný bod ve světových souřadnicích U_1, V_1 a W_1 transformujeme na souřadnicový systém kamery s body X_1, Y_1 a Z_1 . Dle vnitřního uspořádání kamery dále transformujeme tento bod na souřadnice pixelu snímaného obrazu u_1 a v_1 . Tento proces bude dále matematicky popsán v kapitole 2.

1.1.2 Aktivní triangulace

U metody aktivní triangulace je důležitým prvkem systému laser nebo jiný aktivní prvek, který osvětluje snímaný předmět. Tento prvek společně se snímaným předmětem a kamerou tvoří trojúhelník. Vzdálenost b mezi aktivním prvkem a kamerou se nazývá triangulační optická báze. Úhel α , který svírá paprsek aktivního prvku s triangulační bází je neměnný. Úhel β , který svírá odražený paprsek s triangulační

bázi je proměnlivý a díky tomu dochází k osvětlení jiného bodu na obrazovém snímáči (viz Obr. 1.2). Vzdálenost bodu může být poté vypočtena z hodnot těchto dvou úhlů, délky triangulační báze a parametrů kamery.[2]

V případě že chceme vypočítat vzdálenost pouze jednoho bodu (1D) nám jako aktivní prvek poslouží laserový paprsek. Většinou však chceme určit vzdálenost několika bodů nebo celého objektu a v tomto případě poté využijeme světelný pruh pro 2D triangulaci nebo strukturovaný světelný svazek pro 3D triangulaci. Běžně se poté využívá více laserových nebo jiných světelných zdrojů, které tyto pruhy vytvářejí za pomoci různých optických čoček. Promítnutý vzor se deformuje dle tvaru objektu na který dopadá. Z této informace lze poté určit tvar objektu.[1]



Obr. 1.2: Schéma systému využívající aktivní triangulaci

Pomocí rovnice 1.1 můžeme vypočítat šířku projekce světelného paprsku na obrazový snímáč, kde c je velikost čipu snímáče v milimetrech, pxl celkový počet pixelů snímáče a n n-tý pixel snímáče, na který dopadá světelný paprsek.[2]

$$a = \frac{c * n}{pxl} \quad [mm] \quad (1.1)$$

S pomocí velikosti projekce můžeme vypočítat úhel β dle rovnice 1.2, kde f je ohnisková vzdálenost čočky.[2]

$$\beta = \arctg\left(\frac{\frac{c}{2} - a}{f}\right) + 90^\circ \quad [^\circ] \quad (1.2)$$

Výsledná vzdálenost l může být poté vypočtena pomocí rovnice 1.3 kde b je délka triangulační báze.[2]

$$l = \frac{b * \sin(\beta)}{\sin(180^\circ - (\alpha + \beta))} \quad [mm] \quad (1.3)$$

V reálném prostředí může docházet k míchání různých světelných paprsků, které vyzařují jiné kamery v okolí nebo slunce. Tato šumová informace negativně ovlivňuje kvalitu naměřených výsledků. Metoda aktivní triangulace je tudíž citlivá na vlivy okolního prostředí a různost povrchů snímaného materiálu. V praxi se tato metoda nejčastěji využívá ve vnitřních prostorech v aplikacích jako je například kontrola kvality výrobků na páse, kde je možné odstranit vnější vlivy a vytvořit si tak co nejlepší podmínky k osvětlování výrobků a reflexi paprsků.[3]

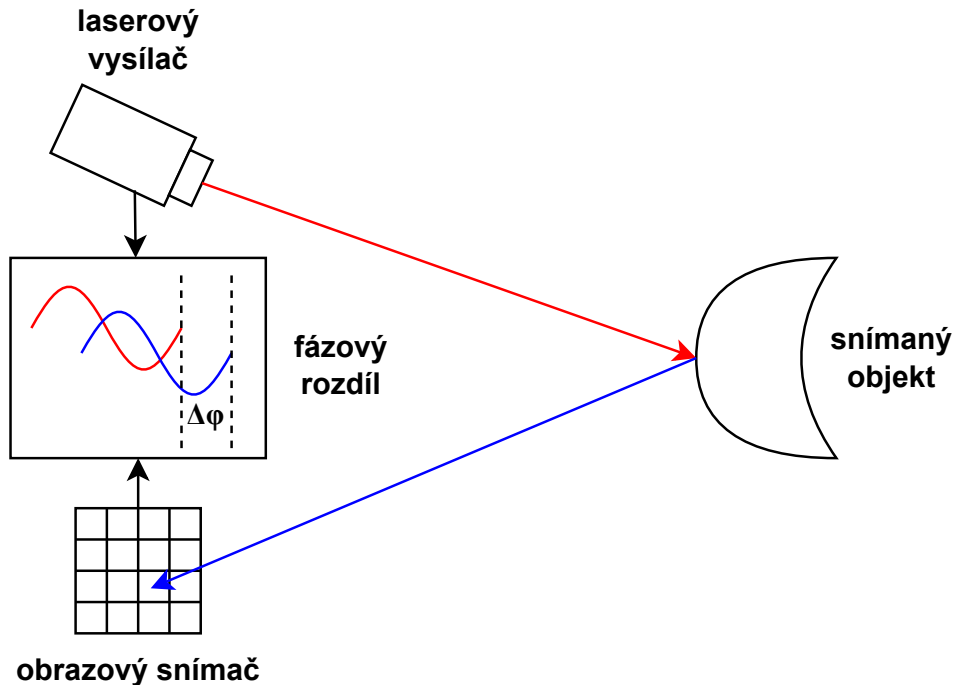
Důležité je také nastavení světelného zdroje tak, aby byla osvětlena co největší část snímaného objektu. Vzhledem k tvarům objektů může docházet k tomu, že některá místa budou zastíněna a světelný pruh se na ně nepromítne. Kameru je poté nutné správně umístit tak, aby snímala všechna místa co jsou osvětlena a nesnímala neosvětlená místa, aby bylo možné využít plné rozlišení kamery. Neosvětlená místa není možné kamerou rozpoznat.[3]

1.2 Měření doby letu

Metoda měření doby letu (anglicky Time of flight method) je metoda založená na určování vzdálenosti objektu za pomoci známé rychlosti světla. Měřicí systém obsahuje laser, který vyšle paprsek světla na měřený objekt a obrazový snímač, který zachytí odražený paprsek od objektu (viz Obr. 1.3). Systém poté vypočítá rozdíl fáze vyslaného a přijatého paprsku. Díky této hodnotě je možné vypočítat vzdálenost předmětu podle rovnice (1.4):

$$l = \frac{c}{2} \cdot \frac{\Delta\varphi}{2\pi f} \quad [m] \quad (1.4)$$

kde c je rychlost světla, $\Delta\varphi$ je rozdíl fázového posunu a f je frekvence signálu. Důležité je také zmínit, že vysílač i přijímač jsou většinou zabudovány v jednom pouzdře.[4]



Obr. 1.3: Schéma systému měření doby letu

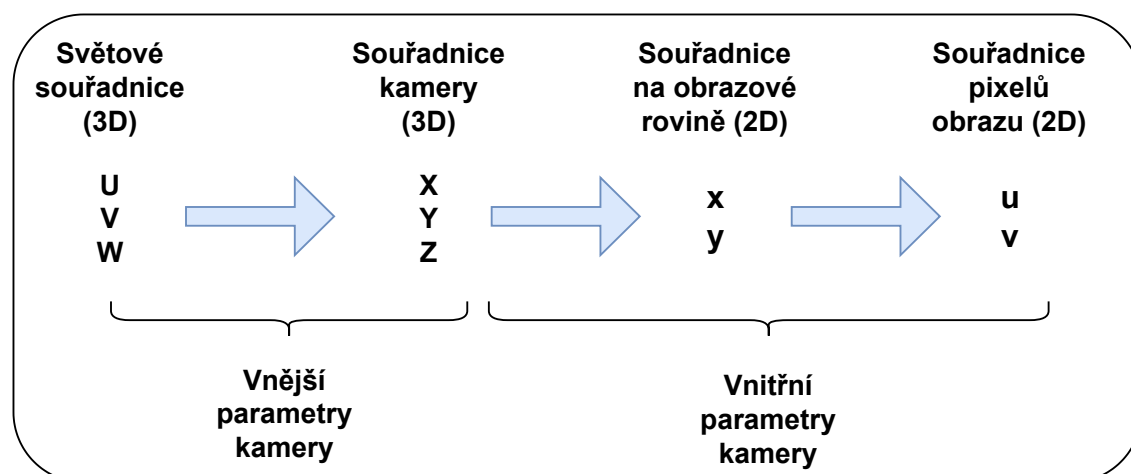
Vzhledem k tomu, že rychlost světla je rovna přibližně 300 000 km/s je možné tuto metodu využít na větší vzdálenosti. Tato rychlost ovšem klade velké nároky na přesnost celého systému, což je jeho velikou nevýhodou. Stejně jako v případě metody aktivní triangulace může přesnost měření ovlivňovat infračervené světlo vyzařované z další kamery v okolí nebo jiný zdroj infračerveného záření jako například sluneční paprsky. V případě objektů nepravidelných tvarů může také docházet k odražení paprsků do různých směrů mimo snímač.[1]

1.3 Ostatní metody

Další metodou měření 3D objektů je například metoda optické interferometrie. Při této metodě je vyslaný paprsek světla rozdělen na dvě části, které putují po jiných drahách. Referenční paprsek se odrazí od zrcadla zpět do snímače a jeho dráha je známa. Druhý paprsek se odrazí od snímaného předmětu. Výsledná vzdálenost je určena na základě rozdílu fází odražených paprsků.[1]

2 Matematický popis snímání scény

Při snímání trojrozměrné scény (3D) dochází k transformaci souřadnic jednotlivých bodů obrazu do dvourozměrného snímku (2D). Jednotlivé snímané pixely prochází hned několika transformacemi (viz Obr. 2.1).



Obr. 2.1: Transformace souřadnic při snímání obrazu kamerou

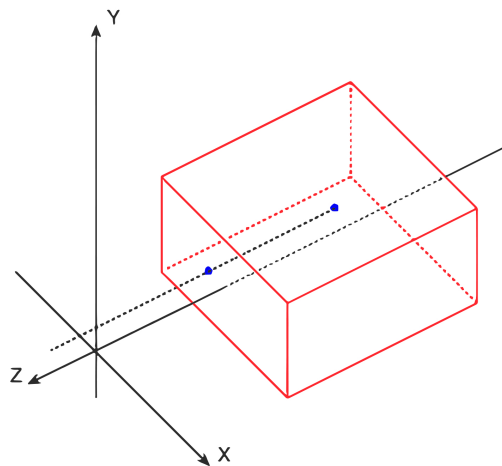
Matematický popis těchto transformací je nutné znát pokud chceme rekonstruovat 3D model objektu z jeho 2D snímku. V dalších kapitolách budeme tyto transformace mezi souřadnicovými systémy popisovat prostřednictvím vnějších a vnitřních parametrů kamery. Jako vnější parametry kamery se označuje poloha a orientace kamery v prostoru snímané scény. Pomocí těchto parametrů se přepočítává obraz ze světových souřadnic do souřadného systému kamery. Vnitřními parametry poté označujeme vlastnosti kamery, které transformují obraz z souřadnicového systému kamery na souřadnicový systém pixelů obrazu.[6]

2.1 Zobrazovací projekce

Projekce udávají jakým způsobem proběhne transformace bodů 3D scény do 2D obrazu. Typů projekcí existuje hned několik z nichž nejpoužívanější je projekce perspektivní a poté projekce ortografická nebo také paralelní. Při projekci používáme takzvanou projekční matici, kterou násobíme body 3D scény. [5]

2.1.1 Ortografická projekce

Nejjednodušším způsobem projekce je projekce ortografická. Při ortografické projekci je velikost všech objektů na scéně konstantní neohledně na vzdálenosti objektů od kamery. Všechny projekční přímky jsou zde rovnoběžné a střed projekce je tudíž v nekonečnu (viz Obr. 2.2).



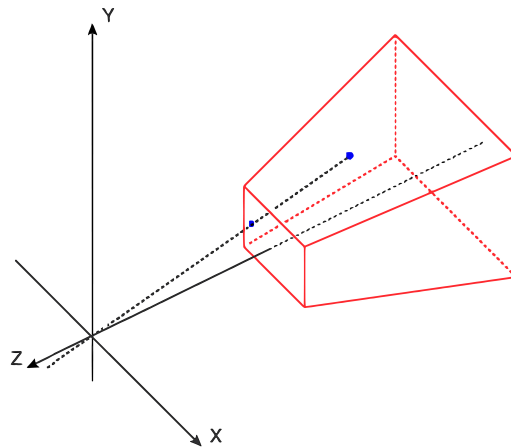
Obr. 2.2: Ortografická projekce

Ortografická projekce je aproximace modelu čočky s dlouhou ohniskovou vzdáleností (teleobjektivu). Tato projekce se používá u objektů, jejichž hloubka je zanedbatelná vzhledem k jejich vzdálenosti od objektivu (například pozorování objektů ve vesmíru).[6]

2.1.2 Perspektivní projekce

Nejběžněji používanou projekcí je projekce perspektivní. Tato projekce bere v úvahu perspektivu snímané scény. Objekty, které jsou blíže k objektivu kamery vypadají větší, než objekty které jsou od objektivu dále. Promítané body jsou děleny velikostí jejich z složky. U perspektivní projekce jsou projekční přímky různoběžkami (viz Obr. 2.3).

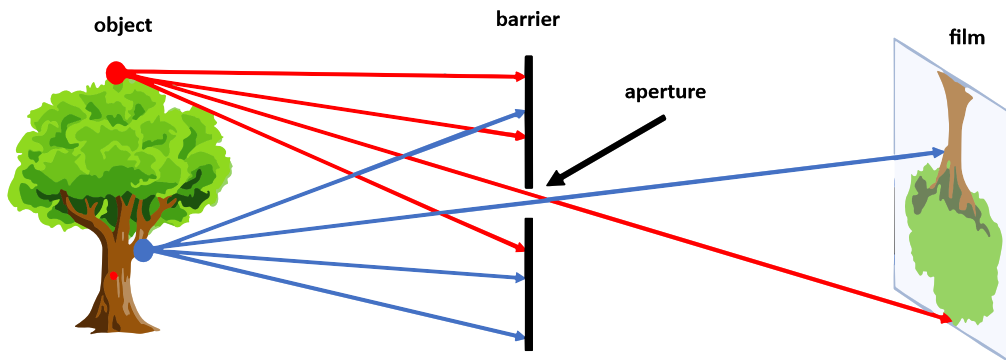
Perspektivní projekce popisuje funkci snímání obrazu pomocí modelu dírkové kamery nebo modelu tenké čočky. Tato projekce se používá v počítačové grafice nebo v úlohách počítačového vidění.[6]



Obr. 2.3: Perspektivní projekce

2.2 Model dírkové kamery

Model dírkové kamery (pinhole camera) je zjednodušený model kamery, který využívá transformaci pomocí perspektivní projekce. Model dírkové kamery funguje tak, že mezi snímaný objekt a sensor kamery postavíme bariéru, kterou neprojde světlo. Tato bariéra obsahuje jedinou mezeru v bodě O , která se nachází ve středu bariéry. Touto mezerou můžou procházet světelné paprsky vyzářené z každého bodu snímaného objektu (viz Obr. 2.4).[7]



Obr. 2.4: Model dírkové kamery [7]

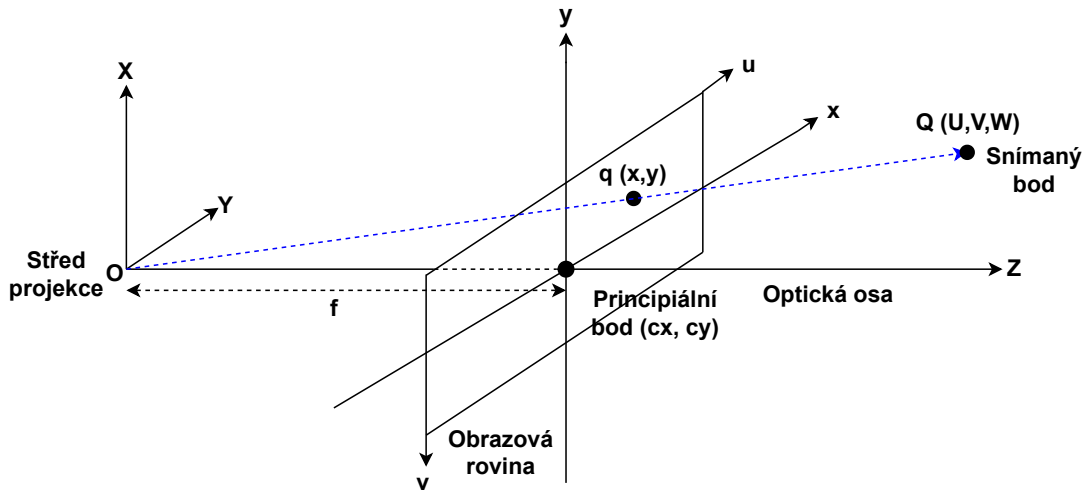
Pokud by mezi senzorem a objektem nebyla bariéra, tak by výsledný obraz byl směsí všech možných barev, protože paprsky, které vyzařuje snímaný objekt se šíří do všech směrů. Dále tedy přichází v úvahu, co nejvíce zmenšit mezeru uprostřed bariéry. Tím ovšem docílíme průchodu méně světelných paprsků a výsledný obraz tak

bude sice ostřejší, ale tmavší a expoziční doba bude delší. Dále také může docházet k efektu difrakce. Řešením tohoto problému je použití optických čoček tak, abychom směřovali světelné paprsky do požadovaného bodu.[8]

Dírkový model kamery se používá jako zjednodušující model při zjišťování vnitřních parametrů kamery.

2.3 Vnitřní parametry kamery

Vnitřní parametry kamery neboli intrinsické parametry definuje vnitřní uspořádání kamery. Pomocí těchto parametrů přepočítáváme jednotlivé body obrazu ze souřadnicového systému kamery (X, Y, Z) do souřadnicového systému obrazu (u, v) . Následující obrázek popisuje perspektivní projekci snímaného bodu pomocí modelu dírkové kamery na obrazovou rovinu (snímač) (viz Obr. 2.5).



Obr. 2.5: Promítnutí pozorovaného bodu scény

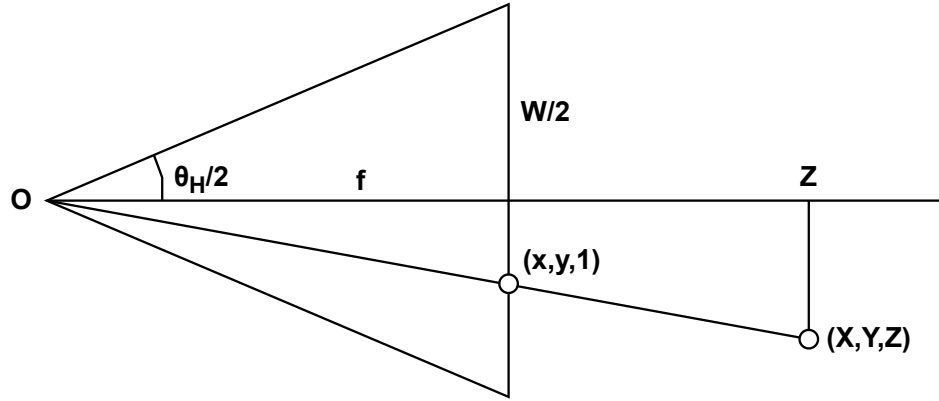
Vzdálenost f mezi obrazovou rovinou a středem projekce O se nazývá ohnisková vzdálenost. Ta společně se souřadnicemi principiálního bodu cx a cy tvoří matici vnitřních parametrů K . [9]

$$K = \begin{bmatrix} fx & \alpha & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

V matici se také někdy může vyskytovat parametr α (osové zkreslení), který udává nepřesnost v kolmosti obou sensorových os. U moderních kamer jsou ovšem

použité senzory dostatečně přesné na to, abychom mohli tento parametr zanedbat.[9]

V momentě kdy stanovíme počátek přibližně do středu obrazu (cx, cy) můžeme uvažovat jednotnou ohniskovou vzdálenost 2.2. Tento vztah vyjadřuje obrázek 2.6.



Obr. 2.6: Vztah mezi ohniskovou vzdáleností a horizontálním zorným úhlem [6]

$$K = \begin{bmatrix} fx & \alpha & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} f & 0 & cx \\ 0 & f & cy \\ 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

Ohniskovou vzdálenost můžeme poté vypočítat jako:

$$\tan\left(\frac{\theta_H}{2}\right) = \frac{W}{2f} \Rightarrow f = \frac{W}{2} [\tan\left(\frac{\theta_H}{2}\right)]^{-1} \quad (2.3)$$

kde W je šířka snímače a θ_H horizontální zorný úhel kamery.[6]

2.3.1 Zkreslení snímků

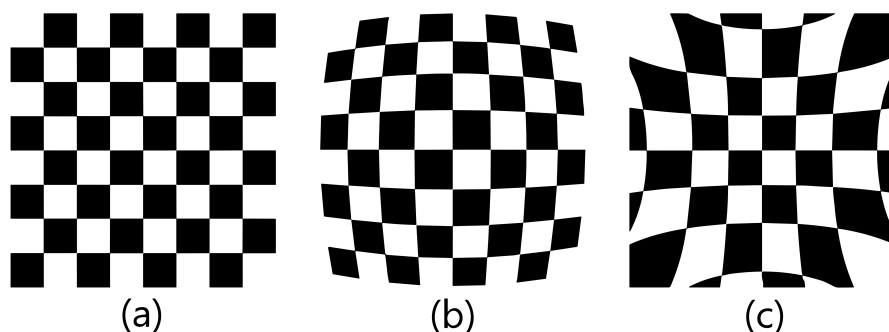
Mezi vnitřní parametry kamery kromě matice K ještě řadíme distorzi, neboli zakřivení kamery. Tento jev vzniká proto, že v reálných kamerách používáme systém různě zakřivených čoček. Tyto čočky mají nedokonalosti, v jejichž důsledku dochází ke zkreslení obrazu.(viz Obr. 2.7) Zkreslení se většinou nejintenzivněji projevuje na krajích snímku. Rozlišujeme mezi dvěma hlavními typy zkreslení a to radiálním a tangenciálním. U běžných čoček se hlavně projevuje zkreslení radiální a tangenciální zkreslení je většinou zanedbáváno.[9] Tento jev se řeší při kalibraci kamery získáním radiálních koeficientů distorze a přepočtením bodů zkreslených distorzí (viz 2.4):

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \hat{u}(1 + k_1r^2 + k_2r^4 + k_3r^6) \\ \hat{v}(1 + k_1r^2 + k_2r^4 + k_3r^6) \end{bmatrix} \quad (2.4)$$

kde k_1 , k_2 a k_3 jsou koeficienty radiální distorze a \hat{u} a \hat{v} jsou body snímku ovlivněné distorzí. Symbol r z předchozí rovnice poté vyjádříme jako:

$$r = \sqrt{(\hat{u} - c_x)^2 + (\hat{v} - c_y)^2} \quad (2.5)$$

kde c_x a c_y jsou souřadnice principiálního bodu.



Obr. 2.7: (a)Obrázek bez zkreslení, (b)Pozitivní radiální zkreslení ($k_1 > 0$), (c) Negativní radiální zkreslení ($k_1 < 0$) [10]

2.4 Vnější parametry kamery

Matice vnějších parametrů kamery neboli extrinsická matice nám popisuje vztah mezi světovými souřadnicemi snímaného objektu a souřadnicemi kamery v 3D prostoru. Tato matice obsahuje rotační a translační vektor, které udávají pozici a natočení kamery vůči světovému souřadnicovému systému. [6]

Matice má tvar:

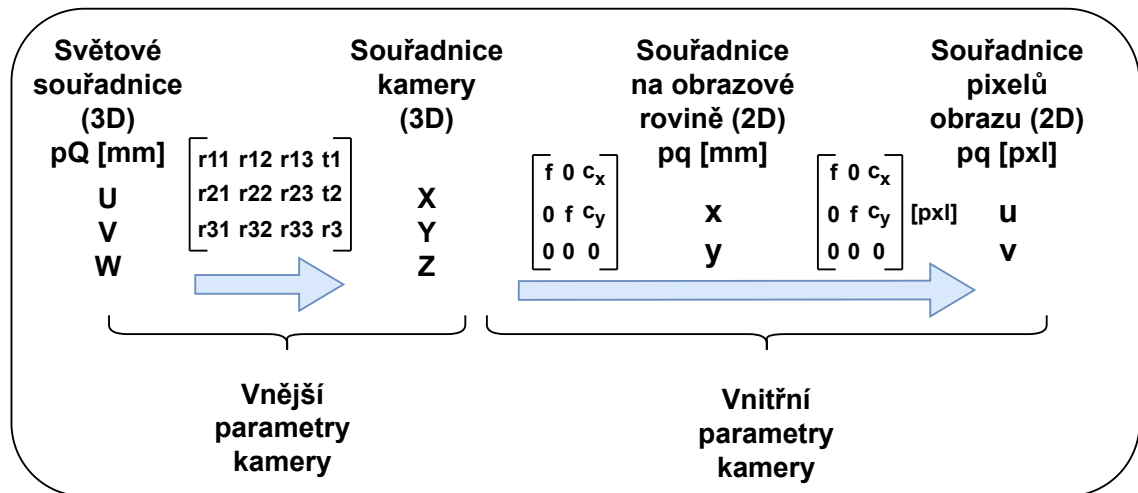
$$R_T = [R|t] * pQ \Rightarrow R_T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix} \quad (2.6)$$

kde R je rotační vektor, T je translační vektor a pQ je pozorovaný bod ve světových souřadnicích. Výsledný přepočítání mezi souřadnicovými systémy dle obrázku 2.1 lze poté modelovat takto:

$$P * pq = K[R|t] * pQ \quad (2.7)$$

$$P \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & cx \\ 0 & f & cy \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix} \quad (2.8)$$

kde pQ je pozorovaný bod ve světových souřadnicích a pq je bod souřadnice pixelu obrazu. Situaci popisuje obrázek 2.8. Pomocí extrinsické matice kamery získáme bod v souřadnicovém systému kamery a poté s pomocí intrinsické matice kamery získáme bod na obrazové rovině v mm nebo bod určitého pixelu snímače. Tento rozdíl závisí na použitých jednotkách v intrinsické matici kamery.



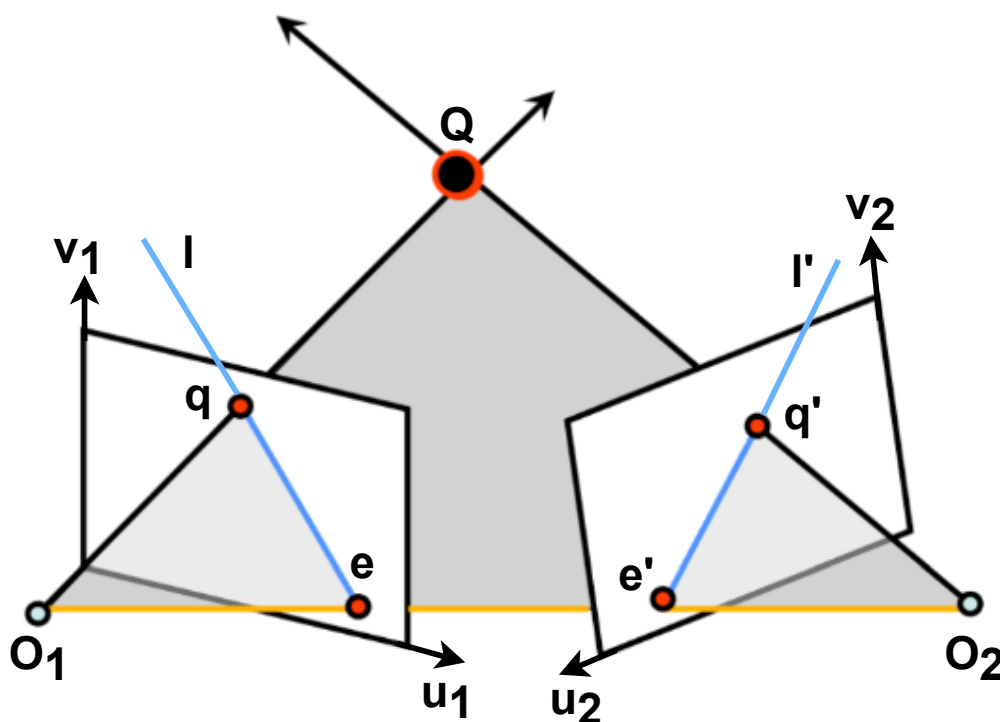
Obr. 2.8: Transformace souřadnic při snímání obrazu kamerou

2.5 Stereo systém

V předchozích kapitolách byla popsána vnitřní a vnější matice v případě použití jedné kamery a tudíž získání jednoho snímku scény. Pro rekonstrukci 3D modelu scény jsou ovšem zapotřebí minimálně dva snímky a tudíž stereo systém dvou a více kamer (dále budeme uvažovat stereo systém se dvěma kamerami). Po získání výše popsaných matic u obou kamer je nutné nalézt vztah mezi stejnými body v obou vyfocených snímcích tak, abychom byli schopni získat hloubkovou souřadnici scény, kterou potřebujeme pro vytvoření 3D modelu. Důležitou podmínkou stereo systému je překrytí pohledů obou kamer tak, aby se snímáný objekt nacházel na obou kamerách zároveň. Vztah mezi korespondujícími body ve dvou snímcích popisuje epipolární geometrie.

2.6 Epipolární geometrie

Obrázek 2.9 ilustruje vztah mezi dvěma kamerami a sledovaným bodem Q . Projekce sledovaného bodu Q se zobrazuje na obrazových rovinách levé a pravé kamery v bodech q a q' . Středů projekce kamer se nachází v bodech O_1 a O_2 a přímka, která je spojuje se nazývá báze. Plocha mezi bodem Q a středů kamer se nazývá epipolární rovina. Místa, kde báze protíná obrazové roviny kamer e a e' se nazývají epipóly. Přímky, které protínají průniky obrazových rovin s epipolární rovinou l a l' se nazývají epipolární přímky nebo také epipoláry. Tyto přímky protínají bázi v epipólech.[11]



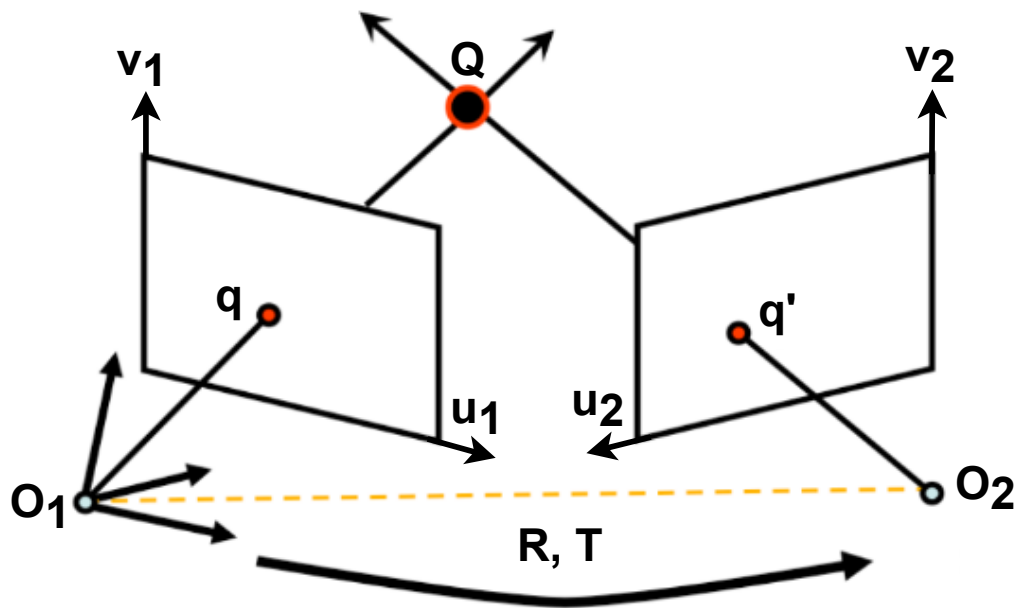
Obr. 2.9: Epipolární geometrie [11]

Speciální případ epipolární geometrie nastává v případě, že jsou obě kamery umístěny paralelně vedle sebe (obrazové roviny jsou paralelní). Báze je poté paralelní k těmto obrazovým rovinám a epipolární přímky ji tudíž nikdy neprotnou. Epipóly se tak nacházejí v nekonečnu. Toto uspořádání se nazývá rektifikovaný pár a v některých případech může být výhodné pro další výpočty.[11]

Známymi parametry z předchozích kapitol jsou rotace a translace kamer, matice vnitřních parametrů obou kamer a obrazový bod na jedné obrazové rovině kamery. Se znalostí středů kamer O_1 , O_2 a bodu q jsme schopni definovat plochu epipolární

roviny. Za pomoci epipolární roviny jsme poté schopni vyjádřit epipolární přímky. Se znalostí epipolárních přímek poté můžeme říci, že projekce bodu Q na obrazovou rovinu pravé kamery q' musí nastat v bodech, které leží na epipolární přímce této obrazové roviny. Stejná logika se dá aplikovat i v obráceném případě, kdy budeme vycházet z bodu q' . [11]

V předchozí kapitole 2.4 jsme si definovali matici P , která mapuje body 3D scény ze světového souřadnicového systému do odpovídajících pixelů 2D snímku. Uvažujme nyní, že světový souřadnicový systém je spjat s levou kamerou a pravá kamera je od ní vzdálena o posun T a natočena o rotaci R (viz Obr. 2.10). [11]



Obr. 2.10: Projekce na souřadnicový systém levé kamery [11]

Poté můžeme projekční matice kamer definovat takto:

$$M = K \begin{bmatrix} I & 0 \end{bmatrix} \quad M' = K' \begin{bmatrix} R^T & -R^T T \end{bmatrix} \quad (2.9)$$

V případě, kdy budeme uvažovat, že obě kamery jsou totožné a tudíž $K = K' = I$ můžeme tyto matice vyjádřit takto:

$$M = \begin{bmatrix} I & 0 \end{bmatrix} \quad M' = \begin{bmatrix} R^T & -R^T T \end{bmatrix} \quad (2.10)$$

Lokace bodu q' vůči referenčnímu systému levé kamery je poté $Rq' + T$. Vektory $Rq' + T$ a T leží ve stejné epipolární rovině. Vektorovým součinem $T \times (Rq' + T) = T \times (Rq')$ dostaneme normálový vektor k epipolární rovině. Bod q je poté také normálový k $T \times (Rq')$ a jejich skalární součin je tudíž nulový [11]:

$$q^T \cdot [T \times (Rq')] = 0 \quad (2.11)$$

Vektorový součin můžeme převést na maticově vektorový součin dle:

$$T \times R = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix} = [T_x]R \quad (2.12)$$

a dostaneme výraz $q^T [T_x]Rq' = 0$ kde $[T_x]R = E$ a tudíž:

$$q^T E q' = 0 \quad (2.13)$$

kde E je esenciální matice. Ze souřadnic bodu q a této matice jsme schopni vyjádřit epipolární přímku v pravé obrazové rovině a naopak. Esenciální matice ovšem bere v úvahu pouze vnější parametry kamer (rotaci a translaci).[11]

V reálném případě kamery nejsou plně identické a tudíž musíme vzít v úvahu jejich matice vnitřních parametrů K a K' . Vyjdeme z rovnice 2.9. Projekce bodu Q do levé a pravé obrazové roviny definujeme jako $q_c = K^{-1}q$ a $q'_c = K'^{-1}q'$. Poté můžeme přepsat tvar 2.13 jako[11]:

$$q_c^T [T_x]Rq'_c = 0 \quad (2.14)$$

a po dosazení dostaneme rovnici:

$$q^T K^{-T} [T_x]R K'^{-1} q' = 0 \quad (2.15)$$

$K^{-T} [T_x]R K'^{-1} = F$ a proto můžeme výraz takto substituuovat:

$$q^T F q' = 0 \quad (2.16)$$

kde F značí Fundamentální matici. Fundamentální matice má 7 stupňů volnosti narozdíl od esenciální matice, která jich má pouze 5. Tato matice bere v úvahu nejen vnější parametry kamer (rotaci a translaci) ale také vnitřní parametry kamer definované podle jejich matic K a K' . Podobně jako u esenciální matice můžeme pomocí této rovnice vypočítat epipolární přímku pravé obrazové roviny z projekce bodu q snímaného bodu Q v levé obrazové rovině a naopak[11]:

$$l' = F^T q \quad \text{nebo} \quad l = F q' \quad (2.17)$$

2.7 Způsoby reprezentace 3D modelu

V této kapitole budou rozebrány jednotlivé způsoby reprezentace 3D modelu.

2.7.1 Mračno bodů

Jednou z možností jak reprezentovat 3D model je mračno bodů. Mračno bodů je v podstatě shluk bodů, který se nachází v definovaném souřadnicovém systému. V případě 3D modelu je tedy každý bod definován vektorem se souřadnicemi X, Y a Z v příslušném souřadnicovém systému. Mračna bodů se dále dají rozdělit na organizovaná a náhodná. V případě organizovaných mračen bodů jsou vektory, které definují tyto body logicky seřazeny ve zvoleném pořadí podle některé souřadnice. V případě náhodných mračen bodů jsou tyto vektory reprezentující jednotlivé body seřazeny v náhodném pořadí. Pomocí množství těchto bodů jsme schopni definovat tvary objektu a vytvořit tak jeho reprezentaci. Jednotlivé body v mračcích mohou být barevné i černobílé.[12]

2.7.2 Polygonová síť

Další možností jak reprezentovat model je polygonová síť. Polygonová síť se skládá z ploch polygonů a jejich vrcholů, ve kterých se polygony spojují. Takováto síť vhodně uspořádaných polygonů vytváří a popisuje tvar 3D modelu. Objekt je tedy reprezentován vnějším pláštěm, který definuje jeho rozměry.[13]

2.7.3 Hloubková mapa

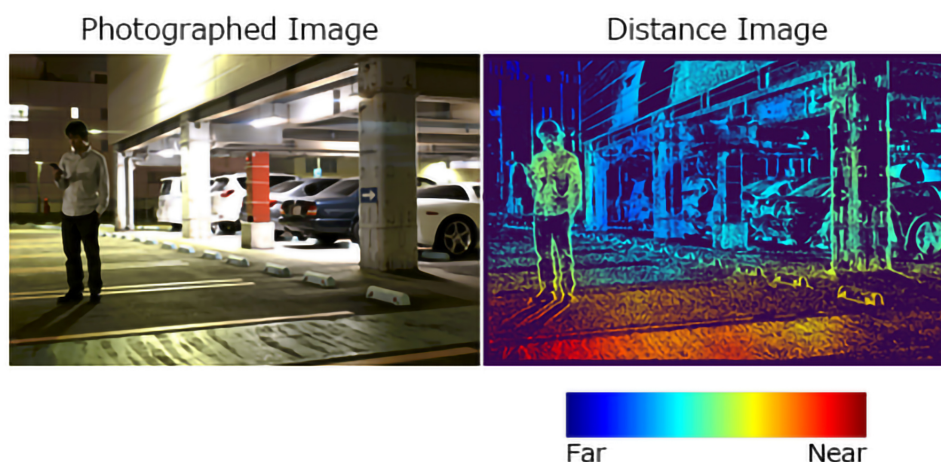
Následující možností reprezentace 3D modelu je hloubková mapa. Hloubková mapa je 2D snímek obsahující hloubkovou informaci objektu. Informace o vzdálenosti jednotlivých bodů je ve snímku zakódována pomocí intenzity barevných přechodů mezi jednotlivými úrovněmi. Například v případě šedotónových snímků jsou body, které se nacházejí blíže ke kameře vyobrazeny světlejšími odstíny než body, které jsou od kamery vzdálenější. Barevné podání může být černobílé nebo například modročervené atd. K hloubkovému snímku je většinou přiřazen ještě barevný RGB snímek a potom se takový snímek označuje jako RGB-D snímek.[13]

2.7.4 Shrnutí

Mezi jednotlivými výše zmíněnými reprezentacemi 3D modelu je možné přecházet pomocí různých algoritmů. Z hloubkové mapy je tak možné vytvořit mračno bodů a z mračna bodů je možné vytvořit polygonovou síť. Vhodnost jednotlivých reprezentací se odvíjí od typu aplikace a požadovaných výsledků.

3 Návrh systému zpracování stereo snímků

V následující kapitole bude popsán návrh stereo systému, který bude využit pro 3D rekonstrukci snímaných objektů. Návrh porovnává různé kombinace dvou běžně dostupných hardwarových prvků, kterými jsou RGB kamery a hloubkové kamery. Výstupem RGB kamer je barevný snímek scény. U hloubkových kamer jsou většinou výstupy dva a to barevný snímek scény a hloubkový snímek udávající vzdálenost jednotlivých bodů od kamery (viz Obr. 3.1). Tyto kamery se také někdy označují jako RGB-D kamery.

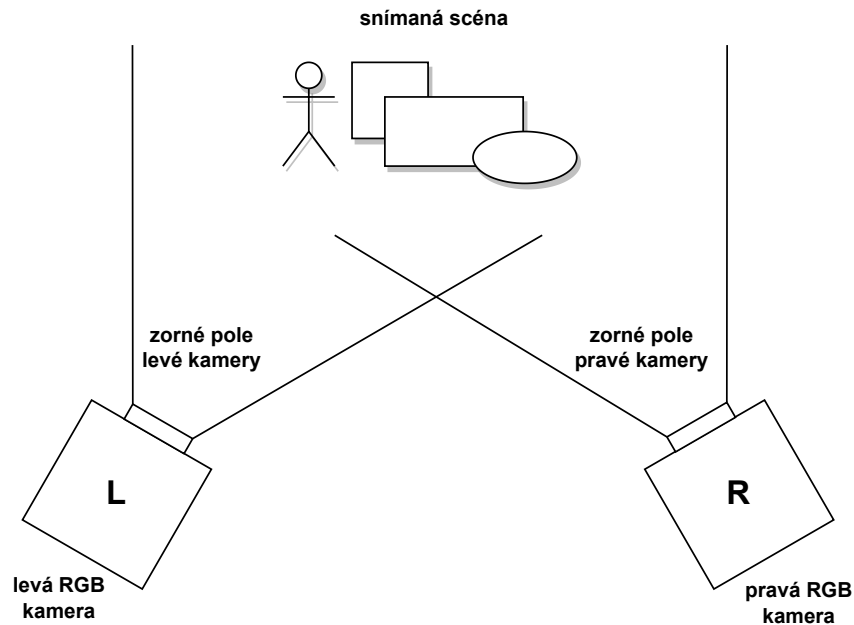


Obr. 3.1: Barevný a hloubkový snímek scény [14]

3.1 Systém obsahující stereo pár kamer

První možností, jak navrhnout tento systém je použití dvou RGB kamer jako stereo páru. Tímto způsobem jsme schopni pomocí metody triangulace získat hloubkovou informaci pro každý pár korespondujících bodů, které ve snímcích nalezneme. Zároveň máme k dispozici i barevný snímek, který můžeme použít později na vytvoření textury 3D modelu. Výhodou tohoto systému je, že nemusíme vyzařovat žádný paprsek k tomu, abychom získali hloubkovou informaci (pasivní triangulace). Většinou také RGB kamery disponují větším rozlišením než 3D kamery. Problémů u tohoto řešení ovšem vyvstává hned několik. Každý bod snímaného objektu u kterého chceme zjistit hloubkovou informaci musí být viditelný na obou kamerách současně a tyto korespondující body musí být poté nalezeny v obou snímcích (viz Obr. 3.2). Tato

podmínka limituje velikost úhlů zorného pole, které můžou obě kamery současně zabírat a snímání větších objektů může být tudíž náročné. Algoritmy, které hledají korespondující body v obou snímcích jsou časově náročné a jejich výsledky mohou být zhoršeny zastíněním některých objektů jinými objekty na jedné z kamer. Tyto zastíněné body se na výsledném 3D modelu projeví prázdnými místy, která je poté potřeba vyplnit pomocí dalších algoritmů.[15]



Obr. 3.2: Stereo systém dvou kamer

3.2 Systém obsahující 3D kameru

Druhou možností návrhu systému je použití samostatné hloubkové kamery. Výstupem tohoto systému je hloubková mapa snímané scény a v případě, že hloubková kamera obsahuje i RGB kameru (RGB-D) máme k dispozici barevný snímek. Výhodou tohoto systému je znalost hloubkové informace přímo z kamery. Díky tomu nemusíme aplikovat časově a výpočetně náročné algoritmy. Dále také nemusíme hlídat překrytí zorných úhlů více kamer a využijeme tak celé rozlišení snímače.

Většina současných RGB-D kamer je založena na principu měření doby letu nebo aktivní triangulace, kde kameru tvoří zdroj záření, který promítá vzor na snímaný objekt a jedna nebo více barevných kamer.

3.2.1 Hloubkové kamery založené na principu doby letu

Hloubkové kamery založené na principu měření doby letu mají pouze jeden úhel pohledu a tudíž jsou méně náchylné na zastínění objektů scény. Kamera musí být schopna rozpoznat časový rozdíl mezi vysláním a přijetím paprsku, který putuje rychlostí světla. Tato změna musí být zaznamenatelná na úrovni jednotlivých pixelů, což vyžaduje vysokou citlivost snímače a proto mají tyto kamery obecně menší senzorové rozlišení. Dále také může vznikat rušení v důsledku vnějšího záření, které ovlivňuje přesnost hloubkového měření.[12]

3.2.2 Hloubkové kamery založené na principu aktivní triangulace

U kamer založeném na principu aktivní triangulace používáme běžné plošné kamery a v důsledku toho mají tyto kamery větší rozlišení než kamery založené na principu doby letu. Nevýhodami jsou ovšem menší snímková frekvence, náchylnost na vnější zdroje záření, které může rušit promítaný vzor a také náchylnost na zastínění objektů. U těchto kamer je také obtížné detekovat hrany objektů.[12]

3.2.3 Zhodnocení systému s jednou hloubkovou kamerou

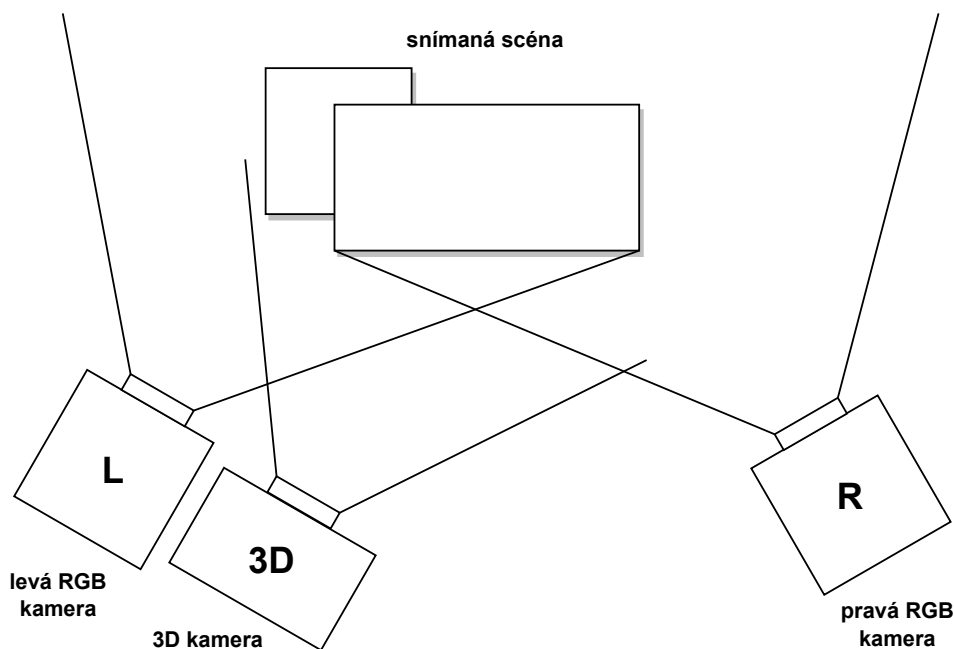
Obecnou nevýhodou hloubkových kamer je menší rozlišení výsledných snímků než u RGB kamer. Tento nedostatek vede k nízké hustotě bodů výsledného 3D modelu a tudíž je tento systém nevhodný pro aplikace vyžadující vysoké rozlišení výsledného modelu. U obou typů kamer popsaných v sekcích výše se vyskytují nevýhody, které limitují jejich samostatné použití. Snahou této práce je navrhnout systém kombinující vícero kamer, čímž by mohlo dojít k odstranění některých těchto nevýhod.

3.3 Systém obsahující 3D a RGB kameru

Další možností, která se v tomto návrhu nabízí je zkombinovat hloubkovou kameru s jednou RGB kamerou. V tomto případě máme k dispozici jednu hloubkovou informaci z 3D kamery a jeden barevný snímek z RGB kamery. Data z barevného snímku je možné použít pro interpolaci nebo filtraci dat z hloubkové kamery. Běžným postupem je předzpracování hloubkových dat a následné promítnutí těchto dat na barevný snímek. Z barevného snímku lze lépe rozpoznat hrany a textury jednotlivých objektů. Problém, který zde ovšem stále zůstává je možné zastínění některých objektů jinými objekty na jedné z těchto kamer a s tím související prázdná místa v textuře výsledného modelu. Tuto chybu se snažíme co nejvíce zmenšit tím, že obě kamery dáme co nejbližší k sobě tak, aby sledovali objekt ze stejného místa.[9]

3.4 Systém s 3D kamerou a stereo párem RGB kamer

Posledním a nejkompexnějším systémem, který se nabízí je kombinace stereo páru dvou RGB kamer a jedné hloubkové kamery (viz Obr. 3.3). Výstupem tohoto systému mohou být dvě hloubkové informace a dva barevné snímky. V tomto případě se nejdříve vypočítá hloubková informace pomocí pasivní triangulace ze stereo páru kamer. Poté se hloubkové informace ze stereo páru i z hloubkové kamery promítnou na společný referenční systém. V posledním kroku se tyto informace spojí a vytvoří finální hloubkovou mapu. Výsledný 3D model má poté vysokou hustotu bodů. Další výhodou tohoto systému je, že hloubková i barevná informace o scéně pochází ze dvou různých pohledů. Je velice pravděpodobné, že hloubková mapa ze stereo páru bude obsahovat prázdná místa, jakožto místa, která viděla pouze jedna kamera. Hloubková kamera se ovšem dívá z jiného úhlu pohledu a tudíž nám některá místa může vyplnit. To samé platí při aplikaci barevné textury na výsledný 3D model.[9]



Obr. 3.3: Systém s 3D kamerou a stereo párem RGB kamer

První možností, kam umístit hloubkovou kameru je doprostřed stereo páru. Existuje zde ovšem i lepší varianta a to umístění 3D kamery na jednu stranu co nejbližší k jedné z RGB kamer stereo páru. Díky tomuto rozložení je systém ještě více odolný proti zastínění některých objektů. Dále tím také zlepšíme reprojekční chybu, což je rozdíl vzdálenosti obrazu mezi promítnutým bodem a změřeným bodem. Tato chyba vzniká v důsledku nepřesné kalibrace zvláště u 3D kamery.[9]

4 Volba hardwarových prostředků

Následující kapitola obsahuje rešerši dostupných kamer na trhu a výběr jednotlivých kamer podle jejich parametrů. V první sekci jsou rozebrány dostupné hloubkové kamery. Další kapitola poté obsahuje výběr RGB kamery a v poslední kapitole je popsána konstrukce stereo systému se zvolenými prvky.

4.1 Výběr RGB-D kamery

Návrh stereo systému se odvíjí od zvolené hloubkové kamery, jejichž výběr na trhu je poměrně úzký. Většina výrobců už vývoj a prodej ukončila nebo se úzce specializují na jedno odvětví.

4.1.1 Kinect

Asi nejznámějším představitelem hloubkových kamer je Kinect od Microsoftu. Tyto kamery se začali vyvíjet a prodávat hlavně pro účely herních konzolí Xbox. Vytvořeny byly dvě verze a to Kinect V1, který fungoval na principu aktivní triangulace a později Kinect V2, který fungoval na principu měření doby letu.

První verze obsahovala RGB senzor s rozlišením 640x480 a infračervený senzor s rozlišením 320x240. Snímková frekvence byla 30 snímků za sekundu a maximální udávaný dosah okolo 4m. První verze Kinectu byla ovšem spíše pro vnitřní použití z důvodu infračervených paprsků slunce rušících senzor. Vývoj i prodej byl ovšem ukončen v roce 2016 a není tak již možné zakoupit novou kameru.[16]



KINECT V1 PRO XBOX 360



KINECT V2 PRO XBOX ONE

Obr. 4.1: Kinect V1 (vlevo) a Kinect V2 (vpravo) [17]

Druhá verze Kinectu byla ve většině ohledů lepší než verze první. Obsahovala RGB senzor s rozlišením 1920x1080 a infračervený senzor s rozlišením 512x424. Snímková frekvence byla taktéž 30 snímků za sekundu. Udávaný maximální dosah zde byl 4,5m. Senzor měl také širší zorné pole a mohl být použit i v aplikacích ve

venkovním prostředí. Vývoj i prodej této kamery byl ovšem také ukončen a tudíž už není možné novou kameru zakoupit. [16]

Tabulka 4.1 obsahuje srovnání parametrů obou verzí Kinectu

Tab. 4.1: Porovnání parametrů obou verzí Kinectu [16]

Parametr	Kinnect V1	Kinnect V2
Technologie	Aktivní triangulace	TOF
Rozlišení RGB senzoru	640x480, 30fps	1920x1080, 30fps
Rozlišení IR senzoru	320x240, 30fps	512x424, 30fps
Operativní vzdálenost	0.8m - 4m	0.4m - 4.5m
Zorné pole RGB senzoru	H:62 °, V:48.6 °	H:84.1 °, V:53.8 °
Zorné pole hloubkového senzoru	H:57 °, V:43 °	H:70 °, V:60 °

Oba Kinecty byly hojně využívány v různých vědeckých pracích a tudíž k nim existuje dobrá dokumentace. Navzdory těmto okolnostem byl vývoj i prodej obou modelů ukončen. V nově vyvíjených aplikacích tak již není vhodné tyto kamery používat.

4.1.2 Asus Xtion

Dalším výrobcem hloubkových kamer na trhu je Asus. Modelů Xtion bylo vyrobeno hned několik z nichž stojí za zmínku Asus Xtion Pro Live a Asus Xtion 2.

První z těchto dvou modelů je založen na principu aktivní triangulace a obsahuje hloubkový senzor s rozlišením 640x480 a RGB senzor s rozlišením 1280x1024. Snímková frekvence je 30 snímků za sekundu u obou senzorů. Udávaný maximální dosah je 3,5m. Horizontální zorné pole senzoru je 58 ° což je poměrně málo. Cenově se tato kamera pohybovala o trochu výše než Kinect. Stejně jako u předchozí verze Kinectu ji není možné použít ve venkovních prostorech. Vývoj i prodej tohoto modelu však byl již také ukončen.

Druhý model Asus Xtion 2 se od předchozí modelu moc nelišil. Stejně jako v případě Kinectu se senzorová technologie změnila na měření doby letu. Vylepšením bylo zvýšené rozlišení RGB kamery na 2592x1944 a větší horizontální zorný úhel senzoru. Stejně ovšem zůstalo rozlišení hloubkové kamery, což je hlavní nevýhodou toho senzoru. Další nevýhodou je také poměrně vysoká cena v porovnání s předchozím modelem.

Z důvodu výše popsaných nevýhod se ani jedna z těchto kamer nedočkala takové oblíbenosti jako v případě Kinectu. Vědeckých článků a obecně informací je k těmto

kamerám poměrně málo. První model již není dostupný a druhý model nenabídl žádné ohromné vylepšení. Výběr těchto kamer tudíž nepřípadá dále v úvahu.

4.1.3 Další výrobci

Na trhu se ještě nachází několik dalších drobnějších prodejců a kamer pro speciální použití.

Za zmínku stojí vývojový kit Microsoftu Azure Kinect DK vytvořený speciálně pro vědecké účely. Kamera funguje na principu měření doby letu a nabízí poměrně zajímavý rozsah rozlišení. Bohužel však stejně jako v případě herních Kinectů byl vývoj a prodej tohoto kitu v roce 2023 ukončen. Nový model již není možné oficiálně zakoupit.

Dalším výrobcem je Structure Sensor Pro. Tato malá kamera je speciálně vyvinuta pro lékařské účely. Funguje jako nástavec na mobilní telefony značka iPhone. Z hlediska vysoké ceny a specifické použitelnosti je ovšem tato kamera nevhodná k použití v této práci.

Dále na trhu existuje několik dalších modelů, které ovšem z důvodu svojí ceny nebo parametrů nejsou použitelné.

4.1.4 Intel RealSense

Poslední zmíněnou firmou na trhu je Intel s jeho kamerami řady RealSense D400. Na výběr je hned z několika modelů, které se rozlišují hlavně maximálním použitelným dosahem a filtrem infračerveného světla (řady s označením f). Všechny nabízené produkty jsou stále podporovány a mají podrobně vytvořenou dokumentaci. Intel také distribuuje vlastní sadu vývojových nástrojů, kde je možné kamery odzkoušet a nastavit různé možné parametry. Tyto kamery jsou tudíž pravděpodobně jedinou solidní možností, která je dnes na trhu dostupná.

Výběr kamery se odvíjel hlavně od maximální snímané vzdálenosti. Ta je v případě snímání vozidel jakožto velkých objektů poměrně důležitá. Řada D455 a D457/456 nabízejí širokoúhlý senzor, jehož maximální snímaná vzdálenost je 6m. Delší snímaná vzdálenost je způsobena prodloužením báze mezi levým a pravým stereo senzorem. Hlavním rozdílem v těchto dvou řadách je větší ochranný stupeň krytí v případě modelu D457 a speciální konektor GMSL. Cena je ovšem o něco vyšší a tudíž byl jako nejvhodnější kandidát zvolen model D455.

Kamera funguje na principu pasivní triangulace s přidaným infračerveným projektozem. Kamera tudíž není závislá na osvětlování snímaných objektů vzorem infračerveného světla jako v případě metody aktivní triangulace. Vnější infračervené signály, které by byly v případě aktivní triangulace považovány za rušivé jsou zde brány jako užitečné. Projektor infračerveného záření, který kamera obsahuje slouží

pouze k zlepšení světelných podmínek v případě nedostatečného osvětlení z vnějšího okolí. V případě metody pasivní triangulace na jejímž principu kamera funguje neovlivňují vnější infračervené signály měřená data. Proto je tato kamera vhodná k vnitřnímu i venkovnímu použití.[18]

Následující tabulka obsahuje parametry zvolené kamery 4.2.

Tab. 4.2: Parametry zvolené hloubkové kamery

Parametr	Intel RealSense D455
Technologie	Stereo Triangulace s IR projektorem
Rozlišení RGB senzoru	1280x800, 30fps
Rozlišení hloubkového senzoru	1280x720, 90fps
Operativní vzdálenost	0.6m - 6m
Zorné pole RGB senzoru	H:90°, V:65°
Zorné pole hloubkového senzoru	H:87°, V:58°
Ohnisková vzdálenost všech senzorů	1,93mm
Clonové číslo všech senzorů	f/2.0
Velikost senzoru	3896 μ m x 2453 μ m
Velikost pixelu senzoru	3 μ m x 3 μ m

Na obrázku 4.2 můžeme vidět zleva (pravý senzor stereo páru, RGB senzor, infračervený projektor a levý senzor stereo páru).



Obr. 4.2: RGB-D kamera Intel RealSense D455[19]

4.2 Výběr RGB kamery

Výběr další kamery se odvíjí od zvolené hloubkové kamery z předchozí kapitoly. Vybraná kamera již obsahuje jeden RGB snímač, který je možné použít jako jednu z kamer stereo páru. Tato RGB-D kamera tak v podstatě zapouzdřuje levou stranu

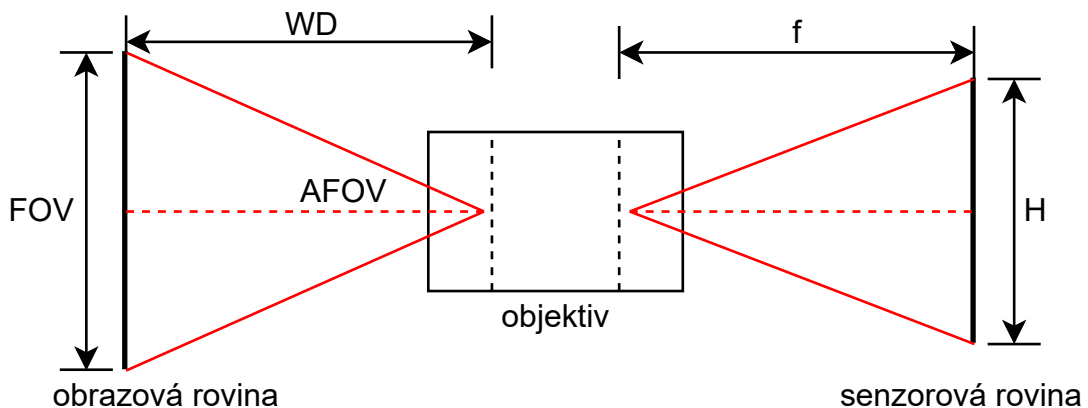
navrženého systému (levou kameru stereo páru a hloubkovou kameru) jak je patrné z obrázku 3.3. Je proto nutné vybrat ještě jednu RGB kameru, která bude tvořit výsledný stereo pár.

Požadované parametry budou víceméně odpovídat parametrům RGB senzoru z hloubkové kamery. V ideálním případě by měla mít kamera rozlišení 1280x720 tak, aby byly oba výsledné snímky stejně velké. Předpokládaný výskyt snímaného objektu bude v maximální vzdálenosti 5m, tak abychom využili rozsah hloubkové kamery. Zorné pole kamery by mělo být podobné a to horizontální úhel 90° a vertikální úhel 65° . Požadovaná snímková frekvence by měla být 30 snímků za sekundu tak, abychom měli stejný počet snímků pohybujícího se objektu. V případě, že se vozidlo bude pohybovat rychlostí 30 km/h tak získáme přibližně 15 snímků vozidla. Situace je znázorněna na obrázku 4.4. Snímač by měl být barevný, abychom mohli na výsledný model aplikovat barevnou texturu. Ohnisková vzdálenost byla vypočtena následovně dle vzorců:

$$FOV = 2 * WD * \tan\left(\frac{AFOV}{2}\right) \quad (4.1)$$

$$f = \left(\frac{H * WD}{FOV}\right) \quad (4.2)$$

kde FOV je šířka zorného pole, AFOV je úhel zorného pole, WD je pracovní vzdálenost snímaného objektu, H je horizontální velikost senzoru a f je vypočtená ohnisková vzdálenost. Situaci popisuje obrázek 4.3.



Obr. 4.3: Vztah mezi ohniskovou vzdáleností a zorným polem

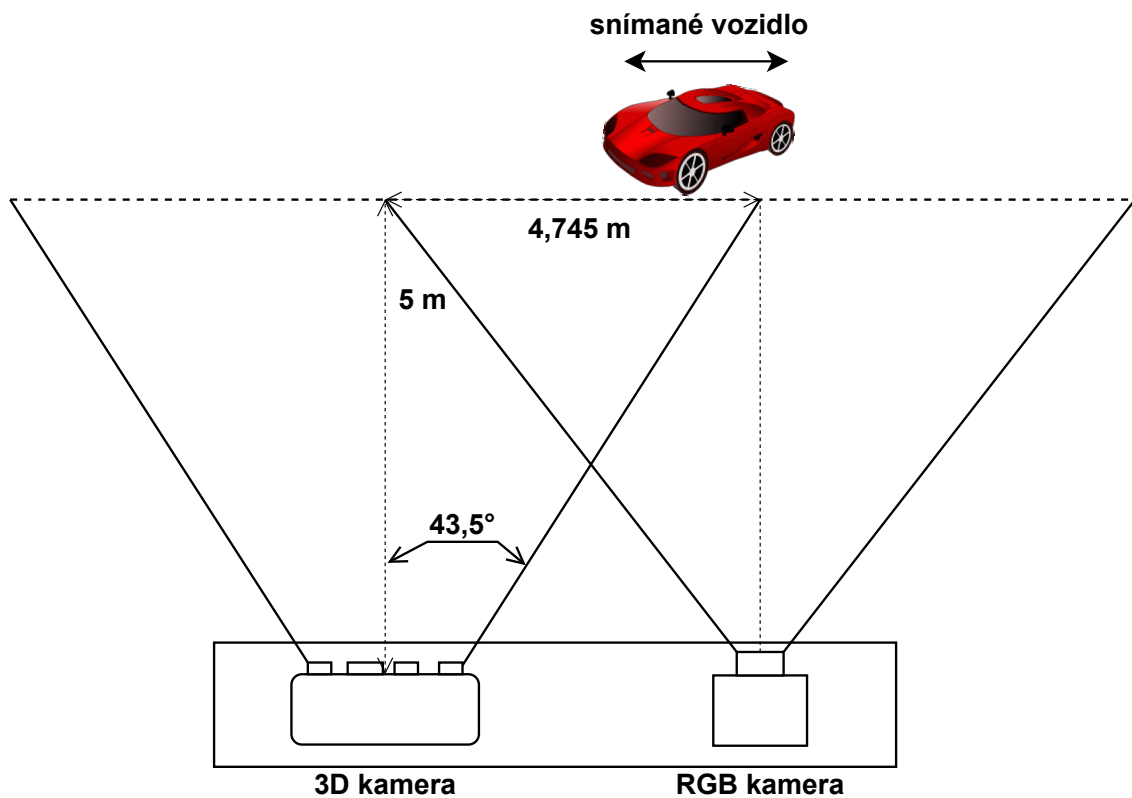
Výsledná ohnisková vzdálenost je 1,22mm. Následující tabulka 4.3 shrnuje požadované parametry RGB kamery.

Tab. 4.3: Požadované parametry RGB kamery

Parametr	RGB kamera
Rozlišení RGB senzoru	1280x720, 30fps
Maximální snímaná vzdálenost	6m
Zorné pole RGB senzoru	H:90°, V:65°
Ohnisková vzdálenost senzoru	1,22mm
Velikost senzoru	3896 μm x 2453 μm
Velikost pixelu senzoru	3 μm x 3 μm

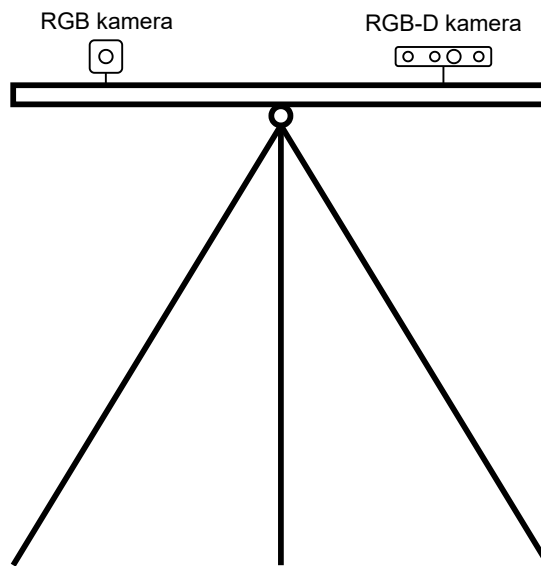
4.3 Návrh systému uchycení kamer

Při návrhu uchycení kamer bylo bráno v potaz hned několik požadavků. Systém by měl být jednoduše přenosný tak, abychom mohli snímat objekty v různých lokacích. Dále by měl být také systém co nejjednodušeji rozebíratelný a upravitelný.



Obr. 4.4: Pohyb objektu vůči kamerám

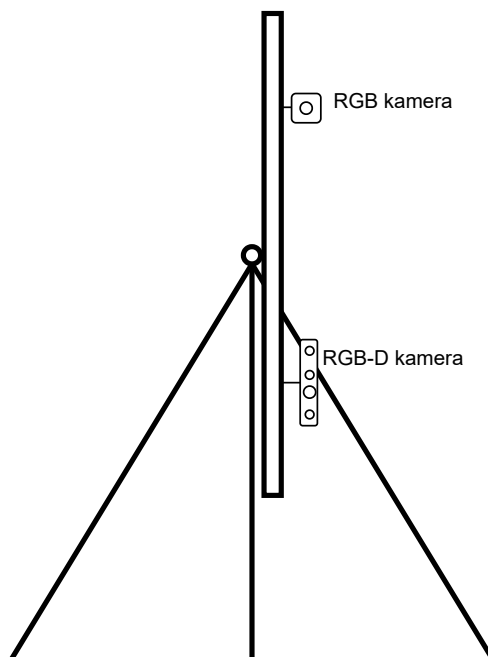
Počátečním konceptem bylo umístění obou kamer horizontálně na připravený nosník (viz obr. 4.5). Návrh se odvíjel od výsuvného teleskopického stativu. Na jeho vrcholu by byla umístěna otočná hlava, abychom mohli kamery natočit do různých úhlů (hlavně ve vertikálním směru). Na vrcholu stativu bude na závitě umístěno rameno na jehož obou stranách by byly přišroubovány kamery. V tomto uspořádání získáme široké horizontální zorné pole. Snímaný objekt se poté pohybuje horizontálně v zorném poli obou kamer (viz obr. 4.4). Obě kamery v tomto případě nasnímají několik snímků projíždějícího objektu pod stejným úhlem a výsledné snímky tak obsahují naprosto stejnou informaci o objektu. Tato informace je tudíž přeby-
tečná a vyvstává otázka jestli nejde toto uspořádání vymyslet efektivněji.



Obr. 4.5: Návrh horizontálního uchycení kamer

Druhou možností, jak kamery umístit bylo uchycení nosníku ve vertikálním směru (viz obr. 4.6). Jedna z kamer tak bude snímat spodní část objektu a druhá (výše postavená) bude snímat vrchní a střešní část objektu. Tímto uspořádáním bude docíleno toho, že získáme co nejvíce různých obrazových informací o objektu. Původní široké horizontální pole kamery tak přesuneme do vertikálního směru a naopak. Nové užší horizontální pole nám v tomto případě nevadí, protože objekty se samy pohybují tímto polem a vždy tak dosáhneme nasnímaní celého objektu. Díky pohybu objektu také vznikne několik snímků v sérii, které jsou potřebné pro stereo rekonstrukci 3D modelu objektu. Stále je ovšem důležité myslet na to, že zorná pole obou kamer se musí překrývat.

Objekty budou takto snímány po jejich celé délce od přední strany až po zadní stranu. Navrhovaný systém poté bude polovinou celkového systému, který by rekon-



Obr. 4.6: Návrh horizontálního uchycení kamer

stručoval model vozidla ze všech stran. K těmto účelům bychom potřebovali identická zařízení a konstrukci, které není z důvodu vysoké ceny možné zakoupit.

Další důležitou otázkou zůstává jednotlivé umístění kamer. 3D kamera může být umístěna na vrchní i spodní pozici nosníku. V případě umístění na spodní pozici bude kamera snímat spíše boční část vozidla a nebude tudíž snímán tak široký úhel, pod kterým vozidlo uvidí. Z vrchní pozice bude kamera snímat boční i střešní část vozidla a mohla by tak lépe vykreslit celý 3D model vozidla.

Nosník je navržen tak, aby bylo možné kamery jednoduše otočit a je tudíž možné otestovat obě varianty.

5 Systém zpracování snímků

V následující kapitole bude popsán systém, který bude zpracovávat vícero snímků objektu zájmu a vytvářet z nich jeden 3D model. Nejvhodnější reprezentací tohoto 3D modelu bude mračno bodů. V průběhu zpracování snímků budeme vytvářet více mračen bodů a tato reprezentace 3D objektu je vhodná pro slučování více objektů. Postup zpracování popisuje obrázek 5.1.

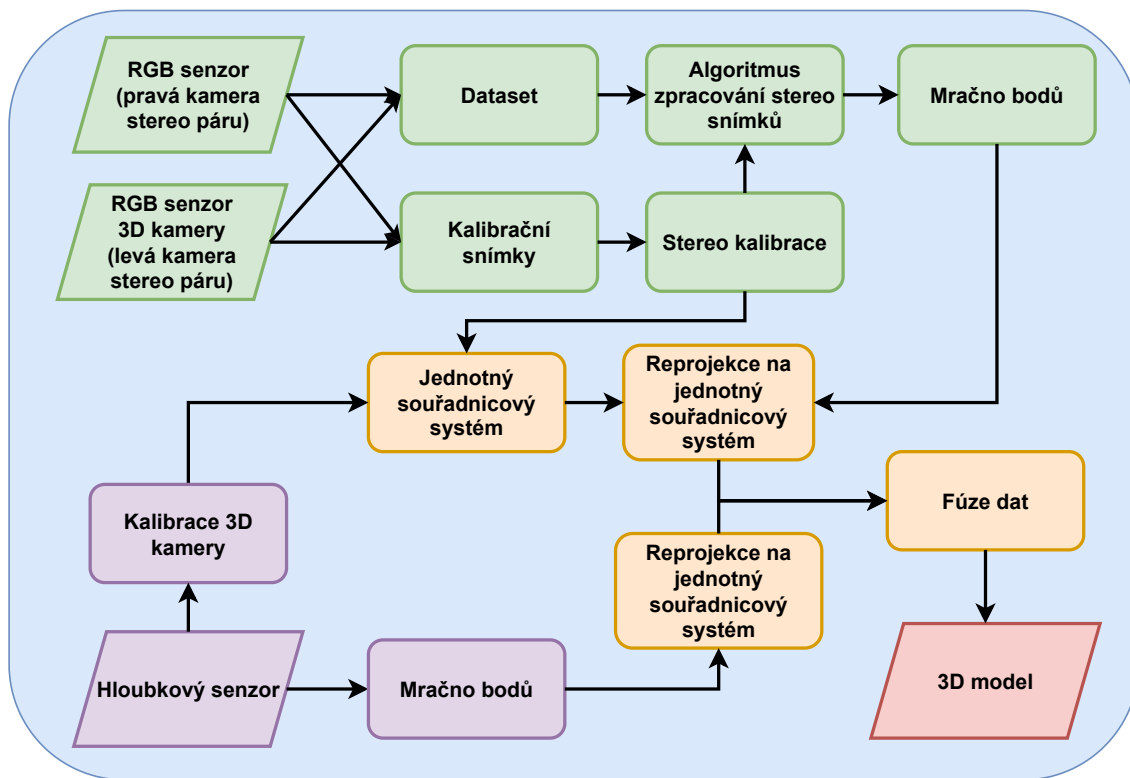
V prvním kroku budou zkalibrovány kamery, které tvoří stereo pár. Pomocí kalibrace získáme parametry potřebné k dalšímu zpracování snímků získaných stereo párem. Poté následuje vícero možností, jak požadované mračno bodů vytvořit. Z existujících řešení se nabízí tvorba hloubkové mapy pomocí které vytvoříme následně mračno bodů, nebo například metoda triangulace. U této metody nejprve nalezneme shodné body v obou snímcích a následně vypočteme jejich hloubkovou souřadnici. Vhodnost jednotlivých algoritmů je v tomto případě nutné otestovat.

Podobně budeme postupovat v případě 3D kamery. Z této kamery už přímo získáme hloubkovou mapu, ze které následovně vytvoříme mračno bodů. Poté musíme vytvořit jednotný souřadnicový systém. Obě získaná mračna bodů transformujeme do tohoto souřadnicového systému a poté jsme schopni tyto mračna bodů sloučit. Výsledkem je poté jeden 3D model získaný ze dvou RGB snímků a jedné hloubkové mapy z 3D kamery.

Při snímání dat ovšem počítáme s tím, že se objekt pohybuje před kamerami a tudíž získáme vícero snímků objektu z různých pohledů v různých časech. Pro každou sadu snímků ve stejném čase poté vytvoříme 3D model pomocí výše popsaného algoritmu. Výsledné 3D modely je poté opět nutné sloučit dohromady a vytvořit tak hustší 3D model snímaného objektu.

5.1 Softwarové prostředky

Jako programovací jazyk pro tuto práci byl zvolen Python. V tomto programovacím jazyku je vytvořeno mnoho knihoven, které jsou vhodné k vytvoření výše popsaného systému. Programování probíhá ve vývojovém prostředí PyCharm. Hlavními použitými knihovnami jsou OpenCV, Open3D, NumPy a PyRealSense. Knihovna OpenCV obsahuje širokou škálu různých funkcí a algoritmů potřebnou v oboru počítačového vidění.[20] Open 3D poté slouží pro práci s mračky bodů a jejich vizualizací.[21] Knihovna Numpy se hodí pro manipulaci s obrazovými daty jakožto maticemi. PyRealSense je poté knihovna s funkcemi pro snímání a zpracování dat z 3D kamery.



Obr. 5.1: Návrh systému zpracování snímků

5.2 Kalibrace stereo páru

Prvním krokem v algoritmu pro zpracování stereo snímků je kalibrace těchto kamer. Kalibrace slouží k získání vnitřních a vnějších parametrů obou kamer a také esenciální a fundamentální matice viz kapitola 2.

Přístupů jak provést kalibraci kamer je hned několik. Tyto přístupy se liší v tom, jaké objekty kalibrační snímky obsahují. První možností je takzvaná auto kalibrace, kdy zaznamenáváme vícero snímků obecné scény. Častěji používanou metodou je ovšem kalibrace kamer pomocí snímků známých objektů v 3D prostoru. Obvykle se k tomuto účelu používá 2D objekt, kterým je obrázek šachovnice. Tento objekt má známé parametry, kterými jsou velikost šachovnice, neboli počet bílých a černých polí na šířku a na výšku a také velikost jednoho čtvercového pole.

Funkce pro kalibraci kamer jsou již v knihovně OpenCV implementovány a fungují na principu kalibračního algoritmu popsaného Zhengyou Zhangem v roce 1998[22].

Zhangův algoritmus je obecně popisován následovně[23]:

1. Je nasnímáno několik snímků šachovnice I_0, \dots, I_{m-1} pod různými úhly. Je možno pohybovat kamerou okolo šachovnice nebo šachovnicí před kamerou.

2. Z každého snímku $I_i (i = 0, \dots, M - 1)$ je extrahováno N obrazových bodů $u_{i,0}, \dots, u_{i,j}, \dots, u_{i,N-1}$, u kterých předpokládáme, že korespondují 1:1 s body na šachovnici.
3. Pro každý pohled i jsou z extrahovaných bodů určeny související homografie H_0, \dots, H_{M-1} (lineární mapování extrahovaných obrazových bodů s body na reálné šachovnici).
4. Pomocí homografií H_i je vypočteno 5 vnitřních parametrů kamery $(\alpha, \beta, \gamma, u_c, v_c)$ vyřešením uzavřeného lineárního tvaru. V tomto momentě je ignorováno zkreslení objektivu kamery. K dosažení jedinečného řešení potřebujeme $M > 3$ pohledů. V případě, že předpokládáme, že je senzor kamery bez osového zkreslení ($\gamma = 0$) postačí jen dva různé snímky šachovnice ($N = 2$). Více snímku obecně vede k přesnějším výsledkům.
5. Se znalostí vnitřních parametrů kamery, jsou vypočteny vnější parametry kamery R_i a t_i pro každý pohled i .
6. Parametry radiálního zkreslení k_0, k_1 jsou poté vypočteny pomocí metody nejmenších čtverců.
7. Nakonec jsou všechny tyto vypočtené parametry jakožto prvotní odhad optimalizovány pomocí nelineárního optimalizačního algoritmu přes všech M pohledů.

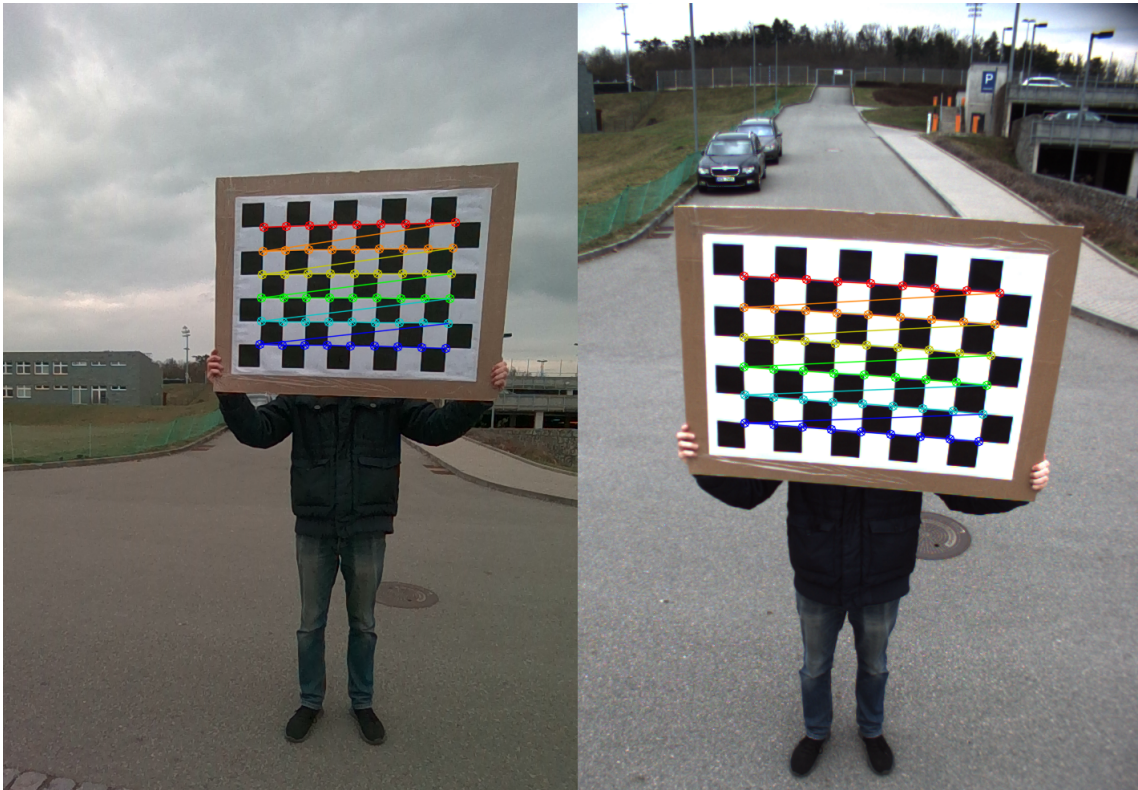
Praktická implementace poté funguje následujícím způsobem:

V prvním kroku je zapotřebí nasnímat sadu snímků zvolené šachovnice pro obě kamery. Šachovnici na snímcích natáčíme pod různým úhlem a přesouváme od jednoho rohu k druhému.

Jako obrazové body nám poslouží rohy jednotlivých polí šachovnice. Tyto místa jsou lehce lokalizovatelná díky velikému rozdílu jasové hodnoty mezi černou a bílou barvou a taky jsou všechna od sebe vzdálena o stejnou jednoznačně definovanou hodnotu. Rohy šachovnice jsou v tomto případě 3D body v prostoru se známými světovými souřadnicemi (U, V, W) . Body se nacházejí na šachovnicové rovině se souřadnicemi U a V , kde souřadnice $W=0$. Pomocí kalibrace tedy získáváme parametry, které definují vztah mezi těmito 3D body a jejich korespondujícími pixely na obrazové rovině (u, v) . Jednotlivé rohy polí nalezneme pomocí funkce `findChessboardCorners()`. K jejich nalezení musíme znát šířku jednoho pole šachovnice a velikost šachovnice tzn. počet polí na šířku i na délku. Krajní řada polí se do velikosti nepočítá. Dále je použita funkce `cornerSubPix()`, která nalezne rohy šachovnice s přesností v subpixelech. Tuto funkci není nutné použít, ale je vhodná k nalezení přesnějších výsledků. Všechny nalezené rohy jsou poté vykresleny na snímku šachovnice z důvodu kontroly viz. obrázek 5.2. Tento postup se opakuje pro všechny dvojice snímků šachovnice.

Po nalezení všech rohů šachovnice následuje kalibrace každé z kamer. Ty jsou

kalibrovány nejprve samostatně za pomoci funkce `calibrateCamera()`. Funkce vrací hodnotu reprojekční chyby, vnitřní matici kamery, koeficienty zkreslení a rotační a translační vektor.



Obr. 5.2: Nalezení rohů šachovnice (spodní kamera, vrchní kamera)

V poslední části přistoupíme ke kalibraci kamer jakožto stereo páru. K tomuto účelu použijeme funkci `stereoCalibrate()`. Funkci ovšem musíme zadat parametr, který zabráni opětovnému výpočtu matic vnitřních parametrů kamer a koeficientů zkreslení, které již máme z předchozí kalibrace. Při kalibraci jednotlivých kamer totiž dostaneme přesnější výsledky a proto je již nechceme znovu přepočítávat [22]. Z výstupu funkce dostaneme reprojekční chybu, původní vnitřní matice obou kamer a původní koeficienty zkreslení obou kamer, rotační a translační vektor, esenciální matici a fundamentální matici.

Reprojekční chyba je vzdálenost mezi sledovaným bodem promítnutým na obrazové rovině a bodem, který byl nalezen pomocí kalibrace. Sledovanými body jsou v našem případě rohy šachovnice se skutečnými souřadnicemi v 3D prostoru promítnutými do pixelů na obrazové rovině. Rozdíl mezi těmito promítnutými body a body nalezenými pomocí kalibrace nám udává reprojekční chybu v pixelech. Výše zmíněné funkce poté vracejí součet reprojekční chyby vypočtené ze všech rohů šachovnice ve

všech snímcích z obou kamer [22]. Hodnoty reprojekční chyby menší než jedna jsou obecně považovány za velmi dobré.

Výsledné hodnoty reprojekční chyby se lišili experiment od experimentu, ale obecně by se dalo říci, že pro jednotlivé kamery se reprojekční chyba pohybovala mezi hodnotami 0.1 až 0.35 a pro společnou stereo kalibraci mezi hodnotami 0.7 až 1.15.

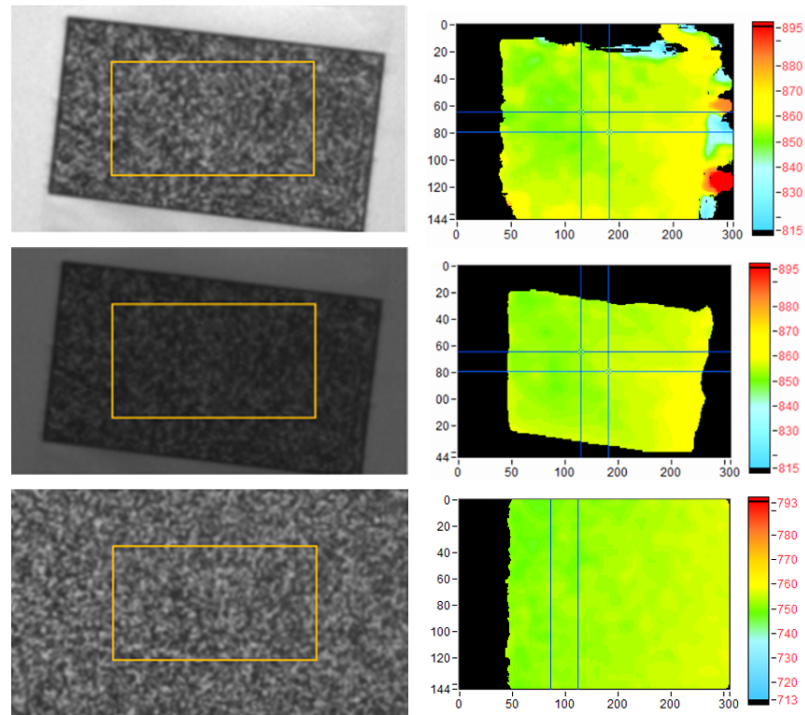
5.3 Kalibrace 3D kamery

U 3D kamery probíhá proces kalibrace zcela odlišně. Kamera je již zkalibrována od výrobce a v případě práce se zcela novou kamerou je potřeba tuto kalibraci pouze zkontrolovat. Výrobce k tomuto účelu doporučuje využít snímek připravené textury, který se umístí před kameru. Při procesu testování se rozlišení kamery zmenší na 256x144 pixelů tak, aby snímek s texturou zabíral většinu zorného pole kamery (viz obr. 5.3). K tomuto účelu se v zorném poli kamery zobrazí čtverec, který označuje hlavní pole zájmu, ve kterém by se měl celý snímek nacházet. Poté je provedena auto-kalibrace v samotném čipu 3D kamery. Výstupem této kalibrace je číslo, které uvádí jak moc se zkalibrovaná kamera odlišuje od ideálního stavu. Pro porovnání je zobrazeno číslo z předchozí kalibrace, které je momentálně uloženo ve vnitřní paměti kamery. Výrobce uvádí, že číslo menší než 0,25 indikuje ideální stav kamery. V případě vyššího čísla je doporučeno tuto kalibraci provést znovu. Při dosažení uspokojivého výsledku je poté nutné stav této kalibrace uložit do vnitřní paměti kamery a přepsat tím předchozí hodnoty.[24] Ovšem jak již bylo uvedeno výše, v případě nové kamery tato auto kalibrace slouží pouze pro kontrolu jejího současného stavu.

Tento celý proces je proveden v programu Intel RealSense Viewer, který slouží pro základní práci s kamerou pomocí připraveného uživatelského rozhraní. V našem případě byl stávající výsledek kalibrace v pořádku (tzn. číslo menší než 0,25), a proto nebylo zapotřebí hodnoty kalibrace přepisovat.

Výrobce dále uvádí, že tuto auto-kalibraci lze provést i na objektech s jednotnou texturou, jako je například bílá stěna nebo koberec. Kromě auto-kalibrace lze také provést takzvanou "Tare"kalibraci při které je navíc uvedena ručně změřená vzdálenost od kalibračního předmětu. Dále je také možno provést kalibraci ohniskové vzdálenosti obou kamer stereo páru 3D kamery. Tato kalibrace by měla vést k jednotné ohniskové vzdálenosti pro obě kamery.[24]

Po kontrole přesnosti tovární kalibrace následuje samotné získání údajů zkalibrované 3D kamery. V tomto případě je zapotřebí získat pouze matici vnitřních parametrů 3D kamery, která je následně zapotřebí pro tvorbu mračna bodů ze snímků této kamery. Tento proces již probíhá v programovacím prostředí v připraveném



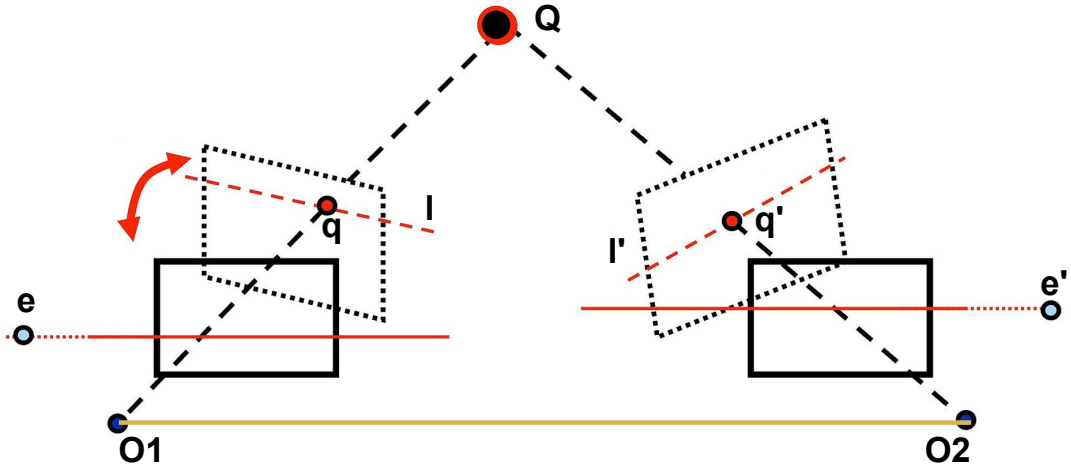
Obr. 5.3: Textura sloužící k auto-kalibraci 3D kamery[24]

skriptu s názvem "Get_3DCam_Intrinsic". Zde jsou po spuštění toku dat z kamery získány hodnoty matice vnitřních parametrů kamery pomocí funkce `get_intrinsics()` z knihovny `PyRealSense`. Z těchto hodnot je poskládána matice vnitřních parametrů kamery v jejím odpovídajícím formátu. Tato matice je následně uložena pomocí knihovny `OpenCV` ve formátu `.xml`, který je vhodný pro následnou práci s uloženými daty. Výsledkem jsou dvě matice a to jedna pro RGB čip a druhá pro stereo pár 3D kamery.

5.4 Rektifikace stereo snímků

Rektifikace snímků je operace při které přeneseme obrazové roviny dvou snímků na jednu společnou rovinu, která je rovnoběžná s přímkou spojující středy obou kamer (viz obr. 5.4). Obrazové roviny jsou poté paralelní vůči sobě. Báze je paralelní k těmto rovinám a epipóly se tudíž nacházejí v nekonečnu, jak již bylo zmíněno v kapitole 2.6. Epipolární přímky v obou snímcích jsou vůči sobě taktéž paralelní.

Toto uspořádání je velice výhodné pro následné hledání korespondencí v obou snímcích. Po rektifikaci obou snímků se hledaný bod z jednoho snímku může nacházet pouze na odpovídající epipolární přímce v snímku druhém. Tímto postupem se hledání korespondujících bodů omezí pouze na procházení jedné souřadnice snímku.



Obr. 5.4: Rektifikovaný pár[25]

V případě, že jsou kamery umístěny ve vertikálním uspořádání, budou tyto přímky probíhat skrz oba snímky ve vertikálním směru.

V programu probíhá proces rektifikace následovně:

Nejprve je zavolána funkce `stereoRectify()` z knihovny OpenCV. Vstupem této funkce jsou matice vnitřních parametrů a koeficienty zkreslení obou kamer, které získáme z jednotlivých kalibrací. Dále také rotační a translační matice ze společné kalibrace. Výstupem jsou poté rotační a projekční matice pro obě kamery. Pomocí těchto rotačních matic dosáhneme rektifikace snímků. Projekční matice P_L a P_R jsou poté nové matice vnitřních parametrů rektifikovaných kamer. Posledním výstupem z této funkce je matice Q , která slouží k transformaci pixelů hloubkové mapy na 3D body v prostoru. V případě vertikálního umístění kamer mají matice následující tvar:

$$P_L = \begin{bmatrix} f & 0 & c_x & 0 \\ 0 & f & c_{y1} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad P_R = \begin{bmatrix} f & 0 & c_x & 0 \\ 0 & f & c_{y2} & T_y * f \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (5.1)$$

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_{y1} \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{-1}{T_y} & \frac{c_{y1} - c_{y2}}{T_y} \end{bmatrix} \quad (5.2)$$

Druhým krokem je vytvoření transformační mapy. Při použití této mapy na zkreslený snímek dostaneme snímek, který je rektifikovaný dle předchozích rotačních matic a neobsahuje žádné zkreslení. Pomocí této transformační mapy pak můžeme

rektifikovat stávající snímky nebo ji použít při zachycování nových snímků a ukládat je již jako rektifikované. Pro vytvoření této mapy slouží funkce `initUndistortRectifyMap()` z knihovny OpenCV. Vstupem jsou matice vnitřních parametrů a koeficienty zkreslení obou kamer a dále také projekční a rotační matice z předchozí funkce. Výstupem je jedna transformační mapa, která je následně uložena ve formátu `.xml` pro další použití. Pro vrchní i spodní snímek je vytvořena jedna transformační mapa.

K následné rektifikaci snímků před dalším použitím slouží funkce `remap()` (viz obr. 5.5). Vstupem této funkce jsou výše zmíněné transformační mapy.



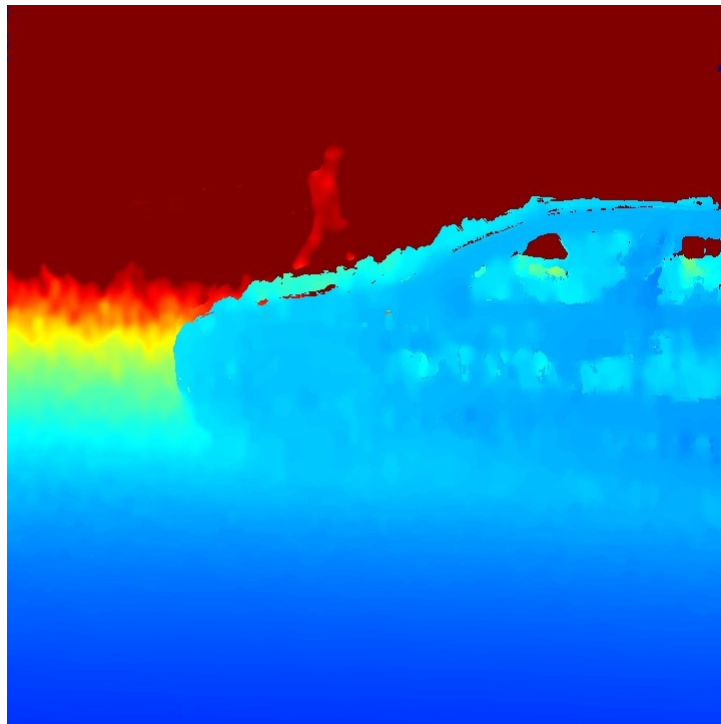
Obr. 5.5: Rektifikované snímky

5.5 Tvorba mračna bodů z 3D kamery

Mračno bodů vytvořené ze snímků 3D kamery je vytvořeno pomocí funkcí z knihovny PyRealSense, které byly k tomuto účelu připravené výrobcem kamery. Nejprve je nutné zachycené snímky upravit. Po otevření toku dat z kamery se zachytí datový rámec pomocí funkce `get_depth_frame()`. Tento rámec obsahuje hloubkovou informaci všech bodů obrazu (viz obr. 5.6). Přijatý rámec je zprvu vyfiltrován pomocí několika filtrů z knihovny kamery. První filtr slouží k zachování ostrých rohů objektu. Druhý filtr vylepšuje stejnorodost hloubkové informace dle předchozích při-

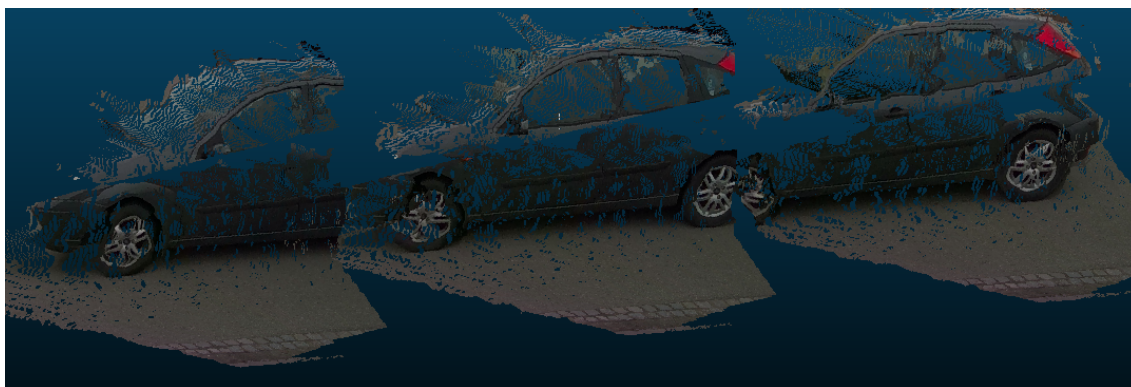
jatých rámců. Posledním filtr poté vyplní místa s chybějící hloubkovou informací pomocí nejbližších sousedních pixelů.

Následně je vytvořen objekt mračna bodů. Vyfiltrované rámce jsou předány tomuto objektu a pomocí funkce `get_vertices()` získáme jednotlivé body v 3D prostoru na základě hloubkové informace jednotlivých pixelů rámce. Tyto body jsou následně uloženy ve formátu `.raw` pro další použití. Tento formát ukládá data v jejich surové, minimálně zpracované podobě. V případě, že bychom například uložili snímek ve formátu `.jpg` dojde při kompresi ke ztrátě hloubkové informace. Výsledné mračno bodů by tak bylo méně přesné. Dále je také uložen barevný snímek ve formátu `BGRA`, díky kterému následně nanese texturu na mračno bodů.



Obr. 5.6: Hloubkový snímek v barevném podání

Výsledné mračno bodů vzniká v programu s názvem `PointCloud_From_3DCam`. Zde jsou načtena všechna surová data obsahující hloubkovou informaci společně s barevnými snímky. Ty jsou následně pomocí funkce `Vector3dVector()` převedena do formátu vhodném pro knihovnu `Open3D`. Ta slouží k ukládání, načítání a obecně práci s mračny bodů. Takto převedená data jsou uložena do výsledného mračna bodů. Toto mračno bodů je poté následně ořezáno pomocí vytvořeného obdélníku jehož obsah je zachován. Výsledné mračno bodů tak obsahuje pouze části objektu snímaného vozidla. Tento proces se opakuje pro všechny snímky ve kterých se vyskytuje sledovaný objekt (viz obr. 5.7).



Obr. 5.7: Jednotlivá mračna bodů z 3D kamery

5.6 Tvorba mračna bodů z barevných snímků

Druhé mračno bodů by dle navrženého systému mělo být vytvořeno z barevných snímků 3D kamery a RGB kamery, jakožto levého a pravého snímku stereo páru. Zde je nutno hloubkovou informaci o nasnímané scéně nejprve vypočítat. K tomuto účelu slouží několik existujících metod které budou popsány v následujících kapitolách.

Barevný snímek z 3D kamery je uložen stejně, jak bylo popsáno v předchozí kapitole s rozdílem formátu, který je tentokrát RGBA. Barevný snímek z RGB kamery je nejprve oříznut na stejnou velikost, jako snímek z 3D kamery (720x1280 pixelů) a poté je uložen ve stejném formátu. Oba snímky jsou pootočený o 90° dle natočení kamery vůči stativu.

Pro všechny snímky z obou kamer jsou navíc uloženy časové značky pro případ jejich potřeby. 3D kamera umožňuje uložení čtyřech časových značek. Dvě jsou z vnitřních hodin zařízení jakožto střední doba osvětlení senzoru a začátek přenosu časového rámce pomocí USB do počítače. Další dvě jsou generované vnitřními hodinami počítače a to konkrétně začátek přenosu dat přes USB stejně jako v předchozím případě a dokončení přenosu datového rámce do počítače.

RGB kamera bohužel žádnou podobnou funkci neobsahuje a časová značka je v jejím případě čas uložení snímku počítaný v programu. Všechny tyto časové značky jsou ukládány ve formátu .json, který je vhodný pro přenos takového typu dat.

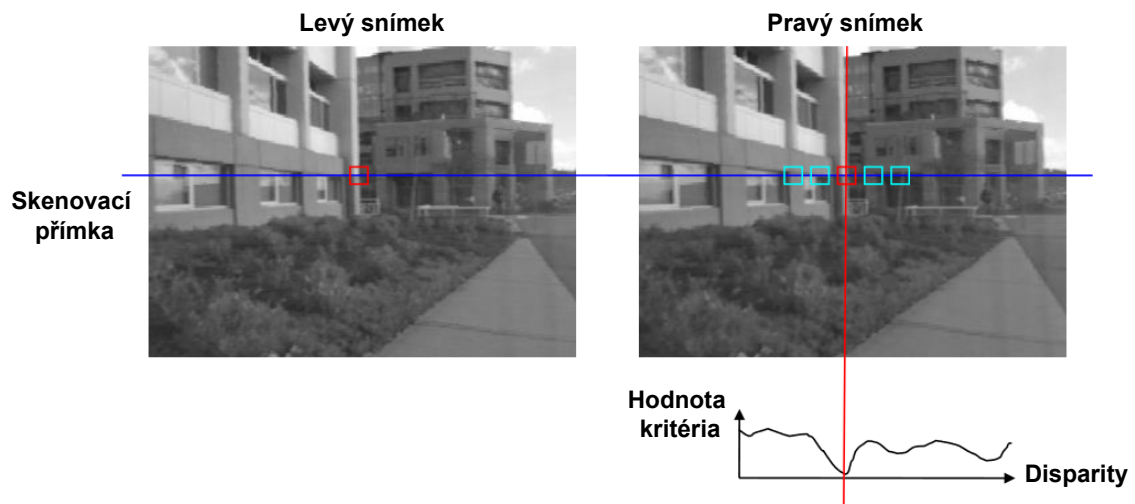
5.6.1 Tvorba mračna bodů z hloubkové mapy

Prvním otestovaným přístupem jak vytvořit mračno bodů ze dvou stereo snímků je tvorba hloubkové mapy. K tomu účelu slouží algoritmy pro hledání stereo korespondencí. Ty se obecně dělí na lokální a globální. U lokálních algoritmů hledáme korespondence v malém okně okolo zvoleného pixelu. V případě globálních algo-

ritmů je tímto oknem celá přímka nebo rovnou celý snímek. Globální metody jsou obecně přesnější než metody lokální, ale jejich výpočetní náročnost je mnohem vyšší. Lokální metody jsou jednodušší a rychlejší za cenu zhoršené přesnosti na hranách objektů a v oblastech s nejasnou texturou.[26]

V knihovně OpenCV je implementována funkce stereoBM() neboli "stereo Block Matching" a funkce stereoSGMB() neboli "stereo Semi Global Block Matching". První jmenovaná funkce je funkcí lokálního typu zatímco druhá funkce využívá přístupu globálních i lokálních metod.

Stereo Block Matching funkce vytvoří malé okno pixelů okolo zvoleného pixelu v referenčním snímku (většinou levý). V druhém snímku jsou pak nalezena podobná okna, podél zvolené přímky (viz obr. 5.8). Na základě vypočtené hodnoty podobnostního kritéria je pak zvoleno výsledné okno, které je označeno jako stejné. Takovýmto kritériem může být například suma absolutních hodnot rozdílů jednotlivých pixelů oken.[26]

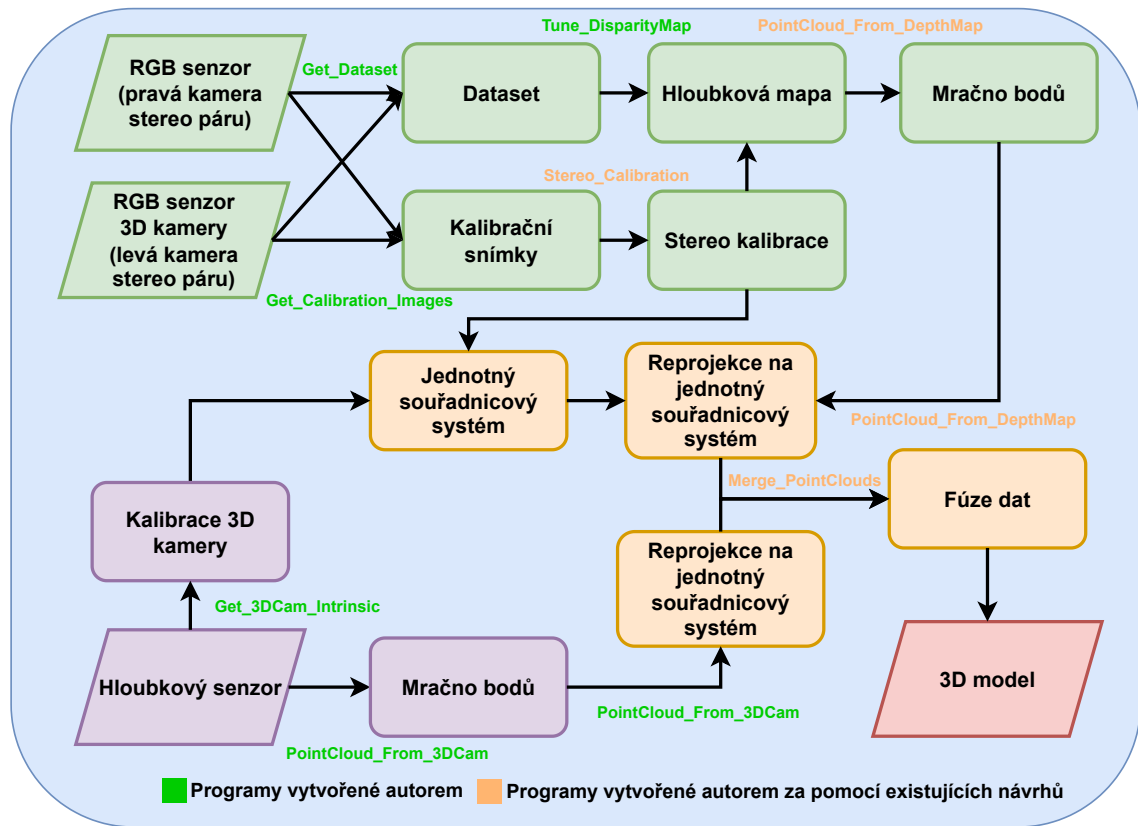


Obr. 5.8: Block Matching algoritmus[33]

Stereo Semi Global Block Matching algoritmus se od předchozího algoritmu liší tím, že podobná okna hledá na více přímkách, které vycházejí symetricky ze zvoleného pixelu do různých směrů. Stereo SGBM algoritmus je obecně o něco přesnější než stereo BM algoritmus za cenu vyšší výpočetní náročnosti.[26] V této aplikaci ovšem rychlost nehraje roli a proto byl jako výhodnější algoritmus zvolen stereo SGBM.

Celkový proces zpracování snímků v případě využití tohoto algoritmu je popsán na obrázku 5.9. Mezi jednotlivými kroky zpracování snímků jsou uvedeny názvy pod-programů ve kterých se tyto kroky vykonávají. Proces vytváření mračna bodů po-

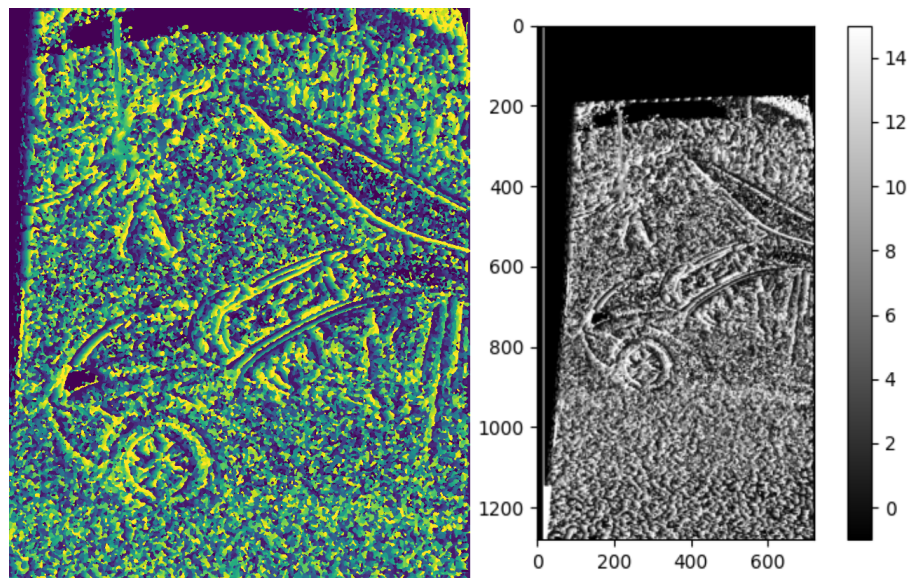
mocí stereo SGBM algoritmu probíhá v části programu s názvem PointCloud_From_DepthMap.



Obr. 5.9: Systém zpracování snímků pomocí hloubkové mapy

Prvním krokem je rektifikace snímků, jak již bylo popsáno v kapitole 5.4. V dalším kroku následuje vytvoření StereoSGBM objektu pomocí funkce StereoSGBM.create(). V této funkci je dále nutné nastavit celou řadu různých parametrů, které ovlivňují kvalitu výstupu tohoto algoritmu. Jsou to například minimální hodnota hledané disparity, počet disparit neboli rozsah disparit od minimální do maximální hodnoty nebo například velikost bloku, respektive velikost hledaného okna pixelů. Větší hledané okno bude mít za následek hladší výslednou hloubkovou mapu na úkor ztráty detailnosti. V případě menšího okna bude výsledná hloubková mapa obsahovat více detailů, ale také více šumu. Kromě těchto tří hlavních parametrů je zde možné nastavit ještě dalších 7 parametrů, které ovlivňují například plynulost výsledné hloubkové mapy, maximální možnou změnu v hodnotě disparit jednotlivých oken nebo maximální povolený rozdíl hodnot jednotlivých pixelů v levém a pravém snímku. Nejlepší kombinaci těchto hodnot není snadné nastavit a z tohoto důvodu byl pro tento účel vytvořen podprogram s názvem Tune_DisparityMap.

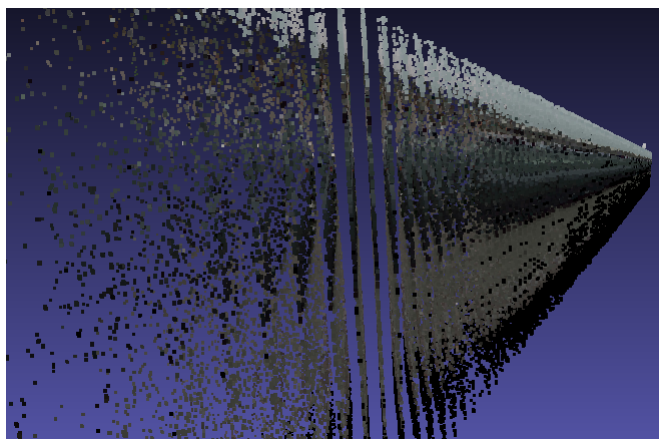
V tomto programu jsou vytvořeny posuvníky pro každý parametr, který je možné ve funkci stereoSGBM nastavovat. Na těchto posuvnících lze nastavit výchozí hodnotu těchto parametrů a jejich rozsah. Pomocí posuvníků je možné upravovat libovolnou hodnotu všech parametrů a živě při tom sledovat jak se mění výsledná hloubková mapa. Tímto způsobem je možné získat optimální kombinaci těchto parametrů. Takto získané parametry je pak jednoduše možné zpětně přepsat do předchozího podprogramu kde se následně vytvoří finální hloubková mapa. Výsledná hloubková mapa je vytvořena pomocí funkce stereo.compute(), jejíž vstupními parametry jsou levý a pravý rektifikovaný snímek. Hloubková mapa získaná tímto způsobem je na obrázku 5.10.



Obr. 5.10: Hloubková mapa získaná pomocí stereoSGBM algoritmu

Po získání hloubkové mapy následuje reprojekce jejích jednotlivých pixelů do 3D prostoru. Ta probíhá pomocí funkce reprojectImageTo3D(). Vstupem této funkce jsou hloubková mapa a transformační matice Q získaná z funkce stereoRectify(). Výstupem jsou poté jednotlivé body v 3D prostoru. Ty jsou následně převedeny do formátu Open3D. Na tyto body je následně nanášena textura pomocí barevného snímku. Vznikne tak výsledné mračno bodů, které můžeme vidět na obrázku 5.11.

Na obrázku 5.11 můžeme pozorovat, že hloubka jednotlivých pixelů není vypočtena správně, protože jednotlivé přechody mezi různými hloubkami nejsou plynulé. Tento problém vychází už ze získané hloubkové mapy. Na hloubkové mapě 5.10 můžeme pozorovat velké množství šumu u kterého nebyla určena hloubka správně. Správně vytvořená mapa by měla obsahovat plynulé přechody odstínu barvy tak, jak se předměty vzdalují od objektivu kamery. Místo toho, zde však můžeme pozoro-



Obr. 5.11: Mračno bodů vytvořené pomocí hloubkové mapy

rovat náhodnou směs barev a tudíž různou hloubku v každé části snímku. Z těchto výsledků tedy vyplývá, že stereoSGBM algoritmus není vhodný pro tuto aplikaci. Je proto nezbytné navrhnout jiný postup pro zpracování stereo snímků.

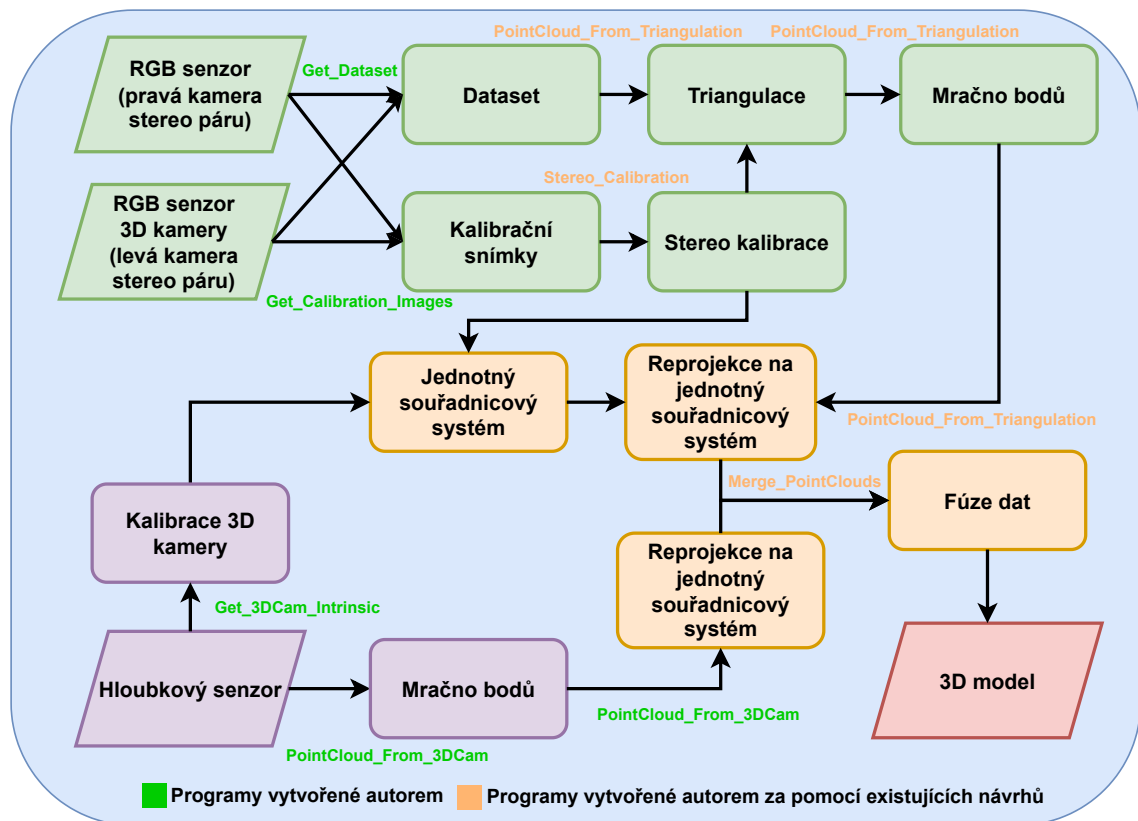
5.6.2 Tvorba mračna bodů triangulací

Další možností jak vytvořit mračno bodů ze dvou stereo snímků je technika triangulace. Tato technika funguje následujícím způsobem. Nejprve je ve snímcích nalezeno větší množství bodů zájmu. Tyto body jsou typicky výrazné body na hranách objektů nebo místa s výraznou změnou barevné intenzity. Po odfiltrování nestabilních bodů zájmu jsou zbývající body popsány většinou 128-rozměrným vektorem. Tento vektor popisuje charakteristické vlastnosti tohoto bodu tak, aby byly jednotlivé body od sebe rozlišitelné. Dále následuje nalezení stejných bodů v obou snímcích dle vypočteného vektoru vlastností.

Díky rektifikaci snímků je hledání těchto bodů omezeno pouze na okolí epipolárních přímků. Se znalostí dvou stejných bodů ve dvou snímcích a příslušných projekčních matic jsme schopni vypočítat polohu tohoto bodu v 3D prostoru za pomoci triangulace. Tímto způsobem získáme 3D souřadnice všech bodů zájmu a vznikne tak výsledné mračno bodů. Celkový proces zpracování snímků v případě využití tohoto algoritmu je popsán na obrázku 5.12.

Klíčovou částí tohoto postupu je výběr algoritmu, který nalezne body zájmu v obou snímcích a následně je popíše. Některé algoritmy pouze popisují nalezené body a jiné zvládají nalezení i popis těchto bodů najednou.

Výše popsáný algoritmus probíhá v podprogramu s názvem `PointCloud_From_Triangulation`. Postup tradičně začíná rektifikací obou stereo snímků. Poté je třeba zvolit algoritmus, který nalezne body zájmu v obou snímcích a popíše je pomocí



Obr. 5.12: Systém zpracování snímků pomocí triangulace

vektoru příznaků. V programu bylo vyzkoušeno hned několik podobných algoritmů. Často užívaným algoritmem tohoto typu je SIFT neboli "Scale-Invariant Feature Transform". Tento algoritmus nalezne klíčové body v obou snímcích a zároveň je i popíše vektorem příznaků.

Dále je zapotřebí nalézt shodné body z levého i pravého snímku. K tomuto účelu jsou v knihovně OpenCV implementovány 2 algoritmy. První se nazývá "Brute Force Matcher". Tento algoritmus projde všechny vektory, které popisují body v jednom snímku a nalezne k nim odpovídající vektory ve snímku druhém. Výběr odpovídajících vektorů probíhá pomocí zvolené metriky vzdálenosti (například euklidovská vzdálenost). Algoritmus projde všechny vektory a vrátí nejbližší odpovídající vektor. [27]

Druhý se nazývá "FLANN based Matcher" neboli "Fast Library for Approximate Nearest Neighbors". Tento algoritmus používá metody k prohledání nejbližšího okolí k nalezení hledaných korespondujících bodů. Obecně je tento algoritmus rychlejší pro velké množství snímků, než algoritmus předchozí. [27] Vhodnost prvního i druhého algoritmu také závisí na zvoleném algoritmu pro nalezení a popis bodů zájmu.

V případě algoritmu SIFT byl použit k nalezení odpovídajících bodů v obou

snímcech algoritmus FLANN. Shodné body jsou ještě následně vyfiltrovány dle zvolené prahové hodnoty přípustné odchylky vzdálenosti. Tyto body jsou následně vykresleny v odpovídajících snímcech společně s epipolárními přímkami, jak je možno vidět na obrázku 5.13.



Obr. 5.13: Stereo snímky s nalezenými body zájmu a epipolárními přímkami

Pomocí tohoto algoritmu bylo nalezeno 1084 shodných bodů. Kromě algoritmu SIFT bylo otestováno ještě několik dalších algoritmů z knihovny OpenCV. Zrychlenou verzí algoritmu SIFT je SURF. Ten je ovšem patentovaný a jeho použití je limitováno zaplacením požadované částky. [28] Z tohoto důvodu byl tento algoritmus přeskočen.

Dalším testovaným algoritmem je BRIEF neboli "Binary Robust Independent Elementary Features". Tento algoritmus slouží pouze k popisu nalezených bodů zájmu. Proto je před jeho implementací nutné použít algoritmy typu STAR nebo FAST, které slouží k nalezení těchto bodů zájmu. Výhodou toho algoritmu je jeho malá paměťová náročnost, jelikož výsledné vektory popisující nalezené body jsou ukládány v binární podobě. Je také o něco rychlejší v porovnání s algoritmem SIFT nebo SURF.[29]

Algoritmus SIFT byl dříve taktéž patentovaný a tudíž bylo za jeho použití nutné zaplatit. Z toho důvodu byl vývojáři OpenCV vytvořen algoritmus ORB. ORB jakožto "Oriented FAST and Rotated BRIEF" je kombinací těchto dvou výše popsaných algoritmů. Pomocí algoritmu FAST nalézá body zájmu a pomocí algoritmu

BRIEF tyto body popisuje. Je tudíž efektivní alternativou k výše zmíněným place-
ným algoritmům.[30]

Následujícím testovaným algoritmem je BRISK neboli "Binary Robust Invariant Scalable Keypoints". Tento algoritmus hledá i popisuje body zájmu a stejně jako ORB je alternativou k algoritmu SIFT.[31] Posledním vyzkoušeným algoritmem je FREAK neboli "Fast Retina Keypoint". Tento algoritmus slouží pouze k popisu nalezených bodů zájmu a je ho tedy nutné použít společně s algoritmy jako SIFT nebo FAST pro nalezení těchto bodů.[32]

Všechny výše popsané algoritmy byly otestovány společně s použitím algoritmů FLANN nebo Brute Force k nalezení shodných bodů v obou snímcích. Všechny algoritmy byly testovány na stejné dvojici snímků z obrázku 5.13. Jelikož rychlost algoritmů v této aplikaci nehraje důležitou roli, tak jako hlavní hodnotící kritérium byl zvolen výsledný počet nalezených bodů. Srovnání všech algoritmů dle tohoto kritéria je uvedeno v tabulce 5.1.

Tab. 5.1: Porovnání počtu nalezených bodů zájmu různými algoritmy

Algoritmus	SIFT	BRISK	ORB	BRIEF	FREAK
Počet společných bodů	1084	382	174	74	70

Z výsledků vyplývá, že nejlépe fungujícím algoritmem v tomto případě je SIFT. Nalezené shodné body jsou následně transformovány do 3D souřadnic pomocí funkce `triangulatePoints()`. Vstupem této funkce jsou body z levého i pravého snímku a projekční matice nalezené funkcí `stereoRectify()`. Z výsledných 3D bodů je následně pomocí knihovny `Open3D` vytvořeno mračno bodů.

U metody triangulace byly provedeny 2 pokusy. Z prvního pokusu vyplynulo, že snímky, které jsou získány z nesynchronizovaných kamer vytvářejí mračna bodů, která neodpovídají reálné hloubce scény. V případě nesynchronizovaných kamer je vozidlo na jednom snímku v rozdílné pozici než vozidlo na snímku druhém, protože je mezi jednotlivými snímky z různých kamer časová prodleva. Tento efekt se projeví na vypočtené hloubce a umístění shodných nalezených bodů. Hardwarová synchronizace v tomto případě nebyla proveditelná z důvodu použití dvou různých kamer. 3D kamera obsahuje piny pomocí kterých je možné hardwarovou synchronizaci provést, ale RGB kamera takovouto synchronizaci neumožňuje. Z tohoto důvodu bylo potřebné nalézt alternativní řešení.

Při druhém pokusu byla provedena fiktivní synchronizace. Snímky byly získávány postupem, kdy vozidlo vždy změnilo svoji pozici a poté na chvíli zastavilo v pozici nové. Tímto způsobem bylo dosaženo stejné pozice vozidla na obou snímcích z

obou kamer. Výsledné mračno bodů poté odpovídalo reálnému rozložení objektů ve snímané scéně. Výsledné mračno bodů vytvořené při druhém pokusu je na obrázku 5.14.



Obr. 5.14: Mračno bodů vytvořené pomocí triangulace

Na výsledném mračnu bodů lze pozorovat, že 3D rekonstrukce objektů pomocí metody triangulace nevytváří hustá mračna bodů, jako například v případě rekonstrukce pomocí hloubkové mapy. Tato metoda je omezena počtem nalezených bodů zájmu v obou snímcích. Počet nalezených bodů ovlivňuje typ snímaného objektu. Tyto body je například obtížné nalézt na vozidlech, která mají jednotlivou texturu karoserie a tudíž jednotlivé body nelze nijak rozlišit. Tento fakt byl patrný ze všech výsledných snímků výše uvedených algoritmů. Většina bodů je tak nalezena mimo objekt zájmu v jeho okolí viz obrázek 5.13. Testování bylo provedeno i na snímcích s oříznutým objektem vozidla. V tomto případě často algoritmy našly minimální až žádný počet společných bodů. Výsledné mračno bodů tak nepopisuje dostatečně snímaný objekt. Navržený postup tudíž není vhodný k použití v této aplikaci a je nutné přijít s jiným přístupem k stereo rekonstrukci.

5.6.3 Tvorba mračna bodů pomocí bundle adjustmentu

Poslední otestovanou metodou stereo rekonstrukce je metoda bundle adjustmentu. U této metody je předpokládáno, že pohyblivým objektem je kamera a snímaný objekt je stacionární. Dochází tudíž pouze ke změně pozic kamery a snímaný objekt je tak zachycován z různých úhlů pohledu. Běžně je tato metoda používána s technikami typu SLAM neboli "Simultaneous Localization and Mapping" nebo SFM neboli "Structure from Motion". U těchto technik se většinou využívají autonomní vozidla nebo bezpilotní letouny či drony.

Metoda bundle adjustmentu se snaží minimalizovat reprojekční chybu mezi 3D bodem v prostoru a jeho promítnutým bodem na obrazové rovině. K dosažení optimálního výsledku jsou neustále upravovány matice vnitřních a vnějších parametrů kamery. Dle těchto matic se upravují vypočtené body v 3D prostoru dokud není dosaženo minimální reprojekční chyby. Reprojekční chyba mezi 3D bodem v prostoru a promítnutým bodem na obrazové rovině je vypočtena pomocí Euklidovské vzdálenosti.[34] Počáteční hodnoty vnitřní a vnější matice obou kamer jsou získány pomocí stereo kalibrace. V případě stereo rekonstrukce vypadá minimalizační funkce následovně:

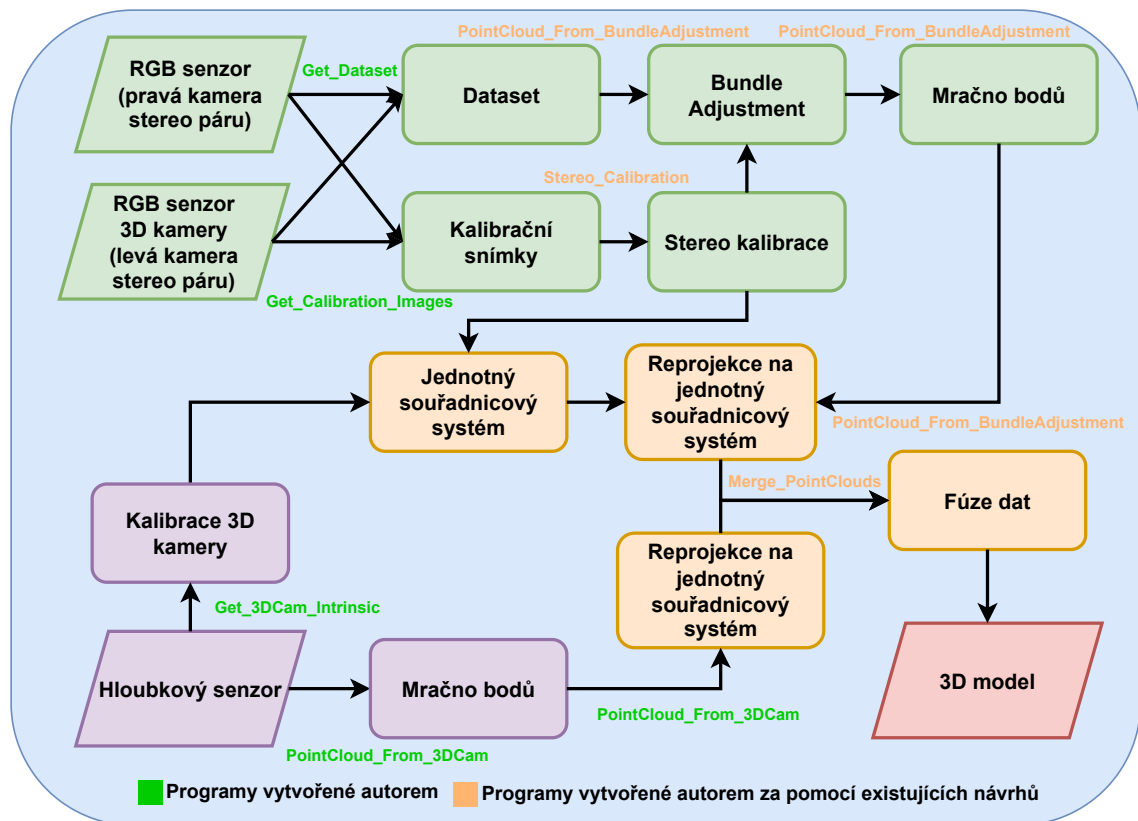
$$\operatorname{argmin}_{A,P} \sum_{j=1}^J \sum_{k=1}^K \sum_x d(p_{j,k,x}, A_{k,x}P_j)^2 \quad (5.3)$$

kde d značí Euklidovskou vzdálenost, P_j jsou body v 3D prostoru, $p_{j,k,x}$ jsou odpovídající body promítnuté na obrazové rovině levé nebo pravé kamery a $A_{k,x}$ jsou projekční matice levé nebo pravé kamery. [34]

Metoda bundle adjustmentu je vytvořena v podprogramu s názvem `PointCloud_From_BundleAdjustment`. Proces zpracování snímků s využitím metody bundle adjustmentu je zobrazen na obrázku 5.15.

Tato metoda využívá některé funkce podobně jako u metody triangulace z předchozí kapitoly. U této metody se na počátku snímky nerektifikují na rozdíl od předchozích zmíněných metod. Proces této metody probíhá v cyklu pro každý snímek, který je vždy porovnáván se snímkem následujícím. Tuto metodu lze tudíž použít i pouze pro dva snímky objektu. U načteného snímku jsou nejprve nalezeny body zájmu pomocí algoritmu SIFT. Poté jsou vždy nalezeny odpovídající body v následujícím snímku pomocí algoritmu FLANN stejně jak bylo popsáno v předchozí kapitole. Následně je pomocí odpovídajících bodů vypočtena Fundamentální matice a k ní odpovídající Esenciální matice. Poté je použita funkce `recoverPose()`. Tato funkce pomocí Esenciální matice, matice vnitřních parametrů kamery a odpovídajících bodů ve dvou snímcích vypočte rotační a translační vektory, které tvoří matici vnějších parametrů kamery. Díky matici vnějších parametrů kamery může být vypočtena projekční matice. Pomocí této matice získáme 3D body v prostoru za použití funkce `triangulatePoints()` stejně jako v předchozí kapitole. V posledním kroku je aplikován algoritmus bundle adjustmentu, který se snaží dosáhnout minimální reprojekční chyby, která je v každém kroku přepočítána. Celý proces se takto opakuje pro všechny následující snímky dokud nejsou nalezeny optimální body tvořící výsledné mračno bodů.

Z výsledků je nutné konstatovat, že metodou bundle adjustmentu bylo na stejném datasetu vytvořeno hustší mračno bodů. Vytvořené mračno bodů obsahuje 13 324 bodů. Nevýhodou této metody je ovšem její výpočetní náročnost, která roste s



Obr. 5.15: Systém zpracování snímků pomocí bundle adjustmentu

počtem parametrů, které chceme optimalizovat. Těmi jsou i vypočtené 3D body a s větším počtem snímků tudíž roste čas, potřebný k vytvoření výsledného mračna bodů.

Na výsledných mračnecích bodů bylo možné pozorovat, že při použití této metody se vyskytují stejné problémy, jako v případě metody triangulace. Výsledné body jsou opět nalezeny všude v okolí vozidla a na samotném vozidle je nalezen minimální počet společných bodů. Výsledné mračno bodů tak nereprezentuje dostatečně požadovaný objekt vozidla.

5.6.4 Tvorba mračna bodů pomocí dalších metod

V průběhu práce byly vyzkoušeny další metody, které mohou být využity k rekonstrukci 3D modelu vozidla. Tyto metody ovšem nesouvisí přímo se stereo rekonstrukcí a proto je nelze dle zadání této práce považovat za využitelné. Metody jsou zde uvedeny spíše pro dodatečné srovnání a otestování alternativních návrhů v případě budoucího postupu.

V prvním testovaném přístupu byly využity prostředky umělé inteligence. K

tomuto účelu slouží neuronová síť s názvem MiDaS.[35] Tato síť slouží k odhadu hloubkové mapy z jednoho vloženého snímku. Uživatel si nejprve zvolí natrénovaný model dle požadované přesnosti a vloží jeden snímek objektu u kterého chce provést odhad hloubkové mapy. Odhad hloubkové mapy pomocí tohoto nástroje je na obrázku 5.16.



Obr. 5.16: Odhad hloubkové mapy pomocí nástroje MiDaS

Hloubkové mapy z tohoto nástroje obsahují plynulé přechody a výsledná kvalita těchto map je tudíž poměrně dobrá. Z vygenerovaných hloubkových map lze následně vytvořit výsledné mračno bodů pomocí knihovny Open3D.

Takto vygenerovaná mračna bodů obsahují lepší textury výsledného vozidla v porovnání s modelem z 3D kamery. Přesnost výsledné hloubky je ovšem v tomto případě horší, protože se jedná pouze o odhad vzdálenosti a metoda se neopírá o skutečné měření.

Druhým testovaným nástrojem je program s názvem COLMAP. Tento nástroj slouží pro vytváření 3D modelů objektů a rozsáhlých struktur z velkého množství seřazených i neseřazených snímků. Nástroj využívá techniky používané metodou SFM zmíněné v předchozí kapitole. Je zde využita i technika bundle adjustmentu. Program obsahuje grafické rozhraní, ve kterém lze nahrát celou sadu snímků a spustit rekonstrukci modelu. Výsledkem je několik různě hustých modelů scény ze snímků.[36]

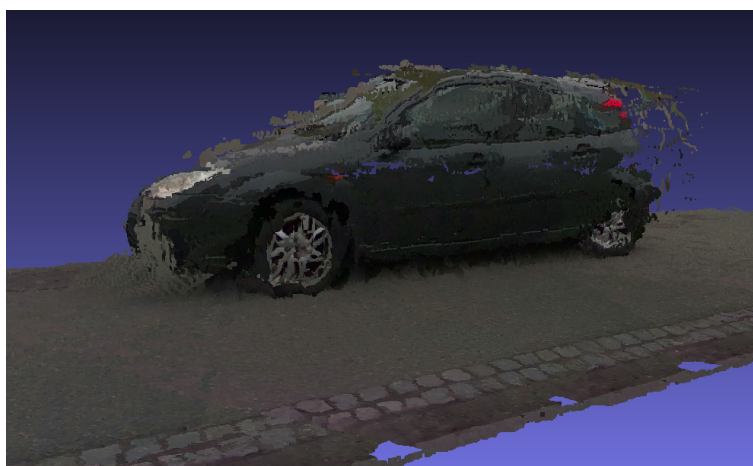
Při testování se ovšem projeví stejné problémy jako v průběhu celé této práce. Výsledné mračno bodů obsahovalo pouze okolní scénu a požadovaný objekt vozidla zde nebyl patrný.

6 Zhodnocení nasnímaného datasetu

V následující kapitole budou srovnány výsledné modely vozidel vytvořené pomocí 3D kamery a nebo pomocí neuronové sítě MiDaS z předchozí kapitoly.

6.1 Modely z 3D kamery umístěné na spodní pozici

Následující modely byly vytvořeny za pomoci 3D kamery. Z výsledných modelů byly vybrány pouze části obsahující vozidlo pomocí obdélníku s připravenými rozměry k vyříznutí požadované části modelu. Na snímku 6.1 a 6.2 můžeme vidět mračno bodů získané z 3D kamery umístěné na spodní pozici nosníku. U prvního modelu je objekt tvořen převážně boční stranou vozidla a malou částí jeho přední strany.



Obr. 6.1: Mračno bodů vytvořené 3D kamerou (Ford)



Obr. 6.2: Mračno bodů vytvořené 3D kamerou (Škoda)

U druhého modelu si můžeme povšimnout chybějících bodů na pruhu karoserie z boční strany vozidla. Zde se opět ukazuje, že jednolitě části objektů, jako například karoserie vozidel nejsou ideálními objekty pro snímání. Na lesklých površích se často vytvářejí odlesky světla, které způsobují šum a tudíž nepřesné vyhodnocení hloubky jednotlivých bodů. Na modelu si ovšem můžeme povšimnou přesahujícího nákladu ze zadní části vozidla, který je správně označen pomocí červené látky. Na tomto modelu lze demonstrovat, že kontrolu u nestandardně naloženého nákladu lze provést i v případě nedokonalých modelů.

6.2 Modely z 3D kamery umístěné na vrchní pozici

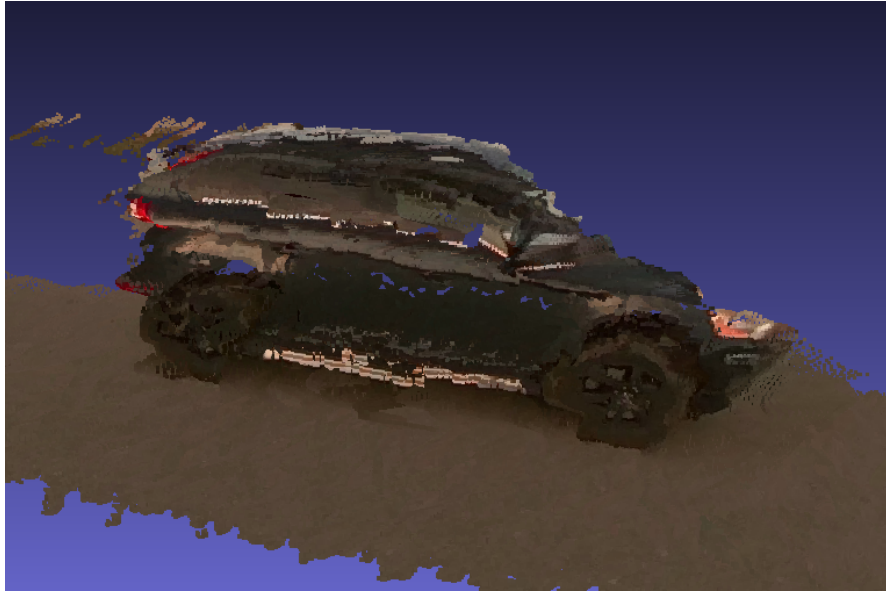
Modely v této kapitole jsou vytvořeny ze snímků 3D kamery umístěné na vrchní pozici nosníku. U prvního modelu vozidla viz. 6.3 si můžeme povšimnou silného šumu. Části tohoto modelu vozidla byly jako jediné vyříznuty z jeho scény ručně namísto připraveného obdélníku v programu popsaného výše. To má za následek nedostatečné odstranění vzdálených bodů šumu.



Obr. 6.3: Mračno bodů vytvořené 3D kamerou (Volkswagen)

U druhého modelu vozidla viz. 6.4 už není šum tolik patrný. Na obou snímcích si můžeme povšimnou jiného barevného podání oproti předchozím dvěma modelům. To je způsobeno jinými světelnými podmínkami při snímání datasetu v jiné části dne. Zde je patrné, že venkovní proměnlivé prostředí ovlivňuje svými světelnými podmínkami kvalitu výsledných modelů.

Obecně lze ale v tomto případě konstatovat, že modely vytvořené pomocí 3D kamery umístěné na vrchní pozici nosníku mají horší kvalitu a menší hustotu bodů,



Obr. 6.4: Mračno bodů vytvořené 3D kamerou (Volvo)

než modely z 3D kamery umístěné na spodní pozici nosníku. Srovnání je uvedeno v tabulce 6.1.

Tab. 6.1: Porovnání hustoty bodů výsledných modelů vozidel

Spodní pozice 3D kamery		
Model vozidla	Ford	Škoda
Počet bodů [N]	3.021.929	2.207.343
Vrchní pozice 3D kamery		
Model vozidla	Volkswagen	Volvo
Počet bodů [N]	1.119.625	1.006.332

6.3 Porovnání kvality hloubky modelu z 3D kamery s modelem z neuronové sítě

Tato kapitola slouží k porovnání kvality hloubky mračna bodů získaného z 3D kamery a mračna bodů vygenerovaného z hloubkových map, které odhadla neuronová síť.

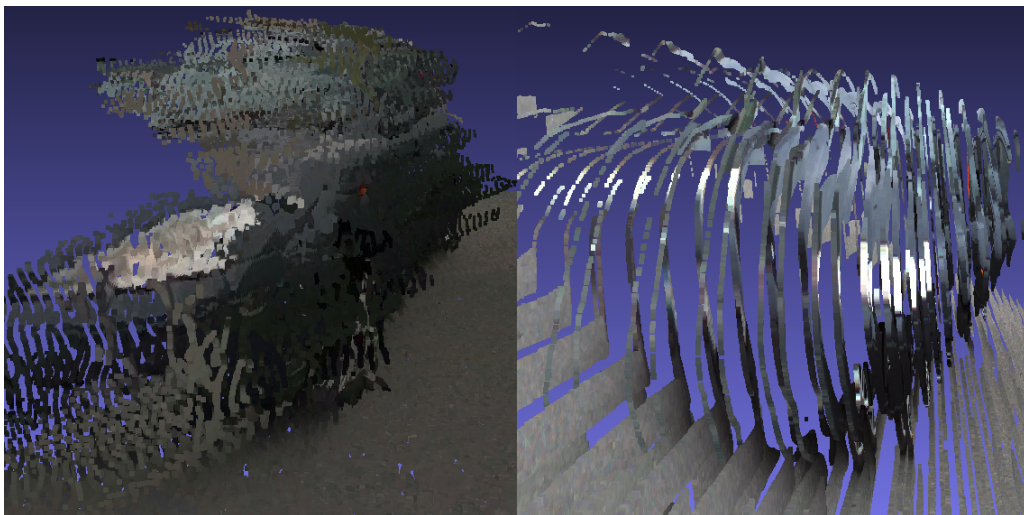
Na prvním obrázku viz. 6.5 je výsledný model který byl vytvořen z hloubkových map, které byly vygenerovány neuronovou sítí. Z obrázku je patrné že snímky z RGB

kamery mají lepší barevné podání než RGB čip u 3D kamery a proto je textura tohoto modelu lepší než u předchozích modelů.



Obr. 6.5: Mračno bodů vytvořené pomocí neuronové sítě

Druhý obrázek viz. 6.6 srovnává kvalitu hloubky modelu z předchozího snímku s modelem stejného vozidla z 3D kamery. Z toho obrázku je patrné, že 3D kamera dokáže lépe určit reálnou hloubku objektů a vytvořit její plynulý přechod. V případě odhadu hloubkové mapy je výsledné mračno bodů složeno z pásů dílčích hloubek. Tyto pásy neobsahují plynulé přechody a výsledný model tak není kvalitní reprezentací rozměrů vozidla.



Obr. 6.6: Hloubka mračna bodů z 3D kamery (vlevo) a z odhadnuté hloubkové mapy (vpravo)

Závěr

Cílem této práce bylo navrhnout a implementovat systém vhodný k rekonstrukci 3D modelů vozidel. Za pomoci takového systému by bylo možné kontrolovat vozidla v místech kritické dopravní infrastruktury.

V první kapitole jsou uvedeny metody, pomocí kterých lze rekonstruovat 3D model ze získaných 2D snímků. Následující kapitola obsahuje matematický popis snímání scény, který je pro tuto práci potřeba. Ve třetí kapitole je poté rozebráno několik návrhů systému kombinace 2D a 3D kamer. Dle jejich výhod a nevýhod je poté vybráno nejlepší řešení a to konkrétně kombinace jedné 3D kamery a jedné RGB kamery. Čtvrtá kapitola obsahuje rešerši dostupných 3D kamer na trhu z nichž je vybrána nejvhodnější kamera Intel RealSense D455. K ní je poté doplněna RGB kamera dle stanovených parametrů. Na konci kapitoly je ještě navrhnout systém uchycení těchto kamer. V páté kapitole je poté navrhnout software sloužící k snímání scény i rekonstrukci modelu. V poslední kapitole jsou pak zhodnoceny výsledné modely.

V průběhu práce bylo nasnímáno několik různých datasetů s různou konfigurací umístění kamer. 3D kamera byla umístěna na vrchní i spodní pozici nosníku. Z výsledných modelů ovšem vyplynulo, že hustší a méně zašuměné modely vznikly při umístění 3D kamery na spodní pozici.

Navrhované softwarové řešení počítalo s rekonstrukcí modelu z hloubkových dat získaných 3D kamerou a následnou rekonstrukcí druhého modelu pomocí dvou RGB snímků z 3D kamery a RGB kamery. Poté by následovalo jejich spojení.

V případě rekonstrukce z RGB snímků bylo navrženo a otestováno několik metod. První navrženou metodou bylo vytvoření hloubkové mapy dostupnými algoritmy a následné vytvoření mračna bodů pomocí této mapy. Touto metodou se ovšem nepodařilo dosáhnout rozumných výsledků. Hloubková mapa nereprezentovala dostatečně vzdálenosti ve snímané scéně a výsledné mračno bodů tak neodpovídalo skutečné hloubce scény.

Následující otestovanou metodou byla metoda triangulace. Postupem této metody je nalezení schodných bodů ve dvou snímcích z obou kamer a následné vypočtení jejich hloubky. V tomto případě ovšem nebylo možné nalézt body, které by dostatečně reprezentovali model vozidla. Jednoduchá struktura karoserie neumožňovala nalezení takto významných bodů, které by bylo možné nalézt v obou snímcích.

Podobný problém nastal i v případě 3. metody s názvem bundle adjustment. Mračna bodů zde byla hustší než v případě metody triangulace, ale rekonstruovaná scéna obsahovala body mimo snímané vozidlo stejně jako v případě triangulace.

Na závěr byly otestovány metody, které nesouvisí se stereo rekonstrukcí dle zadání této práce. Tyto metody jsou zde uvedeny pro srovnání. Jednalo se o použití

neuronové sítě k získání odhadu hloubkové mapy z jednoho snímku scény. Výsledný model vytvořený z takto získané hloubkové mapy obsahoval kvalitnější texturu, ale hloubka jednotlivých bodů neodpovídala plynulému rozložení jako například u modelů rekonstruovaných z dat z 3D kamery. Druhou alternativní metodou bylo použití nástroje sloužícího k rekonstrukci modelů metodou struktura z pohybu. Ani u toho nástroje však nebylo dosaženo kvalitních výsledků.

Výsledné modely uvedené v poslední kapitole jsou tedy vytvořeny pouze pomocí dat získaných 3D kamerou. I tak lze ale konstatovat, že vytvořené modely jsou dostatečnou reprezentací vozidel. Na příkladech uvedených v poslední kapitole lze provést kontrolu umístění nákladu přesahujícího obrys vozidla.

Z výsledků tedy vyplývá, že všechny body zadání této práce byly splněny. V případě RGB kamery lze konstatovat, že je pro tyto účely nevhodná a nadbytečná. Do budoucna by bylo vhodné navrhnout novou konfiguraci kamer a jejich umístění. Návrh by například mohl obsahovat použití čtyřech těchto 3D kamer a tudíž rekonstrukce celého modelu vozidla. Otázkou poté zůstávají možnosti financování takového řešení.

Literatura

- [1] KALOVÁ, Ilona a HORÁK, Karel. *Optické metody měření 3D objektů*. Online. Elektrorevue. 2005. Dostupné z:
https://www.researchgate.net/publication/333845234_Opticke_metody_mereni_3D_objektu. [cit. 2023-10-17].
- [2] KALOVÁ, Ilona a LISZTWAN, Marek. *Active Triangulation Technique*. Online. In 5th International Conference of PhDStudents. 2005. Miskolc: University of Miskolc, p. 99-104. ISBN: 963-661-673- 6 Dostupné z:
https://www.researchgate.net/publication/333776357_Active_Triangulation_Technique. [cit. 2023-10-18].
- [3] CHVALOVÁ, Klára. *3D rekonstrukce objektů pomocí stereovize*. Praha: ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Dostupné z:
https://dspace.cvut.cz/bitstream/handle/10467/85064/F2-DP-2019-Chvalova-Klara-DP_Chvalova_2019.pdf?sequence=-1&isAllowed=y. [cit. 2023-10-18].
- [4] HANSARD, Miles; LEE, Seungkyu; CHOI, Ouk a HORAUD, Radu. *Time of Flight Cameras: Principles, Methods, and Applications*. Springer Science & Business Media. 2012, roč. 2012, s. 103. Dostupné z:
https://www.researchgate.net/publication/281885252_Time_of_Flight_Cameras_Principles_Methods_and_Applications/related. [cit. 2023-10-23].
- [5] *Scratchapixel 3.0*. Dostupné z:
<https://www.scratchapixel.com/lessons/3d-basic-rendering/perspective-and-orthographic-projection-matrix/projection-matrix-introduction.html>. [cit. 2023-11-08].
- [6] SZELISKI, Richard. *Computer Vision: Algorithms and Applications*. Online. 2nd Edition. Springer Cham, 03 January 2022n. 1. ISBN 978-3-030-34372-9. Dostupné z:
<https://szeliski.org/Book/>. [cit. 2023-11-08].
- [7] HATA, Kenji a SAVARESE, Silvio. *CS231A Course Notes 1: Camera Models*. Online. In: Stanford University. Dostupné z:
https://web.stanford.edu/class/cs231a/course_notes/01-camera-models.pdf. [cit. 2023-11-08].
- [8] BERGAMASCO, Filippo *The Pinhole Camera Model*. Online, Presentace. Venice: DAIS, Ca' Foscari University of Venice, Academic year 2016/2017.

- Dostupné z:
https://www.dsi.unive.it/~bergamasco/teachingfiles/cvslides/11_pinhole_camera_model.pdf. [cit. 2024-02-13].
- [9] ZANUTTIGH, Pietro; MARIN, Giulio; DAL MUTTO, Carlo; DOMINIO, Fabio a MINTO, Ludovico. *Time-of-Flight and Structured Light Depth Cameras*. Online. 1. Springer International, 24 May 2016n. 1. ISBN 978-3-319-30973-6. Dostupné z:
<https://link.springer.com/content/pdf/10.1007/978-3-319-30973-6.pdf>. [cit. 2023-11-10].
- [10] *Camera Calibration and 3D Reconstruction*. Online. OpenCV 2.4.13.7 documentation. 2011, Dec 31, 2019 Dostupné z:
https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html. [cit. 2023-11-10].
- [11] HATA, Kenji a SAVARESE, Silvio. *CS231A Course Notes 3: Epipolar Geometry* Online. In: Stanford University. Dostupné z:
https://web.stanford.edu/class/cs231a/course_notes/03-epipolar-geometry.pdf. [cit. 2023-11-10].
- [12] KADAMBI, Achuta; BHANDARI, Ayush a RASKAR, Ramesh. 3D Depth Cameras in Vision: Benefits and Limitations of the Hardware. Online. In: *Computer Vision and Machine Learning with RGB-D Sensors*. Springer, Cham, 15 July 2014n. 1., s. 24. ISBN 978-3-319-08651-4. Dostupné z:
https://web.media.mit.edu/~achoo/tr/3d_benefits_limits.pdf. [cit. 2023-11-19].
- [13] *From Pixels to 3D Shapes: An Overview of 3D Data Representations*. Online. Medium. Jul 30, 2023. Dostupné z:
<https://medium.com/@deepyachowdary/from-pixels-to-3d-shapes-an-overview-of-3d-data> [cit. 2023-11-19].
- [14] KOIFMAN, Vladimir. *Toshiba Proposes Single-Lens RGB-Depth Camera*. Online. In: Image Sensors World. Tuesday, June 21, 2016. Dostupné z:
https://1.bp.blogspot.com/-Ni2fK33cHV4/V218mtmygkI/AAAAAAAAAPao/Tfgua0wfHYs1WCa676UBPCW94d9QrK_wgCLcB/s1600/Toshiba%2BRGB-Z%2B1606_01_2.jpg. [cit. 2023-11-19].
- [15] BHATTI, Asim. *Stereo Vision*. Online. Janeza Trdine 9, 51000 Rijeka, Croatia: IntechOpen, 2008, s. 382. ISBN 978-953-7619-22-0. Dostupné z:

- <https://books.google.cz/books?id=5jehDwAAQBAJ&printsec=copyright&hl=cs#v=onepage&q&f=false>. [cit. 2023-11-19].
- [16] AL-NAJI, Ali Abdulelah a CHAHL, Javaan. *Detection of Cardiopulmonary Activity and Related Abnormal Events Using Microsoft Kinect Sensor*. Online. *Sensors*. 03/2018n. l., roč. 2018, č. 03, s. 18. Dostupné z: <https://doi.org/10.3390/s18030920>. [cit. 2023-12-10].
- [17] NĚMEC, Filip. *Xbox Kinect – Nejlepší pohybové hry pro Kinect a jeho (ne)slavný osud (TÉMA)*. Online. Alza. 2022. Dostupné z: <https://www.alza.cz/gaming/xbox-kinect-nejlepsi-hry-neslavny-osud>. [cit. 2023-12-10].
- [18] *Beginner's guide to depth (Updated)*. Online. July 15, 2019. Dostupné z: <https://www.intelrealsense.com/beginners-guide-to-depth/>. [cit. 2023-12-11].
- [19] *D455*. Online. In: Intel RealSense. 2021. Dostupné z: <https://www.intelrealsense.com/wp-content/uploads/2021/10/D455.png>. [cit. 2023-12-11].
- [20] G, Bradski. *The OpenCV Library*. Online. Dr. Dobb's Journal of Software Tools. 2000. Dostupné z: <https://opencv.org/>. [cit. 2024-04-23]
- [21] ZHOU, Qian-Yi; PARK, Jaesik a KOLTUN, Vladlen. *Open3D A Modern Library for 3D Data Processing*. Online. ArXiv:1801.09847. 2018. Dostupné z: <https://www.open3d.org/>. [cit. 2024-04-23]
- [22] *Camera Calibration and 3D Reconstruction*. Online. OpenCV. Dostupné z: https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html#ga687a1ab946686f0d85ae0363b5af1d7b. [cit. 2024-02-23].
- [23] BURGER, Wilhelm. *Zhang's Camera Calibration Algorithm: In-Depth Tutorial and Implementation*. Online. ResearchGate. May 2016, s. 56. Dostupné z: <https://doi.org/10.13140/RG.2.1.1166.1688/1>. [cit. 2024-02-23].
- [24] GRUNNET-JEPSEN, Anders; SWEETSER, John a KHUONG, Tri. *Intel® RealSense™ Self-Calibration for D400 Series Depth Cameras*. Online. S. 35. Dostupné z: <https://dev.intelrealsense.com/docs/self-calibration-for-depth-cameras>. [cit. 2024-03-17].

- [25] SAVARESE, Silvio. *Model used for image rectification example*. Online. Wikipedia, the free encyclopedia. Dostupné z: https://en.wikipedia.org/wiki/Image_rectification#/media/File:Lecture_1027_stereo_01.jpg. [cit. 2024-03-18].
- [26] VERMA, Dr. Rachna. *A Comparative Evaluation of Leading Dense Stereo Vision Algorithms using OpenCV*. Online. In: International Journal of Engineering Research & Technology (IJERT). ETRASCT' 14 Conference Proceedings, s. 6. ISSN 2278-0181. Dostupné z: <https://www.ijert.org/research/a-comparative-evaluation-of-leading-dense-stereo-vision-algorithms-using-opencv-IJERTCONV2IS03023.pdf>. [cit. 2024-03-18].
- [27] OPENCV. *Feature Matching*. Online. Dostupné z: https://docs.opencv.org/3.4/dc/dc3/tutorial_py_matcher.html. [cit. 2024-04-22].
- [28] SURF: (*speeded up robust feature*) *algorithm based localization method and robot (China)*. Přihl.: 2013-12-13. Uděl.: 2014-09-03. CN103640018A. Dostupné z: <https://patents.google.com/patent/CN103640018A/en>. [cit. 2024-04-22].
- [29] OPENCV. *BRIEF (Binary Robust Independent Elementary Features)*. Online. Dostupné z: https://docs.opencv.org/4.x/dc/d7d/tutorial_py_brief.html. [cit. 2024-04-22].
- [30] OPENCV. *ORB (Oriented FAST and Rotated BRIEF)*. Online. Dostupné z: https://docs.opencv.org/3.4/d1/d89/tutorial_py_orb.html. [cit. 2024-04-22].
- [31] ALSULAIMAN, Hasan. *Feature-matching using BRISK*. Online. Medium. Dec 20, 2020. Dostupné z: <https://medium.com/analytics-vidhya/feature-matching-using-brisk-277c47539e8>. [cit. 2024-04-22].
- [32] *A tutorial on binary descriptors – part 5 – The FREAK descriptor*. Online. Gil's CV blog. Dostupné z: <https://gilscvblog.com/2013/12/09/a-tutorial-on-binary-descriptors-part-5-the-freak>. [cit. 2024-04-22].
- [33] *Stereo*. Online. In: 16-385 Computer Vision. Carnegie Mellon: The ROBOTICS INSTITUTE, 2020, s. 130. Dostupné z: <http://www.cs.cmu.edu/~16385/lectures/lecture13.pdf>. [cit. 2024-03-18].

- [34] KURZ, Christian; THORMAHLEN, Thorsten a SEIDEL, Hans-Peter. *Bundle Adjustment for Stereoscopic 3D. Online.* Online. S. 12. Dostupné z: https://www.informatik.uni-marburg.de/~thormae/paper/MIRA11_2.pdf. [cit. 2024-04-04]
- [35] RANFTL, René; LASINGER, Katrin a HAFNER, David. *Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-shot Cross-dataset Transfer.* Online. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. S. 14. Dostupné z: <https://doi.org/https://doi.org/10.48550/arXiv.1907.01341>. [cit. 2024-04-05]
- [36] SCHONBERGER, Johannes L. a FRAHM, Jan-Michael. *Structure-from-Motion Revisited.* Online. In: Conference on Computer Vision and Pattern Recognition (CVPR). University of North Carolina at Chapel Hill, 2016, s. 10. Dostupné z: https://openaccess.thecvf.com/content_cvpr_2016/papers/Schonberger_Structure-From-Motion_Revisited_CVPR_2016_paper.pdf. [cit. 2024-04-05]

Seznam symbolů a zkratek

CCD	Charge-coupled device – Zařízení vázané na náboj
CMOS	Complementary Metal Oxide Semiconductor – Komplementární polovodič s oxidem kovu
RGB	Red-Green-Blue – Způsob míchání barev (červená-zelená-modrá)
TOF	Time of Flight – Měření doby letu
IR	Infrared (light) – Infračervené světlo
USB	Universal Serial Bus – Univerzální sériová sběrnice

A Obsah přiložené SD karty

V této příloze se nachází seznam všech souborů, které se nachází na přiložené SD kartě.

```
/.....kořenový adresář přiloženého archivu
├── Diplomová_Práce_Jan_Boch.pdf ..... diplomová práce v pdf
├── 3D_Reconstruction_Boch ..... vytvořený program
│   ├── Data.....složka pro vstupní a výstupní data programu
│   ├── Podprogramy.py ..... vytvořené dílčí skripty
│   ├── README.txt ..... návod pro práci s programem
│   └── requirements.txt ..... soubor pro stažení balíčků použitých v práci
├── Nasnímané_Datasety ..... složka obsahující nasnímané datasety v průběhu práce
└── Vytvořené_Modely ..... složka obsahující vytvořené 3D modely
```