

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

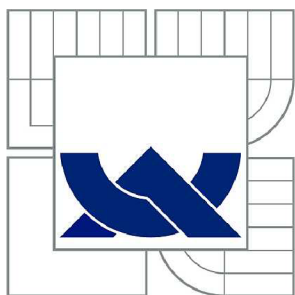
MĚŘENÍ VZDÁLENOSTI STEREOSKOPICKÝM SENZOREM

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

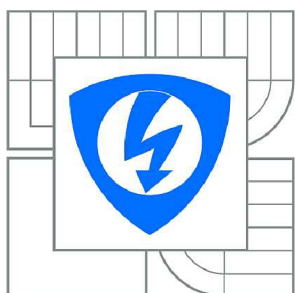
Bc. ONDŘEJ VAVROŠ

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

# MĚŘENÍ VZDÁLENOSTI STEREOSKOPICKÝM SENZOREM

STEREOSCOPIC SENSOR FOR DISTANCE MEASUREMENT

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

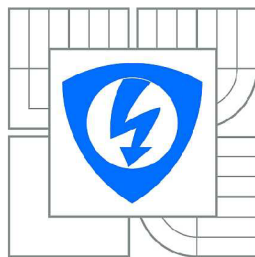
**Bc. ONDŘEJ VAVROŠ**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Ing. KAMIL ŘÍHA, Ph.D.**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Diplomová práce

magisterský navazující studijní obor  
Telekomunikační a informační technika

**Student:** Bc. Ondřej Vavroš

**ID:** 125691

**Ročník:** 2

**Akademický rok:** 2014/2015

**NÁZEV TÉMATU:**

## Měření vzdálenosti stereoskopickým senzorem

### POKYNY PRO VYPRACOVÁNÍ:

Cílem studenta bude za pomoci soustavy, tvořené dvojicí kalibrovaných kamer, zpracovat pořízené stereoskopické snímky do podoby hloubkové mapy. Z takto vypočtené hloubkové mapy a známých kalibračních parametrů, které student získá z kalibrace kamer, dále vypočte vzdálenosti předmětů před soustavou kamer.

Praktickým výstupem diplomové práce bude návrh aplikace, pracující s pořízeným stereoskopickým záznamem, která zobrazí vzdálenosti sledovaných předmětů před soustavou kamer. Pro zvýšení přesnosti měření bude uvažován výpočet disparitní mapy se sub-pixelovou přesností.

### DOPORUČENÁ LITERATURA:

[1] Andrew Harlley and Andrew Zisserman. Multiple view geometry in computer vision (2. ed.) . Cambridge University Press, 2006.

[2] BRADSKI, Gary; KAEHLER, Adrian. Learning OpenCV. [s.l.] : O'Reilly, 2008.

[3] CYGANEK, B.; SIEBERT, P., J.; An Introduction to 3D Computer Vision Techniques and Algorithms: Wiley 2009.

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 26.5.2015

**Vedoucí práce:** doc. Ing. Kamil Říha, Ph.D.

**Konzultanti diplomové práce:**

**doc. Ing. Jiří Mišurec, CSc.**

*Předseda oborové rady*

### 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 vás provede teoretickým postupem, který vám umožní určit vzdálenost objektu od stereoskopického senzoru. Součástí práce je popis kroků pro dosažení cíle, tzn. získání obrazu, provedení kalibrace, rektifikace. Dále vás práce provede přehledem algoritmů pro vytvoření disparitní mapy, subpixelové přesnosti a určením vzdálenosti objektu od senzoru. V následující části se práce věnuje implementaci těchto postupů do aplikace, jejímž cílem je měření vzdálenosti.

## **KLÍČOVÁ SLOVA**

Stereoskopie, stereo obraz, epipolární geometrie, rektifikace, disparita, disparitní mapa, subpixelová přesnost, algoritmus, měření vzdálenosti, webkamera

## **ABSTRACT**

This master's thesis will take us through theoretical procedure that allows us to determine the distance of an object by stereoscopic sensor. Part of this work presents the description of the steps to achieve image of objects, calibration and rectification. At the next part our study provides an overview of algorithms for creating disparity maps, subpixel accuracy and determining the distance of the object from sensor. In the following part of thesis deals with the implementation of these processes into application which aim is to measure the distance.

## **KEYWORDS**

Stereoscopy, stereo picture, epipolar geometry, rectification, disparity, disparity map, subpixel accuracy, algorithm, distance measurement, webcam



## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Měření vzdálenosti stereoskopickým senzorem“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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)

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Martinovi Hasmandovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

(podpis autora)



Faculty of Electrical Engineering  
and Communication  
Brno University of Technology  
Technická 12, CZ-61600 Brno  
Czech Republic  
<http://www.six.feec.vutbr.cz>

## PODĚKOVÁNÍ

Výzkum popsany v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno .....

.....

(podpis autora)



EVROPSKÁ UNIE  
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ  
INVESTICE DO VAŠÍ BUDOUCNOSTI



# OBSAH

Úvod	11
<b>1 Specifikace cíle a popis problémů</b>	<b>12</b>
<b>2 Analýza a návrh řešení</b>	<b>13</b>
2.1 Získání obrazu . . . . .	13
2.2 Předzpracování obrazu pro stereoskopii . . . . .	13
2.2.1 Dírkový model kamery . . . . .	13
2.2.2 Stereoskopie . . . . .	16
2.2.3 Epipolární geometrie . . . . .	16
2.2.4 Fundamentální matice . . . . .	19
2.3 Srovnávací metrika . . . . .	20
2.3.1 Sum of absolute differences . . . . .	21
2.3.2 Sum of squared differences . . . . .	21
2.3.3 Normalized cross-correlation . . . . .	21
2.4 Subpixelová přesnost . . . . .	22
2.5 Změna vzájemné polohy a úhlu kamer . . . . .	24
2.5.1 Vliv změn poloh kamer na měření vzdálenosti . . . . .	24
2.6 Určení vzdálenosti . . . . .	26
<b>3 Realizace</b>	<b>29</b>
3.1 Kamery . . . . .	29
3.2 Software . . . . .	29
3.2.1 Zobrazení menu a nastavení vstupních parametrů . . . . .	30
3.2.2 Výběr z aktuálně dostupných kamer . . . . .	31
3.2.3 Kalibrace kamer . . . . .	31
3.2.4 Tvorba disparitní mapy, subpixelová přesnost a určení vzdálenosti . . . . .	31
3.3 Aplikace . . . . .	39
3.4 Měření . . . . .	39
3.4.1 Metodika měření . . . . .	39
3.4.2 Naměřené hodnoty . . . . .	42
3.4.3 Zhodnocení měření . . . . .	47
3.4.4 Faktory ovlivňující měření . . . . .	47
<b>4 Závěr</b>	<b>49</b>
<b>Literatura</b>	<b>50</b>

<b>Seznam zkratk</b>	<b>52</b>
<b>Seznam příloh</b>	<b>53</b>
<b>A Tabulky měření</b>	<b>54</b>
A.1 Měření pro SGBM disparitu . . . . .	54
A.2 Měření pro vlastní implementaci disparity . . . . .	55
A.3 Porovnání metrik . . . . .	56
<b>B Optický disk</b>	<b>58</b>
B.1 Diplomová práce . . . . .	58
B.2 OpenCV . . . . .	58
B.3 Zdrojový kód aplikace . . . . .	58
B.4 Zkompilovaná aplikace . . . . .	58
<b>C Návod k použití aplikace</b>	<b>59</b>

# SEZNAM OBRÁZKŮ

2.1	Perspektivní projekce [7] . . . . .	13
2.2	Ukázka provádění kalibrace . . . . .	16
2.3	Epipolární geometrie, pohled dvou kamer . . . . .	18
2.4	Hledání podobností . . . . .	21
2.5	Subpixelový odhad hodnoty disparity[6] . . . . .	22
2.6	Situační náčrt posunu jedné z kamer do strany . . . . .	24
2.7	Situační náčrt vzájemně natočených kamer a posunu jedné z kamer do strany . . . . .	25
2.8	Vliv změny polohy na disparitu . . . . .	26
2.9	Diagram zachycení objektu kamerami v kanonické konfiguraci . . . . .	27
2.10	Příklad triangulace . . . . .	28
3.1	Použitá kamera Sweex webcam WC035V2[5] . . . . .	29
3.2	Zjednodušený diagram struktury softwaru . . . . .	30
3.3	Ukázka disparity BM, SGBM a VAR . . . . .	33
3.4	Obrazový set Tsukuba s ground truth disparitní mapou . . . . .	36
3.5	Graf závislosti chyby na velikosti agregovacího okna . . . . .	37
3.6	Ukázka postupu kalibrace . . . . .	40
3.7	Výřez disparitní mapy - nejmenší měřitelná vzdálenost . . . . .	41
3.8	Ukázka špatně a dobře exponovaného snímku . . . . .	41
3.9	Graf měření, kamery paralelně, rozteč 4,5 cm . . . . .	43
3.10	Graf měření, kamery paralelně, rozteč 10 cm . . . . .	43
3.11	Graf měření, kamery soustředěně, rozteč 10 cm . . . . .	44
3.12	Graf měření, kamery soustředěně, rozteč 4,5 cm . . . . .	44
3.13	Graf měření, SGBM, pixelová přesnost . . . . .	45
3.14	Graf měření, vlastní implementace, pixelová přesnost . . . . .	45
3.15	Graf měření, SGBM, subpixelová přesnost . . . . .	46
3.16	Graf měření, vlastní implementace, subpixelová přesnost . . . . .	46
C.1	Aplikace - Konzolové okno . . . . .	59
C.2	Aplikace - Určení kamer . . . . .	59
C.3	Aplikace - Snímání a kalibrace . . . . .	60
C.4	Aplikace - Vizuální kontrola provedené kalibrace . . . . .	60
C.5	Aplikace - Snímání pro měření - metoda SGBM . . . . .	61
C.6	Aplikace - Snímání pro měření - vlastní implementace . . . . .	61

## SEZNAM TABULEK

A.1	Měření vzdálenosti, kamery paralelně, rozteč 4,5 cm . . . . .	54
A.2	Měření vzdálenosti, kamery paralelně, rozteč 10 cm . . . . .	54
A.3	Měření vzdálenosti, kamery soustředěně, rozteč 10 cm . . . . .	54
A.4	Měření vzdálenosti, kamery soustředěně, rozteč 4,5 cm . . . . .	55
A.5	Měření vzdálenosti, kamery paralelně, rozteč 4,5 cm . . . . .	55
A.6	Měření vzdálenosti, kamery paralelně, rozteč 10 cm . . . . .	55
A.7	Měření vzdálenosti, kamery soustředěně, rozteč 10 cm . . . . .	56
A.8	Měření vzdálenosti, kamery soustředěně, rozteč 4,5 cm . . . . .	56
A.9	Závislost chyby na velikosti agregačního okna, Tsukuba . . . . .	56
A.10	Závislost chyby na velikosti agregačního okna, Teddy . . . . .	57
A.11	Závislost chyby na velikosti agregačního okna, Cones . . . . .	57

# ÚVOD

Práce se zabývá využitím obyčejných webových kamer pro snímání stereo obrazu. Využití stereo obrazu je široké, například promítat vzdálený objekt pro každé oko zvlášť a zprostředkovat tím tak prostorový vjem tohoto vzdáleného objektu. Jinou variantou využití stereo obrazu je jeho počítačové zpracování, například do hloubkové mapy. Hloubková mapa vyjadřuje vzdálenost jednotlivých soustav ploch od kamery. Pro zvýšení přesnosti hloubkové mapy se práce zabývá také subpixelovou přesností. Pokročilejším zpracováním lze dosáhnout 3D modelu pozorované scény. V tomto případě je cílem zpracování stereo obrazu do hloubkové mapy se sub-pixelovou přesností a z ní určení vzdálenosti pozorovaného objektu od kamery.

Jak již bylo řečeno, stereoskopický senzor bude tvořen dvěma běžně dostupnými webovými kamerami. Dále bude tento stereoskopický senzor doplněn o aplikaci naprogramovanou v jazyce C/C++ za současného využití knihoven pro počítačové vidění známé jako OpenCV. V první části práce si specifikujeme cíle. V druhé části probereme teorii o zpracování obrazu pro stereoskopii a určování vzdálenosti. V třetí části bude realizace práce.

Výsledný produkt této práce může být základem k napodobení Kinectu od firmy Microsoft, který se používá k ovládní herní konzole Xbox. Obecně se k určení hloubky u Kinectu využívá infračerveného senzoru s infračerveným projektorem a běžné kamery. Projektor na snímanou scénu promítá "flíčkový" vzor, který je v celé ploše ortogonální a vytváří tak souřadnicovou mapu pro infračervený senzor, jedná se o kameru snímající v infračerveném spektru světla, a spolu s běžnou kamerou tvoří stereoskopickou sestavu podobnou jako v této práci.[8]. Přesné podrobnosti o způsobu získání hloubkové mapy z Kinectu nejsou uveřejněny, protože jsou předmětem know-how společnosti Microsoft.



# 1 SPECIFIKACE CÍLE A POPIS PROBLÉMŮ

Tato práce je založena na využití běžného počítače, v tomto případě notebooku s operačním systémem Windows, a příslušenství, tj. webkamer. Webkamery komunikují s počítačem prostřednictvím USB (Universal Serial Bus, univerzální sériová sběrnice). Jelikož se jedná o takzvané generické zařízení, odpadá tím pro nás nutnost manuální instalace ovladačů pro webkamery a s tím spojená možná nekompatibilita s operačním systémem. Jelikož tyto ovladače pro generická zařízení má operační systém v základu, provede se instalace webkamer automaticky. Pro stereoskopický efekt budou webkamery umístěny tak, aby byly objektivy nasměrované rovnoběžně k objektu zájmu a měly rozestup odpovídající minimálně rozestupu očí.

Pro úspěšné snímání stereo obrazu je zapotřebí prozkoumat problematiku kalibrace webkamer. Z toho plyne, že musí být zajištěna vzájemná obrazová synchronizace. Jinými slovy je třeba odstranit geometrické poruchy, například soudkovitost, způsobené nedokonalou optikou webkamer, dále je nutné zajistit řádkovou synchronizaci, např. aby konkrétní roh snímaného objektu byl na stejném řádku jak v levém, tak v pravém snímku páru.

Dalším bodem bude zpracování nasnímaného páru snímků. Pro jeho zpracování již existují více či méně složité algoritmy, které nám vytvoří takzvanou hloubkovou mapu, která se odborněji nazývá disparitní mapa. Abychom mohli určit, který algoritmus je pro naše účely nejvhodnější, musíme prostudovat současné existující řešení a problematiku stereoskopie. Nabyté vědomosti využijeme i při určování vzdálenosti objektů z hloubkové mapy.

Na základě nastudovaných znalostí o kalibraci kamer a stereoskopii budeme uvažovat o možnosti vzájemného posuvu a změny úhlů obou webkamer. Představa je, že změnou rozestupu anebo změnou vzájemného úhlu kamer se zvýší přesnost a maximální měřitelná vzdálenost systému. O vlivu těchto změn pro měření vzdálenosti budeme diskutovat.

Vývoj stereoskopického systému bude probíhat ve vývojovém nástroji Visual Studio se svobodnou multiplatformní knihovnou pro manipulaci s obrazem OpenCV. Díky použití OpenCV, která disponuje funkcemi využívající OpenCL (Open Compute Language, průmyslový standard pro paralelní programování) nebo CUDA (Compute Unified Device Architecture, architektura umožňující běh programů na technologii OpenCL), lze na vybavenějším počítači užít paralelního zpracování a snížit tak prodlení mezi zachycením obrazu a výslednou hloubkovou mapou s určením vzdálenosti. Rovněž díky multiplatformnosti lze, s menší úpravou kódu, provozovat zpracování obrazu na operačním systému založeném na Linuxovém jádře a na embeded zařízeních. Toho může být využito v budoucích pracích.

## 2 ANALÝZA A NÁVRH ŘEŠENÍ

### 2.1 Získání obrazu

K pořízení snímků stereo obrazu budou použity dvě USB webkamery. Jedná se o běžné, komerčně dostupné webkamery, které lze zakoupit v supermarketech, resp. v jejich odděleních elektroniky.

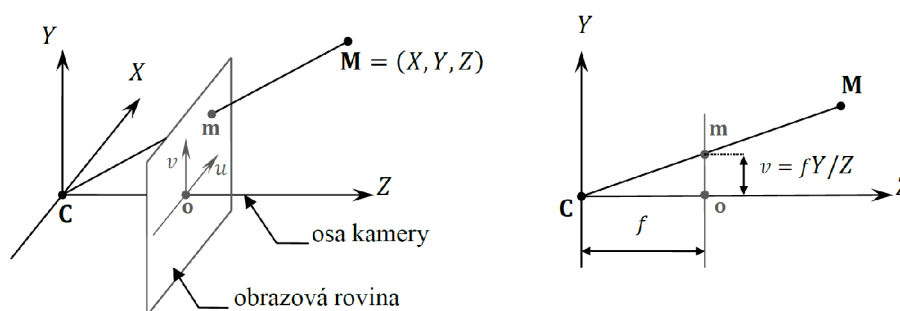
Detaily, jak kamery získávají a zpracovávají obrazovou informaci není třeba znát, protože tyto úkony obstarávají webkamera a počítač sami v rámci hardwarových a softwarových implementací dodávaných výrobcem těchto zařízení. Můžeme se tak zaměřit přímo na stereoskopické měření.

### 2.2 Předzpracování obrazu pro stereoskopii

Budeme snímat cílový 3D objekt do dvou 2D snímků. Abychom pochopili, jak jsou jednotlivé body 3D prostoru promítány obrazy v kamerách, poslouží nám model kamery.

#### 2.2.1 Dírkový model kamery

Dírkový model kamery je jeden z jednodušších modelů kamery. Používá takzvané perspektivní projekce (viz Obr. 2.1)[6]. Model pracuje s 16 parametry: 4 vnitřními,



Obr. 2.1: Perspektivní projekce [7]

popisujícími vlastnosti kamery, a 12 vnějšími, udávajícími polohu a natočení kamery.

Souřadnice bodu  $\mathbf{m} = (u, v)^T$  lze získat pomocí podobnosti trojúhelníků. Platí, že

$$u = \frac{fX}{Z}, \quad v = \frac{fY}{Z} \quad (2.1)$$

S využitím vlastností homogenních souřadnic je možné použít maticový zápis

$$\tilde{\mathbf{m}} \doteq \mathbf{P}\tilde{\mathbf{M}}, \quad (2.2)$$

kde:

$\tilde{\mathbf{m}}$  a  $\tilde{\mathbf{M}}$  jsou body v homogenních souřadnicích,  
 $\mathbf{P}$  značí transformační matici perspektivní projekce.

Symbol  $\doteq$  ovšem označuje rovnost pro nenulový parametr, protože detailně je tento vztah možné pro jednoduchý případ transformační matice  $\mathbf{P}$  rozepsat jako

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \doteq \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{pmatrix} \frac{fX}{Z} \\ \frac{fY}{Z} \\ 1 \end{pmatrix} \quad (2.3)$$

nebo také

$$\begin{pmatrix} su \\ sv \\ s \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \quad (2.4)$$

kde  $s \neq 0$  je výše zmíněný nenulový parametr a v tomto případě je jeho hodnota rovna  $Z$ . Vztah (2.2) je tedy možné zapsat jako

$$Z\tilde{\mathbf{m}} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{M}. \quad (2.5)$$

Tento vztah je základní rovnicí perspektivní projekce. Udává matematickou formulaci promítání libovolného bodu ve 3D prostoru  $\mathbf{M} = (X, Y, Z)^T$  do obrazové roviny bodového modelu kamery, resp. vznik promítaného bodu vyjádřeného v homogenních souřadnicích  $\tilde{\mathbf{m}} = (u, v, 1)^T$  [7].

Pro případy promítání, kdy střed projekce se nenachází v počátku globálních souřadnic 3D prostoru a osa promítání není shodná s osou  $Z$ , je třeba projekční matici  $\mathbf{P}$  popisující model kamery doplnit o další parametry, konkrétně vnitřní a vnější, jak je uvedeno v úvodu kapitoly. Transformační matice je rozšířena na

$$\mathbf{P} = \mathbf{A}\mathbf{P}_0\mathbf{G}, \quad (2.6)$$

která po dosazení do rovnice (2.2) vypadá následovně

$$\tilde{\mathbf{m}} \doteq \mathbf{A}\mathbf{P}_0\mathbf{G}\tilde{\mathbf{M}} \quad (2.7)$$

Pro názornost je možné rovnici 2.7 rozepsat na

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \cong \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \left[ \begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{array} \right] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.8)$$

Kde:

- $(u, v, 1)^T$  je vektor bodu  $\tilde{\mathbf{m}}$  v homogenních souřadnicích,
- $c_x, c_y$  udávají polohu středu obrazu,
- $f_x, f_y$  jsou ohniskové vzdálenosti v upravených jednotkách (k pixelu),
- $r_{mn}$  jsou prvky matice rotace  $\mathbf{R}$ ,
- $t_m$  značí prvky vektoru posunutí  $\mathbf{t}$ ,
- $(X, Y, Z, 1)^T$  je vektor bodu  $\tilde{\mathbf{M}}$  v homogenních souřadnicích[7]

Rovnici (2.8) lze dále zjednodušit

$$\tilde{\mathbf{m}} \cong \mathbf{A} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11}X + r_{12}Y + r_{13}Z + t_1 \\ r_{21}X + r_{22}Y + r_{23}Z + t_2 \\ r_{31}X + r_{32}Y + r_{33}Z + t_3 \\ 1 \end{bmatrix}, \quad (2.9)$$

v dalším kroku zkrátíme na

$$\tilde{\mathbf{m}} \cong \mathbf{A} \begin{bmatrix} r_{11}X + r_{12}Y + r_{13}Z + t_1 \\ r_{21}X + r_{22}Y + r_{23}Z + t_2 \\ r_{31}X + r_{32}Y + r_{33}Z + t_3 \end{bmatrix} = \mathbf{A} [\mathbf{RM} + \mathbf{t}]. \quad (2.10)$$

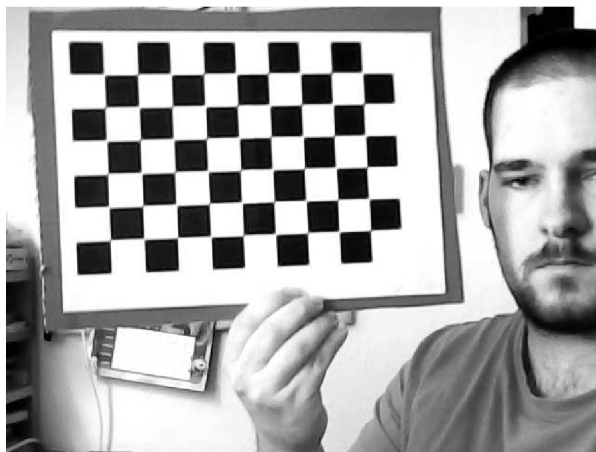
S využitím rovnice (2.5) je výsledek tohoto postupu možné zapsat pomocí úplné rovnosti jako

$$Z\tilde{\mathbf{m}} = \mathbf{A} [\mathbf{RM} + \mathbf{t}]. \quad (2.11)$$

U běžných kamer výrobci vnitřní parametry neuvádí vůbec, uvádí je jen u některých kamer pro profesionální použití, proto vnitřní parametry musíme získat pomocí kalibrace.

Metod kalibrace kamer existuje více[9], v našem případě se spokojíme s kalibrací na základě známé scény, objektu či obrazce. Kameře předložíme obrazec, kalibrační metoda jeho strukturu zná a proto rozpozná jeho záchytné body. Z pozic záchytných bodů se odhadnou vnitřní případně korekční parametry kamery. Korekční parametry opravují radiální zkreslení jako jsou soudkovitost a poduškovitost. K tomuto účelu opět využijeme knihovnu OpenCV, která obsahuje funkce pro odhad ze známých záchytných bodů.

Pro kalibraci kamer bude stačit před ní podržet v různých polohách a na určitý čas černobílou šachovnicí. Počet a polohy umístění šachovnice závisí na její velikosti. A to, zda vykryje najednou celý záběr obou kamer tak, aby nechyběla žádná část šachovnice. Pokud ne, šachovnicí zakrýváme alespoň část záběru, potom je vhodné polohovat šachovnicí tak, abychom iterativně zakryli celý záběr.



Obr. 2.2: Ukázka provádění kalibrace

### 2.2.2 Stereoskopie

Porovnávat dva dvourozměrné obrazce a hledat v nich vzájemně korespondující pixely by bylo neúnosně složité a zdlouhavé. Epipolární geometrie nám pomůže zredukovat prohledávaný prostor. S její pomocí provedeme rektifikaci obrazu. Po této úpravě si jednotlivé řádky stereoskopického páru obrazu budou korespondovat a nám se zredukuje vyhledávaný prostor do jednoho rozměru.

### 2.2.3 Epipolární geometrie

Obrázek 2.3 znázorňuje zobrazovací konfiguraci zahrnující dva projektivní systémy. Vytvářejí tak systém získávání stereoskopického obrazu. Je založen na dvou dírkových modelech kamer, každý složený z projektivní roviny  $\Pi_i$  (kde index  $i$  se změní na l pro levou rovinu a na r pro pravou rovinu) s příslušnými projekčními středy  $O_i$ . Přímka procházející projekčním středem  $O_i$  a je kolmá k projekční rovině  $\Pi_i$  protíná tuto rovinu v bodě nazývaném jako *hlavní bod*. Vzdálenost od projektivního středu k projekční rovině se nazývá *ohnisková vzdálenost*. Linie spojující středy  $O_l$  a  $O_r$  se nazývá *základní linie*. V místech protnutí základní linie a obrazových rovin  $\Pi_i$  se nachází *epipolární body*. Ve zvláštním případě, kdy základní linie neprotíná rovinu  $\Pi_i$ , leží epipolární body v nekonečnu.

Rovina tvořená bodem v prostoru  $\mathbf{M}$  a projekčními středy  $\mathbf{O}_1$  a  $\mathbf{O}_r$  se nazývá *epipolární rovina*  $\Pi_e$ . Epipolární rovina  $\Pi_e$  protíná obrazové roviny  $\Pi_1$  a  $\Pi_r$ , kde jejich průniky jsou *epipolární přímky*. Role epipolárních přímek může být pochopena například prostřednictvím analýzy bodu  $\mathbf{m}_1$  levého obrazu (bod je reprezentován vektorem) prostorového bodu  $\mathbf{M}$ . V tomto případě střed  $\mathbf{O}_1$  a bod  $\mathbf{m}_1$  definuje polopřímku  $\overline{\mathbf{O}_1\mathbf{m}_1}$ . Můžeme konstatovat, že bod  $\mathbf{m}_1$  je obrazem bodu  $\mathbf{M}$ , ale také všech bodů ležících na polopřímce  $\overline{\mathbf{O}_1\mathbf{m}_1}$ . Znamená to, že bod  $\mathbf{M}$  může ležet kdekoli na této polopřímce a má stále stejný obraz. Proto pro určení přesné polohy prostorového bodu  $\mathbf{M}$  nelze vycházet z jednoho obrazu. Pro upřesnění prostorové polohy bodu  $\mathbf{M}$  potřebujeme druhý bod obrazu z jiného pohledu. Tím je, například, bod obrazu  $\mathbf{m}_r$  na obrazové rovině  $\Pi_r$ . Bod  $\mathbf{m}_r$  a  $\mathbf{O}_r$  tvoří tentokrát polopřímku  $\overline{\mathbf{O}_r\mathbf{m}_r}$ . Tato polopřímka je v projekčním středu  $\mathbf{O}_r$  fixovaná a současně se volně pohybuje po polopřímce  $\overline{\mathbf{O}_1\mathbf{m}_1}$ , průsečík obou polopřímek se nachází v prostorovém bodě  $\mathbf{M}$ . Navíc, polopřímky  $\overline{\mathbf{O}_1\mathbf{m}_1}$  nebo  $\overline{\mathbf{O}_r\mathbf{m}_r}$  prochází příslušnými obrazovými rovinami  $\Pi_1$  nebo  $\Pi_r$  v místech, kde leží epipolární přímky. Stejně tak projekce těchto polopřímek na protějších obrazových rovinách tvoří epipolární přímky. Z toho vyplývá takzvaná *epipolární podmínka*:

Každý bod obrazu  $\mathbf{m}_i$  prostorového bodu  $\mathbf{M}$  leží pouze na epipolární přímce odpovídající obrazové rovině.

Toto omezení hraje důležitou roli při hledání obrazových bodů (neznáme jejich prostorové body), omezuje vyhledávání z celé obrazové roviny na hledání podél epipolární přímky. V obecném případě nejsou pozice epipolárních přímek předem známy. Nicméně, pro speciální případ kanonické kamerové konfigurace jsou pozice epipolárních přímek známy, tato skutečnost může značně zjednodušit vyhledávání odpovídajících bodů. Kanonická kamerová konfigurace je taková, kdy si obrazové řádky kamer odpovídají, tj. jsou horizontálně zarovnané. Optické osy kamer jsou v této konfiguraci paralelní, epipóly se přesunou do nekonečna a epipolární přímky v obrazových rovinách se stanou rovněž paralelní.

Každá z kamer stereo systému disponuje vlastním souřadnicovým systémem, jehož výchozí bod se nachází ve středovém bodu kamery  $\mathbf{O}_i$ . Osa  $Z$  tohoto souřadnicového systému odpovídá optické ose kamery. V obou souřadných systémech vektory  $\mathbf{M}_1 = (X_1, Y_1, Z_1)^T$  a  $\mathbf{M}_r = (X_r, Y_r, Z_r)^T$  představují stejný prostorový bod  $\mathbf{M}$ . Na druhou stranu, na obrazových rovinách vektory  $\mathbf{m}_1 = (x_1, y_1, z_1)^T$  a  $\mathbf{m}_r = (x_r, y_r, z_r)^T$  určují dva různé obrazy prostorového bodu  $\mathbf{M}$ . Navíc si můžeme všimnout, že  $z_1 = f_1$  a  $z_r = f_r$ , kde  $f_1$  a  $f_r$  jsou ohniskové vzdálenosti levé a pravé kamery v pixelech.

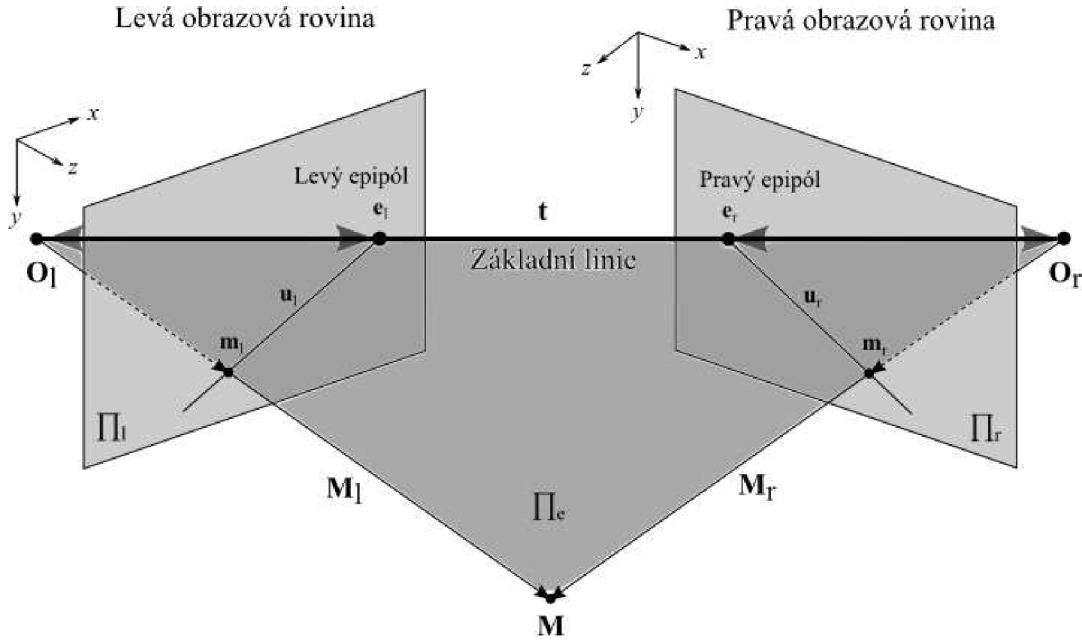
Jak již bylo naznačeno v předchozí části, každá kamera je popsána souborem vnějších parametrů určujících vztah umístění kamery k vnějšímu souřadnicovému

systemu. Tím, že každá kamera stereo systému disponuje vlastním souřadným systémem, lze provést převod z jednoho souřadnicového systému do druhého pomocí translace  $\mathbf{t} = \mathbf{O}_r - \mathbf{O}_l$  a rotace popsanou ortogonální maticí  $\mathbf{R}$ . Tedy pro dva vektory  $\mathbf{M}_l$  a  $\mathbf{M}_r$  ukazující na stejný bod  $\mathbf{M}$  v prostoru platí následující:

$$\mathbf{M}_r = \mathbf{R} (\mathbf{M}_l - \mathbf{t}) . \quad (2.12)$$

Epipolární rovina  $\Pi_e$  v souřadnicovém systému spjatém s levou kamerou je určena vektory  $\mathbf{t}$  a  $\mathbf{M}_l$ . Proto také vektor  $\mathbf{M}_l - \mathbf{t}$  patří do této roviny. To znamená, že jejich smíšený součin musí být roven nule. Vztah vypadá takto[6]:

$$(\mathbf{M}_l - \mathbf{t}) \cdot (\mathbf{t} \times \mathbf{M}_l) = 0 \quad (2.13)$$



Obr. 2.3: Epipolární geometrie, pohled dvou kamer

Součin (2.13) může být zapsán v maticovém tvaru jako součin matice  $\mathbf{S}$  a vektoru  $\mathbf{P}_1$ , který je v následující rovnici:

$$\begin{aligned} \mathbf{T} \times \mathbf{P}_1 &= \begin{vmatrix} t_1 & t_2 & t_3 \\ M_{11} & M_{12} & M_{13} \\ \mathbf{i} & \mathbf{j} & \mathbf{k} \end{vmatrix} \\ &= \mathbf{i} (t_2 M_{13} - t_3 M_{12}) - \mathbf{j} (t_1 M_{13} - t_3 M_{11}) + \mathbf{k} (t_1 M_{12} - t_2 M_{11}) \\ &= \begin{bmatrix} -t_3 M_{12} + t_2 M_{13} \\ t_3 M_{11} - t_1 M_{13} \\ -t_2 M_{11} + t_1 M_{12} \end{bmatrix} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix} \begin{bmatrix} M_{11} \\ M_{12} \\ M_{13} \end{bmatrix} = \mathbf{S} \mathbf{M}_1 \end{aligned} \quad (2.14)$$

kde  $\mathbf{i} = (1, 0, 0)^T$ ,  $\mathbf{j} = (0, 1, 0)^T$ ,  $\mathbf{k} = (0, 0, 1)^T$  jsou jednotkové vektory a  $\mathbf{S}$  je diagonálně symetrická matice. Nyní provedeme substituci rovnic (2.12) a (2.14) do rovnice (2.13), tím získáme

$$\left(\mathbf{R}^{-1}\mathbf{M}_r\right)^T \mathbf{S}\mathbf{M}_l = 0. \quad (2.15)$$

Vzhledem tomu, že matice  $\mathbf{R}$  je ortogonální, lze ji jednoduše přeskádat

$$\mathbf{M}_r^T \mathbf{R}\mathbf{S}\mathbf{M}_l = 0, \quad (2.16)$$

$$\mathbf{M}_r^T \mathbf{E}\mathbf{M}_l = 0, \quad (2.17)$$

kde matice

$$\mathbf{E} = \mathbf{R}\mathbf{S} \quad (2.18)$$

se nazývá *esenciální matice*, která díky matici  $\mathbf{S}$  z rovnice (2.14) má také hodnost dva.

## 2.2.4 Fundamentální matice

Body  $\mathbf{m}_r$  a  $\mathbf{M}_r$ , stejně tak jako body  $\mathbf{m}_l$  a  $\mathbf{M}_l$  jsou propojeny vztahem (2.1). Tedy rovnici (2.17) lze zapsat jako

$$\mathbf{m}_r^T \mathbf{E}\mathbf{m}_l = 0, \quad (2.19)$$

kde  $\mathbf{m}_r$  a  $\mathbf{m}_l$  jsou obrazové body na obrazových rovinách. Vzhledem k tomu, že odpovídající si body mohou ležet pouze na odpovídajících epipolárních přímkách,  $\mathbf{E}\mathbf{m}_l$  v rovnici (2.19) je rovnicí epipolární přímky na pravé obrazové rovině, která prochází přes bod  $\mathbf{m}_r$  a, jako všechny epipolární přímky, přes epipól. Proto můžeme obě epipolární přímky vyjádřit jako

$$\mathbf{u}_r = \mathbf{E}\mathbf{m}_l, \quad (2.20)$$

$$\mathbf{u}_l = \mathbf{E}^T \mathbf{m}_r. \quad (2.21)$$

Pro daný bod  $\mathbf{m}_k$ , kde  $k = 1, 2, 3, \dots, N$  je třeba nalézt její souřadnice v pixelech:

$$\mathbf{A}_k \mathbf{m}_k = \tilde{\mathbf{m}}_k, \quad (2.22)$$

kde  $\mathbf{A}_k$  je vnitřní matice kamery pro  $k$ -tý obraz,  $\mathbf{m}_k$  bod v kamerovém souřadnicovém systému a  $\tilde{\mathbf{m}}_k$  je homogenní pixelová souřadnice. Rovnice (2.19) může být napsána jako

$$\left(\mathbf{A}_r^{-1}\tilde{\mathbf{m}}_r\right)^T \mathbf{E}\mathbf{A}_l^{-1}\tilde{\mathbf{m}}_l = 0, \quad (2.23)$$

$$\tilde{\mathbf{m}}_r^T \mathbf{A}_r^{-T} \mathbf{E}\mathbf{A}_l^{-1}\tilde{\mathbf{m}}_l = 0. \quad (2.24)$$

Nakonec je získána rovnice

$$\tilde{\mathbf{m}}_r^T \mathbf{F}\tilde{\mathbf{m}}_l = 0, \quad (2.25)$$



kde matice

$$\mathbf{F} = \mathbf{A}_r^{-T} \mathbf{E} \mathbf{A}_l^{-1} \quad (2.26)$$

se nazývá *fundamentální matice*. Ta popisuje epipolární geometrii v rámci pixelových souřadnic, kdežto esenciální matice v rovnicích (2.17) a (2.19) je použita pro homogenní kamerové souřadnice.

Vidíme tedy, že matice  $\mathbf{F}$  a  $\mathbf{E}$  jsou svázány rovnicí (2.26). Substitucí (2.18) do (2.26) získáme

$$\mathbf{F} = \mathbf{A}_r^{-T} \mathbf{R} \mathbf{S} \mathbf{A}_l^{-1}. \quad (2.27)$$

Vezmeme-li v úvahu, že hodnota matice  $\mathbf{E}$  je dva, hodnota matice  $\mathbf{F}$  bude také dva[6].

## 2.3 Srovnávací metrika

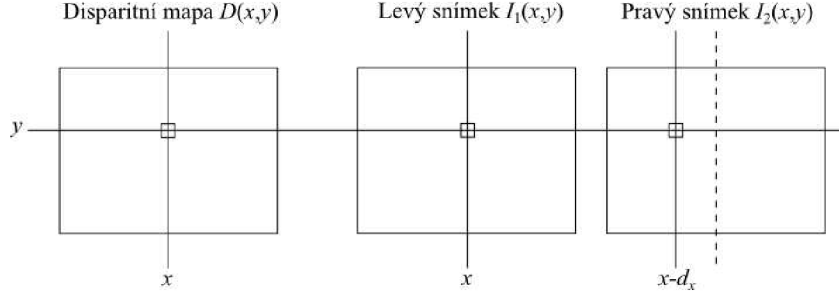
Z výše uvedeného víme, že byt se prohledávaný prostor zmenšil do jedné dimenze, musíme odhadnout, kam se zkoumaný pixel z jednoho obrazu posunul na druhém obrazu. Pokud by pixel odpovídal pixelu pozadí, bude jeho poloha stejná nebo blízká poloze v druhém obraze.

Pro vyjádření vzdálenosti mezi pixely byl zaveden pojem disparita (jedná se o inverzní vyjádření hloubky). Pixely odpovídající si polohou mají nulovou disparitu. Odhad disparity pro každý pixel nám zajistí algoritmy pro tvorbu hloubkové mapy, případně též disparitní mapy. Porovnávání po jednom pixelu by vedlo k velkému rozptylu disparity mezi sousedními pixely a výsledná mapa by neměla valnou vypovídací hodnotu, poněvadž by neměla daleko k šumu, proto se využívá složitějších metod. Tyto metody pracují s celou skupinou (blokem) pixelů. Předpokládá se totiž, že v obraze budou velké či celé plochy s podobným jasem a jen občas se vyskytne nějaký rozdíl, kde nastane změna hloubky v obraze.

V této části analyzujeme tři zástupce nejčastěji používaných srovnávacích metrik, které určují, jak se různé oblasti v různých snímcích liší a přiřadí k nim odpovídající hodnotu disparity.

Pro následující srovnávací metriky předpokládejme, že  $I_1$  a  $I_2$  jsou jasové hodnoty oblastí porovnávaných snímků.  $I_1$  je určen kolem referenčního bodu  $(x, y)$ , vyjádřený v místním souřadnicovém prostoru,  $I_2$  je určen okolo bodu  $(x - d_x, y)$  ve svém místním souřadnicovém prostoru. Pro obě oblasti je definována množina posunů  $U$ , měřených od jejich referenčních bodů, tedy  $(x, y)$  a  $(x - d_x, y)$ . Odpovídající oblasti nemusí být nutně jednolité. Dále předpokládejme, že všechny prvky množiny  $U$  spadají do rozsahů platných pro oblasti  $I_1$  a  $I_2$ . Hodnoty pixelů mohou být skalární, vektorové, maticové nebo také tenzorové, tj. pro všechny druhy matematických objektů, pro které jsou definovány příslušné operátory a normování. Následující metriky jsou jednoznačně definovány pouze pro skalární hodnoty. Navíc

máme zájem získat skalární hodnoty jako výsledek porovnávání. Při porovnávání oblastí se obvykle zajímáme o nalezení centrálních bodů  $(x, y)$  a  $(x - d_x, y)$  anebo hodnotu  $d_x$  pro kterou porovnávací metrika dosahuje extrémní hodnoty.



Obr. 2.4: Hledání podobností

Na obrázku 2.4 vidíme grafickou formu výše uvedeného. Disparitní mapa může v nejjednodušší formě nabývat hodnoty  $D(x, y) = d_x$ . Pro vyjádření podobnosti dvou pixelů, případně bloků se využívá několik druhů metrik[6]. Běžně používané metriky jsou  $D_{\text{SAD}}$ ,  $D_{\text{SSD}}$  a jako třetí varianta byla zvolena  $D_{\text{NCC}}$ .

### 2.3.1 Sum of absolute differences

Zkráceně též SAD, znamená to suma absolutních rozdílů. Tento algoritmus je z uvedených algoritmů výpočetně nejméně náročný, proto se používá tam, kde je prioritou rychlý výpočet nebo kde není k dispozici velký výpočetní výkon.

$$D_{\text{SAD}} = \sum_{i \in U} |I_1(x + i) - I_2(x + d_x + i)| \quad (2.28)$$

### 2.3.2 Sum of squared differences

Značeno zkratkou SSD, což znamená suma kvadratických rozdílů. Můžeme si všimnout jisté podobnosti s SAD, je však rozšířen o kvadratickou část. Metoda je citlivá na změnu jasu a kontrastu, což se může projevit např. na dvojici obrazů, které mají různou střední hodnotu jasu.

$$D_{\text{SSD}} = \sum_{i \in U} (I_1(x + i) - I_2(x + d_x + i))^2 \quad (2.29)$$

### 2.3.3 Normalized cross-correlation

Zkráceně též NCC, značí normalizovanou křížovou korelaci, ve statistice často nazývanou jako lineární korelační koeficient. Ten má vítanou vlastnost invariance vůči

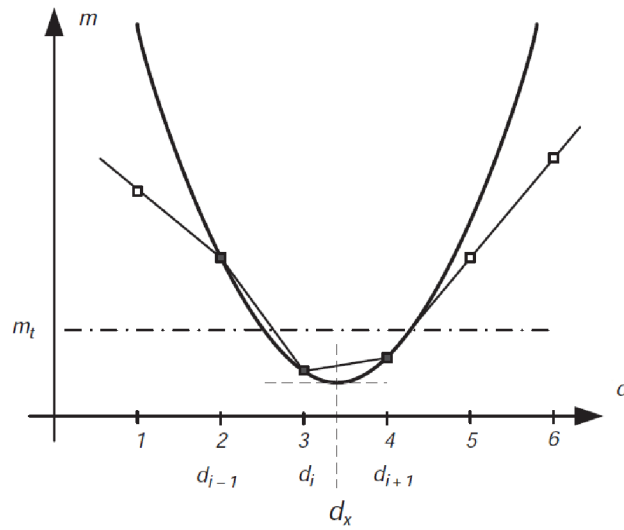
lokálním změnám kontrastu a jasů.

$$D_{\text{NCC}} = \frac{\sum_{i \in U} (I_1(x+i) - \overline{I_1(x)}) \cdot (I_2(x+d_x+i) - \overline{I_2(x+d_x)})}{\sqrt{\sum_{i \in U} (I_1(x+i) - \overline{I_1(x)})^2 \cdot \sum_{i \in U} (I_2(x+d_x+i) - \overline{I_2(x+d_x)})^2}} \quad (2.30)$$

## 2.4 Subpixelová přesnost

Disparity jsou počítány jako rozdíly mezi polohami odpovídajících bodů. Vzhledem k tomu, že tyto polohy jsou omezeny na mřížku obrazových bodů, mohou vypočítané disparity nabývat pouze celočíselných hodnot. Ovšem v některých situacích, například při použití snímků s nízkým rozlišením, to vede k nadměrným chybám při rekonstrukci prostoru.

Jeden ze způsobů, jak zmírnit tento problém je využít tvar většího počtu vypočtených hodnot srovnávací metriky. Nejběžnější technikou zvýšení přesnosti je proložení těchto hodnot parabolou, kde v minimu paraboly nalezneme novou hodnotu disparity, nyní se subpixelovou přesností.



Obr. 2.5: Subpixelový odhad hodnoty disparity[6]

Obrázek 2.5 ukazuje příklad proložení vypočtených hodnot parabolou, kde čtveřky znázorňují celočíselné hodnoty disparit získaných z některé ze srovnávacích metrik a jejich minimum je označeno jako  $d_i$ . Naším úkolem je vzít  $d_i$  a jeho dva nejbližší sousedy, tj.  $d_{i-1}$  a  $d_{i+1}$ , a dosadit je do polynomu třetího řádu. Z takto doplněného polynomu získáme novou hodnotu  $d_x$ , která již není vázána na obrazovou mřížku. Přesněji, získáme tři páry hodnot:

$$\{d_{i-1}, m_{i-1}\}, \{d_i, m_i\}, \{d_{i+1}, m_{i+1}\}, \quad (2.31)$$

kde  $m_i = m(d_i)$  je odpovídající hodnota pro posunutí  $d_i$ . Na jejich základě určíme koeficienty polynomu třetího řádu daného jako

$$ad_i^2 + bd_i + c = m_i, \quad (a \neq 0). \quad (2.32)$$

Bod minima  $d_x$  najdeme dvojnásobnou diferencí (2.32) podle  $d_i$ , pak se rovnice první derivace rovná nule za podmínky:

$$2ad_x + b = 0 \quad \text{a} \quad 2a > 0, \quad (2.33)$$

tedy

$$d_x = -\frac{b}{2a} \quad \text{a} \quad a > 0. \quad (2.34)$$

Vložením (2.31) do (2.32) získáme tyto tři rovnice

$$\begin{cases} ad_{i-1}^2 + bd_{i-1} + c = m_{i-1} \\ ad_i^2 + bd_i + c = m_i \\ ad_{i+1}^2 + bd_{i+1} + c = m_{i+1} \end{cases}, \quad (2.35)$$

kteří vyřešíme pro  $a$  a  $b$ . Je však snazší vypočítat  $b/a$  pomocí Cramerova pravidla. Dále může být rovnice zjednodušena tím, že  $d_{i-1} = -1$ ,  $d_i = 0$  a  $d_{i+1} = 1$ . To odpovídá posunutí počátku souřadného systému do bodu  $(d_i, 0)$ , který bude spočten na konci výpočtů. Tedy

$$\frac{b'}{a'} = -\frac{\begin{vmatrix} 1 & m_{i-1} & 1 \\ 0 & m_i & 1 \\ 1 & m_{i+1} & 1 \end{vmatrix}}{\begin{vmatrix} -1 & m_{i-1} & 1 \\ 0 & m_i & 1 \\ 1 & m_{i+1} & 1 \end{vmatrix}} = -\frac{m_{i-1} - m_{i+1}}{m_{i-1} - 2m_i + m_{i+1}}, \quad (2.36)$$

kde  $a'$  a  $b'$  jsou v posunutém souřadném systému. Nyní rovnici (2.36) vložíme do (2.33), resp. (2.34) a posunutí  $d_i$ , získáme tak subpixelovou disparitu  $d_x$ :

$$d_x = d_i + \frac{m_{i-1} - m_{i+1}}{2(m_{i-1} - 2m_i + m_{i+1})}, \quad (2.37)$$

pro kterou je předpokládáno, že

$$(m_{i-1} - m_i) + (m_{i+1} - m_i) > 0, \quad (2.38)$$

kteří je jednoduše důsledkem předpokladu  $a > 0$ .

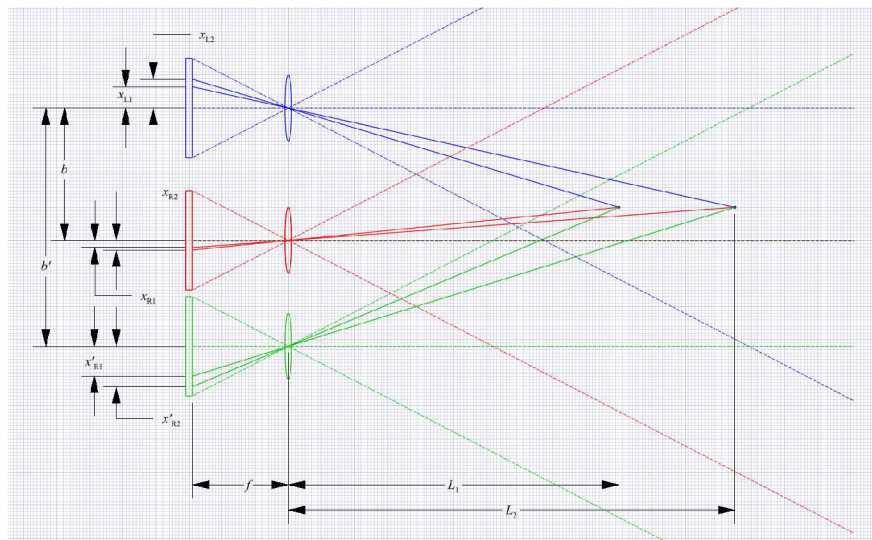
Výše uvedeným postupem je získána reálná hodnota disparity s vyšší přesností přesahující přesnost disparity vypočtenou některou ze srovnávacích metrik využívajících celočíselné hodnoty pixelů vázaných na obrazovou mřížku.

## 2.5 Změna vzájemné polohy a úhlu kamer

Z dřívějších kapitol víme, že s každou změnou vzájemné polohy kamer se změní vnější parametry kamer, vždy tak změni jejich kalibraci a parametry kalibrace je třeba znovu přepočítat pro novou polohu. Z toho vyplývá, že po zkalibrování kamer musíme zamezit dalšímu pohybu kamer. Dále se změni podmínky triangulace (příp. podobnosti trojúhelníků), kdy se změni dosah a přesnost triangulace. Dále si ukážeme vliv změny polohy na tvorbu disparitní mapy. Nebudeme se zde zabývat složitými matematickými rozbory nebo úvahami, pouze si názorně ukážeme dopady těchto změn.

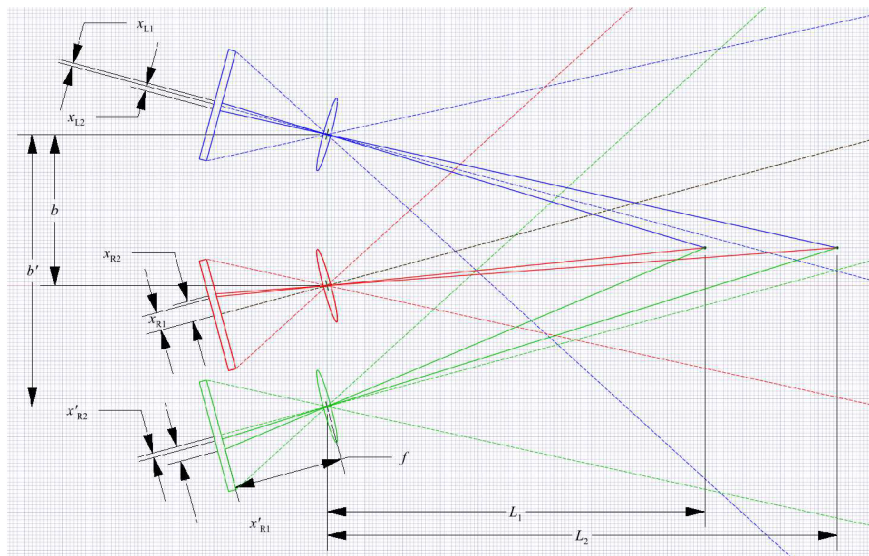
### 2.5.1 Vliv změn poloh kamer na měření vzdálenosti

Abychom následující znázornění pochopili, popíšeme si následující situaci. Levá kamera s fixní polohou je modrá. Pravá kamera před změnou polohy je červená a po změně polohy zelená. Přerušovanou čarou jsou znázorněny osy kamer a jejich zorná pole. Plnou čarou jsou uvedeny paprsky znázorňující promítání bodů zájmů na senzory kamer.  $x_{L1}, x_{L2}, x_{R1}, x_{R2}, x'_{R1}, x'_{R2}$  jsou jednotlivé vzdálenosti od os jednotlivých kamer k místu promítání bodů na senzorech.  $b$  a  $b'$  jsou vzdálenosti mezi kamerami před a po změně polohy.  $f$  je ohnisková vzdálenost.  $L_1$  a  $L_2$  jsou vzdálenosti od bodů zájmů ke kamerám.



Obr. 2.6: Situační náčrtek posunu jedné z kamer do strany

V prvním případě je znázorněn vliv rovnoběžného posunu kamer, jenž je zobrazen na Obr. 2.6. V tomto případě, kdy osy objektivů jsou rovnoběžné, nemůžeme



Obr. 2.7: Situační nákres vzájemně natočených kamer a posunu jedné z kamer do strany

přesně určit hranici nulové disparity. Teoreticky se tato hranice limitně blíží nekonečnu. Avšak vlivem digitalizace se ztrácí mnoho informací, kde míra ztrát je určena především rozlišovací schopností a počtem kvantovacích hladin kamer. Hranice nulové disparity se při použití běžné webové kamery nachází v řádu jednotek až desítek metrů od kamer. Hranici maximální disparity můžeme určit v momentě, kdy se sledovaný objekt ztratí ze zorného pole jedné z kamer. Jinak řečeno se tato hranice vyskytuje v místě protnutí hranic zorného pole obou kamer. Zvýšením odstupů obou kamer oddálíme hranici s maximální disparitou. Dále si můžeme všimnout, že změnou vzdálenosti z  $b$  na  $b'$  došlo ke změně vzdáleností z  $x_{R1}, x_{R2}$  na  $x'_{R1}, x'_{R2}$ . Tato změna je odpovídající reakcí podle rovnice o podobnosti trojúhelníků

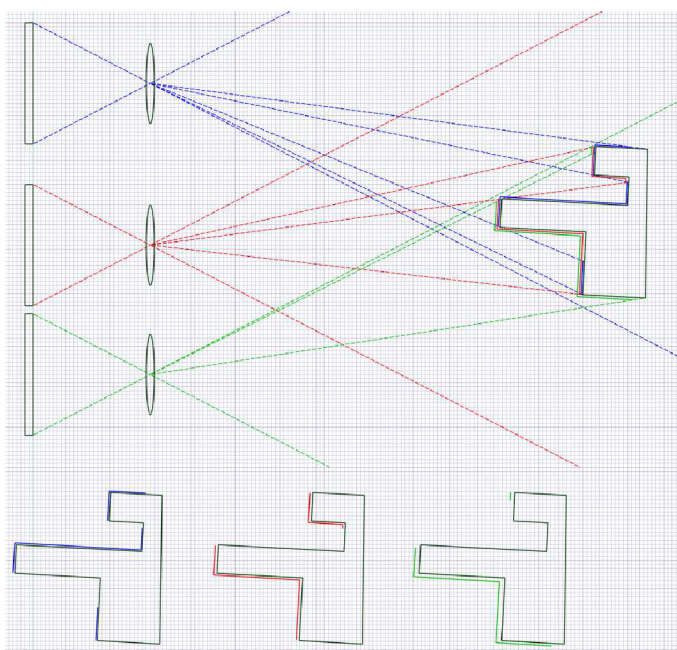
$$\frac{L}{b} = \frac{f}{x_R - x_L}, \quad (2.39)$$

o které se více dozvíme v následující kapitole.

V druhém případě, na Obr. 2.7, byl změněn vzájemný úhel kamer a osy objektivů se protínají. Rovněž i hranice zorných polí obou kamer se protínají na více místech a určují nám jak hranici maximální disparity, tak hranici nulové disparity. Protože známe polohy hranic disparity, můžeme přesněji určit vzdálenost objektu od kamer, ale jen v intervalu těchto hranic. Zvýšením odstupů se tento interval zvětší. Oproti předchozímu případu je zde účelněji využita sensorová plocha. Můžeme si zde všimnout, že poměry vzdáleností si neodpovídají jako v rovnici 2.39, to však i v případě nahrazení rozdílu  $x_R - x_L$  disparitou  $D$  vadí, neboť i disparita je vázaná na pozici pixelu na senzoru kamer.



Na Obr. 2.8 se nachází vyznačené oblasti snímaného předmětu zachycené jednotlivými kamerami. Můžeme si všimnout, že ani jedna z kamer nezachycuje celý čelní obrys předmětu. To je dáno polohou předmětu vůči kamerám a především geometrickou povahou předmětu. Zjednodušeně, jedna kamera nevidí to samé co druhá, protože jí v tom brání nějaký výčnělek snímaného předmětu nebo kamery snímají různé části předmětu, každá z jiné strany, která s předchozí stranou není podobná. Důležité ovšem pro srovnávací metriku a tedy tvorbu disparitní mapy je, že se vyznačené oblasti z větší či menší části překrývají a mají vzájemnou korelaci. Změnou vzájemného odstupu kamer anebo polohy a natočení předmětu lze dosáhnout lepšího překrytí oblastí a poskytnout tak srovnávací metrice výhodnější snímky.



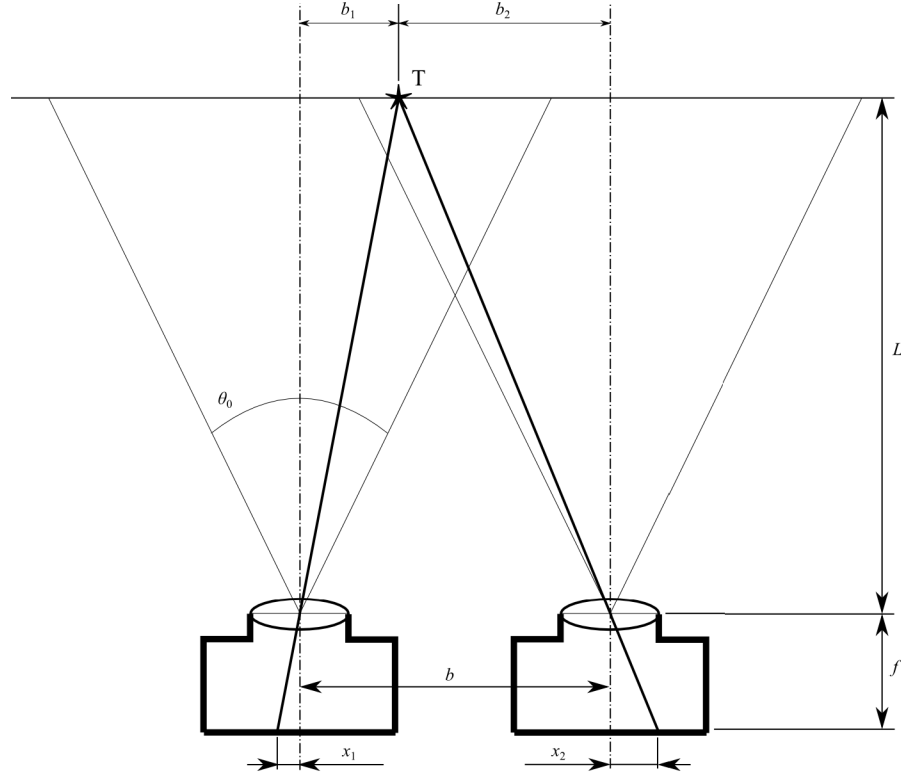
Obr. 2.8: Vliv změny polohy na disparitu

## 2.6 Určení vzdálenosti

Nyní se dostáváme k meritu věci, tím je měření vzdálenosti. Mnoho článků a návodů, které lze nalézt na internetu, se spoléhalo na změření závislosti disparity na vzdálenosti a z průběhu hodnot byly určeny rovnice pro převod z hodnot disparity na reálnou hodnotu vzdálenosti, buď empirickým určením, proložením známou křivkou nebo regresní analýzou.

Další varianta měření vzdálenosti, nalezená v akademickém periodiku[10] a použitá použita v této práci, se zakládá na podobnosti trojúhelníků, resp. triangulace.

Některé detaily o triangulaci můžeme nalézt v předchozí kapitole. Na obrázku 2.9



Obr. 2.9: Diagram zachycení objektu kamerami v kanonické konfiguraci

můžeme vidět použití podobnosti trojúhelníků, kde  $\mathbf{T}$  je zachycovaný objekt,  $b$  je vzdálenost mezi kamerami,  $f$  je vzdálenost od snímáče k objektivu nebo také ohnisková vzdálenost,  $L$  je vzdálenost od objektu k objektivu,  $b_1$  a  $b_2$  jsou vzdálenosti od objektu  $\mathbf{T}$  k ose levé, resp. pravé kamery.  $\theta$  je zorný úhel. Vzdálenosti  $b$ ,  $b_1$ ,  $b_2$  a  $L$  uvažujeme v reálných jednotkách.  $x_1$  a  $x_2$  jsou vzdálenosti od os jednotlivých kamer k bodu promítání objektu  $\mathbf{T}$  na senzoru odpovídající kamery, tyto vzdálenosti jsou v pixelech. Vzdálenost  $f$ , resp. ohnisková vzdálenost, nemusí být v reálných jednotkách, pojí však reálnou vzdálenost k velikosti pixelů. Podobnost trojúhelníků je vyjádřena rovnicemi

$$\frac{b_1}{L} = \frac{-x_1}{f}, \quad \frac{b_2}{L} = \frac{x_2}{f}. \quad (2.40)$$

Jelikož  $b = b_1 + b_2$ , pak

$$b = \frac{L}{f} (x_2 - x_1). \quad (2.41)$$

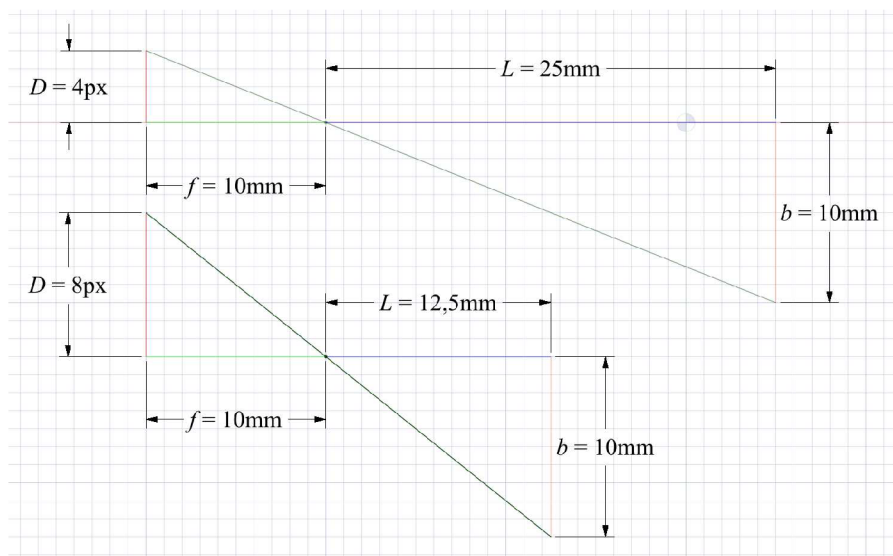
A protože nás zajímá  $L$ , bude rovnice vypadat takto:

$$L = \frac{bf}{x_2 - x_1}. \quad (2.42)$$



Na závěr můžeme rozdíl  $x_2 - x_1$  nahradit disparitní hodnotou v pixelech  $D$  získanou z některé výše popsaných srovnávacích metrik

$$L = \frac{bf}{D}. \quad (2.43)$$



Obr. 2.10: Příklad triangulace

Na obrázku 2.10 je příklad použití rovnice (2.43). Jednotlivé proměnné byly popsány v předchozím odstavci.

## 3 REALIZACE

K vývoji softwaru byl využit běžný notebook, který neoplývá současnými moderními akceleračními technologiemi (CUDA, OpenCL). Tyto technologie ani nebyly v žádné míře implementovány a využity v softwaru. Má to sice tu nevýhodu, že veškeré výpočty provádí procesor a od něj se odvíjí rychlost výpočtů. Software tak není vázaný na jakoukoli technologii, která se rychle objevila a může stejně tak rychle zaniknout nebo být nahrazena jinou.

### 3.1 Kamery

Bylo rozhodnuto jít nejdostupnější a levnou cestou. Byly pořízeny dvě kamery Sweex Webcam WC035V2, jejichž technické specifikace lze najít na [5]. Jedná se o obyčejné, tzv. generické webové kamery, které lze pořídit pod různými značkami od různých prodejců, zároveň splňují minimální standardy, s nimiž je umožněno k těmto kamerám přistupovat jedním generickým ovladačem. Zároveň je však třeba poznamenat, že generická kamera neznamena nutně zařízení vyrobené podle jednoho vzoru a s identickými parametry, ale o způsob zpracování obrazu a komunikace tak, aby byl generický ovladač schopen tyto požadavky splnit. V praxi to znamená, že trh je zaplaven relativně levnými zařízeními, která mají víceméně stejné parametry, ale jejich zásadní výhodou je, že jsou doslova „*Plug and play*”. Uživatel, resp. programátor, tak nemusí řešit možné komplikace s instalací ovladačů a podružného softwaru a zároveň to usnadňuje vývoj softwaru.



Obr. 3.1: Použitá kamera Sweex webcam WC035V2[5]

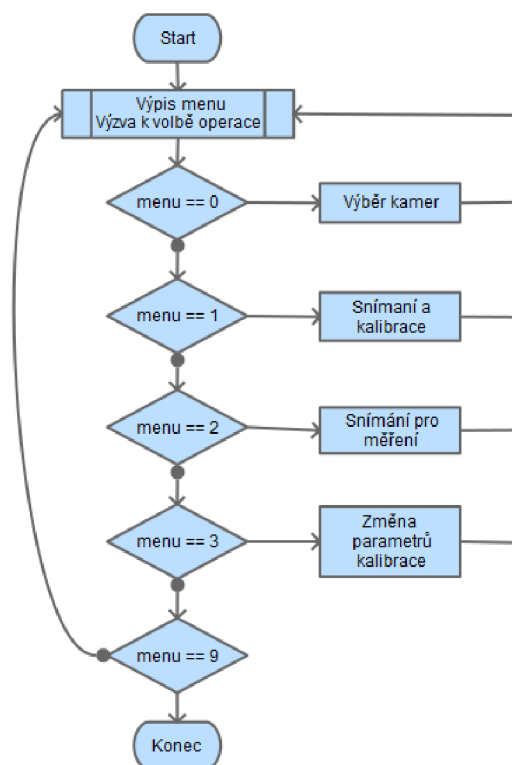
### 3.2 Software

Software byl vyvíjen ve vývojovém prostředí Microsoft Visual Studio 2010 a doplněný o otevřenou knihovnu pro počítačové vidění OpenCV[3].

Software byl pro snazší programování rozdělen do několika částí. Jednotlivé části se starají o:

- Zobrazení menu a nastavení vstupních parametrů
- Výběr z aktuálně dostupných kamer
- Kalibraci kamer
- Tvorbu disparitní mapy, subpixelovou přesnost a určení vzdálenosti

Abstraktně si tyto části můžeme představit na zjednodušeném diagramu na Obr. 3.2



Obr. 3.2: Zjednodušený diagram struktury softwaru

### 3.2.1 Zobrazení menu a nastavení vstupních parametrů

Na diagramu (Obr. 3.2) vidíme, že hlavní smyčka se stará o vypsání nabídky operací a vyzývá k výběru některé z operací, která se následně vykoná. Dále se tato část stará o obsluhu položky *Změna parametrů kalibrace*, kde se nastavují rozměry kalibrační šachovnice a rozteč kamer.

### 3.2.2 Výběr z aktuálně dostupných kamer

Tato část softwaru nalézají všechny připojené kamery, interní nebo dodatečně připojené přes USB. Právě větší počet připojitelných kamer neumožňuje fixní určení kamer, která z kamer je levá, resp. pravá. Každá z kamer však po připojení do počítače získává speciální identifikátor, který je v OpenCV zjednodušen na číselný index, který se s každou další připojenou kamerou inkrementuje. Určení levé či pravé kamery je prováděno v součinnosti s uživatelem. A to tak, že mu jsou postupně zobrazovány živé náhledy jednotlivých kamer a uživatel potom klávesovým vstupem zadá, která je levá, pravá nebo se má vynechat. Toto nastavení se udržuje po dobu aktivní aplikace, s dalším spuštěním aplikace je třeba výběr kamer provést znovu.

### 3.2.3 Kalibrace kamer

Tato část obstarává párové nasnímání obrazů z kamer během předkládání kalibrační šachovnice. Následně proběhne detekce šachovnicového vzoru, zároveň dochází k odstranění těch párů snímků, kde zcela nebo částečně chybí šachovnicový vzor. Se zbylými korektními snímky proběhne kalibrace kamer a rektifikace podle teorie z kapitoly 2.2.3 a 2.2.4. Po ukončení výpočtu kalibračních hodnot může uživatel vizuálně zkontrolovat zkalibrované a rektifikované snímky. Kalibrační hodnoty se ukládají do souboru pro použití v další části softwaru a zároveň jsou kalibrační hodnoty zachovány i pro další spuštění aplikace.

### 3.2.4 Tvorba disparitní mapy, subpixelová přesnost a určení vzdálenosti

Poslední část softwaru aplikuje kalibrační hodnoty na aktuální obrazy z kamer. Uživatel tak na obrazovce počítače vidí opravené a rektifikované obrazy z kamer. Během náhledu je vypočítávána disparitní mapa, disparitní mapa se subpixelovou přesností, jejichž obecné principy nalezneme v kapitole 2.3 a 2.4, a přepočítání z disparity na vzdálenost (viz kap. 2.6).

Z programátorského hlediska lze výpočet disparitní mapy provádět dvěma způsoby. První, obecný způsob provede výpočet metrik a jejich agregaci v jednom kroku. Tento postup se používá převážně pro GPU výpočty, protože je méně náročný na paměť, ale je pomalejší na výpočet. Druhý způsob provádí výpočet metrik a agregaci odděleně. V jednom kroku se provede výpočet metrik a v druhém agregace. Tento postup je náročnější na paměť, protože si musí v paměti ponechat výsledky z výpočtů metrik pro agregační krok, ale zato je rychlejší, protože každý krok provádí monotónní operace.

Otevřené knihovny OpenCV[3] nabízí hned tři metody tvorby disparitní mapy. Jsou to Block matching, Semiglobal block matching a Variational matching. Hlavní rysy jednotlivých metod budou stručně shrnuty v následujících odstavcích.

### **Block matching**

Tato metoda rozdělí referenční obraz stereopáru do makrobloků, které jsou posléze v jednom směru posouvány a porovnávány s druhým obrazem stereopáru. Porovnávání úseky jsou vyhodnocovány pomocí součtu absolutních rozdílů. Z makrobloků, majících nejnižší součet absolutních rozdílů či součet kvadratických rozdílů, jsou získány jednotlivé disparity makrobloků, souhrnem disparit ze všech makrobloků je získána disparitní mapa. Jedná se o nejjednodušší variantu tvorby disparitní mapy, která je výpočetně nenáročná. Metoda je však citlivá na kvalitu provedené rektifikace a rozdílou střední hodnotu celkového jasů obrazů stereopáru.

### **Semiglobal block matching**

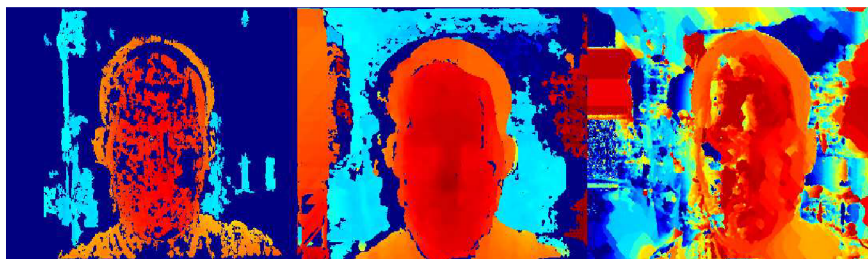
Zatímco Block matching metoda prohledává shodné makrobloky pouze v jednom směru, Semiglobal block matching metoda může prohledávat makrobloky až v osmi směrech. Vyhodnocování makrobloků se provádí pomocí váhy cest pro každý směr. Disparita je určena jako cesta s nejmenší vahou. Na závěr se provádí disparitní zpřesnění, kde je kontrolována konzistence mapy, jsou odstraněny extrémní hodnoty a interpolací jsou dopočítávána prázdná místa v mapě. Metoda poskytuje kvalitní výstup za cenu vyšší výpočetní a paměťové náročnosti.

### **Variational matching**

Varianční metoda je globální metodou v tom smyslu, že se snaží minimalizovat energii funkce. Metoda je varianční, protože využívá variančního počtu, a to zejména Euler-Lagrangeovy rovnice. Varianční metody jsou často nasazovány pro řešení problému optického toku, kde se odhaduje pole posunutí mezi dvěma obrazy. Optický tok je dvourozměrný, zatímco stereo korespondence obvykle předpokládá, že pole posunutí je jednorozměrné. Jelikož metoda využívá diferenciálního a intergrálního počtu ve dvou rozměrech, je výpočetně a paměťově nejnáročnější z výše uvedených metod.

Na Obr. 3.3 můžeme porovnat rozdíly disparitních map vytvořených jednotlivými metodami. Tyto mapy byly normovány a pro větší názornost převedeny do barevné mapy JET[12]. Z obrázku je zřejmé, že výstup metody Semiglobal block matching vytváří souvislejší plochy bez prudkých změn a celkově podává hloubku nejméně.

A právě způsob tvorby disparitní mapy, její zpřesnění a kvalitní výstup činí metodu Semiglobal block matching vhodnou k implementaci do softwaru.



Obr. 3.3: Ukázka disparity BM, SGBM a VAR

Pro lepší pochopení principu tvorby disparitní mapy byla vytvořena vlastní implementace tvorby disparitní mapy na bázi Block matching metody (viz výše). Algoritmus této metody můžeme rozdělit do několika kroků:

1. Matching
2. Agregace
3. Vyhledání

### Matching

Pro matching zde byly implementovány metriky SAD, SSD i NCC, které jsou detailně popsány v kapitole 2.3. Avšak pro zrychlení algoritmu je agregační část rovnic prováděná odděleně v dalším kroku. Vyjímkou je metrika NCC, kde nelze agregační část snadno oddělit. Pro daný okamžik lze vybrat a pracovat jen s jednou z metrik. Algoritmus oba vstupní obrazy prochází pixel po pixelu na stejných souřadnicích obou obrazů, jejich jasová složka je porovnána některou z metrik a výsledná hodnota metriky je zapsána do diferenční mapy na stejných souřadnicích. Takto se postupuje přes všechny pixely obrazů. Následně se jeden z obrazů posune o jeden pixel a celý proces porovnávání obrazu se opakuje. Pro každé posunutí je výstupní diferenční mapa opatřena patřičným indexem odpovídajícím posunu obrazů. Algoritmus matchingu si předvedeme na úryvku kódu funkce SAD metriky:

```
Mat SAD(int disparita, Mat levy, Mat pravy)
{
    Mat temp;
    temp.create(levy.rows,levy.cols,CV_64FC1);
    temp.setTo(0);

    for(int i=0;i<levy.rows;i++)
    {
        for(int j=0;j<levy.cols;j++)
        {
```

```

        if(j-disparita>=0)
        {
            temp.at<double>(i,j)=abs(levy.at<uchar>(i,j)-
            pravy.at<uchar>(i,j-disparita));
        }
    }
}
return temp;
}

```

Výstup tohoto kroku si můžeme představit jako trojrozměrnou matici hodnot, kde osy  $x$  a  $y$  značí souřadnice diferenční mapy a osa  $z$  značí posun, resp. disparitu vstupních obrazů.

### **Agregace**

Agregace provádí řádkové sčítání hodnot diferenčních map v nastavené délce okna. Zjednodušeně vzato hodnota daného pixelu agregované diferenční mapy je součtem hodnot stejného pixelu a následujících hodnot pixelů sousedících z jedné strany na stejném řádku v délce nastaveného agregáčního okna. Lepší představu o agregaci poskytne příklad v podobě kódu funkce agregátoru:

```

Mat Agregator(Mat vstup, int okno)
{
    Mat tempA;
    tempA.create(vstup.rows, vstup.cols, CV_64FC1);

    double aggreg = 0;
    for(int r = 0; r < vstup.rows; r++)
    {
        for(int s = 0; s < vstup.cols - okno; s++)
        {
            for(int ok = 0; ok < okno; ok++)
            {
                aggreg += vstup.at<double>(r, s + ok);
            }
            tempA.at<double>(i,j) += aggreg;
            aggreg = 0;
        }
    }
    return tempA;
}

```

Z kódu funkce si můžeme všimnout, že vstupem je dvourozměrná matice, provede se tedy agregace pro jednu disparitu. Proto musíme tuto funkci volat iterativně pro každou disparitu diferenčních map.

### Vyhledání

Krok vyhledání prochází trojrozměrnou maticí agregovaných diferenčních map v ose  $z$ , tedy skrze disparitu, a vyhledává nejmenší agregovanou difereční hodnotu. V místě nález nejmenší agregované difereční hodnoty nyní známe index disparity a tento index přeneseme do dvourozměrné matice na odpovídající souřadnice. Pokud toto provedeme přes všechny souřadnice v osách  $x$  a  $y$ , získáme disparitní mapu s pixelovou přesností.

Při použití metriky NCC je postup obdobný, ale vyhledává se největší agregovaná hodnota diferenčních map. Je to dáno opačnou logikou výstupu metriky NCC oproti metrikám SAD a SSD. Kód tohoto kroku vypadá následovně:

```
double nejmensi=DBL_MAX;
if(NCCFlag==1)
{
nejmensi=DBL_MIN;
}
int indexNejmensiho=0;
for(int i=0;i<mapa[0].rows;i++)
{
    for(int j=0;j<mapa[0].cols;j++)
    {
        for(int disp=dmin;disp<=dmax;disp++)
        {
            if(NCCFlag==1)
            {
                if(mapa_aggr[disp].at<double>(i,j)>nejmensi)
                {
                    nejmensi=mapa_aggr[disp].at<double>(i,j);
                    indexNejmensiho=disp;
                }
            }
            else
            {
                if(mapa_aggr[disp].at<double>(i,j)<nejmensi)
                {
                    nejmensi=mapa_aggr[disp].at<double>(i,j);
```

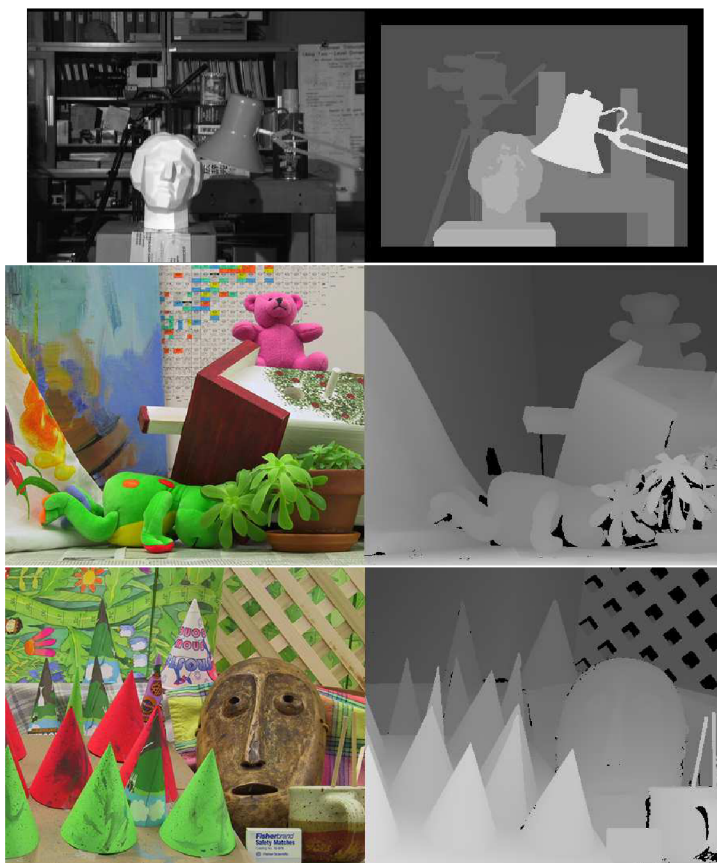


```

        indexNejmensiho=disp;
    }
}
}
hlmapa.at<uchar>(i,j)=indexNejmensiho;
nejmensi=DBL_MAX;
if (NCCFlag==1){nejmensi=DBL_MIN;}
indexNejmensiho=0;
}
}

```

Nyní se můžeme podívat na porovnání jednotlivých metrik. Porovnání metrik probíhá na základě výpočtu chyby od tzv. ground truth disparitní mapy. K porovnávání byly použity obrazové sady Tsukuba, Teddy a Cones, které můžeme vidět na Obr. 3.4 [14].



Obr. 3.4: Obrazový set Tsukuba s ground truth disparitní mapou

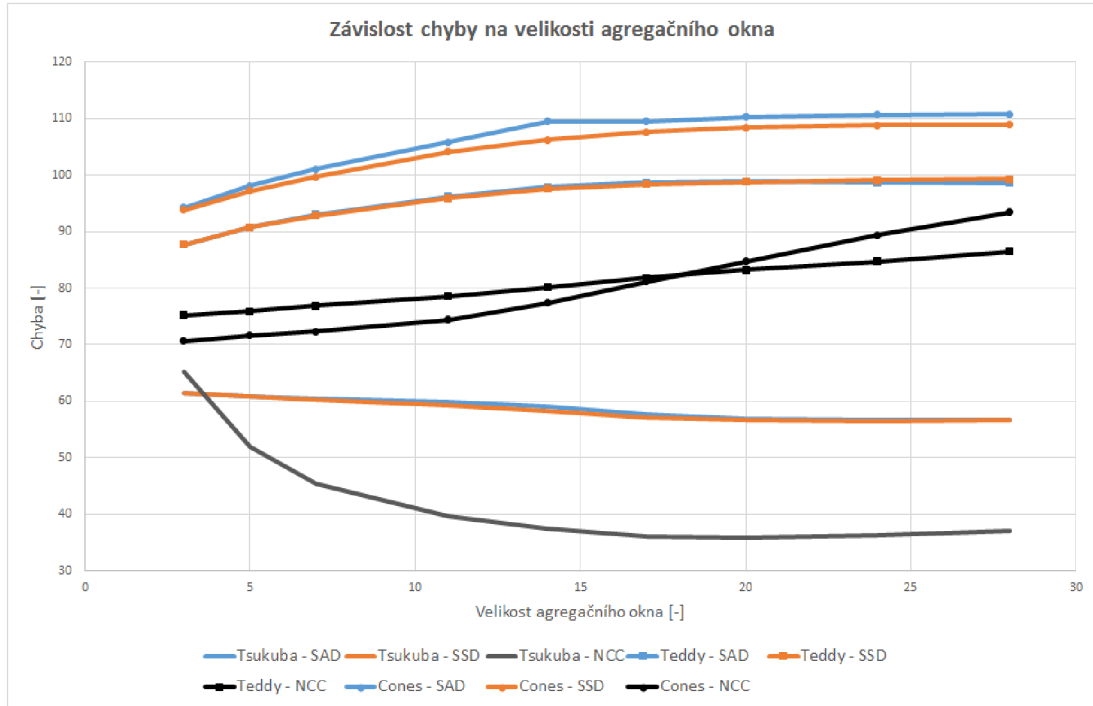
Rovnice výpočtu chyby, která vyjadřuje odlišnost vypočtené disparitní mapy a ground

truth disparitní mapy, vypadá takto:

$$Chyba = \frac{\sum_{i=0}^P |A_i - B_i|}{P}, \quad (3.1)$$

kde  $A_i$  značí vypočtenou disparitu,  $B_i$  ground truth disparitu a  $P$  pak celkový počet pixelů v disparitní mapě.

Chyby jednotlivých metrik byly vypočteny pro různé velikosti agregačních oken, zaneseny do tabulky v příloze A.3 a znázorněny v grafu na Obr. 3.5. V grafu



Obr. 3.5: Graf závislosti chyby na velikosti agregovačního okna

(Obr. 3.5) si můžeme všimnout, že všechny průběhy chyb se mají s rostoucím agregačním oknem od okna velikosti 20 tendenci ustálovat. Další zvětšování agregačního okna nevede k výraznějším změnám v chybě. U metrik SAD a SSD vidíme, že jejich průběhy chyb jsou si podobné, neboť výpočet jejich metrik má stejný základ lišící se jen o kvadratickou část. Dále vidíme, že průběhy chyb metriky NCC dosahuje výrazně nižších chyb než u metrik SAD a SSD. To je ale vykoupeno výrazně vyšší výpočetní složitostí (rovnice 2.30) než u metrik SAD (rovnice 2.28) a SSD (rovnice 2.29), tudíž je z těchto metrik nejpomalejší.

Dalším úkolem této části je výpočet disparitní mapy se subpixelovou přesností. K výpočtu subpixelové přesnosti je použita metoda prokládání parabolou, která je uvedená v kapitole 2.4. Pro výpočet disparity se subpixelovou přesností je zapotřebí aktuální pixelová disparita a agregované diferenční hodnoty z aktuální disparity

a ze dvou nejbližších sousedních disparit. Výsledná disparitní mapa se subpixelovou přesností disponuje hodnotami v rozsahu reálných čísel. Úryvek kódu funkce výpočtu subpixelové přesnosti nám opět poskytne lepší představu:

```
double SubPixPres(int di, double m0, double m1, double m2)
{
    double dx=0.0;
    if(NCCFlag==1) //u NCC je "logika opacneho extremu"
    {
        m0=1-m0;
        m1=1-m1;
        m2=1-m2;
    }
    if (((m0-m1)+(m2-m1))>0.0) //je splnena podminka dle rovnice 2.38?
    {
        dx = di + ((m0 - m2)/(2.0 * (m0 - 2.0*m1 + m2)));
    }
    else //pri nesplneni podminky skopiruj hodnotu
    {
        dx=double(di);
    }
    return dx;
}
```

Posledním úkolem této části je určení vzdálenosti. Obecně se provádí přepoččet podle rovnice (2.43) uvedené v kapitole 2.6. Tato rovnice ovšem vyžaduje ohniskovou vzdálenost  $f$ , kterou doposud neznáme a výrobci se ve většině případů neobtěžují uvádět detailní technické specifikace kamer[5]. Naštěstí ohniskovou vzdálenost  $f$  můžeme získat z reprojekční matice  $Q[2]$ , kterou lze sestavit z kalibračních parametrů.

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{-1}{T_x} & \frac{(c_x - c'_x)}{T_x} \end{bmatrix} \quad (3.2)$$

Stále však nezískáváme vzdálenost v reálných jednotkách, protože právě takto získaná ohnisková vzdálenost  $f$  není v reálných jednotkách, pouze pojí reálnou vzdálenost k velikosti pixelů. Výsledkem je upravená rovnice (2.43)

$$L = \frac{bf}{D} \cdot K, \quad (3.3)$$

kde  $K$  je korekční hodnota doplňující ohniskovou vzdálenost  $f$  o vazbu na reálné jednotky. Korekční hodnota  $K$  je získána pokusem, kdy je měřen objekt ve známé vzdálenosti a výpočtem podle rovnice (2.43) je získána hodnota bez vazby na reálné jednotky. Podílem vypočtené a skutečné hodnoty získáme korekční hodnotu  $K$ . Pro zvýšení přesnosti je korekční hodnota průměrem pěti pokusů. Poněvadž se ohnisková vzdálenost mění s každou kalibrací kamer, je nutné získat novou korekční hodnotu po každé kalibraci.

Další a lepší varianta získání ohniskové vzdálenosti je využití rovnice (2.43), kde dosadíme známé proměnné  $b$ ,  $D$  a za  $L$  dosadíme vzdálenost k objektu, o kterém víme, že se nachází v uvedené vzdálenosti  $L$ . Rovnice pro určení ohniskové vzdálenosti pak vypadá takto

$$f = \frac{LD}{b}. \quad (3.4)$$

Takto získaná ohnisková vzdálenost má vazbu k reálné vzdálenosti a není třeba jí pomáhat korekcí jako u předchozí varianty získání ohniskové vzdálenosti.

### 3.3 Aplikace

Aplikace byla vytvořena s využitím výše popsaných metod, funkcí a poznatků. Aplikace pro svůj chod vyžaduje dynamické knihovny zkompileované ze zdrojového kódu otevřené knihovny OpenCV[3] se zapnutou podporou Qt Frameworku[13]. Cílem aplikace je demonstrovat fungující určení vzdálenosti a ověřit přesnost měření. V aplikaci je dále možné si vybrat použitou disparitu, resp. vybrat si vlastní implementaci disparity nebo SGBM metodu z OpenCV.

Návod k použití aplikace nalezneme v příloze C.

### 3.4 Měření

V následující části se dozvíme o metodice měření, kvůli zajištění stejných podmínek pro měření. Vlastní měření je rozdělené do čtyř částí, kde se mění rozteč a vzájemné natočení kamer. Měří se závislost vypočtené vzdálenosti z pixelové a subpixelové disparity na skutečné vzdálenosti.

#### 3.4.1 Metodika měření

Aby byly zajištěny konzistentní výsledky měření všech poloh kamer, byl určen pracovní postup a prostor pro měření. Než bylo dospěno k vhodnému pracovnímu postupu a vhodnému prostoru pro měření, bylo zjištěno, jaké faktory do měření vstupují. Jejich vlivům je vyhrazena zvláštní kapitola.

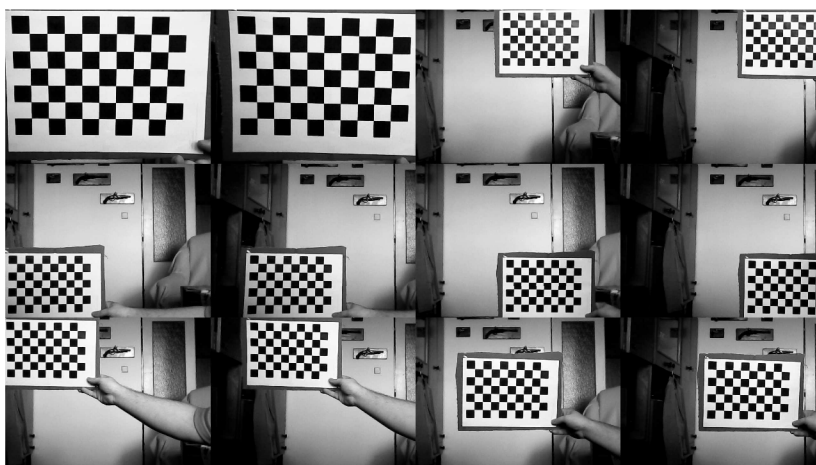
## Pracovní postup

### 1. Umístění kamer.

Kamery jsou umístěny vedle sebe paralelně nebo soustředěně tak, aby nebyly oslněny přímým světlem, nejlépe tak, že paprsky světla směřují do záběru. Dále je třeba zajistit, aby po provedení kalibrace zůstaly kamery fixní.

### 2. Kalibrace kamer.

Při kalibraci je kalibrační obrazec vždy umísťován tak, aby nechyběla žádná jeho část v levém ani pravém snímku. Nejprve je umístěn co nejbližší k vykrytí co největší snímané plochy, posléze je obrazec vzdálen na jeden metr od kamer a přesunován postupně přes všechny rohy a střed levého snímku tak, jak je uvedeno na obrázku 3.6. Po kalibraci je provedeno získání hodnoty ohniskové vzdálenosti.

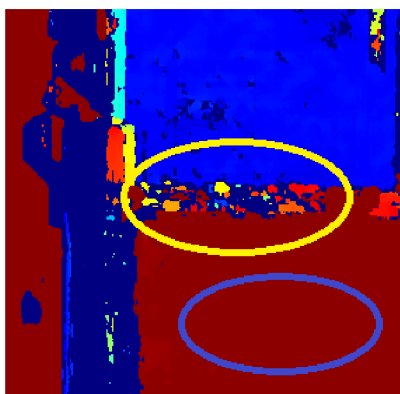


Obr. 3.6: Ukázka postupu kalibrace

### 3. Vlastní měření

Měření je rozvrženo v celých půlmetrových délkách, přičemž první a poslední měřená vzdálenost může být odlišná. První vzdálenost se odvíjí od schopnosti použité disparitní metody určit nejmenší vzdálenost a poslední měřená vzdálenost se odvíjí od fyzických limitů místnosti, tedy stěny místnosti, kde bylo prováděno měření. Měření probíhalo tak, že po umístění měřeného objektu bylo vyčkáno alespoň 10 sekund, po ustálení disparitní mapy byla odečtena hodnota měřené vzdálenosti z disparitní mapy bez a se subpixelovou přesností. Takto bylo postupováno přes všechny měřené vzdálenosti a všechny konfigurace poloh kamer. U poloh kamer se měnila rozteč z 4,5 cm na 10 cm a vzájemné natočení kamer z paralelního pohledu na soustředěný v jednom metru od kamer, čemuž by odpovídal úhel natočení o  $2,6^\circ$  u 4,5cm rozteče nebo  $5,8^\circ$

u 10 cm rozteče. Na obrázku 3.7 vidíme výřez disparitní mapy měřeného ob-



Obr. 3.7: Výřez disparitní mapy - nejmenší měřitelná vzdálenost

jektu umístěného co nejbližší kamerám. Žlutě ohraničená část značí místo, kde není možné přibližovaný objekt dále zpracovávat disparitní metodou a souvislá plocha, modře ohraničená část, se začne rozpadat do artefaktů.

### Prostor pro měření

Jako nejvhodnější místnost pro měření byla určena místnost rovnoměrně osvětlená zářivkovými světly nebo osvětlená nepřímým slunečním světlem. Takto zajištěné osvětlení nevytváří ostré stíny, přexponované a podexponované snímky. Jsou-li snímky správně exponované, neztrácejí kresbu a plasticitu snímané scény. Na Obr. 3.8 na levém snímku můžeme vidět příklad špatně exponovaného snímku



Obr. 3.8: Ukázka špatně a dobře exponovaného snímku

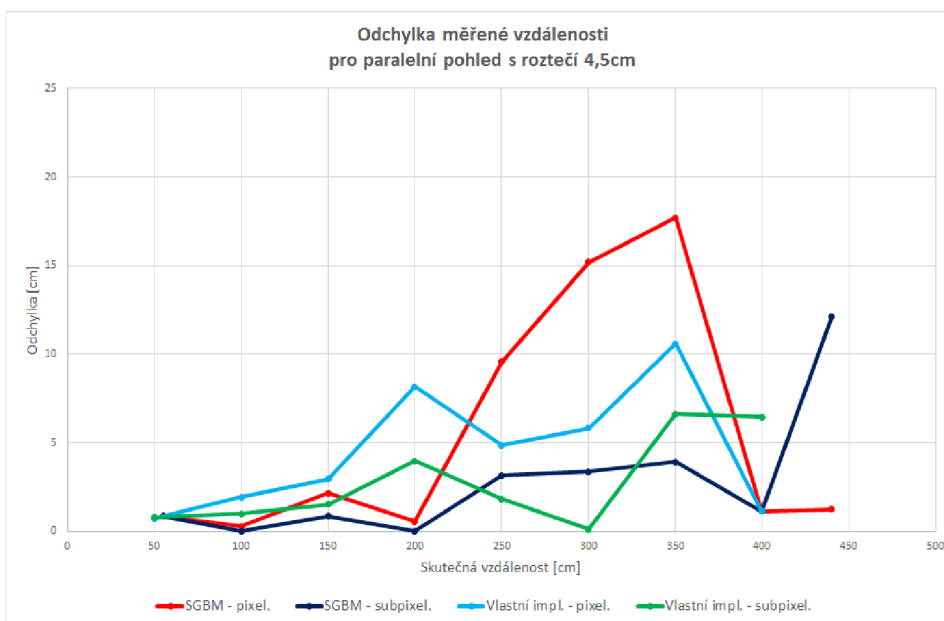
pořízeného při ostrém slunečním světle. Snímek je současně přexponovaný a podexponovaný. V místě vysokého jasu by měla být vidět police a v místě nízkého jasu by měl být vidět rukáv trika s pruhovaným vzorem, jak je to na pravém snímku, který je exponován lépe. Dále si můžeme všimnout, že ostré světlo vrhá znatelné

stíny, patrné na obličejích. Právý snímek je pořízen ve chvíli, kdy se zatáhlo a zlepšily se světelné podmínky pro fotografování.

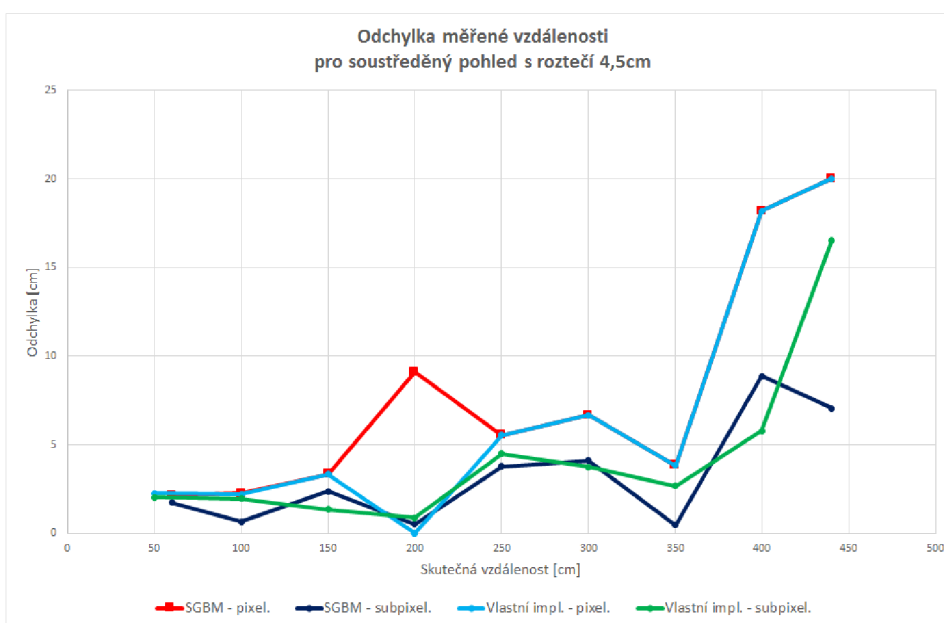
### **3.4.2 Naměřené hodnoty**

Naměřené hodnoty disparit, vzdáleností a vypočtených odchylek pro jednotlivé polohy kamer a použitých disparitních metod jsou uvedeny v tabulkách, které nalezneme v příloze A. Zde nalezneme grafy, kde porovnáme odchylky pro obě měřené disparitní metody a pro různé polohy.

## Srovnání podle vzájemné pozice kamer

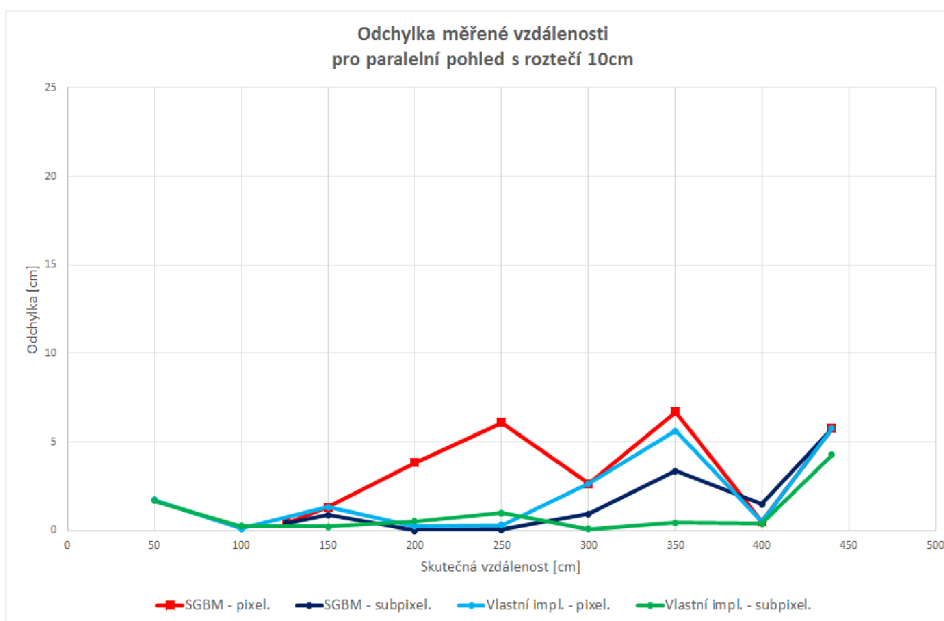


Obr. 3.9: Graf měření, kamery paralelně, rozteč 4,5 cm

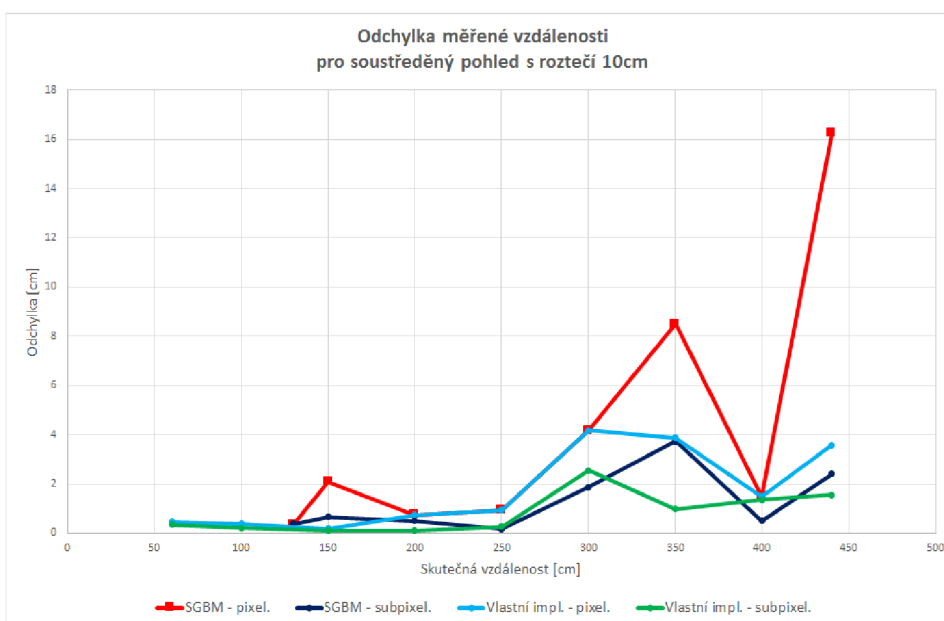


Obr. 3.10: Graf měření, kamery paralelně, rozteč 10 cm



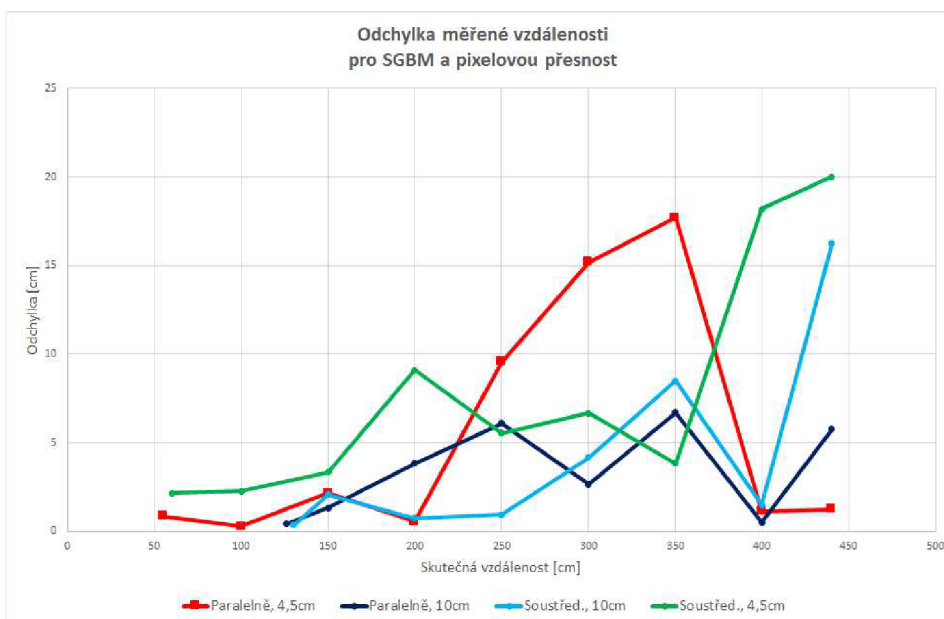


Obr. 3.11: Graf měření, kamery soustředěně, rozteč 10 cm

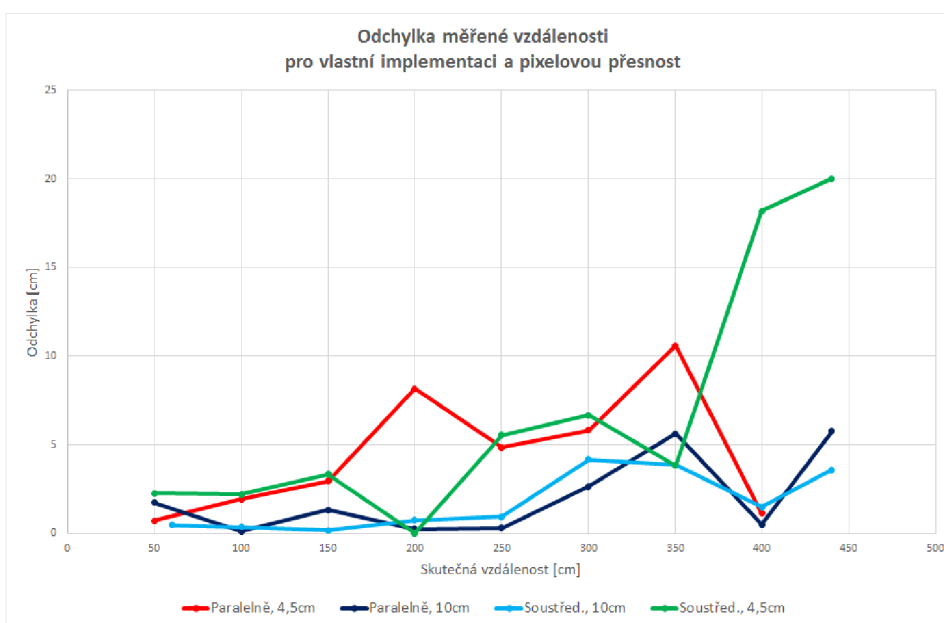


Obr. 3.12: Graf měření, kamery soustředěně, rozteč 4,5 cm

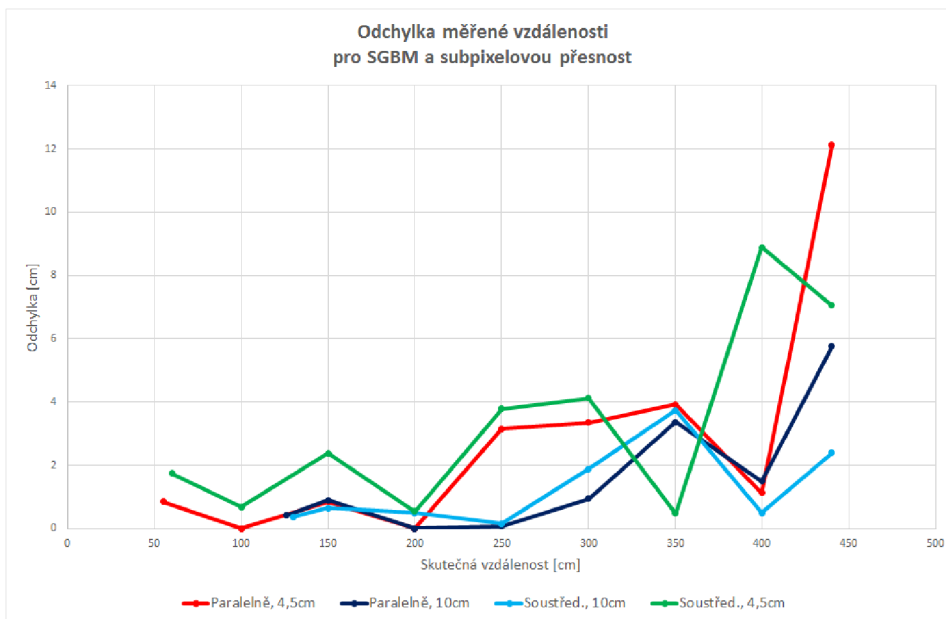
## Srovnání podle použité disparitní metody



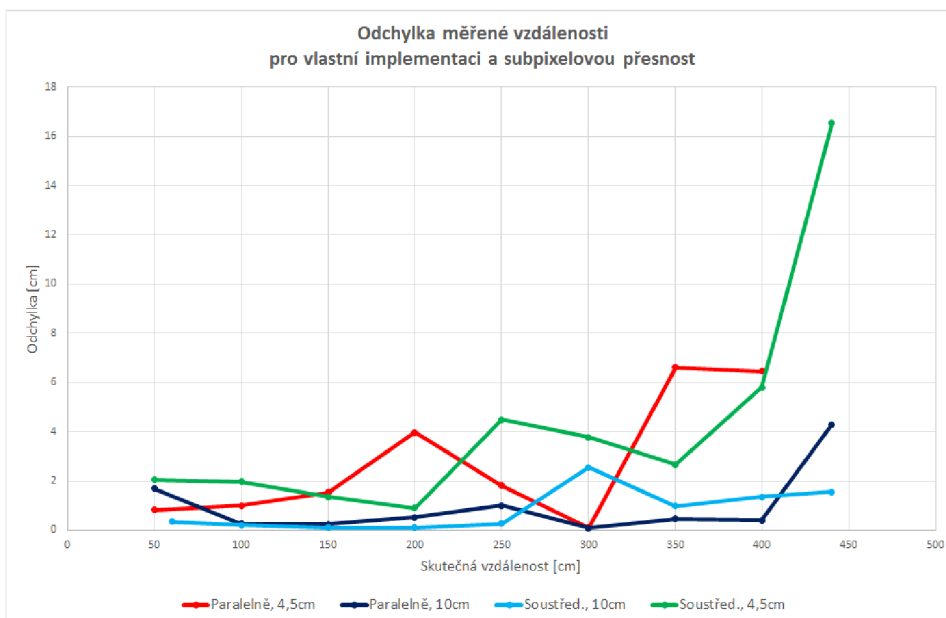
Obr. 3.13: Graf měření, SGBM, pixelová přesnost



Obr. 3.14: Graf měření, vlastní implementace, pixelová přesnost



Obr. 3.15: Graf měření, SGBM, subpixelová přesnost



Obr. 3.16: Graf měření, vlastní implementace, subpixelová přesnost

### 3.4.3 Zhodnocení měření

Z grafů naměřených odchylek si můžeme hned všimnout, že výpočet subpixelové přesnosti přináší snížení odchylek měřené vzdálenosti u všech poloh kamer, ale i v obou disparitních metodách, a je tedy přínosem. U metody SGBM se odchylka snížila průměrně o 42% a u vlastní implementace disparitní metody o 55%. Dále si můžeme všimnout, že metoda SGBM vykazuje vyšší odchylky než vlastní implementace disparitní metody. Metoda SGBM používá různé pre-filtry a post-filtry, které spíše než ke zvýšení přesnosti výpočtu disparit vedou ke snížení citlivosti na kvalitu pořízených stereosnímků, jejich kalibraci a rektifikaci. Nicméně to vede k věrnější rekonstrukci 3D prostoru i za cenu nižší přesnosti disparit. Změna z paralelní konfigurace kamer na soustředěnou nemá vliv na velikost odchylek, neboť je tato změna eliminována při kalibraci kamer, resp. při dedistorzi a rektifikaci stereosnímků na základě kalibračních údajů. Změna rozteče kamer z 4,5 cm na 10 cm způsobí snížení odchylek a zvýšení maximální měřitelné vzdálenosti, ale i zvýšení minimální měřitelné vzdálenosti. Změnil se tak rozsah měření vzdáleností a snížení odchylek je způsobené využitím výhodnější části křivky odchylek. Můžeme si to představit tak, že bychom vzali křivku odchylek pro rozteč kamer 4,5cm a posunuli ji v grafu směrem vpravo.

### 3.4.4 Faktory ovlivňující měření

Zde je uveden stručný přehled vlastních postřehů získaných během různých pokusů o měření vzdálenosti. Většina postřehů je převážně spojená s kvalitou, resp. nekvalitou pořizovaných snímků z používaných kamer.

#### Optika kamer

1. Neostrost v rozích
2. Zkreslení obrazu
3. Nemožnost stejného zaostření obrazu na obou kamerách

Kromě neostrosti na okrajích záběru si proces kalibrace na obrazec s ostatními vadami poradí. Neostrosti na okrajích záběru naštěstí nevadí, protože lze kamery nebo měřený objekt nasměrovat do záběru tak, aby nebyl neostrostmi zasažen.

#### Automatika kamer

Zde je největším problémem nezávislost řízení snímačů kamer, které se starají o optimální expozici, vyvážení bílé barvy, případně odstraňují blikání v případě zářivkového či jiného blikajícího osvětlení. Mnohokrát se při měření stalo, že automatika jedné z kamer v nevhodnou chvíli vyhodnotila současné světelné podmínky za nevhovující a změnila expozici. Výsledkem byla skoková změna hodnot celé disparitní

mapy a samozřejmě měřených vzdáleností, které neodpovídaly skutečnosti. Bohužel se nepodařilo nezávislost automatik kamer potlačit a proto bylo přikročeno k opatřením, které ve výsledku nenutí automatiky kamer ke změnám (viz kapitola 3.4.1, Prostor pro měření).

### **Osvětlení**

Ostré přímé sluneční nebo umělé bodové světlo způsobuje přexponování nebo podexponování snímků. Způsobují tak ztrátu kresby a plasticity snímané scény. Často bývají podexponované také oblasti se stíny. Tyto nežádoucí jevy lze odstranit nebo alespoň zmírnit rozptýlením ostrého světla použitím matného skla, pořizovat snímky v místnosti na neslunečné straně budovy či vyčkat na vhodné počasí, což může být časově náročné. Nebo použít zářivkové světlo v místnosti s omezeným nebo žádným přístupem vnějšího světla.

## 4 ZÁVĚR

V diplomové práci jsem se v teoretické části věnoval rozboru postupů pro předzpracování obrazu pro stereoskopii, srovnávací metriky, subpixelovou přesnost a určení vzdálenosti. Teorii jsem úspěšně aplikoval v praktické části, kde je výsledkem aplikace napsaná v jazyce C++ využívající otevřené knihovny OpenCV. V praktické části jsem se věnoval jednotlivým částem aplikace, kde popisují jejich činnost, principy a důvody použití. Rovněž zde popisují vlastní implementaci disparitní metody na bázi Block matchingu, se kterou jsem si ověřil princip tvorby disparitní mapy.

Aplikace pro měření vzdálenosti disponuje dvěma metodami, metodou Semiglobal block matching(dále SGBM metoda) z knihovny OpenCV a vlastní implementací disparitní metody na bázi Block matchingu. Obě metody mají implementovaný výpočet subpixelové přesnosti proložením parabolou.

Pro určení vzdálenosti jsem použil podobnost trojúhelníku, kde je zapotřebí zjistit ohniskovou vzdálenost kamer. Původní záměr byl použít ohniskovou vzdálenost získanou z kalibračních hodnot. Nicméně takto získaná ohnisková vzdálenost neposkytovala dostatečnou přesnost. Proto jsem zvolil jednodušší a přímější variantu získání ohniskové vzdálenosti.

Během prvních měření vzdáleností s hotovou aplikací jsem narazil na řadu problémů. Řadu z nich způsobovala nezávislá automatika kamer, která se stará o řízení snímačů a má poskytovat optimální obraz z hlediska expozice a vyvážení bílé barvy. Tato nezávislost vytvářela při měření rozporupné výsledky, protože automatika jedné z kamer, vždy v nevhodnou chvíli, vyhodnotila aktuální světelné podmínky jako nedostatečné a změnila parametry snímače. To způsobilo značné změny a výsledná disparitní mapa nijak nekorelovala se skutečnou podobou prostoru. Abych tyto problémy snížil, vypracoval jsem metodiku měření, se kterou lze zajistit shodné podmínky po celou dobu měření a získat tak konzistentní výsledky. Jedná se o souhrn úkonů měření a popis, jak má být místnost pro měření osvětlena.

Finální měření vzdáleností bylo provedeno pro různé konfigurace poloh kamer. Naměřené hodnoty jsou uvedeny v tabulkách v příloze a odchylky od skutečné vzdálenosti byly zaneseny do grafů.

Při plnění cílů zadaných v diplomové práci byly teoretické poznatky ověřeny v praxi. Věřím, že zadání bylo splněno bezesbytku.

## LITERATURA

- [1] HARTLEY, Richard a Andrew ZISSERMAN. *Multiple view geometry in computer vision*. 2.edice, Cambridge: Cambridge University Press, 2003, 655 s. ISBN 05-215-4051-8.
- [2] BRADSKI, Gary a Adrian KAEHLER. *Learning OpenCV*. Sebastopol: O'Reilly, 2008, 555 s. ISBN 978-0-596-51613-0.
- [3] OpenCV. OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. [online]. 2013-12-30 [cit. 2013-12-30]. Dostupné z: <http://opencv.org>
- [4] *Laboratorní cvičení: Stereovidění*. [online]. 2011 [cit. 2013-12-30]. Dostupné z: <http://www.fbmi.cvut.cz/files/predmety/3528/public/Stereo%20vid%C4%9Bn%C3%AD%20-%20podklady.pdf>
- [5] *Parametry webové kamery Sweex WC035V2*. [online]. 2009 - 2014 [cit. 2014-5-18]. Dostupné z: <http://www.sweex.com/en/assortiment/sound-vision/webcams/WC035V2/>
- [6] CYGANEK, Boguslaw. *An Introduction to 3D computer vision techniques and algorithms*. Chichester: Wiley, 2009, 483 s. ISBN 978-0-470-01704-3.
- [7] ŘÍHA, Kamil. *Pokročilé techniky zpracování obrazu [elektronicky]*. První vydání. Brno: Vysoké učení technické v Brně, Fakulta elektroniky a komunikačních technologií, Ústav telekomunikací, Technická 12, 616 00 Brno, 2013.
- [8] *Kinect for Windows Sensor Components and Specifications*. [online]. 2014 [cit. 2014-07-08]. Dostupné z: <http://msdn.microsoft.com/en-us/library/jj131033.aspx>
- [9] JÄHNE, Bernd, Horst HAUSSECKER a Peter GEISLER, *Handbook of computer vision and applications*. San Diego: Academic Press, 1999, ISBN 0-12-379771-3.
- [10] MAHAMMED, Manaf A., Amara I. MELHUM a Faris A. KOCHERY. *Object Distance Measurement by Stereo Vision* [online]. Dubai: ICCTE, 2013 [cit. 2014-07-28]. ISSN 2278-3083. Dostupné z: <http://warse.org/pdfs/2013/icctesp02.pdf>
- [11] BIRCHFIELD, S. a C. TOMASI. *A pixel dissimilarity measure that is insensitive to image sampling*. IEEE Transactions on Pattern Analysis and Machine

- Intelligence [online]. 1998, vol. 20, issue 4, s. 401-406 [cit. 2014-08-20]. DOI: 10.1109/34.677269.
- [12] *ColorMaps in OpenCV*. [online]. 2014 [cit. 2014-08-20]. Dostupné z: <http://docs.opencv.org/trunk/modules/contrib/doc/facerec/colormaps.html>
- [13] OpenCV. Installation by Making Your Own Libraries from the Source Files. [online]. 2015 [cit. 2015-04-15]. Dostupné z: [http://docs.opencv.org/doc/tutorials/introduction/windows\\_install/windows\\_install.html](http://docs.opencv.org/doc/tutorials/introduction/windows_install/windows_install.html)
- [14] The Middlebury Computer Vision Pages: Middlebury Stereo Evaluation. *Middlebury* [online]. 2014 [cit. 2015-05-16]. Dostupné z: <http://vision.middlebury.edu/stereo/eval/>



## SEZNAM ZKRATEK

USB Universal Serial Bus, univerzální sériová sběrnice

C/C++ multiparadigmatický programovací jazyk, který vyvinul Bjarne Stroustrup

OpenCV Open source Computer Vision, svobodná knihovna pro manipulaci s  
obrazem

OpenCL Open Compute Language, průmyslový standard pro paralelní  
programování

CUDA Compute Unified Device Architecture, architektura umožňující běh  
programů na technologii OpenCL

RGB Red - Green - Blue, barevný model využívající 3 barev

CMOS Complementary Metal - Oxide - Semiconductor, technologie integrovaných  
obvodů

CCD Charge - Coupled Device, technologie snímání obrazu

2D Two Dimensional, dvou rozměrný (prostor)

3D Three Dimensional, tří rozměrný (prostor)

PnP Plug and play, doslova připoj a hrej, technologie umožňující jednodušší  
rozpoznávání a konfiguraci hardwaru

BM Block matching, nejjednodušší metoda pro tvorbu disparitních map

SGBM Semiglobal block matching, pokročilá metoda pro tvorbu dispartiních map

# SEZNAM PŘÍLOH

<b>A</b>	<b>Tabulky měření</b>	<b>54</b>
A.1	Měření pro SGBM disparitu . . . . .	54
A.2	Měření pro vlastní implementaci disparity . . . . .	55
A.3	Porovnání metrik . . . . .	56
<b>B</b>	<b>Optický disk</b>	<b>58</b>
B.1	Diplomová práce . . . . .	58
B.2	OpenCV . . . . .	58
B.3	Zdrojový kód aplikace . . . . .	58
B.4	Zkompilovaná aplikace . . . . .	58
<b>C</b>	<b>Návod k použití aplikace</b>	<b>59</b>

# A TABULKY MĚŘENÍ

## A.1 Měření pro SGBM disparitu

Tab. A.1: Měření vzdálenosti, kamery paralelně, rozteč 4,5 cm

Skutečná vzdálenost	55	100	150	200	250	300	350	400	440	cm
Pixelová disparita	79	44	29	22	17	14	12	11	10	-
Subpixelová disparita	79	44,125	29,25	22,0625	17,875	14,875	12,75	11	10,3125	-
Pixelová vzdálenost	55,854	100,284	152,155	200,56	259,559	315,179	367,708	401,136	441,25	cm
Subpixelová vzdálenost	55,854	100	150,855	200	246,853	296,639	346,078	401,136	427,879	cm
Odchylka pixel. vzdál.	0,854	0,284	2,155	0,56	9,559	15,179	17,708	1,136	1,25	cm
Odchylka subpixel. vzdál.	0,854	0	0,855	0	3,147	3,361	3,922	1,136	13,121	cm

Tab. A.2: Měření vzdálenosti, kamery paralelně, rozteč 10 cm

Skutečná vzdálenost	126	150	200	250	300	350	400	440	cm
Pixelová disparita	79	66	49	39	33	28	25	23	-
Subpixelová disparita	79	66,1875	49,9375	39,9375	33,1875	28,8125	25,0625	23	-
Pixelová vzdálenost	126,424	151,326	203,827	256,09	302,652	356,696	399,5	434,239	cm
Subpixelová vzdálenost	126,424	150,897	200	250,078	300,942	346,638	398,504	434,239	cm
Odchylka pixel. vzdál.	0,424	1,326	3,827	6,09	2,652	6,696	0,5	5,761	cm
Odchylka subpixel. vzdál.	0,424	0,897	0	0,078	0,942	3,362	1,496	5,761	cm

Tab. A.3: Měření vzdálenosti, kamery soustředěně, rozteč 10 cm

Skutečná vzdálenost	130	150	200	250	300	350	400	440	cm
Pixelová disparita	77	66	50	40	33	28	25	22	-
Subpixelová disparita	77,4375	66,625	50,0625	40,125	33,25	28,375	25,0625	22,9375	-
Pixelová vzdálenost	130,357	152,083	200,75	250,938	304,167	358,492	401,5	456,25	cm
Subpixelová vzdálenost	129,621	150,657	200,499	250,156	301,88	353,744	400,499	437,602	cm
Odchylka pixel. vzdál.	0,357	2,083	0,75	0,938	4,167	8,492	1,5	16,25	cm
Odchylka subpixel. vzdál.	0,379	0,657	0,499	0,156	1,88	3,744	0,499	2,398	cm

Tab. A.4: Měření vzdálenosti, kamery soustředěně, rozteč 4,5 cm

Skutečná vzdálenost	60	100	150	200	250	300	350	400	440	cm
Pixelová disparita	74	45	30	22	18	15	13	11	10	-
Subpixelová disparita	74,5	45,6875	30,1875	22,9375	18,125	15,125	13,125	11,25	10,625	-
Pixelová vzdálenost	62,162	102,272	153,333	209,091	255,556	306,667	353,846	418,182	460	cm
Subpixelová vzdálenost	61,745	100,684	152,381	200,545	253,793	304,122	350,476	408,889	432,941	cm
Odchylka pixel. vzdál.	2,162	2,272	3,333	9,091	5,556	6,667	3,846	18,182	20	cm
Odchylka subpixel. vzdál.	1,745	0,684	2,381	0,545	3,793	4,122	0,476	8,889	7,059	cm

## A.2 Měření pro vlastní implementaci disparity

Tab. A.5: Měření vzdálenosti, kamery paralelně, rozteč 4,5 cm

Skutečná vzdálenost	50	100	150	200	250	300	350	400	cm
Pixelová disparita	87	45	30	23	18	15	13	11	-
Subpixelová disparita	87,8324	44,5659	29,7207	22,509	17,7795	14,7139	12,8501	11,2124	-
Pixelová vzdálenost	50,718	98,056	147,038	191,848	245,139	294,167	339,423	401,136	cm
Subpixelová vzdálenost	50,816	99,011	148,466	196,032	248,179	299,887	343,382	393,539	cm
Odchylka pixel. vzdál.	0,718	1,944	2,962	8,152	4,861	5,833	10,577	1,136	cm
Odchylka subpixel. vzdál.	0,816	0,989	1,534	3,968	1,821	0,113	6,618	6,461	cm

Tab. A.6: Měření vzdálenosti, kamery paralelně, rozteč 10 cm

Skutečná vzdálenost	50	100	150	200	250	300	350	400	440	cm
Pixelová disparita	193	100	66	50	40	33	29	25	23	-
Subpixelová disparita	193,248	99,6138	66,4779	49,8065	39,7925	33,2813	28,5732	24,9433	22,9224	-
Pixelová vzdálenost	51,749	99,875	151,326	199,75	249,688	302,652	344,379	399,5	434,239	cm
Subpixelová vzdálenost	51,687	100,262	150,238	200,526	250,99	300,093	349,541	400,408	435,71	cm
Odchylka pixel. vzdál.	1,749	0,125	1,326	0,25	0,312	2,652	5,621	0,5	5,761	cm
Odchylka subpixel. vzdál.	1,687	0,262	0,238	0,526	0,99	0,093	0,459	0,408	4,29	cm

Tab. A.7: Měření vzdálenosti, kamery soustředěně, rozteč 10 cm

Skutečná vzdálenost	60	100	150	200	250	300	350	400	440	cm
Pixelová disparita	166	100	67	50	40	33	29	25	23	-
Subpixelová disparita	166,329	100,167	66,8711	50,1606	40,1085	33,1762	28,5989	25,0089	22,8934	-
Pixelová vzdálenost	60,467	100,375	149,813	200,75	250,938	304,167	346,121	401,5	436,413	cm
Subpixelová vzdálenost	60,347	100,208	150,102	200,107	250,259	302,552	350,975	401,357	438,444	cm
Odchylka pixel. vzdál.	0,467	0,375	0,187	0,75	0,938	4,167	3,879	1,5	3,587	cm
Odchylka subpixel. vzdál.	0,347	0,208	0,102	0,107	0,259	2,552	0,975	1,357	1,556	cm

Tab. A.8: Měření vzdálenosti, kamery soustředěně, rozteč 4,5 cm

Skutečná vzdálenost	50	100	150	200	250	300	350	400	440	cm
Pixelová disparita	88	45	30	23	18	15	13	11	10	-
Subpixelová disparita	88,3862	45,1148	30,3904	23,1034	18,0746	15,1432	13,0433	11,3554	10,0758	-
Pixelová vzdálenost	52,273	102,222	153,333	200	255,556	306,667	353,846	418,182	460	cm
Subpixelová vzdálenost	52,044	101,962	151,364	199,105	254,5	303,767	352,671	405,807	456,538	cm
Odchylka pixel. vzdál.	2,273	2,222	3,333	0	5,556	6,667	3,846	18,182	20	cm
Odchylka subpixel. vzdál.	2,044	1,962	1,364	0,895	4,5	3,767	2,671	5,807	16,538	cm

## A.3 Porovnání metrik

Tab. A.9: Závislost chyby na velikosti agregačního okna, Tsukuba

Okno	SAD	SSD	NCC
3	61,4436	61,4001	65,2292
5	60,8169	60,7806	51,8584
7	60,4638	60,2675	45,4362
11	59,8543	59,2315	39,6627
14	59,1378	58,3244	37,5224
17	57,6775	57,0047	36,1713
20	56,9818	56,6544	35,7999
24	56,7926	56,5696	36,2322
28	56,7459	56,6777	36,972

Tab. A.10: Závislost chyby na velikosti agregačního okna, Teddy

Okno	SAD	SSD	NCC
3	87,6331	87,68	75,1726
5	90,788	90,7495	75,8867
7	93,0595	92,8609	76,8376
11	96,1693	95,9425	78,5633
14	97,8614	97,6128	80,1402
17	98,6848	98,3898	81,78
20	98,9094	98,7869	83,1845
24	98,6772	99,0438	84,6986
28	98,5497	99,276	86,4783

Tab. A.11: Závislost chyby na velikosti agregačního okna, Cones

Okno	SAD	SSD	NCC
3	94,2625	93,8309	70,6416
5	98,0423	97,1542	71,6022
7	101,069	99,7072	72,2757
11	105,799	104,117	74,392
14	109,487	106,242	77,3532
17	109,487	107,579	81,1576
20	110,273	108,387	84,7485
24	110,695	108,75	89,353
28	110,783	108,905	93,4113

## **B OPTICKÝ DISK**

Příložený optický disk obsahuje následující

### **B.1 Diplomová práce**

Jedná se o elektronickou verzi této práce uloženou ve formátu PDF.

### **B.2 OpenCV**

Zkomprimovaný soubor s překompilovanou otevřenou knihovnou počítačového vidění OpenCV s podporou Qt funkcí v použité verzi vydání (2.4.11). Bez této knihovny nelze použít hotovou aplikaci ani zkompileovat zdrojový kód aplikace.

### **B.3 Zdrojový kód aplikace**

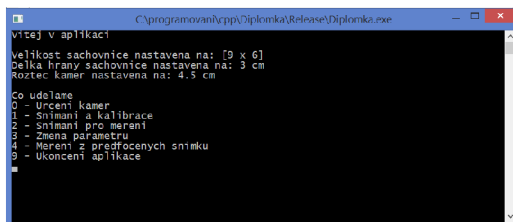
Zdrojový kód aplikace napsaný v jazyce C++.

### **B.4 Zkompilovaná aplikace**

Spustitelná aplikace doplněná o dynamické knihovny potřebné pro běh aplikace.

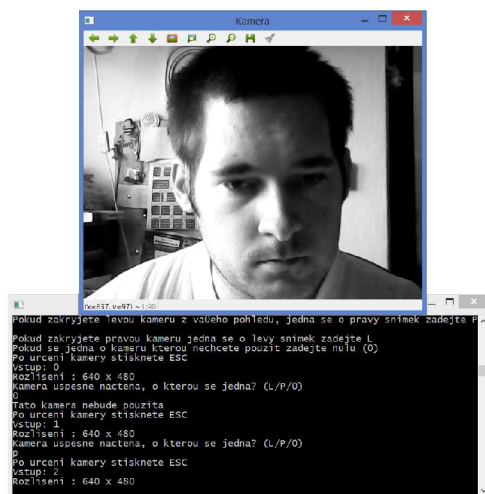
## C NÁVOD K POUŽITÍ APLIKACE

Umístěte vhodně obě kamery a zapojte je do počítače. Po spuštění aplikace se objeví konzolové okno s výpisem aktuálních parametrů pro kalibraci a menu s očíslovanými operacemi.



Obr. C.1: Aplikace - Konzolové okno

V tuto chvíli aplikace neví, které kamery má použít. Zvolíte proto operaci *Urceni kamer*, stisknete nulu a potvrdíte klávesou Enter.

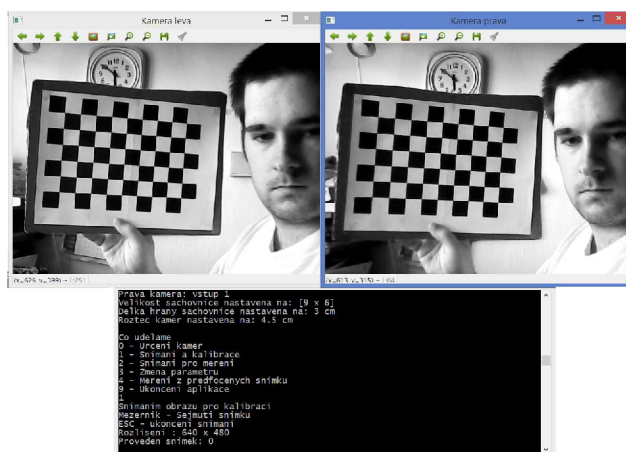


Obr. C.2: Aplikace - Určení kamer

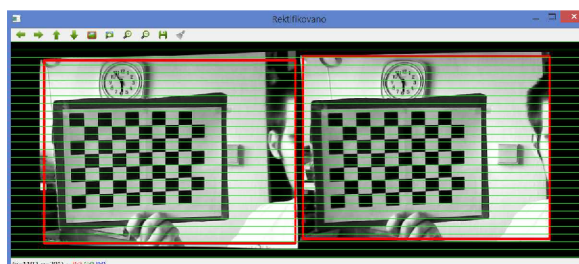
Objeví se okno s obrazovým vstupem některé z kamer. Postupným zakrýváním kamer zjistíte, o kterou kameru se jedná. Stisknete klávesu Esc a v konzoli se objeví žádost o zadání L, P nebo nuly. L nebo P zadáváte tehdy, jedná-li se o levou nebo pravou kameru z pohledu kamer. Nulu zadáte pouze tehdy, když máte připojených kamer více (např. integrovaná kamera v notebooku) a nehodí se do měření a potvrdíte klávesou Enter. Nyní se můžete pustit do kalibrace zvolením *Snimani a kalibrace*, stisknete 1 a klávesu Enter. Nyní se objeví dvě okna, každé pro jednu kameru a jsou stranově označena. V této chvíli můžete ručně doostřit kamery a upřesnit jejich



polohu. Posléze před kamery předkládejte kalibrační obrazec tak, aby byl v záběru obou kamer a pokaždé stiskněte klávesu Mezerník. V konzoli můžete zkontrolovat, zda byly snímky zachyceny. Je třeba nasnímat alespoň 2 polohy kalibračního obrazce pro úspěšnou kalibraci. V praxi je lepší nasnímat alespoň 5 poloh kalibračního obrazce pro vyšší přesnost. Snímání ukončíte stisknutím klávesy ESC. Proběhne validace nasnímaných obrazců, vyřadí se nevyhovující a vyhovující postoupí do kalibrace, která může zabrat delší čas. Na konci kalibrace se objeví okno s opravenými a rektifikovanými obrazy z kamer. Slouží především jako kontrola, zda se kalibrace povedla. Pokud se rektifikace nepovedla, proveďte kalibraci znovu.



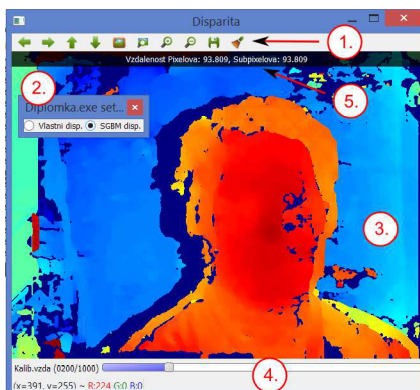
Obr. C.3: Aplikace - Snímání a kalibrace



Obr. C.4: Aplikace - Vizuální kontrola provedené kalibrace

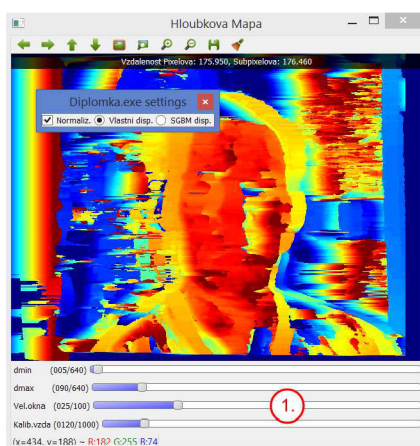
Dále můžete pokračovat volbou *Snímání pro měření*. Objeví se okno zobrazující disparitní mapu vytvořenou metodou SGBM, které můžete vidět na Obr. C.5. Na obrázku můžete vidět očíslované body. Bod 1. slouží otevření podokna s přepínačem volby disparitní metody (Bod 2.). Bod 3. zobrazuje vlastní disparitní mapu, která je pro lepší čitelnost převedena do JET barev. U bodu 4. se nachází posuvník se kterým si můžeme nastavit známou vzdálenost objektu zachycenou v disparitní

mapě. Nastavením kurzoru myši na známý objekt a stisknutím levého tlačítka myši dojde k vypočtení ohniskové vzdálenosti, potřebná pro výpočet vzdálenosti. Nyní by se v bodě 5. měla zobrazovat korektně pixelová a subpixelová vzdálenost. Zobrazované vzdálenosti jsou platné pro objekty v disparitní mapě nacházející se pod kurzorem myši.



Obr. C.5: Aplikace - Snímání pro měření - metoda SGBM

Pokud u přepínače volby disparitní metody zvolíte vlastní implementaci, objeví se stejné okno, jen v bodě 1. přibudou další posuvníky, *dmin*, *dmax* a *Vel.okna*. Těmito posuvníky můžete nastavovat minimální a maximální disparitu a velikost agregačního okna. Vhodným nastavením agregačního okna můžete dosáhnout kvalitnější disparitní mapy. Nastavením minima a maxima disparity změníte rozsah zpracovávaných disparit, vhodným nastavením rozsahu disparit můžete urychlit výpočet disparitní mapy, ovšem na úkor úplnosti disparitních map.



Obr. C.6: Aplikace - Snímání pro měření - vlastní implementace

Měření můžete kdykoliv ukončit stiskem klávesy Esc, vrátíte se tak do základního

menu. Pod položkou *Zmena parametru* můžete změnit jednotlivé parametry kalibrace jako velikost šachovnice, délka hrany jednoho pole šachovnice a rozteč kamer. Pokud změňte některý z parametrů, je vhodné provést znovu kalibraci kamer.

Další položkou v menu je *Mereni z predfocenyh snimku*, kde můžete provést výpočet disparitní mapy a měření vzdálenosti z předfocené stereo snímku. Tyto snímky však už musí být zkalibrované a zrektifikované. Pokud se soubory se snímky nachází ve stejném adresáři stačí zadat jen jména těchto souborů spolu s koncovkou (např. tsukuba\_1.png) nebo můžete zadat úplnou cestu k souborům (např. C:\Diplomka\tsukuba\_1.png). Po zadání souborů opět uvidíte okno stejné jako na Obr. C.5 nebo C.6. podle zvolené disparitní metody. Pokud budete mít zvolenou vlastní implementaci a budete měnit parametry  $dmin$ ,  $dmax$  a *Vel.okna*, stiskněte klávesu Mezerník pro aktualizaci disparitní mapy. Pro návrat do menu použijte klávesu Esc.

Poslední položka v menu ukončí aplikaci.