



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A
BIOMECHANIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND
BIOMECHANICS

APLIKACE STEREOVIZE A POČÍTAČOVÉHO VIDĚNÍ

COMPUTER VISION AND STEREO VISION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MARTIN BUBÁK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JOSEF VEJLUPEK

BRNO 2014

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav mechaniky těles, mechatroniky a biomechaniky
Akademický rok: 2013/2014

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Martin Bubák

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Mechatronika (3906T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Aplikace stereovize a počítačového vidění

v anglickém jazyce:

Computer vision and stereo vision

Stručná charakteristika problematiky úkolu:

Cílem práce bude zhodnocení využití algoritmů sledování pohybu a stereovize pro lokalizaci mobilního robotu (mobilního objektu) v prostoru při využití stacionární případně pohyblivé kamery. Dále pak zhodnocení využitelnosti této informace při plánování trajektorie mobilního robotu.

Cíle diplomové práce:

- Nastudujte dokumentaci programového vybavení "Computer Vision System Toolbox" pro MATLAB a proveďte rešerši relevantních algoritmů
- Vytvořte aplikaci v prostředí MATLAB Simulink, která bude v reálném čase sledovat pomocí kamery scénu a detekovat polohu určitého objektu (například pomocí značky)
- Vytvořte aplikaci v prostředí MATLAB Simulink, která bude v reálném čase vytvářet pomocí dvou kamer stereoskopický model scény / prostoru
- Propojte vytvořené aplikace a využijte je pro odhad polohy objektu ve 3D prostoru.

Seznam odborné literatury:

Pratt, W. K.: Digital Image Processing (Second Edition), New York: Wiley 1991

Klíma, M.; Bernas, M.; Hozman J.; Dvořák, P. : Zpracování obrazové informace

<http://www.mathworks.com/products/computer-vision/index.html>


<http://www.zam.fme.vutbr.cz/~druck/>

Vedoucí diplomové práce: Ing. Josef Vejlupek


Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2013/14.

V Brně, dne 22.11.2013





prof. Ing. Jindřich Petruška, CSc.
Ředitel ústavu



prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan

ABSTRAKT

Tato diplomová práce se zabývá využitím softwarového nástroje Computer Vision System Toolbox pro vytvoření aplikací v počítačovém vidění.

Na začátku práce je provedena rešerše snímání obrazu a jeho reprezentace pomocí barevných modelů. Následuje popis epipolární geometrie a nakonec je uveden popis Computer Vision System Toolboxu.

V další části se zabýváme nastavením použitých kamer Basler a zpracováním snímaného obrazu. Následuje popis vytvoření aplikace pro detekci objektu. Po tomto popisu se seznámíme s aplikací pro vytvoření hloubkové mapy prostoru.

KLÍČOVÁ SLOVA

Computer Vision System Toolbox, Stereovize

ABSTRACT

This dissertation work is describing the usage of the software tool Computer Vision System Toolbox to create applications in computer vision.

At the beginning of the work is performed background research of image scanning and its representation by using colour models. It is followed by a description of epipolar geometry and lastly is stated a description of the Computer Vision System Toolbox.

In the next section of the work we deal with setting of used Basler cameras and processing of the scanned image. The following is a description how to create applications for object detection and after this description, we get to know applications for creation of depth maps area.

KEYWORDS

Computer Vision System Toolbox, Stereo vision

BIBLIOGRAFICKÁ CITACE

BUBÁK, Martin. *Aplikace stereovize a počítačového vidění*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2014. 46 s. Vedoucí diplomové práce Ing. Josef Vejlupek.

ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Josefa Vejlupka a s použitím literatury uvedené v seznamu.

V Brně dne 25. května 2014

.....

Martin Bubák

PODĚKOVÁNÍ

Tímto bych chtěl poděkovat vedoucímu diplomové práce Ing. Josefu Vejlupkovi za cenné rady a připomínky.

OBSAH

Úvod.....	9
1 Rešeršní studie.....	11
1.1 CCD snímače.....	11
1.2 CMOS snímače.....	12
1.3 Barevné snímání obrazu.....	13
1.4 Barevné modely.....	14
1.4.1 RGB model.....	14
1.4.2 CMY a CMYK model.....	15
1.4.3 YUV model.....	15
1.4.4 HSV model.....	16
1.5 Epipolární geometrie.....	16
1.6 Fundamentální matice.....	18
1.7 Computer Vision System Toolbox.....	19
2 Realizace aplikací.....	22
2.1 Použité kamery.....	22
2.2 Použité objektivy.....	23
2.3 Nastavení parametrů kamery.....	24
2.4 Kalibrace obrazu.....	26
2.5 Detekce polohy.....	29
2.6 Výpočet hloubkové mapy.....	34
2.7 Výsledná aplikace.....	39
Závěr.....	43
Seznam obrázků.....	46

ÚVOD

Užití počítačového vidění má v dnešní době velmi široké uplatnění. Můžeme se s ním setkat nejen v průmyslových aplikacích, kde má za úkol například kontrolovat počet výrobků v daném obalu, kvalitu výrobků nebo v elektrotechnice k výsledné kontrole správnosti osazení DPS, ale i v medicínské oblasti.

Účelem nasazení počítačového vidění v průmyslu je většinou snížení nákladů na výrobu a její zrychlení. V některých případech bývá využíváno v oblastech, kde selhává lidské oko nebo v prostředích kam lidské oko nemůže.

Jeho využití je také důležité v medicínské oblasti, kde se využívá zpracování obrazu za účelem diagnózy. V této oblasti se obraz získává z mikroskopu, rentgenu, ultrazvuku či tomografie a jeho následné zpracování může sloužit například k nalezení nádorů.

Jedním z novějších využití počítačového vidění je navádění autonomních vozidel. Toho se například využívá k výzkumu vesmíru a v poslední době se počítačové vidění nasazuje i do automobilů. Právě v těchto případech se využívá stereovize jako prostředek pro vytvoření mapy prostředí za účelem detekce překážek.

Problematika počítačového vidění je hodně široká a neustále se vyvíjí. Vytváření algoritmů v tomto oboru je náročné nejen programátorsky, ale i časově. Proto pro nás, jako techniky jsou vyvíjeny nástroje pro usnadnění naší práce v tomto oboru. Jedním takovým nástrojem hojně užívaným v praxi je Vision Builder od firmy National Instruments. Další nástroj je od firmy Mathworks Computer Vision System Toolbox. Jedná se o poměrně nový nástroj, který je součástí Matlabu od roku 2011 a neustále se intenzivně vyvíjí.

Použití uvedených nástrojů ovšem není úplně tak jednoduché a vyžaduje určité znalosti v tomto oboru. Úspěchem zpracování obrazu není jenom samotný software, ale stejnou vahou se na něm podílí i hardware. Hardwarem v tomto případě nejsou myšleny jen použítá snímací zařízení-kamery, ale i různé filtry a osvětlovače, které nám mají za úkol ulehčit detekci daného předmětu či eliminovat vliv okolních podmínek jako osvětlení.

Nasnímaný obraz je tvořen velkým množstvím dat, která je potřeba zpracovat a to většinou v co nejkratším časovém úseku. Proto při návrhu hardwaru je potřeba zvážit vhodné komunikační rozhraní, kterým může být Gigabitový ethernet, USB 3.0 či Camera Link nebo CoaXPress. Dále výběr procesorového vybavení mezi „běžným“ výkonným procesorem a programovatelným hradlovým polem FPGA.

Tato práce si klade za cíl seznámit se s již zmíněným softwarovým nástrojem Computer Vision System Toolbox a s jeho využitím při minimálních znalostech v oblasti počítačového vidění. Dále dosáhnout znalostí v tomto oboru potřebných pro vytvoření aplikace, která bude schopná detekce určitého objektu. Dalším úkolem, kterým se tato práce zabývá je vytvoření aplikace, která bude v reálném čase vytvářet hloubkovou mapu prostoru za účelem odhadu polohy určitého objektu. Následně bude využito aplikace detekce v aplikaci pro vytvoření hloubkové mapy.

1 REŠERŠNÍ STUDIE

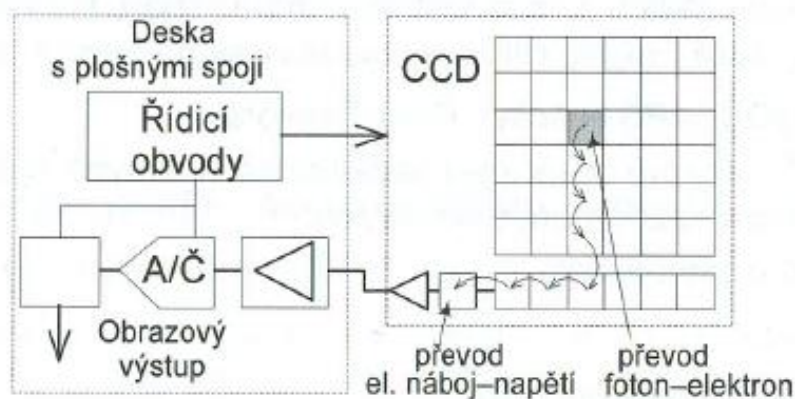
Světlo, které vidíme, je vlastně elektromagnetické vlnění. Jedná se o dvojdimenzionální spojité signál, které je nutný pro zaznamenávání digitalizovat. Každému obrazovému bodu (pixelu) musíme přiřadit hodnotu jasu u černobílého snímku nebo barevnou informaci u barevného snímku.

Pro tento účel se v dnešní době používají dva typy obrazových senzorů. Jedná se o typ snímačů CCD a o typ CMOS. Oba typy senzorů jsou složeny z fotocitlivých buněk, ve kterých optické záření generuje elektrický náboj. Tento náboj je úměrný dopadajícímu optickému záření.

1.1 CCD SNÍMAČE

Jedná se o polovodičový čip se schopností transportu elektrického náboje. V každé buňce senzoru se generuje elektrický náboj, jehož velikost odpovídá intenzitě optického záření, které na buňku dopadá. Tento náboj se následně posouvá z jedné buňky do druhé až je následně přiveden na okraj, kde je převeden na digitální podobu signálu a uložen do paměti [Obr. 1]. [1]

- + velká citlivost
- + malý šum
- vzájemné ovlivňování sousedních pixelů
- nemožnost adresovat jednotlivé pixely

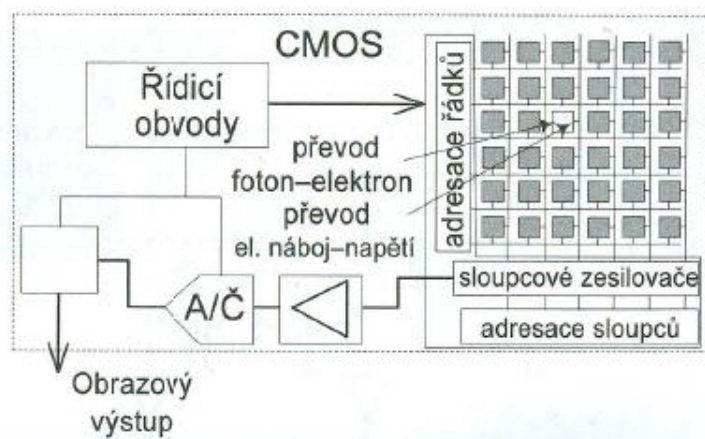


Obr. 1 CCD snímač [1]

1.2 CMOS SNÍMAČE

Tyto snímače mají světlocitlivé buňky podobné jako CCD snímače, ale velikost náboje je převáděna na napětí přímo v jednotlivých buňkách. Existují dva typy těchto senzorů a to pasivní a aktivní. [1]

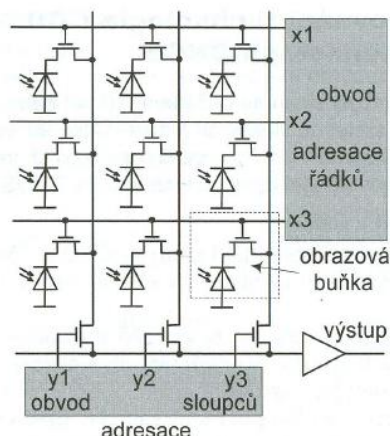
- + nízká spotřeba
- + vyšší rychlost vyčítání
- + adresovatelnost pixelů
- + menší pravděpodobnost rozmazání obrazu
- + levná výroba
- nižší činitel plnění
- větší šum



Obr. 2 CMOS snímač [1]

PASIVNÍ SNÍMAČ

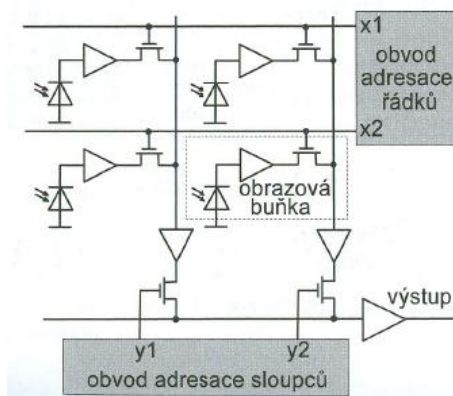
Jeho buňka je většinou tvořena fotodiodou a tranzistorem MOS-FET, který slouží jako spínač. Díky tomu lze adresovat jakoukoliv buňku.



Obr. 3 CMOS pasivní snímač [1]

AKTIVNÍ SNÍMAČ

Buňka tohoto snímače ještě navíc obsahuje zesilovač signálu. Díky tomu je dosaženo zlepšení poměru signál/šum a zvětšení citlivosti.



Obr. 4 CMOS aktivní snímač [1]

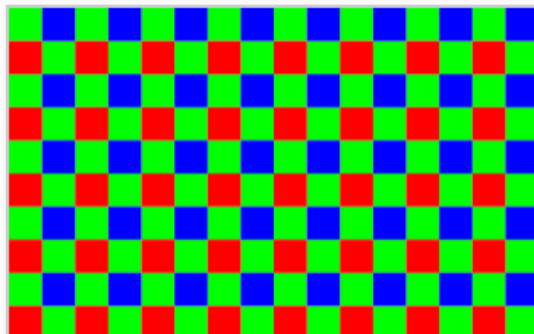
1.3 BAREVNÉ SNÍMÁNÍ OBRAZU

Uvedené snímače jsou schopny měřit pouze intenzitu dopadajícího záření, a proto pro barevné snímání je potřeba použít jednu z těchto metod. [1]

- První metoda spočívá ve snímání každé barvy zvlášť. Pro snímání jednotlivých barev jsou před senzor postupně vkládány barevné filtry. Nevýhodou této metody je nízká rychlost.

- Druhá metoda využívá také barevné filtry, které jsou ale vloženy před každou buňku senzoru. Tyto senzory využívají větší citlivosti lidského oka na žlutozelenou barvu. To vedlo k vývoji několika barevných uspořádání filtrů. Nejčastěji

používané je Bayerovo uspořádání G-R-G-B. Nevýhodou je snížení rozlišitelnosti čipu.



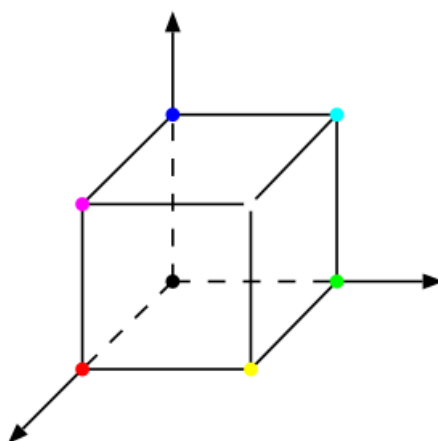
Obr. 5 Bayerovo uspořádání barevného filtru [3]

- Třetí metoda využívá optický hranol pro rozdělení jednotlivých barevných složek. Následně jsou tyto složky snímány každá zvlášť na samostatném čipu. Nevýhodou je nutnost použití tří senzorů.

1.4 BAREVNÉ MODEL Y

1.4.1 RGB MODEL

Jedná se o nejvíc rozšířený barevný model, který je využíván v běžné elektronice jako televizích, monitorech, fotoaparátech a podobně. Tento model je aditivně složen ze tří barev: červené = 700 nm, zelené = 546,1 nm a modré = 435,8 nm. Výslednou barvu dostane složením jednotlivých složek, které jsou reprezentovány svou mohutností. V desítkové soustavě má každá barva mohutnost od 0 do 255. Čím větší je součet mohutností, tím je výsledná barva světlejší. Abychom tedy získali bílou barvu musí mít všechny složky maximální mohutnost. Tento barevný model lze reprezentovat pomocí jednotkové krychle [Obr. 6].



Obr. 6 Jednotková krychle RGB [4]

Pro počítačové vidění často používáme obraz v odstínech šedi a tak je potřeba model RGB převést na hodnoty jasu. Z důvodu různé citlivosti lidského oka na jednotlivé barvy se používají následující váhové koeficienty. [2]

$$f = 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B \quad (1)$$

1.4.2 CMY A CMYK MODEL

Tento model využívá barev Cyan - modrozelená, Magenta - purpurová, Yellow - žlutá. Na rozdíl od RGB je tento model subtraktivní. Tedy čím větší mohutnost barvy, tím je výsledná barva tmavší.

Smícháním těchto tří barev však nedostaneme dokonalou černou barvu, proto se model CMY rozšiřuje na model CMYK, kde je ještě navíc červená barva. [2]

Převod mezi RGB a CMY s předpokladem jasové úrovně (0,1)

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

1.4.3 YUV MODEL

Jedná se o model, který využívá tříprvkový vektor [Y,U,V]. Y reprezentuje jasovou složku s rozsahem od 0 do 1. Složka U reprezentuje zastoupení modré barvy a složka V zastoupení červené barvy. Obě složky mají rozsah od -0,5 do 0,5. [5]

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,147 & -0,289 & 0,436 \\ 0,615 & -0,515 & -0,1 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1,137 \\ 1 & -0,397 & -0,58 \\ 1 & 2,034 & 0 \end{bmatrix} * \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \quad (4)$$

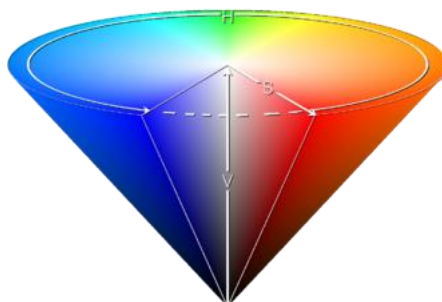
1.4.4 HSV MODEL

Tento model nejvíce odpovídá lidskému vnímání barev. Lze reprezentovat pomocí kuželu a skládá se ze tří složek. [6]

Hue - barevný odstín - je dán polohou barvy na standartním barevném kole (120° = zelená, 240° = modrá, 360° = červená)

Saturation - sytost barvy - udává množství šedi v procentech k barevnému odstínu

Value - hodnota jasu - udává relativní světlost nebo tmavost barvy

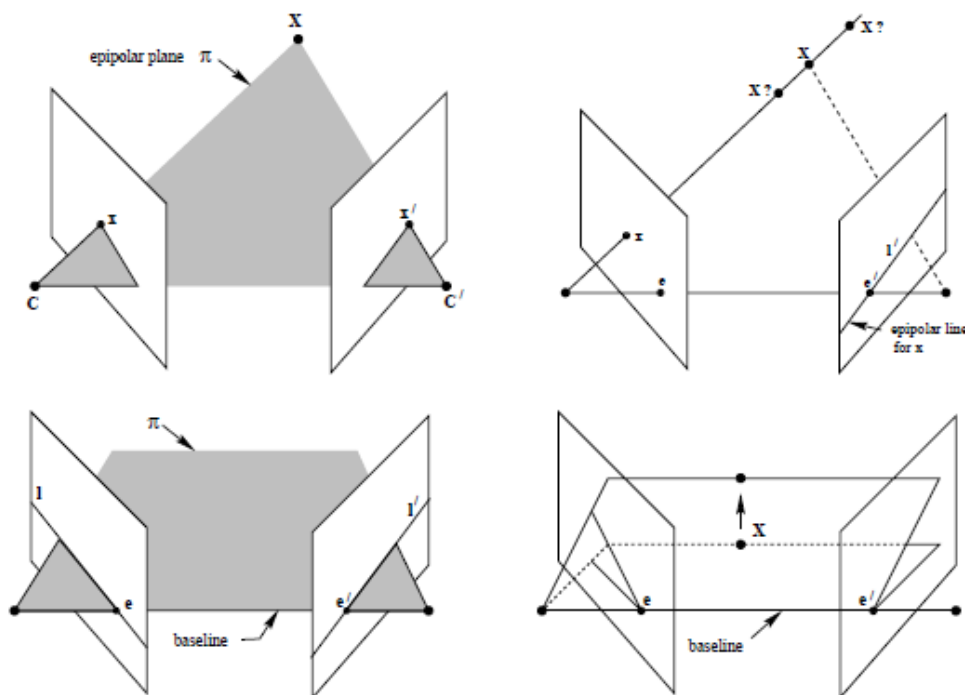


Obr. 7 HSV kužel [6]

1.5 EPIPOLÁRNÍ GEOMETRIE

Epipolární geometrie je vnitřní projektivní geometrie, která slouží pro popis vztahu mezi dvěma snímky téže scény. Důležitou vlastností této geometrie je, že není závislá na struktuře scény a závisí pouze na vnitřních parametrech kamer a jejich vzájemné pozici. Díky tomu můžeme určit vztah mezi odpovídajícími si body obou snímků. [11]

Mějme bod v prostoru, který označíme \mathbf{X} . Tento bod promítneme do obrazové roviny prvního snímku jako \mathbf{x} a do druhého snímku jako \mathbf{x}' . Dále si označíme středy promítání kamer \mathbf{C} a \mathbf{C}' . Nyní můžeme definovat několik důležitých pojmů epipolární geometrie.



Obr. 8 Epipolární geometrie [11]

Epipól - průsečík obrazové roviny se spojnicí středů kamer C a C' . Tento bod označujeme jako e a e' . V případě, kdy jsou osy kamer rovnoběžné, leží epipól v nekonečnu.

Epipolární rovina - rovina, která je definována spojnicí středů kamer C , C' a bodem v prostoru X . Tato rovina je označena jako π .

Epipolární přímka (epipolára) - průsečnice obrazové roviny a epipolární roviny. Epipoláru označujeme jako l a l' . Všechny epipoláry l se protínají v epipólu e a epipoláry l' v e' .

S měnícím se prostorovým bodem X rotuje epipolární rovina kolem spojnice středů kamer C a C' . Tímto získáme množinu epipolárních rovin, kterou nazveme jako epipolární svazek.

Těchto vlastností se využívá pro zjednodušení hledání odpovídajících si bodů. Z [Obr. 8] je vidět, že X , x , x' , C , C' leží v jedné epipolární rovině. V případě, kdy známe x , C a C' , tak máme definovanou epipolární rovinu π a epipoláru l . Hledaný bod tedy musí ležet v epipolární rovině π a zároveň v obrazové rovině. Bod x' tedy hledáme na epipoláře l' . [11]

1.6 FUNDAMENTÁLNÍ MATICE

Fundamentální matice \mathbf{F} je algebraickou reprezentací epipolární geometrie. Jedná se o matici velikosti 3×3 jejíž hodnost je 2. [11]

Vztah mezi epipóly a fundamentální maticí:

$$Fe = 0 \quad (5)$$

$$F^T e = 0 \quad (6)$$

Vztah mezi epipolární přímkou a fundamentální maticí:

$$l = F^T x' \quad (7)$$

$$l' = Fx \quad (8)$$

Pro odpovídající si obrazové body \mathbf{x} a \mathbf{x}' platí

$$x'Fx = 0 \quad (9)$$

Stejně jako epipolární geometrie, tak i fundamentální matice nezávisí na struktuře scény. Proto pro její výpočet z obrazových dat potřebujeme dostatečný počet odpovídajících si bodů.

Zapíšeme-li $\mathbf{x} = (x, y, 1)^T$ a $\mathbf{x}' = (x', y', 1)^T$ pro dvojici odpovídajících si bodů, dostaneme lineární rovnici jejíž neznáme jsou prvky fundamentální matice.

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0 \quad (10)$$

Pro n shodných bodů tedy můžeme sestavit soustavu n lineárních rovnic o devíti neznámých.

$$Af = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} f = 0 \quad (11)$$

Pro jedinečné řešení soustavy (11) je potřeba znát alespoň 8 dvojic odpovídajících si obrazových bodů. V tom případě může být řešení soustavy nalezeno pomocí lineární metody.

Pro zvýšení přesnosti fundamentální matice se většinou využívá znalosti více než 8-mi dvojic bodů. [11]

1.7 COMPUTER VISION SYSTEM TOOLBOX

Je softwarový nástroj, který obsahuje nástroje a algoritmy pro počítačové vidění. Jedná se o nový nástroj, který je součástí Matlabu teprve od roku 2011 a podléhá neustálému vývoji. Pro plné využití potenciálu tohoto toolboxu je potřeba jeho kombinace ještě s dalšími dvěma toolboxy.

První z nich je Image Processing Toolbox, který poskytuje nástroje pro zpracování obrazu. Tyto nástroje lze využít pro analýzu nebo segmentaci obrazu a jeho geometrickou transformaci.

Druhým toolboxem je Data Acquisition Toolbox jenž umožňuje přístup k hardwaru pro snímání obrázků a videa.

Algoritmy, které Computer Vision Toolbox využívá, lze rozdělit do několika následujících skupin.

DETEKCE PŘÍZNAKŮ

Tato skupina je vybavena algoritmy pro nalezení význačných bodů - příznaků v obraze. Dále umožňuje porovnání dvou sad nalezených příznaků a zobrazení výsledku.

- Shi & Tomasi, Harris a FAST metody pro detekci rohů v obraze

- metody SURF a MSER pro detekci krajů a regionů

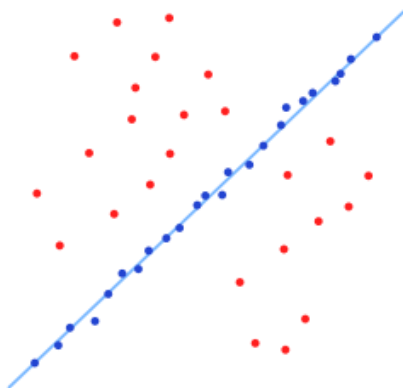
REGISTRACE A STEREOVIZE

Tato sada obsahuje algoritmy pro odhad geometrických vztahů mezi snímky zejména s využitím příznaků. Algoritmy najdou využití při stabilizaci videa nebo ve stereovizi.

Při určování geometrických vztahů můžeme využít metodu nejmenších čtverců či metodu RANSAC.

Výhoda metody RANSAC spočívá ve vynechání odlehlých bodů z množiny detekovaných bodů při odhadu parametrů. Na [Obr. 9] máme detekované body,

kteřé chceme proložit přímku. Tento algoritmus vybere náhodné body a proloží je matematickým modelem. Následně zkontroluje, jak tento model odpovídá ostatním bodům. Tento postup je opakován a nakonec je vybrán model, který nejlépe popisuje většinu bodů.



Obr. 9 Algoritmus RANSAC [7]

DETEKCE OBJEKTU, POHYBU A JEHO SLEDOVÁNÍ

Zde máme nástroje pro detekci objektu ve snímku. Můžeme využít vyhledávání podle předlohy s využitím detekce příznaků. Dále je k dispozici detektor regionů založený na vyhledávání oblasti ve snímku s určitou vlastností.

Zajímavým detektorem, který tato sada nástrojů obsahuje je algoritmus Viola-Jones. Jedná se o objektový detektor, který byl natrénován za účelem vyhledávání lidského obličeje. Kromě obličeje zahrnuje i vyhledávání nosu, očí a dalších částí lidského těla.

Pro detekci pohybu lze použít algoritmus optického toku nebo algoritmus, který pracuje na principu odhadu pozadí.

Pro sledování pohybu je k dispozici algoritmus CAMShift, který využívá změny histogramu za účelem pohybu detekčního okna spolu se sledovaným předmětem. Dále je zde Kanade-Lucas-Tomasi jenž sleduje pohyb vybraných příznaků v obraze. Tento algoritmus najde také uplatnění při stabilizaci videa. Pro případ sledování více objektů v obraze je možné tyto algoritmy kombinovat s Kalmanovým filtrem.

KALIBRACE KAMERY

Tento nástroj se používá pro odhad vnitřních a vnějších parametrů kamery a také k odhadu zkreslení obrazu objektivem. Nástroj ke kalibraci je ve formě aplikace, která zjednodušuje jeho obsluhu.

ZPRACOVÁNÍ VIDEO A ZOBRAZENÍ

V této části nalezneme nástroje ke zpracování videa. Tyto nástroje umožňují čtení či zápis běžných video formátů jako AVI, MPEG a WMV. Navíc umožňují převzorkování videa nebo vkládání textů a obrázků do snímků. Dále zde nalezneme video přehrávač.

Nakonec se je potřeba ještě zmínit, že uvedený toolbox lze využívat jak v Matlabu, tak i v Simulinku. Většina uvedených funkcí umožňuje generování ANSI/ISO C kódu s využitím Matlab Coder, Simulink Coder nebo Embedded Coder. Navíc je možné generovat data pro C kód jak s plovoucí čárkou, tak s pevnou čárkou. [10]

2 REALIZACE APLIKACÍ

Pro realizaci byl použit Matlab verze 2013b.

2.1 POUŽITÉ KAMERY

Pro snímání obrazu byly použity průmyslové kamery od německé firmy Basler. Kamery byly vybírány tak, aby byla jejich podpora s Matlabem.



Obr. 10 Kamera Basler [8]

Parametry použité kamery BASLER acA 1300 - 60gc:

Rozlišení	1280 x 1024 pixelů
Velikost pixelu	5,3 μm x 5,3 μm
Snímkovací frekvence	60 snímků za sekundu
Typ	barevná
Typ senzoru	CMOS
Rozhraní	GigE adaptér
Formát videa	Bayer RG 8, Bayer RG 12, Bayer RG 12 packed, YUV 4:2:2 packed, YUV 4:2:2 (YUYV) packed
Barevná hloubka	12 bitů
Velikost	42 x 29 x 29
Napájení	12 VDC nebo PoE

2.2 POUŽITÉ OBJEKTIVY

Použité kamery byly osazeny varifokálním objektivem firmy Computar. Objektiv má proměnnou ohniskovou vzdálenost s manuální clonou. Nastavení ohniskové vzdálenosti se provádí manuálně pomocí kroužku na objektivu. Pro zajištění jeho stabilní pozice je vybaven jistícím šroubkem.



Obr. 11 Objektiv Computar [9]

Parametry použitého objektivu H2Z0414C - MP:

Ohnisková vzdálenost	4 - 8 mm
Clona	F1,4 - 16
Optická velikost	1/2
Minimální vzdálenost (M.O.D.)	0,5 m
Rozsah spektra	viditelné
Velikost	Ø 41,6 x 48,8 mm
Horizontální úhel záběru	90,4° - 47°
Vertikální úhel záběru	67° - 35,3°

2.3 NASTAVENÍ PARAMETRŮ KAMERY

Použité kamery byly připojeny k PC pomocí ethernetového kabelu CAT5E ke Gigabitovým síťovým adaptérům. Výhodou tohoto připojení je jeho flexibilita a nízká cena kabeláže. Navíc tyto kamery lze využít v běžné síťové infrastruktuře. Nevýhodou jejich připojení je vyšší zátěž procesoru, která se později ukázala jako problematická. Pro napájení kamer nebylo využito napájení po ethernetovém kabelu, ale byl využit druhý konektor 12 VDC.

Po připojení kamer k PC je nutné nastavit jejich IP adresu a masku podsítě. Matlab vyžaduje, aby každá kamera měla nastavenou jinou masku podsítě. Dále je potřeba povolit na síťových adaptérech tzv. „Jumbo frames“. Toto nastavení je z důvodu odesílání velkých packetů po ethernetu.

Ke kamerám se v Matlabu přistupuje pomocí Data Acquisition Toolboxu. Při nainstalovaném Pylon driver jsou kamery přístupné přes adaptér winvideo. Tento adaptér umožňuje snadný výběr formátu videa a jeho rozlišení. Problém však nastal s připojením obou kamer současně. Přes winvideo se podařilo připojit jako videovstup v Matlabu pouze jednu z kamer. Pro řešení tohoto problému bylo využito podpory adaptéru GigE, který podporují nejen použité kamery, ale i novější verze Matlabu. Aby bylo možné použít adaptér GigE, tak nesmí být v PC nainstalovaný Pylon driver. Po odinstalování driveru bylo již možné připojit kamery přes GigE a hlavně bylo možné v Matlabu vytvořit videovstupy pro obě kamery.

Na snímky má velký vliv okolní osvětlení. Tento vliv se projevil poměrně tmavými snímky i při osvětlení v místnosti, které je pro lidské oko dostatečné. Použité kamery mají tři možnosti pro zesvětlení snímků.



Obr. 12 Snímek bez úpravy

Na [Obr. 1] může vidět snímek pořízený za dne v místnosti s osvětlením pomocí halogenových žárovek. Nastavení kamer je základní a tomuto nastavení odpovídá expozice 5 ms.

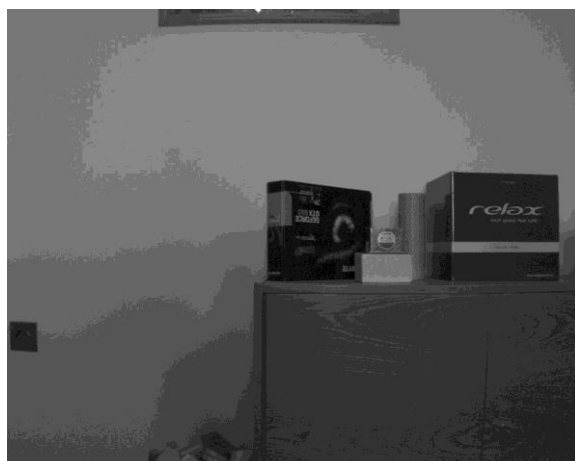
První možností zesvětlení snímku je analogové zesílení jasové složky. Výhodou je, že se nemusí zvyšovat expoziční doba, ale se zesílením se zesiluje také šum. Kamery neumožňovaly příliš velké zesílení a tudíž snímky byly stále tmavé.

Druhou možností je zvýšit expoziční dobu [Obr. 13]. Na tomto snímku se již tmavé předměty neztrácí a lze je snadno rozlišit. Naopak bílé předměty jsou poměrně přesvětlené. Doba expozice odpovídá hodnotě 70 ms. Při pořizování snímku ve venkovním prostředí za velmi slunečného dne je naopak potřeba expoziční dobu zkrátit na hodnotu okolo 2,5 ms, jelikož snímky byly velmi přesvícené. Kamery umožňují použít automatické nastavení expozice, ale takto pořízené snímky trpí malým přesvícením.



Obr. 13 Snímek s úpravou expozice

Třetí možností je použití gama korekce [Obr. 14]. A to gama korekce RGB. Takto pořízené snímky ve vnitřních i venkovních prostorech nebyly příliš tmavé a ani příliš světlé.

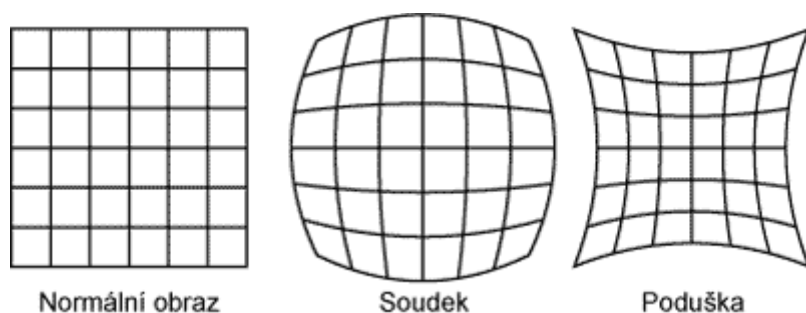


Obr. 14 Snímek s gama korekcí

2.4 KALIBRACE OBRAZU

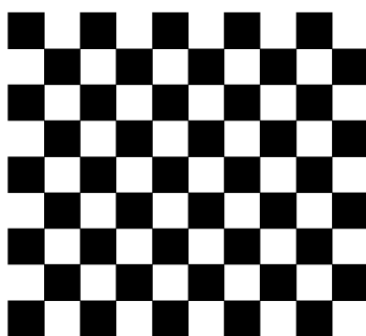
Kvalitu snímaného obrazu neovlivňuje jen použitý obrazový čip, ale také objektiv. Objektiv se skládá ze soustavy čoček, jejichž kvalita výrazně ovlivňuje obraz, který je zobrazen na snímací čip.

Čočky jsou zatíženy různými optickými vadami. Vada, která je výrazně pozorovatelná na pořízených snímcích je zkreslení obrazu. Toto zkreslení je způsobeno různým zvětšením bodů vzdálených od optické osy. Nejčastěji je pozorovatelné u širokoúhlých a zoom objektivů. Rozeznáváme dva druhy zkreslení poduška a soudek [Obr. 15].



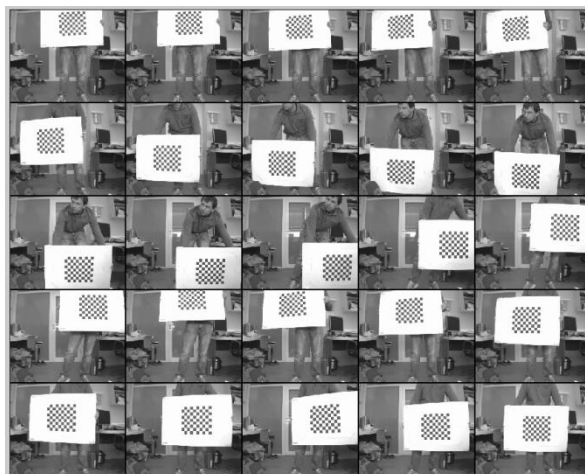
Obr. 15 Typy zkreslení obrazu [12]

Pro odstranění zkreslení obrazu lze využít Camera Calibration Toolbox. Pro kalibraci obrazu se používá kalibračním obraz, kterým je nejčastěji šachovnice. Kalibrační šachovnice nesmí být čtvercová, aby bylo možné rozlišit směr x a y . Jednotlivé čtverce šachovnice naopak musí mít stejné rozměry [Obr. 16].



Obr. 16 Kalibrační šachovnice

Připravenou šachovnici je nutné umístit na rovný podklad a vytvořit ideálně 10 až 20 snímků. Jelikož máme zoom objektiv, tak je potřeba používat stále stejné nastavení. Kalibrační šachovnicí je nutné pokrýt celý snímávaný obraz, aby kalibrace byla co nejpřesnější. Dále byly kalibrační snímky umístěny do vzdálenosti od kamer tak, aby pokryly vzdálenost zájmu.



Obr. 17 Kalibrační snímky

Při načítání obrázků toolbox sám rozpozná šachovnici v obraze. Po úspěšném načtení a identifikování jsou tyto snímky zobrazeny. Před provedením odhadu parametrů je nutné nastavit parametry, které jej ovlivňují.

RADIÁLNÍ ZKRESLENÍ

Toto zkreslení je popsáno polynomy [13]

$$x_{zkreslené} = x(1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6) \quad (12)$$

$$y_{zkreslené} = y(1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6) \quad (13)$$

$$r^2 = x^2 + y^2 \quad (14)$$

kde k_1 , k_2 , k_3 jsou koeficienty polynomu, které je potřeba odhadnout

Pro použité objektivy bylo dostačující odhadnout koeficienty k_1 a k_2 . V případě použití širokoúhlých objektivů je nutné počítat i s koeficientem k_3 .

TANGENCIÁLNÍ ZKRESLENÍ

K tomuto zkreslení dochází, když obrazová rovina a roviny objektivu nejsou rovnoběžné. I když vliv tohoto zkreslení byl malý, tak pro dobré výsledky je potřeba uvažovat.

Toto zkreslení je popsáno polynomy [13]

$$x_{zkreslené} = x + [2 \cdot p_1 \cdot y + p_2 \cdot (r^2 + 2 \cdot x^2)] \quad (15)$$

$$y_{zkreslené} = y + [p_1 \cdot (r^2 + 2 \cdot y^2) + 2 \cdot p_2 \cdot x] \quad (16)$$

$$r^2 = x^2 + y^2 \quad (17)$$

kde p_1 , p_2 jsou koeficienty polynomu, které je potřeba odhadnout

ZKOSENÍ

U některých kamer nemusí být vždy osa x a y kolmá, a proto je potřeba odhadovat i tento parametr. U použitých kamer byla uvažována kolmost mezi osami, tudíž tento parametr nebyl odhadován.



Obr. 18 Kalibrace snímku

Na [Obr. 18] vlevo můžeme vidět snímek pořízený kamerou bez úpravy. Na cihlové zdi je patrné zkreslení obrazu typu soudek, které se se vzdáleností od středu zvětšuje. Vpravo je snímek po odstranění zkreslení. Na něm můžeme vidět, že došlo k „narovnání“ spár cihel.

2.5 DETEKCE POLOHY

Pro detekování polohy objektu budeme uvažovat případ statické kamery. U objektu uvažujeme oba případy, tedy případ pohybu a případ klidu. Jako příklad použití budeme uvažovat detekci polohy vozidla Car4 pomocí značky.

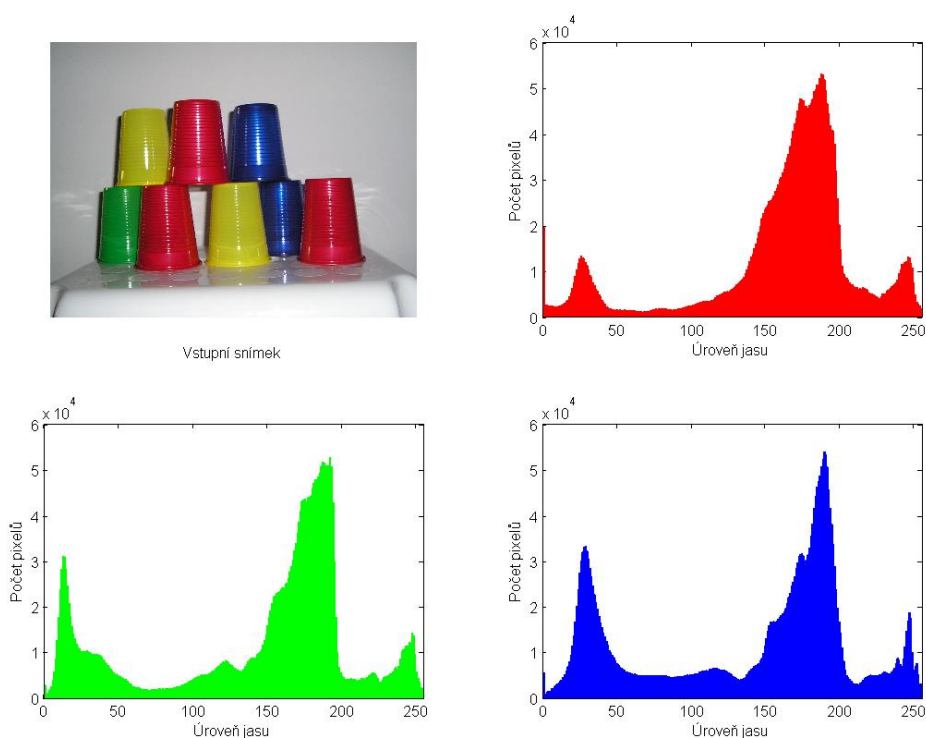
První metodou je nalezení významných bodů - příznaků značky, kterou chceme detekovat. Tyto nalezené příznaky následně porovnáme s příznaky nalezenými ve snímku prostoru, ve kterém chceme danou značku detekovat. Ne všechny nalezené shodné body si skutečně odpovídají. Pro vyloučení „špatných bodů“ je potřeba provést odhad geometrické transformace nalezených bodů.

Z uvedeného popisu vyplývá, že tato metoda není příliš rychlá. Pro správnou funkčnost vyžaduje, aby značka měla dostatek příznaků vhodných pro detekci. Vzhledem k možné rotaci značky by bylo potřeba zajistit tyto příznaky v rozsahu 360°. Z uvedené plyne, že tato metoda není příliš vhodná nejen její rychlostí detekování, ale i vysokými požadavky na značku.

Další možností je vyhledání značky na základě její charakteristiky. Jako charakteristiku značky můžeme uvažovat její barvu nebo její tvar. Tato metoda spočívá v oddělení značky na základě její charakteristiky od pozadí. Za účelem

detekce tedy snímek převedeme do binární podoby tak, že pozadí potom bude reprezentováno hodnotou 0 a značka hodnotou 1.

V našem případě budeme uvažovat jako charakteristiku barvu značky, kterou je nyní potřeba oddělit od pozadí. Abychom určili práh jasu jednotlivých barev, tak je potřeba určit histogram jednotlivých RGB složek snímku. Pomocí histogramu určíme četnost jednotlivých úrovní jasu barevných složek. Na [Obr. 19] můžeme vidět snímek obsahující různé barvy a také jeho histogramy složek RGB.

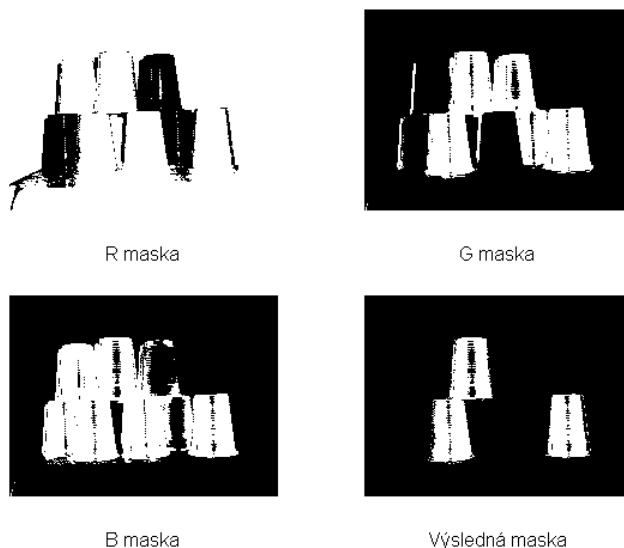


Obr. 19 Snímek a jeho histogramy

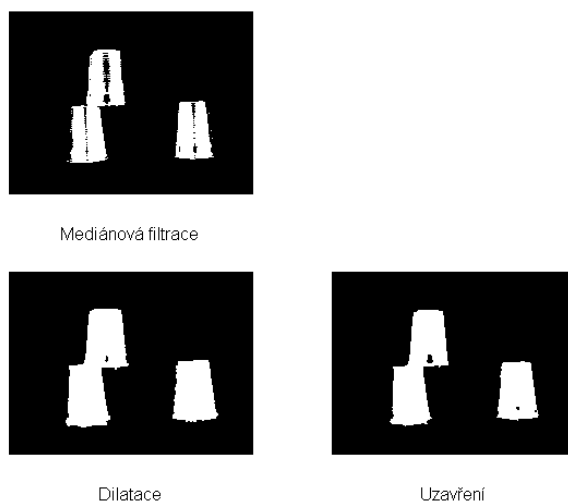
Nyní na základě histogramů určíme prahy jednotlivých složek. Pokud budeme chtít vyloučit červenou barvu, tak interval jasu složky R vybereme v místě s největší četností. Naopak složky G a B chceme zahrnout do okolí, tudíž jejich intervaly budou obsahovat nejnižší četnosti jasu. Nakonec složíme jednotlivé masky dohromady a dostaneme výslednou masku, kde hledaná červená barva je reprezentována bílými regiony v binárním obrazu. Tyto masky můžeme vidět na [Obr. 20].

Pro určení prahů jednotlivých jasových složek můžeme také využít funkci pro jeho automatické nalezení jenž je obsažena v Matlabu. Automatické nalezení prahu nedosahovalo tak dobrých výsledků jako manuální nastavení.

Na výsledné masce červené barvy můžeme stále vidět černé body, které zakrývají červenou barvu. Toto je způsobeno nejen šumem v obraze, ale také odlesky, který jsou patrné na vstupním snímku. Pro odstranění těchto jevů je potřeba výslednou masku filtrovat.



Obr. 20 Bitová maska



Obr. 21 Filtrace masky

Výslednou masku můžeme filtrovat pomocí mediánové filtrace. Vlivem velkého černého pruhu je v tomto případě tato metoda nevhodná. Je nutné použít porovnávací matici o velikosti alespoň 8x8. Tato operace je potom časově náročná a to v řádu desetin sekundy.

Další možností je použití operace dilatace. Tato operace poskytla mnohem lepší výsledek a to za cenu pouze několika setin sekundy. Prvkem pro dilataci byla použita kružnice.

Operace uzavření poskytla téměř shodný výsledek jako dilatace při použití stejného prvku a to za téměř shodnou dobu.

Na takto připravený snímek můžeme již využít nástroj pro detekci, kterým je detektor regionů. Pro zpřesnění detekovaného objektu ještě nastavíme minimální nebo maximální počet pixelů daného objektu. Detektor je schopný vrátit střed detekovaného regionu nebo souřadnice obdélníku, který daný region obklopuje.

Uvedený detekční algoritmus je poměrně rychlý. Operace úpravy snímku a detekce trvá několik setin sekundy. Díky rychlosti algoritmu lze tento způsob detekce využívat v každém kroku.



Obr. 22 Detekovaná značka

Ne všechny detekční algoritmy jsou tak rychlé. Jako příklad si můžeme vzít detektor Viola-Jones. Tento detektor využijeme pro detekci obličeje ve snímku. Právě detekování obličeje trvá v řádu několika desetin sekundy, což je opravdu rozdíl oproti předchozí detekci.

Řešením je využití nástroje pro sledování pohybu. Nejprve je obličej detekován Viola-Jonesem a je určena jeho poloha. Jako další přijde na řadu sledování pohybu. Nástroj, který máme k dispozici, je algoritmus CAMShift využívající změnu histogramu v závislosti na pohybu detekovaného obličeje. Vyhledávání obličeje tedy není prováděno v každém kroku a detekční obrazec se v obraze posouvá v závislosti na pohybu obličeje.

Problém nastane, když se detekovaný obličej ztratí mimo záběr kamery a po té se znovu vrátí. V tom případě tento algoritmus není schopen zpět obličej detekovat.

Když nastane tento případ je nutné opět provést detekci zmíněným detektorem. Při zpětném detekování však dochází k výraznému zpomalení. Po nalezení nové polohy se opět využije algoritmu CAMShift.

Dalším problémem při použití detektoru Viola-Jones je špatné identifikování objektu. Tento případ byl u pokusech s tímto detektorem častý. Ke špatnému detekování docházelo v případech rozmanitého pozadí snímku, kdy byly některé předměty vyhodnoceny jako obličeje. K odstranění špatných detekcí v případě obličeje je možné ve snímku ještě detekovat například nos. Porovnáním polohy detekovaného nosu a obličeje potom můžeme předcházet chybným detekcím, avšak za cenu času potřebného pro vyhledání obou prvků.



Obr. 23 Viola-Jones detekce obličeje

Rychlost detektoru regionů včetně úpravy snímku a algoritmu CAMShift je prakticky stejná. Výhodou detektoru regionů je přesnější poloha detekovaného objektu. Navíc detektor regionů je imunní proti ztrátě detekovaného předmětu ze záběru a jeho opětovnému návratu.

2.6 VÝPOČET HLOUBKOVÉ MAPY

Prvním krokem pro výpočet hloubkové mapy je rektifikace obrazu. Mezi obrazy z obou kamer je nutné určit geometrii tak, aby odpovídající si body v obou obrazech ležely na jedné epipolární přímce.

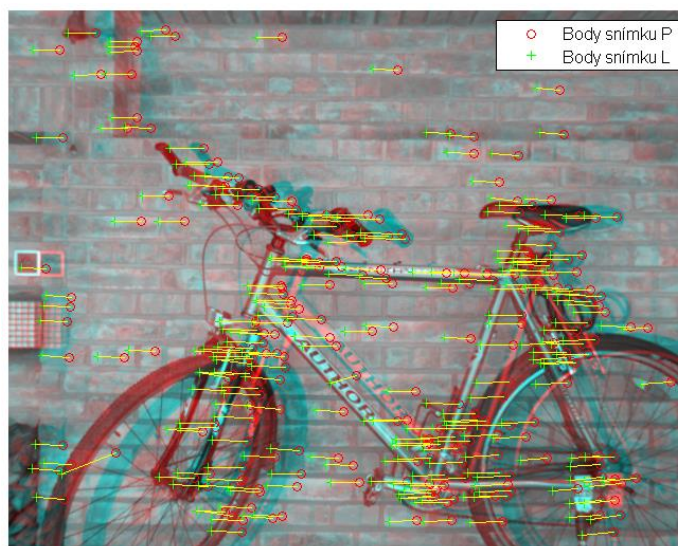
Pro rektifikaci snímků je nutné určit jejich příznaky. K určení příznaků byla použita metoda SURF. Výhodou příznaků SURF je, že jsou neměnné při změně měřítka, natočení, či změně jasu nebo kontrastu.



Obr. 24 Detekované příznaky

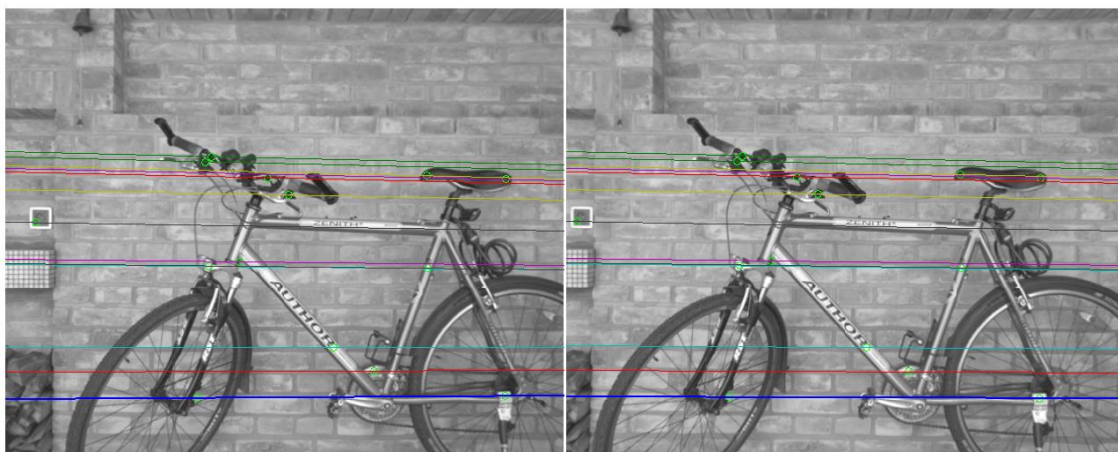
Na [Obr. 24] můžeme vidět deset nejsilnějších detekovaných příznaků metodou SURF. Tyto nalezené příznaky jsou reprezentovány na snímku polohou označenou + a kružnicí, která reprezentuje jejich velikost. Kromě těchto zobrazených parametrů obsahuje každý příznak hodnotu jeho významnosti a informaci o tom, zda jde o světlý objekt na tmavém pozadí či naopak.

Když máme určeny příznaky v jednotlivých snímcích, tak je potřeba určit odpovídající si příznaky mezi snímky. Jednotlivé příznaky jsou mezi snímky porovnány a jsou přiřazeny k sobě ty odpovídající si. Na [Obr. 25] můžeme vidět tyto vykreslené polohy příznaků. Na obrázku je také vidět, že shodnost některých příznaků není určena správně. Chybně určené příznaky můžeme vyloučit pomocí epipolární geometrie.



Obr. 25 Určení shodných příznaků

K určení epipolární geometrie je potřeba určit fundamentální matici. K jejímu určení použijeme již nalezené příznaky. V Matlabu je implementováno několik metod pro její určení. Při pokusech se nejvíce osvědčila metoda Ransac a Least Trimmed Squares. Tyto metody umožňují nastavení maximální kontrolované vzdálenosti mezi odpovídajícími si body při odhadu fundamentální matice. Metoda Least Median of Squares a metoda osmi bodového algoritmu se používá pouze k odhadu fundamentální matice a to v případě, že si všechny příznaky v obou snímcích odpovídají.



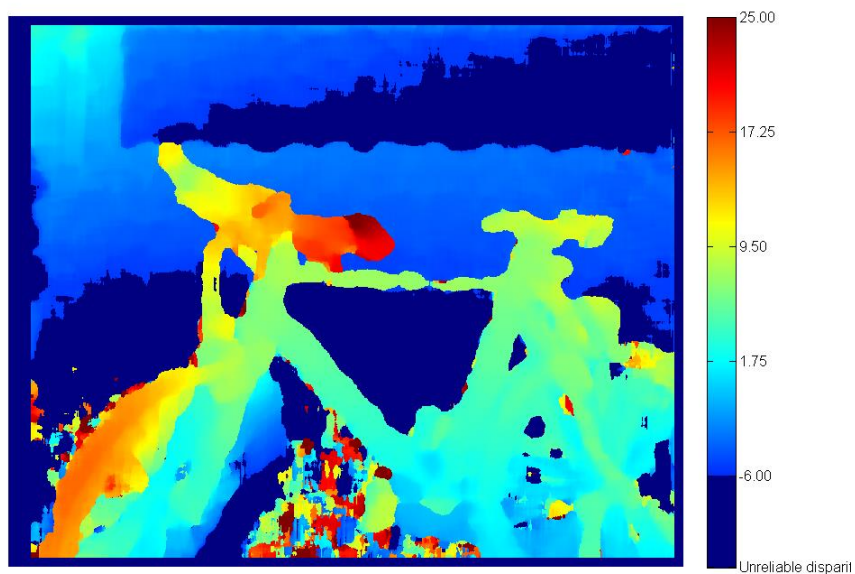
Obr. 26 Epipolární přímky

Ze znalosti fundamentální matice a odpovídajících si bodů můžeme určit epipolární přímky, které lze nyní vykreslit [Obr. 26]. Další bodem rektifikace obrazu je odhadnout geometrickou transformaci mezi těmito snímky tak, aby epipolární přímky byly vodorovné a odpovídající si body ležely na stejné epipolární přímce. Na [Obr. 26] lze také vidět, že se jedná o snímky před geometrickou transformací. Jednotlivé epipolární přímky jsou stále skloněné.

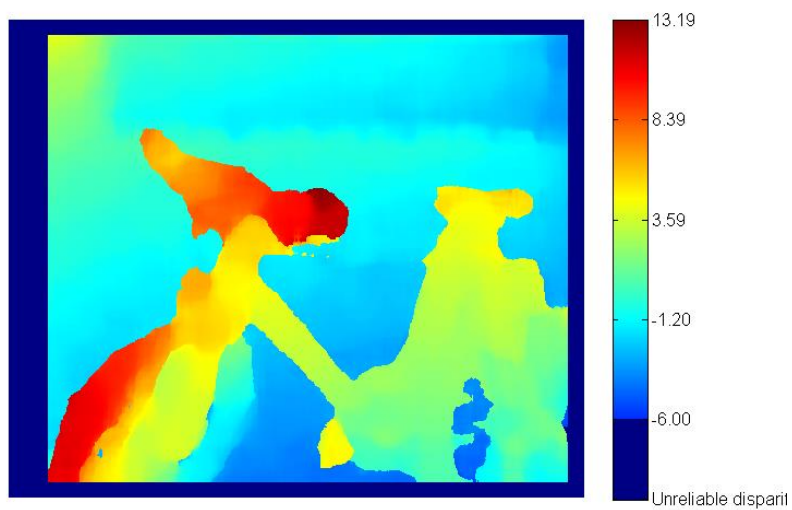
Pro výpočet hloubkové mapy je důležité, aby odhadnutá fundamentální matice byla co nejpřesnější. Ve více než 50% případů odhadů geometrické transformace, která vychází z fundamentální matice, bylo na transformovaném snímku patrné, že se jedná o nesprávný výsledek.

Po geometrické úpravě snímků může již následovat výpočet hloubkové mapy. V Matlabu je využíván algoritmus Block matching. Pro dobré výsledky je potřeba provést několik nastavení. Prvním je velikost bloku pixelů, které jsou mezi sebou porovnávány. Výsledkem je rozdíl vzdálenosti mezi shodnými pixely ve snímcích. Vypočítané vzdálenosti je přiřazena hodnota z nastaveného relativního rozsahu. Tento rozsah se nastavuje jako další parametr. Nejvíce se osvědčil rozsah 0 až 64.

Na [Obr. 27] a [Obr. 28] můžeme vidět vypočítané hloubkové mapy. Pro jejich vykreslení byla použita verze Matlabu 2012a, jelikož tato verze umožňovala při vykreslení zobrazit stupnici relativní hloubky. Při srovnání hloubkové mapy při rozlišení 1280x1024 a 640x512 můžeme vidět několik rozdílů. Při větším rozlišení je na obrázku vidět více detailů než při polovičním rozlišení. Více detailů je však vykoupeno větším šumem.



Obr. 27 Hloubková mapa při rozlišení 1280x1024

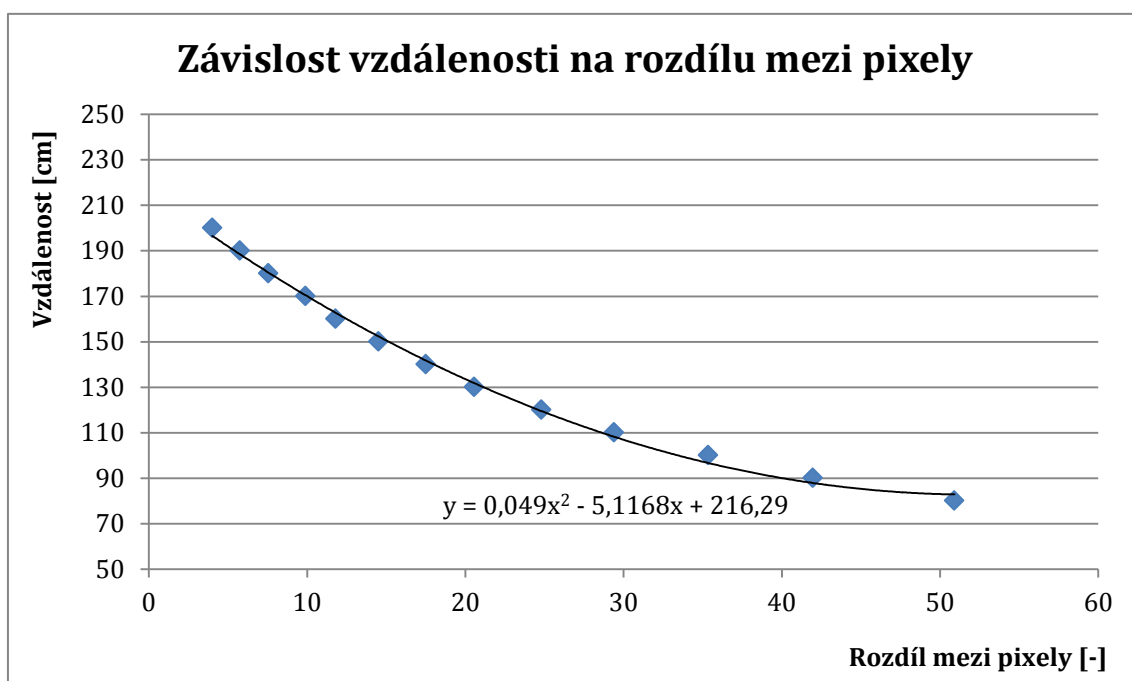


Obr. 28 Hloubková mapa při rozlišení 640x512



Obr. 29 Měření vzdálenosti

Abychom mohli určit skutečnou vzdálenost bylo nutné určit její přepočít. Za tímto účelem bylo provedeno experimentální měření. K měření bylo použito prkno, jehož vzdálenost byla měřena pomocí metru a potom pomocí výpočtu hloubkové mapy. Vzdálenost byla měřena v rozsahu 80 až 220 cm každých 10 cm. Měření bylo čtyřikrát zopakováno a následně byly průměrné hodnoty vyneseny do grafu, kde byly proloženy polynomem. Výsledný polynom určuje vztah mezi relativní vzdáleností určenou výpočtem hloubkové mapy a skutečnou vzdáleností v centimetrech.



Graf. 1 Závislost vzdálenosti

2.7 VÝSLEDNÁ APLIKACE

Jako výsledná aplikace byla vytvořena aplikace pro detekci překážek s využitím v projektu Car4. Kamery budou namontovány na vozidlo a budou snímat scénu před vozidlem za účelem detekce překážek. Informace o překážce potom bude použita pro změnu trajektorie jeho pohybu.

V případě detekování překážky je potřeba si položit otázku jakým způsobem jí vlastně budeme detekovat. Výše uvedené způsoby detekce na základě příznaků nebo na základě charakteristiky překážky použít nejdou. Takto bychom byli schopni zaznamenávat pouze předem předpokládané překážky s předem danou vlastností. Další možností detekování je detekce na základě pohybu. Tu ale také nelze použít. Vzhledem ke kameře by se nepohybovali jen překážky, ale i okolí.

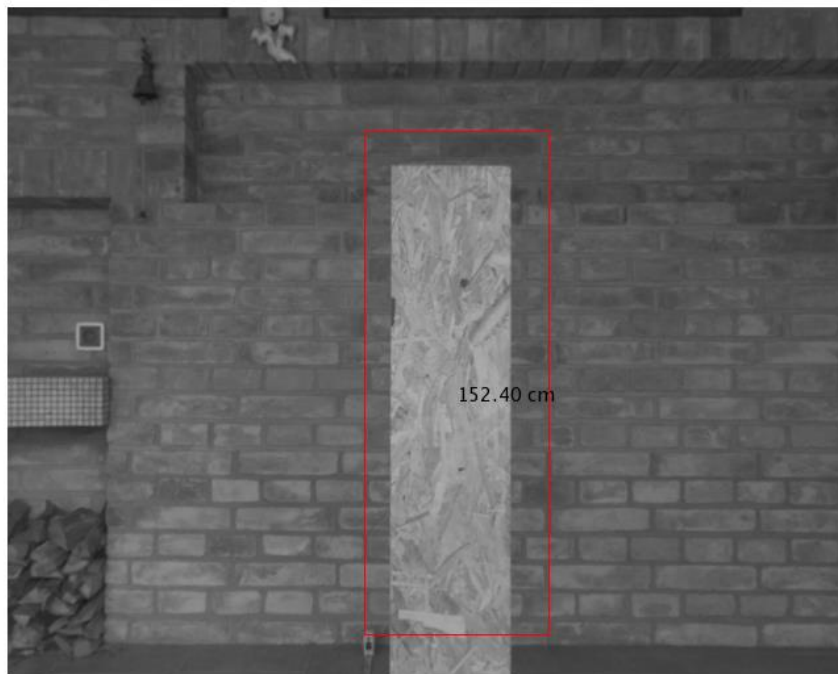
Pro detekování překážek tedy musíme použít jiný způsob. Tento způsob spočívá v určení hloubkové mapy a na jejím základě detekování překážky. Jako překážku budeme brát předměty, které jsou ve vzdálenosti od kamery blíže než 180 cm. Vhledem k minimální vzdálenosti dané objektivem 50 cm budeme uvažovat minimální vzdálenost od kamery 70 cm.

S přibližováním kamery k překážce vypočítaná relativní vzdálenost roste. Abychom mohli překážku detekovat, tak ve vypočítané hloubkové mapě budeme vyhledávat hodnoty relativní vzdálenosti větší, než které odpovídají vzdálenosti 180 cm. Tyto hodnoty nahradíme jedničkou a ostatní nulou. Tímto způsobem dostaneme snímek logických hodnot, kde bílé regiony odpovídají překážkám. Na takto upravený snímek můžeme použít již zmíněný detektor regionů, jehož návratová hodnota bude střed regionu. Abychom zamezili detekování šumu, tak nastavíme velikost minimálního detekovaného regionu na větší hodnotu počtu pixelů.

Když máme detekovanou překážku, tak je potřeba vypočítat její vzdálenost. Pro výpočet vzdálenosti bychom mohli použít hodnotu určenou středem detekovaného regionu. Pokud bychom použili pouze jen jednu hodnotu, tak výsledná vzdálenost by nemusela u členitějších povrchů dosahovat vysoké přesnosti. Pro zvýšení přesnosti je tedy nutné počítat s více hodnotami. Výsledná aplikace používá pro výpočet vzdálenosti 81 hodnot, které jsou v okruhu detekovaného středu překážky. Z těchto hodnot je vytvořena průměrná hodnota, která je použita pro výsledný výpočet skutečné vzdálenosti. Při ověření přesnosti dosahoval uvedený postup přesnosti ± 10 cm.

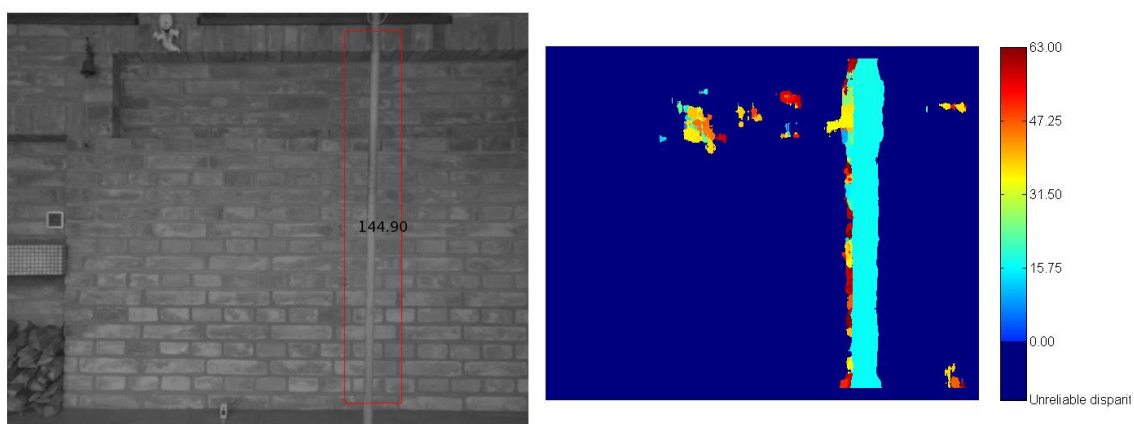
Na [Obr. 30] je vidět použití výsledné aplikace při detekci překážky. Překážka je detekována a je vypočítána její vzdálenost, která je zobrazena u překážky.

Vzdálenost vypočítaná z hloubkové mapy je 152,40 cm. vzdálenost překážky, kterou jsme určili metrem, je 160 cm.

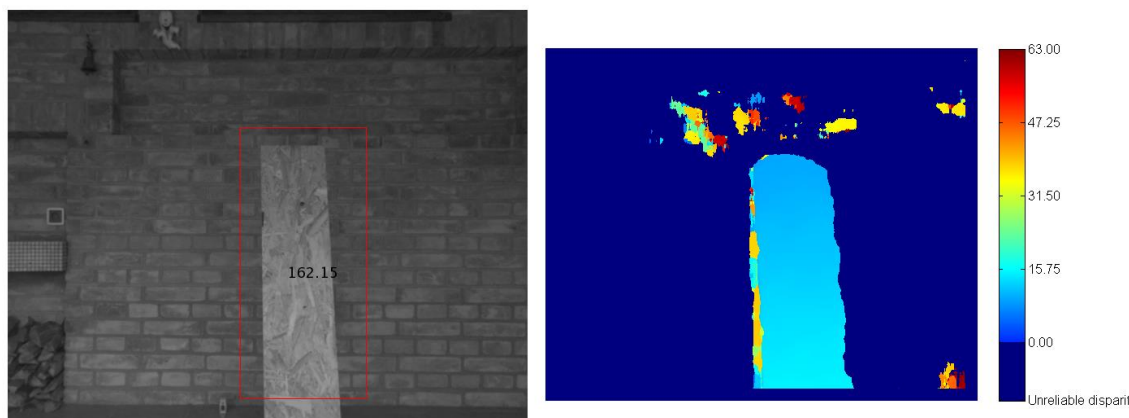


Obr. 30 Detekovaná překážka

Pro ověření detekčního algoritmu bylo vyzkoušeno několik překážek. Na [Obr. 31] můžeme vidět překážku v podobě tyče o průměru 2,5 cm. Vpravo je zobrazena hloubková mapa a vlevo vidíme detekovanou tyč s vypočítanou vzdáleností 144,90 cm. Skutečná vzdálenost tyče byla 150 cm.

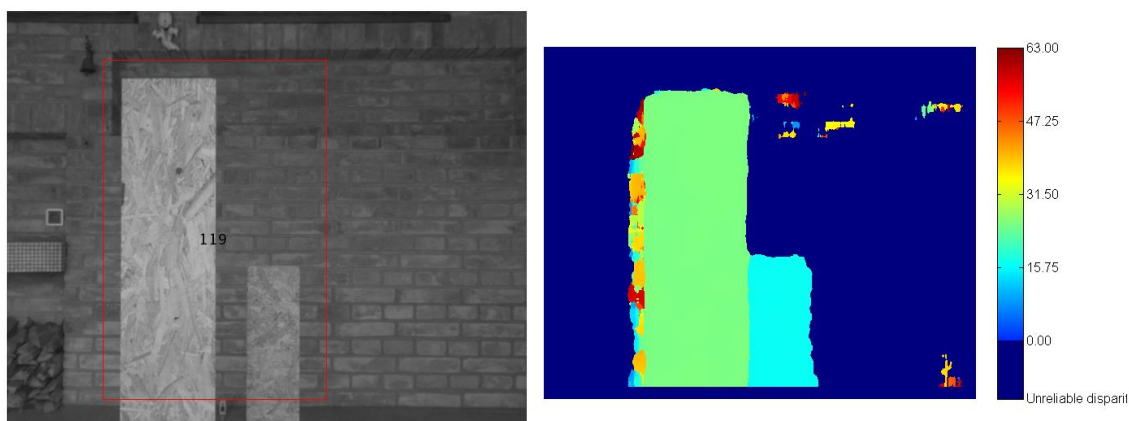


Obr. 31 Detekce tyče



Obr. 32 Detekce nakloněné překážky

Dalším případem je nakloněná překážka ať ve vertikální ose nebo v ose horizontální. Na [Obr. 32] máme překážku nakloněnou v horizontální ose. S detekcí algoritmus problém neměl, ale vypočítaná vzdálenost je 162,15 cm. Spodní hrana překážky, která byla blíže ke kamerám, byla vzdálena 140 cm. Algoritmus výpočtu bral v úvahu hodnoty kolem středu překážky, který byl od kamer vzdálen více než 140 cm. Obdobná chyba výpočtu vzdálenosti by nastala při naklonění ve vertikální ose.



Obr. 33 Detekce dvou překážek

Případ, kdy máme dvě překážky blízko sebe můžeme vidět na [Obr. 33]. Z hloubkové mapy vpravo je patrné, že levá překážka je blíže než pravá. Detektor však tyto dvě překážky sloučil. Vypočítaná vzdálenost je 119 cm a naměřená vzdálenost levé překážky byla 120 cm a pravé překážky 165 cm. Výpočetní algoritmus opět vycházel z hodnot okolo středu detekčního obrazce.

Detekce překážky a výpočet její vzdálenosti musí být rychlý, aby při jejím detekování mohla být změněna trajektorie ještě před srážkou. Při spuštění vytvořené aplikace v reálném čase nastal problém s přetížením procesoru i přes použitou výkonnou počítačovou sestavu. Použitý procesor Intel Core i5-3570 dosahoval při výpočtech hloubkové mapy při maximálním rozlišení 1280x1024 vytížení až 100%. To mělo za následek ztrátu síťových paketů, které nebyl procesor schopen zpracovat, a pád programu. Řešením bylo snížení rozlišení a tím i snížení objemu přenášených a zpracovávaných dat. Při pokusu nastavit kamery na poloviční rozlišení z prostředí Matlabu bylo toto nastavení kamer ignorováno. Tudíž byla použita funkce v Matlabu pro zmenšení rozlišení snímků na polovinu až po jejich sejmutí. Tímto způsobem klesla zátěž procesoru o pouhých 8 až 10% a při zatížení procesoru ještě ostatními procesy opět docházelo ke ztrátě paketů. Výsledná aplikace je tedy náročná na výpočetní výkon, zejména samotný výpočet hloubkové mapy.

Jelikož vozidlo Car4 bylo v době vývoje aplikace nepojízdné, tak nebyl uvedený algoritmus detekce překážek vyzkoušen přímo na vozidle.

ZÁVĚR

Cílem této práce je vytvoření aplikace pro detekci objektu a aplikace využívající stereovize pro výpočet hloubkové mapy dané scény s použitím Computer Vision System Toolboxu.

Pokud chceme využívat nástroje stereovize, tak je důležité se seznámit s epipolární geometrií. V práci se seznámíme nejen s touto geometrií, ale i s jejím matematickým popisem - fundamentální maticí.

Práce se také věnuje popisu Computer Vision System Toolboxu. Popis je rozdělen do několika skupin, ve kterých jsou popsány nástroje jednotlivých kategorií.

Aplikace počítačového vidění vyžadují nejen správný výběr hardwaru, ale také jeho správné nastavení v dané aplikaci. Velký vliv zejména na aplikaci detekce podle jeho barvy mělo osvětlení. Snímek není ovlivněn pouze osvětlením, ale také objektivem. Tento vliv se projevuje zkreslením obrazu, které je potřeba odstranit.

Dále se zabýváme vytvořením aplikace pro detekci objektu. Jsou rozebrány možnosti detekce a je uveden popis výsledné aplikace pro detekování objektu. Výsledná aplikace je založena na detekci značky určité barvy. Nejenom, že tato detekce dosahuje dobré rychlosti, ale i vyšší přesnosti než sledování objektu na základě změny histogramu.

Na tuto aplikaci navazuje aplikace určená pro výpočet hloubkové mapy. Jsme seznámeni s rektifikací obrazu, která je důležitým krokem pro určení hloubkové mapy.

Nakonec jsme seznámeni s aplikací pro výsledné využití na projektu Car4. Její využití je v detekci překážek před vozidlem a určení jejich vzdálenosti.

Z důvodu nepojízdnosti vozidla Car4 nebyl výsledný algoritmus přímo využit na vozidle.

POUŽITÉ INFORMAČNÍ ZDROJE

- [1] DOLEČEK, Jaroslav. *Moderní učebnice elektroniky : 3. díl Optoelektronika*. Praha : Ben, 2007. *Obrazové senzory*, s. 97-107.
- [2] DOBEŠ, Michal. *Zpracování obrazu a algoritmy v C#*. Praha : Ben, 2008. *Digitální fotografie a digitalizace obrazové informace*, s. 9-29.
- [3] Wikipedia [online]. [cit. 2014-05-13]. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Charge-coupled_device#Sn.C3.ADm.C3.A1n.C3.AD_obrazu>.
- [4] Wikipedia [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/RGB>>.
- [5] Wikipedia [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/YUV>>.
- [6] Wikipedia [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/HSV>>.
- [7] Wikipedia [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://en.wikipedia.org/wiki/RANSAC>>.
- [8] Basler [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://www.baslerweb.com/products/ace.html?model=372>>.
- [9] VISIONX [online]. [cit. 2014-05-13]. Dostupné z WWW: <<http://www.visionx.cz/produkty/podle-vyrobce/cbc-group/varifokalni-objektivy-computar/>>.
- [10] Mathworks [online]. [cit. 2013-02-20]. Datasheet Computer Vision System Toolbox. Dostupné z WWW: <http://www.mathworks.com/products/computer-vision/index.html?s_tid=brdcrb>.
- [11] HARTLEY Richard, ZISSERMAN Andrew. *Multiple View Geometry in computer vision*. Cambridge University Press, 2003
- [12] Fotoroman [online]. [cit. 2014-05-13]. Dostupné z WWW: <http://www.fotoroman.cz/glossary2/2_sfera.htm>.
- [13] Mathworks [online]. [cit. 2014-05-13]. Camera calibration toolbox. Dostupné z WWW:

<<http://www.mathworks.com/help/vision/ug/find-camera-parameters-with-the-camera-calibrator.html>>

- [14] Elektrorevue [online]. [cit. 2014-05-13]. Epipolární geometrie.
Dostupné z WWW:

<<http://www.elektrorevue.cz/clanky/05017/index.html>>.

- [15] VISIONX [online]. [cit. 2014-05-13]. Basler ACE. Dostupné z WWW:

<<http://www.visionx.cz/produkty/podle-vyrobce/basler/ace-gige-vision/>>.

- [16] [cit. 2014-05-13]. Dostupné z WWW:

<http://courses.engr.illinois.edu/cs543/sp2011/lectures/Lecture%2023%20-%20Epipolar%20Geometry%20and%20Stereo%20-%20Vision_Spring2011.pdf>.

- [17] [cit. 2014-05-13]. Dostupné z WWW:

<http://www.nti.tul.cz/cz/images/7/71/Prednaska_barva.pdf>.

SEZNAM OBRÁZKŮ

Obr. 1 CCD snímač [1]	11
Obr. 2 CMOS snímač [1].....	12
Obr. 3 CMOS pasivní snímač [1]	13
Obr. 4 CMOS aktivní snímač [1].....	13
Obr. 5 Bayerovo uspořádání barevného filtru [3].....	14
Obr. 6 Jednotková krychle RGB [4]	15
Obr. 7 HSV kužel [6].....	16
Obr. 8 Epipolární geometrie [11].....	17
Obr. 9 Algoritmus RANSAC [7]	20
Obr. 10 Kamera Basler [8].....	22
Obr. 11 Objektiv Computar [9]	23
Obr. 12 Snímek bez úpravy	24
Obr. 13 Snímek s úpravou expozice.....	25
Obr. 14 Snímek s gama korekcí	26
Obr. 15 Typy zkreslení obrazu [12]	26
Obr. 16 Kalibrační šachovnice	27
Obr. 17 Kalibrační snímky	27
Obr. 18 Kalibrace snímku	29
Obr. 19 Snímek a jeho histogramy	30
Obr. 20 Bitová maska	31
Obr. 21 Filtrace masky	31
Obr. 22 Detekovaná značka	32
Obr. 23 Viola-Jones detekce obličeje	33
Obr. 24 Detekované příznaky.....	34
Obr. 25 Určení shodných příznaků	35
Obr. 26 Epipolární přímky	36
Obr. 27 Hloubková mapa při rozlišení 1280x1024	37
Obr. 28 Hloubková mapa při rozlišení 640x512.....	37
Obr. 29 Měření vzdálenosti.....	38
Obr. 30 Detekovaná překážka	40
Obr. 31 Detekce tyče.....	40
Obr. 32 Detekce nakloněné překážky	41
Obr. 33 Detekce dvou překážek	41