

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

REKONSTRUKCE 3D OBJEKTU Z OBRAZOVÝCH DAT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ONDŘEJ AMBROŽ

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

REKONSTRUKCE 3D OBJEKTU Z OBRAZOVÝCH DAT

3D OBJECTS RECONSTRUCTION FROM IMAGE DATA

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

VEDOUCÍ PRÁCE
SUPERVISOR

ONDŘEJ AMBROŽ

Ing. MICHAL ŠPANĚL

BRNO 2008

Abstrakt

Práce se zabývá 3D rekonstrukcí objektu otáčejícího se před statickou kamerou o přesně daný úhel. Pokouší se vysvětlit základní princip této rekonstrukce za předpokladu, že jsou známy všechny parametry kamery.

Klíčová slova

Epipolární geometrie, parametry kamery, vnitřní parametry kamery, vnější parametry kamery, fundamentální matice, významné body, OpenCV.

Abstract

The work concerns 3D reconstruction of an object making a strictly set angle turn in front of a static camera. It tries to explain the basic principle of this reconstruction given all the camera parameters are known.

Keywords

Epipolar geometry, camera parameters, intrinsic camera parameters, extrinsic camera parameters, fundamental matrix, feature points, OpenCV.

Citace

Ondřej Ambrož: Rekonstrukce 3D objektu z obrazových dat, bakalářská práce, Brno, FIT VUT v Brně, 2008

Rekonstrukce 3D objektu z obrazových dat

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Michala Španěla

.....
Ondřej Ambrož
12. května 2008

Poděkování

Děkuji vedoucímu práce panu Ing. Michalu Španělovi za rychlé odpovědi na mé časté emaily a za to, že si vždy našel čas, pokud jsem potřeboval poradit.

© Ondřej Ambrož, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Grafické a matematické základy	4
2.1	Popis bodu	4
2.2	Homogenní souřadnice	4
2.3	Transformace ve 3D	5
2.4	Perspektivní projekce	6
2.5	SVD	7
2.6	Definice a geometrický význam SVD	8
3	Zpracování obrazu	10
3.1	Významné body	10
3.2	Harrisův detektor	10
3.3	SIFT	11
3.4	Korespondence významných bodů	12
4	Teorie pro 3D rekonstrukci	13
4.1	Vnitřní parametry kamery	13
4.2	Vnější parametry kamery	14
4.3	Projekční matice model projektivní kamery	14
4.4	Fundamentální matice a epipolární geometrie	14
4.5	Lineární triangulační metoda	15
5	Návrh řešení	17
5.1	Získání testovacích dat	17
5.2	Kalibrace kamery	18
5.3	Vyhledání významných bodů	18
5.4	Nalezení korespondencí ve vyhledaných bodech	18
5.5	Výpočet fundamentální matice	19
5.6	Výpočet epipolárních přímek	19
5.7	Triangulace korespondencí	19
5.8	Zpětná projekce	19
5.9	Pootočení získaných bodů o deset stupňů	20
6	Implementace	21
6.1	Získání testovacích dat	21
6.2	Kalibrace kamery	21
6.3	Vyhledání významných bodů	22

6.4	Nalezení korespondencí ve vyhledaných bodech	22
6.5	Výpočet fundamentální matice	22
6.6	Výpočet epipolárních přímk	23
6.7	Triangulace korespondencí	23
6.8	Zpětná projekce	23
6.9	Pootočení získaných bodů o deset stupňů	23
7	Výsledky	24
7.1	Rekonstrukce kalibračního vzoru.	24
7.2	Rekonstrukce ze dvou pohledů	25
7.3	Zhodnocení výsledků	25
7.4	Návrhy na další práci	26
8	Závěr	30

Kapitola 1

Úvod

Termín rekonstrukce 3D objektů spadá do kategorie v počítačové grafice, jež se nazývá *počítačové vidění* (angl. *Computer vision*). Tím se dostávám i k základu využití této techniky. Hlavní myšlenkou bylo umožnit robotům, automaticky řízeným autům atd., poznat hloubku objektů, a tak se pohybovat v prostoru a rozeznávat objekty. V 70. letech, kdy byl počáteční rozvoj této myšlenky, se dospělo ke klamnému zdání, že se nebude jednat o nic extrémně složitého. S postupem času se však ukázalo, že úplně jednoduché to nebude. Až v roce 2005 se povedlo dokončit závod DARPA Grand Challenge, ve kterém se mělo automaticky řízené auto dostat z bodu A do bodu B. Závod se konal v poušti. Po tomto úspěchu se dnes již jezdí nový závod DARPA Urban Challenge, ve kterém se mají automatické automobily pohybovat ve městském provozu a reagovat na dopravní situace, dokonce i automaticky parkovat. První závod se konal v roce 2007. Více na [6].

Tato bakalářská práce se však nezabývá problémem rozeznávání objektů a pohybu v prostoru, ale snaží se vysvětlit postup rekonstrukce statického objektu, který se nachází na otočném stolku a je snímán digitálním fotoaparátem nebo kamerou. Objektem si při každém pořízení snímku otočím o deset stupňů. Tak vznikne soubor 36 snímků, z nichž se pak snažím zrekonstruovat původní body v prostoru. Postup rekonstrukce a metody, které jsou k ní potřeba popisují v této práci.

Ve druhé kapitole se zabývám geometrickými a grafickými základy. Například popisem bodu. Následuje kapitola věnovaná zpracování obrazu a vyhledávání významných bodů. Věnuje se detektoru SIFT a Harrisově detektoru. Ve třetí kapitole je pak uveden nástin teorie pro 3D rekonstrukci. V kapitole návrh řešení je jednoduchým způsobem představen program, který by měl vzniknout a postup jednotlivých kroků, které jsou pro rekonstrukci nutné. V následující kapitole je popsána samotná implementace tohoto programu a také funkce pro řešení jednotlivých kroků z předešlé kapitoly. V sedmé kapitole uvádím výsledky práce a v poslední kapitole zhodnocuji dosažené výsledky a možnosti dalšího postupu práce.

Kapitola 2

Grafické a matematické základy

2.1 Popis bodu

[1] Základem veškeré geometrie v počítačové grafice je bod, ať už v ploše (čili 2D) nebo v prostoru (3D). S body jako takovými pracujeme málokdy. Ovšem všechna tělesa rozkládáme na jednotlivé jednoduché prvky a jejich popis téměř vždy závisí na bodech. Jakmile se s nimi naučíme pracovat, naučíme se pracovat s libovolně složitými tělesy.

Bod popisujeme zejména v kartézských souřadnicích. Jiné souřadné systémy jsou také možné, ale v grafice se používají zřídka. Bod zapisujeme uspořádanou dvojicí či trojicí a čísla můžeme zapsat do matice řádkové, či sloupcové. Tedy např. jako ve vztahu 2.1.

$$P = [x, y, z]^T = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.1)$$

V české literatuře se častěji používá sloupcová notace, v anglosaské řádková. Ani jeden systém nemá zvláštní výhodu. Jen se musí dávat pozor na to, který se používá.

2.2 Homogenní souřadnice

$$P = [x, y, z, w]^T = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \quad (2.2)$$

Pro snazší provádění všech lineárních transformací bodů a vektorů v prostoru je vhodné všechny body (včetně koncových bodů úseček a vektorů) specifikovat ve čtyřrozměrném prostoru, přičemž poslední souřadnice (označovaná písmenem w) je nastavena na hodnotu 1.0 pro body a 0.0 pro vektory. Tato souřadnice se podle své úlohy při výpočtech nazývá váha (weight - z toho vyplývá i její jednopísmenné označení). Po tomto rozšíření původních 3D souřadnic do čtyřrozměrného prostoru je možné pro specifikaci lineárních transformací a současně i perspektivní projekce používat transformační matice o velikosti 4x4 prvků [17].

Homogenní souřadnice bodu jsou zároveň jednotnou reprezentací afinních transformací. Umožňují nám vyjádřit všechny druhy základních transformací (včetně posunutí) jednou transformační maticí a aplikovat je násobením matic a vektorů.

Definice 2.2.1 *Homogenní souřadnice* bodu ve 3D s kartézskými souřadnicemi $[x, y, z]$ je uspořádaná čtveřice $[X, Y, Z, w]$ pro kterou platí $x = X/w$, $y = Y/w$, $z = Z/w$. Bod je svými homogenními souřadnicemi určen jednoznačně. Souřadnici w nazýváme váhou bodu. Hodnota váhy je většinou $w = 1$, v případě lineárních transformací [11].

2.3 Transformace ve 3D

Geometrické transformace ve 3D jsou rozšířením 2D transformací. Transformované body jsou popsány homogenními 3D souřadnicemi $P(x, y, z, w)$. Transformační matice T pro posun ve 3D je pouze rozšířením dimenze 2D matice (viz. vztah 2.3).

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ d_x & d_y & d_z & 1 \end{bmatrix} \quad (2.3)$$

Poněkud jiná situace nastává u rotační transformace ve 3D. Stejně jako ve 2D probíhá rotace okolo počátku souřadného systému. Máme však různé transformační matice R_x , R_y , R_z pro rotaci okolo souřadných os X , Y a Z (viz. rotace kolem Y 2.8).

$$R_y = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & \cos \alpha & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

Musíme si zde dávat pozor jak transformační matice používáme. Uvedené matice platí při používání transformace ve smyslu $[x', y', z', 1] = [x, y, z, 1] \cdot M$, kde M je transformační matice. Pokud však používáme transformace podle vztahu 2.5, mění se translační i rotační matice na vztahy 2.6 až 2.9[1].

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = M \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2.5)$$

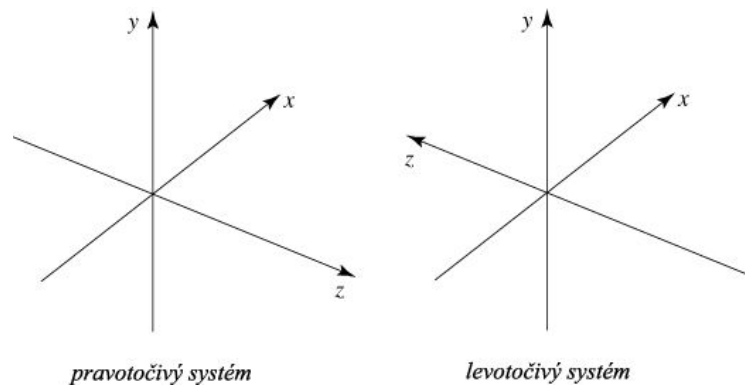
$$T = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

$$R_y = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

$$R_z = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

Matice pro otočení jsou uvedené pouze v pravotočivé formě. Levotočivá varianta znamená změnu směru otáčení - tj. místo úhlu α dosadíme úhel $-\alpha$. Takže se znaménka u funkce \cos nezmění, neboť $\cos \alpha = \cos \alpha$. Znaménka u \sin se prohodí, protože $\sin -\alpha = -\sin \alpha$.



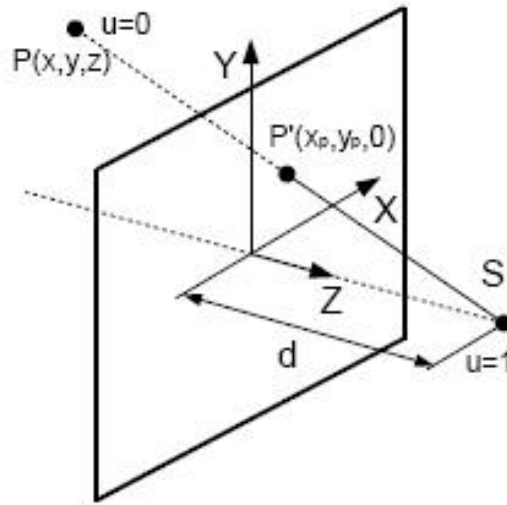
Obrázek 2.1: Pravotočivý a levotočivý systém [1]

2.4 Perspektivní projekce

Definice 2.4.1 *Perspektivní projekce* je nelineární neafinní projekce, která zobrazuje vrcholy promítaných objektů prostřednictvím paprsků protínajících se v jednom bodě, ve středu projekce. Střed projekce je zároveň většinou místem pozice pozorovatele. Velikost průmětů objektů je nepřímouměrně závislá na jejich vzdálenosti od průmětny. Čím je objekt blíže průmětny, tím je jeho obraz větší a naopak. Rovnoběžnost promítaných hran není v této projekci zachována[11].

Perspektivní projekce odpovídá našemu lidskému vidění reálného světa. Proto se nejčastěji

používá pro realističtější zobrazení ve virtuální realitě, architektuře, počítačových 3D hrách, atd.



Obrázek 2.2: Schéma odvození perspektivní projekce [11]

Perspektivní projekci můžeme vyjádřit prostřednictvím transformace s maticí P_{per} (viz. vztah 2.13 a 2.14). Její odvození, vycházející z parametrického vyjádření promítacího paprsku, najdeme ve vztazích 2.10 - 2.12 (viz. obr. 2.2).

$$x = x - xu, \quad y = y - yu, \quad z = z - (z + d)u \quad (2.10)$$

$$u \in \langle 0, 1 \rangle, \quad u = 0 \rightarrow (x, y, z), \quad u = 1 \rightarrow (0, 0, d) \quad (2.11)$$

$$z_P = 0 \rightarrow u = \frac{z}{z + d}, \quad x_P = x \frac{1}{1 + \frac{d}{z}}, \quad y_P = y \frac{1}{1 + \frac{d}{z}} \quad (2.12)$$

$$(x_P, y_P, z_P, w_P) = (x, y, z, w) \cdot P_{per}, \quad w_P = 1 + \frac{z}{d} \quad (2.13)$$

$$P_{per} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \frac{1}{d_z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.14)$$

2.5 SVD

Singulární dekompozice matice (angl. Singular Value Decomposition), jedná se o rozklad obdélníkové reálné nebo komplexní matice, který nalézá uplatnění například ve zpracování signálu a statistice. Použijí ji k řešení lineárních rovnic. Další použití podle [19] je např.:

1. výpočet pseudoinverzní matice
2. řešení homogenních lineárních rovnic
3. minimalizace pomocí metody nejmenších čtverců
4. aproximace matic

2.6 Definice a geometrický význam SVD

Definice 2.6.1 A je matice $A \in R^{m \times n}$. Pak existuje *ortogonální matice*¹ $U \in R^{m \times m}$, ortogonální matice $V \in R^{n \times n}$ a diagonální matice $D \in R^{m \times n}$ tak, že platí

$$A = UDV^T$$

kde

$$D = \begin{pmatrix} \delta_{11} & \dots & \dots & 0 \\ \vdots & \delta_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ \vdots & & & \delta_{nn} \\ 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{pmatrix}$$

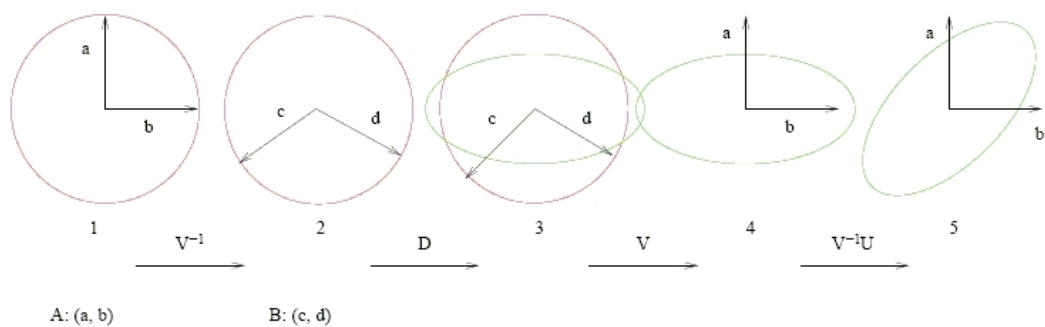
přičemž $\delta_{11} \geq \delta_{22} \geq \dots \geq \delta_{nn} \geq 0$.

Čísla na diagonále matice D jsou tzv. *singulární čísla* matice A a vektory obsažené v maticích U a V^T nazýváme levé, resp. pravé singulární vektory matice A . Rozklad SVD matice A nám také vypovídá informaci o hodnotě matice A , tedy přesněji počet nenulových hodnot na diagonále matice D je hodnota matice A .

Geometrický význam SVD nám pomáhá určit, jak matice A změní vektory po její aplikaci. To si je možné představit tak, že matici A budeme považovat za zobrazení určitého vektoru x do jiného vektoru y podle předpisu $y = Ax$. Potom aplikace matice V má za následek pouze změnu báze vstupního vektoru x . Matice D prodlouží nebo zkrátí daný vektor. Nakonec matice U znamená zpětný přechod od nové báze k té původní. Názorný příklad, jaký je geometrický význam při zobrazení jednotkové kružnice maticí $A_{2 \times 2}$, je možné pozorovat na obrázku 2.3. Jednotlivé kroky znázorněné na obrázku odpovídají následujícím operacím:

1. **1** → **2** - Změna báze z A do B maticí přechodu V^{-1} .
2. **2** → **3** - Změna měřítka podél souřadných os maticí D .
3. **3** → **4** - Změna báze z B do A pomocí matice V .
4. **4** → **5** - Rotace provedená násobením součinem matice $V^{-1}U$.

¹Ortogonální matice je taková čtvercová matice U , pro kterou platí $U^T U = U U^T = I$ a $U^T = U^{-1}$.



Obrázek 2.3: Geometrický význam SVD - A,B jsou jednotlivé báze [12].

Vysvětlení je platné, pokud uznáme následující fakta:

$$\begin{aligned}
 A &= UDV^T = (UV^{-1})VDV^{-1} \\
 V^T &= V^{-1}
 \end{aligned}
 \tag{2.15}$$

rovnice 2.15 vyplývá z důvodu ortogonality matice V [12].

Kapitola 3

Zpracování obrazu

3.1 Významné body

Významný bod (*angl. feature point nebo interest point*) - jedná se o bod, resp. body, které nás v obrázku zajímají a jsou pro další zpracování obrázku důležité. Např. se jedná o rohy, nebo body, které se dají znovu dobře vyhledat. To je jejich základní vlastnost, možnost znovu vyhledat ten samý bod i v jiném obrázku, kde je scéna jinak pootočená.

Významný bod má následující charakteristiky[18][4]:

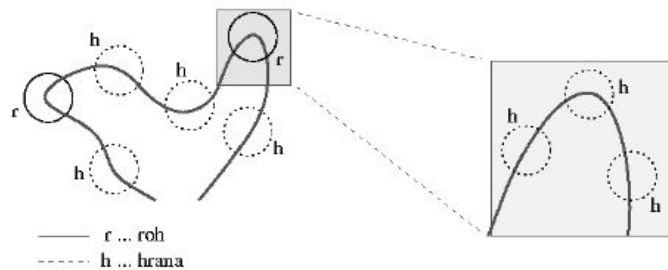
- přesnou, matematicky dobře podloženou definici
- přesně definovanou pozici v obrazovém prostoru
- lokální strukturu v obraze kolem významného bodu bohatou na informace vhodné pro pozdější zpracování vizuálním systémem
- je stabilní z hlediska působení lokálních a globálních deformací v obrazové doméně tak, aby byl bod opět nalezen s vysokým stupněm opakovatelnosti
- je vhodné, aby obsahoval atribut měřítka pro možnost vypočítat ho z reálné změny obrázku stejně jako s měnícím se měřítkem

Detektorů významných bodů je poměrně hodně. Mezi nejznámější patří Harrisův detektor a SIFT ¹.

3.2 Harrisův detektor

Jedná se o detektor významných bodů, který vyhledává rohy a hrany. Byl vytvořen v roce 1988 Chrisem Harrisem a Mikem Stephensem. Vychází z Moravcova operátoru, ale liší se od něj v používání takzvané lokální autokorelační funkce[3]. Použitím této funkce dosahuje Harrisův detektor výborných výsledků. Je nezávislý na rotaci a posunu a odolný vůči šumu. Nedokáže si však poradit se změnou měřítka. Jak je vidět na obrázku 3.1.

¹ SIFT - Scale-invariant feature transform, autorem je David Lowe



Obrázek 3.1: Znázornění problému vyhledávání významných bodů pomocí měřítkově závislého Harrisova detektoru. Jsou zkoumány body znázorněné křivky. Kolečka představují tzv. kruhová okénka, pomocí kterých se vyhodnocuje zakřivení v okolí bodu. Pomocí nepřerušovaných kroužků se znázorňují body vyhodnocené jako významné. Křivka z těchto okének vystupuje pod výrazně jiným směrem, než vstupuje. Všechny ostatní body křivky jsou vyhodnoceny jako hrany. V pravé části je část stejné křivky, ale je znázorněna ve větším měřítku. Je na ní použit opět Harrisův detektor se stejným okénkem jako v prvním případě. Díky většímu měřítku už se ale směr křivky uvnitř okénka nikde nezmění natolik, aby byl bod vyhodnocen jako roh. Body vyhledané pomocí měřítkově závislých metod tedy nelze použít pro porovnávání struktur s různými měřítky, neboť se po změně měřítku v obrázku vyhledají jiné struktury[4].

3.3 SIFT

SIFT - Scale-invariant feature transform, jehož autorem je David G. Lowe, je jedním z nej-používanějších detektorů významných bodů v současnosti. A zmínka je o něm i ve většině prací zabývajících se počítačovým viděním. Je popsán v článku [14]. Zde uvedu jen základní vlastnosti a popis. Každý významný bod, který se pomocí SIFTu najde, dostane přiřazen tzv. deskriptor. Jedná se o vektor 128 hodnot celých čísel. Deskriptory jsou invariantní vůči následujícím operacím s obrazem[4]:

- zvětšení, zmenšení - tedy změna měřítku
- posun, 2D rotace
- změna kontrastu, jasu, osvětlení, šum
- částečně invariantní vůči prostorové změně úhlu pohledu

Proces získávání významných bodů z analyzovaného obrázku lze rozdělit do čtyř fází[4]:

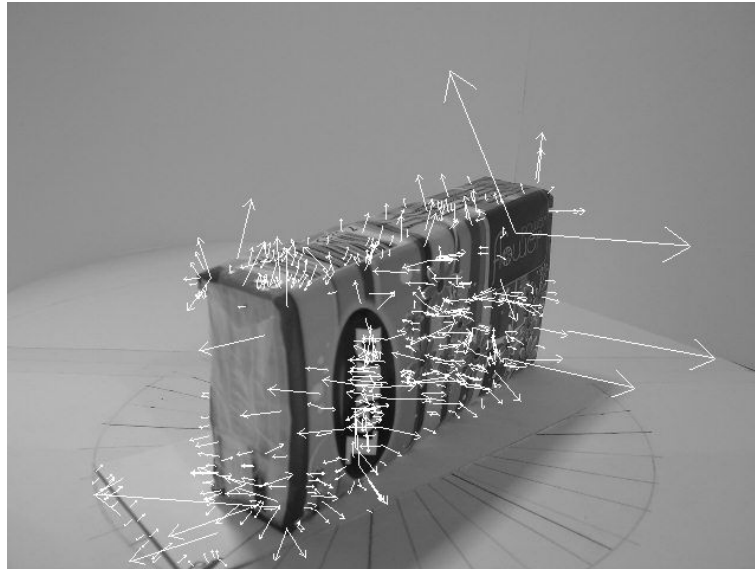
1. Sestrojení scale-space a vyhledání lokálních extrémů:
V první fázi výpočtu se musí zkontrolovat měřítkově nezávislá reprezentace obrázku, tzv. scale-space. V něm se dále vyhledávají přes všechna měřítká lokální extrémy. Pro zvýšení výkonnosti je pro generování scale-space použita místo Laplacianu Gaussovy funkce (LoG) jeho aproximace pomocí rozdílu Gaussových funkcí (DoG).
2. Přesnější lokalizace klíčových bodů:
Body získané v první fázi se podrobí podrobnějšímu zkoumání. Ty s nedostatečnou stabilitou se vyloučí a u zbylých se pomocí interpolace upřesní jejich poloha v prostoru.

3. Přiřazení orientací:

Každému významnému bodu je přiřazena dominantní orientace na základě orientací gradientů² v okolí bodu. To je potřeba pro zjištění nezávislosti na rotaci.

4. Tvorba deskriptorů:

Na základě orientací gradientů v okolí bodu je pro každý významný bod vypočten deskriptor. Pomocí něj jsou potom porovnávány významné body mezi analyzovanými snímky.



Obrázek 3.2: Ukázka vyhledání významných bodů pomocí SIFTu

3.4 Korespondence významných bodů

Jak bylo řečeno, odpovídající si body na dalších obrázcích mají stejné (nebo velmi podobné) deskriptory. Pokud tedy hledáme podobné body, porovnáváme deskriptory jednotlivých bodů pomocí Euklidovské vzdálenosti. Dva body s nejmenší vzdáleností by si pak měly odpovídat a mělo by se jednat o korespondence. V praxi se však používá mírně odlišný výpočet a to takový, že se najdou pro jeden bod dva odpovídající body s nejmenší Euklidovskou vzdáleností. Pokud je tato vzdálenost v poměru menším jak 0.6 jedná se u nejmenší vzdálenosti s velkou pravděpodobností o korespondenci. Hodnota 0.6 byla zjištěna testováním jako hodnota s nejlepšími výsledky.

²Gradient je diferenciální operátor, jehož výsledkem je vektorové pole vyjadřující směr a velikost změny skalárního pole.

Kapitola 4

Teorie pro 3D rekonstrukci

4.1 Vnitřní parametry kamery

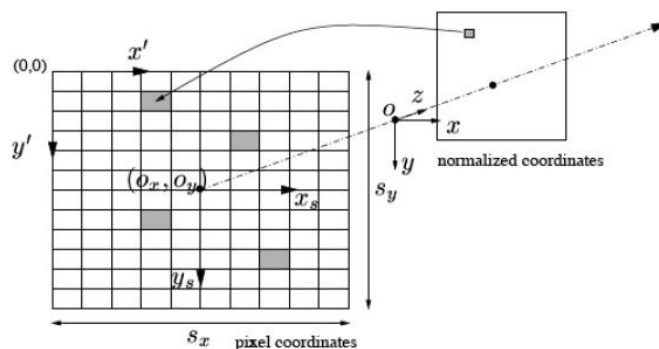
Parametry, které jsou nutné pro převod ze souřadnic obrázku do souřadnic pixelů obr. 4.1. Parametry charakterizují optický, geometrický a číselný model kamery. Matice vnitřních parametrů má rozměr 3x3:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Matice K se nazývá *matice vnitřních parametrů* (angl. *intrinsic camera matrix*), kde c_x a c_y jsou souřadnice základního bodu (většinou se jedná o střed snímku), f_x a f_y jsou ohniskové vzdálenosti uváděné jednotkách vycházejících z pixelů. Matice vnitřních parametrů kamery je nezávislá na scéně a jako taková může být používána od jejího výpočtu až do změny ohniskové vzdálenosti. [10].

Dále lze f_x a f_y vypočítat také pomocí vzorců:

$$f_x = \text{ohnisková vzdálenost[mm]} / (\text{šířka pixelu[um]} / 10^3)$$
$$f_y = \text{ohnisková vzdálenost[mm]} / (\text{výška pixelu[um]} / 10^3)$$



Obrázek 4.1: Transformace z obrazových souřadnic do pixelů[20]

Pokud je matice vnitřních parametrů známa, je možné provést přesnou rekonstrukci. Tedy získat přesnou rekonstrukci 3D objektu a ne pouze jeho podobnost. Tím je myšleno, že je možné zrekonstruovat jeho přesné rozměry.

4.2 Vnější parametry kamery

Vnější parametry kamery popisují vztah mezi světovým souřadným systémem a souřadným systémem kamery. Jedná se o rotační matici R a vektor posunutí t .

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$t = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

4.3 Projekční matice model projektivní kamery

Vztah mezi bodem M ve 3D a jeho projekcí do bodu m je popsán rovnicí.

$$m = K[Rt]M, \quad (4.1)$$

kde K je matice vnitřních parametrů kamery, (R, t) jsou vnější parametry kamery: rotační matice R a vektor posunu t , které převádějí světový souřadný systém do souřadného systému kamery[10].

4.4 Fundamentální matice a epipolární geometrie

Epipolární geometrie popisuje základní geometrický vztah mezi dvěma perspektivními kamerami nezávisle na pozorované scéně. V následující části popíšeme jen základní vztahy mezi dvěma kamerami.

Na obr. 4.2 vidíme dvě kamery určené středy promítání C_i, C_p a příslušnými projekčními rovinami, bod v prostoru P a jeho projekce na levé a pravé průmětně p_p, p_l [5].

Epipolární geometrie je algebraicky reprezentovaná esenciální maticí E a z ní vycházející fundamentální maticí F . Esenciální matice je definována vztahem:

$$P_p^T E P_l = 0 \quad (4.2)$$

kde P_p, P_l jsou zápisy bodu P v souřadných systémech levé a pravé kamery. Pokud známe transformaci mezi souřadnými systémy kamer:

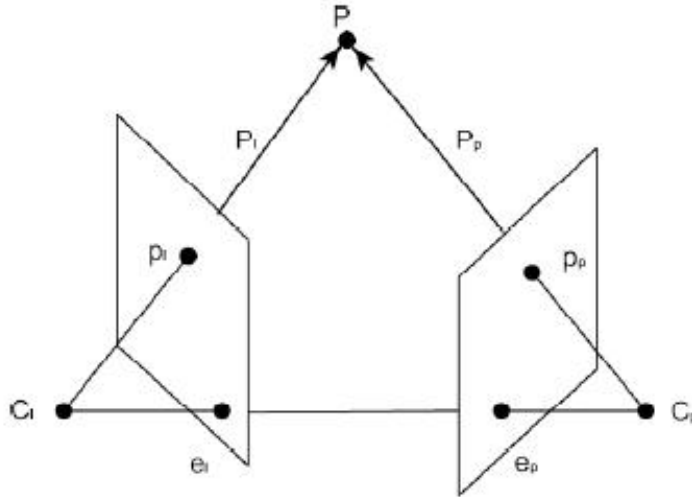
$$P_p = R(P_l - T) \quad (4.3)$$

můžeme matici E vyjádřit vztahem:

$$E = SR \quad (4.4)$$

kde R , je matice rotace viz. 4.3 a

$$S = \begin{pmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{pmatrix} \quad (4.5)$$



Obrázek 4.2: Epipolární geometrie dvou pohledů [5].

kde T_x, T_y, T_z jsou prvky vektoru T viz. 4.3. Fundamentální matice je definována vztahem:

$$p_p^T F p_l = 0 \quad (4.6)$$

kde p_p, p_l jsou projekce bodu P na průmětny pravé a levé kamery v pixelových souřadnicích[5].

4.5 Lineární triangulační metoda

V této kapitole popíši jednoduchou lineární triangulační metodu, kterou používám i v programu.

Každému obrázku odpovídají rovnice $x = PX, x = P'X$, kde P a P' jsou projekční matice dvou kamer. Tyto rovnice mohou být zkombinovány do nové rovnice $AX = 0$, jenž je lineární rovnicí pro X .

Jako první je odstraněn homogenní měřítkový faktor pomocí křížového součinu daných tří rovnic pro každý bod obrázku, z čehož dvě jsou lineárně nezávislé. Například pro první obrázek, $x \times (PX) = 0$ rozepsáním vzniká:

$$x(p^{3T}X) - (p^{1T}X) = 0 \quad (4.7)$$

$$y(p^{3T}X) - (p^{2T}X) = 0 \quad (4.8)$$

$$x(p^{2T}X) - y(p^{1T}X) = 0 \quad (4.9)$$

kde p^{iT} jsou řádky matice P . Tyto rovnice jsou lineárními komponenty bodu X . Rovnice ve formě $AX = 0$ lze pak sestavit jako:

$$A = \begin{bmatrix} xp^{3T} - p^{1T} \\ yp^{3T} - p^{2T} \\ x'p'^{3T} - p'^{1T} \\ y'p'^{3T} - p'^{2T} \end{bmatrix} \quad (4.10)$$

kde z každého obrázku náleží dvě rovnice, celkově tedy čtyři rovnice pro čtyři homogenní neznámé. Rovnici $AX = 0$ lze vyřešit například SVD funkcí [8]. Při použití SVD je pak výsledkem poslední sloupec matice V^T .

Kapitola 5

Návrh řešení

Výsledkem by měl být program, kterému se zadá 36 snímků objektu, kde na každém snímku je objekt pootočen o deset stupňů, dále tři kalibrační vzory, každý také otočený o deset stupňů proti předchozímu. Výstupem programu je datový soubor obsahující souřadnice bodů zrekonstruovaného objektu.

Jak již bylo řečeno v této verzi by měl být výstupem programu soubor, jež bude obsahovat všechny zrekonstruované body v prostoru a na jejich zobrazení se použije externí program. Body se budou ukládat do souboru, kdy každému bodu odpovídá jeden řádek se souřadnicemi X , Y , Z v tomto pořadí.

Postupy rekonstrukcí jsou různé podle toho, co je známé a čeho chceme dosáhnout. Zjednodušeně by se ale daly přepsat na kalibraci kamer, dále vyhledání významných bodů, pak spárování těchto bodů z různých obrázků (vyhledání korespondencí) a nakonec výpočet bodů v prostoru. Návrh řešení, kterého se budu držet při implementaci je následující:

1. Vytvoření/získání snímků pro rekonstrukci.
2. Kalibrace kamery - výpočet vnitřních a vnějších parametrů.
3. Vyhledání významných bodů ve dvou po sobě následujících obrázcích.
4. Nalezení korespondencí ve vyhledaných bodech.
5. Výpočet fundamentální matice.
6. Výpočet epipolární přímky pro bod v obrázku a nalezení odpovídajícího bodu na přímce v druhém obrázku.
7. Triangulace korespondencí.
8. Zpětná projekce a eliminování špatně zrekonstruovaných bodů.
9. Pootočení nově získaných bodů o deset stupňů proti předchozím.
10. Zpět na krok 3. a použití dalšího obrázku v řadě.

5.1 Získání testovacích dat

Testovací data získám z digitálního fotoaparátu. Přičemž mezi každým snímkem pootočím objekt o 10 stupňů. Tím vznikne soubor 36 fotografií postupně se otáčejícího objektu.

Fotoaparát se nebude hýbat a jeho ohnisková vzdálenost (*angl. focal length*) se nebude měnit, což ulehčí další výpočty, neboť kalibrace vnitřních parametrů kamery bude pro všechny obrázky stejná.

5.2 Kalibrace kamery

Kalibrace kamery se nejčastěji provádí pomocí tzv. *kalibračního vzoru* (*angl. calibration rig*). Jedná se například o šachovnici u které známe její přesné rozměry. Kalibrační vzor vyfotíme z různých úhlů. Pomocí získaných snímků můžeme zjistit vnitřní a vnější parametry kamery. Vnitřní parametry se během celé rekonstrukce nemění, jelikož rozlišení a zaostření zůstávají stejná pro všechny snímky. Vnitřní parametry kamery stačí tedy získat jen jednou a dále používat ty samé. Teorie o vnitřních parametrech viz. kapitola 4.1.

Ze snímků kalibračního vzoru lze získat i vnější parametry kamery, ty se však za normálních okolností nedají dále použít, protože s posunutím a rotací objektu se mění. Šachovnici mám však před kamerou umístěnu tak, že rozdíl mezi prvním a druhým kalibračním vzorem je stejný jako následné pootočení objektu. Mohu tedy použít i vnější parametry kamery a sestavit tak projekční matice.

Tímto se rekonstrukce velice zlehčuje, protože není nutné získávat parametry jinými výpočty.

5.3 Vyhledání významných bodů

Pro vyhledávání významných bodů je možno použít několik detektorů. Rozhodl sem se pro vyzkoušení detektorů Harrisova a SIFT. Problémem u Harrisova detektoru je, že nemá žádný implicitně vytvářený deskriptor pro své nalezené body. Proto mi přišlo vhodnější použít pro vyhledávání bodů program SiftDemoV4, který hledá, jak to již název napovídá, body pomocí SIFT algoritmu.

Program SiftDemoV4 vytvořil přímo David Lowe pro studijní účely a demonstraci. Nelze ho použít pro komerční implementaci, algoritmus SIFT je chráněn patentem. Velkou výhodou je, že zároveň pro vyhledané body vytvoří i deskriptor. Deskriptor je složen ze 128 celočíselných hodnot pro každý bod. Odpadá tedy vytváření vlastního deskriptoru, jak by to bylo třeba u Harrise.

5.4 Nalezení korespondencí ve vyhledaných bodech

Pokud máme nalezené všechny významné body v prvním a druhém obrázku, je potřeba zjistit, které body si odpovídají. Tedy najít korespondující si body. Korespondence je taková oblast na obrázcích, která představuje stejný objekt ve 3D světě.

Princip nalezení korespondencí spočívá ve vypočítání Euklidovské vzdálenosti a to takto:

1. Ke každému významnému bodu z prvního obrázku zjistíme dva nejbližší body pomocí Euklidovské vzdálenosti, rovnice pro SIFT deskriptor 5.1.
2. Vypočítáme poměr dvou nejbližších bodů. Pokud je poměr menší jak 0.6, je nejbližší bod s největší pravděpodobností korespondencí k bodu z prvního obrázku. Pokud ne, nejspíše bod z prvního obrázku korespondenci v druhém nemá.

Poměr 0.6 byl zjištěn testováním a dosahoval nejlepších výsledků.

$$d = \sqrt{\sum_{i=1}^{128} (k1_i - k2_i)^2} \quad , \text{ kde } k1_i, k2_i \text{ jsou } i\text{-té hodnoty deskriptorů} \quad (5.1)$$

5.5 Výpočet fundamentální matice

Pokud máme ve dvou obrázcích vyhledané korespondující si body, můžeme spočítat fundamentální matici. Pro výpočet fundamentální matice existuje několik metod. Jejich hlavní rozdíl spočívá v tom, kolik používají korespondujících bodů pro výpočet. Existuje osmi, sedmi a dokonce i pětibodový algoritmus (více v pracích [8] [15] [16]). Z těchto je nejvíce využíván osmibodový algoritmus. Jelikož korespondujících bodů se většinou najde více, vzniká tak velké množství fundamentálních matic. Aby se vybrala ta správná, využívá se algoritmu RANSAC. Jedná se o iterativní algoritmus, který vybere fundamentální matici nejlépe odpovídající všem bodům (více např. [7]).

5.6 Výpočet epipolárních přímek

Po spočítání fundamentální matice z předchozího kroku, můžeme vypočítat epipolární přímky. Pro jakýkoliv bod z prvního obrázku získáme přímku v druhém snímku, na které se má nacházet korespondence k vybranému bodu. Takto lze rychlejším způsobem hledat nové korespondence ve snímcích. Nemusí se totiž prohledávat pro každý bod celý obrázek, ale jen prostor kolem vypočítané epipolární přímky a na ní.

V programu například používám epipolární přímky v programu pouze ke zjištění, jestli se opravdu jedná o správné korespondence a pro ověření funkčnosti pro další použití v programu jako možnost získávat více korespondencí, a tak přesnější strukturu rekonstruovaného objektu.

5.7 Triangulace korespondencí

Pokud již máme zjištěnou rotační a translační matici kamer, vnitřní kalibraci, známe odpovídající si body v obou obrázcích, můžeme sestavit projekční matice a tyto všechny prvky pak použít v triangulačním algoritmu popsaném v kapitole 4.5. Pro výpočet je možné použít SVD (viz. 2.5), kde výsledné souřadnice bodu jsou řádky z posledního sloupce matice V^T . Nyní jsme získali body v prostoru ze dvou snímků (stereo páru).

5.8 Zpětná projekce

V předchozím kroku jsme získali body objektu v prostoru. Pokud chceme eliminovat všechny chybně rekonstruované body, provedeme u všech bodů projekci zpět do souřadnic snímku a porovnáme s hodnotami v původním obrázku. Pokud se liší souřadnice bodů o více jak námi zvolená mez, body vyloučíme. Jedná se nejspíše o špatné korespondence, nebo jiné chyby jako špatný výsledek triangulace, který použitím SVD může nastat.

5.9 Pootočení získaných bodů o deset stupňů

Jelikož z předchozích kroků máme vypočítané body rekonstruovaného objektu v prostoru a na výpočet používáme stále stejné vnější parametry kamery, body nejsou v prostoru spočítány správně a je nutné je pootočit o určitý počet stupňů.

Při získávání testovacích dat jsem objekt otáčel s každým snímkem přesně o deset stupňů. Teoreticky by tedy mělo stačit pootočit získané body o $S \cdot 10^\circ$, kde S je již zpracovaný počet snímků. Pootočení se provádí kolem osy Y , pokud je střed otáčení objektu shodný s osou Y počátku souřadného systému. Touto geometrickou transformací se získají nové body v prostoru, které již mají odpovídat rekonstruovanému objektu.

Kapitola 6

Implementace

V této kapitole popisují implementaci jednotlivých bodů, tak jak byly navrženy v předešlé části.

Pro implementaci používám knihovnu OpenCV[9] a program píše v jazyce C/C++. Dále na vyhledávání významných bodů využívám program Davida Loweho SiftDemoV4[13] a pro zobrazení bodů ve 3D program gnuplot[2].

6.1 Získání testovacích dat

Testovací data jsem získal pomocí digitálního fotoaparátu Konica Minolta DiMAGE Z10, při rozlišení snímků 640x480px a stálém zaostření 20cm a vypnutém blesku. Fotoaparát byl umístěn napevno a objekt před ním se otáčel na podložce při každém snímku o deset stupňů. Takto byly získány tři snímky šachovnice a 36 snímků objektu pro další zpracování.

6.2 Kalibrace kamery

Pro získání matice vnitřních a vnějších parametrů je v OpenCV implementována funkce `cvCalibrateCamera2`, která pro provedení úspěšné kalibrace potřebuje souřadnice bodů ve 3D v jednom poli a ve druhém poli souřadnice odpovídajících bodů, které jsou projekcí ve snímku kamery.

Jelikož používám šachovnici pro kalibraci, mohl jsem využít funkce:

`cvFindChessboardCorners,`

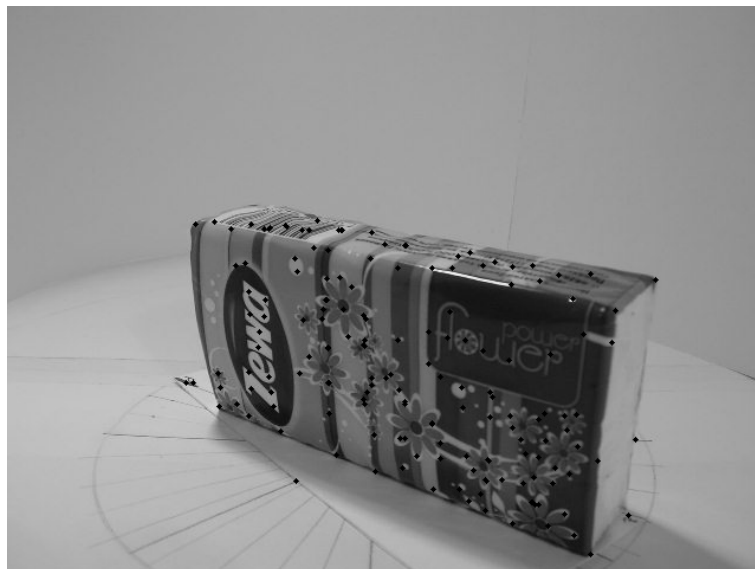
která vyhledá v odkazovaném obrázku všechny vnitřní rohy šachovnice a vrátí jejich souřadnice v poli. V mém případě se jedná o 49 vnitřních bodů. Tyto body pak přidám k předešlým nalezeným bodům tak, aby se pak daly poslat do funkce pro kalibraci kamery.

Při používání šachovnice jako kalibračního vzoru se také snadno počítají reálné souřadnice 3D bodů. Například první horní roh zvolíme jako bod o souřadnicích $[0, 0, 0]$, a pak přidáváme rozměry čtverců šachovnice tak, jak se body postupně vyhledávají na řádcích a sloupcích. Druhý roh šachovnice má například souřadnice $[18, 0, 0]$ a první bod v druhém řádku $[0, 18, 0]$ při rozměru čtverce $18mm$. Souřadnice šachovnice v prostoru se s dalšími snímky nemění, protože z teoretického hlediska se kamera otáčí kolem objektu. Jeho souřadnice jsou tedy stále stejné.

Jakmile jsou všechny kalibrační obrazy zpracovány, lze již volat přímo kalibraci kamery. Tato funkce v parametrech vrátí vnitřní matici kamery a rotační a translační vektor pak pro každý kalibrační snímek.

6.3 Vyhledání významných bodů

Jako první jsem použil Harrisův detektor významných bodů, který je přímo implementován v OpenCV. Použití je pomocí funkce `cvGoodFeaturesToTrack`, parametry jsou vstupní obrázek a nastavení hodnot pro vyhledávání. Výstupem jsou nalezené body. Tato funkce použije Harrisův detektor pouze pokud se jí to v parametru zadá. Výsledek je možno vidět na obrázku [6.1](#)



Obrázek 6.1: Významné body při použití vyhledávání pomocí Harrisova detektoru implementovaného v knihovně OpenCV funkcí `cvGoodFeaturesToTrack`.

Z důvodů, které jsem uvedl již dříve, jsem se rozhodl použít nakonec program `SiftDemoV4`. Jedná se o konzolový program, který se spouští `siftWin32 <pict.pgm> pict.key` v prostředí Windows více na [\[13\]](#). Jak je vidět z příkladu, program pracuje s obrázky ve formátu `pgm`. Jeho výstupem je pak textový soubor, jenž obsahuje všechny nalezené významné body se svými deskriptory. Zobrazené vyhledané body jsou vidět na obrázku [3.2](#).

6.4 Nalezení korespondencí ve vyhledaných bodech

Vyhledané významné body je nutno přečíst z textového souboru. K tomu v programu slouží funkce `readFile`. Po načtení se použijí další funkce pro výpočet Euklidovské vzdálenosti a její porovnávání.

Vyhledané korespondence ukládám do dvou polí. První pole jsou body z prvního obrázku a ve druhém poli jsou korespondující body v druhém obrázku.

6.5 Výpočet fundamentální matice

Pro výpočet opět používám funkci OpenCV `cvFindFundamentalMat`, které se jako parametry zašlou dvě pole korespondujících si bodů a zvolí se metoda jakou se má fundamentální

matice vypočítat. Volím osmi bodový algoritmus a RANSAC pro vrácení nejlépe ohodnocené fundamentální matice.

6.6 Výpočet epipolárních přímek

V OpenCV je pro výpočet epipolárních přímek připravena funkce:

```
cvComputeCorrespondEpilines,
```

která z fundamentální matice a bodů ve snímku vytvoří pole vektorů přímek. Každá přímka je popsána vektorem $[a, b, c]$ odpovídající rovnici $ax + by + c = 0$.

6.7 Triangulace korespondencí

Podle vzorce 4.5 sestavím matici a pro její rozklad použiji OpenCV funkci `cvSVD`. Tato funkce rozloží zasloupanou matici. Je možno použít i jiné parametry pro rozklad.

6.8 Zpětná projekce

Jedná se pouze o použití projekčních matic, otestování vypočtených 3D bodů, jejich projekci zpět do souřadnic snímku a následné otestování zda odpovídají vyhledaným významným bodům.

6.9 Pootočení získaných bodů o deset stupňů

Pootočení probíhá pomocí transformačních matic v prostoru, kde se nejprve použije posunutí do osy Y , a pak pootočení o daný počet stupňů podle toho kolikátý snímek je právě zpracováván. Po získání nových souřadnic jsou tyto uloženy do textového souboru.

Kapitola 7

Výsledky

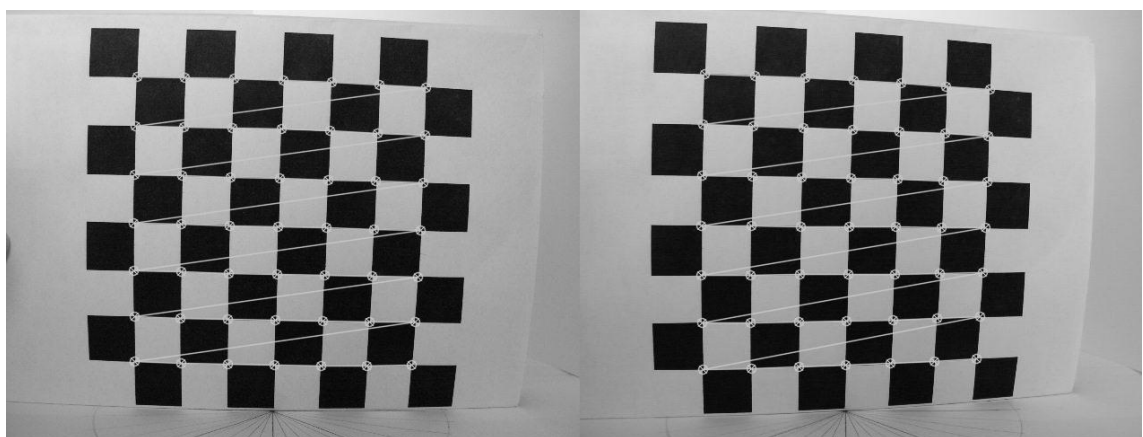
Tato kapitola se zabývá zhodnocením a ukázkou jednotlivých kroků 3D rekonstrukce, nejdříve kalibračním vzorem šachovnice, poté rekonstrukcí ze dvou pohledů, a dále uvádí problém více pohledů. Poslední část se zaměřuje na možnosti dalšího vývoje a vylepšení programu.

7.1 Rekonstrukce kalibračního vzoru.

Jako první ukáží výsledky přímo na kalibračním vzoru, tedy šachovnici. Zde není nutné provádět vyhledávání korespondencí, protože OpenCV nalezne vždy všechny vnitřní rohy a ty si na obou snímcích odpovídají.

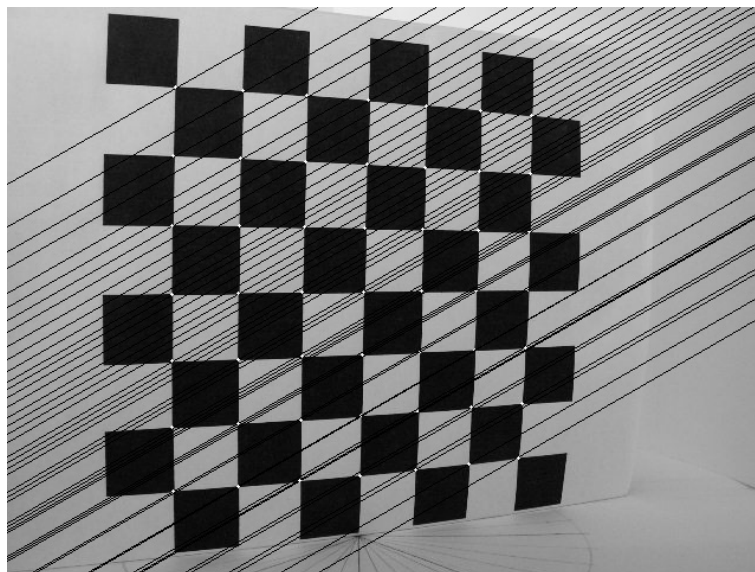
Kalibrace kamery probíhá v pořádku. Tato část se provádí vždy pro rekonstrukci jakéhokoliv objektu. Jsou vyhledány všechny vnitřní rohy šachovnice, jak je vidět na obrázku 7.1. Matice vnitřních parametrů pak vychází:

$$\begin{pmatrix} 680.01 & 0 & 318.13 \\ 0 & 677.88 & 219.57 \\ 0 & 0 & 1 \end{pmatrix}$$



Obrázek 7.1: Kalibrační matice s vyhledanými vnitřními rohy šachovnice.

Vypočtené a vykreslené epipolární přímky v druhém snímku jsou pak ukázány na obr. 7.2. Je vidět, že každým bodem prochází právě jedna epipolární přímka. Tento bod rekonstrukce je také správně.



Obrázek 7.2: Vykreslené epipolární přímky pro druhý snímek šachovnice.

Na obrázku 7.3 je rekonstrukce šachovnice ze dvou pohledů. Ta je velice přesná a odchylky bodů jsou minimální. Pokud se ale pokusím provést rekonstrukci ze tří snímků pootočené šachovnice (obr. 7.4) a provedu posun do osy Y i rotaci, lze již pozorovat relativně velkou odchylku. Body, které by měly být blízko u sebe, protože se jedná o ty samé, jsou vzdálenější, než by se očekávalo.

7.2 Rekonstrukce ze dvou pohledů

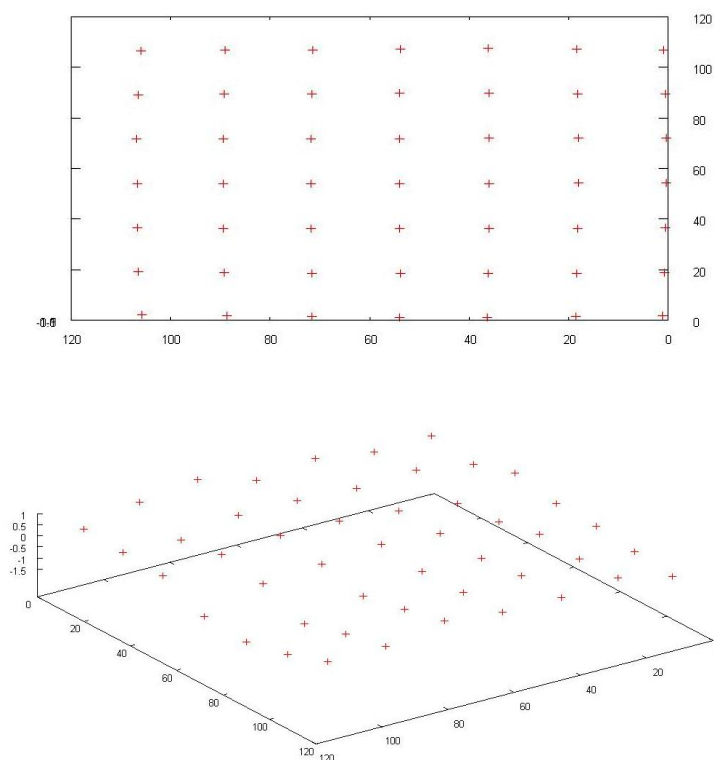
Nyní ukáži rekonstrukci ze dvou pohledů. Na nich je dobře vidět rekonstrukce boční stěny objektu. Po kalibraci kamery se vyhledají korespondující body ve snímcích (obr. 7.5).

Vypočítají se a vykreslí epipolární přímky pro druhý ze dvou snímků (obr. 7.6).

Rekonstrukce na obrázku 7.7 je bez posunutí a rotace. Je jasně vidět boční stěna s mírným prohnutím. Rekonstrukce ze dvou pohledů probíhá správně. Pokud ale použijeme posunutí a rotaci, aby body byly ve správných souřadnicích, dojde k deformaci a body v žádném případě neodpovídají předpokladu (viz. 7.8).

7.3 Zhodnocení výsledků

Jak je možno si odvodit z obrázků, rekonstrukce nepracuje správně pro všechny snímky. Jakmile se použije posunutí a rotace pro přepočítání zrekonstruovaných bodů, dojde k deformaci. Chyba mohla nastat při pořizování snímků a posunu fotoaparátu mezi získáváním snímků kalibračního vzoru a samotného objektu. Jedná se s největší pravděpodobností právě o tuto chybu.

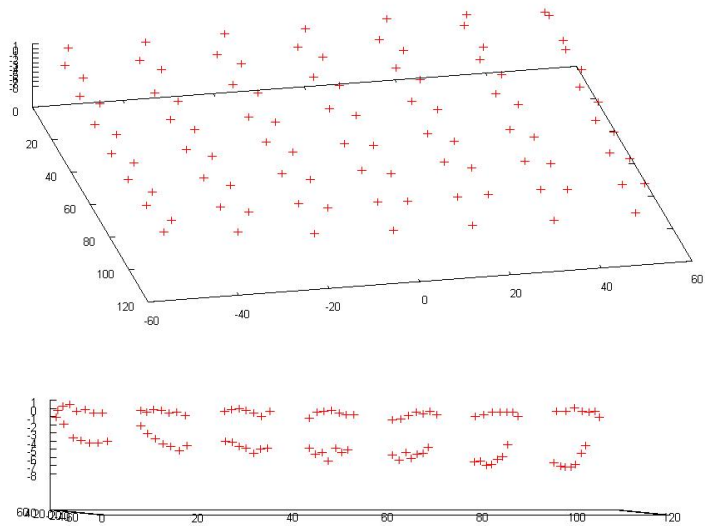


Obrázek 7.3: Zrekonstruované body šachovnice ze dvou pohledů.

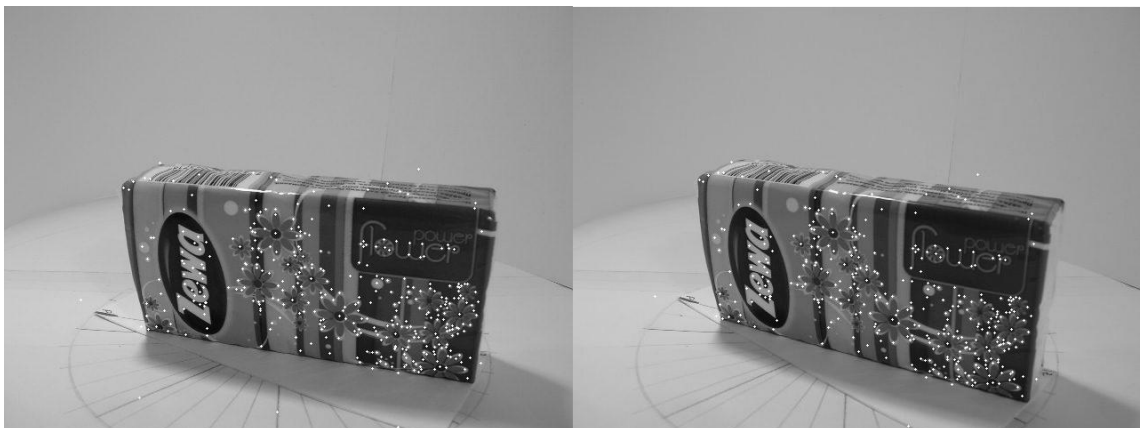
Jednotlivé části programu však podle všeho fungují dobře. Jedná se vyhledání rohů v kalibračním vzoru, kalibrace kamery, nalezení korespondujících si bodů i práci s fundamentální maticí a epipolárními přímkami. Rekonstrukce ze dvou pohledů vyhází také podle dosažených výsledků správně a tedy i triangulace korespondencí je v pořádku.

7.4 Návrhy na další práci

Prioritním problémem je zjištění, kde přesně se nachází chyba a eliminace této chyby. Jako další by se dalo uvažovat o zakomponování vyhledávání významných bodů přímo do programu. Použití externího programu SiftDemoV4 je omezující, proto by nebylo špatné použít například již nějaké hotové knihovny pro SIFT, nebo použít Harrisův detektor, který má v OpenCV podporu a vytvářet si pouze vlastní deskriptor pro každý bod.



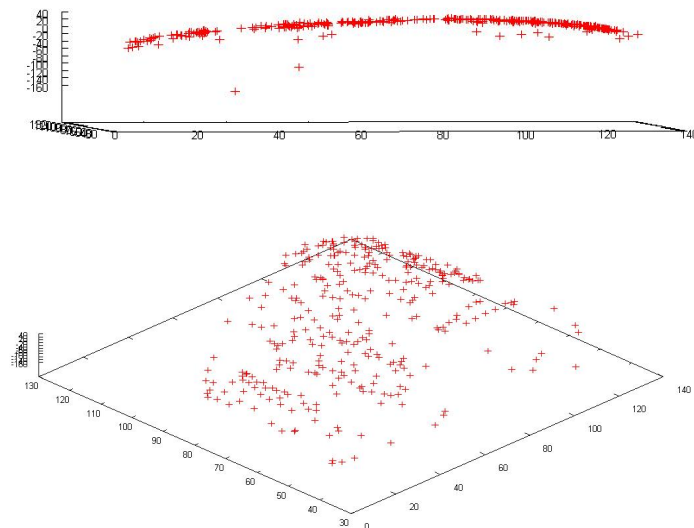
Obrázek 7.4: Zrekonstruované body šachovnice ze 3 snímků.



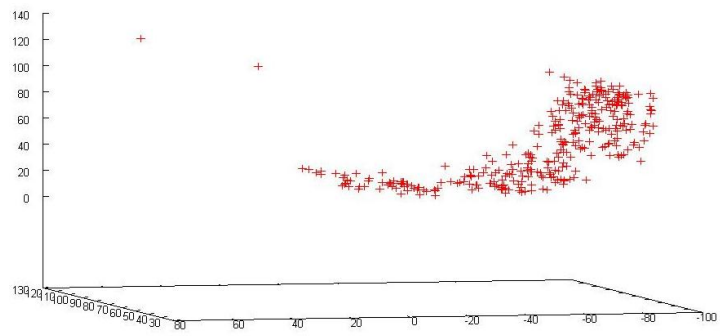
Obrázek 7.5: Vyhledané korespondující body pro dva snímky.



Obrázek 7.6: Vykreslené epipolární přímky pro druhý ze dvou snímků.



Obrázek 7.7: Rekonstrukce boční stěny objektu.



Obrázek 7.8: Rekonstrukce boční stěny objektu po nastavení posunu a rotace.

Kapitola 8

Závěr

V této bakalářské práci jsem se pokusil vytvořit program, který bude provádět 3D rekonstrukci objektu z několika jeho snímků. Jak vyplývá z kapitoly 7 je v této fázi zatím ještě nepoužitelný pro celkovou rekonstrukci. Pro stereo rekonstrukci se dosahuje výsledků mnohem lepších a relativně přesných. Správně pracuje, jak kalibrace kamery, zjištění vnitřních a vnějších parametrů, tak i pokusy s fundamentální maticí a epipolárními přímkami. Další práci bych viděl ve zlepšení výsledků. Zjištění, z jakých důvodů přesně dochází k hlavní chybě a následné deformaci zrekonstruovaných bodů.

Literatura

- [1] Analytická geometrie pro počítačovou grafiku I. [online], [cit. 2008-04-27].
URL <<http://herakles.zcu.cz/education/ZPG/cviceni.php?no=1&lang=cs>>
- [2] gnuplot homepage. [online], [cit. 2008-05-07].
URL <<http://www.gnuplot.info/>>
- [3] Harris/Plessey Operator. [online], [cit. 2008-04-26].
URL <<http://www.cim.mcgill.ca/~dparks/CornerDetector/mainHarris.htm>>
- [4] Bílek, P.: *Významné body v obraze: detekce, korespondence a lokalizace ve 3D*.
Bakalářská práce, FEL ČVUT, 2007.
- [5] Bystřický, V.: Určení pozice bodu v prostoru pomocí dvou kamer. Technická zpráva,
Západočeská univerzita v Plzni, Fakulta aplikovaných věd, 2007.
- [6] DARPA: Urban Challenge. [online], [cit. 2008-05-05].
URL <<http://www.darpa.mil/GRANDCHALLENGE/index.asp>>
- [7] Forsyth, D. A.; Ponce, J.: *Computer Vision A Modern Approach*. Pearson Education
International, 2003, ISBN 0-13-191193-7.
- [8] Hartley, R.; Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge
Univeristy Press, druhé vydání, 2003.
- [9] Intel: Open Source Computer Vision Library. [online], [cit. 2008-05-05].
URL <<http://www710.univ-lyon1.fr/~bouakaz/OpenCV-0.9.5/docs/>>
- [10] Intel: *OpenCV Reference Manuals*.
- [11] Kršek, P.: *Základy počítačové grafiky IZG*, Studijní opora, dostupno pro zapsané v
předmětu IZG na FIT VUT v Brně.
- [12] Kunc, L.: *Automatická 3D rekonstrukce z 2D snímku otáčejícího se objektu*.
Diplomová práce, České vysoké učení technické v Praze, 2007.
- [13] Lowe, D. G.: Demo Software: SIFT Keypoint Detector. [online], [cit. 2008-04-26].
URL <<http://www.cs.ubc.ca/~lowe/keypoints/>>
- [14] Lowe, D. G.: Distinctive image features from scale-invariant keypoints. *International
Journal of Computer Vision*, ročník 60, 2 2004: s. 99–110.
- [15] Ma, Y.; Soatto, S.; Košecká, J.; aj.: *An Invitation to 3-D Vision: From Images to
Geometric Models*. Springer, 2004, ISBN 0-387-00893-4.

- [16] Nistér, D.: An Efficient Solution Problem to the Five-point Relative Pose Problem. In *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, ročník 26, IEEE, 6 2004.
- [17] Tišnovský, P.: OpenGL a nadstavbová knihovna GLU. [online], [cit. 2008-05-09].
URL <http://www.root.cz/clanky/opengl-a-nadstavbova-knihovna-glu-10//>
- [18] Wikipedia: Interest point detection. [online], [cit. 2008-04-26].
URL http://en.wikipedia.org/wiki/Interest_point_detection
- [19] Wikipedia: Singular value decomposition. [online], [cit. 2008-04-30].
URL http://en.wikipedia.org/wiki/Singular_value_decomposition
- [20] Yang, A. Y.: Image Formation and Camera Models. [online], [cit. 2008-04-29].
URL <http://inst.eecs.berkeley.edu/~ee225b/sp07/lectures/lec12.pdf>

Seznam použitých zkratk

2D Two-dimensional

3D Three-dimensional

RANSAC Random Sample Consensus

SIFT Scale-Invariant Feature Transform

SVD Singular Value Decomposition

Seznam příloh

Příloha 1. CD se zdrojovými kódy a dokumentu v pdf formátu.