

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

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

BC. FILIP CÍR

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

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

BC. FILIP CÍR

VEDOUCÍ PRÁCE
SUPERVISOR

ING. MICHAL ŠPANĚL

BRNO 2008

Zadání diplomové práce

Řešitel: **Cír Filip, Bc.**
Obor: Počítačová grafika a multimédia
Téma: **Rekonstrukce 3D objektu z obrazových dat**
Kategorie: Počítačová grafika

Pokyny:

1. Prostudujte základy zpracování obrazu. Zaměřte se zejména na problematiku detekce hran a význačných bodů.
2. Prostudujte dostupné materiály na téma 3D rekonstrukce objektů z obrazu a zorientujte se v současných metodách.
3. Vyberte vhodnou metodu a vygenerujte syntetická data pro její testování.
4. Experimentujte s vaší implementací a případně navrhnete vlastní modifikace metod.
5. Porovnejte dosažené výsledky a diskutujte možnosti budoucího vývoje.
6. Vytvořte stručný plakát prezentující vaši diplomovou práci, její cíle a výsledky.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části diplomového projektu je požadováno:

- Splnění prvních tří bodů zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese
<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVR-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Španěl Michal, Ing.,** UPGM FIT VUT

Datum zadání: 24. září 2007

Datum odevzdání: 19. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2

L.S.



doc. Dr. Ing. Pavel Zemčík
vedoucí ústavu

**LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO**

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Bc. Filip Cír**
Id studenta: 89499
Bytem: Kostelecká 1830, 547 01 Náchod
Narozen: 25. 03. 1983, Náchod
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

**Článek 1
Specifikace školního díla**

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Rekonstrukce 3D objektu z obrazových dat
Vedoucí/školitel VŠKP: Španěl Michal, Ing.
Ústav: Ústav počítačové grafiky a multimédií
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě	počet exemplářů: 1
elektronické formě	počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....

Nabyvatel



.....
Autor

Abstrakt

Tato práce se zabývá 3D rekonstrukcí z obrazových dat. Jsou popsány možnosti a přístupy k optickému skenování. Ruční optický 3D skener se skládá z kamery a zdroje čárového laseru, který je vzhledem ke kameře upevněn pod určitým úhlem. Je navržena vhodná podložka se značkami a je popsán algoritmus pro jejich real-time detekci. Po detekci značek lze vypočítat pozici a orientaci kamery. Na závěr je popsána detekce laseru a postup při výpočtu bodů na povrchu objektu pomocí triangulace.

Klíčová slova

Eroze, dilatace, Houghova transformace, aktivní kontury, optické 3D skenování, triangulační skener, kalibrace kamery, pozice a orientace kamery, triangulace.

Abstract

This paper deals with 3D reconstruction of objects from image data. There is describes theoretical basis of the 3D optical scanning. Handheld 3D optical scanner setup is described composed of a single camera and a line laser whose position is fixed with respect to the camera. Set of image markers and a simple real-time detection algorithm are proposed. Detected markers are used to estimate position and orientation of the camera. Finally, laser detection and triangulation of points lying on object surface are discussed.

Keywords

Erode, dilate, Hough transform, active contours, optical 3D scanning, triangulation scanner, camera calibration, position and orientation of the camera, triangulation.

Citace

Cír Filip: Rekonstrukce 3D objektu z obrazových dat. Brno, 2008, diplomová práce, FIT VUT v Brně.

Rekonstrukce 3D objektu z obrazových dat

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Michala Španěla. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jméno Příjmení
Datum

Poděkování

Rád bych poděkoval vedoucímu mé diplomové práce Ing. Michalu Španělovi za odborné rady a za výrobu 3D skeneru.

© Filip Cír, ROK.

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

Obsah	1
Seznam obrázků	2
1 Úvod	4
2 Zpracování rastrového obrazu	5
2.1 Barevné modely a histogram obrazu	5
2.2 Metrické a topologické vlastnosti obrazu	8
2.3 Sousednost a souvislost pixelů	9
2.4 Morfologické operace	10
2.5 Shluková analýza (clustering)	13
2.6 Aktivní kontury a Houghova transformace	14
3 Optické 3D skenování	17
3.1 Aktivní - laserové	17
3.2 Pasivní	23
4 Návrh algoritmu	25
4.1 Kalibrace kamery a laseru	26
4.2 Detekce značek na podložce	27
4.3 Výpočet pozice kamery	29
4.4 Detekce laserového paprsku	30
4.5 Triangulace – výpočet 3D souřadnic bodů	33
4.6 Výstup - VRML	35
5 Implementace	36
5.1 Ruční 3D Skener	36
5.2 Aplikační rozhraní	37
5.3 Konzolová aplikace	39
5.4 Instalační program	43
6 Analýza výsledků	44
6.1 Triangulace – 3D rekonstrukce	44
6.2 Výpočet pozice kamery	46
6.3 Přesnost rekonstrukce	47
6.4 Rychlost rekonstrukce	49
7 Závěr	50
Literatura	51
Seznam příloh	53

Seznam obrázků

obr 2.1: Aditivní skládání barev (RGB) [11].....	5
obr 2.2: Jednotková RGB krychle [11].....	6
obr 2.3: HSV model – šestiboký jehlan postavený „na špičce“ [11]	6
obr 2.4: Model HLS – dva zrcadlově souměrné kužely [11].....	7
obr 2.5: Okolí okamžitého obrazového elementu [6]	9
obr 2.6: Diskrétní úsečky [6]	9
obr 2.7: Bodové množiny [6].....	10
obr 2.8: Dilatace [6].....	11
obr 2.9: Eroze [6].....	12
obr 2.10: Rozdělení shluků na oblasti [9].....	14
obr 2.11: Algoritmus aktivní kontury [13].....	15
obr 2.12: Detekce kružnic pomocí Houghovy transformace [10].....	16
obr 3.1: FARO Laser ScanArm V3 [18].....	18
obr 3.2: Ukázka práce s FARO Laser ScanArm V3 [18]	19
obr 3.3: Skener Handyscan 3D [19]	20
obr 3.4: Zjištění polohy pomocí zařízení HandyScan [19].....	20
obr 3.5: Snímání pomocí zařízení HandyScan [19].....	21
obr 3.6: Ukázky naskenovaných objektů skenerem Handyscan 3D [19]	21
obr 3.7: Skener NextEngine [20].....	22
obr 3.8: Ukázka 3D modelů vytvořených pomocí skeneru NextEngine [20].....	22
obr 3.9: Systém Atos [21].....	23
obr 3.10: Ukázka práce skeneru Atos [21]	23
obr 4.1: Návrh algoritmu.	25
obr 4.1: Kalibrace laseru.....	26
obr 4.3: Kalibrace kamery	27
obr 4.4: Návrh podložky	27
obr 4.5: Filtrace barev v RGB.....	28
obr 4.6: Výpočet pozice kamery	29
obr 4.7: Odstíny vzniklé aplikací k-means na histogram.....	30
obr 4.8: 3D histogram a hranice oblastí na podstavě [2]	31
obr 4.11: Filtrační maska	32
obr 4.10: Triangulace - boční pohled.....	33
obr 4.11: Triangulace - pohled shora.....	33
obr 4.12: Triangulace - pohled na promítací rovinu.	34

obr 4.13: Triangulace - poloha pomocného bodu M.	35
obr 5.1: Ruční 3D skener	36
obr 5.2: Ukázka práce se skenerem	37
obr 5.3: Záložka Kalibrace	38
obr 5.4: Záložka Skenování	38
obr 5.5: Záložka Výstup	39
obr 5.6: Záložka Nastavení	39
obr 5.7: Paprsek filtrovaný pomocí 4 různých Gaussových rozložení	40
obr 5.8: Filtrovaný paprsek popsáný více Gaussovými rozložení	41
obr 5.9: Detekce laseru v RGB	41
obr 5.10: Aktuálně detekované a identifikované značky na podložce	42
obr 5.11: Real-time 3D vizualizace	42
obr 5.12: Instalační program	43
obr 6.1: Scéna ve 3D studiu MAX	44
obr 6.2: Objekt, na který dopadá laserový paprsek	44
obr 6.3: Zrekonstruovaný objekt ve VRML	45
obr 6.4: Zrekonstruovaný objekt + originální model (červeně)	45
obr 6.5: Detekce značek	46
obr 6.6: Zobrazení pozice kamery	46
obr 6.7: Skenování podložky	47
obr 6.8: Zrekonstruovaná podložka	47
obr 6.9: Graf výšek bodů nad rovinou podložky	48

1 Úvod

Cílem této práce je návrh algoritmu umožňující real-time 3D rekonstrukci a jeho následná implementace.

V posledních letech dochází k výraznému rozvoji optického skenování. Optické skenování nachází uplatnění především v průmyslové výrobě pro odhalení výrobních vad výrobků nebo při reverse engineeringu pro získání geometrie existujícího výrobku.

Existuje mnoho přístupů k optickému skenování. V zásadě se dělí na aktivní a pasivní. V případě pasivních skenerů se často jedná o princip stereoskopického vidění. U aktivního přístupu je na povrch snímaného objektu vysílán čárový laserový paprsek a na základě pokrivení tohoto paprsku lze zjistit tvar snímaného objektu. Tato práce vychází z principu triangulačního skeneru s jednou kamerou.

První kapitola se zabývá možnostmi zpracování a úpravami rastrového obrazu. Jsou zde popsány barevné modely a základní morfologické operace. Druhá kapitola teoreticky rozebírá možnosti a postupy při 3D optickém skenování. Další kapitola pojednává o návrhu algoritmu pro 3D rekonstrukci. Je zde nastíněno pořadí jednotlivých kroků v algoritmu a jednotlivé kroky jsou následně popsány. Ve čtvrté kapitole je popsána implementace, aplikační rozhraní a ruční 3D skener. Poslední kapitola analyzuje výsledky, rychlost a přesnost rekonstrukce. V závěru jsou diskutovány problémy a možné vylepšení.

V příloze je uveden postup pro odvození vzorců pro výpočet pozice kamery a vzorců pro triangulaci. Jsou zde ukázány dosažené výsledky a je zde uživatelská příručka popisující ovládání programu. V poslední části je zobrazena podložka se značkami a šachovnice pro kalibraci kamery.

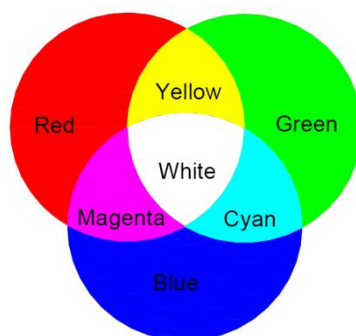
2 Zpracování rastrového obrazu

2.1 Barevné modely a histogram obrazu

Světlo je elektromagnetické vlnění v rozsahu $4,3 \cdot 10^{14} - 7,5 \cdot 10^{14}$ Hz. Barva tělesa je dána schopností pohlcovat a odrážet záření o určitých frekvencích. Barva (barevný tón) závisí na dominantní frekvenci. Jas (svítivost) odpovídá intenzitě světla. Sytost čistota barvy světla je definována tak, že čím větší je sytost, tím užší je spektrum barevných frekvencí ve světle obsažených. Světlost je velikost achromatické složky ve světle s dominantní frekvencí. Barevnost slučuje sytost a dominantní frekvenci [5].

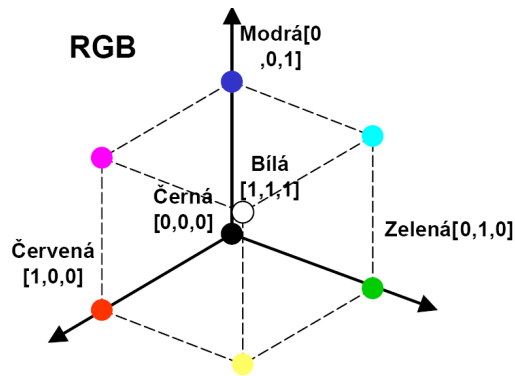
2.1.1 Barevný model RGB (Red Green Blue)

V tomto barevném modelu se barva určuje pomocí zastoupení tří základních barev: červené, zelené a modré. Je odvozen ze způsobu, jakým obrazovka zobrazuje barvy. Jeden bod na obrazovce je tvořen optickým splynutím tří různých bodů – červeného, zeleného a modrého. Poměr intenzit těchto bodů vytváří konkrétní barvu. Tento barevný model je aditivní (obr 2.1), což znamená, že výsledná barva je tím světlejší, čím vyšší je intenzita jednotlivých složek. Plná intenzita všech tří barev pak dává bílou.



obr 2.1: Aditivní skládání barev (RGB) [11]

Barva v modelu RGB je reprezentována třemi hodnotami z intervalu $\langle 0, 1 \rangle$ (vžil se termín barevný vektor). Dost často se také používá ukládání v celočíselném rozsahu $\langle 0, 255 \rangle$. Celý barevný rozsah je pak možné zobrazit pomocí jednotkové RGB krychle (obr 2.2). Černá se nachází v bodě $(0, 0, 0)$, bílá v bodě $(1, 1, 1)$. Na jednotlivých osách leží tři základní barvy, ve zbylých vrcholech pak barvy doplňkové [4].

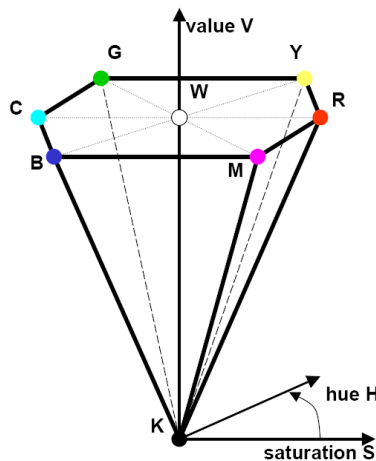


obr 2.2: Jednotková RGB krychle [11]

2.1.2 Barevný model HSV (Hue Saturation Value)

Také tento model používá k určení barvy tři složky, nejde však již o základní barvy. Tyto složky se nazývají odstín, sytost a jasnost. Odstín určuje převládající spektrální barvu (od červené po fialovou), sytost je dána množstvím příměsi ostatních barev. Množství bílého světla v barvě pak určuje jasnost. Tento přístup je lidskému chápání barev mnohem bližší. Člověk obvykle ví, jaký odstín hledá a poté ho upraví pomocí sytosti a jasu na požadovanou úroveň.

K číselné reprezentaci modelu se opět používá tříslučkový barevný vektor. Prostorovým zobrazení barevného rozsahu daného HSV modelem však již není krychle. HSV model se reprezentuje jako šestiboký jehlan, postavený „na špičce“ (obr 2.3). Vrchol jehlanu je tvořen černou barvou. Barvy spektra se nalézají na obvodu podstavy. Každý vrchol udává jednu základní, resp. hlavní doplňkovou barvu v pořadí: červená, žlutá, zelená, azurová, modrá, fialová. Ve středu podstavy se nachází bílá barva. Význam tří složek barevného modelu je následující. Odstín je dán úhlem v rozsahu $\langle 0^\circ, 360^\circ \rangle$. Ten určuje odklon od spojnice červená-bílá a tím i barvu ve spektru. Vzdálenost od osy jehlanu určuje sytost barvy (odstíny šedé leží na ose jehlanu – mají nulovou sytost). Konečně výška (vzdálenost od vrcholu) odpovídá jasnosti barvy.



obr 2.3: HSV model – šestiboký jehlan postavený „na špičce“ [11]

Model HSV je mnohem intuitivnější než technicky orientované modely, ale má některé drobné nedostatky. Např. z hlediska výpočtů by bylo mnohem vhodnější, kdyby se bod, probíhající barevné spektrum pohyboval po kružnici a ne po šestiúhelníku. Druhým nedostatkem je to, že největší „výběr“ barev máme při nejvyšší hodnotě jasu. To neodpovídá způsobu, jakým barvy vnímá lidské oko, které je nejcitlivější při středních hodnotách jasu [4].

Hbarevný tón (hue), $\langle 0, 360^\circ \rangle$ udává převládající spektr.

Ssyťost (saturation), $\langle 0, 1 \rangle$ určuje čistotu barvy (příměš jiných barev)

V (B) ...jasová hodnota (value, brightness) $\langle 0, 1 \rangle$ dána množstvím bezbarvého světla

2.1.3 Barevný model HLS (Hue Lumination Saturation)

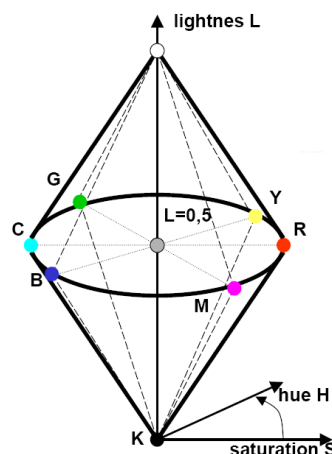
Zmíněné nedostatky modelu HSV se snaží odstranit model HLS. Význam jednotlivých složek je velmi obdobný: odstín, světlost a syťost. Ve skutečnosti jde pouze o formální odlišení. Rozdílný je ovšem způsob mapování těchto hodnot na příslušnou barvu.

Prostorovým zobrazením barevného modelu HLS jsou dva zrcadlově souměrné kužely, sdílející stejnou podstavu (obr 2.4). Barevné spektrum se nachází po obvodu společné podstavy. Syťost je dána vzdáleností od osy kuželů, světlost pak výškou, resp. vzdáleností od vrcholu spodního kužele (černá barva). Konkrétní barevný odstín ve spektru je opět určen úhlem. Filozofie modelu HLS je tedy stejná, jako u modelu HSV, ale s ohledem na výše zmíněné požadavky (výpočetně jednodušší kružnice a maximální rozsah pro střední hodnotu jasu) [4].

H..... barevný tón (hue), $\langle 0, 360^\circ \rangle$ udává převládající spektr.

L světlost (lightness), $\langle 0, 1 \rangle$ velikost achromatické složky

S..... syťost (saturation) $\langle 0, 1 \rangle$ určuje čistotu barvy (příměš jiných barev)



obr 2.4: Model HLS – dva zrcadlově souměrné kužely [11]

2.1.4 Histogram obrazu

Důležitým klíčem k charakterizaci obrazu je histogram, který kvantifikuje množství a frekvenci barev obsažených v obraze. Matematicky je histogram vektor absolutních četností hodnot zastoupených v obraze. Jinými slovy řečeno, hodnota histogramu H pro index i říká, kolik pixelů v obraze má intenzitu i . Pokud je původní obraz složen z jasových složek, je histogram jednorozměrný vektor, v případě složek RGB je složen ze tří vektorů [1].

Histogram je statistickou veličinou. Můžeme ho také chápat jako pravděpodobnost výskytu pixelu barvy v obraze. Histogram kvantifikuje jasové poměry v obraze a nenese žádnou informaci o jejich plošném rozložení. Histogram velmi často bývá jedinou globální informací o obraze.

2.2 Metrické a topologické vlastnosti obrazu

V digitálním zpracování obrazů se používá několik definic vzdálenosti. Mějme dva obrazové body se souřadnicemi (x, y) , (h, k) . *Euklidovská vzdálenost* D_E je známa z elementární geometrie a každodenní zkušenosti,

$$D_E((x, y), (h, k)) = \sqrt{(x-h)^2 + (y-k)^2}$$

Její výhodou je názornost. Nevýhodou je složitější výpočet kvůli odmocnině a skutečnost, že výsledkem je neceločíselná hodnota, což neodpovídá kvantovaným hodnotám v digitálních obrazech.

Vzdálenost mezi dvěma body si lze také představit jako nejmenší počet kroků jednotkové vzdálenosti nutných pro přesun z výchozího do cílového bodu v pravoúhlé diskretizační mřížce. Jsou-li povoleny jen kroky o jedno pole v mřížce buď ve vodorovném, nebo ve svislém směru, získáme vzdálenost D_4 , které se říká vzdálenost v městských blocích díky analogii se vzdáleností ve městě s pravoúhlou sítí ulic,

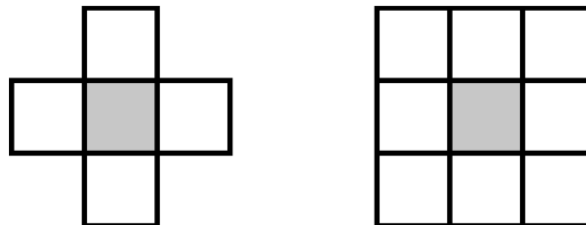
$$D_4((x, y), (h, k)) = |x-h| + |y-k|$$

Když navíc připustíme i jednotkové kroky v diagonálním směru, je odpovídající vzdáleností D_8 , které se také říká šachovnicová vzdálenost, protože odpovídá nutnému počtu kroků krále na šachovnici,

$$D_8((x, y), (h, k)) = \max\{|x-h| + |y-k|\}$$

2.3 Sousednost a souvislost pixelů

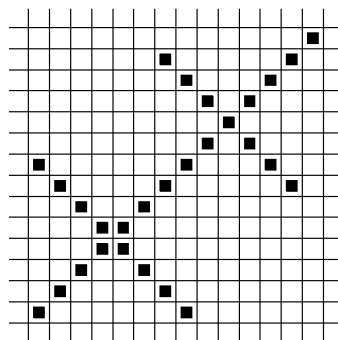
Pro digitální obrazy je dalším potřebným pojmem sousednost pixelů. Dva pixely jsou 4-sousedy, pokud je jejich vzdálenost $D_4 = 1$. Podobně jsou dva pixely 8-sousedy, když $D_8 = 1$. Znárodnění 4-sousedství a 8-sousedství na obr 2.5.



obr 2.5: Okolí okamžitého obrazového elementu [6]

Dva pixely v obraze, mezi nimiž existuje cesta, se nazývají souvislé pixely. Alternativně bychom mohli říci, že oblast tvoří takové pixely, jejichž každá dvojice je souvislá. Relace „být souvislým“ je reflexivní, symetrická a transitivní. Definuje tudíž rozklad množiny (v našem případě obrázku) na třídy ekvivalence (oblasti).

Sousednost a souvislost definovaná na diskretní čtvercové mřížce výše uvedeným způsobem vede k paradoxům. Představme si úsečku v digitálním obraze se sklonem 45° . Při uvažování 4-okolí je však v každém obrazovém elementu nesouvislá. Protíná-li úsečku jiná úsečka, může nastat situace, viz obr 2.6, že buď mají (průsečík vlevo dole), nebo nemají (průsečík vpravo nahoře) společný bod.



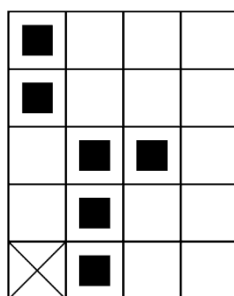
obr 2.6: Diskretní úsečky [6]

2.4 Morfologické operace

Matematická morfologie využívá vlastností bodových množin, výsledky z geometrie a topologie. Výchozím předpokladem je představa, že reálné obrázky lze modelovat pomocí *bodových množin* libovolné dimenze (např. N -rozměrný euklidovský prostor). Dvojměrný euklidovský prostor \mathcal{E}^2 a systém jeho podmnožin je přirozeným definičním oborem pro popis rovinných útvarů.

V počítačovém vidění se používá digitální protějšek euklidovského prostoru. Pro binární matematickou morfologii zpracovávající dvojúrovňový obrázek je základním popisným kamenem množina dvojic celých čísel ($\in \mathbb{Z}^2$). Pro šedotónovou matematickou morfologii zpracovávající obrázek s více úrovněmi jasu jsou základem popisu trojice ($\in \mathbb{Z}^3$).

Binární obraz lze vyjádřit jako 2D bodovou množinu. Body objektů v obraze reprezentují množinu X , což odpovídá pixelům s hodnotou jedna. Počátek má souřadnice (0,0) a souřadnice ostatních bodů (x, y) mají stejný význam, jako je v matematice obvyklé. Obr 2.7 ukazuje příklad bodové množiny. Body objektu jsou zapsány jako malé černé čtverce [6]. Příklad diskrétního bodového obrazu. $X = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$.



obr 2.7: Bodové množiny [6]

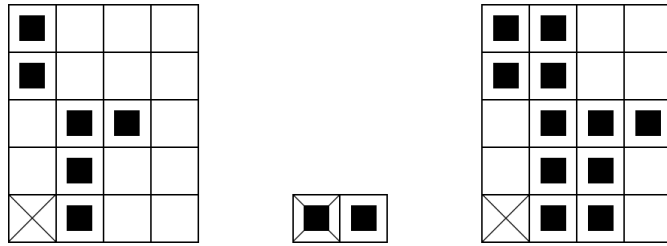
2.4.1 Dilatace

Dilatace \oplus skládá body dvou množin pomocí vektorového součtu (též Minkowského množinový součet, např. $(a,b) + (c,d) = (a+c, b+d)$). Dilatace $X \oplus B$ je bodovou množinou všech možných vektorových součtů pro dvojice pixelů, vždy pro jeden z množiny X a jeden z množiny B

$$X \oplus B = \{p \in \mathcal{E}^2 : p = x + b, \quad x \in X, b \in B\}$$

Ukázka dilatace obr 2.8

$$\begin{aligned} X &= \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\} \\ B &= \{(0,0), (1,0)\} \\ X \oplus B &= \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4), (2,0), (2,1), (2,2), (3,2), (1,3), (1,4)\} \end{aligned}$$



obr 2.8: Dilatace [6]

Vlastnosti dilatace:

Dilatace je komutativní operací

$$X \oplus B = B \oplus X$$

Je také asociativní

$$X \oplus (B \oplus D) = (X \oplus B) \oplus D$$

Dilataci můžeme vyjádřit jako sjednocení posunutých bodových množin

$$X \oplus B = \bigcup_{b \in B} X_b$$

Dilatace je invariantní vůči posunu

$$X_h \oplus B = (X \oplus B)_h$$

Dilatace se používá samostatně k zaplnění malých děr, úzkých zálivů a jako stavební kámen složitějších operací. Dilatace zvětšuje objekty. Má-li se zachovat jejich původní rozměr, kombinuje se dilatace s erozí [6].

2.4.2 Eroze

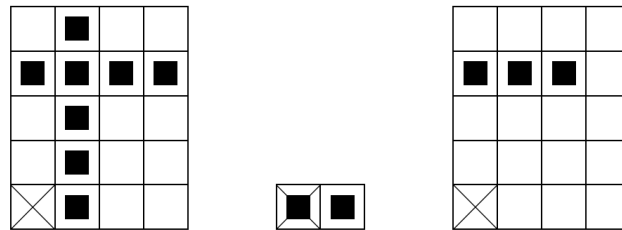
Eroze je duální operací k dilataci. Ani dilatace ani eroze nejsou invertovatelné. Eroze \otimes skládá dvě množiny podle předpisu

$$X \otimes B = \{p \in \mathcal{E}^2 : p + b \in X \text{ pro každé } b \in B\}$$

Předchozí vztah říká, že pro každý bod obrazu p se ověřuje, zda pro všechna možná $p + b$ leží výsledek v X . Pokud ano, zapíše se v reprezentativním bodě do výsledného obrázku 1 a v opačném

případě 0. Obr 2.9 ukazuje bodovou množinu X erodovanou strukturním elementem B , který není všesměrový. Svislá čára zmizí.

$$\begin{aligned} X &= \{(1,0), (1,1), (1,2), (0,3), (1,3), (2,3), (3,3), (1,4)\} \\ B &= \{(0,0), (1,0)\} \\ X \otimes B &= \{(0,3), (1,3), (2,3)\} \end{aligned}$$



obr 2.9: Eroze [6]

Eroze se používá pro zjednodušení struktury objektů – objekty tloušťky 1 se ztratí, a tak se složitější objekt rozdělí na několik jednodušších.

Ekvivalentní vyjádření eroze - B_p označuje množinu B posunutou o vektor p

$$X \otimes B = \{p \in \mathcal{E}^2 : B_p \subseteq X\}$$

Erozi můžeme také interpretovat jako systematické posouvání strukturního elementu B po obrazu X . Když je B posunutý o vektor p obsažen v obrazu X , potom bod odpovídající reprezentativnímu bodu B patří do eroze $X \otimes B$.

Implementace eroze se může zjednodušit, když si všimneme, že erozi obrazu X strukturním elementem B můžeme vyjádřit jako průnik všech posunů obrazu X o vektory $-b \in B$

$$X \otimes B = \bigcap_{b \in B} X_{-b}$$

Eroze je také invariantní vůči posunu

$$X_h \otimes B = (X \otimes B)_h,$$

$$X \otimes B_h = (X \otimes B)_{-h}$$

Eroze se také od dilatace svými vlastnostmi liší. Eroze není na rozdíl od dilatace komutativní [6].

$$X \otimes B \neq B \otimes X.$$

2.5 Shluková analýza (clustering)

Shluková analýza je metoda, která umožňuje v rozsáhlých nestrukturovaných souborech dat nacházet jejich vnitřní strukturu v podobě tzv. *shluků*, což jsou místa s větší koncentrací dat než jejich okolí. V histogramu clustering hledá oblasti se zvýšeným výskytem [8].

K-means clustering

Algoritmus K-means iterativně hledá hodnoty vektorů tak, že minimalizuje střední odchylku mezi zadanou množinou dat a vektory (vzdálenost bodu od etalonu shluku), které mají k těmto datům nejmenší euklidovskou vzdálenost a rozděluje je do předem daného počtu shluků (tříd) K : C_1, C_2, \dots, C_k . Základní algoritmus používá Euklidovskou vzdálenost a μ_j je aritmetický průměr bodu ve shluku (střední hodnota ve třídě) – etalon.

Vstupem algoritmu je množina dat x_1, x_2, \dots, x_l a číslo K udávající počet vektorů $\mu_j, j = 1, \dots, k$. Na začátku se inicializují vektory $\mu_j, j = 1, \dots, k$, na náhodně zvolenou hodnotu nebo použitím nějaké vhodně zvolené heuristiky (např. využívající apriorní znalost o úloze). Po inicializaci se začnou iterativně opakovat následující dva kroky:

1. Klasifikace: Všechna data $x_i, i = 1, \dots, l$, se klasifikují do tříd určených vektory $\mu_i, i = 1, \dots, k$, podle minima euklidovské vzdálenosti. Tedy vzor x_i je přiřazen do třídy y_i podle

$$y_i = \arg \min_j \|x_i - \mu_j\|$$

2. Přepočítání vektorů μ_j : Vypočítají se nové hodnoty vektorů μ_j jako střední hodnoty dat x_i , které byly klasifikovány do třídy určené příslušným vektorem μ_j . Tedy nová hodnota μ_j se spočte podle vztahu $\mu_j = \frac{1}{l_j} \sum_{i=1, y_i=j}^l (x_i)$, kde l_j je počet vzorů x_i klasifikovaných v druhém kroku do třídy určené vektorem μ_j .

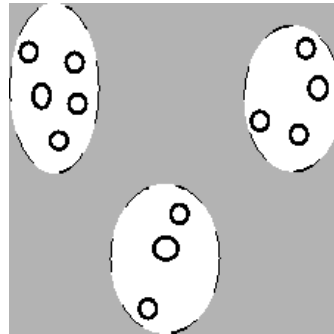
Kroky 1 a 2 se opakují do té doby, dokud se alespoň jeden vektor x_i klasifikuje do jiné třídy, než byl klasifikován v předcházejícím kroku.

Vlastnosti k-means:

- jednoduchý
- prvky se mohou přeskupovat mezi shluky
- pouze pro metrická data
- konverguje v konečném počtu kroků k nějakému řešení

- může existovat více řešení v závislosti na počátečních podmínkách

Experimentální zkušenosti naznačují, že je tento algoritmus pro účely vektorové kvantizace velmi vhodný jak z hlediska kvality rozkladu množiny do shluků, tak i poměrně rozumnými výpočetními nároky [8]. Na obr. 2.10 je ukázka rozdělení shluků na oblasti.



obr 2.10: Rozdělení shluků na oblasti [9].

2.6 Aktivní kontury a Houghova transformace

2.6.1 Aktivní kontury

Aktivní kontury (snakes, active contours) je metoda postupného tvarování kontur až ke hraně objektu v obraze. Model aktivní kontury je řízená uzavřená kontura, která se deformuje vlivem tzv. vnitřních, obrazových a vnějších sil. Vnitřní síly kontrolují hladkost průběhu, obrazové síly směřují tvarování kontury směrem ke hraně objektu a vnější síly jsou výsledkem počátečního umístění kontury.

Mějme konturu:

$$v(s) = (x(s), y(s)) \quad s \in (0,1) \quad (1)$$

Výsledná pozice aktivní kontury koresponduje s lokálním minimem energie kontury:

$$E_{snake}^* = \int_0^1 E_{snake}(v(s)) ds = \int_0^1 (E_{int}(v(s)) + E_{image}(v(s)) + E_{con}(v(s))) ds \quad (2)$$

kde E_{int} je vnitřní energie kontury (ohyb, zlom), E_{image} reprezentuje energii obrazu a E_{con} energii omezení.

Standardní algoritmus nedosahuje při složitějších konturách dostatečných kvalit, není schopen se přimknout k prudkým zlomům hranic objektu, vytváří falešné kontury či smyčky. Ji a Yan navrhl modifikovanou verzi, která některé tyto nedostatky odstraňuje [15] a [16]. Ukázka aktivní kontury je na obrázku 2.11.



obr 2.11: Algoritmus aktivní kontury [13]

2.6.2 Houghova transformace

Umožňuje detekovat přímky, kružnice, elipsy, atd. Je však nutné znát matematický popis tvaru, který se má detekovat. Houghova transformace je málo citlivá na šum a necitlivá k porušení hranic. Funguje i při překrytí části obrazu. Jde o transformaci z Kartézského souřadnicového systému do polárního.

Parametrické vyjádření přímky je:

$$\rho = x \cos \theta + y \sin \theta \quad (3)$$

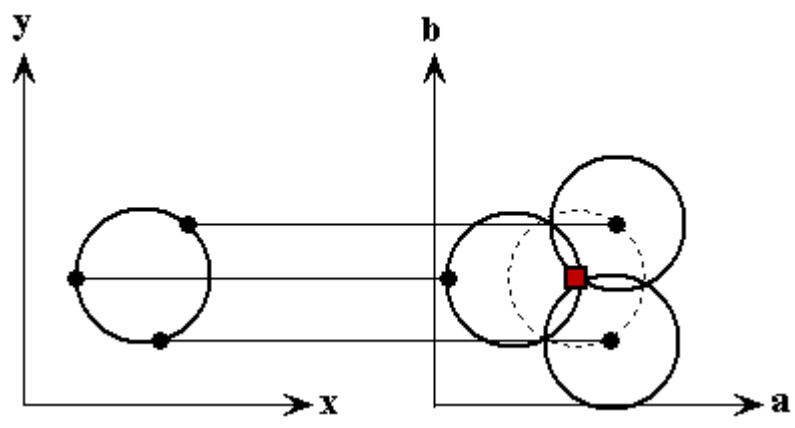
Kde ρ je normalizovaná vzdálenost přímky od počátku a θ je úhel vzhledem k ose x . Body obrazu jsou prezentovány křivkami v parametrickém prostoru. Jedná-li se o kolineární body (z jedné přímky), křivky v parametrickém prostoru se nejčastěji protínají právě v bodě reprezentující parametry hledané křivky (přímky) [17].

Detekce kružnic pomocí Houghovy transformace

Kružnice s poloměrem r a středem (a, b) je popsána parametrickou rovnicí:

$$\begin{aligned} x &= a + r \cos(t) \\ y &= b + r \sin(t) \end{aligned} \quad (4)$$

Střed kružnice lze nalézt tak, že body na kružnici (x, y) se budou brát jako středy kružnice a na průniku těchto kružnic bude ležet hledaný střed (a, b) , viz. obrázek 2.12. Tento postup lze modifikovat i pro detekci elips, tak že parametr r bude rozdílný v ose x a y . Délky hlavní a vedlejší osy elipsy lze následně nalézt algoritmem „ellipse fitting“.



obr 2.12: Detekce kružnic pomocí Houghovy transformace [10]

3 Optické 3D skenování

3.1 Aktivní - laserové

Laserové skenovací systémy umožňují bezkontaktní určování prostorových souřadnic, 3D modelování a vizualizaci staveb, konstrukcí, interiérů, podzemních prostor, terénů i drobnějších předmětů. Naskenovaný objekt může být příslušným softwarem zobrazen jako mračno bodů, z něhož lze vytvořit 3D model, který lze jakkoli upravovat a dotvářet v CAD systému. Je nutné si ovšem uvědomit, že i tato metoda má svá fyzikální a technologická omezení, která mohou měření znehodnotit [3].

Mezi největší výhody laserového skenování patří:

- automatický a systematický sběr bodů
- vysoká hustota těchto bodů a rychlost jejich zaměření (statisíce bodů za sekundu)
- výpočty souřadnic v reálném čase (během měření, nebo hned po něm)
- některé skenery dokážou k 3D souřadnicím bodu přiřadit i hodnotu odrazivosti skenovaného povrchu

V praxi se používá především těchto druhů 3D skenerů:

- statické, pevné fixované na místě (př. na výrobních linkách kontrolují kvalitu)
- mobilní systémy umístované na stativy pro měření na krátké a střední vzdálenosti
- systémy leteckého skenování pro topografické aplikace

Z hlediska použité technologie lze laserové skenery rozdělit:

- triangulační 3D skenery (laboratorní, přenosné)
- laserové skenovací systémy dálkoměrné (stacionární, mobilní, letecké)

Laserové skenovací systémy našly uplatnění v mnoha oborech lidské činnosti jako je například stavebnictví, fotogrammetrie, topografie, dálkový průzkum Země, dokumentace památkových objektů, archeologie, strojírenský průmysl a mnoha dalších [3].

3.1.1 Laserové skenery s měřením délek (ranging scanners)

Systémy založené na impulsním dálkoměru

Laserový impuls je vyslán směrem na měřený objekt a vzdálenost mezi vysílačem a povrchem objektu je spočítána z času, který uběhne mezi vysláním a přijetím signálu. Jelikož je rychlost elektromagnetických vln $3 \cdot 10^8 \text{ m} \cdot \text{s}^{-1}$, je tranzitní čas malý, např. pro délku 30 m je tranzitní čas jen

$2 \cdot 10^{-7}$ s, jsou tedy kladeny vysoké nároky na přesnost měření tranzitního času. (Tento princip měření délek je dobře znám z elektronických dálkoměrů a některé motorizované totální stanice lze naprogramovat jako skenovací zařízení.) Střední chyba měření těchto skenerů se pohybuje řádově v milimetrech a odvíjí se také od přesnosti úhlového určení laserového paprsku [3].

Systémy založené na fázovém dálkoměru

Tato metoda je také dobře známá z elektronických dálkoměrů. V tomto případě je vysílaný signál modulován harmonickým vlněním a vzdálenost se spočítá z fázového rozdílu vyslaného a přijatého signálu. Z uživatelského hlediska se tato metoda příliš neliší od impulsní. Snad podstatnějším rozdílem může být omezenější dosah, protože je nutné přijímat kvalitní signál, což může mít za následek více špatných nebo nekvalitních bodů [3].

Systémy založené na frekvenčním dálkoměru

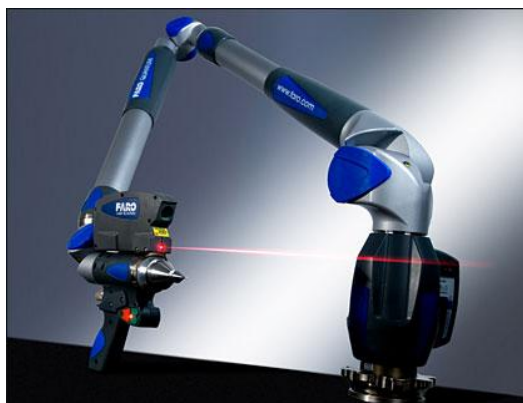
Poněkud méně používaná, ale také známá dálkoměrná metoda. Principem frekvenčního dálkoměru je určení záznamu vynikajícího jako produkt skládání časově posunutých frekvenčně modulovaných vln [3].

3.1.2 Triangulační skenery

Triangulační skener je založen na znalosti polohy kamery, promítací roviny a z popisu roviny čárového laserového paprsku. Pořadí jednotlivých kroků u triangulačního skeneru je:

- kalibrace kamery a polohy laseru
- detekce podložky a výpočet polohy kamery
- detekce laseru a triangulace

Polohu kamery lze zjistit například pomocí vhodné podložky se značkami, na kterou se skenovaný objekt položí. Znalost polohy těchto značek a znalost jejich reálných vzdáleností umožní výpočet pozice kamery. Důležitým údajem je také ohnisková vzdálenost kamery.

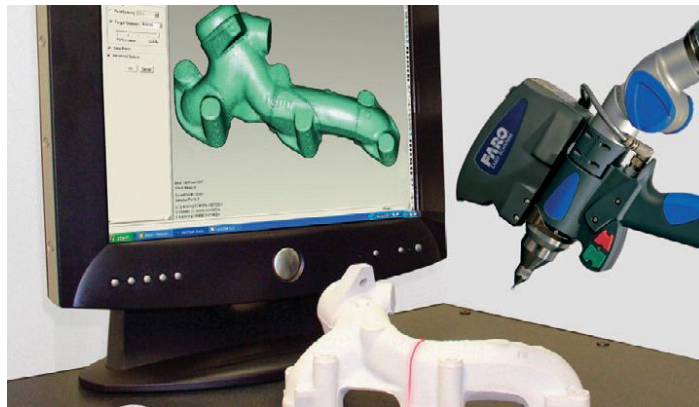


obr 3.1: FARO Laser ScanArm V3 [18]

S jednou kamerou

Tento typ skeneru má na jedné straně základny zařízení emitující laserové záření, které vytváří na objektu stopu bodovou, nebo, pomocí vhodného optického prvku tvořícího laserovou rovinu, stopu čárovou. Na druhé straně základny je umístěna CCD kamera, která detekuje stopu laseru na objektu. Souřadnice respektive vzdálenost se vypočte řešením trojúhelníka. Přesnost takového systému závisí především na vzdálenosti od základny, kterou není většinou možno měnit, proto se uplatní především na malé vzdálenosti a na malé předměty [3].

FARO Laser ScanArm V3 (obrázek 3.1) se používá pro porovnávání s CAD modely, vývoj prototypů, reverse engineering a 3D modelování. Uživatel může sejmout jednotlivé body pevným dotekem, poté naskenovat oblasti požadující podrobnější vyhodnocení (s rychlostí snímání 19 000 bodů za sekundu). Systém je kompatibilní se softwary Geomagic, Polyworks, Rapidform a mnoha dalšími [18].



obr 3.2: Ukázka práce s FARO Laser ScanArm V3 [18]

FARO Laser ScanArm má Implementovány funkce pro skenování tmavých a reflexních povrchů. Rychlost snímání až 19 200 bodů za sekundu. Možnost použití pevného doteku a laserového skeneru současně. Přesnost skeneru je $\pm 0,055$ mm. Na obrázku 3.2 je ukázka práce skeneru [18].

S dvěma kamerami

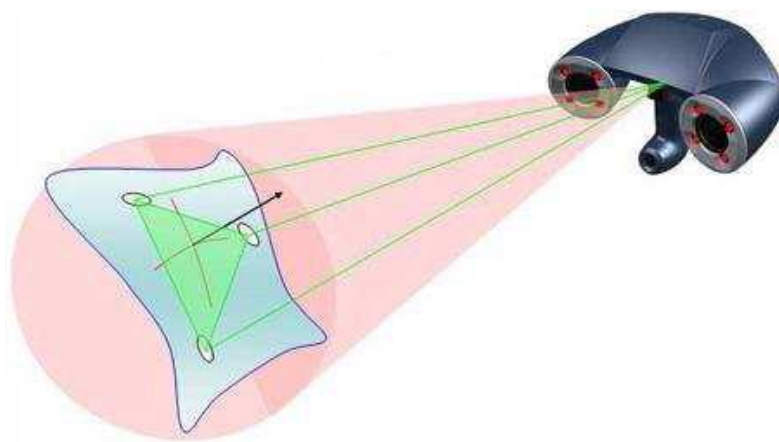
Při tomto řešení je na každém konci základny po jedné CCD kameře. Detekuje se stopa laserového paprsku, který vysílá zvláštní projektor, který však nemá žádnou měřicí funkci. Detekována nemusí být jen laserová stopa, ale i pruhy či jiný strukturovaný vzor. Při výpočtu polohy bodů se vychází opět z řešení trojúhelníka. Dosahuje se stejné přesnosti jako v případě s jednou kamerou a rovněž použití je stejné [3].

Příklad triangulačního skeneru se dvěma kamerami je obrázku 3.3. Skener Handyscan 3D vytváří přímo polygonovou síť, výslednou síť lze uložit do formátu STL. Formát STL patří ke standardním formátům používaných v CAD ale i FEA aplikacích. HandyScan 3D je přímo integrován v některých aplikacích, např. Geomagic, Polyworks, lze pracovat přímo s nasnímanými daty [19].



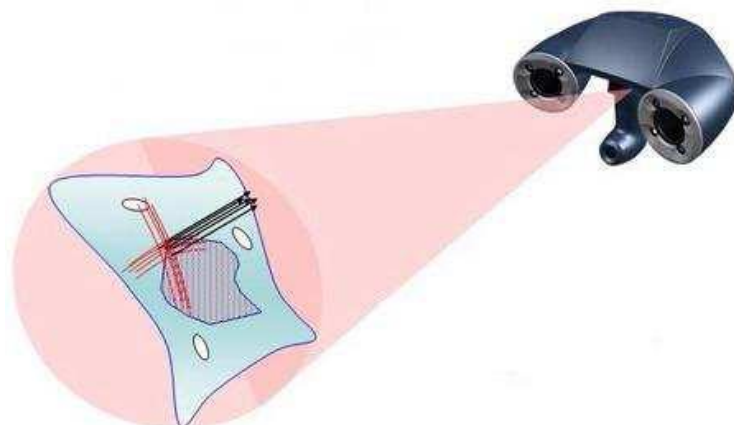
obr 3.3: Skener Handyscan 3D [19]

Během snímání Handyscan 3D zjišťuje v reálném čase polohu reflexních značek, vypočítá jejich vzájemnou polohu a určí pozici scanneru vůči snímanému objektu. Reflexní značky vytvářejí nepravidelnou síť a tu si scanner zaznamenává. Poloha scanneru vůči této síti je jedinečná, protože reflexní značky jsou umístěny náhodně. Tato síť pomáhá určit polohu scanneru stejně, jako GPS používá satelity. Přesnost Handyscanu je $\pm 0,1$ mm [19].



obr 3.4: Zjištění polohy pomocí zařízení HandyScan [19]

Snímání pomocí zařízení HandyScan je založeno na principu triangulace (obrázek 3.4). Reflexní značky mohou být umístěny na podložce, nebo přímo na snímaném tělese. Pokud obě kamery současně vidí minimálně 3 reflexní značky, je systém schopen automaticky určit polohu snímaného tělesa. Pro umožnění pohybu je nutné vidět 4 reflexní značky, vzájemná poloha reflexních bodů je jedinečná a systém tedy kdykoli dokáže navázat na již nasnímané body dalším snímáním [19].



obr 3.5: Snímání pomocí zařízení HandyScan [19]

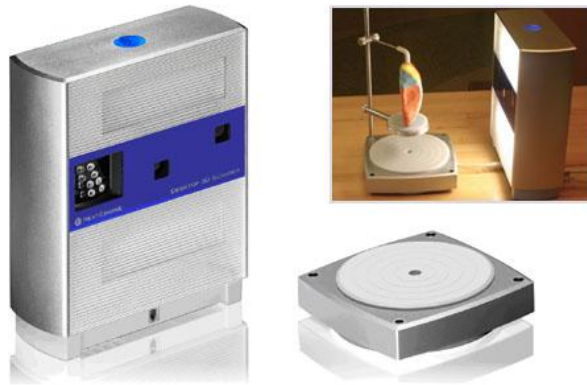
Obě kamery snímají laserový kříž z povrchu tělesa, a systém tak získá informace o ploše (obrázek 3.5). Program zaznamenává všechny informace ze snímání, na základě nastaveného rozlišení a pracovního prostoru pak generuje síť. Při změně nastavení systém plochu znovu přepočítá. Vzniklou síť je možné v programu upravovat, automaticky vymazávat izolované malé části sítě. Rozlišení je definované na základě velikosti pracovního prostoru ve třech úrovních (nízké, střední, vysoké), ve vybrané oblasti je možné rozlišení zlepšovat [19].

Ukázky naskenovaných objektů skenerem Handyscan 3D jsou na obrázku 3.6.



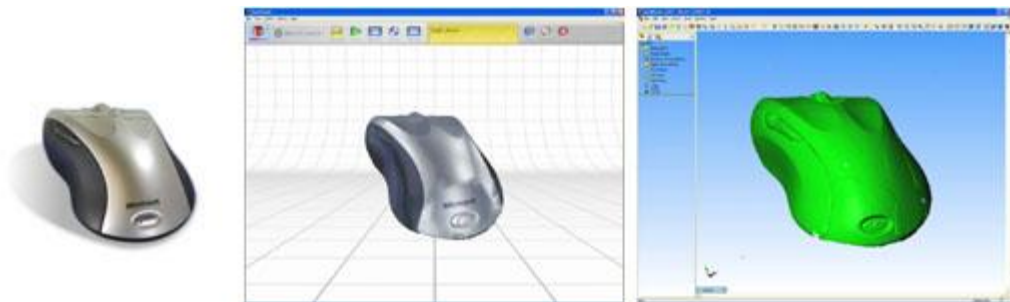
obr 3.6: Ukázky naskenovaných objektů skenerem Handyscan 3D [19]

Dalším dvoukamerovým skenerem je NextEngine (obrázek 3.7). Proces měření je založen na principech optické triangulace. Na povrch objektu jsou promítány 4 pruhy světla, které jsou snímány pomocí dvou kamer. Obslužný program z těchto snímků vypočítá prostorové souřadnice jednotlivých bodů. Přesnost skeneru NextEngine je $\pm 0.13\text{mm}$ [20].



obr 3.7: Skener NextEngine [20]

Zároveň je objekt vyfocen a vzniklý obrázek je namapován na snímaný povrch. Složení jednotlivých snímků do jednoho celku je zajištěno pomocí referenčních značek umístěných na snímaném objektu. Pro nasnímaní celého modelu lze použít otočný stůl nebo je možné libovolně pohybovat skenerem i snímaným objektem [20]. Ukázka 3D modelů vytvořených pomocí skeneru NextEngine obrázek 3.8.



obr 3.8: Ukázka 3D modelů vytvořených pomocí skeneru NextEngine [20]

Pro vysokou hustotu dat (1 snímek - přibližně 400 000 pixelů) lze použít optické 3D souřadnicové zařízení, systém Atos (Advanced Topometric Sensor) obrázek 3.9. Uplatnění tedy nalezne v technologiích CAD, CAM nebo FEM. Měření je založeno na principu triangulační metody a digitálního image processingu [21].

Před vlastním skenováním se na objekt pravidelně umístí pomocné značky, pomocí kterých systém vypočte pozici měřících senzorů a změřené hodnoty transformuje přímo do souřadného systému objektu. Na povrch objektu jsou promítnuty proužky světla a pomocí dvou kamer jsou snímány z různých úhlů. Digitálním zpracováním obrazu se s vysokou přesností vypočtou 3D souřadnice každého pixelu na snímku pro každou kameru zvlášť. Protože triangulace je založena na CCD geometrii a projektor je při měření současně kalibrován, nedochází při změnách okolního osvětlení ke zkreslení souřadnic. Přesnost měření, která se pohybuje v rozmezí $\pm 0,05\text{mm}$, se dá srovnat s přesností mechanických měřících přístrojů [21].

Celé zařízení je koncipováno tak, aby jeho zvládnutí kladlo minimální požadavky na uživatele. Měřicí hlava přístroje se snadno nastavuje kolem snímaného objektu jelikož je umístěna na

fotografickém stavu. Není nutné objekt skenovat po pravidelných úsecích (např. po 30°), ale stačí udělat snímky nepravidelně a díky pomocným značkám systém vyhodnotí přesnou pozici snímku [21].



obr 3.9: Systém Atos [21]

Tuto metodu 3D digitalizace lze aplikovat pro libovolné objekty (obrobky, modely, formy, části lidského těla,...) o rozměrech 10 až 500mm bez přímého kontaktu s nimi. V CAD technologiích je zjištěná data možno použít např. pro vytvoření počítačového modelu z fyzického objektu (reverzní inženýrství), pro zpětnou kontrolu rozměrů výrobku nebo pro přenos změn provedených na vyrobené součásti do CAD programu. Data jsou vhodná také pro vytvoření řídicích programů pro výrobu na NC frézách a systémech Rapid Prototyping [21]. Ukázka práce skeneru Atos obrázek 3.10.



obr 3.10: Ukázka práce skeneru Atos [21]

3.2 Pasivní

Pasivní skenery nevysílají žádné záření, pouze detekují ambientní záření. Většina skenerů tohoto typu detekuje viditelné světlo. Některé typy skenerů detekují také infračervené světlo. Pasivní skenování není finančně náročné, protože ve většině případů není třeba žádný speciální hardware [7].

Stereoskopické vidění

Stereoskopické systémy obvykle využívají dvě mírně vzdálené videokamery, které sledují stejnou scénu. Analýzou rozdílů mezi obrazy, které vidí obě kamery, je možné určit vzdálenost každého bodu obrazu. Tato metoda funguje na stejném principu, na jakém funguje také stereoskopické vidění u člověka [7].

Silueta

Tento typ 3D skenerů používá siluety vytvořené ze sekvence fotografií kolem trojrozměrného objektu proti kontrastnímu pozadí. Tato technika neumožňuje detekovat některé konkávní tvary (např. vnitřek válce) [7].

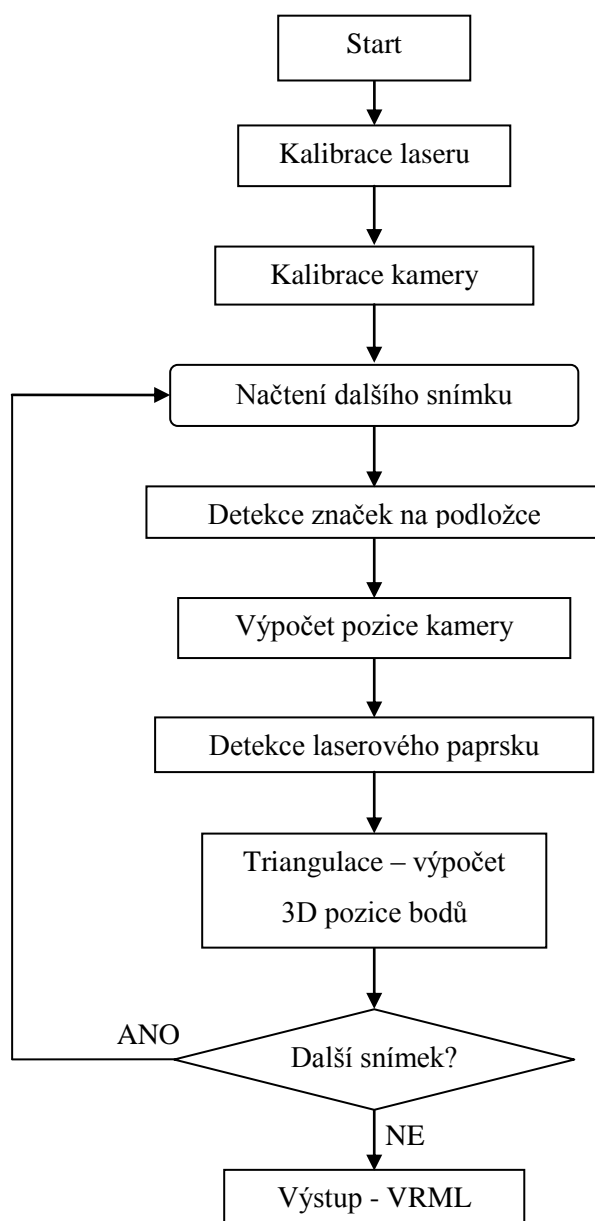
S asistencí uživatele

Tato metoda je založena na asistenci uživatele při detekci a identifikaci některých rysů a tvarů na množině rozdílných obrázků. Tato technika je užitečná pro rychlou aproximaci jednoduchých objektů, které mají jednoduché tvary, jako jsou např. budovy. Komerční programy založené na této metodě jsou iModeller, D-Sculptor nebo RealViz-ImageModeler [7].

4 Návrh algoritmu

Pro implementaci jsem si zvolil skenování pomocí triangulačního skeneru s jednou kamerou. Triangulační skener s jednou kamerou má nejnižší náklady na pořízení, stačí jedna kamera a jeden zdroj laserového paprsku. Výhodný je i z hlediska výroby vlastního skenovacího zařízení, postačí držák, na jehož jeden konec se umístí kamera a na druhý laserový zdroj. Pozice bodů na povrchu objektu respektive jejich vzdálenost se vypočte řešením trojúhelníka.

Návrh algoritmu:

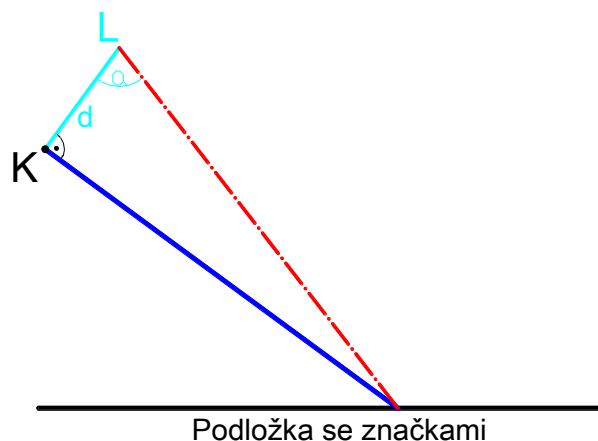


obr 4.1: Návrh algoritmu.

4.1 Kalibrace kamery a laseru

4.1.1 Kalibrace laseru

Při každé změně úhlu sklonu laserového paprsku je nutné provést kalibraci laseru. K výpočtu úhlu lze použít stejnou podložku, jaká se bude využívat pro zjištění pozice kamery.



obr 4.2: Kalibrace laseru

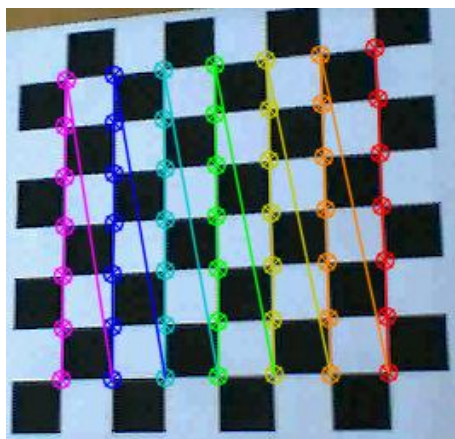
Před kalibrací je třeba vypočítat polohu kamery a vypočítat střední hodnotu polohy laserového paprsku na promítací rovině, dále se využívá se stejný vzorec jako u triangulace, s tím rozdílem, že neznámou není poloha, ale úhel sklonu laserového paprsku. Poloha bodu na promítací rovině je známa, bod leží na průsečíku přímky jdoucí kamerou a bodem na promítací rovině a rovinou podložky, obrázek 4.2. Pro minimalizaci chyby se úhel se vypočítá několikrát, např. dvacetkrát a hodnota se zprůměruje.

4.1.2 Kalibrace kamery

Pro výpočet 3D souřadnic bodů na snímaném objektu je nutné znát kromě úhlu sklonu roviny s laserem vůči kameře také parametry kamery. Mezi parametry kamery patří ohnisková vzdálenost a konstanty pro přepočtení vzdáleností v ose X a v ose Y. Další parametr je matice popisující pokrivení obrazu (distortion).

Kalibrace probíhá tak, že se vezme etalon, v počítačovém vidění se nejčastěji používá šachovnice. U etalonu se nadetekují důležité body, u šachovnice jsou to rohy jednotlivých políček, na základě znalosti polohy těchto bodů lze vypočítat parametry kamery. Pro co nejpřesnější výpočet parametrů je třeba provést detekci důležitých bodů a následný přepočtení několikrát a z různých pohledů na šachovnici. Parametry kamery se změnou pohledu nemění, jsou invariantní.

Pro výpočet lze využít funkci z knihovny OpenCV, kde již je funkce na kalibraci kamery implementována, obrázek 4.3.

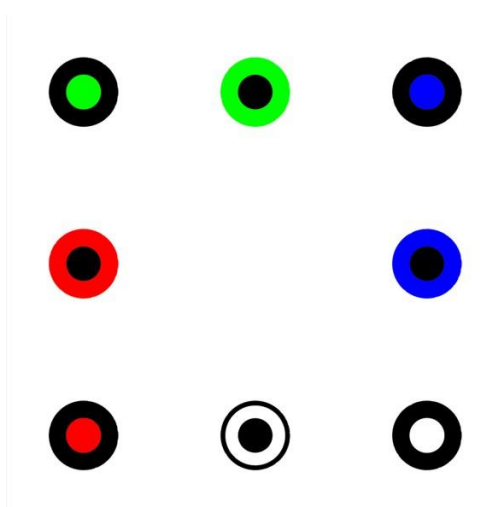


obr 4.3: Kalibrace kamery

4.2 Detekce značek na podložce

K určení polohy kamery lze použít několika metod. Jenou z metod je použití nějakého referenčního 3D objektu se známými rozměry. Na projekční rovině se pak hledají body odpovídající bodům skutečného 3D tělesa. Pro tento postup je v OpenCV [14] již implementována funkce POSIT, která po zadání rozměrů původního 3D objektu a 2D souřadnic odpovídacích promítnutých bodů spočítá translační a rotační matici. Nevýhodou je to, že snímaný objekt může referenční 3D objekt zakrýt a zabránit tak výpočtu rotační a translační matice.

Další možností je použití podložky ve tvaru šachovnice a výpočet translační a rotační matice na základě odpovídajících si bodů. Pro detekci rohů na šachovnici je v OpenCV již implementován algoritmus. Opět je zde stejný problém jako v předešlém případě když snímaný objekt šachovnici překryje a nebude možné vypočítat translační a rotační matice.



obr 4.4: Návrh podložky

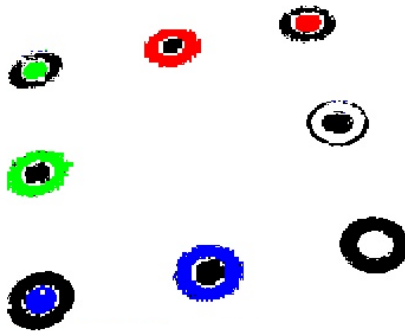
4.2.1 Návrh podložky

Podložka musí umožňovat určení pozice i v případě když by byla část podložky zastíněna snímaným objektem. Měla by obsahovat několik značek, každá značka musí být snadno a jednoznačně identifikovatelná. Značka na podložce by měla mít jednoduchý geometrický tvar, např. kružnice.

Podložka na obrázku obr 4.4 umožňuje přesně identifikovat každou značku na základě jedinečné barevné kombinace. Vzdálenosti mezi sousedními značkami jsou stejné.

4.2.2 Detekce značek

Nejprve se provede filtrace barev v RGB (viz kapitola 4.4.2), obrázek 4.5. Tento postup rychle vyfiltruje černou, modrou, červenou a zelenou barvu.



obr 4.5: Filtrace barev v RGB

Pro samotnou detekci značek je vhodné použít kombinaci různých metod. Například metody aktivních kontur a detekce pomocí Houghovy transformace. První metodou lze získat hranice objektů a vhodně nastavenými prahy pak vybrat ty které tvarem odpovídají značkám na podložce. Pro tento výběr se používají momentové charakteristiky. Druhou metodou je detekce elips pomocí Houghovy transformace.

4.2.3 Identifikace značek

Při identifikaci značky se zjistí barva v jejím středu. Pokud je takto zjištěna červená, zelená, modrá nebo bílá barva lze již tuto značku přesně identifikovat. Pokud je zjištěna barva černá je nutné prozkoumat okolní body a zjistit barvu prvního nečerného pixelu. Po určení barvy prvního nečerného pixelu, již lze i tuto značku identifikovat. Při zjišťování barvy pixelu je nutné zjišťovat i barvy sousedních pixelů a hodnotu zprůměrovat, aby se zabránilo chybám.

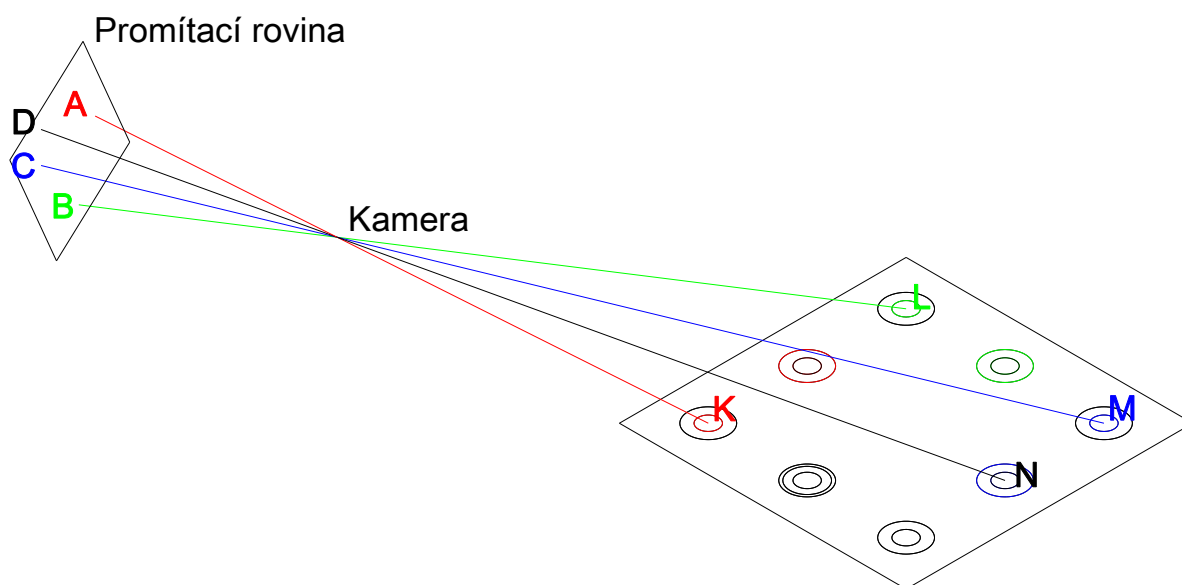
Určení barvy se provádí na základě rozdílnosti barevných složek, tento přístup poskytuje poměrně velkou rychlost, protože k určení barvy se používají přímo barvené složky RGB. Například červená barva má výrazně vyšší hodnotu R a hodnoty G a B jsou nižší. Při návrhu podložky jsou použité pouze barvy červená, zelená, modrá, bílá a černá. Tyto barvy mají barevné složky mezi sebou velmi rozdílné a lze je tak poměrně snadno popsat.

4.3 Výpočet pozice kamery

Počátek souřadného systému se položí na pozici kamery, osy systému budou ortogonální s pohledem kamery. Nejprve se vypočítá obecná rovnice podložky z pohledu kamery a následně se souřadný systém převede tak, aby nový střed souřadného systému byl ve středu podložky a osy souřadného systému byly ortogonální s podložkou.

Pro výpočet pozice kamery je nutné detekovat alespoň 4 značky na podložce. Kamera bude v počátku souřadného systému. Body na promítací rovině mají souřadnici Z rovnu ohniskové vzdálenosti a pozice pixelu na promítací rovině je vyjádřena souřadnicemi X a Y . Jsou tedy známy dva body potřebné pro výpočet parametrické rovnice přímky procházející počátkem souřadného systému a pixelem na promítací rovině.

Ohniskovou vzdálenost po kalibraci známe v pixelech, souřadnice bodu na promítací rovině jsou rovněž v pixelech. Rovnice parametrické rovnice se tedy vypočítají ze souřadnic, jejichž jednotkou je pixel, ale dále bude již možné počítat v milimetrech, protože pomocí pixelů se určí pouze směrnice přímek a u směrníc nezáleží na jednotkách.



obr 4.6: Výpočet pozice kamery

Vzdálenosti značek mezi sebou na reálné podložce jsou známy, díky tomu lze parametrem jedné parametrické rovnice přímky určit polohu roviny, ve které leží podložka. Parametr je třeba zvolit tak, aby vzdálenosti značek na vypočtené rovině co nejpřesněji odpovídaly reálným vzdálenostem na podložce. Hodnota parametru se postupně mění od nuly do jedné s daným krokem.

Protože v počátku souřadné systému je kamera je třeba je převést stávající souřadný systém do systému nového. Počátek nového souřadného systému se položí do středu podložky, osy souřadného systému budou ortogonální s rovinou podložky. Kamera bude mít vzhledem k středu podložky souřadnice:

$$\begin{aligned}
 Kamera_x &= x - S_x = a \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{iX} + B_{(i+4)X}}{2}}{4} \\
 Kamera_y &= y - S_y = b \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{iY} + B_{(i+4)Y}}{2}}{4} \\
 Kamera_z &= z - S_z = c \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{iZ} + B_{(i+4)Z}}{2}}{4}
 \end{aligned} \tag{5}$$

Podrobné odvození vzorců pro výpočet pozice kamery je uveden v příloze 2.

4.4 Detekce laserového paprsku

Při detekci laserového paprsku dopadajícího na povrch skenovaného objektu, záleží na povrchu objektu. Lesklé povrchy téměř nelze triangulačním skenerem s laserem snímat, neboť laserový paprsek je příliš odražen a rozptýlen a není tak možné ho detekovat.

4.4.1 Gaussovo normální rozložení

První možností je filtrace pomocí Gaussova normálního rozložení. Oblasti (shluky) se získají shlukováním na 2D histogramu vytvořeném ze snímku s laserovým paprskem v barevném modelu HSL nebo HSV. Tyto barevné modely umožňují vhodnější popis barev než model RGB. Histogram se vytvoří ze dvou složek například H a L. Pro shlukování se použije algoritmus k-means, jehož vstupem je histogram a výstupem je jeho rozdělení na oblasti. Každá oblast pak popisuje jeden barevný odstín. Na obrázku 4.7 jsou zobrazeny barevné odstíny (střední hodnoty) oblastí vzniklých po aplikaci algoritmu k-means na histogram.

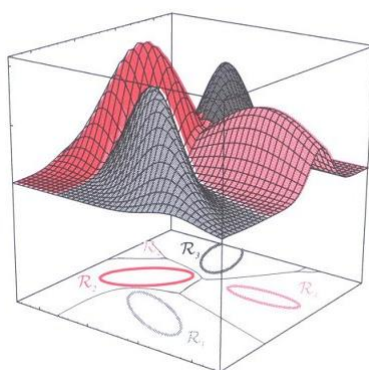


obr 4.7: Odstíny vzniklé aplikací k-means na histogram

Výpočet střední hodnoty a kovariační matice

Vstupem shlukové analýzy jsou shluky bodů. Každý bod ve shluku má souřadnice definované podle zvoleného barveného modelu HLS resp. HSV. Souřadnice bodů tak vytvoří matici, která bude mít dva nebo tři řádky podle toho v kolika rozměrech počítáme. Z této matice se spočítá vektor středních hodnot, který bude mít opět dva nebo tři řádky a stejně řádků bude mít i vypočtená čtvercová kovariační matice.

Výsledkem shlukové analýzy bude pro každý shluk vypočtená střední hodnota a kovariační matice, které budou tuto oblast popisovat. Shluky budou mít eliptický tvar. Na obrázku 3D histogramu (obr 4.8) je vidět jak se pro jednotlivé shluky (kopečky) po shlukové analýze vytvoří hranice, které jsou znázorněny na podstavě.



obr 4.8: 3D histogram a hranice oblastí na podstavě [2]

Pro výpočet pravděpodobnosti lze využít pravděpodobnostní funkce Gaussova normálního rozložení. Parametry této funkce jsou vektor středních hodnot, který má 2 nebo 3 řádky, a čtvercová kovariační matice, která má stejně řádků jako vektor středních hodnot. Proměnnými v pravděpodobnostní funkci jsou vektory souřadnic v barevném modelu HLS nebo HSV. Výsledkem pravděpodobnostní funkce je pravděpodobnost od nuly do jedné, s jakou testovaný pixel leží v jisté oblasti.

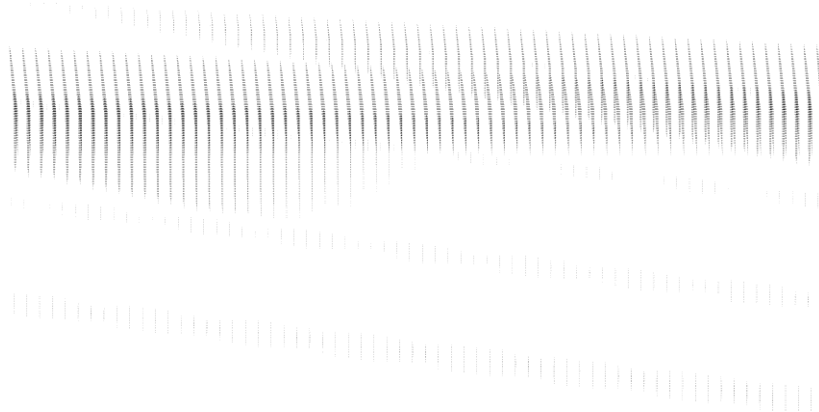
Pravděpodobnostní funkce

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \cdot \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \cdot \Sigma^{-1} \cdot (\mathbf{x} - \boldsymbol{\mu})\right) \quad (6)$$

- \mathbf{x} d-rozměrný sloupcový vektor
- $\boldsymbol{\mu}$ d-rozměrný vektor středních hodnot
- Σ kovarianční matice velikosti $d \times d$
- $|\Sigma|$ determinant kovarianční matice
- Σ^{-1} inverzní matice ke kovarianční matici

Filtrační maska

Z několika Gaussových rozložení popsaných vektorem středních hodnot a kovariační maticí lze vytvořit bitovou filtrační masku, která pro každý možný pixel v barevném modelu RGB určí, zda leží nebo neleží (s jistou předem danou pravděpodobností) v oblasti popsané Gaussovým rozložením. Filtrační maska je trojrozměrné bitové pole, které pro každou hodnotu indexů R,G a B vrátí hodnotu true nebo false, podle toho zda leží či neleží v oblasti popsané Gaussovým rozložením.



obr 4.9: Filtrační maska

Tímto způsobem lze vytvořit sadu těchto masek, pro různé osvětlení a pro různé skenované materiály. V barevném modelu RGB má každá složka rozsah 0-255, tzn. velikost masky bude $256^3 = 16777216$ bitů = 2 MB. Ukázka filtrační masky jen na obrázku 4.9. Filtrační maska umožní mnohem rychlejší detekci laserového paprsku, protože není nutný převod do jiných barevných modelů a není ani nutné počítat pravděpodobnostní funkci.

4.4.2 Detekce laseru v RGB

Předchozí metoda, pokud se nepoužívá filtrační maska, kde není zásah uživatele nutný, vyžaduje zásah uživatele, který musí manuálně vybrat barevné odstíny (shluky) podle kterých se má laser filtrovat.

Druhou možností, která nevyžaduje zásah uživatele je detekovat laserový paprsek prostým filtrováním v barevném modelu RGB, tak že se vhodně zvolí prahy jednotlivých složek a popíše jejich vzájemná rozdílnost. Například pokud se filtruje modrá barva, je zřejmé že modrá složka bude větší než složka zelená nebo červená, bez ohledu na její absolutní velikost.

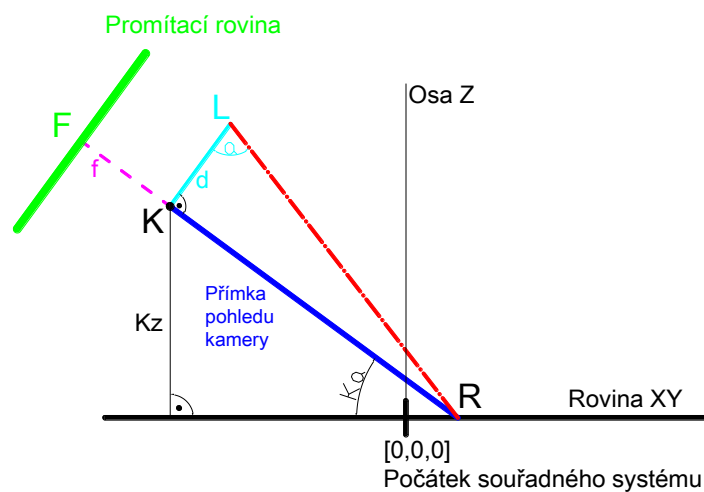
Tato varianta bude jistě rychlejší, protože pracuje přímo s barevnými složkami RGB a není nutný převod do jiných barevných modelů, je však nutné vhodně určit prahy. Výhodou pak je, že pokud bude filtrovaná barva popsána rozdílností složek, tak kvalita její filtrace by měla být invariantní vůči změně okolního osvětlení.

Po detekci se ještě použije morfologické uzavření (dilatace následovaná erozí) nebo mediánový filtr pro eliminaci případných šumů.

4.5 Triangulace – výpočet 3D souřadnic bodů

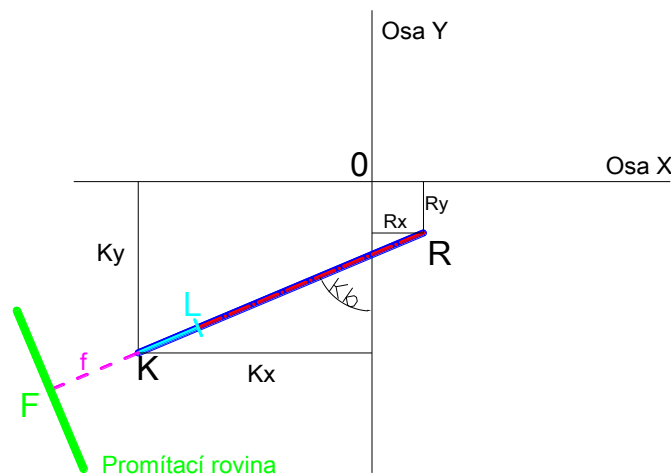
Počátek souřadného systému se položí do středu podložky (obr 4.10). Souřadnice kamery vůči počátku budou tedy $[Kx, Ky, Kz]$. Osy souřadného systému budou ortogonální s rovinou podložky. K_α je úhel, který udává sklonění kamery oproti rovině s podložkou. Hodnota K_α je v intervalu $(\pi/2, 0)$, při hodnotě $\pi/2$ se kamera „dívá“ kolmo na rovinu s podložkou. K_β je otočení kamery podle osy Z a K_γ je otočení kamery podle osy kolmé na promítací rovinu, procházející kamerou.

Další vstupní hodnoty jsou ohnisková vzdálenost f , sklon laseru vzhledem ke kameře α a vzdálenost laseru od kamery d (obr 4.10). Posledními vstupními hodnotami jsou souřadnice pixelů x a y a koeficienty k_x a k_y které udávají kolik pixelů je jeden milimetr.



obr 4.10: Triangulace - boční pohled.

Postup odvození: Cílem je najít obecnou rovnici, která popisuje rovinu čárového laserového paprsku, a parametrickou rovnici přímky procházející kamerou a pixelem na promítací rovině a nalézt jejich průsečík. Je třeba vypočítat 3D souřadnice pixelů na promítací rovině a najít jeden pomocný bod pro výpočet obecné rovnice laseru.



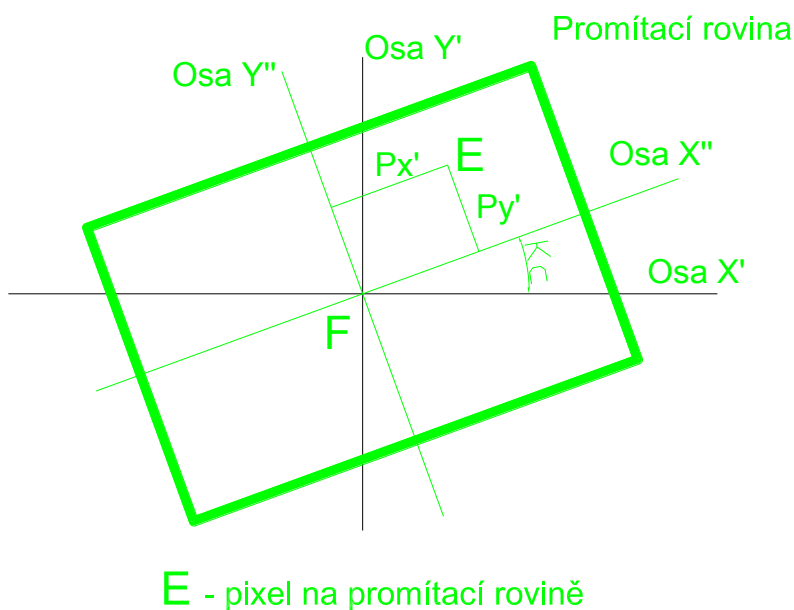
obr 4.11: Triangulace - pohled shora.

Výpočet parametrické rovnice pohledu kamery

- Vypočítat pomocný bod R , průsečík roviny ve které leží podložka a přímky pohledu kamery.
- Výpočet parametrické rovnice přímky pohledu kamery.

3D souřadnice pixelu (bod E) na promítací rovině jsou:

$$\begin{aligned} E_X &= K_X + \sin K_\alpha \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \cdot \sin K_\beta + \cos K_\beta \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} + \cos K_\alpha \cdot \sin K_\beta \cdot f \\ E_Y &= K_Y + \sin K_\alpha \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \cdot \cos K_\beta + \sin K_\beta \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} + \cos K_\alpha \cdot \cos K_\beta \cdot f \\ E_Z &= K_Z - f \sin K_\alpha + \cos K_\alpha \cdot \cos(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \end{aligned} \quad (7)$$



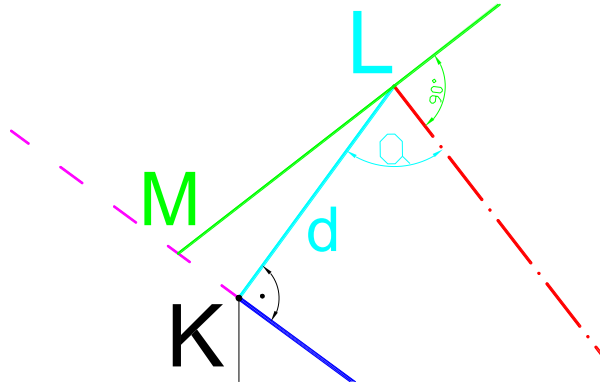
obr 4.12: Triangulace - pohled na promítací rovinu.

Výpočet 3D souřadnice zdroje laserové paprsku, bod L .

$$\begin{aligned} L_X &= K_X + \sin K_\alpha \cdot \cos K_\gamma \cdot d \cdot \sin K_\beta + \cos K_\beta \cdot \sin K_\gamma \cdot d \\ L_Y &= K_Y + \sin K_\alpha \cdot \cos K_\gamma \cdot d \cdot \cos K_\beta + \sin K_\beta \cdot \sin K_\gamma \cdot d \\ L_Z &= K_Z + \cos K_\alpha \cdot \cos K_\gamma \cdot d \end{aligned} \quad (8)$$

Výpočet pomocného bodu M . Pro vyjádření obecné rovnice roviny laseru, je nutné znát 3 body na rovině nebo je nutné vypočítat parametrickou rovnici přímky kolmé na rovinu,

$$\begin{aligned} M_X &= K_X - \cos K_\alpha \cdot \frac{d}{\tan \alpha} \cdot \sin K_\beta \\ M_Y &= K_Y - \cos K_\alpha \cdot \frac{d}{\tan \alpha} \cdot \cos K_\beta \\ M_Z &= K_Z + \sin K_\alpha \cdot \frac{d}{\tan \alpha} \end{aligned} \quad (9)$$



obr 4.13: Triangulace - poloha pomocného bodu M.

Pro výpočet obecné rovnice laseru se použijí body L a M jejichž pomocí se vypočítá směrnice vektor přímky kolmé na hledanou rovinu a protože směrnice vektor přímky kolmé na obecnou rovinu je stejný jako normálový vektor této roviny, najdeme tak obecnou rovnici roviny laseru:

$$(L_X - M_X) \cdot x + (L_Y - M_Y) \cdot y + (L_Z - M_Z) \cdot z - [(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z] = 0 \quad (10)$$

Parametr t lze získat dosazením parametrické rovnice přímky procházející bodem K a pixelem E (7) do obecné rovnice laseru (10).

$$t = \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_X - E_X) + (L_Z - M_Z) \cdot (K_X - E_X)} \quad (11)$$

Dosazením vypočteného parametru t do parametrické rovnice přímky procházející pixelem E a bodem K se vypočítají 3D souřadnice hledaného bodu na povrchu snímaného objektu:

$$\begin{aligned} x &= K_X + (K_X - E_X) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_X - E_X) + (L_Z - M_Z) \cdot (K_X - E_X)} \\ y &= K_Y + (K_Y - E_Y) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_X - E_X) + (L_Z - M_Z) \cdot (K_X - E_X)} \\ z &= K_Z + (K_Z - E_Z) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_X - E_X) + (L_Z - M_Z) \cdot (K_X - E_X)} \end{aligned} \quad (12)$$

Podrobné odvození vzorců pro triangulaci je uvedeno v příloze 3.

4.6 Výstup - VRML

Vypočtené 3D body je třeba vhodně vizualizovat. Asi nejvhodnější prostředkem je formát VRML, který umožňuje snadné vytvoření a následné prohlížení vytvořeného zrekonstruovaného objektu. Ve formátu VRML lze vypočtené body zadávat bod po bodu. Soubory ve formátu VRML lze snadno zobrazit u i ve standardních internetových prohlížečích [12].

5 Implementace

Pro implementaci jsem si vybral prostředí .NET 2005. Pro konzolovou aplikaci jsem zvolil programovací jazyk C++ s knihovnamí OpenCV a OpenGL. Aplikační rozhraní je vytvořeno v programovacím jazyce C# a je zde použito rozhraní MCI pro přehrávání video souborů.

5.1 Ruční 3D Skener

Ruční skener (obrázek 5.1) se skládá z webové kamery a čárového laseru umístěných na opačných koncích držáku. Webová kamera má jeden stupeň volnosti, je možné měnit náklon snímání. Čárový zdroj i kamera se připojuje pomocí USB rozhraní k počítači. Držák vyrobil a 3D skener sestavil Ing. Michal Španěl.

Pro správnou funkčnost kamery je nutné doinstalovat ovladače. Kamera funguje pod operačními systémy Windows XP i Windows Vista 32 i 64 bitů.

Parametry 3D Skeneru:

- Vzdálenost kamery a zdroje čárového laseru je 160 mm
- Pracovní rozsah: 0 až 2 metry od počítače, limitováno délkou kabelu
- Snímací rozsah: 0,2 až 1,5 metru
- Dva možné přístupy práce se skenerem: 1. Kamera nahoře, čárový laser dole (obrázek 5.1)
2. Čárový laser nahoře, kamera dole
- Cena: do 3 000,- Kč



obr 5.1: Ruční 3D skener

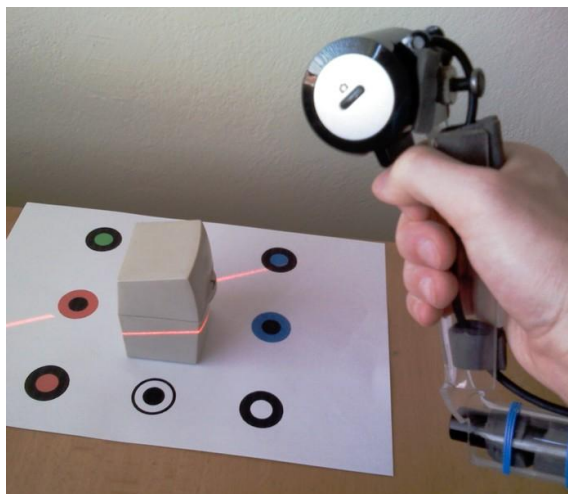
Philips SPC 900NC/00 PC kamera:

- Senzor: CCD
- Maximální rozlišení při snímání videa (neinterpolované): 640x480

- Maximální počet snímků za sekundu: 90
- Barevná hloubka: 24bit
- Rozhraní: USB 1.1
- Cena: ~ 1 400,- Kč

Čárový laser:

- Bezpečnostní třída laseru: IIIa (3R – značení dle IEC 60825 standardu)
- Výkon laseru: 3mW
- Vlnová délka: 635 nm (jasně červená barva)
- Odběr (maximální): 45 mA
- Napájení: 5V AC
- Cena: ~ 1 000,- Kč

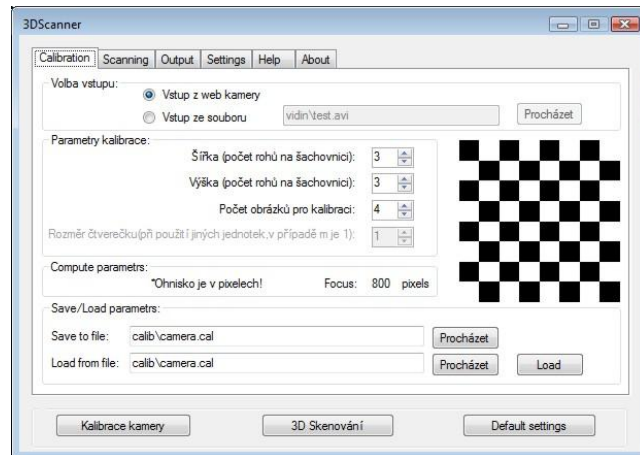


obr 5.2: Ukázka práce se skenerem

5.2 Aplikační rozhraní

Je vytvořeno v prostředí Visual studio.NET 2005 v programovacím jazyce C#. Pro spuštění je nutné mít nainstalován framework 2.0 nebo vyšší. Funkčnost aplikace byla testována na systémech Windows XP 32bit a Windows Vista 64bit. Při prvním spuštění je nutné v záložce settings stisknutím tlačítka „Spustit“ registrovat knihovny nutné k běhu konzolového programu.

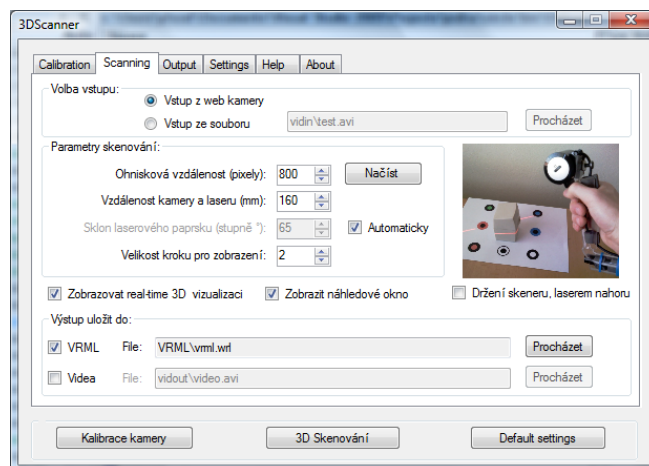
Aplikační rozhraní umožňuje nastavování parametrů pro konzolovou aplikaci. V první záložce lze nastavit parametry kalibrace kamery. Kalibraci lze provádět real-time pomocí připojeného 3D skeneru nebo načtením videa ze souboru. Parametry kalibrace jsou šířka a výška měřené v počtu rohů na šachovnici a počet obrázků, ze kterých se budou kalibrační data počítat. Minimální výška a šířka jsou 3 rohy, minimální počet snímků pro výpočet kalibračních dat je 4. Vypočtené parametry se ukládají do externího souboru. Lze také načíst již vypočtené parametry ze souboru.



obr 5.3: Záložka Kalibrace

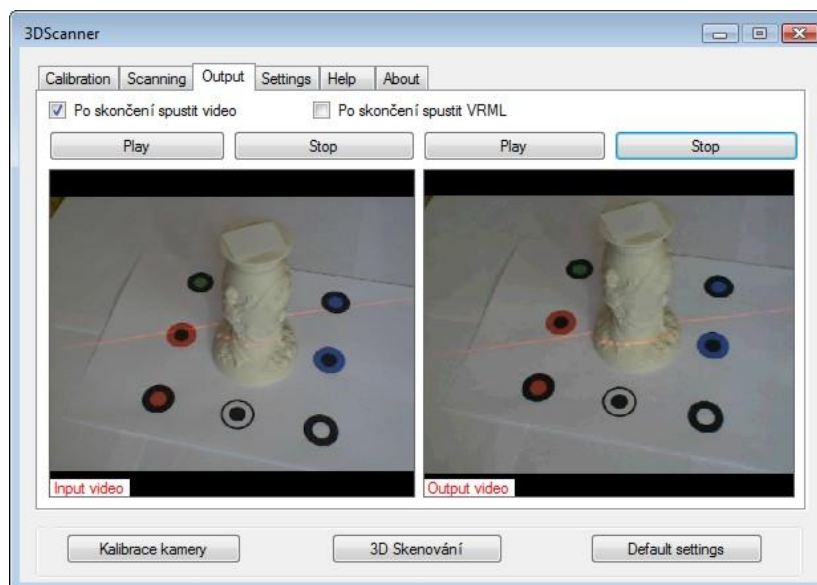
Druhá záložka umožňuje nastavení 3D skenování. Opět lze skenování provádět real-time pomocí 3D skeneru nebo z uloženého videa. Parametry skenování jsou ohnisková vzdálenost v pixelech (tuto vzdálenost lze načíst z kalibračního souboru), vzdálenost kamery a laseru v mm, sklon laserového paprsku ve stupních a velikost kroku pro zobrazení. Sklon laserového paprsku lze vypočítat automaticky, zaškrtnutím příslušného políčka. Při zapnutí automatického výpočtu je po spuštění skenování nejprve naskenovat prázdnou podložku a po stisku tlačítka „m“ začít skenovat.

Poslední parametr umožňuje měnit hustotu bodů ve výsledném 3D modelu. Dále je možné nastavit parametry zobrazení, kterými jsou real-time 3D vizualizace v OpenGL a zobrazení náhledového okna pro zobrazení jaké značky na podložce jsou detekovány a jak je detekován laserový paprsek. Nakonec je možné nastavit uložení výstup skenování. Je možné uložit výsledný 3D model ve formátu VRML a lze uložit video zaznamenané 3D skenerem při tvorbě tohoto modelu.



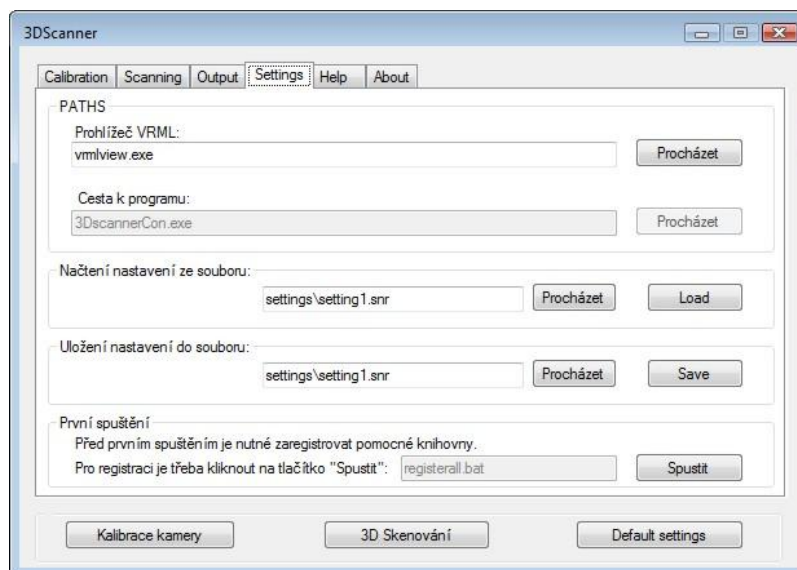
obr 5.4: Záložka Skenování

Záložka Output umožňuje přehrát video, které je na vstupu (v případě že není požitá možnost real-time skenování a je použito skenování ze souboru) a také lze přehrát video zaznamenané 3D skenerem při skenování. V této záložce lze také nastavit, aby se po dokončení skenování v externím prohlížeči VRML zobrazil výsledný 3D model. Pro přehrávání videí je použit Media Control Interface – MCI.



obr 5.5: Záložka Výstup

V záložce Settings je možné nastavit cestu k externímu prohlížeči VRML. Dále lze uložit nebo načíst nastavení skenování a kalibrace ze souboru. Důležitou položkou v této záložce je registrace knihoven nutných pro spuštění konzolové aplikace. Tato registrace se provádí pouze při prvním spuštění.



obr 5.6: Záložka Nastavení

5.3 Konzolová aplikace

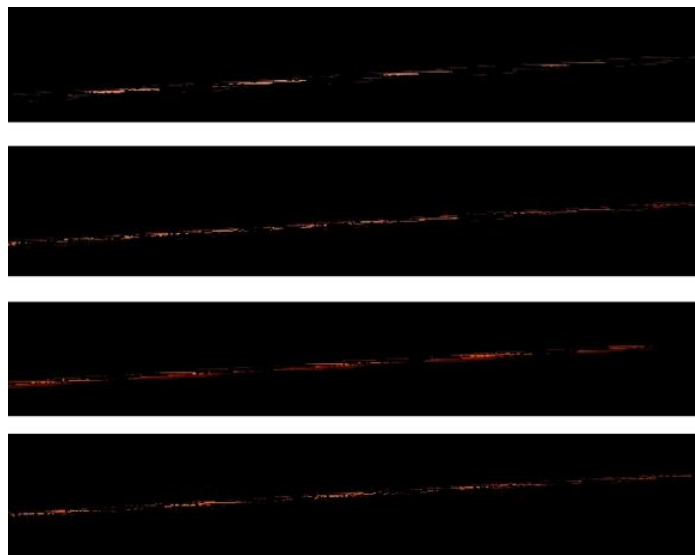
Konzolová aplikace provádějící samotné 3D skenování je vytvořena v prostředí Visual studio.NET 2005 v programovacím jazyce C++. Jsou použity knihovny OpenCV [14] a GLUT (The OpenGL Utility Toolkit) [22]. Funkčnost aplikace byla testována na systémech Windows XP 32bit a Windows Vista 64bit.

Aplikace je spuštěna na dvou vláknech. Na prvním vlákně se realizuje 3D rekonstrukce a na druhém OpenGL vizualizace. Aplikace postupně zpracovává snímek po snímku a detekuje nejprve značky na podložce, dále vypočítá pozici kamery, detekuje laserový paprsek, provede výslednou triangulaci a výsledky uloží do souboru. Před rekonstrukcí je třeba kalibrovat kameru a laser. Pro kalibraci kamery se využívají funkce z knihovny OpenCV.

Detekce značek probíhá pomocí dvou různých metod, složením výsledků těchto metod se určí poloha jednotlivých značek. První metodou je detekce kontur v obraze a následná selekce eliptických tvarů pomocí momentových charakteristik. Kontury a momentové charakteristiky jsou vypočítané pomocí funkcí z knihovny OpenCV. Druhá metoda pro detekci značek je detekce elips pomocí Houghovy transformace. Částečně byla použita implementace pro detekci kružnic z OpenCV, která byla následně výrazně upravena pro detekci elips a z knihovny OpenCV byla použita funkce `cvFitEllipse` pro výpočet délky hlavní a vedlejší osy elipsy.

Detekce laserového paprsku

Jsou implementovány dva způsoby detekce laserového paprsku. První způsob používá detekci pomocí Gaussových rozložení. Při klasifikaci se postupně berou pixely jeden za druhým zdrojového obrazu v barevném modelu HLS a počítají se jejich pravděpodobnosti, se kterou leží v jisté oblasti (např. oblasti popisující laserový paprsek).



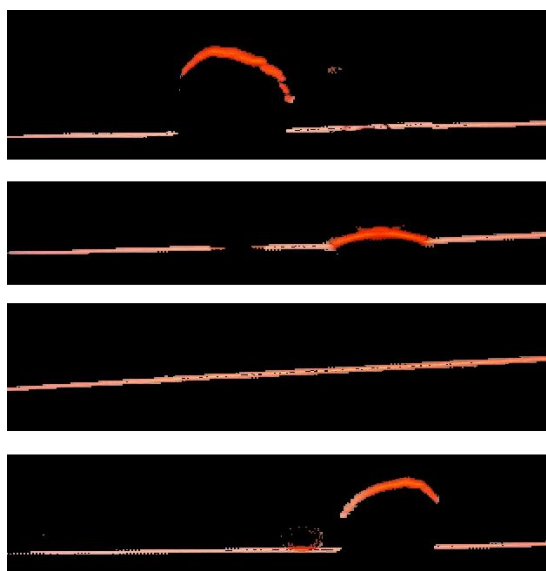
obr 5.7: Paprsek filtrovaný pomocí 4 různých Gaussových rozložení

Nejvyšší pravděpodobnost určí příslušnost k oblasti. Jedna oblast může být popsána i více Gaussovými rozloženími, výsledná pravděpodobnost se pak rovná součtu všech pravděpodobností jednotlivých vrstev. Na obrázku 5.7 je zobrazen laserový paprsek filtrovaný jednotlivě pomocí 4 Gaussových rozdělení. Na obrázku 5.8 je pak zobrazen filtrovaný paprsek popsáný více Gaussovými rozloženími.



obr 5.8: Filtrovaný paprsek popsaný více Gaussvými rozložení

Druhou možností je detekce paprsku pomocí RGB hodnot. Detekce laserového paprsku se provádí na základě rozdílu jednotlivých barevných složek. Pracuje se přímo s hodnotami R, G a B což umožňuje velmi rychlou Detekci. Ukázka filtrování laserového paprsku pomocí RGB hodnot je na obrázku 5.9.



obr 5.9: Detekce laseru v RGB

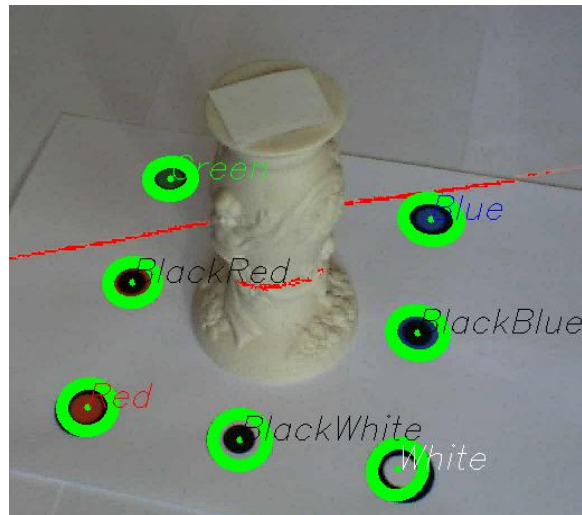
Z obrázků je zřejmé, že lepších výsledků dosahuje detekce přímo pomocí RGB hodnot, proto byla právě tato metoda použita v demonstrační aplikaci.

5.3.1 Struktura konzolové aplikace

Celý projekt 3DScanner se skládá z celkem 26 souborů. V každém souboru jsou seskupeny funkce podle funkčních celků, lze tak například použít funkce pro kalibraci samostatně bez použití ostatních funkcí. Hlavním souborem, který postupně zpracovává snímek za snímek a volá jednotlivé funkce, je `processvid.cpp`. Hlavním souborem pro kalibraci kamery je `calib.cpp`. Vizualizaci zajišťuje `3Dview.cpp`. Triangulaci `2Dto3D.cpp` a výpočet pozice kamery pak `campos.cpp`.

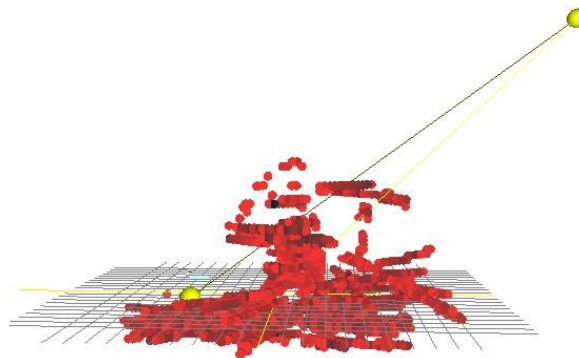
5.3.2 Vizualizace

Pro real-time vizualizaci je využita knihovna OpenGL. Vizualizace běží na jiném vláknu než samotná triangulace. Díky tomu lze i během výpočtu 3D rekonstrukce za pomoci myši plynule prohlížet výsledný 3D model. Na real-time vizualizaci se zobrazuje pozice kamery a zrekonstruovaný model. Během skenování také lze zobrazit okno s aktuálně detekovanými identifikovanými a identifikovanými značkami (obrázek 5.10).



obr 5.10: Aktuálně detekované a identifikované značky na podložce

Při real-time vizualizaci (obrázek 5.11) se polohy bodů na povrchu rekonstruovaného objektu kvantují, to umožňuje tmavším odstínem zobrazit polohy (stavy) do kterých se zaokrouhlí více pozic. Platí pravidlo čím je kvantovaný bod tmavší, tím více bodů reprezentuje a jeho poloha je tedy „jistější“. První žlutá kulička zobrazuje pozici kamery, druhá kulička ležící na rovině podložky zobrazuje kam se kamera „dívá“, tzn. průsečík osy pohledu s rovinou podložky.



obr 5.11: Real-time 3D vizualizace

5.4 Instalační program

Instalační program umožňuje instalaci a odinstalování programu 3DScanner, včetně všech potřebných knihoven. Instalační program je vytvořen v prostředí Visual studio.NET 2005. Instalaci lze provést standardně pomocí exe souboru nebo pomocí Windows installer-msi balíčku. Po instalaci se na plochu přidá ikonka programu. Odinstalování se provádí přes ovládací panely – přidat/odebrat programy.

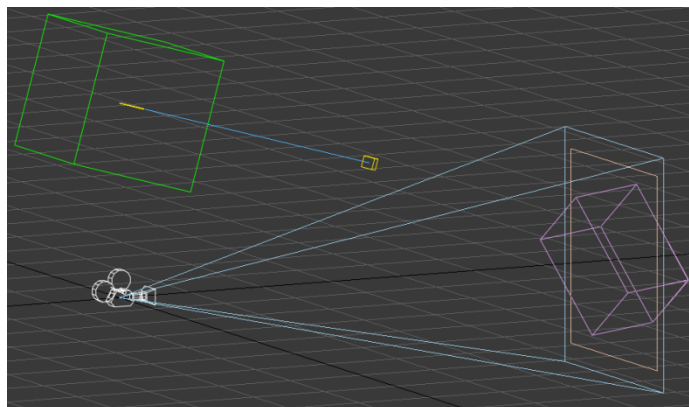


obr 5.12: Instalační program

6 Analýza výsledků

6.1 Triangulace – 3D rekonstrukce

Pro ověření funkčnosti algoritmu pro triangulaci bylo nutné vytvořit sadu syntetických dat. Syntetická data umožňují otestovat funkčnost samotného algoritmu 3D rekonstrukce neboť model se nejprve vytvoří ve vhodném nástroji, např. 3D studio MAX, takže pozice kamery i její natočení je známé. Ve výsledku se tedy neobjevují nepřesnosti vzniklé při výpočtu pozice kamery, odpadají rovněž problémy s detekcí laserového paprsku.



obr 6.1: Scéna ve 3D studiu MAX

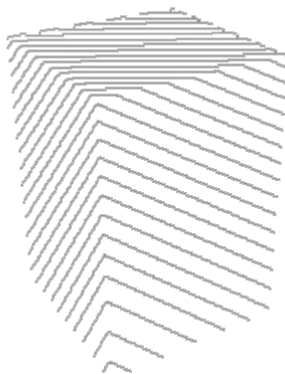
Na obrázku 6.1 je ukázka scény v 3D studiu MAX. Ve scéně je kvádr, na něhož se nechá dopadat laserový paprsek. Zelená krychle nad kamerou je modifikátor, který umožňuje měnit sklon laserového paprsku. Celou scénu pak snímá dírková kamera s definovanou ohniskovou vzdáleností. Jsou tak známé všechny parametry nutné pro triangulaci. Parametry jsou 6 stupňů volnosti kamery, vzdálenost zdroje laserového paprsku od kamery, sklon laserového paprsku a ohnisková vzdálenost kamery. Z takto připravené scény se vygeneruje série snímků, ve kterých se bude po jednom stupni měnit sklon laserového paprsku obrázek 6.2.



obr 6.2: Objekt, na který dopadá laserový paprsek.

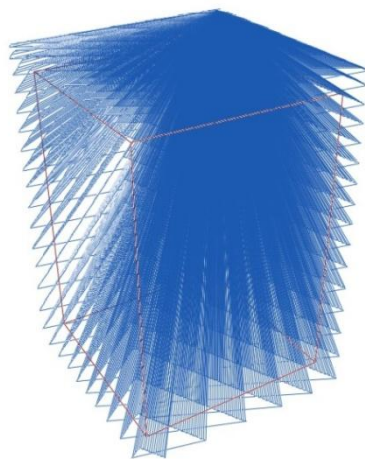
Protože všechny potřebné parametry jsou známy, berou se nyní postupně pixely laserového paprsku na promítací rovině a pro každý se provede přepočítání na 3D souřadnice podle vzorců uvedených v příloze 3.

Obrázek 6.3 zobrazuje vizualizaci rekonstruovaného objektu. Každá čára na povrchu objektu znamená jeden vstupní snímek s jiným sklonem laserového paprsku.



obr 6.3: Zrekonstruovaný objekt ve VRML

Na obrázku 6.4 je původní a zrekonstruovaný objekt. Červenou barvou je znázorněn originální (vzorový) kvádr a modrou barvou je zrekonstruovaný objekt. Jak je vidět originální kvádr je o něco menší než vypočtený objekt, ale hrany obou těles jsou rovnoběžné. Rozdíl ve velikosti je daný špatně nastaveným poměrem pro převod z pixelů na milimetry, tento problém paradoxně odpadá při reálném snímání 3D skenerem neboť tento poměr si lze vypočítat pomocí znalosti vzdálenosti reálných značek na podložce.



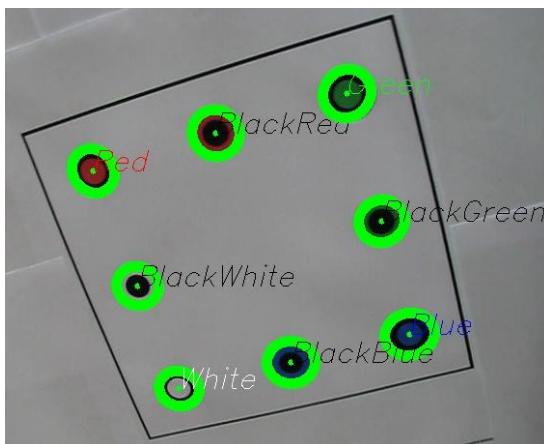
obr 6.4: Zrekonstruovaný objekt + originální model (červeně)

Z obrázku 6.4 lze tedy vyčíst, že algoritmus pro triangulaci, tak jak je odvozen v příloze 3. funguje a dokáže zrekonstruovat objekt s velkou přesností, která je v tomto případě omezena pouze schopností správně zadat poměr pro převod z pixelů na milimetry.

6.2 Výpočet pozice kamery

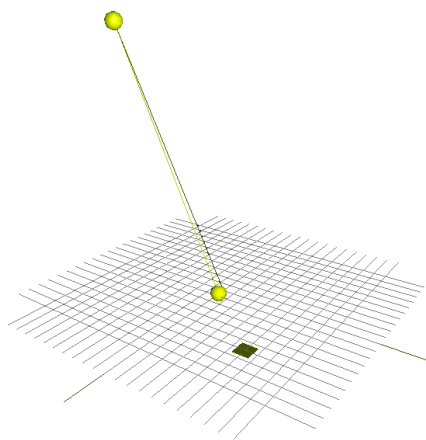
Pro výpočet pozice kamery je klíčová přesná detekce značek na podložce. Z těchto značek se pak výpočtem uvedeným v příloze 2 vypočítá pozice kamery vzhledem k podložce v milimetrech. Dále se vypočítají úhlu sklonu kamery a převodní poměr z pixelů na milimetry. Před výpočtem pozice kamery je třeba znát ohniskovou vzdálenost kamery, kterou lze zjistit kalibrací kamery.

Pro přesné určení polohy značek je použito dvou rozdílných metod, jsou jimi detekce aktivních kontur a Houghova transformace. Složením těchto metod lze docílit velké přesnosti při určování polohy. Obě metody jsou značně robustní a invariantní vůči změně osvětlení, což zajišťuje funkčnost i za horších podmínek.



obr 6.5: Detekce značek

Na obrázku 6.6 je žlutou kuličkou zobrazena pozice kamery. Druhá žlutá kulička ležící v rovině podložky znázorňuje místo tak se kamera „dívá“, průsečík osy pohledu s rovinou podložky. Čtvereček na podložce zobrazuje, kde se nachází značka „White“ z obrázku 6.5.



obr 6.6: Zobrazení pozice kamery

6.3 Přesnost rekonstrukce

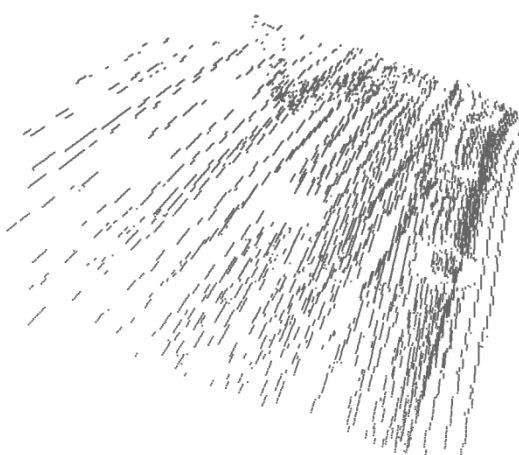
Přesnost rekonstrukce závisí především na kvalitním zjištění pozice kamery. Při rekonstrukci ze syntetických dat je možno dosáhnout takřka sto procentní přesnosti, protože všechny parametry jsou přesně známy.

Při výpočtu odchylky je možné pomocí ručního 3D skeneru nasnímat podložku na kterou dopadá laserový paprsek. Po rekonstrukci by měla vyjít série čar ležících v jedné rovině. Tyto čáry by měly mít vzdálenost od podložky (souřadnici Z) blízkou nule. Při tomto „pokusu“ se již projeví vliv přesnosti detekce značek a schopnosti detekovat laserový paprsek bez případných šumů.



obr 6.7: Skenování podložky

Na obrázku 6.7 je jeden snímek ze skenování podložky. Obrázek 6.8 ukazuje již zrekonstruovanou podložku. Na obrázku 6.8 jsou vidět „díry“ na podložce které jsou na místě značek.



obr 6.8: Zrekonstruovaná podložka

Nespojitost laserových paprsků u zrekonstruované podložky je způsobena diskretizací výsledných 3D souřadnic. Diskretizace se provádí z důvodu plynulosti real-time vizualizace. Pokud

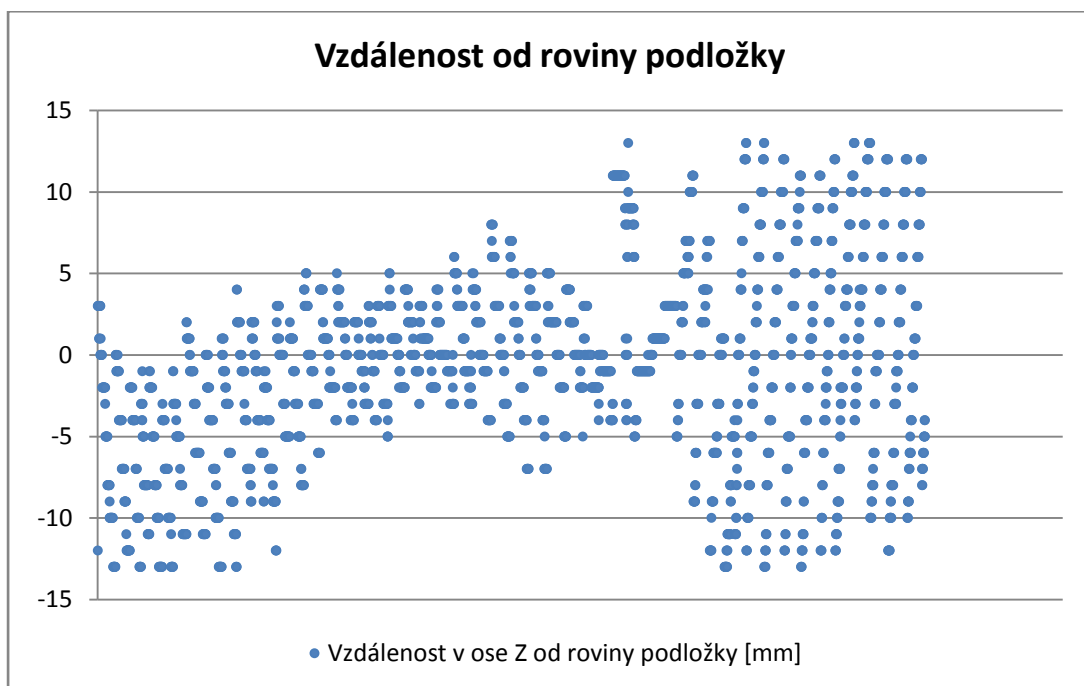
by se vizualizovaly všechny vypočtené body došlo by v důsledku obrovského množství bodů k velkému zpomalení real-time vizualizace a výsledný model uložený ve VRML by měl řádově několik megabajtů. Vhodným kvantováním poloh lze snížit tuto velikost na desítky kilobajtů a umožní také plynule zobrazovat 3D vizualizaci při skenování. Velikost kvantovacího kroku (rozlišení) lze nastavit v aplikačním rozhraní, minimální hodnota je jeden milimetr.

Průměrná výška bodů nad rovinou podložky je $-0,89 \text{ mm}$. Průměrná odchylka je $4,33 \text{ mm}$, tato hodnota udává přesnost algoritmu 3D rekonstrukce. Průměrná odchylka je bohužel částečně zkreslena šumem, který má mnohem vyšší vzdálenost nad rovinou podložky.

Průměr [mm]	Odchylka [mm]	Medián [mm]
-0,86	4,33	-1

Tabulka vypočtených hodnot

Na grafu 6.9 jsou zobrazeny výšky bodů nad podložkou. Z důvodu názornosti byla vybrána pouze část dat z obrázku 6.8.



obr 6.9: Graf výšek bodů nad rovinou podložky

6.4 Rychlost rekonstrukce

Rychlost rekonstrukce závisí na použitém rozlišení obrazu, který se bude zpracovávat. V tabulce jsou naměřené hodnoty dosažené při použití vstupních snímků s rozlišením 640x480.

Délka videa [s]	Snímků za sekundu	Snímků celkem	Čas zpracování [s]	Zpracovaných snímků za sekundu
9	25	225	25	9
10	30	300	37	8,11
8	30	240	23	10,43
8	30	240	26	9,23
10	30	300	29	10,34
13	30	390	41	9,51
19	30	390	47	8,30

Průměr je 9,27 snímků za vteřinu. Ne pro všechny snímky videa se však počítá pozice kamery a provádí 3D rekonstrukce, tyto výpočty se provádí jen pro snímky, ve kterých se podařilo detekovat alespoň čtyři značky na podložce. Proto rychlost rekonstrukce nejpřesněji popíše nejpomalejší rychlost a ta je 8,3 snímků za vteřinu.

Testovací sestava:

- Procesor - Intel Core Duo CPU T7300 2.00GHz 64 bitů
- Paměť - RAM 2046 MB
- Grafická karta - NVIDIA GeForce 8600M GT 512 MB
- Operační systém – Windows Vista Business 64 bitů

Testování proběhlo na videu s rozlišením 640x480 se zapnutou real-time vizualizací v OpenGL.

U rekonstrukce, která probíhá real-time pomocí připojeného 3D skeneru se omezená rychlost zpracování snímků projeví tak, že rekonstrukce bude sice probíhat real-time, ale z datového proudu se budou snímky brát až v době, kdy se povede zpracovat snímek předchozí. To znamená, že některé snímky, které kamera nasnímá, se vůbec nepoužijí. Například pokud bude kamera snímat 30 snímků za vteřinu, použije se pouze každý třetí snímek a ostatní snímky zůstanou nevyužity.

7 Závěr

Cílem diplomové práce bylo navrhnout a implementovat algoritmus pro rekonstrukci objektů z obrazových dat. Zadání se mi povedlo splnit, navrhl jsem algoritmus a vytvořit program demonstrující jeho funkčnost. Pomocí vztahů analytické geometrie jsem odvodil vzorce pro výpočet pozice kamery a pro 3D rekonstrukci – triangulaci. Správnost těchto vzorců jsem ověřil analýzou výsledků. Přesnost rekonstrukce je $\pm 4,3$ mm.

Navržený algoritmus pro rekonstrukci umožňuje volný pohyb ručního 3D skeneru za současné změny sklonu laserového paprsku. Implementována je ale pouze funkce automatického výpočtu sklonu laserového paprsku před skenováním. Jsou dva možné přístupy práce se skenerem, kamerou nahoře a laserem dole nebo kamerou dole a laserem nahoře.

Pro demonstraci výsledků jsem vytvořil aplikaci s grafickým rozhraním a 3D vizualizací. Pomocí této aplikace lze provádět real-time snímání objektů. Před snímáním je nutné kalibrovat kameru a zjistit tak její ohniskovou vzdálenost. Aplikaci lze snadno nainstalovat a používat na platformách Windows XP i Windows Vista 32 i 64 bitů. Během snímání lze zobrazit 3D scénu v OpenGL, ve které se zobrazuje aktuální stav rekonstrukce a pozice kamery. Výsledný zrekonstruovaný model je možné uložit ve formátu VRML. Program dokáže při real-time snímání při rozlišení 640x480 zpracovat osm snímků za vteřinu.

Detekce laserového paprsku je optimalizována na skenování objektů, které mají matný bílý povrch. Se skenerem je třeba pracovat pomalu, při rychlejších pohybech je obraz snímaný kamerou rozmazán a nelze ho zpracovat.

Větší přesnosti by se dalo dosáhnout změnou vzhledu podložky, značky na podložce by nemusely být rozmístěny pravidelně, vzdálenosti mezi sousedními značkami by nemusely být stejné. Větší přesnost by přineslo i použití kamery s vyšším rozlišením, současná kamera má rozlišení 0,3 miliónů pixelů, profesionální skenery používají kamery s rozlišením o řád vyšším. Dalším vylepšením by bylo použití kvalitnějšího zdroje laserového paprsku, současný laserový zdroj má nestejnou tloušťku laserového paprsku, uprostřed je paprsek silnější než na okrajích. Současný laser má rovněž poměrně malý výkon, pouze 3mW.

Literatura

- [1] Žára, J., Beneš, B., Sochor, J. Moderní počítačová grafika, vydání druhé, Brno, Computer press, 2004, s. 155-157
- [2] Štanclová, J. AIL072 - Rozpoznávání vzorů, Matematicko-fyzikální fakulta UK v Praze, 27. prosince 2007. Dokument dostupný na URL http://kocour.ms.mff.cuni.cz/~stanclova/ail072/03%20-%20Diskriminant_funkce.pdf (prosinec 2007).
- [3] Řezníček, J. 3D laserové skenování, Fakulta stavební ČVUT, 27. prosince 2007. Dokument dostupný na URL <http://ifgm.fsv.cvut.cz/data/vvt/s1/laserteorie3d.pdf> (prosinec 2007).
- [4] Vraštil, Z. Barvy a barevné modely, 27. prosince 2007. Dokument dostupný na URL <http://web.quick.cz/vrastilz/CG/Colors.html> (prosinec 2007).
- [5] Mudrová, M. Barvy v počítačové grafice, Ústav počítačové a řídicí techniky Vysoké školy Chemicko-Technologické v Praze, 22. dubna 2006. Dokument dostupný na URL <http://uprt.vscht.cz/ucebnice/zob/prednasky/02-BARVY/Barvy.pdf> (duben 2006).
- [6] Hlaváč, V., Sedláček, M. Zpracování signálu a obrazu, Elektrotechnická fakulta ČVUT v Praze, 27. prosince 2007. Dokument dostupný na URL <http://neuron.tuke.sk/~pluchta/Pocitacove%20Videnie/Prednasky/NIECO/HLAZSO.PDF> (prosinec 2007).
- [7] Wikipedia - 3D scanner, 27. prosince 2006. Dokument dostupný na URL http://en.wikipedia.org/wiki/3D_scanner (prosinec 2007).
- [8] Kelbel, J., Šilhán, D. Shluková analýza, Elektrotechnická fakulta ČVUT v Praze, 27. prosince 2007. Dokument dostupný na URL <http://gerstner.felk.cvut.cz/biolab/X33BMI/slides/KMeans.pdf> (prosinec 2007).
- [9] Matoušek, Z. Shluková analýza, Fakulta informatiky MU v Brně, 22. dubna 2006. Dokument dostupný na URL http://nlp.fi.muni.cz/projekty/podobnost_dokumentu/node19.html (duben 2006).
- [10] Circle Hough Transform, 11. duben 2008. Dokument dostupný na URL <http://www.cis.rit.edu/class/simg782.old/talkHough/HoughLecCircles.html> (duben 2008).
- [11] Kršek, P. Barevný obraz a barevné modely, Fakulta informačních technologií VUTBR, 27. prosince 2007. Dokument dostupný na URL https://www.fit.vutbr.cz/study/courses/IZG/private/lecture03/izg_barev_modely.pdf (prosinec 2007).
- [12] VRML Wiki, 11. dubna 2008. Dokument dostupný na URL <http://en.wikipedia.org/wiki/VRML> (duben 2008).

- [13] Active Contouring (Snakes) in IDL, 17. duben 2008. Dokument dostupný na URL http://www.dfanning.com/ip_tips/snakes.html (duben 2008).
- [14] OpenCV Wiki, 27. prosince 2007. Dokument dostupný na URL <http://opencvlibrary.sourceforge.net> (prosinec 2007).
- [15] Španěl, M. Obrazové segmentační techniky, 11. duben 2008. Dokument dostupný na URL <http://www.fit.vutbr.cz/~spanel/segmentace/.cs.iso-8859-2> (duben 2008).
- [16] Kybic, J. Active Contours — Snakes, 11. duben 2008. Dokument dostupný na URL <http://cmp.felk.cvut.cz/cmp/courses/DZO/slidy/snakes.pdf> (duben 2008).
- [17] Zemčík, P. Detekce geometrických tvarů, 11. duben 2008. Dokument dostupný na URL https://www.fit.vutbr.cz/study/courses/POV/private/lectures/POV_HT_objekty.pdf (duben 2008).
- [18] Faro - Laser ScanArm V3, 11. duben 2008. Dokument dostupný na URL <http://www.faro.com/content.aspx?ct=di&content=pro&item=1&subitem=58> (duben 2008).
- [19] Handyscan - přenosný ruční 3D scanner, 11. duben 2008. Dokument dostupný na URL <http://www.handyscan.cz> (duben 2008).
- [20] Nextengine, 11. duben 2008. Dokument dostupný na URL www.nextengine.com (duben 2008).
- [21] Optické 3D skenery - Atos, 11. duben 2008. Dokument dostupný na URL <http://robo.hyperlink.cz/3dskenery/main05.html> (duben 2008).
- [22] GLUT - The OpenGL Utility Toolkit, 19. dubna 2008. Dokument dostupný na URL <http://www.opengl.org/resources/libraries/glut/> (duben 2008).

Seznam příloh

Příloha 1. Ukázky výsledků

Příloha 2. Odvození rovnic pro výpočet pozice kamery

Příloha 3. Odvození rovnic pro 3D rekonstrukci

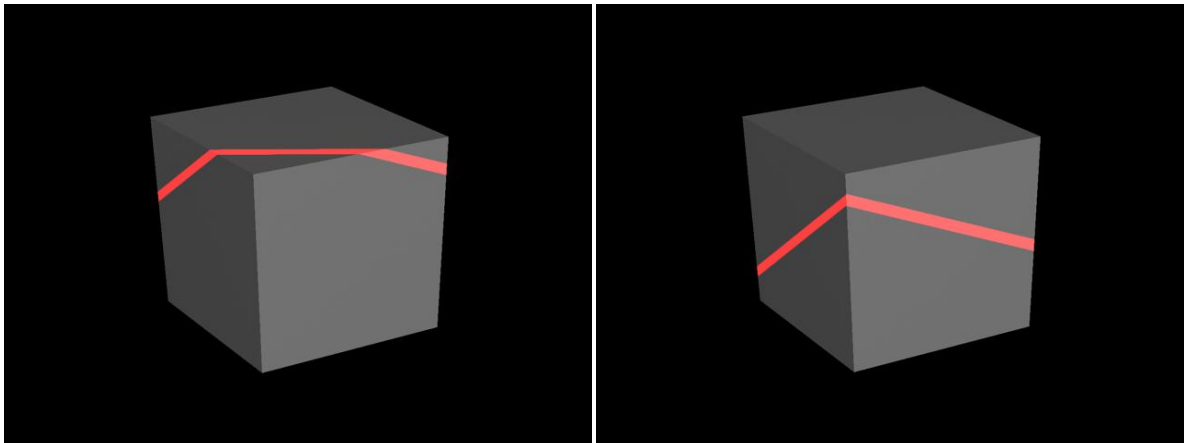
Příloha 4. Uživatelská příručka

Příloha 5. Podložka (skutečná velikost)

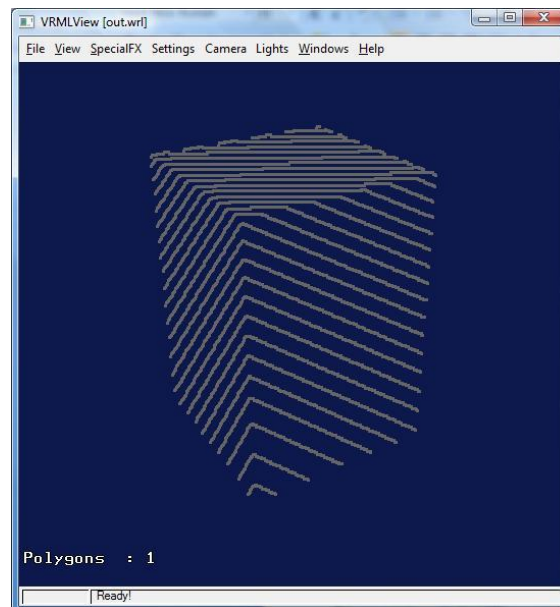
Příloha 6. Kalibrační šachovnice

Příloha 7. DVD

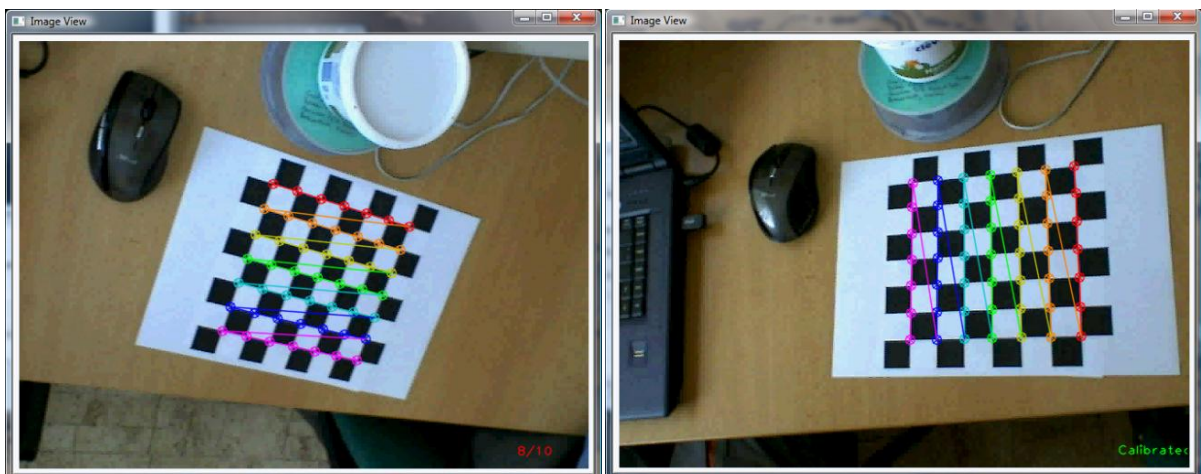
Příloha 1. Ukázky výsledků



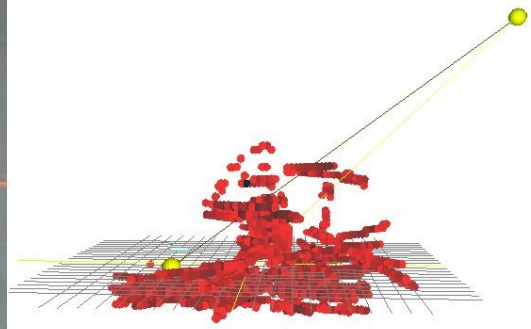
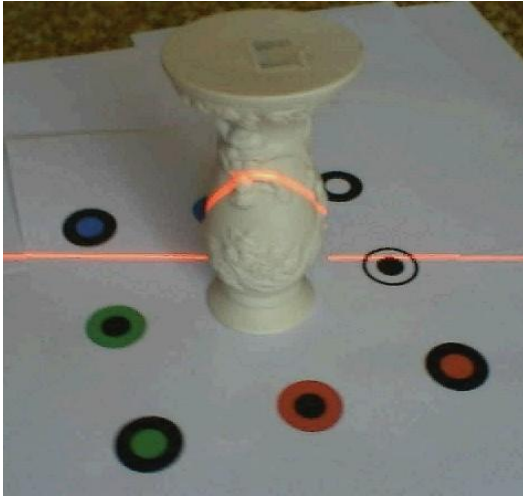
obr P1.1: Vygenerované obrázky s laserovým paprskem.



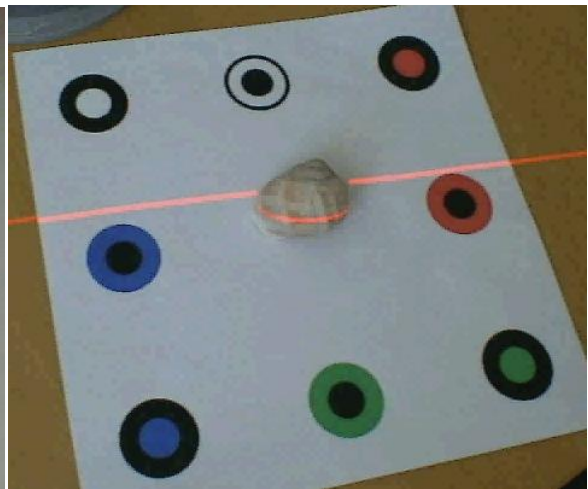
obr P1.2: Zrekonstruovaný objekt.



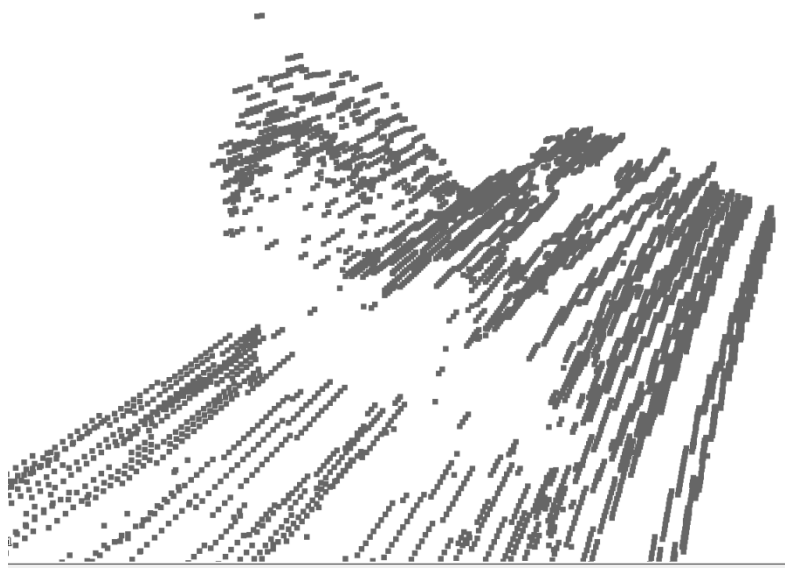
obr P1.3: Kalibrace kamery



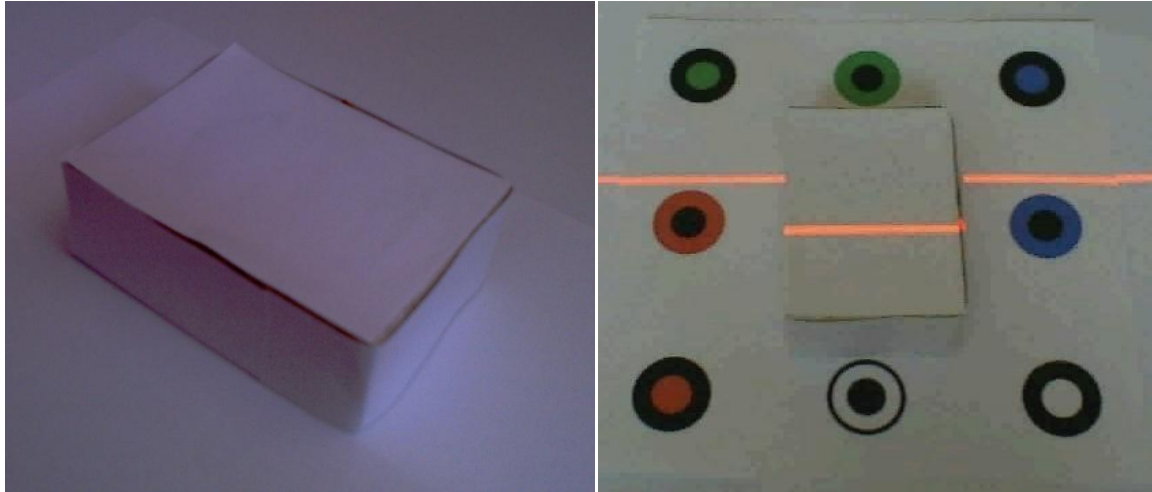
obr P1.4: Vlevo snímek z videa při skenování, vpravo real-time 3D rekonstrukce v OpenGL



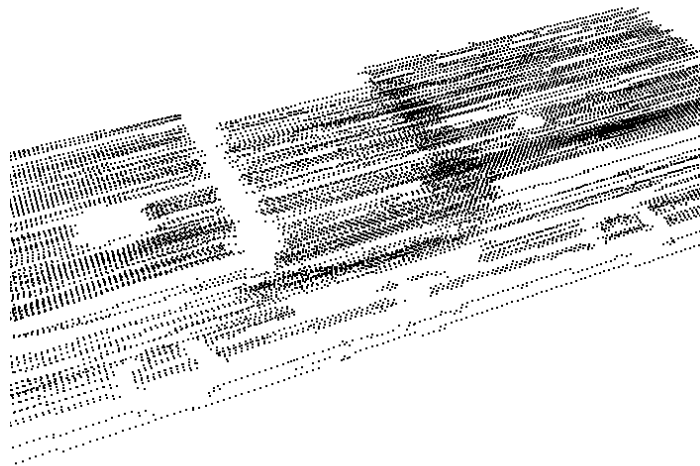
obr P1.5: Ulita



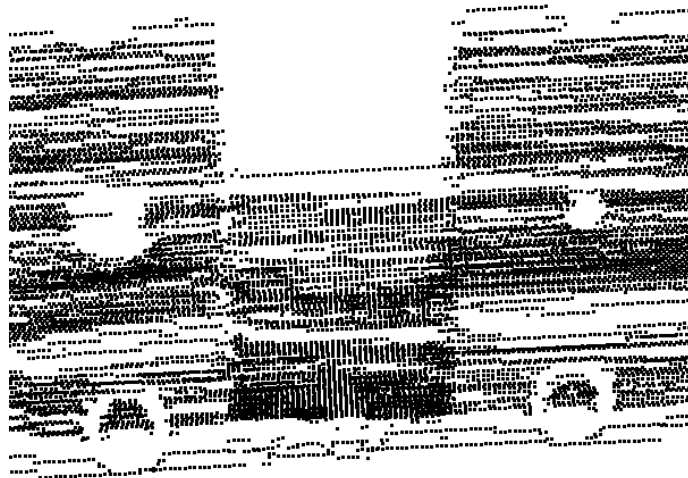
obr P1.6: Rekonstrukce ulity - VRML



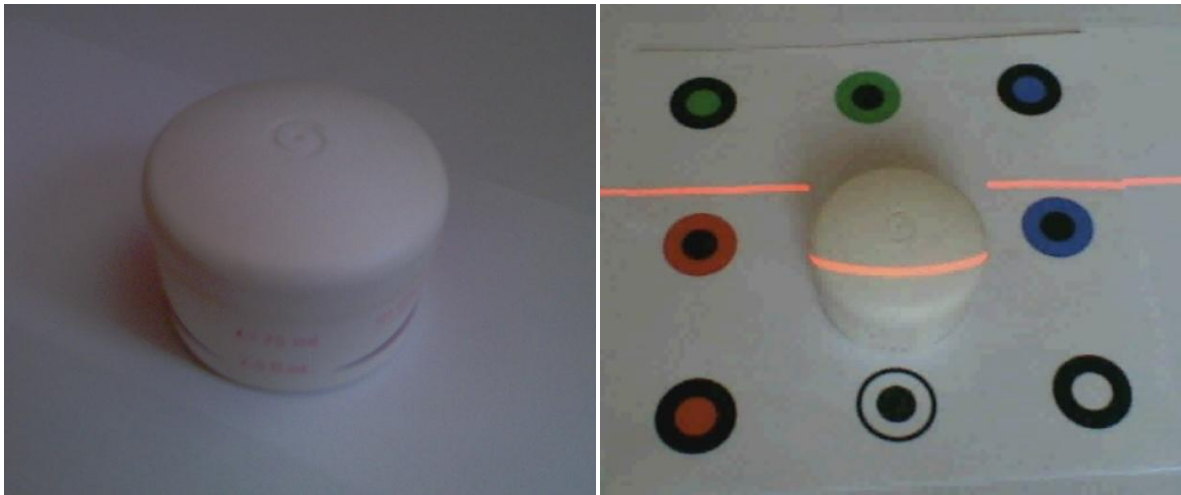
obr P1.7: Krabička A



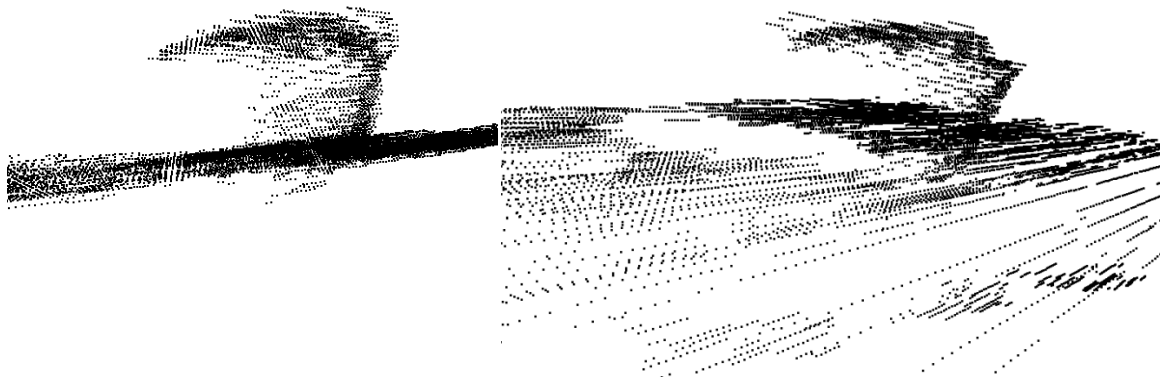
obr P1.8: Rekonstrukce krabičky A - VRML



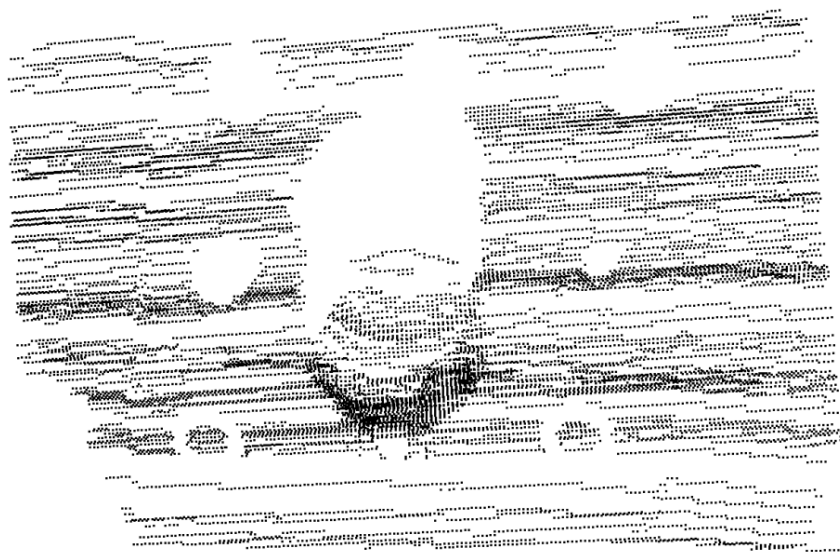
obr P1.9: Rekonstrukce krabičky A - VRML



obr P1.10: Krabička B

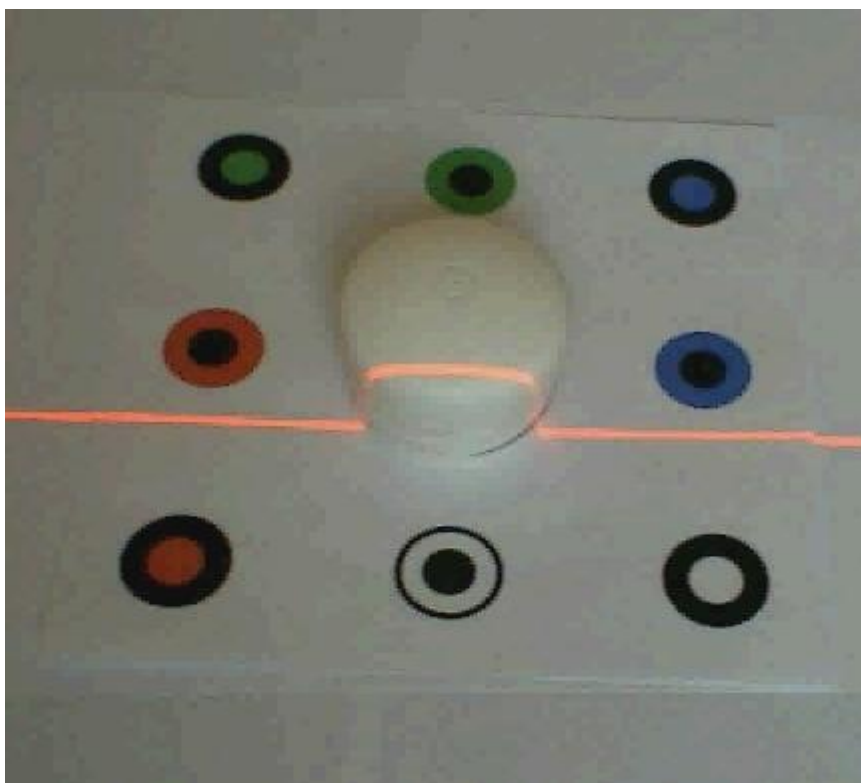


obr P1.11: Rekonstrukce krabičky B - VRML

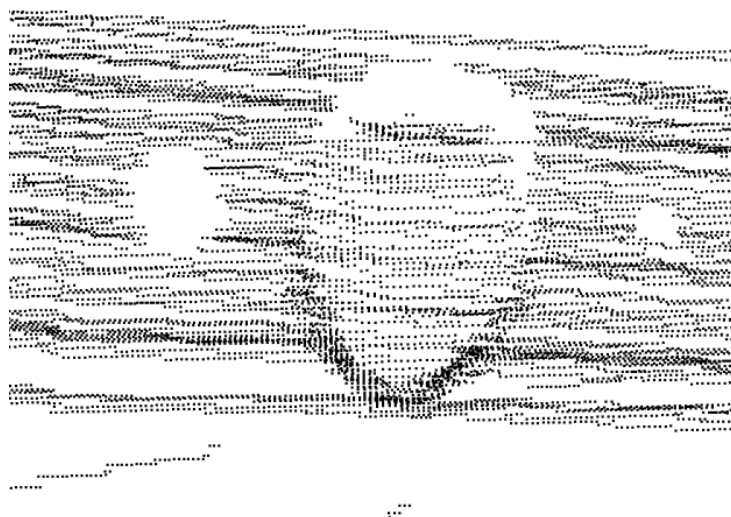


obr P1.12: Rekonstrukce krabičky B - VRML

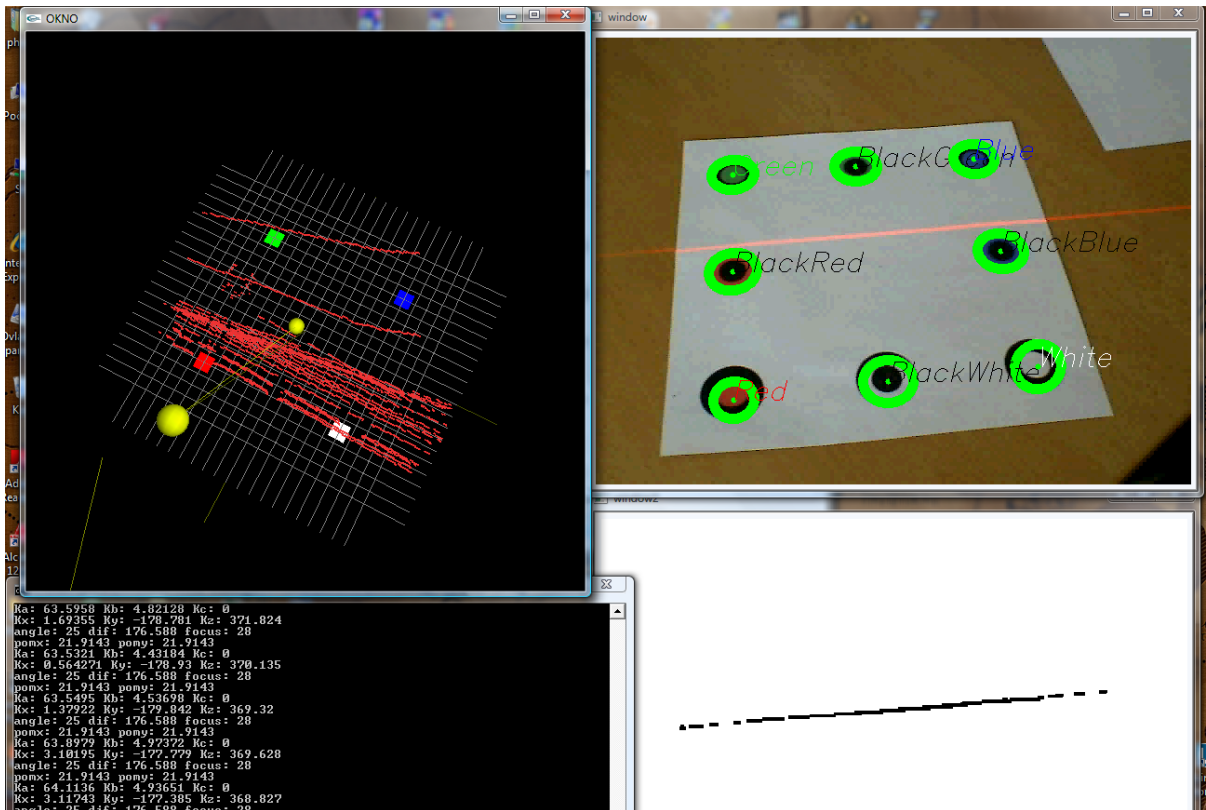
Na obrázku P1.13 je díky opačnému držení 3D skeneru (laserem nahoru) patrný jiný sklon laseru než na obrázku P1.10, to se ve výsledku promítne i při triangulaci na zrekonstruovaném objektu. Za objektem na obr. P1.14 je oproti objektu na obrázku P1.12 kratší „stín“.



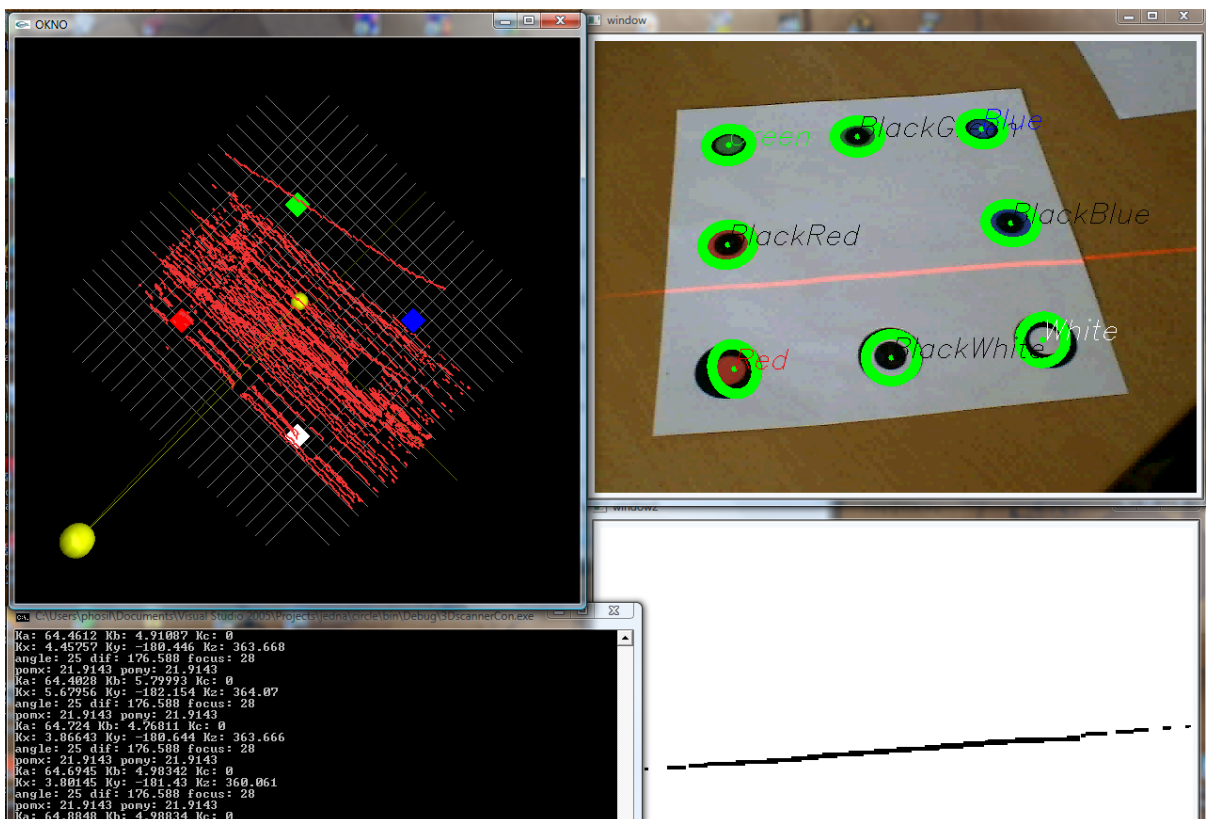
obr P1.13: Krabička B – obrácené držení skeneru



obr P1.14: Rekonstrukce krabičky B – VRML - obrácené držení skeneru



obr P1.15a: Ukázka činnosti programu



obr P1.15b: Ukázka činnosti programu

Příloha 2. Odvození rovnic pro výpočet pozice kamery

Polohu značky na reálné podložce lze vyjádřit pomocí parametrické rovnice přímky procházející počátkem a značkou A na promítací rovině. Poloha reálné značky K tak bude vyjádřena:

$$\begin{aligned}K_X &= x_A \cdot t_A \\K_Y &= y_A \cdot t_A \\K_Z &= f \cdot t_A\end{aligned}\tag{13}$$

Kde x_A a y_A je poloha značky A na promítací rovině, f je ohnisková vzdálenost a t_A je koeficientem parametrické rovnice. Jednotkou x_A , y_A a f jsou pixely. Obdobně lze vyjádřit polohy ostatních značek. Zároveň jsou ale známé vzdálenosti značek na reálné podložce, lze tak vyjádřit polohu všech ostatních značek na reálné podložce v závislosti na pozici značky K . Nechť $|KL|$ je reálná vzdálenost bodů K a L , potom platí kvadratická rovnice:

$$(x_B^2 + y_B^2 + f^2) \cdot t_B^2 - 2 \cdot (x_A x_B + y_A y_B + f^2) \cdot t_A t_B + (x_A^2 + y_A^2 + f^2) \cdot t_A^2 = |KL|^2 \tag{14}$$

Proměnné x_B , y_B a t_B jsou koeficienty parametrické rovnice přímky, na které leží bod L podobně jako je to při výpočtu polohy bodu K v rovnici (13). Parametr t_B lze tedy vyjádřit v závislosti na t_A . Stejně lze v závislosti na parametru t_A vyjádřit polohu dalších značek. To umožňuje volbou jednoho parametru určovat polohy dalších bodů, z těchto bodů lze již snadno vypočítat obecnou rovnici roviny. Parametry t_C a t_D lze v závislosti na t_A vyjádřit takto:

$$\begin{aligned}(x_C^2 + y_C^2 + f^2) \cdot t_C^2 - 2 \cdot (x_A x_C + y_A y_C + f^2) \cdot t_A t_C + (x_A^2 + y_A^2 + f^2) \cdot t_A^2 &= |KM|^2 \\(x_D^2 + y_D^2 + f^2) \cdot t_D^2 - 2 \cdot (x_A x_D + y_A y_D + f^2) \cdot t_A t_D + (x_A^2 + y_A^2 + f^2) \cdot t_A^2 &= |KN|^2\end{aligned}\tag{15}$$

V závislosti na t_A jsou vyjádřeny tři další parametry, dosazením do parametrických rovnic lze vypočítat polohy reálných značek L , M a N .

$$\begin{aligned}L_X &= x_B \cdot t_B; \quad L_Y = y_B \cdot t_B; \quad L_Z = f \cdot t_B \\M_X &= x_C \cdot t_C; \quad M_Y = y_C \cdot t_C; \quad M_Z = f \cdot t_C \\N_X &= x_D \cdot t_D; \quad N_Y = y_D \cdot t_D; \quad N_Z = f \cdot t_D\end{aligned}\tag{16}$$

Protože závislost parametrů je popsána kvadratickou rovnicí má značka dvě možné polohy, celkem tedy tyto tři body popisují $2^3=8$ možných rovin. Parametr t_A je třeba zvolit tak, aby vzdálenosti značek na vypočtené rovině co nejpřesněji odpovídaly reálným vzdálenostem na podložce. Hodnota parametru t_A se postupně mění od nuly do jedné s daným krokem.

Pro každou hodnotu t_A se vypočtou polohy značek L , M a N vypočítá obecná rovnice roviny podložky.

$$\begin{aligned}u &= (L_X - M_X, L_Y - M_Y, L_Z - M_Z) \\v &= (L_X - N_X, L_Y - N_Y, L_Z - N_Z) \\w &= u \times v = (a, b, c)\end{aligned}\tag{17}$$

Obecná rovnice:

$$ax + by + cz + d = 0 \tag{18}$$

Pro každé t_A tedy vyjde osm obecných rovin, protože každá ze značek má dvě možné polohy.

Nyní se vypočítá průsečík všech parametrických rovnic přímek s touto rovinou, pro každou detekovanou a identifikovanou značku na promítací rovině se tak získá poloha odpovídající reálné značky. Tímto postupem tak lze získat až osm poloh reálných značek, čtyři z těchto značek popisují přímo obecnou rovinu.

V dalším kroku se vypočítají vzájemné vzdálenosti takto vypočtených bodů, tyto vzdálenosti se zapíší na odpovídající index do matice D_V . V matici D_R jsou naopak zapsány skutečné vzdálenosti značek. Optimální obecná rovnice musí splňovat následující vztah. Hledá se minimální odchylka skutečných vzdáleností od vzdáleností vypočtených:

FOR $t_A = 0$ *TO* 1 *je třeba zvolit vhodný krok, např. 0.001*

FOR $k = 0$ *TO* 8 *pro každé t_A je osm kombinací poloh značek*

$$\text{minimum} = \min \left[\left(\sum_{i=0}^8 \sum_{j=i}^8 D_R[i, j] - D_V[i, j] \right), \text{minimum} \right];$$

vypočítají se minimální vzdálenosti všech osmi značek od reálných vzdáleností

D_R - matice reálných vzdáleností mezi značkou s indexem i a j

D_V - matice vypočtených vzdáleností mezi značkou s indexem i a j

ENDFOR

ENDFOR

Nyní je již známa obecná rovnice roviny, která má nejmenší odchylku vzdáleností značek od reálných vzdáleností. Následujícím krokem je převedení stávajícího souřadného systému do systému nového. Počátek nového souřadného systému se položí do středu podložky, osy souřadného systému budou ortogonální s rovinou podložky.

Dále se vypočítají souřadnice kamery z pohledu podložky. Obecná rovnice roviny s podložkou je:

$$ax + by + cz + d = 0 \tag{19}$$

Parametrická rovnice přímky kolmé na rovinu procházející bodem $[0,0,0]$, ve kterém leží kamera:

$$\begin{aligned} x &= a \cdot t \\ y &= b \cdot t \\ z &= c \cdot t \end{aligned} \tag{20}$$

Dosazením parametrických rovnic do obecné rovnice lze zjistit hodnotu parametru t :

$$t = \frac{-d}{a^2 + b^2 + c^2} \tag{21}$$

Souřadnice kamery vůči počátku ležícímu v rovině podložky (bod leží v obecné rovině podložky):

$$\begin{aligned} x &= a \cdot \frac{-d}{a^2 + b^2 + c^2} \\ y &= b \cdot \frac{-d}{a^2 + b^2 + c^2} \\ z &= c \cdot \frac{-d}{a^2 + b^2 + c^2} \end{aligned} \tag{22}$$

Vzdálenost z_p kamery od nového středu se vypočítá z obecné rovnice dosazením 0 za x a y :

$$a \cdot 0 + b \cdot 0 + cz + d = 0$$

$$z_P = \frac{-d}{c} \quad (23)$$

Kolmá vzdálenost z_k kamery od obecné roviny:

$$z_k = c \cdot \frac{-d}{a^2+b^2+c^2} \quad (24)$$

Úhel přímky pohledu s rovinou podložky K_α lze tedy vypočítat:

$$K_\alpha = \sin^{-1} \frac{z_k}{z_P} \quad (25)$$

Body na podložce mají souřadnice B_{i_X} , B_{i_Y} a B_{i_Z} střed podložky S lze vypočítat:

$$\begin{aligned} S_X &= \frac{\sum_{i=0}^4 \frac{B_{i_X} + B_{(i+4)_X}}{2}}{4} \\ S_Y &= \frac{\sum_{i=0}^4 \frac{B_{i_Y} + B_{(i+4)_Y}}{2}}{4} \\ S_Z &= \frac{\sum_{i=0}^4 \frac{B_{i_Z} + B_{(i+4)_Z}}{2}}{4} \end{aligned} \quad (26)$$

Výpočet úhlu K_β pomocí výpočtu úhlu dvou vektorů.

$$\cos \varphi = \frac{\vec{u} \cdot \vec{v}}{|\vec{u}| \cdot |\vec{v}|} \quad (27)$$

Je třeba zvolit vhodné vektory, které budou oba ležet v rovině podložky. Jedním vektorem může být vektor (1,0) tedy kladná osa x. Druhý vektor může být definován např. takto:

$$\begin{aligned} d_{X_X} &= B_{3_X} - B_{2_X} \\ d_{X_Y} &= B_{3_Y} - B_{2_Y} \\ d_{X_Z} &= B_{3_Z} - B_{2_Z} \end{aligned} \quad (28)$$

Úhel K_β , otočení kamery kolem osy Z lze vyjádřit:

$$K_\beta = \cos^{-1} \frac{d_{X_X}}{\sqrt{d_{X_X}^2 + d_{X_Y}^2 + d_{X_Z}^2}} \quad (29)$$

Pro následnou triangulaci je třeba, aby střed souřadnic nového souřadného systému ležel ve středu podložky, proto je nutné od souřadnic, které vyjadřují pozici kamery, odečíst souřadnice středu podložky a tím posunout střed nového systému do tohoto středu. Kamera bude mít vzhledem k středu podložky souřadnice:

$$\begin{aligned} K_X &= x - S_X = a \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{i_X} + B_{(i+4)_X}}{2}}{4} \\ K_Y &= y - S_Y = b \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{i_Y} + B_{(i+4)_Y}}{2}}{4} \\ K_Z &= z - S_Z = c \cdot \frac{-d}{a^2+b^2+c^2} - \frac{\sum_{i=0}^4 \frac{B_{i_Z} + B_{(i+4)_Z}}{2}}{4} \end{aligned} \quad (30)$$

Příloha 3. Odvození rovnic pro 3D rekonstrukci

Postup odvození:

Cílem je najít obecnou rovnici, která popisuje rovinu čárového laserového paprsku, a parametrickou rovnici přímky procházející kamerou a pixelem na promítací rovině a nalézt jejich průsečík. Je třeba vypočítat 3D souřadnice pixelů na promítací rovině a najít jeden pomocný bod pro výpočet obecné rovnice laseru.

Výpočet parametrické rovnice pohledu kamery

- a) Nejprve je nutné vypočítat pomocný bod R , průsečík roviny ve které leží podložka a přímky pohledu kamery.

Bod R bude mít zjevně souřadnici Z rovnu nule. Souřadnice X a Y se vypočítají pomocí dvou postupných kroků, nejprve se z úhlu K_α a ze souřadnice K_Z vypočítá Δ_{XY} :

$$\Delta_{XY} = \frac{K_Z}{\tan K_\alpha} \quad (31)$$

Nyní je nutné Δ_{XY} rozložit do os X a Y :

$$\begin{aligned} \Delta_X &= \Delta_{XY} \cdot \sin K_\beta \\ \Delta_Y &= \Delta_{XY} \cdot \cos K_\beta \end{aligned} \quad (32)$$

Po připočtení k souřadnicím X a Y bodu K dostaneme souřadnice bodu R :

$$\begin{aligned} R_X &= K_X + \Delta_X \\ R_Y &= K_Y + \Delta_Y \\ R_Z &= 0 \end{aligned} \quad (33)$$

Po dosazení:

$$\begin{aligned} R_X &= K_X + \frac{K_Z}{\tan K_\alpha} \cdot \sin K_\beta \\ R_Y &= K_Y + \frac{K_Z}{\tan K_\alpha} \cdot \cos K_\beta \\ R_Z &= 0 \end{aligned} \quad (34)$$

- b) Výpočet parametrické rovnice přímky pohledu kamery.

Pro výpočet parametrické rovnice je nutné vypočítat směrový v vektor přímky:

$$\begin{aligned} v &= (R_X - K_X, R_Y - K_Y, R_Z - K_Z) \\ v &= \left(K_X + \frac{K_Z}{\tan K_\alpha} \cdot \sin K_\beta - K_X, K_Y + \frac{K_Z}{\tan K_\alpha} \cdot \cos K_\beta - K_Y, 0 - K_Z \right) \\ v &= \left(\frac{K_Z}{\tan K_\alpha} \cdot \sin K_\beta, \frac{K_Z}{\tan K_\alpha} \cdot \cos K_\beta, -K_Z \right) \end{aligned} \quad (35)$$

Parametrická rovnice přímky pohledu je:

$$\begin{aligned} x &= K_X + v_X \cdot t \\ y &= K_Y + v_Y \cdot t \\ z &= K_Z + v_Z \cdot t \end{aligned} \quad (36)$$

Po dosazení:

$$\begin{aligned}x &= K_X + \frac{K_Z}{\tan K_\alpha} \cdot \sin K_\beta \cdot t \\y &= K_Y + \frac{K_Z}{\tan K_\alpha} \cdot \cos K_\beta \cdot t \\z &= K_Z - K_Z \cdot t\end{aligned}\quad (37)$$

Výpočet bodu F (střed promítací roviny) na přímce pohledu kamery, ve vzdálenosti f od bodu K .

Souřadnice F_X a F_Y jsou jasné:

$$\begin{aligned}F_X &= K_X + \cos K_\alpha \cdot \sin K_\beta \cdot f \\F_Y &= K_Y + \cos K_\alpha \cdot \cos K_\beta \cdot f\end{aligned}\quad (38)$$

Bod F leží na přímce pohledu kamery, proto lze vypočítat parametr t přímky.

$$x = F_X \text{ a } y = F_Y$$

Dosazením:

$$\begin{aligned}K_X + \frac{K_Z}{\tan K_\alpha} \cdot \sin K_\beta \cdot t &= K_X + \cos K_\alpha \cdot \sin K_\beta \cdot f \\t &= \sin K_\alpha \cdot \frac{f}{K_Z}\end{aligned}\quad (39)$$

Pro kontrolu:

$$\begin{aligned}K_Y + \frac{K_Z}{\tan K_\alpha} \cdot \cos K_\beta \cdot t &= K_Y + \cos K_\alpha \cdot \cos K_\beta \cdot f \\t &= \sin K_\alpha \cdot \frac{f}{K_Z}\end{aligned}\quad (40)$$

Dosazením za t do parametrické rovnice přímky získáme souřadnice bodu R :

$$\begin{aligned}F_X &= K_X + \cos K_\alpha \cdot \sin K_\beta \cdot f \\F_Y &= K_Y + \cos K_\alpha \cdot \cos K_\beta \cdot f \\F_Z &= K_Z - f \sin K_\alpha\end{aligned}\quad (41)$$

Výpočet 3D souřadnic pixelů ležících na promítací rovině.

Body na promítací rovině mají souřadnice x'_p a y'_p jsou v pixelech, proto je nutné pomocí koeficientů k_x a k_y převést na souřadnice x' a y' v milimetrech.

$$\begin{aligned}x' &= x'_p \cdot k_x \\y' &= y'_p \cdot k_y\end{aligned}\quad (42)$$

Promítací rovina je natočena o úhel K_γ a proto je třeba souřadnice přepočítat, souřadnice pixelu svírají s úhlem ω k tomuto úhlu se připočte úhel K_γ vypočítají se nové souřadnice pixelu:

$$IF \ x' = 0 \ \wedge \ y' > 0 \ \Rightarrow \ \omega = \frac{\pi}{2}$$

$$\begin{aligned}
IF \ x' = 0 \wedge y' < 0 &\Rightarrow \omega = \frac{3 \cdot \pi}{2} \\
IF \ x' > 0 \wedge y' \geq 0 &\Rightarrow \omega = \tan^{-1} \frac{y'}{x'} \\
IF \ x' < 0 \wedge y' < 0 &\Rightarrow \omega = \pi + \tan^{-1} \frac{y'}{x'} \\
IF \ x' < 0 \wedge y' \geq 0 &\Rightarrow \omega = \pi - \tan^{-1} \frac{y'}{\text{abs}(x')} \\
IF \ x' > 0 \wedge y' < 0 &\Rightarrow \omega = 2 \cdot \pi - \tan^{-1} \frac{\text{abs}(y')}{x'}
\end{aligned} \tag{43}$$

Po přičtení úhlu K_Y :

$$\begin{aligned}
x'_0 &= \sin(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2} \\
y'_0 &= \cos(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2}
\end{aligned} \tag{44}$$

Změna Δ_Z v ose Z a Δ_{XY} v rovině XY :

$$\begin{aligned}
\Delta_Z &= \cos K_\alpha \cdot y'_0 \\
\Delta_{XY} &= \sin K_\alpha \cdot x'_0
\end{aligned} \tag{45}$$

3D souřadnice pixelu (bod E) na promítací rovině jsou:

$$\begin{aligned}
E_X &= K_X + \Delta_{XY} \cdot \sin K_\beta + \cos K_\beta \cdot x'_0 + \cos K_\alpha \cdot \sin K_\beta \cdot f \\
E_Y &= K_Y + \Delta_{XY} \cdot \cos K_\beta + \sin K_\beta \cdot x'_0 + \cos K_\alpha \cdot \cos K_\beta \cdot f \\
E_Z &= K_Z - f \sin K_\alpha + \Delta_Z
\end{aligned} \tag{46}$$

Po dosazení:

$$\begin{aligned}
E_X &= K_X + \sin K_\alpha \cdot \sin(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2} \cdot \sin K_\beta + \cos K_\beta \cdot \sin(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2} + \cos K_\alpha \cdot \sin K_\beta \cdot f \\
E_Y &= K_Y + \sin K_\alpha \cdot \sin(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2} \cdot \cos K_\beta + \sin K_\beta \cdot \sin(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2} + \cos K_\alpha \cdot \cos K_\beta \cdot f \\
E_Z &= K_Z - f \sin K_\alpha + \cos K_\alpha \cdot \cos(\omega + K_Y) \cdot \sqrt{x'^2 + y'^2}
\end{aligned} \tag{47}$$

Výpočet 3D souřadnice zdroje laserové paprsku, bod L .

Laserový zdroj je pootočen o úhel K_Y k rovině XY , další výpočet bude obdobný jako při výpočtu bodu E :

$$\begin{aligned}
\Delta_Z &= \cos K_\alpha \cdot \cos K_Y \cdot d \\
\Delta_{XY} &= \sin K_\alpha \cdot \cos K_Y \cdot d
\end{aligned} \tag{48}$$

Bod L :

$$\begin{aligned}
L_X &= K_X + \Delta_{XY} \cdot \sin K_\beta + \cos K_\beta \cdot \sin K_Y \cdot d \\
L_Y &= K_Y + \Delta_{XY} \cdot \cos K_\beta + \sin K_\beta \cdot \sin K_Y \cdot d \\
L_Z &= K_Z + \Delta_Z
\end{aligned} \tag{49}$$

Dosazením:

$$\begin{aligned}
 L_X &= K_X + \sin K_\alpha \cdot \cos K_\gamma \cdot d \cdot \sin K_\beta + \cos K_\beta \cdot \sin K_\gamma \cdot d \\
 L_Y &= K_Y + \sin K_\alpha \cdot \cos K_\gamma \cdot d \cdot \cos K_\beta + \sin K_\beta \cdot \sin K_\gamma \cdot d \\
 L_Z &= K_Z + \cos K_\alpha \cdot \cos K_\gamma \cdot d
 \end{aligned} \tag{50}$$

Výpočet pomocného bodu M . Pro vyjádření obecné rovnice roviny laseru, je nutné znát 3 body na rovině nebo je nutné vypočítat parametrickou rovnici přímky kolmé na rovinu.

$$\begin{aligned}
 g &= \frac{d}{\tan \alpha} \\
 \Delta_Z &= \sin K_\alpha \cdot g \\
 \Delta_{XY} &= \cos K_\alpha \cdot g
 \end{aligned} \tag{51}$$

Bod M :

$$\begin{aligned}
 M_X &= K_X - \Delta_{XY} \cdot \sin K_\beta \\
 M_Y &= K_Y - \Delta_{XY} \cdot \cos K_\beta \\
 M_Z &= K_Z + \Delta_Z
 \end{aligned} \tag{52}$$

Dosazením:

$$\begin{aligned}
 M_X &= K_X - \cos K_\alpha \cdot \frac{d}{\tan \alpha} \cdot \sin K_\beta \\
 M_Y &= K_Y - \cos K_\alpha \cdot \frac{d}{\tan \alpha} \cdot \cos K_\beta \\
 M_Z &= K_Z + \sin K_\alpha \cdot \frac{d}{\tan \alpha}
 \end{aligned} \tag{53}$$

Výpočet parametrické rovnice přímky procházející bodem K a 3D souřadnicí pixelu (bodem E) na promítací rovině.

$$\begin{aligned}
 x &= K_X + (K_X - E_X) \cdot t \\
 y &= K_Y + (K_Y - E_Y) \cdot t \\
 z &= K_Z + (K_Z - E_Z) \cdot t
 \end{aligned} \tag{54}$$

Dosazením:

$$\begin{aligned}
 x &= K_X + \left(-\sin K_\alpha \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \cdot \sin K_\beta - \cos K_\beta \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} - \cos K_\alpha \cdot \sin K_\beta \cdot f \right) \cdot t \\
 y &= K_Y + \left(-\sin K_\alpha \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \cdot \cos K_\beta - \sin K_\beta \cdot \sin(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} - \cos K_\alpha \cdot \cos K_\beta \cdot f \right) \cdot t \\
 z &= K_Z + \left(f \sin K_\alpha - \cos K_\alpha \cdot \cos(\omega + K_\gamma) \cdot \sqrt{x'^2 + y'^2} \right) \cdot t
 \end{aligned} \tag{55}$$

Výpočet obecné rovnice laseru, pro výpočet se použijí body L a M jejichž pomocí se směrnice vektor přímky kolmé na hledanou rovinu a protože směrnice vektor přímky kolmé na obecnou rovinu je stejný jako normálový vektor této roviny, najdeme tak obecnou rovnici roviny laseru.

Obecná rovnice má tvar:

$$ax + by + cz + d = 0 \quad (56)$$

Dosazením směrnice vektoru kolmé přímky za vektor normálový:

$$(L_X - M_X) \cdot x + (L_Y - M_Y) \cdot y + (L_Z - M_Z) \cdot z - [(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z] = 0 \quad (57)$$

Parametr t lze získat dosazením parametrické rovnice přímky procházející bodem K a pixelem E (47) do obecné rovnice laseru (57).

$$t = \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_Y - E_Y) + (L_Z - M_Z) \cdot (K_Z - E_Z)} \quad (58)$$

Dosazením vypočteného parametru t do parametrické rovnice přímky (55) procházející pixelem E a bodem K se vypočítají 3D souřadnice hledaného bodu na povrchu snímaného objektu:

$$\begin{aligned} x &= K_X + (K_X - E_X) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_Y - E_Y) + (L_Z - M_Z) \cdot (K_Z - E_Z)} \\ y &= K_Y + (K_Y - E_Y) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_Y - E_Y) + (L_Z - M_Z) \cdot (K_Z - E_Z)} \\ z &= K_Z + (K_Z - E_Z) \cdot \frac{(L_X - M_X) \cdot L_X + (L_Y - M_Y) \cdot L_Y + (L_Z - M_Z) \cdot L_Z}{(L_X - M_X) \cdot (K_X - E_X) + (L_Y - M_Y) \cdot (K_Y - E_Y) + (L_Z - M_Z) \cdot (K_Z - E_Z)} \end{aligned} \quad (59)$$

Zjednodušený výpočet:

Pro zjednodušený výpočet lze použít rovnici kde úhly K_α , K_β a K_γ jsou rovny 0 a také K_X , K_Y a K_Z jsou nulové. Rovnici této zjednodušené triangulace lze zapsat v tomto tvaru:

$$\frac{y'}{f} \cdot d \cdot \tan \alpha \cdot z + d \cdot z - d^2 \cdot \tan \alpha = 0 \quad (60)$$

Proměnné x' a y' udávají souřadnice pixelů na projekční rovině v milimetrech, f je ohnisková vzdálenost, d je vzdálenost laseru od kamery a α je úhel sklonu laserového paprsku.

Výpočet z-tové souřadnice:

$$z = \frac{d^2 \cdot \tan \alpha}{\frac{y'}{f} \cdot d \cdot \tan \alpha + d} \quad (61)$$

U poslední rovnice platí podmínka:

$$\frac{y'}{f} \cdot d \cdot \tan \alpha + d > 0 \quad (62)$$

Při nesplnění uvedené podmínky dojde ke zkreslení výsledného zrekonstruovaného objektu.

Výpočet souřadnic x a y :

$$x = \frac{x'}{f} \cdot z \quad (63)$$

$$y = \frac{y'}{f} \cdot z \quad (64)$$

Vzdálenosti x' a y' jsou většinou v pixelech, je proto nutné znát převodní koeficienty z pixelů na milimetry.

Příloha 4. Uživatelská příručka

Instalování driverů pro kameru

Pro správnou funkčnost kamery použité na 3D skeneru je nutné doinstalovat ovladače. Pro Windows XP jsou ovladače [zde](#). Pro Windows Vista [zde](#).

Instalace programu

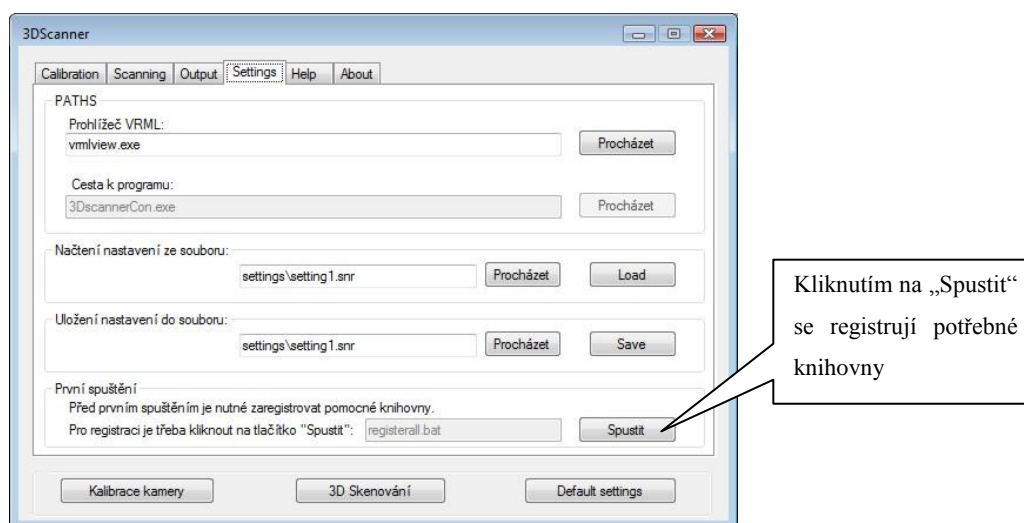
Program lze nainstalovat pomocí standardního instalátoru spuštěním souboru setup.exe, druhou možností je instalace pomocí msi balíčku 3Dscanner.msi. U obou možností lze vybrat místo na disku kam se program nainstaluje. Po úspěšné instalaci přidá instalační program na plochu ikonku pro spuštění programu. Pro instalaci je třeba 10MB místa na disku. Součástí instalace je freewarový program VRMLView, který umožňuje prohlížení výsledných modelů ve formátu VRML.



obr P4.1: Instalátor

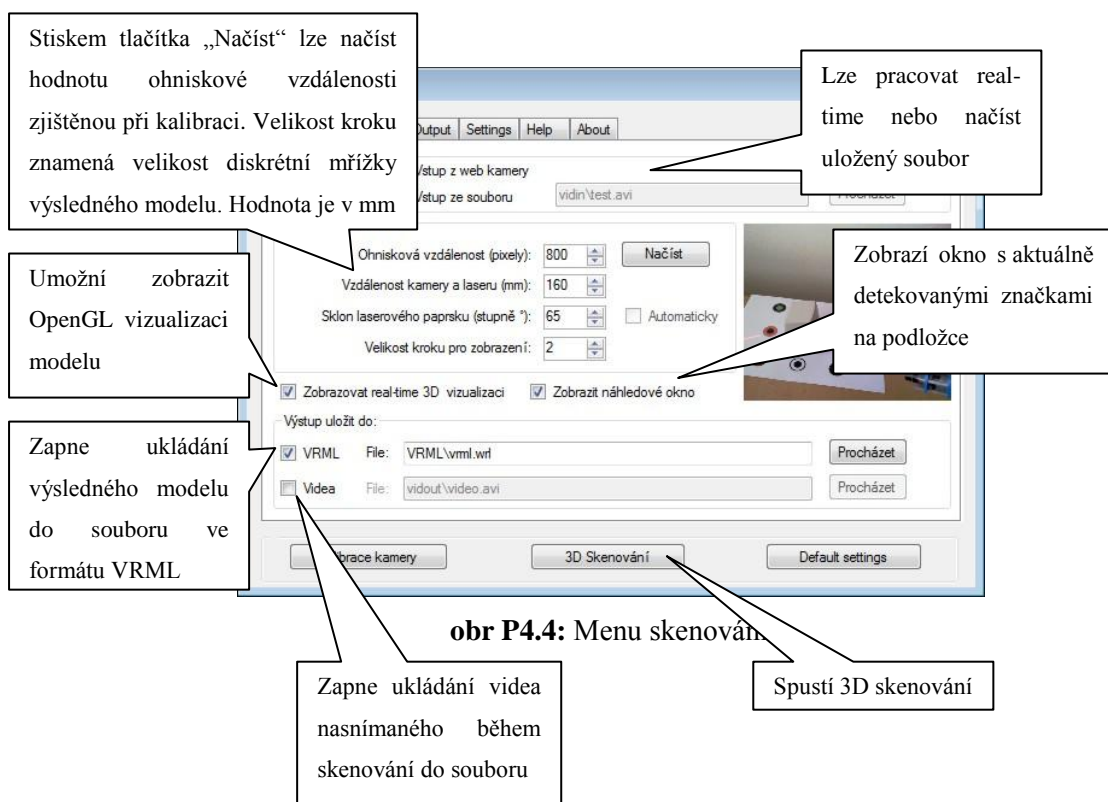
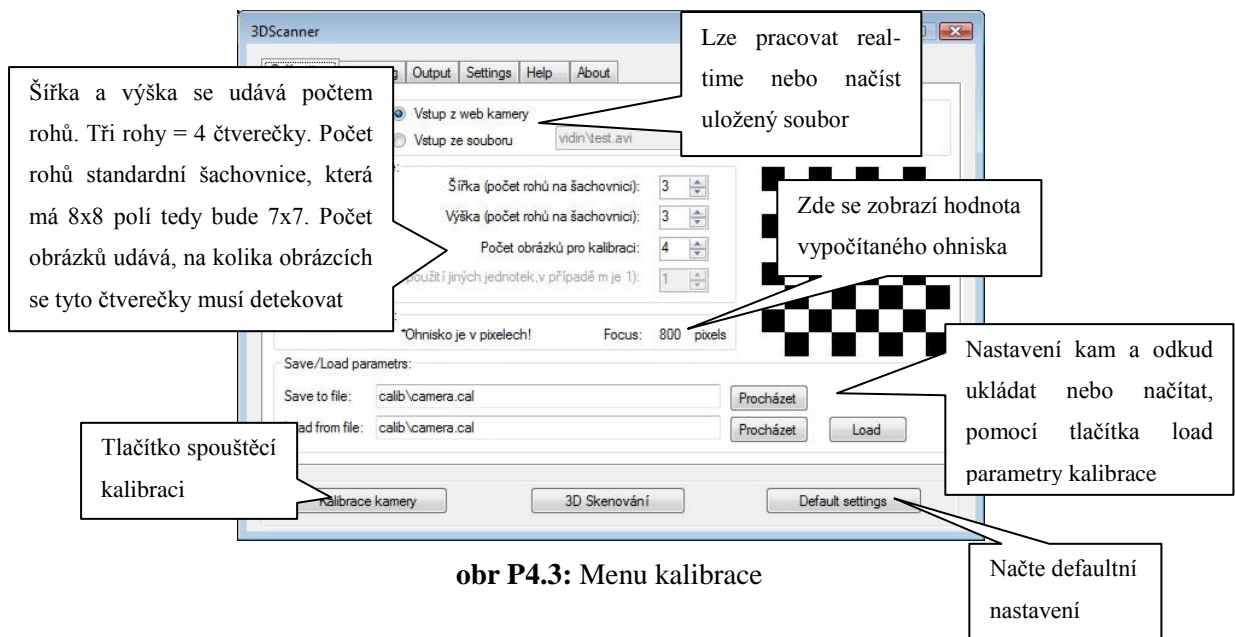
První spuštění programu

Před prvním použitím programu 3Dscanner je třeba registrovat knihovny, které obsluhují kameru. Registrace se provede stisknutím tlačítka „Spustit“ v záložce „Settings“

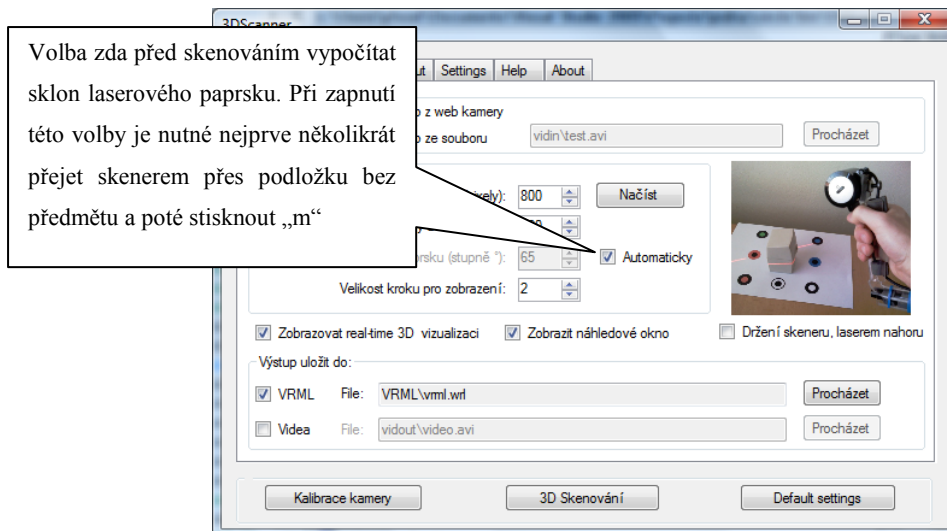


obr P4.2: Registrace knihoven

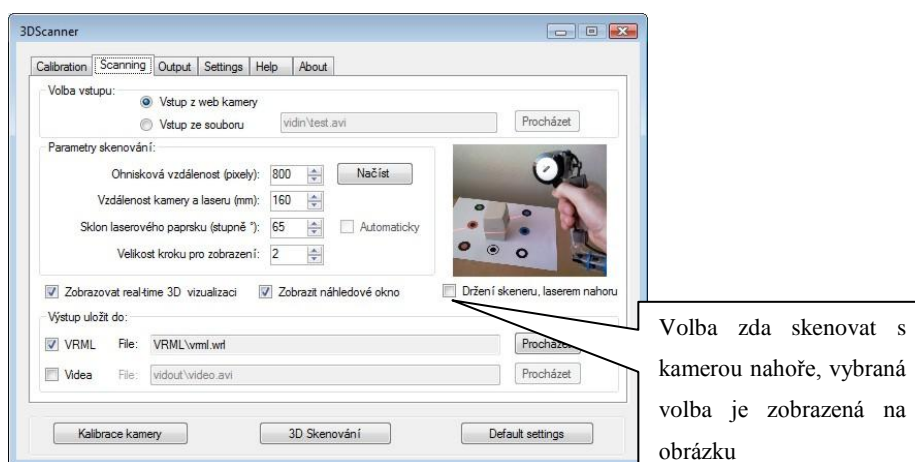
Ovládání programu



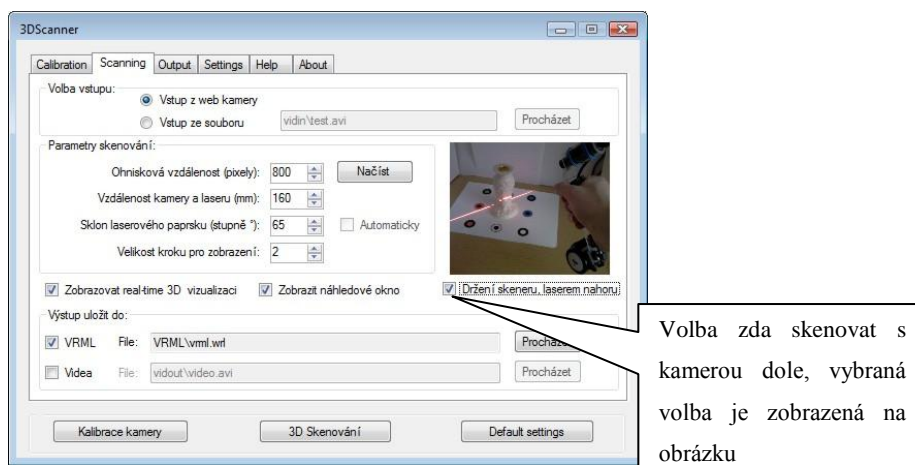
Doporučené držení skeneru je zdrojem laserového paprsku dole, skenování je optimalizováno pro skenování bílých matných předmětů.



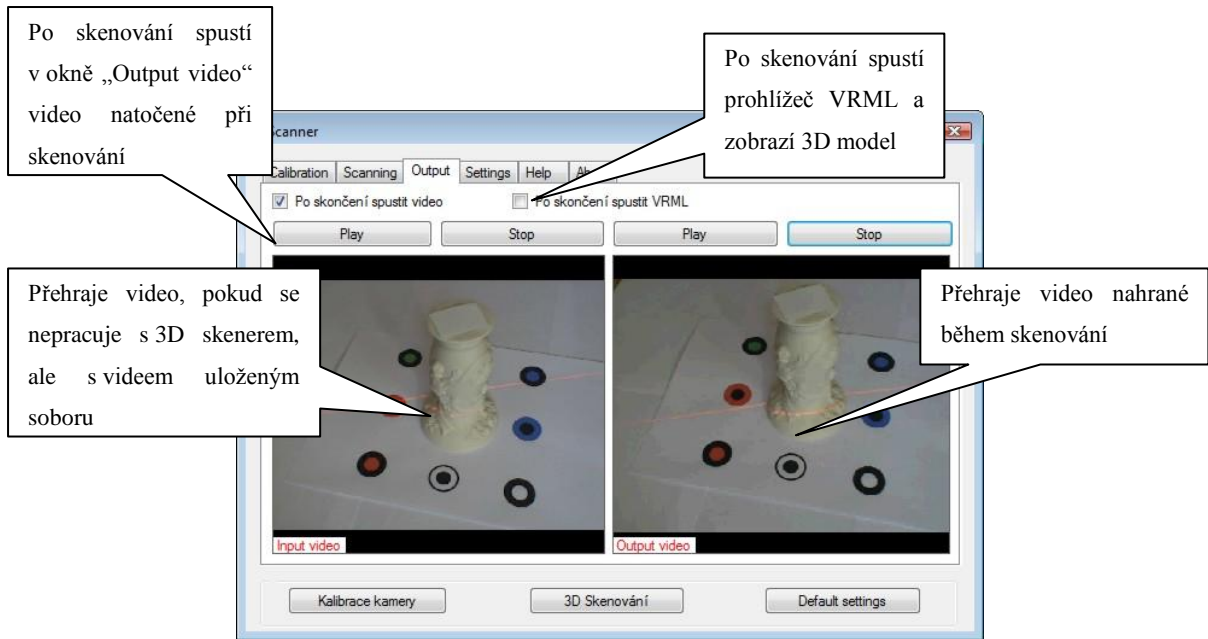
obr P4.6: Menu skenování – automatické zjištění úhlu



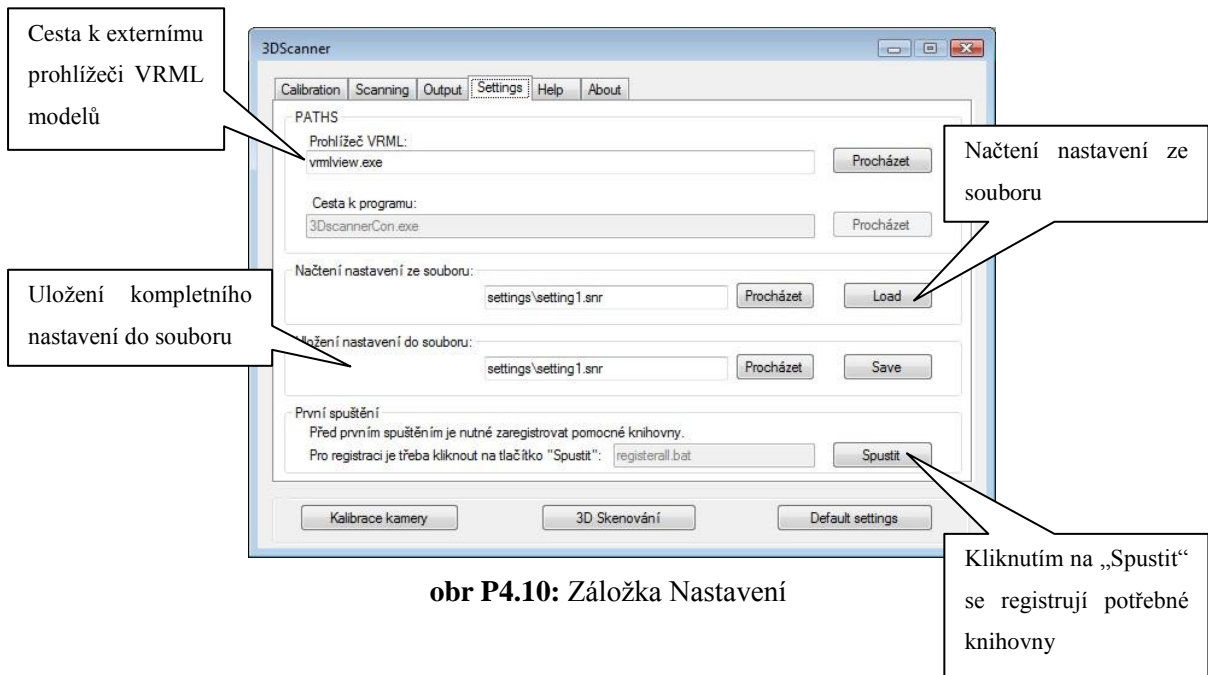
obr P4.7: Menu skenování – držení skeneru, laserem dole



obr P4.8: Menu skenování – držení skeneru, laserem nahore



obr P4.9: Záložka Výstup



obr P4.10: Záložka Nastavení

Práce se skenerem

Detekce laserového paprsku je optimalizována na skenování objektů, které mají matný bílý povrch.

Se skenerem je třeba pracovat pomalu, při rychlejších pohybech je obraz snímaný kamerou rozmazán a nelze ho zpracovat. V některých případech je třeba vypnout u kamery funkci automatického vyvážení bílé.

Postup při skenování:

1. kalibrace kamery - šachovnice
2. vhodné nastavení sklonu kamery
3. kalibrace úhlu kamery (automatický výpočet) - prázdná podložka
4. skenování

Adresářová struktura

Adresář:

- „calib“ – zde se ukládají data vytvořená při kalibraci
- „outpic“ – po stisknutí tlačítka „d“ během skenování zde ukládá aktuální obrázek
- „settings“ – slouží pro ukládání souborů s nastavením
- „vidin“ – zde se ukládají videa pro zpracování
- „vidout“ – uložení videí které byly nahrány během skenování
- „VRML“ – zde se ukládají 3D modely ve formátu VRML

Ukázkové příklady

Po spuštění programu 3Dscanner lze v záložce „Settings“ – Načtení nastavení ze souboru tlačítkem „Procházet“ načíst nastavení pro ukázkové příklady. Ukázkové příklady nejsou součástí instalace, jsou na přiloženém DVD v adresáři „samples“. Na DVD jsou příklady označené písmeny od „A“ do „G“. V každém ukázkovém příkladu je v adresáři „settings“ nastavení, které je nutné pomocí výše uvedeného postupu načíst. U každého příkladu jsou tři přednastavené nastavení, první pro kalibraci kamery, druhé pro kalibraci laseru a třetí pro skenování.

Popis příkladů:

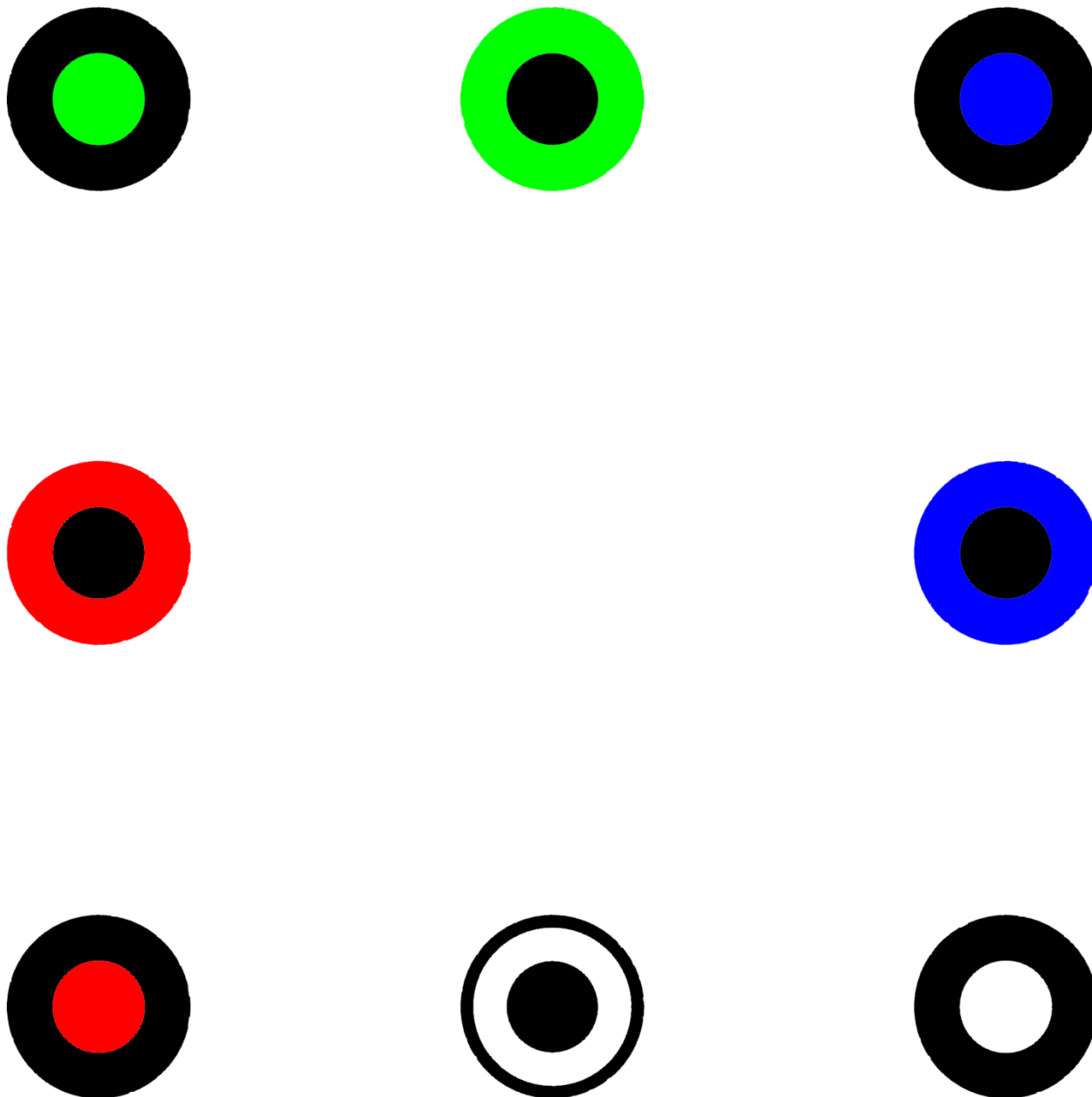
- „A“, „B“ a „C“ – skenování krabičky B
- „D“ a „E“ – skenování krabičky A
- „F“ a „G“ – skenování krabičky B obráceným držením skeneru

Odinstalování

Program 3Dscanner lze odinstalovat standardním způsobem: „Ovládací panely“ -> „Přidat/odebrat programy“

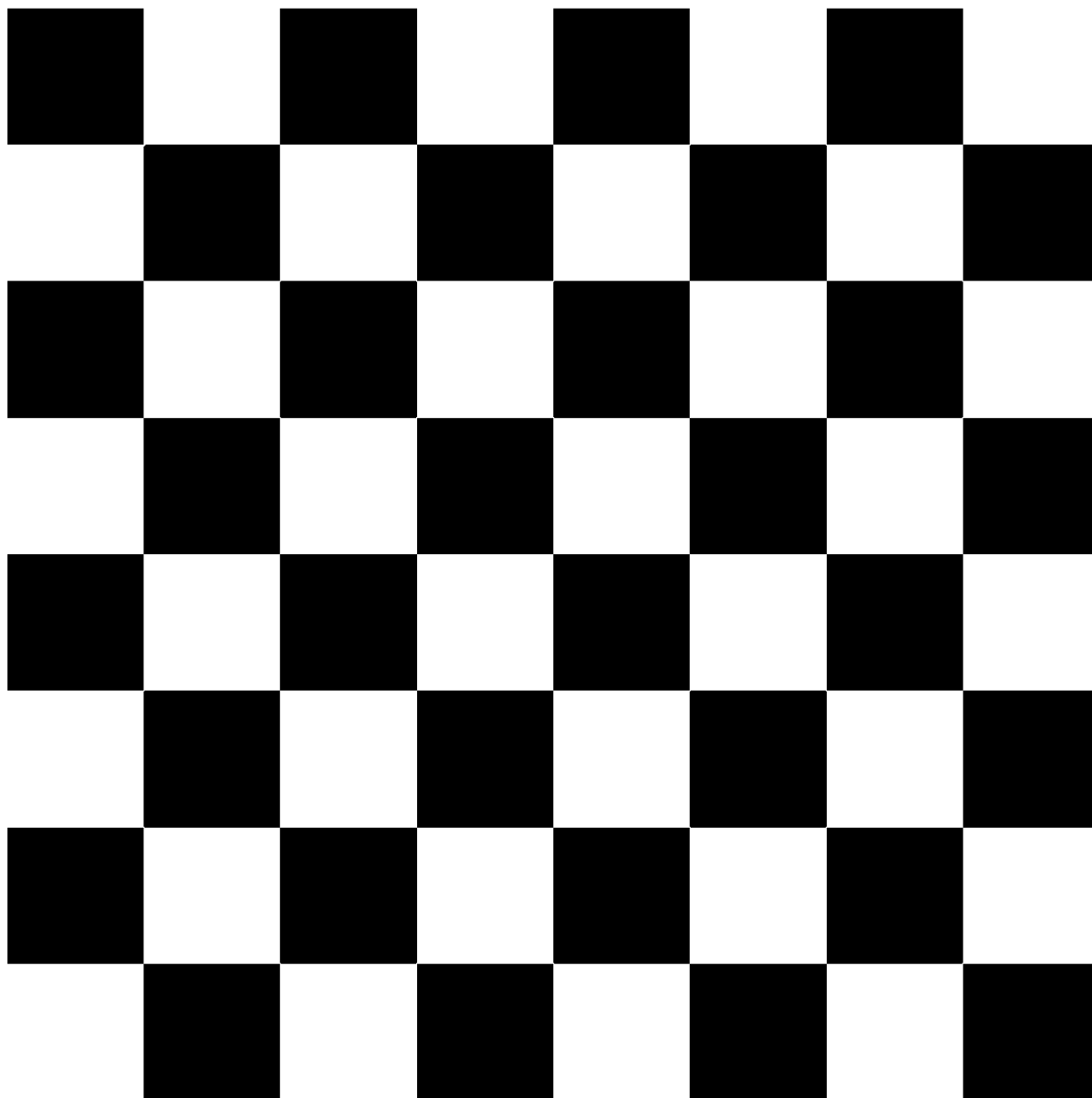
Příloha 5. Podložka (skutečná velikost)

Vzdálenosti středů značek na podložce jsou 70 mm, poloměr malé kružnice je 7 mm, poloměr velké je 14 mm.



obr P5.1: Skutečná velikost podložky

Příloha 6. Kalibrační šachovnice



obr P6.1: Kalibrační šachovnice