

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Tvorba aplikace pro rekonstrukci povrchu z mračna bodů

Bakalářská práce

Vedoucí práce:
Ing. Jaromír Landa, Ph.D.

Filip Jandora

Brno 2015

Rád bych poděkoval panu Ing. Jaromíru Landovi, Ph.D. za odbornou pomoc při realizaci této bakalářské práce, ochotu a trpělivost, kterou mi během jejího vypracování věnoval.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Tvorba aplikace pro rekonstrukci povrchu z mračna bodů**

vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

Brno, 8. května 2015

.....

Abstract

Jandora, F. Creating application for surface reconstruction from a point cloud. Bachelor thesis. Mendel University in Brno, Faculty of Business and Economics, 2015.

This bachelor thesis is dedicated to the surface reconstruction from a point cloud. With this application it is possible to smooth out and reconstruct loaded point cloud and then use it as an object. Application is based on the C++ programming language supported by Qt and PCL libraries.

Keywords: Point cloud, surface reconstruction, C++, Qt, PCL

Abstrakt

Jandora, F. Tvorba aplikace pro rekonstrukci povrchu z mračna bodů. Bakalářská práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta, 2015.

Tato bakalářská práce je věnována problematice rekonstrukce povrchu z mračna bodů. Pomocí této aplikace je možné vyhladit a zrekonstruovat nahraný mrak bodů a dále jej pak používat jako objekt. Aplikace je sestavena na základě programovacího jazyka C++, za podpory knihoven Qt a PCL.

Klíčová slova: Mračno bodů, rekonstrukce povrchu, C++, Qt, PCL

Obsah

1	Úvod a cíl práce	8
1.1	Úvod	8
1.2	Cíl práce	9
2	Laserové skenování a mračna bodů	10
2.1	Laserové skenování	10
2.2	Způsoby pořízení mračna bodů v praxi	11
2.2.1	Letecké skenování	11
2.2.2	Mobilní skenování	12
2.2.3	Pozemní skenování	13
2.3	Ukládání mračna bodů	13
2.3.1	Textové soubory	13
2.3.2	Binární soubory	14
3	Využití mračna bodů	15
4	Problematika rekonstrukce 3D povrchů	16
5	Analýza současných metod pro rekonstrukci povrchů	18
5.1	Marching cubes	18
5.2	Marching tetrahedra	18
5.3	Delaunay triangulation	19
5.4	Ball pivoting	20
5.5	Poisson surface reconstruction	20
6	Metodika	21
7	Technologie	22
7.1	C++	22
7.2	Qt	22
7.3	Point Cloud Library	23
7.3.1	Moving Least Squares	23
7.3.2	Greedy surface triangulation	23
8	Vlastní práce	25
8.1	Načítání souboru	25
8.2	Zjištění základních údajů a vizualizace	25
8.3	Zobrazení normál	26
8.4	Úprava mračna bodů	26
8.5	Rekonstrukce povrchu	27
8.6	Krok zpět	28
8.7	Ukládání souboru	28

9	Možná rozšíření	29
9.1	Typy souborů	29
9.2	Možnosti rekonstrukce	29
9.3	Barevná mračna	29
9.4	Práce po rekonstrukci	29
10	Diskuze	30
10.1	Vývojové prostředí a operační systém	30
10.2	Verze Qt	30
10.3	Nedostatečná dokumentace PCL	31
10.4	Proměnná prostředí	31
10.5	Zhodnocení aplikace	31
11	Závěr	32
12	Literatura	33

Seznam obrázků

- Obrázek 1: Příklad mračna bodů (Mederos, Velho a Figueiredo, 2003). 10
- Obrázek 2: Ukázka několikanásobné reflexe paprsku při průchodu vegetací (Tamás, 2010a). 12
- Obrázek 3: Nejtypičtější problém při rekonstrukci povrchu - šum. Ve spodní části pak můžeme vidět jeho vyhlazení (Marton, Rusu a Beetz, 2009). 16
- Obrázek 4: Možné výsledky povrchu pro jednu z rychlých algoritmu marching cubes (Lorensen a Cline, 1987). 18
- Obrázek 5: Ukázka rozložení krychle na čtyřstěny (Bourke, 1994). 19
- Obrázek 6: Ukázka algoritmu ve 2D prostoru. Při rozhodování mezi červenou a zelenou hranou, bude podle pravidla pro minimalizaci úhlů upřednostněna zelená hrana (Maur, 2002). 19
- Obrázek 7: Ilustrace jednotlivých kroků Poissonovy metody (Kazhdan, Bolitho a Hoppe, 2006). 20
- Obrázek 8: Na levé straně můžeme vidět mračno bodů obsahující šum. Vedle je možno shlédnout normály poté, co prošly algoritmem Moving Least Squares (Rusu, 2009). 23
- Obrázek 9: Zobrazení aplikace po načtení souboru bal.ply, který obsahuje mračno bodů. 26

1 Úvod a cíl práce

1.1 Úvod

S výpočetní technikou se lze v současné době setkat prakticky na každém kroku. Neobejdou se bez ní ani obory, jejichž posláním nebyla primárně práce s počítači. Mezi takové obory patří zejména ekonomie, architektura, sociologie a mnohé další. V dnešním moderním světě je celá řada úkonů na výpočetní technice přímo úměrně závislá. Její přínos pak narůstá s rozsahem zpracovávaných dat. Nasazení výpočetní techniky ale neznamená pouze nakoupit počítače, stroje či další komponenty. Jedná se o složitý proces, který zahrnuje taktéž aplikaci algoritmů, jež následně umožňují zadané úkoly řešit. K dosažení správného výsledku je neméně důležitá i interakce techniky s jejím uživatelem a současně také zobrazení ve správné podobě. Pro lidské oko bývá zpravidla nejsrozumitelnější právě výstup grafický.

Postupem času se grafické výstupy stávaly stále složitějšími. Jejich interpretace tak položila základní kámen vzniku zcela novému oboru, který byl nazván počítačovou grafikou. Ta má základy již v 60. letech minulého století, kdy bylo poprvé toto slovní spojení použito. Následovalo využití CRT obrazovek či světelného pera. Koncem 80. let se do popředí dostalo grafické uživatelské rozhraní a současně byl položen základ pro 3D grafiku, která získala popularitu především v 90. letech. Zájem lidí o 3D grafiku rostl zvláště díky počítačovým hrám a prvním animovaným filmům.

Úkolem počítačové grafiky ale nebylo pouze upravovat výstupy aplikací. Do popředí se dostala schopnost pracovat s informacemi z reálného světa, zvláště pak došlo k zaměření se na jejich získání a úpravu. Uživatel má více možností, jak data získat. Jednou z nich je vytvoření si svých vlastních dat, např. malováním či modelováním. Jedná se však o zdoluhavý proces, jehož realizace může být s přibývajícím detaily na objektu hůře proveditelná a tedy i časově náročnější. Grafická data lze také získat použitím hardwaru (ať už se jedná o digitální fotoaparáty či skenery), který dokáže fyzický objekt zdigitalizovat. S takto získanými daty pak můžeme snadno manipulovat v oblasti výpočetní techniky. Z těchto vstupních zařízení uživatel získá zpravidla obrazová data. Lze se však setkat i s netradičními výstupy, tzv. mračny bodů. Tato mračna bývají produktem tzv. 3D skenerů, které do jejich podoby dokáží převést reálné objekty.

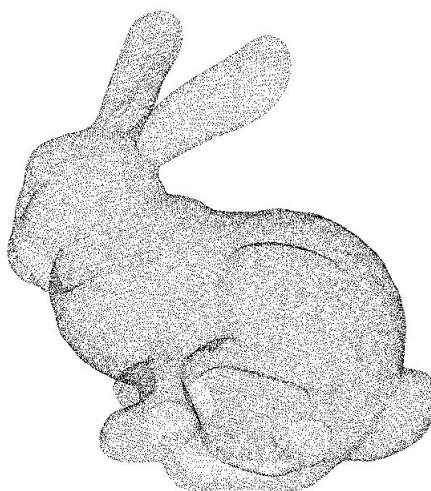
Mračno bodů si lze představit jako množinu dat nacházející se v trojrozměrném souřadnicovém systému. Jednotlivým bodům je ukládána pozice na každé ze tří os. Přesto, že se mračna dají za pomoci grafického softwaru zobrazit, převádějí se na tělesa. K tomuto převodu slouží právě rekonstrukce povrchu. Tato tělesa se pak dále využívají v celé řadě odvětví, např. v lékařství, georeferencování map či reverzním inženýrství.

1.2 Cíl práce

Za cíl práce jsem si vytyčil vytvoření grafické aplikace, která bude schopna importovat získaná mračna bodů a za pomoci implementovaných funkcí zrekonstruovat jejich povrch. Rovněž jsem si stanovil určitá kritéria, která byla nutno pro tvorbu programu dodržet. První spočívalo v jednoduchosti ovládání. Aplikace by neměla být příliš složitá, naopak její ovládání by mělo být zcela intuitivní, tedy takové, aby v ní mohli pracovat i uživatelé, kteří neznají problematiku rekonstrukce povrchů do hloubky. Další podmínkou bylo vytvoření přehledného prostředí, jenž nebude narušováno větším množstvím prvků než je nezbytně nutné.

2 Laserové skenování a mračna bodů

Běžný uživatel si může mračno představit jako shluk bodů nacházející se v 3D prostoru. Jinými slovy lze říci, že se jedná o množinu bodů, jež je součástí trojrozměrného souřadného systému. Každému bodu v mračnu je přiřazena souřadnice x , y a z . Souřadnice jsou nepostradatelné při určování polohy bodů na dané ose. Mimo ně mohou jednotlivé body v množině podávat další informace o vlastnostech, které se týkají daného bodu, např. jeho barva, velikost či průhlednost.



Obrázek 1: Příklad mračna bodů (Mederos, Velho a Figueiredo, 2003).

2.1 Laserové skenování

K laserovému skenování se běžně využívají 2D a 3D skenery. 2D skenery představují nejjednodušší způsob přenosu dat, např. vkládání fotografií do počítače. Mnohem větší využití však mají právě 3D skenery, které umožňují vkládat do počítače prostorové objekty. Pomocí těchto skenovacích přístrojů dochází k určování prostorových souřadnic. V reálném světě mají široké spektrum využití, např. při 3D modelování, vizualizaci staveb, podzemních prostor, ale i drobnějších objektů. Po naskenování dojde k transformaci tělesa na tzv. mračno bodů, ze kterého je později možné vytvořit 3D model.

Neexistuje pouze jeden, univerzální způsob, jak mračno bodů pořídit. V současné době je naopak známá celá řada metod, které dokáží převést reálný objekt do digitální podoby. Uživatel se nejčastěji setkává s metodou triangulace a doby letu, tzv. time-of-flight.

Samotná **triangulace** může probíhat ve více technikách: aktivní triangulace, pasivní triangulace, technika měřících systémů s teodolitem. Techniky se vzájemně liší pouze v maličkostech. Základní princip, tedy podstata triangulace však zůstává stejná (Jähne, Haussecker a Geissler, 1999). Na povrch snímaného tělesa je vyslán

laserový paprsek, který je převáděn na souřadnice bodů. Tyto souřadnice vznikají na základě shody dvou senzorů snímajících laser. Je tedy zřejmé, že tyto senzory musí být vůči sobě kalibrovány a musí se nacházet v poloze, jež je přístroji předem známá. Na základě těchto informací je zařízení schopno vypočítat vzdálenost od tělesa i jeho zakřivení a tato data následně uložit. Metody lze využít pouze v případě, kdy mezi zařízením a snímaným objektem neleží žádná překážka (Mechelke, Kersten a Lindstaedt, 2007).

Time-of-flight metoda je založená na odrazu světelného paprsku od povrchu. Skener vyšle na povrch snímaného tělesa puls světla. Při kontaktu paprsku s povrchem se odrazí a vrací se zpět ke skeneru. Zároveň s tímto procesem dochází k měření časové prodlevy mezi vysláním paprsku a jeho příletem zpět. Uplynulá doba je následně vydělena dvěma (neboť paprsek urazí vzdálenost k tělesu a zpět k přístroji) a za pomoci konstanty pro rychlost světla může dojít k výpočtu vzdálenosti snímaného tělesa od skeneru. Důležitým faktorem je přesnost měřícího zařízení. Rychlost světla, se kterou se počítá, je extrémně vysoká a z tohoto důvodu může i drobná odchylka měření ovlivnit (Bernardini a Rushmeier, 2002).

S laserovým skenováním je spojena řada výhod:

- dochází k automatickému a současně systematickému získávání bodů,
- umožňuje jejich rychlé zaměření,
- vysoká hustota bodů v mračnu,
- souřadnice jsou vypočítávány již během měření nebo bezprostředně po něm,
- existují skenery, které dokáží obohatit vlastnosti bodů i o odrazivost povrchu.

2.2 Způsoby pořízení mračna bodů v praxi

V předchozí kapitole bylo stručně představeno trojrozměrné laserové skenování. V současné době je tento proces hojně využíván k získávání dat nejen o zemském povrchu. Využívá se především následujících typů 3D skenerů:

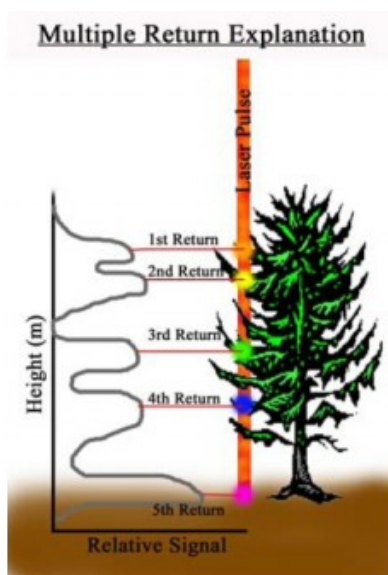
- statický, fixovaný na konkrétní místo (kontrola kvality na výrobních linkách),
- mobilní 3D skener využívající se při měření na krátké a střední vzdálenosti,
- letecké skenovací zařízení napomáhající při tvorbě map.

V následujících odstavcích uvádím podrobnější informace o leteckém, mobilním a pozemním mapování terénu.

2.2.1 Letecké skenování

Letecké skenování se v posledních letech stává stále oblíbenějším nástrojem pro mapování rozsáhlých povrchů, a to nejen v České republice, ale i ve světě. Není se čemu divit, protože výsledky získané tímto způsobem jsou v porovnání s jinými metodami,

použitými v obdobných podmínkách, zpravidla nesrovnatelně lepší. Výhodou není však pouze kvalita výstupu, ale také úspora času při jeho zpracování, které z velké části probíhá automaticky. Laserové paprsky jsou schopny procházet vegetací. Princip tohoto průniku spočívá v odrazení paprsku nikoliv jako celku, ale pouze po částech. V případě, že část paprsku narazí na překážku, odrazí se zpět, zatímco zbývající část pokračuje v původním směru. Tento efekt se opakuje až do doby, kdy je vrácena každá část paprsku. Pokud senzor získá informace z více odrazů, tak dojde k vyhodnocení posledních dat jako polohy georeliéfu. Tento proces může být též chybový, a to v případě průchodu obzvláště hustou vegetací, kdy paprsek nemusí proniknout až na zem. Paprsky jsou pohlcovány sněhem i vodou, a proto není vhodné provádět mapování během zimy. Další nevýhodou představuje vysoká nákladovost na realizaci leteckého skenování. V praxi se často kombinuje s leteckým snímkováním, které je o poznání méně finančně náročné (Pavelka, 2012).



Obrázek 2: Ukázka několikanásobné reflexe paprsku při průchodu vegetací (Tamás, 2010a).

2.2.2 Mobilní skenování

Mobilní skenování je v mnoha ohledech velmi podobné mapování leteckému. Rozdíl je především v upevnění skenovacího zařízení na střeše automobilu, na rozdíl od letadla či helikoptéry. Díky této skutečnosti dochází k zachycení menšího rozsahu ploch. Je to však zařízení přesnější, mobilnější, s většími možnostmi využití a menší finanční náročností. Vyskytuje se ve dvou režimech, a to tzv. „*stop-and-go*“ a „*on-the-fly*“. Podstatou prvního ze zmíněných režimů je statické mapování. Jakmile dojde k pořízení snímku, vozidlo se přesune a proces se následně opakuje. Druhý režim naopak skenuje po celou dobu pohybu vozidla, čímž se podobá leteckému mapování (Tamás, 2010b).


```
0 1 1
0 1 0
1 0 0
1 0 1
1 1 1
1 1 0
4 0 1 2 3           {začátek seznamu stěn}
4 7 6 5 4
4 0 4 5 1
4 1 5 6 2
4 2 6 7 3
4 3 7 4 0
```

Hlavní výhodou textových souborů při ukládání mračna bodů je možnost jejich zápisu a čtení. Jejich nevýhodou oproti binárním souborům představuje delší doba načítání a jejich velikost.

2.3.2 Binární soubory

Druhým typem výstupů vycházejícím ze 3D skenerů jsou binární soubory. Stejně jako textové soubory jsou i tyto považovány za jedny ze základních způsobů ukládání mračna bodů. Přes to, že i zde je možnost vytvoření vlastní struktury, zpravidla se využívají již vytvořené struktury standardizovaných formátů.

Zástupcem binárních souborů je formát LAS (LASer). Obdobně jako PLY, i LAS se skládá z hlavičky a určitých údajů týkajících se jednotlivých bodů v mračnu. Jedná se např. o odrazivost či klasifikaci, jež jsou hojně využívány. Za velkou výhodou je považována schopnost formátu uchovávat i dodatečné informace o konkrétních bodech. LAS také odstraňuje zásadní nevýhodu PLY, kterou byla jeho rychlost. Tento formát dokáže pracovat mnohonásobně rychleji (LAS Specification - Version 1.3, 2010).

3 Využití mračna bodů

Předchozí kapitola se zabývala teoretickým úvodem k problematice mračna bodů. Bylo v ní definováno mračno bodů, jeho získání a uložení do základních struktur. Tato kapitola se bude naopak věnovat jeho využitím v reálném světě.

Málokdo ví, co si pod pojmem mračno bodů představit a přitom se s ním lidé setkávají prakticky denně. Jak už bylo řečeno v úvodu, mračna bodů mají své zastoupení téměř v každém oboru.

Metody pracující s mračny bodů jsou hojně využívány v medicíně. Zejména pak ve spolupráci s lékařskými přístroji, kterými jsou např. CT či magnetická rezonance, z jejichž výstupů mračno bodů vzniká. Samotné mračno bodů má zde pouze velmi nízké využití. Proto obvykle dochází k jeho přeměně na 3D povrch, který je následně upraven a poté vyrenderován tak, aby lékaři získali adekvátní výstup (Bibb, 2015).

Další oblast, v níž je možné se s mrakem bodů setkat, představuje tzv. georeferencování. Tento proces spočívá ve sjednocení klíčových bodů dvou map. Cílem je předat souřadný systém té, která jím nedisponuje. Postup je často aplikován na historické mapy, u kterých souřadný systém před georeferencováním chyběl.

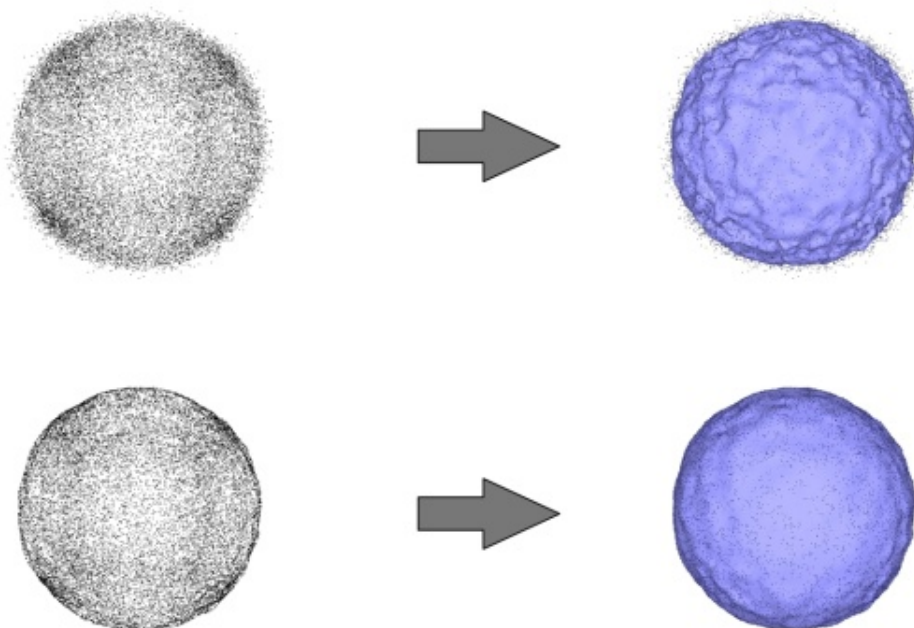
Mračno bodů je neméně důležité i pro reverzní inženýrství. Jeho podstata spočívá v odhalení základní myšlenky fungování zkoumaného objektu. Výstupem reverzního inženýrství je obnovení návrhu, který může být podkladem k vytvoření nového podobného produktu. Z tohoto důvodu je pro tento obor mračno bodů nepostradatelné a stává se tak samotným základem tohoto odvětví. Mračna bodů jsou zde velmi často využívána, a to především pro získání modelu tělesa, které již existuje. Příklad reverzního inženýrství můžeme najít třeba v architektuře, kde je používáno pro rekonstrukci 3D modelu skutečné budovy.

Zábava může mít mnoho podob a záleží pouze na člověku, které z nich dává přednost. Na zábavní průmysl jsou v současné době kladeny vysoké nároky. Neustálé zdokonalování výpočetní techniky umožňuje společností přicházet na trh s novými inovacemi. Spojení zábavního průmyslu s mračnem bodů se ukázalo jako dobrý krok, který položil základní kámen tvorbě animovaných filmů. Mračna bodů je též využíváno ve videoherních konzolích typu Kinect, jež snímají pohyb před sebou a převádí jej v interakci na obrazovce.

4 Problematika rekonstrukce 3D povrchů

Problematika rekonstrukce 3D povrchů je v současné době stále diskutovanějším tématem, neboť dochází ke snižování nákladů na výrobu skenovacích zařízení, čímž se stávají pro mnohé uživatele dostupnějšími. V celé řadě vědních oborů se současně zvyšuje jejich důležitost. Jedná se o zařízení, která prozatím není možné substitovat. Získanému 3D mračnu bodů je pro jeho další užití nutné zrekonstruovat povrch. Cílem rekonstrukce je nalezení množiny polygonů, jež bude co nejpřesnější napodobeninou zkoumaného tělesa v reálném světě. Pro rekonstrukci je optimální mít hustou síť bodů v mračnu, tzn. bez šumu (Hoppe, 1994).

Ideálnímu stavu se po získání množiny bodů ze skeneru přiblížíme jen velice těžce. Mnohem častěji se stává, že mračno obsahuje velké množství nedostatků. Tyto chyby je možné odstranit již existujícími algoritmy, které tak umožňují mračno bodů opravit. Jedná se především o odstranění výše zmíněného šumu či dopočítání chybějících bodů ve špatně naskenovaných místech. Mimo tyto kroky se neustále usiluje o minimalizaci spotřeby výpočetní paměti a doby trvání výpočtů, zatímco se zvyšuje přesnost zrekonstruovaného povrchu k předloze ve skutečném světě (Marton, Rusu a Beetz, 2009).



Obrázek 3: Nejtypičtější problém při rekonstrukci povrchu - šum. Ve spodní části pak můžeme vidět jeho vyhlazení (Marton, Rusu a Beetz, 2009).

Samotná rekonstrukce se skládá z následujících čtyř kroků:

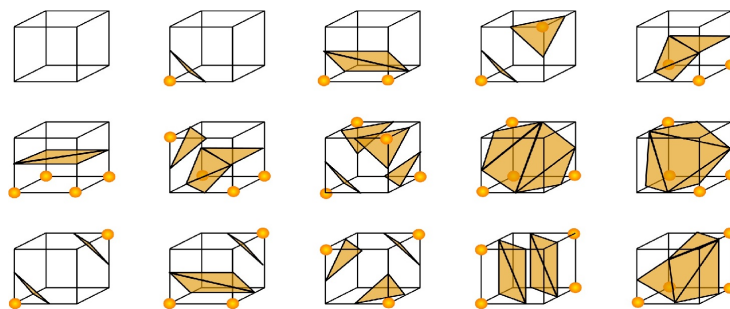
- V úvodu je potřeba provést samotnou analýzu mračna bodů, jež bylo získáno na vstupu. Podstatou je zjistit základní vlastnosti množiny mezi které patří např. výpočet průměrné vzdálenosti mezi body či určení ohraničujícího tělesa.

- Dále dochází k využití informací získaných z předchozího kroku a přiblížení mračna co možná nejlíže k ideálnímu stavu. Na základě průměrné vzdálenosti mezi body lze např. zjistit hustotu a šum mračna. Tyto problémy se následně snaží odstranit.
- Následuje výpočet normál všech bodů mračna a jejich orientace, která se zpravidla odvozuje podle geometrického středu tělesa. Většinou se jedná o složité výpočty. U příliš členitých povrchů těles se pak mohou objevovat chyby.
- Závěr rekonstrukce je spojen s výpočtem trojúhelníkové sítě, k tomuto výpočtu se využívají všechny dostupné údaje o vstupním mračnu bodů. Výslednou síť nakonec vrátíme jako seznam orientovaných trojúhelníkových polygonů. Jejich orientace je důležitá, protože nám při vykreslování tělesa dovolí vykreslit o polovinu polygonů méně a tím ušetřit jak paměť stroje, tak trvání celého procesu.

5 Analýza současných metod pro rekonstrukci povrchů

5.1 Marching cubes

Metoda Marching cubes byla vyvinuta zejména pro potřeby medicíny. Využívá se v lékařských přístrojích, kterými jsou především počítačová tomografie (CT) a magnetická rezonance (MR). Algoritmus projde trojrozměrné vstupní pole a rozdělí jej na krychle. Poté pracuje s jednotlivými krychlemi a u každého vrcholu rozhodne, zda patří dovnitř či vně. Podle jeho orientace mu pak přidá hodnotu 0 nebo 1. Nulou se označují body, jež jsou mimo dané těleso. Bodům, které se nacházejí uvnitř je přiřazena hodnota jedna. Jelikož má krychle osm vrcholů, získáme osmicifrné binární číslo. Krychle nabízí 256 různých kombinací. Ty jsou zúženy do 15 možností (další vznikají jejich rotací). Za pomoci zjištěných informací pak ze seznamu vybereme příslušnou reprezentaci povrchu a vložíme ji do výsledné polygonální sítě (Lorenson a Cline, 1987).



Obrázek 4: Možné výsledky povrchu pro jednu z krychlí algoritmu marching cubes (Lorenson a Cline, 1987).

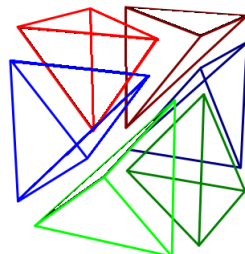
Nespornou výhodou této metody je schopnost vysokého rozlišení výsledné polygonální sítě. Negativa získává možnou produkcí děr či množstvím malých trojúhelníků.

5.2 Marching tetrahedra

Tento postup vhodně rozšiřuje metodu Marching cubes z předchozí sekce. Algoritmus opět rozdělí trojrozměrné vstupní pole na krychle, každou krychli pak na dalších šest čtyřstěňů. Dále se postupuje obdobně jako u Marching cubes. Prochází se všechny body čtyřstěňu a každému je přiřazena hodnota na základě polohy bodu vůči tělesu (0 - bod se nachází vně, 1 - bod se nachází uvnitř). Podle těchto údajů je vybrána plocha ze seznamu 16 možností. Vhodná plocha se následně vloží do výsledné polygonální sítě (Treece, Prager a Gee, 1998).

Pozitivní na tomto postupu je možnost použití i pro data v nepravidelné mřížce a vyřešení výskytu děr, které vznikaly v algoritmu Marching Cubes. Nevýhodou

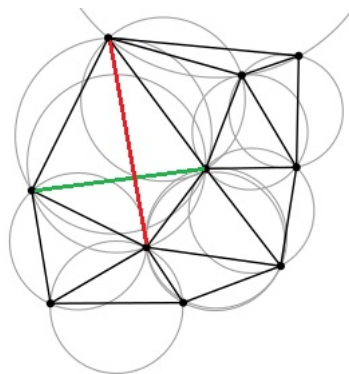
zůstává množství malých trojúhelníků, jež díky rozdělení krychle na čtyřstěny ještě vzrostlo.



Obrázek 5: Ukázka rozložení krychle na čtyřstěny (Bourke, 1994).

5.3 Delaunay triangulation

Triangulace byla původně vytvořena výhradně pro práci ve dvoudimenzionálním prostoru. Postupným vývojem však došlo k zobecnění Delaunayho triangulace z 2D prostoru na N -dimezionální prostor. V počítačové grafice se uplatňuje především metoda určená pro trojrozměrný prostor. Pro představu si nejdříve velice zjednodušeně popíšeme algoritmus ve 2D. Vezmou se tři body a proloží se jimi kružnice. Pokud uvnitř kružnice neleží žádný bod, vytvoří se trojúhelník. Pokud ano, je potřeba použít odlišnou trojici bodů. Problém může nastat při rozdělování čtyřúhelníku na dva trojúhelníky. V takovém případě se postupuje dle pravidla pro minimalizaci úhlů. Dochází tedy k rozdělení takové dvojice úhlů, jež svírají vyšší hodnotu.



Obrázek 6: Ukázka algoritmu ve 2D prostoru. Při rozhodování mezi červenou a zelenou hranou, bude podle pravidla pro minimalizaci úhlů upřednostněna zelená hrana (Maur, 2002).

Hlavní myšlenka Delaunayho triangulace je samozřejmě zachována i ve vyšších dimenzích. S přidáním dalších rozměrů přibývají i situace, se kterými mohou být spojeny i určité problémy. Abychom tyto situace ve 3D zabezpečili, stačí k podmínce ze dvourozměrného prostoru přidat jednu nebo dvě další. Tyto situace napomáhá řešit např. omezení délky hran potenciálního polygonu (Maur, 2002).

5.4 Ball pivoting

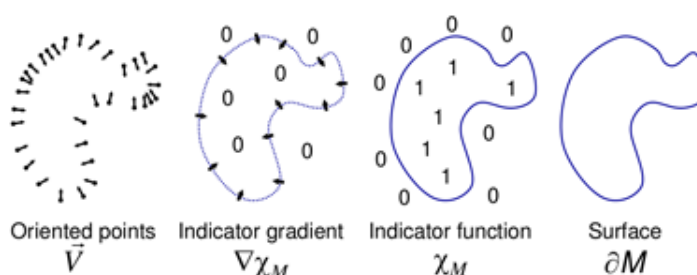
Princip této metody je opět velice jednoduchý a je podobný Delaunayho triangulaci, která je popsána v předchozích řádcích. Po vložení vstupní množiny je ze tří bodů vytvořen trojúhelník, který je považován za trojúhelník počáteční. Kolem hran tohoto obrazce rotuje koule, jejíž velikost stanoví sám uživatel. Jakmile se koule dotkne dalšího bodu, připojí nový trojúhelník. Tento postup se opakuje, dokud koule neotestuje všechny hrany obrazce. Pokud jsou již všechny hrany ověřeny a stále existují body, které nebyly testovány, zvolí se nový počáteční trojúhelník. Proces se následně opakuje. Výhodou této metody je nízká spotřeba paměti a malá časová náročnost (Bernardini a spol., 1999).

5.5 Poisson surface reconstruction

Poslední algoritmus pro rekonstrukci povrchu, kterým se budu ve své práci zabývat, se značně odlišuje od těch předchozích. Specifický je především přístup ke zpracování vstupních dat. Tato metoda nepřístupuje ke každému bodu množiny samostatně, ale pracuje s ní jako s celkem. Tento postup jí napomáhá být odolnou vůči mračnům, která obsahují značný šum či nerovnoměrné rozložení.

Funkce probíhá ve 4 funkcích:

- výpočet normál jednotlivých bodů,
- přiblížení gradientu indikátorové funkce pomocí normál,
- výpočet indikátorové funkce z gradientu z předchozího bodu,
- extrakce polygonové sítě, která reprezentuje hledané těleso (Kazhdan, Bolitho a Hoppe, 2006).



Obrázek 7: Ilustrace jednotlivých kroků Poissonovy metody (Kazhdan, Bolitho a Hoppe, 2006).

6 Metodika

V první části své bakalářské práce jsem se zabýval zejména teorií týkající se mračna bodů. Nyní již čtenář ví, co mračno bodů je, jakým způsobem se získává a následně ukládá. Stručně bylo popsáno i jeho využití v reálném světě. Práce se dále zaměřuje na problematiku rekonstrukce povrchů a některé jeho metody.

Pro rekonstrukci povrchu z mračna bodů je důležité použít správný algoritmus. K výběru adekvátního algoritmu nám napomohou znalosti nabyté právě v první části práce. Další neméně důležitý krok představuje zvolení programovacího jazyka, který by měl být schopen pracovat jak s převzatou částí kódu, tak současně vytvořit grafické rozhraní, jež je pro uživatele zpravidla nejpřehlednější.

Na teoretickou část navazuje část praktická, tzn. vlastní práce na projektu. Cílem bylo vytvořit takovou aplikaci, jež bude schopna zrekonstruovat povrch na základě vstupní množiny bodů. Přijal jsem určitá rozhodnutí týkající se vzhledu a funkcí vyvíjené aplikace. Ta je založena především na jednoduchosti designu a praktickém ovládní, jež zvládne běžný uživatel. Následovalo programování zvolených funkcí vedoucích k rekonstrukci povrchu. Do aplikace bylo samozřejmě nutné zavést i základní operace, kterými jsou např. načítání či ukládání souboru.

Dále proběhlo testování všech funkcí, zabezpečení aplikace od předčasného ukončení a odstranění dalších chyb. Všechny funkce projektu jsou popsány a doplněny o informace týkající se přesného fungování.

V závěru práce se nachází diskuze, v níž jsem aplikaci zhodnotil. Je zde popsán výsledek, kterého bylo při vytváření celé práce dosaženo a jeho přínos a porovnání s ostatními aplikacemi pro rekonstrukci povrchu z mračna bodů.

7 Technologie

K práci s mračnem bodů a jeho rekonstrukcí se mi zdála nejvhodnější knihovna Point Cloud Library, která se těmito úkoly zabývá již několik let. Pro vytvoření aplikace jsem tak hledal programovací jazyk, který by byl ve vzájemné interakci s touto knihovnou. Skvělým kandidátem se stal objektově orientovaný jazyk C++, který dokáže s PCL výborně spolupracovat. Za pomoci další knihovny s názvem Qt jsem vytvořil jednoduché a efektivní grafické uživatelské rozhraní.

7.1 C++

V roce 1985 vznikla rozšířením jazyka C verze, kterou již lze považovat za řádný objektově orientovaný programovací jazyk. Známe ji pod názvem C++. Jazyk C++ se postupem času neustále zdokonaloval. Současná verze je dnes stejně úspěšná jako její předchůdce. Hlavní výhodou je téměř úplná kompatibilita C++ s C, která programátorům usnadnila přechod k objektově orientovanému programování a zaručila použití již existujících kódů napsaných v jazyce C (Virus, 2011).

C++ se stalo jedním z nejdůležitějších programovacích jazyků 90. let 20. století a jeho oblíbenost neklesala ani po roce 2000. Jazyk C++ navazuje na tradici výkonného, uceleného, rychlého a přenositelného jazyka, kterou vybudoval jeho předchůdce. Objektově orientované vlastnosti umožňují jazyku C++ novou metodiku programování, jež byla navržena tak, aby si poradila se stupňující se obtížností moderních úkolů. Jeho nově rozšířené možnosti šablon přinášejí další způsob programování, kterým je obecné programování. O jazyku C++ můžeme tedy říci, že spojuje tři oddělené programovací tradice:

- tradici procedurálního jazyka reprezentovanou C,
- tradici objektově orientovaného jazyka reprezentovaného rozšířením C o třídy C++,
- obecné programování podporované šablonami jazyka C++ (Gregoire, Solter a Kleper, 2011).

7.2 Qt

Qt představuje knihovnu programovacího jazyka C++. Není však určena výhradně pro tento jazyk, ale mohou ji využívat i jazyky jiné, např. Python, Perl, Java. Jedná se o multiplatformní nástroj, díky němuž je možné vyvíjet konzolové či GUI aplikace. Obrovskou výhodou je přehledně zpracovaná dokumentace a také vývojové programy. Aplikace vytvořené pomocí Qt používají nativní vzhled operačního systému (Blanchette a Summerfield, 2008).

7.3 Point Cloud Library

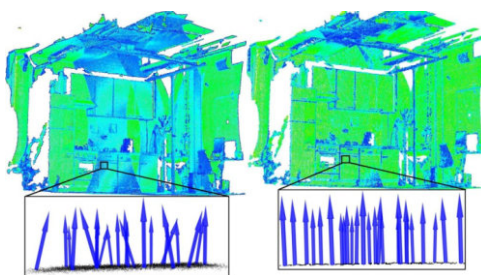
Point Cloud Library (neboli PCL) je rozsáhlý otevřený projekt pro 2D či 3D zpracování obrazu a mraku bodů. PCL obsahuje nespočet nejmodernějších algoritmů včetně filtrování, odhadování funkcí, rekonstrukce povrchu, zapisování, přizpůsobování a segmentace. Tyto algoritmy mohou být použity např. pro vyčištění šumu v datech, spojení 3D mraku bodů, rozdělení jednotlivých částí obrazu či vytvoření povrchu z mraku bodů a jeho zobrazení.

PCL je taktéž multiplatformní, může být úspěšně zkompileován a spuštěn na operačních systémech Linux, MacOS, Windows i Android/iOS. Pro zjednodušení vývoje je PCL rozdělena do několika dílčích kódovacích knihoven, které mohou být kompilovány samostatně (PCL, 2014).

7.3.1 Moving Least Squares

Některé nesrovnalosti v datech, způsobené např. chybami při měření z malé vzdálenosti, jsou velmi komplikované na odstranění. Pro vytvoření úplných modelů se musí brát ohled nejen na hladký povrch, ale také na absorpci dat modelu. V situacích, kdy nebylo možné získat další skeny, vznikl algoritmus pro převzorkování dat (Moving Least Squares, 2014).

Klasická převzorkovací metoda, jež vznikla v počítačové grafice, našla své uplatnění i v robotice a mluví se o ní jako o Moving Least Squares (MLS). MLS poskytuje interpolaci povrchu ze získané množiny bodů přidáváním polynomů vyššího řádu mezi dva sousední body. Po provedení převzorkování se mohou opravit malé chyby a vyhladit se tzv. „dvojitě stěny“, které jsou výsledkem vícenásobného skenování (Rusu, 2009).



Obrázek 8: Na levé straně můžeme vidět mračno bodů obsahující šum. Vedle je možno shlédnout normály poté, co prošly algoritmem Moving Least Squares (Rusu, 2009).

Algoritmus dokáže rozpoznat v mračnu díry, jež vznikly chybným skenováním. Ty jsou detekovány na základě analýzy úhlů, které svírají jednotlivé sousední body. K zakrytí těchto děr jsou v jejich místech vytvořeny body nové (Rusu, 2009).

7.3.2 Greedy surface triangulation

Tato metoda vytváří seznam bodů, ze kterých může být sestaveno těleso, tzv. „krajní“ body. Do tohoto seznamu jsou přidávány body sousední, dokud nejsou

všechny spojeny. Tomuto algoritmu nedělají problém ani neuspořádaná data, která mají více společných částí, neboť obvykle vycházejí z jednoho nebo více skenů. Nejlepšího výsledku je dosaženo, pokud je vstupní množina bodů vyhlazena a existuje zde plynulý přechod mezi částmi s odlišnou hustotou bodů (Greedy surface triangulation, 2014).

8 Vlastní práce

V této kapitole jsem popsal veškerou implementaci zvoleného algoritmu aplikace, která byla vytvořena na základě mé bakalářské práce. Jak již bylo zmíněno dříve, aplikace je programována v jazyce C++ za podpory knihoven Qt a PCL. Následný popis programu je strukturován chronologicky do sekcí, jež odpovídají hlavním funkcím, s nimiž může uživatel pracovat po zapnutí programu.

8.1 Načítání souboru

Prvním krokem po spuštění programu by zcela intuitivně mělo být načtení mračna bodů. Program uživateli neumožní provést žádnou jinou operaci, dokud nemá soubor k dispozici, neboť by neměl s čím pracovat.

Aplikace je napsána tak, aby načetla výhradně soubory s koncovkou `.ply`, která je standartním výstupním formátem ze 3D skenerů. Struktura tohoto formátu je velice jednoduchá a bude-li uživatel znát pár základních pravidel, může si mračno bodů stvořit i sám.

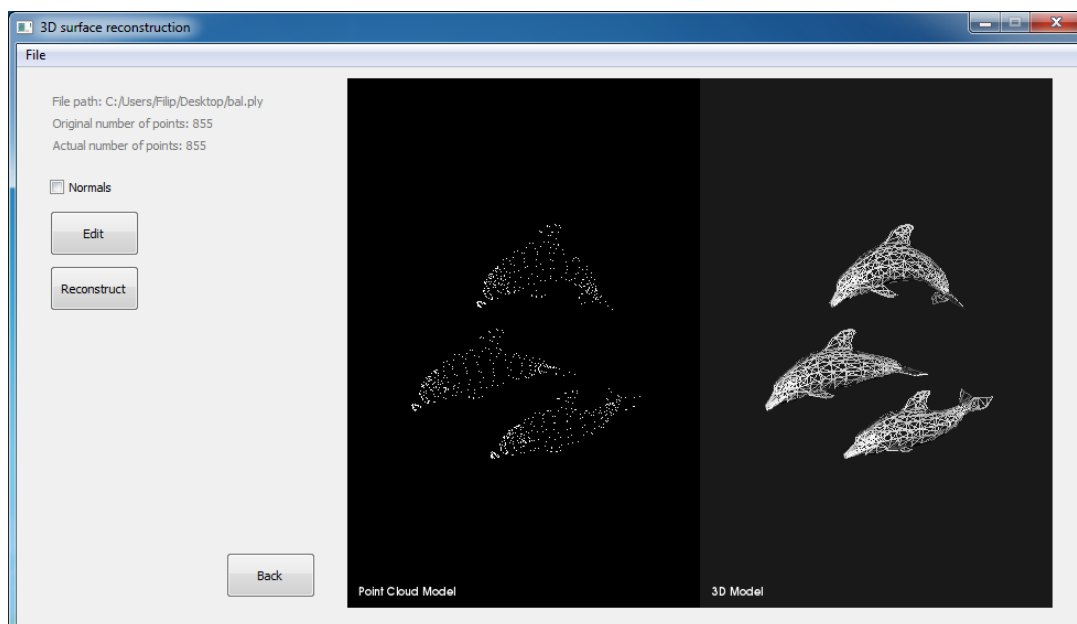
Postup načítání mračna bodů je velmi jednoduchý. Po kliknutí na položku *Open file* v nabídce *File* se otevře dialogové okno, které zobrazuje pouze soubory s koncovkou `.ply`, jiné tedy vybrat ani nelze.

Na konci tohoto procesu tak dojde k úspěšnému načtení mračna bodů.

8.2 Zjištění základních údajů a vizualizace

Druhou fází, jež následuje bezprostředně po načtení množiny bodů do programu, je zjištění základních údajů a následná vizualizace. Tento krok provede aplikace sama, uživatel jej nemůže žádným způsobem ovlivnit. Největší změna, které si lze ihned povšimnout, je zobrazení mračna bodů v pravé části okna aplikace. S tímto mračnem se již dá v reálném čase pohybovat, otáčet či jej přibližovat. Uživatel si tak může udělat dokonalou představu o tom, jak načtená množina vypadá. Pokud byl načten soubor, u něhož již rekonstrukce proběhla, pak se vedle vizualizace mračna bodů taktéž zobrazí model tělesa, které množina reprezentuje. Při manipulaci s mračnem bodů dochází současně ke změnám a přesunům daného modelu. Uživatel tak může tyto dva obrazy neustále porovnávat a má naprostou jistotu, že se kamera v obou oknech nachází na stejné pozici.

Načtení mračna bodů spouští funkci, jež zobrazí uživateli o mračnu ty nezákladnější údaje. V aplikaci je lze nalézt v levé horní polovině okna. Jedná se o samotný název souboru, cestu k souboru, počet bodů, které obsahuje mračno uložené v souboru a taktéž aktuální počet bodů množiny, se kterou je právě pracováno. Po načtení souboru budou tyto dvě hodnoty samozřejmě totožné. Lišit se začnou až v situaci, kdy bude s mračnem určitým způsobem manipulováno. Samotné načtení množiny bodů také uživateli odemyká všechny funkce, kterými program disponuje, např. tlačítko pro rekonstrukci povrchu mračna bodů či možnost uložení výstupu.



Obrázek 9: Zobrazení aplikace po načtení souboru bal.ply, který obsahuje mračno bodů.

8.3 Zobrazení normál

Načtení mračna bodů tedy uživateli umožnilo odemknout všechny funkce aplikace. Za nejjednodušší funkci považují zobrazení normál. Nejdříve je potřeba vyjasnit si, co značí normála v trojrozměrném prostoru. Následně se budu zabývat jejím vlivem na chod programu.

Obecně lze normálu definovat jako přímku, jež je kolmá na prostor, což znamená, že je možné určit normály tělesa v jeho jednotlivých bodech. Vektor určující směr normály se nazývá normálový vektor. V počítačové grafice je nepostradatelným ukazatelem pro osvětlování jelikož podává informaci o úhlu, který v daném bodě svírá paprsek s povrchem. Normálové vektory jsou však také velmi důležité pro tvorbu povrchu tělesa.

Pokud uživatel programu potřebuje z jakéhokoliv důvodu normály zobrazit, stačí zaškrtnout položku *Normals*. Program je schopen si sám normály spočítat a následně je také zobrazit. Není potřeba vkládat žádná další data, aplikace výpočty provádí ze vstupního mračna bodů.

Tyto normály jsou následně přidány do vizualizace množiny bodů.

8.4 Úprava mračna bodů

Pokud má být rekonstrukce povrchu provedena kvalitně, pak je nutné přiblížit mračno bodů co možná nejvíce jeho ideálnímu stavu. K tomu uživateli pomůže algoritmus Moving Least Squares, který je součástí programu.

V situaci, kdy je potřeba redukovat šum či jinak poupravit vstupní množinu bodů, stačí kliknout na tlačítko s názvem *Edit*. Zobrazí se okno, ve kterém je nutno nastavit prohledávací rádius funkce. Tato hodnota se liší v závislosti na rozložení mračna bodů. Pokud uživatel pracuje s množinou, jejíž body se od sebe nacházejí ve větších vzdálenostech, pak je potřeba nastavit hodnotu vyšší. Standardně je zde však nastavena univerzální hodnota, od které se může uživatel určitým způsobem odrazit.

Dokončit operaci umožňuje tlačítko *Finish*. Kliknutí na něj představuje pro program signál, aby provedl vše potřebné. Změny provedené na mračnu bodů se téměř okamžitě projeví ve vizualizační části programu. Obměnou může rovněž projít položka *Actual number of points* v informační části, která značí aktuální počet bodů v pracovní množině.

8.5 Rekonstrukce povrchu

Nejdůležitější část celé aplikace představuje samotná rekonstrukce povrchu z mračna bodů. Spuštění této funkce má téměř totožný postup, jaký byl popsán v předchozí sekci. Nejprve je potřeba v hlavním okně aplikace stisknout tlačítko s názvem *Reconstruct*, které uživateli otevře nastavení jednotlivých parametrů před použitím algoritmu na rekonstrukci.

- *Search Radius* určuje v jaké vzdálenosti od bodu, kterým algoritmus právě prochází, se budou připojovat další body. Tímto algoritmem lze prakticky určit maximální délku hrany každého trojúhelníku.
- Položky *MU* a *Maximum Nearest Neighbors* nastavují parametry sousedních bodů. První parametr určuje maximální přijatelnou vzdálenost zvažovanou pro určitý bod, a to vzhledem ke vzdálenosti bodu nejbližšího. Druhý parametr pak specifikuje, kolik sousedních bodů ke každému bodu by měl algoritmus vyhledat.
- Minimální a maximální úhel každého trojúhelníku se nastavuje pomocí *Minimum Angle* a *Maximum Angle*, kde první parametr není vždy garantován, zatímco druhý, tedy maximální úhel, je dodržen vždy.
- Poslední dvojicí, která může výsledek ještě upravit je *Maximum Surface Angle* a *Normal Consistency*. Tyto parametry se dokáží vypořádat s případy, kdy se v mračnu nacházejí ostré hrany či rohy. Algoritmus dokáže tyto chyby ošetřit díky normálám, jež si sám spočítá. I přesto, že se body nacházejí dostatečně blízko sebe, nemusí dojít k jejich spojení. Tato situace nastane, pokud úhel, který svírají jejich normály je větší než stanová hranice.

I u těchto parametrů jsou nastaveny počáteční hodnoty. Každé mračno bodů je ale specifické, a proto je potřeba tyto údaje měnit s ohledem na konkrétní situaci. V případě, že je vše nastaveno, nezbyvá nic jiného než použít tlačítko *Finish*. V okně pro vizualizaci by se měla následně objevit požadovaná rekonstrukce povrchu.

8.6 Krok zpět

Uživatel nemusí být s výstupem algoritmu spokojen. Může ho také zajímat, jak mračno bodů zareaguje na změnu toho či onoho parametru. Existují dvě možnosti, jak tuto situaci řešit. Množinu bodů lze opětovně načíst a provést na ní pouze dílčí změny. Druhou možností je využít tlačítko *Back*, které jej vrátí o jeden krok zpět, nikoliv však opakovaně. Vždy je možná pouze jedna změna.

8.7 Ukládání souboru

Provedené změny v mračnu bodů je samozřejmě také potřeba uživatelem uložit. Pokud by se tak nestalo, změny na mračnu by byly ztraceny a uživatel by s nimi později již nemohl pracovat.

Ukládání se provádí kliknutím na položku *Save file* v nabídce *File*, která uživateli otevře dialogové okno. Zde je potřeba soubor pojmenovat a dále zvolit cestu do složky, do níž má být soubor uložen. Soubor je automaticky ukládán ve formátu *.ply* a není možné jej uložit jiným způsobem.

9 Možná rozšíření

Jak už sám název kapitoly napovídá, v této sekci se budu věnovat možným rozšířením a vylepšením, které by mohly do budoucna zvýšit využitelnost aplikace, jež se stala součástí mé bakalářské práce.

9.1 Typy souborů

Již několikrát bylo zmíněno, že současná aplikace dokáže načítat a ukládat pouze do souborů typu PLY. Program by však měl daleko větší využití, pokud by byl kompatibilní i s odlišnými typy souborů. Za klíčové tedy považuji podporu dalších základních formátů (textových či binárních), které jsou určeny pro ukládání mračna bodů. Do budoucna by se také dalo uvažovat o souborech složitějších.

9.2 Možnosti rekonstrukce

Druhá možnost rozšíření, taktéž stěžejní, spočívá v přidání dalších algoritmů pro rekonstrukci povrchu mračna bodů. Uživatel by si tak před samotným procesem mohl vybrat, kterým způsobem chce mračno bodů rekonstruovat. Na základě výběru by následně došlo k provedení rekonstrukce. Kvalita aplikace by se zavedením tohoto kroku několikanásobně zvýšila. Uživatel by tak mohl algoritmy lehce porovnávat a díky tomu následně zvolit ten nejvhodnější.

9.3 Barevná mračna

Program je v současné době schopen pracovat pouze s bezbarvým mračnem bodů. Nemuselo by tomu tak však být pokud by aplikace byla schopna uchovávat barvy jednotlivých bodů v případě, kdy jsou součástí vstupních souborů. Toto rozšíření by zvýšilo nejen atraktivitu programu, ale také jeho využití a význam, protože i uchování barvy je jedním z klíčových parametrů mračna bodů.

9.4 Práce po rekonstrukci

Vytvořením polygonální sítě by pro uživatele práce na projektu nemusela skončit. Aplikaci by bylo možné využívat i po samotné rekonstrukci povrchu mračna bodů. Tím mám na mysli úpravu samotného povrchu, možnost přidávat barevné odstíny, textury či mnohá další.

10 Diskuze

V průběhu tvorby aplikace jsem musel velmi často řešit vznikající překážky, které značně zpomalovaly práci na projektu. Tato kapitola se bude věnovat jednotlivým problémům, které bylo potřeba odstranit.

10.1 Vývojové prostředí a operační systém

Aplikaci jsem vytvářel za pomoci programu Visual Studio 2010 (VS2010), který podporuje spolupráci s knihovnou PCL 1.6.0. U obou aplikací již existují i jejich novější nástupci. Bohužel však nedokázali splnit potřebné požadavky, a proto jsem se rozhodl využít dvou výše jmenovaných.

Instalaci těchto programů jsem zahájil na operačním systému Windows 8.1. Již při ní se vyskytl první zádrhel. Nebylo možné nainstalovat OpenNI framework, který byl součástí instalačního balíčku PCL a bez jehož pomoci by následně nemohlo dojít k vizualizaci testovacího mračna bodů v program VS2010. Na trhu je již k dispozici OpenNI 2, který je kompatibilní s novějšími operačními systémy. OpenNI 2 však bohužel neobsahuje knihovny, které bylo potřeba v mé práci využít.

Bylo tedy nutné operační systém přinstalovat. Zvolil jsem Windows 7 x64, na kterém již vše fungovalo zdánlivě bez potíží. Problém nastal až v okamžiku, kdy program Qt Creator nahlásil, že se pokouším zkompileovat 32bitový kód na 64bitovém operačním systému. Qt Creator představuje program, v němž jsem vytvářel grafické uživatelské prostředí vyvíjené aplikace.

Problém byl vyřešen konečnou reinstalací operačního systému na 32bitovou verzi Windows 7. Závěrečný úkon tak umožnil spojit knihovnu PCL s programem Qt Creator.

10.2 Verze Qt

Od samotného začátku jsem si byl jist, jakou verzi programu Visual Studia a knihovny PCL použít pro úspěšné spuštění aplikace. Váhal jsem však při výběru verze Qt, jakožto knihovny pro grafické uživatelské rozhraní programovacího jazyka C++.

Rozhodl jsem se pro stažení její nejnovější verze a to konkrétně knihovny Qt 5.2.0. Problém nastal ve chvíli, kdy jsem se pokusil připojit k programu Qt Creator knihovny pro práci s mračnem bodů. Aplikace mi tento krok bohužel neumožnila. Obdobně jako u instalace operačního systému či frameworku OpenNI, bylo vše několikrát přinstalováno, neboť nebylo zřejmé, co vyvolané chyby způsobuje.

Teprve až instalace Qt ve starší verzi, konkrétně verzi Qt 4.8.0, tento problém vyřešila a umožnila knihovny připojit.

10.3 Nedostatečná dokumentace PCL

Informace obsažené na webových stránkách *www.pointclouds.org* byly pro mou práci velkým přínosem. Uživatel zde nalezne opravdu rozsáhlou dokumentaci a několik velmi užitečných tutoriálů.

I přes tento fakt jsem však na těchto stránkách některé údaje postrádal. Sice jsem našel tutoriál věnovaný propojení programu Qt Creator s knihovny PCL, byl však nepřehledný a neúplný. Chybělo zde hned několik klíčových informací a zpravidla ani ty, které jsem měl k dispozici, nebyly z mého pohledu dostatečné.

10.4 Proměnná prostředí

Nejvíce mě potrápila chybějící proměnná prostředí v operačním systému, která musela být přidána ručně. Nutno říci, že tato chyba se velmi těžko odhalovala a nebylo vůbec jednoduché ji najít. Po jejím opravení a vyřešení ještě několika dalších menších zádrhelů, se program podařilo úspěšně zkompilovat.

10.5 Zhodnocení aplikace

Nutno zdůraznit, že aplikace nyní umožňuje uživateli pouze základní funkce pro práci s mračnem bodů, včetně jeho rekonstrukce. Existuje zde velký prostor pro zlepšování a další vývoj programu. V kapitole devět jsem uvedl možná rozšíření projektu, která sice nejsou nezbytně nutná, avšak aplikace by díky jejich realizaci získala nový rozměr.

I v základní formě je dle mého názoru přínos aplikace značný. Vyzdvihl bych především jednoduchou ovladatelnost, při níž běžnému uživateli nepřekáží žádné funkce, kterým by nerozuměl. Často se lze na trhu setkat s programy, které mají velké množství rozmanitých funkcí, čímž splňují požadavky a nároky odborníků v daném odvětví. Zapomíná se však na uživatele, kteří neznají problematiku až natolik do hloubky a informační technologie jsou pouze jejich koníčkem. Právě pro ně je tato aplikace, jež obsahuje pouze nutná nastavení pro schopnost rekonstrukce mračna bodů na povrch, určena. Dostupnost programu v kombinaci s neustálým snižováním cen 3D skenerů by mohla podnítit zájem lidí o tuto problematiku a vyvolat tak diskuzi zaměřenou na rozsáhlejší využívání programu.

Aplikace, jež vznikla k mé bakalářské práci, však nemusí nutně sloužit pouze k rekonstrukci povrchů malých objektů. Dala by se stejně tak dobře využít i k rekonstrukci budov a při správném nastavení dokonce k rekonstrukci daleko větších povrchů, kterými mohou být města či rozsáhlejší krajiny. Pokud by se dostala do rukou správné společnosti, jež se obdobnou problematikou zajímá, mohla by se proměnit ve velmi užitečného pomocníka.

11 Závěr

V první části své bakalářské práce jsem se zabýval teoretickým úvodem k mračnu bodů. V závěru práce bych rád shrnul průběh praktické části. Cílem bylo vytvořit aplikaci, pomocí níž je možné zrekonstruovat mračno bodů do podoby 3D povrchu. Vytýčený cíl jsem dle mého názoru splnil. Vytvořil jsem aplikaci, jež je jednoduchá, pochopitelná pro běžné uživatele a bez rušivých elementů.

Celý proces tvorby aplikace doprovázelo množství potíží, jež se týkaly verze operačního systému, programů a knihoven a jejich kompatibility. Problémy se též objevily z důvodů nedostatečné dokumentace a nebo chybějící proměnné prostředí v systému. Do budoucna by bylo možné provést několik rozšíření, díky nimž by vzrostla kvalita programu. Mezi ně patří přidání rekonstrukčních metod, uchování si barev po načtení souboru, možnost pracovat i s odlišnými typy souborů a postrekonstrukční proces.

Osobně se domnívám, že program má široké spektrum využití. Primárně je určen běžným uživatelům, avšak své uplatnění by mohl najít i u specializovaných firem. Obzvláště pokud by došlo k realizaci výše uvedených rozšíření.

Práce by také mohla zvednout informovanost i zájem běžných uživatelů k tomuto tématu. Domnívám se, že tato oblast informačních technologií ještě není zcela probádána a jakýkoliv impuls může pomoci.

Byl bych rád, kdyby má bakalářská práce sloužila jako základ práce diplomové, v níž by bylo velmi zajímavé v dané oblasti pokračovat a dále ji rozvádět.

12 Literatura

- BERNARDINI, FAUSTO A HOLLY RUSHMEIER *The 3D Model Acquisition Pipeline* [online]. [cit. 2015-05-15]. Dostupné z: http://graphics.cs.yale.edu/site/sites/files/3d_acquisition.pdf.
- BERNARDINI, FAUSTO, JOSHUA MITTLEMAN, HOLLY RUSHMEIER, CLÁUDIO SILVA, GABRIEL TAUBIN *The ball-pivoting algorithm for surface reconstruction* [online]. [cit. 2015-05-04]. Dostupné z: http://www.research.ibm.com/vistechology/pdf/bpa_tvce.pdf.
- BIBB, RICHARD *Medical modelling: the application of advanced design and rapid prototyping techniques in medicine*. 2. vyd., 2015, 500 s. ISBN 978-1-78242-300-3.
- BLANCHETTE, JASMIN A MARK SUMMERFIELD *C++ GUI programming with Qt 4*. 2. vyd. Upper Saddle River, N.J.: Prentice Hall, 2008, 718 s. ISBN 978-0-13-235416-5.
- BOURKE, PAUL *Polygonising a scalar field* [online]. [cit. 2015-05-19]. Dostupné z: <http://paulbourke.net/geometry/polygonise>.
- Fast triangulation of unordered point clouds*. [online]. [cit. 2014-12-09]. Dostupné z: http://pointclouds.org/documentation/tutorials/greedy_projection.php#greedy-triangulation.
- GREGOIRE, MARC R, NICOLAS SOLTER A SCOTT KLEPER *Professional C++*. 2. vyd. Indianapolis, IN: Wiley Publishing, Inc., 2011, 1072 s. ISBN 978-0-470-93244-5.
- HARTLEY, RICHARD A ANDREW ZISSERMAN *Multiple view geometry in computer vision*. 2. vyd. UK: Cambridge University Press, 2003, 655 s. ISBN 0-521-54051-8.
- HOPPE, HUGUES *Surface Reconstruction from Unorganized Points* [online]. [cit. 2015-05-01]. Dostupné z: <http://research.microsoft.com/en-us/um/people/hoppe/recon.pdf>.
- HRUBÝ, MARTIN *Geografické Informační Systémy (GIS) Studijní opora* [online]. [cit. 2015-05-15]. Dostupné z: <http://perchta.fit.vutbr.cz/vyuka-gis/uploads/1/GIS-final2.pdf>.
- JÄHNE, BERND, HORST HAUSSECKER A PETER GEISSLER *Handbook of computer vision and applications*. 3. vyd. San Diego: Academic Press, 1999, 623 s. ISBN 0-12-379771-3.
- KAZHDAN, MICHAEL, MATTHEW BOLITHO A HUGUES HOPPE *Poisson surface reconstruction* [online]. [cit. 2015-05-04]. Dostupné z:

- <http://research.microsoft.com/en-us/um/people/hoppe/poissonrecon.pdf>.
- LANDA, JAROMÍR *Automatizovaná detekce objektů v mraku bodů*. Disertační práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta, Ústav informatiky, 2014, 116 s..
- LAS Specification - Version 1.3* [online]. [cit. 2015-05-15]. Dostupné z: http://www.asprs.org/a/society/committees/standards/LAS_1_3_r11.pdf.
- LORENSEN, WILLIAM E. A HARVEY E. CLINE *Marching cubes: A high resolution 3D surface construction algorithm* [online]. [cit. 2015-05-03]. Dostupné z: <https://www.cs.virginia.edu/johntran/GLunch/marchingcubes.pdf>.
- MARTON, ZOLTAN CSABA, RADU BOGDAN RUSU A MICHAEL BEETZ *On fast surface reconstruction methods for large and noisy point clouds* [online]. [cit. 2015-05-01]. Dostupné z: http://ias.cs.tum.edu/_media/spezial/bib/marton09icra.pdf.
- MAUR, PAVEL *Delaunay triangulation in 3D* [online]. [cit. 2015-05-03]. Dostupné z: <https://www.kiv.zcu.cz/site/documents/verejne/vyzkum/publikace/technicke-zpravy/2002/tr-2002-02.pdf>.
- MEDEROS, BORIS, LUIZ VELHO A LUIZ HENRIQUE DE FIGUEIREDO *Moving Least Squares Multiresolution Surface Approximation* [online]. [cit. 2015-05-15]. Dostupné z: <http://w3.impa.br/~boris/sib2003.html>.
- MECHELKE, KLAUS, THOMAS P. KERSTEN A MAREN LINDSTAEDT *Comparative investigations into the accuracy behaviour of the new generation of terrestrial laser scanning systems* [online]. [cit. 2015-05-15]. Dostupné z: <http://archive.cyark.org/temp/hcuhanburgmechelkeetal2007.pdf>.
- NEDBAL, MAREK *Rekonstrukce povrchu chaotických množin bodů* [online]. Bakalářská práce. Masarykova univerzita v Brně, Fakulta informatiky, 2014 [cit. 2015-05-03]. Dostupné z: http://is.muni.cz/th/357293/fi_b_a2/thesis.pdf.
- PAVELKA, KAREL *Letecké laserové skenování v ČR a možnosti využití dat pro dokumentaci historické těžby nerostných surovin* [online]. [cit. 2015-05-18]. Dostupné z: <http://lfgm.fsv.cvut.cz/data/RIV/j%C3%A1chymovsko.pdf>.
- PLY Files an ASCII Polygon Format* [online]. [cit. 2015-05-15]. Dostupné z: <http://people.sc.fsu.edu/~jburkardt/data/ply/ply.html>.
- RUSU, RADU BOGDAN *Semantic 3D Object Maps for Everyday Manipulation in Human Living Envrionments* [online]. [cit. 2015-05-17]. Dostupné z: <http://files.rbrusu.com/publications/RusuPhDThesis.pdf>.
- RUSU, RADU BOGDAN *What is PCL?* [online]. [cit. 2014-12-09]. Dostupné z: <http://pointclouds.org/>.

- Smoothing and normal estimation based on polynomial reconstruction.* [online]. [cit. 2014-12-09]. Dostupné z: <http://pointclouds.org/documentation/tutorials/resampling.php#movingleast-squares>.
- TAMÁS, LOVAS *Data acquisition and integration, Airborne laser scanning* [online]. [cit. 2015-05-15]. Dostupné z: http://www.tankonyvtar.hu/hu/tartalom/tamop425/0027_DAI4/ch01s02.html.
- TAMÁS, LOVAS *Data acquisition and integration, Mobile laser scanning* [online]. [cit. 2015-05-15]. Dostupné z: http://www.tankonyvtar.hu/hu/tartalom/tamop425/0027_DAI4/ch01s04.html.
- TAMÁS, LOVAS *Data acquisition and integration, Terrestrial laser scanning* [online]. [cit. 2015-05-15]. Dostupné z: http://www.tankonyvtar.hu/hu/tartalom/tamop425/0027_DAI4/ch01s03.html.
- TREECE, G. M., R. W. PRAGER A A. H. GEE *Regularised marching tetrahedra: improved iso-surface extraction* [online]. [cit. 2015-05-03]. Dostupné z: <http://www.ks.uiuc.edu/Research/vmd/projects/ece498/surf/treece98regularised.pdf>.
- VIRIUS, MIROSLAV *Jazyky C a C++: kompletní průvodce*. 2. vyd. Praha: Grada, 2011, 367 s. ISBN 978-80-247-3917-5.
- ŽÁRA, JIŘÍ *Moderní počítačová grafika*. 2. vyd. Brno: Computer Press, 2004, 609 s. ISBN 80-251-0454-0.