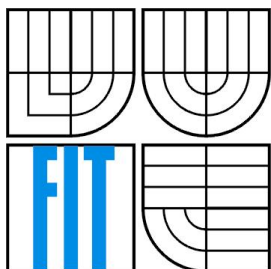




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ  
FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

# INTERAKTIVNÍ 3D MODELOVÁNÍ LIDSKÝCH TKÁNÍ NA ZÁKLADĚ CT/MR DAT

INTERACTIVE HUMAN TISSUES 3D MODELING BY CT/MR DATA

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Tomáš Satinský

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Kršek Přemysl, PhD.

BRNO 2008

## **Abstrakt**

Tato práce se skládá z teoretického rozboru, návrhu a implementace systému pro tvorbu trojrozměrných počítačových modelů tkání lidského těla. Tyto modely budou vytvářeny na základě dat získaných pomocí počítačové tomografie (CT) a magnetické rezonance (MR). Výstupem lékařských přístrojů založených na těchto principech je trojrozměrný rastr, který není příliš vhodný pro vizualizaci či vytváření reálných modelů tkání. Pro extrakci požadovaných tkání a vytváření jejich modelů se používají automatické nebo poloautomatické metody, kdy je nutná spolupráce odborníka na vytváření modelu. Tato práce je zaměřena na modelování lineárních struktur lidského těla jako jsou žíly a tepny, nervy, kanálky a další. Tyto struktury jsou totiž automatickými metodami problematicky rozpoznatelné a proto je vhodná asistence odborníka znalého lidské anatomie.

## **Klíčová slova**

počítačová grafika, anatomie, modelování, tkáně, počítačová tomografie, magnetická rezonance

## **Abstract**

This work concerns / deals with theoretical analysis, design and implementation of system for creation three-dimensional computer models of human body tissues. These models will be created by computer tomography (CT) and magnetic resonance (MR) data. Output of medical apparatuses based on these principles is a three-dimensional raster, which is not good for visualization and making real models. For required tissues extraction are used automatic or semiautomatic methods, where is necessary an operation of a specialist. This work is concerned with human body tissues with linear structure such as veins and arteries, nerves, bone canals etc. These structures are problematic recognizable by automatic methods. Therefore is convenient an operation of a specialist, which is adept in human anatomy.

## **Keywords**

computer graphics, anatomy, modeling, tissues, computer tomography, magnetic resonance

## **Citace**

Tomáš Satinský: Interaktivní 3D modelování lidských tkání na základě CT/MR dat, diplomová práce, Brno, FIT VUT v Brně, 2008

# Interaktivní 3D modelování lidských tkání na základě CT/MR dat

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením

Ing. Přemysla Krška, PhD. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Tomáš Satinský  
19.5.2008

©Tomáš Satinský, 2008.

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

# Obsah

Obsah.....	1
1 Úvod.....	2
2 Teoretický rozbor.....	3
2.1 Zdrojová CT/MR data.....	3
2.1.1 Data získaná na základě CT.....	3
2.1.2 Data získaná na základě MR.....	4
2.1.3 Struktura dat.....	6
2.1.4 Přístup k datům a jejich parametry.....	6
2.2 Zobrazení CT/MR dat.....	7
2.2.1 Volume rendering.....	7
2.2.2 Multiplanární zobrazení.....	10
2.2.3 Zobrazení pomocí hraniční reprezentace.....	12
2.3 Segmentace tkání.....	14
2.4 Základní koncepce programu.....	17
2.5 Zobrazování trojrozměrných dat.....	18
2.6 Vykreslování křivek.....	20
3 Návrh.....	22
3.1 Uživatelské rozhraní t.....	22
3.2 Architektura programu.....	24
4 Implementace .....	28
5 Závěr.....	30
Literatura.....	31
Seznam příloh.....	32

# 1 Úvod

Současná medicína disponuje úžasnými prostředky, které umožňují nahlédnout do lidského těla bez fyzického zásahu. Mezi principy, jež toto umožňují patří počítačová tomografie (angl. Computer Tomography, dále jen CT) a magnetická rezonance (angl. Magnetic Resonance, dále jen MR). Přístroje pracující na těchto principech poskytují data o pacientově těle ve formě trojrozměrného rastru. Každé políčko 3D rastru informuje o určitých vlastnostech pacientova těla v odpovídajícím místě.

To umožňuje zkoumat lidské tělo neinvazivní metodou s vysokou přesností, za živa a v místech, kde by to dříve nebylo možné bez porušení životních funkcí. Trojrozměrná rastrová data se ovšem těžce zobrazují tak, aby zobrazení pokrylo požadovanou oblast zájmu a ostatní informace byly potlačeny. Také sestavení reálných fyzických modelů vybraných tkání není možné přímo dat, jež poskytují dané přístroje.

Řešením je segmentace dat a následné vytvoření modelů tkání, jež chceme zkoumat. Tímto získáme pouze informace o požadované oblasti a zjednodušený model nám poskytne lepší možnost zobrazení či výroby reálného fyzického modelu. U tkání, jež při použití určité metody zobrazení vykazují výrazně jiné parametry než tkáně sousední je možné pro vytvoření modelu použít některou z automatických metod segmentace. Jiné tkáně je možné modelovat pouze za asistence odborníka znalého anatomie.

## 2 Teoretický rozbor

Tato kapitola se zabývá popisem trojrozměrných dat, jež nám poskytují současné medicínské přístroje založené na principu počítačové tomografie (CT) a magnetické rezonance (MR), jejich strukturou a možnostmi zobrazení. Dále jsou popsány možnosti, jak dané tkáně segmentovat a modelovat.

### 2.1 Zdrojová CT/MR data

CT/MR metody využívají pro vyšetření pacienta rentgenového záření (CT) nebo jevu magnetické jaderné rezonance (MR). Příslušná zařízení jsou ovládána výkonným počítačem, který provádí zpracování dat získaných při vlastním vyšetření. Výsledkem jsou digitální počítačová data, která diskrétně popisují rozložení fyzikálních hodnot (rentgenová hustota-CT, hustota protonových jader-MR) v nasnímaném objemu [1]. Přístroje postavené na principu CT a MR provádějí snímání vybrané oblasti pacientova těla.

#### 2.1.1 Data získaná na základě CT

Počítačová tomografie používá k vyšetření pacienta rentgenových paprsků, které jsou schopny částečně projít lidským tělem. Při průchodu paprsků dochází k jejich tlumení na podle vlastností tkání, kterými procházejí. Zdrojem rentgenových paprsků je tzv. rentgenka. Pokud rentgenkou prozáříme tělo a na opačné straně zaznameneáme intenzitu prošlého záření, tak získáme běžný rentgenový snímek (viz. obrázek 2.1). Problémem takového zobrazení je to, že neříká nic o tom, jak jsou zobrazené tkáně rozmístěny v prostoru, protože ty se na snímku překrývají.

Řešení nabízí počítačový tomograf. Ten provádí snímání požadované oblasti z více míst. Z takto získaných dat je pomocí náročných výpočtů získán „3D rentgenový snímek“. Výsledný 3D snímek udává rentgenovou hustotu v jednotlivých mikroobjemech snímaného objektu. Lze jej chápat jako sérii „řezů“ (viz obrázek 2.2).

Nevýhodou používání CT je - stejně jako u rentgenu - ozáření radioaktivním zářením. Dávka radiace, které je pacient při vyšetření vystaven, závisí na objemu zkoumané oblasti, počtu a typu snímání a požadované přesnosti a kvalitě zobrazení.

Na snímcích z počítačové tomografie jsou dobře odlišitelné tkáně, které různou měrou pohlcují rentgenové paprsky. Nejvíce záření pohlcují kosti a tak mají na snímcích z CT vysokou intenzitu.



Obrázek 2.1: Rentgenový snímek



Obrázek 2.2: Jeden řez z CT

### **2.1.2 Data získaná na základě MR**

Přístroje založené na principu magnetické rezonance využívají elektromagnetické vlnění s vysokou frekvencí spolu se silným magnetickým polem. Jádra různých atomů hmoty mají různou tzv. rezonanční frekvenci. Ta závisí na síle magnetického pole, v němž se prvek nachází a na vnitřní struktuře jádra. Různé atomy se tedy v daných podmínkách chovají různě, což se v MR využívá k jejich rozlišení. V reálných aplikacích je sledována rezonance jader vodíku, který se v lidském těle vyskytuje nejvíce.



Stejně jako u CT jsou po náročných výpočtech získána prostorová data. Hodnoty jsou závislé na mnoha proměnných. U magnetické rezonance je možné dosáhnout daleko vyššího rozlišení než u počítačové tomografie. Dále je možné použití kontrastní látky. Ta může na snímku zvýraznit tkáně, jež by byly jinak oproti svému okolí nerozlišitelné. Interval snímání je v současné době tak nízký, že je možné snímat například srdce v reálném čase.

Výhodu oproti počítačové tomografii je, že pacient není během zákroku vystaven žádnému radioaktivnímu záření. Kontraindikací vyšetření je přítomnost některých kardiostimulátorů, implantátů či kovových tělísek v těle pacienta. Dále jsou to rozsáhlá tetování ve vyšetřované oblasti či první trimestr těhotenství.

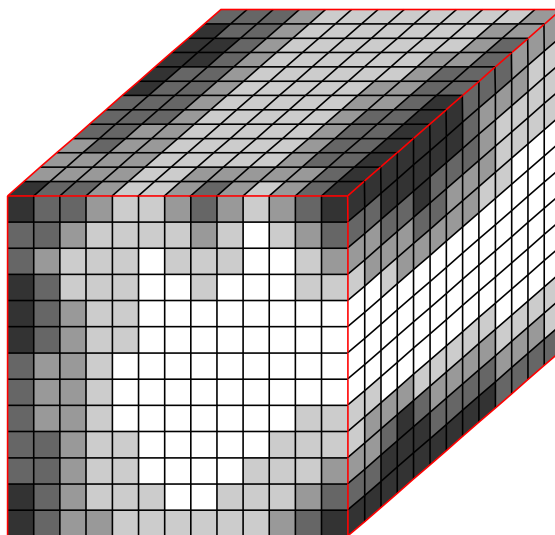
Na snímcích z magnetické rezonance jsou dobře odlišitelné tkáně s různým obsahem vodíku. Například kosti, které obsahují velmi málo vody mají na snímcích z MR velmi malou intenzitu.



Obrázek 2.3: Řez lidskou hlavou získaný pomocí MR [5].

### 2.1.3 Struktura dat

Data získaná na principu CT a MR jsou uložena v pravidelné ortogonální trojrozměrné mřížce. Z pohledu počítačové grafiky se jedná o trojrozměrný rastr. Každý prvek rastru (tzv. voxel) udává určitou informaci o odpovídajícím mikro-objemu ve snímaném objemu. K nasnímanému objemu je přiložena také informace o jeho rozměrech a orientaci.



Obrázek 2.4: Příklad 3D rastru s různými intenzitami voxelů

### 2.1.4 Přístup k datům a jejich parametry

Pro sdílení a přenos medicínských dat se používá standardní obrazový a komunikační formát DICOM 3.0 (Digital Imaging and Communication in Medicine). Výzkumná skupina „Medical group“, která funguje na FIT VUT v Brně má zajištěn přístup k potřebným datům ve spolupráci s Klinikou zobrazovacích metod u sv. Anny v Brně. Pro rychlý a interaktivní přenos dat ve formátu DICOM mezi jejím pracovištěm, radiologickými pracovišti a klinickými aplikačními pracovišti v nemocnicích v rámci ČR, je napojena na metropolitní síť MeDiMed (Metropolitan Digital Imaging System in Medicine). V akutních případech je možno získat požadovaná CT/MR data (nebo jiná obrazová data) ihned po provedení příslušného vyšetření, než stačí pacient opustit diagnostické pracoviště.

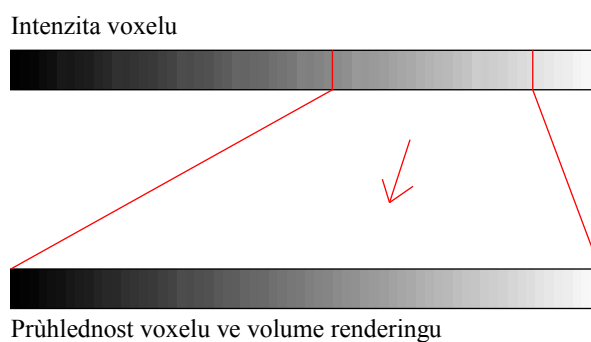
Pro načítání a manipulaci s daty je možné použít například knihovnu MDSTK, jež vznikla na FIT VUT v Brně. MDSTK umožňuje například načítání 2D řezů a 3D objemů ve formátu DICOM, jejich analýzu a filtraci a další.

## 2.2 Zobrazení CT/MR

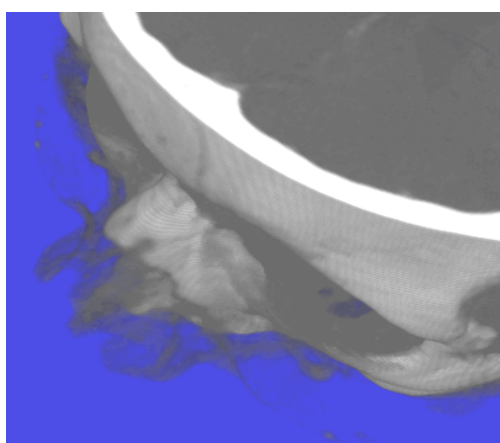
Objemová data se dají zobrazovat buďto přímo, nebo je možné zobrazit až vyextrahované tkáně, které chceme zkoumat.

### 2.2.1 Volume rendering

Mezi přímé metody patří různé metody tzv. „volume renderingu“, kdy je zobrazeno objemové těleso. V takovém případě je potřeba pracovat s průhledností, aby byly viditelné i voxely uvnitř tělesa. Určité voxely jsou tedy průhledné více, jiné méně. Různit se může také barva voxelů. Nejjednodušší způsob je ten, kdy je průhlednost voxelu lineárně závislá na jeho intenzitě. Také je možné mapovat na průhlednost pouze určitý interval intenzit. Chceme-li se zaměřit hlavně na kosti, pak můžeme pro získání průhlednosti použít pouze tu oblast intenzit, kde se vyskytují kosti (viz. obrázek 2.5).

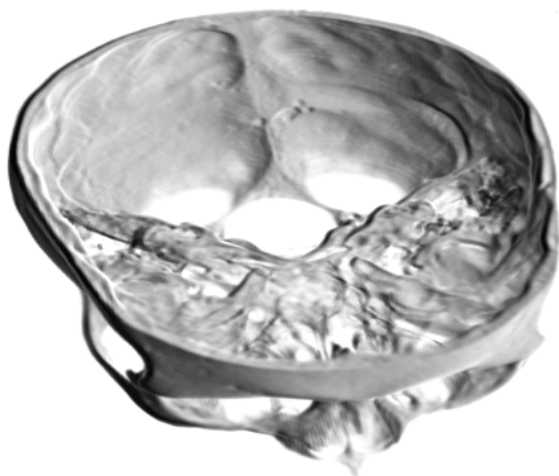


Obrázek 2.5: Příklad mapování intenzit na průhlednost.

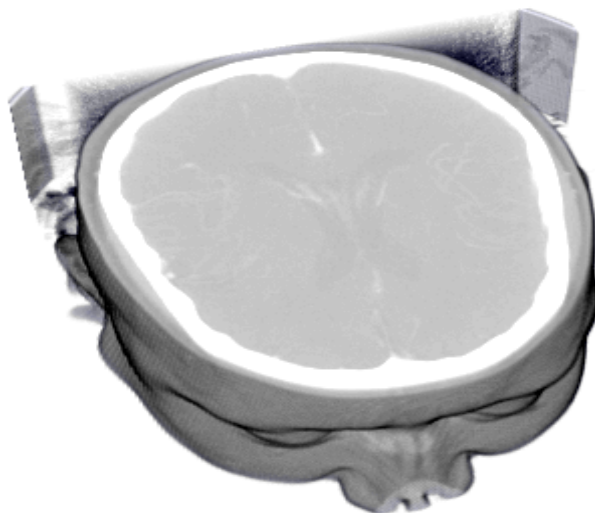


Obrázek 2.6: Vykreslování s mapováním intenzit na průhlednost i barvu voxelu.

Důležitým vodítkem pro lidskou orientaci v prostoru je také osvětlení a stínování zkoumaného objektu. Mluvíme-li o stínování, jedná se vlastně o stínování povrchu tělesa. Volumetrická data ale explicitně neobsahují žádné informace o površích v tělese. Výše zmíněným vykreslováním s potlačením určitých tkání ale iluze povrchu vzniká a tak je vhodné aby tyto povrchy byly stínované. Pro to je potřeba znát normálový vektor daného voxelu. Jak jej ale získat? Iluze povrchu vzniká tam, kde se rychle mění průhlednost (viz obrázek 2.6). Na základě změn intenzit v objemu je možné odhadnout pomyslný normálový vektor v daném místě a pomocí něho simulovat osvětlení, jež podpoří dojem prostorovosti a učiní výsledný obraz přehlednějším. Normálový vektor iluzorního povrchu lze tedy definovat jako normalizovaný gradient průhlednosti daného voxelu.

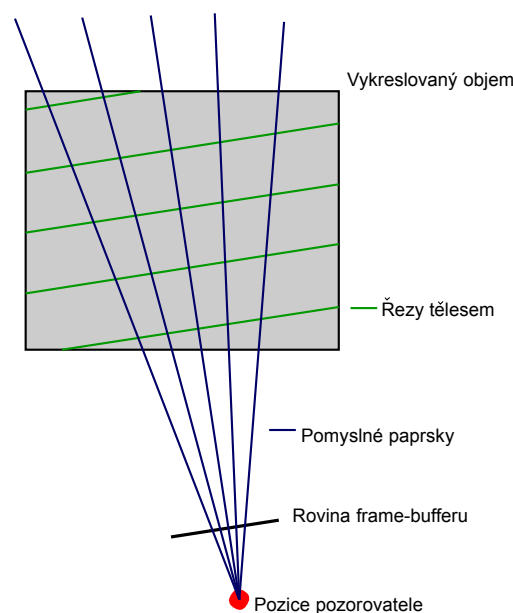


Obrázek 2.7: Volume rendering CT snímku lidské lebky se zaměřením na tkáně s vyšší intenzitou.  
Se simulovaným osvětlením.



Obrázek 2.8: Stejný snímek s mapováním celého rozsahu intenzit na průhlednost.  
Bez simulovaného osvětlení.

Dnešní grafické akcelerátory příliš nepodporují vykreslování voxelů, jsou optimalizovány pro vykreslování trojúhelníků. Z tohoto důvodu je potřeba nalézt takový způsob, kdy budou vykreslovaná volumetrická data zobrazována pomocí ploch. Jeden způsob vychází z tzv. ray-castingu, kdy jsou z pozice pozorovatele scény vysílány pomyslné paprsky přes každý pixel výsledného rastru (frame-buffer) a následně se testuje, zda protínají vykreslované těleso. Pracuje s řezy objemovým tělesem, jež jsou vždy natočeny k pozorovateli (viz. obrázek 2.9). Pro správné vykreslení průhlednosti se řezy vykreslují od nejvzdálenějšího k nejbližšímu. Při současných hodnotách operačních paměti grafických akcelerátorů je možné ukládat data přímo na kartě. Lze využít i tzv. 3D textur. Normály pro výpočet osvětlení můžou být počítány dynamicky za použití pixel-shaderu nebo můžou být předpočítány a uloženy na kartě.



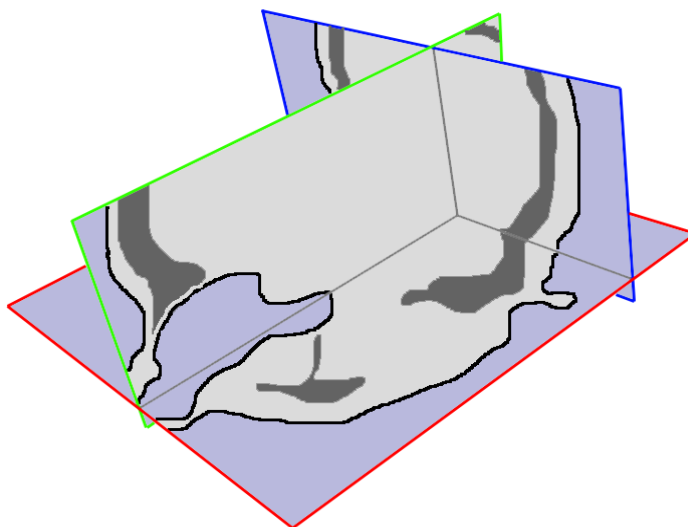
Obrázek 2.9: Zjednodušená verze ray-castingu pro zobrazení objemových těles.

Výhodou metod volume-renderingu je, že zobrazují celý objem v jediném okamžiku a tak je možné udělat si celkový přehled. Nevýhodou je nemožnost označit v datech určitý bod, například při vyznačování obrysů tkání a podobně. Dále není příliš možné zaměřit se na malou oblast v objemu a tu lépe prozkoumat.

## 2.2.2 Multiplanární zobrazení

Při multiplanárním zobrazení se vykreslují řezy objemem. Jelikož data jsou uložena v pravidelné pravoúhlé mřížce, tak se zdá být jako nejschůdnější zobrazovat řezy kolmé na osy objemu. Řezy musí být pohyblivé, aby bylo možné zkoumat či označit libovolné místo v objemu.

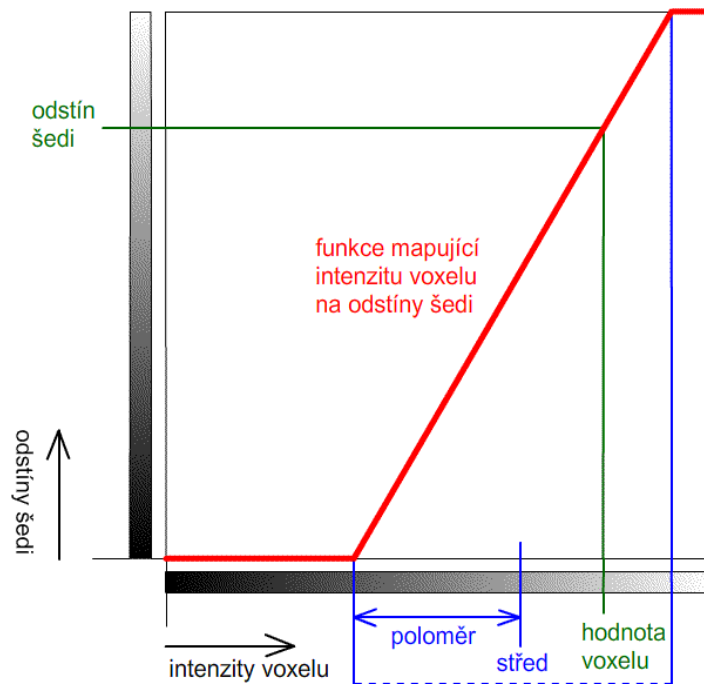
Multiplanární zobrazení má tu výhodu, že je možno označit pomocí myši určité místo na řezu a tím vlastně kdekoli v objemu.



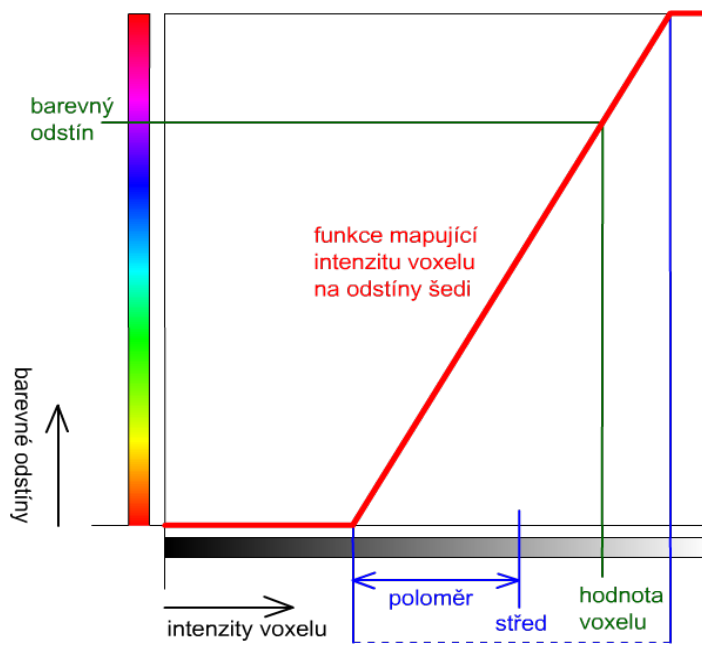
Obrázek 2.10: Možný vzhled lidské lebky zobrazený pomocí tří základních řezů.

Hodnoty voxelů z CT/MR jsou nejčastěji ukládány s přesností 12 bitů. Pokud bychom hodnotu voxelu brali jako odstín šedi a takto jej zobrazovali, pak bychom měli 4096 odstínů ve spektru od bílé po černou. Takové množství není lidské oko schopno rozlišit a tak jej nepodporují ani grafické karty. Běžně se používá 8 bitů, tj 256 hodnot. V tomto případě je tedy potřeba nějak namapovat 4096 hodnot intenzity na 256 odstínů šedé. Lze například vybrat určitý interval intenzit ten lineárně namapovat na celý interval odstínů šedi. Velice intuitivně lze interval definovat pomocí jeho středu a poloměru (viz. obrázek 2.11).

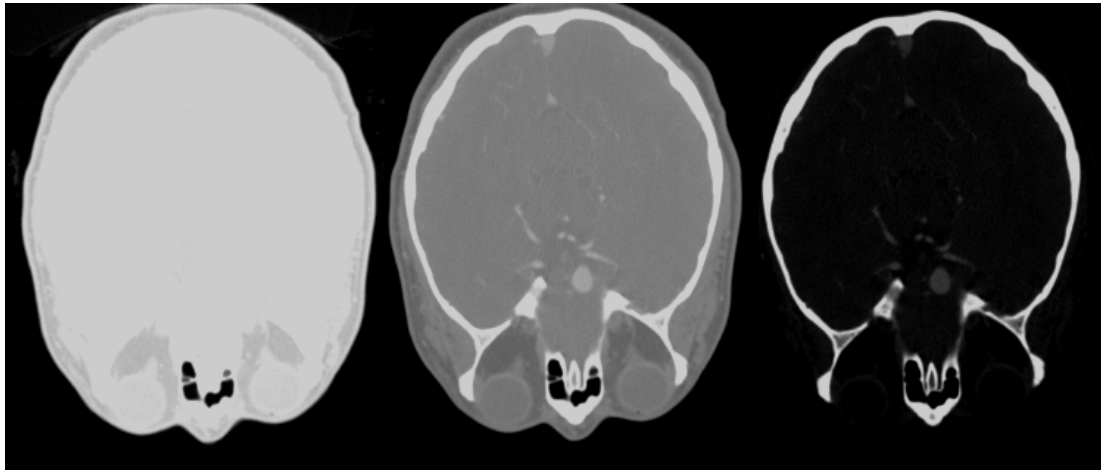
Obecně nám jde o optické rozlišení různých druhů tkání v obraze. Výše popsaný způsob využívá schopnosti lidského oka rozlišit intenzity. Další způsob je postaven na rozlišování barev. Intenzity voxelů jsou mapovány na spektrum barevných odstínů (viz. obrázek 2.12). Každá hodnota intenzity má přiřazenu hodnotu složku H v barevném modelu HSL (hue, saturation, lightness – odstín, sytost, světlost). Zbylé dvě složky (S, L), lze nastavit na libovolnou vhodnou hodnotu.



Obrázek 2.11: Mapování intenzit na odstíny šedi.



Obrázek 2.12: Mapování intenzit na barevné odstíny

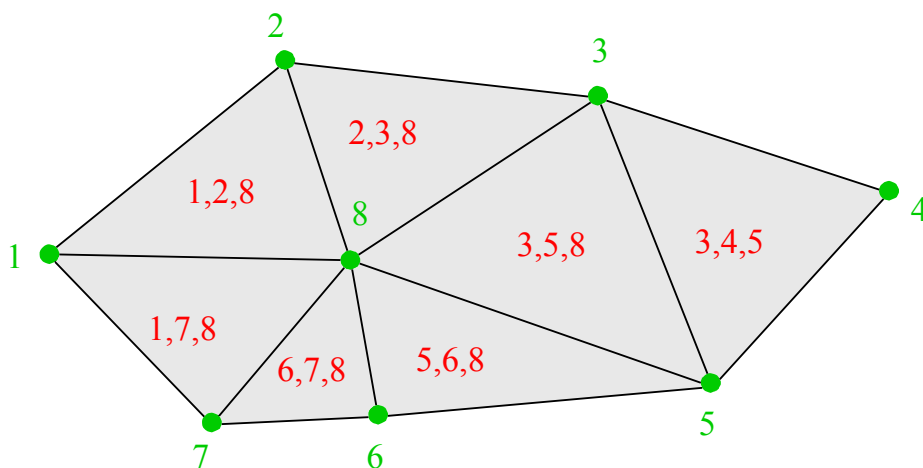


Obrázek 2.13: Lineární mapování intenzit na odstíny šedi v různém nastavení.  
(poloměr intervalu 500, střed 500, 1000 a 1500)

### 2.2.3 Zobrazení pomocí hraniční reprezentace

Vykreslovat objemová data z CT/MR pomocí hraniční reprezentace není možné přímo, ale až po segmentaci a vytvoření příslušného modelu v hraniční reprezentaci (tzv. B-rep od anglického boundary representation). Takto definované těleso (např. určitý druh tkáně) je definován pomocí hranice, která ho dělí od jeho okolí.

Nejpoužívanější hraniční reprezentací je síť trojúhelníků. Síť nazýváme množinu trojúhelníků, které sdílejí své hrany. Síť trojúhelníků je možné popsat pomocí pole bodů (část geometrická) a pole trojúhelníků (část topologická). Struktura popisující trojúhelník se odkazuje do pole bodů.



Obrázek 2.14: Příklad trojúhelníkové sítě.



Výhodou této reprezentace je podpora pro vykreslování trojúhelníků na současných grafických akcelerátorech včetně transformací, osvětlení a podobně. Nevýhodou je, že nejdříve je potřeba data segmentovat a převést do této reprezentace. Dále tento způsob není vhodný pro zobrazování dat s plynulým přechodem do svého okolí, protože podporuje pouze ostré hranice.

Existuje i komplexnější reprezentace – tzv. okřídlená hrana. Je vhodná pro složitější operace nad trojúhelníkovými sítěmi, jako jsou například logické operace. S použitím této reprezentace se snáze udržuje uzavřenost tělesa a podobně. Pro ruční modelování povrchu těles se zase používají některé z interpolačních křivek a ploch (NURBS, T-Spline)

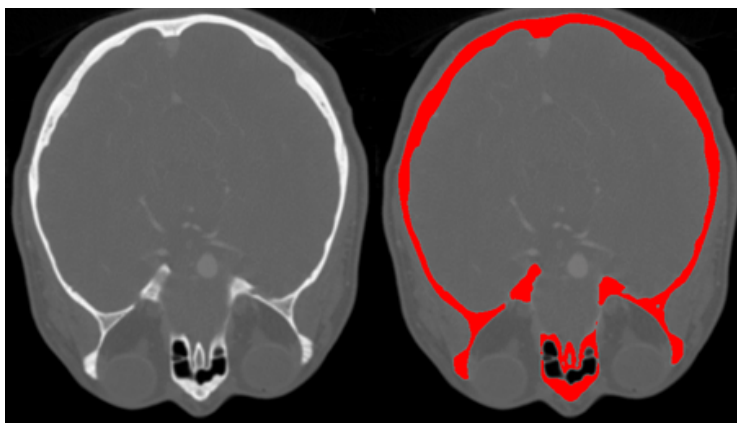
## 2.3 Segmentace tkání

Existuje několik způsobů jak segmentovat objemová CT/MR data. Různé metody se liší měrou automatickosti. Některé vyžadují pouze minimální zásah obsluhy, jiné jsou na asistenci odborníka přímo založené.

Dobře prozkoumán je způsob sestávající z následujících kroků:

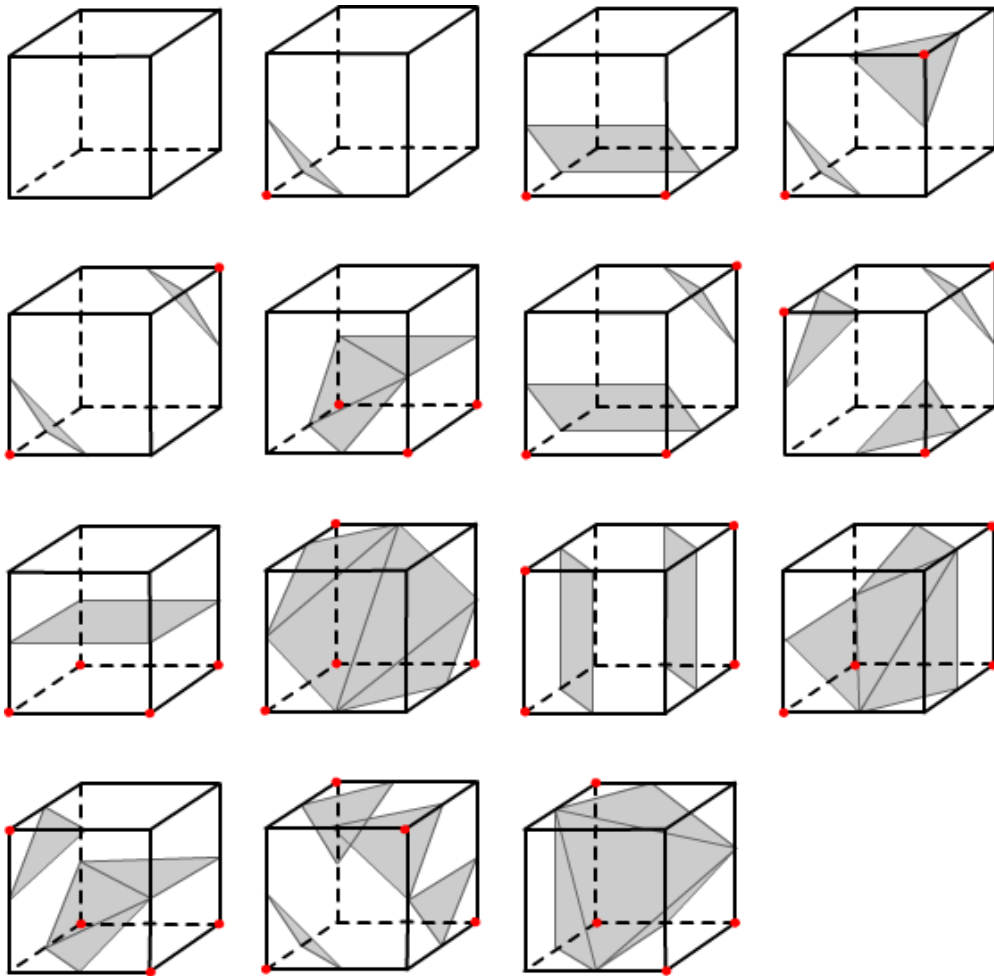
- Výběr voxelů, jež považujeme za součást hledané tkáně.
- Převod vybrané tkáně na hraniční reprezentaci.
- Vyhlazení povrchu.
- Snížení počtu trojúhelníků pro snadnější zobrazení a manipulaci.

Výběr voxelů provádí například prahováním. Může být použit jeden nebo více prahů. Nejprve zjistíme, jaké hodnoty intenzity na CT/MR snímku má tkáň našeho zájmu. Pokud se jedná o souvislý interval, pak určíme prahy a voxely, jež leží v tomto intervalu označíme jako součást tkáně. Pokud daná tkáň vykazuje na snímcích široké rozmezí hodnot a v tomto rozmezí se nachází i jiné tkáně, pak tato metoda není vhodná. I v těch nejpříhodnějších případech se jedná o určitou kombinaci zmíněných extrémů. Proto je potřeba odhadnout vhodné prahy tak, abychom označili co nejvíce voxelů, jež náleží označované tkáni a co nejméně těch, které ji nenáleží.



Obrázek 2.15: Výběr voxelů pomocí prahování.

Pro některá použití je vhodné znát hraniční reprezentaci dané tkáně. Pro převedení vybraných voxelů do hraniční reprezentace se používá algoritmus Marching cubes. Ten používá pomyslnou kostku, jenž postupně prochází celé objemové těleso tak, že v každém rohu kostky je voxel. Uvnitř kostky jsou konstruovány trojúhelníky podle toho, zda voxely v rozích náleží segmentované tkáni. Máme tedy osm rohů, z nichž každý může náležet nebo nenáležet segmentované tkáni. Celkově může nastat 256 kombinací, jež se dají redukovat na 15 základních případů. Rotací a zrcadlením základních případů lze získat všech 256 možností.

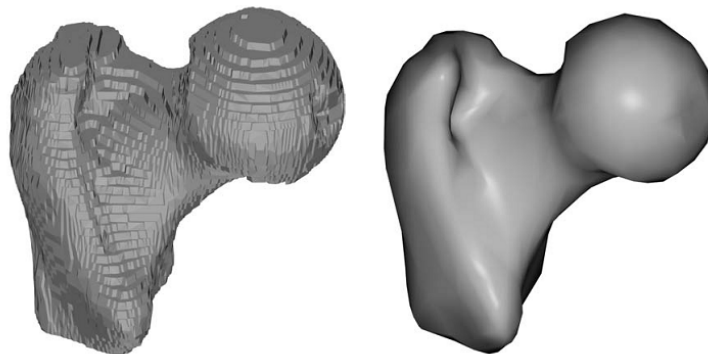


Obrázek 2.16: Základní případy algoritmu Marching cubes.

Jak je vidět na obrázku 2.17, samotný algoritmus Marching cubes nestačí. Rozhodování o tvaru trojúhelníku, které vytváří bere v úvahu pouze osm sousedních vrcholů a tak nedokáže zachytit charakter povrchu v daném místě z globálnějšího pohledu. Výsledný model má ostré hrany a jsou viditelné vrstvy původní voxelové mřížky. Povrchy tkáně lidského těla mají většinou hladký průběh a tak vyhlazení povrchu pravděpodobně přiblíží model skutečnosti. Existují různé metody vyhlazování. Liší se například podle toho, zda zachovávají hrany a vyhlazují pouze větší plochy nebo ne. Pro

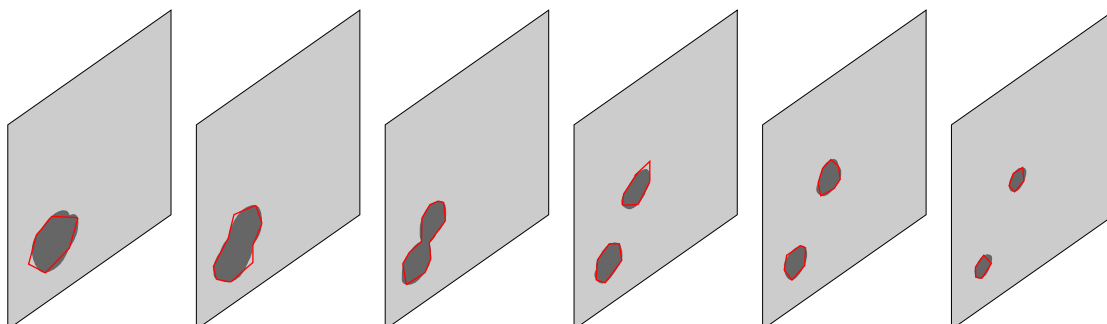
organické modely je vhodnější použít některou z jednodušších metod, která hrany nezachovává. Nebezpečím vyhlazování je možná degradace lokálních detailů. Proto je potřeba vhodně nakonfigurovat vyhlazovací algoritmus tak, abychom dostali dostatečně hladký povrch a zároveň zachovali dostatek detailů.

Jednoduchou metodou je Laplaceovo vyhlazování, které provádí přesun uzlů trojúhelníkové sítě. Nová poloha uzlu se určí jako průměr poloh sousedních uzlů (po hraně). Pokud chceme určovat míru vyhlazení, pak nepřesunujeme pouze částečně směrem k místu průměru. Takto je možno provést několik iterací, až je dosaženo požadovaného výsledku. Tato metoda je vhodná zvláště pro tento případ, kdy je model definován velkým množstvím trojúhelníků. V opačném případě by docházelo k přílišné degradaci modelu (zanikání detailů, zmenšování). Pak by bylo potřeba použít některou z pokročilejších variant tohoto algoritmu, jež se snaží kompenzovat vznikající chyby a můžou používat širší okolí (po dvou hranách a více). Jsou to například algoritmy Laplace+HC nebo dolní propust podle Taubina.

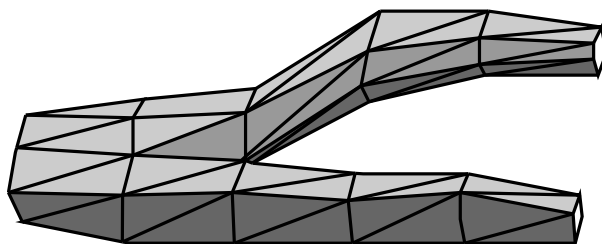


Obrázek 2.17: Stehenní kost po marching cubes a po vyhlazení [3].

Další metoda je založena na ručním vyznačování hranic tkání v objemových datech. To se nejčastěji provádí pomocí zakreslování hranic tkáně v řezech objemem. Řezů mohou být desítky či stovky a tak. Vyznačování v každém řezu zvlášť by bylo velice pracné a tak se používá označování každého n-tého řezu. Označení tkáně v mezilehlých řezech je nutno odhadnout. Z výsledné sady hranic tkáně v řezech se automatickou metodou zkonstruuje hraniční reprezentace modelované tkáně.



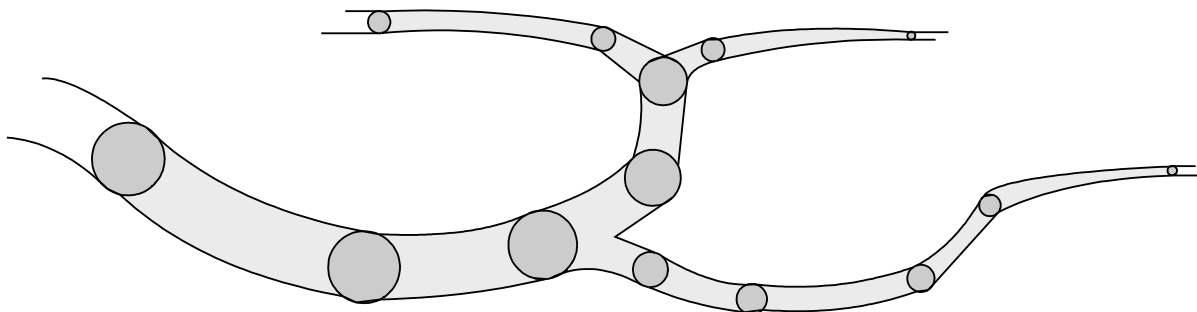
Obrázek 2.18: Zakreslování hranic tkáně v řezech.



Obrázek 2.19: Trojúhelníková síť vytvořená na základě vyznačených hranic.

Ve skutečnosti ovšem tkáň nemusí procházet řezy takto ideálně. Může se stáčet a procházet rovnoběžně s rovinou řezu, což komplikuje její vyznačování. Tvorba hraniční reprezentace na základě hranic v rovnoběžných řezech v tomto případě není triviální.

Máme-li tkáň vyznačené v jednotlivých řezech a označení tvoří uzavřenou křivku, pak můžeme zjistit, které voxely náleží označené tkáni a které ne. To znamená, že můžeme použít výše zmíněnou metodu s použitím Marching cubes, vyhlazení a redukce počtu trojúhelníků. Pro modelování žil, tepen, nervů a podobných tkání, které mají přibližně kruhový průřez a zároveň při průchodu tělem nemění tvar příliš dramaticky je možné navrhnout speciální způsob modelování. Možná by stačilo zadat polohu omezeného množství bodů, kterými tkáň prochází a hodnotu průměru tkáně v těchto bodech. Zbylé úseky mezi těmito body by se aproximovaly. Průměr by bylo také možné aproximovat a následně vygenerovat hraniční reprezentaci nalezené tkáně.



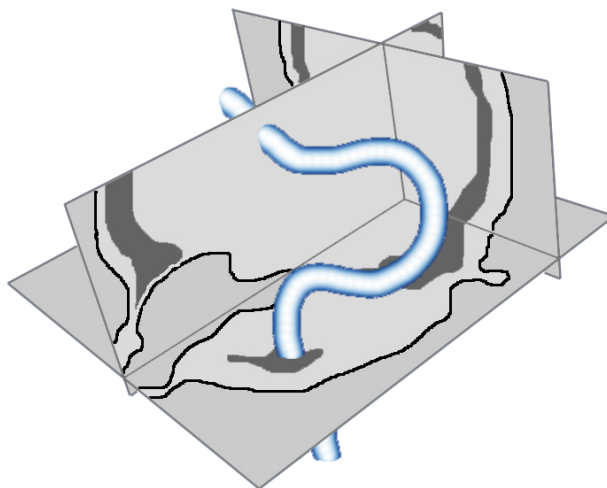
Obrázek 2.20: Ilustrace modelování pomocí bodů průchodu s daným průměrem.

Tento dokument se bude dále zabývat návrhem systému pro modelování tkání tímto způsobem.

## 2.4 Základní koncepce programu

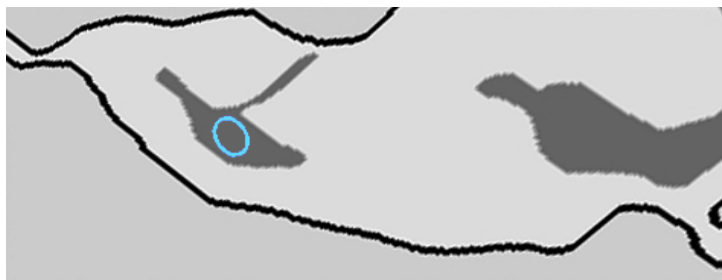
Navrhovaný program by měl uživateli umožnit načtení objemových dat z CT nebo MR. Tato data budou dále zobrazována multiplanárním způsobem, protože ten umožňuje přehledné procházení objemovým tělesem a snadné zadávání bodů v prostoru. Intenzity pixelů budou mapovány na odstíny šedi tak, jak je to popsáno v kapitole 2.2.2. Uživatel by měl mít možnost nastavit střed mapovaného intervalu a jeho poloměr tak, aby si mohl určit tu část spektra ve které tkáň jeho zájmu co nejlépe vyniknou.

Hlavním účelem programu je modelování lineárních tkání definovaných pomocí řídicích bodů. Tyto body definují místa průchodu tkáně a průměr tkáně v těchto místech. Průběh a průměr tkáně mezi těmito body bude aproximován. Program by měl tedy umožňovat přidávání a mazání těchto bodů a určování jejich průměru.



Obrázek 2.21: Zobrazení všech řezů i tkáně v perspektivě.

Pro přesnější zkoumání a definici průchodu tkáně bude použit druhý způsob zobrazení. V něm bude zobrazena pouze jedna rovina řezu v ortogonálním zobrazení a modelovaná tkáň bude vyznačena tam, kde tuto rovinu protíná (viz. obr. 2.22).



Obrázek 2.22: Ortogonální zobrazení řezu a vyznačeným místem průchodu tkáně.

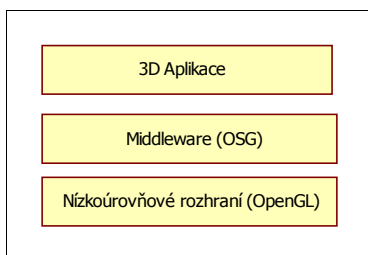
## 2.5 Zobrazování trojrozměrných dat

Zadaná úloha potřebuje pro svou realizaci zobrazovat trojrozměrná data v reálném čase. V dnešní době toto dokážou i běžné domácí počítače, jež téměř vždy disponují grafickou kartou a 3D akcelerátorem. Důvodem jsou hlavně požadavky herního průmyslu. Masová výroba grafických akcelerátorů umožňuje pořídit velice výkonná zařízení za nízkou cenu.

Chceme-li přistupovat k funkcím grafického akcelerátoru, pak můžeme využít dvou hlavních rozhraní OpenGL a Direct3D. OpenGL poskytuje pouze nízkoúrovňové rozhraní pro vykreslování základních primitiv (bodů, čar, trojúhelníků a bitmap). Direct3D v „retained“ módu dříve nabízelo funkcionalitu na vyšší úrovni jako je například správa grafu scény. Ovšem v novějších verzích už není tento mód podporován. Na rozdíl od OpenGL, které má své implementace na mnoha platformách, je Direct3D vázáno na platformu Microsoft Windows. Tato vlastnost činí Direct3D méně použitelným pro výzkumné účely. Pro implementaci navrhovaného systému bude použito OpenGL.

Zajímavou možností je také použití nástrojů, jež využívají zmíněná rozhraní a dodávají další pokročilé funkce jako například objektově orientovaný přístup a další. Nízkoúrovňové rozhraní sice poskytuje programátorovi větší kontrolu nad prováděnými operacemi, ale vývoj je pomalý. Také existuje mnoho často využívaných požadavků, jejichž řešení je známo a tak není potřeba aby je každý programátor implementoval sám. Pro takovéto nástroje se vžil název „middleware“, protože stojí mezi nízkoúrovňovým rozhraním a vytvářenou aplikací. Ve výzkumných aplikacích je často využívána knihovna Open Scene Graph (dále jen OSG). OSG pracuje s rozhraním OpenGL. Mezi jeho vlastnosti patří:

- Podpora načítání a ukládání mnoha 2D a 3D souborových formátů.
- Objektové rozhraní ke všem funkcím OpenGL (osvětlení, texturování, shadery atd.)
- Správa scény pomocí grafu scény.
- Další pokročilé funkce (trojúhelníkové sítě, animace, manipulování s objekty atd.)
- Platformně nezávislá tvorba okna a práce s událostmi.
- Dobře navržený model tříd, otevřený zdrojový kód v C++.
- Možnost rozšiřovat funkcionalitu knihovny prostřednictvím modulů.



Obrázek 2.23: Jednotlivé vrstvy 3D aplikace

Nejdůležitější složkou OSG je graf scény. Většina ostatních tříd OSG s ním nějak souvisí a samy o sobě nemají smysl. Samotný strom je tvořen objekty třídy *Node* a jejími potomky. Chceme-li, aby měl uzel více potomků, pak musí být typu *Group*. Třídy odvozené od *Group* přidávají další funkcionalitu, jako například 3D transformace nad potomky, přepínání aktivních potomků, ořezávání podle rovin, a další. Chceme-li do scény vložit objekty pro vykreslení, pak v ní musí existovat uzel typu *Geode*. Ten obsahuje pole tříd pro přímé vykreslení, které musí být typu *Drawable*. To jsou například *Geometry* pro vykreslování za pomoci primitiv, *DrawPixels* pro vykreslování bitmap, *ShapeDrawable* pro základní tvary (koule, kvádr, ...) a další.

Každý uzel může obsahovat vlastní *StateSet*, který umožňuje měnit stav OpenGL za použití jednoduchého objektového rozhraní. Různé oblasti se nastavují prostřednictvím tříd zděděných z třídy *StateAttribute*. Jsou to například *Fog*, *Light*, *PolygonMode*, *Texture*, *Viewport* a podobně. Při vykreslování je průchod stromu optimalizován tak, aby k přepínání stavů docházelo co nejméně. To šetří čas vykreslování.

Má-li určitý uzel nebo *Drawable* reflektovat stav dat, která zobrazujeme, pak jej si na něj můžeme udržovat odkaz a měnit ho přímo, nebo použijeme elegantní způsob s použitím callbacků. Uzlu jednoduše přiřadíme objekt třídy odvozené od *NodeCallback*. Při aktualizaci scény se automaticky zavolá *operator()(Node\* node, NodeVisitor\* nv)*. V těle této metody můžeme pomocí odkazu *node* změnit libovolný parametr uzlu.

Graf scény je možné procházet pomocí objektů tříd odvozených od *NodeVisitor*. Například chceme-li zjistit průnik nějakého objektu se scénou, pak použijeme *IntersectionVisitor*. K získání statistik o scéně slouží *StatsVisitor*. Dále existuje několik tříd tohoto typu pro optimalizaci scény (LOD, tessellace, převod polí trojúhelníků na pásy atd.). Před každým vykreslením se použije *UpdateVisitor*, který kromě jiného volá callbacky jednotlivých uzlů.

Součástí OSG je také třída *Viewer*. Vytváří okno v okenním systému dané platformy, přijímá události a umožňuje jejich preposílání objektům třídy *EventHandler*. Udržuje také odkazy na kamery a kořen grafu scény.

OSG obsahuje také třídy pro podporu geometrie. Jsou to vektory a matice různých typů, pole vektorů, indexů a podobně. Implementovány jsou i operace mezi těmito třídami (násobení vektoru maticí apod.).

Kompletní systém pro programování 3D aplikací dělá z OSG podpora mnoha 2D a 3D formátů. Pomocí modulu *osgDB* je možné načítat a ukládat obrázky, 3D modely nebo celé části scény. V současné verzi jsou podporovány například formáty 3DS, BMP, DDS, DXF, GIF, IVE, JPEG, MD2, OBJ, PNG, STL, TGA, TIFF a další. OSG disponuje také vlastním formátem pro ukládání a načítání celých (pod)grafů scény.

## 2.6 Vykreslování křivek

Jak bylo zmíněno dříve, modely tkání budou definovány omezeným počtem bodů s danou polohou a poloměrem. Úseky mezi těmito body budou aproximovány co do průběhu i poloměru. Průběhy modelovaných tkání mají hladký spojitý charakter. Pro aproximaci takového průběhu je vhodné použít některý ze známých druhů křivek.

Křivky bývají reprezentovány jako parametry určité rovnice, jež je následně zobrazována. Tato rovnice může být ve tvaru explicitním, implicitním nebo parametrickém. Explicitní tvar ( $y=f(x)$ ) je vhodný pouze u křivek, jež jsou zároveň funkcemi. Implicitní tvar ( $F(x, y)=0$ ) je zase obecně náročný na sestavení. V oblasti počítačové grafiky je nejvíce používám tvar parametrický ( $x=x(t), y=y(t)$ ). Je možné chápat jej jako postupný výpočet dráhy pohybujícího se bodu. Souřadnice bodu jsou funkcemi parametru času. Parametr  $t$  se obvykle pohybuje v rozsahu  $\langle 0, 1 \rangle$ . Průběh křivky je možné vyčíslvat postupně se zvoleným krokem a průběh mezi jednotlivými kroky nahradit přímkou. Volbou kroku je možné vyvážit výpočetní náročnost a vizuální kvalitu. Složitější tvary bývají složeny z více křivek s plynulým napojením.

Základním druhem křivek jsou v počítačové grafice polynomiální křivky. Mají tvar:

$$Q_n(t) = a_0 + a_1t + \dots + a_nt^n$$

Nejčastěji se používají tzv. kubiky, kde má  $n$  hodnotu 3. Často používané interpolační křivky jsou tzv. Hermitovské kubiky (jinak též Fergusonovy kubiky). Tato křivka je definována počátečním a koncovým bodem ( $P_0, P_1$ ) a tečnými vektory v těchto bodech ( $pv_0, pv_1$ ). Přimykání křivky k tečnému vektoru je dáno velikostí vektoru.



Obrázek 2.24: Definice Hermitovské kubiky

Parametrická rovnice má následující tvar:

$$Q(t) = P_0F_1(t) + P_1F_2(t) + pv_0F_3(t) + pv_1F_4(t)$$

Kde  $F_1, F_2, F_3, F_4$  jsou kubické Hermitovské polynomy tvaru:

$$F_1(t) = 2t^3 - 3t^2 + 1$$

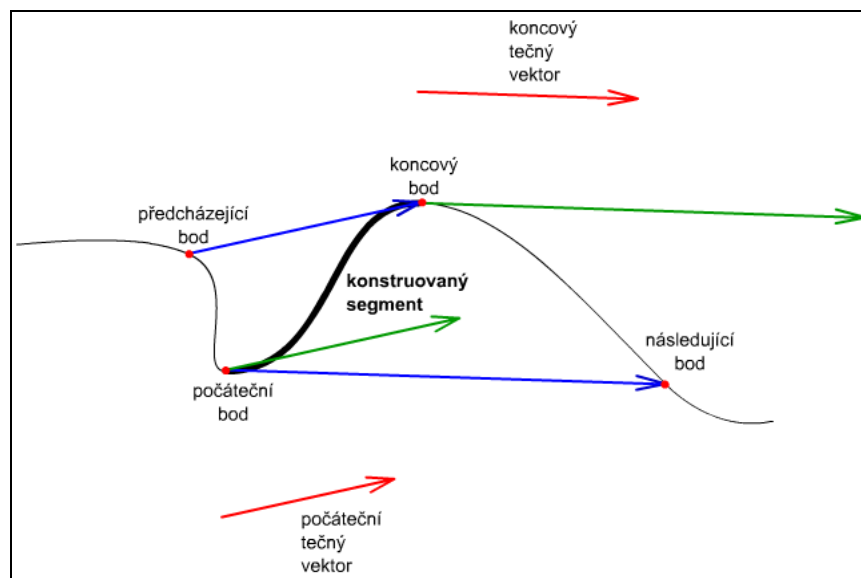
$$F_2(t) = -2t^3 + 3t^2$$

$$F_3(t) = t^3 - 2t^2 + t$$

$$F_4(t) = t^3 - t^2$$



Pro aproximaci průběhu tkání mezi zadanými budou použity tyto křivky. Celkový průběh tkáně bude po po částech složen z jednotlivých Hemirtovských křivek (segmentů). Aby na sebe jednotlivé křivky plynule navazovaly, je potřeba zajistit, aby tečné vektory jednotlivých křivek měly v bodech napojení stejný směr. Pokud mluvíme o tzv. geometrické spojitosti. Pokud je shodná i délka vektorů, pak se jedná o tzv. parametrickou spojitost. Pro výpočet tečných vektorů bude použit následující algoritmus (termíny viz. obrázek 2.25). Směr počátečního tečného vektoru segmentu má stejný měr směr jako vektor od přecházejícího bodu ke koncovému segmentu. Nápodobně koncový tečný vektor má směr stejný jako vektor od počátečního bodu segmentu k následujícímu bodu. Tyto směry jsou na obrázku znázorněny zeleně. Ale při konstrukci křivky hraje významnou roli také délka tečných vektorů. Ta určuje, nakolik se bude křivka k vektoru přichylovat. Proto je nutné délku vektorů upravit. V navrhovaném programu se vektory budou normalizovat, poté se vynásobí délkou konstruovaného segmentu a dále se může násobit vhodnou konstantou pro doladění přichylnosti křivek.



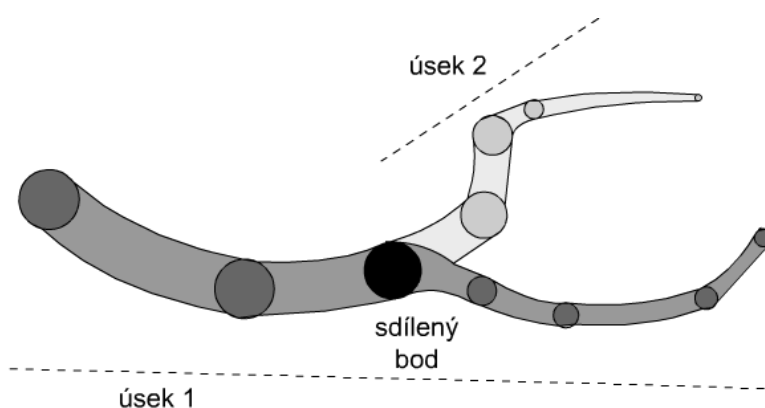
Obrázek 2.25: Konstrukce počátečních a koncových tečných vektorů segmentu křivky.

## 3 Návrh

### 3.1 Uživatelské rozhraní

Uživatelské rozhraní bude obsahovat především oblast, ve které se budou zobrazovat CT/MR objemová data a modelovaná tkáň. Hlavně pomocí myši bude probíhat natáčení 3D scény a editace modelované tkáně.

Modelování tkání bude probíhat postupným přidáváním bodů, kterými tkáň prochází. Posloupnost bodů bude tvořit tzv. úsek. Jednotlivé úseky modelované tkáně mohou sdílet společné body a tím vytvářet rozvětvení tkání.



Obrázek 3.1: Dvojměrný příklad modelování tkáně pomocí úseků se sdílenými body.

Pro efektivní ovládání programu není vhodné, aby bylo zapotřebí často vybírat položky z menu a podobně. Modelování tkání, určování úhlu pohledu či posun rovin řezu se nejnadhěji provádí za pomoci myši. Mód použití myši se dá efektivně určovat stiskem kláves ALT, CTRL a SHIFT a tlačítek myši.

Program bude poskytovat dva základní druhy zobrazení. Prvním bude perspektivní zobrazení všech tří pravoúhlých rovin. Řídící body se v tomto zobrazení budou jevit jako kuličky, modelovaná tkáň jako „potrubí“ mezi těmito body. Ovládání programu v tomto módu zobrazení:

- Rotace pohledu na scénu – Stisk pravého tlačítka myši a tah myši. Pokud bude tento postup použit v pravoúhlém módu, tak se pohled automaticky přepne do perspektivního s rotací nastavenou na rovinu, kterou jsme pozorovali.
- Zadávání bodů průchodu tkáně- Stisk klávesy CTRL a levého tlačítka myši nad vybraným místem na řezu. Během následujícího tažení myši se určí průměr tkáně v daném místě. Nový bod automaticky propojí s označeným bodem.
- Změna průměru bodu – Klik na již existující bod pro označení a tah myši nahoru/dolů.

V druhém módu je zobrazena pouze jedna rovina řezu pomocí pravouhlého promítání. Řídící body jsou zde viditelné pouze pokud je daná rovina řezu umístěna v jejich blízkosti a jeví se jako křížky. Z modelované tkáně je viditelný pouze její průnik s danou rovinou řezu. To umožňuje přesněji sledovat, zdali se vytvořený model shoduje s reálnou tkání na snímku. Ovládání programu v tomto módu zobrazení:

- Přesouvání bodů – nejprve je potřeba umístit rovinu řezu poblíž daného bodu, aby se tento bod zobrazil. Následuje stisk levého tlačítka myši nad bodem pro jeho označení a tah myši.
- Zadávání bodů průchodu tkáně – stisk levého tlačítka a tah myši pro určení polohy bodu. Tento mód je určený především pro upřesnění modelu a tak je možné přidávat body pouze mezi jiné již existující. Místo kliku se tedy musí nacházet v blízkosti středu modelované tkáně.

Operace nezávislé na módu zobrazení:

- Přiblížení / oddálení pohledu na scénu – Stisk prostředního tlačítka myši a tah nahoru/dolů.
- Posouvání řezů objemovými daty – Stisk klávesy levého tlačítka myši nad vybraným řezem a pro jeho označení a následně tažení myši nahoru/dolů nebo použití šipek nahoru a dolů na klávesnici.
- Mazání bodů – Označení bodů stiskem levého tlačítka myši a klávesa DELETE.

Pro přechod mezi jednotlivými pohledy se použijí klávesy:

- F1 pro pohled shora.
- F2 pro pohled zepředu.
- F3 pro pohled z boku.

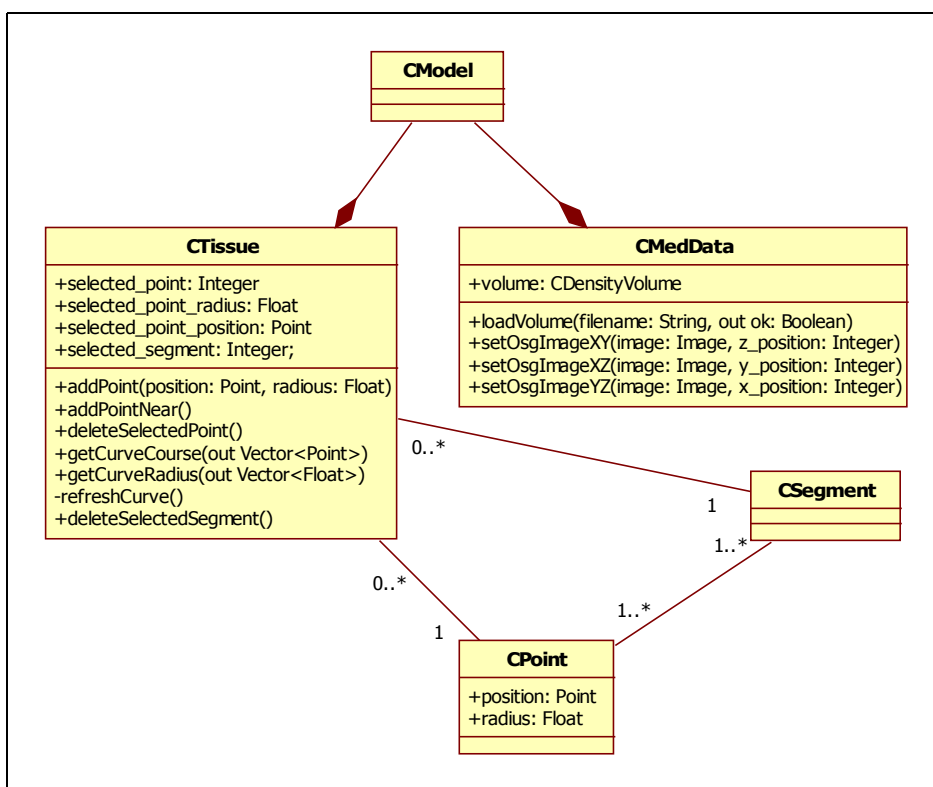
## 3.2 Architektura programu

Program je navrhován pomocí objektově orientovaného přístupu. Z hlediska architektury nebude nikterak složitý. V systému se vyskytují tyto tři hlavní oblasti:

- Uživatelské rozhraní s 3D zobrazením
- Medicínská data z CT nebo MR
- Modelované tkáně

Jistě bude výhodné použít návrhový vzor Model-View-Controller (model, pohled, ovládání) pro oddělení daných domén.

- Objekt třídy CModel bude zapouzdřovat data, se kterými se pracuje a funkce nad těmito daty. Daty je zde míněn 3D rastr z CT/MR a modelovaná tkáň.
- Objekt třídy CView bude zajišťovat zobrazení těchto dat.
- Objekt třídy CController zajistí editaci dat na základě vstupů z klávesnice a myši.

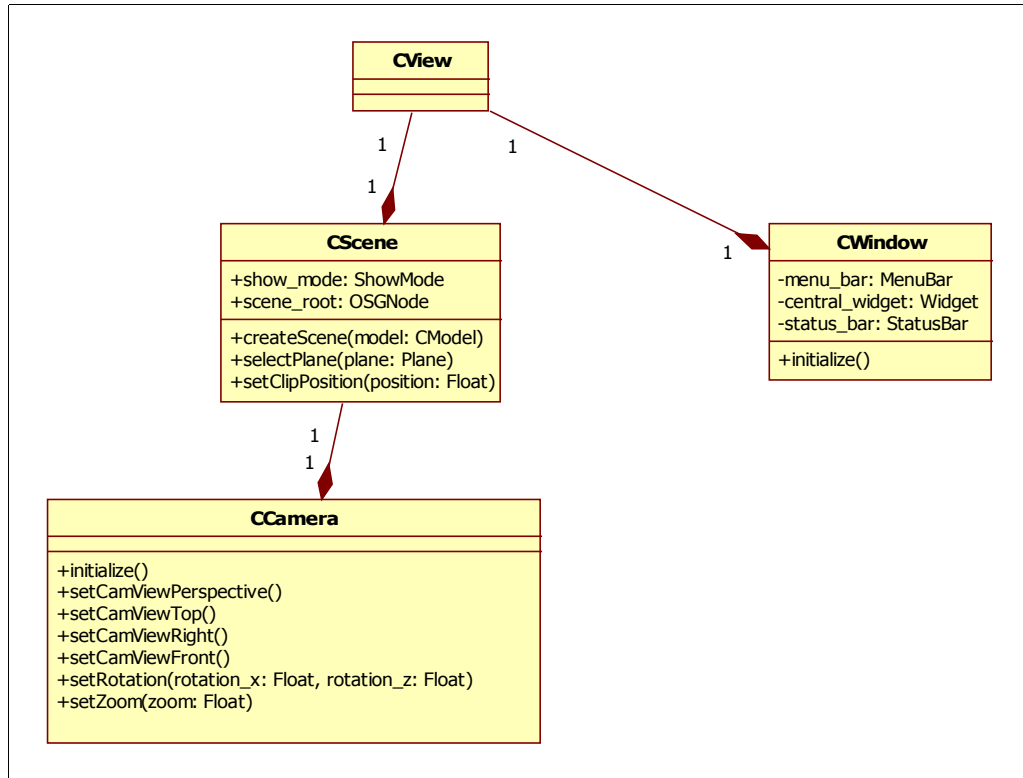


Obrázek 3.2: Diagram tříd zobrazující třídu CModel a její složky.

Jak je vidět z obrázku 3.2, Třída CModel nemá žádnou vlastní funkčnost. Důležité jsou její složky CTissue, která spravuje modelovanou tkáň a CMedData, jež se stará o načítání objemových dat z CT/MR a práci s nimi.

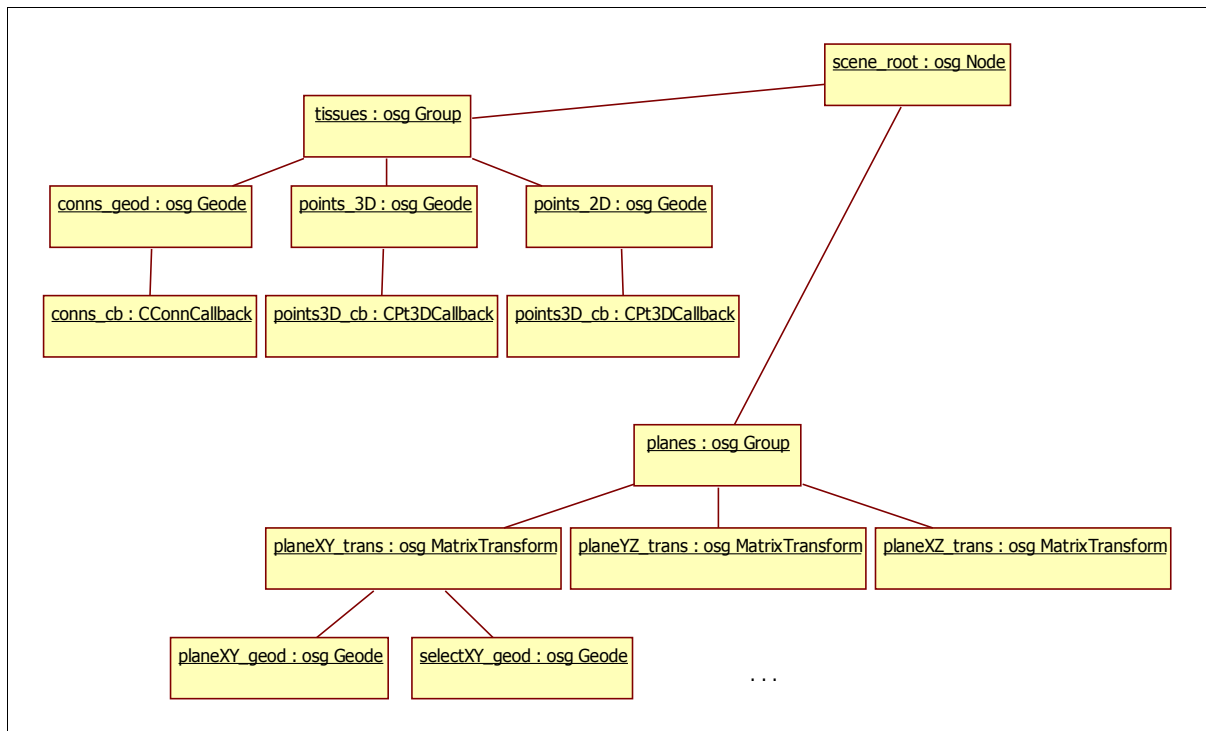
Třída CTissue udržuje body tkáně a její segmenty. Umožňuje jejich přidávání a ubírání. Vrací také průběh tkáně v prostoru a také průběh jejího průměru.

Třída CMedData udržuje objemová data a dokáže naplnit zadaný obrázek daty z řezu kolmého na některou z os X, Y, Z.



Obrázek 3.3: Diagram tříd zobrazující třídu CView a její složky.

Podobně jako u třídy CModel, tak i u třídy CView jsou nejdůležitější složky. CWindow slouží hlavně pro konstrukci hlavního okna programu. CScene spravuje trojrozměrnou scénu a umožňuje různé pohledy na ni v závislosti na módu zobrazení (perspektivní, pravoúhlé – atribut show\_mode). To zahrnuje hlavně změny stavů jednotlivých uzlů a listů grafu scény. Také označení roviny či bodu se musí projevit ve scéně, aby se o něm uživatel dozvěděl. Hlavní složkou je však atribut scene\_root, jenž odkazuje na kořen grafu scény. Graf scény je zobrazen na obrázku 3.4.



Obrázek 3.4: Graf scény v době běhu programu.

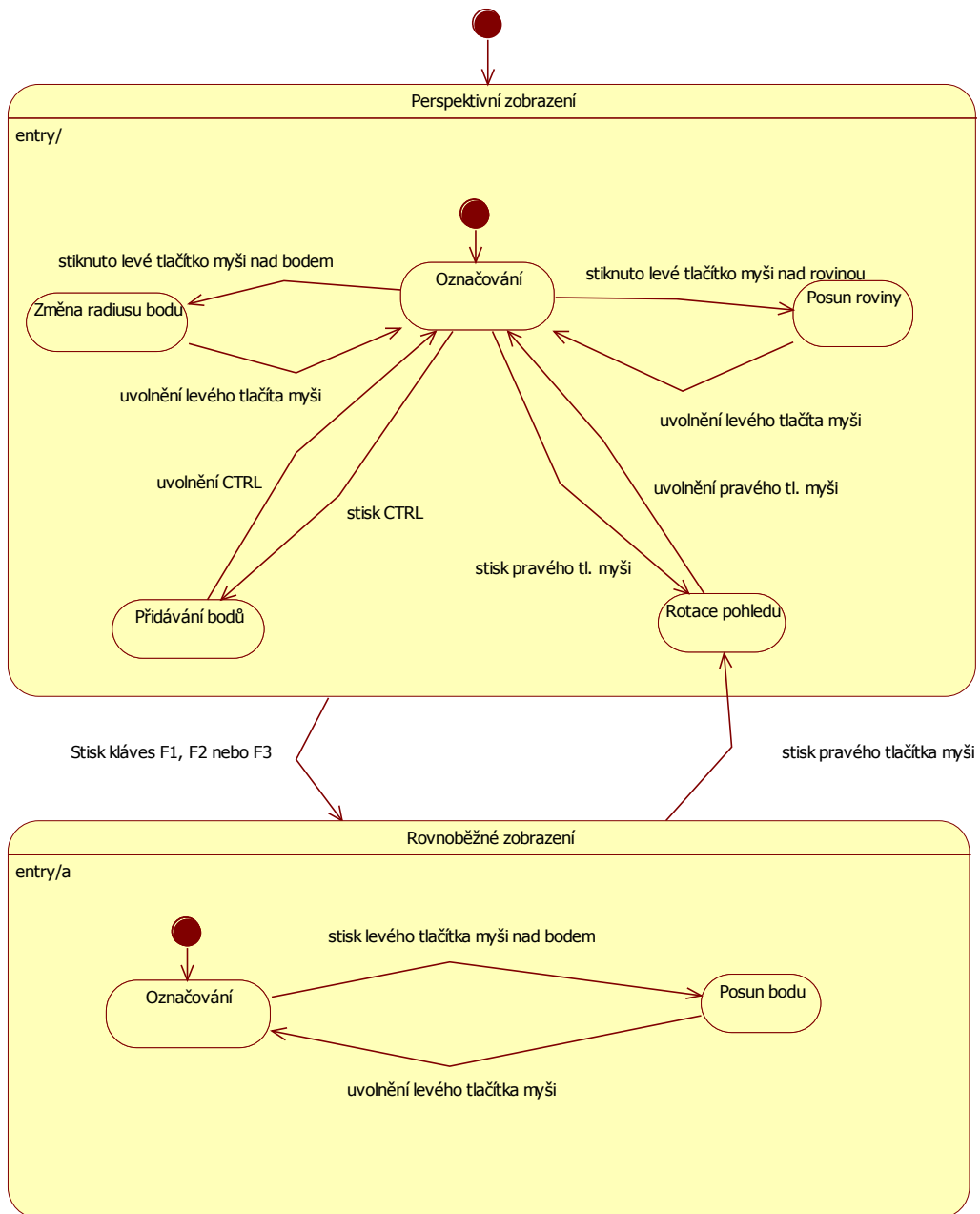
Graf scény obsahuje:

- scene\_root – Skupina obsahující modelované tkáně a řezy objemem.
- tissues – Skupina pro všechny složky tkání.
- points\_3D – Geode pro „kuličky“ jenž znázorňují místa, kudy tkáň prochází v perspektivním zobrazení.
- points\_2D – Geode pro „křížky“ jenž znázorňují místa, kudy tkáň prochází v pravoúhlém zobrazení.
- conns\_geod - Geode pro propojení mezi těmito body, tedy samotnou tkáň.
- planes – Skupina obsahující řezy objemem.
- plane<rovina>\_trans – Transformační uzel umožňující posouvání řezů.
- plane<rovina>\_geod – Geode pro řez samotný.
- select<rovina>\_geod – Geode pro linii kolo řezu znázorňující jeho označení.

Všechny Geode stromu obsahují ještě další prvky typu Drawable, kde jsou uloženy zobrazované elementy. Graf scény musí reflektovat stav zobrazených dat. To zajišťuje buď objekt CScene nebo se tak děje pomocí callbacků. Callbacky mají:

- Řezy objemem – Je potřeba aktualizovat texturu na řezu podle jeho polohy.
- Body průchodu tkáně – Změna velikosti a barvy podle toho, zda je bod označen.
- Tkáň samotná – Reflektuje změny bodů.

Objekt třídy CController především přijímá události z klávesnice a myši. Zjišťuje také, na jaký objekt ve scéně se kliklo. Podle těchto informací dává povely dalším objektům a mění svůj vlastní stav.

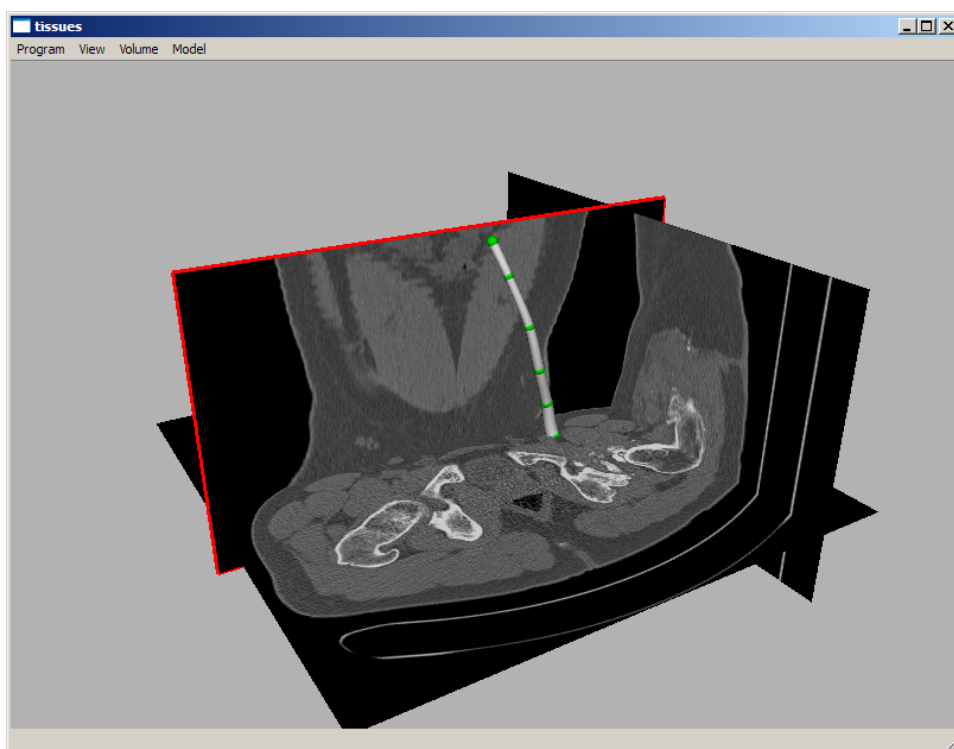


Obrázek 3.5: Stavový diagram uživatelského rozhraní

## 4 Implementace

Program je implementován pomocí jazyka C++ s využitím knihoven Open Scene Graph a MDSTK. Pro tvorbu dalších okenních elementů jako například menu byla použita knihovna QT. Při vývoji programu je využívána správa verzí pomocí systému Subversion. Komentování kódu umožňuje vygenerování programové dokumentace systémem Doxygen.

Implementovány jsou všechny navržené funkce kromě napojování jednotlivých úseků, což znemožňuje vytváření rozvětvených struktur. Označení řezu je znázorněno červenou linkou okolo řezu. V perspektivním zobrazení jsou body zobrazeny jako kouličky. Neoznačené body jsou zelené, označené jsou obarveny na červeno. Propojení mezi body se jeví jako šedá „trubka“. V Pravoúhlém zobrazení se body zobrazují pouze pokud se nachází v blízkosti roviny na kterou pohled směřuje. Jeví se jako křížky, které je možno uchopit a přesouvat. Z modelované tkáně se zobrazí pouze její průnik s danou rovinou (viz. obrázek 3.4).

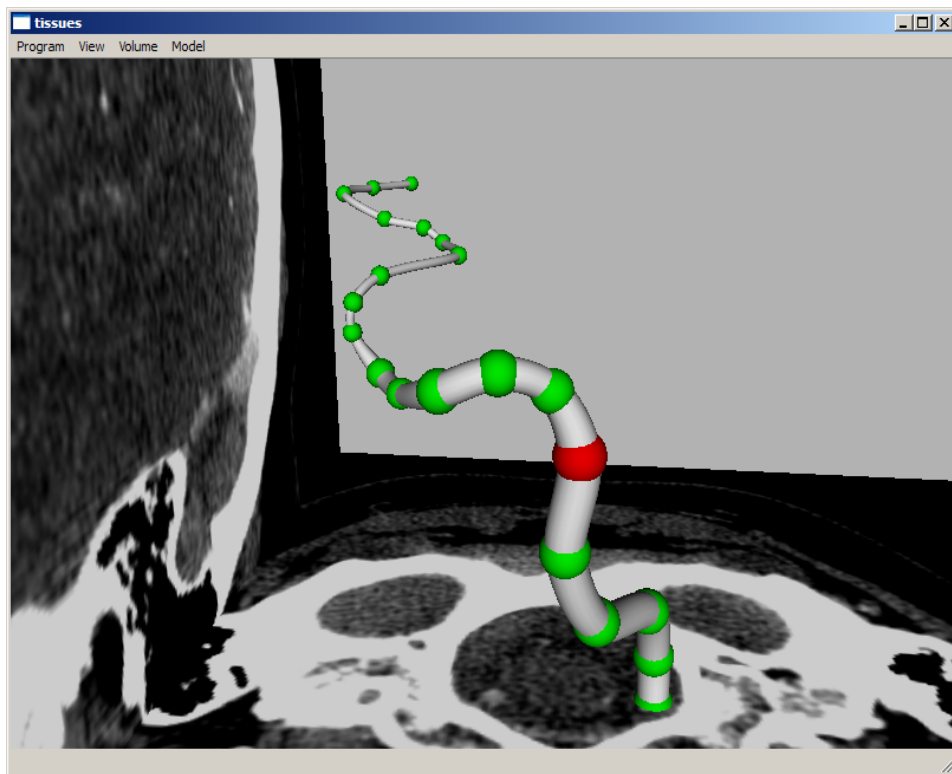


Obrázek 4.1: Zobrazení řezů objemem s označenou rovinou.

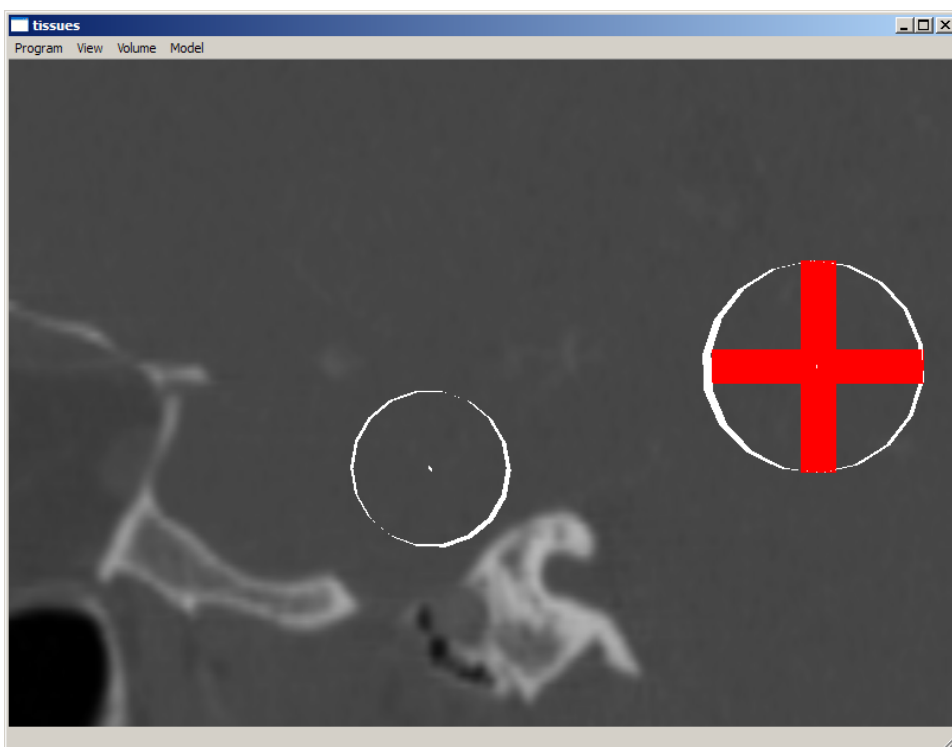
Prostřednictvím menu je možné: ukončit program, měnit mapování intenzit voxelů na odstíny šedi řezů, načíst jiný objem než standardní *head.vlm*, ukládat a načítat modelovanou tkáň nebo ji exportovat do formátu *osg* či *obj*.

Program obsahuje malou chybu, kterou se mi nepodařilo napravit. V pravoúhlém zobrazení v některých případech mizí zobrazovaný řez.





Obrázek 4.2: Perspektivní zobrazení vyznačené tkáně.



Obrázek 4.3: Tkáň zobrazená v pravouhlém zobrazení s označeným bodem průchodu.

## 5 Závěr

V tomto dokumentu byla zkoumána data, jež jsou výstupem medicínských přístrojů pracujících na principech počítačové tomografie a magnetické rezonance. Byly zkoumány také možnosti segmentace tkání a zobrazení určitých druhů tkání. Dále je v teoretickém rozboru nastíněna základní koncepce koncepce programu pro modelování lineárních tkání a oblasti nutné k jeho realizaci.

Dalším krokem byl návrh systému pro interaktivní modelování specifického druhu tkání – tenkých žil, nervů, kanálků a podobných - na základě těchto dat. Program je navržen jako objektově orientovaný. K popisu architektury byla použita notace UML.

V posledním kroku byl tento systém implementován. Nyní umožňují načítání volumetrických dat z medicínských přístrojů a jejich zobrazení pomocí řezů. Je možné zaměřit se na určitý rozsah intenzit původních dat a tak co nejlépe zvýraznit spektrum, ve kterém se nachází tkáň našeho zájmu. Tu je možné modelovat zadáváním bodů, kudy tato tkáň prochází průměry tkáně v těchto bodech. Zbytek tkáně je aproximován co do průběhu i průměru tkáně. Takto vytvořený model je možné uložit, znovu načíst či exportovat jako polygonální síť.

Nabízí se několik možností, jak by se dal program této koncepce dále rozvíjet. Pro efektivní modelování je velice důležité zobrazení volumetrických dat a tak by bylo vhodné navrhnou komfortnější procházení scénou než je tomu v současné verzi. U multiplanárního zobrazení máme přehled o tom, kde se jaké intenzity nacházejí, ale pouze co se týká roviny řezu. Pro zjištění jejich průběhu v prostoru je nutné posouvání řezu. Při použití volume-renderingu máme přehled o prostorovém rozložení, ale pomocí 2D polohovacího zařízení (myši) nemůžeme označit určitý bod v prostoru. Jistě by bylo přínosné prozkoumat možnosti kombinace těchto přístupů. Například zobrazení řezu objemem běžným způsobem s tím, že jeho blízké okolí by se zobrazovalo pomocí některé metody volume-renderingu.

Model tkáně získaný výše popsáním způsobem je pouze přibližný. Možná by se dal nalézt algoritmus, který by automaticky zpřesnil model tak, aby kopíroval modelovanou tkáň.

# Literatura

- [1] Drastich, A.: Zobrazovací systémy v lékařství, 1. vyd., Brno, Rektorát VUT v Brně, 1990.
- [2] Žára, J. aj.: Moderní počítačová grafika, 1. vyd., Brno, Computer press, 2004
- [3] Kršek P.: Přímá tvorba FEM modelů na základě CT/MR dat pro aplikace v biomechanice [dizertační práce], Brno
- [4] Počítačová tomografie [online]. Wikipedie, 2008 cit. [2008-1-3]. Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Počítačová\\_tomografie](http://cs.wikipedia.org/wiki/Počítačová_tomografie)>.
- [5] Magnetická\_rezonance [online]. Wikipedie, 2008 cit. [2008-1-3]. Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Magnetická\\_rezonance](http://cs.wikipedia.org/wiki/Magnetická_rezonance)>.
- [6] Medical Group [online]. FIT VUT v Brně- Ústav počítačové grafiky a multimédií, 2008 cit. [2008-1-3]. Dostupný z WWW: <<http://www.fit.vutbr.cz/research/groups/pgmed>>
- [7] Nečas, O.: Vyhlazování polygonálních modelů [bakalářská práce], Brno
- [8] Page-Jones, M.: Základy objektově orientovaného návrhu v UML, 1. vyd., Praha, Grada Publishing, 2001
- [9] Lorensen, W. E., Cline, H. E.: *Marching Cubes: A high resolution 3D surface construction algorithm*. In: ACM SIGGRAPH Computer Graphics, 21, New-York, 1987, s. 163

# Seznam příloh

Příloha 1. CD se zdrojovými kódy programu, programem zkompilovaným pro platformu Windows a vygenerovanou dokumentací programu.