



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

## SEGMENTACE MEDICÍNSKÝCH OBRAZŮ S VYUŽITÍM Grafové reprezentace

MEDICAL IMAGE SEGMENTATION BASED ON GRAPH CUT WITH SHAPE PRIOR

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Dominika Kozlová

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Roman Jakubíček

BRNO 2016



# Bakalářská práce

bakalářský studijní obor **Biomedicínská technika a bioinformatika**

Ústav biomedicínského inženýrství

**Studentka:** Dominika Kozlová

**ID:** 164210

**Ročník:** 3

**Akademický rok:** 2015/16

## NÁZEV TÉMATU:

### Segmentace medicínských obrazů s využitím grafové reprezentace

#### POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s problematikou zpracování obrazů, zejména segmentace. Prostudujte segmentační metody založené na grafové reprezentaci. 2) Zpracujte literární rešerši na danou problematiku. Obzvláště se zaměřte na grafové metody segmentace a návrhy její architektury s využitím znalosti o tvaru segmentovaného objektu. 3) Navrhněte vhodné postupy pro vytvoření architektury grafu na základě znalosti segmentovaného obrazu. 4) Realizujte a ověřte metodu grafové segmentace v programovém prostředí Matlab na simulovaných datech. 5) Otestujte segmentační metodu na reálných medicínských CT datech. 6) Diskutujte dosažené výsledky a možné využití této metody (square graph-cut ) v oblasti zpracování medicínských obrazů.

#### DOPORUČENÁ LITERATURA:

[1] LÉZORAY, O., GRADY, L. J. Image processing and analysis with graph: theory and practice. Boca Raton: CRC Press, c2012, 537 s. ISBN 978-1-4398-5507-2.

[2] EGGER, Jan, Tina KAPUR, Thomas DUKATZ, Malgorzata KOLODZIEJ, Dženan ZUKIĆ, Bernd FREISLEBEN a Christopher NIMSKY. Square-Cut: A Segmentation Algorithm on the Basis of a Rectangle Shape. PLoS ONE [online]. 2012, roč. 7, č. 2, s. e31064.

**Termín zadání:** 8.2.2016

**Termín odevzdání:** 27.5.2016

**Vedoucí práce:** Ing. Roman Jakubiček

**Konzultant bakalářské práce:**

**prof. Ing. Ivo Provazník, Ph.D., předseda oborové rady**

#### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## ABSTRAKT

Tato práce se zabývá grafovou metodou segmentace obrazu a jejím zdokonalením prostřednictvím znalosti tvaru objektu pro tvorbu architektury grafu (šablony). Popsány jsou základy teorie grafů, ze které grafové segmentační metody vychází. Navržený segmentační algoritmus byl realizován ve 2D provedení s grafickým uživatelským prostředím v MATLABu. Pro segmentaci objemových dat byla metoda rozšířena do 3D. Implementovaná metoda byla testována na simulovaných datech a na reálných CT a MRI snímcích obratlů a mozku. Získané výsledky byly zhodnoceny a porovnány s původní grafovou metodou bez využití šablony.

## KLÍČOVÁ SLOVA

Segmentace obrazu, Teorie grafů, Řez grafem, Šablona

## ABSTRACT

This thesis deals with a graph-based image segmentation and its improvement by using the information about the shape of the object for creating specific graph architecture (template). There are described basics of the graph theory, which is the basis of the graph segmentation methods. Designed segmentation algorithm was realized in 2D with graphical user interface in MATLAB. For segmentation of volume data, the method was extended into 3D. Implemented method was tested on simulated data and on real CT and MRI images of vertebra and brain. Obtained results were evaluated and compared with the original method without using the template.

## KEYWORDS

Image segmentation, Graph Theory, Graph Cut, Template

KOZLOVÁ, D. *Segmentace medicínských obrazů s využitím grafové reprezentace*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2016. 54 s. Vedoucí bakalářské práce Ing. Roman Jakubiček.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Segmentace medicínských obrazů s využitím grafové reprezentace“ jsem vypracovala samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení §11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Ráda bych poděkovala vedoucímu bakalářské práce panu Ing. Romanu Jakubíčkovi za odborné vedení, konzultace, ochotu a cenné rady k práci.

Brno .....

.....

(podpis autora)

# Obsah

Úvod .....	8
1 Teorie grafů .....	9
1.1 Váhováný graf .....	9
1.2 Orientovaný a směřovaný graf .....	9
1.3 Zdrojová a cílová funkce .....	10
1.4 Cesty a konektivita .....	10
1.5 Nejkratší cesta grafu .....	11
1.6 Stromy a minimální kostra grafu .....	11
1.7 Síť a tok grafem .....	12
1.8 Řez grafem .....	13
2 Realizace segmentačního algoritmu .....	14
2.1 Vytvoření grafu .....	14
2.2 Tvorba šablony .....	15
2.3 Segmentační energie .....	17
2.3.1 Regionální vlastnosti .....	17
2.3.2 Hraniční vlastnosti .....	18
2.4 S-t řez .....	19
3 Realizace ve 3D .....	20
3.1 Tvorba šablony krychle .....	20
3.2 Tvorba 3D šablony dle binární předlohy .....	22
4 Grafické uživatelské prostředí .....	24
5 Výsledky .....	26
5.1 Testování na simulovaných datech .....	26
5.1.1 Obdélník .....	26
5.1.2 Pětúhelník .....	28
5.1.3 Hvězda .....	30
5.2 Testování na reálných datech .....	33
5.2.1 Segmentace obratlů .....	33
5.2.2 Segmentace snímků mozku .....	40

5.3 Testování 3D dat.....	43
6 Závěr.....	45
Literatura .....	46

# Úvod

Segmentace patří k základním krokům obrazové analýzy. Obraz je pomocí segmentace rozdělen do vymezených oblastí se společnými vlastnostmi, které se nepřekrývají a společně pokrývají celou oblast obrazu. Vzniklé segmenty představují jednotlivé objekty separované od pozadí v obraze. Výsledkem segmentace je pak obraz s vyznačenými oblastmi segmentů, například barevně či indexy.

Segmentace obrazových dat je využívána k detekci a klasifikaci objektů. U medicínských obrazů získaných pomocí zobrazovacích systémů, jako je počítačová tomografie či magnetická rezonance, přispívá segmentace dat ke stanovení diagnózy. Využívá se zejména pro rozlišení různých typů tkání, 3D modelování a vizualizaci struktur, jako jsou orgány, kosti, cévy i různé patologie.

Publikováno bylo již velké množství různých segmentačních technik vhodných u různých situací. Metody lze rozdělit dle určitých společných znaků do několika oblastí. Některé jsou aplikovány globálně na celý obraz, jiné fungují lépe pouze lokálně na předdefinovaných místech obrazu. Mezi hlavní oblasti segmentačních metod patří kategorie podle homogenity oblastí (př. prahování), metody založené na detekci hran (př. Houghova transformace), regionově orientované techniky (př. metoda rozvodí a metoda dělení a spojování oblastí) a segmentace dle porovnání vzorů [1]. Během posledních let hrají důležitou roli v segmentaci obrazů postupy vycházející z teorie grafů, kdy je obraz převeden na jeho grafovou reprezentaci. Právě segmentací využívající grafovou reprezentaci obrazu a jejím zdokonalením pomocí začlenění informace o tvaru objektu se zabývá tato práce.

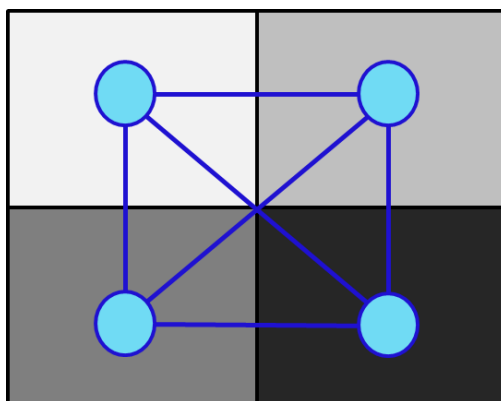
O základech teorie grafů, ze které grafové segmentační metody vychází, pojednává následující kapitola. Ve druhé kapitole je popsána realizace segmentačního algoritmu, konkrétně je vysvětlena tvorba grafu z obrazových dat s využitím znalosti tvaru objektu, dále segmentační energie a provedení řezu grafem. Implementovaná metoda byla také rozšířena do 3D pro segmentaci objemových dat, o čemž pojednává kapitola třetí. Dále bylo pro snadnou interakci uživatele s programem pro 2D segmentaci vytvořeno grafické uživatelské prostředí, popsané v kapitole čtvrté. Realizovaný algoritmus byl testován na simulovaných datech i reálných snímcích obratlů a mozku. Výsledky těchto segmentací jsou uvedeny a diskutovány v páté kapitole.



# 1 Teorie grafů

Grafy jsou struktury v matematice s dlouhou historií, které již byly aplikovány v mnoha vědeckých i technických oblastech a informatice. Dále jsou popsány a vysvětleny základní pojmy nutné pro práci s teorií grafů a její využití.

Grafy představují množinu prvků a párových vztahů mezi těmito prvky. Graf je tedy dvojice množin  $G = (V, E)$ , kde  $V$  představuje množinu uzlů (též vrcholů) a  $E$  vztahy mezi nimi, množinu hran. Každá hrana je podmnožinou všech možných dvojic navzájem různých vrcholů  $E \subseteq V \times V$  [2]. V obrazu odpovídají uzly  $v \in V$  obrazovým pixelům (nebo voxelům) a hrany  $e_{i,j} \in E$  korespondují se zvoleným systémem propojení sousedních pixelů [3]. Graf reprezentující obraz se 4 pixely lze vidět na Obr. 1.1, kde jsou uzly znázorněny modrými kruhy a hrany modrými čarami, které je spojují.



Obr. 1.1: Příklad grafu (modře) reprezentující obraz se 4 pixely (šedě)

## 1.1 Váhováný graf

Každý graf může být váhováný. Váhování vrcholů grafu je funkce  $\hat{w}: V \rightarrow \mathbb{R}$  a váhování hran funkce  $w: E \rightarrow \mathbb{R}$ . Pro zjednodušení zápisu se používá pro váhu uzlu značení  $w_i$  nebo  $w(v_i)$  a pro váhu hrany spojující dva uzly označení  $w_{i,j}$  nebo  $w(v_i, v_j)$ , kde  $i, j$  značí pořadí uzlů v daném grafu.

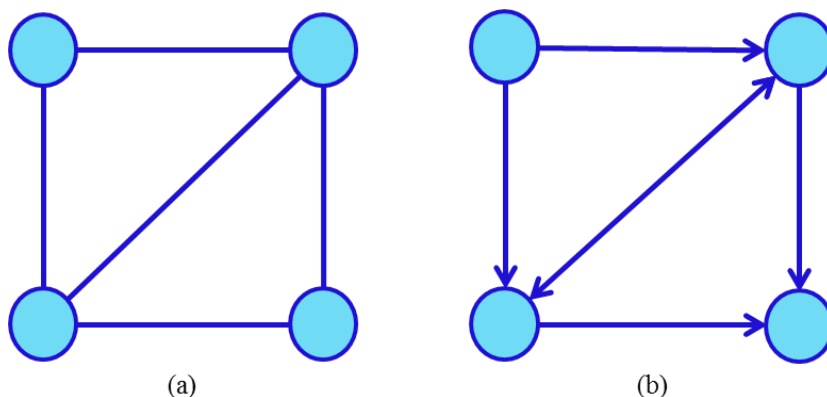
Pokud není váhování specifikováno, všechny váhy uzlů a hran jsou považovány za rovny jedné. V případě, že je váha hrany rovna nule, hrana nepatří do množiny hran daného grafu [2]. Váhy hran mohou být využity pro reprezentaci afinit daných uzlů k objektu zájmu, vzdáleností mezi uzly a podobně. Lze si je představit i jako cenu, kterou je nutné zaplatit za průchod danou hranou [3].

## 1.2 Orientovaný a směrovaný graf

V grafu je každá hrana orientovaná, některé hrany jsou navíc směrované. Orientace hrany znamená, že hrana obsahuje stanovené pořadí uzlů. Hrana je směrovaná, pokud váhování v jednom směru hrany neodpovídá jejímu váhování ve směru opačném, tedy  $w_{i,j} \neq w_{j,i}$ . Graf,

ve kterém je alespoň jedna hrana směřována, je nazýván směřovaný (*directed*) graf. V opačném případě, kdy  $w_{i,j} = w_{j,i}$  se graf nazývá nesměrovaný (*undirected*). Směřovaný graf je více obecný, a tudíž všechny jeho algoritmy mohou být použity i pro nesměrované, ale naopak nikoli.

Orientace hrany nesměrovaného grafu bývá značena šipkou, jak je znázorněno na Obr. 1.2 a slouží k určení znaménka toku touto hranou. Například průtok proudu z uzlu  $v_i$  do uzlu  $v_j$  je kladný a z uzlu  $v_j$  do uzlu  $v_i$  záporný. V tomto smyslu je proud směrovanou hranou obvykle kladný [2].



Obr. 1.2: Příklad grafu (a) neorientovaný (b) orientovaný

### 1.3 Zdrojová a cílová funkce

Zvažujeme dvě funkce  $s, t: E \rightarrow V$ , kde  $s$  je zdrojová funkce (*source*) a  $t$  představuje cílovou funkci (*target*). Hrana  $e_{i,j} = (v_i, v_j) \in E$  má počátek  $s(e_{i,j}) = v_i$ , též zdroj a koncový bod  $t(e_{i,j}) = v_j$ , neboli stok (spotřebič). U hrany  $e_k \in E$  jsou vrcholy  $s(e_k)$  a  $t(e_k)$  nazývány hranice hrany  $e_k$  a výraz  $t(e_k) - s(e_k)$  je mez hrany  $e_k$ . Obecně může mít graf více zdrojů a stoků, v dalším textu je však uvažován pouze jeden zdroj a stok grafu [2].

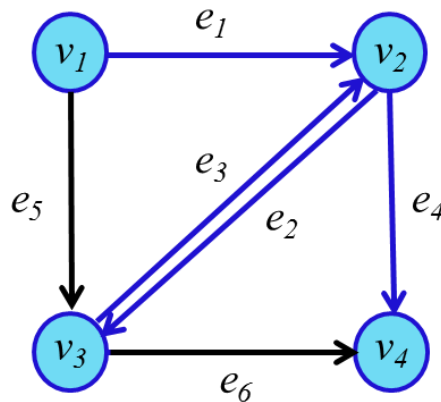
### 1.4 Cesty a konektivita

Grafy utvářejí vztahy mezi uzly, které umožňují definovat, zda jsou dva uzly spojeny řadou párových vztahů. Sled (*walk, chain*) z vrcholu  $v_i$  do  $v_j$  je daná posloupnost uzlů  $\pi(v_1, v_n)$  spojených hranami. Může obsahovat i více opakování jednoho uzlu či hrany. Jestliže je počáteční bod roven koncovému  $v_i = v_j$ , pak je sled uzavřený.

Sled se nazývá tah (*trail*), pokud se v něm každá hrana vyskytuje pouze jednou. Otevřený sled je cesta (*path*), v případě, že se v něm pouze jednou nachází každý uzel. Uzavřený tah tvoří kružnici (*circuit*). Kružnice je nazývána cyklus (*cycle*), pokud jsou všechny její uzly rozdílné.

Dva uzly jsou propojené, pokud existuje sled těchto uzlů, tedy  $\exists \pi(v_i, v_j)$  nebo  $\exists \pi(v_j, v_i)$ . Pokud se v grafu vyskytuje alespoň jeden sled uzlů, nazývá se propojený. Dva

uzly jsou propojeny silně, pokud  $\exists \pi(v_i, v_j)$  i  $\exists \pi(v_j, v_i)$ , tedy pokud mezi nimi existuje sled bodů v jednom i v opačném směru. V nesměrovaném grafu jsou tedy všechny uzly propojené silně [2]. Na Obr. 1.3 je znázorněn sled, tah a cesta v propojeném grafu.



Obr. 1.3: Znázornění propojeného grafu: sled (modře), tah a cesta (černě)

## 1.5 Nejkratší cesta grafu

Směrování v grafech, zejména hledání nejkratší cesty, patří mezi nejstarší a nejběžnější problémy v oblasti teorie grafů. Výpočet nejkratší cesty je třeba u počítačového zobrazování pro interaktivní obrazovou segmentaci.

Uvažujeme-li váhovaný graf  $G = (V, E)$  s váhovou funkcí  $w$  nabývající reálných hodnot pro každou hranu, potom je celková váha sledu uzlů  $l(\pi(v_i, v_j))$  dána součtem vah  $w(e_k)$  všech hran patřících tomuto sledu  $\pi(v_i, v_j)$ . Dle konvence je délka (tj. váha) cesty rovna nekonečnu, pokud uzly  $v_i, v_j$  nejsou spojeny a celková váha sledu  $l(\pi(v_i, v_j))$  je pak rovna nule.

Minimální délka mezi dvěma uzly je získána pomocí minima celkové váhy sledu. Sled s minimální vzdáleností je nazýván nejkratší cesta (*shortest path*), která nemusí být jen jedna. Pro výpočet nejkratší cesty mohou být použity různé algoritmy pro různé situace, například pro její hledání z jednoho uzlu ke všem zbylým (např. Dijkstraův algoritmus) nebo pro cestu mezi všemi páry uzlů [2].

## 1.6 Stromy a minimální kostra grafu

Strom je nesměrovaný propojený graf bez cyklů, u něhož odstranění jakékoli hrany vede k rozpojení grafu a vytvoření nové hrany vede k vytvoření cyklu. Částečný graf propojeného grafu se nazývá kostra grafu. Existuje alespoň jedna kostra pro každý propojený graf, a pokud navíc minimalizuje cenu (váhy) grafu, nazývá se minimální kostra grafu (*minimum spanning trees*).

Pro řešení problému minimální kostry grafu existuje několik algoritmů, například Primův algoritmus, který začíná od hrany s minimální hodnotou vah a iterativně přidává

do stromu hrany s minimálními vahami mezi všemi možnými hranami pro zachování neacyklického stromu. Hledání minimální kostry grafu je využíváno například pro obrazové segmentace a hierarchickou reprezentaci obrazu [2].

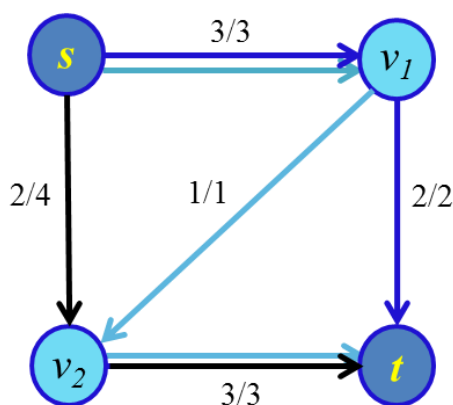
## 1.7 Síť a tok grafem

Transportní síť je uspořádaná pětice  $(V, E, s, t, c)$ , kde  $(V, E)$  je váhovaný orientovaný graf, u kterého existuje jeden počáteční uzel  $s \in V$ , nazývaný zdroj (*source*), kterému žádný uzel nepředchází a jeden koncový uzel  $t \in V$ , nazývaný stok (*sink*), po kterém žádný uzel nenásleduje [4]. Nezáporná váha směřované hrany  $e_{i,j}$  se nazývá kapacita hrany  $c(e_{i,j})$ .  $A^+(v_i) = \{e_{i,j} \in E\}$  značí množinu dovnitř směřovaných hran do bodu  $v_i$  a obdobně  $A^-(v_i) = \{e_{i,j} \in E\}$  značí množinu hran směřujících z bodu  $v_i$  ven. V grafu musí existovat alespoň jedna cesta ze zdroje do stoku.

Funkce  $\varphi: E \rightarrow \mathbb{R}^+$  se nazývá tok sítě, pokud každý uzel  $v_i \notin \{v_s, v_t\}$  splňuje zachování podmínky, že výsledný součet proudů v uzlu je nulový (viz následující rovnice), též zvanou jako Kirchhoffův zákon proudu [2].

$$\sum_{e_{j,i}} \varphi(e_{j,i}) - \sum_{e_{i,j}} \varphi(e_{i,j}) = 0 \quad (1.1)$$

Tok hranou je množství, které je hranou právě přenášeno. Tok dané hrany musí být menší než kapacita hrany  $c(e)$ , zároveň nesmí nabývat záporných hodnot. Rezerva hrany značí, kolik kapacity hrany zbývá, neboli jaké množství by mohlo hranou ještě protékat. Tok v uzlech nevzniká ani nezániká, ztráty v uzlech se tedy neuvažují. Velikost toku v celém grafu je rovna součtu toků ze zdroje do stoku. Cílem je zajistit maximální tok ze zdroje do stoku, ne naplnit celou síť. Na Obr. 1.4 je znázorněn příklad sítě s tokem, různě barevnými šipkami jsou zobrazeny možné cesty ze zdroje  $s$  do stoku  $t$ . Hrany jsou značeny hodnotou toku před lomítkem a kapacitou hrany za lomítkem.



Obr. 1.4: Tok sítě a možné cesty grafem (černě, světle a tmavě modře)

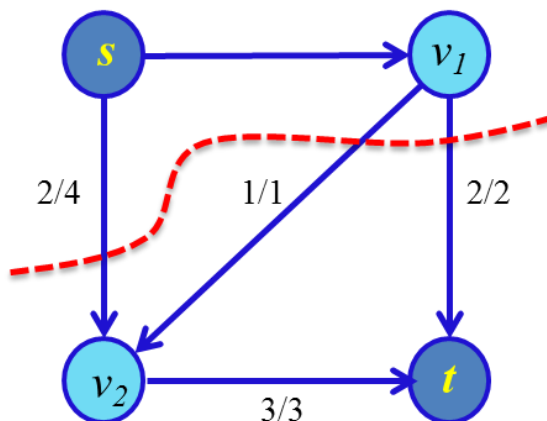
Základní problém, který je řešen při segmentaci, je hledání toku co největší velikosti, které může být dosaženo, tedy maximálního toku (*max flow*) [4]. Pro jeho výpočet slouží například Fordův-Fulkersonův algoritmus (viz *Ford-Fulkerson algoritm* [5]).

## 1.8 Řez grafem

Dle terminologie je *s-t* řez podmnožinou hran  $C \subset E$ , jejichž odstranění rozdělí uzly do dvou podmnožin  $S$  a  $T$ . Řezem tedy vznikají dvě podmnožiny vrcholů patřících buď ke zdroji, nebo ke stoku [2]. U segmentace obrazu je řez jeho grafem využíván pro rozdělení uzlů zastupujících pixely do množiny přiřazené ke zdroji, který reprezentuje objekt a ke stoku, který představuje pozadí obrazu. Řezem jsou tedy získány hranice segmentovaného objektu.

Kapacita (váha) řezu je rovna součtu kapacit všech jeho hran [3]. Hlavním problémem je hledání řezu co nejmenší možné kapacity, tedy minimálního řezu (*min cut*) [4]. Minimální kapacita řezu vytvoří segmentaci, která je optimální pro zahrnuté vlastnosti, které jsou začleněny v rámci vah hran. Minimální *s-t* řez grafu se dvěma koncovými body  $s$  a  $t$  patří mezi klasické kombinatorické úlohy, které bývají řešeny různými algoritmy [3].

U každého váhovaného směřovaného grafu existuje maximální tok a minimální řez. Velmi využíván je fakt, že velikost maximálního toku ze zdroje do stoku je rovna celkové kapacitě minimálního řezu [2], [4]. Na Obr. 1.5 je zobrazen příklad *s-t* řezu grafem se dvěma vnitřními vrcholy  $v_1, v_2$ . Celková hodnota zázorněného řezu je rovna součtu kapacit jeho hran, a tedy číslu 7.



Obr. 1.5: Příklad *s-t* řezu grafem (červeně)

Metody založené na grafových řezech získaly mnoho pozornosti díky tomu, že využívají hraniční i oblastní informace a dosahují globálně optimálních výsledků pro energiovou funkci, popsanou v kapitole 2.3. Navíc lze tyto postupy aplikovat i na  $N$ -dimenzionální obrazy. Mezi standartní algoritmy grafových řezů patří například tzv. „Max-Flow“ nebo „Push-Relabel“ [6].

## 2 Realizace segmentačního algoritmu

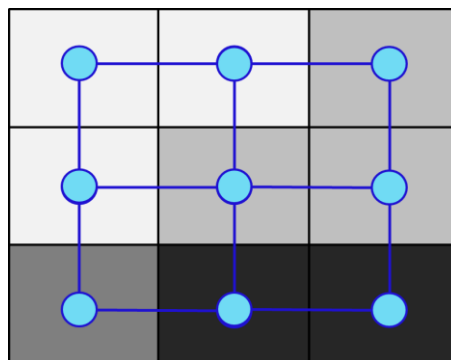
Implementovaná segmentační metoda vychází z článku [7], kde je představen „Square-Cut“ algoritmus využívající grafovou reprezentaci s návrhem její architektury na základě znalosti tvaru segmentovaného objektu.

Hlavním bodem grafových segmentačních metod je utvoření váhovaného grafu  $G = (V, E)$  s množinou uzlů  $V$  a množinou hran  $E$ , kde uzly  $v \in V$  odpovídají pixelům obrazu a hrany  $\langle v_i, v_j \rangle \in E$  spojují uzly  $v_i, v_j$  dle zvoleného systému. Každá hrana grafu má svou váhu, která může představovat určitou míru preference, zda odpovídající pixely patří do objektu zájmu, nebo rozdíl jasu uzlů, které spojuje. V závislosti na aplikaci a použitém algoritmu může být konstruovaný graf směřovaný či nesměřovaný. Zde je využit graf směřovaný.

Typické grafové algoritmy, které jsou využívány pro segmentaci obrazu, zahrnují minimální kostry grafů (*minimum spanning trees*), hledání nejkratší cesty (*shortest paths*) a grafové řezy (*graph-cuts*), které jsou využity právě v této práci [2].

### 2.1 Vytvoření grafu

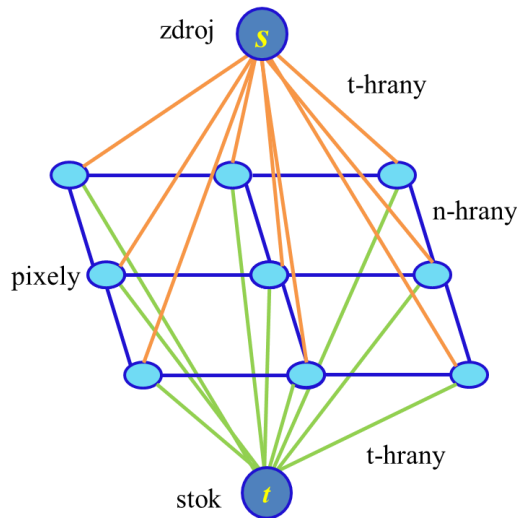
Konstruovaný graf obrazu má podobu pravoúhlé mřížky uzlů a hran. Vnitřní vrcholy grafu reprezentují jednotlivé pixely obrazu, imaginární vrcholy zdroj  $s$  a stok  $t$  představují označení objektu a pozadí. Sousední vrcholy jsou spojeny hranami, které se nazývají n-hrany (z anglického slova *neighborhood*). K tvorbě n-hran grafu může být využito 4-okolí nebo 8-okolí pro definici sousedních vrcholů. U 4-okolí je vrchol spojen n-hranami se čtyřmi sousedními vrcholy, jak lze vidět na Obr. 2.1. N-hrany obsahují informaci v podobě vah  $w(e)$  o rozdílu hodnot jasu dvou sousedních pixelů, které spojují.



Obr. 2.1: Spojení uzlů n-hranami do 4-okolí

Do grafu jsou přidány dva imaginární terminální vrcholy zdroj  $s$  a stok  $t$ , které jsou spojeny tzv. t-hranami (z anglického slova *terminal*) se všemi pixely obrazu, jak je znázorněno na Obr. 2.2. Každý pixel  $p$  má dvě t-hrany spojující ho s každým terminálem,

tedy se zdrojem  $\{p, S\}$  a se stokem  $\{p, T\}$ . Všem t-hranám jsou přiřazeny nezáporné váhy  $w(e)$  na základě pravděpodobnosti výskytu jasu pixelu v histogramu objektu či pozadí [7] [8].



Obr. 2.2: Grafová reprezentace obrazu

Výpočty vah n-hran i t-hran vychází z minimalizace segmentační energie, která je popsána v kapitole 2.3.

## 2.2 Tvorba šablony

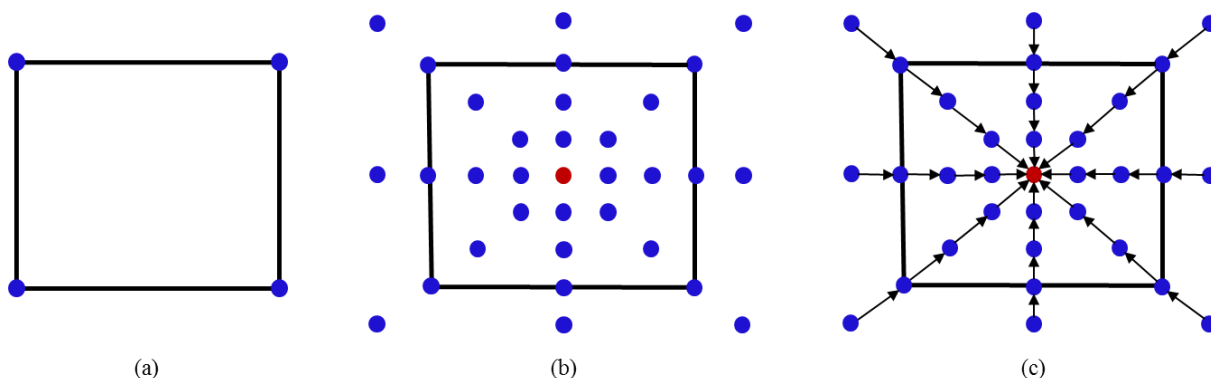
Metody založené na grafových řezech lze klasifikovat do tří kategorií: „speed up-based graph cut“, „interactive-based graph cut“ a „shape prior-based graph cut“, který využívá informaci o tvaru objektu [9]. Poslední uvedený algoritmus rozděluje uzly grafu v obrazu rovnoměrně a až poté je přidán parametr pro preferování určitého tvaru u segmentačního řezu. Tento postup ale nedovoluje preferovat určitou strukturu především v případech, kdy jsou části objektu jasem prakticky nerozeznatelné od pozadí. Tento problém je řešen pomocí zadaného tvaru objektu při tvorbě grafu, neboť pak vzorkované uzly grafu mohou být dle vytvořené šablony rozčleněny nerovnoměrně [7].

Realizovaný algoritmus pak začíná nastavením orientovaného grafu šablony z počátečního bodu (*seed point*) definovaného uživatelem. Počáteční bod by měl být umístěn uvnitř objektu pro segmentaci, ideálně zhruba uprostřed (v těžišti). Pro tvorbu je také nutné znát souřadnice krajních bodů objektu po směru hodinových ručiček. Tyto krajní body musí vhodně popisovat objekt, který chceme segmentovat. Například u čtvercové šablony jsou zadány 4 souřadnice rohových bodů, u pětiúhelníku bude 5 bodů a podobně. Zadané souřadnice jsou následně spojeny bod po bodu do uzavřeného obrysu, jak lze vidět na Obr. 2.3 (a) pro čtvercovou šablonu.

Poté jsou z počátečního bodu vyslány paprsky radiálně přes vytvořený obrys. Uzly grafu jsou pak vzorkovány podél těchto paprsků, jak je znázorněno na Obr. 2.3 (b). Průsečík jednoho paprsku s obrysem šablony poskytuje vzdálenost mezi uzly jen pro tento paprsek. Všechny paprsky mají stejný počet uzlů, ale mohou mít různé vzdálenosti mezi uzly.

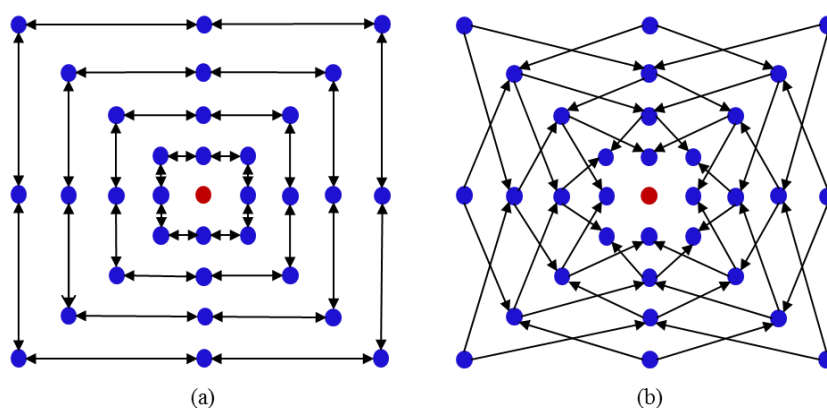
Pro průsečíky, které se nachází blíže k výchozímu bodu, budou vzdálenosti mezi uzly menší a naopak. Rozmístěné uzly v grafu budou mít odpovídající množinu hran. N-hrany spojují uzly šablony a t-hrany uzly se zdrojem  $s$  a stokem  $t$  pro výpočet s-t řezu.

Existují dva typy váhovaných n-hran mezi uzly, z-křivky  $A_z$  a r-křivky  $A_r$ . Z-křivky, znázorněny na Obr. 2.3 (c), jsou mezi uzly daného paprsku, přičemž  $z$  udává počet navzorkovaných bodů podél jednoho paprsku. Zajišťují, aby všechny uzly podél obrysu objektu v grafu byly zahrnuty do vytvoření uzavřené množiny, tedy aby byl vnitřek objektu daným způsobem oddělen od zevnějšku.



Obr. 2.3: Konstrukce grafu čtvercové šablony. (a) Spojené rohové body šablony. (b) Uzly šablony. (c) Z-křivky [7].

R-křivky  $A_r$  spojují uzly mezi sousedními paprsky,  $r$  značí počet paprsků vycházejících z počátečního bodu šablony. R-křivky omezují množinu možných segmentací a určují variabilitu výsledného obrysu objektu pomocí parametru  $\Delta_r$ . Čím je hodnota  $\Delta_r$  vyšší, tím je větší i počet možných segmentací. R-křivky šablony s parametrem  $\Delta_r = 0$  a  $\Delta_r = 1$  jsou zobrazeny na Obr. 2.4.



Obr. 2.4: R-křivky čtvercové šablony. (a) Parametr  $\Delta_r = 0$ . (b) Parametr  $\Delta_r = 1$  [7].

Po konstrukci grafu, je vybrána v grafu uzavřená množina s minimální cenou řezu. S-t řez vytvoří optimální segmentaci objektu pod vlivem parametru  $\Delta_r$ . Hodnota tohoto parametru rovna nule zajistí, že výsledkem segmentace bude přesně tvar předdefinované šablony (čtverce). S rostoucí hodnotou  $\Delta_r$  je možná větší variabilita tvaru objektu [7].



## 2.3 Segmentační energie

Pevná omezení (*hard constrains*) určují dvě podmnožiny pixelů, první patří objektu „*obj*“ (*object*) a druhá náleží pozadí „*bkg*“ (*background*). Tyto pixely jsou označeny uživatelem jako tzv. semínka (*seeds*) a poskytují informace o hodnotě jasu pixelů objektu a pozadí. Segmentační metoda poté na základě definovaných omezení určí rozdělení zbylých neoznačených pixelů.

Omezení samotná nejsou pro dobrou segmentaci dostačující, proto se využívá kombinace omezení a segmentační energie. Tato energie obsahuje dvě složky, hraniční (*boundary term*) a oblastní, též regionální složku (*region term*), které jsou popsány níže.

Mějme množinu pixelů  $P$  a množinu  $N$ , která představuje jejich sousední systém (zde 4-okolí) všech neseřazených párů  $\{p, q\}$  sousedních pixelů. Dále mějme binární vektor  $A$ , jehož složky  $A_p$  specifikují zařazení pixelů  $p$  v množině  $P$ . Binární vektor může nabývat hodnot  $A \in \{0,1\}$ , kde pixely  $p$  náležící objektu  $A_p = „obj“$  získají hodnotu rovnu 1 a všechny  $p$  náležící pozadí  $A_p = „bkg“$  hodnotu 0. Každé  $A_p$  tedy může patřit buď objektu, nebo pozadí, a tudíž vektor  $A$  definuje segmentaci. Celková segmentační energie, která zahrnuje hraniční a regionové vlastnosti, pak představuje váhovou funkci:

$$E(A) = \lambda \cdot R(A) + B(A), \quad (2.1)$$

kde koeficient  $\lambda \geq 0$  udává relativní důležitost oblastní vlastnosti  $R(A)$  oproti vlastnosti hraniční  $B(A)$ . Poměr těchto složek má značný vliv na výsledek segmentace. V případě, že zvolíme hodnotu  $\lambda$  vyšší, bude segmentace provedena více dle intenzity pixelů než jejich vzájemné souvislosti a naopak. Cílem je nalézt segmentaci odpovídající globálnímu minimu segmentační energie  $E(A)$  z uvedené rovnice, která splňuje uživatelem definovaná pevná omezení [6].

### 2.3.1 Regionální vlastnosti

Regionální vlastnosti segmentační energie udávají míru příslušnosti pixelu k objektu či pozadí. Hodnota oblastní složky segmentační energie  $R(A)$  je vyjádřena jako:

$$R(A) = \sum_{p \in P} R_p(A_p), \quad (2.2)$$

kde  $R_p(A_p)$  představuje sankci za přiřazení pixelu do objektu nebo pozadí.  $R(A)$  udává, jak intenzita pixelů  $p$  vyhovuje histogramu objektu a pozadí. Je tedy nutné znát tyto histogramy, které jsou vytvořeny z hodnot pixelů zařazených do podmnožin objektu či pozadí dle počátečního určení pevných omezení uživatelem. Hodnoty sankcí regionálních vlastností jsou pak získány z rovnic:

$$R_p(„obj“) = -\ln Pr(I_p | „obj“), \quad (2.3)$$

$$R_p(„bkg“) = -\ln Pr(I_p | „bkg“), \quad (2.4)$$

kde  $Pr(I_p | „obj“)$  je pravděpodobnost, že intenzita pixelu náleží do histogramu objektu a  $Pr(I_p | „bkg“)$  pravděpodobnost, že intenzita pixelu náleží do histogramu pozadí.

Do grafu obrazu se regionální vlastnosti promítnou jako váhy t-hran, tedy  $R_p(„obj“)$  jako váhy t-hran mezi jednotlivými pixely a zdrojem a  $R_p(„bkg“)$  mezi pixely a stokem. Pro pixely označené jako semínka objektu jsou nastaveny hodnoty vah t-hran jiné. Výsledné váhy všech t-hran jsou uvedeny v následující tabulce [6].

Tabulka 1: Váhy t-hran [6]

Hrana	Váha	Pro
$\{p, T\}$	$\lambda \cdot R_p(„bkg“)$	$p \in P, p \notin „obj“ \cup „bkg“$
	$\infty$	$p \in „obj“$
	0	$p \in „bkg“$
$\{p, S\}$	$\lambda \cdot R_p(„obj“)$	$p \in P, p \notin „obj“ \cup „bkg“$
	0	$p \in „obj“$
	$\infty$	$p \in „bkg“$

### 2.3.2 Hraniční vlastnosti

Hodnoty hraničních vlastností segmentace  $B(A)$ , které vyjadřují rozdíl mezi sousedními pixely, jsou definovány:

$$B(A) = \sum_{\{p,q\} \in N} B_{p,q} \cdot \delta(A_p, A_q), \quad (2.5)$$

$$\delta(A_p, A_q) = \begin{cases} 1 & A_p \neq A_q \\ 0 & \text{jinak} \end{cases}, \quad (2.6)$$

kde koeficient  $\delta(A_p, A_q)$  určuje, že jsou počítány jen pixely na hranici objektu a pozadí.  $B_{\{p,q\}}$  větší než nula interpretuje souvislost mezi pixely  $p$  a  $q$ , hodnoty jsou vypočteny dle rovnice:

$$B_{\{p,q\}} = \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \cdot \frac{1}{\text{dist}(p,q)}, \quad (2.7)$$

kde  $I_p$  a  $I_q$  značí intenzity pixelů  $p$  a  $q$ ,  $\sigma$  je rozptyl intenzit pixelů v obraze a  $\text{dist}(p,q)$  eukleidovská vzdálenost daných pixelů. Hodnota  $B_{\{p,q\}}$  je vyšší, pokud jsou intenzity pixelů

$p$  a  $q$  podobnější a je téměř nulová, pokud jsou velmi rozdílné. V grafu obrazu jsou pak hraniční vlastnosti v podobě hodnot vah  $n$ -hran mezi jednotlivými pixely [6].

## 2.4 S-t řez

Získané hodnoty  $t$ -hran a  $n$ -hran následně zpracovává „min-cut/max-flow“ algoritmus Y. Boykova a V. Kolmogorova [10], který vyhledá maximální tok sítě a zjistí minimální  $s$ - $t$  řez grafu, tedy výsledek segmentace. Použitá knihovna pro implementaci tohoto algoritmu do MATLABu je dostupná na stránkách MathWorks® [11].

Všechny uzly grafu jsou algoritmem rozděleny do dvou podmnožin „*obj*“ a „*bkg*“, které reprezentují objekt a pozadí obrazu. Výsledkem je matice stejných rozměrů jako výchozí obraz se značením pixelů objektu hodnotou 1 a pozadí 0.

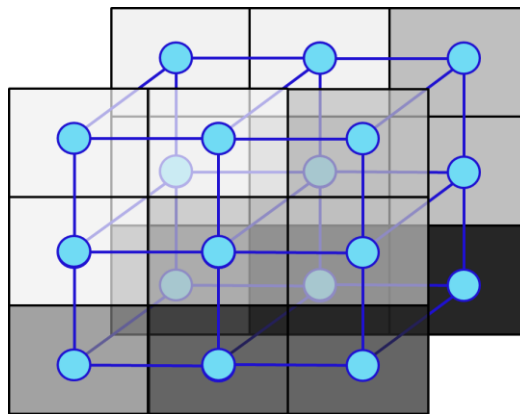
Tento proces je dosažen minimalizováním segmentační energie, neboť by měla být minimální právě na hranici objektu a pozadí [9]. Případné nedokonalosti výsledku metody jsou tedy v přímém vztahu s váhami hran grafu, čehož může být využito k nastavování úprav segmentace [6].

Jak bylo vysvětleno, modifikovaná grafová metoda, na rozdíl od původní, nevyužívá pro tvorbu specifické architektury grafu všechny pixely obrazu, ale do výsledné šablony spojuje jen některé. Provedením  $s$ - $t$  řezu takto zkonstruovaným grafem proto není získáno rozdělení všech pixelů do podmnožin objektu a pozadí, ale zařazeny jsou jen ty, které tvoří šablonu. Pro získání konečného výsledku je tedy navíc nutné provést následující operace. Nejprve jsou nalezeny poslední vysegmentované uzly každého paprsku šablony, které náleží objektu. Tyto body jsou spojeny do výsledného obrysu a celý prostor uvnitř je vyplněn. Získán je tedy ucelený tvar objektu a případné otvory uvnitř tohoto útvaru nejsou brány v úvahu. Z tohoto důvodu není metoda vhodná u objektů s různými otvory, jejichž segmentace je vyžadována.

### 3 Realizace ve 3D

Pro možnost segmentace objemových dat, získaných z MRI či CT vyšetření, namísto pouhé 2D segmentace v jednotlivých řezech obrazu, byla grafová metoda i s využitím šablony rozšířena do třetí dimenze.

Obdobně jako ve 2D realizaci, algoritmus vytvoří směřovaný graf se dvěma terminálními uzly (tzv. s-t síť), uzly grafu ale odpovídají obrazovým voxelům. Váhování t-hran terminálních uzlů, které se zdrojem  $s$  a stokem  $t$  opět spojují všechny uzly, reprezentují afinitu daného voxelu k objektu (zdroji) a k pozadí (stoku). Množina váhovaných n-hran mezi neterminálními uzly grafu je tvořena v této práci dle systému 6-okolí, které je ilustrováno na Obr. 3.1. Jak lze vidět, každý uzel je oproti 2D provedení navíc spojen se dvěma uzly, které leží v jiných řezech 3D obrazu. Mezi další typy sousedství patří například 18-okolí či 26-okolí. Po konstrukci grafu je spočítán minimální s-t řez, který rozdělí uzly grafu do dvou podmnožin reprezentujících objekt či pozadí.



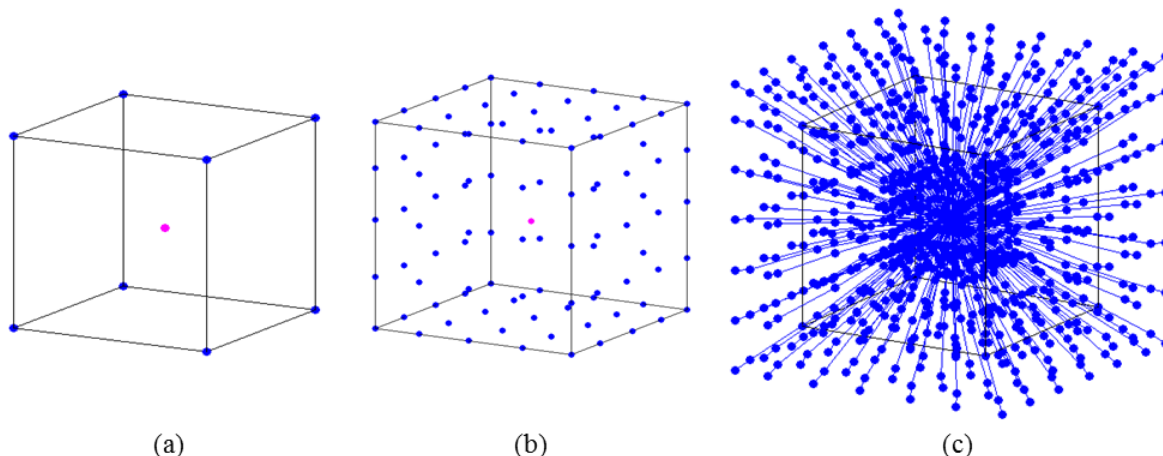
Obr. 3.1: Znárodnění spojení voxelů obrazu n-hranami do 6-okolí

#### 3.1 Tvorba šablony krychle

Pro tvorbu 3D šablony byl výchozí článek [8], prezentující „Cube-Cut“, grafovou metodu využívající šablonu krychle pro objemovou segmentaci obrazů. Tvorba šablony je oproti 2D provedení podobná, ale parně složitější ve spojování uzlů grafu a vzorkování stran šablony. Algoritmus je ve 3D provedení též více náročný z hlediska zpracování velkého objemu dat.

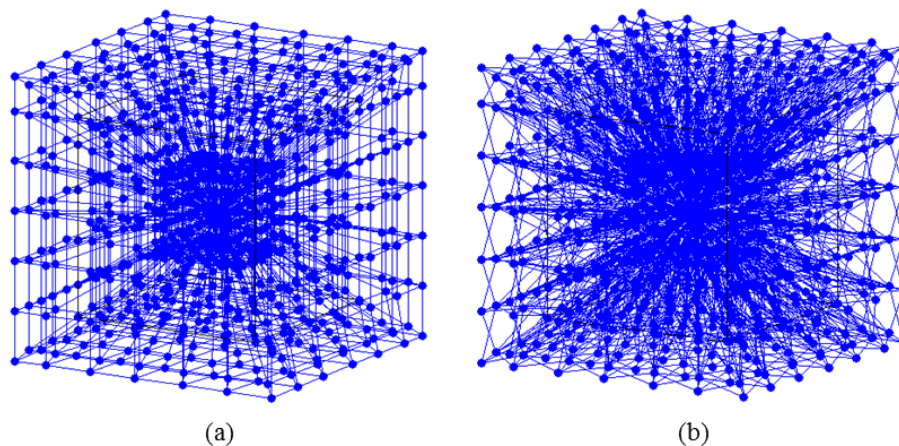
Segmentační algoritmus vytvoří orientovaný graf, jehož uzly patří do podmnožiny voxelů korespondujících s definovaným kubickým tvarem. Výchozími body pro tvorbu šablony jsou opět počáteční bod zhruba uprostřed segmentovaného objektu a krajní body, určené uživatelem. Souřadnice krajních bodů jsou zadány po směru hodinových ručiček pro přední a následně zadní stranu objektu. Zadané souřadnice jsou poté navzájem propojeny do uzavřeného obrysu, znázorněném na Obr. 3.2 (a). U utvořeného obrysu jsou uzly vzorkovány na spojnicích mezi krajními body a následně rovnoměrně na všech stěnách krychle dle zvoleného kroku mezi paprsky šablony, jak lze viděn na Obr. 3.2 (b).

Přes navzorkovaný povrch krychle jsou poté z počátečního bodu vyslány paprsky do vzdálenosti dle zvolené velikosti přesahu. Podél těchto paprsků jsou dle stanoveného kroku vzorkovány uzly tak, aby voxely ve stejné vrstvě vždy utvářely tvar krychle. Všechny paprsky mají tedy stejný počet uzlů, ale vzdálenosti mezi nimi bývají různé. Pokud je  $k$  uzlů na každém paprsku, pak je utvořeno  $k - 1$  krychlí různé velikosti se středem ve výchozím uzlu všech paprsků, tedy v počátečním bodě. Na povrchu každé utvořené krychle jsou voxely rozmístěny ve stejných vzdálenostech od sebe, se zvětšujícími objemy krychlí rostou i vzdálenosti voxelů na jejich povrchu. Uvnitř paprsků jsou spojeny body tzv. z-křivkami. Všechny uzly utvořené šablony s propojením z-křivkami jsou zobrazeny na Obr. 3.2 (c).



Obr. 3.2: Konstrukce grafu šablony krychle. (a) Spojení krajních bodů do obrysu krychle. (b) Uzly obrysu šablony. (c) Všechny uzly šablony spojené z-křivkami.

R-křivky opět spojují uzly mezi sousedními paprsky, horizontálně i vertikálně. Dle zvoleného systému je jimi každý uzel propojen se 4 uzly sousedních paprsků. Variabilita výsledného obrysu objektu oproti tvaru krychle je určena pomocí parametru  $\Delta_r$ . Při hodnotě parametru  $\Delta_r = 0$  má výsledek segmentace odpovídat tvaru krychle, tudíž jsou uzly mezi různými paprsky propojeny na povrchu zvětšujících se krychlí, jak lze vidět na Obr. 3.3 (a). Se zvyšující se hodnotou tohoto parametru je pak možná vyšší variabilita tvaru objektu, jak lze pozorovat u propojení r-křivkami s parametrem  $\Delta_r = 1$  na Obr. 3.3 (b).

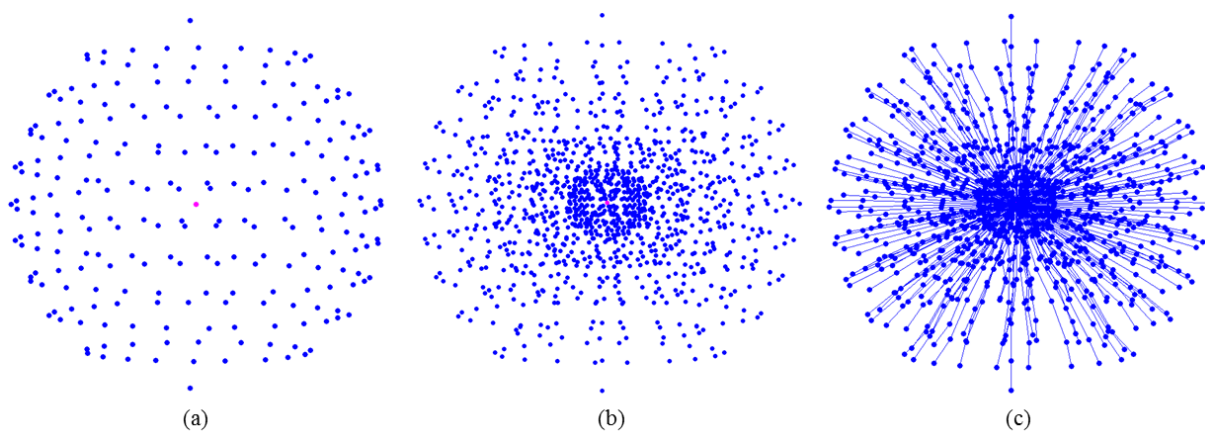


Obr. 3.3: Šablona krychle s r-křivkami. (a) Parametr  $\Delta_r = 0$ . (b) Parametr  $\Delta_r = 1$ .

## 3.2 Tvorba 3D šablony dle binární předlohy

Šablona krychle vyžaduje zadání souřadnic krajních bodů přední a následně zadní stěny. Obdobným způsobem by bylo možné vytvořit například šablonu kvádrů. Avšak utváření dalších, více komplikovaných tvarů by vedlo k pracnému či zdlouhavému zadávání krajních bodů i náročnému následnému zpracování. Z toho důvodu byl pro další tvorbu zvolen postup vycházející ze zvolené předlohy, která má podobu 3D binárního obrazu.

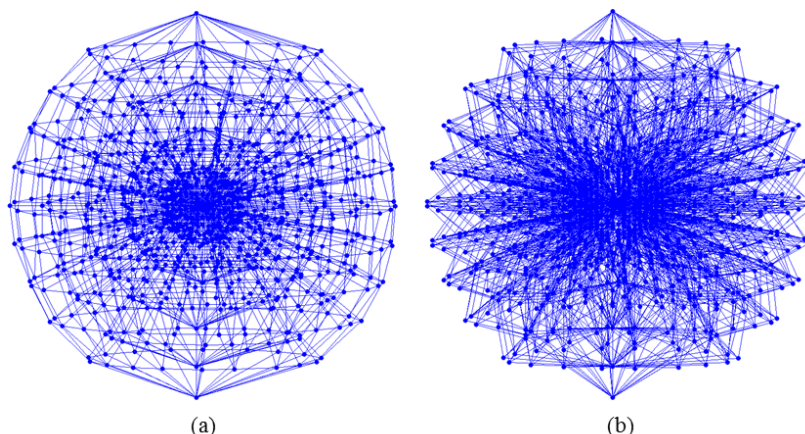
Nejprve je opět třeba zadat počáteční bod zhruba uprostřed segmentovaného objektu. Dále je vybrána binární předloha, ze které jsou získány krajní body hranovou detekcí, a to jen v řezech dle zvoleného kroku mezi paprsky šablony. Hrana je místo v obrazu s dostatečně velkou a rychlou změnou jasů. Hranová detekce je provedena na základě tzv. Sobelova lokálního operátoru, který aproximuje první derivaci diferencemi pro hledání maximálního gradientu [1]. Získán je binární obraz, ve kterém bílé pixely představují pozice hran a černé pixely místa, kde hrana není. Nalezené uzly jsou poté se stejným krokem vzorkovány. Výsledné krajní body lze vidět u šablony kulovitého tvaru na Obr. 3.4 (a). Následně jsou pozice získaných bodů posunuty podle určeného počátečního bodu v obrazu do požadované vzdálenosti. Dále jsou z počátečního bodu vyslány paprsky do všech krajních bodů a podél nich vzorkovány zbylé uzly šablony dle stanoveného kroku mezi nimi, jak je znázorněno na Obr. 3.4 (b). Podél paprsků opět spojují uzly vedle sebe z-křivky, zobrazené na Obr. 3.4 (c).



Obr. 3.4: Konstrukce grafu šablony koule. (a) Počáteční bod růžově a krajní body modře. (b) Všechny uzly šablony. (c) Z-křivky.

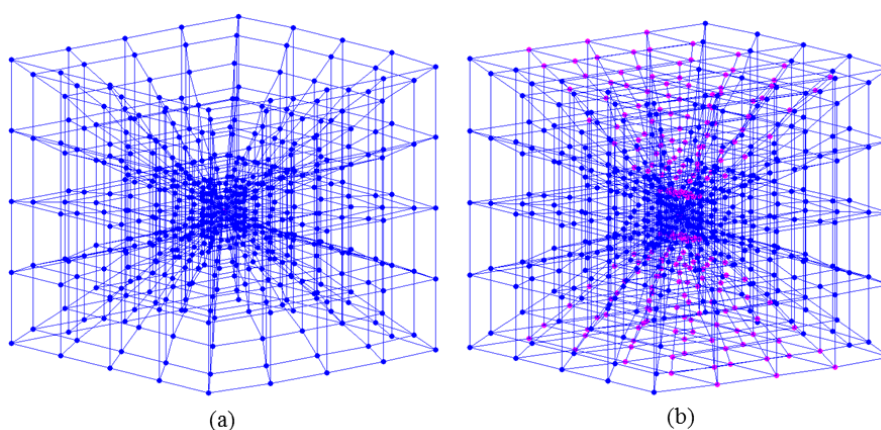
Výsledná šablona je získána propojením uzlů sousedních paprsků r-křivkami dle parametru  $\Delta_r$ . Graf zkonstruované šablony koule je zobrazen na Obr. 3.5 (a) s parametrem  $\Delta_r = 0$  a na Obr. 3.5 (b) s  $\Delta_r = 1$ . Jak lze pozorovat na Obr. 3.5 (a), v porovnání se šablonou krychle není každý uzel spojen r-křivkami jen se 4 okolními uzly, ale propojení může mít více, jak je tomu například u prvního a posledního bodu (na obrázku horní a dolní uzel), které jsou r-křivkami spojeny se všemi body v sousedním řezu. Pokud se vyskytuje ve vedlejší vrstvě stejný počet bodů, jsou mezi sebou postupně propojeny r-křivkami každý s každým (jak je tomu i u šablony krychle). Jelikož ale v sousedním řezu šablony stejný počet

uzlů být nemusí, pak je v tomto případě ke každému bodu z vrstvy, kde se jich nachází více, nalezen vždy jeden nejbližší souseď, se kterým je spojen.



Obr. 3.5: Šablona koule s  $r$ -křivkami. (a) Parametr  $\Delta_r = 0$ . (b) Parametr  $\Delta_r = 1$ .

Jak si lze povšimnout u vytvořené šablony koule na Obr. 3.5, v prvním a posledním řezu je vždy jen jeden její bod, výsledkem je tedy ucelený tvar. Avšak vyskytují se případy, kdy je vhodné k získaným krajním bodům hranovou detekcí z binární předlohy přidat body do plochy přední či zadní stěny, proto byla tato možnost do procesu tvorby přidána. Například právě u tvaru krychle či kvádrů je třeba doplnit uzly do prvního i poslední plochy šablony, neboť je hranovou detekcí získán vždy jen obrys. Tento problém by bylo také možné řešit i provedením hranové detekce 3D Sobelovými maskami. Uzly první a poslední plochy by ale bylo třeba navzorkovat obdobným způsobem, jako jsou přidávány v uvedeném řešení. Rozdíl mezi šablonou krychle zkonstruovanou jen z nalezených krajních bodů z binární předlohy a vytvořenou i z přidanych uzlů přední i zadní stěny je demonstrován na Obr. 3.6.

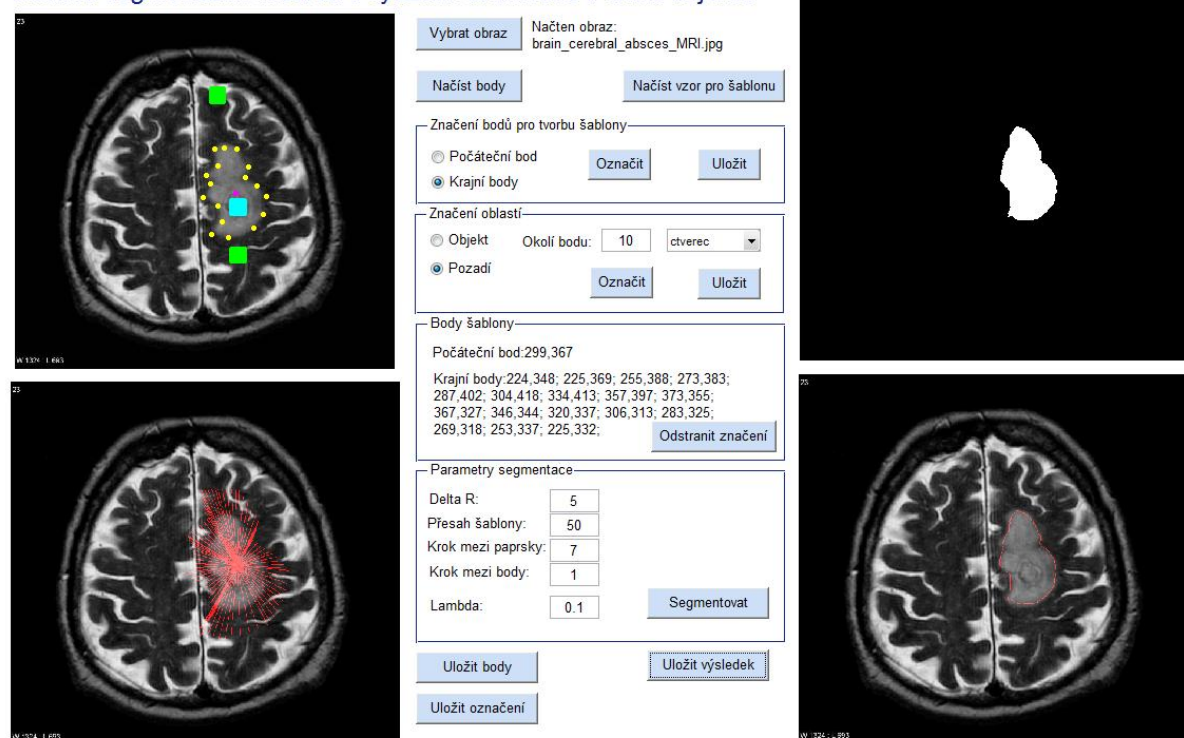


Obr. 3.6: Zkonstruovaná šablona krychle dle binární předlohy. (a) Tvorba z nalezených krajních bodů. (b) Po přidání krajních bodů přední a zadní stěny, znázorněných růžovou barvou.

## 4 Grafické uživatelské prostředí

Realizovaná segmentační metoda není zcela automatická, z toho důvodu bylo pro snadnou interakci uživatele s programem vytvořeno grafické uživatelské prostředí. Jeho názorná ukázka je představena na Obr. 4.1 s výsledkem segmentace cerebrálního abscesu na MRI snímku mozku v axiálním pohledu.

Grafová segmentační metoda s využitím informace o tvaru objektu



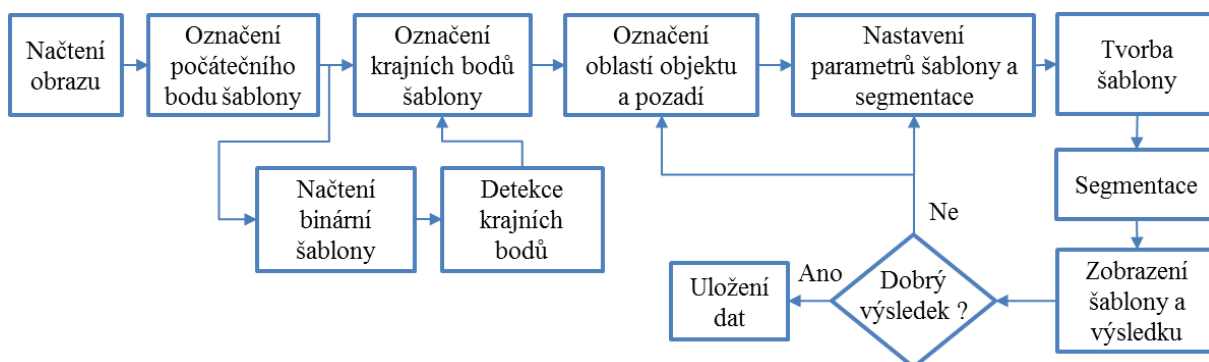
Obr. 4.1: Ukázka grafického uživatelského prostředí programu u segmentace cerebrálního abscesu na snímku z MRI.

Nejprve je načten výchozí obraz pro segmentaci, který je zobrazen v levém horním rohu okna programu. Poté jsou uživatelem zadány počáteční bod a krajní body pro tvorbu šablony, které je možné vyznačit ručně v načteném obraze, nebo lze použít binární obraz jako vzor pro šablonu. Ze zvoleného vzoru jsou krajní body získány detekcí hran nebo rohů dle výběru. Při detekci hran je volen krok mezi získanými body, u detekce rohů jejich maximální počet. Detekce hran je vhodnější u objektů oblého tvaru, u kterých se rohy vyskytují v minimálním počtu či vůbec. U rohových objektů je zase více žádoucí volit detekci rohů, neboť je pak vyloučena možnost, že by krajní body v rohových místech nebyly kvůli zvolenému kroku mezi nimi. Nalezené krajní body jsou následně posunuty podle zadaného počátečního bodu v obraze, od kterého jsou vzdáleny o zvolený počet pixelů. U šablony se dále definuje její přesah přes zadané krajní body, krok mezi jejími paprsky a body a parametr  $\Delta_r$ , popsáný výše.



Dále je třeba ručně vyznačit jednu či více oblastí objektu a pozadí v obraze buď formou jednotlivých bodů, nebo čtverce s definovanou velikostí okolí kolem zadaného bodu. Všechny označené body v původním načteném obraze lze vidět na Obr. 4.1 vlevo nahoře, počáteční bod jako růžový bod, krajní body žlutou barvou, zadaná oblast náležící objektu jako modrý čtverec a dvě oblasti pozadí jako čtverce zelené. Na tomto obrázku je též dobře patrný důvod, proč lze značit více oblastí objektu či pozadí, neboť má pozadí oproti objektu vyšší i nižší úroveň jasu.

Posledním zadávaným parametrem je koeficient  $\lambda$ , který souvisí se segmentační energií. V případě, že je zvolena hodnota vyšší, segmentace je provedena více dle intenzity pixelů než jejich vzájemné sousednosti a naopak. Po segmentaci jsou zobrazeny body šablony, které lze pozorovat na Obr. 4.1 vlevo dole a výsledky segmentace, zobrazené na Obr. 4.1 vpravo formou binárního obrázku a výsledného obrysu objektu v původním obraze pro lepší zhodnocení úspěšnosti segmentace. Při neuspokojivém výsledku lze přidat další oblasti objektu či pozadí, měnit parametry šablony i segmentace a proces opakovat. Poté lze všechny výsledky, parametry, body a značení uložit pro další účely. Schéma segmentačního programu je znázorněno na Obr. 4.2.



Obr. 4.2: Schéma realizovaného programu pro segmentaci.

Pro snazší orientaci uživatele v programu byla přidána „tooltip“ nápověda, která zobrazí jednoduchý popisek vedle kurzoru při jeho najetí na konkrétní nápis. Implementována byla i kontrola zadávaných vstupních parametrů, která uživatele upozorní na případné nedostatky a neumožní spuštění segmentace, dokud nejsou všechny údaje zadány správně.

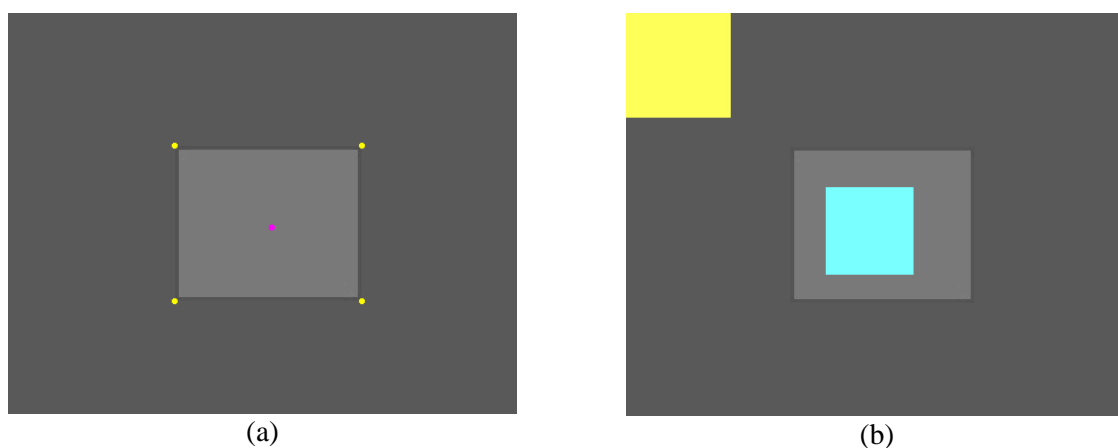
## 5 Výsledky

### 5.1 Testování na simulovaných datech

Nejprve byl segmentační algoritmus testován na simulovaných datech. Byl vytvořen obrázek obdélníku, pětiúhelníku a hvězdy s bitovou hloubkou 8 bitů a průměrnou hodnotou jasu pozadí 91 a objektu 122. Pro názornou ukázkou vyšší efektivity segmentace modifikované grafové metody byly porovnávány výsledky segmentace s využitím grafových řezů bez a s použitím vytvořené šablony. U původní grafové metody jsou v grafu obrazu spojeny všechny uzly n-hranami dle systému 4-okolí. Při segmentování se šablonou n-hrany spojují jen uzly dle grafu vytvořeného podle zadaných bodů a parametrů, uvedených dále. Testování bylo provedeno na snímcích s různou úrovní Gaussovského šumu, definovanou hodnotou jeho směrodatné odchylky.

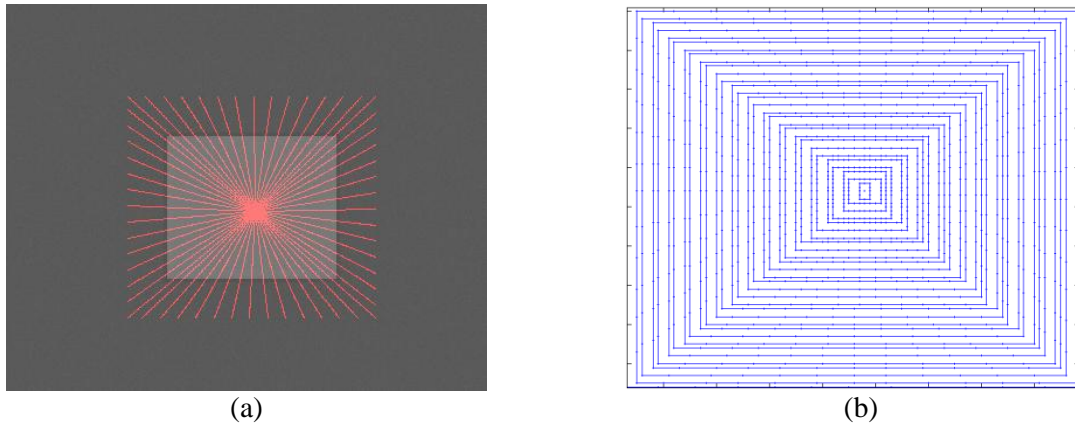
#### 5.1.1 Obdélník

Pro tvorbu šablony byl zadán počáteční bod a krajní body objektu dle Obr. 5.1 (a). Body označené jako pevná omezení pro výpočet histogramů potřebných k segmentaci (se šablonou i bez ní) jsou zobrazeny na Obr. 5.1 (b).



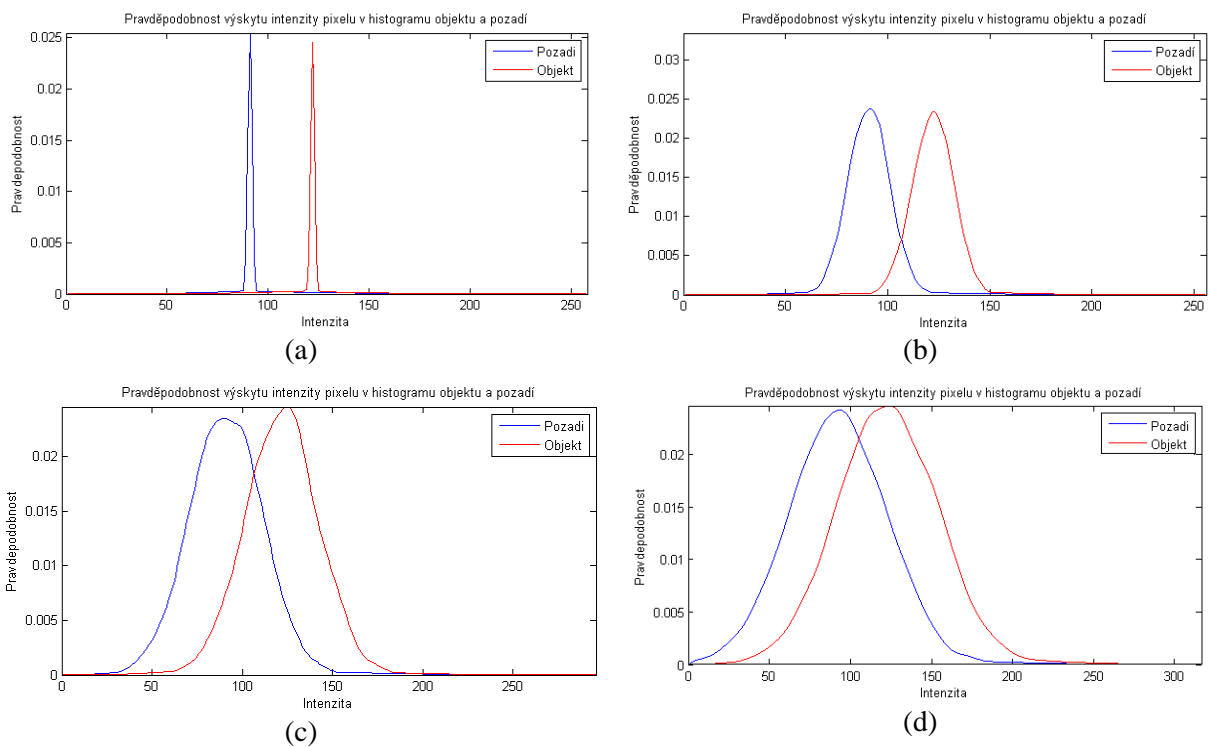
Obr. 5.1: Vstupní obraz obdélníku se zadanými body. (a) Počáteční bod šablony růžově a krajní body žlutě. (b) Označené body objektu modře a pozadí žlutě.

Uzly grafu implementované čtvercové šablony jsou pak zobrazeny na Obr. 5.2 (a). Vzhledem k tomu, že má objekt v uměle vytvořeném obrazu rovné hrany, byl u n-hran šablony nastaven parametr  $\Delta_r = 0$ . Bylo zjištěno, že s n-hranami tvořenými z-křivkami podél paprsku a r-křivkami mezi paprsky nejsou výsledky segmentací optimální. Následně byly pro lepší výsledky z-křivky ze všech grafů šablon odstraněny. Výsledné spojení uzlů n-hranami mezi paprsky středové části šablony je pak vykresleno na Obr. 5.2 (b).



Obr. 5.2: Graf implementované šablony čtverce. (a) Uzly šablony. (b) N-hrany středové části šablony.

Z histogramů pixelů značených v rámci pevných omezení se určuje pravděpodobnost výskytu jasu pixelu v histogramu objektu a pozadí, která je zobrazena v grafech na Obr. 5.3. Od úrovně šumu se směrodatnou odchylkou 10 již dochází k překrývání křivek objektu a pozadí, viz Obr. 5.3 (b).

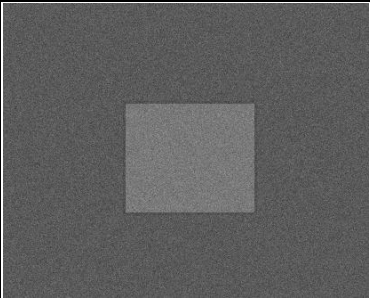
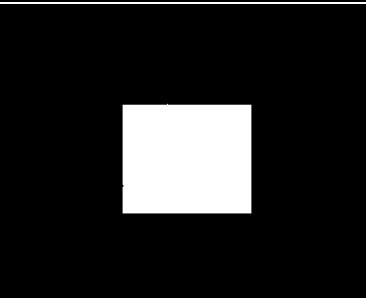
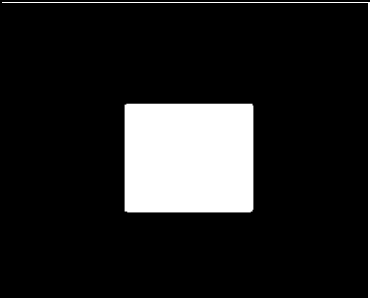
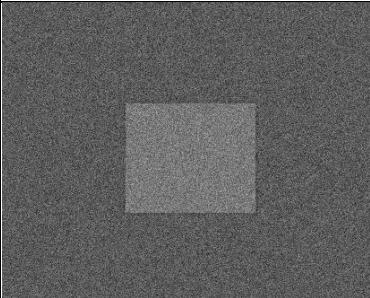
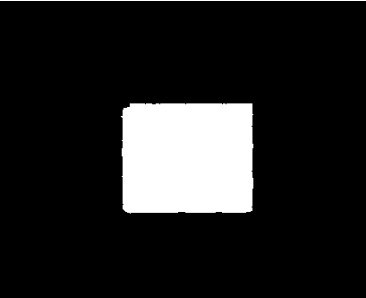
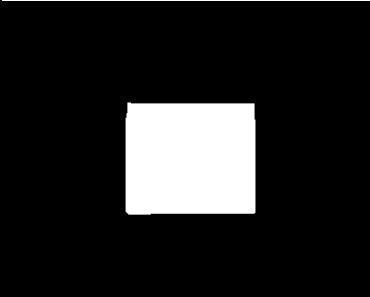
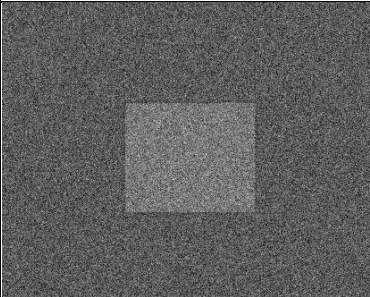
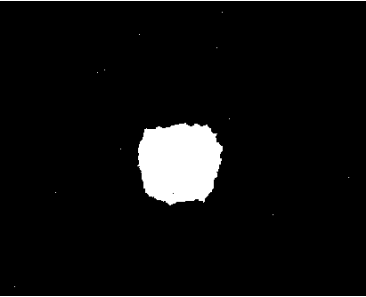
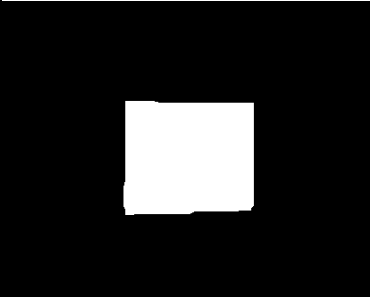


Obr. 5.3: Pravděpodobnosti výskytu jasu pixelu obrazu v histogramu objektu (červeně) a pozadí (modře) pro různé úrovně šumu. (a) 1. (b) 10. (c) 20. (d) 30.

U hodnoty směrodatné odchylky šumu rovné 1 až 10 se výsledky prakticky vůbec nelišily. Plně úspěšná byla tedy i segmentace bez využití šablony. Od úrovně šumu vyšší než 10 již ale byly výsledky metod odlišné. Následující Tabulka 2 proto uvádí vždy původní obraz a výsledek segmentace bez a se šablonou pro úrovně šumu 10 až 30. Hodnota parametru  $\lambda = 0,01$  byla pro segmentaci volena s ohledem na přidávaný šum do obrazu a souvislý tvar

objektu. Pro srovnání výsledků se hodnota  $\lambda$  neměnila pro všechny segmentace v následující tabulce.

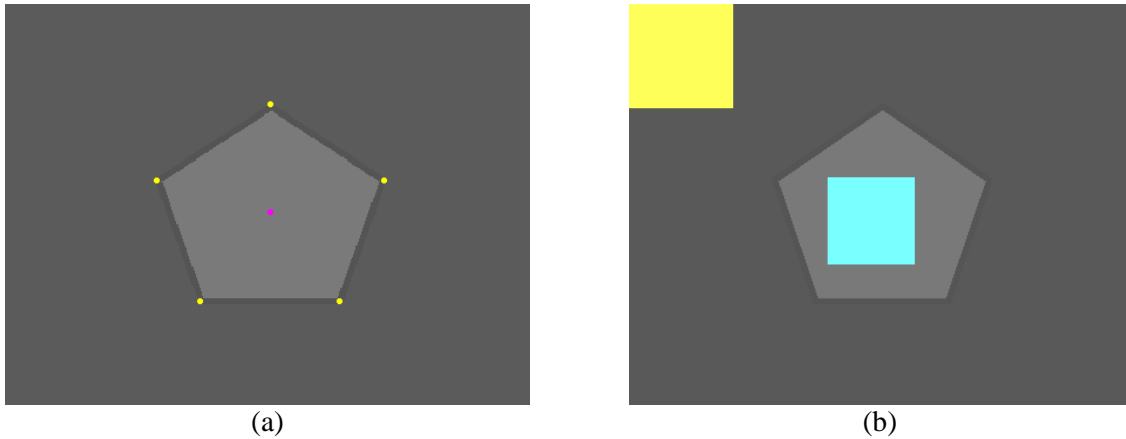
Tabulka 2: Porovnání výsledků „Graph-Cut“ segmentace obdélníku bez a se šablonou pro směrodatnou odchylku šumu 10-30

Šum	Původní obraz	„Graph-Cut“ bez šablony	„Graph-Cut“ se šablonou
10	 (a)	 (b)	 (c)
20	 (d)	 (e)	 (f)
30	 (g)	 (h)	 (ch)

U obrazů s takovou úrovní šumu, že dochází k výraznému překrytí histogramů objektu a pozadí, již tedy „Graph-Cut“ segmentace samotná selhává. Přínosem je pak využití šablony, neboť má díky spojení jen jejích uzlů dle zadaných parametrů přesnější výsledky, jak ukazuje Tabulka 2.

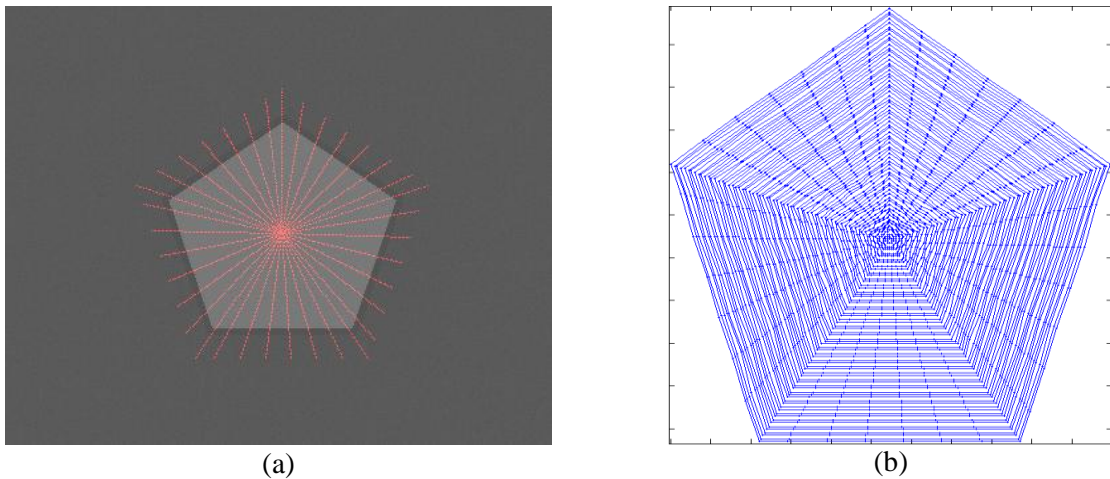
### 5.1.2 Pětiúhelník

U obrazu s pětiúhelníkem bylo pro tvorbu šablony zadáno s počátečním bodem dalších 5 krajních bodů objektu, jak je uvedeno na Obr. 5.4 (a). Dále byly označeny body objektu a pozadí jako pevná omezení, které jsou vyznačeny na Obr. 5.4 (b).



Obr. 5.4: Vstupní obraz pětiúhelníku se zadanými body. (a) Počáteční bod šablony růžově a krajní body žlutě. (b) Označené body objektu modře a pozadí žlutě.

Dle zadaných bodů byl sestaven graf šablony, jehož uzly jsou zobrazeny na Obr. 5.5 (a). Objekt má v uměle vytvořeném obrazu rovné hrany, proto byl volen opět parametr  $\Delta_r = 0$ . Spojení uzlů n-hranami použité pro segmentaci je pak vykresleno na Obr. 5.5 (b).

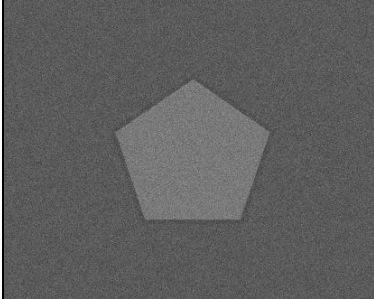
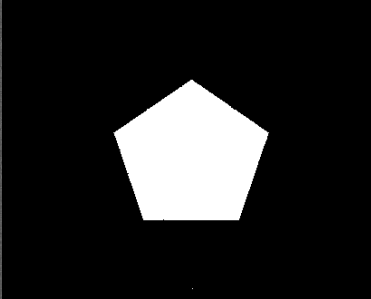
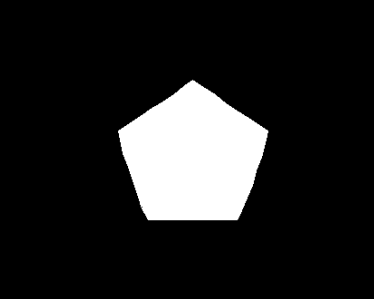
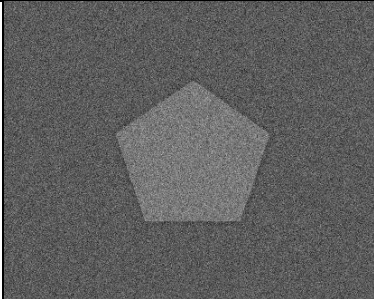
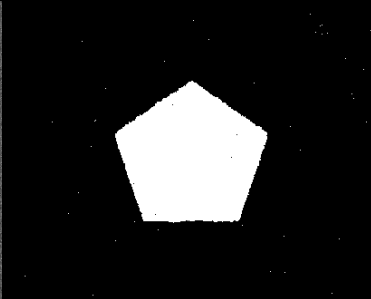
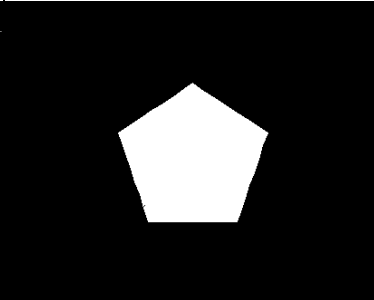
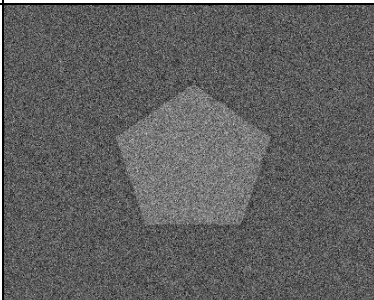
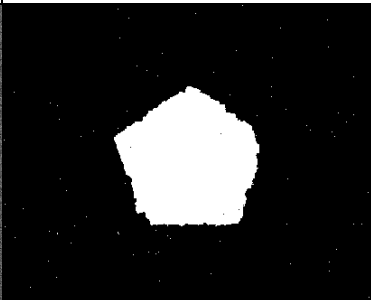
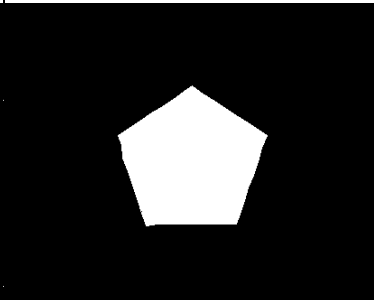


Obr. 5.5: Graf implementované šablony pětiúhelníku. (a) Uzly šablony. (b) N-hrany šablony.

Testovány byly opět obrázky se šumem v rozmezí hodnot jeho směrodatné odchylky 10 až 30. Tabulka 3 zobrazuje původní obraz a výsledek „Graph-Cut“ segmentace bez a s použitím šablony. Hodnota parametru  $\lambda$  byla nastavena na hodnotu 0,03, která je nízká opět kvůli přidávanému šumu, ale vyšší než u obdélníku kvůli tvaru objektu.

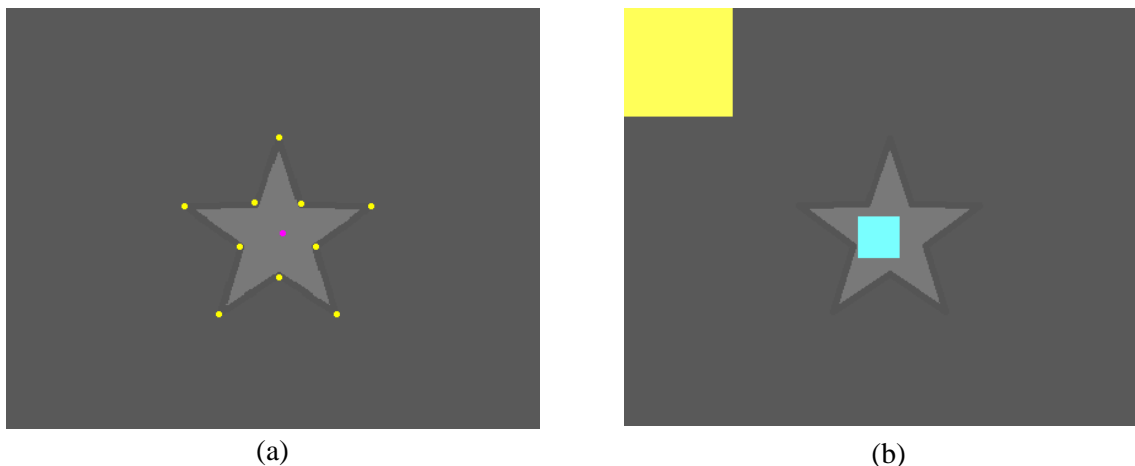
U směrodatné odchylky šumu 10 se zdá být přesnější segmentace bez šablony, což je patrně způsobeno nepřesným zadáním krajních hodnot pro tvorbu šablony jen odhadem. Také čím více paprsků tvoří šablonu, tím přesnější výsledek segmentace jsme schopni získat. U obrazu se šumem 20 je již zřetelná výhoda použití šablony, neboť nejsou špatně segmentovány body pozadí kvůli šumu, jak tomu nastalo u segmentace bez šablony, viz Tabulka 3 (e).

Tabulka 3: Porovnání výsledků „Graph-Cut“ segmentace pětiúhelníku bez a se šablonou pro směrodatnou odchylku šumu 10-30

Šum	Původní obraz	„Graph-Cut“ bez šablony	„Graph-Cut“ se šablonou
10	 (a)	 (b)	 (c)
20	 (d)	 (e)	 (f)
30	 (g)	 (h)	 (ch)

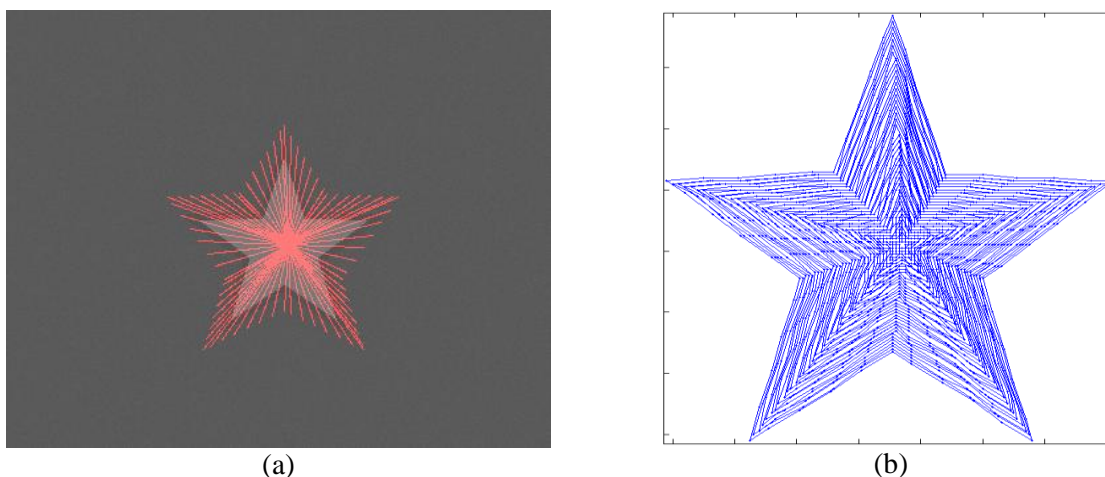
### 5.1.3 Hvězda

Hvězda je pro segmentaci tvar již poměrně složitý, pro tvorbu šablony hvězdy muselo být zadáno 10 souřadnic vrcholů objektu, znázorněných na Obr. 5.6 (a). Zadané body pevných omezení segmentace jsou na Obr. 5.6 (b).



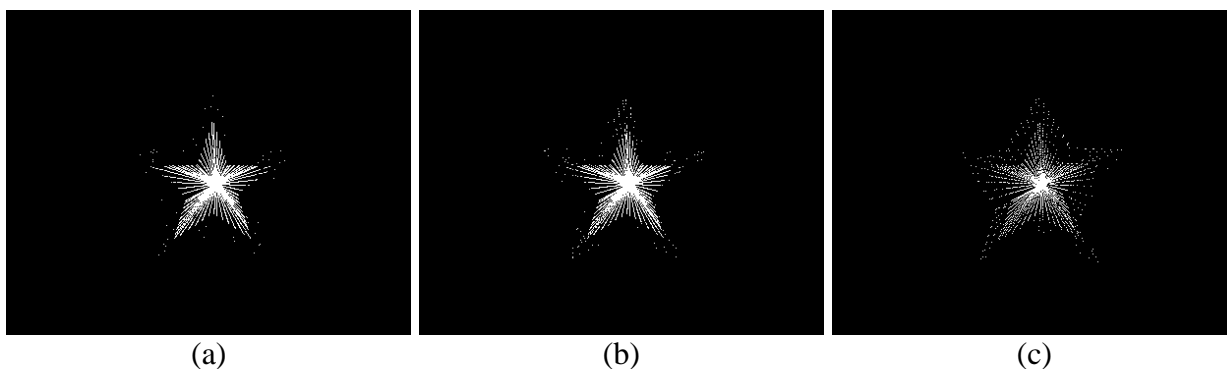
Obr. 5.6: Vstupní obraz hvězdy se zadanými body. (a) Počáteční bod šablony růžově a krajní body žlutě. (b) Označené body objektu modře a pozadí žlutě.

Uzly grafu vytvořené šablony hvězdy lze vidět na Obr. 5.7. Parametr  $\Delta_r$  byl opět nastaven na nulovou hodnotu a vzniklé n-hrany šablony hvězdy jsou znázorněny na Obr. 5.7 (b). Vzhledem k tomu, že má hvězda poměrně členitý tvar, byla zvolena vyšší hodnota parametru  $\lambda = 1$ , výsledek segmentace je proto náchylnější k šumu.



Obr. 5.7: Graf implementované šablony hvězdy. (a) Uzly šablony. (b) N-hrany šablony.

Na Obr. 5.8 lze vidět, že byly vysegmentovány i oddělené body šablony, které zřejmě k objektu nepatří. Na základě znalosti tvaru segmentovaného objektu lze tyto body odstranit při využití šablony pomocí kontroly, zda jsou segmentovány i body sousedních paprsků, se kterými má být tento uzel spojen. Také je využito ověření, zda mezera mezi daným bodem a zbylými segmentovanými uzly není mezera větší než stanovená mez (např. o číslo větší než zvolený krok mezi uzly paprsku). Problémem by mohl být otvor v segmentovaném objektu, který ale při získání výstupu u modifikované metody není brán v úvahu, jak je popsáno v kapitole 2.4. Tabulka 4, s rozdíly „Graph-Cut“ segmentací bez a s použitím šablony, pak obsahuje výsledky segmentace s využitím těchto kontrolních podmínek. Rozdíl segmentací je patrný již od úrovně šumu 10 právě kvůli složitějšímu tvaru hvězdy a nutnosti vyšší hodnoty  $\lambda$ .



(a)

(b)

(c)

Obr. 5.8: Vysegmentované body šablony hvězdy u obrazů s různou úrovní šumu. (a) 10. (b) 20. (c) 30.

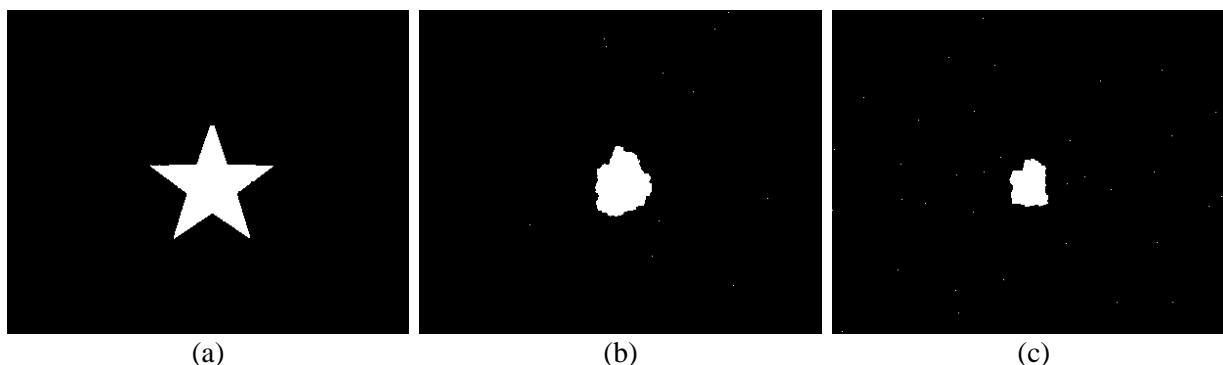
Tabulka 4: Porovnání výsledků „Graph-Cut“ segmentace hvězdy bez a se šablonou pro směrodatnou odchylku šumu 10-30

Šum	Původní obraz	„Graph-Cut“ bez šablony	„Graph-Cut“ se šablonou
10			
	(a)	(b)	(c)
20			
	(d)	(e)	(f)
30			
	(g)	(h)	(ch)

U segmentace bez využití šablony je výsledek sice lepší při nízké hodnotě parametru  $\lambda = 0,01$ , ale při vyšších hodnotách šumu se již ztrácí tvar hvězdy ve výstupu segmentace, jak lze pozorovat na Obr. 5.9. Na složitější tvary objektů k segmentaci v obrazech se šumem



či velmi podobnými hodnotami jasů objektu a pozadí tedy již prostý „Graph-Cut“ bez využití šablony nestačí. Zato s jejím využitím lze segmentaci získat lepší výsledky.



Obr. 5.9: Výsledky „Graph-Cut“ segmentace bez šablony s  $\lambda = 0,01$  u obrazů s různou úrovní šumu. (a) 10. (b) 20. (c) 30.

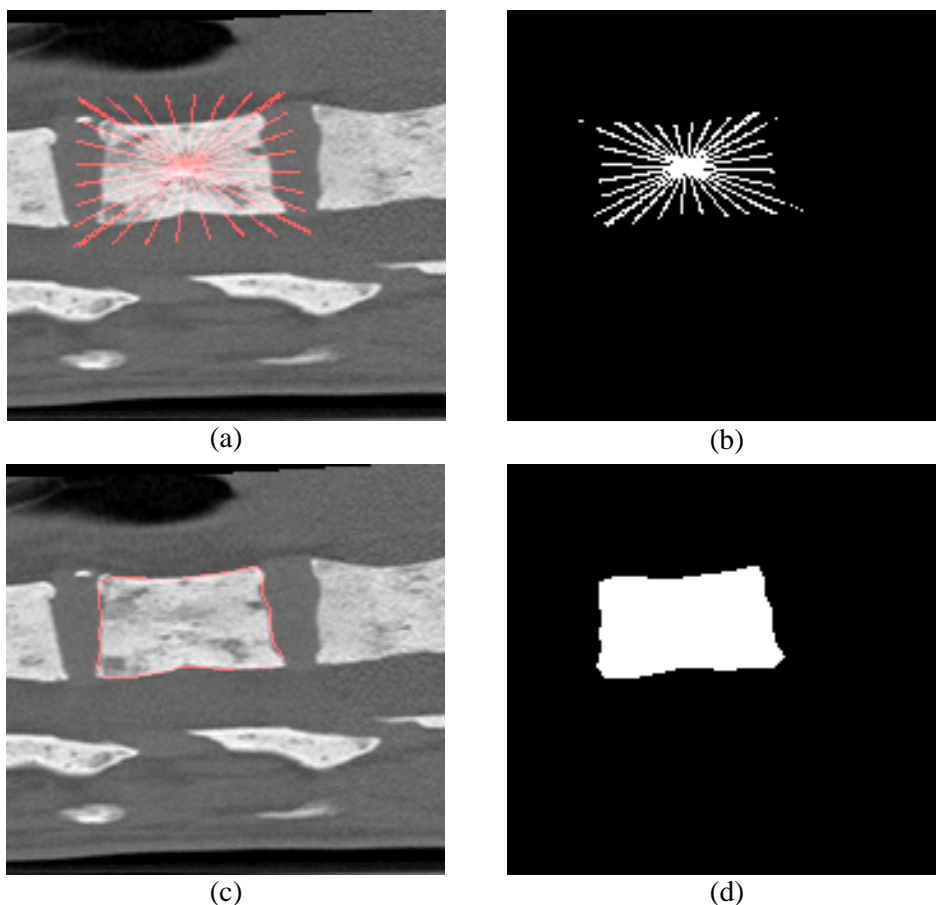
## 5.2 Testování na reálných datech

Realizovaná segmentační metoda byla dále testována na reálných CT a MRI snímcích obratlů, jejichž výsledky jsou uvedeny v následující podkapitole. Jedná se o reálná patientská data poskytnutá Vědeckým onkologickým institutem IRST v rámci spolupráce s Ústavem biomedicínského inženýrství na výzkumu. Dále byl algoritmus vyzkoušen na snímcích mozku s různými abnormalitami, o čemž pojednává podkapitola 5.2.2.

### 5.2.1 Segmentace obratlů

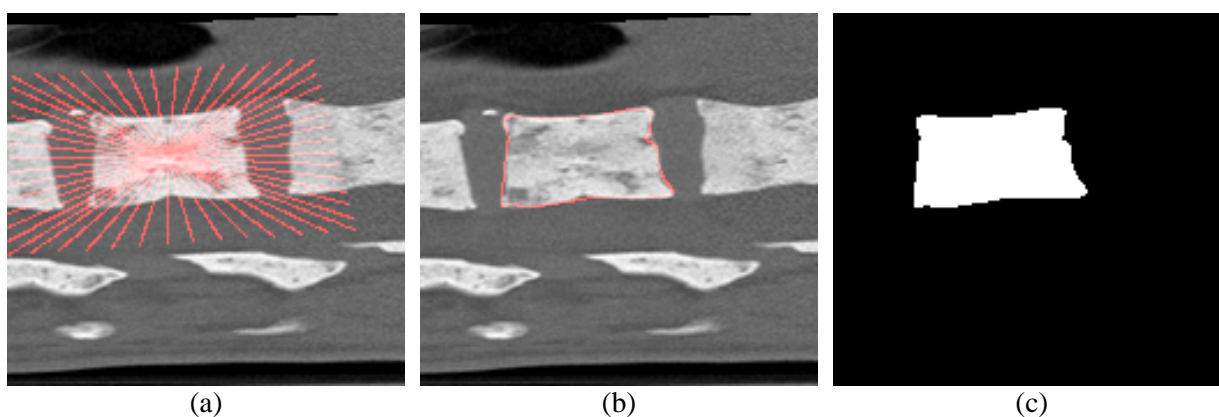
Segmentační algoritmus se čtvercovou šablonou byl použit na reálné snímky obratlů v sagitálním pohledu. Po zadání výchozího bodu a čtyřech rohů objektu byla vytvořena šablona pro segmentaci, která je zobrazena na vstupním obrazu na Obr. 5.10 (a). Vzhledem k tomu, že segmentovaný objekt není přesně tvaru čtverce, byla pro tvorbu hran mezi paprsky šablony zvolena hodnota parametru  $\Delta_r = 3$ . Dále se pixely objektu úrovní jasů výrazně liší od pozadí a v obraze není zřejmý výrazný šum, proto lze nastavit vyšší hodnotu  $\lambda = 3$ .

Na Obr. 5.10 (b) jsou zobrazeny všechny vysegmentované body šablony, přičemž některé zřejmě k objektu nepatří. Na základě znalosti tvaru segmentovaného objektu byly tyto body odstraněny pomocí podmínek obdobných jako u segmentace hvězdy (viz kapitola 5.1.3). Na Obr. 5.10 (c) je pak vyobrazen výsledný obrys segmentovaného obratle, vzniklý spojením vysegmentovaných krajních bodů šablony. V Obr. 5.10 (d) lze pozorovat výsledek segmentace, kde jsou všechny pixely přiřazené objektu bílé a pozadí černě.



Obr. 5.10: Segmentace obratle (real\_data\_11) v sagitálním pohledu. (a) Body vytvořené šablony. (b) Vysegmentované body šablony. (c) Obrys objektu. (d) Všechny segmentované body objektu.

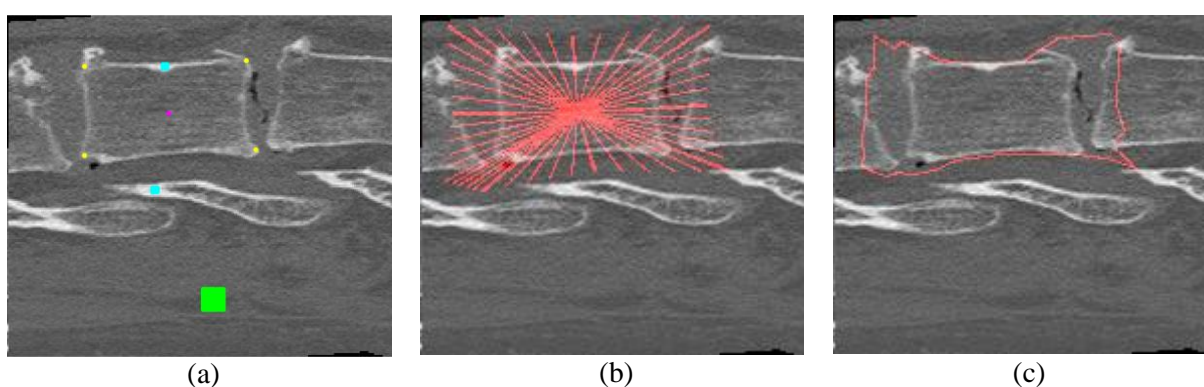
Vliv na podobu výsledku segmentace má i zvolený přesah šablony přes označené krajní body objektu. U obratle z předešlého snímku, u kterého se body uvnitř objektu úrovní jasu výrazně liší od pozadí, větší přesah uzlů šablony přes jeho krajní body nečiní pro segmentaci viditelný problém. Výsledek se šablonou s paprsky až k okraji snímku je zobrazen na Obr. 5.11 a odpovídá tvarem objektu.



Obr. 5.11: Segmentace obratle (real\_data\_11) s větším přesahem šablony. (a) Body šablony. (b) Výsledný obrys. (c) Výsledek.

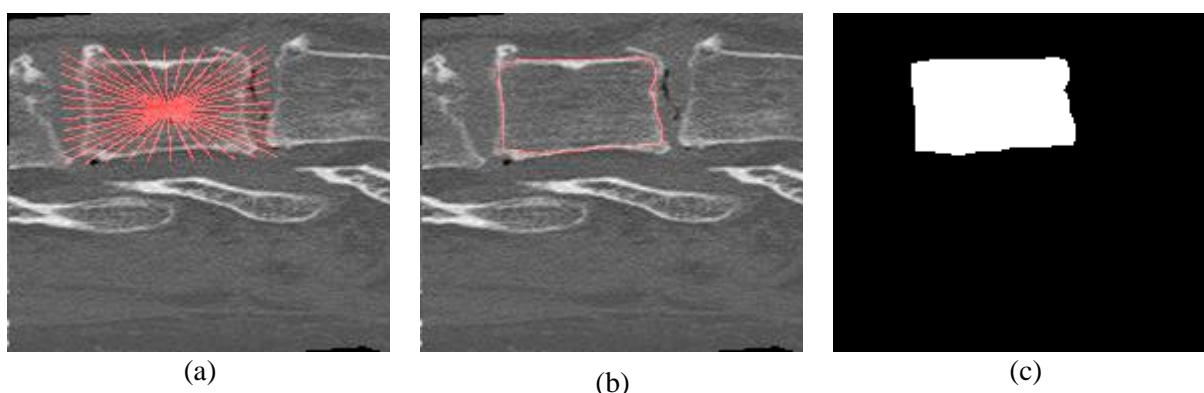
Jak lze vidět na Obr. 5.11 (a), použitá šablona přesahuje přes okolní obratle, které mají podobnou úroveň jasu jako obratel segmentovaný. Z toho důvodu byla zvolena nižší hodnota parametru  $\lambda = 0,5$ .

Problém při velkém přesahu šablony nastává u obratlů, jejichž pixely jsou uvnitř objektu úrovní jasu téměř či zcela shodné s pixely pozadí a liší se pouze body na jejich okraji. Pak je třeba pro segmentaci značit oblasti objektu na těchto krajních bodech odlišných od pozadí, jak lze vidět na Obr. 5.12 (a), kde je značení objektu modrou a pozadí zelenou barvou. Právě u takovýchto případech činí nesnáze vysoký přesah paprsků šablony, především pokud sahají přes další objekty, jako jsou například další obratle na snímku. Příklad segmentace, u které se z uvedených důvodů nezdařil získat uspokojivý výsledek, je znázorněn na Obr. 5.12.



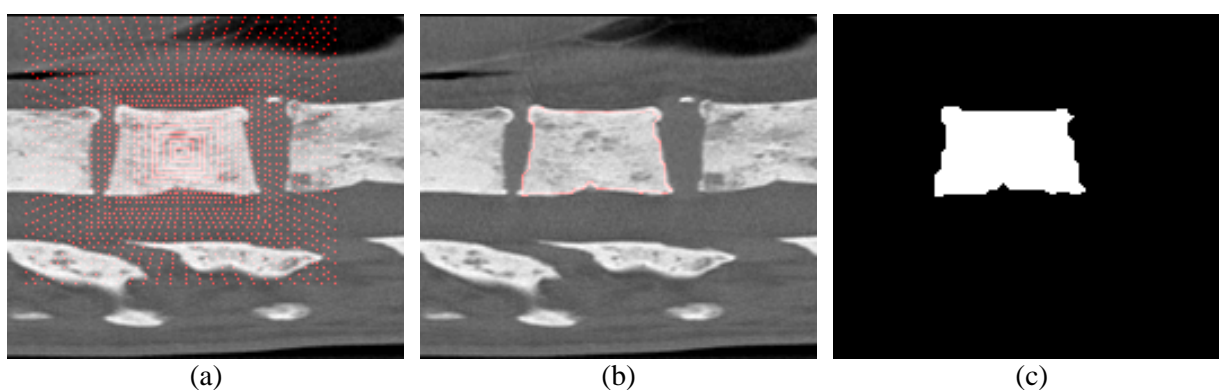
Obr. 5.12: Problematická segmentace obratle (real\_data\_8). (a) Značení. (b) Výsledný obrys. (c) Neuspokojivý výsledek.

Jelikož je výsledek segmentace uvedený na Obr. 5.12 prakticky nepoužitelný, lze v tomto případě využít menšího přesahu šablony, aby svými uzly zasahovala jen minimálně či vůbec do okolních obratlů. Výsledek segmentace stejného obratle, při použití šablony, která sahá přes sousední obratle jen nepárně, je zobrazen na Obr. 5.13. Jak lze vidět, získaný výsledek je již přijatelný vzhledem k obtížným podmínkám na snímku.



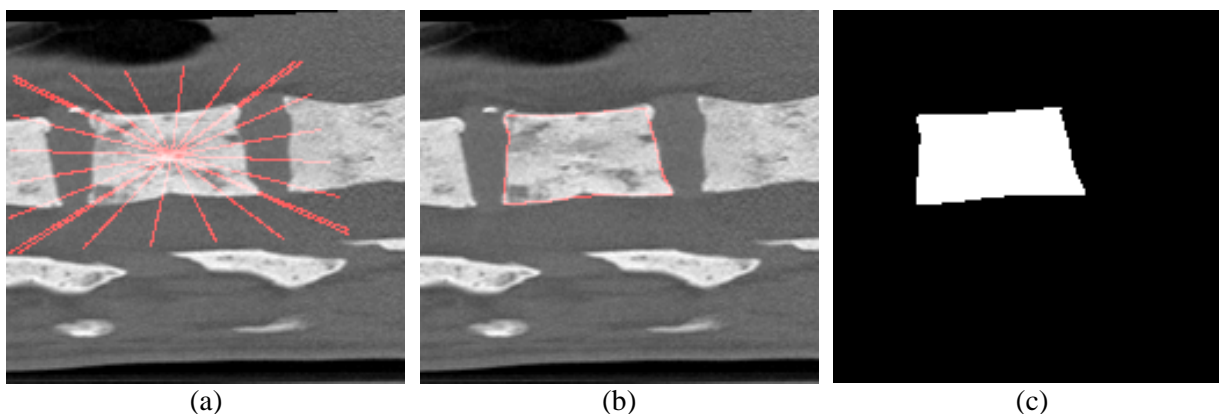
Obr. 5.13: Problematická segmentace obratle (real\_data\_8). (a) Body šablony. (b) Výsledný obrys. (c) Výsledek.

Z hlediska konstruované šablony mají vliv na výsledek segmentace také volené kroky mezi jejími paprsky, který odpovídá vzdálenosti mezi posledními body paprsků, tedy s největší vzdáleností od počátečního bodu, a krok mezi body podél paprsku. Tyto parametry musí být vhodně nastaveny tak, aby výsledný tvar, získaný segmentací, dostatečně odpovídal hledanému tvaru objektu. Při provedených segmentacích byl nejčastěji volen krok mezi body podél paprsků roven 1, aby bylo zajištěno, že body šablony „zachytí“ přesně obrys objektu. Větší krok mezi body totiž znamená řidší vzorkování utvářené sítě uzlů grafu, a tím vede ke snížení přesnosti, neboť se kontura objektu může nacházet právě v bodech, které jsou vynechány. Příklad negativního vlivu na přesnost segmentace lze vidět na Obr. 5.14 s krokem 4 mezi body. Výsledky pro různě velký krok mezi body paprsků pak znázorňuje v příloze A Tabulka 7.



Obr. 5.14: Segmentace obratle (real\_data\_12). (a) Šablona s krokem 4 mezi body paprsku. (b) Výsledný obrys. (c) Výsledek.

Mezi paprsky bývá volen krok větší, nejčastěji v rozmezí 10 až 20, především s ohledem na velikost snímku a segmentovaného objektu, případně dle komplikovanosti jeho tvaru. Ukázka nevhodně nastaveného kroku mezi paprsky na příliš vysokou hodnotu je uvedena na Obr. 5.15. Jak lze pozorovat, výsledek neodpovídá tvaru obratle v místech mezi paprsky šablony, pro větší přesnost je tedy třeba zvolit menší krok. Výsledky segmentace obratle s různým nastavením velikosti kroku mezi paprsky šablony uvádí Tabulka 6 v příloze A.

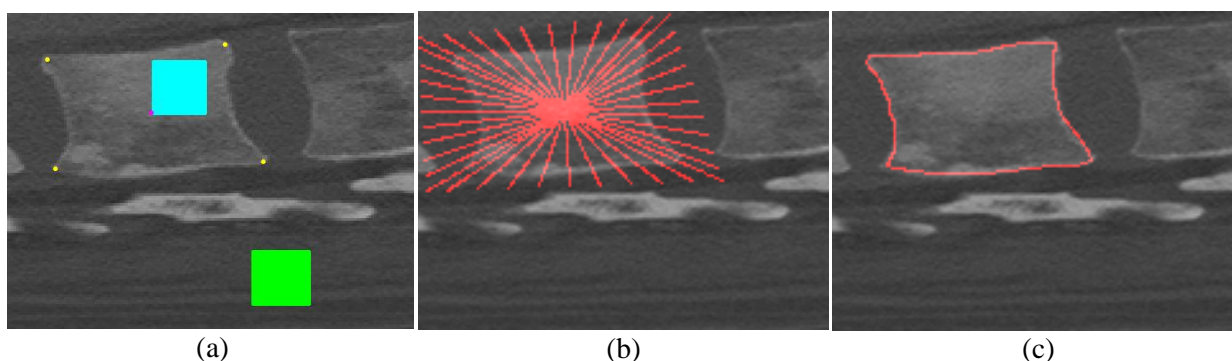


Obr. 5.15: Segmentace obratle (real\_data\_11). (a) Šablona s krokem 24 mezi paprsky. (b) Výsledný obrys. (c) Výsledek.

Je tedy patrné, že čím jsou kroky mezi body či paprsky menší, tím je získaný výsledek přesnější, neboť jsou aproximovány menší vzdálenosti mezi body při jejich spojování do uceleného tvaru. Při zvolení většího kroku bodů nebo paprsků ale proběhne segmentační proces rychleji, neboť je pro výpočet zahrnuto méně obrazových uzlů. Nastavení je tedy třeba provést s ohledem na parametry obrazu a požadavky na výsledek.

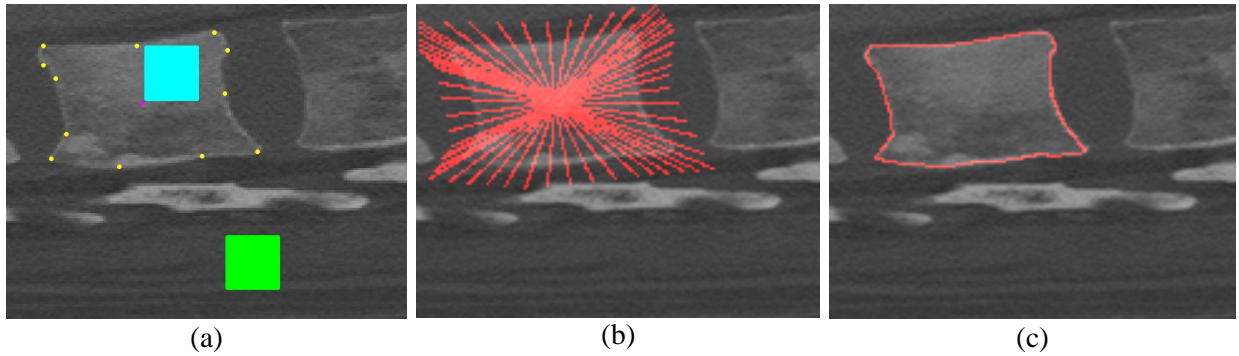
Dále byl sledován vliv na výsledek segmentace z hlediska zadaného počtu krajních bodů. Při zadání jejich většího počtu je tvar objektu přesněji vymezen a vytvořená šablona pak i více napodobuje jeho tvar. Následkem toho je možné nastavit nižší hodnotu parametru  $\Delta_r$ , který při použití čtvercové šablony musí být vyšší, aby byla možná větší variabilita segmentovaného tvaru vůči vytvořené šabloně. Navíc bylo pro usnadnění, a tím i urychlení procesu značení více souřadnic krajních bodů, vytvořeno grafické uživatelské rozhraní.

Pro srovnání byly u stejného snímku obratle zadány nejprve 4 krajní body, jak lze vidět na Obr. 5.16 (a) žlutou barvou. Utvořená šablona má potom přibližně čtvercový tvar, znázorněný na Obr. 5.16 (b). Pro segmentaci byl zvolen jako optimální parametr  $\Delta_r = 3$ . U výsledného obrysu, získaného touto segmentací, uvedeného na Obr. 5.16 (c), lze pozorovat odlišnost od tvaru obratle zejména v jeho rozích, ačkoli jeho segmentovaný tvar není příliš složitý.



Obr. 5.16: Segmentace obratle (real\_data\_5) s označením 4 krajních bodů. (a) Značení. (b) Utvořená šablona. (c) Výsledek segmentace.

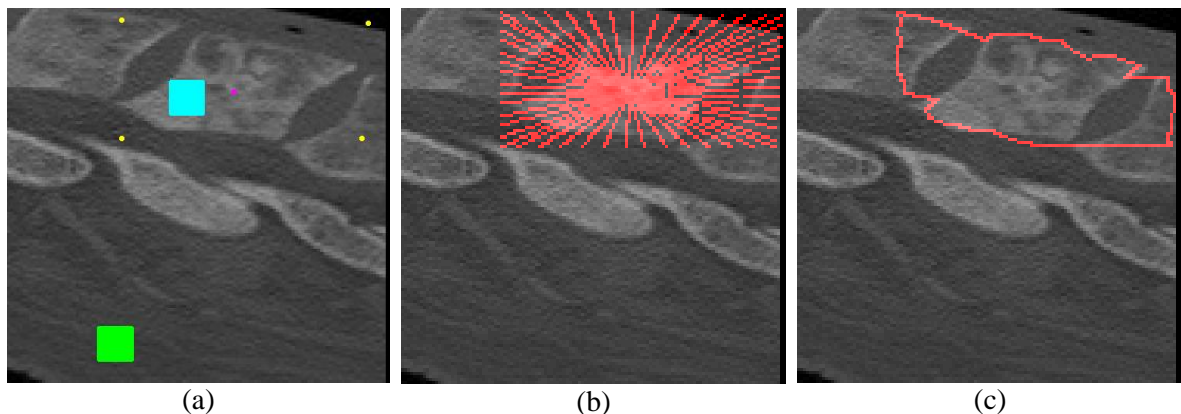
Dále bylo na totožném snímku obratle zadáno celkem 12 krajních bodů tak, aby vhodně popisovaly jeho tvar, znázorněny jsou žlutou barvou na Obr. 5.17 (a). Zkonstruovaná šablona pak obrysem více kopíruje tvar obratle, jak lze pozorovat na Obr. 5.17 (b), proto byl zvolen parametr  $\Delta_r = 0$ . Získán byl výsledek skutečně odpovídající tvaru segmentovaného objektu, zobrazen na Obr. 5.17 (c).



Obr. 5.17: Segmentace obratle (real\_data\_5) s označením 12 krajních bodů. (a) Značení. (b) Šablona. (c) Výsledek.

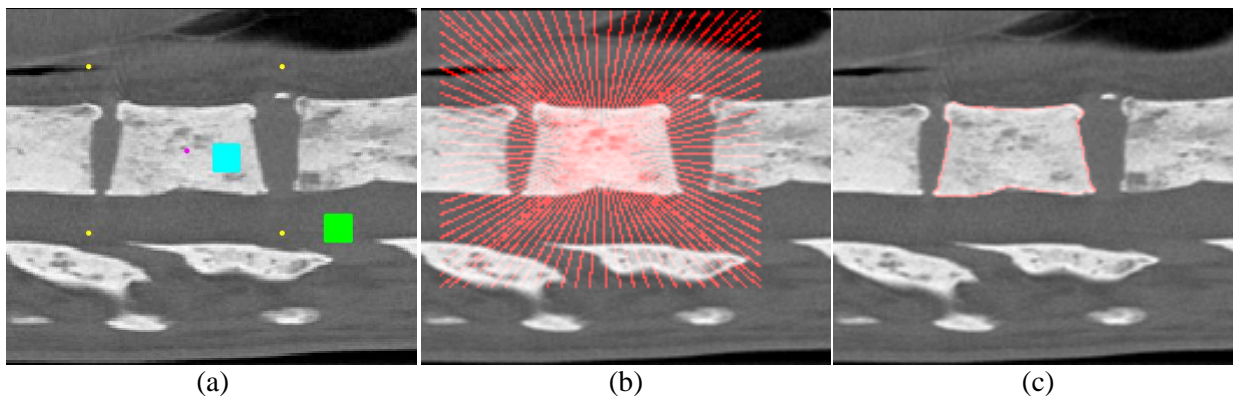
Dále tedy bylo na snímcích obratlů zadáváno více krajních bodů pro tvorbu šablony, aby její tvar více napodoboval vybraný obratel. Další ukázky výsledků segmentace obratlů vzhledem k zadanému počtu krajních bodů jsou uvedeny v příloze A, Tabulka 5.

Při použití binární předlohy pro tvorbu šablony je třeba dbát především na to, zda tvarem skutečně alespoň přibližně odpovídá segmentovanému objektu, a také zda je vzhledem k němu ve vhodném natočení. Například čtvercovou či obdélníkovou šablonu lze použít pro segmentaci obratle jen v případech, kdy na snímku není obratel nijak výrazně nakloněn či pootočen. Příklad nevhodné aplikace šablony dle binární předlohy obdélníku je ukázán na Obr. 5.18 u obratle, jehož tvar je vzhledem k použité šabloně značně odlišný. Z toho důvodu se nezdařilo získat uspokojivý výsledek, což potvrzuje klíčovost správné formy použité šablony pro segmentaci. Výsledek segmentace tohoto obratle při použití vhodné šablony bez binární předlohy lze vidět v příloze A, Tabulka 5 (g) až (k).



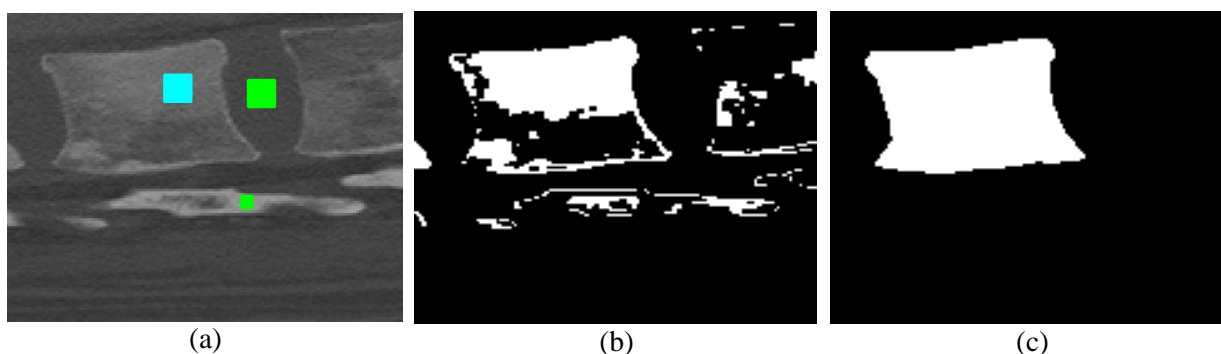
Obr. 5.18: Segmentace obratle (real\_data\_6) s nevhodným použitím obdélníkové binární předlohy pro šablonu. (a) Značení. (b) Obdélníková šablona. (c) Výsledek.

Příklad přijatelného použití binární předlohy čtverce pro konstrukci šablony je uveden na Obr. 5.19. Vzhledem k tomu, že se obrys obratle na daném CT snímku od čtvercového tvaru neodchyluje výrazně, byl zvolen parametr  $\Delta_r = 1$  a bylo dosaženo optimálního výsledku. Binární předlohu jednoduchého tvaru tedy pro segmentaci obratle v některých případech skutečně lze využít.



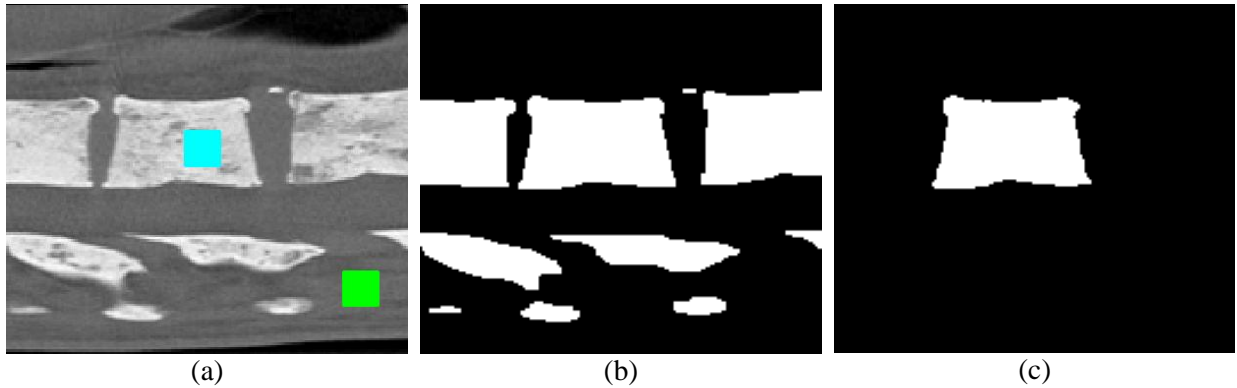
Obr. 5.19: Segmentace obratle (real\_data\_12) s vyhovujícím použitím čtvercové binární předlohy pro šablonu. (a) Značení. (b) Čtvercová šablona. (c) Výsledek.

Při patřičném zadání všech zmíněných bodů, označení a parametrů, případně po jejich úpravě, lze ve většině případů docílit přijatelného výsledku segmentace obratle. Pro srovnání byly snímky obratlů segmentovány i původní nemoifikovanou grafovou metodou, tedy bez využití šablony. Na Obr. 5.20 je zobrazeno porovnání segmentace u obratle, jehož část je úrovní jasu špatně rozlišitelná od pozadí. Lze vidět, že metoda modifikovaná, využívající šablonu, uvedenou na Obr. 5.17, je zde úspěšná a metoda původní selhává.



Obr. 5.20: Segmentace obratle (real\_data\_5). (a) Značení oblastí. (b) Výsledek původní metody. (c) Výsledek metodou s využitím šablony.

Na Obr. 5.21 je uvedeno srovnání metod u obratle, jenž se naopak svou úrovní jasu liší od pozadí výrazně. Metodou s využitím čtvercové šablony, zobrazené na Obr. 5.19, je vysegmentován pouze zvolený obratel. Původní metodou jsou vysegmentovány všechny objekty stejného jasu v obraze, jako vybraný obratel. Je tedy na uživateli, co požaduje. Výsledky dalších porovnání jsou uvedeny v příloze A, Tabulka 8.

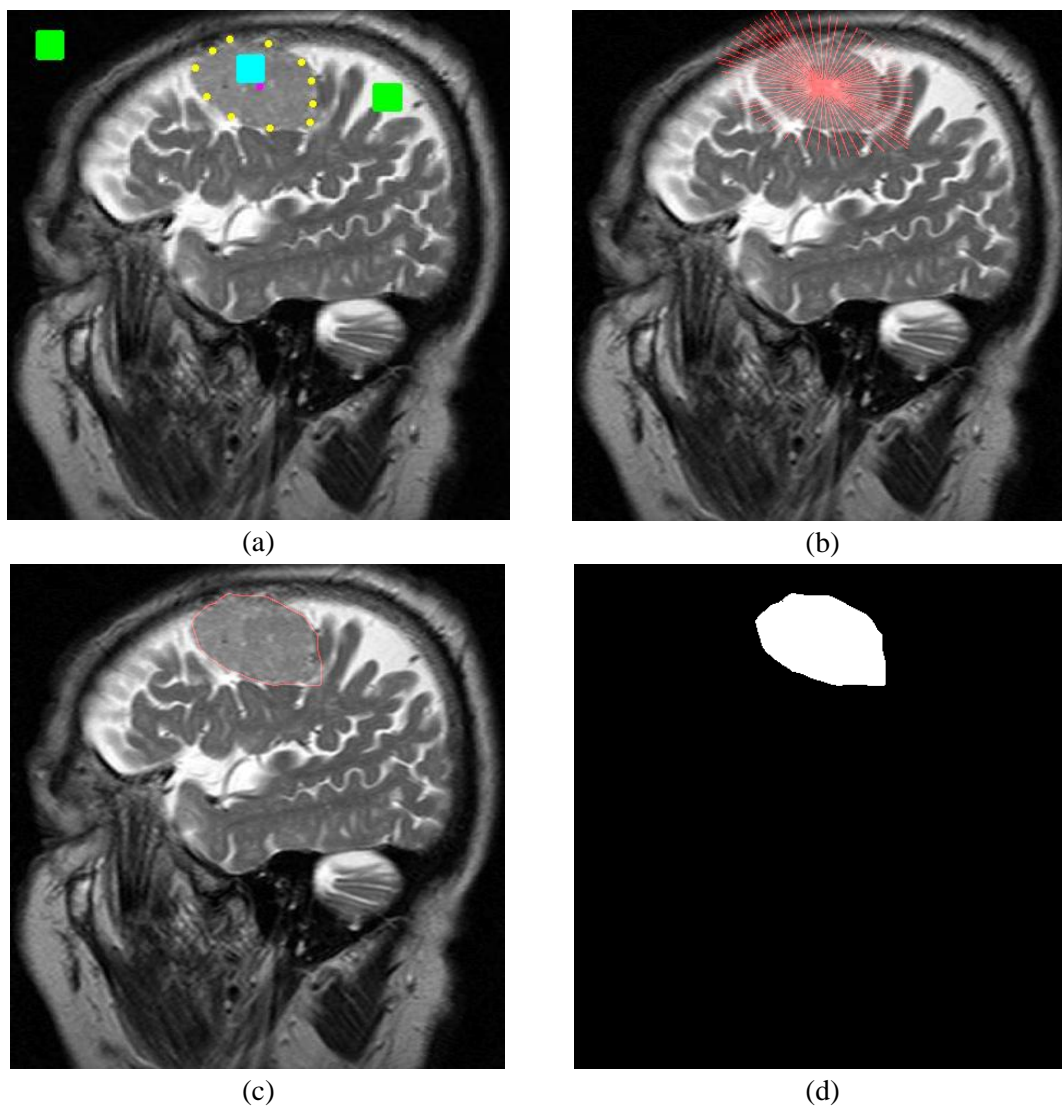


Obr. 5.21: Segmentace obratle (real\_data\_12). (a) Značení oblastí. (b) Výsledek původní metody. (c) Výsledek metodou s využitím šablony.

### 5.2.2 Segmentace snímků mozku

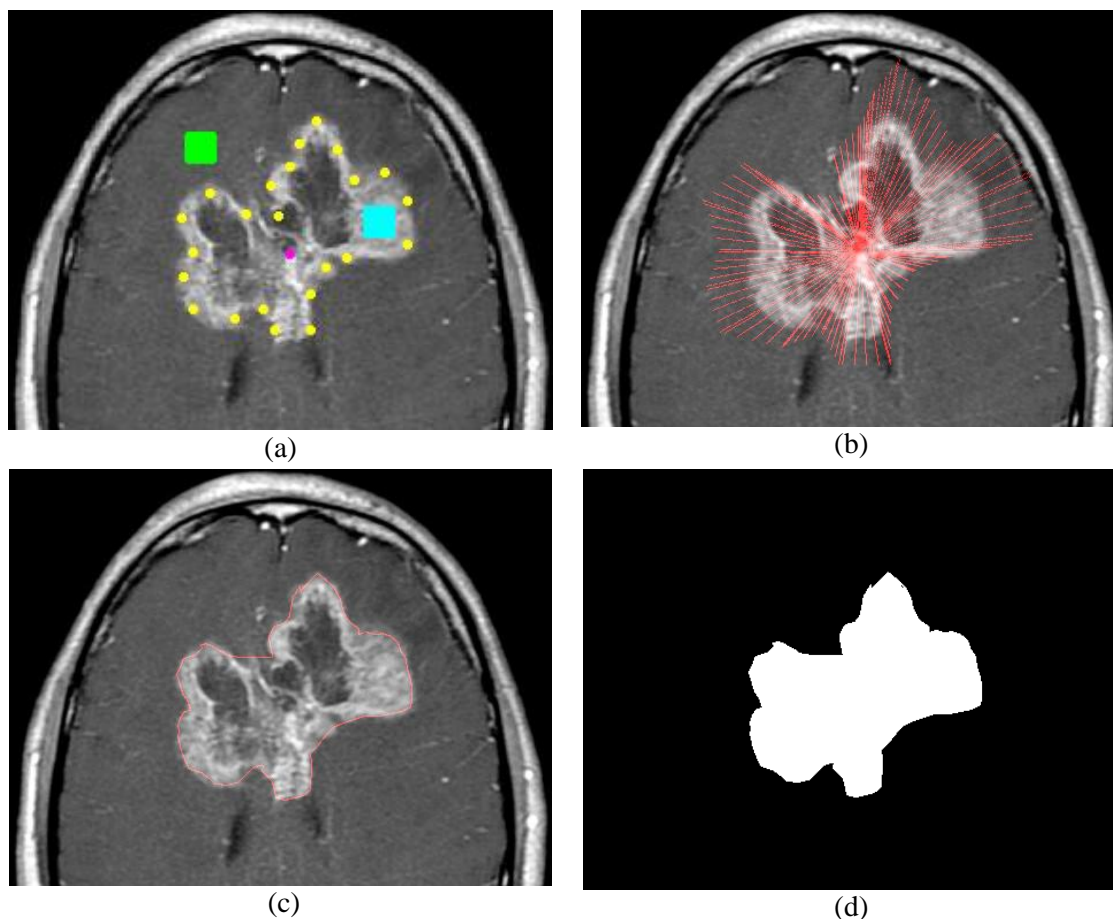
Realizovaná segmentační metoda byla dále testována na reálných CT a MRI snímcích mozku s různými abnormalitami, spojenými především s nádorovými onemocněními. Vzhledem k velké různorodosti segmentovaných útvarů v mozku byly pro tvorbu šablony krajní body zadávány ručně, bez využití binárních předloh. U značení oblastí objektu a pozadí se ukázala výhodná možnost označení více oblastí s různou úrovní jasu pro tvorbu histogramů. Využití značení vícero oblastí pozadí lze pozorovat na Obr. 5.22 se segmentací meningiomu, jehož část okraje je u lebky špatně rozlišitelná od pozadí. Jako oblasti pozadí byly označeny jak světlejší a tmavší oblast kůry mozkové, tak i tmavé okolí hlavy, na Obr. 5.22 (a) jsou zelenou barvou. Dále jsou na Obr. 5.22 (b) zobrazeny body použité šablony, která svými paprsky sahá až k okraji obrazu, a tedy zahrnuje uzly téměř všech úrovní jasu v obrazu. Na Obr. 5.22 (c) lze pozorovat dosažený výsledek, který téměř odpovídá hledanému obrysu.





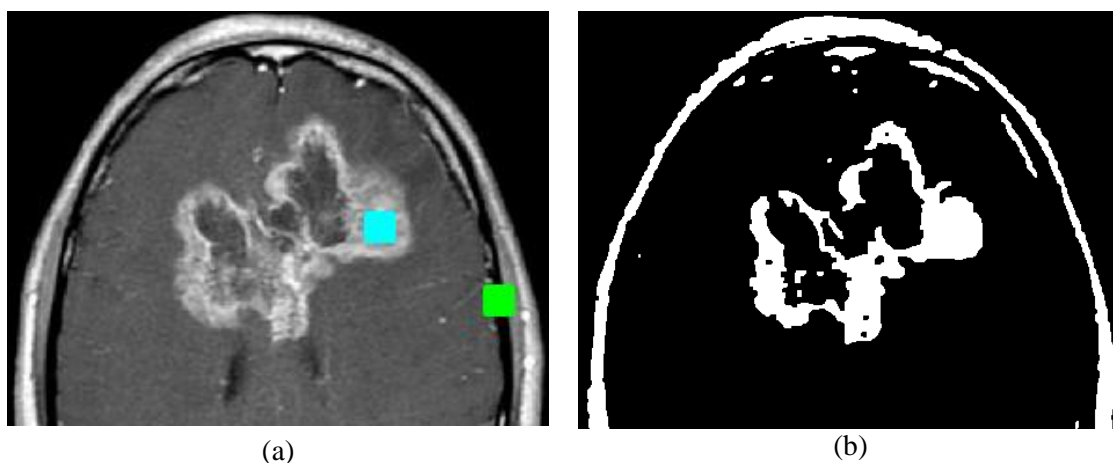
Obr. 5.22: Segmentace meningiomu na MRI (T2) snímku hlavy v sagitálním pohledu. (a) Značení bodů a oblastí. (b) Utvořená šablona. (c) Výsledný obrys. (d) Výsledek. [12]

Na následující ukázce segmentace glioblastomu na Obr. 5.23 lze vidět, že si realizovaný segmentační algoritmus dokáže poradit i s více komplikovaným tvarem objektu. Klíčové je zadání dostatečného množství krajních bodů tak, aby dostatečně vystihly útvar pro segmentaci a samozřejmě vhodně vyznačit oblasti objektu a pozadí, jak je ukázáno na Obr. 5.23 (a). Krok mezi paprsky je třeba zvolit tak, aby nebyla vynechána žádná změna ve tvaru obrysu objektu, jak lze vidět u vykreslené šablony na Obr. 5.23 (b). Paprsky konstruované šablony také sahají jen k vnitřnímu okraji lebky, neboť při jejím přesahu již nastává problém obdobný, jako byl popsán u obratle na Obr. 5.12, a segmentovány jsou i pixely lebky. I přes složitost segmentovaného útvaru byl získán poměrně přesný výsledek segmentace, který je uveden na Obr. 5.23 (c) a (d). Na získaných výsledcích lze také pozorovat, že u objektu nejsou segmentovány žádné otvory uvnitř jeho struktury, neboť touto metodou nejsou při zpracování brány v úvahu, jak bylo vysvětleno v kapitole 2.4.



Obr. 5.23: Segmentace glioblastomu „Butterfly glioma“ na snímku hlavy v axiální rovině. (a) Značení bodů a oblastí. (b) Utvořená šablona. (c) Výsledný obrys. (d) Výsledek. [13]

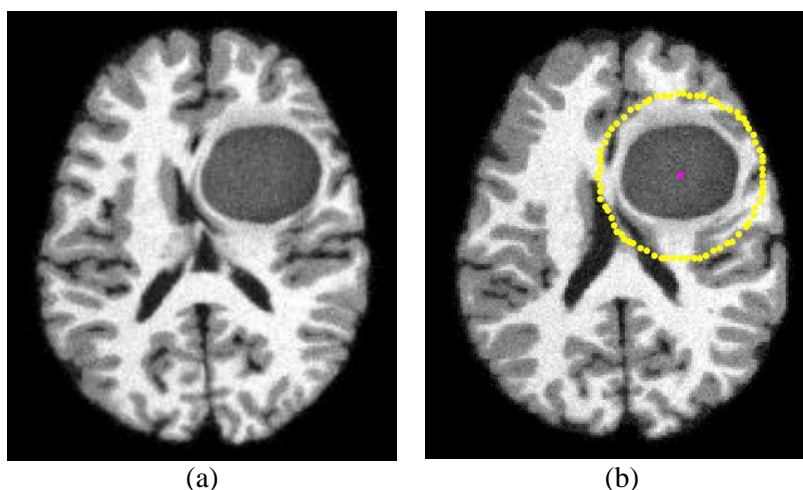
Ve srovnání s původní grafovou metodou, jejíž výsledek je uveden na Obr. 5.24, jsou sice segmentovány i otvory v objektu, ale i všechny světlé části v obrazu. Další ukázky provedených segmentací na snímcích mozku jsou k nahlédnutí v příloze B, Tabulka 9.



Obr. 5.24: Segmentace glioblastomu. (a) Značení oblastí. (b) Výsledek původní metody.[13]

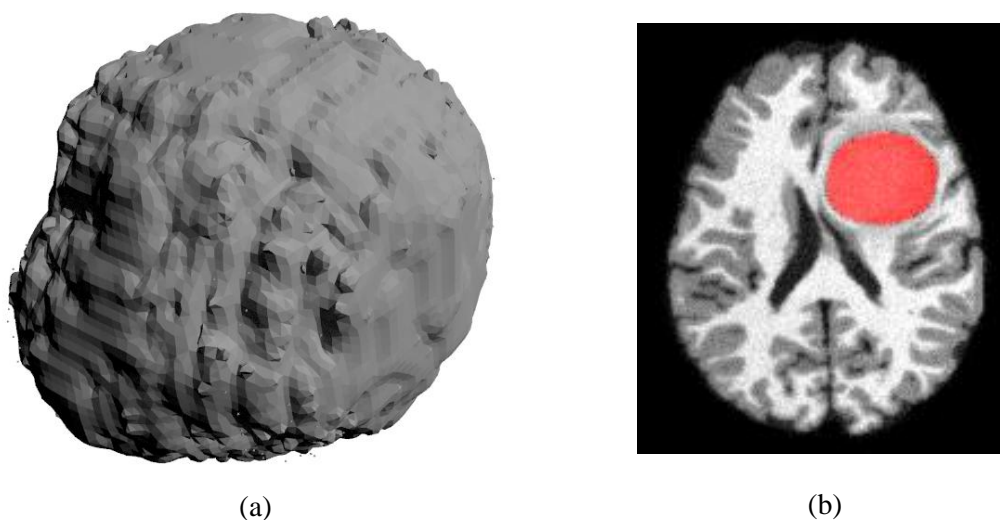
### 5.3 Testování 3D dat

Pro testování metody ve 3D provedení byla použita reálná patientská data získaná v rámci výzkumů probíhajících na Ústavu biomedicínského inženýrství. Dále je uveden příklad segmentace nádoru na mozku. Na Obr. 5.25 (a) je zobrazen T1-váhovaný MRI snímek řezu zhruba středem tohoto nádoru. Použita byla šablona koule, zkonstruovaná dle binární předlohy, jejíž tvorba je popsána v kapitole 3.2. Vzhledem k tomu, že tvar nádoru není přesně kulovitého tvaru, byl zvolen parametr  $\Delta_r=1$ . Pro lepší představu je na Obr. 5.25 (b) znázorněn počáteční bod a krajní body šablony jen orientačně v daném řezu snímku.



Obr. 5.25: (a) T1-váhovaný MRI snímek řezu středem nádoru. (b) Počáteční bod a krajních body šablony.

Výsledek získaný segmentací, tedy objem celého nádoru je uveden na Obr. 5.26 (a). Vysegmentované body jsou také vyznačeny v daném řezu z předchozího obrázku na Obr. 5.26 (b) červenou barvou.



Obr. 5.26: Výsledek segmentace nádoru mozku. (a) 3D vizualizace nádoru. (b) Vysegmentované body v řezu zhruba středem nádoru.

Jak lze vidět na Obr. 5.26 (b) s řezem přibližně uprostřed nádoru, je pro tuto část výsledek segmentace zdařilý, stejně jako ve většině segmentovaného objemu tohoto nádoru. Nedokonalosti ve výsledku nastávají pouze u krajních řezů, patrně v důsledku použití šablony koule, od které se tvar nádoru výrazně liší. Komplikací je také především proces spojování vysegmentovaných bodů šablony pro získání výsledku, který je značně komplikovaný a zavádí patrné nerovnosti ve výsledném obrysu a další případné nedostatky a odchylky v dosaženém tvaru.

## 6 Závěr

Cílem bakalářské práce bylo prostudování segmentačních metod založených na grafové reprezentaci, zejména těch s návrhem architektury grafu s využitím znalosti o tvaru segmentovaného objektu. Dále měl být proveden návrh postupu pro vytvoření šablony pro segmentaci na základě znalosti segmentovaného obrazu, realizace navržené segmentační metody a její testování na simulovaných i reálných medicínských datech v programovém prostředí MATLAB<sup>®</sup>.

V první kapitole jsou popsány základní pojmy z teorie grafů, ze kterých grafové segmentační metody obrazů vychází. Pro realizaci segmentační metody byl zvolen algoritmus hledající minimální řez utvořeným grafem. Samotná realizace segmentačního algoritmu je podrobně vysvětlena ve druhé kapitole. Konkrétně je popsáno vytvoření grafové reprezentace obrazu, tvorba implementované šablony a využití segmentační energie pro řez grafem, který je proveden pomocí algoritmu Boykova a Kolmogorova v MATLABu.

Grafová metoda byla i s využitím šablony navíc rozšířena do třetí dimenze pro možnost segmentace objemových dat, jak je uvedeno ve třetí kapitole. Pro snadnou interakci uživatele s programem bylo vytvořeno grafické uživatelské rozhraní pro definování šablony, bodů a parametrů potřebných k provedení segmentace, popsané ve čtvrté kapitole segmentace.

V páté kapitole jsou uvedeny a diskutovány výsledky testování nejprve na simulovaných datech, konkrétně obrázku obdélníku, pětiúhelníku a hvězdy včetně postupu tvorby použité šablony. Porovnány jsou výsledky segmentace s využitím grafových řezů bez a s vytvořenou šablonou na obrazech s různou úrovní Gaussovského šumu. Dle získaných výsledků lze usuzovat vyšší efektivitu segmentace při využití definované šablony, zejména při vyšší úrovni šumu v obrazu a složitějším tvaru segmentovaného objektu.

Dále bylo provedeno testování realizované segmentační metody na reálných medicínských CT a MRI snímcích obratlů a mozku. Z výsledků vyplývá, že lze vytvořený program s výhodou využít i pro segmentace objektů obtížně rozlišitelných od pozadí. I když metoda není zcela automatická a uživatel musí zadat různá značení a parametry, pomocí postupných úprav může dospět k výsledku požadované úrovně.

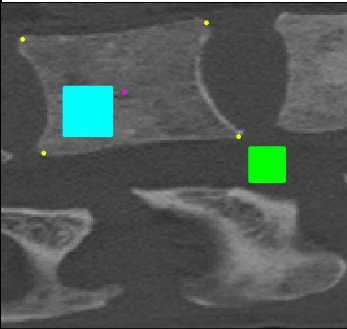
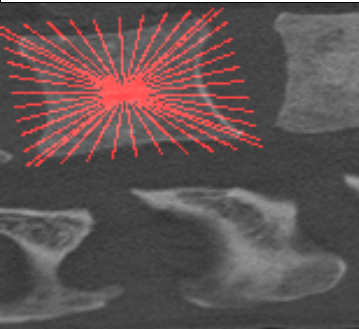
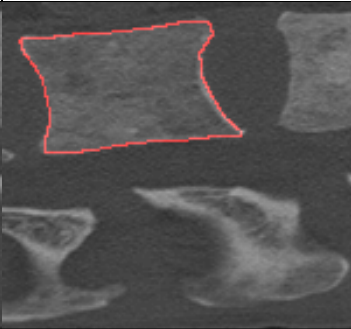
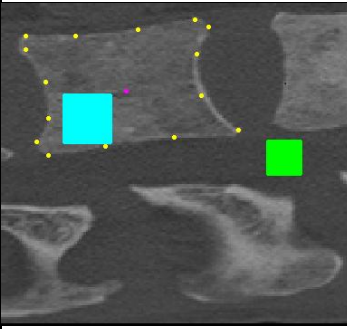
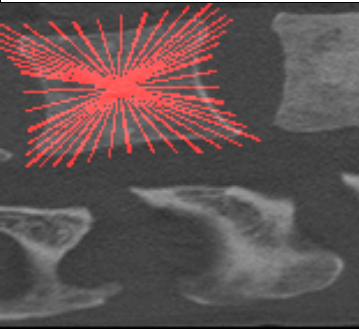
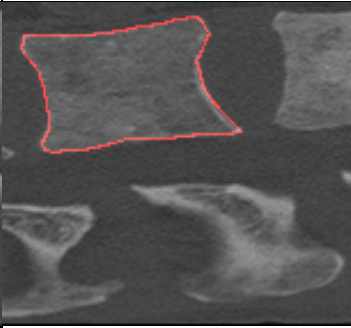
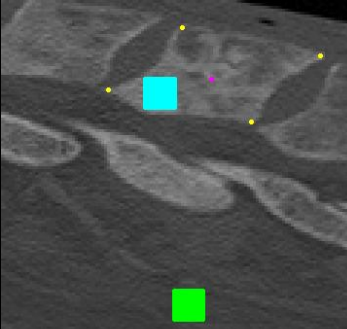
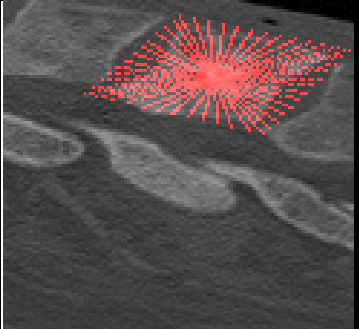
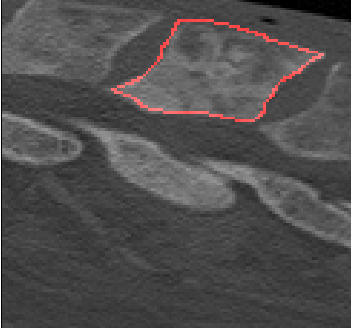
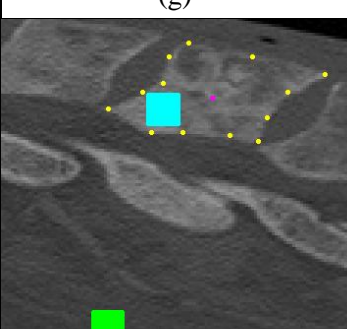
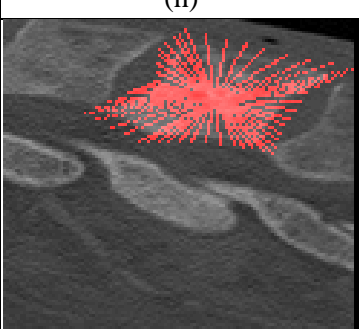

## Literatura

- [1] JAN, Jiří. *Medical Image Processing, Reconstruction and Restorations: Concepts and Methods*. Boca Raton: Taylor, 2006. ISBN 08-247-5849-8.
- [2] LÉZORAY, Olivier a Leo GRADY. *Image processing and analysis with graphs: theory and practice*. Boca Raton, FL: Taylor, 2012, xxiv, 537 p. ISBN 978-143-9855-072.
- [3] KANG LI, XIAODONG WU, D.Z. CHEN a M. SONKA. Optimal Surface Segmentation in Volumetric Images-A Graph-Theoretic Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* [online]. 2006, **28**(1): 119-134 [cit. 2015-10-11]. DOI: 10.1109/TPAMI.2006.19. ISSN 0162-8828. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1542036>
- [4] MAREŠ, Martin. *Krajinou grafových algoritmů: průvodce pro středně pokročilé*. Vyd. 1. Praha: ITI, 2007, 71 s. ISBN 978-80-239-9049-2.
- [5] FORD, L a D FULKERSON. *Flows in networks*. New Jersey: Princeton University Press, 2011, xix, 194 s. princeton landmarks in mathematics. ISBN 978-0-691-14667-6.
- [6] BOYKOV, Y. a M.-P. JOLLY. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. Proceedings of „International Conference on Computer Vision“. IEEE Computer Society. July 2001, Vol. 1, pp. 105-112. ISBN 0-7695-1143-0.
- [7] EGGER, Jan, Tina KAPUR, Thomas DUKATZ, Malgorzata KOLODZIEJ, Dženan ZUKIĆ, Bernd FREISLEBEN, Christopher NIMSKY a Yong FAN. Square-Cut: A Segmentation Algorithm on the Basis of a Rectangle Shape. *PLoS ONE* [online]. 2012-2-21, **7**(2): e31064- [cit. 2015-10-07]. DOI: 10.1371/journal.pone.0031064. ISSN 1932-6203. Dostupné z: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3283589/>
- [8] SCHWARZENBERG, Robert, Bernd FREISLEBEN, Christopher NIMSKY, Jan EGGER, Yong FAN, Bernd FREISLEBEN, Christopher NIMSKY a Yong FAN. Cube-Cut: Vertebral Body Segmentation in MRI-Data through Cubic-Shaped Divergences. *PLoS ONE* [online]. 2014-4-4, **9**(4): e93389- [cit. 2015-10-07]. DOI: 10.1371/journal.pone.0093389. ISSN 1932-6203. Dostupné z: <http://dx.plos.org/10.1371/journal.pone.0093389>
- [9] YI, Faliu a Inkyu MOON. Image segmentation: A survey of graph-cut methods. *2012 International Conference on Systems and Informatics (ICSAI2012)*. IEEE, 2012, : 1936-1941. DOI: 10.1109/ICSAI.2012.6223428. ISBN 978-1-4673-0199-2. Dostupné také z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6223428>

- [10] BOYKOV, Y. a V. KOLMOGOROV. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **26**: 1124-1137. DOI: 10.1109/TPAMI.2004.60. ISSN 0162-8828. Dostupné také z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1316848>
- [11] RUBINSTEIN, Miki. A wrapper library for Boykov and Kolmogorov maxflow/min-cut implementation. In: Mathworks [online]. 2008 [cit. 2014-12-22]. Dostupné z: <http://www.mathworks.com/matlabcentral/fileexchange/21310-maxflow>
- [12] AL SALAM, Hani. Meningioma. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-03-30]. Dostupné z: <http://radiopaedia.org/cases/meningioma>
- [13] GAILLARD, Frank. Butterfly glioma. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-03-29]. Dostupné z: <http://radiopaedia.org/cases/butterfly1802>
- [14] RABOU, Ahmed Abd. Acoustic schwannoma. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://radiopaedia.org/cases/acoustic-schwannoma-20>
- [15] GAILLARD, Frank. Basal ganglia bleed. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://radiopaedia.org/cases/basal-ganglia-bleed>
- [16] GAILLARD, Frank. Cerebral absceses. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://radiopaedia.org/cases/cerebral-absceses>
- [17] RABOU, Ahmed Abd. Glioblastoma. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://radiopaedia.org/cases/glioblastoma-20>
- [18] GAILLARD, Frank. Primary CNS lymphoma: B-cell in the setting of HIV infection. In: *Radiopaedia.org* [online]. 2016 [cit. 2016-03-29]. Dostupné z: <http://radiopaedia.org/cases/primary-cns-lymphoma-b-cell-in-the-setting-of-hiv-infection>

## A. Ukázky segmentace snímků obratlů

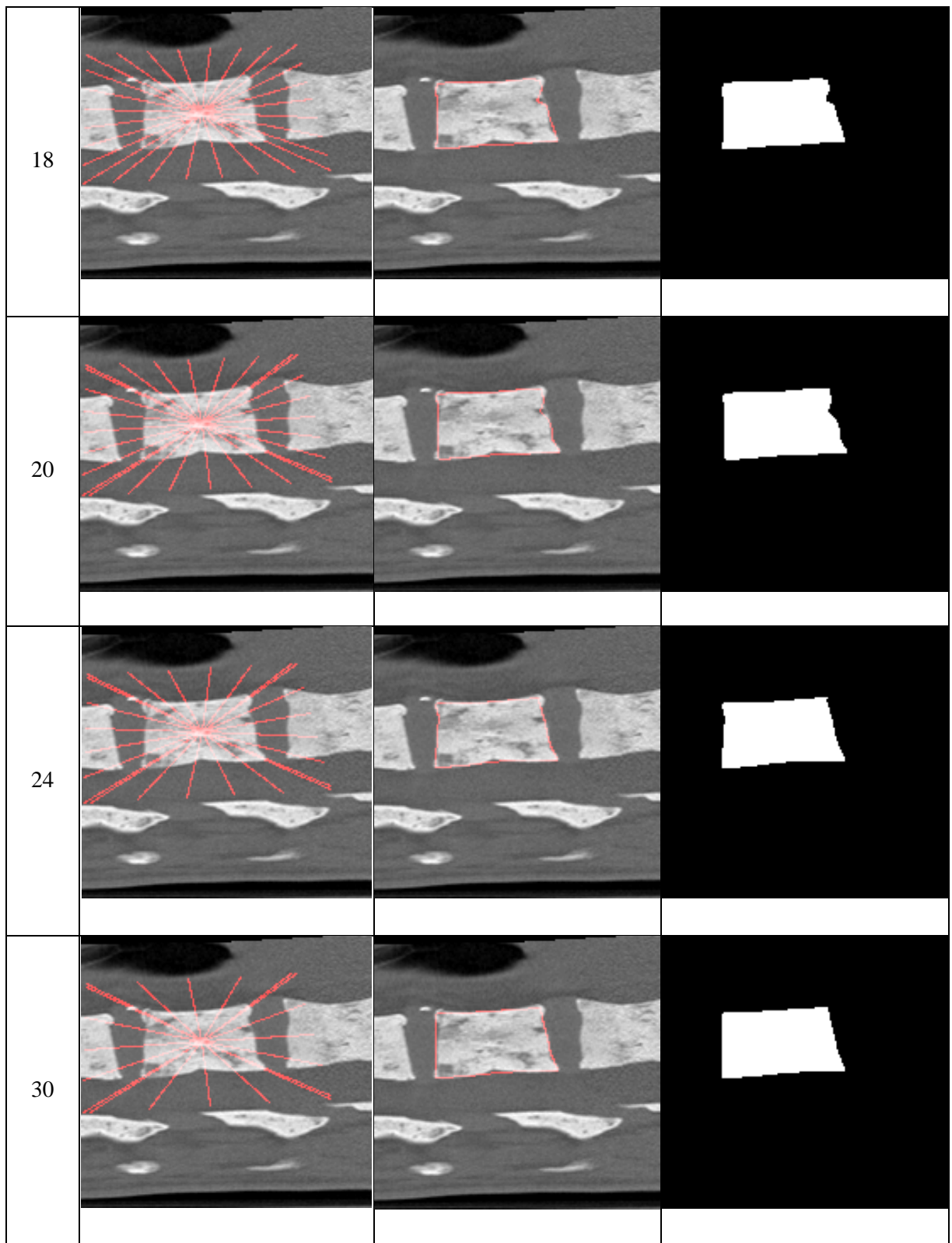
Tabulka 5: Vliv počtu zadaných krajních bodů na výsledek segmentace snímku obratle.  
(a - f) Real\_data\_7. (g - k) Real\_data\_6.

Body	Označení bodů a oblastí.	Šablona	Výsledý obrys
4			
	(a)	(b)	(c)
15			
	(d)	(e)	(f)
4			
	(g)	(h)	(ch)
15			
	(i)	(j)	(k)

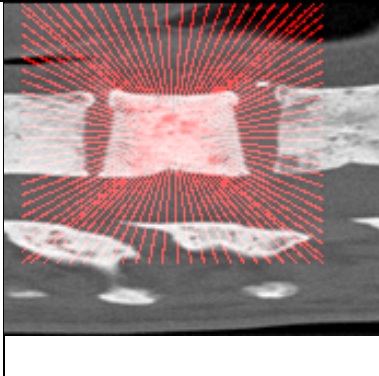
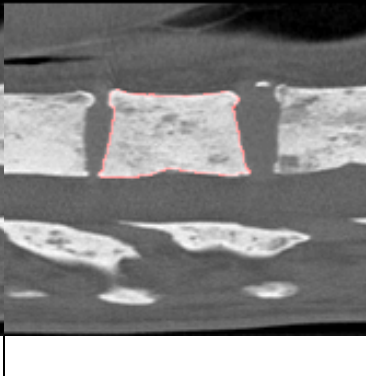

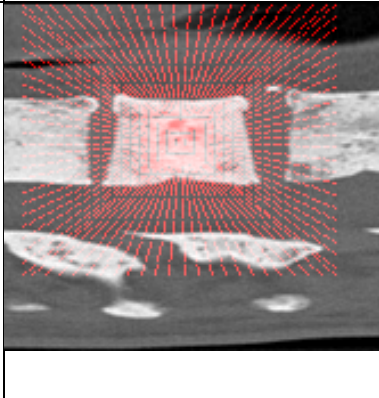
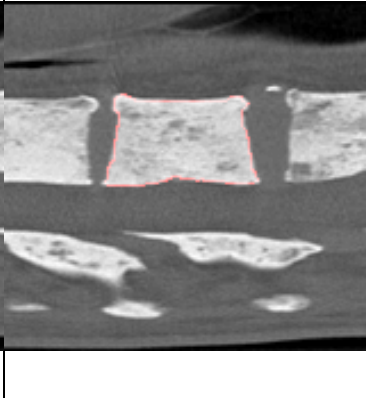

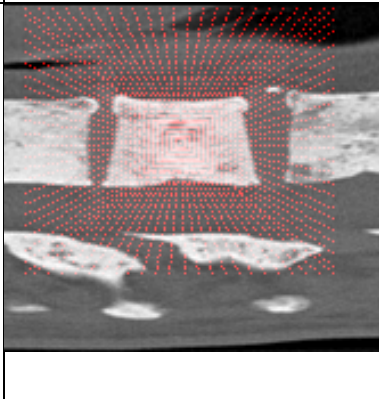
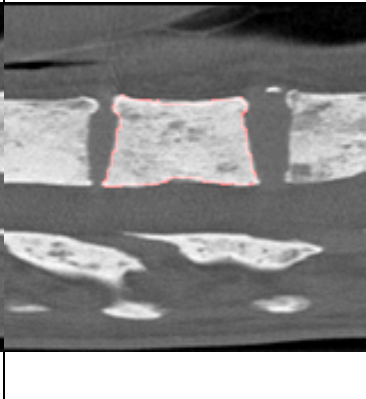

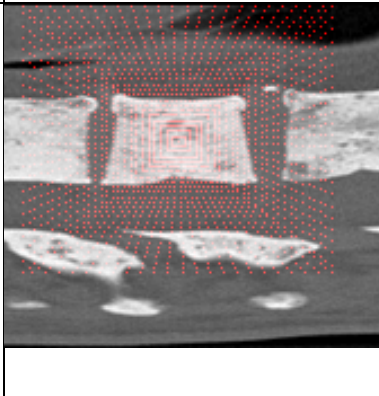
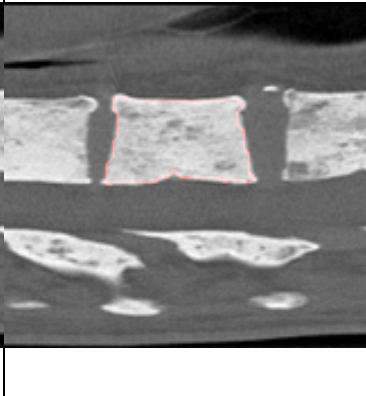



Tabulka 6: Vliv velikosti kroku mezi paprsky šablony na segmentaci obratle (real\_data\_11).

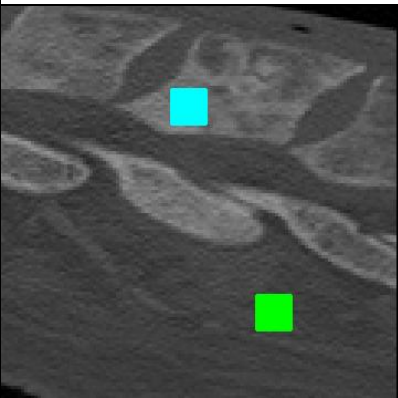


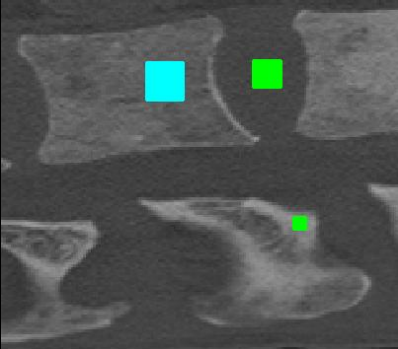


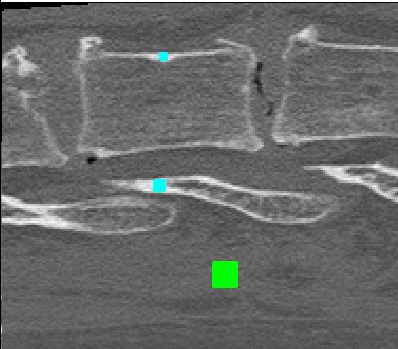
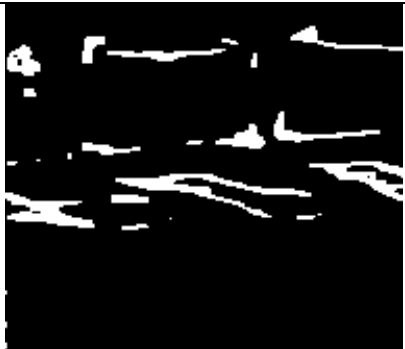

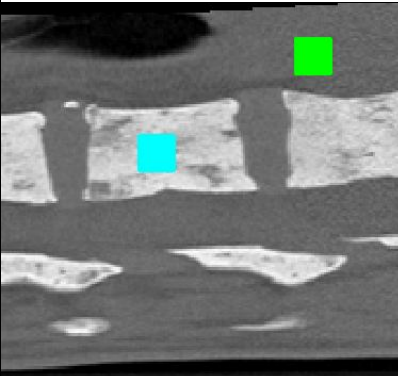


Krok	Šablona	Výsledý obrys	Výsledek
10			
12			
14			
16			



Tabulka 7: Vliv velikosti kroku mezi body paprsků šablony na segmentaci obratle (real\_data\_12) s využitím binární předlohy čtverce pro tvorbu šablony.



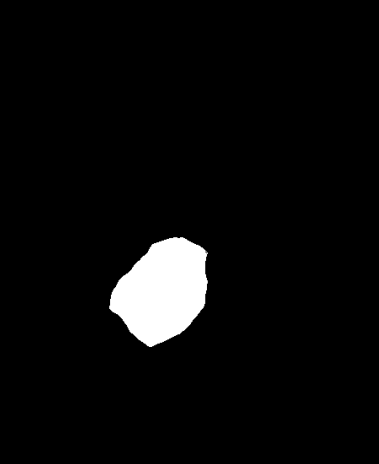


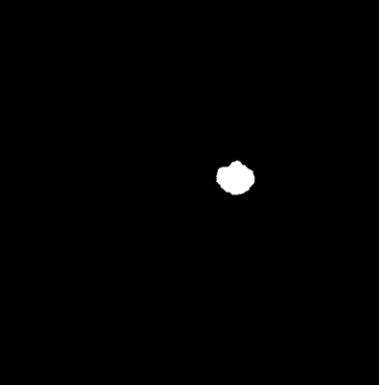
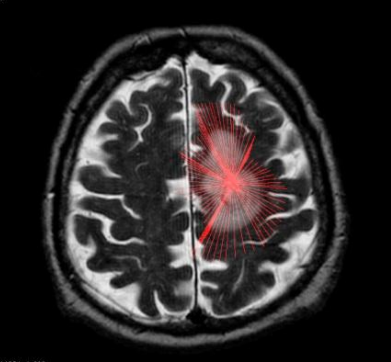
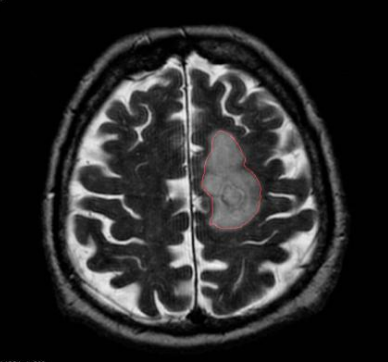

Krok	Šablona	Výsledý obrys	Výsledek
1			
2			
3			
4			

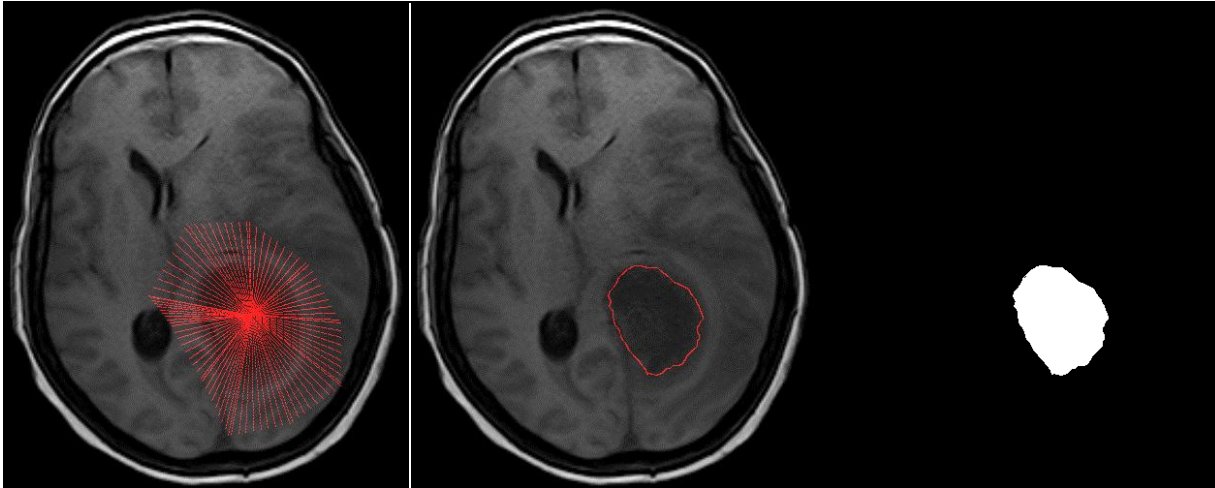
Tabulka 8: Srovnání výsledků segmentace obratlů původní nemodifikovanou grafovou metodou a její upravenou verzí s využitím šablony.

Značení (bez šablony)	Bez šablony	Se šablonou
		
Real_data_6		
		
Real_data_7		
		
Real_data_8		
		
Real_data_11		

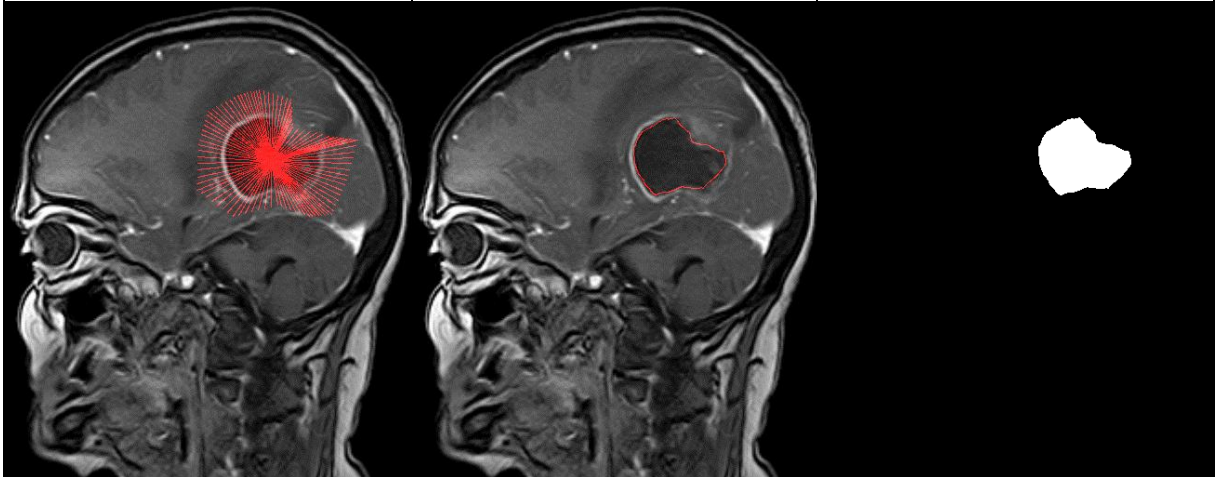
## B. Ukázky segmentace u snímků mozku

Tabulka 9: Segmentace různých abnormalit na CT a MRI snímcích mozku.

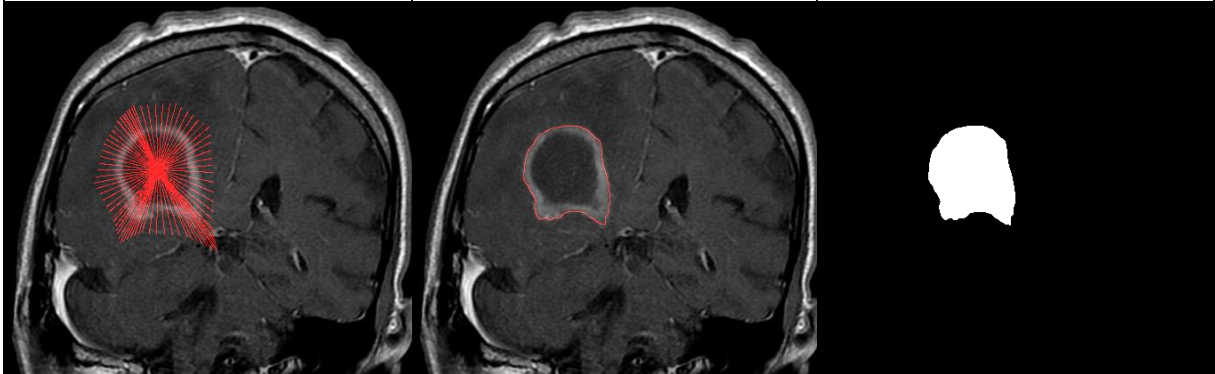
Šablona	Výsledný obrys	Výsledek
		
Vestibular schwannoma [14]		
		
Krvácení basálních ganglií [15]		
		
Cerebrální absces [16]		



Glioblastom [17]



Glioblastom [17]



Lymfom [18]