

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra kybernetiky

BAKALÁŘSKÁ PRÁCE

**Fúze výsledků segmentace
multimodálních obrazových dat**

Plzeň, 2016

Miroslav Bulka

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta aplikovaných věd
Akademický rok: 2015/2016

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Miroslav BULKA**
Osobní číslo: **A13B0601P**
Studijní program: **B3918 Aplikované vědy a informatika**
Studijní obor: **Kybernetika a řídicí technika**
Název tématu: **Fúze výsledků segmentace multimodálních obrazových dat**
Zadávající katedra: **Katedra kybernetiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Prostudujte doporučenou literaturu týkající se předzpracování obrazu a jeho segmentace.
2. Zaměřte se především na zpracování obrazových dat v lékařství.
3. Prostudujte metody registrace obrazu a fúze obrazových dat.
4. Vybrané metody otestujte na reálných datech.
5. Diskutujte získané výsledky a navrhněte možná vylepšení.

Rozsah grafických prací: **dle potřeby**
Rozsah kvalifikační práce: **30-40 stránek A4**
Forma zpracování bakalářské práce: **tištěná**
Seznam odborné literatury:

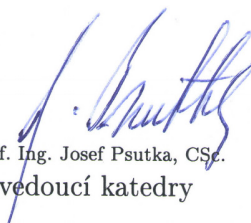
- Milan Šonka, Václav Hlaváč, Roger Boyle: **Image Processing, Analysis and Machine Vision.**
- Miloš Železný: **Přednášky k předmětu ZDO.**
- Milan Šonka, J. Michael Fitzpatrick: **Handbook of Medical Imaging.**
- Isaac N. Bankman: **Handbook of Medical Image Processing and Analysis.**

Vedoucí bakalářské práce: **Ing. Tomáš Ryba**
Nové technologie pro informační společnost

Datum zadání bakalářské práce: **1. listopadu 2015**
Termín odevzdání bakalářské práce: **13. května 2016**



Doc. RNDr. Miroslav Lávička, Ph.D.
děkan



Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

V Plzni dne 30. října 2015

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 2. května 2016

.....

vlastnoruční podpis

Poděkování

Zvláštní poděkování patří Ing. Tomáši Rybovi z oddělení umělé inteligence na Katedře kybernetiky Fakulty aplikovaných věd Západočeské univerzity v Plzni nejen za poskytnutí mnoha odborných materiálů a cenných rad, ale i za jeho obrovskou ochotu seznámit mě s danou problematikou.

Anotace

Cílem této práce je aplikovat nastudovanou teorii registrace obrazu na jednoduchá obrazová data a následně pomocí testování zhodnotit efektivitu, přesnost a vhodnost využití jednotlivých metod v daných případech. Nejdříve bude pro jednoduchost testována 2D registrace na umělých binárních obrazech, což rozhodne o tom, které způsoby registrace povedou k lepším výsledkům. Následující část se bude zabývat výběrem optimálních parametrů pro 3D registraci obrazu, a to zejména CT snímků. Nakonec budou uvedeny další způsoby registrace obrazu coby vylepšení známých metod.

Klíčová slova

Registrace obrazu, geometrická transformace, míra vzdálenosti, optimalizátor.

Annotation

The aim of this work is to apply obtained knowledge of image registration theory into simple images and then with help of testing to evaluate effectiveness, accuracy and suitability of the use each method in stated cases. At first, to simplify, there will be tested 2D registration on artificial binary images, which will decide what registration methods will lead to better results. The following part will concern with selection of the optimal parameters of 3D image registration, especially of CT images. At the end, another registration methods will be presented as improvement of the known ones.

Keywords

Image registration, geometric transformation, distance measure, optimizer.

Obsah

1	Úvod	2
2	Registrace obrazu	4
2.1	Transformace obrazu	4
2.2	Realizace	7
2.3	Průběh testování	7
2.3.1	Transformace vztažného obrázku	7
2.3.2	Registrace obrazu	10
3	Testování 2D registrace	14
3.1	Translace	14
3.2	Nerigidní transformace	17
3.3	Afinní transformace	21
3.4	Transformace B-spline funkcemi	25
3.5	Dva rozdílné snímky	27
3.5.1	Nerigidní transformace	28
3.5.2	B-spline transformace	30
4	Testování 3D registrace	32
4.1	Výběr parametrů 3D registrace	32
4.1.1	Testování	33
4.2	Další způsoby registrace CT snímků	40
4.2.1	Testování registrace CT snímků	42
5	Závěr	47
	Literatura	49
	Seznam tabulek	50
	Seznam obrázků	52

Kapitola 1

Úvod

Představme si dvě fotografie jednoho konkrétního objektu, kde první fotografie bude pořízena na šířku a druhá na výšku. Obě digitální fotografie si otevřeme v prohlížeči. Nejdříve si prohlédneme snímek pořízený na šířku, označíme ho jako vztážený a zapamatujeme si jej, přičemž budeme mít informaci o tom, že na druhém snímku se nachází tentýž objekt jako na vztáženém. Jelikož je ale druhá fotografie pořízena na výšku, bude tento objekt otočený o 90° . Otočení můžeme chápat jako geometrickou transformaci, v tomto případě rotaci o hodnotě 90° . Člověk, který si obrázky prohlíží, automaticky rozpozná, že se jedná o dvě stejné fotografie, z nichž jedna je otočená, a tu následně otočí zpět tak, aby si obě navzájem odpovídaly. Jeho činnost můžeme popsat jako nalezení geometrické transformace, pomocí které by se druhý (otočený) obrázek převedl na vztážený. Tento proces se nazývá registrace obrazu, kterou se zabývají Fitzpatrick aj. v publikaci [1], a jedná se o významnou disciplínu počítačového vidění. Cílem této práce bude seznámit se s teorií registrace obrazu, aplikovat získané poznatky při implementaci registračních metod v jazyku Python a následně provést řadu testování, jejichž výsledkem bude optimální nastavení parametrů pro registraci lékařských obrazových dat, v našem případě CT snímků.

Algoritmus registrace bude nutné vhodně implementovat, aby bylo možné provést jeho testování. Z tohoto důvodu jsem pro implementaci zvolil programovací jazyk Python s využitím knihovny SimpleITK [2], která nabízí široký výběr metod a parametrů registrace. V první řadě bude důležité si uvést řadu parametrů, jejichž nastavení budeme během testování registrace měnit. Budou jimi geometrická transformace, míra vzdálenosti mezi dvěma obrazy a optimalizátor, který se míru vzdálenosti snaží minimalizovat. V oblasti geometrických transformací se budu zabývat translacemi, nerigidními transformacemi za využití translace, rotace a změny měřítka, afinními transformacemi a B-spline transformacemi, o kterých pojednávají Fitzpatrick aj. v publikaci [1]. Z optimalizátorů věnuji pozornost metodám LBFGS a klesání gradientu a jako míru vzdálenosti budu používat součet kvadrátů odchylek, korelaci a vzájemnou informaci.

Hlavním úkolem bude prozkoumat vliv zvolených parametrů registrace na její výslednou kvalitu, jejímž důležitým ukazatelem bude hodnota míry vzdálenosti na konci procesu. Abych co nejlépe demonstroval výsledky registrace, budou se testování provádět nejdříve na umělých binárních 2D obrazech, kde budou registrovány dva obrázky, které se od sebe

budou lišit pouze tím, že jeden z obrázků bude vytvořen transformací druhého snímku. Před spuštěním registrace si parametry této geometrické transformace zaznamenám a následně je porovnám s parametry geometrické transformace, která bude jedním z výstupů registračního procesu. Další důležitou hodnotou, kterou při registraci obrazu obdržíme, bude výsledná míra vzdálenosti mezi oběma obrazy na konci výpočtu. Testování registrace binárních 2D obrazů budu provádět za použití všech vybraných typů geometrické transformace, míry vzdálenosti a optimalizátoru. Na základě výsledků vyberu optimalizátor, který povede k nejlepším výsledkům ve smyslu kvality registrace. Následně přidám do jednoho z uměle vytvořených obrazů další objekty, které se ve druhém vyskytovat nebudou, což je situace, kterou očekávám u práce s lékařskými obrazovými daty. Rovněž změním poměr intenzit objektů v obou obrazech a opustím oblast binárních dat. Testování 2D registrace za přítomnosti aditivních objektů v jednom ze vstupních obrazů povede k určení nejhodnější míry vzdálenosti pro registraci lékařských obrazových dat.

V dalších kapitolách se již budu věnovat 3D registraci CT snímků. Dodávám, že medicínská data byla poskytnuta Fakultní nemocnicí v Plzni v rámci smlouvy o spolupráci mezi FN Plzeň a ZČU v Plzni. Veškerá data byla anonymizována, tj. zbavena veškerých osobních a citlivých údajů (jméno, adresa, rodné číslo). Oba CT snímky, které budou do registračního procesu vstupovat, byly pořízeny ve dvou rozdílných fázích, což má za následek jejich vzájemnou odlišnost v rozdělení jasů. A jelikož mezi oběma fázemi nastala určitá časová prodleva, byly snímky rovněž pořízeny v jiný okamžik. Mezi časy pořízení obou snímků tak mohlo dojít ke změně, kterou interpretujeme jako geometrickou transformaci. Vzhledem k tomu, že každý z dvojice vstupních obrazů byl získán během jiné fáze v jiném časovém okamžiku, můžeme data chápat jako multimodální, přestože pocházejí ze stejné modalit, kterou je CT. Během práce s lékařskými obrazovými daty bude nutné si uvědomit, že oblastí našeho zájmu jsou játra, tedy tkáň, jejíž změny budou spočívat převážně v transformaci elastického charakteru, proto jako geometrickou transformaci pro registraci CT snímků zvolíme B-spline transformaci. Testování registrace CT snímků bude prováděno za účelem zvolení optimálních hodnot parametrů, kterými jsou konstanta učení čili parametr optimalizátoru [2] a velikost mřížky, což je parametr B-spline transformace (viz [3]).

V poslední řadě bych se rád zaměřil na pokus o možné vylepšení 3D registrace, kterým bude použití 2D registrace po řezech, kdy místo celých CT snímků budou registrovány jejich jednotlivé řezy, které budou odpovídat stejné úrovni v ose z . Následně obě metody registrace 3D obrazu porovnám a uvedu, za jakých podmínek bych doporučil zvolit právě 2D registraci po řezech.

Kapitola 2

Registrace obrazu

2.1 Transformace obrazu

Velmi důležitou disciplínou počítačového vidění je registrace obrazu, která slouží k vyrovnání dvou rozdílných obrazů pomocí geometrické transformace. Problematikou se podrobně zabývají Fitzpatrick aj. v publikaci [1]. Úkolem registrace je najít vztah mezi oběma obrazy určením parametrů transformace T , která mapuje body z prostoru X do prostoru Y tak, že převede bod v prostoru X reprezentovaný vektorem x na transformovaný bod x' :

$$x' = T(x). \quad (2.1)$$

Transformace se dělí na dva základní typy: rigidní a nerigidní. Rigidní transformace, na rozdíl od nerigidní, zachovává všechny vzdálenosti a přímost čar. Transformace je tvořena dvěma složkami, translací a rotací, a její zápis vypadá následovně:

$$x' = Rx + t, \quad (2.2)$$

kde R je ortogonální matice 3×3 a t je vektor translace. Pro úlohy, které mají fyzikální význam, je vhodnější využít nerigidní transformace. Tyto transformace, jak už bylo řečeno výše, nezachovávají vzdálenosti. Nejjednodušší nerigidní transformaci získáme přidáním prvku násobící složky do rigidní transformace:

$$x' = RSx + t, \quad (2.3)$$

kde S je diagonální matice, jejíž prvky reprezentují násobící složky podél souřadnicových os. V této práci budu využívat transformaci, kde je násobení izotropní. Předpis transformace bude dán následujícím vztahem:

$$x' = sRx + t, \quad (2.4)$$

kde s je kladný skalár. Složitější nerigidní transformací je velice rozšířená afinní transformace:

$$x' = Ax + t. \quad (2.5)$$

Afinní transformaci lze reprezentovat pomocí homogenních souřadnic, kde jsou A a t složeny do matice M o rozměrech 4×4 , jejíž prvky jsou definovány následovně: $m_{ij} = a_{ij}$; $m_{i4} = t_i$; $m_{4j} = 0$; $m_{44} = 1$; $i = 1, 2, 3$; $j = 1, 2, 3$.

$$u' = \begin{pmatrix} u'_1 \\ u'_2 \\ u'_3 \\ 1 \end{pmatrix} = Mu = \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ a_{31} & a_{32} & a_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ 1 \end{pmatrix}, \quad (2.6)$$

kde $u_i = x_i$ a $x'_i = u'_i$ pro $i = 1, 2, 3$ a $u_4 = u'_4 = 1$. Afinní transformace zachovává přímost čar a rovinnost povrchů a je vhodná pro zkosené obrázky.

Obecněji lze rozdělit nerigidní transformace na projektivní transformace, které zachovávají přímost čar a rovinnost povrchů, a na transformace křivkami, které tak nečiní. Základní tvar projektivní transformace:

$$x' = (Ax + t) / (p \cdot x + \alpha) \quad (2.7)$$

lze zapsat v homogenních souřadnicích:

$$u' = \begin{pmatrix} u'_1 \\ u'_2 \\ u'_3 \\ u'_4 \end{pmatrix} = Mu = \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ a_{31} & a_{32} & a_{33} & t_3 \\ p_1 & p_2 & p_3 & \alpha \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ 1 \end{pmatrix}, \quad (2.8)$$

kde již není nutné, aby $u'_4 = 1$, a kde $m_{4j} = p_j$, $m_{44} = \alpha$ a $x'_i = \frac{u'_i}{u'_4}$ pro $i, j = 1, 2, 3$.

Pokud transformace nezachovává přímost čar, jedná se o transformace křivkami. Jejich nejjednodušší forma má tvar:

$$x' = \sum_{ijk}^{IJK} c_{ijk} x^i y^j z^k, \quad (2.9)$$

kde c_{ijk} je tříprvkový vektor koeficientů pro i, j, k v polynomiálním výrazu pro složky x', y', z' obrazu x' .

Výsledná transformace je definována jako dělení prostoru do množiny třídídimenzionálních kvádrů pomocí tří množin řezných rovin, z nichž každá je kolmá k jedné z os kartézské soustavy souřadnic. Uvnitř vytvořených kvádrů i, j, k dimenzí u_i, v_j, w_k platí:

$$T = P_{ijk}^{(x)}(x) P_{ijk}^{(y)}(y) P_{ijk}^{(z)}(z), \quad (2.10)$$

kde každý $P()$ je jednorozměrný polynom stupně m definovaný pouze na příslušném kvádru. Rozumnou volbou koeficientů polynomu můžeme zařídit, aby se polynomy spojovaly plynule přes hranu kvádru. Takovýmto transformacím se říká spline funkce a jednotlivé rohy x_{ijk} kvádrů se nazývají uzly. Nejčastějším případem je kubická spline křivka, pro kterou platí, že T a její první i druhá derivace jsou spojitě.

B-spline křivky jsou křivky volného tvaru, které se skládají ze segmentů Bézierových křivek stejného stupně m , viz [4]. Kubické B-spline funkce, jejichž každá část je polynom třetího řádu, jsou nejčastějším případem, jak uvádějí Šonka aj. ve své práci [5], kde jsou B-spline křivky dány předpisem:

$$x(s) = \sum_{i=0}^{n+1} v_i B_i(s), \quad (2.11)$$

kde v_i jsou koeficienty reprezentující spline křivku a $B_i(s)$ jsou bázové funkce, jejichž tvar je dán řádem spline.

V ojedinělém případě mohou být B-spline funkce vyjádřeny tak, jak se uvádí v publikaci [1]:

$$x'(l, m, n, \alpha, \beta, \gamma) = \sum_{ijk} B_{i-l}(\alpha) B_{j-m}(\beta) B_{k-n}(\gamma) c_{ijk}, \quad (2.12)$$

kde každé $B_q(t)$ je segmentem kubické B-spline křivky a $B_q(t) = 0$, s výjimkou $-1 \leq q \leq 2$. Pro $l, m, n, \alpha, \beta, \gamma$ platí, že l, m, n jsou celá čísla a $0 \leq \alpha, \beta, \gamma \leq 1$.

2.2 Realizace

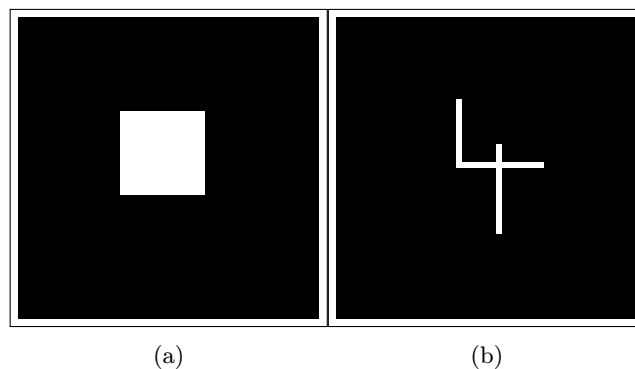
Nastudovanou teorii jsem se rozhodl implementovat v programovacím jazyku Python, pro který existuje řada knihoven umožňujících práci s digitalizovaným obrazem. Jednou z nich je i ITK, jejíž zjednodušenou verzi SimpleITK [2] budu v algoritmech využívat, jelikož ITK je obzvláště v oblasti zpracování lékařských obrazových dat velmi rozšířeným nástrojem.

Pro implementaci algoritmu registrace budeme rovněž využívat důležitou knihovnu NumPy a pro vykreslování obrázků a grafů je možné využít knihovnu matplotlib.

2.3 Průběh testování

- Načtení vztažného obrázku,
- transformace vztažného obrázku dle zvolených parametrů, čímž vytvoříme snímek, který označíme jako registrovaný,
- registrace obrazu,
- grafické znázornění výsledků a jejich analýza.

2.3.1 Transformace vztažného obrázku



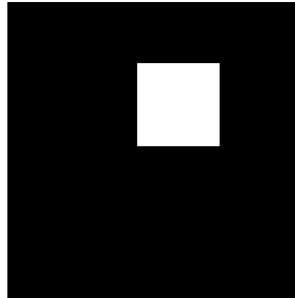
Obrázek 2.1: Výchozí obrázky

Existuje řada geometrických transformací obrazu [1]. V následujícím odstavci se zaměřím především na tyto typy transformací:

- translace,
- nerigidní transformace (translace + rotace + změna měřítka),
- afinní transformace,
- transformace B-spline funkcemi.

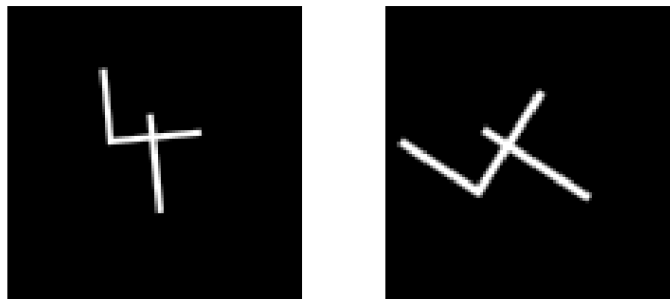
Každou z těchto transformací aplikujeme na jeden ze dvou vytvořených výchozích obrázků (obr. 2.1).

Při translaci se posune výchozí objekt určitým směrem, přičemž zachovááme všechny vzdálenosti. Požadujeme, aby čtverec uprostřed obrazu byl po transformaci posunutý doprava nahoru.



Obrázek 2.2: Registrovaný obrázek

Nyní kromě translace použijeme ještě rotaci a změnu měřítka obrázku. Zde se už nezachovávají vzdálenosti. Tvar objektů zůstává po transformaci nezměněn.



(a)

(b)

Obrázek 2.3: Výsledek nerigidní transformace

Do transformace nyní zahrneme i zkosení objektu. Taková transformace se nazývá afinní (obrázek 2.4).

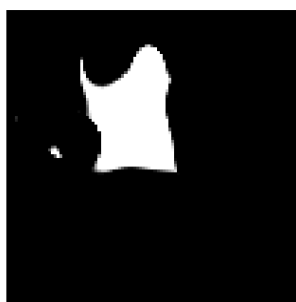
Velmi známou nerigidní transformací je i transformace s použitím B-spline funkcí, jejíž výsledek můžeme vidět na obrázku 2.5.



(a)

(b)

Obrázek 2.4: Výsledek afinní transformace



Obrázek 2.5: Výsledek B-spline transformace

2.3.2 Registrace obrazu

Registrace je nejdůležitějším krokem tohoto testování. Před zahájením registrace musíme zvolit několik parametrů, které udávají, kterou dvojici snímků budeme registrovat a jakým způsobem to budeme provádět. Nejdůležitějšími parametry jsou podle SimpleITK [2]:

- **dvojice vstupních obrazů** - vztažený a registrovaný snímek,
- **geometrická transformace**, podle které budeme upravovat registrovaný snímek,
- **míra vzdálenosti** - kritérium, které znázorňuje, jak si oba obrazy navzájem odpovídají,
- **optimalizátor** - nastavuje parametry tak, aby optimalizoval míru vzdálenosti.

Při registraci obrazu se ukázalo jako vhodné vybrat konkrétní geometrickou transformaci a postupně u ní otestovat průběh všech níže uvedených typů míry vzdálenosti. V některých případech jsem rovněž využil i různé optimalizátory.

Připomeňme si, že transformací vztažného obrázku jsme získali transformovaný registrovaný obraz. Prvním krokem v registračním procesu byl výběr transformace. Pro účely testování efektivity a přesnosti metod byl volen vždy typ transformace, pomocí které jsme získali registrovaný obraz.

Dalším a rovněž velmi důležitým krokem v procesu registrace je výběr míry vzdálenosti. Obvykle se pro toto kritérium používá označení míra podobnosti. Vzhledem k tomu, že optimalizátory v SimpleITK [2] se snaží míru minimalizovat, nižší hodnota míry by znamenala vyšší podobnost mezi obrazy, což by bylo zavádějící. Vhodnější tedy bude používat označení míra vzdálenosti, jejíž snižování bude znamenat zvyšování vzájemné podobnosti obou obrazů. Ze známých typů míry vzdálenosti jsem vybral čtyři, o kterých budou pojednávat následující odstavce.

Součet kvadrátů odchylek

Pro obrazy A a B s voxely i se snažíme minimalizovat hodnotu SSD (z anglického Sum of Squares of Differences), jejíž výpočet je uveden v publikaci [1]:

$$SSD = \sum_i |A(i) - B'(i)|^2. \quad (2.13)$$

Podle knihovny SimpleITK [2] bude v ideálním případě SSD na hodnotě 0, pokud budou mít vzájemně odpovídající si voxely stejnou hodnotu, což by platilo pro identické obrazy.

Korelace

Optimalizátory v knihovně SimpleITK [2] se vždy snaží míru vzdálenosti minimalizovat, tudíž se pro výpočet korelace používá záporný normalizovaný korelační koeficient, který

se budeme snažit minimalizovat:

$$NC = -\frac{\sum_i (A_i \cdot B_i)}{\sqrt{\sum_i A_i^2 \cdot \sum_i B_i^2}}, \quad (2.14)$$

kde A_i , resp. B_i , jsou i -té voxely obrazů A , resp. B .

Pro jednoduchost budeme dále pro záporný normalizovaný korelační koeficient, který nabývá záporných hodnot, používat označení korelační koeficient. Jeho hodnota bude v ideálním případě rovna -1, nikoliv 1. Využití korelačního koeficientu jako míry vzdálenosti je vhodné v případě, že jsou intenzity objektů v lineárním vztahu, jak je popsáno v publikaci [1].

Vzájemná informace

Vzájemnou informací se detailně zabývají Fitzpatrick aj. v publikaci [1]. Sdružená PDF (z anglického probability distribution function) je funkcí intenzit v n obrazech. K jejímu odhadu slouží sdružený histogram dimenze n , na jehož osách jsou hodnoty intenzit pro jednotlivé obrazy a hodnota v každém bodě znázorňuje počet voxelů, které těchto intenzit nabývají.

Dalším pojmem je sdružená entropie H , která podle teorie informace udává informaci od množiny symbolů $\{s\}$ a jejich pravděpodobností $\{p(s)\}$, které slouží k jejímu výpočtu:

$$H = -\sum_s p(s) \log p(s). \quad (2.15)$$

V registraci obrazu se entropie využívá k vyšetření sdružené PDF a histogramu. Chybná registrace vede ke zvýšení hodnoty entropie, tudíž je hlavním úkolem entropii minimalizovat:

$$H = -\sum_{j,k} PDF[j, k] \log PDF[j, k]. \quad (2.16)$$

Tento způsob není příliš robustním řešením. Vzájemná informace (mutual information, MI) normalizuje entropii a může ji nahradit jako míru vzdálenosti během registračního procesu. Na rozdíl od entropie, kterou bylo nutno minimalizovat, je naopak snahou registrace vzájemnou informaci maximalizovat:

$$MI(A, B') = H(A) + H(B') - H(A, B'), \quad (2.17)$$

kde $H(A, B')$ je sdružená entropie a $H(A)$ a $H(B')$ jsou marginální entropie, které se vypočítají pomocí vztahů:

$$H(A) = \sum_j \left(\sum_k PDF[j, k] \log \sum_l PDF[j, l] \right), \quad (2.18)$$

$$H(B') = \sum_k \left(\sum_i PDF[i, k] \log \sum_j PDF[j, k] \right). \quad (2.19)$$

Vzájemná informace však nemusí být funkční pro některé typy klinických snímků, obzvláště pro ty, které obsahují velké množství šumu. Z tohoto důvodu existuje upravená normalizovaná vzájemná informace:

$$NMI(A, B') = \frac{H(A) + H(B')}{H(A, B')}. \quad (2.20)$$

Stejně jako u korelace se setkáváme s tím problémem, že i vzájemná informace by měla být teoreticky maximalizována. SimpleITK [2] se však snaží vždy míru vzdálenosti minimalizovat, a tak místo vzájemné informace počítá negativní vzájemnou informaci, která nabývá záporných hodnot. Rovněž si pro jednoduchost stanovme, že pod pojmem vzájemná informace budeme mít na mysli právě negativní vzájemnou informaci, kterou se bude optimalizátor snažit minimalizovat.

Knihovna SimpleITK nám nabízí tyto typy vzájemné informace:

- **joint histogram mutual information**, pro kterou si kvůli jednoduchosti zavedeme označení JHMI,
- **Mattes mutual information**, pro kterou si zavedeme označení MMI.

Když je nastavený typ míry vzdálenosti, zbývá určit optimalizační metodu, podle které se budou nastavovat parametry za účelem minimalizovat hodnotu jedné z výše zmíněných měř vzdálenosti.

Nejčastějšími optimalizátory uváděnými v knihovně SimpleITK [2] jsou:

- metoda klesání gradientu:
 - pomocí pravidelného kroku,
 - pomocí metody line search,
- LBFGB - z anglického Limited memory Broyden Fletcher Goldfarb Shannon minimization.

S výběrem optimalizátoru je rovněž vhodné si nastavit jeho parametry, jakými jsou například zastavovací podmínky a maximální počet iterací, aby registrace proběhla v rozumnou dobu. Výsledkem optimalizačního procesu je optimalizovaná míra vzdálenosti a parametry transformace, které se následně využijí k převedení registrovaného snímku na obraz, který by měl co nejlépe odpovídat vztažnému snímku.

Pro každou transformaci jsem použil všechny čtyři míry vzdálenosti. Ve většině případů byla každá z nich upravována dvěma rozdílnými optimalizátory. Směrodatný je pro nás především průběh míry vzdálenosti. Pokud bude optimalizována, získaná transformace by měla převést registrovaný obrázek na vztažný.

Pro ověření správnosti výsledků je rovněž možné vedle míry vzdálenosti zkontrolovat i výsledné parametry transformace, které registrační proces vyprodukuje. Je ovšem nutné si uvědomit, že registrace se snaží najít transformaci, která by převedla registrovaný obrázek na vztažný, a protože nastavená transformace T převádí v našem případě vztažný obraz

na registrovaný, jednalo by se o opačnou transformaci. Původní transformace, kterou lze získat registrovaný obraz, má podle rovnice 2.1 tvar:

$$x' = T(x).$$

Registrace hledá transformaci, která převede x' na x :

$$x = T^{-1}(x'). \quad (2.21)$$

Například translace bude mít tvar:

$$x' = x + t,$$

a tedy:

$$\begin{aligned} x &= x' - t, \\ x &= x' + t', \end{aligned} \quad (2.22)$$

kde $t' = -t$ jsou hledané parametry transformace, která bude výsledkem registračního procesu a její tvar bude:

$$x = T^{-1}(x') = x' + t' = x' - t. \quad (2.23)$$

Druhou možností je zaměnit vstupní obrazy registrace, a tedy hledat transformaci, která by vztažný obraz převedla na registrovaný, tedy přesně to, co bylo provedeno za účelem získání registrovaného obrazu. Za předpokladu přesné registrace by tato metoda vedla k nalezení transformačních parametrů, které jsme nastavili. Vzhledem k tomu, že ve vytvořených experimentálních obrazech mají hlavní objekty vhodnou velikost a tvar, jsem zvolil variantu, kdy získáme opačné parametry a transformujeme registrovaný obrázek.

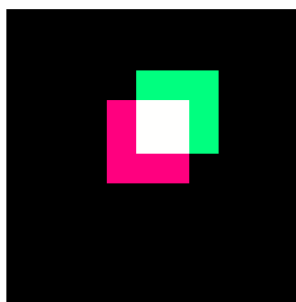
Na začátku znázorníme obrázek (obr. 3.1), kde budou sloučeny vztažný a registrovaný snímek. Fialová barva představuje vztažný obrázek, zelená registrovaný a bílá barva znázorňuje shodu, tj. místa, která souhlasí pro oba obrázky. Čím úspěšnější registrační metoda bude, tím více bude ve výsledném obrázku bílých pixelů, jelikož je žádoucí, aby se vztažný obrázek co nejvíce shodoval s transformovaným, tedy s registrovaným obrazem transformovaným podle nalezených parametrů.

Kapitola 3

Testování 2D registrace

3.1 Translace

V případě, že objekt transformujeme pouze pomocí translace, stačí pouze nastavit parametr $t = \begin{pmatrix} -10 & 10 \end{pmatrix}$.



Obrázek 3.1: Stav na začátku registračního procesu

Z tohoto jednoduchého příkladu, kdy zaznamenáváme pouze posun objektu, lze snadno předem odhadnout, že v případě úspěšného průběhu registrační metody bude zelená oblast na obrázku 3.1 postupně mizet vlivem posunu registrovaného obrazu směrem ke vztažnému (fialová barva). Pro translaci jsem použil všechny čtyři míry vzdálenosti a pro každou dva optimalizátory.

Součet kvadrátů odchylek

Účelem zavedení součtu kvadrátů odchylek jako míry vzdálenosti je tuto míru minimalizovat, tudíž hodnota 0, která byla výsledkem na konci procesu (viz tabulka 3.1), je důkazem toho, že registrační metoda pro tuto míru proběhla přesně a vedla k požadovaným výsledkům.

Optimalizátor	Míra vzdálenosti
Metoda line search	0
LBFGB	0

Tabulka 3.1: Výsledná míra vzdálenosti

Nalezené parametry transformace jsou například pro použití optimalizační metody klesání gradientu:

$$t = \begin{pmatrix} 9.999999999999966 & -9.999999999999964 \end{pmatrix}.$$

Můžeme si všimnout, že jednotlivé složky vektoru translace mají opačná znaménka než u nastavených parametrů. Je nutné ovšem vzít v úvahu, že během vytváření registrovaného obrazu jsme transformovali vztažný snímek, zatímco úkolem registrace je najít vztah, podle kterého se převede registrovaný obraz na vztažný a transformace tudíž probíhá opačným směrem. Pro lepší přehled bychom tedy mohli na začátku registračního procesu předpokládat, že požadované parametry transformace budou $t = \begin{pmatrix} 10 & -10 \end{pmatrix}$. Po tomto upřesnění je mezi požadovanými a získanými parametry transformace jen zanedbatelný rozdíl, který se ve vizualizaci výsledků neprojevívá.

Korelace

Optimalizátor	Míra vzdálenosti
Metoda line search	-1
LBFGB	-1

Tabulka 3.2: Výsledná míra vzdálenosti

Snahou je minimalizovat korelační koeficient. Hodnota -1 značí přesnou registraci, což v tomto případě platí rovněž pro oba optimalizátory. Například pro optimalizátor LBFGB jsou získané parametry:

$$t = \begin{pmatrix} 10.000000017263721 & -10.0000000172637 \end{pmatrix}.$$

Vzájemná informace

Protože snahou optimalizátoru je minimalizovat vzájemnou informaci, bylo nejlepším řešením použití MMI s optimalizátorem klesání gradientu pomocí metody line search, což můžeme usoudit nejen z hodnoty vzájemné informace v tabulce 3.3, ale i z toho, že parametry translace nejlépe odpovídají požadovanému $t' = -t = \begin{pmatrix} 10 & -10 \end{pmatrix}$. Naopak největší odchylku od požadovaných parametrů vykazoval případ použití optimalizátoru

Optimalizátor	Míra vzdálenosti	t
Metoda line search	-0.5451700	$\begin{pmatrix} 9.9925 & -9.9962 \end{pmatrix}$
LFGS	-0.2215712	$\begin{pmatrix} 10.0266 & -16.6091 \end{pmatrix}$

Tabulka 3.3: Výsledek registrace v případě použití míry vzdálenosti MMI.

Optimalizátor	Míra vzdálenosti	t
Metoda line search	-0.5282200	$\begin{pmatrix} 10.0066 & -9.9729 \end{pmatrix}$
LFGS	-0.5206918	$\begin{pmatrix} 9.96754 & -9.9690 \end{pmatrix}$

Tabulka 3.4: Výsledek registrace v případě použití míry vzdálenosti JHMI.

LFGS pro stejný typ vzájemné informace, tedy MMI, což můžeme vyčíst již z hodnoty míry vzdálenosti v tabulce 3.3, která je oproti ostatním svého typu výrazně vyšší.

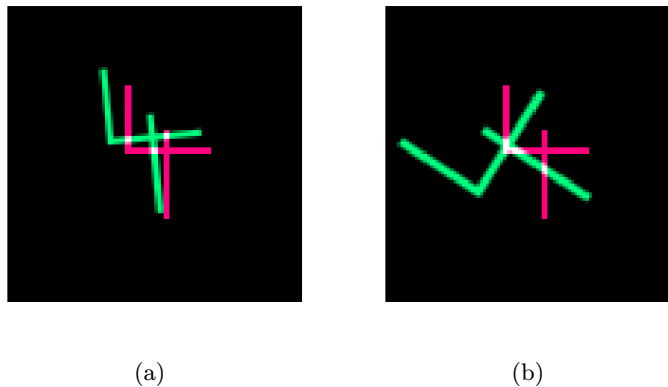
Přestože registrace za použití vzájemné informace nevykazovala příliš velké odchylky od požadovaných parametrů, pro translaci by bylo vhodnější registrovat obraz s mírou vzdálenosti v podobě součtu kvadrátů odchylek, nebo bychom, pokud by intenzity obou vstupních obrazů byly v lineárním vztahu, použili korelaci. V následujících odstavcích se ale budeme zabývat složitějšími transformacemi, kde je možné očekávat, že vzájemná informace povede naopak k přesnějším výsledkům.

3.2 Nerigidní transformace (translace + rotace + změna měřítka)

Na obrázku 3.2b je výsledek po zesílení nastavených parametrů tak, aby byla transformace výraznější.

Obrázek	Translace	Rotace (rad)	Násobení
3.2a	$\begin{pmatrix} 4 & 4 \end{pmatrix}$	0.1	0.9
3.2b	$\begin{pmatrix} 4 & 4 \end{pmatrix}$	1	0.7

Tabulka 3.5: Zvolené parametry transformace



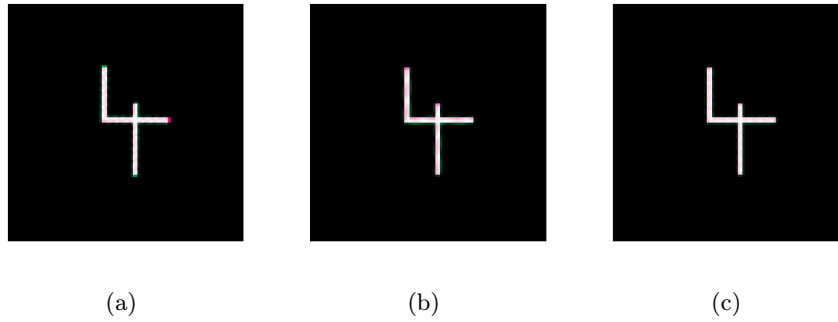
Obrázek 3.2: Stav před začátkem registrace

Součet kvadrátů odchylek

Optimalizátor	Míra vzdálenosti
Metoda line search	76176780
Pravidelný krok	6002767
Pravidelný krok + výraznější transformace	4959931

Tabulka 3.6: Výsledná míra vzdálenosti

Ačkoliv obrazy po registraci, které můžeme vidět na obrázku 3.3, vypadají podobně a velmi přesně, z výstupních dat v tabulce 3.6 je zřejmé, že klesání gradientu pomocí metody line search není tak přesné jako použití druhého optimalizátoru, jehož hodnota je o řád nižší. Připomeňme, že se snažíme míru vzdálenosti minimalizovat.



Obrázek 3.3: Optimalizátory zleva: klesání gradientu pomocí metody line search, klesání gradientu pomocí pravidelného kroku, klesání gradientu pomocí pravidelného kroku (výraznější transformace).

Korelace

Stejně jako v případě sumy kvadrátů odchylek se jako vhodnější optimalizátor jevila metoda klesání gradientu pomocí pravidelného kroku. Pro naprosto přesnou registraci bychom

Optimalizátor	Míra vzdálenosti
Metoda line search	-0.3522
Pravidelný krok	-0.9552
Pravidelný krok + výraznější transformace	-0.9779

Tabulka 3.7: Výsledná míra vzdálenosti

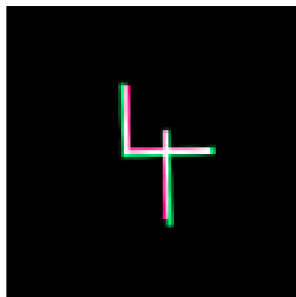
dostali míru vzdálenosti o hodnotě -1. První případ klesání gradientu (metoda line search) vedl k hodnotě, která se výrazně liší od požadované -1.

Vzájemná informace

Optimalizátor	Míra vzdálenosti
Metoda line search	-0.0949
Pravidelný krok	-0.0629
Pravidelný krok + výraznější transformace	-0.0077

Tabulka 3.8: Výsledek registrace při zvolení MMI jako míry vzdálenosti.

V případě použití MMI coby míry vzdálenosti se na rozdíl od předchozích výsledků ukázal pravidelný krok jako méně úspěšný typ metody klesání gradientu.

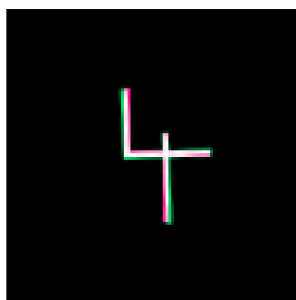


Obrázek 3.4: Výsledek registrace (MMI, pravidelný krok)

Optimalizátor	Míra vzdálenosti
Metoda line search	-0.0813
Pravidelný krok	-0.1358
Pravidelný krok + výraznější transformace	-0.4990

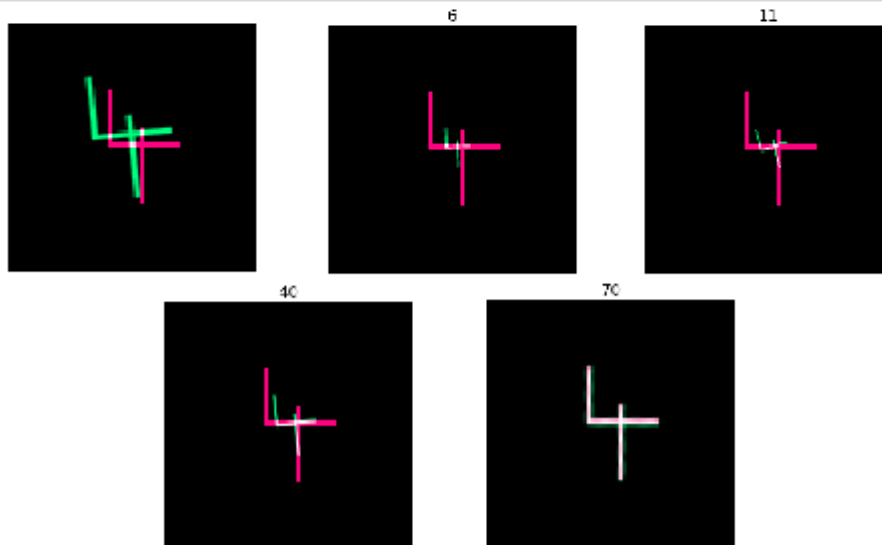
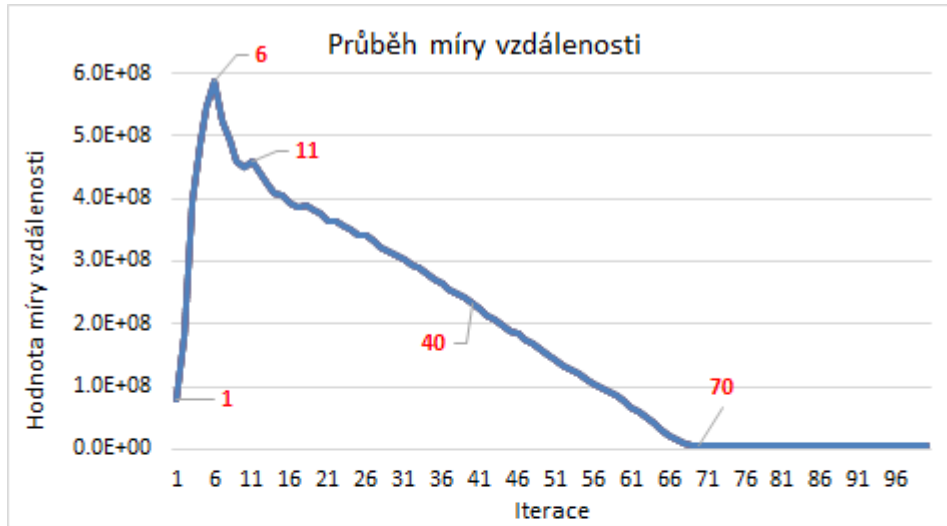
Tabulka 3.9: Výsledek registrace při zvolení JHMI jako míry vzdálenosti.

Klesání gradientu pomocí pravidelného kroku byl při zvolení JHMI coby míry vzdálenosti, stejně jako tomu bylo u korelace a součtu kvadrátů odchylek, lepším optimalizátorem, hledíme-li na její výslednou hodnotu v tabulce 3.9.



Obrázek 3.5: Výsledek registrace (JHMI, pravidelný krok)

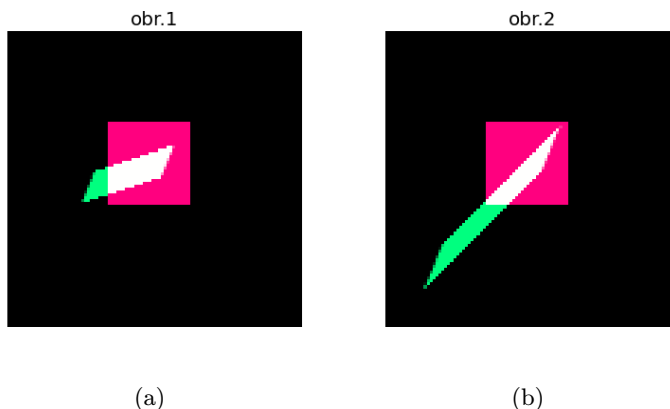
Výsledky získané z registrace za použití translace, rotace a změny měřítka ukazují, že optimalizační metoda klesání gradientu je pro registraci obrazu vhodná, poněvadž vedla k úspěšnému výsledku téměř ve všech případech. Chybné výsledky byly zaznamenány pouze u některých případů, kdy jsme používali metodu line search. Klesání gradientu pomocí pravidelného kroku se ukázalo jako spolehlivější a ve většině případů i přesnější řešení volby optimalizátoru. V následujících pokusech se tedy zaměřím na tento typ metody.



Obrázek 3.6: Průběh míry vzdálenosti jako sumy kvadrátů odchylek za použití optimalizátoru klesání gradientu pomocí pravidelného kroku.

3.3 Afinity transformace

Při afinní transformaci jsem nenastavoval jednotlivé parametry, ale matici A . Znovu máme dva obrazy, z nichž u jednoho (3.7b) je transformace výraznější.



Obrázek 3.7: Stav po transformacích

$$\text{Obraz 3.7a: } A = \begin{bmatrix} 1.2 & 0.5 \\ 1 & 3 \end{bmatrix}, \text{ obraz 3.7b: } A = \begin{bmatrix} 1.2 & 0.5 \\ 3 & 3 \end{bmatrix}.$$

Ještě před zobrazením výsledků je nutno poznamenat, že metodou klesání gradientu budeme nyní myslet pouze její případ pomocí pravidelného kroku, který se v předešlých testováních ukázal jako spolehlivější.

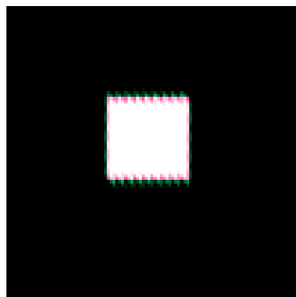
Součet kvadrátů odchylek

Optimalizátor	Míra vzdálenosti
LBFGB	190.99
Klesání gradientu	191.02
Klesání gradientu (obr. 3.7b)	278.83

Tabulka 3.10: Výsledná míra vzdálenosti

Oba optimalizátory vykazovaly podobnou přesnost. I přes zdánlivě vysokou míru vzdálenosti v tabulce 3.10 šlo o úspěšnou registraci obrazu, což ukáže i obrázek 3.8 coby výsledek registrace například pro klesání gradientu.

Na okrajích objektu na obrázku 3.8 jsou k vidění mírné odlišnosti, avšak ve srovnání s obrázkem na začátku registrace (obr. 3.7) lze považovat registrační proces za úspěšný. Nepřesnosti jsou způsobeny velkým zkosením obrázku při transformaci, což je faktor, který v tak velké míře pro budoucí registraci lékařských obrazových dat neočekáváme.



Obrázek 3.8: Výsledek registrace

Korelace

Optimalizátor	Míra vzdálenosti
LBFGBS	-0.96205
Klesání gradientu	-0.96201
Klesání gradientu (obr. 3.7b)	-0.96178

Tabulka 3.11: Výsledná míra vzdálenosti

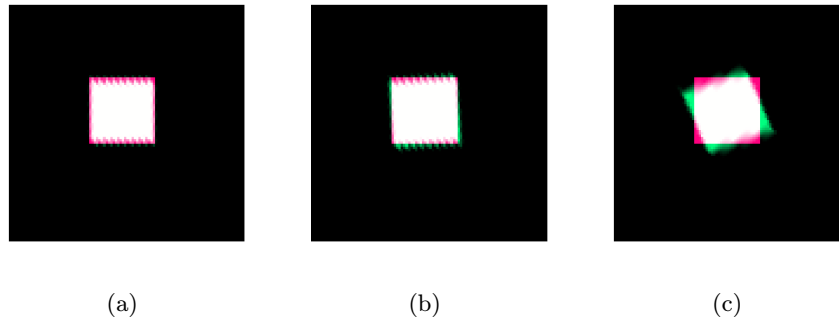
Z hodnot výsledného korelačního koeficientu v tabulce 3.11 lze usoudit, že použití korelace coby míry vzdálenosti vedlo ke kvalitní registraci, a to pro všechny optimalizační metody.

Vzájemná informace

Optimalizátor	MMI	A
LBFGBS	-0.2852	$\begin{bmatrix} 1.1542 & 0.4896 \\ 0.9145 & 2.7612 \end{bmatrix}$
Klesání gradientu	-0.2726	$\begin{bmatrix} 1.2925 & 0.7109 \\ 0.8935 & 2.9136 \end{bmatrix}$
Klesání gradientu (obr. 3.7b)	-0.2762	$\begin{bmatrix} 2.3819 & 1.7016 \\ 2.3917 & 2.6335 \end{bmatrix}$

Tabulka 3.12: Výsledek registrace za použití MMI jako míry vzdálenosti.

Míry vzdálenosti pro jednotlivé optimalizátory se v tabulce 3.12 od sebe příliš neliší, ale budeme-li hledět na výslednou matici A , zjistíme, že se pro různé optimalizátory, na rozdíl od míry vzdálenosti, od sebe některé prvky značně liší, což se projeví i ve výsledném obrázku po skončení registrace. Zleva na obrázku 3.9 vidíme optimalizátory LBFGSB, klesání gradientu pro obraz 3.7a a pro obraz 3.7b.



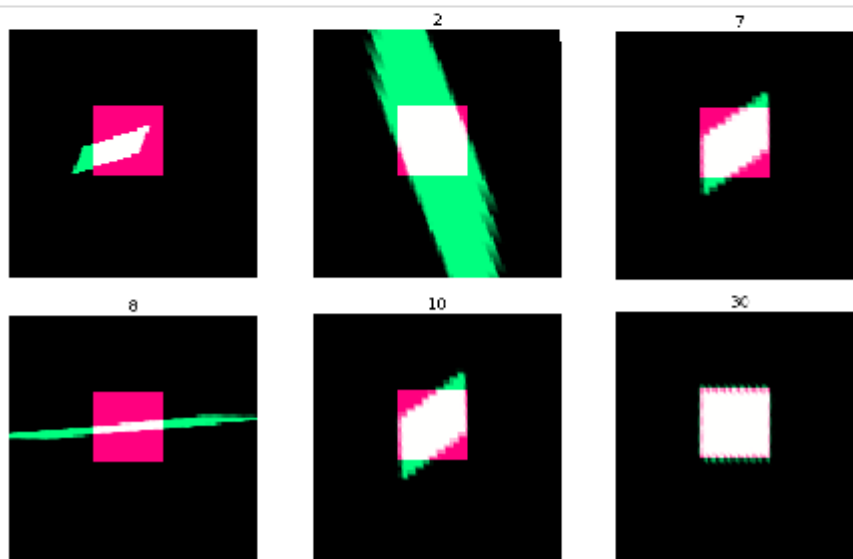
Obrázek 3.9: Výsledky registrace

Zde na obrázku 3.9 je možné lépe zhodnotit výsledek registrace. Pro mírněji nastavené parametry transformace je vzájemná informace MMI stejně vhodná jako například součet kvadrátů odchylek s tím, že optimalizátor LBFGSB způsobí, že výsledný objekt nepokryje některé pixely (fialová barva), ve kterých by se podle vztažného obrázku nacházet měl. Klesání gradientu tuto chybu zmírní, v některých místech (zelená barva) však bude objekt přesahovat. Obrázek 3.9c ukazuje, že pro větší parametry matice A je vzájemná informace nevhodná.

Optimalizátor	JHMI	A
LBFGSB	-0.4717	$\begin{bmatrix} 0.5029 & 0.2107 \\ -0.1296 & 0.9999 \end{bmatrix}$
Klesání gradientu	-0.2480	$\begin{bmatrix} 0.9813 & 0.0069 \\ -0.0029 & 1.0001 \end{bmatrix}$
Klesání gradientu (obr. 3.7b)	-4.6979	$\begin{bmatrix} 0.2874 & 0.0540 \\ -0.0667 & 0.2718 \end{bmatrix}$

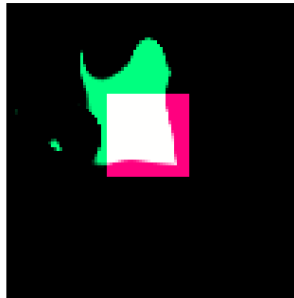
Tabulka 3.13: Výsledek registrace za použití JHMI jako míry vzdálenosti.

K rozhodnutí, zda je míra vzdálenosti JHMI vhodná, nám rovněž pomůže podívat se na výslednou matici A v tabulce 3.13, kde se ve všech případech velmi liší od požadované matice. Tento typ vzájemné informace je tedy pro afinní transformaci nevhodný.



Obrázek 3.10: Průběh míry vzdálenosti jako sumy kvadrátů odchylek za použití optimalizátoru LBFGSB.

3.4 Transformace B-spline funkcemi



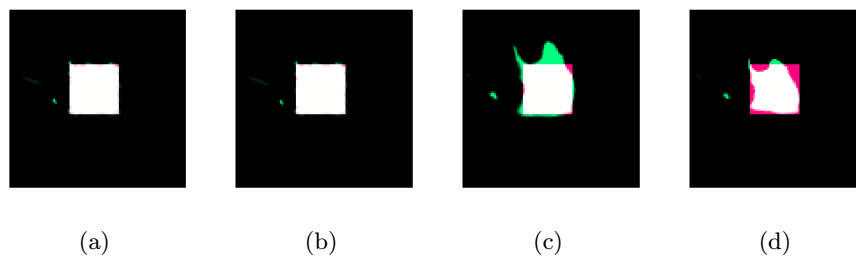
Obrázek 3.11: Stav před začátkem registrace

Jako optimalizační metoda byla využita metoda klesání gradientu.

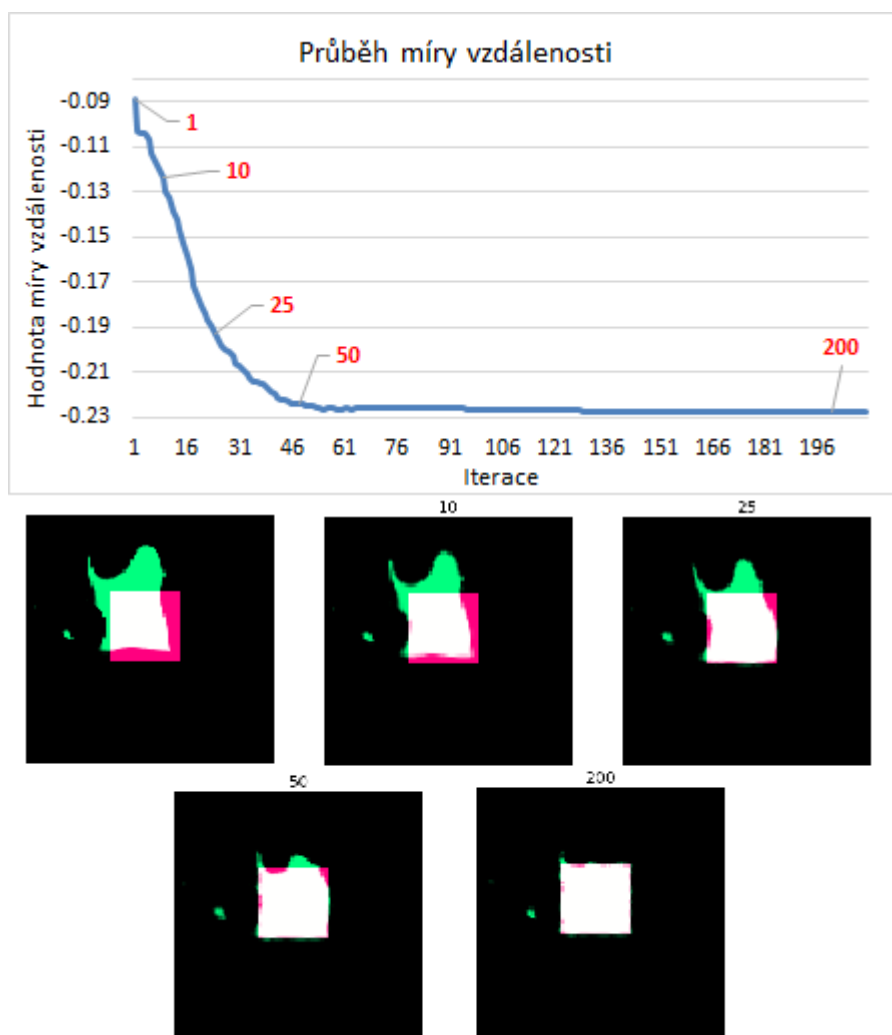
Typ míry vzdálenosti	Hodnota míry vzdálenosti
Součet kvadrátů odchylek	49.7431
Korelace	-0.9906
MMI	-0.2275
JHMI	-0.3074

Tabulka 3.14: Výsledky pro jednotlivé míry vzdálenosti

Korelace a součet kvadrátů odchylek vedly k přesnější registraci než vzájemná informace, což je vidět na obrázku 3.12, kde jsou znázorněny výsledky pro jednotlivé míry vzdálenosti.



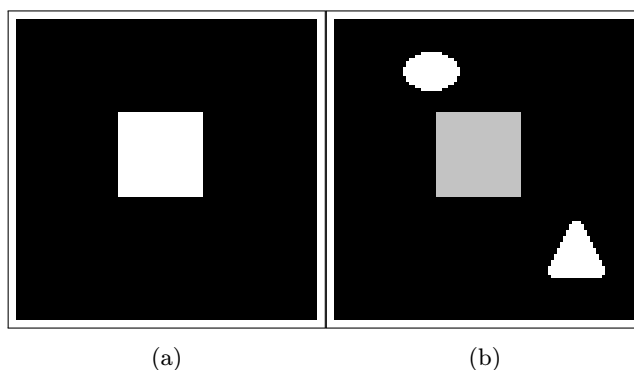
Obrázek 3.12: Zleva: součet kvadrátů odchylek, korelace, MMI, JHMI.



Obrázek 3.13: Detailní průběh míry vzdálenosti jako součtu kvadrátů odchylek.

3.5 Dva rozdílné snímky

Až doposud jsme pracovali s obrazy, které znázorňovaly jeden bílý objekt a černé pozadí, a ty jsme dále transformovali. Na vztažném i registrovaném obrázku se nacházel jen jeden objekt, navíc se stejnou intenzitou. V praxi je ovšem nutné počítat s odlišným poměrem mezi intenzitami objektů a pozadí pro oba obrazy. Také se může stát, že na jednom obrázku se vyskytne objekt, který se ve druhém neobjevil. Dalším úkolem je tedy ověřit průběh registrace pro případ, kdy se budou nacházet v registrovaném obrázku objekty, které se ve vztažném nevyskytují. Rovněž změníme poměr intenzity objektu, který budeme transformovat, k pozadí.



Obrázek 3.14: Dvojice vstupních obrazů

Všimněme si, že čtverec na obrázku 3.14b má šedou barvu a přidané objekty, které se ve vztažném obrázku 3.14a nenacházejí, mají stejnou barvu jako objekt (čtverec) ve vztažném obrázku, tedy bílou. Obrázek 3.14b se ještě před spuštěním registrace transformuje a označí jako registrovaný.

Transformace

Následující testování budeme provádět za použití geometrických transformací, se kterými se pravděpodobně setkáme u registrace lékařských obrazových dat. Jsou jimi nerigidní transformace za využití translace, rotace a změny měřítka a transformace B-spline křivkami. Afinní transformace, která zahrnuje i zkosení obrázku, se u lékařských snímků jeví málo pravděpodobná.

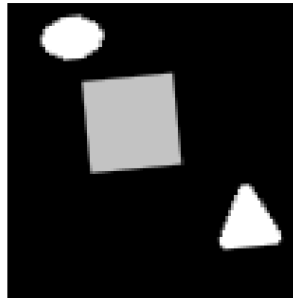
Míra vzdálenosti

Z předešlých testování je zřejmé, že na výsledek registrace pro jednotlivou transformaci má největší vliv výběr míry vzdálenosti, méně důležitý je pak výběr optimalizační metody. Za tu zvolíme metodu klesání gradientu s využitím pravidelného kroku, což se ukázalo jako nejspolehlivější řešení ve většině předchozích testování. Důležitým úkolem je najít míru vzdálenosti, která se pro danou transformaci ukáže jako nejvhodnější, proto je nezbytné testování provést pro všechny čtyři již známé typy měř.

3.5.1 Nerigidní transformace (translace, rotace, změna měřítka)

Translace	Rotace (rad)	Násobení
$\begin{pmatrix} 4 & 4 \end{pmatrix}$	0.1	0.9

Tabulka 3.15: Zvolené parametry



Obrázek 3.15: Registrovaný obrázek

Typ míry vzdálenosti	Hodnota míry vzdálenosti
Součet kvadrátů odchylek	246881043.5770
Korelace	-0.4767
MMI	-0.1632
JHMI	-0.2930

Tabulka 3.16: Výsledek registrace

V tabulce 3.16 je na první pohled vidět, že míry vzdálenosti nabývají odlišných hodnot, než tomu bylo v ideálních případech v předchozím testování. Tento problém způsobují právě přidané objekty do registrovaného obrazu a není možné transformovat obraz tak, aby naprosto přesně odpovídal vztažnému obrázku. Odpovídat si budou pouze objekty, které mají oba vstupní obrazy společné, v tomto případě tedy čtverec uprostřed obou snímků. Z hodnot měř vzdálenosti tedy nebude možné na první pohled určit, jak úspěšná registrace byla provedena.

Parametry transformace získané registrací

Před zkoumáním správnosti výsledků je nutné si opět uvědomit, že obraz označený jako registrovaný vznikl transformací vztažného obrázku podle nastavených parametrů. Ty ovšem nebudou odpovídat parametrům získaným, neboť úkolem je transformovat registrovaný obraz tak, aby jeho objekty odpovídaly objektům ve vztažném obrázku. Hledáme tedy zpětnou transformaci.

Určení požadovaných parametrů:

$$s' = s^{-1} = 1.1\bar{1}$$

$$t' = -t = (-4 \ -4)$$

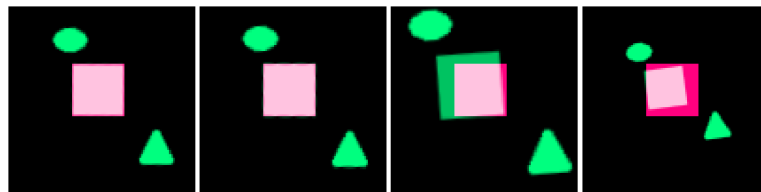
$$R' = -R = -0.1,$$

kde s , t , R jsou nastavené parametry transformace.

Typ míry vzdálenosti	Translace	Rotace (rad)	Násobení
Součet kvadrátů odchylek	$\begin{pmatrix} -5.6035 & -4.1956 \end{pmatrix}$	-0.1041	1.1489
Korelace	$\begin{pmatrix} -5.7056 & -4.2755 \end{pmatrix}$	-0.1014	1.1129
MMI	$\begin{pmatrix} -0.5979 & -2.0450 \end{pmatrix}$	-0.0273	0.9148
JHMI	$\begin{pmatrix} -0.5699 & -2.0249 \end{pmatrix}$	0.0352	1.4923

Tabulka 3.17: Získané parametry

Vzájemná informace jako míra vzdálenosti není tedy pro tento typ nerigidní transformace dobrou volbou, jelikož hodnoty jednotlivých parametrů v tabulce 3.17 se výrazně liší od požadovaných, což ukáže i následující grafické znázornění.

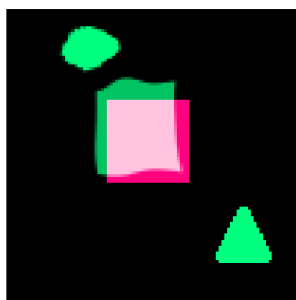


Obrázek 3.16: Zleva: suma kvadrátů odchylek, korelace a vzájemné informace.

Prohlédneme-li si obrázek 3.16, zjistíme, že ani v případě úspěšné registrace nenajdeme ve výsledném obrazu bílé pixely, které znázorňují shodu. To je způsobeno odlišností intenzit objektů pro vztažný a registrovaný obrázek. Je však patrné, že fialová barva, která znázorňuje objekt ve vztažném obrázku, je v místech překrytí objektů světlejší. Světlost fialové barvy je úměrná rozdílu intenzit pro odpovídající objekty.

3.5.2 B-spline transformace

Oproti předchozím případům B-spline trochu zmírníme parametry transformace.

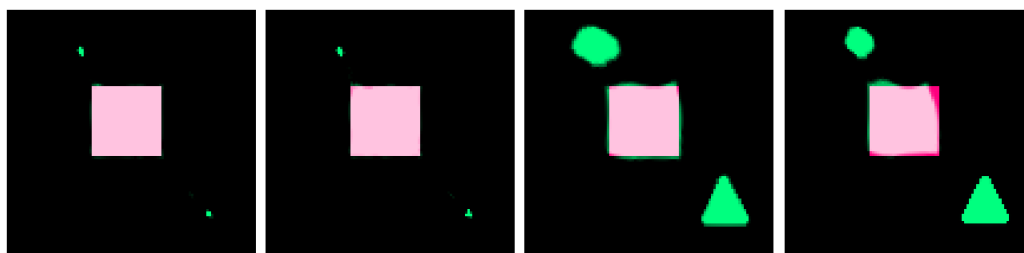


Obrázek 3.17: Stav na začátku registrace

Typ míry vzdálenosti	Hodnota míry vzdálenosti
Součet kvadrátů odchylek	29326297.7625
Korelace	-0.9695
MMI	-0.2674
JHMI	-0.2856

Tabulka 3.18: Výsledek registrace

Korelační koeficient se v tomto případě blíží ideální hodnotě -1 (viz tabulka 3.18), což v případě předešlé transformace neplatilo. Odlišnost od míry vzdálenosti způsobují objekty, které se nepřekrývají. V těchto místech si obrazy neodpovídají. Je ale známo, že míra vzdálenosti nemůže být ideální, pokud se vyžaduje, aby se tyto přídavné objekty zobrazily ve výsledném obraze. Malá odchylka korelace od hodnoty -1 naznačuje, že se ve výsledném obraze budou přidat objekty, tedy elipsa a trojúhelník, vyskytovat jen v nepatrné míře, což je možné ověřit výsledným obrázkem 3.18.



Obrázek 3.18: Zleva: suma kvadrátů odchylek, korelace a vzájemné informace.

Z grafického znázornění na obrázku 3.18 je čitelné, že vzájemná informace je vhodnější než součet kvadrátů odchylek nebo korelace za předpokladu, že jako geometrická transformace bude použita metoda B-spline funkcí. Právě B-spline transformaci je vhodné použít v případech, kde očekáváme transformace elastického charakteru, což lze předpokládat u CT snímků, jejichž registraci se budu věnovat v kapitole 4.

Kapitola 4

Testování 3D registrace

V následujících kapitolách se budeme věnovat 3D registraci CT snímků, tedy lékařských obrazových dat, kde se zaměříme především na oblast jater. Jelikož oba CT snímky, které budou vstupovat do registrace, byly získány během dvou odlišných fází, můžeme předpokládat, že v době mezi pořízením obou snímků došlo v těle pacienta ke změnám. V případě měkkých tkání se bude jednat převážně o geometrické transformace elastického charakteru. Těm bude z mnoha prozkoumaných transformací nejlépe odpovídat transformace pomocí B-spline křivek.

4.1 Výběr parametrů 3D registrace

Předešlé kapitoly se týkaly především volby nejvhodnější míry vzdálenosti a optimalizační metody. Jako nejvhodnější optimalizátor se ukázala metoda klesání gradientu pomocí pravidelného kroku, která se snaží minimalizovat vzájemnou informaci, jež byla posouzena jako nejvhodnější míra vzdálenosti pro registraci za použití B-spline transformace (viz kapitola 3.5.2). Po určení těchto prvků se již můžeme zabývat jejich parametry, které mají zásadní vliv na kvalitu registrační metody. Podle SimpleITK [2] mezi ně patří:

- parametry B-spline transformace - velikost mřížky,
- parametry optimalizátoru - konstanta učení, maximální počet iterací.

Mřížka

Velikost mřížky je zřejmě nejdůležitějším parametrem B-spline transformace. Především má vliv na počet hledaných parametrů transformace, a tím pádem i na dobu trvání jedné iterace registračního procesu. V knihovně ITK je B-spline transformace, jak je uvedeno v dokumentaci [3], reprezentována pomocí deformovatelné transformace bodů z jednoho N -dimenzionálního prostoru do druhého. Deformace je dána sítí kontrolních bodů $\vec{\lambda}_j$ a mění se jejich deformací $\vec{g}(\vec{\lambda}_j)$.

Pro následující odstavce označme velikost mřížky pro jednu konkrétní dimenzi velkým M . Poté bude platit:

$$M = n - m, \tag{4.1}$$

kde m je řád spline a n je počet kontrolních bodů v dané dimenzi. V případě 3D transformace bude velikost mřížky reprezentovat tříprvkový vektor $[M_x, M_y, M_z]$. V našem případě budeme volit velikost mřížky pro všechny dimenze stejnou, pro zjednodušení tedy budeme používat pouze velké M , pro které bude platit: $M = M_x = M_y = M_z$.

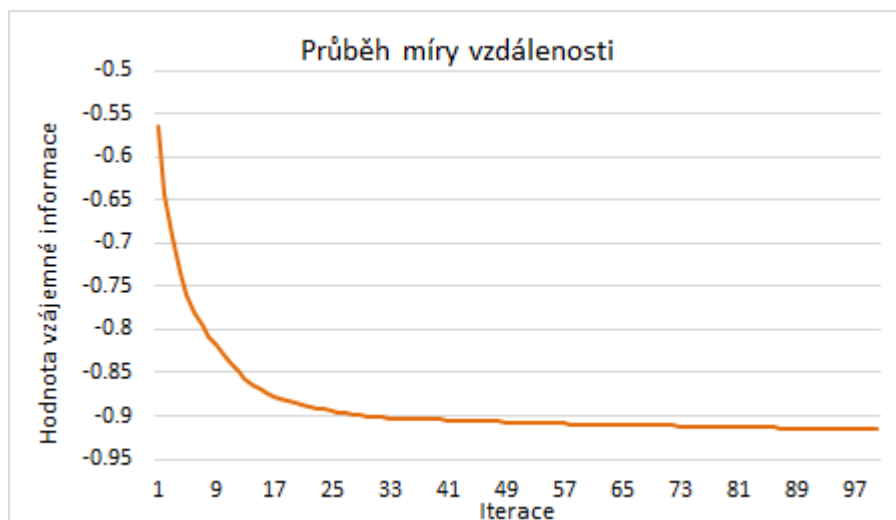
Parametry optimalizátoru

V případě optimalizační metody klesání gradientu pomocí pravidelného kroku budeme nastavovat zejména konstantu učení a maximální počet iterací, který slouží pro omezení doby běhu procesu registrace, viz [2]. Dalším způsobem zastavení registrace je nastavení parametru minimálního kroku, který přeruší výpočet ve chvíli, kdy změna míry vzdálenosti oproti předešlé iteraci bude nižší než stanovený práh. Tento způsob jsem se rozhodl neaplikovat, jelikož v některých případech míra vzdálenosti v prvních několika iteracích kmitá a může se stát, že její změna bude v těchto místech dostatečně malá k tomu, aby byla registrace zastavena.

Kmitání míry vzdálenosti je z velké části způsobeno zvolenou hodnotou konstanty učení, která se používá pro přepočítání parametrů transformace v každé iteraci a udává, jak moc se tyto parametry změní. S parametry transformace se v jednotlivých iteracích mění i míra vzdálenosti, která je hlavním ukazatelem kvality registrace, z čehož plyne, že i míra učení má na výslednou kvalitu registrace značný vliv. Pro míru učení zavedeme zkratku LR (z anglického learning rate).

4.1.1 Testování

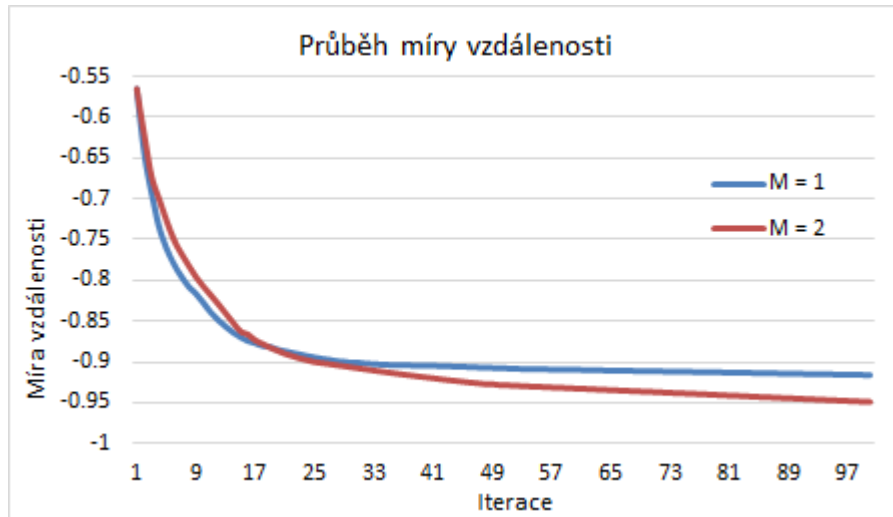
Pro určení kvality registrace budeme sledovat průběh míry vzdálenosti. Jelikož jako míra vzdálenosti je nastavena negativní vzájemná informace, kterou se optimalizátor snaží minimalizovat, bude se kvalitnější registrace vyznačovat tím, že její výsledná míra vzdálenosti bude nižší. Pro začátek nastavíme konstantu učení i velikost mřížky na hodnotu 1.



Obrázek 4.1: Graf průběhu vzájemné informace

Výsledná míra vzdálenosti má po 100 iteracích hodnotu -0.91607 . I po 100 iteracích měla vzájemná informace tendenci klesat, jak ukazuje obrázek 4.1.

Nyní zkusíme zvýšit velikost mřížky na hodnotu 2. Konstanta učení zatím zůstane beze změny na hodnotě 1.



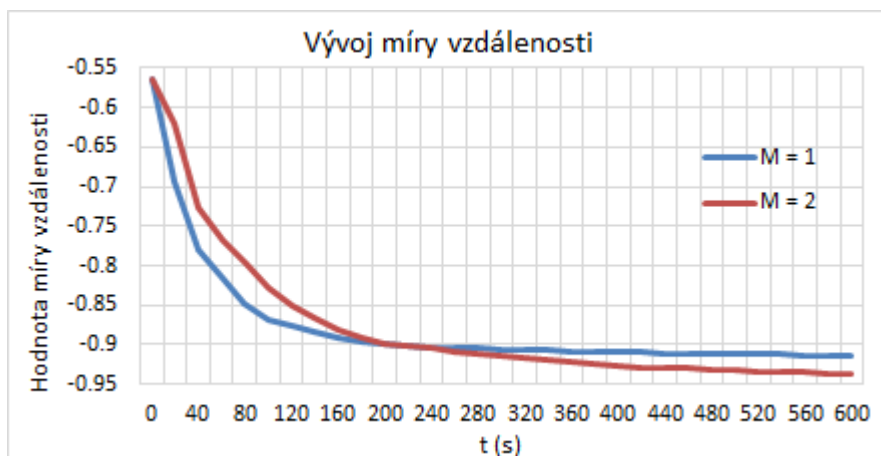
Obrázek 4.2: Graf průběhu vzájemné informace

Zvýšení velikosti mřížky vedlo ke snížení hodnoty míry vzdálenosti. V prvních iteracích klesala míra vzdálenosti lépe pro $M = 1$, ale již od 20. iterace dosahovala registrace se zvýšenou velikostí mřížky M nižších hodnot vzájemné informace. Důkazem, že velikost mřížky má vliv na dobu trvání jedné iterace, je výsledná doba trvání registračního procesu pro 100 iterací, jejíž hodnoty jsou zaznamenány v tabulce 4.1.

Velikost mřížky M	Doba trvání (s)
1	675
2	835

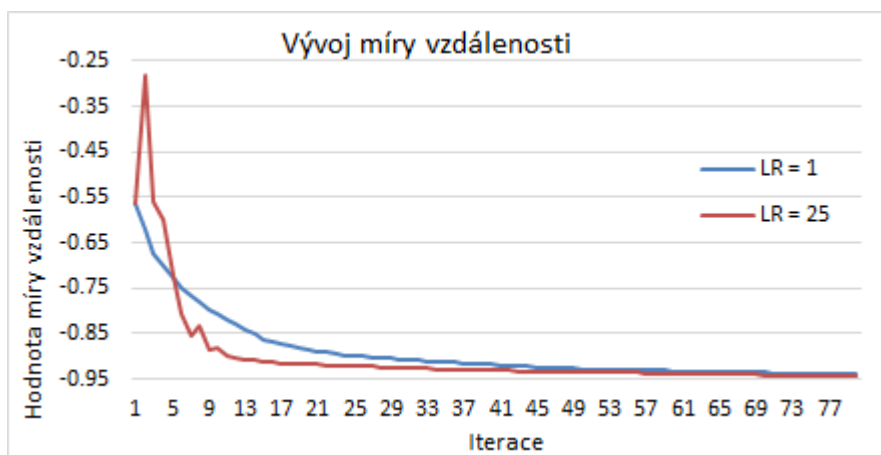
Tabulka 4.1: Výsledná doba trvání

Vzhledem k tomu, že se bude registrace testovat pro různé velikosti mřížky, byla by doba trvání velice proměnlivá, pokud bychom volili stejný počet iterací pro každý pokus. Jelikož cílem je vytvořit program, který bude použitelný v praxi, bude účelnější sledovat vývoj míry vzdálenosti v čase (viz obrázek 4.3), nikoliv v závislosti na počtu iterací, jak tomu bylo doposud.



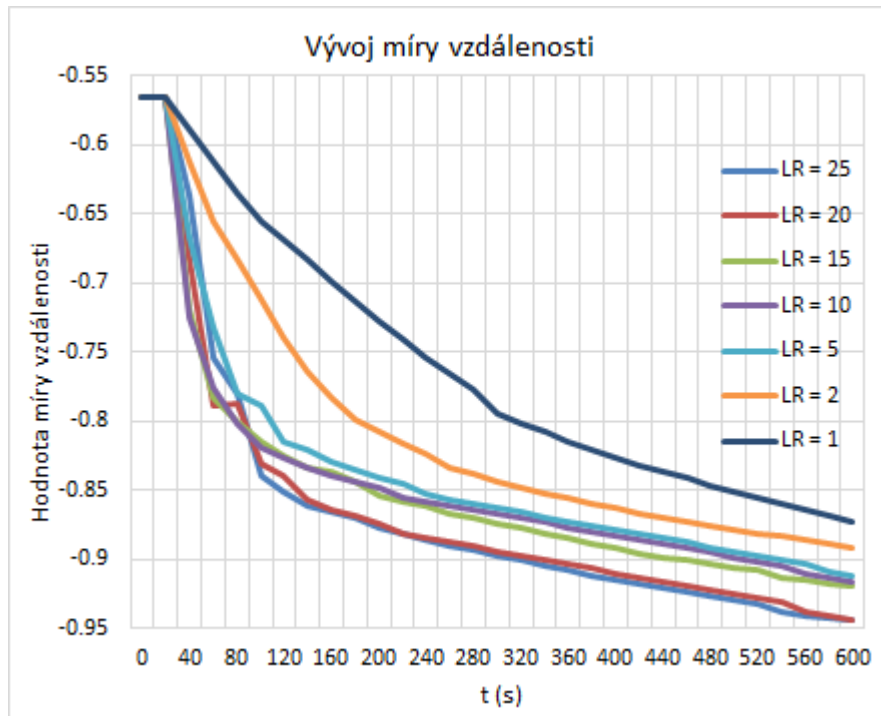
Obrázek 4.3: Graf průběhu vzájemné informace v čase

Nyní zvýšíme také konstantu učení, a to na 25. Průběh vzájemné informace pro 80 iterací vidíme na obrázku 4.4.



Obrázek 4.4: Graf průběhu vzájemné informace pro $M = 2$

Vyšší konstanta učení $LR = 25$ vedla ke zlepšení výsledné míry vzdálenosti. Z grafu na obrázku 4.4 lze však vyčíst, že vykazovala během prvních iterací značné kmitání. Bylo by žádoucí, aby program zvládl provést takový počet iterací, po kterých by tyto oscilace stihly ustát. Pro $M \geq 6$ již může vzhledem k vysoké časové náročnosti každé iterace docházet ke kmitům i po několika minutách. Testované velikosti mřížky tedy budou v rozmezí 2 až 5.

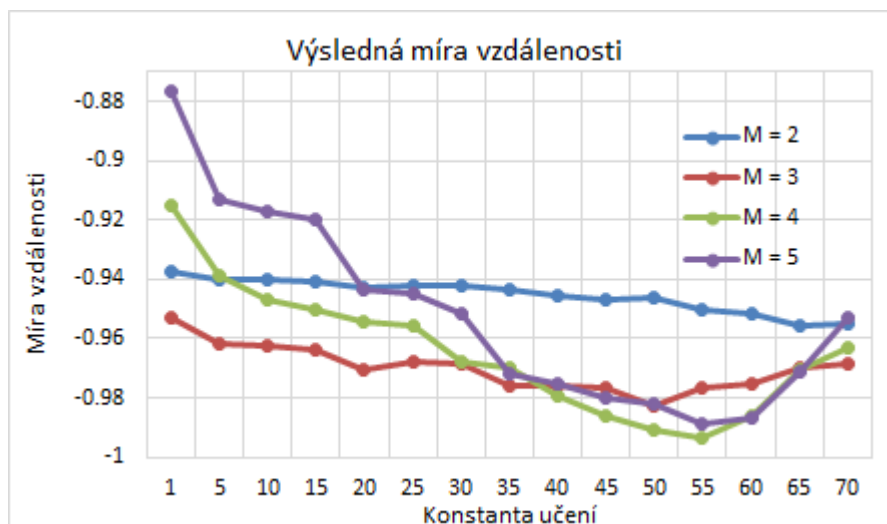


Obrázek 4.5: Graf průběhu vzájemné informace v čase pro $M = 5$ a různé míry učení LR .

Graf na obrázku 4.5 napovídá, že vyšší hodnota míry učení vede ke snížení výsledné míry vzdálenosti. Jedním z úkolů bude zjistit optimální konstantu učení, po jejímž zvýšení už by nedošlo ke zlepšení kvality registrace, ale naopak ke zhoršení.

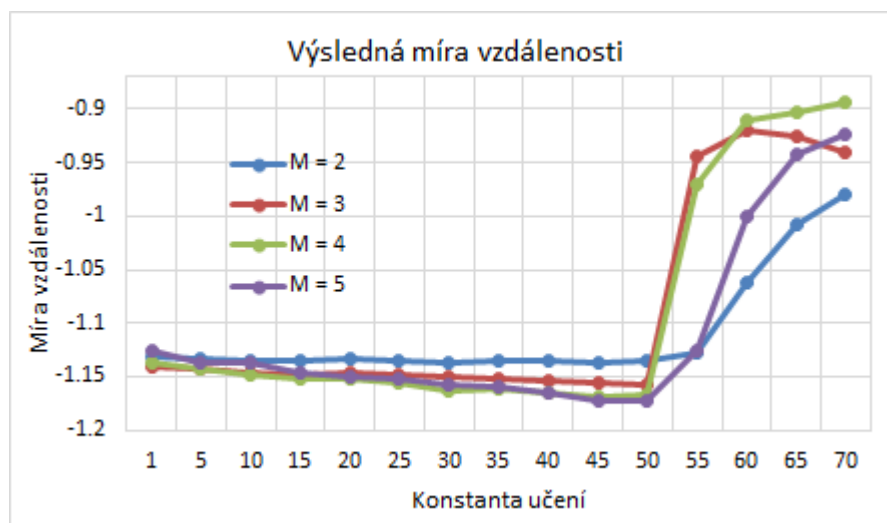
Rovněž si nastavíme dobu trvání registrace, a to na 10 minut. Po uplynutí 600 sekund bude registrační proces zastaven a výsledná hodnota vzájemné informace zaznamenána. Za tuto dobu by registrace o velikosti mřížky 6 stihla provést pouze 16 iterací, po kterých by ještě míra vzdálenosti nemusela být ustálená, což by mohlo mít značný dopad na výsledek registrace. Omezení na testování velikosti mřížky na hodnoty 2 až 5 je tedy na místě.

Obrázek 4.6 znázorňuje, jakých hodnot dosáhla vzájemná informace po 600 sekundách v závislosti na volbě konstanty učení. Při nižších konstantách učení dosahovala lepších výsledků mřížka o nižších velikostech 2 a 3. Se zvyšující se konstantou učení však hodnota vzájemné informace pro tyto zvolené rozměry mřížky neklesala tak silně, jako tomu bylo u velikostí 4 a 5, které již pro míry učení $LR \geq 45$ vykazovaly nejlepší výsledky. Spolu se zvolenou konstantou učení $LR \geq 30$ evidentně nebude vhodné nastavit velikost mřížky M na hodnotu 2, která nebude reagovat na zvyšující se konstantu učení takovým zlepšením výsledné míry vzdálenosti jako rozměry $M \geq 3$. Obecně lze konstatovat, že větší rozměry mřížky povedou k výraznějšímu klesání vzájemné informace v závislosti na zvyšující se míře učení. Zvyšování kvality registrace u vyšších hodnot konstanty učení má však omezení. V našem případě, jak je možné se přesvědčit z obrázku 4.6, je tímto omezením velikost míry učení $LR = 55$. Další zvyšování LR již vede naopak ke zvýšení míry vzdálenosti, a to znovu nejvýrazněji u vysokých hodnot M .



Obrázek 4.6: Hodnota vzájemné informace po 600 sekundách

Možným řešením by bylo zvolit rozměry mřížky a nastavit konstantu učení na takovou hodnotu, při které nabývá příslušná křivka na grafu minima, tedy například $M = 4$ a $LR = 55$. Provedme stejné testování pro jinou dvojici CT snímků.

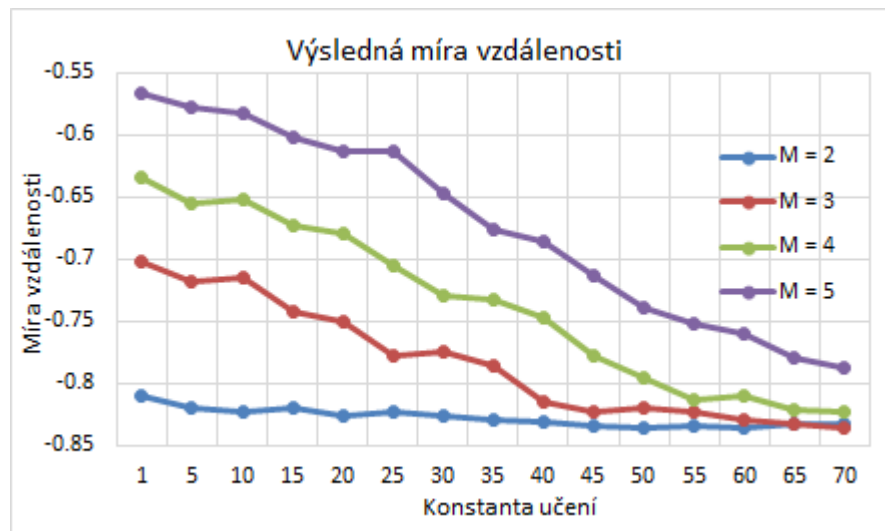


Obrázek 4.7: Hodnota vzájemné informace po 600 sekundách

Obrázek 4.7 ukazuje, že kdybychom zvolili pro registraci těchto dat dříve zvolené parametry $M = 4$ a $LR = 55$, dostali bychom jednoznačně nejhorší výsledky v porovnání se všemi nižšími hodnotami konstanty učení, a to pro jakoukoliv velikost mřížky. Z grafu je patrné, že dostatečně vysoká konstanta učení by měla fatální následky na kvalitu registrace. Nepomůže ani zvolení co nejmenších rozměrů mřížky, kde bychom předpokládali slabší reakci na zvýšení konstanty učení. Úkolem je tedy nastavit optimální míru učení, u které by k takovému propadu nedošlo. Ze všech testovaných dvojic CT snímků již nebyla nalezena data, pro která by se kvalita registrace začala zhoršovat už pro nižší hod-

notu konstanty než 50, která byla zpozorována v tomto případě. Berme tedy hodnotu 50 jako horní mez intervalu $\langle 30, 50 \rangle$, který označíme jako nejvhodnější volbu konstanty učení pro registraci CT snímků.

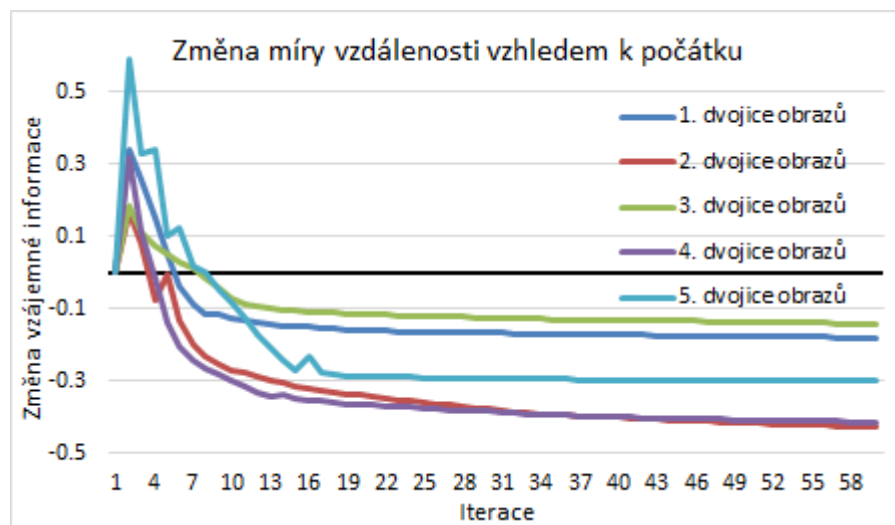
Zbývá určit optimální velikost mřížky M . Tu jsme volili z množiny $\{2, 3, 4, 5\}$. Hodnota $M = 1$ se během pozorování ukázala jako nejhorší varianta pro kvalitu registrace téměř ve všech případech a do výběru optimální hodnoty parametru nebyla zařazena. Pro nízkou konstantu učení by bylo nejvhodnější zvolit hodnotu $M = 2$, která však s rostoucí mírou učení nevedla ke klesání vzájemné informace tak rapidně jako hodnoty $M \geq 3$. Tento nedostatek se projevil především v intervalu míry učení $\langle 30, 50 \rangle$, který byl zhodnocen jako optimální. Fakt, že pro vyšší hodnoty M dochází k rychlejšímu poklesu míry vzdálenosti v grafu, má nevýhodu v tom, že po dosažení horní meze optimálního intervalu míry učení dojde k o to rychlejšímu růstu. Značnou roli hraje rovněž i kvalita registrace na začátku grafu, tedy pro konstantu učení $LR = 1$, která může být pro nižší M o tolik lepší, že vyšší M se jí vyrovnají až při hodnotách, které se blíží výše zvolené horní mezi 50, nebo ji dokonce přesahují. Po zvolení konstanty učení z optimálního intervalu by mohlo dojít k situaci, kdy by se stále ještě ukázala jako nejvhodnější velikost mřížky o hodnotách 2 nebo 3. Graf na obrázku 4.8 ukazuje výsledek testování 3D registrace u jedné z dvojic CT snímků.



Obrázek 4.8: Hodnota vzájemné informace po 600 sekundách

Na obrázku 4.8 si všimněme obrovského rozdílu v hodnotě vzájemné informace na začátku grafu, který má za následek, že i přes rychlý pokles hodnot míry vzdálenosti pro mřížku velkých rozměrů bude pro celý interval hodnoty míry učení $\langle 30, 50 \rangle$ dosahovat nejlepších výsledků velikost mřížky $M = 2$. Nutno poznamenat, že z celého testování se jedná o ojedinělý případ, kdy by vzájemná informace začala stoupat až při nezvykle vysokých hodnotách, které jsou naprosto mimo náš zvolený interval, na kterém by volba $M \in \{4, 5\}$ znamenala fatální zhoršení celé registrace oproti nižším hodnotám. Kompromisem bude zvolit jako nejvhodnější mřížku o velikosti 3, která, jak si opět můžeme všimnout v gra-

fech na obrázcích 4.6, 4.7 a 4.8, dosahuje přijatelných kvalit při nízkých konstantách učení a zároveň zvládá průměrně rychlým růstem kvality registrace sekundovat vyšším hodnotám i u konstant učení z definovaného optimálního intervalu. Z důvodu bezpečnosti jako optimální konstantu učení ne zvolíme horní mez optimálního intervalu, která je stanovena na $LR = 50$, ale o něco nižší hodnotu, a to $LR = 45$. Výsledkem testování 3D registrace CT snímků tedy bude stanovení jejich parametrů na $LR = 45$ a $M = 3$. Pro toto nastavení si můžeme znázornit průběh míry vzdálenosti za použití různých dvojic vstupních obrazů. Jelikož se jednotlivé testované dvojice CT snímků od sebe liší, míra vzdálenosti dosahuje na začátku i na konci registrace zcela odlišných hodnot. Zajímavé bude si ukázat vývoj změny vzájemné informace vzhledem k její hodnotě v počátku (obr. 4.9). Na začátku grafu tedy bude změna rovna 0 a po 60 iteracích bude dosahovat záporných hodnot, pokud registrace proběhne v pořádku.



Obrázek 4.9: Průběh změny hodnoty míry vzdálenosti po iteracích pro různé dvojice vstupních dat.

Poznamenejme si, že $\Delta MI = MI(i) - MI(0)$, kde $MI(i)$ je hodnota vzájemné informace v i -té iteraci a $MI(0)$ je její hodnota na začátku registračního procesu.

4.2 Další způsoby registrace CT snímků

Nabízejí se dva základní způsoby registrace CT snímků. Prvním z nich je plná 3D registrace, již jsme se věnovali v kapitole 4.1. Druhým způsobem je 2D registrace pro každý řez, kde do registračního procesu vstupují 2D obrazy, které představují jednotlivé řezy konkrétních CT snímků.

3D registrace

Knihovna SimpleITK [2] umožňuje registrovat 3D obrazy stejným způsobem, jako tomu bylo u 2D registrace s tím rozdílem, že registrace hledá i parametry transformace pro osu z . Počet hledaných parametrů se zvýší v případě některých typů geometrické transformace i mnohonásobně. U B-spline transformace, kterou reprezentují stovky parametrů, lze předpokládat výrazné zvýšení časové náročnosti algoritmu.

2D registrace po řezech

Průběh této registrační metody spočívá ve výběru indexu konkrétního řezu a vytažení jemu odpovídajícím 2D snímkům pro vztažený a registrovaný obraz. Hlavní výhodou této metody spočívá v tom, že pro nastavení parametrů, které by vedlo k pomalejší 3D registraci, dosahuje metoda lepších výsledků, co se týče doby trvání jedné iterace. Registrace je navíc jednodušší v tom smyslu, že se snaží najít geometrickou transformaci o menším počtu parametrů než 3D registrace. Pokud je hledaná transformace popsána právě B-spline funkcemi, jak tomu v našem případě je, a je navíc 3D, tak může počet hledaných parametrů dosáhnout počtu několika set až tisíců. Registrace 2D snímků po řezech počet parametrů několikanásobně sníží díky absenci třetí dimenze, a právě proto se tato registrace stává jednodušší.

Je nutné si ovšem uvědomit, že 2D registrace se musí provádět pro každý řez, kterých v CT snímcích zpravidla bývají desítky. Snížení doby potřebné pro jednu iteraci musí být tedy natolik výrazné, aby i přes nutnost opakovat proces pro všechny řezy vedlo ke snížení celkové doby registrace.

Zásadní nevýhodou, která má navíc vliv i na kvalitu registrace, je fakt, že indexy řezů, které reprezentují stejnou úroveň v ose z , se v mnoha případech neshodují. Například chceme-li manuálně vybrat konkrétní řez, získáme ho výběrem 2D obrazu, který je umístěn na indexu i 3D snímku. Jako druhý snímek vstupující do registrace vybereme z příslušného 3D obrazu ten řez, který vizuálně, ve většině případů i mírou vzdálenosti, nejlépe odpovídá 2D obrazu určenému v předchozím kroku. Vybraný řez se bude nacházet na indexu j druhého 3D snímku, přičemž $i \neq j$. To je způsobeno právě geometrickou transformací v ose z . Pro jednoduchost si představme, že transformace v ose z představuje pouze translaci o velikosti t_z . Poté bude platit $j = i + t_z$. Pro registraci odpovídajících snímků lze využít dvě možnosti.

První z nich je vybírání dvou podobných snímků. Pro každý řez vztažného 3D snímku se vybere takový řez z druhého 3D registrovaného obrazu, který mu je nejvíce podobný. To

by však vedlo k nutnosti spočítat míru vzdálenosti mezi vztažným řezem a všemi ostatními registrovanými řezy. Mohl by navíc nastat případ, kdy by se řezy periodicky opakovaly a nalezený snímek by se nacházel o několik úrovní jinde. Rovněž hrozí nebezpečí, že pro více různých vztažných řezů by byl nalezen tentýž registrovaný.

Druhou variantou je nalezení geometrické transformace, podle které bude možné upravit jeden z obrazů tak, aby si jednotlivé řezy odpovídaly. Metoda spočívá v nalezení jejich parametrů a následné transformaci především po ose z . Otázkou zůstává, jak nalézt parametry geometrické transformace. I zde se nabízejí dva způsoby.

Registrace obrazu

Pomocí jednoduché registrace se nalezne libovolná geometrická transformace a pomocí ní se jeden ze vstupních obrazů upraví pro následnou 2D registraci řezů. To ovšem znamená vyšší časovou náročnost vzhledem k tomu, že se jedná o 3D registraci.

Odhad translačního parametru

Jelikož se snažíme především o to, aby byly registrovány vždy dva obrazy představující stejný řez, omezíme se pouze na translaci v ose z , která by v ideálním případě upravila jeden z obrazů tak, že bychom mohli vybírat postupně snímky, které by se nacházely na indexu i svého 3D obrazu a zároveň si odpovídaly. Pokud bychom se k tomu rozhodli použít 3D registraci, jednalo by se pouze o speciální případ první varianty, kterou jsem zhodnotil jako nevhodnou. Hledaná transformace obsahuje vektor translačních parametrů $t = (t_x, t_y, t_z)$. Jelikož cílem je posunout obraz po ose z , bude pouze t_z nabývat nenulových hodnot: $t = (0, 0, t_z)$. Hledání se tedy omezuje pouze na parametr t_z . Je důležité si uvědomit, že CT snímky, které registrujeme, jsou výsledky segmentace jater, což je zároveň oblast našeho zájmu. Výhodou toho, že snímky již prošly segmentací, je dostupnost binární masky, která říká, kde se jaterní tkáň na snímku nachází. Jelikož je žádoucí, aby si jednotlivé řezy obou snímků odpovídaly zejména v oblasti jater, nalezneme parametr t_z translace t , která posune jeden z obrazů tak, aby rozložení binární masky na ose z bylo pro dvojici snímků co nejvíce podobné. Aby se z informace o rozložení jater mohl určit parametr translace, je možné například počítat aritmetický průměr indexů, na kterých se jaterní tkáň nachází, tedy tam, kde binární maska nabývá nenulových hodnot. Pro 3D snímek bude výsledkem trojrozměrný vektor $(\bar{x}, \bar{y}, \bar{z})$, ze kterého nás zajímá pouze jeho složka \bar{z} . Translační parametr t_z je poté možno stanovit pomocí vztahu:

$$t_z = \bar{z}_1 - \bar{z}_2, \quad (4.2)$$

a tedy:

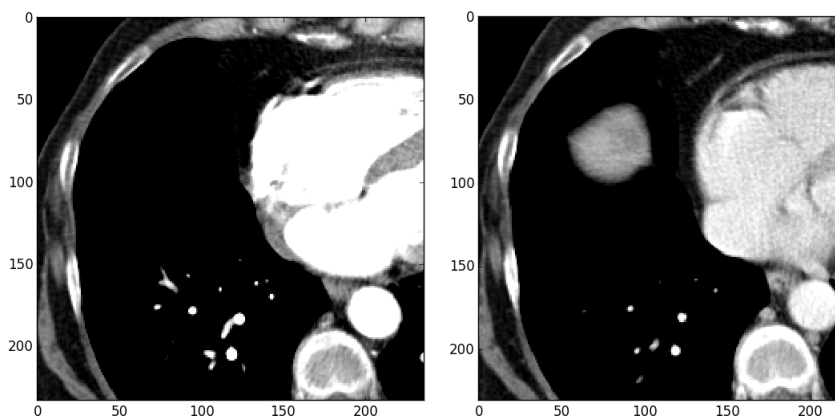
$$t = (0, 0, \bar{z}_1 - \bar{z}_2), \quad (4.3)$$

kde \bar{z}_1 a \bar{z}_2 jsou aritmetické průměry indexů nenulových hodnot binárních masek prvního a druhého snímku v ose z a t bude výsledná translace, podle které budeme transformovat registrovaný snímek před 2D registrací řezů.

Ještě před spuštěním registrace je často nutné obrazy oříznout, neboť translace jednoho ze snímků má za následek vytvoření jednoho či několika prázdných řezů na začátku, nebo na konci obrazu podle toho, zda je hodnota t_z kladná, nebo záporná. Prázdné řezy je tedy nutno odstranit a s nimi i odpovídající řezy z druhého snímku vstupujícího do registrace, jelikož oba vstupní obrazy musí být stejné velikosti.

4.2.1 Testování registrace CT snímků

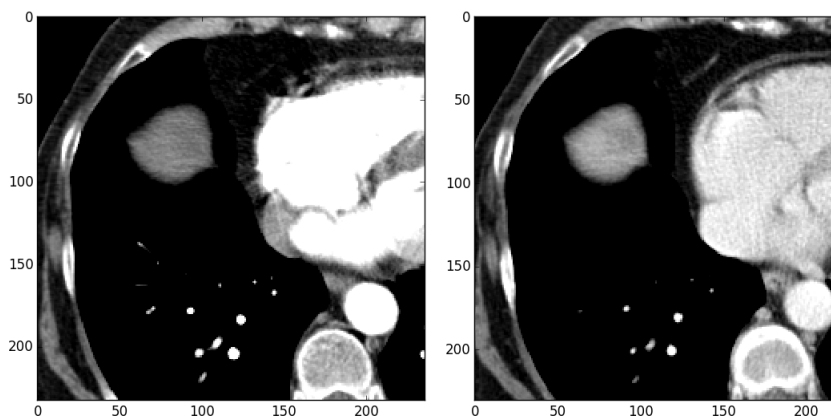
V tomto testování nepozorujeme pouze kvalitu registrace, ale také její dobu trvání. Výsledek testování určí, která ze dvou výše zmíněných metod bude rychlejší a která bude vykazovat vyšší kvalitu. Před začátkem testování je velmi důležité poznamenat, že standardní metodou pro registraci 3D snímků je klasická 3D registrace, která je navíc podporována v rámci knihovny SimpleITK [2]. 2D registraci po řezech, která je pokusem o její náhradu, by bylo rozumné posoudit jako lepší pouze v případě, že by se metoda vyznačovala mnohem vyšší kvalitou i rychlostí oproti klasické 3D registraci. 2D registraci po řezech lze navíc provést pouze za určitých podmínek, o kterých budou pojednávat další odstavce. Pro porovnání efektivity obou metod je nutné, aby byly testovány za stejných podmínek, tedy za použití stejných vstupních obrazů a rovněž stejných parametrů registrace. Pro testování byly vybrány 2 CT snímky (obr. 4.10) získané během dvou odlišných fází.



Obrázek 4.10: Vlevo na obrázku vidíme 2. řez vztažného snímku, vpravo je k vidění 2. řez registrovaného snímku.

Z obrázku 4.10 je zřejmé, že si oba obrazy v některých oblastech neodpovídají. Zejména v levé horní části registrovaného snímku vidíme vrstvu objektu (jaterní tkáň), který se ve vztažném snímku nevyskytuje. Nyní budou z 3D obrazů znovu vytaženy 2 řezy. V případě vztažného snímku se však bude jednat o řez na indexu 3 a řez registrovaného snímku zůstane stejný.

Novou situaci je možné vidět na obrázku 4.11. V levé horní oblasti vztažného snímku se ukázala vrstva objektu, který vidíme v 2. řezu registrovaného snímku. Oba obrazy nyní s velkou pravděpodobností odpovídají stejné úrovni na ose z . Aby si obrazy odpovídaly i indexem řezu ve svých 3D snímcích, musel by být jeden z nich posunut po ose z .



Obrázek 4.11: Vlevo na obrázku vidíme 3. řez vztažného snímku, vpravo je k vidění 2. řez registrovaného snímku.

Nabízí se tedy možnost transformovat registrovaný snímek pomocí translace o velikosti $t = (0, 0, -1)$. Velikost posunu po ose z zjistíme výše zmíněnou metodou nalezení parametru t_z translace $t = (t_x, t_y, t_z)$ jako rozdílu průměrů indexů výskytu binární masky jater na ose z pro oba obrazy. Výsledek $t_z = -0.9603$ se příliš neliší od hrubého odhadu hodnoty $t_z = -1$. První řez registrovaného obrazu zůstane však po translaci prázdný a bude nutné jej odstranit z obou snímků, aby mohla registrace proběhnout bez potíží.

Registrace s $LR = 1$ a $M = 1$

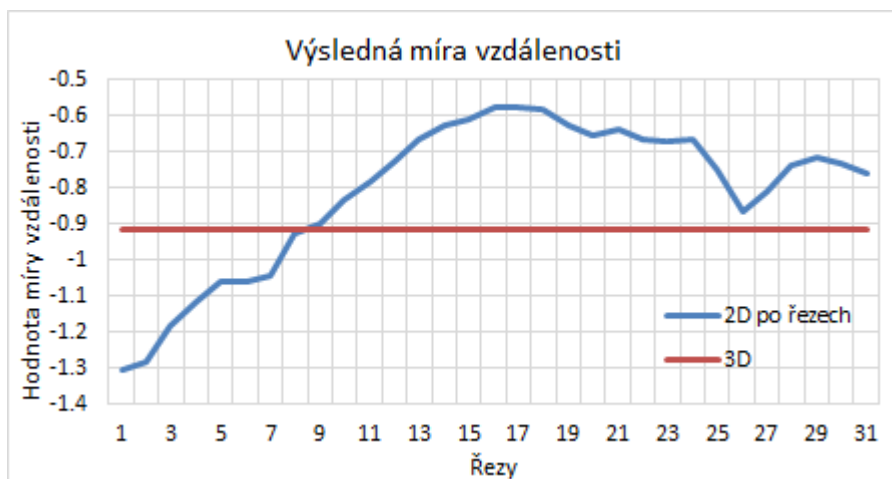
Typ registrace	Doba trvání (s)
2D po řezech	248
3D	675

Tabulka 4.2: Doby trvání výpočtu pro 100 iterací

Obrázek 4.12 poslouží pro porovnání kvality obou typů registrace. 2D registrace byla provedena pro každý řez zvlášť, a tím pádem dosahovala pro jednotlivé řezy různých hodnot vzájemné informace. Oproti tomu byla 3D registrace provedena pouze jednou pro celé obrazy, tudíž její výslednou hodnotu míry vzdálenosti znázorníme v grafu jako konstantní funkci.

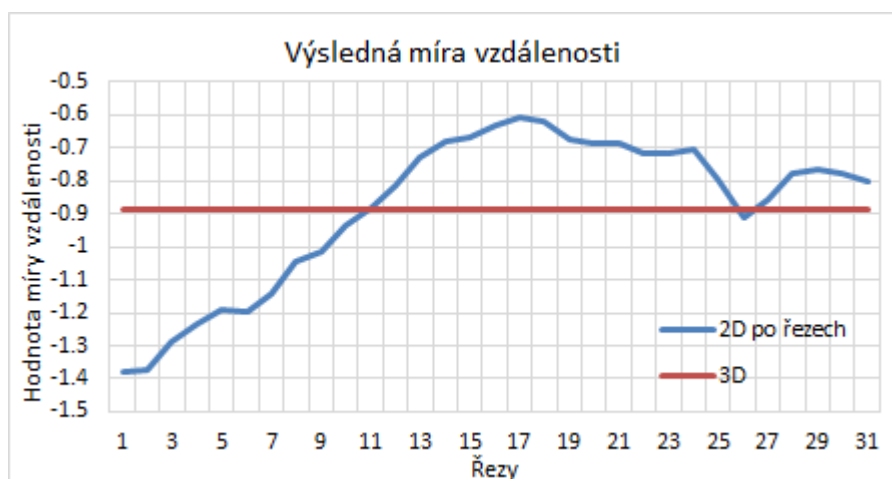
Pro 3D registraci dosáhla míra vzdálenosti hodnoty -0.91607 . U 2D registrace bylo dosaženo hodnot nižších než -0.9 jen v prvních řezech. V místech, kde se vyskytovala oblast našeho zájmu, tedy játra, dosahovala vzájemná informace výrazně vyšších hodnot. Například pro 13. řez byla výsledná míra vzdálenosti rovna -0.66749 .

2D registrace pro každý řez zvlášť vykazovala mnohem vyšší rychlost než klasická 3D registrace, která se však vyznačovala vyšší kvalitou ve smyslu hodnoty míry vzdálenosti na konci procesu.



Obrázek 4.12: Výsledná hodnota míry vzdálenosti pro příslušné řezy.

Registrace s $LR = 1$ a $M = 5$



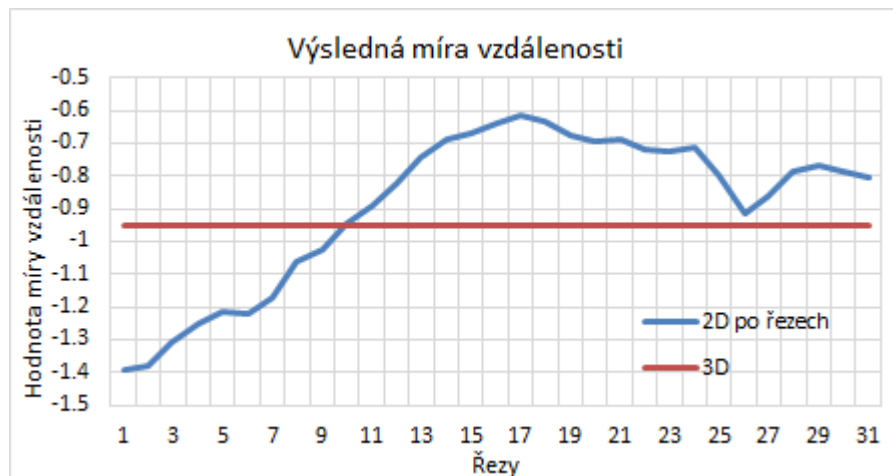
Obrázek 4.13: Výsledná hodnota míry vzdálenosti pro příslušné řezy.

Typ registrace	Doba trvání (s)
2D po řezech	118
3D	669

Tabulka 4.3: Doby trvání výpočtu pro 35 iterací

Stejně jako v predešlém případě platí, že pro většinu řezů se 2D registrace vyznačovala vyšší hodnotou vzájemné informace, tedy nižší kvalitou, o čemž se můžeme přesvědčit pomocí obrázku 4.13.

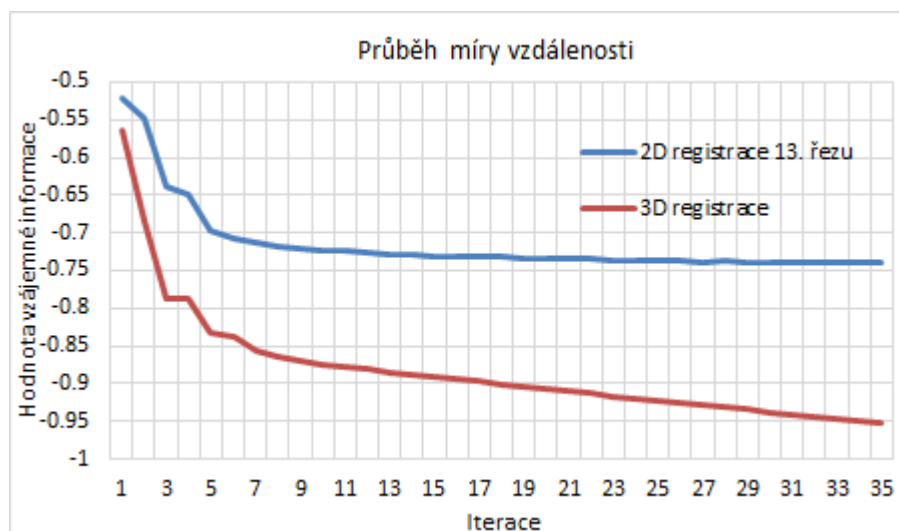
Registrace s $LR = 20$ a $M = 5$



Obrázek 4.14: Výsledná hodnota míry vzdálenosti pro příslušné řezy.

Obrázek 4.14 ukazuje, že zvýšená konstanta učení vedla ke snížení hodnoty míry vzdálenosti 3D registrace, která opět vykazovala nižší hodnoty vzájemné informace než 2D registrace pro většinu řezů.

Pro 2D registraci řezů se výsledná míra vzdálenosti se zvyšující konstantou učení příliš neměnila. Naopak 3D registrace vykazovala s rostoucí konstantou učení lepší výsledky. Zvýšení M však mělo u 2D registrace po řezech značný vliv na změnu výsledné hodnoty vzájemné informace, a to dokonce větší než u klasické 3D registrace. Vliv parametrů na výslednou hodnotu vzájemné informace pro klasickou 3D registraci je detailně popsán v kapitole 4.1.1.



Obrázek 4.15: Graf průběhu vzájemné informace

Míra vzdálenosti měla po 35 iteracích klasické 3D registrace obrazu větší tendenci klesat než u 2D registrace řezů, což ukazuje obrázek 4.15. Například výsledná míra vzdálenosti

13. řezu pro 200 iterací měla hodnotu -0.75553 a registrační proces pro všechny řezy trval 628 sekund, tedy přibližně stejnou dobu jako 35 iterací u 3D registrace.

Za předpokladu, že rozdíl obou registrovaných obrazů v souřadnici z bude možno vyřešit jednoduchou translací jednoho z obrazů po této ose, lze metodu 2D registrace jednotlivých řezů aplikovat. Musí však být k dispozici binární maska jater pro oba snímky, což jsou aspekty, které nelze jednoznačně předpokládat. 2D registraci po řezech doporučuji využít v případě, že je kladen důraz na rychlost běhu programu a nároky na výslednou kvalitu nejsou vysoké. Metoda je zhruba 2.72 krát rychlejší než klasická 3D registrace při zvolení $M = 1$. Pro $M = 5$ však metoda vykazovala rychlost 5.67 krát vyšší. Pokud tedy volíme větší rozměry mřížky, stojí rovněž za zvážení využít metodu 2D registrace řezů. V obecném případě je robustnějším řešením aplikovat klasickou 3D registraci obrazu, která oproti registraci řezů vykazuje větší rozdíly ve výsledné míře vzdálenosti při změnách konstanty učení.

Kapitola 5

Závěr

Cílem této práce bylo prozkoumat metody registrace obrazu a následně vybrat takovou, která by byla pro registraci trojrozměrných CT snímků nejvhodnější. Nejprve jsem se věnoval představení různých geometrických transformací, kdy jsem se snažil nejen uvést potřebnou teorii, ale i demonstrovat, jak se obrazová data po transformacích změni. Ze známých typů transformací, uváděných Fitzpatrickem aj. v jejich publikaci [1], jsem se zaměřil na translaci, nerigidní transformaci za využití translace, rotace a změny měřítka, afinní transformaci a B-spline transformaci. Z dalších důležitých parametrů registrace, jakými jsou například míra vzdálenosti a optimalizátor, jsem rovněž představil několik nejpoužívanějších typů v knihovně SimpleITK [2]. Z optimalizátorů jsem zkoumal metody LBFGS a klesání gradientu a ze známých typů měř vzdálenosti jsem se věnoval součtu kvadrátů odchylek, korelaci a vzájemné informaci.

Dále následovalo testování registrace obrazu za účelem výběru nejlepších parametrů, což bylo pro jednoduchost prováděno nejdříve na umělých binárních 2D obrazech. Testování 2D registrace bylo prováděno pro všechny zmíněné typy geometrické transformace, míry vzdálenosti a optimalizátoru. Důležitým ukazatelem pro hodnocení kvality registrace byla výsledná míra vzdálenosti. Na základě výsledků testování jsem jako nejvhodnější optimalizátor používaný v knihovně SimpleITK [2] vybral metodu klesání gradientu pomocí pravidelného kroku, která vedla ve většině případů k nejkvalitnější registraci.

Další testování 2D registrace opustilo oblast binární dat. Objekty ve vstupních obrazech již neměly stejný poměr intenzit a navíc byly do registrovaného obrázku přidány další objekty, které ve vztažném chyběly. Výsledkem testování bylo zvolení vzájemné informace jako nejvhodnější míry vzdálenosti, přičemž nebyl brán ohled pouze na její hodnotu na konci registračního procesu, ale také na výsledný registrovaný obraz.

Pro testování registrace trojrozměrných obrazových dat, v našem případě CT snímků, jsem použil B-spline transformaci, tedy jednu z elastických transformací, které byly v lékařských obrazových datech očekávány. Jejím důležitým parametrem se ukázala velikost mřížky (viz [3]), která měla vliv nejen na dobu trvání jedné iterace, ale i na kvalitu registrace. Dalším parametrem, který ovlivňoval průběh míry vzdálenosti, byla konstanta učení, tedy parametr optimalizátoru [2]. Testování 3D registrace bylo prováděno za účelem zvolení optimální velikosti mřížky a konstanty učení pro registraci CT snímků. Z důvodu

testování různých velikostí mřížky, které ovlivňovaly rychlost výpočtu v každé iteraci, jsem věnoval velkou pozornost rovněž době trvání registrace. Na základě výsledků testování jsem určil hodnotu 45 jako optimální konstantu učení a velikost mřížky jsem nastavil na hodnotu 3.

V poslední kapitole 4.2 jsem se zabýval jiným způsobem registrace CT snímků. 2D registraci po řezech jsem zkoumal coby náhradu za klasickou 3D registraci. V následujícím testování jsem oba způsoby porovnal a uvedl jsem, za jakých podmínek by bylo vhodné 2D registraci po řezech použít a kdy naopak nikoliv. 2D registraci po řezech jsem testoval i z toho důvodu, že 3D registrace byla v porovnání s registrací 2D snímků pomalejší, což byl jeden z hlavních problémů, na které jsem během práce na tomto projektu narazil. Pro implementaci jednotlivých metod v jazyku Python jsem využíval knihovnu SimpleITK [2], která nabízí v oblasti registrace obrazu obrovské možnosti a svými vysoce optimalizovanými metodami pomáhá redukovat časovou náročnost výpočtu, čímž mi práci na projektu usnadnila, a tedy i zpříjemnila. Dalším směrem, kterým by se měl výzkum metod registrace obrazu ubírat, bych zvolil případnou časovou optimalizaci 3D registrace, kterou bych shledal velice přínosnou pro výzkum v oblasti analýzy lékařských obrazových dat.

Literatura

- [1] FITZPATRICK, J. Michael; HILL, Derek L. G.; MAURER Jr., Calvin R. *Handbook of medical imaging: Image registration*, 2000, 2: 447-513.
- [2] JOHNSON, Hans J.; MCCORMICK, Matthew M.; IBANEZ, Luis. *The ITK software guide*. Kitware, Inc., USA, 2016.
- [3] BSplineTransform, *ITK - Doxygen dokumentace*. [online]. 21.3.2016 [cit. 2016-03-21]. Dostupné z: http://www.itk.org/Doxygen/html/classitk_1_1BSplineTransform.html
- [4] BASTL, Bohumír. *B-spline, NURBS křivky*. [online]. Plzeň, Západočeská univerzita v Plzni, Fakulta aplikovaných věd, [cit. 2016-03-21]. Dostupné z: http://geometrie.kma.zcu.cz/index.php/www/content/download/1156/3275/file/GPM_NURBS.pdf?PHPSESSID=05c59e8bf9cc6c31b703704908efa66e
- [5] SONKA, Milan; HLAVAC, Vaclav; BOYLE, Roger. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

Seznam tabulek

3.1	Výsledná míra vzdálenosti (translace, SSD)	15
3.2	Výsledná míra vzdálenosti (translace, korelace)	15
3.3	Výsledek registrace (translace, MMI)	16
3.4	Výsledek registrace (translace, JHMI)	16
3.5	Zvolené parametry nerigidní 2D transformace	17
3.6	Výsledná míra vzdálenosti (nerigidní transformace, SSD)	17
3.7	Výsledná míra vzdálenosti (nerigidní transformace, korelace)	18
3.8	Výsledek registrace (nerigidní transformace, MMI)	18
3.9	Výsledek registrace (nerigidní transformace, JHMI)	19
3.10	Výsledná míra vzdálenosti (afinní transformace, SSD)	21
3.11	Výsledná míra vzdálenosti (afinní transformace, korelace)	22
3.12	Výsledek registrace (afinní transformace, MMI)	22
3.13	Výsledek registrace (afinní transformace, JHMI)	23
3.14	Výsledky registrace za použití B-spline transformace pro jednotlivé míry vzdálenosti	25
3.15	Zvolené parametry nerigidní 2D transformace (přídavné objekty)	28
3.16	Výsledné míry vzdálenosti (nerigidní transformace, přídavné objekty)	28
3.17	Získané parametry nerigidní transformace (přídavné objekty)	29
3.18	Výsledek 2D registrace (B-spline, přídavné objekty)	30
4.1	Doba trvání registrace CT snímků	34
4.2	Doby trvání 100 iterací 3D registrace a registrace 2D po řezech	43
4.3	Doby trvání 35 iterací 3D registrace a registrace 2D po řezech ($LR = 1$, $M = 5$)	44

Seznam obrázků

2.1	Výchozí 2D obrázky	7
2.2	Registrovaný 2D obrázek (translace)	8
2.3	Výsledek nerigidní transformace 2D obrázku	8
2.4	Výsledek afinní transformace 2D obrázku	9
2.5	Výsledek B-spline transformace 2D obrázku	9
3.1	Stav na začátku 2D registrace za použití translace	14
3.2	Stavy před začátkem 2D registrace za použití nerigidní transformace	17
3.3	Výsledek 2D registrace (nerigidní transformace, SSD) pro různé optimalizační zátory	18
3.4	Výsledek 2D registrace (nerigidní transformace, MMI, pravidelný krok)	19
3.5	Výsledek 2D registrace (nerigidní transformace, JHMI, pravidelný krok)	19
3.6	Průběh míry vzdálenosti (nerigidní 2D transformace, SSD, pravidelný krok)	20
3.7	Stav po afinní 2D transformaci	21
3.8	Výsledek 2D registrace (afinní transformace, SSD)	22
3.9	Výsledek 2D registrace (afinní transformace, MMI)	23
3.10	Průběh míry vzdálenosti (afinní 2D transformace, SSD, LBFGSB)	24
3.11	Stav po 2D B-spline transformaci	25
3.12	Výsledky 2D registrace za použití B-spline transformace	25
3.13	Průběh míry vzdálenosti (2D B-spline transformace, SSD)	26
3.14	Dvojice rozdílných vstupních obrazů	27
3.15	Registrovaný obrázek s přidávanými objekty	28
3.16	Výsledky 2D registrace (nerigidní transformace, přídavné objekty)	29
3.17	Stav na začátku 2D registrace rozdílných snímků (B-spline)	30
3.18	Výsledky 2D registrace (B-spline transformace, přídavné objekty)	30
4.1	Průběh vzájemné informace (registrace CT snímků)	33
4.2	Průběh vzájemné informace ($LR = 1$)	34
4.3	Průběh vzájemné informace v čase (3D registrace)	35
4.4	Průběh vzájemné informace pro $M = 2$	35
4.5	Průběh vzájemné informace v čase pro $M = 5$ a různé míry učení LR	36
4.6	Hodnota vzájemné informace po 600 sekundách (3D registrace)	37
4.7	Hodnota vzájemné informace po 600 sekundách (jiná data)	37
4.8	Hodnota vzájemné informace po 600 sekundách (jiná data)	38

4.9	Průběh změny hodnoty vzájemné informace	39
4.10	Ukázka řezů vztažného a registrovaného 3D obrazu	42
4.11	Ukázka jiného vztažného a původního registrovaného řezu	43
4.12	Výsledná hodnota míry vzdálenosti pro příslušné řezy ($LR = 1, M = 1$) . .	44
4.13	Výsledná hodnota míry vzdálenosti pro příslušné řezy ($LR = 1, M = 5$) . .	44
4.14	Výsledná hodnota míry vzdálenosti pro příslušné řezy ($LR = 20, M = 5$) .	45
4.15	Průběh vzájemné informace u 3D registrace a registrace 2D po řezech . . .	45