



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

INTERNETOVÁ KNIHOVNA METOD A ALGORITMŮ PRO REGISTRACI MEDICÍNSKÝCH OBRAZŮ

**INTERNET LIBRARY OF METHODS AND ALGORITHMS USED FOR MEDICAL IMAGE
REGISTRATION**

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

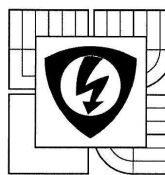
AUTOR PRÁCE
AUTHOR

JAKUB KOŇAŘÍK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. ROMAN PETER

BRNO 2011



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav biomedicínského inženýrství

Bakalářská práce

bakalářský studijní obor
Biomedicínská technika a bioinformatika

Student: Jakub Koňářik

Ročník: 3

ID: 109606

Akademický rok: 2010/11

NÁZEV TÉMATU:

**Internetová knihovna metod a algoritmů pro registraci medicínských
obrazů**

POKYNY PRO VYPRACOVÁNÍ:

Navrhněte strukturu a uživatelské rozhraní multimediální internetové knihovny pro prezentaci metod používaných v oblasti registrace medicínských obrazů. Tento informační systém realizujte, naplňte několika příklady spustitelných algoritmů, příslušnými teoretickými formulacemi a dostupnými obrazovými daty. Funkci celého systému včetně vybraných algoritmů s možností vzdáleného výpočtu ověřte. Vytvořte uživatelskou a administrátorskou příručku celého systému.

DOPORUČENÁ LITERATURA:

- [1] JAN, J. Medical Image Processing, Reconstruction and Restoration Concepts and Methods, CRC Press, Boca Raton, USA 2005
- [2] GILMORE W.J. Velká kniha PHP a MySQL 5 - kompendium znalostí pro začátečníky i profesionály, 2007, 80-86815-53-6, ZONER Press
- [3] ZITOVÁ B., FLUSSER J. Image registration methods: a survey, Image and Vision Computing, Volume 21, Issue 11, October 2003, Pages 977-1000, ISSN 0262-8856

Termín zadání: 7.2.2011

Termín odevzdání: 27.5.2011

Vedoucí práce: Ing. Roman Peter

Konzultanti 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

Registrace obrazů je postup vedoucí k nalezení geometrické transformace, která zajistí shodnou pozici, orientaci a velikost korespondujících objektů v obrazech. Jaké jsou tyto metody a jak fungují? Tyto metody a principy jsou vysvětleny v první části této práce. Druhá část práce je zaměřena na výklad teorie informačních systémů a praktickou tvorbu internetové databáze. Praktická část byla realizována v redakčním systému Drupal.

ABSTRACT

Image registration is a procedure which is heading to search for a geometric transformation, which ensures same position, orientation and size of corresponding objects in pictures. How does these methods work? These methods and principles are explained in first part of this work. Second part of this work is aimed to explain theory about information systems and practical creation of the internet database. These practical part was realized in editorial system called Drupal.

KLÍČOVÁ SLOVA

lícování, registrace, transformace, algoritmus, interpolace, informační systém

KEY WORDS

fitting, registration, transformation, algorithm, interpolation, information system

KOŇAŘÍK, J. *Internetová knihovna metod a algoritmů pro registraci medicínských obrazů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 44 s. Vedoucí bakalářské práce Ing. Roman Peter.

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma *Internetová knihovna metod a algoritmů pro registraci medicínských obrazů* jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 27. května 2011

.....
podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Romanovi Peterovi, za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mého semestrálního projektu.

V Brně dne 27. května 2011

.....
podpis autora

OBSAH

BRNO UNIVERSITY OF TECHNOLOGY	1
OBSAH.....	5
ÚVOD.....	8
1. LÍCOVÁNÍ OBRAZŮ	9
1.1. GEOMETRICKÉ TRANSFORMACE 2D	9
1.1.1. Posunutí obrazu (<i>Translate</i>).....	9
1.1.2. Rotace, neboli otočení.....	9
1.1.3. Změna měřítka.....	10
1.1.4. Afinity transformace.....	10
1.1.5. Projektivní transformace.....	11
1.2. GEOMETRICKÉ TRANSFORMACE 3D	11
1.2.1. <i>Translate</i>	11
1.2.2. <i>Rotace</i>	12
1.2.3. <i>Zrcadlení</i>	13
1.2.4. Změna měřítka.....	13
1.2.5. <i>Střih obrazu</i>	14
1.3. INTERPOLACE.....	14
1.3.1. <i>Aproximace jasové funkce</i>	14
1.3.2. <i>Interpolace metodou nejbližšího souseda</i>	15
1.3.3. <i>Bilineární interpolace</i>	16
1.3.4. <i>Bikubická interpolace</i>	16
1.4. KRITERIÁLNÍ FUNKCE	17
1.4.1. <i>Odečítání obrazů (SSD)</i>	17
1.4.2. <i>Korelační koeficient (CC)</i>	17
1.4.3. <i>Koeficient stejnorodosti obrazu</i>	18
1.4.4. <i>Vzájemný histogram a vzájemná distribuce pravděpodobnosti</i>	18
1.4.5. <i>Vzájemná entropie</i>	18
1.4.6. <i>Maximalizace vzájemné informace (MI)</i>	19
1.4.7. <i>Normalizovaná vzájemná informace</i>	19
1.5. OPTIMALIZAČNÍ METODY.....	19
1.5.1. <i>Role optimalizační metody</i>	19
1.5.2. <i>Deterministické algoritmy</i>	20
1.5.3. <i>Stochastické algoritmy</i>	20
2. MULTIMEDIÁLNÍ INTERNETOVÁ KNIHOVNA.....	22
2.1. STRUKTURA IS	22
2.1.1. <i>Databáze</i>	22
2.1.2. <i>Server</i>	22
2.1.3. <i>Klient</i>	22
2.2. POUŽÍVANÉ TECHNOLOGIE.....	22
2.2.1. <i>Technologie serveru</i>	22
2.2.2. <i>Technologie databáze</i>	23
2.2.3. <i>Technologie klienta</i>	24

2.3. REDAKČNÍ SYSTÉMY	24
2.3.1. <i>Drupal 6</i>	24
2.3.2. <i>Joomla 1.5</i>	25
2.3.3. <i>WordPress</i>	25
2.3.4. <i>PhpRS</i>	25
3. TVORBA STRÁNEK	26
3.1. TVORBA VZHLEDU STRÁNEK	26
3.2. TVORBA OBSAHU STRÁNEK	26
3.3. TVORBA ALGORITMŮ	29
3.3.1 <i>Tvorba php skriptu</i>	29
3.3.2. <i>Tvorba exe souborů</i>	29
3.3.3. <i>Tvorba tlačítek na stránkách</i>	32
4.1. MANUÁL K ZPROVOZNĚNÍ INTERNETOVÉ KNIHOVNY	33
4.2. MANUÁL K POUŽÍVÁNÍ STRÁNEK	37
4.2.1 <i>Spouštění algoritmů</i>	39
5. ZÁVĚR	41
6. SEZNAM LITERATURY	43

ÚVOD

Lícováním obrazů se nazývá proces, v kterém se hledá geometrická transformace dvou obrazů, například *pre* a *post* kontrastních. Snaží se tedy o překrytí odpovídajících si bodů obrazů pořízených lékařskými přístroji, například rentgenem, ultrazvukem, CT (výpočetní tomografie), NMR (nukleární magnetická rezonance), PET (pozitronová emisní tomografie), SPECT (single photon emission tomography) a dalšími.

Proces registrace (lícování) obrazů má dvě části, které jsou geometrická transformace a metoda, jež nám ji vypracuje. Každá z těchto částí má mnoho variant a svých dalších podmnožností. V první části této práce jsou vysvětleny hlavně základní geometrické transformace a metody pracující s jasovými funkcemi obrazu.

Praktickou částí práce je multimediální internetová knihovna. Ta je prezentována informačním systémem. Informační systém má několik částí, u kterých je vysvětlena jejich funkčnost a význam. Tvorba internetové knihovny je vysvětlena v poslední části práce.

1. LÍCOVÁNÍ OBRAZŮ

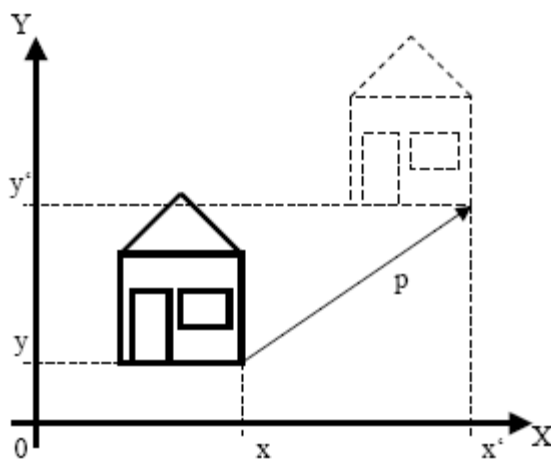
1.1. geometrické transformace 2D

Pod pojmem geometrická transformace rozumíme transformaci obrazu v prostoru. V tomto případě v 2D.

1.1.1. Posunutí obrazu (Translace)

Tato metoda spočívá v posunutí objektu v souřadném systému, při zachování úhlů a vzdáleností. Translace je dána pouze vektorem posunutí $p=(x_t, y_t)$.

V maticovém zápisu: $[x', y', w'] = [x, y, w] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_t & y_t & 1 \end{bmatrix}$ (1.1)



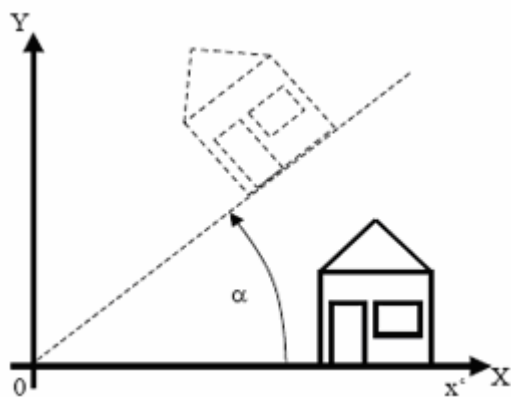
Obrázek 1 (přejato z [15])

1.1.2. Rotace, neboli otočení

Všechny úhly a vzdálenosti objektu jsou opět zachovány, pouze dojde k jeho otočení o určitý úhel α .

Pro transformaci souřadnic platí následující vztah: $x' = x \cos \alpha - y \sin \alpha$
 $y' = x \sin \alpha + y \cos \alpha$

V maticovém zápisu: $[x', y', w'] = [x, y, w] \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$ (1.2)



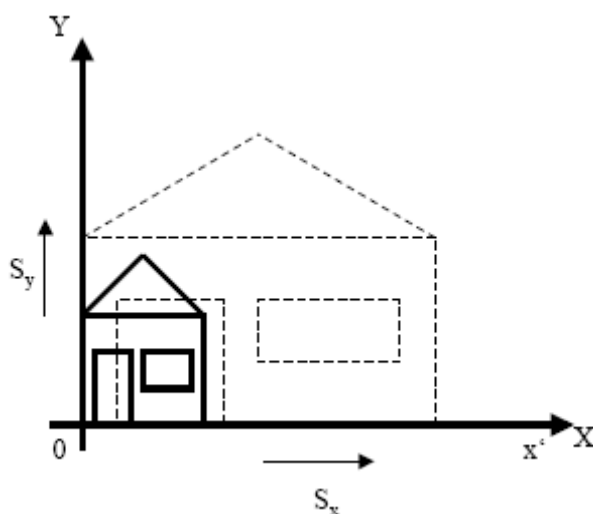
Obrázek 2 (přejato z [15])

1.1.3. Změna měřítka

Zde již dochází ke změně vzdáleností a velikostí úhlů, zachováváme pouze rovnoběžnost. Měníme měřítko ve směru os.

Platí pro ni následující: $x' = s_x x$ a $y' = s_y y$

V maticovém zápisu: $[x', y', w'] = [x, y, w] \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$ (1.3)

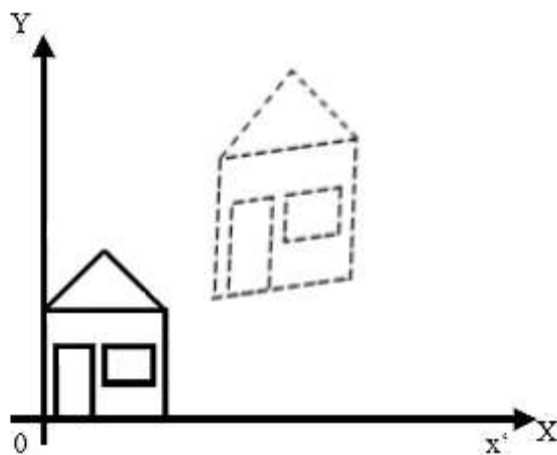


Obrázek 3 (přejato z [15])

1.1.4. Afinní transformace

Spočívá ve zkosení objektů. Zachovávají se rovnoběžky a poměr stran. Přímký se nám tedy netransformují v křivky.

V maticovém zápisu: $[x', y', w'] = [x, y, w] \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ p_1 & p_2 & 1 \end{bmatrix}$, kde $p=(p_1, p_2)$ a $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ (1.4)



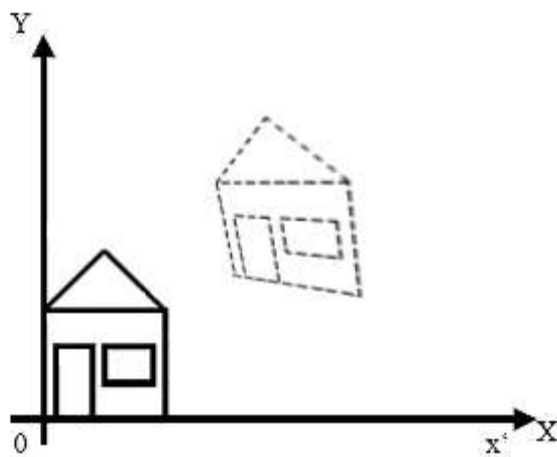
Obrázek 4 (přejato z [15])

1.1.5. Projektivní transformace

Poslední geometrická transformace, kde zachováváme pouze přímost přímek. Rovnoběžky se můžou stát různoběžkami. Transformační matice může mít libovolné hodnoty prvků.

Využívá se pro vyjadřování perspektivy.

V maticovém zápisu: $[x', y', w'] = [x, y, w] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$ (1.5)



Obrázek 5 (přejato z [15])

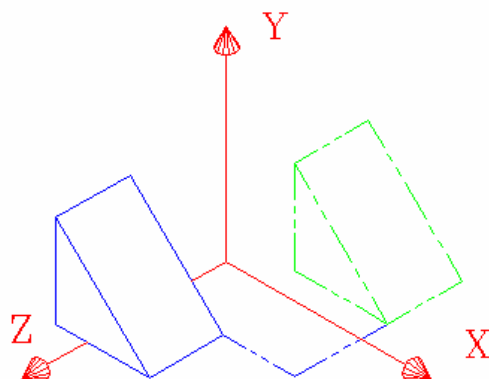
1.2. geometrické transformace 3D

1.2.1. Translace

Translací měníme polohu každého z bodů (x, y, z) na $(x + D_x, y + D_y, z + D_z)$, při zachování vzdáleností a úhlů mezi těmito body.

V maticovém zápisu:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & D_x \\ 0 & 1 & 0 & D_y \\ 0 & 0 & 1 & D_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (1.6)$$



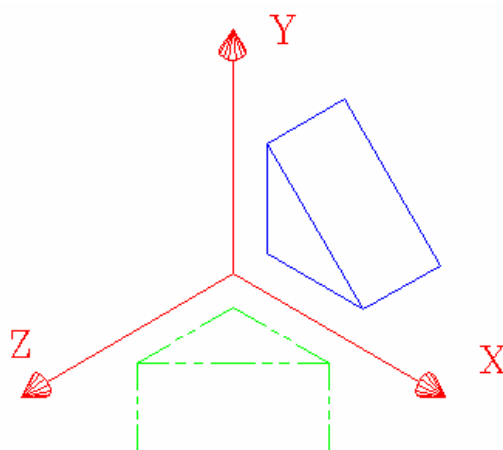
Obrázek 6 (přejato z [10])

1.2.2. Rotace

Zde dojde k rotaci objektu dle všech tří os o určitý úhel, při zachování všech úhlů a vzdáleností objektu.

V maticovém zápisu pro rotaci kolem každé z os (řecká písmena prezentují úhel rotace):

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R_y(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\phi & 0 & \cos\phi & D_z \\ 0 & 0 & 0 & 1 \end{bmatrix} R_z(\omega) = \begin{bmatrix} \cos\omega & -\sin\omega & 0 & 0 \\ \sin\omega & \cos\omega & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.7)$$

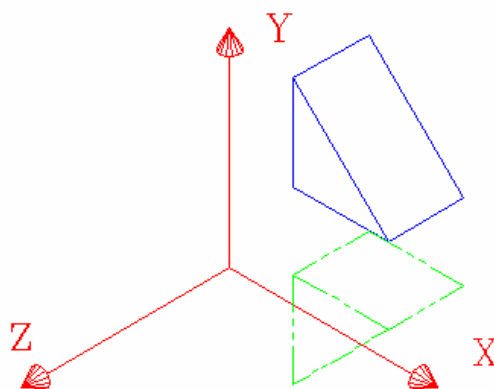


Obrázek 7 (přejato z [10])

1.2.3. Zrcadlení

Zrcadlení obrazu, při zachování vzdáleností a úhlů. Dojde pouze k přetočení obrazu.

V maticovém zápise: $T_{xy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $T_{yz} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $T_{xz} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ (1.8)



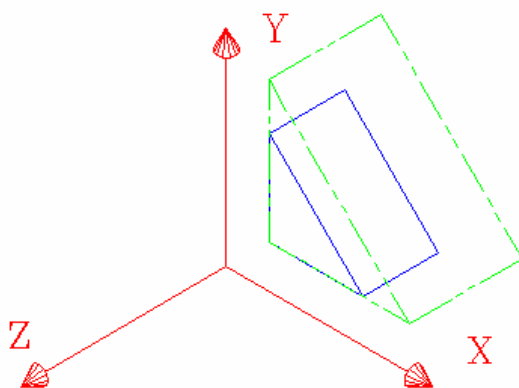
Obrázek 8 (přejato z [10])

1.2.4. Změna měřítka

Dochází ke změně vzdáleností a velikostí úhlů, zachováváme pouze rovnoběžnost. Měníme měřítko ve směru všech tří os.

Platí pro ni následující: $x' = s_x x$ a $y' = s_y y$ a $z' = s_z z$

V maticovém zápise: $\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$ (1.9)



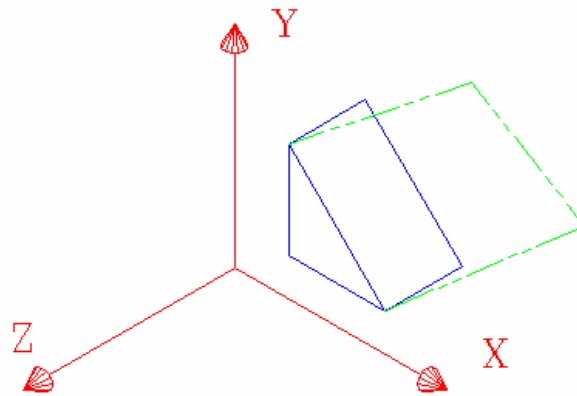
Obrázek 9 (přejato z [10])

1.2.5. Střih obrazu

Dochází ke změně vzdáleností a úhlů. Změna každé souřadnice je dána lineární kombinací všech tří souřadnic.

V maticovém zápisu:

$$\begin{bmatrix} 1 & a & b & 0 \\ c & 1 & d & 0 \\ e & f & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + ay + bz \\ cx + y + dz \\ ex + fy + z \\ 1 \end{bmatrix} \quad (1.10)$$



Obrázek 10 (přejato z [10])

1.3. Interpolace

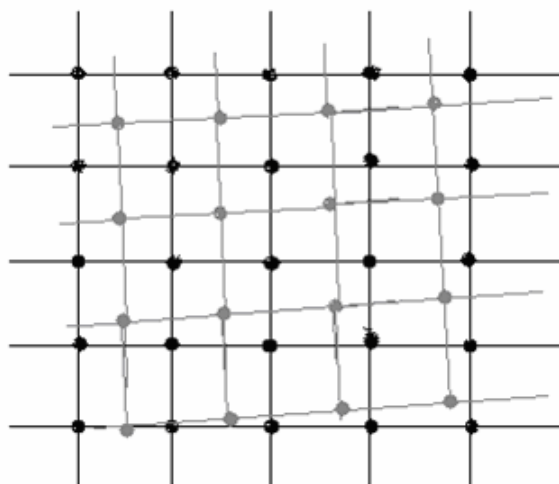
Výsledky transformace obrazů nám vždy nemusí vyjít celočíselně či stejně orientovány. Proto potřebujeme náš transformovaný obraz převést do původního diskrétního prostoru – interpolovat. Proto musíme vypočítat hodnoty nových elementů, jejichž souřadnice jsou celými čísly. K čemuž slouží několik interpolačních technik.

1.3.1. Aproximace jasové funkce

Tato interpolace spočívá v proložení několika původních bodů, v okolí nového, aproximační funkcí (nejčastěji polynomem). Aproximaci lze ale vyjádřit jako dvojrozměrnou konvoluci. Obrazovou funkci $f(x,y)$ máme jen v vzorkované verzi $g_s(l\Delta x, k\Delta y)$. Výsledná funkce bude po aproximaci:

$$f_n(x, y) = \sum_{l=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} g_s(l\Delta x, k\Delta y) h_n(x - l\Delta x, y - k\Delta y) \quad (1.11)$$

kde h_n je interpolační jádro, obvykle má malé pokrytí, a Dx , Dy jsou diskretizační kroky v jednotlivých směrech.



Obrázek 11 (přejato z [11])

1.3.2. Interpolace metodou nejbližšího souseda

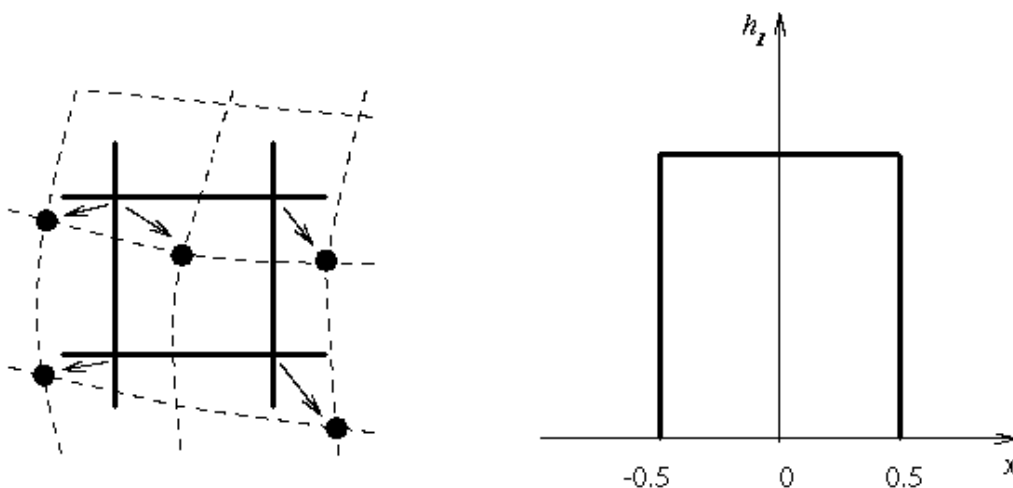
Tato interpolace je nejjednodušší a nejrychlejší, což má za důsledek ne vždy uspokojivou kvalitu transformovaného obrazu.

Funkce interpolace:

$$f_n(x, y) = g_s(\text{round}(x), \text{round}(y)), \quad (1.12)$$

kde *round* je zaokrouhlení. Jako konvoluce, má interpolační jádro h v jednorozměrném případě charakter:

$$h_1^1(x, y) = \begin{cases} 1, & t \in \langle -0,5; 0,5 \rangle \\ 0, & \text{jinde} \end{cases} \quad \text{v dvourozměrném pak dostaneme } h_1(x, y) = h_1^1(x) \cdot h_1^1(y) \quad (1.13)$$



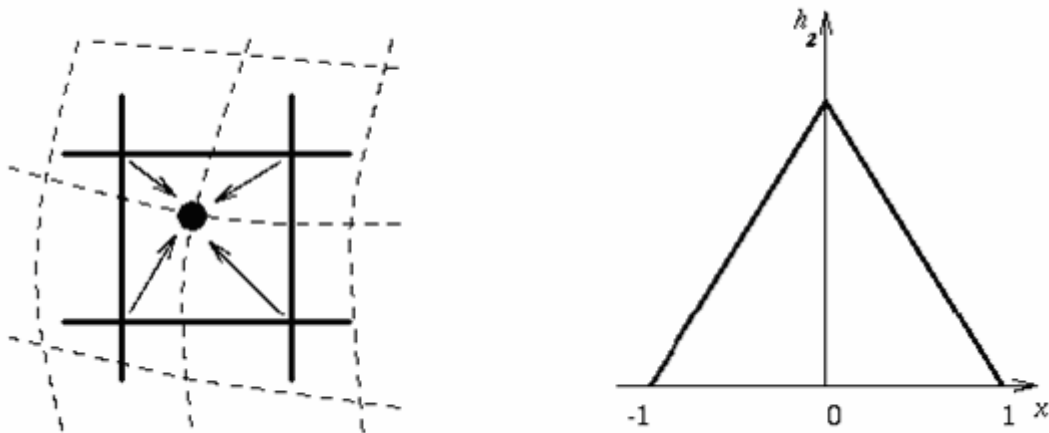
Obrázek 12 (přejato z [11])

1.3.3. Bilineární interpolace

Metoda početně náročnější než předešlá, ovšem stále ji řadíme mezi jednodušší, postup je v proložení 4 nejbližších bodů třemi přímkami. Jednou přímkou proložíme dva body a druhou přímkou dva sousední body. Třetí pak bude procházet mezi našimi dvěma přímkami a nad hledaným bodem. Nový bod je pak vážená suma čtyř okolních bodů.

Ve 2D máme: $h_2(x, y) = h_2^1(x) \cdot h_2^1(y)$

$$\text{V konvoluci pak dostaneme: } h_2^1(x, y) = \begin{cases} 1-t, t \in \langle 0;1 \rangle \\ 1+t, t \in \langle -1;0 \rangle, \\ 0, \text{ jinde} \end{cases} \quad (1.14)$$



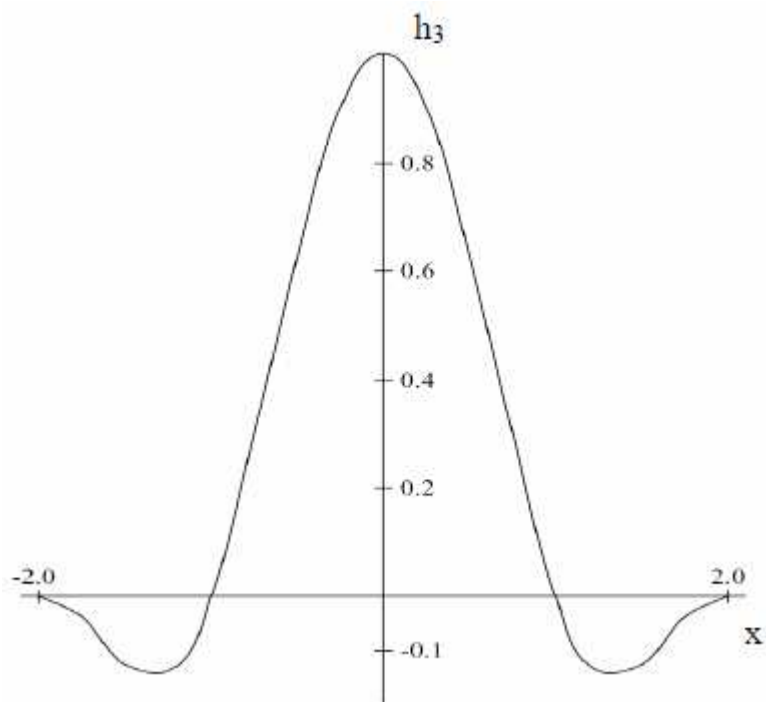
Obrázek 13 (přejato z [11])

1.3.4. Bikubická interpolace

Tato metoda využívá k nalezení hodnot nových bodů aproximaci původní funkce polynomem třetího řádu. Výpočetní náročnost této metody je velká, protože je třeba prokládat okolních 16 bodů. Výhodou metody je ostrost transformovaného obrazu, nevýhodou je, že se při generování můžeme dostat do záporných hodnot. Interpolační jádro h má charakter vlny souměrné dle osy y v řezu a jeho zápis vypadá takto:

Pro 2D: $h_3(x, y) = h_3^1(x) \cdot h_3^1(y)$

$$\text{Kovoluce : } h_3^1(x, y) = \begin{cases} 1-2|t|^2+|t|^3, |t| < 1 \\ 4-8|t|+5|t|^2-|t|^3, 1 \leq |t| < 2 \\ 0, \text{ jinde} \end{cases} \quad (1.15)$$



Obrázek 14 (přejato z [11])

1.4. Kriteriační funkce

1.4.1. Odečítání obrazů (SSD)

Tato metoda je založena na odečítání intenzit jednotlivých bodů vstupních obrazů. Metoda je velmi přesná pokud nám obrazy přesně pasují na sebe. V opačném případě, tedy pokud obrazy nejsou shodné a přesně na sebe nepasují, či pokud jsou zašuměny, dojde k registrační chybě a metoda je nepoužitelná. Funkce má tvar:

$$SSD = \frac{1}{N} \sum_i^N |R(i) - L'(i)|^2 \quad (1.16)$$

kde N je počet shodných bodů, i je příslušný bod, \mathbf{R} je vstupní obraz a \mathbf{L}' je zpětně transformovaný obraz. Funkci počítáme pro $i \in \mathbf{R} \cap \mathbf{L}'$.

1.4.2. Korelační koeficient (CC)

Vyjadřuje nám míru shodnosti obrazů.

$$CC = \frac{\sum_i (R(i) - \bar{R})(L'(i) - \bar{L}')}{\sqrt{\sum_i (R(i) - \bar{R})^2 \sum_i (L'(i) - \bar{L}')^2}} \quad (1.17)$$

kde \mathbf{R}, \mathbf{L}' jsou příslušné střední hodnoty intenzit bodů obrazů. Funkci počítáme pro $i \in \mathbf{R} \cap \mathbf{L}'$.

1.4.3. Koeficient stejnorodosti obrazu

U této metody počítáme v každém kroku hodnotu koeficientu Q . Tento koeficient počítáme jako poměr intenzit transformovaného a původního obrazu. Algoritmus je navržen tak, aby minimalizoval směrodatnou odchylku:

1. Vypočti koeficienty obrázku $Q(i)=L'(i)/R(i)$ (1.18)

2. Vypočti střední hodnotu koeficientu Q : $\mu_Q = \frac{1}{Q} \sum_i Q(i)$ (1.19)

3. Vypočti směrodatnou odchylku koeficientu Q : $\sigma_Q = \frac{1}{Q} \sum_i (Q(i) - \mu_Q)^2$ (1.20)

4. RIU=normalizovaná standardní odchylka: $\frac{\sigma_Q}{\mu_Q}$ (1.21)

1.4.4. Vzájemný histogram a vzájemná distribuce pravděpodobnosti

Zde v jednotlivých krocích počítáme vzájemný histogram vstupního a transformovaného obrazu. Z tohoto histogramu pak hledáme jejich transformační vztah, pokud by si tedy obrazy naprosto odpovídaly, byl by výsledný histogram přímkou, která by byla osou úhlu soustavy souřadnic (ve 2D). Normalizovaný histogram, pak přejde ve vzájemnou distribuční funkci pravděpodobnosti - *PDF*. Známe tedy transformační vztah vstupního a transformovaného obrazu a vypočteme *PDF* pro rozdělení jednotlivých intenzit:

1. Vytvořme prostor N_a na N_b prvků pro histogram $HIST[j,k]$

2. Inicializujme histogram: $HIST[j,k]=0$ pro všechny j,k

3. pro každý bod $i \in \mathbf{R} \cap \mathbf{L}'$ vypočteme hodnoty intenzit $\mathbf{R}(i)$ a $\mathbf{L}'(i)$, vypočteme polohu intenzit čísel r, l odpovídajících bodům $\mathbf{R}(i)$ a $\mathbf{L}'(i)$ a inkrementujeme $HIST[j,k]$

4. Vypočteme $\sum_{j,k} HIST[j,k]$

5. Normalizujeme histogram, abychom mohli vypočítat *PDF*:

$$PDF[j,k] = \frac{HIST[j,k]}{\sum_{j,k} HIST[j,k]} \quad (1.22)$$

1.4.5. Vzájemná entropie

Zde pracujeme s mírou informace v obrazu, kterou vyjádříme opět v *PDF*. Intenzity bodů se nám budou shlukovat a pokud utvoří přímkou v histogramu budou obrazy dokonale překryté. Kriteriační funkce:

$$H = \sum_{j,k} PDF[j,k] \log PDF[j,k] \quad (1.23)$$

1.4.6. Maximalizace vzájemné informace (MI)

Tato metoda odhaduje obecnou závislost dvou souborů dat, může se tedy použít pro měření podobnosti dvou obrazů, pořízených různými zařízeními. Oproti PDF, která je jen pro místa obrazů, které se překrývají, nám MI normalizuje histogramy vzhledem k částečným neurčitostem a nelinearitám obrazů. Minimalizací funkce $MI(\mathbf{R}, \mathbf{L}')$ najdeme transformaci t :

1. Vypočti $PDF[j,k]$ obrázků \mathbf{R}, \mathbf{L}'
2. Vypočti jejich vzájemnou entropii $H(\mathbf{R}, \mathbf{L}')$ pomocí (1.5.5)
3. Vypočti okrajové entropie $H(\mathbf{R})$ a $H(\mathbf{L}')$:

$$H(R) = \sum_j (\sum_k PDF[j,k] \log \sum_l [j,l]) \quad (1.24)$$

$$H(L') = \sum_k (\sum_i PDF[i,k] \log \sum_j [j,k]) \quad (1.25)$$

4. Vyhodnot' vzájemnou informaci $MI(\mathbf{R}, \mathbf{L}')$: $MI(R, L') = H(R) + H(L') - H(R, L')$ (1.26)

1.5.7. Normalizovaná vzájemná informace

Normalizovaná vzájemná informace vznikla, protože MI nedokáže vyřešit všechny registrační problémy, zvláště ne ty, kde se v obrazu vyskytuje mnoho šumu.

Spočívá pouze v náhradě funkce $MI(\mathbf{R}, \mathbf{L}')$ funkcí $NMI(\mathbf{R}, \mathbf{L}')$.

$$NMI(R, L') = \frac{H(R) + H(L')}{H(R, L')} \quad (1.27)$$

1.5. Optimalizační metody

1.5.1. Role optimalizační metody

Hlavním úkolem optimalizačních metod je vyhledání vhodné geometrické transformace, aby došlo ke slícování vstupního a referenčního obrazu. Tento algoritmus je nadřazený všem výše uvedeným metodám, proto je volba správné optimalizační metody velice důležitá pro správné vyřešení našeho problému.

Optimalizační metody dělíme do dvou skupin. Na deterministické a stochastické. Deterministické vychází z hledání minim funkcí s využitím prvních a druhých derivací. Jejich nevýhoda je v možnosti použití pouze na spojitě hladké funkce a to, že ve většině případů najdou pouze lokální řešení. Stochastické mají za úkol zavést do procesu náhodné jevy, které nám odstraní nedostatky deterministických metod. Další metoda počítá s přijmutím horšího řešení, což v důsledku zvýšení kritériální funkce vede k přesunu od lokálního minima ke globálnímu. Tato varianta může oproti dvou předchozím pracovat s nespojitou funkcí.

1.5.2. Deterministické algoritmy

1.5.2.1. Newtonova metoda

Tato metoda vychází z toho, že víme, že v bodech extrémů vyšetřované funkce má první derivace hodnotu nula. Pak nám stačí jen najít bod jehož první derivace se rovná nule a druhou derivací zjistit zda se jedná o minimum. Iterační vzorec:

$$x_{k+1} = x_k - H(x_k)^{-1} \nabla f(x_k), \quad (1.28)$$

kde $H(x_k)^{-1}$ je inverzní Hessova matice (hessián) optimalizované funkce pro bod x_k a $\nabla f(x_k)$ je gradient v bodě x_k .

1.5.2.2. Quasi-Newtonova metoda s algoritmem BFGS

Jelikož je počítání hessiánu v každém kroku nereálné, byly vyvinuty quasi newtonovské metody, které pomocí informací o křivost funkce v jednotlivých bodech, aproximují hessián a určí směr kde má být nalezeno minimum. Řídí se tímto vzorcem:

$$x_{k+1} = x_k + \alpha * d_k, \quad (1.29)$$

kde α je délka iteračního kroku daná Wolfovy podmínkami a d_k je směr optimalizace.

Otázkou zůstává, jak nalézt inverzní hessián figurující ve směru hledání minima:

$$d_k = -H(x_k)^{-1} \nabla f(x_k), \quad (1.30)$$

Pro vyhledání aproximace hessiánu, bylo tedy vyvinuto spousta metod. Jednou z nich je metoda BFGS (zkratka vznikla dle autorů metody: Broyden, Fletcher, Goldfarb, Shanno):

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k}, \text{ kde } s_k = x_{k+1} - x_k, q_k = \nabla f(x_{k+1}) - \nabla f(x_k) \quad (1.31)$$

1.5.3. Stochastické algoritmy

1.5.3.1. Simulované žhání

Simulované žhání je fyzikální jev, který popisuje tvorbu krystalické mřížky v materiálu, který byl zahřátý na vysokou teplotu a pak pomalu chladne. Mřížka se formuje do stavu s co nejmenší energií, pokud by však materiál byl zchlazován rychleji dojde k defektům při tvorbě mřížky.

Optimalizovaná funkce $f(x)$ prezentuje energii $E=f(x)$, vektor x odpovídá momentálnímu stavu a T je teplota systému. Funkcí $x_n = G(x, T)$ je náhodně generován nový stav systému. Funkci $h(\Delta E, T)$ pojmenujme jako funkci pravděpodobnosti přijetí nového řešení:

$$h(\Delta E, T) = \exp\left(-\frac{\Delta E}{T}\right) \quad (1.32)$$

kde ΔE je rozdíl energií, $\Delta E = f(\mathbf{x}) - f(\mathbf{x}_n)$, $f(\mathbf{x})$ je hodnota energie předchozího stavu a $f(\mathbf{x}_n)$ je hodnota současného stavu.

Počet náhodných generování nového stavu označíme N_T , a začneme zchlazování dle ochlazovacího plánu, který může mít tvar:

$$T_n = \alpha T$$

kde α je tzv. koeficient chlazení.

Algoritmus vypadá takto [15]:

Inicializace: Definuj počáteční teplotu T_0 , koncovou teplotu T_{min} a počet náhodných prohledávání N_T , $n = 0$, $T = T_0$.

1. krok: Vygeneruj nový stav \mathbf{x}_n pomocí $G(\mathbf{x}_n, T)$
2. krok: Přijmi nové řešení s pravděpodobností vyčíslenou pomocí $h(\Delta E, T)$. Pokud $h(\Delta E, T) > 1$, vždy přijmi nové řešení a $\mathbf{x} = \mathbf{x}_n$.
3. krok: Přiřaď $n = n + 1$. Pokud $n < N_T$ skoč na krok 1.
4. krok: Sniž teplotu T dle chladícího schématu.
5. krok: Pokud $T > T_{min}$ opakuj algoritmus od kroku 1. Přiřaď $n = 0$.

Výhodou algoritmu je velká robustnost a snadná aplikace. Nevýhodou je složité zadávání vstupních parametrů.

2. MULTIMEDIÁLNÍ INTERNETOVÁ KNIHOVNA

Multimediální internetová knihovna bude vlastně informační systém umístěný na internetu. Se strukturou zaměřenou na prezentování druhů registrace obrazů a algoritmů, které jsou v této práci probrány výše. *IS* bude obsahovat popisy, formální vyjádření, příslušné algoritmy a grafické přílohy jednotlivých ověřených přístupů.

2.1. Struktura IS

Informační systém se skládá ze tří hlavních částí kterými jsou: databáze, server a klient.

2.1.1. Databáze

Pod pojmem databáze myslíme určitou zpracovanou množinu informací uloženou na nějakém paměťovém médiu, která tyto data poskytuje další částem systému. Do databáze se informace vkládají pomocí programovacího jazyka, například *PHP*.

2.1.2. Server

Je v struktuře IS prostřední část, která plní požadavky klienta tím, že vyhledává a získává informace z databáze a poté je posílá klientovi.

2.1.3. Klient

Je softwarová aplikace nainstalovaná na uživatelově počítači, která uživateli prezentuje data, které jí zaslal server.

2.2. Používané technologie

Server, databáze i klient používají různé technologie, které jsou zde popsány.

2.2.1. Technologie serveru

2.2.1.1. PHP

PHP je skriptovací programovací jazyk, pro programování dynamických internetových stránek. Tyto skripty jsou vytvářeny přímo na serveru a výsledek jejich činnosti vidí uživatel v prohlížeči. Je hojně využívaný díky jednoduchosti použití a tomu že vznikl jako kombinace několika jiných programovacích jazyků. Funguje na mnoha operačních systémech.

PHP umí ukládat, měnit a mazat data. Vše se odehrává na webovém serveru (kde jsou uloženy zdrojové kódy webových stránek). *PHP skript* se nejprve provede na serveru a potom odešle prohlížeči pouze výsledek. Rozdíl mezi *PHP* a například *JavaScriptem* je ten, že *PHP* vše zpracuje na serveru a výsledek odešle prohlížeči, oproti tomu *JavaScript* vše provádí přímo v prohlížeči. Zdrojový kód *PHP* narozdíl od *JavaScriptu* a *HTML* nelze zobrazit.

Pomocí *PHP* je možné vytvořit diskuzní fórum, knihu návštěv, počítadlo, anketu, graf a další úpravy obsahu webu. Je zde možnost propojit stránky s databázemi, např. *MySQL*.

V *PHP* lze pro web vytvořit šablona s definovaným formátováním. Pokud je pak třeba změnit formátování webu, stačí provést změnu pouze v *PHP skriptu* a změny, například nadpisů a patiček se změní na celém webu najednou.

Příkazy *PHP* jsou vkládány přímo do *HTML* kódu a jsou od něj odděleny tagy `<? a ?>` nebo `<?php a ?>`.

2.2.1.2. Apache

Apache je softwarový webový server. Je multiplatformní, takže s ním mohou pracovat *Linux*, *Microsoft Windows* a mnoho dalších systémů. Dodává internetové stránky prohlížečům na celém světě.

V současné době má *Apache* mezi internetovými servery více než 50% podíl. Ve standardní konfiguraci podporuje několik základních vlastností, které jsou nezbytné pro bezproblémový běh *www* serveru.

Apache je ideální pro tvorbu dynamických stránek, protože podporuje různé druhy programovacích jazyků. Výhodou je, že tyto jazyky není nutné mít na svém počítači vůbec nainstalované, ale stačí, nahrát příslušný modul, který podporuje právě požadovaný programovací jazyk. Pokud se má na serveru používat například *PHP*, stačí pouze stáhnout patřičný modul namísto celé *PHP* instalace. Výhodou je, že funkčnost je stejná, ale modul je mnohem menší.

2.2.2. Technologie databáze

2.2.2.1. MySQL

MySQL je multiplatformní databáze, která je k dispozici bezplatně pod licencí *GPL* nebo pod placenou komerční licencí. Jelikož je multiplatformní, funguje na *Linux*, *Microsoft Windows* a dalších operačních systémech. V současné době je využívána ve velkém množství díky jejímu volnému bezplatnému šíření.

Do *MySQL* lze ukládat různá data (texty, obrázky atd.), s nimiž lze dále jednoduše pracovat, třídit, řadit, filtrovat a podobně. Nejčastěji se *MySQL* používá ve spojení s jazykem *PHP*, které umožňuje přístup k uloženým datům. Každá databáze v *MySQL* obsahuje tabulky, každá tabulka má sloupce a řádky – v každém řádku jsou záznamy předem určeného typu.

Pro jednoduchou správu *MySQL* databází přes webové rozhraní se používá nástroj *PhpMyAdmin*. *PhpMyAdmin* je volně šiřitelný program napsaný v *PHP*, který umožňuje zálohování, vytváření tabulek, vkládání, editaci a mazání záznamů v tabulkách, vytváření databází a další úkony.

MySQL tvoří s serverovým programem *Apache* a skriptováním *PHP* tzv. triádu, trojici programů nejčastěji instalovanou k vytváření databázových aplikací. *MySQL* se také díky své relativní jednoduchosti poměrně snadno učí.

Nevýhody *MySQL* pramení z jejích výhod. Nepodporuje složitější programátorské konstrukce a nemá dostatečný výkon v opravdu náročných (zatěžovaných) webových aplikacích. Tehdy se používají konkurenční databáze, například *PostgreSQL* nebo *Oracle*.

2.2.3. Technologie klienta

2.2.3.1. HTML

HTML (HyperText Markup Language) je značkovací jazyk pro hypertext. *HTML* je v dnešní době stále nejoblíbenější jazyk pro vytváření jednoduchých *www* stránek. Značky (*tagy*) jazyka se dělí na popisné a stylistické. Popisné nám popisují povahu obsahu a stylistické nám vytvářejí vzhled.

Pojem značkovací jazyk znamená, že s jeho pomocí označujeme části textu tak aby bylo prohlížeči jasné, co mu sdělujeme. Značky se uzavírají do špičatých závorek. *Tagy* dělíme také na párové a nepárové. Párové mají počáteční a konečnou značku, přičemž konečná se od počáteční liší přidáním lomítka, například: `<p>text</p>`, tento párový tag vymezuje odstavec. Nepárový tag je například `
`, který zalomí řádek .

HTML formátování lze použít i v jiných operačních systémech (*Linux*, *MacOs*). Aby internetový prohlížeč poznal, že se jedná o internetovou stránku (*HTML* dokument), bývají uloženy s příponou *htm*, nebo *html*.

2.2.3.2. CSS

CSS (tabulky kaskádových stylů) je jazyk sloužící pro popis zobrazení stránek napsaných například v *HTML*. Jejich smysl je v oddělení vzhledu od obsahu, což v *HTML* nejde.

CSS umožňuje definovat vlastnosti určitých elementů např. barvy textu, pozadí, zarovnání, velikost písma a mnoho dalších vlastností, které nelze nahradit ničím v *html*. V *CSS* lze jednoduše připravit stránku se stejnými nadpisy, odstavci, seznamy atd. *CSS* se může zapisovat přímo do *HTML* dokumentu, nebo do externího souboru s příponou *.css*. Výhodou je, že lze připojit jeden styl uložený v **.css* souboru k více stránkám.

Pro tvorbu *CSS* stačí jednoduchý textový editor, a ovládání jazyka *HTML*.

2.3. Redakční systémy

Redakční systémy umožňují kompletní správu obsahu webu. Základní funkcí redakčních systémů je publikování textů. Redakční systémy obsahují, nebo lze do nich doinstalovat celou řadu doplňků, např. diskuzní fóra, fotogalerie, internetový obchod atd.

2.3.1. Drupal 6

Drupal je open source (volně dostupný) redakční systém, který staví na několika základech, které jsou důležité pro jeho fungování a vývoj. *Drupal* umožňuje díky svému modulárnímu systému vytvořit blog, e-shop, fórum, korporátní web. *Drupal* umožňuje vytvořit malé, ale stabilní a rychlé jádro s dobrým rozhraním a moduly, na kterých staví. Každý uživatel si může vytvořit svůj vlastní modul. Seznam modulů pro stažení, je udržován na domovské stránce *Drupalu*.

Do jádra *Drupalu* se nedostávají neověřené patche, protože jeho úprava není volně přístupná, ale je administrována tvůrci tohoto systému. Jádro má rovněž velmi dobře navrženou strukturu. To z něj dělá bezpečný a stabilní systém. *Drupal* je open source – tedy

volně šiřitelný, podporuje *PHP* programovací jazyk. Pracuje s databázemi *MySQL* a *PostgreSQL*, v přípravě je také podpora pro náročnější systémy *MS SQL* a *Oracle*.

2.3.2. Joomla 1.5

Joomla je profesionální redakční systém určený pro kompletní správu webového obsahu. Obsahuje velké množství pluginů a dostupných rozšíření, podporuje změny vzhledu, blogy, diskuzní fóra, hlasování, kalendář, apod. *Joomla* disponuje kvalitní základnou vývojářů a přispěvatelů, je licencována pod *GNU GPL* a tedy šiřitelná zdarma.

2.3.3. WordPress

WordPress je redakční systém zaměřený převážně na tvorbu osobních stránek a blogů. Mezi jeho hlavní přednosti patří snadné ovládání a správa, výtečná rozšiřitelnost a kvalita výstupu.

WordPress je vyvíjen jako open source projekt, celý zdrojový kód je popsán v dokumentaci a dává tak možnost každému programátorovi se zapojit do jeho vývoje, upravovat jeho části a přizpůsobit si ho podle vlastních představ, lze ho tak použít téměř pro cokoli. Toho také využívají stovky lidí a podílí se na vývoji *WordPress*, čímž ho staví před komerční produkty, které tvoří jen pár jedinců a nikoliv široká veřejnost.

2.3.4. PhpRS

PhpRS je český redakční a publikační systém. Je naprogramován v *PHP* a využívá databázi *MySQL*. Má spoustu vlastností: publikování článků, vytváření anket a novinek, správu redaktorů, jednoduchou úpravu vzhledu, vyhledávání, management reklam, rozšiřitelnost pluginy, zálohování a další zajímavé vlastnosti.

PhpRS je distribuován pod licencí *GNU GPL*. Nechybí podpora pluginů, layoutů, správa uživatelů, kdy se rozlišují typy uživatelů a jejich přístupová práva. Práva lze přidělit i pro jednotlivé moduly. Pro články je připraveno několik zobrazovacích šablon a nastavení (rubrika, typ článku, autor, datum vydání, související články, přidělená anketa apod.). Tento redakční systém má velké možnosti jak ve vzhledu, tak ve funkčnosti.

3. TVORBA STRÁNEK

Pro tvorbu stránek byl z vyjmenovaných redakčních systémů vybrán *Drupal* ve verzi 6, kvůli jeho stabilitě a dobré rozšiřitelnosti prostřednictvím velkého množství modulů. Jako server byl vybrán program *WampServer*, který pracuje na *Apache* a je kompatibilní s *Drupalem*. Zprovoznění těchto programů je rozebráno v kapitole 4.

3.1. Tvorba vzhledu stránek

Tvorba stránek začala nastavením vzhledu. Na domovských internetových stránkách *Drupalu*: <http://drupal.org/project/themes> jsou ke stažení stovky témat vzhledu od autorů *Drupalu* i obyčejných uživatelů. V tomto případě bylo ale vybráno jedno ze základních témat vzhledu, které se pak v souborech *style.css* a *page.tpl.php* upravovalo. Měnilo se barevné rozvržení změnou HTML kódů jednotlivých barev pozadí, písma, velikost stránky a její adaptace na velikost prohlížeče, se nastavila pro plnou adaptaci na velikost a rozlišení prohlížeče uživatele. Dále se upravil druh písma, jeho velikost a barva, přidalo se logo FEKT VUT a stejné logo se přidalo i do miniatury, která se zobrazuje vedle adresy stránky v prohlížeči.

3.2. Tvorba obsahu stránek

Další částí po vytvoření vzhledu byla tvorba obsahu. K tvorbě obsahu byl využit modul *Book*, který je jeden ze základních modulů jádra *Drupalu*. Po jeho aktivování přibude možnost vkládat speciální typ článku *Book page* = *Stránka knihy*. Tyto jednotlivé stránky jsou pak součástí velké knihy, do které jsou přiřazeny.

Book page se vkládají přes menu administrace -> *Vytvořit obsah* -> *Book page*. Po kliknutí na tuto položku v menu se zobrazí v prohlížeči tvorba příspěvku *Book page*.

The screenshot shows the Drupal administration interface for creating a new *Book page*. On the left, there is a sidebar with two main sections: 'Navigace na webu' (Navigation on the website) containing 'Teorie registrace obrazů' (Theory of image registration), and 'Admin menu' (Admin menu) containing 'Můj účet' (My account), 'Poslední příspěvky' (Last posts), 'Vytvořit obsah' (Create content) with sub-items 'Anketa' (Survey), 'Book page', 'Stránka' (Page), and 'Článek' (Article), 'Agregátor zdrojů' (Source aggregator), 'Administrace' (Administration), and 'Odhlásit' (Logout). The main content area is titled 'Domů > Vytvořit obsah' (Home > Create content) and 'Vytvořit Book page' (Create Book page). It features a 'Nadpis:' (Title) field with a red asterisk, a 'Nastavení menu' (Menu settings) section, a large 'Text' area for the page content, and a 'Formát vstupu' (Input format) dropdown menu. At the bottom, there are links for 'Osnova knihy' (Book outline) and 'Informace o revizi' (Revision information).

Obrázek 15 – úvodní strana při vytváření book page

Zde lze vyplnit nadpis, a pak vlastní text stránky. Tento text může být formátován pomocí *HTML* nebo *PHP*. Pod textovým polem, kde se text zadává je položka „Formát vstupu“. Zde si zvolíme v jakém formátování jsme text vyplňovali. V práci je text většinou formátován v *HTML*, a proto jako formát vstupu byl vybrán „Full HTML“, kde jsou povoleny všechny *HTML* značky. Další položkou je „Osnova knihy“. Po kliknutí na ni se rozbalí další nabídka, kde první položkou je *Knih*a. Zde můžeme vybrat <žádné>, <vytvořit novou knihu> nebo příspěvek zařadit do již existující knihy. Pokud bychom vybrali <vytvořit novou knihu> byl by výše vytvořený text a jeho nadpis první stránkou nově vytvořené knihy. Pokud se vybere již existující kniha, načte se nové pole, kde je nutno vybrat nadřazenou položku nového příspěvku v této knize.

Obrázek 16 – zařazení příspěvku do knihy

Řazení příspěvků probíhá pomocí váhy. Uvedu příklad: pro příspěvek 1.1. Posunutí obrazu je nadřazenou položkou 1. Geometrické transformace 2D, stejně tak je tato položka nadřazená i pro další příspěvky 1.2., 1.3. atd. Jejich řazení mezi sebou je pak dáno metodou vah, kdy příspěvek 1.1. má váhu -15, příspěvek 1.2. má váhu -14 atd. Nejvýše jsou tedy řazeny příspěvky s nejnižšími hodnotami vah a ty s vyššími následují po nich. Pokud jsou některým stránkám zadány stejné váhy, řazení se provede podle abecedního pořadí jejich názvů. Rozsah hodnot vah je <-15,15>. Jedna položka tedy může mít maximálně 31 podřazených příspěvků. Tímto váhováním tedy byla vytvořena hierarchistická struktura stránek a přes *HTML* kódy se pak vkládal jejich obsah. K vytvoření textu a vložení obrázků bylo použito několik tagů. Vzorce a matice pro jednotlivé metody a algoritmy byly také vkládány formou obrázků, protože tvorba tohoto formátování je v *HTML* prakticky nemožná a nepříliš estetická.

Formátování textu:

<p style="text-align: justify">"text"</p>, kde

<p> </p>	je párový tag, který vymezuje odstavec
<style>	je stylizace daného odstavce
<text-align>	určuje, že se bude text zarovnávat
<justify>	určuje, jaké zarovnání se provede, zde zarovnání do bloku

****, kde

 	párový tag, text mezi značkami bude tučně zvýrazněn
 	párový tag, vymezuje text, který bude formátován
<style>	stylizace textu umístěného mezi tagy a
<color>	mění barvu stylizovaného textu

Vkládání obrázků:

``, kde

<code></code>	tímto tagem se volí cesta k obrázku, který vkládáme
<code><align></code>	nastavení obtékání textu kolem obrázku nebo jeho umístění na řádku
<code><alt></code>	popis obrázku, v některých prohlížečích se zobrazí po přejetí myší
<code><width a height></code>	zadávání šířky a výšky obrázku, v pixelech

Další použité značky:

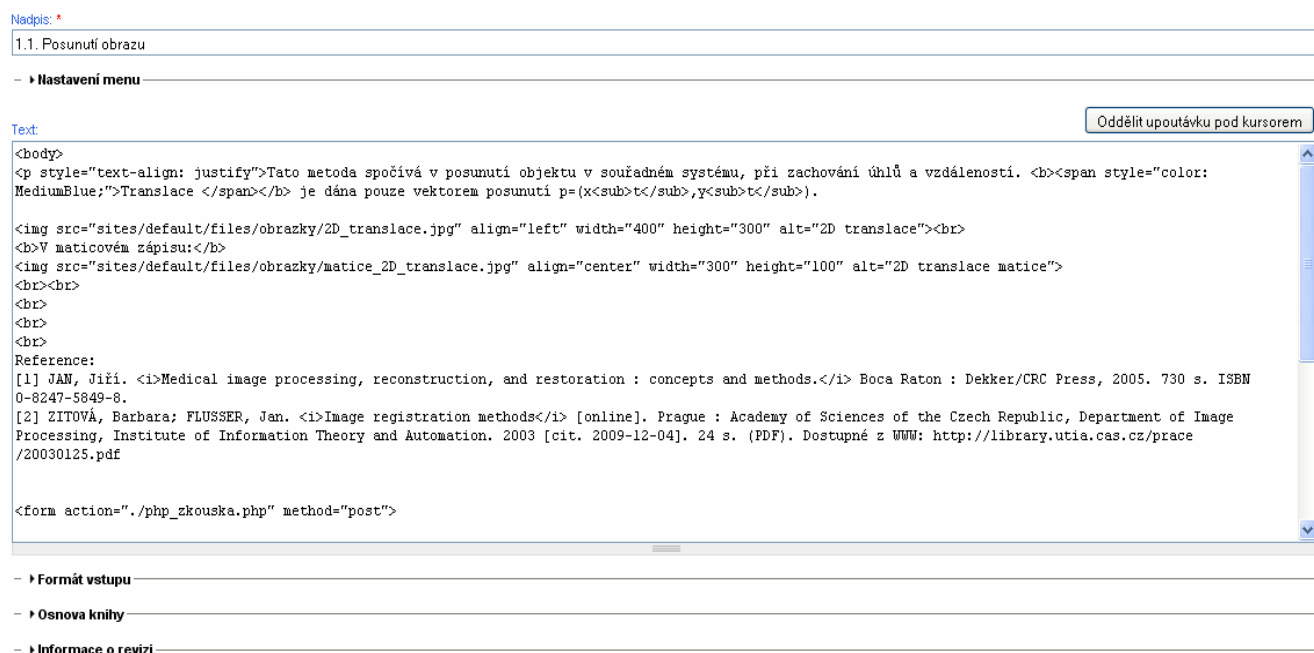
<code><i> </i></code>	párový tag, text mezi značkami bude kurzívou
<code>
</code>	zalomení řádku

Symbols:

Symbols se v HTML vkládají ve formě kódů.

delta Δ má v *HTML* kód: `Δ`

alfa α má v *HTML* kód `α`



Obrázek 17 – vzhled formátování příspěvku v HTML

3.3. Tvorba algoritmů

Příklady funkčních algoritmů byly namyšleny jako funkce, přeložené do *exe* souborů a následně přenesené do internetových stránek. Pro tuto realizaci bylo nutné vytvořit v programu *Matlab* funkce jednotlivých metod používaných v registraci obrazů, které se následně překládaly do *exe* souborů. Další potřebnou částí bylo vytvoření skriptu v programovacím jazyku *php*, který slouží ke zpracování *exe* souborů. Třetí částí byly tlačítka na internetových stránkách sloužící uživateli k výběru vytvořených *exe* souborů a spuštění jejich zpracování.

3.3.1 Tvorba php skriptu

Vytvořený skript se nachází ve složce *C:\wamp\www\drupal* a je pojmenovaný *php_zkouska.php*, lze jej otevřít v textovém editoru pro zobrazení kódu. Jak je zmíněné výše v kapitole o *php*, je programování v tomto jazyce propojeno s *HTML*, což je vidět i ve vytvořeném skriptu. *Php* může být vnořeno do *HTML*, nebo i opačně.

Skript je založen na funkci *exec()*, která slouží právě pro zpracování *exe* souborů.

```
// exec - spustí program a uloží výstup
if(function_exists('exec'))

    @exec($cfe,$res);
    return $res;
```

Další důležitý a v *php* nejvíce používaný příkaz je *echo*, který slouží například k volání proměnných. Přes tento příkaz jsou na stránce s konečným výstupem *exe* souboru zobrazeny popisy:

<code>echo "Byl vybrán soubor:";</code>	- vypíše tento text na stránku
<code>echo \$_POST['path'];</code>	- vypíše přesnou adresu v počítači odkud byl vybrán spuštěný <i>exe</i> soubor
<code>echo "Vysledek nasi fce:";</code>	- vypíše tento text na stránku
<code>echo \$src;</code>	- zobrazí výstupní obrázek spuštěného <i>exe</i> souboru

3.3.2. Tvorba exe souborů

Základem *exe* souborů bylo vytvoření funkcí pro registraci obrazů v programu *Matlab*. Ty byly vytvořeny z geometrické transformace a podobnostní funkce. Byly vytvořeny pro translaci, změnu měřítka a afinní transformaci, vše ve 2D. Vykreslovací plocha obrázku byla rozdělena do šachovnice po stejně velkých čtvercích:

```
ctverec_velikost = 8;
sach = checkerboard(ctverec_velikost,256/(ctverec_velikost*2));
sach (sach>0.5)=1;
sach2=1-sach;
```

```
orig_sach=orig.*sach;
orig_vec=orig(:);
```

Pro každou z funkcí byly nadefinovány vstupní parametry, pro posun a měřítko dva, pro afinní čtyři:

```
parametry = [0 %translate v ose X
              0]; %translate v ose Y
```

Těmto počtům odpovídaly i počty kroků změn jednotlivých parametrů:

```
krok = [1 %krok translate v ose X
        1]; %krok translate v ose Y
```

Poté následoval výpočet(u *afinní*) nebo definice(*translate* a *měřítko*) kombinací, kdy každá proměnná mohla v každém kroku dosáhnout hodnoty 0, 1, -1 a všechny tyto kombinace uložit do jedné matice:

```
kombinace = [ 0 0 1 1 -1 0 -1 -1 1
              0 1 0 1 0 -1 -1 1 -1];
```

,zde pro translaci tři hodnoty pro dva parametry: $3^2 = 9$ možností, bylo tedy lehké je vypsát ručně, pro afinní $3^4 = 81$ se musela matice vytvořit v *Matlabu*.

V dalším cyklu, bylo kritériem $\cos_krit \leq 0.99$ tedy dokud bude \cos_krit menší než 0.99 bude cyklus probíhat. V tomto cyklu se vypočítávaly nové parametry registrovaného obrazu prostřednictvím *Matlab* funkce *maketform*, která může mít transformační typy například: *afinní*, *projektivní* či *volitelnou*. Pro všechny skripty byla použita *afinní*, u které byla upravována transformační matice dle potřeby. Například pro translaci vynecháním zkosení a změny měřítka:

```
while cos_krit <= 0.99
for i=1:9

    parametry_nove = parametry + krok.*kombinace(:,i);

    % affini transformace
    transform = maketform('affine',[1 0 0;...
                                    0 1 0;...
                                    parametry_nove(1) parametry_nove(2) 1]);

    reg_transformed=imtransform(reg,transform,'XData',[1 size(orig,2)],'YData',[1
size(orig,1)]);
    reg_transformed_vec = reg_transformed(:);

    cos_krit(i,:)=(sum(reg_transformed_vec.*orig_vec))/(sqrt(sum(reg_tran
sformed_vec.^2))*sqrt(sum(orig_vec.^2)));

end
```

V závěru se transformované parametry spojily s základní šachovnicí do výstupní proměnné:

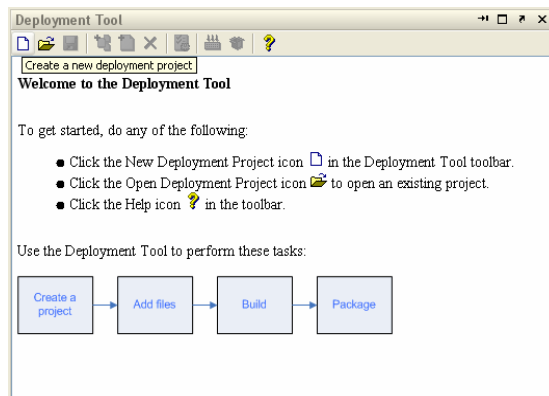
```
translace_vystup = reg_transformed_sach+orig_sach;
```

Ta byla přes příkaz *imwrite* uložena do *.jpg* obrázku na určenou adresu v počítači, kvůli zobrazení *php* skriptem:

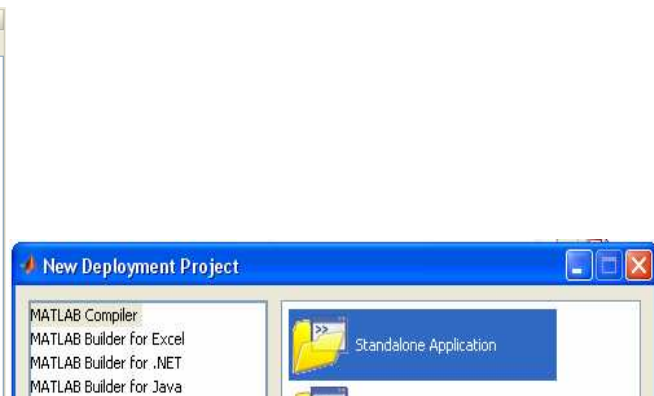
```
% nazev pod jakym se vystupni obraz ulozi
imageName = 'translace_finalni.jpg';

% nasmerovani ulozeni vystupniho obrazu a nastaveni cesty pro zpracovani
% php skriptem
cestaPC = strcat('C:\wamp\www\drupal\img\' , imageName);
cestaWeb = strcat('img/' , imageName);
imwrite(translace_finalni,cestaPC);
fprintf('%s', cestaWeb);
```

Překlad *m-file* skriptů *Matlabu* probíhal přes funkci *deploytool*, která je součástí základní instalace *Matlabu*. V příkazovém okně se odesláním příkazu *deploytool* otevřelo nové okno, kde se vybral nový *projekt*, zadal název a jako typ vybrala samostatná aplikace.

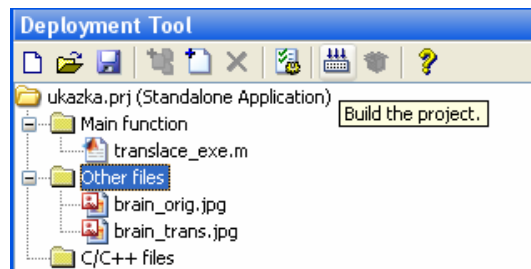


Obrázek 18



Obrázek 19

Dále se znovu otevřelo okno *deploytool*, kde se do položky „Main function“ zařadil *m-file* s funkcí registrace obrazu a do položky „Other files“ všechny obrázky potřebné pro funkci, či další, pro funkci důležité, soubory. Pak už jen kliknout na „Build the project“ a *Matlab* začal sám z dostupných dat vytvářet *exe* soubor.



Obrázek 20 – prostředí *deploytool*

3.3.3. Tvorba tlačítek na stránkách

Tlačítka byly na stránkách vytvořeny *HTML* kódováním. Konkrétně *tagem* `<form>`, který slouží pro vytváření *formulářů*:

```
<form action="./php_zkouska.php" method="post">
```

- začátek formuláře a odkázání na *php skript*

```
<input type="file" name="path"
onchange="document.getElementById('copy').value = this.value"/>
```

- tlačítko pro vyhledávání *exe* souborů na disku

```
<input type="text" id="copy" value="" style="display:none;">
```

- pomocné textové pole odkud čte *php skript* cestu

```
<input type="submit" value="Zpracuj" name="neco">
```

- tlačítko pro spuštění zpracování *exe* souboru

```
</form>
```

- ukončení formuláře



Obrázek 21 – vytvořená tlačítka

4.1. Manuál k zprovoznění internetové knihovny

1. Nejprve se musí vložit CD do počítače a otevřít jeho obsah. Doporučuji složku ve formátu zip rozbalit a její obsah uložit například na plochu operačního systému, kvůli další manipulaci s těmito soubory.
2. **Instalace WampServeru:**

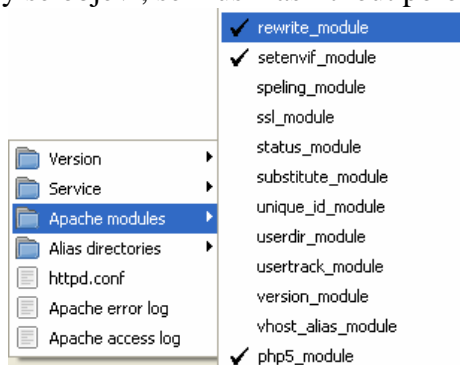
Soubor *WampServer2.0h.exe* se musí zkopírovat na uživatelskou plochu OS a spustit. V 1. kroku se spustí instalace tohoto programu. V 2. kroku se potvrdí licenční podmínky programu, v 3. kroku se musí vybrat složka, kam se program nainstaluje. Nechá se původní (standardně by se mělo objevit *c:\wamp*). V dalším kroku se vytvoří, pokud uživatel chce, ikona programu na ploše a ikona v položce snadné spuštění (v *OS Microsofti Windows* vlevo na liště, vedle nabídky Start). V posledním kroku kliknutím na *Install* se program nainstaluje. Pokud je v počítači nainstalovaná *Mozilla Firefox*, program se po instalaci zeptá, zda má být *Mozilla Firefox* použita jako prohlížeč pro zobrazení s tímto programem. Odsouhlasí se ano. Posledním krokem se nastaví *PHP* e-mail adresa, nechá se přednastavená, není nutné ji vyplňovat. V posledním okně kliknutím na *Finish* se zakončí instalace a spustí *WampServer*.
3. Spuštěný program se ukáže vpravo na liště ve spuštěných aplikacích vedle hodin OS *Windows*. Jeho ikona se podobá polovině ciferníku analogových hodin.



Obrázek 22 – v pořádku spuštěný Wampserver

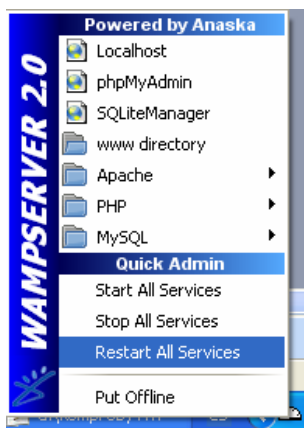
4. Instalace Drupalu:

- Celá složka souboru *drupal-6.16* se musí zkopírovat do složky *C:\wamp\www*. Aby byl Drupal v češtině, musí se otevřít složka *drupal_čeština* a veškerý její obsah (zkopírovat pouze obsah ne celou složku, tedy soubory *modules*, *profiles*, *themes* a txt soubory: *LICENSE.cs*, *STATUS.cs*) zkopírovat do složky *C:\wamp\www\drupal-6.16*. Windows otevře okno přepisování souborů a zvolením *přepsání všech* proběhne přepis anglických souborů na soubory přeložené do češtiny.
5. Pokračuje se kliknutím na spuštěný *WampServer* na liště vedle hodin. Po zobrazení nabídky se vybere záložka *Apache*, dále pak *Apache modules* a v následujícím abecedně seřazeném seznamu, který se objeví, se musí zaškrtnout položka *rewrite_module*,



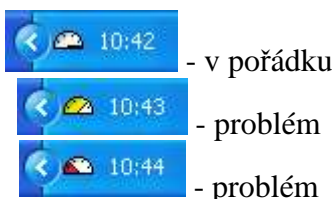
Obrázek 23 – povolení rewrite modulu

kteřá se tím zaktivuje. Tato aktivace je nutná pro část instalace, která bude následovat a pro následné fungování webové aplikace. Znovu kliknutím na ikonu *WampServeru* a zvolením *Restart All services* se tato změna načte.



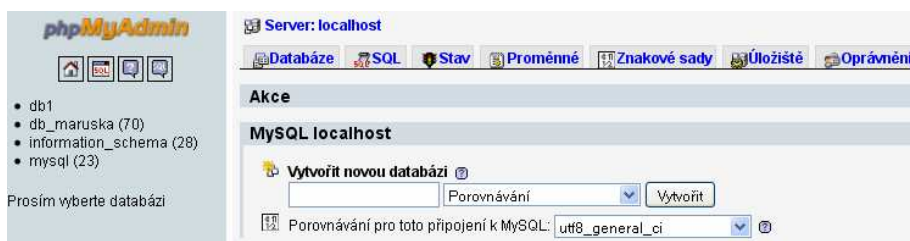
Obrázek 24 – restart Wampserveru

V posledním kroku se klikne na *WampServer* a zvolí *Put Online*. To, zda je server online a běží v pořádku, se zjistí jednak tím, že po najetí kurzorem myši na ikonu se zobrazí „WAMPSEVER – server Online“ a jednak tím, že „ciferník“ ikony je čistě bílý. Pokud se zde vyskytuje jiná barva (červená, žlutá), je někde chyba a server neběží správně (například pokud je zaplý program *Skype*, tak *WampServer* nefunguje, pracují na stejném portu a vzájemně se ruší).



Obrázek 25 – tři stavy Wampserveru, které mohou nastat

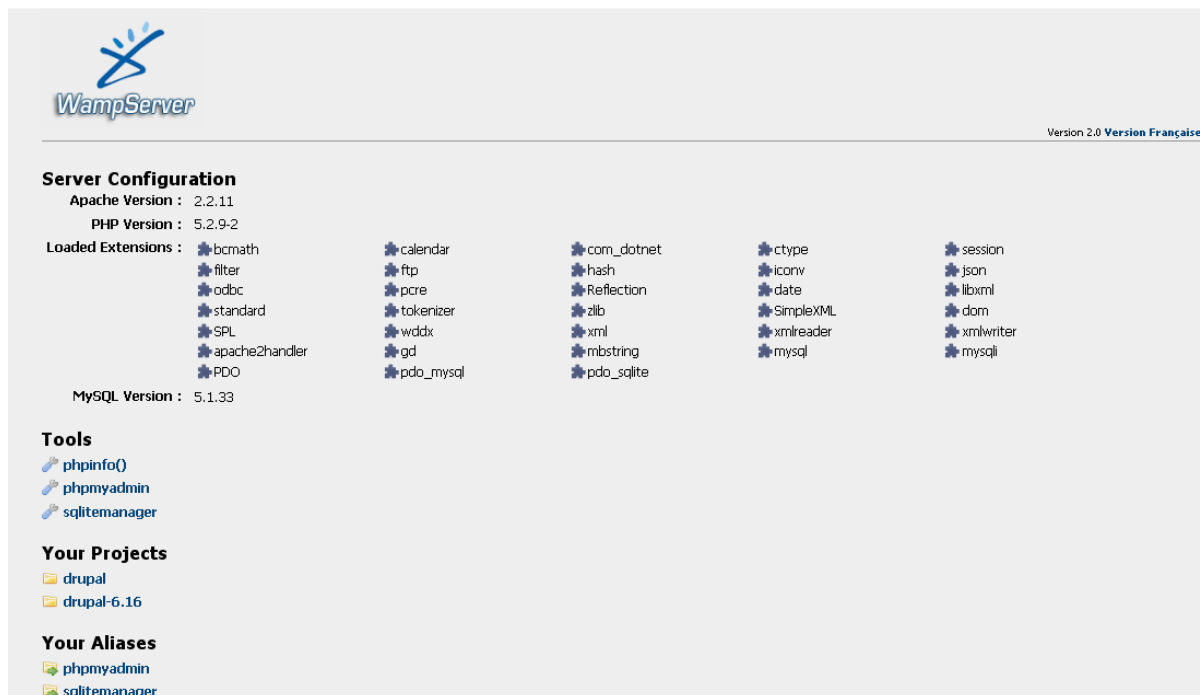
6. Dále je nutno vytvořit databázi. Kliknutím na ikonu *WampServeru* a zvolením *phpMyAdmin* se otevře stránka v prohlížeči (v *Mozilla Firefox*, pokud je nastavena) s adresou <http://localhost/phpmyadmin/>. Nalezením textového pole vlevo s nadpisem „Vytvořit novou databázi“ a napsáním názvu databáze do textového pole, např. „db1“, se vytvoří databáze „db1“.



Obrázek 26 – vytvoření nové databáze v phpMyAdmin

Nová databáze se pak nachází v seznamu vlevo. Kliknutím na „db1“ se databáze otevře v prohlížeči a nahoře na stránce výběrem položky „Oprávnění“ se objeví dvouřádková tabulka s hodnotami: uživatel, počítač, typ, oprávnění, přidělování, akce. Tato stránka se ponechá otevřená, bude ještě k zapotřebí. Opětovným kliknutím na ikonu *WampServeru* a zvolením

„Localhost“ se v prohlížeči otevře další okno, případně ve stejném okně další panel s adresou <http://localhost/> a názvem strany WAMPSEVER Homepage. Vlevo dole na této stránce jsou položky „Tools“, „Your Projects“, „Your Aliases“. V „Your Projects“ (v překladu „Vaše Projekty“) by měla být složka *drupal-6.16*, kliknutím na ni se otevře okno s instalací *Drupalu*.



Obrázek 27 – vzhled adresy <http://localhost/>

7. Prvním krokem je výběr jazyka. Pomocí Select language (zvolit jazyk) se může vybrat požadovaný jazyk, v nabídce se nachází i čeština. V dalším kroku se objeví červená tabulka s chybou, kde se píše problém s nastavením a závislostmi. Odstraní se úpravou souboru „default.settings.php“, který se nachází v adresáři C:\wamp\www\drupal-6.16\sites\default\. Musí se vytvořit kopie tohoto souboru, uložit ve stejném adresáři, přejmenovat na „settings.php“ a následně soubor otevřít (pro práci s *PHP* a *HTML* soubory se při tvorbě této práce nejvíce osvědčily textové editory Notepad++ a PSPad). Otevře se *PHP* soubor, kde vše, co je označeno hvězdičkou (*), je komentář nebo návod pro nějaké úpravy tohoto souboru.

Důležitý je 92. řádek, kde je username:password@localhost/databasename. Tyto hodnoty se musí změnit podle údajů databáze. Ty se nacházejí v otevřeném okně, kde je vytvořená databáze se zvolenou položkou „Oprávnění“ (viz výše). Zde je vidět, že „username“ je „root“, heslo není, takže se nebude vyplňovat, localhost se ponechá (pracuje na lokálním serveru nikoliv na vzdáleném) a nakonec se vyplní název databáze, v tomto případě „db1“. Upravený řádek bude vypadat následovně:

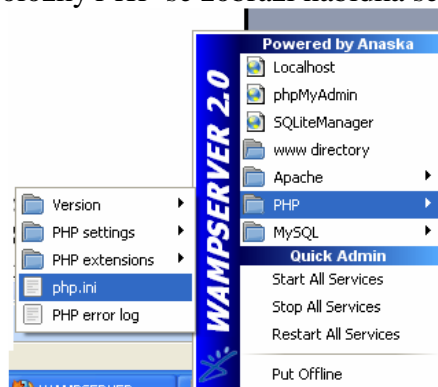
\$db_url = 'mysql://root@localhost/db1';. Soubor se musí uložit a může zavřít.

Následujícím navrácením do okna prohlížeče, kde se instaluje *Drupal*, se musí stránka s červenou tabulkou aktualizovat. Tabulka po aktualizaci zmizí a začne instalace programu a překladu rozhraní. Po automatickém ukončení těchto úkonů následuje krok „Nastavení webu“. Zde si uživatel dle potřeby vyplní uživatelský profil, tedy jméno a heslo (tento profil bude zároveň sloužit jako administrátor), dále email, název stránky (lze dodatečně změnit) a časové pásmo. Velice důležité je nastavení čisté *URL*, díky které může být adresa stránky bez znaku „?q=“. Při správném postupování je možné u čistých

URL zadat „Povoleno“ (dáno hlavně zapnutím *rewrite_module*, viz výše), zadá se tedy „Povoleno“ a uloží. Poté proběhnou různé importy překladů a rozhraní a instalace je kompletní.

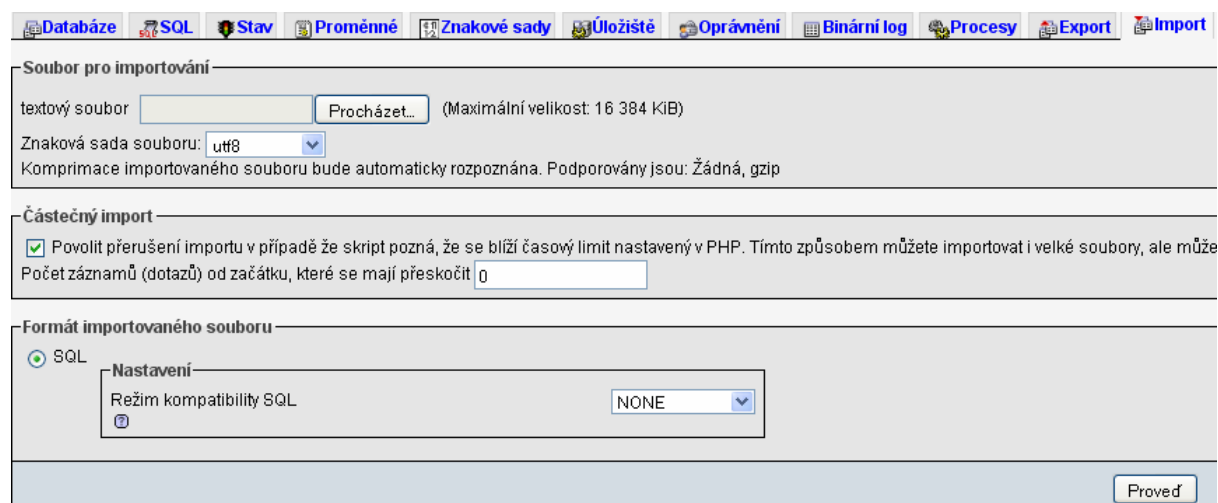
8. Spuštění www stránek

V tomto bodě bude rozebráno, jak spustit stránky, které představují praktickou část bakalářské práce. Nejprve se musí zkopírovat celý soubor *drupal* a soubor *index.php*, oba opět do složky *C:\wamp\www*. Pro soubor *index.php* zobrazí Windows přepisovací upozornění, kde se musí zvolit „přepsat“. Dále se musí zkopírovat databáze, předtím ale je nutné upravit parametr, jak velký soubor je schopen *PHP* přijmout. Kliknutím na ikonu *WampServeru*, nalezením položky *PHP* se zobrazí nabídka se souborem *php.ini*.



Obrázek 28 – přístup k *php.ini*

Soubor se otevře v textovém editoru, ve kterém se musí najít tyto dva řádky: *Maximum size of POST data that PHP will accept* a *post_max_size = 8M* (hledání pomocí CTRL+F a klíčového slova „max“). První řádek říká, jakou maximální velikost datového souboru *PHP* přijme a na druhém řádku se tato hodnota určuje konkrétně. Hodnota 8 Mb je základní, změňte ji například na 16Mb nebo na 32Mb. Druhou podobnou změnu je potřeba udělat na těchto dvou řádcích: *Maximum allowed size for uploaded files* a *upload_max_filesize = 2Mb*. Hodnotu 2Mb také změňte na 16Mb nebo 32Mb. Soubor se musí uložit a zavřít. Tato změna se načte pomocí „Restart All Services“ nacházející se ve *WampServeru*. Ve *WampServeru* se zvolí „phpMyAdmin“, otevře se okno prohlížeče se stránkou, kde se tvořila původní databáze, nyní se ale bude databáze importovat. Po kliknutí na položku „Import“ v horní nabídce na stránce se pak načte stránka, kde se zmáčkne tlačítko „Procházet“.



Obrázek 29 – okno pro import databáze v *phpMyAdmin*

Objeví se okno, ve kterém se musí v rozbalené složce souborů této práce vyhledat soubor *db_maruska.sql*, který se načte přes tlačítko „Otevřít“. Tento soubor je záloha databáze internetových stránek, které jsou vytvořeny. V pravém dolním rohu na této stránce tlačítkem „Proved“ se databáze nahraje a bude se nacházet v seznamu nalevo na stránce. Nyní je instalace dokončena.



Obrázek 30 – seznam vytvořených databází

4.2. Manuál k používání stránek

Stránky se zobrazí po kliknutí na ikonu spuštěného *WampServeru* a v nabídce, která se objeví výběrem „Localhost“. Otevře se okno prohlížeče, kam se může do adresy stránky zadat <http://localhost/drupal/>, nebo na stránce kliknout na složku *drupal* v sekci „Your Projects“.

V prohlížeči se pak otevrou stránky představující praktickou část této bakalářské práce. Na levé straně je blok s nadpisem „Navigace na webu“ a je zde v modrém poli položka „Teorie registrace obrazů“. Po kliknutí na ni, se zobrazí celé menu navigace. Obsah je hierarchicky řazený a hlavní kapitoly jsou *Geometrické transformace 2D*, *Geometrické transformace 3D*, *Interpolace*, *Kritérium podobnosti* a *Optimalizace*. V jednotlivých kapitolách jsou pak příklady metod a algoritmů využívaných při registraci obrazů, jejich stručný popis, grafická příloha, případně metody výpočtu a přepočet souřadnic.

Navigace na webu

- Teorie registrace obrazů
 - 1. Geometrické transformace 2D
 - 1.1. Posunutí obrazu
 - 1.2. Rotace obrazu
 - 1.3. Změna měřítka
 - 1.4. Afinní transformace
 - 1.5. Projektivní transformace
 - 2. Geometrické transformace 3D
 - 3. Interpolace
 - 4. Kritérium podobnosti
 - 5. Optimalizace

Domů > Teorie registrace obrazů > 1. Geometrické transformace 2D

1.1. Posunutí obrazu

Možila Jakub Konarik, Ne, 05/02/2010 - 14:47

Tato metoda spočívá v posunutí objektu v souřadném systému, při zachování úhlů a vzdáleností. **Translace** je dána pouze vektorem posunutí $p=(x_t, y_t)$.

V maticovém zápisu:

$$\begin{bmatrix} x' & y' & w' \end{bmatrix} = \begin{bmatrix} x & y & w \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_t & y_t & 1 \end{bmatrix}$$

Reference:

[1] JAN, Jiří. *Medical image processing, reconstruction, and restoration: concepts and methods*. Boca Raton: Dekker/KRC Press, 2005. 730 s. ISBN 0-8247-5849-8.

[2] ZITOVÁ, Barbara; FLUSSER, Jan. *Image registration methods* [online]. Prague: Academy of Sciences of the Czech Republic, Department of Image Processing, Institute of Information Theory and AI (PDF). Dostupné z WWW: <http://library.utia.cas.cz/prace/20030125.pdf>

Procházet...

Zpracuj

Obrázek 31 – vzhled adresy http://localhost/drupal/2D_posunuti_obrazu

Pro přechod do menu administrace je nutné se přihlásit jako administrátor. V prohlížeči se do adresy stránky zadá <http://localhost/drupal/user/> , v okně se pak otevře přihlášení přes Uživatelské jméno a Heslo.

Ty jsou pro administrátora nastaveny takto:

Uživatelské jméno: Jakub Konarik

Heslo: RegistracE1.

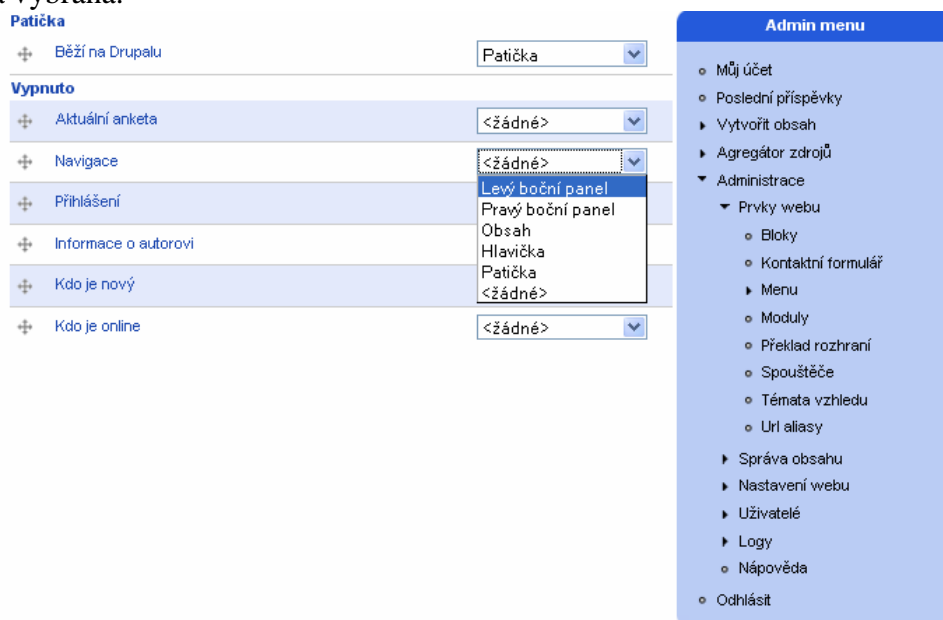
Formát hesla s velkými a malými písmeny, čísla a interpunkcí, je vyžadován *Drupalem* kvůli zabezpečení proti dekódování hesla. Po vyplnění údajů stačí kliknout na „Přihlásit“.

Dalším krokem po přihlášení je v prohlížeči zadání adresy <http://localhost/drupal/admin/>, po kterém se uživatel dostane do administračního menu stránek.

Správa obsahu	Prvky webu
Spravovat obsah webu.	Nastavit vzhled a funkce webu.
<ul style="list-style-type: none"> Agregátor zdrojů Kategorie Knihy Komentáře Nastavení posílání příspěvků Obsah RSS exporty Typy obsahu 	<ul style="list-style-type: none"> Bloky Kontaktní formulář Menu Moduly Příklad rozhraní Spouštěče Témata vzhledu Uri aliasy
Uživatelé	Nastavení webu
Spravovat uživatelské účty, role a přístup uživatelů k jednotlivým možnostem.	Upravit základní nastavení webu.
<ul style="list-style-type: none"> Oprávnění Profily Přístupová pravidla Role Uživatelské nastavení Uživatelé 	<ul style="list-style-type: none"> Akce Datum a čas Hlášení chyb IMCE Informace o webu Jazyky Logování a výstrahy Nahrávání souborů Nastavení vyhledávání Systém souborů Toolkit pro práci s obrázky Téma vzhledu administrace Unychlení Vstupní formáty Výkon Údržba webu Čistá URL
Logy	
Zobrazit systémové logy a další informace o stavu webu.	
<ul style="list-style-type: none"> Nejnovější záznamy v logu Nedávná zobrazení Nejhledanější fráze Nejvíce odkazující stránky Seznam nejčastějších chyb „přístup odepřen“ Seznam nejčastějších chyb „stránka nenalezena“ Nejlepší stránky Nejčastější návštěvníci Nastavení logu přístupů Dostupné aktualizace Hlášení stavu 	

Obrázek 32 – vzhled administračního menu

Zde je možné si stejně jako je blok „Navigace na webu“ zobrazit i „Admin menu“. Vpravo v nabídce „Prvky webu“ je třeba kliknout na „Bloky“. Zobrazí se stránka, kde je dole seznam zapnutých a vypnutých bloků. V seznamu vypnutých se musí najít blok „Navigace“, kliknout na jeho rolovací tlačítko a vybrat Levý nebo Pravý boční panel. Po tomto vybrání se blok automaticky přesune mezi zapnuté a po uložení změn dole na stránce, se zobrazí na straně, která byla vybrána.



Obrázek 33 – vzhled zobrazení administrátorské navigace do panelu

Mezi zapnutými je i blok „Obsah knihy“ se zobrazením na Levém bočním panelu, tento blok prezentuje Navigaci na webu. U každého bloku je vpravo operace „Nastavení“. Zde je možné nastavit, kterým uživatelům, a na kterých stránkách, se blok bude zobrazovat.

Po nastavení a uložení bloku může uživatel, přihlášený jako administrátor, jakkoliv upravovat všechny příspěvky na stránkách a samozřejmě přidávat nové, a mít přitom stále Admin menu zobrazené na nějakém bočním panelu, který si vybral.

Vkládání příspěvků bylo vysvětleno výše. Úprava příspěvků je možná po kliknutí na nějaký příspěvek a zvolení položky „Upravit“.

4.2.1 Spouštění algoritmů

Spouštění probíhá přímo z internetové stránky prostřednictvím dvou vytvořených tlačítek. Po kliknutí na tlačítko procházet se zobrazí okno „Nahrát soubor“, kde uživatel musí vybrat dle svých požadavků a momentálně otevřené metody, jeden z vytvořených, a ve složce „Algoritmy“, uložených *exe* souborů. Vždy je třeba vybírat *exe* soubory ze složky *src*.

Například pro translaci:

`C:\wamp\www\drupal\Algoritmy\2D_Translace\Translace_2D_exe\src\Translace_2D_exe.exe`

Po vybrání příslušného *exe* souboru je třeba kliknout na „Otevřít“ a poté na stránce na tlačítko „Zpracuj“.



Obrázek 34 – načtený *exe* soubor připravený na zpracování

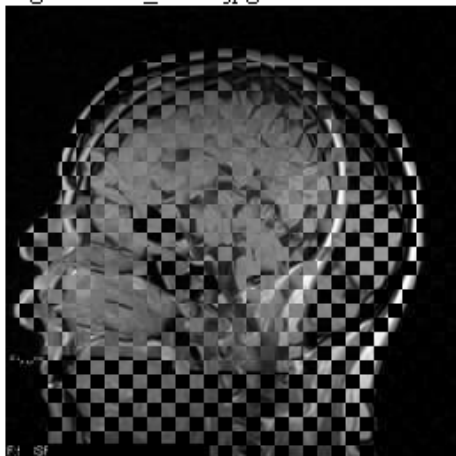
Po této volbě se spustí vybraný *exe* soubor a po jeho provedení se zobrazí nová stránka s výstupním obrázkem. Obrázek se uloží do složky *C:\wamp\www\drupal\img*.

Byl vybrán soubor:

C:\wamp\www\drupal\Algoritmy\2D_Translace\Translace_2D_exe\src\Translace_2D_exe.exe

Výsledek naší fce:

img/translace_finalni.jpg



Obrázek 35 – výsledek translace zobrazený na stránce

5. ZÁVĚR

V první části této práce je rozebrána teorie týkající se registrace medicínských obrazů. Jsou zde popsány 2D a 3D geometrické transformace, dále jednotlivé druhy interpolací, kritériální funkce a optimalizační metody. Tato teorie pak slouží jako zdroj informací pro internetovou databázi metod registrace obrazů.

Druhou částí této práce je Internetová databáze metod pro registraci obrazů. Nejprve je zde vysvětlena teorie technologií, které využívají jednotlivé části informačního systému. Dále je zde vysvětleno co to jsou redakční systémy. Jejich využití již bylo praktickou částí této práce.

Z několika druhů redakčních systémů byl pro tvorbu informačního systému vybrán Drupal, verze 6. Byl vybrán z důvodu jeho uživatelsky přehledného prostředí a velké možnosti rozšiřitelnosti pomocí modulů. Ty jsou volně ke stažení, stejně jako samotný redakční systém, nebo je zde i možnost vytvořit si modul vlastní. Informační systém byl realizován na základě teorie z první a druhé části této práce. Většinou formou HTML znaků, kterými se text formátoval a vkládaly se jím obrázky.

Informační systém byl zrealizován a je funkční. Byl naplněn příslušnými teoretickými formulacemi a obrazovou dokumentací.

Do informačního systému byly uvedeny příklady spustitelných algoritmů. Původní myšlenka byla v vytvoření výběrového systému pro uživatele, kdy si měl vybrat volitelné obrázky, dále zvolit vybranou interpolaci a transformační parametry geometrické transformace. Pokusil jsem se tuto variantu vyřešit prostřednictvím grafického uživatelského prostředí v Matlabu, tzv. GUI (GUIDE), ale u tohoto řešení se vyskytoval problém jak stránce předat výsledný obrázek aby jej mohla zpracovat.

Následně jsem započal druhé řešení, kde jsem se pokoušel volitelné obrázky a metody vytvořit přímo na internetové stránce, zde se ale vyskytl problém s Matlabem a příkazovým řádkem. Pro řešení bylo potřebné aby Matlab mohl číst probíhající argumenty na příkazové řádce OS Windows, což se nakonec ukázalo jako nereálné.

Poslední variantou řešení bylo vytvořit exe soubory bez možnosti výběru vstupů a ponechat uživateli pouze možnost si samostatně jednotlivé soubory spustit. Ovšem i zde se objevilo několik problémů. Prvním z nich je, že vytvořené exe soubory v jednom počítači, nemusejí vždy fungovat na jiném, sám jsem se o tom několikrát přesvědčil. Pokud chce uživatel mít alespoň nějakou šanci na spuštění musí mít v systému nainstalovaný Matlab Component Runtime (MCR), který slouží pro spouštění Matlabovských souborů i na PC, kde Matlab není nainstalovaný, a zároveň by měl být tento program ze stejné verze Matlabu jako ten v kterém byly exe soubory vytvořeny. Dalším problémem jsou internetové prohlížeče. Některé verze Internet Explorer a Mozilly Firefox nezobrazují korektně cestu nahraných exe souborů přes HTML tlačítka, což má za následek, že exe soubory neuvidí php skript a nezpracuje je. Týká se to nových i starších verzí. V prostředí Matlabu je problém u funkcí, jelikož nainstalovaný MCR zabírá 250MB z paměti RAM a samotné průběhy funkcí pro registraci obrazů jsou výpočetně také poměrně náročné, setkal jsem se i s tím, že PC nebylo schopné tyto všechny úkony zvládnout. Další částí tohoto problému je i to, že pokud exe soubor nahraný na stránku probíhá příliš dlouho, tak „stránka už nevydrží čekat na odezvu“ a jednoduše nezobrazí nic. Proto některé mnou vytvořené exe soubory mají ve svých funkcích podobnostní kritérium velmi nízké aby se vůbec stihly vykonat a zobrazit, bohužel na úkor kvality registrace obrazu. Osobně jsem zkoušel hotové stránky a algoritmy na pěti různých počítačích, různých hardwarových i softwarových specifikací. Z nich na dvou se algoritmy nespustily vůbec, na dalších dvou některé, a na jednom téměř všechny. Poslední PC měl nejvýkonnější specifikaci. U všech se vyskytoval zmíněný problém, exe soubor se vykonával, výsledný obrázek se uložil na pevný disk, ale stránka jej nedokázala přečíst. Pokud jsem exe soubor spustil v příkazové řádce, nebo poklepem ve Windows, vždy se v pořádku provedl.

Kromě php jsem našel ještě jedno teoretické řešení, kdy se exe soubor dá dostat na internetové stránky prostřednictvím JavaScriptů, k praktickému pokusu o toto řešení jsem se už nedostal.

Vytvořené stránky v této podobě jsou lehce přenositelné do onlinové podoby. Pokud by však měly být umístěny na nějakou doménu, vyskytl by se problém s využitím funkce `exec()`, kterou v php využívám pro zpracování php skriptů. Tato funkce je na 99% procentech serverů zakázána, z důvodu aby si uživatelé na cizím serveru nemohli spustit libovolně jakýkoliv exe soubor, například s virem. I proto byly stránky tvořeny na lokálním serveru aby se předešlo těmto problémům.

Osobně tedy tuto variantu nepovažuji za nijak výjimečné, ale ani ne úplně špatné řešení. Chce to mít kvalitní, pokud možno novější počítač a trochu štěstí na dobré verze prohlížečů, otázkou je kolik dalších problémů by se objevilo po umístění na nějakou doménu.

V poslední části práce jsou vytvořeny dva manuály, manuál k zprovoznění internetové knihovny a manuál k používání stránek. V prvním z nich je podrobně popsáno, krok po kroku i s obrazovou dokumentací, co je třeba udělat k zprovoznění programů přiložených na CD. V druhém je návod pro orientaci na stránkách, spouštění algoritmů a postup pro přihlášení do administrace stránek, který umožňuje jejich úpravu dle potřeb uživatele.

Před touto prací jsem s tvorbou internetových stránek neměl žádné zkušenosti, což se také mohlo promítnout na kvalitě a rychlosti mé práce. Ovšem po jejím vypracování jsem získal velice dobrý přehled o technologiích používaných v oblasti internetových stránek a jejich tvorbě a zároveň i dobrý přehled o metodách používaných v registraci medicínských obrazů.

Redakční systém Drupal, v kterém jsem stránky tvořil, je přehledný a uživatelsky přívětivý, pokud uživatel pochopí jeho podstatu. Pouze jeho zprovoznění, zvláště napoprvé, bylo poněkud složité. I přesto bych ho doporučil pro začínající uživatele, jako jsem byl já, na začátku této práce.

6. SEZNAM LITERATURY

- [1] ZITOVÁ, Barbara; FLUSSER, Jan. *Image registration methods* [online]. Prague : Academy of Sciences of the Czech Republic, Department of Image Processing, Institute of Information Theory and Automation. 2003 [cit. 2009-12-04]. 24 s. (PDF). Dostupné z WWW: <<http://library.utia.cas.cz/prace/20030125.pdf>>.
- [2] JAN, Jiří. *Medical image processing, reconstruction, and restoration : concepts and methods*. Boca Raton : Dekker/CRC Press, 2005. 730 s. ISBN 0-8247-5849-8.
- [3] KUBEČKA, Libor. *Optimalization methods for image registration* [online]. Brno: VUT, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství. 2003 [cit. 2009-12-04]. 24 s. (PDF). Dostupné z WWW: <http://www.feec.vutbr.cz/EEICT/2003/fsbornik/99-CD/02-Mgr/09-Computer_Intelligent_and_Graphic_Systems/05-kubecka_libor.pdf>.
- [4] RHODY, Harvey. *Geometric image transformations : Lecture 17* [online]. Rochester : Center for Imaging Science Institute of Technology. 2004 [cit. 2009-12-04]. 39 s. (PDF). Dostupné z WWW: <http://www.cis.rit.edu/class/simg782/lectures/lecture_18/lec782_04_17.pdf>.
- [5] OWENS, Robyn. *Computer Vision IT412 : Lecture 5* [online]. 1997 [cit. 2010-05-25]. Geometric Transformations. Dostupné z WWW: <http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT5/node5.html>.
- [6] BREEN, David; REGLI, William; PEYSAKHOV, Maxim. *3D transformations* [online]. Philadelphia : Geometric and Intelligent Computing Laboratory Department of Computer Science Drexel University. [cit. 2009-12-04]. 37 s. (PDF). Dostupné z WWW: <http://www.cs.drexel.edu/~david/Classes/CS430/Lectures/L-04_3DTransformations.6.pdf>.
- [7] PURGATHOFER, Werner. *Geometric transformations* [online]. Wien : TU, 2004 [cit. 2009-12-04]. 4 s. (PDF). Dostupné z WWW: <[http://www.cg.tuwien.ac.at/courses/CG/textblaetter/englisch/04GeometricTransformations\(engl\).pdf](http://www.cg.tuwien.ac.at/courses/CG/textblaetter/englisch/04GeometricTransformations(engl).pdf)>.
- [8] ROHDE, K. Gustavo. *Interpolation artefacts in Biomedical Image Registration* [online]. Pittsburgh : Carnegie Mellon University, Biomedical engineering. 2009 [cit. 2009-12-04]. 31 s. (PDF). Dostupné z WWW: <http://www.andrew.cmu.edu/user/gustavor/reg_artifacts.ppt.pdf>.
- [9] CHUANG, Jung Hong. *Chap 5 Geometric Transformation* [online]. Taiwan : National Chiao Tung University. 2000 [cit. 2010-05-25]. 32 s. (PDF). Dostupné z WWW: <<http://cggmwww.csie.nctu.edu.tw/courses/cgu/2000/cgu2000/CourseNotes/5-transformation.pdf>>.
- [10] LEAKE, M. James. *CAD/CAE Tutorials: 3D Geometric Transformations* [online]. Urbana : University of Illinois at Urbana-Champaign, [cit. 2009-12-04]. 12 s. (PPT). Dostupný z WWW: <<http://leake.ge.uiuc.edu/EG%20Lecture%20Slides/GM2.ppt>>.

- [11] HLAVÁČ, V; KYBIC, J. *Geometrické transformace* [online]. Praha : ČVUT, Fakulta elektrotechnická a katedra kybernetiky, Centrum strojového vnímání. [cit. 2009-12-04]. 15 s. (PDF). Dostupné z WWW:
< <http://cmp.felk.cvut.cz/cmp/courses/33zs11zima2005/slidy/geometrickeTransformace.pdf> >.
- [12] CHEN, Ying; HAO, Pengwei; ZHANG, Chao. *Shear-resize factorizations for fast image registration* [online]. London : Department of Computer Science, University of London. 2005 [cit. 2009-12-04]. 4 s. (PDF). Dostupné z WWW:
< <http://www.eecs.qmul.ac.uk/~phao/Papers/ICIP04.pdf> >.
- [13] HOLZSCHLAG, Molly E.; BOUDA, Jan. *HTML a CSS : jdi do toho*. Praha : Grada, 2006. Vytváření stránky v HTML, s. 263. ISBN 80-247-1454-X.
- [14] POLZER, Jan. *Drupal : podrobný průvodce tvorbou a správou webů*. Brno : Computer Press, 2008. Tvorba webu, s. 262. ISBN 978-80-251-1946-4.
- [15] BARTOŠ, Michal. *Lícování obrazů lékařských zobrazovacích modalit*. Brno : Vysoké učení technické, 2007. 66 s. Diplomová práce. VUT, Fakulta elektrotechniky a komunikačních technologií.
- [16] ČESKÝ WEBHOSTING s.r.o. *Redakční systémy* [online]. [cit. 2009-12-04] Dostupné z WWW: <http://navody.c4.cz/redakcni-systemy>