

# 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

POKROČILÉ ZPRACOVÁNÍ OFTALMOLOGICKÝCH VIDEO SEKVENCÍ  
RETINÁLNÍCH OBRAZŮ

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

Bc. PAVEL ŘÍHA

BRNO 2015



**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

## **POKROČILÉ ZPRACOVÁNÍ OFTALMOLOGICKÝCH VIDEO SEKVENCÍ RETINÁLNÍCH OBRAZŮ**

ADVANCED PROCESSING OF OPHTHALMOLOGIC VIDEOSERQUENCES OF RETINAL IMAGES

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. PAVEL ŘÍHA**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**prof. Ing. JIŘÍ JAN, CSc.**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav biomedicínského inženýrství

# Diplomová práce

magisterský navazující studijní obor  
Biomedicínské a ekologické inženýrství

**Student:** Bc. Pavel Říha

**ID:** 133802

**Ročník:** 2

**Akademický rok:** 2014/2015

## NÁZEV TÉMATU:

**Pokročilé zpracování oftalmologických video sekvencí retinálních obrazů**

## POKYNY PRO VYPRACOVÁNÍ:

1) Provedte literární rešerši v oblastech a) metod mono- a multimodálního lícování, se zaměřením zvláště na lícování multispektrálních obrazů, b) metod odhadu pohybu ve videosekvencích se zaměřením na časovou interpolaci. 2) Prozkoumejte možnosti interpolace a fúze barevných komponent specifického oftalmologického videa za účelem zvýšení časového a příp. i prostorového rozlišení složených barevných rámců. 3) Navrhněte algoritmy pro tuto konkrétní aplikaci a ověřte je v experimentální verzi softwaru. 4) Navrhněte strukturu výsledného programu (vstup: videodata z experimentální fundus kamery, výstup: videodata s barevnými rámci v kadenci odpovídající původnímu snímání R-G-B rámců), včetně kompatibilního uživatelského prostředí, vhodného k rutinnímu využití. 5) Implementujte program na vhodném hardwaru, s ohledem na přiměřenou minimalizaci výpočetní náročnosti. 6) Aplikujte program na dostupná oftalmologická data a zhodnoťte dosažené výsledky (podle možnosti i ve spolupráci s medicínskými experty).

## DOPORUČENÁ LITERATURA:

[1] JAN, J. Medical Image Processing, Reconstruction and Restoration. CRC Tailor and Francis (USA), 2006, ISBN 0-8247-5849-8.

[2] HAJNAL, J. V., HILL, D.L.G.: Medical Image Registration. CRC Press - Tailor and Francis Group (USA), 2001, ISBN 9780849300646.

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 22.5.2015

**Vedoucí práce:** prof. Ing. Jiří Jan, CSc.

**Konzultanti diplomové práce:**

**prof. Ing. Ivo Provazník, Ph.D.**

*Předseda oborové rady*

## ABSTRAKT

Tato práce se zabývá registrací a analýzou snímků z experimentální nízkonákladové fundus kamery dosahující nízkého SNR (kolem 10 dB) a nízkého časového i prostorového rozlišení. Cílem diplomové práce je prozkoumání možností digitálních úprav vedoucích k vytvoření videosekvence mající reálný přínos pro lékařskou diagnostiku.

K registraci je použit známý program elastix; předregistrační filtry i interpolace snímků jsou implementovány v Matlabu. Program poskytuje široké možnosti nastavení, z nichž bylo mnoho variant otestováno a vyhodnoceno. K posouzení dosažené přesnosti jsou vyhodnocovány prostorové odchylky v detekovaném pohybu cév.

Nejlepších výsledků s přesností pod 0.3 px bylo dosaženo použitím pásmové propusti, vhodně velké masky a rigidní registrace s metrikou vzájemné informace. Testované sekvence se podařilo registrovat dostatečně přesně pro vizuální hodnocení i základní výpočetní analýzy. Hlavním přínosem této práce jsou zregistrované sekvence a vytvořené aplikace, které mohou být využity při dalším vývoji experimentální kamery.

## KLÍČOVÁ SLOVA

Nízkonákladová fundus kamera, retinální snímek, monomodální a multimodální lícování, registrace, elastix, časová interpolace

## ABSTRACT

The diploma thesis deals with registration and analysis of images from the experimental low-cost fundus camera that reaches a low SNR (around 10 dB) and low temporal and spatial resolution. The aim of the diploma thesis is to explore the possibilities of digital processing leading to the creation of a videosequence that has real benefits for medical diagnostics.

The well-known program elastix is used for registration. Preprocessing filters and interpolation are implemented in Matlab. The program provides a wide range of setting options, out of which many combinations were tested and evaluated. To assess the accuracy achieved, spatial variations in the detected motion of blood-vessels are evaluated. Best results with a precision below 0.3 px were achieved by using a band-pass filter, a suitably sized mask, rigid registration and a metric of the mutual information. Test sequences were registered precisely enough both for visual assessment and basic computational analysis. Registered sequences and the developed application that both can be used in the further development of the experimental camera are the main contributions of the diploma thesis.

## KEYWORDS

Low-cost fundus camera, monomodal and multimodal registration, elastix, interpolation

ŘÍHA, Pavel *Pokročilé zpracování oftalmologických video sekvencí retinálních obrazů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2015. 109 s. Vedoucí diplomové práce prof. Ing. Jiří Jan, CSc..

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Pokročilé zpracování oftalmologických video sekvencí retinálních obrazů“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Děkuji především svému vedoucímu práce, panu prof. Ing. Jiřímu Janovi, CS. za odborné vedení, konzultace, připomínky k práci a předání mnoha cenných rad. Dále děkuji doc. Ing. Radimu Kolářovi, Ph.D. za konzultace a podnětné návrhy ke zpracování a analýze oftalmologických dat a Bernhardu Höherovi, Dipl. Ing. za informace o experimentální kameře a poskytnutí vytvořených snímků pro tuto diplomovou práci. Tato by práce by nevznikla bez podpory rodičů, díky nimž jsem se mohl věnovat celému studiu. Uvedeným, stejně jako všem těm, kteří přispěli jakoukoliv formou k vzniku práce, chci touto cestou poděkovat.

# OBSAH

|  |           |
|--|-----------|
| Úvod   | 10        |
| <b>1 Fundus kamera a retinální snímky</b>      | <b>12</b> |
| 1.1 Diagnostické využití                       | 12        |
| 1.2 Metody oftalmoskopie                       | 13        |
| 1.3 Experimentální kamera pro telemedicínu     | 14        |
| 1.3.1 Návrh experimentální fundus kamery       | 14        |
| 1.3.2 Konstrukce                               | 14        |
| 1.3.3 Potlačení odlesků                        | 14        |
| 1.3.4 Pozicování kamery                        | 14        |
| 1.3.5 Digitální zpracování                     | 14        |
| 1.3.6 Shrnutí vlastností kamery                | 14        |
| 1.4 Retinální snímek                           | 15        |
| 1.5 Zpracování retinálních snímků              | 16        |
| 1.6 Kontrast zelené složky                     | 16        |
| <b>2 Registrace obrazů</b>                     | <b>17</b> |
| 2.1 Přehled registračních metod                | 17        |
| 2.1.1 Rozdělení metod dle způsobu akvizice     | 17        |
| 2.1.2 Princip registrace                       | 18        |
| 2.2 Metody význačných bodů (Feature-based)     | 18        |
| 2.3 Podobnostní metody (Area-based)            | 19        |
| 2.4 Metody využívající Fourierovu transformaci | 21        |
| 2.5 Metody založené na vzájemné informaci      | 21        |
| 2.5.1 2D sdružený histogram                    | 22        |
| 2.5.2 Entropie                                 | 25        |
| 2.5.3 Vzájemná informace                       | 25        |
| 2.5.4 Metrika                                  | 26        |
| 2.6 Formální zavedení registrace               | 27        |
| 2.6.1 Definice registrace                      | 27        |
| 2.6.2 Metrika                                  | 27        |
| 2.6.3 Registrace                               | 28        |
| 2.7 Postup hledání transformace                | 29        |
| 2.7.1 Úplné prohledávání                       | 29        |
| 2.7.2 Gradient descent                         | 29        |
| 2.7.3 Evoluční algoritmy                       | 29        |
| 2.7.4 Pyramidální registrace                   | 29        |

|          |  |           |
|----------|--|-----------|
| 2.8      | Interpolace . . . . .  | 31        |
| 2.9      | Transformace . . . . .   | 32        |
| 2.9.1    | Lineární transformace . . . . .                                | 32        |
| 2.9.2    | Nelineární transformace . . . . .                              | 33        |
| 2.9.3    | Kombinace transformačních metod . . . . .                      | 34        |
| 2.9.4    | Zpětná transformace . . . . .                                  | 35        |
| 2.10     | Maskování . . . . .  | 36        |
| <b>3</b> | <b>Digitální filtrace fundus snímků</b>                        | <b>37</b> |
| 3.1      | Ekvalizace histogramu . . . . .                                | 37        |
| 3.2      | Adaptivní ekvalizace histogramu . . . . .                      | 37        |
| 3.3      | Gama filtrace . . . . .  | 37        |
| 3.4      | Gaussův filtr . . . . .  | 38        |
| 3.5      | Wienerův filtr . . . . .                                       | 38        |
| 3.6      | Mediánový filtr . . . . .                                      | 39        |
| 3.7      | Zvětšení prostorového rozlišení . . . . .                      | 39        |
| <b>4</b> | <b>Předregistrační úprava snímků</b>                           | <b>40</b> |
| 4.1      | Odstranění nekvalitních snímků . . . . .                       | 40        |
| 4.2      | Registrace s důrazem na cévy . . . . .                         | 40        |
| 4.3      | Možnosti detekce význačných bodů a jejich registrace . . . . . | 42        |
| 4.4      | Odstranění šumu filtrem typu dolní propust . . . . .           | 42        |
| 4.5      | Potlačení odlesků diod horní propustí . . . . .                | 43        |
| 4.6      | Využití filtrace pro multimodalitní registraci . . . . .       | 44        |
| 4.7      | Uživatelská filtrace . . . . .                                 | 47        |
| 4.7.1    | CLAHE filtrace . . . . .                                       | 47        |
| <b>5</b> | <b>Popis aplikace</b>  | <b>48</b> |
| 5.1      | Načtení snímků . . . . .                                       | 48        |
| 5.2      | Předregistrace . . . . .                                       | 49        |
| 5.3      | Rozdělení barevných komponent . . . . .                        | 49        |
| 5.4      | Registrace barevných kanálů . . . . .                          | 50        |
| 5.5      | Časová interpolace . . . . .                                   | 51        |
| 5.6      | Registrace snímků . . . . .                                    | 52        |
| 5.7      | Seskládání výsledné sekvence snímků . . . . .                  | 53        |
| 5.8      | Vytvoření videosekvence . . . . .                              | 54        |
| <b>6</b> | <b>Programová implementace</b>                                 | <b>57</b> |
| 6.1      | Celková koncepce . . . . .                                     | 57        |
| 6.2      | Externí aplikace . . . . .                                     | 58        |



|          |   |           |
|----------|---|-----------|
| 6.3      | Grafické a programové uživatelské rozhraní . . . . .                        | 58        |
| 6.4      | Jádro aplikace . . . . .  | 59        |
| 6.5      | Registrace obrazů . . . . .   | 59        |
| 6.5.1    | elastix . . . . .   | 59        |
| 6.5.2    | transformix . . . . .   | 60        |
| 6.5.3    | ElastixFromMatlab Toolbox . . . . .   | 61        |
| 6.5.4    | Vlastní vrstva nad toolboxem . . . . .                                      | 61        |
| 6.5.5    | Shrnutí . . . . .   | 62        |
| 6.6      | Rozhraní programu elastix . . . . .   | 63        |
| 6.7      | Implementace jednotlivých modulů . . . . .                                  | 65        |
| 6.7.1    | Načtení snímků, výběr ROI . . . . .   | 65        |
| 6.7.2    | Registrace barevných komponent . . . . .                                    | 65        |
| 6.7.3    | Interpolace . . . . .   | 68        |
| 6.7.4    | Kompletování barevných snímků . . . . .                                     | 69        |
| 6.7.5    | Vytvoření videa . . . . .   | 71        |
| 6.8      | Rozšiřitelnost programu . . . . .   | 73        |
| <b>7</b> | <b>Výsledky registrace</b>  | <b>74</b> |
| 7.1      | Výchozí nastavení pro elastix . . . . .                                     | 74        |
| 7.2      | Výběr typu transformace . . . . .   | 74        |
| 7.3      | Vliv odlesků . . . . .  | 76        |
| 7.4      | Výsledné snímky . . . . .   | 78        |
| 7.5      | Hodnocení dosažené chybovosti a přesnosti . . . . .                         | 78        |
| 7.6      | Kvantitativní hodnocení registrace . . . . .                                | 80        |
| 7.7      | Kvantitativní hodnocení různých registračních metod . . . . .               | 80        |
| 7.8      | Diskuse zjištěných výsledků a doporučení pro nastavení registrace . . . . . | 85        |
| 7.8.1    | Typ registrace . . . . .  | 85        |
| 7.8.2    | Metrika . . . . .   | 85        |
| 7.8.3    | Počet iterací . . . . .   | 86        |
| 7.8.4    | Velikost masky . . . . .  | 86        |
| 7.8.5    | Registrace barevných kanálů . . . . .                                       | 86        |
| 7.8.6    | Registrace k předchozímu snímku . . . . .                                   | 87        |
| 7.8.7    | Vzorkování pro výpočet metriky . . . . .                                    | 87        |
| 7.8.8    | Předregistrační filtrace . . . . .  | 87        |
| 7.9      | Časová náročnost registrace . . . . .                                       | 87        |
| <b>8</b> | <b>Analýza snímků</b>   | <b>88</b> |
| 8.1      | Kumulační zvýraznění signálu v šumu . . . . .                               | 88        |
| 8.2      | Detekce pulzace pomocí jasových profilů . . . . .                           | 91        |

|          |   |            |
|----------|---|------------|
| 8.3      | Detekce pohybu očí . . . . .                            | 93         |
|          | <b>Závěr</b>  | <b>95</b>  |
|          | <b>Literatura</b>                                       | <b>96</b>  |
|          | <b>Seznam příloh</b>                                    | <b>100</b> |
| <b>A</b> | <b>Instalace</b>  | <b>101</b> |
| A.1      | Softwarové závislosti . . . . .                         | 101        |
| A.2      | Hardwarové požadavky . . . . .                          | 101        |
| A.3      | Instalace . . . . .                                     | 101        |
| A.4      | Konfigurace . . . . .                                   | 102        |
| <b>B</b> | <b>Ovládání vytvořené aplikace</b>                      | <b>103</b> |
| B.1      | Spuštění aplikace . . . . .                             | 103        |
| B.2      | Vytvoření nového projektu . . . . .                     | 103        |
| B.3      | Otevření existujícího projektu . . . . .                | 103        |
| B.4      | Načtení snímků . . . . .                                | 103        |
| B.5      | Výběr ROI . . . . .                                     | 104        |
| B.6      | Nastavení masky . . . . .                               | 104        |
| B.7      | Registrace a interpolace snímků . . . . .               | 105        |
| B.8      | Indikace průběhu výpočtu . . . . .                      | 105        |
| B.9      | Prohlížení mezivýsledků . . . . .                       | 106        |
| B.9.1    | Export výsledků . . . . .                               | 106        |
| B.10     | Ukončení programu . . . . .                             | 106        |
| <b>C</b> | <b>Vytvoření uživatelského předregistračního filtru</b> | <b>107</b> |
| C.1      | Příklad vytvoření a použití CLAHE filtru . . . . .      | 107        |
| C.2      | Popis rozhraní (interface) filtračních funkcí . . . . . | 108        |
| <b>D</b> | <b>Obsah přiloženého CD</b>                             | <b>109</b> |

# ÚVOD

Odhaduje se, že 80 % informací z okolního světa získáváme pomocí zraku, což z očí činí nejdůležitější smyslový orgán. Navíc je sítnice jediným místem těla, kde můžeme pozorovat cévy *in vivo* a včas diagnostikovat i mnohé další nemoci.

Důležitost oka si lidé uvědomovali již v dávné historii, avšak možnosti diagnostiky a léčby oka musely čekat na technologický pokrok. Publikace různých přístupů k oftalmologii lze vysledovat přibližně do začátku 19. století, avšak oftalmologie jako profese neexistovala až do 60. let 20. století. Mezi důležité objevy „předdigitální doby“ vedoucí ke vzniku oftalmologie můžeme zařadit např. uvedení filmu Kodak Tri-X v roce 1954 nebo novou fundus kameru s elektronickým bleskem, lepší optikou a dostupnou cenou představenou Zeissem a Littmannem v roce 1955. Po nástupu digitální techniky v následujících desetiletích, zmiňme např. objev CCD v Bellových laboratořích v roce 1969 nebo představení slavného procesoru 4004 Intelem v roce 1971, již rozvoji oftalmologie nestálo nic v cestě.

V dnešní době je využití fundus kamer v očním lékařství ověřenou a přínosnou technikou. Nacházíme se v bodě, kdy začíná být kladen důraz na automatizované pořizování záznamů a jejich objektivní hodnocení za pomoci výpočetních metod (CAD - Computer-aided diagnosis). Na druhou stranu, pořizovací cena dnešních špičkových kamer se pohybuje v desítkách tisíc amerických dolarů a dělá tyto kamery nedostupnými pro rozvojové země. I proto se v současném trendu vývoje fundus kamer směřujícím ke stále lepší kvalitě vyčlenila nová větev, jejímž cílem je naopak snížení ceny a nabídnutí dostupné kamery i za cenu méně kvalitního obrazu.

A právě snímky z prototypu jedné takto vzniklé nízkonákladové kamery jsou předmětem zpracování v této diplomové práci. Hlavním cílem práce je vyrovnat se se značně horší kvalitou pořízených retinálních snímků a co nejlépe je digitálně upravit a registrovat do videosekvence, která bude mít maximální možnou diagnostickou hodnotu. Vzhledem k zamýšlenému použití kamery v rozvojových zemích (případně i v rámci programů telemedicíny) má být vyšetření neinvazivní, což vznáší požadavek na konstrukci nemydriatické kamery. To v důsledku znamená snížení vzorkovací frekvence a nutnost interpolace snímků do budoucího videa.

V dalších kapitolách této diplomové práce je v první teoretické části detailněji popsána konstrukce experimentální fundus kamery a známé přístupy k registraci medicínských dat. V praktické části je potom popsán návrh i implementace digitálních filtrů, registrace a interpolace obrazů. Součástí práce je v Matlabu napsaná aplikace pro registraci obrazů, která zajišťuje celý proces zpracování dat a s uživatelem komunikuje pomocí přehledného uživatelského rozhraní.

Ná závěr práce je nad rámec zadání uvedena základní analýza registrovaných

snímků, která nastiňuje možnosti jejich využitelnosti a zmiňuje případné další směry vývoje tohoto projektu.

# 1 FUNDUS KAMERA A RETINÁLNÍ SNÍMKY

Fundus kamera je specializovaná kamera určená ke snímání očního pozadí, tj. sítnice, optického disku s cévami a nervy a žluté skvrny. Snímky z fundus kamery se využívají v očním lékařství pro urgentní diagnostiku, screening, sledování vývoje nemocí a ostatní diagnostiku [23]. Příklad klasické fundus kamery je na obrázku 1.1.



Obr. 1.1: Příklad klasické fundus kamery [26]

## 1.1 Diagnostické využití

V lékařské praxi se fundus snímky stávají běžnou diagnostickou metodou pro diagnostiku cévní i necévní patologie. Retina je unikátní oblast lidského těla, kde mohou být cévní struktury přímo pozorovány *in vivo*, což vede k možnostem diagnózy mnoha nemocí v raných stádiích a následné léčbě.

V první řadě jde samozřejmě o oblast oftalmologie, tj. nemoci jako choroidální neovaskularizaci, retinální arteriální okluzi, glaukomy a obecně odhalení patologických změn. V druhé řadě lze hodnocení retinálních snímků použít jako indikátory i mimo oční lékařství pro diagnostiku hypertenze, diabetes mellitu, arterosklerózy, kardiovaskulárních onemocnění a cévních mozkových příhod [5].

Digitální analýza retinálních obrazů v první řadě pomáhá lékařským expertům k efektivní a přesné diagnóze, a následně k účinné léčbě. V druhé řadě by zdokonalení počítačového zpracování snímků mohlo v budoucnu vést k rozšíření automatizovaných screeningových programů [3][4].

## 1.2 Metody oftalmoskopie

Základem oftalmoskopie je prosvícení oka očním zrcátkem, tuto metodu popsal již v roce 1823 Purkyně. V závislosti na optické soustavě se rozlišuje oftalmoskopie přímá a nepřímá.

V následujících stručných bodech jsou pro úplnost uvedeny v současnosti nej-používanější metody oftalmoskopie, tato práce se však bude dále věnovat výhradně snímkům z optické fundus kamery.

- V dnešní době pro snímání barevných snímků sítnice a předního segmentu oka slouží fundus kamery s CCD snímáním obrazu. Fundus kamera je speciálně přizpůsobená kamera namontována na mikroskop, který složitou soustavou čoček a zrcadel umožní fokusaci paprsku na oční pozadí. Je to přímý nástupce přímé oftalmoskopie. Po zabudování barevných filtrů lze provádět fluorescenční angiografii.
- K vysoce detailnímu zobrazení fundu lze použít skenovací laserovou oftalmoskopii (SLO). SLO skenuje fundus fokusovaným laserovým paprskem v sekvencích bod po bodu. Určité množství světla v závislosti na pigmentaci a přítomnosti krve je odraženo zpět a je měřeno fotocitlivým senzorem, jehož signál moduluje jas obrazu.
- Optická koherentní tomografie OCT je neinvazivní bezkontaktní zobrazovací metoda. Umožňuje provádět řezy tkání s vysokou axiální rozlišovací schopností. Je to optická obdoba ultrazvuku, kdy paprsek proniká sítnicí a pozměněný jednotlivými vrstvami sítnice se odráží zpět.
- Pro diagnostiku přes špatně průhledná optická média (např. kvůli zákalům) lze použít ultrazvuková vyšetření.

Z pohledu průběhu (invazivnosti) vyšetření lze kamery rozdělit na dva základní typy:

- mydriatické – s uměle dilatovanou zornicí,
- a nemydriatické – bez uměle dilatované zornice.

## 1.3 Experimentální kamera pro telemedicínu

Ceny klasických fundus kamer se pohybují v rozsahu 20 000–50 000 US\$ a ačkoliv jsou, pokud jde o kvalitu, zlatým standardem, kvůli své ceně jsou nedostupné pro rozvojové země.

A právě s cílem podpořit zdravotnictví v zemích třetího světa vznikl na univerzitě v Erlangenu projekt zaměřený na sestavení levné fundus kamery. Cílem tohoto projektu pod vedením Ing. Bernharda Höhera bylo navrhnout levnou, odolnou a snadno ovladatelnou fundus kameru, která by byla použitelná v uvedených podmínkách a poskytovala co nejkvalitnější retinální snímky. Návazně se předpokládá, že odborný lékař bude diagnostiku provádět vzdáleně, v místě vyšetření bude kameru ovládat pouze méně kvalifikovaný personál [1].

Díky spolupráci mezi univerzitou v Erlangenu a VUT je možné použít prototypové snímky pro účely této diplomové práce.

### 1.3.1 Návrh experimentální fundus kamery

*Tato část dokumentu není přístupná ve veřejné verzi.*

### 1.3.2 Konstrukce

*Tato část dokumentu není přístupná ve veřejné verzi.*

### 1.3.3 Potlačení odlesků

*Tato část dokumentu není přístupná ve veřejné verzi.*

### 1.3.4 Pozicování kamery

*Tato část dokumentu není přístupná ve veřejné verzi.*

### 1.3.5 Digitální zpracování

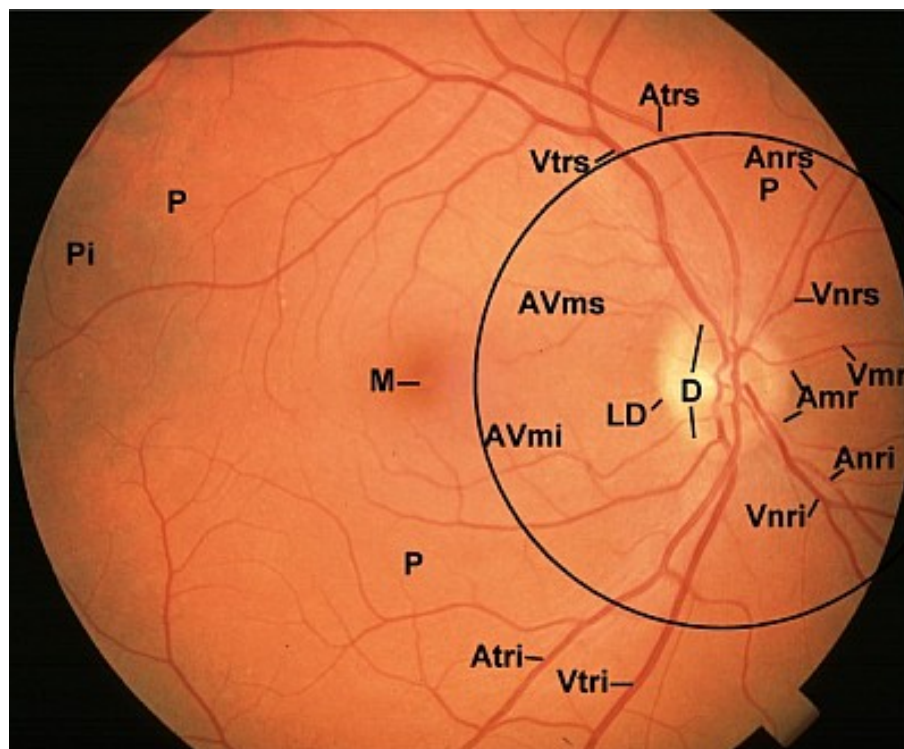
*Tato část dokumentu není přístupná ve veřejné verzi.*

### 1.3.6 Shrnutí vlastností kamery

*Tato část dokumentu není přístupná ve veřejné verzi.*

## 1.4 Retinální snímek

Hlavní strukturu retinálního snímku tvoří optický disk, *a. centralis retinae* dělí se na své 4 hlavní větve a rozvětvená síť menších cév. Příklad retinálního snímku



Obr. 1.2: Oftalmologický snímek zdravého lidského pravého oka včetně popsané anatomie cév sítnice [24]. (Tento snímek byl pořízen klasickou fundus kamerou.)

s vyznačenou anatomí cév a níže uvedenými popisy je uveden obr. 1.2.

- Amr = Arteriola medialis retinae; Anri = Arteriola nasalis retinae inferior; Anrs = Arteriola nasalis retinae superioris; Atri = Arteriola temporalis retinae inferioris; Atrs = Arteriola temporalis retinae superioris; AVmi = Arteriola et Venola macularis inferior; AVms = Arteriola et Venola macularis superior;
- D = Discus nervi optici (= Papilla nervi optici = optic nerve papilla = „slepá skvrna“);
- M = Macula lutea („žlutá skvrna“ = Fovea centralis; spot of highest visual acuity on which only cones are present);
- Vmr = Venola medialis retinae; Vnri = Venola nasalis retinae inferior; Vnrs = Venola nasalis retinae superioris; Vtri = Venola temporalis retinae inferioris; Vtrs = Venola temporalis retinae superioris [24].



## 1.5 Zpracování retinálních snímků

Ke zpracování retinálních snímků se podle účelu přistupuje mnoha způsoby. Zde bude dále diskutována hlavně registrace oftalmologických sekvencí do videa s cílem sledovat změny v čase související s pulzací cév. Významnou diagnostickou hodnotu však mají i jednotlivé statické snímky a po úspěšné registraci může být ze souboru více statických snímků vytvořen jeden s vyšším SNR, což je zvláště u těchto zašuměných obrazů velmi užitečné. Mezi diagnosticky významné vlastnosti patří šířky cév, barvy jednotlivých objektů, zkroucení cév, reflexe povrchů nebo abnormální větvení cév. Vzhledem k velkému významu struktury retinálních cév je kladen velký důraz především na zvýšení jejich kontrastu a následnou segmentaci a analýzu [2], na druhou stranu, například pro hodnocení diabetu je důležitá analýza barev a textur.

## 1.6 Kontrast zelené složky

Fundus snímky z optických kamer jsou obvykle snímány ve všech RGB složkách, tzn. získáváme obraz v reálných barvách, avšak kvalita jednotlivých barevných kanálů není srovnatelná. Nejlepší kontrast mezi cévami a pozadím je v zeleném kanálu, je to způsobeno tím, že zelené světlo je absorbováno krví, ale je odraženo retinálním epitelním pigmentem (RPE - Retinal Pigment Epithelium) [2]. Tato vlastnost je využitelná i při registraci, zelená složka bude v časové řadě zregistrovaná nejlépe a proto na ni budeme registrovat ostatní kanály.

Doplňme, že někteří autoři pro extrakci klíčových bodů, ale i registraci s OCT snímky, pracují pouze se zelenou složkou a zbylé složky vůbec neuvažují.

## 2 REGISTRACE OBRAZŮ

### 2.1 Přehled registračních metod

V posledních přibližně třech desetiletích došlo ke značnému rozšíření digitální akvizice obrazových dat nejen v biomedicínském prostředí. Existence digitálních obrazových dat a dostupnost výkonných počítačů vedla k velkému rozvoji technik počítačového zpracování obrazu. Obsáhlý přehled metod pro registraci obrazů byl Lisou Brownovou publikován již v roce 1992 [6]. Dle článku Barbary Zitové a Jan Flussera z roku 2003 [7] bylo v posledním desetiletí publikováno více než 1000 vědeckých článků na téma registrace obrazů. Od té doby samozřejmě došlo k dalšímu exponenciálnímu růstu, ale v té době prezentované metody jsou stále klasickým úvodem do problematiky a základním kamenem pro tvorbu sofistikovanějších algoritmů. Následující stručný přehled základních registračních metod vychází hlavně z výše citovaných zdrojů.

#### 2.1.1 Rozdělení metod dle způsobu akvizice

Registrace obrazů je široce užívaná ve vzdáleném sledování, medicínském zpracování obrazů, počítačovém vidění apod. Obecně můžeme aplikace registrace rozdělit dle způsobu akvizice do 4 skupin.

- **Akvizice z různých pohledů** (multiview analysis). Cílem je získání většího 2D nebo 3D snímku skenované scény.
- **Snímky získané v různých časech** (multitemporal analysis). Cílem je najít a vyhodnotit změny ve scéně, ke kterým došlo mezi jednotlivými akvizicemi. Medicínským příkladem je sledování vývoje růstu tumoru, účinků terapie apod.
- **Multimodalitní** (multimodal analysis), tj. snímky stejné scény získané ve stejném čase pomocí různých modalit. Cílem je integrovat informace získané z jednotlivých snímků do komplexní obrazové prezentace. Příkladem mohou být multispektrální snímky zpracovávané astronomy nebo v medicíně kombinace informací z různých modalit (MRI, CT, SPECT/PET, UTZ), kdy získáme informace jak o anatomii, tak o fyziologii v jediném obraze, viz obr. 2.1.
- **Modelování** (scene to model registration). Dochází k registraci mezi snímky konkrétní scény a modelem scény. Modelem může být mapa, objekty GIS, jiný pacient, průměrný objekt apod. Cílem je porovnání scény s modelem. Hlavní aplikací v medicíně je porovnání pacientových snímků s digitálními anatomickými atlasy, následná klasifikace chorob apod.

## 2.1.2 Princip registrace

Vzhledem k velmi rozdílným vstupním datům a požadavkům na výsledek dosud neexistuje univerzální registrační metoda aplikovatelná na všechny požadované úlohy. Přesto lze obecně shrnout několik kroků, které se v určité podobě vyskytují ve většině algoritmů:

- detekce význačných vlastností,
- nalezení korespondence mezi vlastnostmi,
- transformace modelu,
- transformace a interpolace obrazu.

## 2.2 Metody význačných bodů (Feature-based)

Zde blíže nedefinovaný pojem vlastnost je klíčovým pro úspěch registračního algoritmu. Vlastnost (v anglické literatuře feature, odtud feature-based algorithm) může být ohraničená plocha/textura, hrana (edge), kontura, těžiště apod. Dokážeme-li v registrovaných obrazech přesně detekovat tyto vlastnosti, ať už automaticky nebo ručně, můžeme vypočítat vhodnou transformaci obrazu a provést velmi přesnou registraci. Problém nastává u obrazů, kdy nalezení takových vlastností s dostatečnou přesností není možné.

Často uváděným příkladem pro registraci obrazů založenou na korespondenci významných bodů je sestavování panoramatických snímků, typicky městské zástavby. Na střechách, věžičkách a oknech se obvykle podaří najít vzájemně korespondující body, provést kvalitní registraci a složit panorama. V počátcích byly vhodné rysy ručně vybírány odborníky, během postupné automatizace registračních procesů však sílila snaha hledat je automaticky<sup>1</sup>.

Je známo několik metod detekce klíčových bodů, mezi nejspolehlivější patří křížení čar (line intersection), jiné metody založené např. na výpočtu těžiště, lokálních extrémů nebo bodů nespojitostí jsou často nepřesné. Byly zkoušeny různé další metody, např. parciální derivace druhého řádu (Kitchen a Rosenfeld), ale zde se jako problém objevil šum v obraze. Později se jako spolehlivější ukázal detektor založený na derivaci prvního řádu. Jako užitečné se ukázaly také více intuitivní algoritmy vycházející ze změny jasových hodnot bez použití derivací.

Zdůrazněme, že na rozdíl od podobnostních (area-based) metod metody význačných bodů (feature-based) nepracují přímo s intenzitou jasu. Význačné rysy reprezentují abstrahovanou informaci na vyšší úrovni.

---

<sup>1</sup>Není bez zajímavosti, že v roce 2014 zabudoval Google do svých galerií na Google Plus funkci, která bez jakékoliv apriorní informace (jen čas pořízení) hledala ve fotografiích nahraných uživateli ty, které lze spojit v panorama, a automaticky tato panoramata vytvářela.

Z naznačeného principu metod založených na význačných bodech plyne důležitý závěr. Tyto metody jsou vhodné pro obrazy obsahující dostatek rozlišitelných a snadno detekovatelných objektů, což medicínské obrazy ne vždy splňují. Konkrétně kvalitní retinální snímky jsou typicky vhodným kandidátem pro detekci míst větvení cév, avšak zde zpracovávané experimentální snímky mají malé ROI a dosahují nízkého SNR, což detekci význačných bodů znesnadňuje [44].

## 2.3 Podobnostní metody (Area-based)

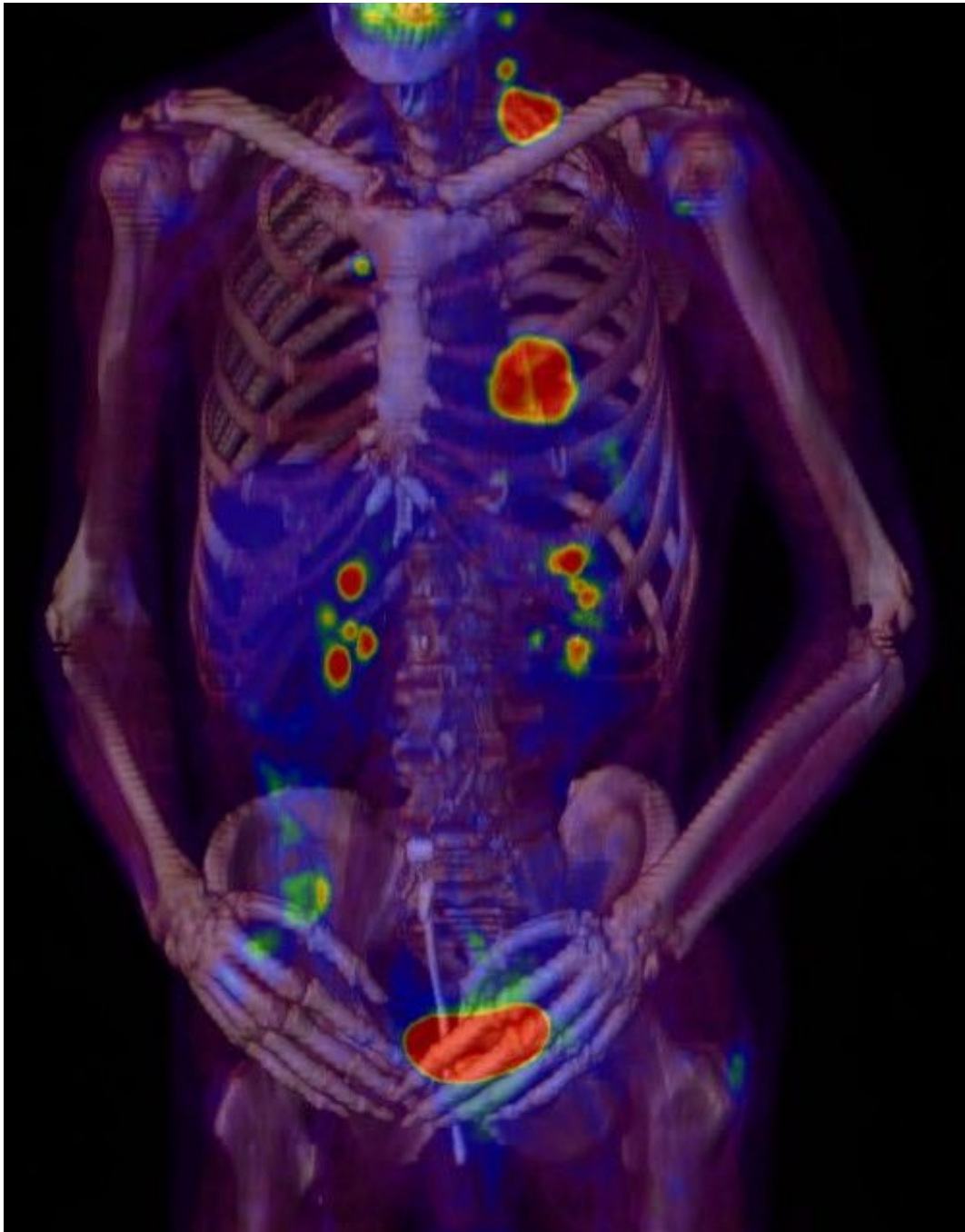
Podobnostní metody vynechávají krok s detekcí specifických rysů a kladou důraz na co nejlepší registraci přímo z jasových ploch v obraze.

Někdy užívaný název korelační metody plyne z principu návrhu kritériálních funkcí, které jsou založeny na posuzování shodnosti vzájemně si odpovídajících bodů. Pro odhad korespondence jsou využívány buď celé obrazy, nebo jejich výřezy. Mezi nevýhodami je uváděno [7], že zvláště u zašuměných obrazů může docházet k vzájemné registraci necharakteristických regionů. Nemyslím si, že šum je problém pouze těchto metod, je to obecný problém a i zde ho můžeme do jisté míry kompenzovat.

Klasická reprezentace podobnostních metod je založená na korelačním koeficientu [9]  $CC(i, j)$  (Cross-Correlation) a jeho modifikacích [8].

$$CC(i, j) = \frac{\sum_W (W - E(W))(I_{(i,j)} - E(I_{(i,j)}))}{\sqrt{\sum_W (W - E(W))^2} \sqrt{\sum_{I_{(i,j)}} (I_{(i,j)} - E(I_{(i,j)}))^2}} \quad (2.1)$$

Korelační koeficient nabývá hodnot z intervalu  $< -1; 1 >$ , přičemž 1 dosáhneme při úplné shodnosti testovaných obrazů. Přístup založený na přímém výpočtu korelačního koeficientu z jasových hodnot je citlivý na změny intenzity jasu a šumu, do jisté míry je ale lze potlačit normalizací snímků. Ze základního korelačního vztahu 2.1 jsou následně odvozovány (výpočetně i implementačně) zjednodušené funkce, například součet absolutních diferencí jasových hodnot mezi oběma obrazy nebo součet čtverců diferencí jasových hodnot mezi dvěma obrazy (Euklidovská vzdálenost, rovnice 2.5).



*Obr. 2.1: Fúze multimodalitních dat v medicíně, spojení anatomických a fyziologických informací do jednoho obrazu. Místa s podezřením na léze detekovaná celotělovým PETem jsou zobrazena na snímku z CT [15]*

## 2.4 Metody využívající Fourierovu transformaci

Fourierovské metody využívají Fourierovy transformace do spektrální oblasti, jejich využití je doporučeno hlavně pro případy, kde jsou zpracovávány obrazy poškozené frekvenčně závislým šumem (vůči němu je metoda mimořádně robustní) nebo je třeba výpočetně zjednodušit registrační proces velkých snímků. Na druhou stranu, s bílým šumem si metoda moc dobře neporadí a používají se další korekce. [6]

Registrační fáze vychází z Fourierova posuvného teorému (Fourier Shift Theorem) [14]

$$\mathcal{F}\{f(t - t_0)\}(s) = e^{-2\pi i s t_0} F(s)$$

a byla původně navržena pro registraci posunutých obrazů. Využívá se toho, že amplituda obrazů je stejná, ale v závislosti na vzájemném posunu se liší jejich fáze.

Máme-li dva vzájemně posunuté obrazy  $f_F$  a  $f_M$ , v prostorové doméně popsané jako

$$f_M(x, y) = f_F(x - d_x, y - d_y),$$

lze vyjádřit korespondenci ve spektrální doméně jako

$$F_M(\omega_x, \omega_y) = e^{-i(\omega_x d_x + \omega_y d_y)} F_F(\omega_x, \omega_y).$$

Během registrace se počítá vzájemné výkonové spektrum obrazů dle 2.2 a hledá se poloha peaku představující poziční shodu. Pokud fázi vzájemného spektra vyjádříme v prostorové doméně, získáme impulzní funkci, kde všechny pozice kromě optimálně registrované budou přibližně nulové.

$$\frac{\mathcal{F}(f_F)\mathcal{F}(f_M)^*}{|\mathcal{F}(f_F)\mathcal{F}(f_m)^*|} = e^{2\pi i(ux_0 + vy_0)} \quad (2.2)$$

De Castro a Morandi [11] navrhli rozšíření metody fázové korelace navíc pro rotační transformaci. Další vývoj vedl i k vyřešení škálování během registrace (polar-log mapping of the spectral magnitude – Fourier-Mellin transform) a registraci 3D obrazů [12]. Výsledky testů (simulované podmínky umělé deformací a zašuměním) byly uspokojivé a to přímo u snímku z fundus kamery [13]. Z novějších článků byla tato metoda publikována např. R. Kolářem na našem ústavu [42].

## 2.5 Metody založené na vzájemné informaci

Vzájemná informace je často označovaná zkratkou MI z anglického Mutual Information. Metody vzájemné informace jsou poslední skupinou podobnostních algoritmů používanou hlavně pro registrování multimodálních obrazů.

Ačkoliv je registrace multimodálních snímků považována za obtížnou úlohu, v medicínských aplikacích je pro integraci maxima diagnostických informací zcela

nezbytná a právě využití vzájemné informace je často klíčové pro registraci dat z odlišných zobrazovacích systémů. Srovnání dříve uvedených korelačních koeficientů a vzájemné informace je rozebíráno např. v *The Correlation Ratio as a New Similarity Measure for Multimodal Image Registration* [10], který popisuje korelační přístup s ohledem na funkční závislost jasových bodů v obrazech.

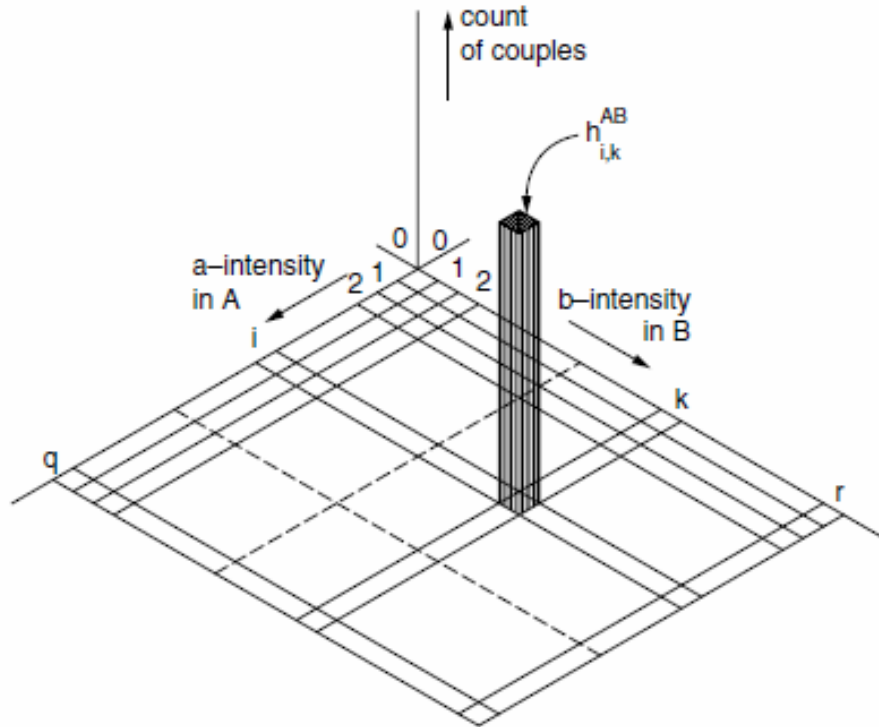
Pro vysvětlení principu vyjděme z 2D sdruženého histogramu.

### 2.5.1 2D sdružený histogram

Pro 2 diskrétní obrazy  $A$  a  $B$  stejných rozměrů, s  $q$ , resp.  $r$  stupni šedi, je 2D sdružený histogram  $h^{AB}$  matice četností současného výskytu různých kombinací jasů na odpovídajících si pozicích v obou obrazech. Formálně sdružený histogram definujeme dle 2.3 a jeho konstrukce je schématicky znázorněna na obr. 2.2 [31].

$$h^{AB} : h_{i,k}^{AB} = \text{count}(D_{i,k}) = \sum_{D_{i,k}} 1, \quad (2.3)$$

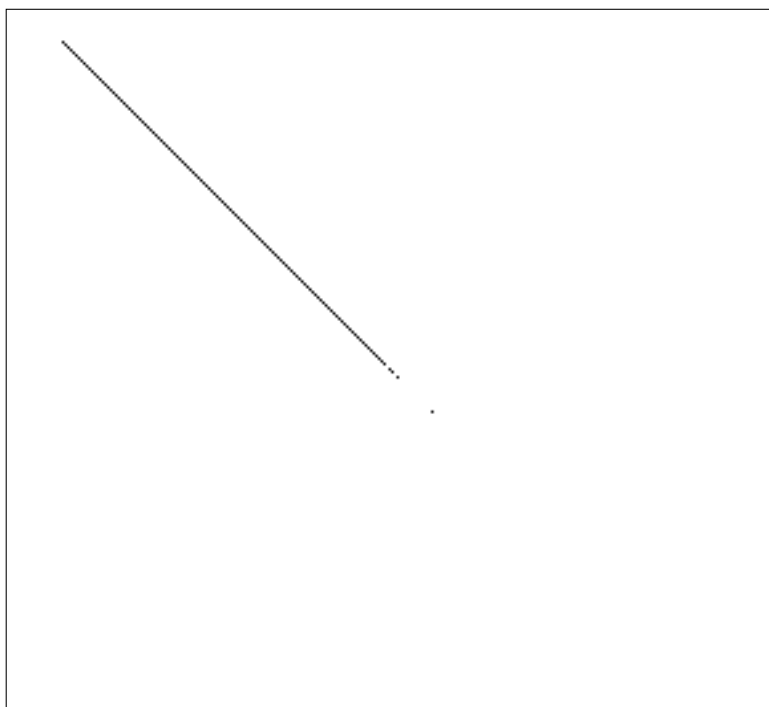
$$D_{i,k} = \{[l, m] : ((a_{l,m} \in A) = i) \wedge ((b_{l,m} \in B) = i)\}$$



Obr. 2.2: Schématické znázornění 2D sdruženého histogramu [31].

Pro lepší popis praktických dopadů uveďme příklady několika sdružených histogramů vypočítaných z různých vstupních obrazů. První histogram 2.3 je vypočítán z jednoho vstupního obrazu –  $2dHist(A, A)$ , takový histogram představuje ideálně

zregistrované obrazy. Protože jsou obrazy schodné, nemohou být body histogramu jinde než na vzájemně si odpovídajících pozicích  $i = j$  v  $hist_{i,j}$  (tj. na úhlopříčce), jejich prostorová distribuce odpovídá distribuci jasových bodů v původním obraze. Tuto podobu lze považovat za referenční při registraci obrazů. Pokud se jas jedné kopie obrazu lineárně transformuje, změní se poloha i strmost, ale stále budou všechny body histogramu náležet jedné přímce.

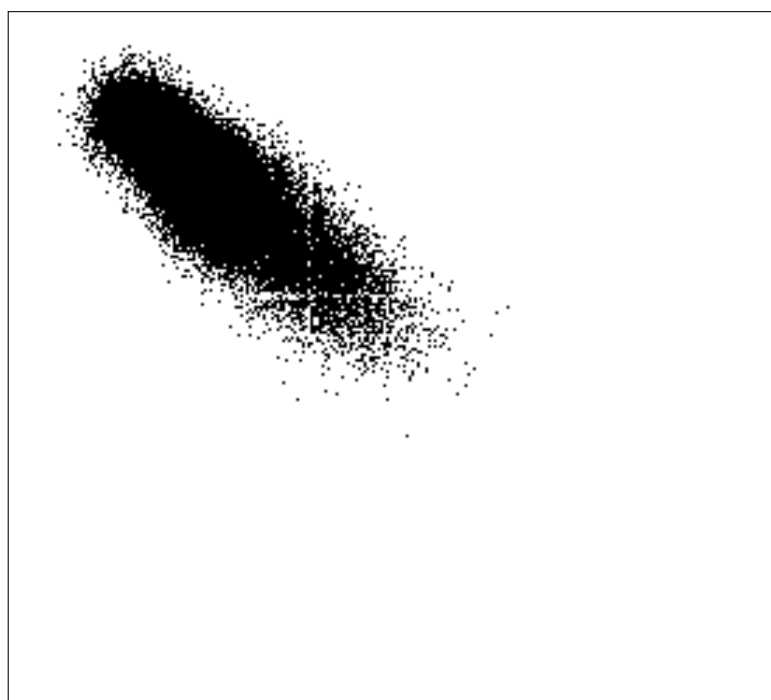


Obr. 2.3: Sdružený histogram téhož snímku.

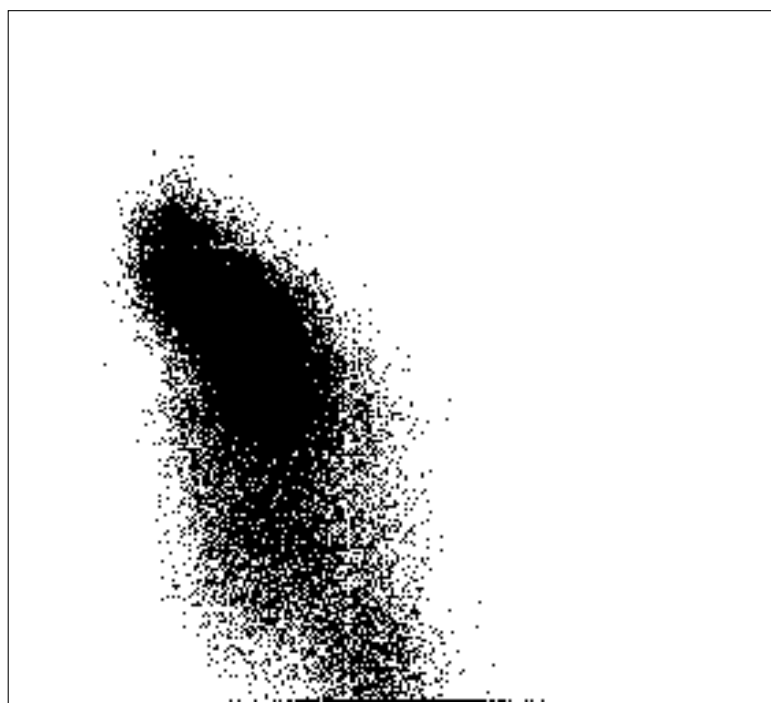
Histogram 2.4 byl vytvořen ze 2 reálných snímků téhož barevného kanálu. Je již více rozptýlený, ale zůstává symetrický a je podobný ideálnímu histogramu. Velký rozptyl bodů je způsoben hlavně šumem.

U multimodalitního lícování ideálního stavu prakticky nelze dosáhnout, jasové hodnoty na stejných pozicích si nikdy nebudou plně odpovídat. Při lícování vycházíme z předpokladu, že jasy téhož objektu ve scéně jsou u každé modality transformovány různými, avšak neměnnými funkcemi, což se ve výsledném sdruženém histogramu projeví shlukováním bodů do několika málo shluků (na rozdíl od histogramu zcela náhodných obrazů, kde by byl celý histogram pokryt rovnoměrně). Je zřejmé, že u slícovaných obrazů dojde k největšímu shlukování a lokální maxima dosáhnou nejvyšších hodnot, tj. jasově významné objekty si vzájemně nejlépe odpovídají. Proto se při registraci snažíme tohoto stavu dosáhnout a nazýváme ho maximalizací vzájemné informace.





*Obr. 2.4: Sdružený histogram registrovaných snímků zeleného kanálu.*



*Obr. 2.5: Sdružený histogram registrovaných snímků červeného a zeleného kanálu.*

Obrázek 2.5 ukazuje sdružený histogram multimodálních snímků různých barevných složek.

## 2.5.2 Entropie

Vzájemná informace, původně formulovaná v teoretické informatice a v teorii pravděpodobnosti, představuje určení vzájemných statistických závislostí mezi různými sadami dat a jako taková je vhodná pro registraci obrazů získaných z různých modalit. Vzájemná informace (MI) mezi dvěma náhodnými vektory  $X$  a  $Y$  je definována jako

$$\begin{aligned} MI(X, Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X, Y) - H(X|Y) - H(Y|X) \\ &= H(X) + H(Y) - H(X, Y), \end{aligned}$$

kde  $H(X)$ , resp.  $H(Y)$  jsou marginální entropie,  $H(X|Y)$ , resp.  $H(Y|X)$  jsou podmíněné entropie a  $H(X, Y)$  je sdružená entropie. Entropie  $H(X)$  vektoru  $X$  s rozdělením pravděpodobnosti  $P(X)$  je definována jako střední hodnota

$$H(X) = -kE(\log(P(X))) = -k \sum_{i=1}^N P(x_i) \log(P(x_i)),$$

kde  $X$  je náhodný vektor.

Pro informační entropii obvykle volíme  $k = \frac{1}{\log 2}$  a uvažujeme výsledný vztah

$$H(X) = - \sum_{i=1}^N P(x_i) \log_2(P(x_i)),$$

jednotkou je potom *1 bit*.

## 2.5.3 Vzájemná informace

Jeden z prvních článků popisujících registrační algoritmus založený na maximalizaci vzájemné informace publikovali Viola a Wells [33], kteří navrhovali použití kritéria MI pro registraci snímků z magnetické rezonance, k maximalizaci použili optimalizaci sestupným gradientem. Již v té době bylo experimentováno s optimalizačními technikami hierarchického prohledávání spolu se simulovaným žiháním, těmito technikám se ale v této práci nebudu detailněji věnovat.

Klasická implementace registračního algoritmu vychází z vyhodnocování vzájemné informace dle výpočetního vztahu

$$MI(X, Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right),$$

kde  $p(x, y)$  je sdružená hustota pravděpodobnosti proměnných  $X, Y$  a  $p(x)$ , resp.  $p(y)$  jsou marginální hustoty pravděpodobnosti  $X$ , resp.  $Y$ . Sdružené pravděpodobnosti jsou získány normalizací patřičných histogramů tak, aby platilo  $\sum_{x \in X} p(x) = 1$ .

Pro praktické použití v programu elastix se kromě dvou vektorů obrazů předává transformační vektor  $\mu$ ,

$$MI(\mu, I_F, I_M) = \sum_{m \in I_M} \sum_{f \in I_F} p(f, m; \mu) \log \left( \frac{p(f, m; \mu)}{p(f)p(m; \mu)} \right), \quad (2.4)$$

kde  $I_F$  představuje referenční obraz a  $I_M$  deformovaný obraz. Sdružená pravděpodobnost je odhadnuta pomocí B-splínového Parzenova okna

$$p(f, m; \mu) = \frac{1}{|\Omega_F|} \sum_{x_i \in \Omega_F} w_F \left( \frac{f - I_F(x_i)}{\sigma_F} \right) \times w_M \left( \frac{m - I_M(T_\mu(x_i))}{\sigma_M} \right),$$

kde  $w_F$  a  $w_M$  reprezentují referenční a deformované B-splínové Parzenovo okno.

## 2.5.4 Metrika

Výše uvedený vztah pro výpočet hodnoty vzájemné informace  $MI(X, Y)$  sice můžeme použít jako kritériální funkci, avšak není skutečnou metrikou v matematickém smyslu.

Doplňme, že určení vzdálenosti mezi dvěma body  $X$  a  $Y$  pomocí vzájemné informace je možné a můžeme definovat metriku (známou jako variace informace)

$$d(X, Y) = H(X|Y) + H(Y|X),$$

která splní všechny axiomy požadované definicí metriky (axiom nezápornosti, axiom totožnosti, axiom symetrie i trojúhelníkovou nerovnost).

Pokud jsou  $X$  a  $Y$  náhodné diskrétní vektory, potom jsou všechny výrazy s entropiemi nezáporné, takže platí  $0 \leq d(X, Y) \leq H(X, Y)$  a můžeme definovat normalizovanou vzdálenost

$$D(X, Y) = \frac{d(X, Y)}{H(X, Y)} \leq 1.$$

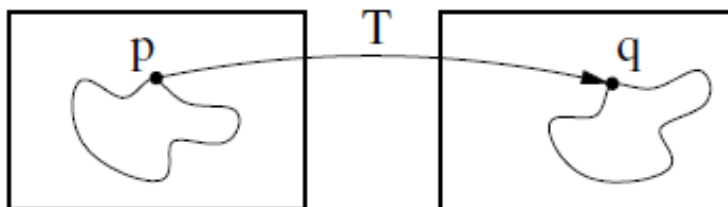
Metrika  $D$  je potom univerzální metrikou [34], [38], [39].

## 2.6 Formální zavedení registrace

### 2.6.1 Definice registrace

Do procesu registrace vstupují dva obrazy, referenční  $I_F(\mathbf{x})$  (z angličtiny označovaný jako fixed nebo sensed, tj. neměnný, referenční) a deformovaný  $I_M(\mathbf{x})$  (moving, tj. pohyblivý). Pohyblivý obraz je deformován tak, aby se co nejlépe shodoval s referenčním obrazem.

Oba obrazy jsou zjevně stejné dimenze  $d$ ,  $\Omega_F \subset \mathbb{R}^d$  a  $\Omega_M \subset \mathbb{R}^d$ . Při nejjednodušším lícování hledáme posun  $\mathbf{u}(\mathbf{x})$  tak, aby byl deformovaný obraz  $I_M(\mathbf{x} + \mathbf{u}(\mathbf{x}))$  zarovnáán k referenčnímu obrazu  $I_F(\mathbf{x})$ . Ekvivalentně můžeme tento proces označit za transformaci  $\mathbf{T}(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x})$  tak, že  $I_M(\mathbf{T}(\mathbf{x}))$  je zarovnáán k  $I_F(\mathbf{x})$ . Transformace zřejmě zobrazuje body z referenčního obrazu na deformovaný  $\mathbf{T} : \Omega_F \subset \mathbb{R}^d \rightarrow \Omega_M \subset \mathbb{R}^d$ , což je schématicky naznačeno na obr. 2.6.



Obr. 2.6: Zobrazení bodu z referenčního obrazu na deformovaný [28]

### 2.6.2 Metrika

Kvalita transformace, neboli aktuální shodnost lícovaných obrazů, je hodnocena metrikou. Dle situace můžeme jako metriku použít různé kriteriální funkce (anglicky často *fitness*), obecně jde buď o funkce 2 vstupních vektorů  $I_F$  a  $I_M$ ,

$$\mathcal{C}(I_F, I_M),$$

které pracují s již transformovaným obrazem  $\mathcal{C}(I_F, T_\mu(I_M))$ , nebo o funkce 3 vstupních parametrů

$$\mathcal{C}(\mu; I_F, I_M),$$

kde je aktuální transformační vektor přímo předáván funkci.

Mezi nejznámější metriky patří Euklidovská vzdálenost, kosinové kritérium, korelační funkce (rovnice 2.6) nebo vzájemná informace. Zvolená metoda závisí na vlastnostech jednotlivých obrazů, u monomodálního lícování dosáhneme často velmi dobrých výsledků i s nejjednodušší metrikou Euklidovské vzdálenosti (rovnice 2.5) vypočítané z jasových hodnot jednotlivých obrazů. U multimodálního lícování, nebo

pokud jsou monomodální obrazy jasově nevyvážené, je třeba použít pokročilejší metody. Kosinovo kritérium porovnává úhel mezi vektory jasů obou obrazů a umožní nám najít shodu i tam, kde je jeden obraz výrazně jasově transformován. Chceme-li lícovat multimodální snímky, je nutné použít metody využívající vzájemné informace (viz rovnice 2.4). Výčet samozřejmě není úplný, pro specifické operace je vhodné použít specializované techniky, např. pro registraci binárních segmentovaných dat se jako kritérium hodí Cohonovo kappa.

$$SSD(\mu, I_F, I_M) = \frac{1}{\Omega_F} \sum_{x_i \in \Omega_F} (I_F(x_u) - I_M(T_\mu(x_i)))^2 \quad (2.5)$$

$$NCC(\mu, I_F, I_M) = \frac{\sum_{x_i \in \Omega_F} (I_F(x_i) - E(I_F))(I_M(T_\mu(x_i)) - E(I_M))}{\sqrt{\sum_{x_i \in \Omega_F} (I_F(x_i) - E(I_F))^2 \sum_{x_i \in \Omega_F} (I_M(T_\mu(x_i)) - E(I_M))^2}} \quad (2.6)$$

[28], [31], [32]

### 2.6.3 Registrace

Lícování je komplexní proces mnoha dílčích úkonů, které jsou opakovaně prováděny s cílem najít co nejlepší funkci deformující vstupní obraz vzhledem k referenčnímu obrazu. Zpravidla postupujeme tak, že apriorně určíme parametry ohraničující prohledávaný prostor. Poté buď náhodně, nebo na základě různých pokročilejších algoritmů, začneme transformovat deformovaný obraz a po každé transformaci vypočítáme (dle zvolené metriky) míru kvality transformace, přičemž hledáme nejlepší transformaci. Po nalezení nejlepšího (resp. dostatečně dobrého) řešení ukončíme prohledávání.

Formálněji, registrace je operace formulovaná jako optimalizační problém hledání transformace  $\hat{T}$  minimalizující kriteriální funkci  $\mathcal{C}$ :

$$\hat{T} = \arg \min_T \mathcal{C}(T; I_F, I_m). \quad (2.7)$$

K řešení uvedeného problému lze využít parametrické, nebo neparametrické metody, tato práce pracuje s parametrickými metodami. Typ transformace předem známe, předmětem optimalizace je hledání vektoru parametrů  $\hat{\mu}$ ,

$$\hat{\mu} = \arg \min_{\mu} \mathcal{C}(\mu; I_F, I_m), \quad (2.8)$$

je zjevné, že jde o zjednodušení obecného problému. Místo prohledávání prostoru funkcí prohledáváme pouze prostor parametrů.

## 2.7 Postup hledání transformace

Problém hledání nejlepší transformace lze matematicky formulovat jako nalezení globálního extrému kriteriální (často také z angličtiny *fitness*) funkce dané použitou metrikou v  $n$ -rozměrném prostoru (nejčastěji 2D a 3D). Hodnotu kriteriální funkce můžeme vypočítat pro libovolný transformační vektor, ale neznáme její analytické vyjádření.

### 2.7.1 Úplné prohledávání

Úplné prohledávání stavového prostoru je nejzákladnější metoda. Implementačně jednoduchá, ale výpočetně velmi náročná. Pro pokročilejší (flexibilní) transformace s mnoha parametry nepoužitelná.

### 2.7.2 Gradient descent

Známa optimalizační metoda. V aktuálně procházeném bodě se určuje gradient (numerická aproximace) a do hledaného minima se postupuje proti jeho směru. Problémem této metody je riziko konvergence do lokálního minima, řešením je použití víceúrovňové pyramidální registrace.

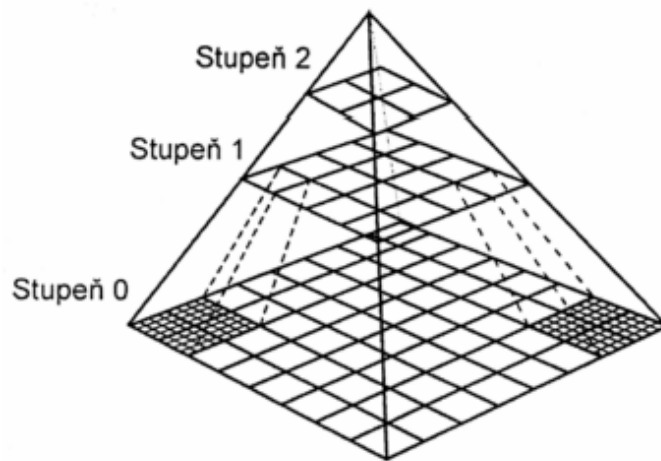
### 2.7.3 Evoluční algoritmy

Evoluční algoritmy často staví na heuristickém přístupu, do popředí se ale dostávají až v posledních letech. Dobrých výsledků dosahují zvláště multiagentní algoritmy. Jejich princip je založen na vygenerování většího množství agentů (každý z nich reprezentuje jeden transformační vektor), kteří jsou na začátku náhodně rozmístěni do celého prohledávaného prostoru a při postupných iteracích směřují do blízkých lokálních extrémů, avšak mohou je za určitých okolností opustit. Je zde předpoklad, že díky podpoře kvalitnějších agentů (selekční tlak) některý z nich skončí v globálním extrému, ale není to zaručeno.

Do této skupiny lze zařadit i simplexovou metodu [35], [32].

### 2.7.4 Pyramidální registrace

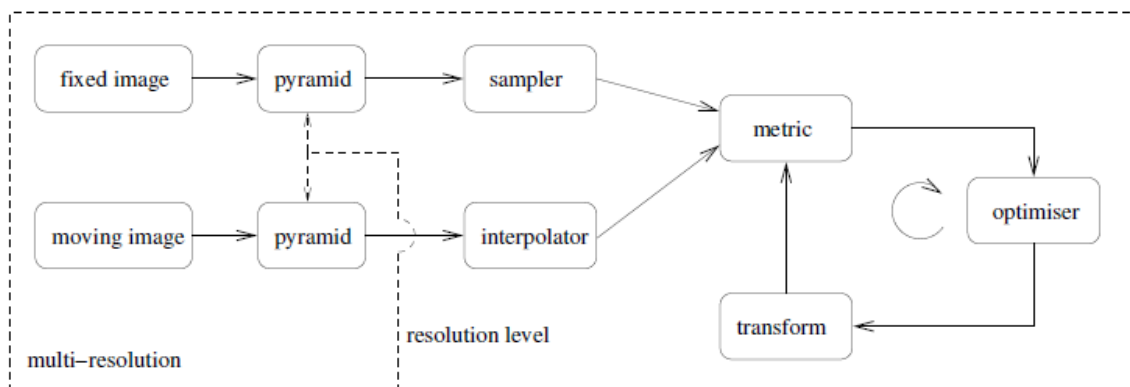
Podstatou pyramidální metody (jak název napovídá) je rozložení celého procesu registrace obrazů do více postupně se zesložitujících částí (schéma je na obr. 2.7). Nejdříve se provádějí ty nejjednodušší (ale nepřesné) operace a pak se program dostane k těm finálním. Registrace obvykle začíná rigidní, případně afinní transformací, jejímž cílem je co nejvíce přiblížit registrované obrazy.



Obr. 2.7: Znárodnění pyramidy s podvzorkovanými snímky

Tato fáze se dále rozpadá do několika dílčích kroků, v prvních krocích jsou registrované snímky rozmazány a podvzorkovány. Rozmazání snižší šanci na uvíznutí v lokálním extrému (použitá metrika díky němu hodnotí shodnost nejvýraznějších objektů v obraze). Podvzorkování snižší výpočetní náročnost, což je výhodné hlavně ze začátku, kdy se prohledává velký stavový prostor. V dalších fázích je obraz rozmazán a podvzorkován méně, přičemž se stále více zužuje prohledávaný prostor pro registraci. Takto se postupuje až do registrace originálního obrazu s plným prostorovým rozlišením.

Po dokončení rigidní nebo afinní transformace se spustí flexibilní transformace, která opět začne rozmazáním, podvzorkováním a postupným kroky se bude zpřesňovat až do finální podoby. Schématicky je celý proces znázorněn na obr. 2.8 [28], [16].



Obr. 2.8: Princip procesu registrace pyramidální metodou v programu elastix [28]

## 2.8 Interpolace

Na počítačích pracujeme s diskretizovanými obrazy. Po provedení přímé geometrické transformace získáme novou sadu pozičních souřadnic, přičemž většina z nich bude vycházet do mezilehlých poloh v původní pravidelné pixelové mřížce. Abychom mohli použít podobnostní kritéria, je třeba přepočítat hodnoty transformovaných pixelů do původní pixelové mřížky. Podobně při zpětné transformaci potřebujeme vypočítat hodnoty zdrojových pixelů v mezilehlých polohách.

Za zlatý standard, který teoreticky dokáže přesně obnovit analogový signál po vzorkování, lze považovat interpolaci pomocí funkce sinc. V praxi je tato metoda (kvůli nekonečné podpoře funkce sinc) nepoužitelná, ale existují různé další metody vzájemně se lišící hlavně kvalitou a výpočetní náročností.

- Metoda nejbližšího souseda – nejjednodušší metoda, nejméně náročná, avšak nepříliš kvalitní. Pixel přebírá hodnotu nejbližšího známého bodu transformovaného obrazu. Jednou z výhod je neměnnost množiny barevných úrovní (výhodné např. u binárních dat, kdy po provedení interpolace požadujeme opět binární snímek).
- Lineární (u 2D obrazů bilineární) – lineární metoda využívá výpočtu koeficientů lineární funkce ze dvou okolních známých bodů a následně výpočet jasové hodnoty mezi nimi. Její modifikace, bilineární interpolace, využívá určení jasu ze zakřivené roviny sestavené z okolních pixelů.
- Kubická (u 2D obrazů bikubická) – velmi často používaná metoda. V principu je stejná jako lineární interpolace, ale využívá polynomy třetího stupně. Bikubická interpolace může být založena na Lagrangeových polynomech, kubických splinech nebo kubické konvoluci.
- Vyšší stupně polynomiální interpolace – jde o zobecnění předchozích 2 metod, avšak vyšší stupně polynomů použité pro interpolaci obvykle nepřináší další zlepšení kvality. Naopak, je zde nebezpečí těžko předvídatelných oscilací (u vysokých stupňů).
- Podle zkrácené funkce sinc – ideálnost neomezené funkce sinc byla uvedena výše. Interpolace podle zkrácené funkce sinc se snaží modifikovat matematicky ideální postup tak, aby byl reálně aplikovatelný. V praxi (v Matlabu) lze implementaci této metody najít pod názvem *lanczos*.

V reálném použití si konkrétní metodu interpolace vybíráme podle účelu. V prvních fázích registrace „na hrubo“ nám obvykle stačí použít rychlou metodu nejbližšího souseda nebo bilineární. Finální obraz, kde požadujeme maximální kvalitu, bývá obvykle interpolován bikubickou metodou [28], [30], [43], [32], [31].



## 2.9 Transformace

Geometrickou transformací obrazu se myslí mapování mezi prostorovými souřadnicemi obrazu  $A$ ,  $\mathbf{r} = (x, y)$ , a prostorovými souřadnicemi transformovaného obrazu  $A'$ ,  $\mathbf{r}' = (x', y')$ . Potom platí  $\mathbf{r}' = \mathbf{T}(\mathbf{r})$ . Z infromatického pohledu můžeme transformace rozdělit na ty, kterými neztrácíme žádnou informaci v obraze, pak můžeme obnovit původní obraz inverzní transformací  $\mathbf{r} = \mathbf{T}^{-1}(\mathbf{r}')$ , a na ty, u kterých zpětnou transformaci provést nelze (ztráta obrazové informace).

Většina používaných transformací je injektivním zobrazením, překážkou pro obnovení obrazu je interpolace diskrétních dat, avšak v praxi to není problém. Vždy se provádí přímá transformace, obnovovat původní obraz není třeba [28]:

$$\mathbf{T} : \Omega_F \subset \mathbb{R}^d \rightarrow \Omega_M \subset \mathbb{R}^d.$$

### 2.9.1 Lineární transformace

**Rigidní transformace** nedeformuje obraz, je možné si ji představit jako pohybování s fotografií po stole. Tuto transformaci lze dále rozdělit na translaci  $\mathbf{T}_\mu(\mathbf{x}) = \mathbf{x} + \mathbf{t}$ , kde  $\mathbf{t}$  je vektor translace. Parametr transformace je definován jako  $\mu = t$ .

Druhou složkou je rotace definovaná jako  $\mathbf{T}_\mu(\mathbf{x}) = R(\mathbf{x} - \mathbf{c}) + \mathbf{c}$ , kde  $R$  je (ortonormální) matice rotace a  $c$  je centrum rotace. Pro dvourozměrné snímky vytvoříme matici rotace jako

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix},$$

přičemž platí  $R^T = R^{-1}$  a  $\det R = 1$ . Parametr transformace je určen úhlem rotace a středem rotace  $\mu = (\theta, c)$  (střed rotace bývá někdy považován za fixní a jako parametr transformace se neuvádí).

Zkombinováním translace a rotace získáme obecnou rigidní (Eulerovu) transformaci

$$\mathbf{T}_\mu(\mathbf{x}) = R(\mathbf{x} - \mathbf{c}) + \mathbf{t} + \mathbf{c}.$$

Maticově lze tuto transformaci (pro 2D snímky) zapsat jako

$$\mathbf{r}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = R\mathbf{r} = \begin{bmatrix} \cos \theta & -\sin \theta & t_x \\ \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}.$$

Ve 2D prostoru je obecná rigidní transformace určena 3 parametry  $\mu = (\theta, t_x, t_y)$ .

**Podobnostní transformace** je velmi podobná rigidní transformaci, ale přidává skalární škálovací parametr  $s$ , kterým násobí matici rotace. Je definována jako

$\mathbf{T}_\mu(\mathbf{x}) = sR(\mathbf{x} - \mathbf{c}) + \mathbf{t} + \mathbf{c}$  a na rozdíl od rigidní transformace umožňuje obraz izotropicky škálovat (tj. měnit měřítko, zvětšovat a zmenšovat).

Další parametry přidává **afinní transformace**, která na rozdíl od předchozích umožňuje zkosení (postupné narůstající translace řádků nebo sloupců). Zkosení ve směru  $x$  a ve směru  $y$  můžeme maticově zapsat jako

$$G_x = \begin{bmatrix} 1 & g_{xy} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, G_y = \begin{bmatrix} 1 & 0 & 0 \\ g_{yx} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Protože nyní připouštíme neizotropní operace, lze předchozí skalární škálovací parametr  $s$  rozdělit do jednotlivých složek a umístit do matice

$$S = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Všechny uvedené složky afinní transformace jsou lineární a lze je aplikovat samostatně v libovolném pořadí, z praktických důvodů ale sestavujeme výslednou matici afinní transformace

$$A = RSG_xG_y,$$

kde  $R$  je výsledná matice rigidní transformace. Transformace obrazu je potom provedena maticovým násobením  $\mathbf{r}' = A\mathbf{r}$  (a interpolací).

## 2.9.2 Nelineární transformace

Pokročilejší flexibilní transformace jsou již nelineární a lze je rozdělit do dvou skupin.

První skupinou jsou projektivní transformace, zachovávají rovinnost povrchů a přímost čar (rovnoběžnost již ale zachována není). Druhou skupinou jsou splinové transformace, které zakřivují přímky i roviny.

**Projektivní transformace** jsou definovány jako

$$\mathbf{r}' = \frac{A\mathbf{r}}{\mathbf{p}^T R + \alpha},$$

kde pro  $\mathbf{p}$  ve 2D prostoru platí  $\mathbf{p}^T = [p_1 \ p_2]$ ,  $A$ ,  $\mathbf{p}$  a  $\alpha$  jsou konstanty. Čitatel je prostá afinní transformace, výsledek transformace závisí na  $R$ . Ve 2D prostoru má projektivní transformace 9 parametrů (ve 3D 16).

**B-splines metoda** je složitější na vysvětlení, ale protože je často doporučovaná pro pružné deformace v programu elastix, uvedu alespoň základní východiska. Celá metoda je založena na kontrolních bodech, které jsou na fixním snímku rozloženy do pravidelné mřížky. B-spliny mají lokální podporu, tj. transformace každého bodu

je vypočítána pouze z několika málo okolních bodů. To je výhodné pro modelování lokální transformace i rychlý výpočet, na druhou stranu, kontrolních bodů mohou být řádově miliony i více (obecně pro rozměrná 3D data). Formálně je B-splínová transformace definována jako

$$\mathbf{T}_\mu(\mathbf{x}) = \mathbf{x} + \sum_{x_k \in N_x} p_k \beta^3 \left( \frac{\mathbf{x} - \mathbf{x}_k}{\sigma} \right),$$

kde  $x_k$  jsou kontrolní body,  $\beta^3(X)$  je kubický multidimenzionální B-splín polynom,  $p_k$  vektor B-splín koeficientů,  $\sigma$  vzdálenosti kontrolních bodů a  $N_x$  je množina všech kontrolních bodů. Na rozdíl od dříve uvedených algoritmů pracujících s význačnými body (features), zde je pro registraci vybíráno velké množství bodů bez apriorní znalosti o obraze.

Další známou metodou pružné transformace je **Thin-plate splines**. Tato transformace je založena na sadě  $K$  landmarků (tj. význačných bodů, terminologie je převzata z [28]) korespondujících mezi fixním  $\mathbf{x}_k^{fix}$  a pohyblivým  $\mathbf{x}_k^{mov}$  obrazem. Transformace je vyjádřena jako suma afinních a pružných komponent:

$$\mathbf{T}_\mu(\mathbf{x}) = \mathbf{x} + \mathbf{A}\mathbf{x} + \mathbf{t} + \sum_{\mathbf{x}_k^{fix}} c_k G(\mathbf{x} - \mathbf{x}_k^{fix}),$$

kde  $G(r)$  je jádro transformace a  $c_k$  koeficienty patřící každému landmarku. Volbou jádrové funkce  $G(r)$  určíme požadované fyzikální chování transformace. Tato metoda je obvykle náročnější než B-splines, ale umožňuje flexibilnější volbu kontrolních bodů [28], [31], [32], [36], [37].

### 2.9.3 Kombinace transformačních metod

Během registrace se postupně může použít více uvedených metod, od rigidní pro základní registraci až po flexibilní registrující lokální odchylky, viz kapitola 2.7.4 Pyramidální registrace.

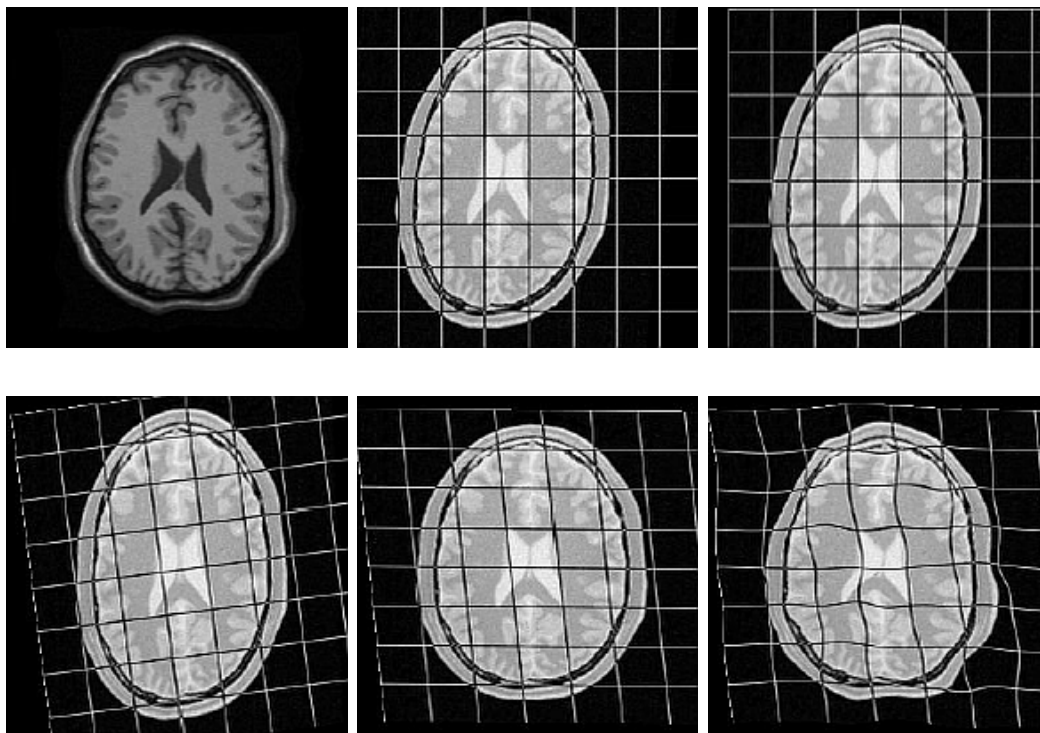
Jednotlivé transformace můžeme kombinovat buď sčítáním

$$\mathbf{T}_\mu(\mathbf{x}) = \mathbf{T}_1(\mathbf{x}) + \mathbf{T}_2(\mathbf{x}) - \mathbf{x},$$

nebo kompozicí

$$\mathbf{T}_\mu(\mathbf{x}) = \mathbf{T}_1(\mathbf{T}_2(\mathbf{x})) = (\mathbf{T}_1 \circ \mathbf{T}_2)(\mathbf{x}).$$

Kompozice obecně vede k lepším výsledkům a je preferovaná [28]. Na obr. 2.9 jsou postupně znázorněny výsledky dosažitelné jednotlivými transformačními metodami.



Obr. 2.9: Porovnání výsledků dosažitelných různými transformačními metodami. Zleva, shora: a) referenční snímek, b) originální registrovaný snímek, c) po translaci, d) po rigidní registraci, e) po afinní registraci, f) po B-spline registraci. Zobrazená mřížka zvýrazňuje provedené transformace, je zde vidět vývoj od horší registrace se zachováním původních proporcí obrazu až po B-spline registraci, která obraz zregistrovala dokonale, ale také ho značně zdeformovala. [28]

#### 2.9.4 Zpětná transformace

Výše uvedené transformační vztahy byly formulovány v základní „dopředné“ podobě, pro praktické výpočty je však často výhodnější využít inverzní transformaci. Zdůrazněme, že doposud uváděný postup vycházel ze souřadnic  $x$  a  $y$ , přičemž byly vypočítávány nové souřadnice  $x'$  a  $y'$ . Problémem tohoto přístupu je, že část vypočtených souřadnic vychází mimo oblast zájmu a interpolovat můžeme až po získání všech hodnot. Pro jednoduchost předpokládejme ortogonální transformaci

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

vypočtené souřadnice  $x'$  a  $y'$  budeme muset ukládat v dočasném seznamu, dokud neproběhne celý výpočet, až pak bude možné provést interpolaci. Některá oblast obrazu bude mít zbytečně příliš vysokou hustotu bodů pro interpolaci a v jiné budou naopak zdrojová data příliš řídká.

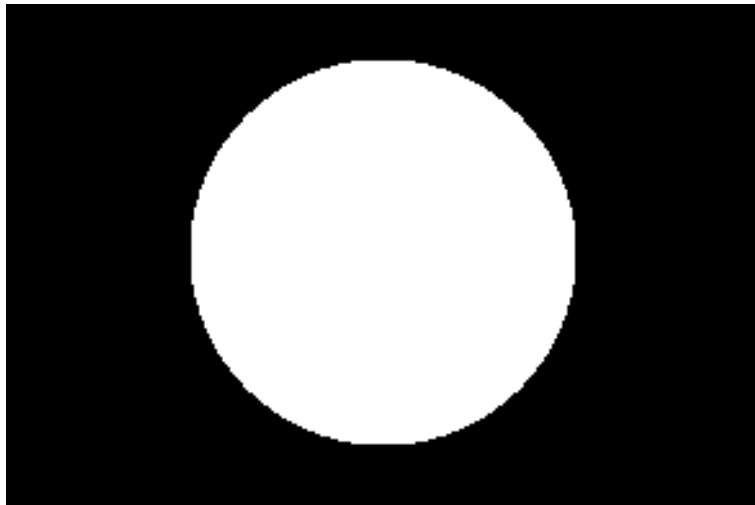
Přístupme k problému opačně, definujme transformaci

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix},$$

yní je parametrem pozice v cílovém obraze, přičemž vypočítáme odpovídající pozici ve zdrojovém snímku. Transformační algoritmus vychází z cílového snímku a vypočítá si zdrojové pozice přesně pro ty body, které potřebuje. Hodnoty těchto bodů jsou získány interpolací ve zdrojovém snímku a přímo dosazeny do výsledného [16].

## 2.10 Maskování

V praxi se často dostáváme do situace, kdy chceme obrazy registrovat pouze podle určité omezené oblasti. Pokud okolní část není vůbec potřeba, je vhodně obraz oříznout a omezit se na ROI. V některých situacích ale není ořez vhodný, např. chceme podle menší ROI registrovat větší obraz nebo má ROI jiný tvar než oříznutelný obdélníkový. V takové situaci se využívají masky, což jsou binární obrazy, kde 1 značí bod, který je podstatný, a 0 bod, který není podstatný.



Obr. 2.10: Příklad masky s ROI ve středovém kruhu. Pro registraci se použije pouze bílá oblast uprostřed, avšak nalezenou transformací je transformován celý snímek.

Dalším případem pro použití masky je „vymaskování“ poškozené části scény vlivem artefaktu, přílišného šumu, přesvětlení apod. Toho je využíváno i v tomto projektu pro potlačení negativního vlivu odlesků osvětlovacích diod v obraze. Masku lze vytvářet adaptivně, např. pro vymaskování přeexponovaných částí snímků lze použít prahování a morfologické operátory.

Příznivým vedlejším efektem maskování je snížení výpočetní náročnosti celého registračního procesu [28].

## 3 DIGITÁLNÍ FILTRACE FUNDUS SNÍMKŮ

Hlavním předmětem zájmu této práce je registrace fundus snímků, avšak praktickým cílem je vytvoření videosekvence, která má co největší diagnostickou hodnotu. Vzhledem k tomuto cíli byly do vytvořeného programu zařazeny některé digitální filtry pro potlačení šumu, ekvalizaci histogramu nebo zvýraznění detailů, jejichž stručný popis je uveden v této kapitole.

### 3.1 Ekvalizace histogramu

Ekvalizace histogramu je neklesající transformace kontrastu, která si klade za cíl vytvořit obraz, který bude mít přibližně rovnoměrné rozložení jasových hodnot v histogramu. Histogram ekvalizovaného obrazu by tedy měl ideálně být plochý a stupně šedi by se v obrazu měly vyskytovat s přibližně stejnou pravděpodobností [32].

Funkci  $T(x)$  transformující jas jednotlivých pixelů lze obecně zapsat jako

$$I_y(n_1, n_2) = T(I_x(n_1, n_2)) \quad (3.1)$$

a v tomto případě globálně aplikované ekvalizace histogramu je určena z celého obrazu (a pro každý pixel je stejná).

### 3.2 Adaptivní ekvalizace histogramu

Problémem předchozí metody je globálnost ekvalizace, která zvýrazňuje nerovnoměrnost v osvětlení a obraz často spíše zhorší. Jako jednoduché řešení se nabízí rozdělit obraz na více menších regionů a každý z nich ekvalizovat samostatně. Tím se odstraní problém nerovnoměrnosti osvětlení, avšak objeví se nový problém, v homogeních oblastech obrazu je příliš zvýrazněn šum.

Řešením je použít adaptivní techniku CLAHE (Contrast-limited adaptive histogram equalization), která po částech ekvalizuje histogram s tím, že kontrast v jednotlivých oblastech je omezován na základě výpočtu lokálních statistik. Velikost omezení kontrastu závisí na parametru *Clip Limit*, který lze uživatelsky upravovat. Detailní informace o matlabové implementaci lze nalézt v manuálu u funkce `adaphisteq` [21], [2], [22].

### 3.3 Gama filtrace

Gama filtrace umožňuje nelineárně transformovat jasové hodnoty obrazu dle

$$I_y(n_1, n_2) = I_x^\gamma(n_1, n_2).$$

Transformační funkce je vždy buď rostoucí a konkávní, nebo klesající a konvexní. Důležité je, že část původního jasového rozsahu dilatuje do větší části nového jasového rozsahu. V takto rozšířené oblasti (buď tmavé nebo světlé) jsou potom lépe prokreslené detaily. Konkrétní výsledek filtrace je závislý na parametru  $\gamma$ , pro  $\gamma < 1$  jsou lépe škálovány světlé oblasti a naopak pro  $\gamma > 1$  tmavé oblasti [32], [31].

### 3.4 Gaussův filtr

Gaussův filtr je zřejmě nejznámější typ konvolučního filtru pro potlačení šumu. Cenou za účinné potlačení šumu je však rozostření hran, což sice může být výhodné pro registraci, ale pro pozorování cév to vhodné není [31].

### 3.5 Wienerův filtr

Zatímco Gaussův filtr funguje jako jednoduchá dolní propust (se všemi svými nedostatky), při návrhu Wienerova filtru bylo pracováno s modelem zkreslení signálu a na jeho základě byl odvozen teoreticky ideální inverzní filtr. Základem pro odvození Wienerova filtru je předpoklad restaurace lineárním systémem a stacionarita náhodných procesů generujících signály. Pro odhadovaný signál  $\hat{x}_w(t)$  platí

$$\hat{x}_w(t) = \int_{-\infty}^{\infty} m(t - \tau)y_w(\tau)d\tau, \quad (3.2)$$

kde  $m(t)$  je hledaná impulsní charakteristika filtru. Potom lze, s využitím principu ortogonality, odvodit

$$M(\omega) = \frac{S_{xy}(\omega)}{S_{yy}(\omega)},$$

což je frekvenční charakteristika optimálního konvolučního filtru pro jakékoliv zkreslení. V praxi však často není možné určit vzájemné spektrum  $S_{xy}(\omega)$ . Proto se zavádí konvoluční model zkreslení signálu s aditivním šumem

$$y_w(t) = \int_{-\infty}^{\infty} h(t - \tau)x_w(\tau)d\tau + \nu(t), \quad (3.3)$$

kde  $x(t)$  je signál a  $\nu(t)$  s ním nekorelovaný šum s nulovou střední hodnotou. Na základě tohoto modelu lze již odvodit tradiční podobu Wienerova filtru [41]

$$M(\omega) = \frac{1}{G(\omega)} \frac{|G(\omega)|^2}{|G(\omega)|^2 + \frac{S_{\nu\nu}(\omega)}{S_{xx}(\omega)}}. \quad (3.4)$$

Ve vztahu 3.4 však stále vystupuje frekvenční charakteristika zkreslujícího filtru a výkonová spektra, která ne vždy známe. Proto různé implementace tohoto filtru

pracují s aproximací jednotlivých proměnných na základě lokálních charakteristik (obvykle střední hodnoty a rozptylu). Matlabová implementace ve funkci `wiener2` používá následující algoritmus:

$$\mu = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} I_x(n_1, n_2), \sigma^2 = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} I_x^2(n_1, n_2) - \mu^2,$$

$$I_y(n_1, n_2) = \mu + \frac{\sigma^2 - v^2}{\sigma^2} (I_x(n_1, n_2) - \mu),$$

kde  $\eta$  je  $N \times M$  okolí každého pixelu a  $v^2$  je průměr všech lokálních rozptylů [20].

### 3.6 Mediánový filtr

Mediánový filtr je známý nelineární filtr, obvykle používaný pro potlačení impulzního šumu. V parametru se zadává velikost okolí pro výpočet mediánu.

### 3.7 Zvětšení prostorového rozlišení

Volba zvětšení prostorového rozlišení nabízí možnost interpolovat existující snímky a zvýšit hustotu vzorkování. K převzorkování je použita interpolační metoda *lanczos*, což je algoritmus vycházející z omezené sinc funkce.

Užitečnost interpolace v tomto kroku je ve zvýšení rozlišení před dalšími registračními kroky a pomocí definované interpolační metody. Podobných výsledků lze ale dosáhnout i později převzorkováním výsledných snímků (a to i real-time při přehrávání videa) [30].



## 4 PŘEDREGISTRAČNÍ ÚPRAVA SNÍMKŮ

Prvním krokem při implementaci programu pro registraci fundus snímků bylo zprovoznění elastixu a otestování kvality registrace se zde používanými snímky, v té době bez dalšího digitálního filtrování. Hned první pokusy bohužel skončily neúspěšně a další snahy zvýšit počet iterací nebo použít jiné registrační metody vedly pouze ke zvýšení výpočetní náročnosti, nikoliv ke zlepšení kvality. Kromě programu elastix byly testovány i funkce dostupné přímo v Matlabu a některé další externí nástroje (různá rozšíření pro Python), avšak žádný z testovaných nástrojů nevedl k uspokojivým výsledkům.

Závěr z těchto experimentů byl takový, že pouhé využití obecného nástroje nebude možné a bude nutné aplikovat další filtry a za použití apriorních informací navrhnout registrační proces pro konkrétní použití.

### 4.1 Odstranění nekvalitních snímků

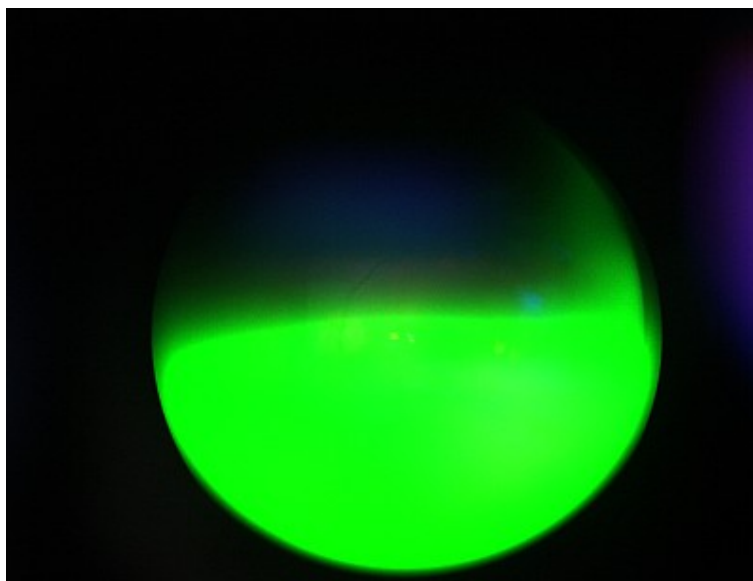
V sériích určených pro zpracování se běžně nalézají jak snímky extrémě přesvětlené (obr. 4.1), tak snímky neosvětlené (obr. 4.2) a samozřejmě různé mezistavy mezi tím. Příčinou je chybné pozicování kamery (jak bylo uvedeno v kapitole 1.3.4 Pozicování kamery), cílem je najít optimální digitální zpracování, které dané odlesky minimalizuje.

Ve zmíněných případech, kdy není šance vyextrahovat nějaké informace, lze snímky před další registrací zcela vyřadit (ale je třeba zvážit porušení jinak ekvidistantní časové osy). Je-li poškozena jenom jedna barevná složka, je vhodnější nahradit ji sousedním snímkem téhož barevného kanálu v lepší kvalitě. Vzhledem k následné interpolaci a fúzi s dalšími složkami to vede k rozumným výsledkům. V začátcích testování bylo zcela vyřazováno cca 1–5 % snímků.

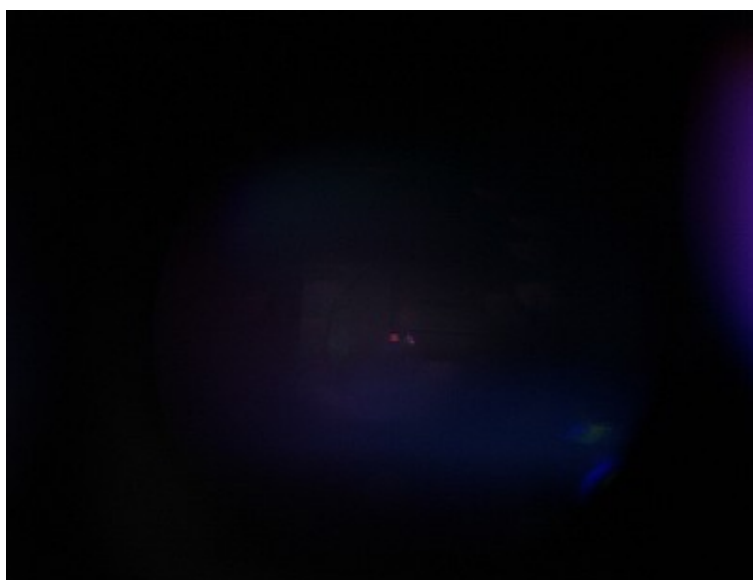
Po dokončení kroku se závěrečným vyvážením jasu (viz obr. 5.9), který nevyrovnaný jas kompenzuje automaticky, již není nutné do snímků manuálně zasahovat.

### 4.2 Registrace s důrazem na cévy

Z porovnání společných vlastností retinálních snímků plyne, že jedinou společnou strukturou, podle které lze rozumně registrovat, jsou cévy. Přitom zastoupení cév je v celém výřezu snímku cca do 5 % px; ukázalo se, že je výhodné během před-registračních úprav tyto cévy zvýraznit. Světlejší středový disk (slepá skvrna) má příliš rozmazané okraje a mezi snímky se často deformuje. Veškeré ostatní pozadí je zatížené značným šumem a registraci spíše poškozuje.

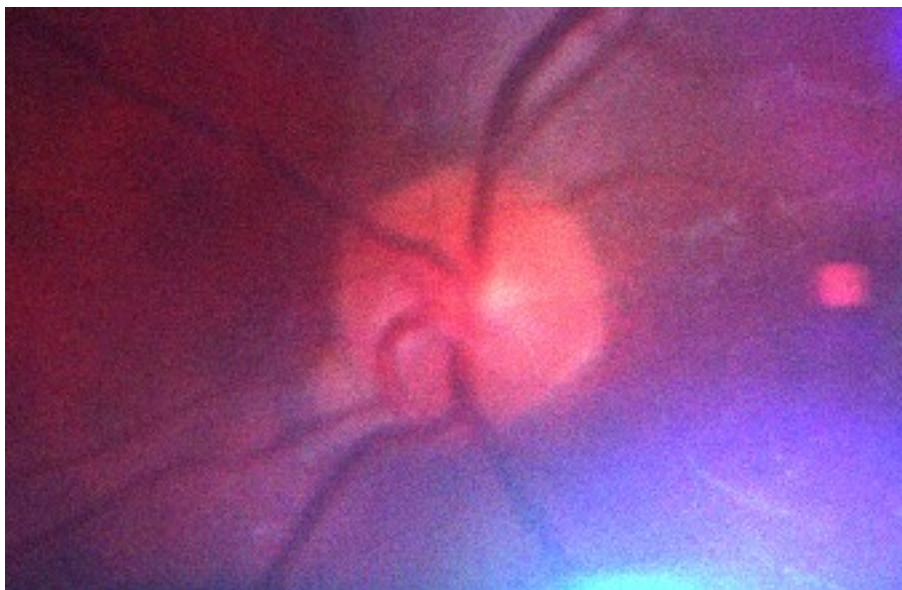


*Obr. 4.1: Příklad neopravitelného snímku. V tomto případě je snímek přesvětlený zelenou diodou.*



*Obr. 4.2: Příklad neopravitelného snímku. V tomto případě je snímek nedostatečně osvětlený.*

Příklad zpracovávaného snímku je uveden na obr. 4.3.



Obr. 4.3: Příklad typického zpracovávaného snímku z experimentální kamery (po oříznutí ROI).

### 4.3 Možnosti detekce význačných bodů a jejich registrace

Jednou ze zvažovaných možností bylo vyzkoušet registraci založenou na význačných bodech, tj. abstrahovat ze snímků vhodné body a na jejich základě registrovat. Značnou (teoretickou) výhodou je šance vyladit 3 modifikované algoritmy, které budou abstrahovat tytéž body následně použité i pro multimodální registraci. Bohužel z analýzy dosavadních výsledků plyne, že k chybám v registraci dochází hlavně ve vertikálním směru, ve kterém je velice obtížné zaměřit vhodné body (zvláště v modré složce). V horizontálním směru, kde by se dalo relativně snadno využít jasového profilu cévy, k problémům nedochází. Z těchto důvodů jsem tuto metodu dále netestoval.

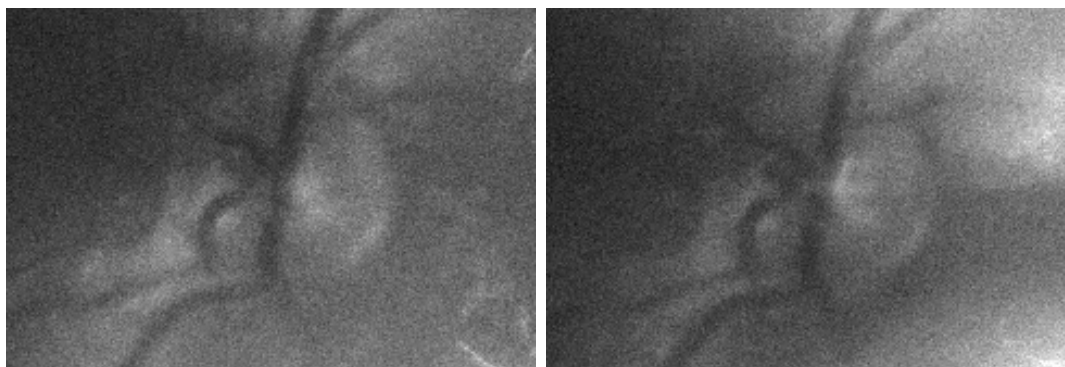
### 4.4 Odstranění šumu filtrem typu dolní propust

Pro odstranění šumu existuje více metod, např. Wienerův filtr popsany v předchozí kapitole. V tomto případě se však ukázalo, že pokročilé metody zachovávající hrany objektů zachovávají i nežádoucí artefakty v obraze. Heuristickým aplikováním různých digitálních filtrů na snímky a hodnocením jasových profilů (viz obr. 4.5) bylo

zjištěno, že nejlepších výsledků dosahuje konvoluční maska o velikosti 7x7 px s hodnotami 1/49 (v angličtině známý *box filter*), která dostatečně potlačuje šum, ale stále ještě ponechává výrazné profily cév (konvoluční maska odpovídající gaussovskému rozložení byla testována, ale nesplnila očekávání). Důsledkem této filtrace je mj. rozostření cév, které se v jasových profilech projeví trojúhelníkovitým průběhem v místech cév, což je pro registraci výhodné.

## 4.5 Potlačení odlesků diod horní propustí

Po odstranění vysokofrekvenčního rušení byla pozornost zaměřena na nízkofrekvenční odlesky způsobené osvětlovacími diodami, které mění jasové poměry v obraze. Lidský zrak se s nimi dokáže vyrovnat velmi dobře, téměř si jich nevšimneme, ale registračnímu algoritmu působí značné potíže.

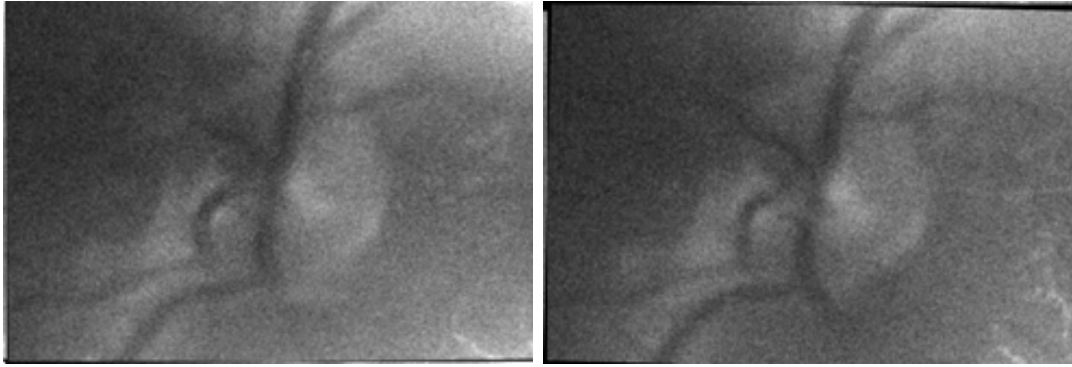


Obr. 4.4: Srovnání různě osvětlených sousedních snímků (40 a 41) modré složky

Ovlivnění jasu u snímků odlišně osvětlených diodami 4.4 je z jasového profilu 4.6 naprosto zřejmé. Dominantní vliv při registraci má pozadí v pravé části snímku a na tmavší cévu v pásmu cca 130 až 140 px je brán pouze minimální ohled, výsledek nesprávné registrace je na obr. 4.5.

Vliv pozadí je do jisté míry potlačen zvolenou maskou, díky které se největší difference v pravé části obrazu při výpočtu kritériální funkce neuplatní, ale i tak zůstávají odlesky značným problémem. Pokud výrazně změním masku a zredukujeme ROI, tak se sice podaří omezit vliv odlesků, ale použitelná část obrazu je již natolik malá, že registrace celkově dosahuje horších výsledků. Projeví se to hlavně neadekvátní rotací, která se obtížně určuje jen podle bodů blízkých středu.

Na základě analýzy odlesků v obraze byla na snímky aplikována jednoduchá horní propust (kvůli výkonu realizovaná násobením ve frekvenční oblasti). Po redukci nízkých harmonických se zbylé jasové hodnoty dostaly i do záporných hodnot a značně se poškodily okraje snímků. To však nepředstavuje velký problém, okraje



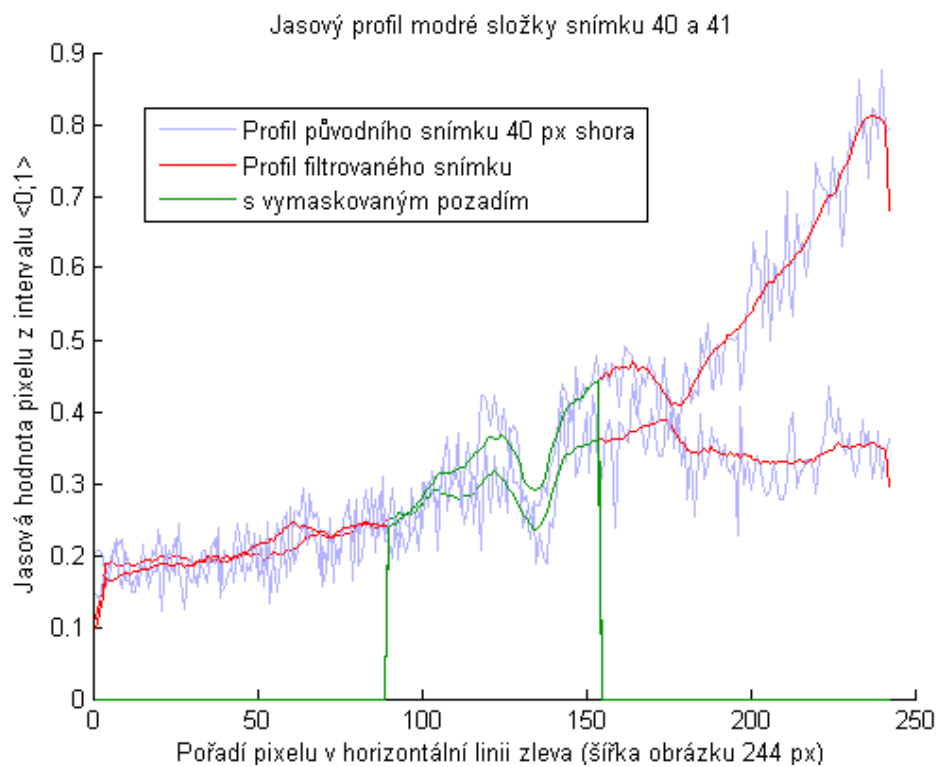
Obr. 4.5: Srovnání výsledků registrace různě osvětlených sousedních blízkých snímků modré složky, odchylka od správné polohy v horní části snímku dosahuje cca 5 px.

jsou stejně vymaskovány a záporné hodnoty pro většinu metrik nepůsobí problémy, pro elastix (hodnoty pixelu se transformují do uint8) stačí přičíst vhodně zvolenou stejnosměrnou složku, se kterou se již elastix vyrovná bez problémů.

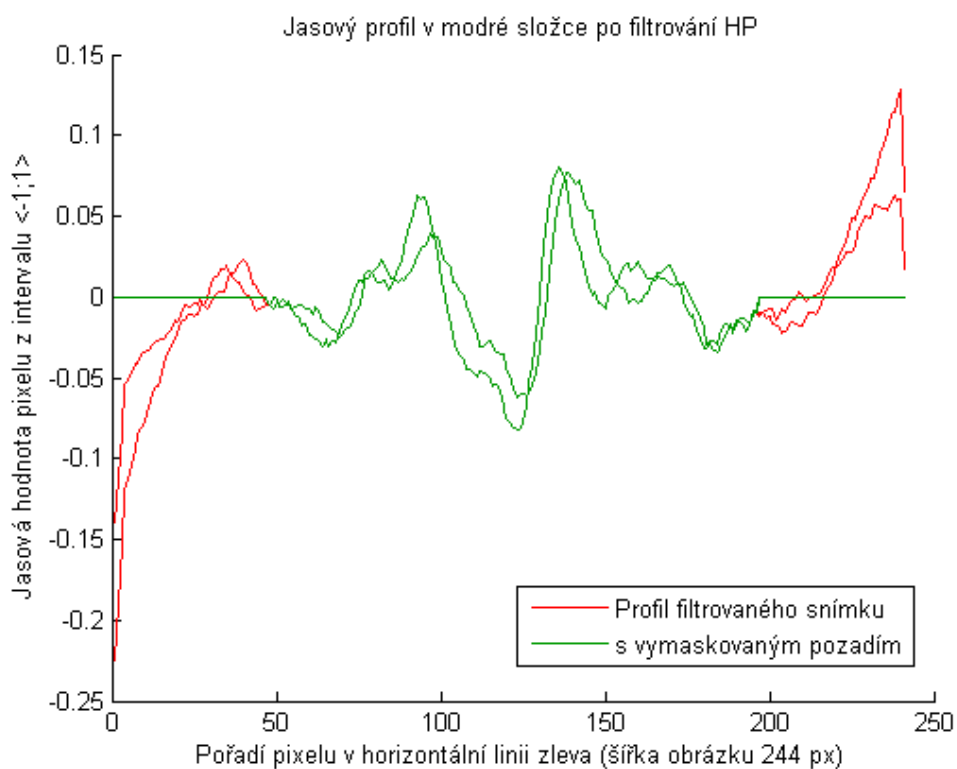
Výsledek popsané úpravy je uveden v grafu 4.7, zelená křivka představuje jasové hodnoty, které se použijí při registraci. Ze zkušeností lze říci, že pro snímky poškozené odlesky byla tato úprava klíčová a po jejím provedení je možné kvalitně registrovat celou sekvenci snímků ve všech barevných kanálech.

## 4.6 Využití filtrace pro multimodalitní registraci

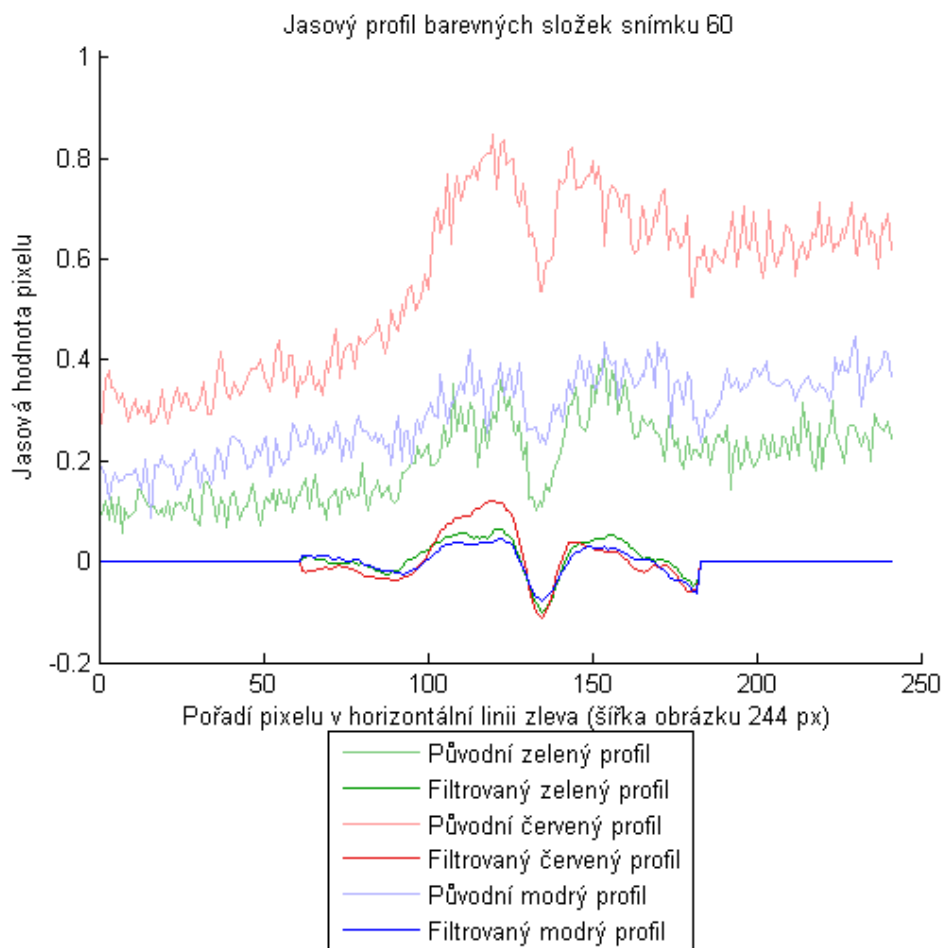
V teoretické části bylo zdůvodněno, proč je nutné přistupovat k registraci mezi různými barevnými kanály jako k multimodalitní registraci. Zde zpracovávané snímky se však neliší natolik jako např. CT + PET (obr. 2.1), všechny jsou pořízeny stejnou optickou kamerou ve viditelném spektru, jenom při osvětlení různými barvami diod. Z analýzy jasových profilů mezi snímky plyne závěr, že výrazné jasové výchyly si velmi dobře odpovídají, jen jsou posunuty o stejnosměrnou složku. Po aplikaci filtrace horní propustí je shoda velmi dobrá a při registraci lze dosáhnout kvalitních výsledků i při použití typicky monomodalitních kritérií. Jasové složky před i po filtraci jsou pro všechny barevné komponenty uvedeny v grafu 4.8.



Obr. 4.6: Jasové profily sousedních snímků v modré složce spolu s vlivem masky



Obr. 4.7: Jasové profily sousedních snímků v modré složce po odstranění nízkých harmonických (zároveň byly odstraněny i příliš vysoké harmonické, viz výše)



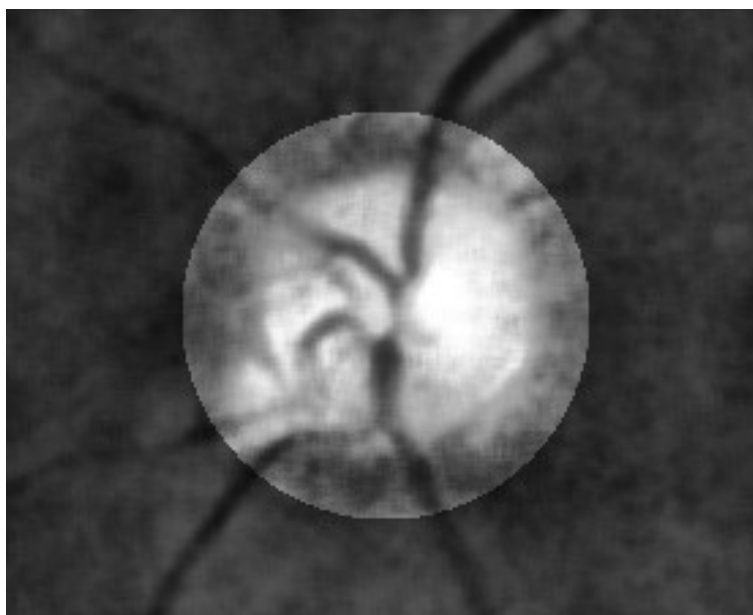
*Obr. 4.8: Porovnání jasových profilů barevných kanálů před a po filtraci. Je zřejmé, že u těchto snímků se jednotlivé barevné kanály liší hlavně stejnosměrnou složkou. Pokud je tato složka odstraněna (filtrace pásmovou propustí), tak filtrované profily dosahují velmi dobré přesné shody a pro jejich registraci by bylo možné použít i monomodální přístup.*

## 4.7 Uživatelská filtrace

Koncept předregistrační filtrace<sup>1</sup> se ukázal být nezbytným pro správnou registraci celé sekvence snímků. Během vývoje této práce byla odladěna zde detailně popsaná pásmová propust, která je považována za dostatečně kvalitní. Přesto je pravděpodobné, že při dalším vývoji by předregistrační filtrace byla jedním z hlavních kandidátů na modifikaci. Proto vytvářená aplikace podporuje použití libovolného uživatelem vytvořeného filtru pro uživatelskou předregistrační filtraci.

### 4.7.1 CLAHE filtrace

Jedním z předem naimplementovaných filtrů je CLAHE filtrace, teoreticky popsaná již v kapitole 3.2 Adaptivní ekvalizace histogramu. Zde ji lze spolu s dolní propustí (protože sama o sobě CLAHE filtrace zvýrazňuje šum) použít pro dosažení výsledků srovnatelných s pásmovou propustí.



Obr. 4.9: Příklad snímku červené složky po provedené předregistrační filtraci filtrem CLAHE a dolní propustí. Ve snímku je navíc naznačeno použití masky. Vnitřní kruh (ROI) je v této podobě použit pro registraci. Jak intenzita filtrů, tak velikost masky jsou plně nastavitelné.

---

<sup>1</sup>Nadále bude předregistrační filtrací nazývána filtrace, která je aplikována na registrované obrazy s cílem najít optimální transformaci. Po jejím nalezení v klasickém registračním procesu je provedena transformace původních nefiltrovaných obrazů a její výsledky jsou vráceny. Tímto se předregistrační filtrace odlišuje od digitální filtrace popisované v kapitole 3, která se aplikuje na originální data a ovlivňuje výslednou podobu obrazů.



## 5 POPIS APLIKACE

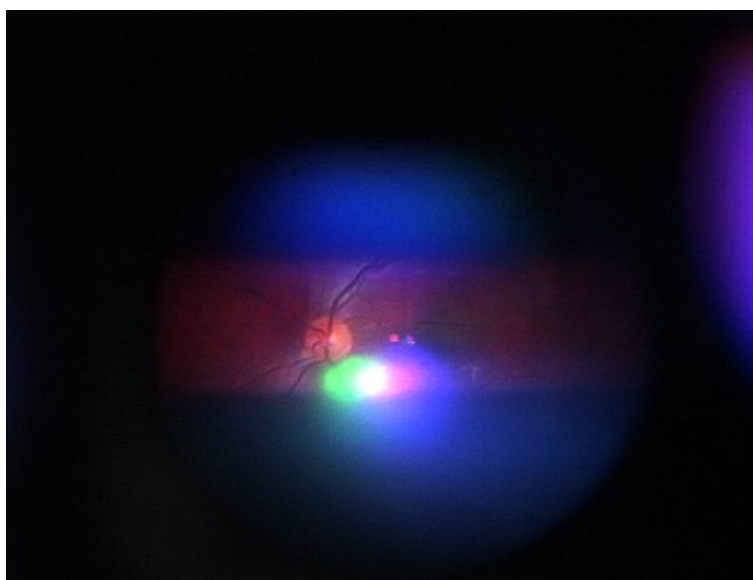
Struktura celé aplikace je rozdělena do několika programových bloků (detailně budou popsány v následující kapitole). Těmito klíčovými funkčními součástmi víceméně odpovídá i návrh grafického rozhraní aplikace a registračního procesu z pohledu uživatele. Jde o následující akce, které jsou vždy vykonávány v uvedeném pořadí.

1. Příprava snímků
2. Registrace RGB kanálů
3. (Filtrace pro zlepšení obrazu)
4. Časová interpolace
5. Fúze snímků
6. (Vytvoření videosekvence)
7. (Analýza obrazu)

### 5.1 Načtení snímků

Vstupní data tvoří několik sérií experimentálních retinálních snímků z fundus kamery. Každá série je tvořena souborem několika desítek barevných PNG obrazů. V těchto obrazech převažuje šumové pozadí, ze kterého je třeba vybrat pouze malou oblast zájmu (ROI) slepé skvrny a blízkého okolí.

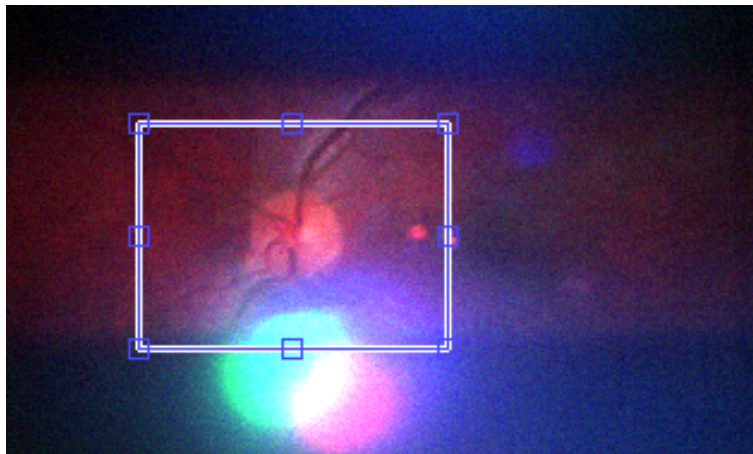
Příklad běžného vstupního snímku je uveden na obr. 5.1, různě se ale objevují i zcela nepoužitelné snímky jako obr. 4.1 nebo obr. 4.2.



*Obr. 5.1: Příklad původního neořezaného snímku, originální rozlišení 1628x1236 px*

## 5.2 Předregistrace

Hledaná oblast slepé skvrny je zadána uživatelským výběrem, viz obr. 5.2, tímto je ponechána možnost určit velikost ROI v závislosti na kvalitě snímků.



Obr. 5.2: Výběr oblasti zájmu (ROI) po předchozím výběru referenčního snímku

Pro přesnost tohoto kroku je důležitý uživatelský výběr referenčního snímku. Je doporučeno vybrat reprezentativní snímek s co nejlepším kontrastem a pokud možno bez odlesků osvětlovacích diod. Vybraná oblast bude následně sloužit jako reference pro hledání ROI ve všech ostatních snímcích.

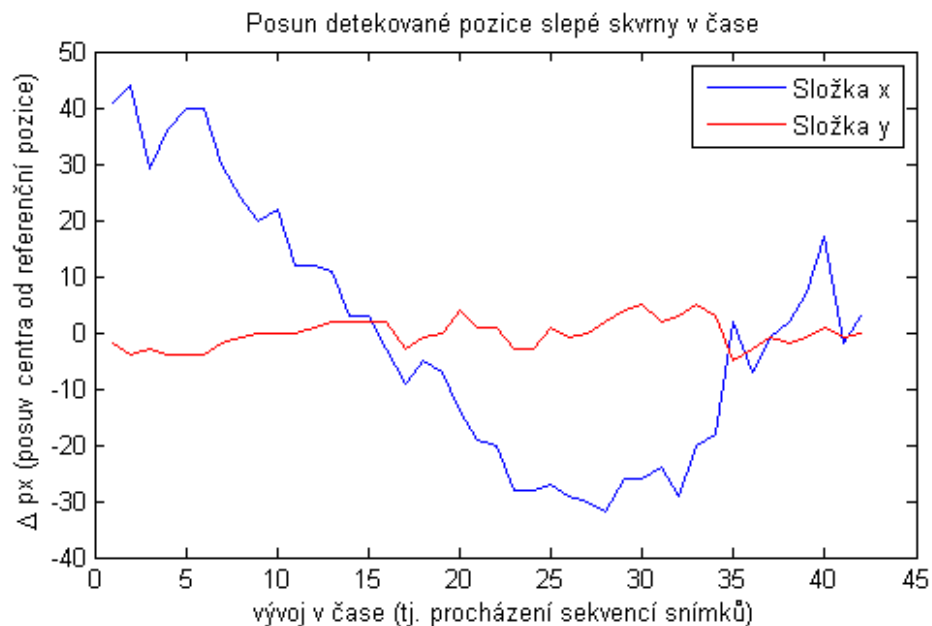
V šedotónovém snímku se pomocí Euklidovské metriky vyhledává nejlepší shoda s referenčním ROI. Využívá se zde apriorní informace z akvizičního procesu o tom, že posuny sousedních snímků jsou malé.

Z grafu 5.3 detekovaných pozic centra slepé skvrny v sérii snímků je patrné, že ve vertikálním směru k pohybu téměř nedochází, zatímco v horizontálním směru se slepá skvrna posouvá až o 150 px. Toho je využito pro volbu obdélníkového prohledávaného okolí.

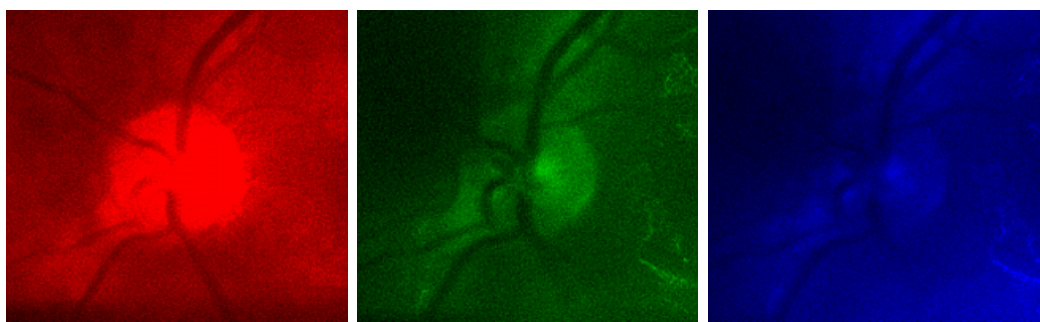
## 5.3 Rozdělení barevných komponent

Dalším bodem je rozdělení jednotlivých barevných komponent. Vstupní barevné snímky v PNG souborech nebyly snímány současně, ale byly uměle sestaveny vždy z trojic po sobě snímáných barevných komponent. Jejich registrace je velmi nepřesná, takže je využita jenom pro předregistraci v šedotónvé škále, ale pro další operace jsou od sebe jednotlivé barevné komponenty odděleny a do dalšího registračního procesu vstupují jako samostatné barevné komponenty.

Na obrázku 5.4 jsou uvedeny oddělené barevné složky v příznivějším případě bez většího poškození odlesky. Na první pohled jsou cévy nejvýraznější v zelené složce



Obr. 5.3: Posun detekované pozice slepé skvrny v čase



Obr. 5.4: Jednotlivé barevné složky

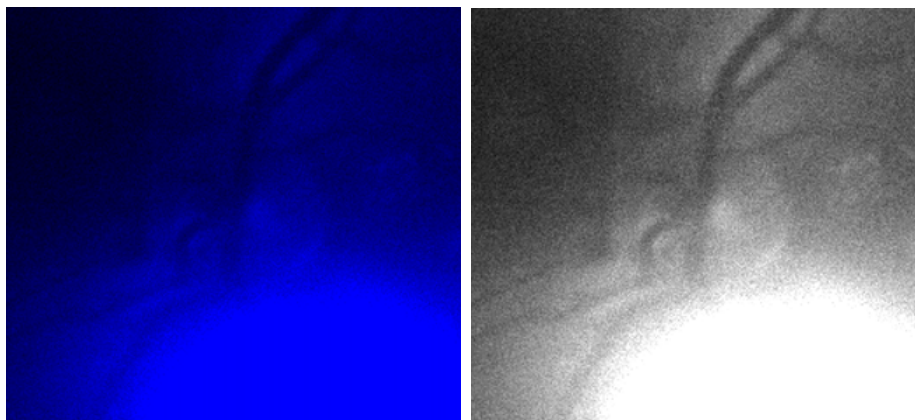
(což bylo zdůvodněno v kapitole 1.6 Kontrast zelené složky), relativně použitelná je ještě červená složka. Kontrast v modré složce je nejhorší.

## 5.4 Registrace barevných kanálů

Účelem registrace barevných kanálů je zafixovat obraz, tj. registrovat všechny obrazy co nejpřesněji na původní ROI vybrané uživatelem. Dalším krokem bude časová interpolace celé sekvence, proto je nezbytné přesně registrovat po sobě jdoucí snímky tak, aby pixely na shodných pozicích skutečně reprezentovaly totožné objekty ve scéně.

Mezi nejdůležitější nastavení v tomto kroku patří:

- volba registračního programu (elastix nebo interní implementace),



*Obr. 5.5: Příklad značného poškození odlesky od osvětlovacích diod v původní modré a pak šedotónové reprezentaci*

- rozhodnutí mezi registrací k referenčnímu nebo sousednímu snímku
- a volba, jestli registrovat barevné kanály se shodným, nebo odlišným nastavením.

Nejlepších výsledků lze dosáhnout pomocí registrace programem elastix.

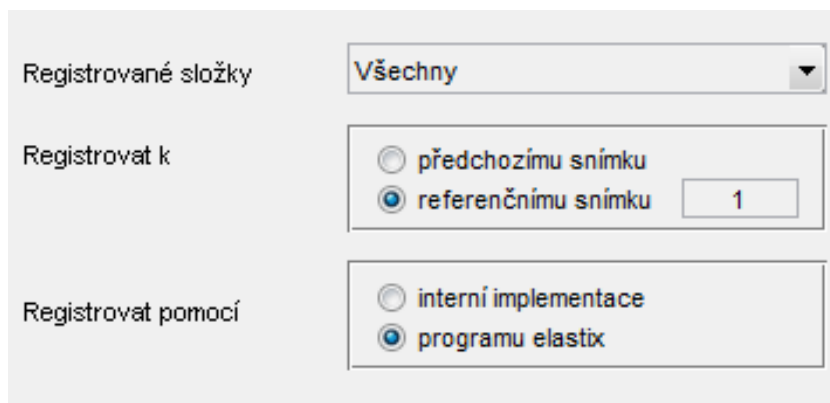
Rozhodnutí, jestli registrovat k referenčnímu nebo sousednímu snímku, je otázkou kompromisu. Pro následnou interpolaci a celkově dobrou návaznost by bylo lepší registrovat k sousednímu snímku, avšak vzhledem k nízké kvalitě vstupních dat je vhodné registrovat ke společnému referenčnímu snímku. Ze zkušeností lze říci, že se ve většině testovacích sekvencích najde snímek, který není registrován zcela přesně a pokud je reference nastavena na předchozí snímek, všechny další snímky v řadě ho pak následují, tj. dochází ke kumulaci chyb. Po ruční kontrole a případné úpravě je možno tuto volbu použít (případně jen pro vybrané kanály).

I když je zde možnost pro registraci jednotlivých kanálů nastavit konfiguraci odděleně, obvykle to není příliš potřeba. Možné využití je například pro afinní registraci obecně kvalitnější zelené nebo červené složky, naopak u modré složky nemá cenu zkoušet cokoli složitějšího než rigidní Eulerovu transformaci.

Uživatelské rozhraní pro nastavení registrace je uvedeno na obrázku 5.6.

## 5.5 Časová interpolace

Cílem časové interpolace je v první řadě získat snímky všech barevných kanálů v týchž časových okamžicích a v druhé řadě zvýšit snímkovou frekvenci budoucího videa. Detaily, které nejsou ve vstupních datech zachyceny, dodatečně zjistit nelze, ale můžeme dopočítat interpolované hodnoty do časových okamžiků, ve kterých bude vytvářen výsledný snímek a zvýšit tím přesnost. Interpolací na vzorkovací frekvenci



Obr. 5.6: Nastavení registrace komponent

10 fps a výše dosáhneme plynulé videosekvence bez viditelně trhaných pohybů.

Vzhledem k charakteru dat je není z principu možné aproximovat žádnou charakteristickou funkcí, proto byla pro interpolaci zvolena metoda kubických splinů, od které lze očekávat rozumné výsledky. Technická realizace je přímo podporovaná Matlabem a bude popsána v části zabývající se implementací.

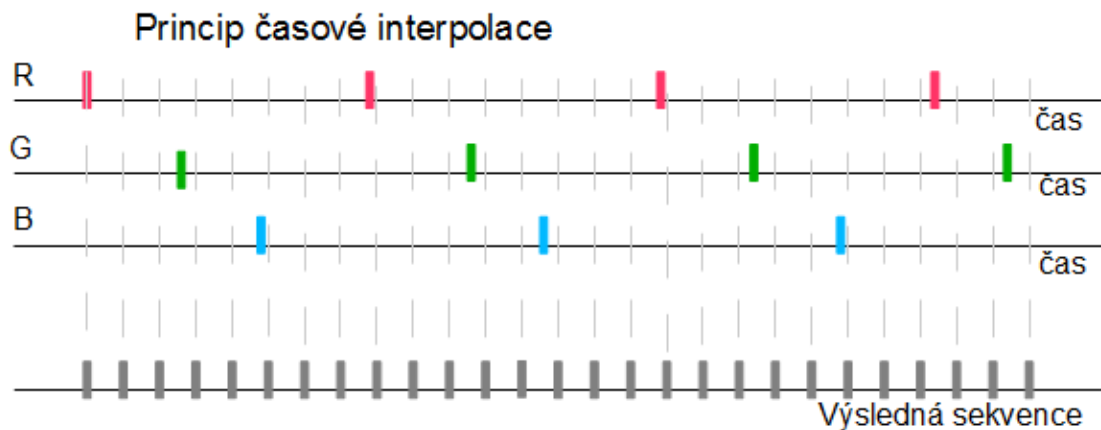
Schématicky je časová interpolace naznačena na obrázku 5.7. V horní části jsou uvedeny časové osy snímání jednotlivých R, G, B komponent a ve spodní výsledné interpolované snímky<sup>1</sup>. Barevnými značkami jsou označeny známé snímky. Vždy mezi nasnímanou trojicí je menší pauza (to je dáno způsobem akvizice, na novějších sadách dat tato pauza již není, schéma ukazuje obecnější případ). Výsledné video musí být plynulé a mělo by proporciálně respektovat reálnou kadenci snímkování, proto je nutné s touto pauzou počítat a interpolovat do ní více snímků. Celkový počet nových snímků závisí na skutečném čase snímání (dáno akvizicí) a požadované vzorkovací frekvenci budoucího videa.

Na obr. 5.8 je uvedeno uživatelské prostředí pro zadání jak informací o akvizici, tak i požadavků na výslednou videosekvenci. Pro informaci obsluhy a kontrolu jsou ihned vypočítány parametry výsledné sekvence, jako počet snímků nebo celkový čas.

## 5.6 Registrace snímků

Po časové interpolaci jsou již v jednotlivých barevných kanálech ve správných časových okamžicích připraveny snímky pro pozdější seskládání do barevných snímků. Než k tomu dojde, je nutné provést registraci mezi jednotlivými barevnými kanály

<sup>1</sup>Ekvidistantně vzorkovaná osa výsledné sekvence představuje vzorkovací frekvenci výsledného videa. Společný výsledný kanál je uveden pouze pro názornost, až do další registrace v rámci snímku se stále pracuje se třemi barevnými kanály odděleně, ale již jsou registrovány do shodného časového okamžiku.



*Obr. 5.7: Schéma časové interpolace*

samostatně pro každý snímek budoucí sekvence. Protože u retinálních snímků dosahuje nejlepšího kontrastu zelená složka (tj. zelená složka je z předchozích kroků nejlépe zaregistrována i časově, což potvrzují praktické zkušenosti), je zelený obraz volen jako referenční (fixní) a postupně je na něj nalícován červený a modrý obraz. Pro tyto účely je již nezbytné použít multimodální registraci.

Uživatelské nastavení umožňuje volit mezi interní implementací registrace a registrace pomocí elastixu. Interní implementaci nelze dále nastavovat, pracuje s úplným prohledáváním a jako kriteriální funkci používá vzájemnou informaci tak, jak bylo definováno v kapitole 2.5.3 Vzájemná informace. Naopak registrace pomocí programu elastix je široce nastavitelná tak, jak bylo popsáno v kapitole 6.6 Rozhraní programu elastix, a vzájemná informace je zde vybrána jako výchozí metrika.

## 5.7 Seskládání výsledné sekvence snímků

Princip sestavování výsledné sekvence byl již naznačen v předchozích kapitolách, především v kapitole 5.5 ve schématu 5.7. Shrňme, že cílem je vytvoření plynulé videosekvence, tj. nainterpolování snímků s dostatečnou vzorkovací frekvencí (fps).

Snímky barevných komponent již nejsou brány po nedělitelných trojicích jak tomu bylo u vstupních dat, ale virtuální čtecí hlava se pohybuje vždy o jediný snímek (jednu barevnou komponentu), zachovává zbylé 2 barevné komponenty z předchozího snímku a vytváří nový barevný snímek. Počet snímků je dle nastavení ještě před seskládáním dále navýšen interpolací.

Barevné složky, které jsou slučovány do jednoho barevného obrazu, byly vzájemně slícovány tak, jak je popsáno v 5.6 Registrace snímků.

Ze vzniklé 3D matice (3 nově vzniklé vrstvy jsou tvořeny původně oddělenými RGB složkami) Matlab snadno vytvoří výstupní barevný snímek, který uloží na disk.

|                            |                          |
|----------------------------|--------------------------|
| Odstup červené složky [s]  | 0.3                      |
| Odstup zelené složky [s]   | 0.2                      |
| Odstup modré složky [s]    | 0.2                      |
| Požadovaná frekvence [fps] | 10                       |
| Dvouprůchodová interpolace | <input type="checkbox"/> |
| Vstupních snímků           | 80                       |
| Výsledných snímků          | 556                      |
| Velikost snímku            | 261x261 px               |
| Celková délka              | 55.6 s                   |

Obr. 5.8: Nastavení časové interpolace

Příklad výsledného snímku je uveden např. na obr. 5.11.

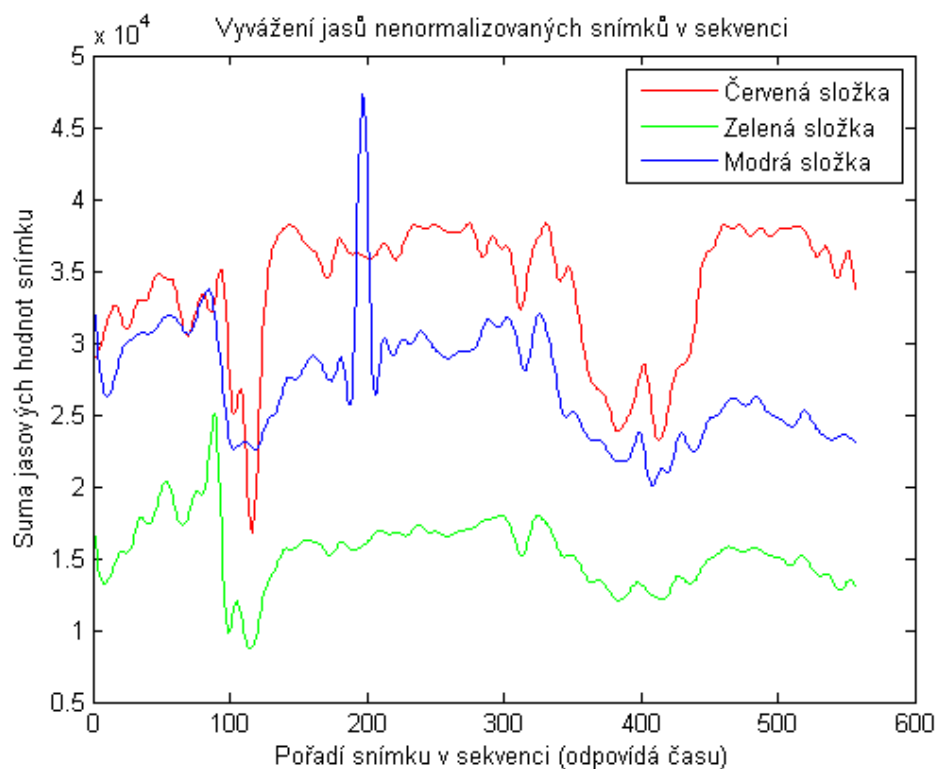
V tomto bodě je dokončen samotný registrační proces, avšak na základě hodnocení vytvořených videí se ukázalo, že bude rozumné ještě upravit úroveň jasu. Osvětlení scény během doby akvizice je značně nerovnoměrné, což vede k velmi nevyváženému jasů v průběhu přehrávání videa, jak je uvedeno v grafu 5.9. Proto byl do úpravy snímků přidán další krok, který umožní vypočítat medián jasů každé barevné složky, normalizovat všechny snímky v řadě a případně na základě uživatelské volby některou ze složek potlačit, nebo naopak zvýraznit. Možnosti uživatelského rozhraní jsou uvedeny na obr. 5.10.

Celkové schéma znázorňující celý registrační proces je na obr. 5.12.

## 5.8 Vytvoření videosekvence

Posledním volitelným krokem tohoto projektu je vytvoření videa ze sekvence snímků. Přestože pro určitou část diagnostiky je nezbytná možnost zobrazit si statické snímky a na jejich tvorbu a úpravy byl v celé práci kladen hlavní důraz, pro jinou část diagnostiky (sledování pulzace cév) je zase nezbytné přehrání videosekvence. Vytvoření videa z připravené sekvence snímků doplňuje možnosti aplikace a společně se statickými snímky tvoří exportovatelné výstupy aplikace.





Obr. 5.9: Nevývážený jas v průběhu videosekvence. Vyrovnáním jasů jde velmi dobře kompenzovat jak dlouhodobější nedostatky akvizice, tak i velká krátkodobá poškození jednoho kanálu (zde např. modrá složka, cca 200. snímek).

Dodatečná zpracování:

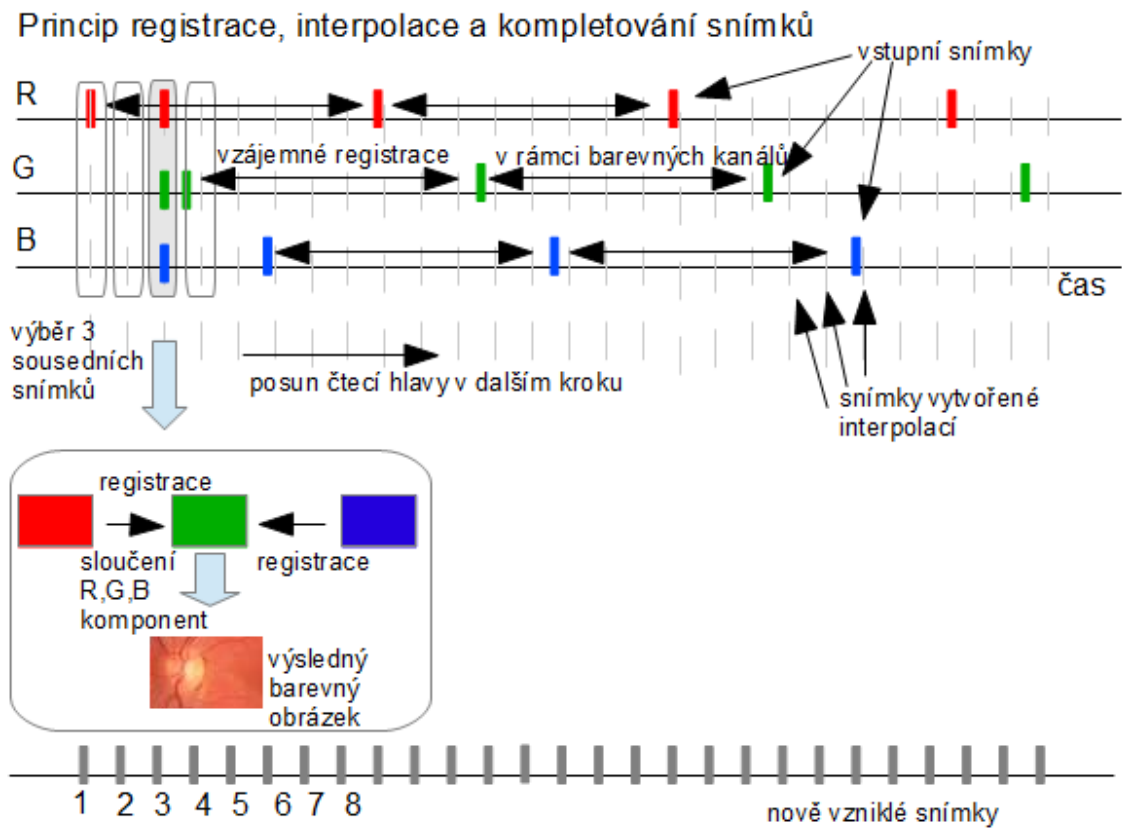
|                                 |                                     |
|---------------------------------|-------------------------------------|
| Dokončit druhý krok interpolace | <input checked="" type="checkbox"/> |
| Normalizovat jas v sekvenci     | <input checked="" type="checkbox"/> |
| Červená složka                  | 1.0                                 |
| Zelená složka                   | 1.0                                 |
| Modrá složka                    | 1.0                                 |
| Celkový jas                     | 1.0                                 |

Obr. 5.10: Nastavení vyvážení jasů ve videosekvenci. Dodatečné úpravy jednotlivých kanálů se zadávají relativně vzhledem k mediánu jasových hodnot celé sekvence.





Obr. 5.11: Výsledný RGB obraz vytvořený fúzí barevných komponent (odpovídá jednotlivým komponentám uvedený na obr. 5.4). (Přinejmenším v elektronické verzi je u levého a horního okraje pozorovatelná hranice při registraci správně posunutého modrého kanálu.)

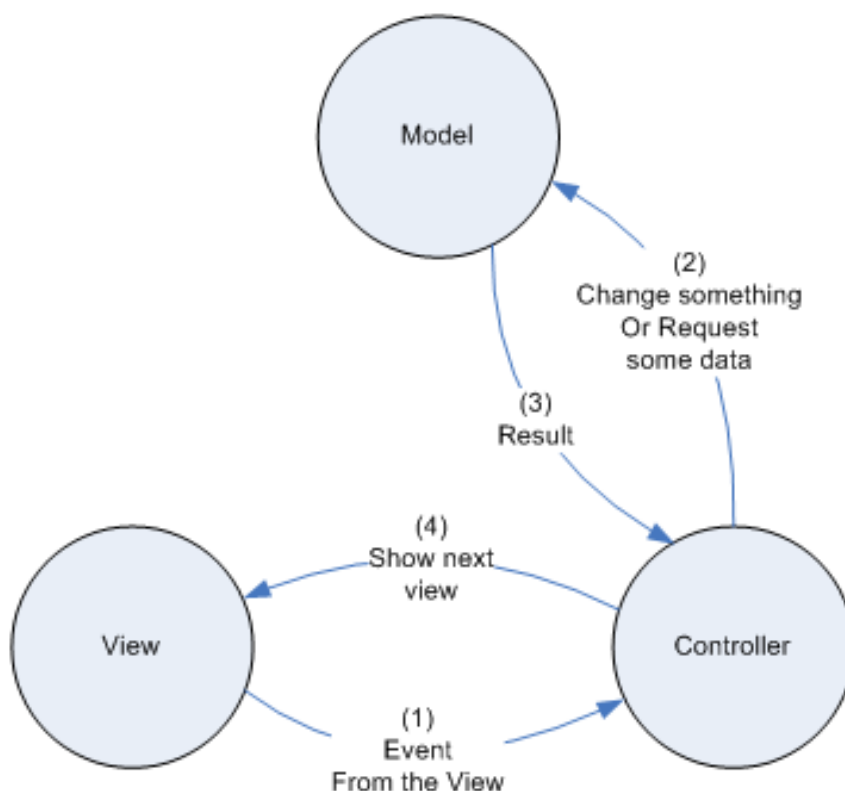


Obr. 5.12: Shrnutí principu registrace, interpolace a kompletování snímků.

## 6 PROGRAMOVÁ IMPLEMENTACE

### 6.1 Celková koncepce

Implementačně je aplikace tvořena zastřešujícím jádrem doplněným (do určité míry) nezávislými podprogramy. Robustní jádro napsané v Matlabu zajišťuje celý proces zpracování dat od surových snímků až po vytvoření videosekvence. Jeho úkolem je především práce se soubory, příprava snímků a v přímé součinnosti s GUI generování konfigurací a spouštění externích programů. V návrhovém MVC modelu<sup>1</sup> odpovídají aplikačnímu jádru s grafickým uživatelským rozhraním vrstvy Controller a View. Komplexní (a pro tuto práci klíčové) operace, tj. registrace snímků a vytvoření videa, jsou napsány jako samostatné a nahraditelné podprogramy volané společným jádrem. Zatímco jádro aplikace je napsané výhradně v Matlabu, jednotlivé moduly využívají i externí programy a v Matlabu mají naimplementované pouze komunikační rozhraní. Tyto podprogramy tvoří Modelovou vrstvu.



Obr. 6.1: Schéma MVC modelu pro návrh všech částí aplikace (specializovaná část aplikace pro registraci obrazů, uživatelské rozhraní a řízení celého procesu) [18].

<sup>1</sup>MVC (Model View Controller, viz obr. 6.1) je návrhový vzor oddělující výkonnou a prezentační vrstvu aplikace.

## 6.2 Externí aplikace

Každý použitý externí program je ovladatelný z příkazové řádky (CLI aplikace, z anglického Command-line interface). Pro složitější konfigurace se využívají parametry uložené do konfiguračních souborů a spouštěnému programu se pak předává pouze cesta k těmto souborům. Komunikační rozhraní v tomto případě zajišťuje mapování matlabových struktur do formátů jednotlivých programů, uložení konfigurace na disk a spuštění programu spolu s předáním potřebných parametrů. Spuštění externího programu z Matlabu přes rozhraní příkazové řádky umožňují systémová volání.

Po vykonání požadované operace je třeba provést opačný proces. Externí program uložil nalezené výsledky do souborů a další výstupy vrátil do výstupního streamu. Všechny tyto informace jsou dostupné pouze jako texty (resp. binární data) a pro jejich efektivní vyhodnocení je nezbytné je naparsovat zpět do připravených matlabových datových struktur. Teprve po tomto zpracování máme k dispozici informace o tom, jestli volání programu uspělo spolu s vrácenými výsledky (registrační vektor, registrovaný obrázek), nebo naopak chybové kódy a hlášky oznamující příčiny neúspěchu.

Výsledkem je, že z vnějšího pohledu se volání externích programů chová jako soubor standardních matlabových funkcí. Lze je libovolně volat bez znalosti rozhraní cílové aplikace a předávat jim matice s obrazovými daty. A naopak, jako návratové hodnoty přijímáme již zpracované výsledky ve formátu podporovaném Matlabem.

Rozhraní klíčových funkčních podprogramů bylo navrženo s ohledem na obecnost a rozšiřitelnost. Přidání další volitelné implementace je otázkou rozšíření konfiguračního souboru (s tím souvisí drobné úpravy v GUI) a vytvoření kompatibilního rozhraní.

## 6.3 Grafické a programové uživatelské rozhraní

Pro snadné použití aplikace bylo vytvořeno grafické uživatelské rozhraní (GUI), které zajišťuje především práci se soubory (výběr snímků pomocí běžného Průzkumníku), interaktivní ořez snímků, snadnou konfigurovatelnost a postupné spuštění jednotlivých modulů.

Záměrem je umožnit začlenění vytvořeného softwarového produktu i do jiných aplikací, proto je veškerá podstatná činnost naimplementována v rámci jádra aplikace a GUI tvoří pouze nejvyšší vrstvu komunikace s uživatelem. Bylo by snadné vedle něj vytvořit programovou API vrstvu pro ovládání aplikace z jiného programu.

## 6.4 Jádru aplikace

Jádru aplikace je klíčovou součástí, která zajišťuje celý registrační proces. Na vstupu přijímá surová data z experimentální videokamery, do pracovního adresáře si uloží jejich kopie pojmenované dle interní konvence a postupně distribuuje požadované snímky do dílčích bloků pro zpracování. Obrazová data nejsou po celou dobu držena v operační paměti, ale jsou vždy ukládána do k tomu určených pracovních adresářů na disk. Toto řešení přináší dvě výhody a jednu nevýhodu. První výhodou je omezení požadavků na operační paměť, aplikace má (kromě části s časovou interpolací<sup>2</sup>) konstantní požadavky na paměť. Druhou výhodou je kontrola činnosti programu, neboť je možné snadno si prohlédnout snímky uložené v jednotlivých fázích registrace. Určitou nevýhodou je potřeba dodatečného výkonu pro režii při načítání a ukládání snímků a přístup na HDD, ta je ale vzhledem k náročnosti samotné registrace zanedbatelná.

## 6.5 Registrace obrazů

Pro samotnou registraci je preferován externí program `elastix`, který považuji za nejkvalitnější dostupný program pro registraci. Pokud není na dané pracovní stanici k dispozici, je možné si vybrat i registraci pomocí rigidní implementace napsané v Matlabu, ta ale podporuje pouze nejzákladnější operace. V dalších kapitolách se na registraci již pohlížejí jako na předdefinovaný proces, který je zde detailněji popsán.

### 6.5.1 elastix

Zájemce o detailní prostudování rozhraní programu `elastix` [27] odkazují na dokumentaci k tomuto programu [28] a případně na detailní popis API na stránkách programu.

Program `elastix` je CLI aplikací, tj. standardně se spouští přes příkazovou řádku.

Typické volání programu `elastix` je

```
elastix -f fixedImage.ext -m movingImage.ext -out  
outputDirectory -p parameterFile.txt,
```

význam jednotlivých parametrů je zřejmý, `fixedImage.ext` a `movingImage.ext` jsou obrazy v podporovaných formátech, `outputDirectory` je adresář, kam jsou uloženy výsledky a `parameterFile.txt` je textový soubor obsahující konfiguraci registrace. K této nejzákladnější podobě se často přidávají alespoň parametry pro použití masky.

---

<sup>2</sup>Interpolační krok je možné za cenu zhoršení efektivity přepsat do podoby s konstantní paměťovou složitostí, pro testovaná data to ale na běžném počítači nebylo potřeba.

Zde uvádím krátkou ukázkou z konfiguračního souboru, cílem je pouze demonstrovat formát konfigurace, kompletní příklady lze nalézt v manuálu [28] a obsáhlý přehled všech podporovaných nastavení je uveden na <http://elastix.isi.uu.nl/dxygen/parameter.html>.

```
//ImageTypes
//Components
// ***** Pyramid
// Total number of resolutions
(NumberOfResolutions 3)
// ***** Transform
//(CenterOfRotation 128 128) center by default
(AutomaticTransformInitialization "true")
(AutomaticScalesEstimation "true")
(HowToCombineTransforms "Compose")
// ***** Optimizer
// Maximum number of iterations in each resolution level:
(MaximumNumberOfIterations 300 300 600)
(AutomaticParameterEstimation "true")
(UseAdaptiveStepSizes "true")
// ***** Metric
```

Výsledkem volání programu elastix je kromě samotného registrovaného obrazu a různých logů o provedené činnosti také soubor s nalezenými transformačními parametry, který lze použít k zopakování transformace.

## 6.5.2 transformix

Transformix je program vyvíjený společně s elastixem, který slouží k transformaci obrazu dle zadaných parametrů. Tyto parametry jsou typicky vygenerovány programem elastix, jak bylo popsáno v předchozím odstavci.

Účelem je umožnit zopakování transformace na jiná data, než byla použita pro původní registraci. Příkladem může být registrace specificky filtrovaných dat (v této práci např. horní propust) a po nalezení optimálního transformačního vektoru se provede transformace původních nefiltrovaných dat. Dalšími příklady použití je transformace masky nebo label obrazu při segmentaci.

Implementačně je práce s programem `transformix` velmi podobná programu `elastix`:

```
transformix -in image.ext -out outDir -tp TransformParameters.txt.
```

### 6.5.3 ElastixFromMatlab Toolbox

Pro komunikaci s programem elastix z Matlabu je použit *ElastixFromMatlab toolbox* [40]. Jde v podstatě o wrapper (mezivrstva sloužící k volání určité knihovny) nad programem elastix, který prostřednictvím sady matlabových funkcí pro registraci obrazů umožňuje pohodlnější práci s programem elastix.

Podstatou činnosti toolboxu je vytvoření matlabových struktur, mj. `StrDataX` pro reprezentaci obrazových dat spolu se `StrToolboxConf` pro uložení konfigurace, a souboru funkcí, které s těmito strukturami pracují. Registrační proces se pak skládá z vytvoření požadovaných struktur (konverzí z běžných matlabových struktur, typicky matic pro obrazová data) a jejich exportu do datových `*.mhd` a konfiguračních souborů, které se uloží v pracovním adresáři. Po přípravě pracovního prostředí se zavolá program elastix a jako parametry se mu předají názvy odpovídajících vygenerovaných souborů.

Klíčovou funkcí volající elastix je `elxElastix` s následující hlavičkou:

```
[StrDataXRegMoving, Transforms, Log, Success, Message] = elxElastix(  
myConf, myParam, FixedImages, MovingImages, 'FixedMask', FixedMask,  
'MovingMask', MovingMask, 'InitialTransform', InitialTransform)
```

Z této jinak obsáhlé funkce uvádím jen řádek

```
[CommandStatus, DummyResult] = system(Cmd);
```

který obsahuje systémové volání dříve sestaveného příkazu v proměnné `Cmd`, tento řádek propojuje Matlab s programem elastix prostřednictvím systémového volání podporovaného konkrétním operačním systémem.

Po provedení registrace program elastix uloží veškerá data do výstupního adresáře, odkud jsou pomocí dalších funkcí toolboxu zpětně naparsovány do pracovních struktur, kde je lze dále využít.

ElastixFromMatlab Toolbox samozřejmě pracuje nejenom s programem elastix, ale i s programem transformix. Strukturu reprezentující transformaci nalezenou v předchozím volání `elxElastix` lze ihned použít pro volání `elxTransformix`.

### 6.5.4 Vlastní vrstva nad toolboxem

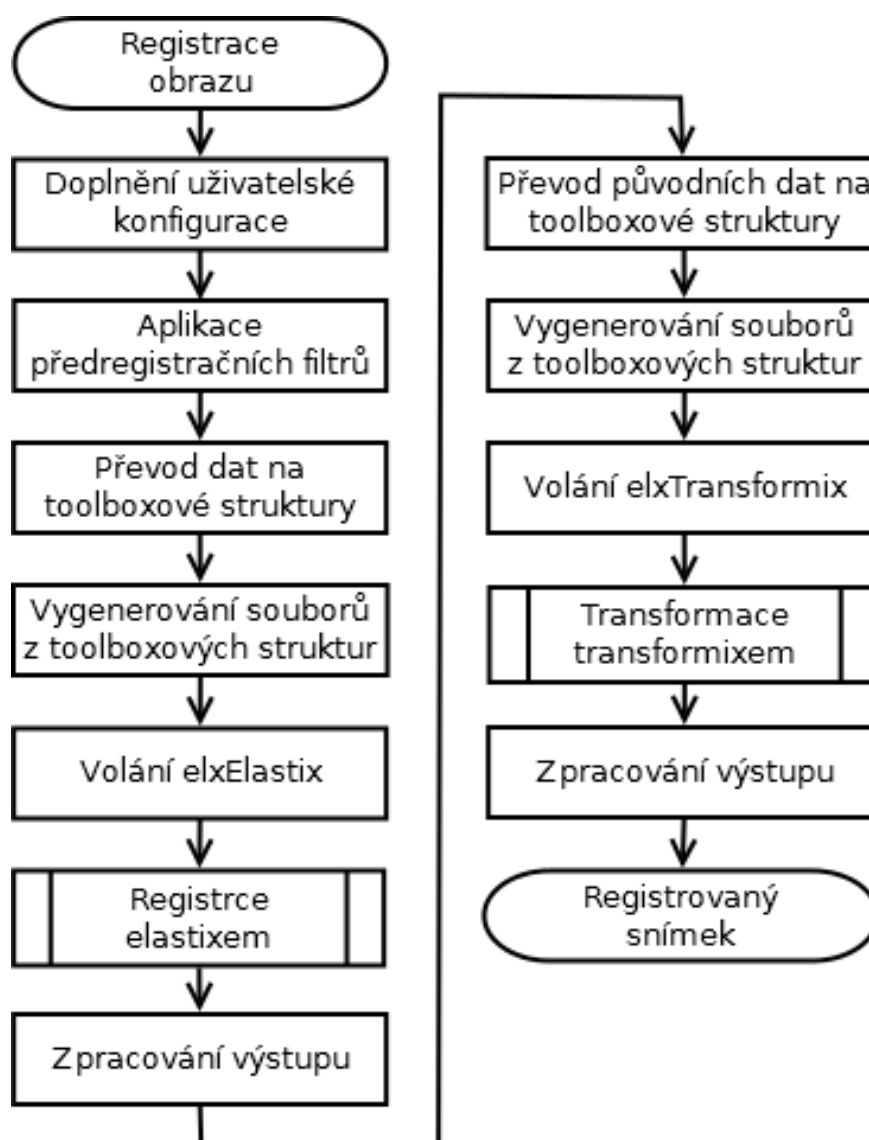
*ElastixFromMatlab toolbox* značně zjednodušuje práci s elastixem, ale pořád jde o obecnou vrstvu, jejíž použití není úplně pohodlné. Pro každou registraci je nutné vytvářet kompletní konfigurační soubor, aplikovat filtry, explicitně přetypovávat obrazy z běžných matic a pak volat elastix a transformix.

Proto byla nad toolboxem vytvořena tenká vrstva, která v parametrech přijímá přímo matice obrazů a relevantní konfiguraci. Uvnitř přijatou konfiguraci do-

plní o neměnná nastavení, aplikuje předregistrační filtry, převede datové struktury a zavoláním elastixu provede registraci. Po zjištění registračního vektoru ihned volá transformix na nefiltrovaná data, získané výsledky přetypuje na matlabové matice a ty vrátí jako svůj výstup.

### 6.5.5 Shrnutí

Shrnující schéma registračního procesu popisující posloupnost akcí při registraci jednoho obrazu je uvedeno na obr. 6.2. V následujících vývojových diagramech se pod blokem *Registrace obrazu* skrývá tento předdefinovaný proces.



Obr. 6.2: Vývojový diagram procesu registrace obrazu.

## 6.6 Rozhraní programu elastix

Program elastix podporuje stovky různých parametrů, avšak do GUI rozhraní pro registraci bylo nutné vybrat pouze ty, které má smysl v konkrétním případě měnit. Další parametry jsou k uživatelské konfiguraci automaticky doplněny na pozadí bez možnosti změny.

V následujícím seznamu jsou uvedeny zpřístupněné parametry, grafické prostředí pro jejich nastavování je na obr. 6.3. V GUI byly ponechány přímo anglické názvy metod odpovídající terminologii elastixu [28].

- Typ transformace
  - translace – `TranslationTransform`
  - eulerova – `EulerTransform`
  - podobnostní – `SimilarityTransform`
  - afinní – `AffineTransform`
  - BSpline – `BSplineTransform`
  - spline kernel – `SplineKernelTransform`
- Počet pyramidálních úrovní
  - 1
  - 2
  - 3
  - 4
- Optimalizátor
  - standardní ve směru gradientu – `StandardGradientDescent`
  - adaptivní stochastický – `AdaptiveStochasticGradientDescent`
  - úplné prohledávání – `FullSearch`
- Metrika
  - euklidovská – `AdvancedMeanSquares`
  - normalizovaný korelační koeficient – `AdvancedNormalizedCorrelation`
  - vzájemná informace – `AdvancedMattesMutualInformation`
  - normalizovaná vzájemná informace – `NormalizedMutualInformation`
- Sampler
  - náhodný – `Random`
  - náhodné souřadnice – `RandomCoordinate`
  - mřížka – `Grid`
  - úplný – `Full`
- Rozlišení histogramu
  - 8
  - 16



- 32
- 64
- 128
- Maximum iterací
  - 10
  - 50
  - 100
  - 250
  - 500
  - 1000

Typy transformace a metrik byly již vysvětleny dříve v kapitole 2.9 Transformace, počet úrovní nastavuje velikost pyramidálního rozkladu (1 je bez rozkladu), viz kapitola 2.7.4 Pyramidální registrace.

Optimalizátor a sampler nastavují postup a optimalizace výpočtu. Určují, jaký transformační vektor bude vybrán po předchozím testu a jaké pixely budou použity pro ověření kvality nové registrace.

Význam rozlišení histogramu a maximálního počtu iterací je zřejmý. Detailní popis všech parametrů je uveden v manuálu [28].

V grafickém prostředí je přednastavena konfigurace, která byla na základě doporučení v manuálu [28] a vlastních zkušeností vybrána jako vhodný kompromis mezi kvalitou a dobou trvání registrace.

| Nastavení pro elastix: |                                   |
|------------------------|-----------------------------------|
| Typ transformace       | EulerTransform                    |
| Počet úrovní           | 2                                 |
| Optimalizátor          | AdaptiveStochasticGradientDescent |
| Metrika                | AdvancedMattesMutualInformation   |
| Sampler                | Random                            |
| Rozlišení histogramu   | 32                                |
| Maximum iterací        | 500                               |

Obr. 6.3: Konfigurační volby programu *elastix* interfacované do GUI

## 6.7 Implementace jednotlivých modulů

Popis jednotlivých modulů byl rozdělen do dvou kapitol. Předchozí kapitola 5 Popis aplikace aplikaci popisuje z obecnějšího metodického pohledu (uvádí příklady zpracovaných snímků, grafy a principiální schémata), tato kapitola se zabývá implementačními detaily (uvádí programové vývojové diagramy).

### 6.7.1 Načtení snímků, výběr ROI

Registrační proces začíná načtením surových snímků z fundus kamery (přesněji řečeno jsou v takové podobě, v jaké byly poskytnuty pro tento projekt, určité zpracování již bylo provedeno). Uživatel vybírá adresář se snímky, očekává se, že zvolený adresář obsahuje pouze požadované snímky inkrementálně pojmenové tak, že seřazená sekvence odpovídá postupné časové akvizici. Další řazení snímků se neprovádí.

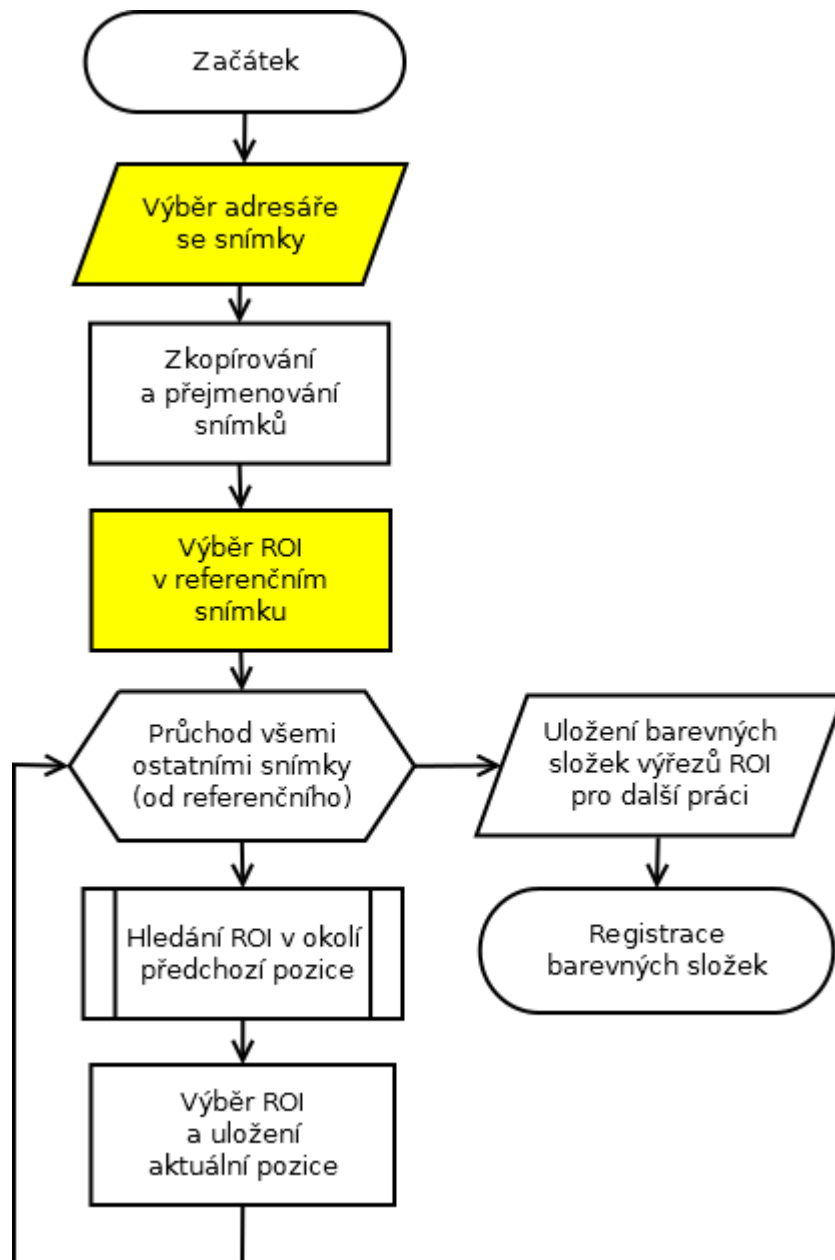
Implementačně je předregistrace zajištěna převodem snímků do šedotónové škály, výřezem referenční ROI (na základě uživatelského výběru, viz obr. 5.2) a jeho vyhledáváním ve všech ostatních snímcích. Algoritmus prochází snímky ve dvou smyčkách, přičemž si při každé iteraci zapamatuje nalezenou pozici a z ní vychází prohledávané okolí v dalším kroku – prohledává se obdélníkové okolí předchozí pozice, protože se předpokládá větší pohyb v horizontálním směru, viz graf 5.3. V prvním cyklu se postupuje od referenčního snímku zpětně k prvnímu snímku, pak se resetuje poziční vektor na referenční a prohledává se od referenčního snímku dál k poslednímu.

Po nalezení pozic ROI v šedotónové škále jsou znovunačteny originální barevné snímky, z nich jsou vyříznuty ROI, odděleny barevné komponenty a ty jsou následně samostatně uloženy pro další zpracování.

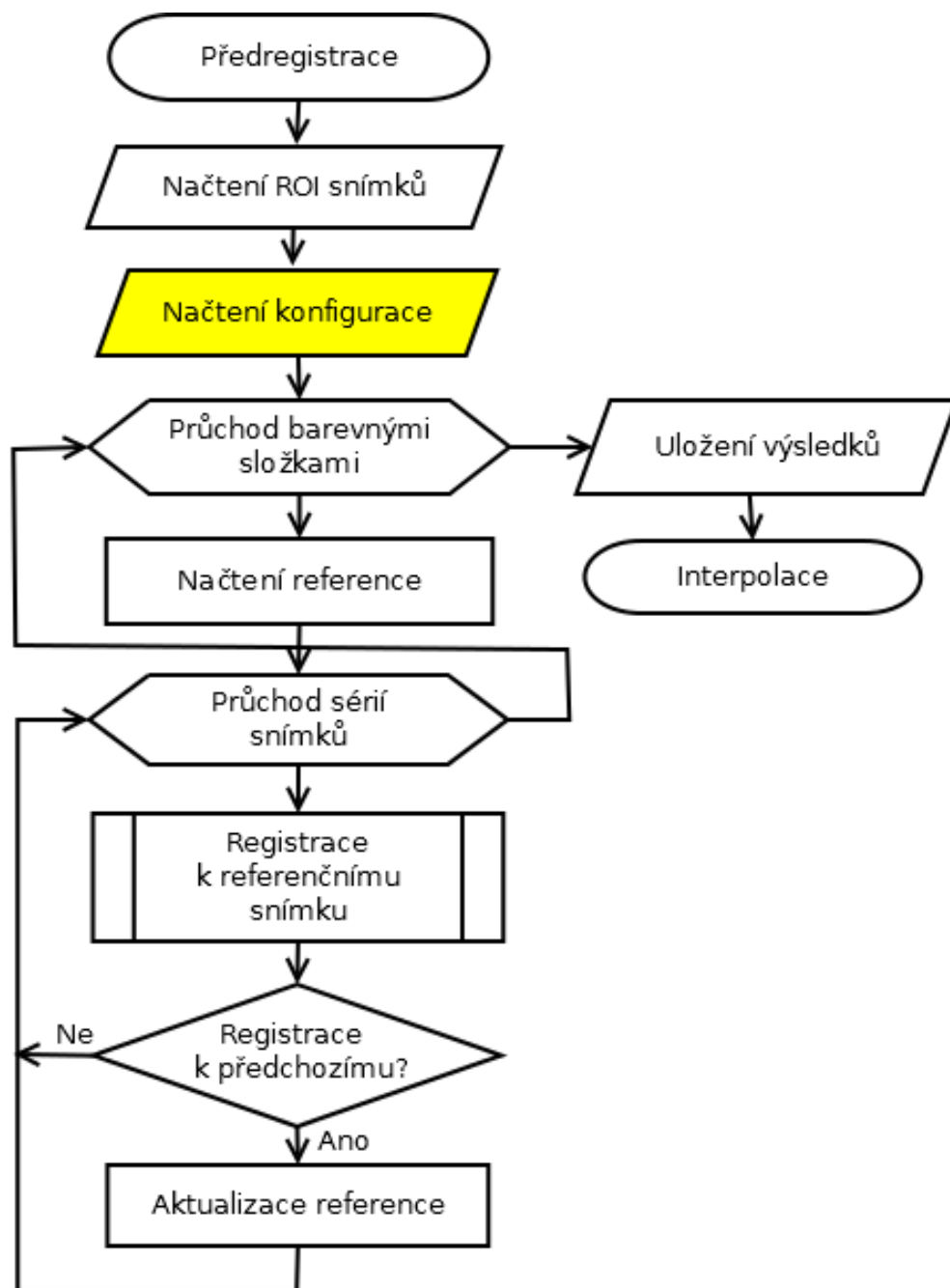
Celý proces je znázorněn na schématu 6.4.

### 6.7.2 Registrace barevných komponent

Souhrnně byl popis modulu uveden kapitole 5.4 Registrace barevných kanálů, popis implementace vychází ze schématu na obr. 6.5. Pro registraci jsou načítány snímky uložené v předchozím kroku, konfigurace se vybírá uživatelsky z přednastavených možností odpovídajících možnostem programu elastix (viz kapitola 6.6 Rozhraní programu elastix). Postupuje se jednotlivě pro všechny barevné komponenty, prvním krokem v každé iteraci (pro každou barevnou složku) je načtení referenčního snímku. V závislosti na nastaveném režimu se jako reference bere buď první snímek, nebo uživatelem zadaný snímek. Po vytvoření referenčního snímku se spustí registrace v zanořeném cyklu. Pokud je nastaveno registrování k předchozímu snímku, je na konci každé iterace aktualizována reference na aktuální snímek.



Obr. 6.4: Vývojový diagram načtení snímků, výběru ROI, předregistrace a uložení výřezů. Žlutě jsou označeny body prováděné interaktivně uživatelem.



Obr. 6.5: Vývojový diagram registrace barevných komponent. Žlutě jsou označeny body prováděné interaktivně uživatelem.

### 6.7.3 Interpolace

Samotná časová interpolace je v Matlabu naimplementována přímočaře pomocí interní funkce `interp1`, která v základní podobě přijímá 3 parametry. Nejdůležitějším z nich je vektor vstupních dat, který v tomto případě představuje soubor totožných bodů napříč  $k$  snímky. Pro interpolaci více takových vektorů funkce `interp1` podporuje práci s 2D maticí, která je naplněna  $k$  snímky o rozměru  $m \times n$  skenovanými po řádcích ( $1 \times mn$ ), tj. interpolovaná matice má rozměr  $k \times mn$ .

V GUI v části konfigurace interpolace uživatel zadává časový odstup vzorkování od předchozího snímku a požadované výsledné vzorkování, na jejichž základě se vypočítá časová osa původního vzorkování i požadovaného vzorkování. Viz obr. 5.8. Počet nových snímků označme  $K$ . Tyto časové osy tvoří první a třetí parametr funkce `interp1`.

Po přípravě dat je pomocí funkce `interp1` vypočítána výsledná datová sada o rozměru  $K \times mn$ . Z této sady se poté ve smyčce odebírají jednotlivé vektory  $1 \times mn$  a transformují se do výsledných obrazů (celkem  $K$  snímků o rozměrech  $m \times n$ ). Celý proces se provádí 3x pro jednotlivé barevné složky samostatně (teoreticky by bylo možné provést vše v jednom kroku, ale museli bychom pracovat s maticí o rozměru  $k \times 3mn$ , resp.  $K \times 3mn$ , což by znamenalo 3x větší nároky na paměť<sup>3</sup>).

Pro interpolaci je využívána metoda kubických splinů, která by (za cenu nejnáročnějšího výpočtu) měla poskytovat nejlépe spojitě výsledky. Pro svoji práci potřebuje minimálně 4 body, což v tomto projektu nepředstavuje problém. V principu lze však použít libovolnou Matlabem nabízenou metodu, v současnosti jsou k dispozici volby 'nearest', 'next', 'previous', 'linear', 'spline', 'pchip', nebo 'cubic' [19].

Výstupem funkce `interp1` je nová matice obsahující interpolované snímky, jejichž hodnoty je ještě třeba omezit do intervalu  $< 0; 1 >$ , protože hodnoty získané interpolací kubickými spliny mohly tento interval přesáhnout. Ukazuje se, že stačí provést saturaci, tj. pixely přesahující meze nastavit na hodnoty mezí.

Schéma interpolace a souvisejících operací je uvedeno na obr. 6.6.

#### Dvouprůchodová interpolace

Takto navrženou interpolaci považuji z hlediska kvality za nejlepší možnou, avšak problémem je její značná pomalost. Pokud zvolíme vyšší snímkovou frekvenci budoucí sekvence, nainterpoluje se velký počet barevných složek, které se musí v dalším kroku po jedné registrovat do barevných snímků, což je velmi pomalé. S cílem zrychlit běh programu byla naimplementována dvouprůchodová interpolace.

<sup>3</sup>Takto naimplementovaná interpolace má největší nároky na operační paměť z celého programu. Lineární paměťovou složitost vzhledem k počtu snímků a kvadratickou vzhledem k velikosti snímků.

Pokud je tato možnost zvolena, tak jsou v první fázi nainterpolovány snímky pouze do časových bodů již existujících snímků, tzn. že pokud je na vstupu  $3 \times 42$  barevných složek ve 126 různých časových bodech, bude na výstupu  $3 \times 126$  obrazů. Po této interpolaci standardně proběhne registrace jednotlivých barevných kanálů a až po ní se spustí druhá fáze interpolace, ve které jsou dle nastavených parametrů dointerpolovány všechny snímky.

Pak můžeme vyjádřit celkový čas jednotlivých registrací v celém procesu jako

$$t = 3 \cdot T_1 \cdot (n - 1) + 2 \cdot 3 \cdot T_2 \cdot n,$$

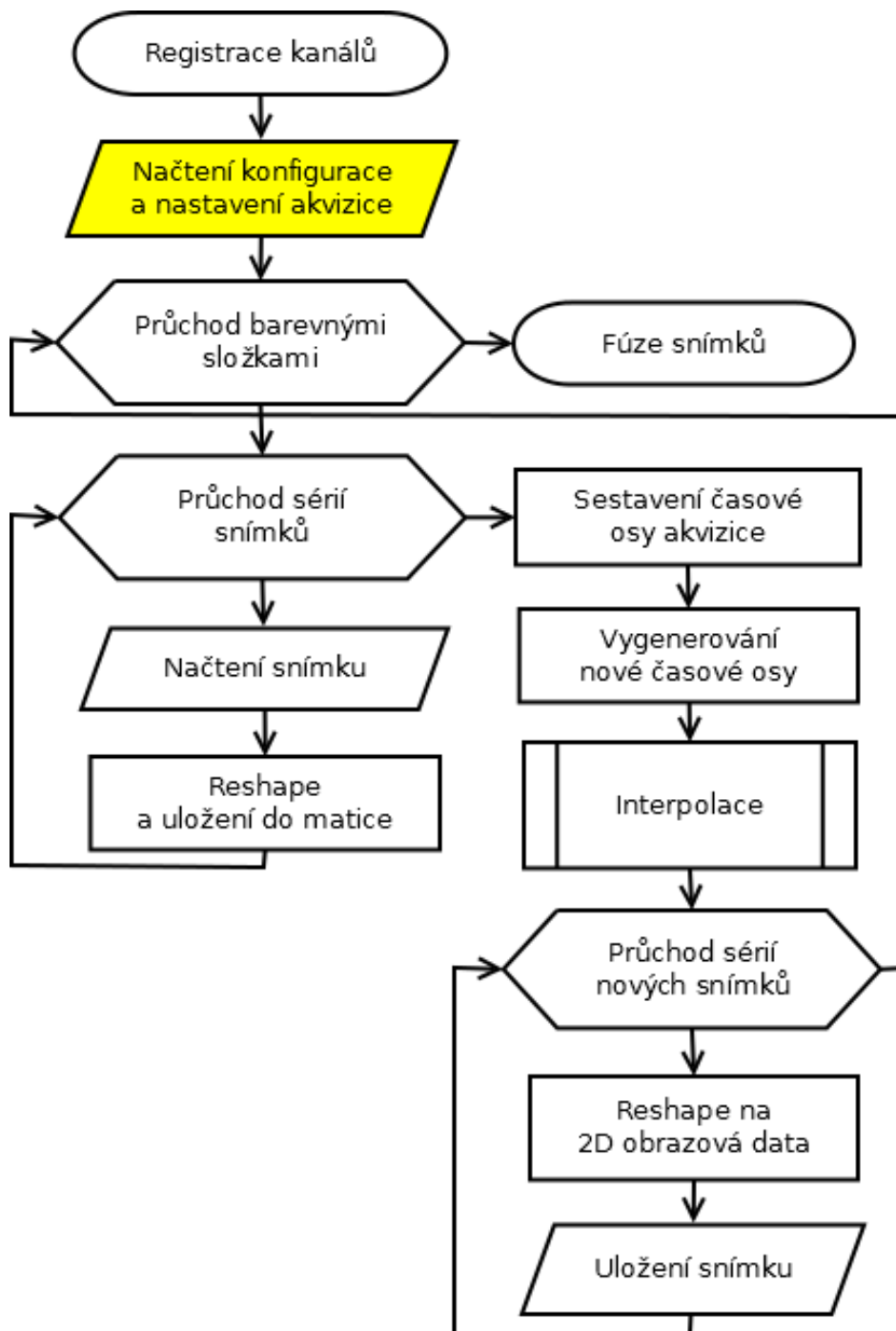
kde  $n$  je počet vstupních barevných snímků a  $T_1$ , resp.  $T_2$  jsou konstanty určující dobu jedné monomodální, resp. multimodální registrace. Čas potřebný pro registraci během fúze komponent se v závislosti na parametrech zkrátí na cca 40 %, avšak zvýší nepřesnost interpolace.

#### 6.7.4 Kompletování barevných snímků

Po předchozí interpolaci jsou již připraveny všechny snímky v dosud oddělených barevných složkách v kadenci odpovídající výsledné videosekvenci. Protože předchozí registrace byla prováděna z časového hlediska, je třeba před sloučením barevných složek provést registraci v rámci každého snímku. Návrhově byl tento krok popsán v kapitole 5.6 Registrace snímků.

Implementačně je tento krok zajištěn procházením sekvence snímků. V každé iteraci jsou načteny všechny tři složky k aktuálnímu snímku a dle uživatelského nastavení registrace jsou červená a modrá složka postupně registrovány na zelenou. V zregistrovaných složkách se vypočítají sumy jasů, které se uloží do paměti pro následnou normalizaci, samotné registrované složky obrazu jsou uloženy na disk a pokračuje se další iterací.

Po proběhnutí prvního cyklu je k dispozici (kromě samotných zregistrovaných snímků) matice, která reprezentuje sumy jasů v celé videosekvenci. Z této matice se, pokud uživatel zvolil možnost normalizace, pro každý barevný kanál určí mediánová hodnota, která představuje normalizovanou úroveň jasu. Na základě uživatelských požadavků je možné vypočítanou cílovou úroveň jasů dále transformovat, tj. snížit či zvýšit jas jednotlivých složek nebo celého barevného snímku. V následujícím cyklu se již naposled prochází celá sekvence snímků a jas se transformuje na vypočítanou úroveň. Vzniklé snímky, reprezentující barevné složky, se nyní sloučí v třírozměrné  $m \times n \times 3$  matice reprezentující barevný obraz v RGB složkách. Posledním krokem je uložení výsledného barevného retinálního snímku.



Obr. 6.6: Vývojový diagram časové interpolace snímků

Schéma registrace barevných komponent do jednoho snímku a normalizace jasu je uvedeno na obr. 6.7. Tečkovaná spojnice mezi registrací a jasovou normalizací snímků naznačuje rozdělení těchto bodů do více uživatelský kroků.

### 6.7.5 Vytvoření videa

Posledním (volitelným) krokem programu je vytvoření videosekvence z registrovaných a interpolovaných PNG snímků.

K vytvoření videa je využita open source konzolová aplikace FFmpeg [17]. Snímky jsou již v dřívějším kroku ukládány s inkrementálním číselným kódem, takže stačí zavolat FFmpeg, který dle předaných parametrů vytvoří video soubor.

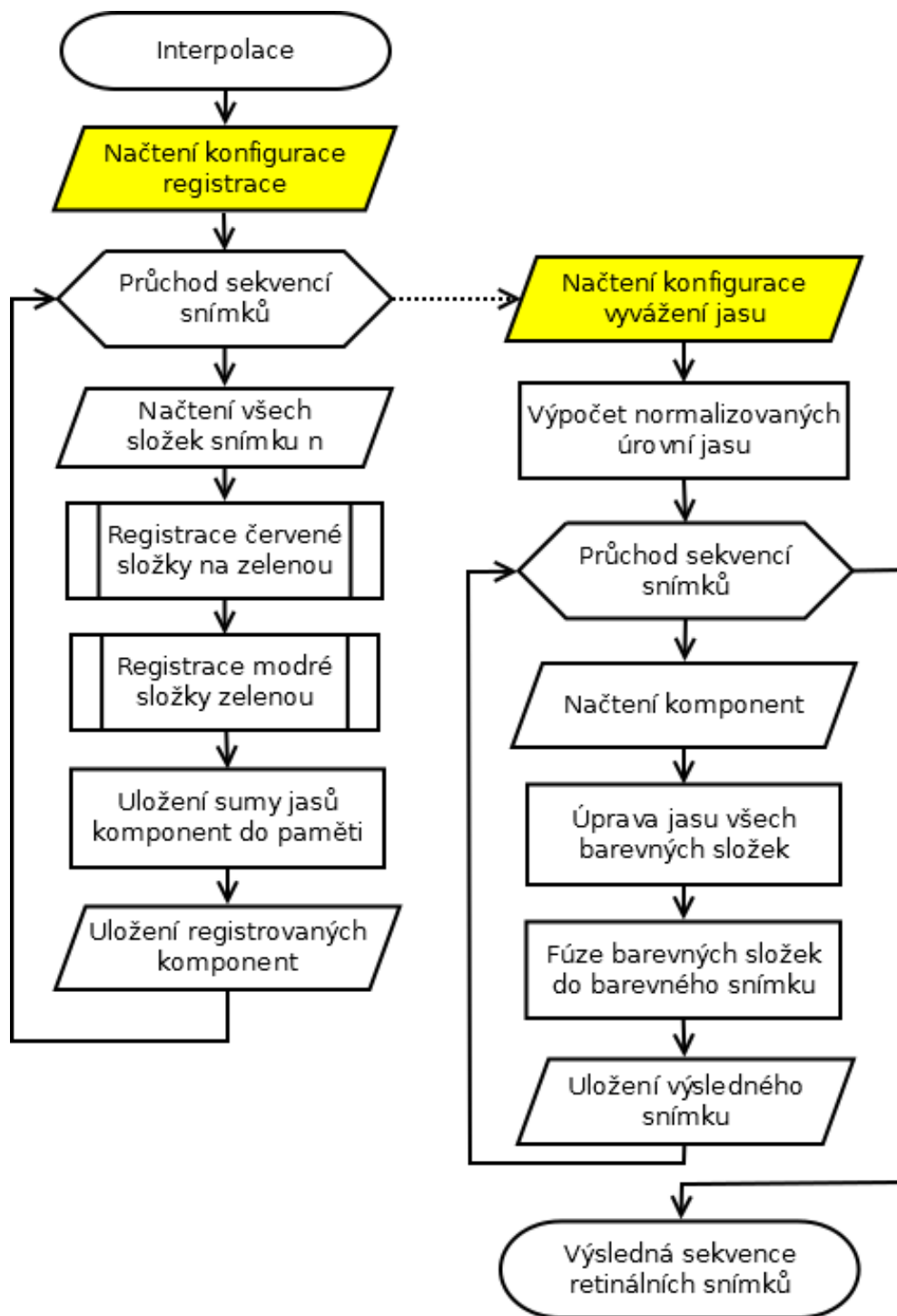
Videosoubory používají pokročilé více či méně ztrátové komprimační metody. Jejich detailní rozbor by přesahoval tematické zaměření této práce, proto se omezím na konstatování, že je využito přednastaveného profilu MPEG-1 kodeku s preferencí kvality (datová velikost je druhořadá).

Důležitým parametrem je především `-framerate` určující vzorkovací frekvenci videa, který je nastaven v souladu s nastavením interpolace, aby výsledné video odpovídalo požadavkům. Druhým důležitým parametrem je `-h:v`, který nastavuje kvalitu videa.

Dále je uveden příklad volaného příkazu pro vytvoření videa z png snímků se vzorkovací frekvencí 10 fps (na tu byly snímky interpolovány), `%03d.png` je maska pro načtení inkrementálně pojmenovaných souborů (tj. `001.png`, `002.png`, ...):

```
ffmpeg.exe -framerate 10 -i cesta/finalSequence/%03d.png  
-b:v 15M cesta/video/video.avi.
```





Obr. 6.7: Vývojový diagram fúze jednotlivých barevných komponent do barevných snímků s úpravou jasu. Žlutě jsou označeny body prováděné interaktivně uživatelem a tečkovaná spojnice naznačuje rozdělení do více uživatelských kroků.

## 6.8 Rozšiřitelnost programu

Během testování různých algoritmů bylo experimentováno zvláště s jednoduššími vlastními metodami, interními registračními funkcemi v Matlabu, některými knihovnamy pro Python a s programem elastix.

Hlavním cílem při testování alternativních řešení bylo porovnání možností, které nabízejí, s programem elastix. Pokud by se podařilo nalézt nějakou zásadní funkčnost, kterou elastix nenabízí, mohla být v podobě další uživatelské volby implementována do vytvářené aplikace. Ukázalo se, že elastix je pro registraci obrazů dostatečně komplexní nástroj a zvládá vše, co ostatní testované programy, proto se zde detailním popisem alternativních nástrojů nezabývám (uvažuji skupinu nástrojů postavených na podobnosti obrazů, nástroje identifikující význačné body patří do odlišné kategorie).

U algoritmů založených na podobnosti obrazů je pro úspěšnost registrace zásadní kritériální funkce hodnotící kvalitu registrace. Související optimalizační metody nebo paralelizace výpočtů mají vliv na dobu trvání registrace a již méně ovlivňují kvalitu samotnou, která je závislá spíše na předregistrační filtraci obrazu.

Z toho důvodů také vytvořená aplikace nabízí kromě nastavení existujících filtrů možnost začlenění vlastních předregistračních filtrů.

Komentovaný příklad uživatelského filtru používající CLAHE filtraci je uveden v příloze C. Zde pouze zdůrazníme, že uživatel má při tvorbě vlastního filtru k dispozici veškeré funkce poskytované Matlabem, tedy i možnost zavolat libovolnou externí aplikaci, podobně jako se používá elastix.

Na základě studia literatury lze říci, že pravděpodobným směrem, kterým by směřovaly kroky dalšího vývoje, by bylo vytvoření filtrů, které budou ve vstupním obraze segmentovat cévy. Takto vzniklé binární obrazy by byly použity pro následnou registraci pomocí programu elastix.

Vzhledem k tomu, že jsou literatuře často uváděny registrační metody založené na detekci význačných bodů, bych rád zdůraznil modulový návrh celé aplikace. Velkou část aplikace (výběr ROI, předregistrační filtraci, práci se soubory, interpolaci, konfiguraci) lze použít i pro metody založené na význačných bodech.

Největší překážkou pro nahrazení elastix modulu libovolným jiným programem je provázanost nastavování konfigurace s uživatelským prostředím. Samotná registrace je ale realizována jen několika málo funkcemi a použití nového modulu je tak pouze otázkou napsání nového wrapperu (s konfigurací pro začátek zapsanou přímo do zdrojových kódů), což je otázka půl dne.

Lze tedy říci, že vytvořená aplikace je rozšiřitelná téměř kterýmkoliv směrem.

## 7 VÝSLEDKY REGISTRACE

Úspěšnost registrace byla průběžně testována již při vývoji aplikace a jednotlivé kroky byly modifikovány tak, aby celý proces vedl k vytyčenému cíli. Vytvořená aplikace se specializuje na registraci snímků z uvedené kamery a maximálně usnadňuje celý registrační proces. Uživatelsky zadávané parametry jsou výsledkem kompromisního rozhodnutí mezi maximální automatizací a možností ovlivnit registraci. I když byla v některých krocích pro uživatele ponechána značná svoboda nastavení, výchozí konfigurace obvykle vede k dobrým výsledkům.

### 7.1 Výchozí nastavení pro elastix

Na základě testování různých metod byla do defaultního nastavení elastixu vybrána rigidní (Eulerova) registrace, metrika se vzájemnou informací (MI), náhodný sampler<sup>1</sup> a maximálně 250 iterací.

### 7.2 Výběr typu transformace

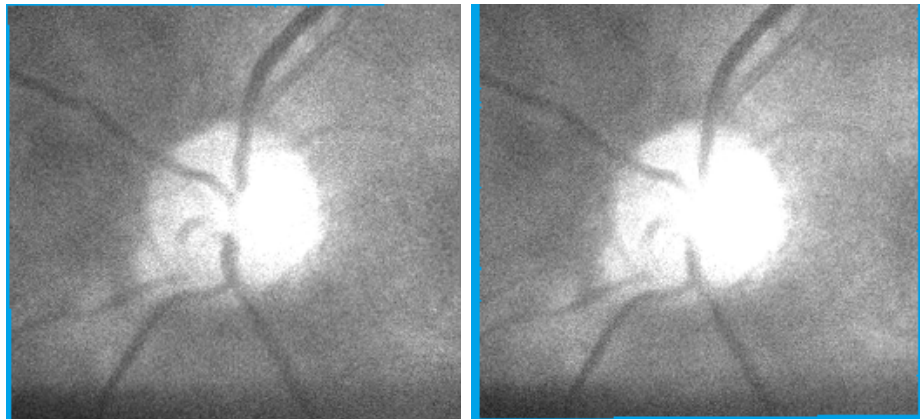
Jedním ze zásadních nastavení, která lze měnit, je typ transformace. Výchozí je rigidní (Eulerova), je možné použít translační, podobnostní, afinní i flexibilní, ale obecně pokročilejší transformace nevedou k lepším výsledkům. Snímky by z principu neměly být flexibilně deformovány, protože se anatomie oka během minutového snímání významně nemění. Maximálně lze uvažovat o změně projekce vlivem pohybu oka nebo kamery, ale ta by měla být tak malá, že bude k její kompenzaci dostačovat afinní transformace (avšak i ta může nežádoucím způsobem deformovat obraz).

Příklad značně sporné afinní registrace je uveden na obr. 7.1, kde bylo pro lepší přehlednost pozadí po registraci obarveno modře. Jde o sousední snímky, ten pravý byl ale oproti levému při registraci značně zúžen, v původním rozlišení o cca 8 px, toto zúžení ale nemá reálné opodstatnění a následující snímek v sérii již zúžen není. Mozaikové zobrazení těchto snímků je uvedeno na obr. 7.2 vlevo a pro srovnání je vpravo zobrazeno mozaikové srovnání týchž snímků při rigidní registraci. Nepřesnosti v registraci jsou otázkou pixelových posunů, ale na základě analýzy snímků v plném rozlišení lze říci, že rigidní transformace nejen že nedeformuje roztažení cév, ale celkově obrazy zregistrovala lépe.

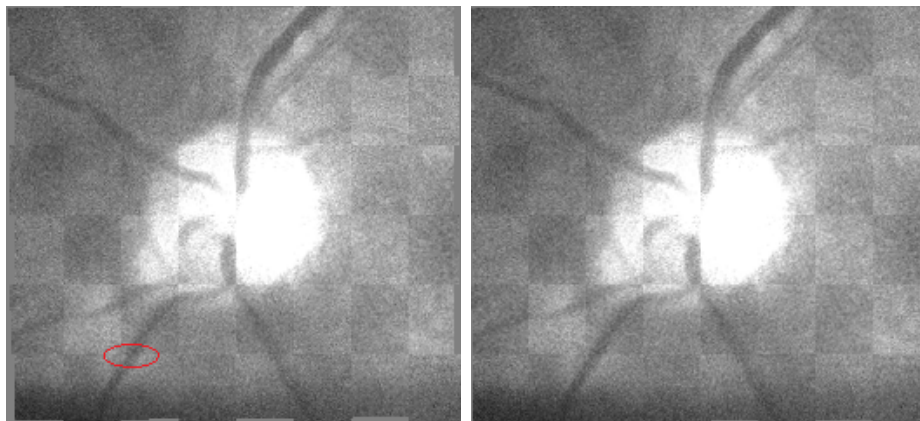
Vzhledem k tomu, že odhalení takových případů vyžaduje detailní analýzu a jejich neodhalení může zkreslit výsledky, doporučuji používat rigidní registraci a jiné metody použít pouze pokud k tomu je důvod.

---

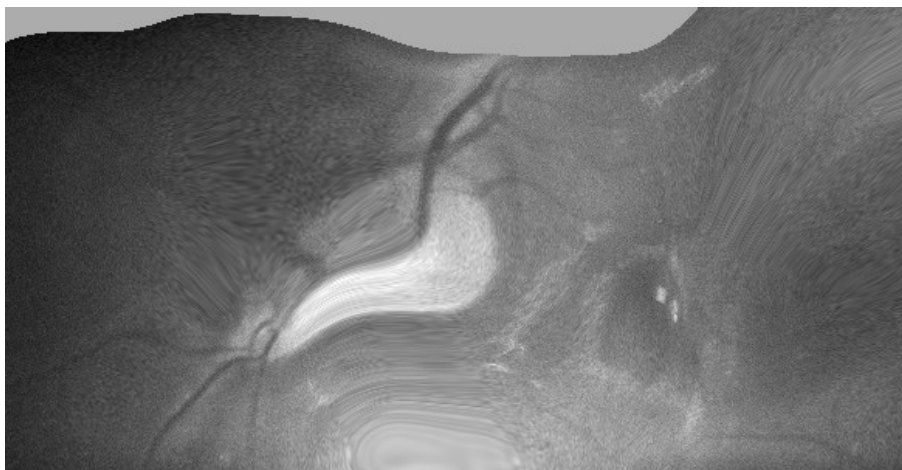
<sup>1</sup>Určuje způsob výběru bodů pro výpočet metriky.



Obr. 7.1: Příklad nesprávné afinní registrace. Snímky jsou příliš zúžené, v původním rozlišení o cca 8 px; tj. značné porušení proporcí v obraze zkreslující pulzace cév, které se mají zachovat.



Obr. 7.2: Mozaika stejných snímků registrovaných afinní (vlevo) a rigidní (vpravo) registrací. Červenou elipsou je označena jedna z mála zde pozorovatelných nepřesností, avšak na základě analýzy snímků v plném rozlišení lze říci, že zatímco po afinní transformaci je registrace v některých místech nepřesná, po rigidní registraci je zcela přesná.



*Obr. 7.3: Zcela chybná flexibilní registrace*

Přestože byla ponechána možnost využít i pokročilé flexibilní transformace, neměla by nastat situace, kdy bude mít jejich použití opodstatnění. Zkušenosti ukazují, že použití flexibilních transformací vede k horším výsledkům, protože se elastix snaží kompenzovat akviziční nedostatky deformací obrazu. Tj. např. deformovat snímky tak, aby si vzájemně odpovídaly odlesky od osvětlení, ačkoliv reálně je ten odlesk na každém snímku v jiné části a je zcela nesmyslné snažit se je vzájemně lícovat (avšak tuto apriorní informaci není možné elastixu předat<sup>2</sup>).

Na obr. 7.3 je uveden příklad naprostého selhání registrace. Nejde sice o typický výsledek flexibilní registrace, je to jeden z prvních pokusů před aplikací dalších filtrů, ale demonstruje, k čemu může vést chybně nastavená flexibilní registrace bez použití vhodných filtrů.

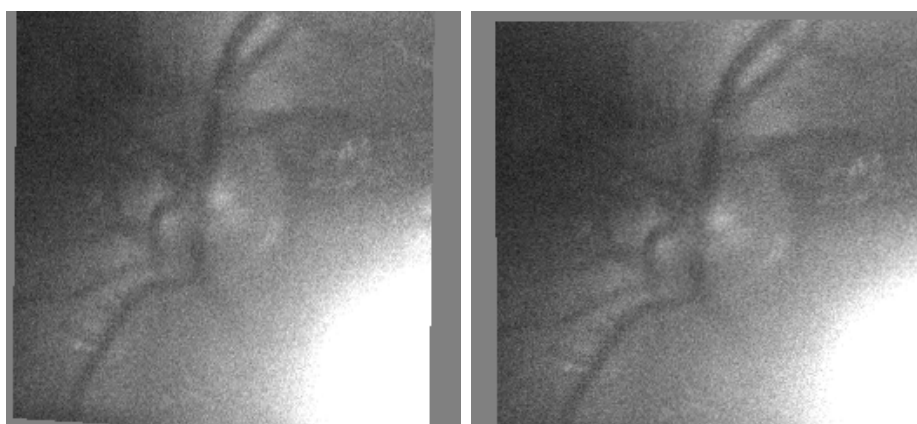
## 7.3 Vliv odlesků

V předchozích kapitolách byla často zdůrazňována potřeba předzpracování snímků digitální filtrací, jejíž konkrétní podoba je uvedena v 4.5 Potlačení odlesků diod horní propustí. Typickým příkladem výsledku rigidní registrace bez aplikace patřičných filtrů je obr. 7.4 vlevo, vpravo je potom tentýž snímek registrovaný po předchozí filtraci. Rozdíl v jednotlivých transformacích je zřejmý, bez zobrazení mozaiky uvádím, že zatímco pravý filtrovaný snímek zcela přesně lícuje s referencí, levý nefiltrovaný je chybně posunut až od 20 px.

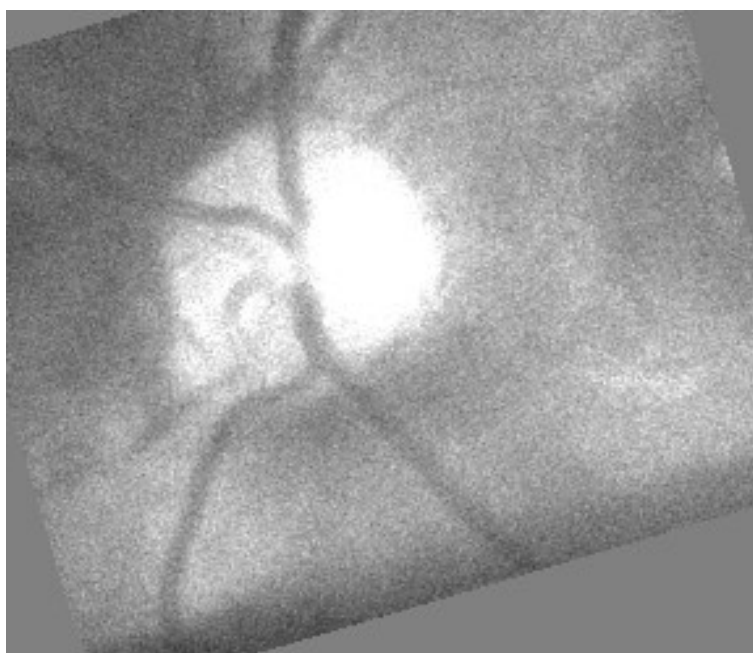
Na obrázku 7.5 je uveden další příklad selhání registrace bez předchozí filtrace. Tentokrát zde nejsou na první pohled patrné žádné odlesky osvětlovacích diod,

---

<sup>2</sup>Bylo testováno adaptivní maskování, které situaci zlepšilo, ale i tak se použití rigidní transformace zdá být výhodnější.



*Obr. 7.4: Registrace snímku s odleskem diody před a po filtraci. Levý snímek byl registrován bez předchozí filtrace a je chybně posunut o cca 20 px (referenční snímek měl slepou skvrnu přesně ve středu). Pravý snímek byl před registrací filtrován pásmovou propustí a vzhledem k referenčnímu snímku (jde o sekvenci uvedenou na obr. 7.2) je registrován zcela přesně. Příklady mozaikového srovnání jsou uvedeny později.*



*Obr. 7.5: Příklad chybné registrace bez předchozí filtrace. Opět jde o sekvenci jako na obr. 7.2, tj. rotace měla být maximálně cca 1°. Chyba je způsobena nevyrovnaným jasem mezi referenčním a registrovaným snímek, jak je ukázáno na obr. 4.6 a detailně popsáno v dané kapitole. Tyto snímky jsou důkazem, že použití předregistrační filtrace je zcela nezbytné.*

přesto, jak bylo uvedeno v jasových profilech v kapitole 4.5 Potlačení odlesků diod horní propustí, zde tento vliv je a pokud není použita filtrace, selže i nejjednodušší rigidní registrace. Žádná z nabízených metrik se s tímto poškozením obrazu sama o sobě nevyrovná.

## 7.4 Výsledné snímky

Subjektivně hodnotím registraci snímků jako dostatečně kvalitní. Podařilo se vytvořit nástroj, který je schopný správně registrovat celé sekvence s takovou kvalitou, že až na výjimky nelze v sousedních snímcích v mozaikovém srovnání (100% přiblížení) najít žádná nepřesně zregistrovaná místa.

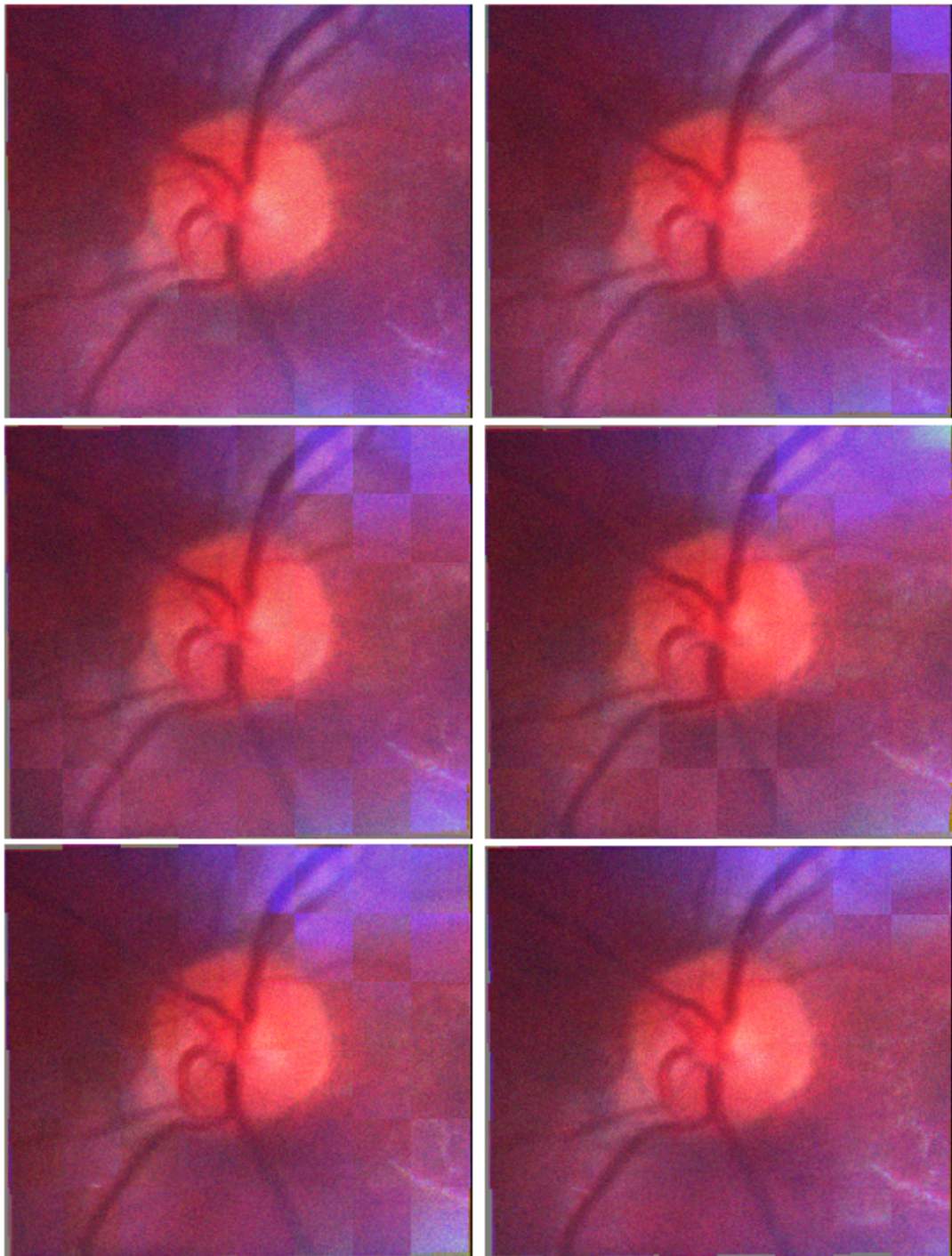
Zregistrované snímky i vytvořené video jsou pro detailnější analýzu k dispozici v elektronických přílohách. V této verzi (pdf nebo papírový výtisk) nelze sekvence o desítkách snímků patřičně prezentovat, proto alespoň pro základní představu na obr. 7.6 uvádím mozaikové srovnání 7 sousedních snímků z jedné ze sekvencí. Hranice mazaiky jsou viditelné jen díky tomu, že byla vybrána část sekvence s většími odlesky. Napojení cév na všech hranicích je při běžném vizuálním pozorování zcela přesné.

## 7.5 Hodnocení dosažené chybovosti a přesnosti

Na základě průběžných zkušeností, kdy byly některé snímky zregistrovány zcela chybně, jsem ručně procházel 4 různé sekvence a kontroloval všechny snímky ve všech třech barevných kanálech. Při takovém hodnocení nelze s pixelovou přesností zhodnotit přesnost registrace, ale viditelně chybě zregistrované snímky jsem našel pouze dva z přibližně 600 kontrolovaných snímků. Oba chybně zregistrované snímky byly z modrého kanálu a byly téměř ze třetiny přesvětlené diodami, zde program prakticky neměl šanci uspět. Do tohoto hodnocení nepočítám dalších cca 5 nesmazaných zcela přesvětlených snímků z různých barevných kanálů, které elastix nechal v původní pozici bez posunu. Odhadovaná chybovost do 0.5 % je vzhledem ke kvalitě vstupních snímků vynikající (na začátku jsem se obával, že bude horší).

U stejných zregistrovaných sekvencí byly vizuálně kontrolovány i videosoubory. I minutové sekvence, ve kterých výchylka v originálních datech dosahovala až 150 px, jsou velmi přesně fixovány na referenční snímek. Pozorovatelné jsou pouze pomalé pohyby s výchylkou dosahující maximálně jednotek pixelů. Různé příklady zregistrovaných sekvencí i videosouborů jsou k dispozici v elektronických přílohách.





*Obr. 7.6: Mozaikové srovnání 7 sousedních snímků po registraci. A1 je srovnáním prvního s druhým, A2 druhého se třetím atd. Pro srovnání byla záměrně vybrána část sekvence s výrazněji odlišným jasem. V části bez odelsků nejsou hranice mozaiky vůbec viditelné.*



## 7.6 Kvantitativní hodnocení registrace

Objektivní zhodnocení dosažené registrace je při neznalosti správné transformace vždy značný problém. V této práci je použito hodnocení shodnosti jasových profilů kolmých na cévy v registrované sekvenci. Podobný přístup byl pro snímky z téže fundus kamery použit již např. R. Kolářem [42].

Podstatou metody je statistické zpracování jasových profilů sejmutých z téže pozice snímku v průběhu celé sekvence. Předpokladem je, že u správně registrovaných snímků by se středy cév neměly pohybovat. Sejmuté profily jsou však v původní podobě kvůli šumu a nerovnoměrné střední úrovni jasu obtížně hodnotitelné. Proto byly jednotlivé vektory filtrovány gaussovým filtrem, interpolovány (kvůli přesnému určení pozice minima) a nakonec od nich byla odečtena minimální hodnota jasu. Provedené úpravy neovlivňují vertikální pozici extrémů, pouze horizontálně normalizují křivky tak, že minimální hodnota je vždy nulová.

Pro číselné vyhodnocení byla detekována vertikální pozice extrému (minima) v průběhu celé sekvence a z tohoto souboru hodnot byla spočítána střední hodnota (7.1) a směrodatná odchylka (7.2).

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N x_i \quad (7.1)$$

$$\sigma(X) = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{X})^2} \quad (7.2)$$

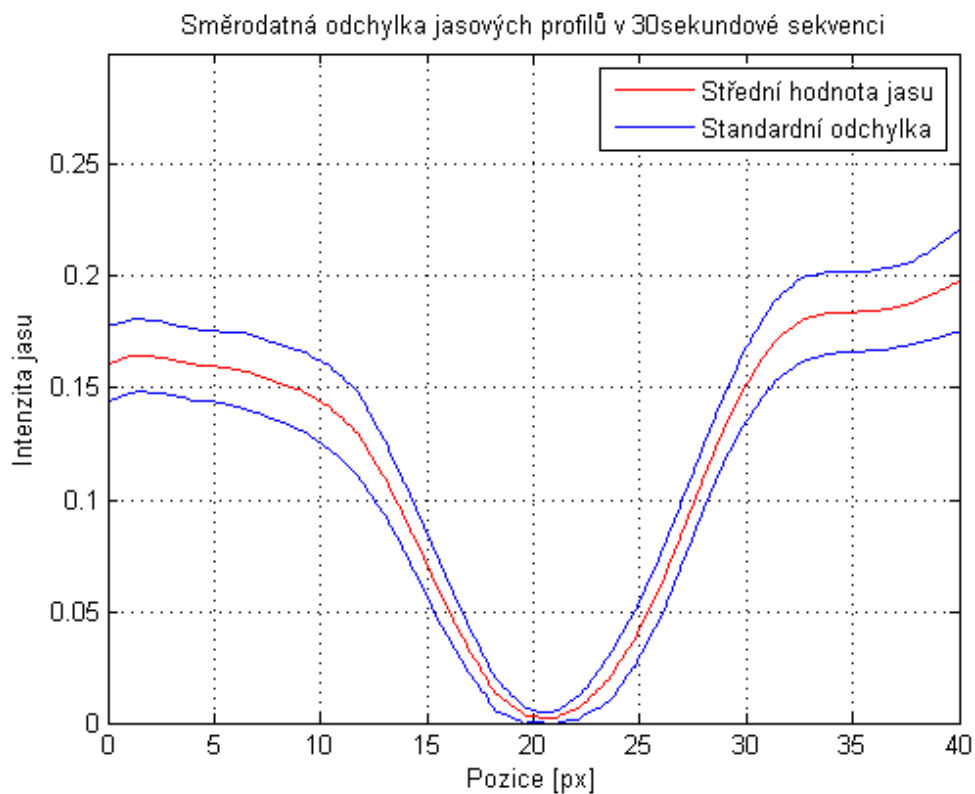
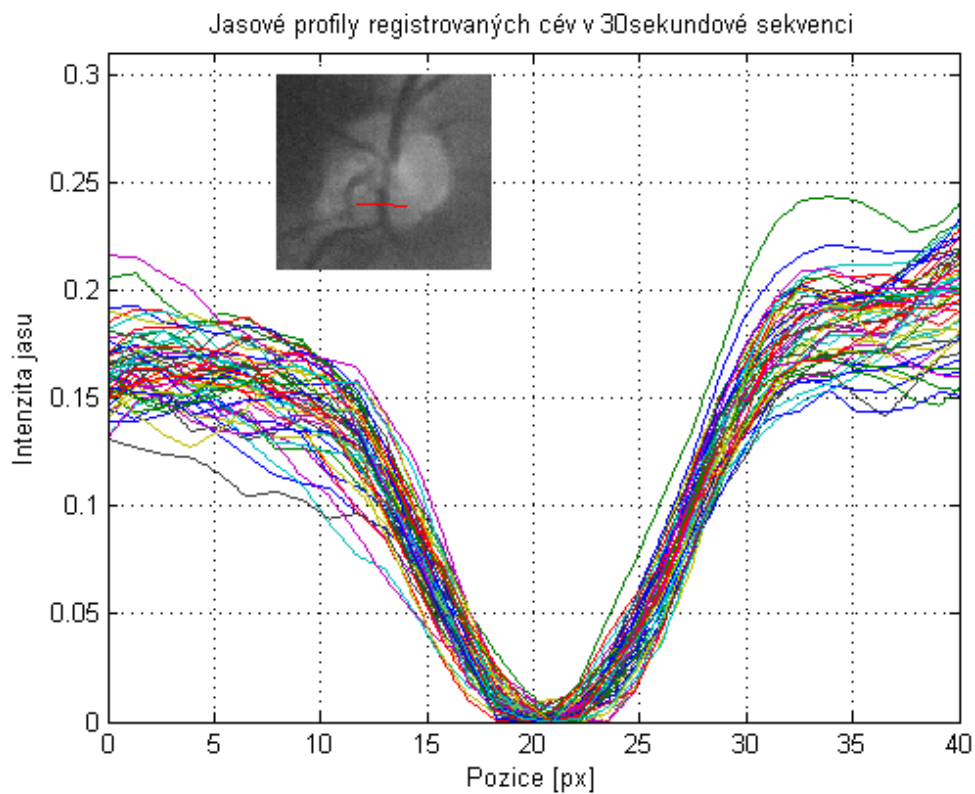
V hodnocené 30sekundové sekvenci dosahuje směrodatná odchylka detekované pozice minima v průměru 0.3 px, v další kapitole bude uveden detailnější přehled.

Střední hodnota i směrodatná odchylka jsou vypočítány pro celou datovou sadu a vyneseny do grafu 7.7, kde křivky směrodatných odchylek  $\sigma$  ohraničují pásmo  $\langle \bar{X} - \sigma, \bar{X} + \sigma \rangle$ . Z grafu je patrné, že standardní odchylka v místě hranic cév dosahuje hodnoty nižší než 1.5 px.

## 7.7 Kvantitativní hodnocení různých registračních metod

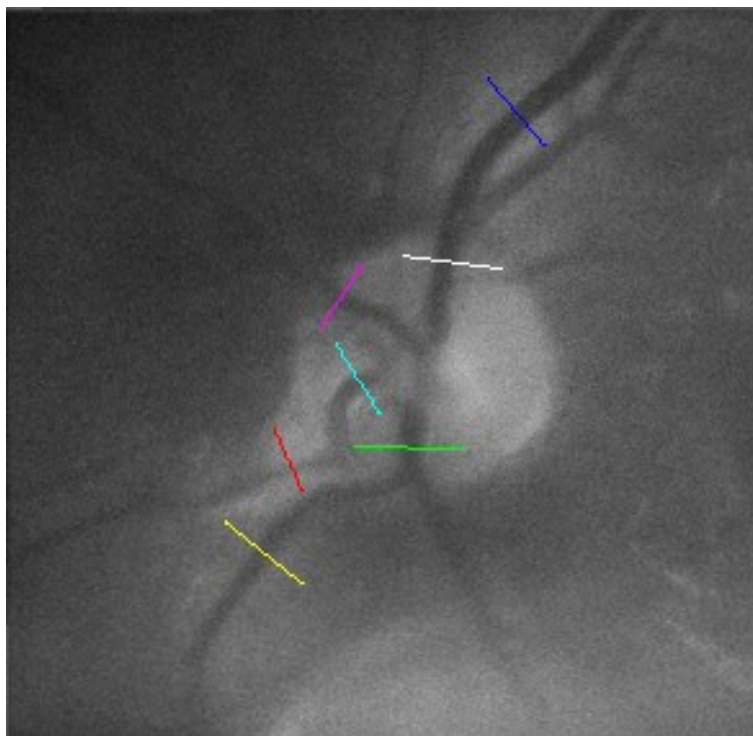
Metoda statistického hodnocení pohybu středů cév v jasových profilech uvedená v předchozí kapitole byla rozšířena pro komplexnější hodnocení přesnosti registrace a srovnání různých registračních metod.

Základem srovnávací metody je porovnání standardních odchylek vyhodnocených ze sekvencí zaregistrovaných s různou konfigurací. Pro srovnání byla vybrána cca 30sekundová sekvence, umístění vyhodnocovaných jasových profilů je na obr.



Obr. 7.7: Hodnocení kvality registrace v jasových profilech. Registrováno s defaultním nastavením: rigidní registrace, vzájemná informace, maximálně 250 iterací. Hodnoty px uvedené v ose x je třeba chápat jen jako uvedení měřítka.

7.8. Tato sekvence (se stejným výběrem ROI) byla opakovaně zregistrována s různým nastavením (viz tabulka 7.1 a 7.2). Výsledkem je soubor původně shodných, ale odlišně registrovaných obrazových dat. V každé jednotlivé sekvenci byly vyhodnoceny shodně umístěné jasové profily a vypočítané charakteristiky lze použít pro hodnocení jednotlivých registračních metod.



*Obr. 7.8: Vyznačení několika jasových profilů, ve kterých je hodnocena kvalita registrace. Snahou bylo jednotlivé profily orientovat do všech směrů a co nejvíce se přiblížit k okrajům snímku. Na druhou stranu, pro správné zpracování je třeba umístit jednotlivé profily kolmo na cévy a do dostatečně kontrastních míst.*

Následující tabulky uvádí přehled přesnosti různých registračních metod.

| Transformace             | Profil 1       | Profil 2       | Profil 3       | Profil 4       | Profil 5       | Průměr         |
|--------------------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Translation,<br>MI, i250 | 0.27650        | 0.46170        | 0.33909        | 0.39486        | 0.22882        | 0.34020        |
|                          | 0.32 0.34 0.54 | 0.44 0.49 2.50 | 0.41 0.40 0.49 | 0.50 0.63 0.76 | 0.27 0.28 0.46 | 0.39 0.43 0.95 |
| Euler,<br>MI, i100       | 0.28598        | 0.30664        | 0.28181        | 0.34696        | 0.17732        | 0.27974        |
|                          | 0.35 0.36 0.57 | 0.42 0.30 2.39 | 0.37 0.35 0.48 | 0.38 0.60 0.72 | 0.23 0.27 0.41 | 0.35 0.38 0.91 |
| Euler,<br>MI, i250       | 0.27954        | 0.31673        | 0.28772        | 0.37037        | 0.18561        | 0.28800        |
|                          | 0.35 0.36 0.54 | 0.41 0.32 2.40 | 0.39 0.37 0.47 | 0.41 0.59 0.73 | 0.24 0.26 0.45 | 0.36 0.38 0.92 |
| Euler,<br>MI, i500       | 0.28647        | 0.33112        | 0.27986        | 0.36956        | 0.18570        | 0.29054        |
|                          | 0.36 0.36 0.57 | 0.41 0.32 2.41 | 0.38 0.36 0.46 | 0.41 0.60 0.74 | 0.22 0.27 0.43 | 0.36 0.38 0.92 |
| Euler,<br>SSD, i250      | 0.29979        | 0.32509        | 0.30532        | 0.40449        | 0.19333        | 0.30560        |
|                          | 0.42 0.34 0.60 | 0.48 0.33 2.41 | 0.43 0.37 0.49 | 0.47 0.63 0.74 | 0.25 0.28 0.45 | 0.41 0.39 0.94 |
| Euler,<br>CORR, i250     | 0.29101        | 0.29823        | 0.30135        | 0.39548        | 0.18534        | 0.29428        |
|                          | 0.37 0.34 0.53 | 0.42 0.30 2.41 | 0.39 0.37 0.46 | 0.47 0.61 0.72 | 0.24 0.28 0.42 | 0.38 0.38 0.91 |
| Affine,<br>MI, i250      | 0.34087        | 0.40559        | 0.30186        | 0.33427        | 0.21518        | 0.31955        |
|                          | 0.44 0.38 0.78 | 0.49 0.35 2.70 | 0.41 0.36 0.47 | 0.43 0.44 0.82 | 0.24 0.34 0.43 | 0.40 0.37 1.04 |
| Affine,<br>MI, i500      | 0.35882        | 0.40580        | 0.28688        | 0.34169        | 0.20818        | 0.32027        |
|                          | 0.45 0.39 0.79 | 0.51 0.35 2.69 | 0.42 0.35 0.47 | 0.46 0.45 0.77 | 0.23 0.33 0.43 | 0.41 0.37 1.03 |

Tab. 7.1: Srovnání translační, Eulerovy a afinní transformace s různými parametry pro masku o poloměru 85 px. V tabulce jsou uvedeny hodnoty směrodatných odchylek (v px) vypočítaných z barevných snímků (černé písmo) i jednotlivých barevných složek (odpovídající barvou písma). Zeleně jsou zvýrazněny hodnoty menší než 1. kvartil, světle zeleně hodnoty mezi 1. a 2. kvartilem a červeně hodnoty větší než 3. kvartil. V tabulce jsou uvedeny hodnoty pouze 5 profilů, průměr je ale vypočítán ze všech 7 měřených profilů.

| Transformace               | Profil 1                  | Profil 2                  | Profil 3                  | Profil 4                  | Profil 5                  | Průměr                    |
|----------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|
| Euler, MI,<br>i250         | 0.27954<br>0.35 0.36 0.54 | 0.31673<br>0.41 0.32 2.40 | 0.28772<br>0.39 0.37 0.47 | 0.37037<br>0.41 0.59 0.73 | 0.18561<br>0.24 0.26 0.45 | 0.28800<br>0.36 0.38 0.92 |
| Euler, MI,<br>i250, full   | 0.28798<br>0.34 0.37 0.58 | 0.31470<br>0.41 0.31 2.40 | 0.28260<br>0.39 0.35 0.47 | 0.37256<br>0.41 0.61 0.73 | 0.18600<br>0.22 0.26 0.42 | 0.28877<br>0.36 0.38 0.92 |
| Euler, MI,<br>i250, prev   | 0.21941<br>0.27 0.34 0.48 | 0.33411<br>0.39 0.40 2.48 | 0.43991<br>0.50 0.51 0.53 | 0.38671<br>0.43 0.62 0.71 | 0.22104<br>0.27 0.33 0.45 | 0.32024<br>0.37 0.44 0.93 |
| Euler, MI,<br>i250, CLAHE  | 0.29734<br>0.36 0.35 0.56 | 0.44687<br>0.62 0.41 2.52 | 0.29512<br>0.44 0.34 0.43 | 0.30691<br>0.30 0.57 0.71 | 0.16596<br>0.25 0.27 0.44 | 0.30244<br>0.39 0.39 0.93 |
| Euler, MI,<br>i100, m130   | 0.35645<br>0.41 0.39 1.93 | 0.18686<br>0.23 0.21 3.40 | 0.23270<br>0.33 0.33 2.48 | 0.56757<br>0.70 0.69 4.43 | 0.24054<br>0.29 0.32 0.53 | 0.31682<br>0.39 0.39 2.55 |
| Euler, MI,<br>i250, m130   | 0.35395<br>0.40 0.39 3.34 | 0.18591<br>0.22 0.21 4.08 | 0.22164<br>0.30 0.31 4.60 | 0.54062<br>0.64 0.65 3.51 | 0.23582<br>0.30 0.32 2.06 | 0.30759<br>0.37 0.37 3.52 |
| Euler, MI,<br>i500, m130   | 0.36968<br>0.43 0.39 3.47 | 0.19292<br>0.21 0.21 4.24 | 0.22177<br>0.30 0.31 4.80 | 0.55129<br>0.66 0.67 4.02 | 0.22817<br>0.29 0.31 3.02 | 0.31276<br>0.38 0.38 3.91 |
| Euler, SSD,<br>i250, m130  | 0.32105<br>0.39 0.37 0.82 | 0.20757<br>0.23 0.24 2.46 | 0.25756<br>0.35 0.32 2.13 | 0.57451<br>0.72 0.68 3.75 | 0.20725<br>0.28 0.29 0.38 | 0.31358<br>0.40 0.38 1.91 |
| Euler, CORR,<br>i250, m130 | 0.35024<br>0.40 0.37 1.03 | 0.18573<br>0.23 0.22 4.08 | 0.24245<br>0.30 0.31 3.42 | 0.55750<br>0.67 0.67 3.82 | 0.21330<br>0.28 0.30 2.74 | 0.30984<br>0.38 0.37 3.02 |

Tab. 7.2: Srovnání dalších parametrů registrace, metodika je schodná jako u předchozí tabulky. Full označuje plné vzorkování (až 5x pomalejší výpočet), prev registraci k předchozímu snímku, i250 maximální počet iterací 250 a m130 masku o poloměru 130 px.

## 7.8 Diskuse zjištěných výsledků a doporučení pro nastavení registrace

V tabulkách srovnávajících různé parametry registrace je koncentrováno velké množství informací. Pro zpřehlednění bylo podbarveno pozadí buněk na základě vypočítaných kvartilů tak, jak je uvedeno v popisu první tabulky. Řádky s největším počtem zelených buněk popisují nejkvalitnější registraci, řádky s převahou červených buněk potom nejméně kvalitní registraci.

Za referenční nastavení lze považovat defaultní nastavení Eulerovy transformace, metriky vzájemné informace (MI) a maxima 250 iterací (i250). Toto referenční nastavení je pak v jednodušším případě modifikováno v jediném parametru, aby se dosáhlo jednoparametrického srovnání. V další tabulce jsou uvedeny různá další nastavení jako změna velikosti masky nebo vzorkování.

### 7.8.1 Typ registrace

Z tabulky i z dříve uvedených snímků plyne, že nejlepších výsledků lze zpravidla dosáhnout při použití rigidní (Eulerovy) transformace. Zatímco translace není dostatečná pro kompenzaci rotací (byť jsou velmi nízké, do  $1^\circ$ ), afinní a vyšší flexibilní transformace snímky příliš deformují.

Pro srovnání jsou důležité řádky s translační, Eulerovou a afinní transformací při jinak shodném MI a 250 iteracích. Eulerova transformace s průměrnou směrodatnou odchylkou 0.28 px dopadla jednoznačně nejlépe, afinní transformace se směrodatnou odchylkou 0.32 px nejhůře.

### 7.8.2 Metrika

Podobně byla jednoparametricky srovnána metrika, jde o řádky Eulerovy transformace s 250 iteracemi postupně pro metriku vzájemné informace (MI), Euklidovskou metriku (SSD) a výpočet korelačního koeficientu. Dosažené hodnoty směrodatných odchylek jsou (v uvedeném pořadí) 0.288 px, 0.306 px a 0.294 px. Je patrné, že ačkoliv jsou rozdíly mezi těmito souhrnnými hodnotami malé, nejlepších výsledků dosáhla metrika vzájemné informace. Takový výsledek byl sice na základě 4.6 Využití filtrace pro multimodalitní registraci předpokládán, avšak závisí na kvalitě předregistrační filtrace, proto doporučuji vždy používat metriku vzájemné informace, která by obecně měla vést ke kvalitnějším výsledkům a eliminovat případnou nedokonalost filtru.

### 7.8.3 Počet iterací

Zvyšující se počet iterací vede v některých případech k horším výsledkům registrace. Tato závislost kvality registrace na počtu iterací je neočekávaná a téměř paradoxní. Konkrétně v případě Eulerovy transformace s kritériem MI lze pro maximální počet iteračních kroků 100, 250 a 500 pozorovat průměrné směrodatné odchylky 0.279 px, 0.288 px a 0.291 px.

Tento výsledek není pravidlem, v některých jiných případech začne kvalita registrace oscilovat kolem určité hodnoty aniž by se zhoršovala, avšak alespoň v případě jednoduchých registrací se již ani významně nezlepší nad hodnotu získanou při 250 iteracích. Příčiny oscilací lze pozorovat odděleně v jednotlivých profilech a v jednotlivých barevných složkách. Jak je snímek posouván, tak se registrace některých regionů zlepšuje, zatímco u jiných se zhoršuje.

Doporučení pro jednoduché transformace je volit počet registračních kroků do 250.

### 7.8.4 Velikost masky

Vliv velikosti masky je srovnatelný jak mezi vzájemně si odpovídajícími řádky, tak i mezi první a druhou tabulkou celkově. Větší maska o poloměru 130 px byla testována kvůli hypotéze, že zohlednění větší části snímku pro registraci povede k lepší kvalitě. Avšak ukázalo se, že vliv osvětlovacích diod na zhoršení registrace je větší, než vliv většího ROI na zlepšení registrace. V tabulce 7.2 lze pozorovat téměř přesné rozdělení zelených řádků pro masku o poloměru 85 px a červených pro větší masku se 130 px. Na základě těchto údajů lze jednoznačně doporučit práci s menší maskou, která nezasahuje do částí snímků poškozených odlesky.

### 7.8.5 Registrace barevných kanálů

Již dříve bylo uvedeno, že největší problémy jsou s registrací modré složky, která má nejnižší SNR a odlesky v ní značně přesahují jasový rozsah scény. Červený a zelený kanál se zpravidla daří registrovat kvalitně bez speciální úpravy nastavení.

Tyto závěry jednoznačně potvrzují naměřené hodnoty směrodatných odchylek. Směrodatné odchylky registrovaných červených a zelených kanálů se až na výjimky drží pod hranicí 0.5 px. Odchylky modrého kanálu dosahují až 3.5 px a v jednom případě velké masky překročily hodnotu 6 px, což je již za hranicí možností této měřicí metodiky<sup>3</sup>. Na druhou stranu, vizuálně se ukázalo, že chybně zregistrovaný

---

<sup>3</sup>Snímané jasové profily jsou pozicovány staticky a jejich délka je omezená, pokud některá hodnota přesahuje hranici o více jak cca 3 px, obvykle to znamená, že část snímků v sekvenci je zregistrována zcela chybně, až o desítky px.

modrý kanál nemá významný vliv na vzhled výsledných sekvencí.

### 7.8.6 Registrace k předchozímu snímku

V tabulce 7.2 je přímo srovnána registrace k referenčnímu snímku (první řádek) a k předchozímu snímku (třetí řádek s parametrem *prev*). Z hodnot vyplývá, že většina profilů je registrována lépe při registraci k referenčnímu snímku, což se odráží i v průměrné odchylce 0.29 px ve srovnání s 0.32 px pro registraci k předchozímu snímku. Výchozí registrace k referenčnímu snímku je stabilnější a lze ji doporučit.

### 7.8.7 Vzorkování pro výpočet metriky

Pro většinu zde uvedených výsledků je používán relativně rychlý náhodný sampler, který pro výpočet metriky používá pouze určitý náhodný výběr bodů obrazu. Pro srovnání bylo několik sekvencí registrováno s nastavením *full* (což je i označení parametru v tabulce), kdy jsou pro výpočet metriky použity všechny dostupné body obrazu. Ukázalo se, že nastavení plného vzorkování nevede k lepším výsledkům. Vzhledem k tomu, že je i výpočetně náročnější a cca 4x pomalejší, nelze ho doporučit.

### 7.8.8 Předregistrační filtrace

Referenční předregistrační filtrace je filtrace pásmovou propustí (dále PP), jako testovaná alternativa byl vybrán uživatelský filtr využívající CLAHE (adaptivní ekvalizace histogramu) spolu s dolní propustí. Odchylka u PP dosahuje 0.289 px, u CLAHE filtru 0.302 px; rozdíly jsou malé, avšak jednoduchá PP vede k lepším výsledkům.

## 7.9 Časová náročnost registrace

Časovou náročnost zpracování celé sekvence určuje především doba registrace v programu elastix (a částečně s tím spojené převody formátů dat). Ta je závislá na nastavených parametrech registrace a dá se předpokládat, že je nejrychlejší možná. Ze strany vytvářené aplikace bylo možné pouze omezit počet prováděných registrací dvouprůchodovou interpolací, což je detailněji popsáno v implementační části práce.

Výsledkem je, že celý proces vytváření videosekvence s cca 50 původními snímky (150 po oddělení barevných složek) může v závislosti na nastavení trvat od deseti minut do několika hodin<sup>4</sup>. Tento čas by pro reálné nasazení nebyl ideální, ale v této fázi jsem kladl důraz spíše na obecnost, nastavitelnost a rozšiřitelnost aplikace než na maximální výkonnostní optimalizaci.

---

<sup>4</sup>Testováno na Intel Core i52430M 2.4 GHz.



## 8 ANALÝZA SNÍMKŮ

Tato kapitola navazuje na hlavní část diplomové práce věnující se registraci. Vzhledem k tomu, že se podařilo dosáhnout kvalitní registrace, byly aplikovány alespoň základní postupy analýzy využívající registrované snímky.

### 8.1 Kumulační zvýraznění signálu v šumu

V jednotlivých snímcích se objevuje značný aditivní šum, který má však náhodnou povahu. Jednou z kvalitních metod pro jeho odstranění, využívající existence většího množství snímků téže scény, je souborové průměrování jednotlivých snímků [41]. Nově vytvořený snímek je sestaven dle

$$I_y(kT) = \sum_{i=k-N}^{k+N} \frac{1}{2N+1} I_x(iT),$$

kde  $k$  je pořadí snímku v sekvenci,  $T$  představuje periodu vzorkování a  $N$  určuje velikost průměrovacího okna.

Příklad nového kumulací vytvořeného snímku ve srovnání s původním snímkem je na obr. 8.1, nahoře je původní snímek, dole snímek vzniklý kumulací.

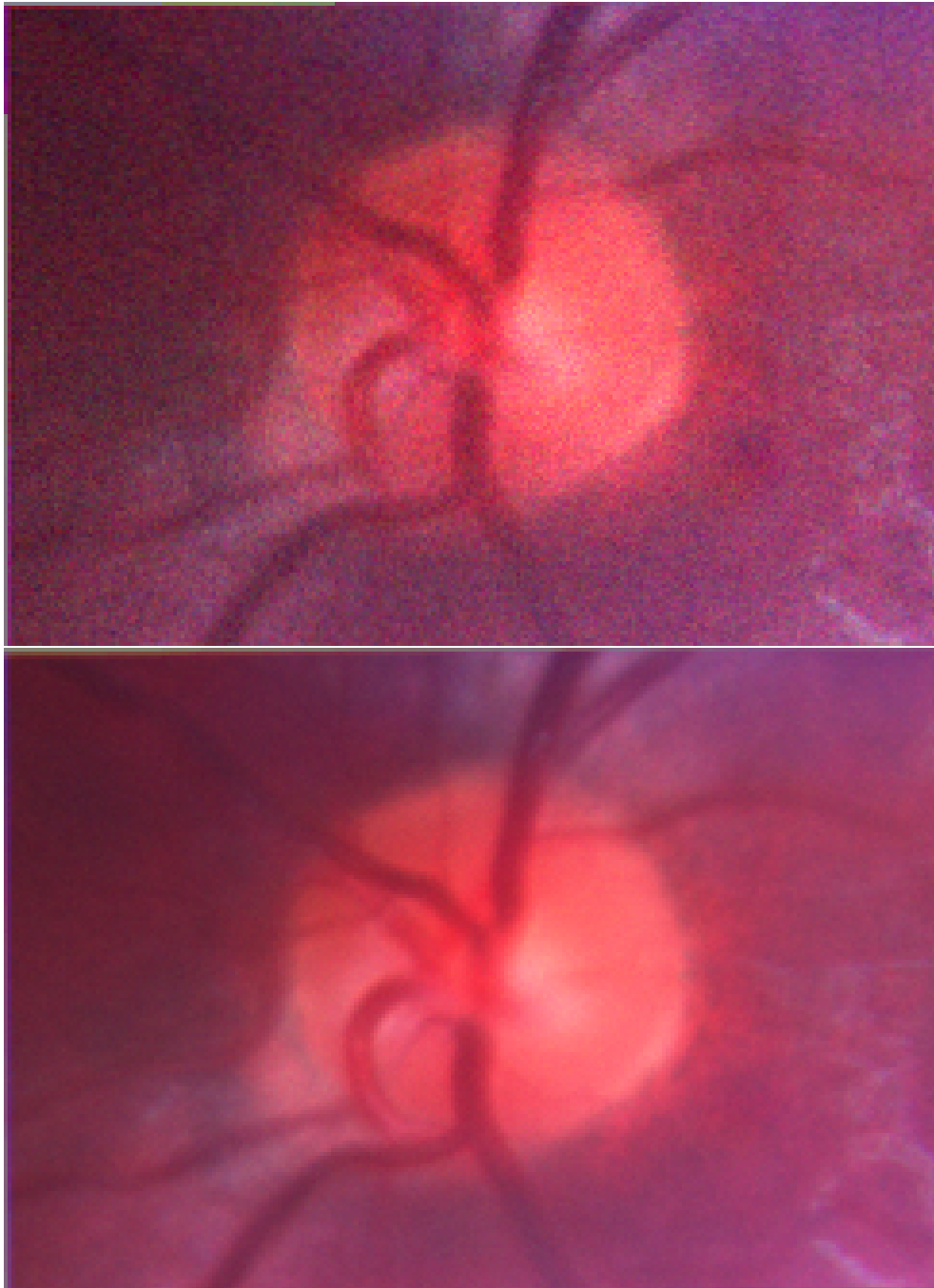
Snímky vzniklé souborovou kumulací přináší nový pohled na data. Nelze v nich sice pozorovat pulzaci cév (kvůli které mohou být ve výsledném snímku hrany cév méně ostré), avšak na druhou stranu v nich jsou značně zvýrazněny strukturní detaily, které nebyly v původním snímku vůbec viditelné. Např. na obr. 8.1 se v horní a levé části objevily cévy, které byly předtím skryté v šumu a taktéž ve střední části jsou nyní pozorovatelné mnohem větší anatomické detaily.

Význam potlačení šumu se ještě více projeví, pokud se ve výsledném snímku pokusíme zvýraznit některé detaily. Na obr. 8.2 jsou uvedeny snímky po aplikaci adaptivního vyrovnání histogramu, tj. adaptivní zvýraznění kontrastu vede ke zvýraznění detailů i v nevhodně osvětlených částech snímku.

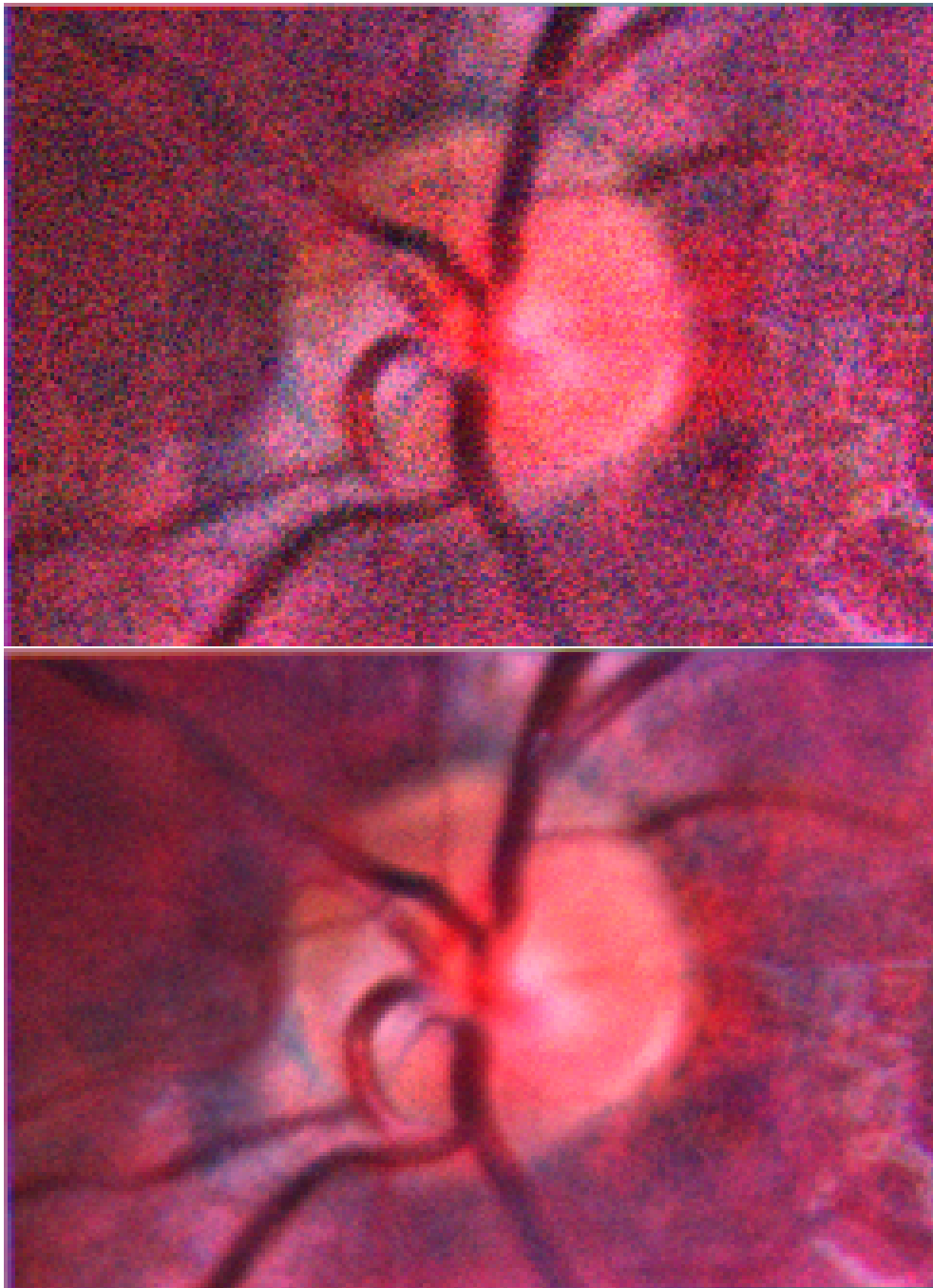
Pro kvantitativní hodnocení zlepšení poměru signálu k šumu bylo z několika co nejvíce homogenních oblastí určeno SNR dle vztahu

$$SNR = 10 \log_{10} \frac{\mu_{sig}}{\sigma_{sig}}.$$

V původních snímcích se SNR pohybuje kolem 10 dB, v souborových průměrech potom kolem 13 dB, průměrné zlepšení je tedy o 3 dB. Pro srovnání, v článku [42] autoři zmiňují zlepšení o 10 dB z 10 dB na 20 dB. Zlepšení SNR zjevně závisí na volbě regionu a na dalším zpracování snímků, mně se pouze registrací a souborovou kumulací podařilo zlepšení maximálně o 4 dB.



*Obr. 8.1: Srovnání původního snímku (nahore) a snímku vzniklého průměrováním uvedeného snímku se sousedními (dole)*



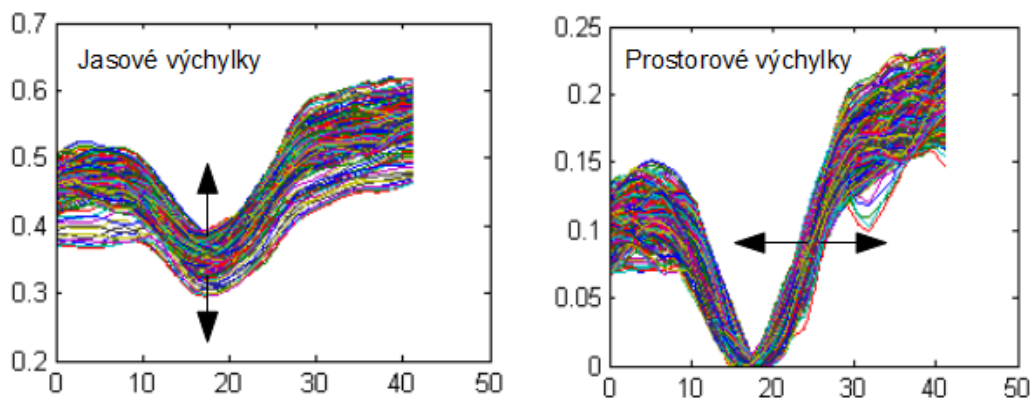
*Obr. 8.2: Srovnání výsledků dosažených adaptivním vyrovnáním histogramu aplikovaným na původní snímek (nahore) a na snímek vzniklý souborovým průměrováním (dole)*

## 8.2 Detekce pulzace pomocí jasových profilů

Jedním z cílů, kvůli kterým byly tyto snímky registrovány, bylo ověření, jestli bude v zaregistrované sekvenci možné pozorovat pulzace cév. Ukázalo se, že registrace je dostatečně kvalitní a pulzace větších cév jsou pozorovatelné. V menších cévách bohužel patrné nejsou, avšak to je způsobeno nízkým SNR, nikoliv nedostatečnou kvalitou registrace.

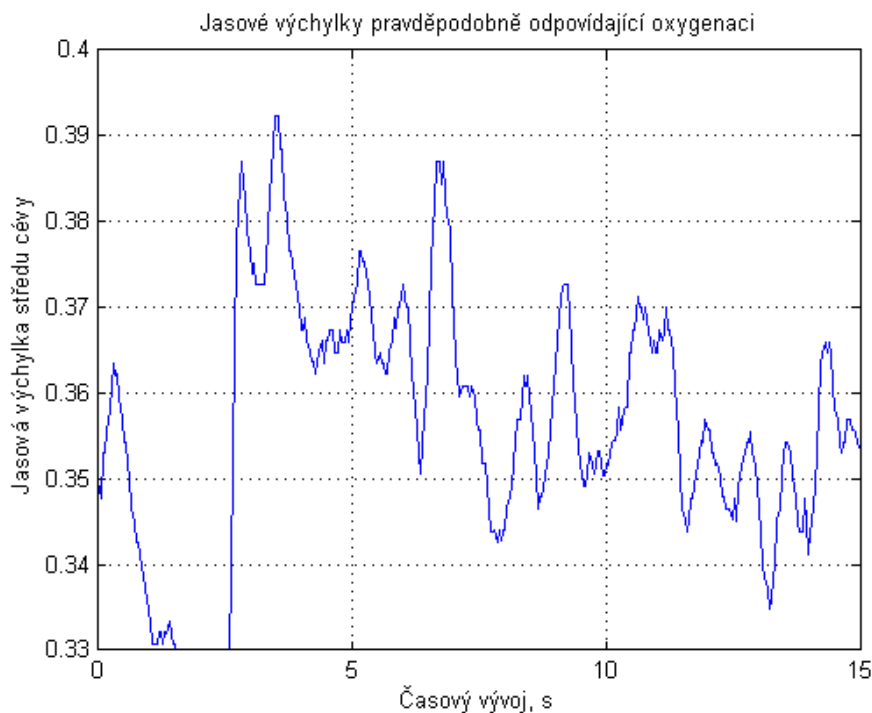
Pro kvantitativní vyhodnocení byla opět použita metoda vycházející z jasových profilů, které jsou snímány podobně jako v kapitole 7.6 Kvantitativní hodnocení registrace. K vyhodnocení pulzace cév byly použity 2 metodiky, viz obr. 8.3.

- První přístup vychází ze změny jasů (barvy) uprostřed cévy, zohledňuje mj. oxygenaci krve. V nasnímaných profilech je nalezena nejnižší jasová hodnota, která je nadále vyhodnocována v časové závislosti, viz graf na obr. 8.4. V takto získané posloupnosti (nebyla provedena horizontální normalizace) jsou i velké skoky zjevně způsobené nerovnoměrným osvětlením, ale vedle nich lze pozorovat menší pravidelné oscilace, zřejmě odpovídající pulzující krvi.
- Druhý přístup vychází z prostorové změny hranic cév, zohledňuje mechanické roztažení cévy. Pro tuto metodiku jsou nasnímané jasové profily opět horizontálně normalizovány<sup>1</sup> a přibližně v poloviční výšce profilu se určí vertikální pozice odpovídající aktuální výchylce cévy v px. Získané výsledky jsou uvedeny na obr. 8.5.



Obr. 8.3: Metody detekce pulzace cév z jasových profilů podle jasových a prostorových výchylek. Tyto grafy jsou spíše schematické, popisy jsou shodné jako na 7.7, tedy  $x$ -ová osa představuje prostorové rozlišení v px,  $y$ -ová osa bezrozměrně vyjádřenou jasovou distribuci.

<sup>1</sup>Kvůli tomuto kroku se výsledky prostorových výchylek stávají částečně závislými na jasových výchylkách, takže toto měření bohužel nelze použít jako nezávislou kontrolní skupinu.

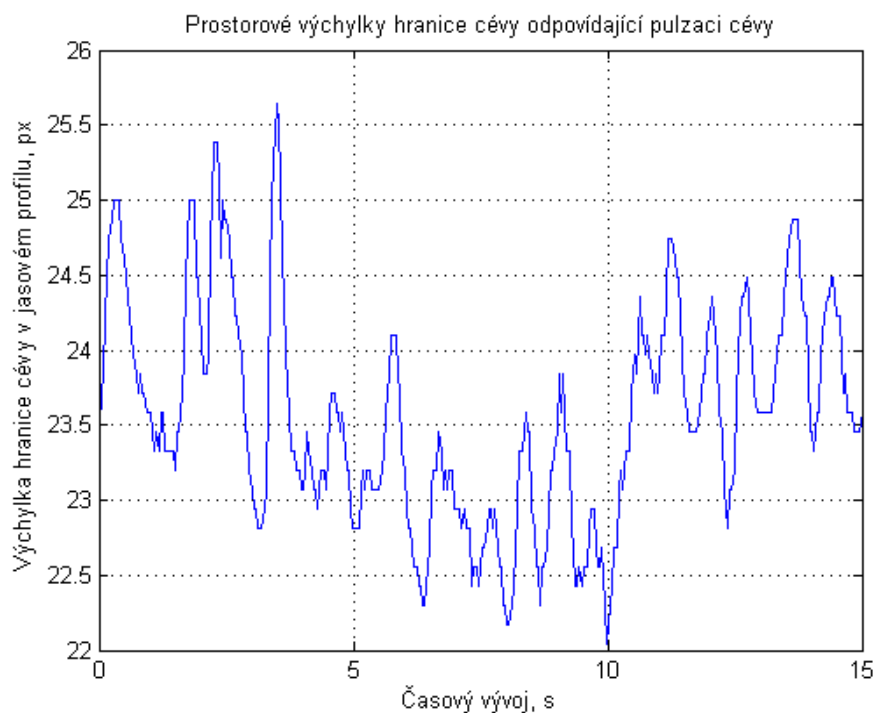


Obr. 8.4: Závislosti detekovaných jasových výchylek středu cévy na čase.

Jednotlivé metody byly původně inspirovány např. analýzou principiálních komponent (PCA) [45] nebo detekcí fluktuace absorpce hemoglobinu [46], avšak pro zde zpracovávané snímky musely být značně zjednodušeny až do podoby zde použité analýzy jasových profilů.

S hodnocením jak přesnosti detekované pulzace, tak i validity výsledků prezentovaných v této podkapitole jsem velmi opatrný, protože neznám přesné časování akvizice a jeden z konzultantů vyjádřil obavy ohledně samotné možnosti pulzaci detekovat<sup>2</sup>. Je třeba zdůraznit, že ačkoliv jsou detekované oscilace přičítány pulzaci cév, nelze vyloučit i jiné vlivy (i když pulzace cév je nejpravděpodobnější). Pokud jde o přesnost, získaná data závisejí na časové interpolaci, časová kalibrace x-ových os grafů nemusí být přesná (nejsou mi známy přesné podmínky akvizice) a pro další analýzu cévních pulzací by jednoznačně bylo nutné zjistit detailní informace o akvizici a znovu zhodnotit celou metodiku.

<sup>2</sup>Některé sekvence byly snímány metodou rychlé akvizice všech 3 barevných kanálů na začátku cca 1sekundových intervalů. Spolu s předpokládanou tepovou periodou cca 1 sekundy by pak bylo kvalitní nasnímáné pulzace prakticky nemožné.

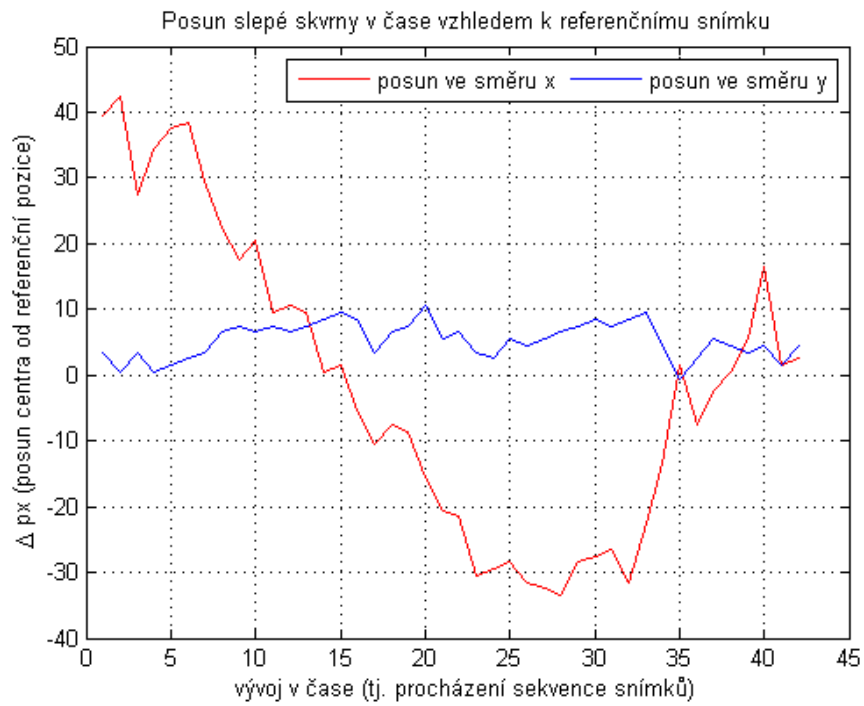


Obr. 8.5: Závislosti detekovaných prostorových výchylek cévní stěny na čase.

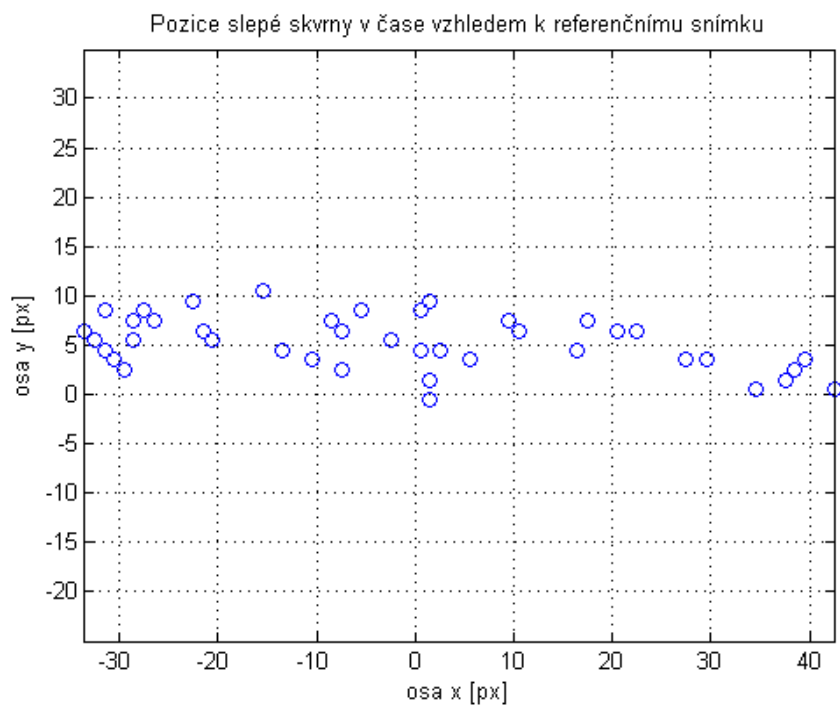
### 8.3 Detekce pohybu očí

Další z vyhodnotitelných charakteristik, které se u retinálních sekvencí někdy zpracovávají, je detekce pohybu oka. V tomto případě je vzhledem k relativně nízké snímkovací frekvenci diskutabilní, jak velkou diagnostickou hodnotu má tato analýza pro rychlé oční pohyby, ale alespoň lze pozorovat pomalejší posuny. Na všech sekvencích je patrný větší pohyb oka v horizontálním směru a jen minimální pohyb ve vertikálním směru.

Detekce pozice slepé skvrny je v této aplikaci nezbytným krokem v registraci obrazů a kvůli její diagnostické hodnotě jsou jednou nalezené pozice na pozadí normalizovány a ukládány pro případné další využití. Kdykoliv po registraci je možné zobrazit si je ve formě grafů, které jsou k dispozici ve 2 základních, v literatuře nejčastěji se objevujících, podobách. První zobrazuje oddělené x-ové a y-ové pozice v závislosti na čase, viz 8.6, a druhá vynáší jednotlivé pozice se zachováním 2D souřadnic bez časové osy 8.7.



Obr. 8.6: Závislosti pohybu oka na čase



Obr. 8.7: Police slepé skvrny v čase se zachováním 2D rozložení

## ZÁVĚR

Tato diplomová práce se věnuje registraci, interpolaci a analýze obrazových dat z experimentální nízkonákladové nemydriatické fundus kamery. Retinální snímky vstupující do procesu zpracování byly pořízeny prototypem nové kamery a předmětem této práce je spíše výzkumné hodnocení možností zpracování těchto snímků a jejich využití pro diagnostiku, než vytvoření softwarového nástroje pro rutinní použití.

V rešeršní části práce jsou uvedeny známé principy a východiska pro registraci a interpolaci obrazových dat s důrazem na techniky využívající podobnostní kritéria.

Druhá část práce poskytuje náhled na samotnou realizaci softwaru, popisuje všechny podstatné kroky od nahrání dat až po vytvoření výsledné snímkové sekvence a videosekvence. Při tvorbě registrační aplikace byl kladen důraz na podstatné a specifické aspekty konkrétní aplikace, nikoliv na „znovuobjevování kola“ opětovným psaním známých algoritmů. Proto byl pro registraci využit velmi kvalitní open source program elastix a vytvořená programová implementace se soustředí na digitální filtraci, analýzu a další úkony navržené přesně pro tento projekt.

Hlavním výstupem této práce je vytvoření softwarového nástroje pro široce nastavitelnou registraci a interpolaci fundus snímků. Kromě širokých možností nastavení parametrů přímo z GUI jsou podporovány libovolné uživatelské filtry (napsané v Matlabu, ale nastavitelné z GUI) pro předregistrační filtraci. Navíc návrh aplikace počítá s tím, že by pro registraci mohl být využit i jiný modul než elastix.

Za pomoci tohoto nástroje byla jedna sekvence zpracována pomocí několika desítek různých kombinací registračních parametrů. V získaných sekvencích byl automaticky detekován pohyb cév (v několika ručně zvolených jasových profilech) a z takto detekovaných pozic byla pro každou metodu určena průměrná směrodatná odchylka, kterou lze považovat za objektivní kritérium hodnocení kvality registrace. Nejúspěšnější metody dosahují směrodatné odchylky menší než 0.3 px. Z výsledků testování softwaru zároveň vyplývá doporučení pro nastavování registračních parametrů, přičemž nejdůležitější závěr je, že nejkvalitnějších výsledků lze dosáhnout použitím rigidní registrace s metrikou vzájemné informace spolu s nezbytným maskováním ROI a předregistrační digitální filtrací potlačující nízkofrekvenční odlesky a vysokofrekvenční šum v obraze.

Kromě objektivního hodnocení kvality lze na základě vizuálního posouzení mozaikového srovnání snímků i videa říci, že dosažená přesnost registrace je dostatečně přesná a splnila očekávání. Zadání práce tedy bylo splněno v plném rozsahu.

Nad rámec zadání byla zpracována kapitola Analýza snímků, která prezentuje výstupy základních diagnostických výpočetních metod. Tato kapitola je zároveň zásadní pro hodnocení diagnostické využitelnosti pořízených snímků a může poskytnout určitý náhled na význam celého projektu.



## LITERATURA

- [1] HÖHER B., MICHELSON G., VOIGTMANN P., SCHMAUSS B., *Low-Cost Non-mydrriatic Color Video Imaging of the Retina for Nonindustrialized Countries*. Teleophthalmology in Preventive Medicine, 2014, ISBN 978-3-662-44974-5.
- [2] PANG K.Y., IZNITA L., FADZIL, HANUNG, HERMAWAN, VIJANTH, *Segmentation of Retinal Vasculature in Colour Fundus Images*. Monash University, Sunway campus, Malaysia, 25th & 26th July 2009.
- [3] CHEN J., TIAN J., *Retinal Vessel Enhancement Based on Directional Field*. Institute of Automation, Chinese Academy of Sciences, Beijing, China.
- [4] KANSKI, JACK J., *Clinical Ophthalmology: A Systematic Approach. 6th. ed.* s.l. : Butterworth-Heinemann, 2007.
- [5] WONG T.Y., *Is retinal photography useful in the measurement of stroke risk?*. THE LANCET Neurology Vol 3 March 2004.
- [6] BROWN L.G., *A survey of image registration techniques*. ACM Computing Surveys 24 (1992) 326–376.
- [7] ZITOVA B., FLUSSER J. *Image registration methods: a survey*. Academy of Sciences of the Czech Republic, 2003.
- [8] PRATT W.K., *Correlation techniques of image registration*. IEEE Transactions on Aerospace and Electronic Systems 10 (1974) 353–358.
- [9] PRATT W.K., *Digital Image Processing*. 2nd ed., Wiley, New York, 1991.
- [10] ROCHE A., MALANDAIN G., PENNEC X., AYACHE N., *The correlation ratio as a new similarity measure for multimodal image registration*. Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'98), Lecture Notes in Computer Science, Cambridge, USA, vol. 1496, 1998, pp. 1115–1124.
- [11] CASTRO E.D. MORANDI C., *Registration of translated and rotated images using finite Fourier transform*. IEEE Transactions on Pattern Analysis and Machine Intelligence 9 (1987) 700–703.
- [12] REDDY B.S., CHATTERJI B.N., *An FFT-based technique for translation, rotation and scale-invariant image registration*. IEEE Transactions on Image Processing 5 (1996) 1266–1271.

- [13] CIDECIYAN A.V., *Registration of ocular fundus images*. IEEE Engineering in Medicine and Biology 14 (1995) 52–58.
- [14] BRACEWELL R.N., *The Fourier Transform and Its Applications*. McGraw-Hill, New York, 1965.
- [15] FREIHERR G., *Time-of-Flight Imaging Propels Entry-Level PET/CT*. <http://www.medscape.com/viewarticle/757972>, 2012.
- [16] O’GORMAN L., SAMMON M., MICHEL S., *Practical Algorithms for Image Analysis*. Cambridge University Press, 2008.
- [17] *About FFmpeg*. FFmpeg. [online]. [cit. 2015-02-12]. Dostupné z: <https://www.ffmpeg.org/about.html>
- [18] Jonathan Dunlap. *Further Reading on Design Patterns*. *Effective Stack*. [online]. [cit. 2015-03-18]. Dostupné z: <http://www.jadbox.com/2009/03/further-reading-on-design-patterns/>
- [19] *interp1*. MathWorks – Support. [online]. [cit. 2014-12-10]. Dostupné z: <http://www.mathworks.com/help/matlab/ref/interp1.html>
- [20] *wiener2*. MathWorks – Support. [online]. [cit. 2015-05-05]. Dostupné z: <http://www.mathworks.com/help/images/ref/wiener2.html>
- [21] SETIAWAN A.W.S., MENGKO T.R., SANTOSO O.S., SUKSMONO A.B., *Color Retinal Image Enhancement using CLAHE*. ICT for Smart Society (ICISS), 2013 International Conference on, 13-14 June 2013, ISBN 978-1-4799-0143-2.
- [22] STARK J.A., *Adaptive Image Contrast Enhancement Using Generalizations of Histogram Equalization*. IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 5, MAY 2000.
- [23] MORAVČÍKOVÁ H., *Funduskamera*. oční ordinace a oční optika. [online]. [cit. 2014-11-23]. Dostupné z: <http://www.mudrmoravcikova.cz/rady-lekare/fundus-kamera/>
- [24] SCHWENN O.. *Ophthalmology*. Johannes Gutenberg University Mainz. [online]. [cit. 2014-11-28]. Dostupné z: <http://www.uni-mainz.de/FB/Medizin/Anatomie/workshop/Klinisches/Retina1E.html>
- [25] SAINÉ P.J., TYLER M.E.. *Fundus Photography Overview*. Ophthalmic Photographers’ Society. [online]. [cit. 2015-02-15]. Dostupné z: <http://www.uni-mainz.de/FB/Medizin/Anatomie/workshop/Klinisches/Retina1E.html>

- [26] New Nidek AFC-230 Auto Fundus Camera with Canon EOS D5. Great Medical Store. [online]. [cit. 2015-02-16]. Dostupné z: <http://great-medical.com/products/New-Nidek-AFC%252d230-Auto-Fundus-Camera-with-Canon-EOS-D5.html>
- [27] KLEIN S., STARING M., MURPHY K., VIERGEVER M.A., PLUIM J.P.W., *elastix: a toolbox for intensity based medical image registration*. IEEE Transactions on Medical Imaging, vol. 29, no. 1, pp. 196 - 205, January 2010.
- [28] KLEIN, S., STARING, M., *Elastix: the manual*. [online]. February 12, 2014. Dostupné z: [http://elastix.isi.uu.nl/download/elastix\\_manual\\_v4.7.pdf](http://elastix.isi.uu.nl/download/elastix_manual_v4.7.pdf).
- [29] NIXON, M., AGUADO, A., *Feature Extraction & Image Processing*. Elsevier Ltd., 2008, ISBN 978-0-1237-2538-7.
- [30] PARKER J.A., KENYON R.V., TROXEL D.E., *Comparison of Interpolating Methods for Image Resampling*. IEEE Transactions on Medical Imaging, vol. MI-2, NO. 1, MARCH 1983.
- [31] JAN J., *Medical Image Processing, Reconstruction and Restoration*. CRC Taylor and Francis (USA), 2006, ISBN 0-8247-5849-8.
- [32] WALEK, P., LAMOŠ, M., JAN, J., *Analýza biomedicínských obrazů, Počítačová cvičení*. Brno: VUT, 2013, ISBN 978-80-214-4792-9.
- [33] VIOLA P., WELLS W.M., *Alignment by maximization of mutual information*. International Journal of Computer Vision 24 (1997) 137–154.
- [34] Wikipedia, the free encyclopedia. *Mutual information*. [online]. [cit. 2015-02-16]. Dostupné z: [http://en.wikipedia.org/wiki/Mutual\\_information](http://en.wikipedia.org/wiki/Mutual_information)
- [35] HAUPT R.L., HAUPT S.E., *Practical Genetic Algorithms*. John Wiley & Sons, New Jersey, 2004, ISBN 978-0-471-45565-3.
- [36] HAJNAL, J.V., HILL D.L.G., *Medical Image Registration*. CRC Press - Taylor and Francis Group (USA), 2001, ISBN 9780849300646.
- [37] GOSHTASBY, A.A., *2-D and 3-D Image Registration for medical, remote sensing, and industrial applications*. Wiley-Interscience, 2005, ISBN 0-471-64954-6.
- [38] PLUIM J.P.W., MAINTZ J.B.A., VIERGEVER M.A., *Mutual Information-Based Registration of Medical Images: A Survey*. IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 22, NO. 8, AUGUST 2003.

- [39] MAES F., VANDERMEULEN D., SUETENS P., *Medical Image Registration Using Mutual Information*. IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 91, NO. 10, OCTOBER 2003.
- [40] CORON A., MAMOU J.. *ElastixFromMatlab toolbox*. The Higher Education and Research forge. [online]. [cit. 2014-11-16]. Dostupné z: <http://great-medical.com/products/New-Nidek-AFC%252d230-Auto-Fundus-Camera-with-Canon-EOS-D5.html>
- [41] JAN J., *Číslíková filtrace, analýza a restaurace signálu*. VUTIUM, 2002, ISBN 80-214-2911-9.
- [42] KOLAR R., HOEHER B., ODSTRCILIK J., SCHMAUSS B., JAN J., *Registration of Image Sequences from Experimental Low-Cost Fundus Camera*. Springer International Publishing, 2014, ISBN 978-3-319-08553-1.
- [43] LI L., REVESZ P., *Interpolation methods for spatio-temporal geographic data*. Computers, Environment and Urban Systems, 2004, ISSN 0198-9715.
- [44] RYAN N., HENEGHAN C., CHAZAL P., *Registration of digital retinal images using landmark correspondence by expectation maximization*. Image and Vision Computing 22, 2004, ISSN 883–898.
- [45] MORET F., POLOSCHEK CH.M., LAGREZE W.A., BACH M., *Visualization of Fundus Vessel Pulsation Using Principal Component Analysis*. Investigative Ophthalmology & Visual Science, July 2011.
- [46] MORGAN W.H., ABDUL-RAHMAN A., YU D., HAZELTON M.L., BETZ-STABLEIN B., LIND R.P., *Objective Detection of Retinal Vessel Pulsation*. PLoS ONE 10(2): e0116475. doi:10.1371/journal.pone.0116475, February 2015.

# SEZNAM PŘÍLOH

|   |            |
|---|------------|
| <b>A Instalace</b>  | <b>101</b> |
| A.1 Softwarové závislosti . . . . .                         | 101        |
| A.2 Hardwarové požadavky . . . . .                          | 101        |
| A.3 Instalace . . . . .                                     | 101        |
| A.4 Konfigurace . . . . .                                   | 102        |
| <b>B Ovládání vytvořené aplikace</b>                        | <b>103</b> |
| B.1 Spuštění aplikace . . . . .                             | 103        |
| B.2 Vytvoření nového projektu . . . . .                     | 103        |
| B.3 Otevření existujícího projektu . . . . .                | 103        |
| B.4 Načtení snímků . . . . .                                | 103        |
| B.5 Výběr ROI . . . . .                                     | 104        |
| B.6 Nastavení masky . . . . .                               | 104        |
| B.7 Registrace a interpolace snímků . . . . .               | 105        |
| B.8 Indikace průběhu výpočtu . . . . .                      | 105        |
| B.9 Prohlížení mezivýsledků . . . . .                       | 106        |
| B.9.1 Export výsledků . . . . .                             | 106        |
| B.10 Ukončení programu . . . . .                            | 106        |
| <b>C Vytvoření uživatelského předregistračního filtru</b>   | <b>107</b> |
| C.1 Příklad vytvoření a použití CLAHE filtru . . . . .      | 107        |
| C.2 Popis rozhraní (interface) filtračních funkcí . . . . . | 108        |
| <b>D Obsah přiloženého CD</b>                               | <b>109</b> |

## A INSTALACE

### A.1 Softwarové závislosti

Základem pro spuštění aplikace je Matlab ve verzi 2010<sup>1</sup> a novějších. Matlab je komerční produkt, pro tuto diplomovou práci byl použit po domluvě s vedoucím a jeho licenci je třeba respektovat. Detailně se s ní lze seznámit na stránkách výrobce MathWorks.

Pro samotnou registraci obrazů je klíčový program elastix [27] spolu s ElastixFromMatlab Toolboxem<sup>2</sup> [40]. Elastix je open source software stavějící na ITK distribuovaný pod licencí odvozenou z BSD, přesná znění licencí jsou opět na stránkách produktů.

Pro vytvoření videa se používá open source aplikace FFmpeg [17]. Její zdrojové kódy sice podléhají licenci GPL, ale pro uživatelské použití (jak je použita zde) je dostupná zdarma bez dalších omezení, detaily opět na stránkách produktů.

Všechny uvedené programy jsou multiplatformní (jsou podporovány ve Windows, Linuxu i OS X), což dělá i vytvořenou aplikaci multiplatformní.

### A.2 Hardwarové požadavky

Aplikace nemá žádné zvláštní nároky, lze očekávat, že pracovní stanice s Matlabem bude dostatečně výkonná pro její spuštění. Nároky aktuální verze Matlabu R2015a jsou uvedeny na stránkách MathWorks v sekci System Requirements. Vzhledem k využitelnosti vícejádrových procesorů programem elastix lze pro rychlejší práci doporučit výkonnější stanice.

### A.3 Instalace

Instalace vytvořené aplikace spočívá především v instalaci softwarových závislostí uvedených výše. Aplikace samotná je napsaná pro Matlab a nevyžaduje speciální instalaci, stačí ji správně nakonfigurovat.

---

<sup>1</sup>Funkčnost na starších verzích nebyla testována.

<sup>2</sup>Při práci s tímto toolboxem bylo nalezeno několik drobných problémů a nekompatibilit s aktuálními verzemi Matlabu. Upravená verze je distribuována společně s ostatními zdrojovými kódy patřícími k této práci.

## A.4 Konfigurace

Součástí zdrojových kódů vytvořené aplikace je konfigurační soubor `ConfigSystem.m`, který obsahuje cesty k pracovním adresářům a externím programům. Konfigurace se zapisuje do k tomu určených proměnných, přičemž detailní pokyny jsou uvedeny v komentářích v daném souboru.

Zde uvádím pouze výčetem, že je třeba nastavit:

- cestu k programu `elastix`,
- cestu k programu `transformix`,
- pracovní adresář pro program `elastix` (musí mít práva k zápisu),
- cestu k programu `ffmpeg`,
- cestu k globální systémové konfiguraci (musí mít práva pro zápis).

Pokud jsou externí programy umístěny v adresářích, do kterých vedou systémové cesty (typicky v proměnné `Path`), lze cesty k nim v konfiguraci vynechat a uvést jen samotný název programu. V tomto stavu je konfigurace distribuována.

## B OVLÁDÁNÍ VYTVOŘENÉ APLIKACE

Tato kapitola se zabývá uživatelským popisem registrování obrazů a vytvářením videosekvence ze surových dat z kamery pomocí vytvořené aplikace. Instalace a konfigurace programu včetně jeho závislostí je popsána v kapitole A Instalace.

### B.1 Spuštění aplikace

S vytvořenou aplikací se pracuje výhradně přes grafické uživatelské prostředí. Ke spuštění nové instance programu slouží script `fundus_register`, který lze volat více způsoby, nejtypičtější je napsat jeho název do *Command Window* v Matlabu, nebo ho otevřít a spustit v Matlabovém Editoru.

### B.2 Vytvoření nového projektu

Aplikace si průběžně ukládá dosažené výsledky, díky tomu je možné opakovat jednotlivé kroky s modifikovaným nastavením parametrů. Aby byla umožněna paralelní práce s více sekvencemi, je cesta do pracovního adresáře parametrizována a právě tuto cestu je třeba zadat unikátně pro každý nový projekt.

Při prvním spuštění aplikace a později při požadavku na vytvoření nového projektu se zobrazí UI dialogové okno s výběrem pracovního adresáře. Konkrétní umístění adresáře je otázkou preferencí uživatele, ale musí být na takové cestě, aby zde měl Matlab práva ke čtení a zápisu souborů.

### B.3 Otevření existujícího projektu

Aplikace si pamatuje umístění naposledy otevřeného projektu, pokud chcete pokračovat v práci s předchozím projektem, bude po znovuspuštění aplikace automaticky otevřen.

Chcete-li otevřít jiný projekt, zvolte možnost „Otevřít“<sup>1</sup> z menu a v dialogovém okně vyberte pracovní adresář s požadovaným projektem.

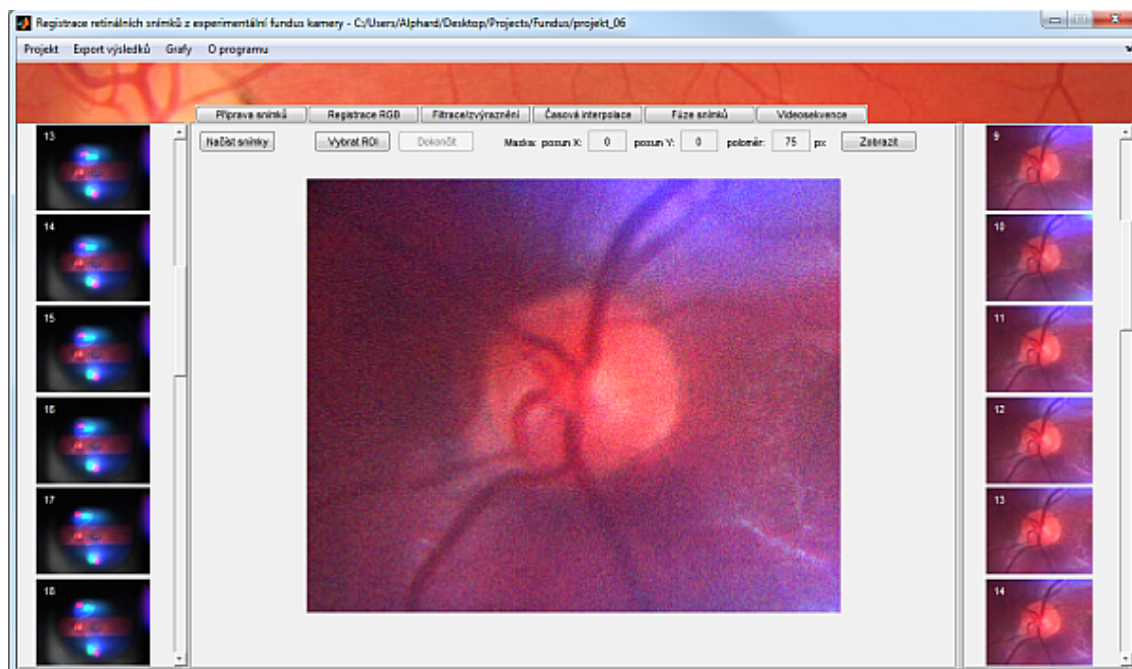
### B.4 Načtení snímků

Po otevření projektu se zobrazí okno podobné obr. B.1. V levném horním rohu hlavní (prostřední) části aplikace klikněte na tlačítko **Načíst snímky**, zobrazí se dialogové

---

<sup>1</sup>Volba „Otevřít“ otevře paralelně nové okno, volba „Otevřít tady“ zavře původní projekt a otevře nový.





Obr. B.1: Screenshot prvního kroku s předregistrací snímků

okno s výběrem adresáře, ve kterém vyberte adresář obsahující obrazová data z fundus kamery. Případné předchozí snímky v projektu budou vymazány.

Po potvrzení se soubory z vybraného adresáře nakopírují do projektového adresáře (originální data nejsou žádným způsobem modifikována). Po dokončení operace se aktualizuje levý průzkumník fotografií, zobrazí se v něm nově načtené snímky.

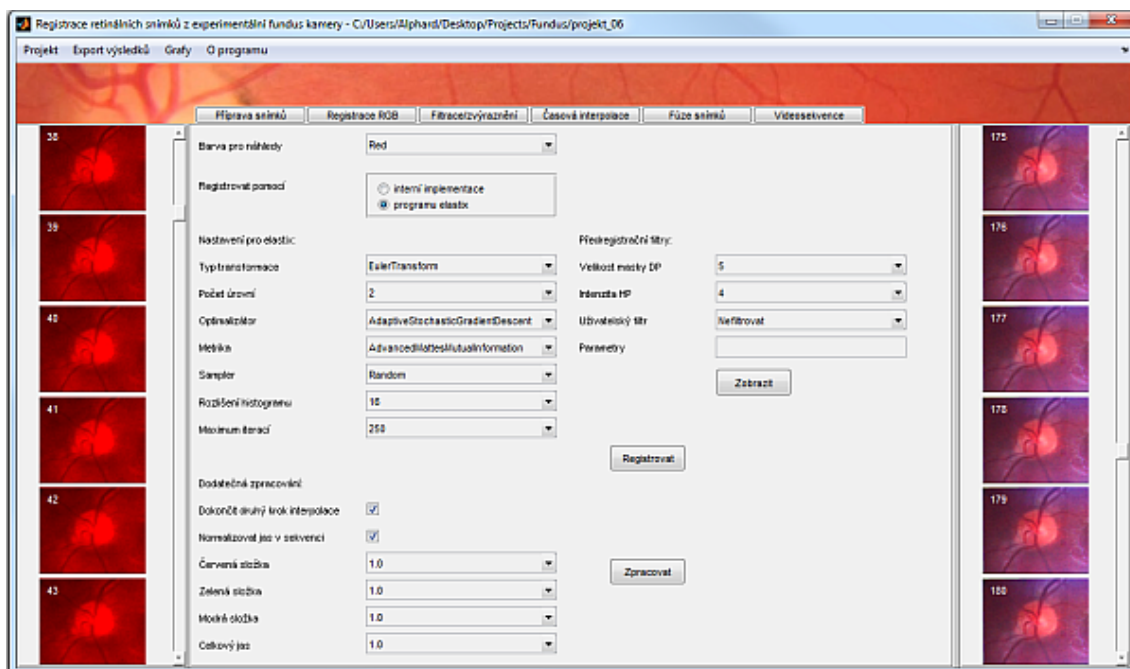
## B.5 Výběr ROI

Po kliku na zmenšeninu snímku v levém průzkumníku se v hlavním okně zobrazí větší verze vybraného obrazu. Projděte vstupní sekvenci a vyberte optimální referenční snímek (bez odlesků a jiných artefaktů).

Po vybrání referenčního snímku se tlačítkem **Vybrat ROI** aktivuje výběr obdélníkové části obrazu, což je indikováno změnou kurzoru na výběrový kříž. Vyberte požadované ROI (viz 5.2) a volbu potvrďte dvojklikem na výběr. Jste-li s výběrem spokojeni, klikněte na tlačítko **Dokončit**. Aplikace následně vyhledá a ořízne ROI ze všech ostatních snímků.

## B.6 Nastavení masky

V tomtéž kroku se nastavuje pozice a velikost masky. Po výběru ROI lze klikem na tlačítko „Zobrazit“ zobrazit aktuální masku (bude zobrazena poloprůhledně na reál-



Obr. B.2: Screenshot závěrečného kroku s fúzí snímků

ném snímku). Nastavením příslušných parametrů lze masku přizpůsobit konkrétním potřebám.

## B.7 Registrace a interpolace snímků

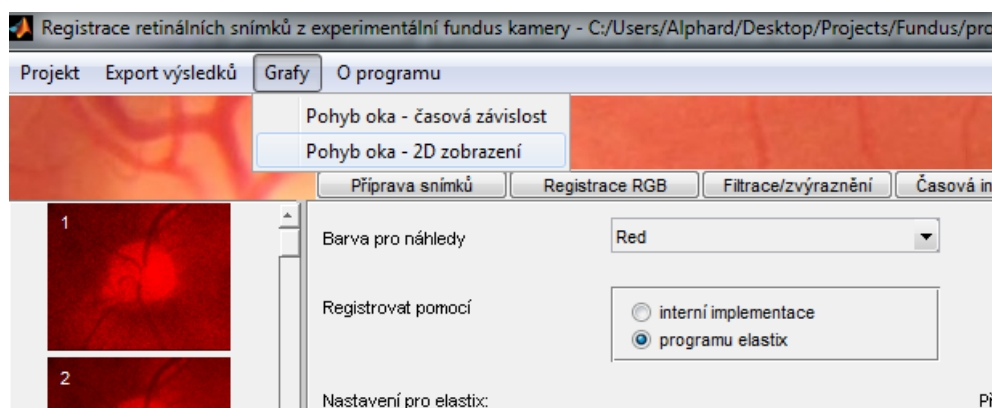
V dalších krocích je již práce se snímky zcela automatická, avšak o to více volitelných parametrů registračního a interpolačního procesu zde lze zadat. Účel všech konfiguračních voleb byl vysvětlen v hlavní části diplomové práce.

Prostřednictvím jednotlivých tlačítek v horním menu **Registrace RGB**, **Filtrace/zvýraznění**, **Časová interpolace**, **Fúze snímků** a **Videosekvence** postupujte zleva doprava a v každém kroku proveďte požadovanou akci. Jednotlivé kroky na sebe navazují, takže není možné je přeskakovat. Můžete se libovolně vrátit zpět a opakovat starší krok s jiným nastavním, ale po každé změně musí být popořadě spuštěny všechny následující akce.

Na obr. B.2 je uveden snímek grafického rozhraní v posledním kroku fúze snímků.

## B.8 Indikace průběhu výpočtu

Registrování velkého množství obrazů může být časově náročné, proto jsou o probíhající operaci uživatelé informováni prostřednictvím progress baru.



Obr. B.3: Detail horního menu. Projekt slouží pro správu projektu (Nový, Otevřít), Export výsledků umožňují exportovat finální snímky a video, Grafy obsahují zobrazení grafů vytvořených v průběhu registračního procesu a v položce O programu jsou uvedeny informace o programu a nápověda.

Klikem na tlačítko **Cancel** lze kdykoliv výpočet validně přerušit<sup>2</sup>, násilné přerušování (**Ctrl+C**) doporučuji nepoužívat, protože může program zanechat v nekonzistentním stavu, ve kterém nelze dále pracovat.

## B.9 Prohlížení mezivýsledků

V libovolném kroku (kromě prvního) lze klikem na zmenšeninu v postranních průzkumnících zobrazit nové okno s vybraným snímkem v plné velikosti. Zobrazené okno obsahuje klasické nástroje jako *Zoom* apod.

Pokud chcete pro prohlížení png obrazů použít jiné nástroje, můžete si soubory otevřít přímo z adresáře s projektem, kde jsou uloženy.

### B.9.1 Export výsledků

Pro jednodušší práci byly do horního menu **Export výsledků** přidány možnosti **Export snímků** a **Export videa**, které exportují výsledné soubory do uživatelem vybraného adresáře.

## B.10 Ukončení programu

Všechny mezivýsledky jsou ukládány na disk. Pokud aktuálně neprobíhá výpočet, lze program bezpečně ukončit zavřením okna, nebo volbou **Ukončit** v horním menu.

<sup>2</sup>Matlab je jednovláknový, kvůli tomu nemusí být odezva okamžitá, ale může chvíli trvat, protože se musí dokončit zpracování aktuálního snímku a řízení programu předat zpět do controlleru.

## C VYTVOŘENÍ UŽIVATELSKÉHO PŘEDREGISTRAČNÍHO FILTRU

Důvody pro implementaci uživatelské filtrace byly uvedeny v kapitole 4.7 Uživatelská filtrace. Připomeňme, že vedle široce nastavitelné registrace elastixem je aplikace vhodných předregistračních filtrů zřejmě nejzásadnějším bodem celého registračního procesu a jako taková má určitě smysl být maximálně nastavitelná.

### C.1 Příklad vytvoření a použití CLAHE filtru

```
0001 % název filtru obsahuje prefix filter_  
0002 % filtr je uložen ve stejnojmenném souboru, např. filter_clahe.m  
0003 %   v adresáři s ostatními scripty  
0004 %  
0005 % do parametru img_in je předána 2D matice představující vstupní  
0006 %   filtrovaný obraz s hodnotami v intervalu <0; 1>  
0007 % do parametru params jsou jako řetězec předány parametry  
0008 %   zadané uživatelem v grafickém rozhraní aplikace  
0009 % ve výstupním parametru img_out se očekává 2D matice o velikosti  
0010 %   vstupní matice a hodnotami v intervalu <0; 1>  
0011  
0012 function img_out = filter_clahe(img_in, params)  
0013     if isempty(params) % kontrola, jestli byl vyplněn parametr  
0014         clipLimit = 0.01; % východzí hodnota  
0015     else  
0016         clipLimit = str2num(params); % konverze řetězce na číslo  
0017     end  
0018     % výkonná část filtru  
0019     img_out = adapthisteq(img_in, 'clipLimit', clipLimit);  
0020 end
```



Obr. C.1: Rozhraní uživatelské předregistrační filtrace v grafickém rozhraní aplikace. Činnost jednou vytvořeného filtru lze modifikovat pomocí parametrů.

## C.2 Popis rozhraní (interface) filtračních funkcí

V předchozí kapitole byl uveden příklad CLAHE filtru, na kterém bude nyní popsán koncept uživatelské filtrace.

Pro zpřístupnění filtru pro předregistrační filtraci je třeba tento filtr pojmenovat s prefixem `filter_` a uložit ho do adresáře spolu s ostatními scripty. Poté bude při startu aplikace automaticky nalezen a nabídnut v selectboxu *Uživatelský filtr* k použití.

Spolu s výběrem vhodného filtru je možné předat tomuto filtru z uživatelského prostředí parametry. Protože obecně mohou být parametry libovolného typu (a může jich být libovolný počet), není možné provádět na straně aplikace implicitní přetypování. Uživatelskému filtru je tedy předán přesně ten řetězec, který byl zadán uživatelem a je třeba ho dále zpracovat (tj. ověřit jeho existenci, konvertovat na číslo, rozdělit více parametrů oddělených čárkami apod.).

Pro případ, že je třeba použít různé parametry pro různé barevné složky, je možné řetězec s parametry zadat ve tvaru `param1;;param2;;param3`, kde `;;` jsou oddělovače. Takový řetězec je aplikací automaticky rozdělen a jednotlivé parametry jsou předány postupně pro filtraci červené, zelené a modré složky (v tomto pořadí). Pokud nejsou oddělovače použity, předá se pro filtraci všech barevných složek stejný parametr (což může být i prázdný řetězec).

Po zpracování parametrů je provedena vlastní filtrace, která může obsahovat jakékoliv operace (včetně volání externích programů), avšak výsledkem musí být obrazová 2D matice velikostně se shodující se vstupní maticí.

## D OBSAH PŘILOŽENÉHO CD

*Ve veřejné verzi nejsou elektronické přílohy k dispozici.*

– /

- **program** – zdrojové kódy vytvořené aplikace (podrobný návod, jak aplikaci nainstalovat a spustit je uveden v kapitolách A.3 a B.1 věnovaných instalaci a spuštění)
- **thesis** – pdf verze této práce
- **data**
  - **input\_fundus\_images** – příklad několika vstupních zpracovávaných sekvencí
  - **registered\_images** – příklad registrovaných sekvencí
  - **comparison** – mozaikové a video srovnání původních a registrovaných snímků