



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

DEPARTMENT OF INTELLIGENT SYSTEMS

**ALGORITMUS PRO REALISTICKOU TVORBU CÉVNÍHO
ŘEČIŠTĚ SYNTETICKÝCH SÍTNIC**

ALGORITHM FOR REALISTIC VASCULAR CIRCULATION OF SYNTHETIC RETINA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MARTIN DVOŘÁK

VEDOUcí PRÁCE

SUPERVISOR

Ing. LUKÁŠ SEMERÁD

BRNO 2019

Zadání bakalářské práce



21963

Student: **Dvořák Martin**
Program: Informační technologie
Název: **Algoritmus pro realistickou tvorbu cévního řečiště syntetických sítnic**
Algorithm for Realistic Vascular Circulation of Synthetic Retina
Kategorie: Umělá inteligence

Zadání:

1. Nastudujte základní informace o sítnici, zejména obvyklý způsob tvorby cévního řečiště.
2. Prozkoumejte existující nástroj uvedený v literatuře pro tvorbu syntetických sítnic.
3. Navrhněte algoritmus pro tvorbu cévního systému, založený na větvení reálných sítnic.
4. Implementujte algoritmus z třetího bodu do existujícího nástroje.
5. Vytvořte novou sadu syntetických sítnic a ověřte reálné rozložení jejich markantních znaků.

Literatura:

- Drahanský M., Orság F., Doležel M. et al.: *Biometrie*, Brno, CZ, Computer Press, 2011, s. 294, ISBN 978-80-254-8979-6.
- Sehu K. W., Weng W. R.: *Ophthalmic pathology: an illustrated guide for clinicians*, Oxford, Blackwell Publishing, 2005, s. 289, ISBN-13: 978-0-727917-79-9
- Tretter Z.: *Generování syntetických obrazů sítnic oka*, Brno, 2017. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Martin Drahanický, Ph.D.
- Semerád L., Biswas S., Skorkovská K., Drahanický M.: *Statistics of Bifurcations*, Springer Publishing, 2018.

Pro udělení zápočtu za první semestr je požadováno:

- První tři body zadání.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Semerád Lukáš, Ing.**
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 29. dubna 2019

Abstrakt

Cílem této práce je zdokonalit existující nástroj na vytváření syntetických sítnic oka. Zlepšení dosavadního softwaru probíhá změnou algoritmu pro tvorbu cévního řečiště. Zvolený problém byl vyřešen navržením algoritmu, který vychází z reálného chování cévního systému v sítnici oka. Skutečné snímky byly statisticky zaznamenány. Matematické programy vytvořily aproximační křivky cév. Byly sledovány směry cév, místa větvení a počet větvených cév. Na základě zjištěných informací a pravděpodobnosti větvení byl navržen model pro generování syntetických snímků. Hlavním výsledkem jsou syntetické sítnice, které obsahují vzory chování skutečných sítnic oka. Tyto snímky mají podobný počet cév a směřují do stejných oblastí jako reálné snímky.

Abstract

The aim of this bachelor thesis is to improve an existing tool for creating synthetic retina of the eye. The improvement of an existing software is carried out by changing the algorithm for creation vascular bed. The problem was solved by designing an algorithm which is based on real behavior of vascular system in the retina of the eye. Real pictures were statistically recorded. Mathematical programs created approximation curves of the blood vessels. Directions of the blood vessels, locations of branching and a number of branching of blood vessels were observed. A model for generation of synthetic pictures was designed based on discovered information and probability of branching. The main results are synthetic retinas which contain patterns of behavior of a real retina of the eye. These pictures have similar number of blood vessels, aiming towards the same location and branching in the same points as real pictures.

Klíčová slova

biometrie, sítnice, syntetická sítnice, C++, algoritmus, automatické generování, statistické pozorování

Keywords

biometry, retina, synthesis retina, C++, algorithm, automatic generation, statistical observation

Citace

DVOŘÁK, Martin. *Algoritmus pro realistickou tvorbu cévního řečiště syntetických sítnic*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Lukáš Semerád

Algoritmus pro realistickou tvorbu cévního řečiště syntetických sítnic

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Lukáše Semeráda. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Martin Dvořák
15. května 2019

Poděkování

Rád bych poděkoval Ing. Lukáši Semerádovi za cenné rady a za jeho vstřícnost při konzultacích.

Obsah

| | | |
|----------|---|-----------|
| 1 | Úvod | 3 |
| 2 | Popis a anatomie lidského oka | 4 |
| 2.1 | Anatomie lidského oka | 4 |
| 3 | Sítnice oka | 6 |
| 3.1 | Anatomické složení sítnice oka | 6 |
| 3.2 | Získání snímků a vyšetření sítnice oka | 9 |
| 4 | Popis existujícího nástroje | 11 |
| 4.1 | Základní myšlenka generátoru | 12 |
| 4.2 | Generované vrstvy | 12 |
| 4.3 | Vrstva pozadí | 14 |
| 4.4 | Vrstva terče zrakového nervu | 15 |
| 4.5 | Vrstva cévního řečiště | 15 |
| 4.6 | Vrstva rámečku snímku | 20 |
| 4.7 | Implementační detaily | 20 |
| 5 | Identifikace tvarů makulárních a nasálních cév krevního řečiště sítnic | 21 |
| 5.1 | Detekce směru a tvaru horních i spodních makulárních cév | 22 |
| 5.2 | Aproximace detekovaných tvarů makulárních cév nelineárními modely | 22 |
| 5.3 | Detekce směru a tvaru horních a spodních nasálních cév | 28 |
| 5.4 | Aproximace detekovaných tvarů nasálních cév nelineárními modely | 28 |
| 5.5 | Shrnutí dosažených výsledků | 31 |
| 6 | Identifikace tvarů a větvení cév vycházejících z makulárních i nasálních cév | 32 |
| 6.1 | Detekce směru a tvaru cév vycházejících z makulárních a nasálních cév | 33 |
| 6.2 | Realizace cév pomocí vektorů | 34 |
| 6.3 | Měření velikostí úhlů v bodech větvení cév | 35 |
| 6.4 | Určení hloubky větvení cév | 36 |
| 6.5 | Aproximace pozic větvení z cév předepsaných matematickou funkcí | 37 |
| 6.6 | Shrnutí dosažených výsledků | 38 |
| 7 | Algoritmus pro tvorbu cévního řečiště syntetických sítnic | 39 |
| 7.1 | Vstupní a výstupní parametry algoritmu | 39 |
| 7.2 | Obecný popis algoritmu | 41 |
| 7.3 | Detailní popis jednotlivých částí algoritmu | 41 |

| | |
|--|-----------|
| 8 Implementace algoritmu | 45 |
| 8.1 Použití konkrétních tříd k jednotlivým bodům navrhnutého algoritmu . . . | 45 |
| 8.2 Zakomponování nového algoritmu do stávajícího programu | 48 |
| 9 Testování větvení algoritmu | 49 |
| 10 Závěr | 51 |
| Literatura | 52 |

Kapitola 1

Úvod

Syntetické snímky sítnice oka můžou značně pomoci biometrii. Jelikož není pořízení reálných snímků příliš snadné jako u jiných biometrických metod, můžou syntetické sítnice značně doplnit tyto databáze. Následně mohou syntetické obrazy pomáhat v rozvoji nástrojů pracujících nad sítnicí oka.

Cílem této práce je zdokonalit systém generování syntetických sítnic oka, který používá výzkumná skupina STRaDe@FIT Vysokého učení technického v Brně. Nástroj byl vytvořen jako diplomová práce. Vytvořený generátor z části reflektuje korektní chování reálných snímků sítnic oka. V jednotlivých snímcích se nachází příliš mnoho cév i bodů větvení. Při agregaci bodů větvení získaných z vygenerované databáze se ve výsledku objevují aliasy a stejné vzory chování.

Zdokonalení generátoru proběhne na základě pozorování vzorů chování jednotlivých cév v nasnímané části sítnice oka. Jednotlivé vzory budou popsány matematickými modely. Na základě stanovených modelů bude navržen algoritmus, který bude reflektovat vypočítané vlastnosti cév. Algoritmus bude implementován do existujícího nástroje, aby nebyla změněna uživatelská přívětivost.

První dvě kapitoly popisují teoretický úvod do problematiky sítnice oka. Nejdříve je biologicky popsáno celé oko. Druhá kapitola se věnuje anatomii sítnice oka, popisuje složení i rozložení jednotlivých částí sítnice. Dále je popsáno získávání snímků krevního řečiště. Čtvrtá kapitola uzavírá teoretickou část a popisuje existující nástroj na generování sítnic oka. Ten byl popsán obecně pro jednotlivé vrstvy, ze kterých se skládá. Zvýšená pozornost byla věnována vrstvě, které vytváří krevní řečiště. Kapitola pět a šest se zabývá identifikací tvaru cévního řečiště. V úvodní části je popsán způsob detekce těchto cév. Následně popisuje zpracování zdetekovaných cév, prezentaci výsledků dosažených zpracováním i ověření správnosti výsledků. Předposlední kapitola se zabývá návrhem algoritmu pro tvorbu cévního řečiště. V úvodní části kapitoly je popsáno vstupní a výstupní rozhraní, které bude muset algoritmus dodržet. Dále je popsána hlavní myšlenka algoritmu v několika bodech. Závěr části se věnuje detailnímu popisu jednotlivých bodů algoritmu. Kapitola osm popisuje implementační detaily. V poslední kapitole je provedeno testování nové databáze snímků.

Kapitola 2

Popis a anatomie lidského oka

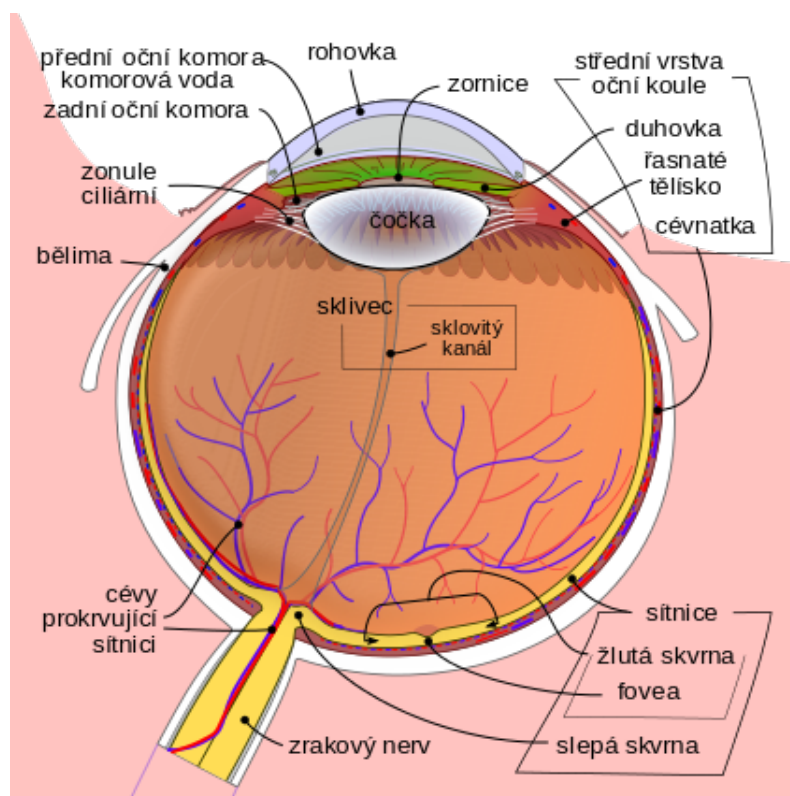
Zrak je jeden z našich hlavních smyslů. Pomocí oka přijímáme až 80 % vjemů. Tento systém je tvořen velkým množstvím částí, které spolu musí komunikovat. V podkapitolách 2.1 a 3.1 budou popsány některé důležité části oka. Zrak zajišťuje oční koule. Skládá se z menší přední polokoule *rohovky* a větší zadní polokoule nazvané *bělma*. Oční koule má předozadní rozměr přibližně 25 mm a svislý rozměr 23,5 mm. [13] [32]

2.1 Anatomie lidského oka

Oko můžeme rozdělit do tří hlavních vrstev. První vrstva je externí vrstva, kterou tvoří *bělmo* a *rohovka*. Druhou prostřední vrstvu můžeme rozdělit na dvě části. Na přední část tvořenou *duhovkou* a *řasnatým tělískem* a na zadní část, kterou tvoří *cévnatka*. Poslední vnitřní vrstvu, která tvoří přes 70 % celkového objemu oka, představuje *sítnice* [17]. Některé části budou popsány níže v následujícím seznamu a jejich umístění v rámci oka zobrazuje obrázek 2.1. [11] [32]

- *Bělmo* (sclera) se nachází ve viditelné části oka a má bílou barvu. Vytváří pomocnou stěnu oční koule a stará se o její stabilitu. Je složeno z kolagenu a elastických vláken. [17] [32]
- *Cévnatka* (choroidea) se nachází mezi sítnicí a bělimou. Později přechází v řasnaté tělísko. Je velmi tenká, nabývá šířky mezi 0,2 mm a 0,4 mm. Obsahuje velké množství cév a pigmentových buněk. Cévnatka zajišťuje vyživování některých částí sítnice. Pomáhá udržet klidové zaostření oka na dálku pomocí napínání řasnatého tělíška. [36]
- *Čočka* (lens) je zavěšena na řasnatém tělísku, které ovlivňuje napětí a tvar čočky. Změna tvaru čočky mění ohniskovou vzdálenost. Pokud oko zaostřuje na blízký objekt, je čočka vyklenuta. Při zaostření na vzdálený objekt sval ochabne a čočka se oploští. [8] [17] [25]
- *Duhovka* (iris) je zbarvený kruhovitý sval, který udává barvu očí. Duhovka vytváří stejné podmínky jako clona ve fotografickém přístroji. Pomocí svalových vláken je roztahována, a tedy propustí více světla dovnitř oka, nebo naopak je stažena a sníží tím značně propuštěné světlo, které dopadá na *tyčinky* a *čípky* (viz podkapitola 3.1). V centru duhovky je okrouhlý otvor *zornice*. [11] [17] [25] [26]

- *Rohovku* (cornea) tvoří průhledná vazivová tkáň. Spolu s čočkou zapříčiňuje lom světelných paprsků do vnitřní části oka. Rohovku chrání slzné žlázy neustálým tvořením slz. Slzy brání vysychání rohovky, vyplavují dráždivé částice a také vyrovnávají nerovnosti na ní. Přebytečné slzy jsou odváděny slznými kanálky. [8] [25]
- *Řasnaté tělísko* (corpus ciliare) je složeno z řídkého vaziva, které obsahuje mnoho pigmentu a také mnoho cév. [26]
- *Sklivec* (corpus vitreum) je rosolovitá a čirá hmota, která vyplňuje většinu vnitřní části oka. Nenacházejí se v ní cévy ani nervy. Tvoří ji síť propletených vláken s tekutinou, která je bohatá na bílkoviny. [26] [32]
- *Sítnice* (retina) zobrazuje viděný obraz skrz fotoreceptory. Ty můžeme rozdělit na tyčinky a čípky. Blíže je popsána v podkapitole 3.1.
- *Zornice* (pupila) se nachází uprostřed duhovky a reguluje spolu s duhovkou množství světla, které dopadá na sítnici oka. Zornice působí černou barvou, jelikož světlo, které vstupuje do oka zornicí, je absorbováno. [4] [17]



Obrázek 2.1: Anatomický popis lidského oka [23].

Kapitola 3

Sítnice oka

Sítnice se nachází na zadní stěně oční bulvy. Je považována za jedinou část *centrálního nervového systému*, kterou lze pozorovat bez invazivního přístupu. Sítnice pokrývá dvě třetiny vnitřní plochy bulvy, pokud se zanedbá místo, kde se nachází zrakový nerv. Světlo, které dopadá do oka, je čočkou zaostřováno na místo, kde se nachází sítnice. Zde je sestaven obraz viděného objektu pomocí *tyčinek* a *čípků*, který je otočen a převrácen. Převrácení a otočení obrazu je provedeno až v mozkové části, kde je zpracován tento vjem. [16] [20] [25]

3.1 Anatomické složení sítnice oka

Sítnice má velice složité uspořádání. Rozprostírá se mezi sklivcem a střední vrstvou oční koule. Lze jí rozdělit na dvě části. Zadní část, která je větší a obsahuje světlocitlivé buňky. Přední část, která je menší, neobsahuje světlocitlivé buňky. Nazývá se slepou částí sítnice. [26]

Vznikla jako výchlipka z mezimozku. Byla z počátku sférická, postupně se přetvořila do tvaru očního pohárku, který se tím následně stal dvojvrstevným. První vnější vrstva se přeměnila v *pigmentový epitel* sítnice. Ve druhé vnitřní vrstvě vznikly světlocitlivé a nervové složky.

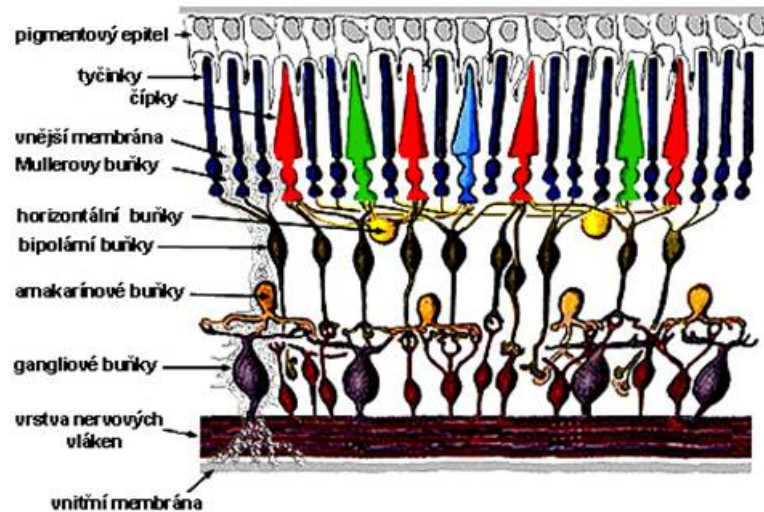
Sítnice má své světlocitlivé prvky, uspořádané v následujících vrstvách 3.1. Nejzevnější vrstva obsahuje *světlocitlivé* elementy, které jsou přiloženy k pigmentovému epitelu. Elementy jsou nervové a nazývají se první neurony sítnice. Vrstva *bipolárních nervových buněk* představuje druhé neurony sítnice. Vrstva se nachází uprostřed tloušťky sítnice a je to její střední vrstva. Třetí neurony sítnice představuje vrstva *gangliových buněk*, nachází se nejbližší nitroočnímu povrchu. Všechny uvedené vrstvy neuronů se spojují ve vrstvách *synapsí* a vytvářejí tím jedenácti vrstvou strukturu sítnice. [36]

Na sítnici se nalézají dvě nejvýraznější části: *makula* 3.3 (žlutá skvrna) a *optický disk* (slepá skvrna) [34]. V následující části budou blíže popsány jedny z nejdůležitějších částí sítnice oka.

3.1.1 Čípky a tyčinky

Jsou to světlocitlivé elementy sítnice. Výživa a kyslík jsou jim dodávány pomocí naléhajícího pigmentového epitelu. V celé sítnici existuje přibližně 6 až 7 miliónů čípků a 120 miliónů tyčinek. Podobné vlastnosti by měl 157 Mpx fotoaparát. [8] [36]

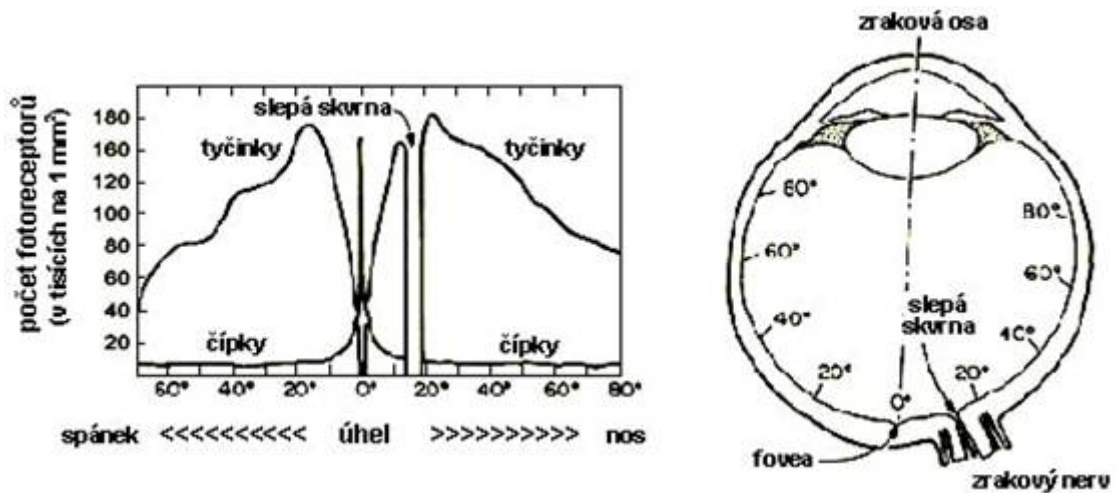
Jsou v sítnici rozloženy nerovnoměrně. V *makule* se nacházejí výhradně čípky. Jejich hustota rapidně klesá se vzdáleností od oblasti žluté skvrny. Tyčinek lze nalézt nejvíce



Obrázek 3.1: Struktura sítnice [21]

ve vzdálenosti 20° od oblasti *fovea*. *Fovea* je místem nejostřejšího vidění. Lze říci, že je zrakovým centrem oka. Ve vzdálenosti 15° od *fovea* se nachází *slepá skvrna*, která neobsahuje žádné světlocitlivé buňky. Demonstruje obrázek 3.2. [25] [36]

V těchto buňkách dochází ke kaskádám chemických i elektrických jevů, které se mění na elektrické impulzy. Tento jev se nazývá *transdukce*. Elektrické impulzy jsou vysílány skrze zrakový nerv do centrálního nervového systému. [8]



Obrázek 3.2: Histogram počtu tyčinek a čípků levého oka [22].

Tyčinky

Ve světlocitlivém segmentu jsou uloženy ploché puchýřky v počtu 600 až 1 000, které jsou tvořeny jemnými membránami. Obsahují zrakové barvivo, to vlivem dopadu světelného paprsku bledne a tvoří nervové vzruchy. Ty vycházejí *neuritem* světlocitlivé buňky. Ten je propojen synapsemi s bipolárními buňkami. [36]

Tyčinky umožňují rozeznávání tvarů předmětů při špatném vidění. Poskytují černobílé vidění [25] [34].

Čípky

Mají podobnou stavbu jako tyčinky, přesto se liší tvarem. Jsou kuželovitého tvaru, kratší a silnější než tyčinky. Jejich světlocitlivý segment má membrány podobné plochým puchýřkům tyčinek. Barvivo, které obsahují čípky při dopadu světla, vyvolává vzruchy, které procházejí *neuritem* čípku. *Neurit* je obdobně spojen synapsemi s bipolárními buňkami. [36]

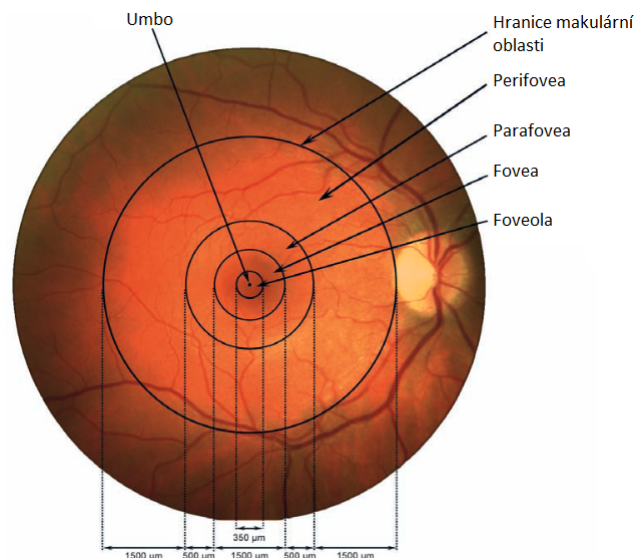
Zajišťují lepší ostrost než tyčinky. Čípky obsahují tři typy svého barviva. Jednotlivé typy jsou různě citlivé na světlo o různých vlnových délkách. Vzniká diferencovaná citlivost k barvám, z kombinací těchto barev je složen výsledný barevný obraz. Jednotlivá barviva jsou nejvíce citlivá na vlnové délky červené, zelené a modré barvy. [8] [36]

3.1.2 Makula (Žlutá skvrna)

Je označována jako nejostřejší místo vidění o průměru přibližně 5 mm. Skládá se převážně z čípků, které jsou protažené a více se podobají tyčinkám. Neobsahuje větší cévní větvení. Kvůli velké koncentraci malých cév má červenější barvu než své okolí. Za žlutou skvrnu je označována, protože makula se zbarví do žluté barvy po smrti člověka. Střed makuly je označován jako *fovea*, v jejímž středu probíhá zorná osa oka. *Fovea* je mírně prohloubená a tvoří jamku. Tato nerovnost umožňuje vměstnání více čípků do oblasti *fovey*. [16] [36]

3.1.3 Optický disk (Slepá skvrna)

Je místo, kam se sbíhají nervová vlákna jdoucí z *ganglových* buněk 3.1. Okrouhlý útvar o průměru přibližně 1,5 mm. Slepá skvrna je zcela zahlcena nervovými vlákny. Oblast je zbarvena do bílé barvy a u lehce vystouplého otvoru je jamka, kterou na sítnici vstupují cévy. Jelikož v oblasti optického disku nejsou světlocitlivé elementy, označuje se toto místo jako slepá skvrna. Jestliže obraz dopadne na místo slepé skvrny, nebude pro mozek viditelný. Mozek v těchto případech dotváří obraz podle zkušeností a znalostí, aby slepé místo vyplnil. Optický disk je na očním pozadí jediným místem, kde je sítnice pevně spojena s hlubšími vrstvami. [16] [34] [36]



Obrázek 3.3: Anatomické zobrazení sítnice oka [30].

3.2 Získání snímků a vyšetření sítnice oka

Pozorování nebo snímání zadní části oka se provádí několika technikami a přístroji. Základní techniky jsou *oftalmoskopie* a *fundoskopie*. Lze je použít i pro orientační měření.

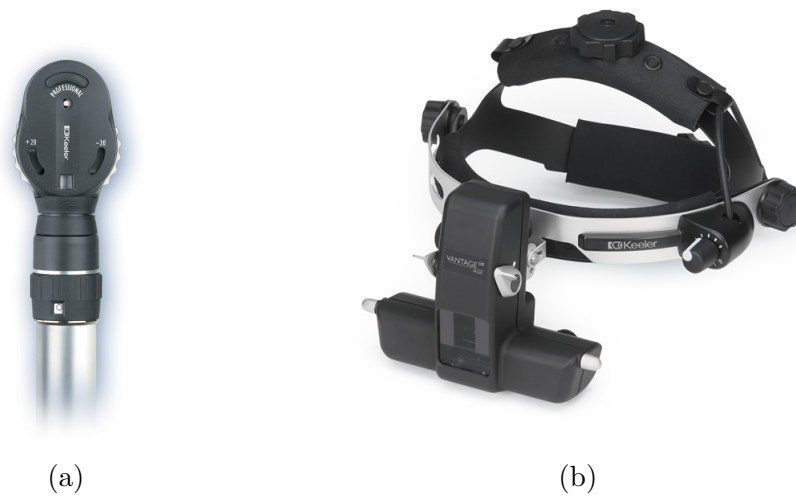
Oftalmoskopy jsou ruční přístroje, které vyšetřují oční pozadí. Zdroj světla se nachází v rukojeti přístroje. Ze zdroje jsou světelné paprsky přes polopropustné zrcadlo do oka vyšetřovaného. Hlava zařízení obsahuje otvor, který je nad osvětlovacím paprskem, sloužící vyšetřujícímu k pozorování sítnice oka. Lze vyměnit i zaostřovací čočku, která umožňuje vidět oční pozadí dostatečně ostře. *Oftalmoskopii* lze provádět dvěma způsoby. [1]

3.2.1 Vyšetření pomocí přímého oftalmoskopu

Vyšetřovací vzdálenost se pohybuje od obvyklých 3 cm do 7 cm. Obraz očního pozadí je zvětšen 16krát. Zvětšení ovlivňuje krátkozrakost i dalekozrakost. Je zobrazeno 5° až 7° očního pozadí 3.2. Čočky slouží k zaostření obrazu nebo ke změně barvy osvětlení. Lze zvolit modré nebo zelené osvětlení, při zeleném osvětlení se cévy jeví jako černé. Obraz očního pozadí je přímý při pozorování. Přístroj lze vidět na obrázku 3.4. [1]

3.2.2 Vyšetření pomocí nepřímého oftalmoskopu

Tento druh vyšetření se od předchozího typu liší tím, že se používá oftalmoskopická spojná čočka. Ta zajišťuje zvětšený, ale stranově a výškově převrácený obraz očního pozadí. Vzdálenost vyšetření je mezi 25 cm a 40 cm. Výsledné zvětšení je menší než u předchozí varianty, dosahuje 4 až 6krát zvětšeného obrazu, ale za cenu většího zorného pole. Oproti přímé oftalmoskopii tato metoda neobsahuje nežádoucí odlesky. Přístroj lze vidět na obrázku 3.4. [1]



Obrázek 3.4: Přímý oftalmoskop [27] (a), nepřímý oftalmoskop [28] (b).

3.2.3 Získání snímku pomocí fundus kamery

Fundus kamera je určena pro fotodokumentaci sítnice. Lze tak zobrazit strukturu fundu a analyzovat možné odchylky od normálního stavu, případně sledovat změny u specifických onemocnění, jako je např. diabetická retinopatie, degenerace makuly [15] apod. Výsledky jsou uloženy v digitální formě. Lze docílit efektivnější diagnostiky na základě porovnání snímků v čase a sledovat tak vývoj nemoci. Osvětlení očního pozadí je provedeno pomocí bílého světla. Fundus kamera pracuje v různých režimech. Jednotlivé režimy slouží pro zvýraznění určité části očního pozadí. Zorné pole kamery bývá 20° až 50° , clonou lze nastavit jeho velikost. Pomocí panoramatického obrazu, pokud se klient dívá do jiných směrů, lze sestavit až 110° zorného pole. Lze použít i přídavné speciální nebo širokoúhlé čočky. Zvětšení očního pozadí se pohybuje mezi 10 až 30násobky. Příklad je uveden na obrázku 3.5. [1]



Obrázek 3.5: Fundus kamera [18].

Kapitola 4

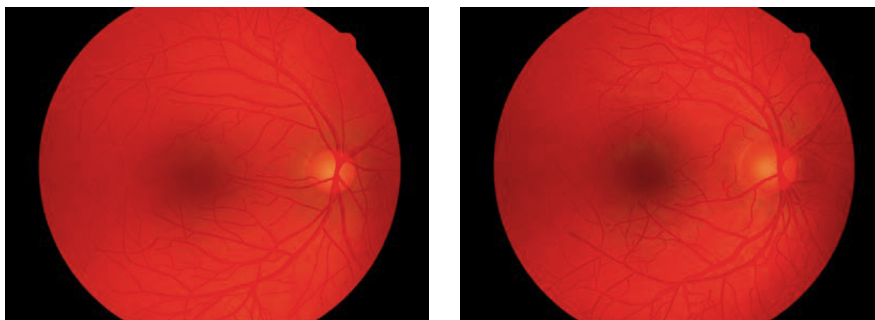
Popis existujícího nástroje

Tato kapitola popíše aktuálně existující nástroj pro tvorbu syntetických obrazů sítnic oka. První část této kapitoly představuje princip funkčnosti celého generátoru. Následují popisy jednotlivých částí. Větší důraz bude kladen na popis generování cévního řečiště sítnice, který se má zdokonalovat. Každá z níže popsaných částí generuje vrstvu výsledného obrazu. Závěr kapitoly ukazuje implementované třídy a další prostředky relevantní pro další část práce.

Nástroj byl vytvořen na fakultě informačních technologií VUT v Brně. Tento generátor má pomoci výzkumné skupině *STRaDe*, [29] při vytváření vlastní databáze syntetických snímků sítnic oka. Generátor vytvořil pan *Ing. Tretter* jako svoji diplomovou práci [30]. Nástroj je implementován v jazyce *C++* a využívá multiplatformní knihovnu *SFML*. Pokud nebude uvedeno jinak, zdrojem informací pro následující kapitolu je tato diplomová práce [30].

Představení jiného existujícího nástroje

Existují i jiné nástroje pro generování syntetických částí obrazů oka. Nástroj pro generování obrazů cévního řečiště sítnice [9] v roce 2014 nebo nástroj pro tvorbu bezcévní části sítnice oka [2] taktéž v roce 2014. Tyto části byly implementovány v softwaru *MATLAB* a dále rozšiřovány i zdokonalovány v projektu *VAMPIRE* (*Vascular Assessment and Measurement Platform for Images of the REtina*). Dosažené výsledky jsou zobrazeny na obrázku 4.1. [3] [31]



Obrázek 4.1: Syntetický snímek sítnice oka pořízený v projektu *VAMPIRE* pomocí dvou různých metod. [3]

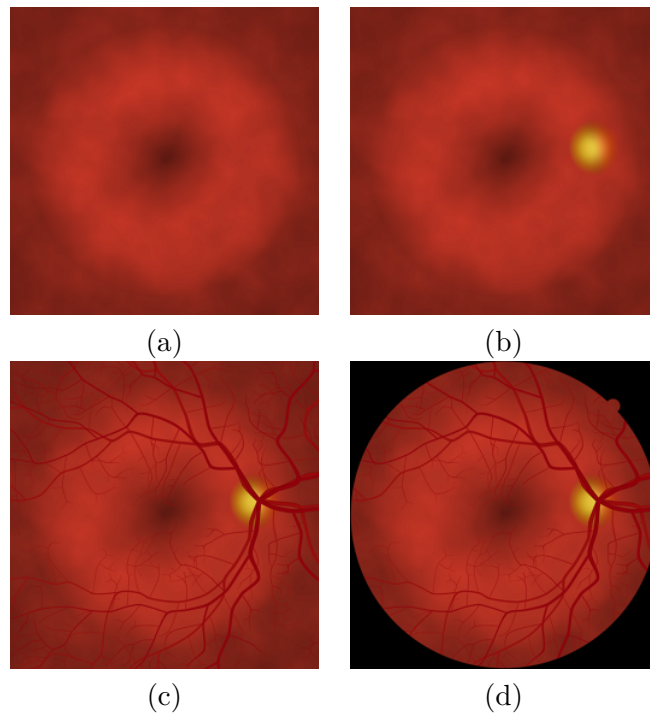
4.1 Základní myšlenka generátoru

Program tvoří syntetické obrazy sítnic oka na základě specifikovaných parametrů, které mu jsou předem zadány. Mezi implementované parametry patří počet takto vytvořených obrazů, rozměry snímku, přiblížení částí snímku a informací o tom, zda se bude generovat pravé nebo levé oko (viz níže).

Prvním krokem při tvorbě snímku je založení vykreslovaného okna o rozlišení, které má splňovat výsledný syntetický snímek. Do tohoto okna jsou postupně vykreslovány další vzájemně se překrývající vrstvy generátoru. Ty jsou generovány samostatně a v posledním kroku překryjí výsledný obraz. Jakmile je dogeneratedována poslední vrstva, je snímek uložen na předem definované místo parametrem programu.

4.2 Generované vrstvy

Reálný snímek sítnice lze rozdělit na 4 různé části, které spolu přímo nejsou spojeny a tedy lze na nich pracovat samostatně. Jednotlivé vrstvy jsou znázorněny na obrázku 4.2. Výsledný syntetický snímek vznikne sloučením jednotlivých částí. Nejspodnější vrstva obsahuje texturu pozadí sítnice oka. Přes ní je generována vrstva, která obsahuje texturu terče a znakového nervu. Následně tyto části překrývá vrstva cévního řečiště a vše následně překrývá textura rámečku.



Obrázek 4.2: Postupné vykreslení jednotlivých vrstev. Vrstva textury pozadí (a), přidání vrstvy textury terče zrakového nervu (b), přidání vrstvy cévního řečiště (c), výsledný obraz s vrstvou rámečku (d).

Vrstva, na kterou se nanáší textura, je nejdříve předzpracována. Její rozměr je vynásoben zadaným parametrem změny měřítka snímku. Generovaná vrstva je větší než zadané

vykreslované okno, a proto musí dojít k vycentrování vrstvy na střed vykreslovacího okna a následně k oříznutí vrstvy na velikost okna. Generátor používá vlastní souřadnicový systém, do kterého mapuje jednotlivé pixely textury. Počátek tohoto souřadného systému je ve středu textury.

4.2.1 Transformace vrstev

Jednotlivé transformace byly navrženy tak, aby byly nezávislé na generování textury. Operace rotace a posunutí nemění pozici vrstvy, ale upravuje souřadný systém. Naopak změna měřítka neupravuje souřadnicový systém, ale mění velikost vrstvy.

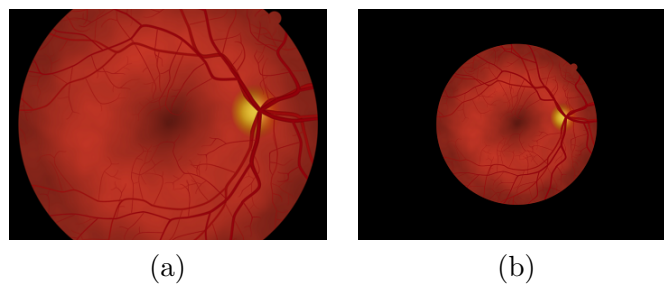
Rotace a posunutí vrstev

Nástroj implementuje tyto transformace z důvodů, které vycházejí z pořizování reálných sítnic oka. Nástroj tedy simuluje možnost, že pořizovaný snímek sítnice oka je mírně či více natočený. K natočení může dojít až už úmyslně při vyšetření konkrétní části sítnice nebo při špatné manipulaci s přístrojem na pořizování snímků.

Obě uvedené rotace se aplikují pouze na souřadný systém, a to v pořadí posunutí a následně rotace. U libovolné vrstvy je možné nastavit individuální hodnoty rotace a posunutí, přičemž tyto transformace ovlivní i další vrstvy, které leží nad danou vrstvou, kde se aplikují transformace.

Změny velikostí snímku

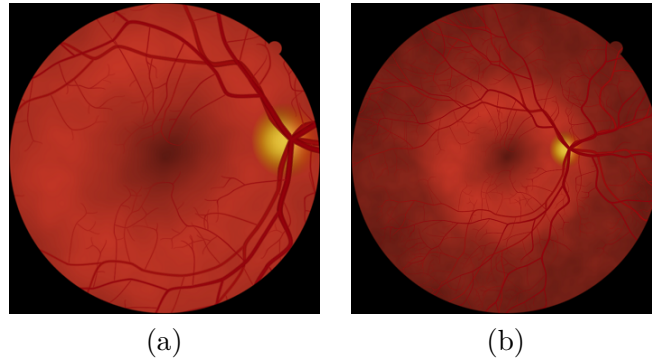
Na některých reálných snímcích nemusí zobrazená sítnice být na celém snímku, nebo naopak se obraz sítnice nemusí celý vměstnat do zobrazeného snímku. Generátor implementuje i tuto možnost a nabízí přiblížení či oddálení snímku. Na obrázku 4.3 lze vidět největší možné a nejmenší možné přiblížení snímku. Program tak mění velikost výsledné vrstvy a nedojde ke snížení kvality výsledného snímku.



Obrázek 4.3: Na prvním snímku (a) je zobrazeno největší možné přiblížení nástroje, naopak druhý snímek (b) zobrazuje největší možné oddálení výsledného obrazu.

Změny velikosti obrazového úhlu snímku

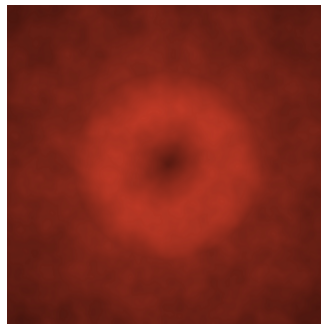
U snímání reálných sítnic je možné měnit velikost úhlu snímání fundus kamery, a tím měnit výsledný snímek sítnice. Tento případ je realizován pomocí parametru, který specifikuje přiblížení snímku sítnice pod rámečkem. Výsledné snímky jsou zobrazeny na obrázku 4.4.



Obrázek 4.4: Na prvním snímku (a) je zobrazen nejmenší možný snímaný úhel snímku sítnice, naopak druhý snímek (b) zobrazuje největší možný snímaný úhel obrazu sítnice.

4.3 Vrstva pozadí

Vrstva pozadí je tvořena načervenalou barvou, která tmavne směrem k exmakulární periferii a v oblasti *fovey*. Pozadí je tvořeno základní funkcí a funkcí šumu, která dodává snímku realističtější vzhled. Základní funkce je rozdělena na dvě části, kde první popisuje podobu textury v makulární oblasti. Pro její nevhodné vlastnosti ve vzdálenějších místech snímku od jeho středu je funkce nahrazena druhou funkcí. Ta simuluje postupné ztmavování pozadí s přibývajícím vzdáleností od středu snímku. Výsledné pozadí lze vidět na obrázku 4.5. Funkce šumu realizuje prosvítání cévnatky pomocí funkce *Perlinova šumu*.



Obrázek 4.5: Výsledná textura pozadí.

Perlinův šum

Je šumová funkce, kterou lze velice rychle vypočítat. Funkci představil roku 1985 *Ken Perlin*. Funkce je statisticky invariantní vůči posunutí a otáčení. Funkce je spojitá a je definovaná na omezeném frekvenčním spektru. Zaručením invariací nebude záležet, kde se začne taková textura mapovat a jak bude natočená, ale textura bude statisticky sama sobě podobná. Teoreticky tato funkce může produkovat nežádoucí alias. Tomuto jevu lze snadno zabránit pomocí přesností vzorkování. Výpočet dané funkce je velice rychlý, výsledné hodnoty jsou náhodné, ale tato nahodilost je však říditelná. [37]

4.4 Vrstva terče zrakového nervu

Terč zrakového nervu je elipsovité oblast zbarvená žlutou barvou. Oblast se zbarvuje až do oranžové barvy u hranice makulární oblasti. Tedy směrem k *fovee* přechází z oranžové přes žlutou do světle žluté barvy. Mimo oblast samotného terče zrakového nervu je vrstva průhledná. Výsledná podoba vrstvy je vidět na obrázku 4.6. Při umístění zrakového terče do vrstvy má terč pevně daný rozměr i místo umístění. Pro simulaci vytvoření různých rozměrů, zbarvení a umístění terče zrakového nervu jsou tyto předem definované hodnoty násobeny s náhodným koeficientem pro vytvoření proměnlivosti snímku.



Obrázek 4.6: Výsledná textura terče zrakového nervu.

Výsledná zbarvení se budou ještě lišit od tohoto obrázku, a to především díky částečné průhlednosti vrstvy. Barvu ovlivní i zbarvení vrstev, na které bude nanášena. U každého snímku bude výsledná pozice a tvar terče zrakového nervu mírně odlišná kvůli náhodnému mírnému posunutí a pootočení. Generováním snímku sítnice levého oka musí být provedena rotace této vrstvy o 180° .

4.5 Vrstva cévního řečiště

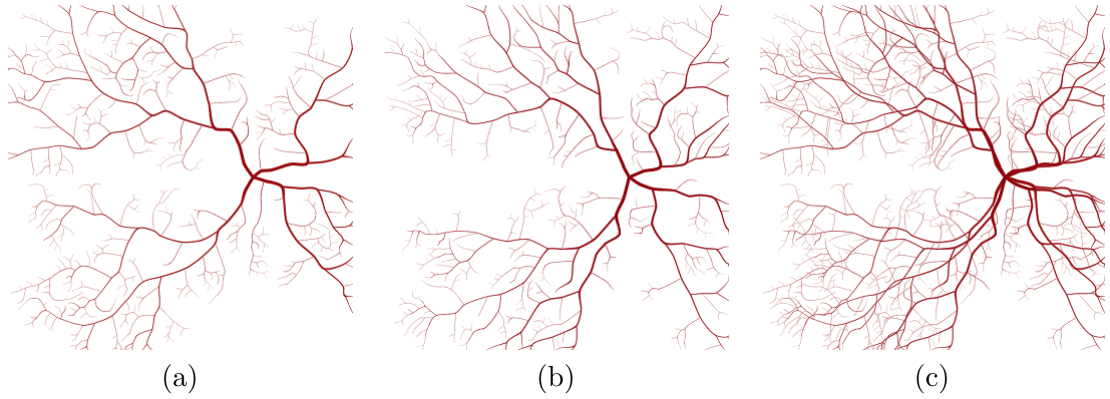
Řečiště sítnice můžeme rozdělit na žilní a tepenné. Obě tyto části se dělí na několik dalších částí podle funkčnosti dané cévy [13]. Nástroj generuje odděleně žilní a tepenné řečiště. Tento krok je nutný kvůli výběru způsobu vykreslování jednotlivých cév. Po vygenerování proběhne spojení žilního a tepenného řečiště.

Spojení těchto řečišť se provádí po pixelech a lze rozdělit na tři možné případy chování. Pokud se na daném místě nenachází céva, je vrstva v tomto místě ponechána průhledná. Jakmile se na sjednocovaném místě nachází jedna céva, tedy buď žilní nebo tepenní, je na výsledné vrstvě právě tato jedna céva zobrazena. Nejkomplikovanější možnost nastane, pokud na daném místě dojde ke sloučení obou cév. Potom je nutné danému místu přepočítat hodnoty barvy. Proces slučování řečišť je vyobrazen na obrázku 4.7.

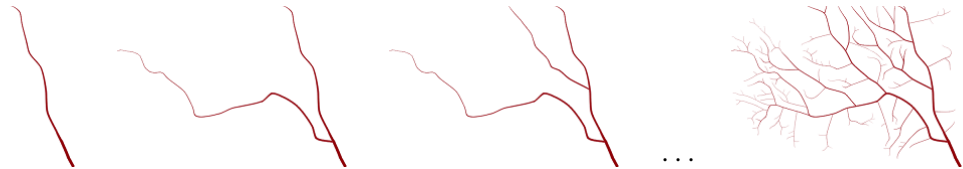
4.5.1 Způsob vykreslování větve cévního řečiště

Pro vytvoření realističtějšího snímku je nutné, aby mezi body větvení křivka neobsahovala výrazně ostré body zlomů. Křivka je spojnice mezi posloupnostmi bodů, nelze ji tudíž v jednom bodě rozdělit na více částí. Proto nástroj generuje strom větvení pro danou cévní větev postupně. Lze vidět na obrázku 4.8.

Toto generování probíhá tak, že se od počátečního bodu vykreslované větve řečiště vykresluje křivka, která prochází předem definovanými body větvení. V bodu větvení pokračuje tvoření křivky širší cévou k dalšímu bodu větvení. Jakmile se céva dogeneruje do jejího



Obrázek 4.7: Textura tepenného řečiště (a), textura žilního řečiště (b), výsledná textura cévního řečiště (c).



Obrázek 4.8: Postupné vykreslení horní temporální větve.

bodou ukončení, je vybrán další bod větvení, který se prohlásí jako počáteční bod křivky. Následuje generování této křivky. Výběr dalšího bodu ke generování ovlivňuje tloušťka cévy. Vždy je vybrán bod, který má aktuálně největší tloušťku. Tento vzorec se opakuje, dokud nejsou vygenerovány všechny cévy dané větve.

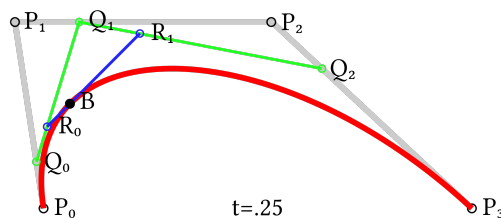
Při vykreslování se používá *kubická Bézierova křivka*. K určení umístění kontrolního bodu Bézierovy křivky je použita vzdálenost od bodu, ke kterému náleží. A ještě vypočítán úhel, který označuje směr, kterým bude kontrolní bod umístěn oproti původnímu bodu větvení.

Bézierova křivka

Jedna z nejpoužívanějších aproximačních křivek, která se používá jak pro modelování ve dvourozměrném prostoru, tak pro modelování třírozměrných objektů. Bézierovu křivku n -tého stupně definuje $n+1$ bodů P_i řídicího polygonu vztahem, který je uveden rovnicí 4.1. B_i^n jsou *Bersteinovy polynomy* n -tého stupně.

$$Q(t) = \sum_{i=0}^n P_i B_i^n(t) \quad (4.1)$$

Jednou z vlastností *Bézierovy křivky* je, že změnou jednoho řídicího bodu P_i je změněn tvar celé křivky. Tato vlastnost zdůvodňuje dělení těchto křivek na segmenty nižšího řádu, které na sebe navazují. Navázání segmentů se provede pomocí ztotožnění posledního řídicího bodu aktuální křivky a prvního řídicího bodu následující křivky. Nejčastěji se používají křivky třetího stupně, tedy *Bézierovy kubiky*, které definují čtyři řídicí body. [10] [37]



Obrázek 4.9: Vytvoření křivky pomocí Béziových křivek [5].

4.5.2 Způsob vykreslení cévy mezi body větvení

Céva je vykreslována postupně po dvojicích bodů větvení jdoucích za sebou. Pro takovou dvojici s příslušnými kontrolními body jsou poté vypočítány dílčí body křivky. Velikost kroku k při výpočtu bodů *Béziové křivky* je stanovena vztahem 4.2, kde neznámá d určuje vzdálenost mezi počátečním a koncovým bodem křivky.

Vypočítané body křivky jsou následně spojeny úsečkami. Jednotlivé úsečky jsou vykresleny pomocí *Bresenhamova algoritmu* 4.5.2. Poté je vykreslena textura kolem každé cévy. Zbarvení textury odpovídá typu cévy a její tloušťce.

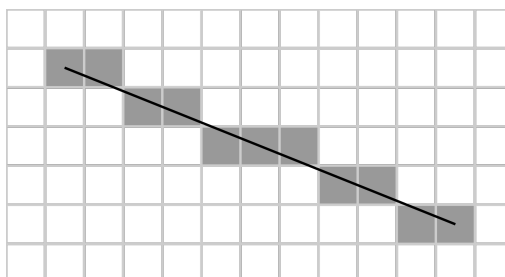
$$k = \frac{1}{2d} \quad (4.2)$$

Bresenhamův algoritmus

Patří mezi nejrozšířenější a nejpoužívanější algoritmus pro rasterizaci úsečky. Algoritmus používá pouze celočíselnou aritmetiku v podobě odečítání, sčítání a porovnávání. Tyto operace jsou velice rychlé a efektivní v procesorech.

Vykreslení konkrétní úsečky není přesné, je pouze aproximované. Je počítáno se směrnici křivky, která je vyjádřena pomocí dvou přirozených čísel ve tvaru zlomku. Postup bude vysvětlen pomocí obrázku 4.10. Nejprve se vykreslí koncový bod úsečky. Následně je nutné spočítat, který bod bude vykreslen o jeden pixel vpravo. Zda bude vykreslen bod se stejnou hodnotou souřadnice Y nebo o jednu pozici níže. Pixel bude vybrán na základě toho, jaká z možností je nejbližší reálné křivce. Toto rozhodnutí provádí *rozhodovací člen*, který je vyjádřen v rovnici 4.3. Po dosazení do této rovnice zjistíme, zda je výsledek kladný, a je změněna souřadnice Y. [10] [37]

$$p_{i+1} = p_i + 2\Delta y - 2\Delta x(y_{i+1} - y_i) \quad (4.3)$$



Obrázek 4.10: Vykreslení křivky pomocí Bresenhamova algoritmu [12].

4.5.3 Generování bodů větvení cévního řečiště

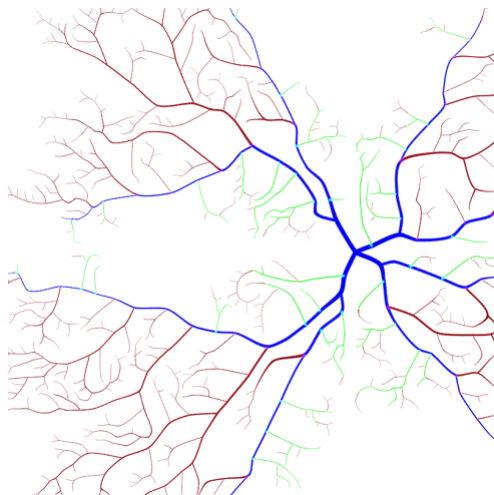
Nutnou prerekvizitou pro vygenerování krevního řečiště je mít vygenerované body větvení, a tedy mít nastavené hodnoty atributů těchto bodů. Body větvení jsou tvořeny samostatně pro každou z hlavních větví tepenného i žilního řečiště. Na generování jednotlivých větví je použit stejný algoritmus, pouze s rozdílnými hodnotami u některých parametrů. Algoritmus rozděluje tvorbu bodů větvení na dvě části. V první části se vytvoří body větvení, které jsou uloženy ve stromové struktuře. Ve druhé části jsou vypočítány pozice jednotlivých bodů větvení s ohledem na pozice rodičovských bodů větvení.

Bod větvení se skládá tedy z několika atributů, které je nutné spočítat.

- pozice bodu
- vzdálenost od předchozího bodu větvení
- šířka cévy
- typ bodu větvení
- typ cévy

Bod větvení, který představuje kořen stromové struktury, se nachází ve středu terče zrakového nervu. Generování bodu do stromu probíhá tak, že každý nově vzniklý bod větvení generuje další dva body větvení. Rekurse generování končí, když atribut *šířka cévy* nabývá hodnoty menší nebo rovno 0. Pokud je šířka cévy záporná, je přepočítána její délka tak, aby byla šířka cévy rovna 0. Následně některé informace vycházejí z publikace [35].

Typ cévy byl specifikován do tří kategorií. Typy demonstruje obrázek 4.11, na kterém jsou vyobrazeny všechny definované typy cév. Hlavní, tedy nejsilnější cévy, jsou zobrazeny modře. Slabé cévy, vycházející z hlavních cév, jsou zobrazeny zeleně. Poslední typ cév je zobrazen červeně a ukazuje všechny ostatní cévy.



Obrázek 4.11: Barevné znázornění typů cév.

Výpočet vzdálenosti od předchozího bodu větvení

Výpočet vzdálenosti se provádí podle funkce 4.4, která má rozdílné chování v závislosti na šířce cévy w_a . Tímto je zaručeno častější větvení slabších cév.

$$d = \begin{cases} \text{rand}(0,15; 0,05) & \text{pro } w_a > 0,15 \\ \text{rand}(0,05; 0,02) & \text{jinak} \end{cases} \quad (4.4)$$

Výpočet typu bodů větvení

Typy bodů větvení mohou nabývat čtyř různých hodnot. Určení konkrétního typu větvení v bodě se odvíjí od jeho pozice na snímku a od typu větvení, které bylo nastaveno na předcích daného bodu větvení.

- První větvení typu Y rozděluje cévu na dvě téměř stejně široké cévy. Menší cévy dále směřují mírně od sebe, a tím vytvářejí obrazec písmena Y .
- Druhý typ T větvení má podobu písmene T . Větvení zde rozdělí cévu na širokou a úzkou.
- Dalším typem je žádné větvení. Céva pouze prochází takto definovaným bodem.
- Posledním typem větvení je ukončení cévy. Místo, kde céva nabývá šířky 0 a zaniká.

Výpočet šířky cévy řečiště

V závislosti na typu větvení je vybrán poměr, kterým se bude céva dělit. V místě dělení nedochází k zúžení cévy, poměr je vyjádřen pomocí vzorce 4.5. V případě větvení typu Y se parametr r pohybuje v hodnotách 0,45 až 0,55. V případě větvení typu T parametr r nabývá hodnot z intervalu 0,01 až 0,05

$$r : (1 - r) \quad (4.5)$$

Hodnota šířky cévy je vypočtena vzorcem 4.6, kde w_p představuje šířku cévy v předchozím bodě. Dříve získaná hodnota r určuje procentuální šířku jedné cévy při jejím větvení. Symbol d zastupuje vzdálenost mezi aktuálním bodem a jeho předchůdcem.

$$w = w_p \sqrt{r} - w_p \frac{d}{10} - \frac{d}{20} \quad (4.6)$$

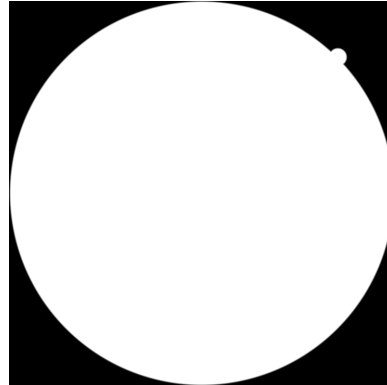
Výpočet pozic bodů větvení

V první řadě jsou vypočteny pozice bodů větvení pro cévy typu 1. Následuje generování bodů pro cévy typu 2, které vycházejí z bodů cév typu 1. Jako poslední se vytvářejí body typu 3. Pořadí generování je především dáno šířkou cévy v daném bodě. Nejprve se tvoří cévy širšího charakteru, a to z důvodu, že ne všechny body musí být následně použity. Pokud je v jednom bodě větvení již šířka cévy nulová, dojde k odstranění bodů, které jsou synovskými uzly s aktuálním bodem větvení.

Výpočet ovlivňuje množina pozic, ve kterých se bod může nacházet. Tuto množinu dále zúží směr vykreslované cévy, do které má daný bod patřit. Pro každou ze zbylých pozic je na základě odchylky od středu intervalu spočítána pravděpodobnost pro umístění tohoto bodu větvení.

4.6 Vrstva rámečku snímku

Na pozorovaných snímcích je sítnice tvarována do kruhu a ostatní prostor, který je na snímku dorovnán do čtvercové nebo obdélníkové části, obsahuje černou barvu. Na některých snímcích lze vidět ještě část sítnice za tímto kruhem. Zobrazení této části sítnice poskytuje informaci čtenáři o správném čtení snímku. Obrázek 4.12 ukazuje výslednou texturu rámečku, aby nedošlo například k otočení snímku a k jeho špatné interpretaci. Nástroj v posledním kroku překryje výsledný snímek tímto rámečkem. Nastavením určitého parametru lze pozměnit umístění značky na snímku sítnice.



Obrázek 4.12: Vrstva textury rámečku s půlkruhovou značkou v prvním kvadrantu.

4.7 Implementační detaily

Nástroj je implementován v jazyce *C++* a využívá externí knihovny *SFML* [24]. Aplikace představuje konzolovou aplikaci, kde se nastavení parametrů snímků provádí přes konfigurační soubor.

Aplikace ihned při spuštění načte konfigurační soubor pro vytváření snímků. Na základě této konfigurace vytváří instance tříd. Budou zde zmíněny pouze ty, které bude nutné upravit či znovu implementovat. Následující třídy jsou důležité z hlediska zdokonalení algoritmu pro tvorbu krevního řečiště: `vascularityLayer`, `vesel` a `bifurcationPoint`. Zmíněné třídy budou stručně popsány.

Stěžejní třídy pro tvorbu cévního řečiště

Třída `vascularityLayer` – tvoří vrstvu krevního řečiště. Pro každou hlavní větev tepenného i žilního řečiště je ihned vytvořena instance třídy `vessel`. Hlavní větve žilního a tepenného řečiště jsou nejprve vykresleny do dvou dílčích textur funkcí `drawVessel()` a následně je provedeno sloučení těchto textur.

Třída `vessel` – tvoří hlavní větve krevního řečiště. Jako počáteční bod každé větve je použita instance třídy `bifurcaionPoint`. Funkce `genBifurcationPointCoord()` následně vypočítá pozici pro první větvení dané hlavní větve krevního řečiště.

Třída `bifurcationPoint` – popisuje konkrétní bod větvení. Na základě hodnot atributů při vzniku dané instance třídy jsou vypočítány vlastnosti následujících bodů větvení a vytvořeny instance třídy `bifurcationPoint`.

Kapitola 5

Identifikace tvarů makulárních a nasálních cév krevního řečiště sítnic

Tato kapitola se zabývá prací s existujícími snímky krevního řečiště. Ty byly získány pomocí databáze *Messidor*. [6] Data byla vybrána zcela náhodně, faktor stejné lokality snímků potlačuje projekt *Messidor* svojí definicí (viz níže). Na snímcích jsou sledovány vlastnosti potřebné k dalším částem práce. Jednotlivé znaky jsou změřeny, statisticky zpracovány a následně vyhodnoceny. Zaznamenávání pozic bodů pro regresní programy bylo provedeno ve webovém programu *Desmos*, který i následně prezentuje aproximační křivky. Naměřená data zpracovávají matematické softwary pro tvorbu nelineárních regresí. Pro zvýšení počtu modelů řešení jsou použity dva matematické programy *CurveExpert Basic* [14] a *RegressionTools* [33].

Databáze *Messidor* vznikla pro vědecké práce zabývající se počítačovou diagnostikou diabetické retinopatie. Obsahuje přes 1 200 barevných snímků sítnice oka, které byly pořízeny z následujících francouzských očních klinik: *Hôpital Lariboisière Paris*, *Faculté de Médecine St Étienne* a *Centre Hospitalier Régional et Universitaire de Brest*. Snímky byly pořízeny za pomoci barevného 3CCD fotoaparátu, který byl nainstalovaný na nemydriatickém retinografu se zorným polem 45° . Data jsou uložena v rozlišení 1440×960 , 2240×1488 nebo 2304×1536 ve stejném formátu *TIFF*. [6]

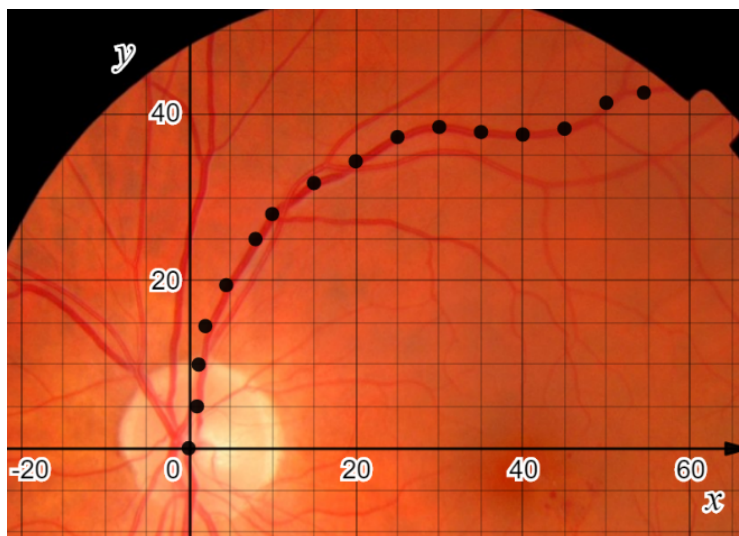
CurveExpert Basic je snadno použitelný program s grafickým rozhraním, kterým lze rychle a jednoduše spočítat lineární a nelineární aproximace křivek pomocí zabudovaných modelů pro aproximace. Nástroj umožňuje definovat vlastní modely. Program pomocí vlastní metriky pro určení nejlepší křivky ukáže uživateli nejvhodnější modely pro dané body. Spočítané aproximace lze zobrazit i v grafech a vizuálně je porovnat mezi sebou. [14]

Webový nástroj *RegressionTools* počítá lineární i nelineární regrese pomocí množiny primitivních funkcí. Tato množina obsahuje přes 100 funkcí. V závislosti na počtu vstupních bodů program skládá primitivní funkce a vytváří z nich komplexnější model. [33]

Aplikace *Desmos* existuje ve webovém provedení. Tento program poskytuje plátno se souřadným systémem, se kterým lze limitovaně komunikovat a vkládat do něj body a matematické křivky. Stěžejní vlastnost pro výběr tohoto softwaru je možnost vložení obrázku do pracovní plochy. Každý vytvořený objekt tvoří vlastní vrstvu. Tyto vrstvy lze překrývat. Tato vlastnost umožňuje přes vložení obrázku vykreslovat křivky a vkládat body. [7]

5.1 Detekce směru a tvaru horních i spodních makulárních cév

Z důvodu časově náročného zpracování jednotlivých snímků bylo zpracováno pouze 80 cév. Zpracování dat probíhalo v programu *Desmos*. Vložený snímek byl zvětšen na velikost 170×120 interních jednotek programu. Následně byl posunut tak, aby bod souřadného systému $[0; 0]$ sdílel stejnou pozici jako vstup cévy do sítnice ve slepé skvrně. Průběh cévy je zaznamenán skupinou bodů, která překrývá danou cévu v konkrétních místech. Vzdálenost mezi jednotlivými body skupiny se pohybuje okolo pěti jednotek. Úvodní zasazení snímku do prostředí, zvětšení, posunutí a proložení cévy je zobrazeno na snímku 5.1.



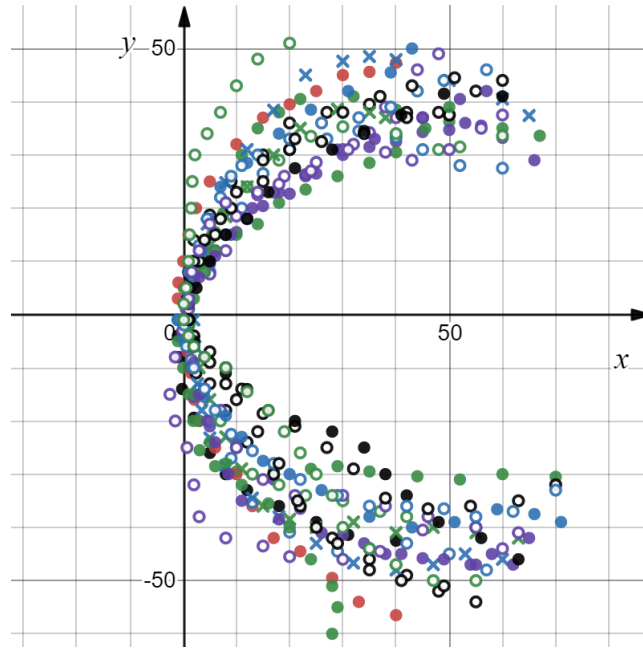
Obrázek 5.1: Snímek z databáze *Messidor* zasazený do prostředí *Desmos* a proložený body horní makulární žíly.

Volba velikosti snímku na 170×120 jednotek určuje definiční obor funkcí a zároveň i jejich obor hodnot. Tento koeficient odpovídá téměř desetinásobku snímku při jeho vložení. Důvodem pro zvětšení všech snímků je lepší práce s regresivními modely. Vybrané programy lépe pracují s intervalem $\langle 0; 170 \rangle$ než s původními hodnotami, které byly z intervalu $\langle 0; 17 \rangle$.

Zpracováním všech náhodně vybraných snímků se stanovil předpoklad, že chování makulárních cév půjde vyjádřit matematickou funkcí. Tento předpoklad bude ověřen v části 5.2. Dosažené výsledky dále vytvářejí předpoklad, že spodní makulární cévy jsou svým tvarem totožné s horními makulárními cévami. Tato vlastnost cév ověřuje také podkapitola 5.2. Výsledky zobrazuje obrázek 5.2.

5.2 Aproximace detekovaných tvarů makulárních cév nelineárními modely

V první části došlo k rozdělení dat na horní a spodní makulární cévy. Tyto dvě sady dat byly předány regresivním programům (*CurveExpert*, *RegressionTools*) pro výpočet aproximačních funkcí. Programy vytvořily seznam matematických funkcí, které seřadily podle



Obrázek 5.2: Zaznamenané horní i spodní makulární cévy.

interních metrik. Byly vybrány nejlépe ohodnocené modely, které dále splňovaly přirozený vývoj za hranicí posledního bodu.

5.2.1 Získané funkce horních makulárních cév

Níže popsané matematické předpisy zobrazují nejlépe hodnocené funkce. Získané parametry funkcí byly zaznamenány do přilehlých tabulek. Závěr části obsahuje grafické znázornění získaných funkcí 5.3.

Získané funkce představují pouze jednu konkrétní cévu, proto bylo nutné, aby jeden předem zvolený parametr funkce nabýval intervalu hodnot. Tyto intervaly byly určeny empiricky tak, aby pokrývaly aspoň 90 % naměřených bodů v dané sadě dat. Vybrané parametry jsou určeny v tabulce 5.4.

První uvedená funkce představuje *racionální model*, zobrazený předpisem 5.1. Hodnoty parametrů jsou zobrazeny v tabulce 5.1. Proměnná b reprezentuje předem definovaný parametr, který nabývá hodnot z intervalu uvedeného v tabulce 5.4. *Gompertzova funkce* 5.2 představuje další uvedený model, hodnoty reprezentuje tabulka 5.2. Obor hodnot intervalového parametru a reprezentuje taktéž tabulka 5.4. Model *druhé odmocniny* 5.3 představuje posledního zástupce této skupiny. Vypočítané parametry jsou uvedeny v tabulce 5.3. Tato funkce nenabízí možnost výběru intervalového parametru, proto je vybrán jediný možný. Interval hodnot zobrazuje tabulka 5.4.

$$f(x) = \frac{a + bx}{1 + cx + dx^2} \quad (5.1)$$

$$f(x) = ae^{-e^{-cx}} \quad (5.2)$$

$$f(x) = a\sqrt{x} \quad (5.3)$$

| parametr | hodnota parametru |
|----------|----------------------|
| a | 4,418165 |
| b | 2,839498 |
| c | 0,037768 |
| d | $3,901618 * 10^{-4}$ |

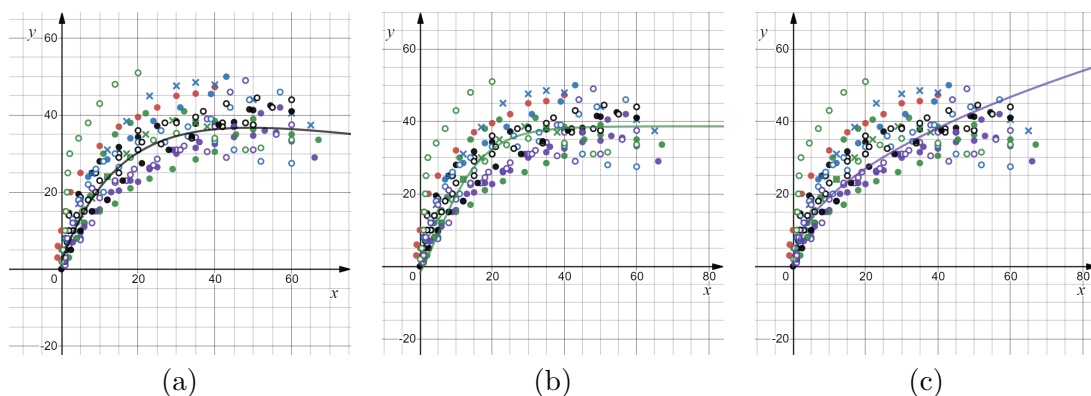
Tabulka 5.1: Vypočtené hodnoty parametrů pro lomenou funkci 5.1.

| parametr | hodnota parametru |
|----------|-------------------|
| a | 48,634152 |
| b | 0,5806228 |
| c | 0,1275869 |

Tabulka 5.2: Vypočtené hodnoty parametrů pro Gompertzovu funkci 5.2.

| parametr | hodnota parametru |
|----------|-------------------|
| a | 6,032641 |

Tabulka 5.3: Vypočtené hodnoty parametrů pro funkci druhé odmocniny 5.3.



Obrázek 5.3: Matematické funkce vykreslené v souřadném systému. Racionální model 5.1 (a), Gompertzova funkce 5.2 (b), funkce druhé odmocniny 5.3 (c), všechny funkce horní makulární tepénky (d).

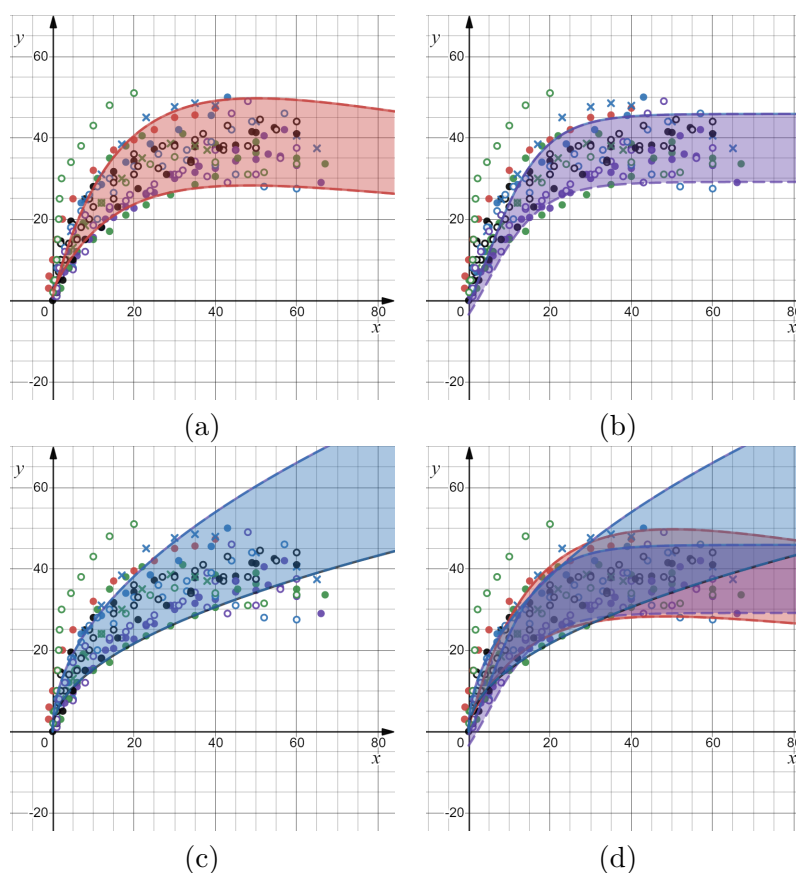
5.2.2 Korekce intervalových parametrů pro modely horních makulárních cév

Tabulka 5.4 popisuje rozpětí jednotlivých matematických modelů. Ty jsou znázorněny v grafech 5.4. Interval hodnot, který může nabývat daný parametr, byl nejdříve opticky nastaven, aby obsahoval největší možný počet bodů. V další fázi proběhla korekce odhadnutých hodnot. Graf d v 5.4 byl přikládán na další náhodné snímky reálných sítnic. To probíhalo způsobem uvedeným v podkapitole 5.1 (vstup cévy do snímku ve *slepé skvrně* odpovídá v souřadném systému bodu $[0;0]$). Ke korekci bylo použito 200 náhodně vybraných cév

z databáze *Messidor*. Parametry byly upravovány dokud se v oblasti nenacházelo alespoň 90 % cév. Tato hranice se jeví jako dostatečná pro další práci. Vytváří to prostor pro vytvoření dalšího matematického modelu, který pokryje další cévy.

| matematická funkce | parametr | začátek intervalu | konec intervalu |
|------------------------|----------|-------------------|-----------------|
| racionální funkce 5.1 | b | 2,1345 | 3,8235 |
| Gompertzova funkce 5.2 | a | 39,143 | 55,876 |
| funkce odmocniny 5.3 | a | 0,0137 | 0,0427 |

Tabulka 5.4: Parametr pro konkrétní funkce a jejich intervalový obor hodnot.



Obrázek 5.4: Oblasti pokryté funkcemi, vykreslené v souřadném systému. Oblast pokrývající racionální funkce 5.1 (a), plocha vymezená Gompertzovou funkcí 5.2 (b), prostor zabírající funkce druhé odmocniny 5.3 (c), všechny vymezené plochy ze předchozích průběhů (d).

5.2.3 Vyhodnocení získaných funkcí spodních makulárních cév

Tato podkapitola je vytvořena identicky s podkapitolou předchozí. Výsledné funkce z regresivních programů udává jejich matematický předpis, který je uveden níže. Ke každému získanému modelu odpovídá i tabulka se získanými hodnotami pro všechny parametry daného modelu.

Mezi uvedené funkce opět patří *racionální model* 5.4. Vypočtené parametry tohoto modelu jsou odlišné a zobrazeny tabulkou 5.5. Taktéž se opakuje *model druhé odmocniny* 5.6, který nabývá odlišné hodnoty parametru v tabulce 5.7. *Gompertzovu funkci* nahrazuje model *logistické regrese* 5.5, kde dosažené hodnoty jsou ukázány tabulkou 5.6.

Získané modely s vyčíslenými parametry zobrazuje skupina grafů 5.5. Vypočtené funkce vykazují vysokou podobnost s modely horních makulárních cév.

$$f(x) = \frac{a + bx}{1 + cx + dx^2} \quad (5.4)$$

| parametr | hodnota parametru |
|----------|----------------------|
| a | -6,832217 |
| b | -2,704067 |
| c | 0,027136 |
| d | $4,183364 * 10^{-4}$ |

Tabulka 5.5: Vypočtené hodnoty parametrů pro lomenou funkci 5.4.

$$f(x) = \frac{a}{1 + be^{-cx}} \quad (5.5)$$

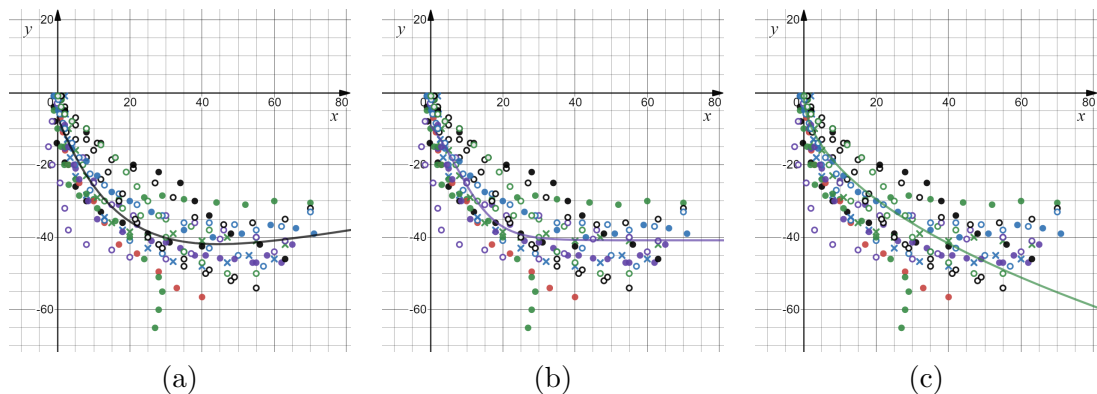
| parametr | hodnota parametru |
|----------|-------------------|
| a | -40,084039 |
| b | 3,591731 |
| c | 0,176445 |

Tabulka 5.6: Vypočtené hodnoty parametrů pro logistickou regresi 5.5.

$$f(x) = -a\sqrt{x} \quad (5.6)$$

| parametr | hodnota parametru |
|----------|-------------------|
| a | 6,59736 |

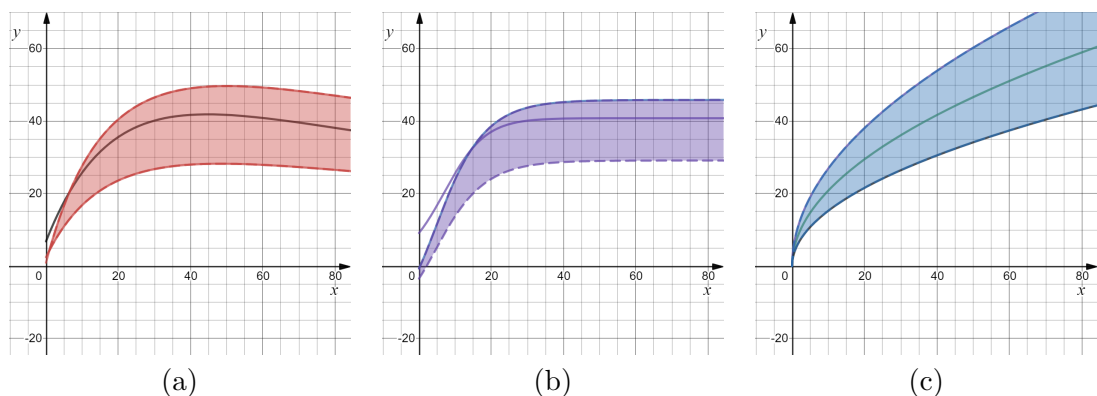
Tabulka 5.7: Vypočtené hodnoty parametrů pro funkci druhé odmocniny 5.6.



Obrázek 5.5: Matematické funkce zobrazené v souřadném systému. Křivka racionálního modelu 5.4 (a), model logistické regrese 5.5 (b), funkce druhé odmocniny 5.6 (c).

5.2.4 Vyhodnocení podobnosti horních a spodních makulárních modelů

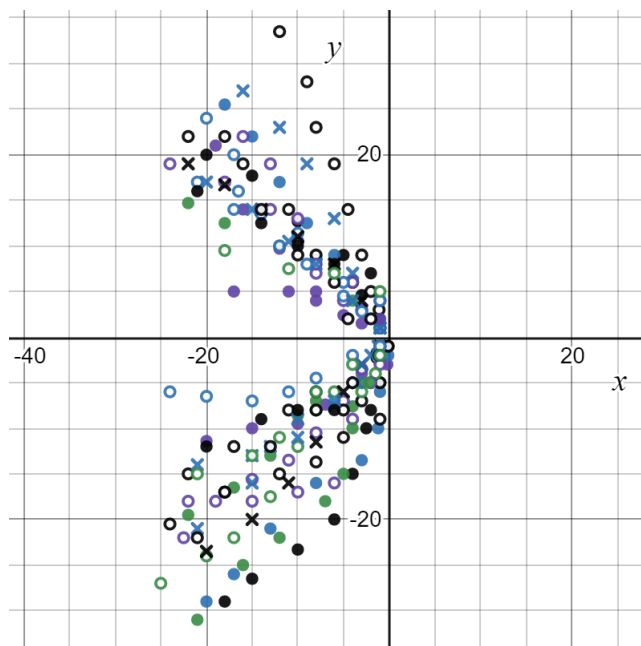
Pro zjištění totožnosti modelů s horními makulárními cévami byly funkce přesunuty ze IV do I kvadrantu souřadného systému. Přesun realizovala funkce absolutní hodnoty. Hodnoty byly zrcadleny přes $osu x$ souřadného systému. Přemístěné funkce se porovnávaly s oblastmi horních makulárních modelů. *Gompertzova* oblast byla asociována s křivkou *logistické regrese*. Zbylé modely byly spojeny s oblastmi, které vytvořily totožné modely. Výsledky reprezentuje soubor grafů 5.6. Z tohoto zjištění je možné pro vytváření spodních makulárních cév použít modely horních makulárních cév. Tuto transformaci lze realizovat vynásobením oboru hodnot s konstantou -1 . Naměřené body spodních makulárních cév se převedly ze IV do I kvadrantu souřadného systému a proběhla korekce parametrů u vyčíslených modelů pro horní makulární cévy.



Obrázek 5.6: Oblasti pokryté modely horních makulárních cév, ve kterých se nacházejí křivky spodních makulárních cév v absolutní hodnotě. Plocha 5.1 a křivka 5.4 reprezentuje racionální model (a), oblast vymezená Gompertzovou funkcí 5.2 s modelem logistické regrese 5.5 (b), prostor zabírající funkce druhé odmocniny 5.3 a křivka stejného modelu 5.6 (c).

5.3 Detekce směru a tvaru horních a spodních nasálních cév

Ke zpracování bylo taktéž použito 80 cév z databáze *Messidor* jako při práci s cévami v makulární oblasti 5.1. Označování bodů probíhalo ve webovém programu *Desmos*. Bylo využito stejného zvětšení a posunutí snímku jako v případě detekce v makulární oblasti. Snímek byl zvětšen na velikost 170×120 a posunut tak, aby vstup cévy do sítnice odpovídal počátku souřadného systému ($[0;0]$). Vzdálenost mezi jednotlivými body dané křivky se blížila pěti interním jednotkám programu. Získaná data zobrazuje obrázek 5.7.



Obrázek 5.7: Snímek z databáze *Messidor* zasazený do prostředí *Desmos* a proložený body horní makulární žíly.

5.4 Aproximace detekovaných tvarů nasálních cév nelineárními modely

Při této aproximaci bylo využito poznatku z podkapitoly 5.2.4, podobnosti horních a spodních makulárních cév. Tento poznatek se přenesl na spodní a horní nasální cévy. Nejdříve proběhlo sjednocení rozdělených nasálních dat. Body spodních nasálních cév byly funkcí absolutní hodnoty transformovány ze III do II kvadrantu souřadného systému. Dvojitý počet dat zpracovaly programy *CurveExpert* a *RegressionTools*. Ty vytvořily seznam nejvhodnějších matematických modelů. Jednotlivé výsledky byly opticky validovány, zda splňují přirozený tvar cév. Tato kontrola se převážně věnovala místu za posledním naměřeným bodem. Následně proběhlo určení intervalových parametrů k jednotlivým funkcím. Závěr podkapitoly popisuje ověření platnosti podobnosti horních a spodních nasálních cév. To proběhlo porovnáním oblastí, které zaplňují jednotlivé modely s naměřenými body pro spodní nasální cévy.

5.4.1 Získané funkce nasálních cév

Vybrané modely s nejlepším hodnocením zobrazují matematické předpisy v této části. K jednotlivému předpisu náleží i tabulka parametrů, které upřesňují danou funkci. Závěr podkapitoly zobrazuje vybrané křivky. Ty zobrazují pouze jednu určitou cévu. Problematikou převedení jedné cévy na celou oblast se zabývá následující část.

První uvedený model představuje lomenou funkci 5.7, vypočtené parametry daného modelu zobrazuje tabulka 5.8. Model mocniny 5.8 reprezentuje další z uvedených křivek. Zjištěné hodnoty parametrů k této křivce obsahuje tabulka 5.9. Poslední zástupce představuje funkci odmocniny 5.9, kde stanovené hodnoty proměnných obsahuje tabulka 5.10. Jednotlivé funkce nasálních cév zobrazuje graf 5.8 v souřadném systému.

$$f(x) = \frac{ax}{x - b} \quad (5.7)$$

| parametr | hodnota parametru |
|----------|-------------------|
| a | 48,712318 |
| b | 26,837965 |

Tabulka 5.8: Určené parametry k lomené funkci 5.7.

$$f(x) = a(-x)^b \quad (5.8)$$

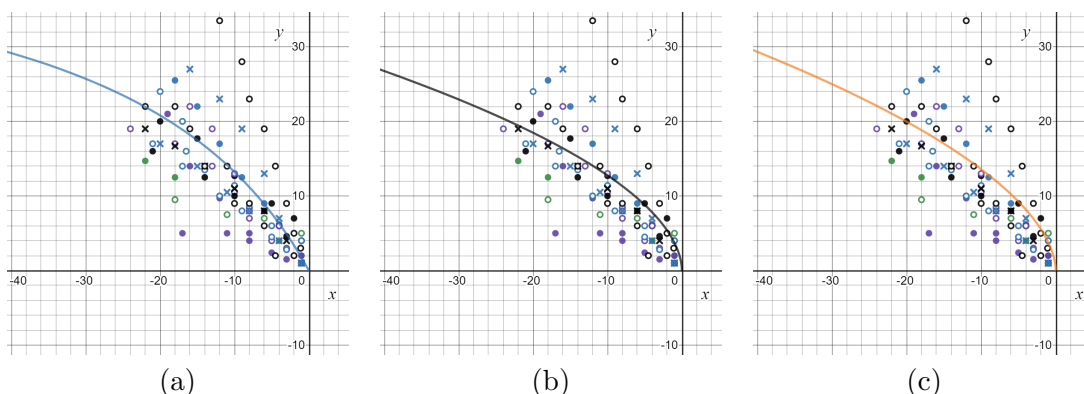
| parametr | hodnota parametru |
|----------|-------------------|
| a | 3,734543 |
| b | 0,533671 |

Tabulka 5.9: Zjištěné parametry k mocninnému modelu 5.8.

$$f(x) = c \left(\frac{bx}{a} \right)^{\left(\frac{1}{b}\right)} \quad (5.9)$$

| parametr | hodnota parametru |
|----------|-------------------|
| a | 0,323413 |
| b | 1,896438 |
| c | 1,412325 |

Tabulka 5.10: Parametry určené pro funkci odmocniny 5.9.



Obrázek 5.8: Matematické modely zobrazené v souřadném systému. Lomená funkce 5.7 (a), mocninový model 5.8 (b), křivka odmocniny 5.9 (c).

5.4.2 Zjištění intervalových parametrů a jejich korekce pro modely nasálních cév

V této části byly zjištěny parametry pro jednotlivé modely nasálních cév. Je nutné, aby předem určený parametr nabýval intervalu hodnot. Tím je zaručena větší náhodnost cév. Meze intervalu se nastavily tak, aby spadalo alespoň 90 % naměřených bodů do vytyčené oblasti. Výsledné hodnoty lze sledovat v tabulce 5.11. Soubor grafů 5.9 zobrazuje plochy, které definují jednotlivé modely svými parametry. Poslední graf prezentuje sjednocení oblastí.

Následně proběhla korekce empiricky zjištěných oborů hodnot. Z databáze *Messidor* bylo náhodně vybráno 100 snímků. Zanesené snímky do souřadného systému byly porovnávány s oblastmi modelů. Jejich parametry se upravovaly dokud v oblastech neleželo alespoň 90 % cév. Tato hranice se jeví jako dostatečně přesná pro další práci. Pokud takto vytvořená přesnost nebude dostačovat, je možné přidat další model, který pokryje určené nedostatky.

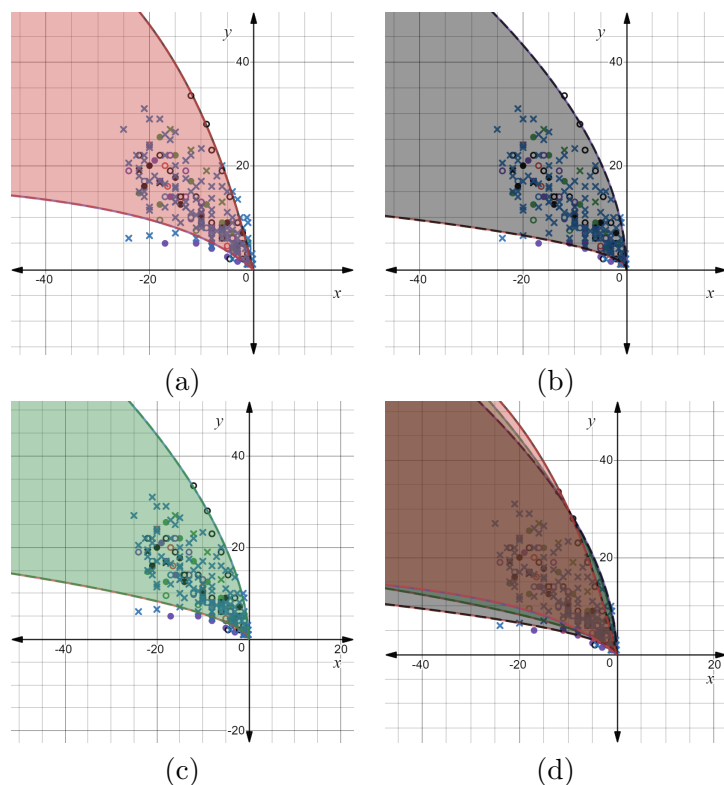
| matematická funkce | parametr | začátek intervalu | konec intervalu |
|----------------------|----------|-------------------|-----------------|
| racionální model 5.7 | a | 22,45632 | 110,4456 |
| model mocniny 5.8 | a | 1,323423 | 8,743423 |
| funkce odmocniny 5.9 | c | 0,812443 | 2,623523 |

Tabulka 5.11: Obor hodnot parametru pro konkrétní modely nasálních cév.

5.4.3 Ověření podobnosti spodních a horních nasálních cév

K ověření podobnosti bylo využito poznatku v podkapitole 5.2.4 o podobnosti horních a spodních makulárních cév. Tento poznatek se použil i pro nasální část sítnice. Ověření platnosti i pro nasální cévy proběhlo porovnáním oblastí jednotlivých matematických modelů s naměřenými body nasálních cév. Druhá fáze ověření porovnávala spodní nasální cévy z existujících sítnic databáze *Messidor* s plochami, které definují spočítané modely.

Funkce pokryly přes 90 % změřených bodů spodních nasálních cév. Úspěšnost ověření pomocí existujících sítnic dosahovala téměř totožných výsledků. Tento poznatek lze použít i u vytváření nasálních cév.



Obrázek 5.9: Oblasti vyplněné funkcemi pro nosální cévy. Plochu zaplňující racionální model 5.7 (a), oblast vymezená mocninou funkcí 5.8 (b), prostor zadaný matematickým modelem odmocniny 5.9 (c), předchozí oblasti umístěné do stejného snímku (d).

5.5 Shrnutí dosažených výsledků

V kapitole proběhla identifikace tvaru hlavních cév, které se nacházejí na viditelné části reálného snímku sítnice oka. Pro každou skupinu hlavních cév byly vypočteny tři matematické modely. Pro každý model proběhlo ověření vypočtených hodnot parametrů a jejich korekce s reálnými snímky sítnic.

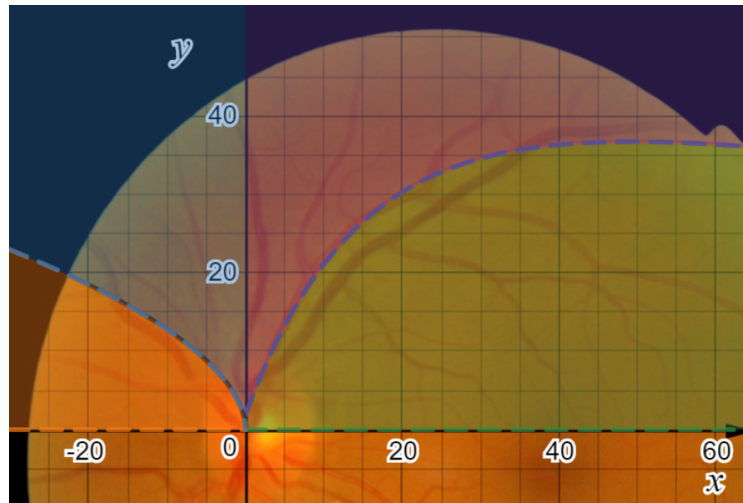
Byl vytvořen předpoklad o stejném chování cév v horních a spodních oblastech. Tento předpoklad potvrdilo řádné testování modelů horních cév na naměřených hodnotách spodních cév na existujících snímcích sítnic oka. Tato vlastnost umožňuje sjednocení logických částí sítnice oka. Bude dále možné pracovat pouze s horní nosální a makulární oblastí. Výsledky této práce byly přeneseny i do odpovídajících spodních oblastí.

Kapitola 6

Identifikace tvarů a větvení cév vycházejících z makulárních i nasálních cév

Tato kapitola se zabývá zpracováním cév, které vznikají větvením z výše popsaných žil a tepének. Pozorovaná část sítnice byla rozdělena na čtyři sektory. Rozdělení popisuje obrázek 6.1. Snímek byl rozdělen na nasální a makulární část. Obě části dělí křivka reprezentující hlavní cévu. Každý z těchto sektorů představuje jiné chování cév, které se v něm nacházejí. Zkoumané snímky pocházejí z dostupné databáze *Messidor*.

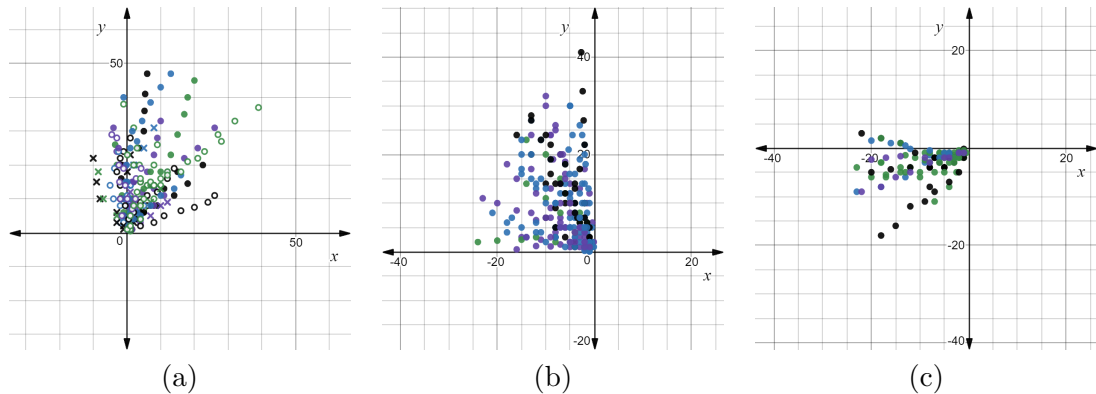
Všechny sektory byly podrobeny pozorování. Nejprve proběhla detekce bodů těchto cév. Nad získanou sadou bodů byl vytvořen matematický aparát, který co nejvíce přiblíží chování těchto cév. Další část kapitoly se věnuje určení správného úhlu mezi dvěma cévami v bodě jejich větvení. K zamezení velkého počtu větvení došlo ke změření maximálního zanoření větvení v podkapitole 6.4.



Obrázek 6.1: Reálný snímek sítnice oka rozdělený na čtyři části. První část reprezentuje makulární cévy nad hlavní makulární cévou. Druhá čtvrtina představuje nasální cévy, které směřují ve směru osy y . Třetí oblast zabírá nasální cévy, které jsou rovnoběžné s osou x . Poslední část představuje plochou největší makulární oblasti, kde cévy směřují do *fovey*.

6.1 Detekce směru a tvaru cév vycházejících z makulárních a nasálních cév

Značení a prezentace bodů probíhala v programu *Desmos*. Snímek byl zvětšen na velikost 170×120 jednotek. Proběhlo posunutí snímku, aby se větvení u zkoumané cévy nacházelo na bodu souřadného systému $[0;0]$. Vzdálenost mezi jednotlivými naměřenými hodnotami se blížila pěti interním jednotkám programu. Získaná data zobrazuje soubor grafů 6.2.



Obrázek 6.2: Naměřené body cév vycházejících z hlavních cév v souřadném systému. Horní makulární sektor (a), horní nasální část (b), spodní nasální cévy (c).

Získaná data zpracovaly programy *CurveExpert* a *RegressionTools*. Výsledné křivky pro každou část sítnice neodpovídaly přesné aproximaci. Výsledné funkce nedosahovaly přesnosti dříve získaných modelů. Hlavním problémem nepřesné aproximace se jevil velice rozetřený vějíř dat, kde data zabírala téměř celý kvadrant souřadného systému s téměř totožným zastoupením bodů. Nejvíce tento jev zobrazuje snímek 6.2 část b.

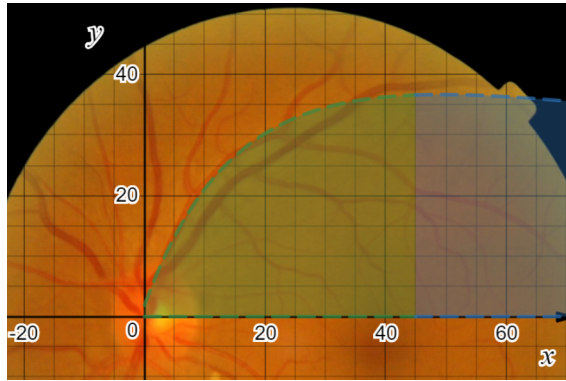
Vypočtené výsledky vyloučily možnost použití matematických funkcí pro reprezentaci naměřených hodnot. Dalším důležitým faktorem, který zabránil použití zmíněného přístupu, byla nemožnost definovat ve funkci pro jednu hodnotu z definičního oboru více hodnot z oboru hodnot.

6.1.1 Detekce směru cév mířících do foveální oblasti

Tato část snímku sítnice oka nebyla změřena stejným způsobem jako zbývající části. Oblast vykazuje jiné a lépe sledované vlastnosti. Nejdůležitějším pozorovaným vzorem chování se jevila cílová destinace cév. Ty obecně směřují, při vložení snímku do souřadného systému 6.2, k souřadné ose x . Tento vzor lze rozdělit na dvě podkategorie podle polohy cévy. V první skupině cévy směřují do *foveální* části. Odlišné chování vykazuje druhá skupina cév, která směřuje k souřadné ose x . Obrázek 6.3 popisuje rozmístění i velikost obou sektorů.

U každé sledované reálné sítnice se hranice oblastí lišily. Ty se i ve většině případů prolínaly a nebylo jednoznačně možné identifikovat bod zlomu. Nejlepší aproximace této hranice se jevilo místo konce *foveální* oblasti.

Cévy směřují do *foveální* oblasti, avšak nejsou v této oblasti již viditelné. Vizualně zanikají na okraji tohoto prostoru, některé mizí již před tímto horizontem.



Obrázek 6.3: Reálný snímek sítnice oka zobrazuje plochou makulární oblast. Ta je rozdělena na dvě části. V první (zelené) ploše cévy směřují do *fovey*. V druhé oblasti (modrá) cévy směřují k souřadné ose x .

6.2 Realizace cév pomocí vektorů

Lepší predispozice představují vektory. Ty už svojí definicí eliminují problém matematických funkcí, a to přiřadit jedné hodnotě z definičního oboru více hodnot z oboru hodnot.

Problém špatné identifikace směrů a tvarů cév vycházejících z hlavních makulárních i nasálních cév nebylo možné vyřešit pomocí jedné proměnné (vektor cévy). Bylo nutné přidat další prvek. Jako vhodný prvek byl vybrán úhel, pod kterým se provede větvení cév.

Kvůli eliminaci problémů bylo vhodné vektory udržovat v normalizovaném tvaru. Ten představuje vektorovou složku x udržovat ve velikosti 1. Každý vektor nabývá ve složce x hodnoty z množiny $\{-1; 1\}$. Složka y představuje informaci o tom, jaký sklon a směr daný vektor reprezentuje.

Vektorová reprezentace odstraňuje realistické prohnutí cév a nastavuje pouze rovnou přímkou. Pro vytvoření realističtějšího chování cév bylo nutné přidat mechanismy pro zakřivení daného vektoru. První mechanismus představuje dlouhodobější řešení. Při každém výpočtu směru cévy byl zohledněn i směr cévy, ze které se větví. Výpočet nového směru reprezentuje vztah 6.1. X a y představují složky nově získaného vektoru pro směr konkrétní cévy. Symboly x_a a y_a představují jednotlivé složky vektoru, který popisuje aktuální cévu. X_p a y_p zastupují vektor cévy, ze kterého se aktuálně počítaná céva větvila. Koeficient podobnosti s rodičovskou cévou c byl nastaven na hodnotu 0,2.

Pokud by rodičovskou cévu reprezentovala matematická funkce a nikoli vektor, bude vypočítána derivace nad touto funkcí a převedena na vektor. Vyhnutí se derivaci by znamenalo velké zjednodušení výpočtu, proto byla realizována alternativa, kdy se vytvoří přibližná derivace odečtením hodnoty $f(x)$ od $f(x + 1)$. Kde f představuje funkci tvaru cévy.

$$\begin{aligned} x &= \frac{x_a + x_p c}{|x_a + x_p c|} \\ y &= \frac{y_a + y_p c}{|x_a + x_p c|} \end{aligned} \tag{6.1}$$

Druhý lokální mechanismus provede náhodné odchýlení od původní přesné hodnoty bodu, kterou céva interpoluje. Zanesení částečné nepřesnosti popisuje matematická rovnice 6.2. Tento mechanismus provede přičtení, popřípadě odečtení maximálně 5% z aktuální

funkční hodnoty. Pro správný chod mechanismu je nutné generovat náhodné hodnoty parametru c v intervalu $\langle -1; 1 \rangle$.

$$y = y + 0,05yc \quad (6.2)$$

Vytvoření nové cévy, která bude reprezentována vektorem vzniká následovně. Je nutné znát vektor, který reprezentuje cévu, ze které probíhá větvení. Pokud tuto cévu reprezentuje funkce, je nutné spočítat derivaci a převést ji na vektor (viz výše). Pomocí vztahů 6.3 se převede aktuální vektor o úhel α . Nově vzniklý vektor musí podstoupit normalizaci předpisem 6.4, protože transformace otočení nezanechá složku vektoru x v jedničkovém tvaru.

$$\begin{aligned} x &= x_p \cos(\alpha) + y_p \sin(\alpha) \\ y &= y_p \cos(\alpha) - x_p \sin(\alpha) \end{aligned} \quad (6.3)$$

V rovnici 6.3 x a y představují složky nově vzniklého vektoru. Symboly x_p a y_p definují původní vektor. Znak α určuje velikost úhlu, o který bude vektor otočený.

$$\begin{aligned} x &= \frac{x_p}{|x_p|} \\ y &= \frac{y_p}{|x_p|} \end{aligned} \quad (6.4)$$

Předpis 6.4 popisuje normalizaci vektoru, která normalizuje složku x z původní $|x_p|$. Provádí se i stejné dělení u y , kde původní hodnotu y_p dělí $|x_p|$.

6.3 Měření velikostí úhlů v bodech větvení cév

K vytvoření větvení cév je nutné znát úhel, pod kterým se budou rozbíhat. V první části proběhlo změření několika cév u existujících sítnic. Kvůli problémům spojených s měřením se přešlo k empirickému zjišťování velikosti úhlu. K zachování kompatibility s existujícím nástrojem bylo nutné do měření zavést procentuální šířku cév. Ve výsledném vztahu 6.5 toto představuje r . Tento symbol udává, kolik procent z původní šířky představuje nově vytvořená céva při větvení. Pokud z větvení vzniknou dvě stejně velké cévy, bude tato neznámá nabývat hodnoty 0,5. Součet šířek cév musí u každého větvení dosáhnout hodnoty 1.

Při měření bylo zjištěno, že i původní céva, ze které se větví další, se vychýlí v opačném směru. Velikost úhlu, o který se původní céva odkloní od svého směru, závisí na velikosti šířky nové cévy. Pro provedení tohoto jevu lze použít vztah 6.5 s šířkou původní cévy a vynásobením velikosti úhlu s číslem -1.

Neznámá r nebývá hodnot z intervalu $\langle 0; 1 \rangle$. Výsledná velikost úhlu α obsahuje hodnoty z prvního kvadrantu goniometrické kružnice $\langle 0; \frac{\pi}{2} \rangle$.

$$\alpha = \frac{\pi}{2} - \frac{r\pi}{2} \quad (6.5)$$

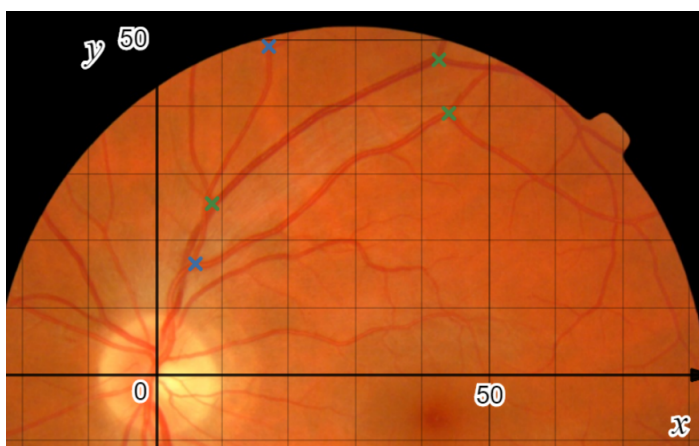
6.3.1 Poměr šířek cév při větvení

Poměrový systém šířek byl převzat z původního generátoru a mírně upraven. Z empirického testování větvení podobně velkých cév dosahovaly šířek z intervalu $\langle 0,55; 0,7 \rangle$. Malé cévy, které převážně míří do oblasti *fovey*, nabývají hodnot z intervalu $\langle 0,04; 0,08 \rangle$.

6.4 Určení hloubky větvení cév

Pro přesnější chování cév byla pozorována hloubka větvení. Tato vlastnost popisuje, kolikrát se daná céva větví nad viditelnou částí sítnice. Měření se aplikovalo na tři oblasti sítnice. Ty zobrazuje obrázek 6.1, nasální části vykazovaly téměř stejné výsledky, proto byly tyto dvě oblasti sloučeny do jedné. Popis postupu zařazování jednotlivých cév do skupin úrovní zanoření ukazuje obrázek 6.4. Měřená céva patří pouze do jedné skupiny, a to podle všech větvení, které zobrazuje viditelná část sítnice.

Snímky sítnic pochází z databáze *Messidor*, kde bylo náhodně vybráno 25 snímků. Měření na každém snímku detekovalo žíly i tepénky sítnice. V každé oblasti proběhlo měření s 50 cévami. Počet se pohybuje na hranici relevantnosti. Výsledky byly řádově jednoznačné, proto nebylo měřeno na větším vzorku dat.



Obrázek 6.4: Snímek sítnice zobrazený v souřadném systému. Modré kříže značí cévu, které patří do skupiny druhé úrovně zanoření. Zeleně označené cévy spadají do první úrovně zanoření. Zaznačeny jsou některé cévy, které se větví nad makulární křivkou.

Dosažené výsledky pro cévy směřující do makulární oblasti obsahuje tabulka 6.1. Ty se nejčastěji větví dvakrát s naměřenou pravděpodobností 56,7 %. Zanedbatelné je větvení u cév, které se větví čtyřikrát, s pravděpodobností 1,8 %.

Tabulka 6.2 popisuje naměřené výsledky cév, které se větví nad makulární křivkou. Ty s více jak dvou třetinovou pravděpodobností se větví pouze jednou na viditelné části. Téměř zanedbatelné výsledky ukazuje skupina tří větvení s pravděpodobností 4,1 %.

Nasální oblasti zobrazuje tabulka 6.3. Ta obsahuje data všech cév, které se větví v nasální oblasti. Ty se v naprosté většině větví pouze jednou s naměřenou pravděpodobností 80,6 %. S minimálním procentuálním zastoupením se objevuje skupina se třemi větveními.

| úroveň zanoření | počet cév | zastoupení v procentech |
|-----------------|-----------|-------------------------|
| 1 | 40 | 23,4 % |
| 2 | 97 | 56,7 % |
| 3 | 31 | 18,1 % |
| 4 | 3 | 1,8 % |

Tabulka 6.1: Hodnoty popisující zanoření u cév směřující do *fovey*. Procenta byla zaokrouhlena na jedno desetinné místo .

| úroveň zanoření | počet cév | zastoupení v procentech |
|-----------------|-----------|-------------------------|
| 1 | 67 | 68,4 % |
| 2 | 27 | 27,6 % |
| 3 | 4 | 4,1 % |

Tabulka 6.2: Výsledky popisující zanoření u cév nad makulární křivkou. Procentuální hodnoty zaokrouhleny na jedno desetinné místo.

| úroveň zanoření | počet cév | zastoupení v procentech |
|-----------------|-----------|-------------------------|
| 1 | 83 | 80,6 % |
| 2 | 18 | 17,5 % |
| 3 | 2 | 2 % |

Tabulka 6.3: Hodnoty popisující zanoření u cév v celé nasální oblasti. Třetí sloupec zaokrouhlený na jedno desetinné místo.

6.5 Aproximace pozic větvení z cév předepsaných matematickou funkcí

V rámci pravidelnějšího větvení cév byl naměřen počet cév, které se větví z hlavní makulární cévy. Hodnoty byly následně empiricky upraveny a zaokrouhleny. Při samotném spuštění generování řečiště pro jednu cévu jsou vygenerovány tyto větvící body. Tyto body jsou uloženy ve frontě a následně z ní odebírány při jejich vytvoření. Empiricky zjištěné hodnoty jsou zobrazeny v tabulce 6.4

První pozice je náhodně vypočtena z intervalu. K výpočtu dalších pozic je připočtena i hodnota vzdálenosti od středu dříve vygenerovaného bodu. Tato korekce zajišťuje, že generované body budou mít mezi sebou určitou vzdálenost.

| Interval | Počet větvení | | |
|----------|-------------------------|--------------------------|--------------------------|
| | I | II | III |
| 1 | $\langle 3; 24 \rangle$ | $\langle 2; 15 \rangle$ | $\langle 2; 15 \rangle$ |
| 2 | | $\langle 20; 40 \rangle$ | $\langle 15; 35 \rangle$ |
| 3 | | | $\langle 35; 60 \rangle$ |

Tabulka 6.4: Počet větvení u hlavních cév a interval pozic, kterých můžou nabývat.

6.6 Shrnutí dosažených výsledků

Kapitola se zabývala cévami, které vycházejí z hlavních křivek a byly vypočteny v předchozí kapitole. První předpoklad použití matematických funkcí se nenaplnil. U pozorovaných cév funkce vykazovaly značné nedostatky. Vhodnější model představují vektory, které jsou v mnoha ohledech flexibilnější.

Proto byl navržen systém větvení s použitím vektorů. Ten využívá transformaci otočení objektu. Vektor je otočený o určitý úhel, který ovlivňuje šířka nově vzniklé cévy. Šířku cévy využívá existující generátor a pro zaručení kompatibility byla tato vlastnost ponechána a využita.

V poslední části se kapitola zabývala počtem větvení jednotlivé cévy. Měření rozdělilo sítnici na tři části z důvodu podobnosti výsledků. Pro každou oblast byl změřen počet větvení.

Kapitola 7

Algoritmus pro tvorbu cévního řečiště syntetických sítnic

Kapitola se zabývá návrhem algoritmu pro tvorbu syntetických sítnic. Ten využije poznatky zjištěné v předchozích kapitolách, především v 5 a 6. Předpis nebude závislý na konkrétních technologiích. Algoritmus musí dodržet vstupní a výstupní formát dat, který je popsán v podkapitole 7.1.

Z důvodu lepší čitelnosti a přehlednosti byl algoritmus rozdělen do dvou podkapitol. V první části jsou popsány jeho základní charakteristiky, v druhé podkapitole byl popsán podrobněji.

Algoritmus si definuje vlastní souřadný systém, který bude popsán v této kapitole. Nový systém byl vytvořen převážně z důvodů definičního oboru a oboru hodnot matematických funkcí. Musela být vytvořena konverzní funkce pro převod mezi tímto a původním souřadným systémem.

7.1 Vstupní a výstupní parametry algoritmu

Pro správné pracování algoritmu s okolními částmi programu musí dodržet vstupní parametry a výstupní formát dat.

Existující nástroj nadefinoval vstupní parametry pro původní algoritmus. Téměř všechny vstupní hodnoty je nutné zachovat pro další interakci s generátorem. Informace, které je nutné uchovat pro další použití, popisuje následující seznam.

- Parametr *type* určuje barvu, kterou dostane nově vygenerovaný strom cévního řečiště. Nabývá hodnot z množiny $\{0; 1\}$. Hodnota 0 reprezentuje tepnu. Na výsledcích je zobrazena světle červenou barvou. Žílu představuje číslo 1, která je prezentována ve výsledcích tmavě červenou.
- *Quadrant* popisuje, ve kterém kvadrantu souřadného systému má být strom cévního řečiště generovaný. Parametr dostává jednu hodnotu z množiny $\{1; 2; 3; 4\}$ podle příslušného kvadrantu.
- *Width* představuje hodnotu nepostradatelnou pro vykreslení cévního řečiště. Ta není spojena s atributem *width* u třídy `bifurcationPoint`.

- Parametry *startX* a *startY* reprezentují souřadnici, které značí vstup tepny a žíly do sítě oka. Tyto parametry představují interní souřadnice generátoru a nabývají hodnot z intervalu $\langle -1; 1 \rangle$.

Výstup musí představovat *stromová struktura* instancí tříd `bifurcationPoint`. Ta bude následně předána původnímu generátoru a vykreslena. Algoritmus musí korektně inicializovat a pracovat s instancemi `bifurcationPoint`. Atributy i jejich významy zobrazuje následující seznam.

- *PositionX* a *positionY* udržují informaci o poloze bodu v souřadném systému. Používá se interní systém, kde souřadnice nabývají hodnot z intervalu $\langle -1; 1 \rangle$.
- *Width* udává relativní šířku cévy v bodě větvení. Ve slepé skvrně je hodnota atributu rovna 1. Pokud má céva nulovou šířku *width* obsahuje 0. Atribut může nosit hodnotu z intervalu $\langle 0; 1 \rangle$.
- *PrevDistance* nosí informaci o vzdálenosti v Eukleidovském prostoru. Udává přímou vzdálenost mezi aktuálním uzlem a jeho *rodičem*. Počítá nad souřadnicemi interního souřadného systému.
- *WidthRatio* popisuje poměr šířek, při jakém se dělí cévy. Pokud aktuální bod nepopisuje větvení, atribut *withRation* je nastaven na hodnotu 1. Při větvení cévy musí součet atributů být 1. Obor hodnot, kterých může nabývat je $\langle 0; 1 \rangle$.
- *DirectionAngle* představuje úhel, který svírá spojnice *rodičovského* a aktuálního uzlu se souřadnou osou *x*, udávaný ve stupních.
- *CurveWeight1*, *curveWeight2* a *curveEndDeviation* představují náhodně zvolené konstanty pro korektní vykreslení části cévního řečiště. *CurveWeight1* a *curveWeight2* nabývají hodnot z intervalu $\langle 0,8; ,2 \rangle$ pomocí rovnoměrného rozložení. Atribut *curveEndDeviation* s 50% šancí obsahuje hodnotu z $\langle 60; 90 \rangle$, jinak hodnotu z intervalu $\langle -90; -60 \rangle$. Obě možnosti používají k určení konkrétní konstanty rovnoměrné rozložení.
- Pomocný atribut *prevPoint* ukazuje na *rodičovský* uzel. Zpětný ukazatel je nutný při vykreslení cévy.
- *NextPointRight* a *nextPointLeft* jsou pomocné ukazatele pro vytvoření stromové struktury. Ukazují na své *potomky*.

7.2 Obecný popis algoritmu

V podkapitole jsou popsány základní charakteristiky algoritmu. Lze tak lépe vystihnout jádro předpisu bez informací, které nejsou důležité pro funkčnost, ale pro jeho větší přesnost. Níže sepsaný seznam prezentuje stručný předpis:

- 1) Provedení inicializace matematické funkce. Vytvoření počátečního bodu.
- 2) Výpočet nového bodu pomocí aktuální funkce. Doplnění do předchozího bodu odkaz na aktuální bod.
- 3) Pokud nově vytvořený bod se nachází uvnitř viditelného snímku pokračuje se bodem 3.A, jinak 4.
 - 3.A) Vyhodnocení, zda se v aktuálním bodě bude vytvářet větvení cév. Při tvorbě větvení se postupuje níže, jinak bodem 3.B.
 - Vložení aktuální matematické funkce do zásobníkové struktury.
 - Vytvoření nové matematické funkce na základě parametrů větvení.
 - Nastavení nové funkce na aktuální a postupuje bodem 2.
 - 3.B) Nastavení aktuálního bodu na průchozí a pokračuje bodem 2.
- 4) Ukončí se práce s aktuální matematickou funkcí.
- 5) Jestliže je zásobníková struktura neprázdná, pokračuje se bodem 5.A jinak 6.
 - 5.A) Vyjmutí vrcholu zásobníku. Nastavení vytažené funkce na aktuální. Postupuje bodem 2.
- 6) Pro každý bod stromové struktury vypočítání atributů potřebných pro kompatibilitu. Provedení převrácení hodnot do správného kvadrantu, pokud vstupní kvadrant není *I* nebo *II*.
- 7) Vratí se strom vypočítaných bodů.

7.3 Detailní popis jednotlivých částí algoritmu

Podkapitola rozvíjí stručně definované kroky v předchozí části. Důležité pasáže budou podrobně rozvedeny v následujících odstavcích. Detaily bodů algoritmu představují převážně zpřesnění postupů na základě informací zjištěných v kapitolách 5 a 6.

7.3.1 Vytvoření nového souřadného systému

Pro efektivnější práci s matematickými funkcemi byl zaveden nový souřadný systém, který odpovídá systému použitého při statistickém zpracování jednotlivých cév. Běh algoritmu se pohybuje mezi souřadnicemi -30 a 100 na ose x . Hranice 60 a -60 udržuje osa y . Počátek souřadného systému $[0;0]$ zobrazuje místo proniknutí cévy do sítnicové oblasti ve *slepé skvrně*.

Vytvořením vlastního souřadného systému je nutné vytyčit i nově hranice viditelnosti pro jednotlivé cévy. Generování cév probíhá pouze v *I* a *II* kvadrantu, pro zbylé kvadranty

proběhne transformace. Maximální hodnota na ose y je 60, minimální nabývá hodnoty -1. Pro osu x minimum leží na čísle -30. Maximum této osy odpovídá hodnotě 80.

Při vytváření cév směřujících do *foveální* oblasti, je nutné nastavit hranici na okraj této části vycházející z pozorování v podkapitole 6.1.1. Oblast byla aproximována na kruhový tvar. Ve vytvořeném souřadném systému se nachází na místě [30;0] s poloměrem 12 jednotek.

Pro správnou funkčnost algoritmu s okolním systémem je nutné nadefinovat transformační funkce z nově vytvořeného souřadného systému do původně používaného. Převodní rovnice pro složku x definuje předpis 7.1, kde x_{old} představuje souřadnici starého souřadného systému. Souřadnice x_{act} reprezentuje nově použité souřadnice. Symboly $minX$ a $maxX$ udržují informaci o minimální a maximální hodnotě systému s konstantními hodnotami -50 a 100. Konstanta 1,08 provádí posunutí hodnot.

$$x_{old} = -2 \frac{x_{act} - minX}{maxX - minX} + 1,08 \quad (7.1)$$

Převodní vztah pro složku y určuje předpis 7.2, kde y_{old} zastává hodnotu starého souřadného systému. Souřadnice y_{act} představuje pozici v novém systému. Symbol $maxY$ uchovává informaci o maximální hodnotě systému s konstantní hodnotou 75.

$$y_{old} = \frac{-y_{act}}{maxY} \quad (7.2)$$

7.3.2 Prvotní inicializace – 1. bod algoritmu

1. Vytvoření *kořenového* bodu pro stromovou strukturu, který se nachází na souřadnicích [0;0].
2. Na základě vstupního parametru, určujícího kvadrant souřadného systému pro danou cévu, vytvořit korektní matematickou funkci.
 - Modely pro nasální část popsané v 5.4, funkce pro makulární oblast v 5.2.
 - Pro náhodně zvolenou funkci nastavit interval parametru, který popisují tabulky 5.4 a 5.11.
 - Hodnotu parametru vytvořit pomocí *rovnoměrného* rozdělení s dříve zjištěnými mezemi.
 - Nastavení velikosti kroku funkce na hodnotu 3, která určuje vzdálenost dvou vygenerovaných sousedních bodů danou funkcí.
3. Vytýčit hranici pro viditelnost cév. Tu popisuje úvod této podkapitoly.

7.3.3 Výpočet nového bodu pomocí aktuální funkce – 2. bod algoritmu

1. Vytvořit nový bod pomocí matematické funkce ve vzdálenosti od předchozího, kterou uvádí aktuální funkce. Vypočítat nové souřadnice lze s určitou odchylkou při použití numerické metody.
2. Provedení náhodné korekce obou souřadnic pomocí vztahu 6.2. Náhodný parametr vygenerovat pomocí *rovnoměrného* rozložení a pro každou souřadnici vytvořit nové náhodné číslo.

3. Pokud není zásobníková struktura prázdná, vybere se vrchol zásobníků (*rodičovská funkce*) a provede se korekce směru aktuální funkce pomocí vzorce 6.1.
4. Nastavení do předchozího *rodičovského* bodu odkaz na nově vzniklý *synovský* bod. *Rodič* je předchůdce aktuálního bodu vytvořený stejnou funkcí. Pokud nově vzniklý bod je prvním vygenerovaným od dané funkce, jeho *rodič* je poslední bod vzniklý funkcí, která se nachází na vrcholu zásobníkové struktury.

7.3.4 Vyhodnocení viditelnosti bodu – 3. bod algoritmu

1. Spočítání, zda se aktuální bod nachází uvnitř výše definovaného obdélníku.
2. Zjištění, zda se aktuální bod nenachází ve *foveální* oblasti. Spočítáním vzdálenosti mezi novým bodem a středem *foveální* části nesmí vzdálenost být menší než předem definovaný poloměr kruhu.

7.3.5 Vyhodnocení vytvoření větvení cévy – bod 3.A algoritmu

1. Zjištění, zda se provede větvení cév v daném bodě. Posuzuje, zda již pro danou cévu proběhl maximální počet větvení následně na základě pravděpodobnosti větvení. Pokud danou cévu popisuje matematická funkce, posuzuje se pozice nového bodu pomocí chování popsaného v 6.5.

Seznam popisující chování při vytvoření nové cévy:

2. Pro nově vzniklou cévu inkrementovat parametr aktuální hloubky větvení popisující 6.4.
3. Nastavení 0 % pravděpodobnosti pro větvení u nové cévní větve.
4. Pokud funkci definuje matematický model, tak se odebere první bod z fronty popisující větvení. V případě prázdné fronty se větvení neprovede.
5. Provedení vynulování aktuální pravděpodobnosti na větvení u příštího bodu, jestliže funkci vytváří vektor.
6. Vypočtení šířky nové cévy pomocí postupu 6.3.1.

Seznam popisující chování při nevytvoření nové cévy:

7. Provedení přičtení 10 % k aktuální pravděpodobnosti na větvení u příštího bodu, jestliže funkci vytváří vektor.

7.3.6 Vytvoření nové matematické funkce při větvení cév – bod 3.A algoritmu

1. Vložení aktuálně používané funkce na vrchol zásobníkové struktury.
2. Nastavení velikosti kroku funkce na hodnotu 3.
3. Vytvoření maximálního počtu větvení u nové cévy pomocí hodnot z 6.4. Určení popisují pravděpodobnosti uvedené v tabulkách ve stejné podkapitole. Použití určité tabulky definuje vstupní kvadrant cévního řečiště.
4. Vypočítání směru nového vektoru. Pomocí vztahu 6.5 se vypočítá jaký úhel bude mezi cévami na základě vlastnosti šířky nové cévy získané při vyhodnocení větvení.

5. Zkopírování *rodičovského* (uložený na vrcholu zásobníkové struktury) vektoru pokud je definovaný. V případě, že *rodiče* určuje matematická funkce, vytvoří se derivace funkce v daném bodě a přetvoří se na vektor pomocí techniky uvedené v 6.2.
6. Otočení získaného vektoru z předchozího bodu o dříve spočítaný úhel pomocí vzorce 6.3 a provedení normalizace předpisem 6.4.
7. Vypočítání úhlu pro korekci *rodičovské* funkce pokud je definovaná vektorem. Ty, které definuje matematická funkce, mají ve staticky zpracované křivce zakomponovanou korekci. Použije se doplněk hodnoty šířky cévy pro novou větev.
8. Otočení *rodičovského* vektoru o úhel z předchozího kroku. Transformace je provedena do opačného směru než u nově vzniklého vektoru.
9. Nastavení nově vytvořené funkce na aktuálně požívanou.

7.3.7 Nastavení aktuálního bodu na průchozí – bod 3.B algoritmu

1. Uložení hodnoty 1 do atributu poměr šířek cévy v daném bodu.

7.3.8 Výpočet kompatibilních atributů a případná transformace cévního řečiště – 6. bod algoritmu

1. Transformování aktuálně počítaného krevního řečiště do požadovaného kvadrantu. Přetvoření se provádí pouze z *I* do *IV* a z *II* do *III* kvadrantu. Realizuje se vynásobením x složky všech získaných bodů stromové struktury s hodnotou -1 .
2. Dopočítání všech atributů pro celou stromovou strukturu bodů. Předpis pro vyčíslení jednotlivých vlastností nutných pro kompatibilitu v systému, které nebylo potřeba používat při běhu algoritmu.
 - *PositionX* a *positionY*: Transformace z aktuálního souřadného systému do systému používaném v generátoru pomocí vztahů 7.1 a 7.2.
 - *PrevDistance*: Vypočítat *Euklidovskou* vzdálenost mezi aktuálním a jeho předchozím bodem.
 - *DirectionAngle*: Vyhodnotit goniometrickou funkci *arkus tangens* v pomyslném trojúhelníku, který vytváří aktuální a předchozí bod. Každý se nachází na jiném konci přepony. Nutné počítat ve stupních.
 - *Width*: Vyčíslit atribut pomocí vztahu 4.6.
 - *CurveWeight1*, *curveWeight1* a *CurveEndDeviation*: Náhodně určené atributy. Možné hodnoty popisuje podkapitola 7.1.
 - *PrevPoint*: Vytvoření vazby mezi *synovským* a *rodičovským* uzlem ze strany potomka.

Kapitola 8

Implementace algoritmu

Kapitola popisuje implementaci algoritmu z předchozí kapitoly. Rozšiřovaný generátor byl sepsaný v objektově orientovaném paradigmatu a v jazyce *C++*, proto bylo toto rozšíření vytvořeno ve stejném paradigmatu i programovacím jazyce. Implementace zachovává uživatelské rozhraní, které má podobu konzolové aplikace s konfiguračním souborem pro vstupní parametry.

Implementace obsahuje tři mírné odlišnosti od navrhnutého algoritmu. Odlišnosti byly zavedeny z důvodu efektivnější implementace. První odlišnost představuje výpočet všech kvadrantů syntetické sítě probíhající v *I* kvadrantu a je následně zrcadlen do ostatních částí. Druhá nepřesnost představuje použití zásobníků. Ten se používá specifickým způsobem, protože ho není nutné implementovat a lze použít zásobník volání operačního systému. Pomocí jednoduché rekurze se dosáhne stejného efektu. Poslední odlišnost představuje použití numerické metody pro výpočet dalšího bodu nad funkcí. Pokud daný bod musí být v určité vzdálenosti od jiného, je vhodné použít numerickou metodu pro získání souřadnice tohoto bodu. Byla použita metoda *binárního půlení*.

Podkapitola 8.1 uvádí, jaká třída nebo skupina tříd implementuje určitý krok algoritmu. Popisuje nutné atributy pro chod algoritmu a další implementační zajímavosti.

Zakomponování nové implementace představovalo i zásah do stávajícího programu. Podkapitola 8.2 popisuje, jaké změny bylo třeba udělat.

8.1 Použití konkrétních tříd k jednotlivým bodům navrhnutého algoritmu

Následující část popisuje použité třídy a mechanismy realizující určité části algoritmu. Společná část implementace představuje hlavičkový soubor `realisticExtensionConstant.h` a třídu `randomGenerator`. Hlavičkový soubor obsahuje použité konstanty napříč celou implementací například maximální i minimální hodnoty v souřadném systému či matematické konstanty. Třída `randomGenerator` je implementována návrhovým vzorem *singleton* a generuje náhodná čísla pomocí *rovnoměrného* rozdělení. Obsahuje metody na vytváření čísla ve specifickém rozsahu nebo v intervalu $\langle 0; 1 \rangle$.

Pro vytváření vektorů je definovaná třída `vector`, která umožňuje vytváření a přenášení vektorů.

8.1.1 Reprezentace matematické funkce

Informace a vlastnosti použité matematické funkce udržuje třída `mathSet`. Obsahuje metody k získání funkční hodnoty v aktuálním bodě `x` `getMathValue`, korekci parametru specifikující matematickou funkci `corectParam`. Uchování funkce reprezentuje atribut `mathfunction`, který je datového typu ukazatel na funkci. Pokud bude uložen směr cévy pomocí vektoru třída, obsahuje atribut `directionVector`.

Vybrání specifické funkce je provedeno v konstruktoru této třídy. Je nutné mít nadefinované funkce z kapitol 5 a 6. Na vybranou funkci se pouze nastaví ukazatel a do atributu `param` se uloží hodnota vygenerovaného parametru.

Třída počítá derivaci funkce v metodě `getDirectionFunction`. Vrací hodnotu vektoru v případě zadané funkce vektorem, jinak je navržena výše definovaná derivace a převedená na vektor.

8.1.2 Logika rozhodující o větvení cév

Logiku větvení představuje třída `bifurcationLogic`. Instance této třídy rozhoduje, zda se bude daná céva větvit. Uchovává informace v attributech o maximální úrovni zanoření `bifurcationMaxLevel`, o aktuální úrovni zanoření `bifuracionLevel`. Udržuje informace o procentuální šanci na větvení. Pokud je daná céva reprezentována matematickou funkcí, obsahuje *pole* souřadnic, ve kterých se bude provádět větvení.

Třída obsahuje dva konstruktory. První slouží pro cévu, která je definovaná pomocí vektoru. Druhý konstruktor vytváří *pole*, ve kterém jsou uchovány hodnoty pro větvení dané cévy.

Stěžejní metoda nad touto třídou je `calculateBifurcationBranch`, která určuje zda v aktuálním místě se provede větvení či nikoliv.

8.1.3 Interakce s matematickými modely a jejich řízení

Nosný prvek celého algoritmu představuje třída `functionControler`. Ta zprostředkovává všechny operace s matematickou funkcí pomocí instance třídy `mathSet`. Udržuje hodnoty větvení v instanci třídy `bifurcationLogic`. `functionControler` popisuje chování jedné konkrétní cévy.

Třída uchovává v attributech pozici počátku i posledního spočítaného bodu, odkaz na instanci třídy `mathSet`, odkaz na instanci rodičovské funkce, z důvodu korekcí aktuálně používané funkce. Obsahuje odkaz na instanci třídy `bifurcationLogic` pro řízení větvení.

V Attributech `startCoordX` a `startCoordY` je uchována informace o vzniku této instance. Každá instance začíná ve svém souřadném systému v místě $[0;0]$ a při výsledném výpočtu bodu jsou k atributům `relativeActualCoordX` a `relativeActualCoordY` připočteny tyto hodnoty posunutí.

Třída obsahuje mnoho metod typu *get* a *set*. Stěžejní metoda pro výpočet pozice nového bodu je `doStep`. Ta počítá pozici nového bodu od původního ve vzdálenosti *delta*. Výpočet je proveden numerickou metodou binárního půlení, kdy na polovinu intervalu je nastavena sečtená hodnota aktuální pozice a *delty*. Po takto vytvořeném bodu se provede jeho náhodné zachvění v osách *x* i *y* pomocí funkce `randomCorection`.

Neméně důležitou metodou představuje `calculateCorection`. Ta upravuje parametry aktuální funkce na základě hodnot funkce *rodičovské*.

8.1.4 Řízení chodu algoritmu

O chod celého algoritmu se stará třída `vascular`. Instanci třídy vytváří třída `createVessel`. Pro každou část krevního řečiště je vytvořena nová instance.

Třída v atributech udržuje informace o počátečním bodě, který se nachází v *optickém disku*. Obsahuje ukazatel na *kořenový* bod celé stromové struktury konkrétní cévy.

Třídní metoda `makeTreeStructure` generuje nové cévní řečiště. V této metodě je inicializována třída `functionController` představující cévu vycházející z optického disku. Pomocí interní funkce `generateVascularBranch` je generována céva instance `functionController`.

Interní metoda `generateVascularBranch` řídí volání metod tříd `functionController` a `bifurcationLogic`. Funkce generuje nové body, dokud hodnoty vypočítané metodou `doStep` jsou stále v předem definovaných mezích. Pokud je provedeno větvení dané cévy, jsou vytvořeny nové instance tříd `functionController` a `bifurcationLogic` a s těmito hodnotami je volána znovu funkce `generateVascularBranch`. Tímto způsobem je využíván zásobník volání jako zásobníková struktura v definici algoritmu.

V této třídě probíhá i propojení aktuálního bodu s nově vzniklým přes ukazatele. Jednotlivé body větvení představují instance třídy `vascularPoint`. Ty uchovávají informaci o své poloze a relativní šířce cévy v atributech `coordX`, `coordY` a `bifurcationRatio`. Pro vytvoření stromové struktury třída obsahuje ukazatele na své potomky v atributu `leftPointer` a `rightPointer`.

8.1.5 Smazání nerealistických prvků

Při generování cévního řečiště je možné, že se dvě stejné cévy překříží u existujících sítnic toto chování neexistuje. Takové chování bylo vyzporováno až při malých šířkách cév. Třída `deleteCrossingVascular` představuje řešení tohoto problému. Tato třída vytváří masku `bool` hodnot, které značí, zda se na aktuálním místě nachází céva či nikoliv.

Vygenerované cévní řečiště se prochází od nejširší cévy k nejslabší. Každá céva je zaznačena do vytvořené boolovské *masky*. Pokud je již v masce uložena informace o tom, že tímto bodem prochází céva, jedná se o situaci, kdy se dané cévy kříží, a aktuální céva bude smazána. Daná céva bude méně významnou, protože zaznačená céva byla dříve vybrána, tedy je širší.

Spojení dvou bodů do úsečky vytváří algoritmus *DDA*. Cévy, které jsou slabší než aktuálně zpracovávaná se ukládají do fronty. Zahájení výpočtu představuje veřejná metoda `clearCrossingController`, která přijímá *kořen* stromové struktury vygenerovaných bodů jako parametr.

8.1.6 Vypočítání kompatibilních atributů

Po vygenerování krevního řečiště a odstranění nevhodných větví je nutné spočítat kompatibilní atributy. Třída `adapter` vytváří řídicí prvek pro tuto transformaci. Veřejná metoda `adapteeTree` vytváří *kořen* nového stromu bodů krevního řečiště a volá třídní funkci `copyToBifurcationPointData`, která prochází rekurzivně stromovou strukturou a vytváří její kopii. Tu tvoří instance třídy `bifurcationPointLight`. Tato třída se stará o správný přepočet atributů. Obsahuje metody `mirrorCoordXInSection` a `mirrorCoordYInSection` na přenesení krevního řečiště do specifikovaného kvadrantu souřadného systému.

8.2 Zakomponování nového algoritmu do stávajícího programu

Propojení nové a staré části kódu proběhlo v konstruktoru třídy `vascularityLayer`. Tato třída se stará o celou vrstvu cévního řečiště. V konstruktoru této třídy se vytvářejí jednotlivé cévy. Konkrétní cévu vytváří třída `vessel`. V této třídě bylo nutné přetížít konstruktor, protože se v něm vytváří celé krevní řečiště pro danou cévu. V novém konstruktoru třídy `vessel` se ukládají pouze vstupní parametry cévy.

Generování nového cévního řečiště probíhá spuštěním metody `createBranch` ve třídě `createVessel`, která vrací ukazatel na *kořenový* uzel nového krevního řečiště. Uložení vygenerovaného řečiště proběhne použitím metody `setStartPoint` ze třídy `vessel`. Následně je řízení programu předáno zpátky původnímu generátoru.

Kapitola 9

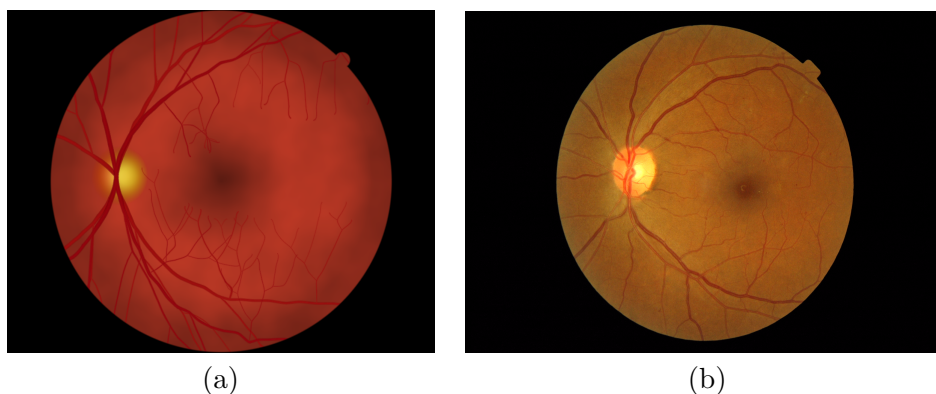
Testování větvení algoritmu

Kapitola se zabývá testováním vygenerované databáze syntetických snímků, které byly vytvořené pomocí nového algoritmu. Ta obsahuje celkem 1 000 snímků. Dílčí ověření návrhu jednotlivých komponent algoritmu proběhlo a následně bylo sepsáno v předchozích kapitolách 5 a 6.

Komplexní testování detekovalo body větvení u jednotlivých snímků. Tyto data byly následně agregovány do jediného snímku. Závěr testování porovnává tento agregovaný snímek s dříve pořízeným výsledkem, který byl pořízen stejnou metodou s databází vygenerovaných snímků předcházející metodou.

Detekci jednotlivých bodů větvení provedl program napsaný Ing. Martinem Presem [19]. Tento software v první části segreguje cévní řečiště ve vstupním obraze. Následně detekuje slepou skvrnu vygenerovaného snímku. V následující části je odhadnuta poloha *fovey* pomocí *anatomického* poznatku. Vzdálenost mezi optickým diskem a žlutou skvrnou je přibližně 2,5 násobek průměru slepé skvrny. V poslední části se naleznou body větvení u cévního řečiště. Jednotlivá místa jsou uložena pomocí *polárních* souřadnic. Počátek souřadného systému je nastaven na střed optického disku. Detekované body větvení agreguje program napsaný Ing. Lukášem Semerádem, který všechny body zanese do stejného snímku.

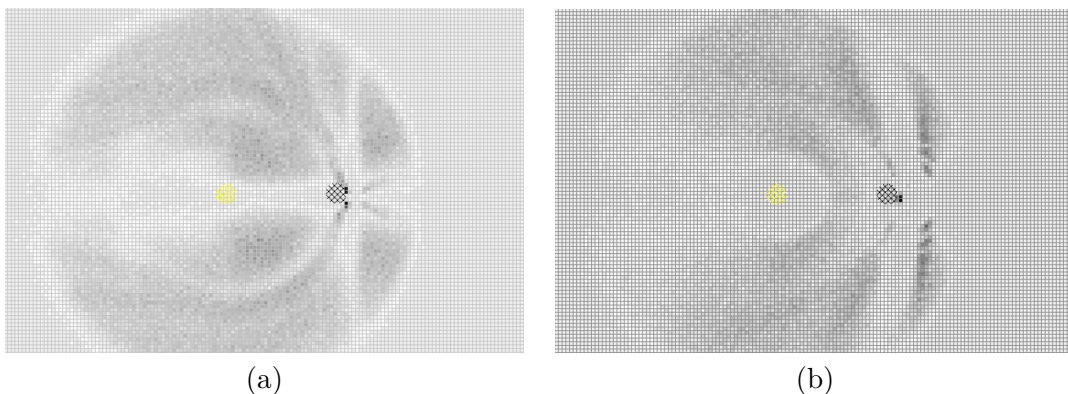
Výsledný snímek v porovnání s nasnímanou sítnicí zobrazuje obrázek 9.1. Vygenerované snímky obsahují srovnatelný počet cév s existujícími sítnicemi. Cévy směřují do stejných oblastí na snímcích s podobným zastoupením. Obsahuje ostřejší přechody a nedokonalosti v makulární oblasti cév. Dříve zvolené zbarvení snímku zvýrazňuje tyto nedokonalosti.



Obrázek 9.1: Vygenerovaná sítnice pomocí navrhnutého algoritmu (a), existující sítnice (b).

9.0.1 Porovnání agregovaných bodů větvení

Tato část se zabývá vyhodnocením bodů větvení u jednotlivých algoritmů generátoru. Detekce větvení proběhla na databázích o stejném počtu snímků. Databáze obsahují 1 000 snímků. Detekované výsledky jsou agregovány do jednoho výsledného obrazu. Výsledky zobrazují obrázky 9.2.



Obrázek 9.2: Detekované body větvení agregované do jednotlivého snímku. Výsledné obrazy z existující databáze snímků (a), z nově vytvořené databáze (b).

Oba snímky obsahují shodný obraz bifurkací v horní i spodní části sítnice. Tento jev způsobuje generování stejných cév, které se později transformují do jiné části obrazu. Výsledek nového algoritmu generátoru ukazuje méně bifurkačních bodů na viditelné části sítnice oka. Bifurkace nezasahují do oblasti fovey. Snímky nového algoritmu neobsahují mnoho větvení za oblastí fovey. Převážná část bifurkací se nachází nad touto oblastí.

Kapitola 10

Závěr

Cílem této práce bylo zdokonalit generování syntetických sítnic oka na základě vzorů chování, které vykazují skutečné nasnímané sítnice. Existující nástroj se podařilo rozšířit o modul, který vytváří krevní řečiště podle detekovaných vzorů chování.

Studium lidského oka a sítnice oka je zaznamenáno v druhé a třetí kapitole. Prozkoumání existujícího nástroje a jeho způsobu vytváření syntetického krevního řečiště dokumentuje kapitola čtyři. Kapitola pět a šest se věnuje přípravě návrhu algoritmu. V těchto kapitolách proběhlo statistické zpracování hlavních vzorů chování cévního řečiště. Matematickými programy byly vypočteny aproximační křivky, které mají představovat hlavní cévy řečiště. Nejlépe hodnocené a použité matematické funkce představuje *racionální* model, *Gompertzova* funkce a model *odmocniny* i *mocniny*. Vypočtené parametry těchto modelů se liší pro makulární a nasální část řečiště. Matematické funkce potvrdily shodnost horní a spodní části řečiště. Drobnější cévy byly následně popsány pomocí matematických vektorů. Vytváření větvení cév bylo popsáno pomocí transformací ve dvourozměrné dimenzi. Dále byl změřen a empiricky upraven úhel, který spolu svírají dvě cévy. Poslední sledovaný vzor chování představovala hloubka větvení cév. V sedmé kapitole bylo stanoveno vstupní a výstupní rozhraní algoritmu a následně i jeho návrh. Ten se zakládá na dříve zjištěných informacích. Hlavní stavební blok představují matematické funkce a vektory. Algoritmus potřebuje ke svému správnému fungování datovou strukturu *zásobník*, ve které jsou uloženy funkce. Osmá kapitola popisuje implementaci dříve navrhnutého algoritmu. Byl použit programový zásobník volání. Program je sepsán rekurzivně. Následně byla vytvořena sada syntetických sítnic. Další kapitola dokumentuje proběhnuté testování, ve kterém se měřilo rozmístění bifurkačních bodů cév. Všechny body zadání bakalářské práce byly splněny.

V práci bych chtěl pokračovat detekcí dalších vzorů chování, které by upřesnily popis skutečných sítnic oka. Věnoval bych se rozdělení pozorovatelné sítnice na více částí a provedl bych pozorování nad menšími oblastmi sítnice. Komplexnější rozšíření by zahrnovalo navrhnout modul pro generování poškozeného cévního řečiště různými chorobami či vadami.

Literatura

- [1] Beneš, P.: *Přístroje pro optometrii a oftalmologii*. Brno: NCONZO, první vydání, 2015, ISBN 978-80-7013-577-8.
- [2] Biasi, M. D.: *Automatic Generation of Retinal Fundus Image Phantoms: Vascular Regions*. Diplomová práce, Università degli Studi di Padova, Dipartimento di Ingegneria dell'Informazione, Padova, 2014, vedoucí práce Prof. A. Ruggeri.
- [3] Bonaldi, L.; Mentia, E.; Ballerinib, L.; aj.: *Automatic Generation of Synthetic Retinal Fundus Images*. *Procedia Computer Science*, ročník 90, č. 6, 2016: str. 54–60, ISSN 18770509.
- [4] Cassin, B.; Solomon, S.: *Dictionary of Eye Terminology*. Gainesville, Florida: Triad Pub Co, první vydání, 1990, ISBN 978-0937404331.
- [5] Chris: *Bézier curve*. 2014, [Online; navštíveno 07.01.2019].
URL https://en.wikipedia.org/wiki/B%C3%A9zier_curve#/media/File:B%C3%A9zier_3_big.svg
- [6] Decencièere, E.; Zhang, X.; Cazuguel, G.; aj.: *Feedback on a publicly distributed database: the Messidor database*. *Image Analysis & Stereology*, ročník 33, č. 3, Srpen 2014: s. 231–234, ISSN 1854-5165, doi:10.5566/ias.1155.
URL <http://www.ias-iss.org/ojs/IAS/article/view/1155>
- [7] Desmos Graphing Calculator: *Desmos Graphing Calculator*. 2016, [Online; navštíveno 10.01.2019].
URL <https://www.desmos.com/calculator>
- [8] Drahanský, M.; Orság, F.; Doležel, M.: *Biometrie*. Brno: M. Drahanský, první vydání, 2011, ISBN 978-80-254-8979-6.
- [9] Fiorini, S.: *Automatic generation of Retinal Fundus Image Phantoms: non-vascular regions*. Diplomová práce, Università degli Studi di Padova, Facoltà di Ingegneria, Padova, 2014, vedoucí práce Prof. Alfredo Ruggeri.
- [10] Foley, J. D.; Dam, A.; Feiner, S. K.; aj.: *Computer Graphics - principles and practice*. Addison-Wesley, 1993, ISBN 0-201-12110-7.
- [11] Garrity, J.; MacMillan, W.; MacMillan, B.: *Structure and Function of the Eyes*. 2016, [Online; navštíveno 05.01.2019].
URL <https://www.msmanuals.com/home/eye-disorders/biology-of-the-eyes/structure-and-function-of-the-eyes>

- [12] Horridus, C.: *Bresenham's line algorithm*. 2007, [Online; navštíveno 07.01.2019].
URL https://en.wikipedia.org/wiki/Bresenham%27s_line_algorithm#/media/File:Bresenham.svg
- [13] Hudák, R.; Kachlík, D.; kolektiv: *Memoria anatomie*. Praha: TRITON, třetí vydání, 2015, ISBN 978-80-7387-959-4.
- [14] Hyams Development: *CurveExpert Basic*. 2016, [Online; navštíveno 10.01.2019].
URL <https://www.curveexpert.net/>
- [15] Hycl, J.; Trybučková, L.: *Atlas oftalmologie*. Praha: Triton, druhé vydání, 2008, ISBN 978-80-7387-160-4.
- [16] Irsch, K.; Guyton, D. L.: *Anatomy of Eyes*. Boston, MA: Springer US, 2009, ISBN 978-0-387-73003-5, s. 11–16.
- [17] Kolb, H.; Nelson, R.; Fernandez, E.; aj.: *Webvision*. Salt Lake City (Utah): University of Utah Health Sciences Center, první vydání, 2007, ISBN 978-1-932811087.
- [18] Lücking, W.; Lücking-Lenkitsch, S.: *Funduskamera*. [Online; navštíveno 10.01.2019].
URL <http://www.augenarzt-luecking.de/index.php?id=25>
- [19] Pres, M.: *Lokalizace bifurkací ve snímcích sítnice*. Diplomová práce, Vysoké učení technické v Brně, Fakulta informačních technologií, Brno, 2016, vedoucí práce Ing. Lukáš Semerád.
- [20] Rak, R.; Matyáš, V.; Říha, Z.: *Biometrie a identita člověka ve forezních a komerčních aplikacích*. Praha: Grada, první vydání, 2008, ISBN 978-80-247-2365-5.
- [21] Reichl, J.: *Encyklopedie fyziky*. [Online; navštíveno 08.01.2019].
URL http://fyzika.jreichl.com/data/optika/32_oko_soubory/image002.jpg
- [22] Reichl, J.: *Encyklopedie fyziky*. [Online; navštíveno 08.01.2019].
URL http://fyzika.jreichl.com/data/optika/32_oko_soubory/image003.jpg
- [23] Rhcastilhos: *Schematic diagram of the human eye in czech*. 2009, [Online; navštíveno 05.01.2019].
URL https://commons.wikimedia.org/wiki/File:Schematic_diagram_of_the_human_eye_cs.svg
- [24] SFML: *Simple and Fast Multimedia Library*. [Online; navštíveno 05.01.2019].
URL <https://www.sfml-dev.org/>
- [25] Silbernagl, S.; Despopoulos, A.: *Atlas fyziologie člověka*. Praha: Avicenum, první vydání, 1984, ISBN 08-026-84.
- [26] Sinělnikov, R. D.: *Atlas anatomie člověka*. Praha: Avicenum, třetí vydání, 1982, ISBN 08-041-80.
- [27] SPIRIT MEDICAL spol. s r.o.: *Oftalmoskop Professional*. 2009, [Online; navštíveno 10.01.2019].
URL <http://www.spiritmedical.cz/uploads/thumbnails/thumbnail-1280960-professional-ophthalmoscope.jpg>

- [28] SPIRIT MEDICAL spol. s r.o.: *Vantage Plus*. 2009, [Online; navštíveno 10.01.2019].
URL <http://www.spiritmedical.cz/uploads/thumbnails/thumbnail-1280960-vantage.jpg>
- [29] STRaDe: *Security Technology Research and Development*. [Online; navštíveno 05.01.2019].
URL <https://strade.fit.vutbr.cz/>
- [30] Tretter, Z.: *Generování syntetických obrazů sítnic oka*. Diplomová práce, Vysoké učení technické v Brně, Fakulta informačních technologií, Brno, 2017, vedoucí práce doc. Ing. Martin Dražanský, Ph.D.
- [31] VAMPIRE: *Vascular Assessment and Measurement Platform for Images of the REtina*. [Online; navštíveno 05.01.2019].
URL <https://vampire.computing.dundee.ac.uk/>
- [32] Videni.cz: *Anatomie lidského oka*. 2010, aktualizováno 2014, [Online; navštíveno 05.01.2019].
URL <http://www.videni.cz/oko/32-anatomie-oka>
- [33] Xuru: *RegressionTools: The Program*. 2016, [Online; navštíveno 10.01.2019].
URL <http://www.xuru.org/rt/RegressionTools.asp>
- [34] Yang, J.; Park, D. S.; Yoon, S.; aj.: *Machine learning and biometrics*. London: IntechOpen, 2018, ISBN 978-1-78923-590-6.
- [35] Zamir, M.; Medeiros, J. A.; Cunningham, T. K.: *Arterial bifurcations in the human retina*. The Journal of general physiology, ročník 74, č. 4, 1979: str. 537–548, ISSN 1540-7748.
- [36] Čihák, R.: *Anatomie 3*. Praha: Grada, druhé vydání, 2004, ISBN 978-80-247-1132-4.
- [37] Žára, J.: *Moderní počítačová grafika*. Brno: Computer Press, vyd 1 vydání, 2004, ISBN 80-251-0454-0.