

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ROZPOZNÁVÁNÍ PODLE SÍTNICE OKA

DIPLOMOVÁ PRÁCE

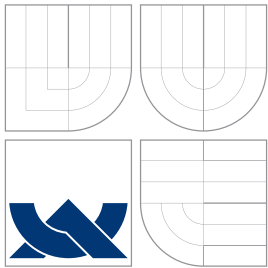
MASTER'S THESIS

AUTOR PRÁCE

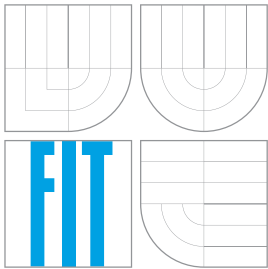
AUTHOR

Bc. JAROSLAV STRUŽKA

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ROZPOZNÁVÁNÍ PODLE SÍTNICE OKA

RECOGNITION BY RETINA

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. JAROSLAV STRUŽKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ KRAJÍČEK

BRNO 2010

Abstrakt

V dnešní době je čím dál víc kladen důraz na bezpečnost zejména na bezpečnost přístupu k informacím. Tím se do popředí dostává biometrie, která se zabývá charakteristickými lidskými vlastnostmi a charakteristickým chováním. Existuje mnoho biometrických vzorů, které můžeme zkoumat. Jedním z nejvíce unikátních je lidská sítnice. Tato diplomová práce obsahuje informace o lidském oku a chorobách, které jej mohou postihnout a tím znehodnotit snímek sítnice. Dále je provedena analýza a návrh aplikace pro rozpoznávání sítnice, který je založen na aplikaci obrazových filtrů. Zejména pak filtrů pro detekci hran, prahování a skeletonizaci. Dále jsou zde uvedeny dosažené výsledky fungování systému.

Abstract

Nowadays, it is more important to focus on security, especially on security of acquiring information. Therethrough biometrics gains ground. Biometrics comprises methods for uniquely recognizing humans based upon one or more intrinsic physical or behavioral traits. There are lot of biometric models, which can be examined. One of the most unique is human retina. This master's thesis contains information about human eye and diseases, that could possibly affect it. Some diseases can devaluate the image of human retina. Next step is analysis and design of application for recognising of retina, which is based on application of image filters. Especially filters for edge detection, thresholding and skeletonization. In final part we issue the outputs of working system.

Klíčová slova

Sítnice, oko, fundus kamera, detekce hran, odstranění šumu, identifikace, verifikace, prahování

Keywords

Retina, eye, fundus camera, edge detection, noise reduction, identification, verification, thresholding.

Citace

Jaroslav Stružka: Rozpoznávání podle sítnice oka, diplomová práce, Brno, FIT VUT v Brně, 2010

Rozpoznávání podle sítnice oka

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana ing. Jiřího Krajíčka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Jaroslav Stružka

24. května 2010

Poděkování

Děkuji panu Ing. Jiřímu Krajíčkovi za vedení a pomoc při tvorbě této diplomové práce. Dále děkuji všem studentům, kteří byly ochotni poskytnout obraz své oční sítnice pro nasnínání do databáze.

© Jaroslav Stružka, 2010.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	3
1.1 Struktura práce	3
2 Oko	5
2.1 Lidské oko	5
2.1.1 Zornice	6
2.1.2 Rohovka	6
2.1.3 Duhovka	7
2.1.4 Čočka	8
2.1.5 Sklivec	8
2.1.6 Sítnice	9
2.2 Oční choroby	11
2.2.1 Makulární degenerace	11
2.2.2 Diabetická retinopatie	12
2.2.3 Glaukom	13
2.2.4 Toxoplazmóza	14
2.2.5 Nádorová onemocnění	15
2.3 Fundus kamery	16
2.3.1 Obecný popis fundus kamery	16
2.3.2 Non-mydriatická funduskamera Canon CR-1	17
2.4 Biometrie a požadavky na systém	17
2.4.1 Univerzalita	17
2.4.2 Unikátnost	18
2.4.3 Konstantnost	18
2.4.4 Získatelnost	18
2.4.5 Výkonnost	18
2.4.6 Akceptovatelnost	18
2.4.7 Ošálitelnost	18
2.4.8 Spolehlivost	19
2.4.9 Statistické koeficienty	19
2.5 Požadavky na systém	22
2.5.1 Nasnímání, extrahování, identifikace	22
2.6 Shrnutí	23
3 Analýza a návrh systému	24
3.1 UML	24
3.1.1 Diagram případů použití	24
3.1.2 Diagram tříd	25

3.2	Analýza	26
3.2.1	Diagram případů užití	26
3.3	Návrh	26
3.3.1	Diagram tříd	26
4	Obrazové filtry	29
4.1	Detekce hran	30
4.2	Odstranění šumu	34
4.2.1	Lineární metody vyhlazování	34
4.2.2	Nelineární metody vyhlazování	34
4.3	Ostatní filtry a funkce	35
4.3.1	Grayscale	35
4.3.2	Prahování	35
4.3.3	Skeletonizace	35
5	Implementace	37
5.1	Použité nástroje	37
5.2	OpenCV	37
5.2.1	cvCvtColor	37
5.2.2	cvSmooth	38
5.2.3	cvThreshold	38
5.2.4	cvAdaptiveThreshold	38
5.2.5	cvCanny	38
5.3	Databáze	39
5.3.1	MySQL	39
5.3.2	XML	39
5.3.3	TXT	40
5.4	Popis metod	40
5.4.1	Kontrola parametrů	41
5.4.2	Získání šablony	41
5.5	Parametry	45
5.6	Shrnutí	46
6	Experimenty a testování	47
6.1	Zařízení a získ vzorků	47
6.2	Výsledky	47
7	Závěr	52
A	Obsah CD	56

Kapitola 1

Úvod

Každý člověk má několik znaků, na jejichž jedinečnosti a výjimečnosti (unikátnosti) lze provádět rozpoznávání (identifikace/verifikace). Ačkoliv některé z těchto vlastností lze oklamat, ať jde o umělé prsty (otisky prstů, bez testování živosti) či speciálně upravené kontaktní čočky (rozpoznávání podle duhovky), jsou tyto metody neustále využívány k určení identity osob na letištích apod. Existují ovšem biometrické vlastnosti, které je velmi obtížné nasimulovat. Jde například o sítnici oka. Ta je u každé osoby tvořena jedinečným cévním řečištěm a lze ji nejen využít k neinvazivnímu pozorování cév, k diagnostice očních chorob, ale také k bezpečnému rozpoznání identity dané osoby (i v případě jednovaječných dvojčat jsou cévní řečiště sítnice rozdílné).

Z retinálního snímku lze extrahovat různé vlastnosti 2D struktur, které se na sítnici vyskytují. Například polohu žluté skvrny, optického disku, či tvar cévního řečiště. Při zjištění anomálií lze na jejich základě odhalit oční choroby jako makulární degeneraci, diabetickou retinopatii, hypertenzi a glaukom.

Existuje několik metod a postupů na extrahování důležitých prvků ze sítnice. Práce [12] se zabývá segmentací cévního řečiště na základě aplikací speciálních 2D filtrů. Dále je to například [9] jako součást nástroje odhalující diabetickou retinopatii. Práce je založená na aplikaci Cannyho operátoru na detekci hran. Jako dalšího zástupce můžeme jmenovat [15], který využívá LoG operátor (Laplacian of Gaussian).

1.1 Struktura práce

Tato diplomová práce se zabývá návrhem metody pro získání vzoru ze sítnice (extrakce krevního řečiště oka). Dále pak metodou pro rozpoznávání dle získaného vzoru pro účely biometrie – jednoznačné rozpoznávání identity osoby (identifikaci a verifikaci). Vstupní snímky jsou získávány pomocí zdravotnického přístroje – non-mydriatickou funduskamerou Canon CR-1.

Tato práce spadá svým tématem do dvou oblastí. První oblast je dána potřebou vývoje specifického software. S tím souvisí všechny fáze, které se v životním cyklu software vyskytují (sběr požadavků, fáze návrhu, implementace, testování). Druhá oblast je dána svým zaměřením, potřebami v oblasti biometrie, konkrétně rozpoznávání podle sítnice oka (s tím souvisí lékařské pozadí kolem sítnice oka, dále pak zejména požadavky biometrie na rozpoznávání a testování).

Podle výše popsaných oblastí je práce rozdělena do několika kapitol, v nichž jsou postupně popsány jednotlivé části.

V následující kapitole se budeme zabývat úvodem do struktury oka (zejména sítnice), chorobami postihující oko a zařízením ke snímání sítnice oka. Dále pak požadavky biometrie a souhrnným seznamem požadavků pro návrh software k extrakci a rozpoznávání.

Ve třetí kapitole se budeme zabývat analýzou a návrhem celkové aplikace pomocí modelovacího jazyka UML. Návrhem funkčních požadavků (těla metod) se pak zabývají následující dvě kapitoly.

Ve čtvrté kapitole se budeme zabývat návrhem metod (posloupností filtrů, algoritmů), které mohou být využity při zpracování (extrakci) vzorů ze vstupních snímků.

V páté kapitole se budeme zabývat provedenou implementací. Seznámíme se s problémy, na které bylo v průběhu implementace naraženo, a vlastnostmi implementačního jazyku a použitých knihoven.

V šesté kapitole se zaměříme na testování aplikace pro praktické, reálné využití. Budeme diskutovat výsledky, kterých bylo dosaženo v rámci testování aplikace. Zejména se zaměříme na DET a ROC křivky.

V závěrečné kapitole zhodnotíme dosažené výsledky a návrhy na možnosti praktického využití aplikace a jejího přizpůsobení/rozšíření.

Kapitola 2

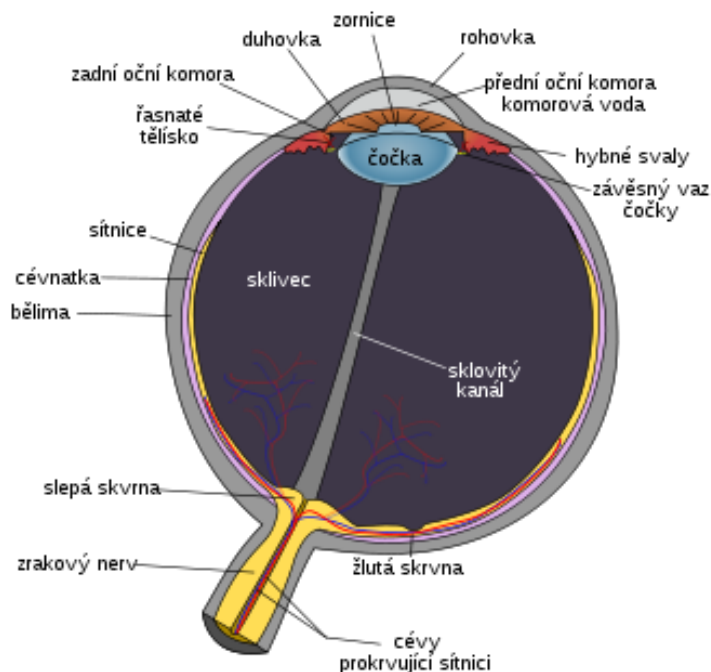
Oko

Oko je párový smyslový orgán reagující na světlo (fotoreceptor), tedy zajišťující zrak. V průběhu evoluce došlo k výraznému rozvoji od světločivných orgánů reagujících pouze na existenci světla, až po oko jednoduché u obratlovců (včetně člověka) a hlavonožců a oko složené např. u členovců.

Mnoho živočichů (většina savců, ptáků, plazů a některých ryb) má oči umístěné na přední straně hlavy, což jim umožňuje trojrozměrné binokulární vidění – jednotlivé obrázky z obou očí se skládají do jednoho, který vnímáme. Oproti tomu existuje monokulární vidění, kdy živočich vnímá současně dva rozdílné obrazy, jako např. zajíc nebo chameleon [14], [27].

2.1 Lidské oko

Lidské oko je párový orgán, který umožňuje člověku vidět. Struktura lidského oka, podrobněji viz. 2.1, se plně přizpůsobuje potřebě zaostřit paprsek světla na sítnici (retina). Všechny části oka, přes které paprsek světla prochází, jsou průhledné, aby co nejvíce zabraňovaly rozptylu dopadajícího světla. Rohovka (cornea) a čočka (lens) pomáhají paprsek světla spojit a zaostřit na zadní stěnu oka – sítnici. Toto světlo pak způsobuje chemické přeměny ve světločivných buňkách (tyčinky a čípky), které vysílají nervové impulsy zrakovým nervem (nervus opticus) do mozku [14], [28].



Obrázek 2.1: Schéma lidského oka

2.1.1 Zornice

Zornice je otvor umístěný uprostřed duhovky a umožňuje světlu projít na sítnici. Zdá se být černá, protože většina světla jí procházející je pohlcena oční tkání. Lidská zornice je kulatá, ale u některých druhů, jako například koček, je zornice štěrbinovitá. Obrázek zornice, tak jak je viděna z venku oka, přímo neodpovídá poloze a velikosti ve skutečnosti, protože je zkreslována rohovkou. Na vnitřní straně leží nápadná struktura, límeček, značící spojnicí embryonické zorniční membrány kryjící embryonickou zornici [7], [11], [29].

2.1.2 Rohovka

Rohovka je průhledná, kopulovitě zakřivená vrstva pokrývající přední část oční bulvy, to znamená duhovku, zornici a přední oční komoru. Rohovka se svou značnou optickou mohutností, ± 43 dioptrií, představuje zhruba dvě třetiny celkové optické mohutnosti oka a společně s čočkou, která má ± 20 dioptrií, se podílí na lomu světla přicházejícího do oka. Celková hodnota refrakce (lomivosti) oka 58,7 d je nižší proto, že je nutné odečíst sílu přední oční komory, která činí přibližně 4 dioptrie. Lidská rohovka je, tak jako u jiných primátů, složena z pěti vrstev. Tyto vrstvy směrem zvnějšku dovnitř jsou:

1. Rohovkový epitel
2. Bowmanova membrána
3. Rohovkové stroma

4. Descemetova membrána

5. Rohovkový endotel

Rohovkou neprocházejí žádné krevní cévy. Je naprosto čirá a má lesklý povrch. Vyživovaná je částečně výměškem slzného aparátu a částečně komorovou vodou z přední oční komory. Rohovka obsahuje mnoho nervových zakončení a tak je extrémně citlivá na dotyk, chemické nebo tepelné podráždění. Tato podráždění spouští rohovkový reflex, který uzavírá víčko a zvyšuje přítok slz. Průměr rohovky dospělého člověka je 11.5 mm . Uprostřed je rohovka $0.5 - 0.6\text{ mm}$ silná, u okraje pak $0.6 - 0.8\text{ mm}$ [7], [11], [30].

2.1.3 Duhovka

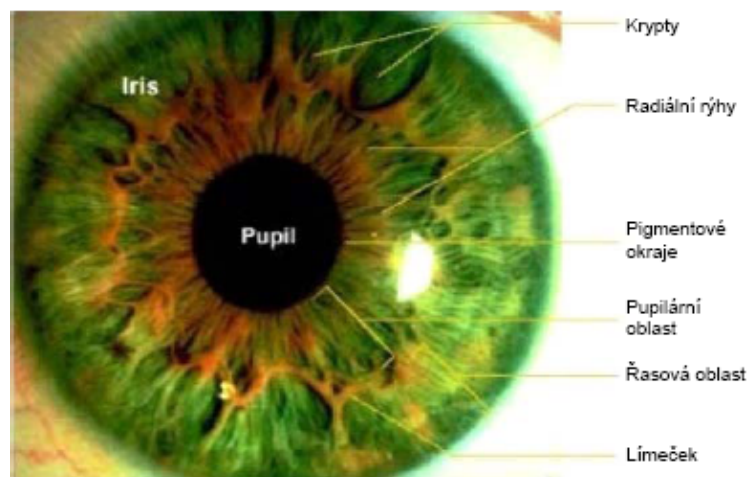
Duhovka je membrána v oku o velikosti cca 11 mm a je odpovědná za regulaci průměru a velikosti rohovky a množství světla, které se dostane na sítnici. V důsledku toho kolik světla má okem projít, svaly připojené k duhovce se stáhnou nebo roztáhnou. Tyto svaly jsou nejmenší v lidském těle. Čím více jsou svaly staženy, tím více světla projde. Barva očí je určena barvou duhovky.

Duhovka se skládá ze dvou vrstev: přední pigmentová fibrovasculární tkáň známá jako stroma, a pod ní se nachází pigmentové epitelální buňky.

Stroma je připojeno ke švěračovým svalům (sphincter pupillae), který ji zavírá, a skupině rozšiřovacích svalů (dilator pupillae), které ji otvírají. Zadní povrch je pokryt těžce pigmentovanou epitelální vrstvou, která má tloušťku dvou buněk. Za to přední povrch žádné epitelium nemá. Větší obsah pigmentu brání průchodu světla skrze duhovku na sítnici. Vnější okraj duhovky, známý též jako kořen, je připojen k očnímu bělmu a přednímu řasnatému tělísku. Duhovka a řasnaté tělísko spolu tvoří přední uveu. Přímo před kořenem duhovky je oblast označovaná jako trabekulární síťovina, skrze kterou je neustále odváděn oční mok z oka. Některé choroby duhovky mají často silný vliv na nitrooční tlak a nepřímo na vidění. Duhovka spolu s řasnatým tělískem poskytují méně důležitou cestu, skrze kterou je odváděn oční mok z oka.

Duhovka je rozdělena na dvě hlavní oblasti viz. 2.2 :

1. Pupilární zóna – vnitřní oblast duhovky, jejíž okraj tvaruje hranici se zornicí
2. Řasnatá zóna – je zbytek duhovky, která sahá od vnějšího okraje duhovky až přibližně do $\frac{2}{3}$ duhovky



Obrázek 2.2: Struktura duhovky oka

Límeček je oblast duhovky která odděluje pupilární část od řasnaté. Je typicky definovaná jako oblast, kde se překrývají svěračové a rozšiřující svaly. Límeček je základem vrstvy embryonické zornice.

Svalové buňky duhovky jsou u savců a obojživelníků hladké, ale u ptáků a plazů jsou tyto buňky příčně pruhované. Mnoho ryb nemá ani jeden typ svalstva, tudíž jejich duhovka se nestahuje ani neroztahuje, a proto zůstává velikost zornice stále stejná [3], [7], [11], [31].

2.1.4 Čočka

Čočka (lens crystallina, aquula) je transparentní, bikonvexní struktura v oku (přední strana je zploštělá). Spolu s rohovkou pomáhá lomit světlo. Změnou svého tvaru umožňuje čočka měnit ohniskovou vzdálenost oka. To umožňuje oku se zaměřit na objekty na různou vzdálenost, čímž se na sítnici dostává ostrý skutečný obraz objektu. Toto přizpůsobení čočky se nazývá akomodace. Je to velice podobné jako zaostřování na fotoaparátu.

U zrakového aparátu dospělého člověka je čočka přibližně 4 mm silná a v průměru má asi 10 mm. Je zavěšená na řasnatém tělisku pomocí zonulárních vláken. Čočka je velmi pružná. Vyklenutí čočky kontrolují ciliární svaly. Vnější povrch čočky je tvořen elastickým pouzdem.

Čočky jsou vytvořené především z průhledných proteinů zvaných krystaliny (asi ze 2 \ 3). Předpokládá se, že krystaliny jsou klíčem k otázce, jak zajistit správnou akomodaci čočky a zároveň propustit většinu světla dovnitř oka. Krystaliny jsou uspořádány do 20000 soustředných vrstev, s indexem lomu pro viditelné světlo od 1.406 v centrální vrstvě do 1.386 v řidších vrstvách okolo. Tento gradient ještě umocňuje optickou mohutnost čočky. Ta je přibližně 18 dioptrií, což tvoří zhruba třetinu celkové mohutnosti oka [7], [11], [32].

2.1.5 Sklivec

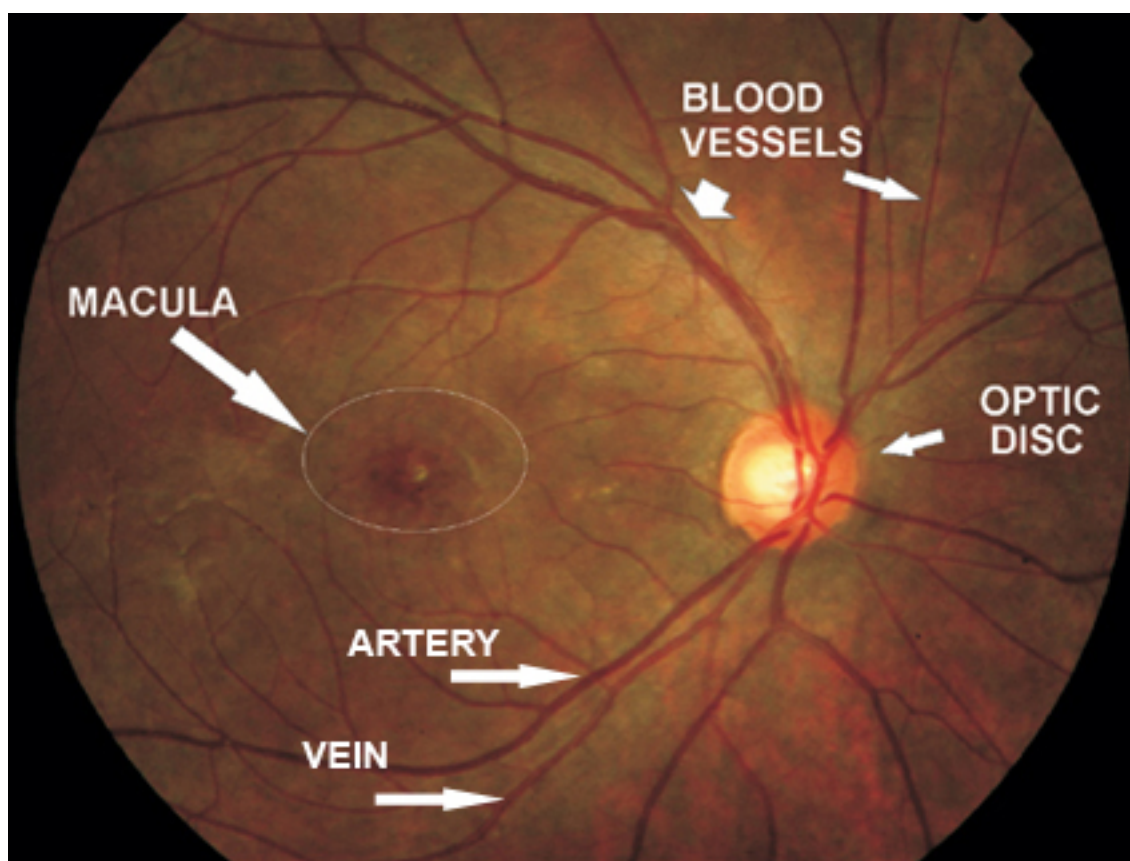
Oční dutinu vyplňuje polotekutý rosolovitý balón, nazývaný sklivec. Sklivec zevnitř naléhá na oční sítnici. Má velice podobné složení jako rohovka. Obsahuje několik málo buněk, často fagocyty, které se starají o odstraňování nechtěných buněčných zbytků ze zorného pole. Z 98 – 99 % je tvořen vodou (rohovka 75 %), dále obsahuje soli, cukry, vitrosin (typ

kolagenu), síť kolagenových vláken typu II s mukopolysacharidovou hyaluronickou kyselinou a také široké spektrum proteinu ve stopovém množství. Je překvapující, že i přes nízký obsah pevné hmoty, drží tvar oka. Sklivce má viskozitu 2x až 4x větší než voda, což mu dává želatinovou konzistenci. Jeho index lomu je 1.336.

Narozdíl od tekutiny v předních částech oka, která je neustále doplňována, hladina gelu ve sklivci je neustále stejná. Pokud se do něj dostane krev, buňky nebo nějaké vedlejší produkty při zánětu, zůstanou v něm do té doby, než jsou odstraněny chirurgicky. Pokud dojde k odtržení sklivce od sítnice, nazývá se tento jev jako odchlípnutí sklivce. Jak stárneme, sklivce řídne a může se zhroutit. To je častější u krátkozrakých očí. A tento jev může také nastat následkem zranění nebo při zánětech oka [7], [11] [33].

2.1.6 Sítnice

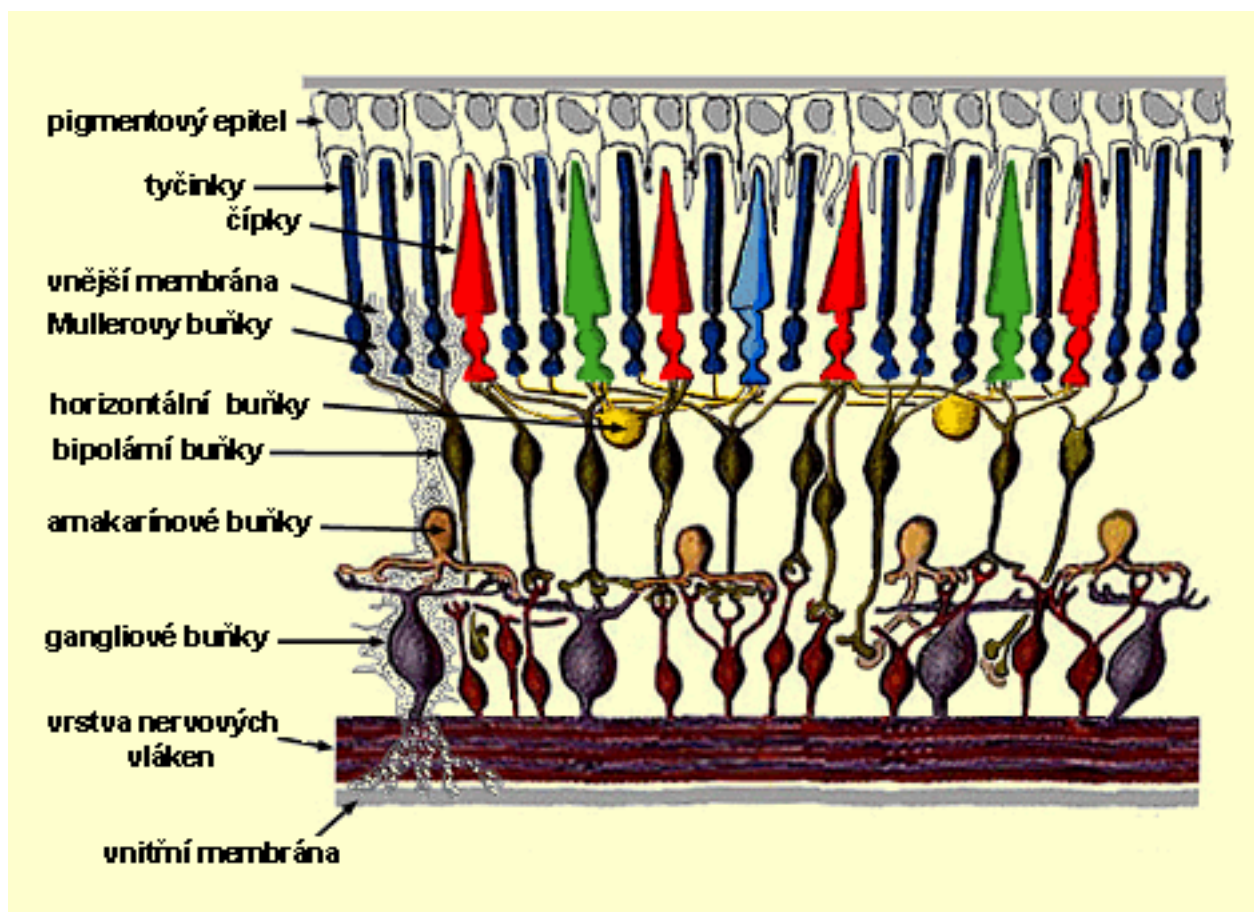
Sítnice je považována za část centrálního nervového systému (CNS) viz. 2.3. Jde o jedinou část CNS, kterou lze pozorovat neinvazivně. Sítnice je světlo-citlivá tkáň pokrývající vnitřní povrch oka. Optická soustava oka vytváří na sítnici obraz. Poskytuje člověku stejnou funkci jako film ve fotoaparátu. Světlo dopadající na sítnici vyvolává řadu chemických a elektrických událostí, které jsou spouští nervových impulzů. Ty jsou rozesílány do různých vizuálních center v mozku přes optický nerv.



Obrázek 2.3: Sítnice

Sítnice je komplexní, vrstvená struktura s několika vrstvami neuronů propojených sy-

napsemi viz. 2.4. Jediné neurony, které přímo reagují na světlo, jsou fotoreceptory. Ty se dělí na dva hlavní typy: tyčinky a čípky. Hlavní funkcí tyčinek je v přítmí poskytnout černobílé vidění, za to čípky podporují vidění za denního světla a vnímání barev. Třetím, velmi vzácným typem fotoreceptoru, jsou gangliové buňky, které jsou důležité pro odraz jasného denního světla.



Obrázek 2.4: Zjednodušené schéma řezu sítnicí

U dospělých lidí sítnice přibližně pokrývá 72 % vnitřní části oka. Celá plocha sítnice obsahuje asi 7 milionů čípků a 75 až 150 milionů tyčinek. Sítnice obsahuje tři zřetelné útvary viz. 2.3. Jsou to:

- Slepá skvrna (scotoma, optický disk) – jde o místo na sítnici, ve kterém se nenachází žádné fotoreceptory a tímto místem prochází veškeré oční nervy a cévy z oka. Má velikost asi 3 mm^2 . Díky nedostatku světlocitlivých buněk není část světla vnímána. Mozek tento nedostatek kompenzuje nahrazením obrazu z okolních bodů a obrazem z druhého oka. Přestože všichni obratlovci mají slepou skvrnu, hlavonožčí oči, které jsou téměř totožné, ji nemají. V nich, je optický nerv připojen k receptorům zezadu, tudíž se nemusí vytvářet chyba v sítnici.
- Žlutá skvrna (makula) – je místo na oční sítnici, kde je největší hustota čípků a tedy i neostřejší vidění. Obraz, na nějž se člověk dívá zpřímá, se promítá právě do

tohoto bodu. Zajímavé je, že žlutá skvrna ve skutečnosti zaživa není vůbec žlutá, nýbrž o něco červenější než okolí. Tento přívlastek však skvrna dostala podle toho, že žlutou se jeví u mrtvoly. Žlutá skvrna má v průměru asi 5 mm. Obsahuje karotenoidy lutein a zeaxantin. V její centrální části se nachází fovea centralis, v centru této je tzv. foveola. Jedná se o termíny označující místa, v nichž postupně přibývá čípků a zvyšuje se ostrost vidění, a v úplném středu nejsou vůbec žádné tyčinky.

- Cévní řečiště – jako každý orgán i oko musí být zásobováno kyslíkem. To se děje pomocí krve. Existují dva zdroje krevního zásobování sítnice: centrální sítnicová tepna a cévnatkové cévy. Cévnatce se dostává největšího krevního průtoku v oku (65 – 85 %) a její hlavní úlohou je údržba vnější sítnice zejména fotoreceptorů. Zbytek krevního toku (20 – 30 %) proudí do sítnice pomocí centrální sítnicové tepny přes optický nerv. Vyživuje vnitřní stranu sítnice. Centrální sítnicová tepna má na sítnici 4 hlavní větve **2.3**.

Soustava tepen a žil vytváří cévní řečiště, to je u každého člověka rozdílné. Dokonce i u jednovaječných dvojčat. Díky tomu se tato část očního aparátu hodí k biometrickému měření [7], [11], [34], [35], [36], [25].

2.2 Oční choroby

Každá část lidského těla může v průběhu života postihnout choroba. Ať už vyléčitelná nebo trvalá. Trvalou chorobou budeme rozumět, že ji nelze chirurgicky či jinak odstranit aniž by to mělo za následek zničení biometrické informace např. amputace. Vyléčitelná choroba je naopak dobře léčitelná či odstranitelná s minimálními následky např. zánět, řezná rána. Oko mohou postihnout oba typy těchto chorob. My se nyní zaměříme na choroby ovlivňující sítnici oka. Ta je v této práci primární. Při rozpoznávání sítnice zkoumáme cévní řečiště. Pokud dojde k jeho narušení může dojít k chybnému vyhodnocení nebo dokonce k možnosti úplného zamítnutí vytvoření vzoru.

2.2.1 Makulární degenerace

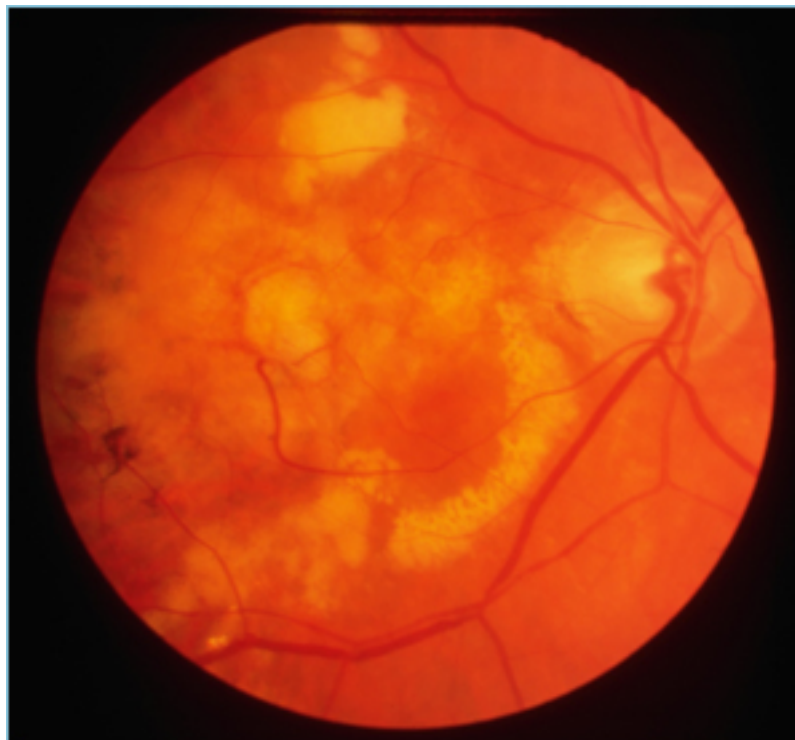
Toto onemocnění vzniká v 90 % s postupujícím věkem, potom hovoříme o stařecké, nebo také senilní makulární degeneraci (ARMD). Ve zbývajícím procentu se makulární degenerace objevuje u dětí nebo mladých lidí ve formě Bestovy makulární degenerace nebo Stargardtovy nemoci. Tato onemocnění vznikají na podkladě dědičnosti. Hovoříme také o makulární juvenilní degeneraci.

U makulární degenerace dochází k porušení oblasti sítnice **2.5**, která vytváří střed zorného pole a umožňuje schopnost barevného vidění. V důsledku toho vzniká závažná porucha centrálního zorného pole. V jeho středu pacient vidí šedý stín až černou skvrnu. Periferní vidění přitom zůstává neporušené.

Makulární degenerace může probíhat ve dvou formách – suché (atrofické) a vlhké (exudativní).

Při suché makulární degeneraci se žlutá skvrna (makula) ztenčuje a pod ní se hromadí části rozpadlé tkáně. Většinou tato forma předchází rozvoji vlhké makulární degeneraci. Ta se také nazývá hemoragická. Pod sítnicí se tvoří nové nakupené cévky, které začínají tláčit a trhat sítnici. Dochází k mikroskopickým krvácením, která vedou k zajištění tkáně sítnice a závažné poruše vidění.

Mezi nejčastější příznaky patří rozmazaná šedá nebo černá skvrna v centru zorného pole (tzv. centrální skotom). Postižený vidí deformované rovné linie, rozmazané písmo nebo neodpovídající tvar různých předmětů. Dochází také k postižení vidění barev, které se zdají vybledlé. Postranní vidění zůstává ostré na jednom nebo obou očích [8], [21], [20], [37].



Obrázek 2.5: Vzhled sítnice postižené makulární degenerací

2.2.2 Diabetická retinopatie

Diabetická retinopatie (DR) je nezánettivé onemocnění oční sítnice. Vzniká jako důsledek celkového postižení cév u diabetu mellitu. Špatně kompenzovaná cukrovka postihuje v očích drobné cévičky, které se ucpávají, tím dojde k porušení prokrvení sítnice (vrstvy, která zajišťuje vidění). Druhý způsob postižení sítnice je, že cévy „netěsní“, vytéká z nich tekutina a způsobuje otok sítnice.

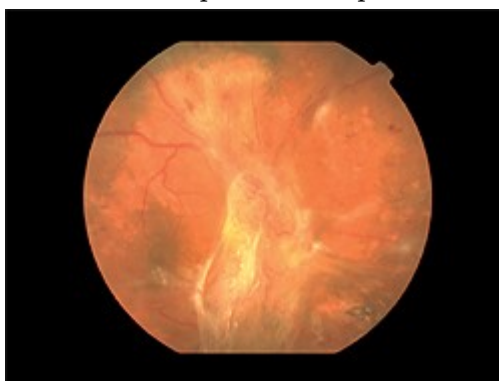
Nedostatečné prokrvení i otok sítnice ničí schopnost vidění. Oko se snaží napravit situaci růstem nových cév (neovaskularizace), ty jsou ale nekvalitní a škodlivé, praskají, mohou prokrvácet oko (hemoftalmus) a způsobit trakční odchlípení sítnice.

Diabetická retinopatie má dvě formy:

1. neproliferativní 2.5
2. proliferativní 2.5



Obrázek 2.6: Sítnice postižená neproliferativní DR

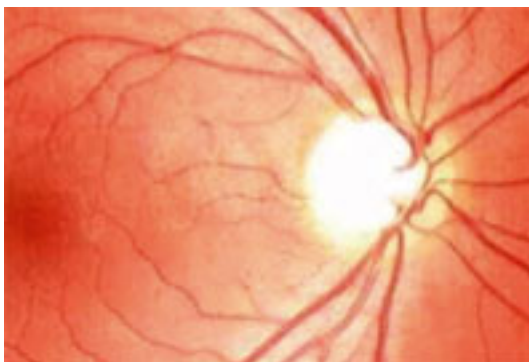


Obrázek 2.7: Sítnice postižené proliferativní DR

Léčba spočívá v kompenzaci diabetu, případně laserové koagulaci změn [16], [38], [24], [26].

2.2.3 Glaukom

Glaukom (lidově zelený zákal) je označení pro určitý typ poškození zraku, jehož podstatou je poškození očního nervu 2.8. Předpokládá se, že obvyklým způsobem vzniku poškození je vysoký nitrooční tlak, ale nemusí to být jediná příčina, protože se vyskytly případy tohoto onemocnění u lidí, u nichž nebyl vysoký nitrooční tlak prokázán. Primárních příčin zvýšení nitroočního tlaku a vzniku glaukomu může být nepřeberně mnoho, od infekce přes poruchu regulace tvorby nitrooční tekutiny až po ucpání odtokových kanálků pro tuto tekutinu [13], [6], [39], [23].



Obrázek 2.8: Sítnice postižena glaukomem

2.2.4 Toxoplazmóza

Toxoplazmóza je onemocnění, které se řadí mezi zoonózy, což jsou nákazy přenosné ze zvířat na člověka. K nákaze je náchylná celá řada volně žijících i doma chovaných zvířat, která se mohou stát zdrojem onemocnění člověka. Vyskytuje se po celém světě, v evropských zemích má protilátky proti toxoplazmóze vytvořeno 10 – 60 % obyvatelstva v závislosti na stravovacích zvyklostech. V České republice se séropozitivita (přítomnost protilátek v krvi) pohybuje kolem 20 – 40 %. Onemocnění se nejčastěji projevuje pouze zvýšenými teplotami, chřipkovitými stavy, bolestmi hlavy, únavností, či zduřením mízních uzlin. Akutní nákaza může někdy přejít do chronického stadia, často ale infekce proběhne úplně nepozorovaně a pozná se pouze podle nálezů specifických antitoxoplazmových protilátek v krvi, které mohou v nízkých hodnotách přetrvávat i po celý život (latentní forma nákazy). Existuje celá řada forem nemoci:

- Uzlinová forma toxoplazmózy – se projevuje zduřením mízních uzlin hlavně podčelistních, ale i podpažních, nebo ve slabinách. Vyskytuje se převážně u mladých lidí a asi 10 % zánětů mízních uzlin je toxoplazmového původu.
- Oční forma – vyznačuje se zánětlivým postižením očního aparátu, nejčastěji zánětem sítnice 2.9.
- Mozková forma toxoplazmózy – s ní se setkáváme u imunodeficitních (s poruchami imunity) osob s nálezem AIDS, kde se jedná zejména o reaktivaci již dříve proběhlé nákazy.
- Gynekologická forma – může se projevit sterilitou, opakovanými potraty, patologickým těhotenstvím, porodem postižených nebo mrtvých dětí.

Ostatní formy toxoplazmózy jsou méně časté. Dramatickou výjimkou obvykle neškodných parazitárních infekcí je infekce lidského plodu – vrozená toxoplazmóza. Může k ní dojít během akutní infekce matky v kterémkoliv období těhotenství [46], [40].



Obrázek 2.9: Sítňice postižena toxoplazmózou

2.2.5 Nádorová onemocnění

Tělo produkuje celou řadu různých typů buněk. Normální zdravé buňky rostou a dělí se na nové podle potřeb organismu, a tento proces udržuje tělo zdravé. Někdy se však buňky začínají dělit a vznikají nové, i bez potřeby organismu. Vzniká tak masa nové tkáně, kterou nazýváme nádorem. Nádor může být benigní nebo maligní.

První skupina jsou nádory nezhoubné (benigní), které rostou většinou pomalu, většinou nevrůstají do okolích tkání a sousední struktury spíše jen utlačují. Nejpodstatnější je, že nepronikají do cév krevního nebo mízního řečiště a nezakládají druhotná ložiska –metastázy. Buňky, ze kterých se skládají, nebývají příliš odlišné od těch, z nichž vznikly. Pokud se tyto nádory podaří odstranit, většinou znovu nenarůstají. Benigní tumory jsou tak jen vzácně hrozbou ohrožení života.

Druhou skupinou, podstatně nepříznivější, jsou nádory zhoubné (maligní). Ty rostou rychle, některé pronikají dříve či později do okolních tkání, které ničí. Takovému šíření říkáme místní šíření nádoru. Brzy potom však agresivní buňky naruší stěny cév a šíří se do organismu cévní a mízní soustavou. Pokud jsou v daném místě příznivé podmínky pro jejich další dělení, vznikají metastázy. Šíření zhoubných buněk krevní nebo lymfogenní cestou se nazývá metastazování [2], [19], [22].



Obrázek 2.10: Sítňice postižena rakovinou

2.3 Fundus kamery

Bohužel firmy, které vyráběly senzory na získání snímku oční sítnice, již neexistují nebo přestaly senzory vyrábět. Proto jediným způsobem, jak získat kvalitní snímek, je použít specializovaného oftalmologického zařízení – fundus kamery.

2.3.1 Obecný popis fundus kamery

V současné době se pro vyšetření očního pozadí (oční sítnice), kde se hodnotí jeho diagnosticky významné části, používají kromě oftalmoskopu i digitální fundus kamery. Jedná se o zobrazovací systém, jehož optika umí korigovat vysoké hodnoty ametropie. Kamery obsahují zdroj bílého světla, kterým lze osvětlit sítnici, jejíž obraz je následně snímán CCD prvkem. Zavedení digitálních fundus kamer ve vysoké míře podporuje diagnostiku různých onemocnění oka i kardiovaskulárního systému. Fundus kamery jsou většinou vybaveny systémy pro automatické nalezení středu sítnice a systémy automatického ostření, které se provádí pomocí frekvenční analýzy snímaného obrazu hodnocením obsahu vysokých frekvencí. Dále obsahují systém automatického řízení intenzity osvětlení sítnice, kdy se vyhodnocují předchozí snímané obrazy a intenzita osvětlení se upravuje na základě průměru jasů v těchto obrazech [12].

2.3.2 Non-mydiatická funduskamera Canon CR-1

Všechna obrazová data pro tuto práci jsou pořízena Non-mydiatickou funduskamerou Canon CR-1. Ta je k dispozici v laboratoři S214. Tato funduskamera je vybavena fotoaparátem CANON EOS-20D. Lze připojit i jiné podporované fotoaparáty [18].

Základní informace	
Typy obrázků	Barevné, Infračervené, Flourescenční angiograf
Úhly pohledu	50° a 43°
Minimální velikost zornice	5.2 mm (v SP módu 4.3 mm)
Velikost pacientových dioptrií	Bez kompenzačních čoček: -10 D až +15 D S „-“ kompenzačními čočkami -7 D až -31 D S „+“ kompenzačními čočkami +11 D až +33 D
Zdroj světla	Halogenová lampa na ohledání , xenonová lampa pro fotografování

2.4 Biometrie a požadavky na systém

Biometrie je automatické rozpoznávání lidí na základě jejich charakteristických anatomických rysů (otisk prstu, obličej, duhovka, ...) a charakteristického chování (chůze, podpis). Každou biometrickou vlastnost lze hodnotit těmito atributy:

- Univerzalita – každá osoba tuto vlastnost má
- Unikátnost (jedinečnost) – jak je tato vlastnost různorodá u každé osoby
- Konstantnost – měří jak moc se vlastnost mění s věkem a časem
- Získatelnost – zda lze tuto vlastnost kvantitativně měřit
- Výkonnost – přesnost, rychlost a robustnost použitého systému
- Akceptovatelnost – ochota lidí tuto metodu používat
- Ošálitelnost – jak jednoduché je vytvořit podvrh
- Spolehlivost – způsob vyhodnocení při ovlivnění okolí

Nyní si provedeme vyhodnocení jednotlivých atributů pro sítnici oka, jež je hlavní náplní této práce.

2.4.1 Univerzalita

Tento atribut říká, že každý člověk musí danou vlastnost nějakým způsobem vlastnit. Každý člověk, jestliže u něj nedojde k nehodě při které přijde o obě oči, či se s vadou nevyvinutých očí (mikrooftalmie) nebo sítnice narodil (tato pravděpodobnost je ovšem velmi nízká), má oči, jejichž složitá stavba 2.1 obsahuje sítnici. Díky tomu je univerzalita vysoká.

2.4.2 Unikátnost

Známá též jako jedinečnost, určuje jak velký rozdíl je mezi danou biometrickou vlastností u různých osob. Pravděpodobnost výskytu stejné vlastnosti musí být co nejnižší. To je v případě sítnice oka splněno. Tato pravděpodobnost je o několik řádů menší než u pravděpodobnosti výskytu stejných otisků prstů. Je to způsobeno zejména tím, že vývoj cévního systému je u každého člověka naprosto rozdílný a to i u jednovaječných dvojčat. Proto je unikátnost sítnice oka velmi vysoká, narozdíl od délky prstů.

2.4.3 Konstantnost

Jde o atribut, který určuje jaká, je změna biometrické vlastnosti v průběhu času. Vývoj oka a jeho reflexů je ukončen přibližně v 6. roku života. Po této době již nedochází k významným změnám. K těm může dojít v případě úrazu či nemoci ovlivňující sítnice 2.2. Při některých nemocích ovšem dochází k nevratným změnám na sítnici. Pokud je zasažena celá sítnice, ztrácí člověk možnost identifikace systémem. Jestliže ale není postižená slepá skvrna, ve které je největší hustota cév, můžeme pro něj vytvořit šablonu zaměřenou pouze na tuto část sítnice a blízké okolí. Tím můžeme konstantnost označit také jako velmi vysokou.

2.4.4 Získatelnost

Tento atribut říká, jak dobře lze získat biometrickou vlastnost pro provedení požadované metody. Měří se doba, kdy osoba podstoupí „vyšetření“ systémem. Doba je delší než například získání obrazu duhovky. Je to způsobeno šikovností operátora přístroje a „kvalitou“ uživatele oka. Například pokud uživatel trpí silnou krátkozrakostí nebo dalekozrakostí, je velmi obtížné přesně zaostřit. Proto se bohužel rozpoznávání sítnice řadí mezi metody s nízkou získatelností.

2.4.5 Výkonnost

Jde o atribut, který ukazuje, jak je daná metoda pro danou vlastnost rychlá, výkonná a robustní. Metoda rozpoznávání sítnice je většinou založena na aplikaci jednoduchých grafických filtrů a operací s obrazem. Tyto operace jsou při správném použití rychlé a výkonné. Míry FNMR a FMR bývají malé. Z tohoto důvodu patří rozpoznávání sítnice oka mezi vysoce výkonné metody.

2.4.6 Akceptovatelnost

Tento atribut ukazuje, jak moc je osoba ochotna testování dané biometrické vlastnosti podstoupit. V tomto případě sítnice pokulháva. Mnoho lidí má z použití dané technologie strach. Jsou přesvědčeni, že bude použito laseru což by jim mohlo způsobit poruchu oka. Tyto obavy jsou ale naprosto zbytečné, protože laser se v tomto případě nikdy nepoužívá. Dalším problémem je samotná procedura získání obrazu sítnice. Ta je zdlouhavá, což může některým uživatelům vadit.

2.4.7 Ošálitelnost

Jde o atribut, který ukazuje bezpečnost systému. Ta je samozřejmě pro každý biometrický systém klíčová. Nezajímá nás ovšem, jak je zabezpečena databáze se vzory, ale to, zda je možné testování vlastnosti nějakým způsobem podvrhnout. Například umělými prsty,

vhodnými kontaktními čočkami apod. Vytvoření falešné sítnice skýtá řadu úskalí. Jedním faktorem je zisk modelované sítnice. Ten bez spolupráce s vlastníkem oka jde velmi těžko. V případě zisku obrázku sítnice, bychom si museli vytvořit umělé oko. Zde je další faktor. Vytvoření umělého oka nebude jistě finančně jednoduchá záležitost. Pokud by došlo i k této alternativě, bude velmi obtížné udržet toto oko v klidu, aby byl pořízen kvalitní snímek. Pokud by například došlo k usmrcení osoby a následnému zkoušení neživého oka, lze testovat živost na barvu žluté skvrny, která je v tomto případě opravdu žlutá, nebo provést nejprve test živosti na reakci duhovky. Až poté provést sken sítnice. Tato metoda by ale musela provádět kontrolu celého záznamu. Z výše popsaných skutečností vyplývá, že ošálitelnost je velmi nízká.

2.4.8 Spolehlivost

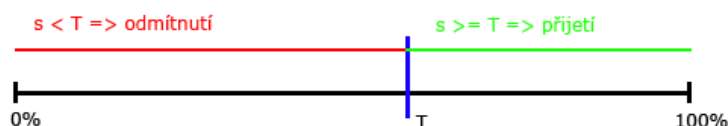
Tímto atributem se značí, jaký vliv má okolí na kvalitu vyhodnocení. Například spolehlivost rozpoznávání podle hlasu je nízká. Velký vliv zde hraje ruch z okolí. Většina výrobců uvádí u svých senzorů vysokou spolehlivost, ale neuvedou, že byly testovány pouze v laboratoři, kde panují ideální podmínky. Pokud jde o snímání sítnice, jeho spolehlivost je vysoká. Existují však podmínky, kdy není možné získat dostatečně kvalitní obrázek sítnice. Jedná se zejména o špatné osvětlení. Pokud bude přístroj na snímání umístěn blízko silnějšího zdroje světla může dojít k přesvětlení snímku, nebo větším odleskům z oka. Také uživatel má při vyšetření vlivem velkého množství světla silně staženou duhovku. To způsobí nemožnost získání obrazu sítnice úplně. Minimální průměr musí být 3 mm. Dalším problémem nastává při již zmíněných chorobách nebo jiných dysfunkcích oka. Možným faktorem též může být vlhkost ovzduší, díky níž může dojít k zarosení optiky.

2.4.9 Statistické koeficienty

Z předchozích podkapitol vyplývá, že celková výkonnost biometrického systému je určena vlastnostmi jako univerzálnost, přesnost, rychlost a paměťová náročnost. Dalšími faktory, které celkovou výkonnost ovlivňují, jsou cena a akceptovatelnost. Jako všechny systémy, tak ani biometrické systémy nejsou dokonalé. Pokud vynecháme technické problémy, se kterými se můžeme setkat např. chyba senzoru, chyba při přenosu dat atd., zůstávají dva hlavní problémy:

- chybné přijetí útočníka jako platného uživatele
- odmítnutí platného uživatele jako útočníka

K těmto chybám dochází v procesu porovnání šablony a předloženého vzoru. Tento proces má za výsledek skóre porovnání s tzv. míra shody. Skóre porovnání udává kvantifikovanou podobnost mezi vzorem a šablonou. Každý biometrický systém má nastavenou hodnotu prahu T . Ta určuje, jaké musí být minimální skóre porovnání, aby došlo k přijetí viz. 2.11



Obrázek 2.11: Oblast přijetí a odmítnutí v závislosti na prahu T

Z tohoto vychází dva chybové stavy, které zapříčiňují již dva zmíněné problémy:

- Dva vzory od dvou odlišných osob jsou rozpoznány jako shodné \implies chybná shoda
- Dva vzory od stejné osoby nasnímané v různé době jsou rozpoznány jako odlišné \implies chybná neshoda

Existuje několik významných statistických koeficientů, které jsou pro hodnocení biometrického systému důležité. Jedná se o:

- FAR (False Acceptance Rate) – Míra chybného přijetí – Pravděpodobnost, že biometrický systém, klasifikuje chybně dva odlišné vzory jako shodné a tím selže při odmítnutí možného útočníka [4].

$$FAR = \frac{\text{počet shodných porovnáání rozdílných vzorů}}{\text{celkový počet porovnáání rozdílných vzorů}} * 100$$

- FRR (False Rejection Rate) – Míra chybného odmítnutí – Pravděpodobnost, že biometrický systém, klasifikuje chybně dva vzory od stejné osoby jako odlišné a tím selže při přijetí oprávněného uživatele [4].

$$FRR = \frac{\text{počet porovnáání vzorů osoby A vedoucích k neshodě}}{\text{celkový počet porovnáání osoby A}} * 100$$

- FTA (Failure To Acquire) – Míra neschopnosti nasnímat – Udává podíl chybných záznamů v automatickém módu záznamu daného senzoru. Tj. zaznamenání biometrické charakteristiky je odmítnuto, ačkoliv je biometrická charakteristika přítomna. Čím vyšší je tato hodnota, tím méně je daný senzor vhodný pro záznam uvedené biometrické charakteristiky. Tím pádem slouží tato míra k hodnocení kvality senzorů [4].

$$FTA = \frac{\text{počet nenásnímaných vzorů}}{\text{celkový počet pokusů o nasnímaní vzorů}} * 100$$

- FTE (Failure To Enroll) – Míra neschopnosti zaregistrovat – Udává podíl uživatelů, které není systém schopen se naučit. Míry FTE se často vyskytují u systémů, které mají kontrolu kvality biometrické charakteristiky, tj. biometrické charakteristiky s nedostatečnou kvalitou nejsou systémem naučeny. V tomto smyslu představuje FTE údaj, který ohodnocuje schopnost algoritmu pracovat i s nekvalitními biometrickými charakteristikami [4].

$$FTE = \frac{\text{počet nezaregistrovaných osob}}{\text{celkový počet osob vyžadujících registraci}} * 100$$

- FTM (Failure To Match) – Míra neschopnosti porovnat – Udává podíl biometrických charakteristik, které nemohly být porovnány se šablonou a nebo jakkoliv jinak zpracovány. Tato míra poukazuje na neschopnost systému učinit rozhodnutí, tj. porovnání nepřinese žádný výsledek [4].

$$FTM = \frac{\text{počet neporovnaných vzorů}}{\text{celkový počet porovnáání vzorů}}$$

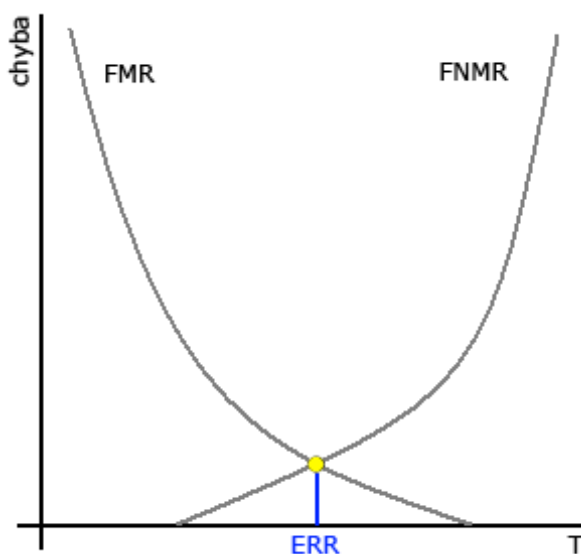
- FMR (False Match Rate) – Míra chybné shody – Udává podíl chybně akceptovaných osob. Na rozdíl od FAR nejsou do celkových součtů brány v potaz pokusy, které byly neúspěšné ještě před samotným porovnáváním [4].

$$FMR = FAR - FTA - FTE$$

- FNMR (False Non-Match Rate) – Míra chybné neshody – Udává podíl chybně neakceptovaných osob. Na rozdíl od FRR nejsou do celkových součtů brány v potaz pokusy, které byly neúspěšné ještě před samotným porovnáváním [4].

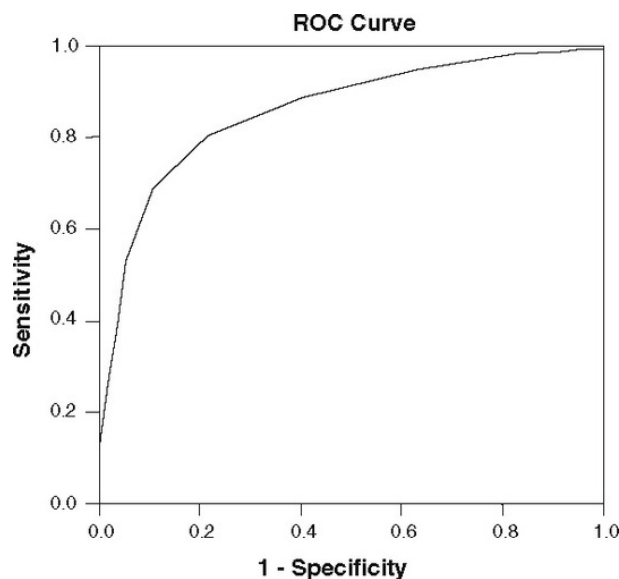
$$FMR = FRR - FTA - FTE$$

- EER (Equal Error Rate) – Míra vyrovnaní chyb – Je definovaná podmínkou $FMR(T) = FNMR(T)$. V praxi se u FMR a FNMR křivek jedná o diskrétní funkce, tj. přesné určení EER není možné. Je tedy možné udat oblast, ve které se obě chybové míry shodují viz. 2.12. Při nastavení porovnávacího prahu T na EER bude chybně akceptován stejně jako odmítnut stejný počet osob. Tím je tedy možné nastavit práh tak, aby hodnoty FMR/FNMR odpovídaly požadavkům [4].



Obrázek 2.12: Závislosti ERR

- ROC (Receiver Operating Characteristic) křivka – FMR i FNMR jsou závislé na hodnotě prahu T . Se změnou hodnoty prahu T se zvyšuje či zmenšuje hodnota FMR a FNMR. Výkonnost systému se proto udává pomocí tzv. ROC křivky. ROC křivky představují v současné době standard při popisu vlastností daného systému dále detekční schopnost funkce FRR vzhledem k FNMR (příp. FAR / FRR) viz. 2.13



Obrázek 2.13: Příklad ROC křivky

2.5 Požadavky na systém

Na vytvářený systém jsou kladeny tyto následující požadavky:

- zařízení na rozpoznávání sítnice musí být schopno použití i jako vestavěné zařízení
- kód musí být v jazyce, který je podporován vestavěnými chipy (jazyk C++)
- systém musí podporovat 3 fáze:
 - nasnímání (enroll)
 - extrahování a generování šablony (template)
 - identifikace/verifikace (porovnání záznamu z databáze)

2.5.1 Nasnímání, extrahování, identifikace

Senzor sejme biometrický vzor (sítnici), který dále putuje do extraktoru k extrakci žil a vytvoření šablony. Výsledná šablona je vstupem pro porovnávací jednotku. Aby bylo možné šablonu porovnat, je potřeba přístup do databáze, která slouží k uchování šablon žil sítnice. Výsledkem porovnání je procentuální shoda, která se dle nastaveného prahu zamění na Ano nebo Ne (v případě verifikace) nebo na nalezenou identitu (v případě identifikace). Tento výsledek je pak předán aplikaci, která buď povolí nebo odmítne dané osobě přístup.

Biometrický systém používáme ve dvou různých specifických režimech. První přichází na řadu registrace uživatele do systému. Jedná se o sejmutí biometrického vzoru a uložení výsledné šablony do databáze, kde bude připravena k použití při porovnávání. V tomto případě je většinou nejprve ověřena identita uživatele na základě nějakých neelektronických dokumentů (občanský průkaz). Registrace uživatele do systému většinou probíhá pod dozorem pověřené osoby, která uživatele zaregistruje. Tento člověk by měl zároveň dohlédnout na správný průběh, tedy aby s přístrojem nebylo manipulováno, nebo nedošlo k pokusu

o přiložení falešného oka. Musí se také zkontrolovat, zda vytvořená šablona je použitelná. Pokud by byla přijata nekvalitní šablona, mohlo by dojít k problémům při dalším přihlašování do systému. Například by mohlo dojít k přihlášení jiné osoby. Druhým režimem je identifikace nebo verifikace uživatele, při které se používá opět všech částí systému. Osoba poskytne biometrický vzor, je vytvořena šablona, která se porovná se šablonou v databázi. Pokud dojde k procentuální shodě (neshodě), je osoba přijata (odmítnuta) [10].

2.6 Shrnutí

Tato práce se zabývá rozpoznáváním sítnicí oka. Tato metoda je v dnešní době silně vytlačována metodou rozpoznávání duhovky oka. Hlavním důvodem je metoda získání vlastnosti. Pro získání duhovky postačuje obyčejný fotoaparát, naopak pro získání kvalitního obrazu sítnice je třeba drahého a specializovaného přístroje. Dalšími důvody jsou neochota a strach lidí z vyšetření. Z hlediska bezpečnosti je sítnice i duhovka přibližně na stejné úrovni. Sítnice má ovšem výhodu, nelze si vytvořit falzifikát a pokusit se jím systém oklamat [5] [41].

Kapitola 3

Analýza a návrh systému

V této kapitole bude provedena analýza systému a jeho návrh pomocí diagramů jazyka UML. Zaměříme se na zejména na diagram tříd, který má pro náš případ největší vypovídací schopnost.

3.1 UML

UML, Unified Modeling Language je v softwarovém inženýrství grafický jazyk pro vizualizaci, specifikaci, navrhování a dokumentaci programových systémů. UML nabízí standardní způsob zápisu jak návrhů systému včetně konceptuálních prvků, jako jsou business procesy a systémové funkce, tak konkrétních prvků, jako jsou příkazy programovacího jazyka, databázová schémata a znovupoužitelné programové komponenty.

UML podporuje objektově orientovaný přístup k analýze, návrhu a popisu programových systémů. UML neobsahuje způsob, jak se má používat, ani neobsahuje metodiku(y), jak analyzovat, specifikovat či navrhovat programové systémy.

Standard UML definuje standardizační skupina Object Management Group (OMG) [42].

3.1.1 Diagram případů použití

Diagramy případů použití jsou hlavní technikou modelování chování systému na úrovni analýzy v UML. Lze vytvořit řadu diagramů, které reprezentují různé aspekty systému na různé úrovni podrobností.

Diagram případu použití má v UML zvláštní postavení, a to ze dvou důvodů. Jednak jeho síla nespočívá v samotném grafickém vyjádření, ale v podstatně větší míře v textové specifikaci případů použití, která odráží požadavky uživatelů. Druhým důvodem je to, že právě tyto specifikace jsou významným zdrojem informací a návodem pro vývojáře prakticky ve všech fázích životního cyklu.

Hlavními vizuálními modelovacími prvky diagramu případů použití jsou případy použití a aktéři. Případ použití reprezentuje nějakou důležitou část celkové požadované funkčnosti vyvíjeného systému. Aktér reprezentuje uživatele systému (osobu nebo jiný systém) v nějaké roli, který přímo komunikuje s případem použití a očekává od něho nějakou zpětnou vazbu – nějaký pozorovatelný výsledek. Někdy je žádoucí vytvořit řadu diagramů případů použití, které zdůrazňují různé aspekty celkového modelu nebo jeho částí [45].

3.1.2 Diagram tříd

Základním diagramem UML pro modelování logického pohledu na statickou strukturu systému je diagram tříd, který vizualizuje třídy (a rozhraní), jejich interní strukturu a vztahy k ostatním třídám. Třída je v UML definována jako popis množiny objektů, které sdílejí tytéž specifikace rysů, omezení a sémantiky. Rysy třídy jsou atributy a operace.

Terminologie objektově orientovaných programovacích jazyků bývá často poněkud odlišná. Rysy se označují jako členy třídy (class members). Atributy jsou datové členy nazývané často členské proměnné (member variables) nebo instanční proměnné (instance variables) či pole (fields). Operace jsou členské funkce (member functions), typicky nazývané metody. Vyvolání služby poskytované metodou se zpravidla označuje jako zaslání zprávy objektu (sending a message, message passing). V jazyce UML se odlišuje operace a metoda. Operace je specifikace části chování objektu. Metoda je její implementací. Z předchozího odstavce je zřejmé, že terminologie programovacích jazyků používá označení metoda i ve smyslu operace. Tam, kde nebudeme potřebovat striktně odlišit specifikaci a implementaci, budeme používat v této studijní opoře v souladu s učebnicí označení metoda. Vztahy v diagramu tříd jsou významná spojení tříd (případně rozhraní). Existují tři typy vztahů, které používáme v diagramu tříd: asociace, agregace a generalizace [45].

- Asociace ve skutečnosti reprezentuje vazby objektů daných tříd, podobně jako vztahy v ER diagramu. Nejčastějším případem asociací jsou binární asociace, což jsou asociace reprezentující vazby mezi dvojicemi objektů, zpravidla dvou různých tříd. Jde-li o objekty téže třídy, nazývá se asociace reflexivní. Obecně ale může participovat ve vazbě více objektů. V případě tří objektů hovoříme o ternární asociaci. U asociace můžeme definovat její jméno, jméno role, násobnost a navigovatelnost. Poslední tři vlastnosti se vztahují vždy ke konkrétnímu konci asociace [45].
- Agregace je v UML speciální druh asociace modelující vztah celek – část, tj. kde instance jedné třídy (celek, agregát) obsahuje instance jiné třídy (části). Vztah celku a části modelovaný agregací je volný. Příkladem může být počítač a periferie. Celek někdy může existovat nezávisle na částech, jindy ne. Podobně části mohou existovat nezávisle na agregátu. U agregace může být dokonce část sdílená více agregáty, tj. násobnost u celku může být větší než jedna.

V UML existuje silnější forma agregace, která vyjadřuje těsnější vazbu celku a částí. Nazývá se kompozice. Graficky se liší od agregace tím, že symbol u celku je vybarven. U kompozice mohou části patřit vždy jen jednomu celku, celek je zodpovědný za vytvoření a zrušení částí, a když je zrušený celek, musí být zrušeny všechny jeho části. Je zřejmé, že sémantika kompozice je velice podobná sémantice atributů a modeluje vlastně totéž. Atribut bychom měli použít, jde-li o primitivní typy nebo třídy, které jsou součástí jazyka nebo v případě, kdy nechceme třídu explicitně ukazovat v modelu. Základním pravidlem by měla být jasnost, užitečnost a srozumitelnost modelu [45].

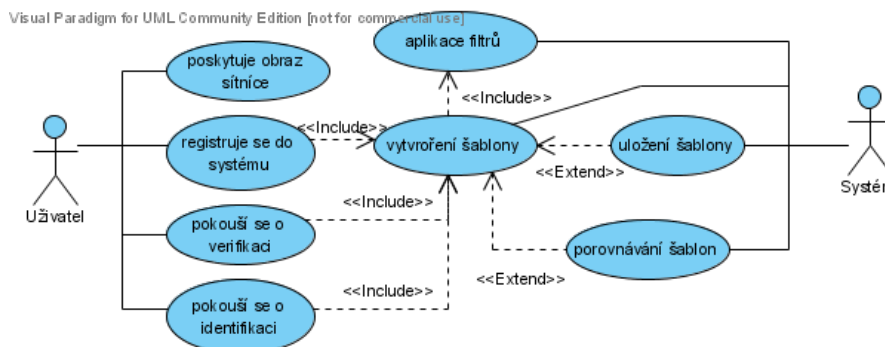
- Generalizace je vztahem mezi obecnějším prvkem a speciálnějším prvkem, kde speciálněší prvek je zcela konzistentní s obecnějším prvkem, ale obsahuje více informací. Oba prvky se řídí principem nahraditelnosti. Ten znamená, že můžeme použít speciálněší prvek všude tam, kde je očekáván prvek obecnější bez narušení systému. V diagramu tříd půjde o vztah mezi třídami. Obecnější třída se nazývá nadtrídou (superclass) a speciálněší třída podtrídou (subclass). Struktura vzniklá uspořádáním tříd podle vztahu generalizace se nazývá hierarchie generalizace [45].

3.2 Analýza

Jak vyplývá z požadavků na systém, je potřeba, aby byl uživatel nejprve zaregistrován do systému. Tato část se skládá z několika částí. Nejprve musí být uživateli vyfocena sítnice oka. Poté tento obraz musí projít řadou filtrů, aby se z něj stala šablona. Ta je uložena do databáze šablon. Další částí je verifikace a identifikace uživatele. Systém dostane obrázek sítnice, který opět projde filtrací a vzniklý vzor je porovnáván se šablonami v databázi a podle nastaveného prahu vyhledá odpovídající záznamy.

3.2.1 Diagram případů užití

Podle analýzy byl vytvořen tento diagram případů užití 3.1.



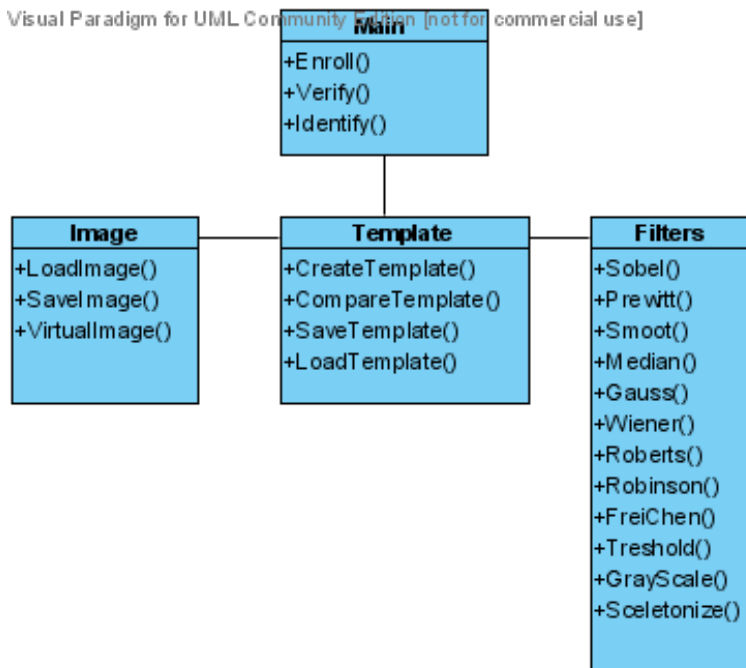
Obrázek 3.1: Diagram případů užití

3.3 Návrh

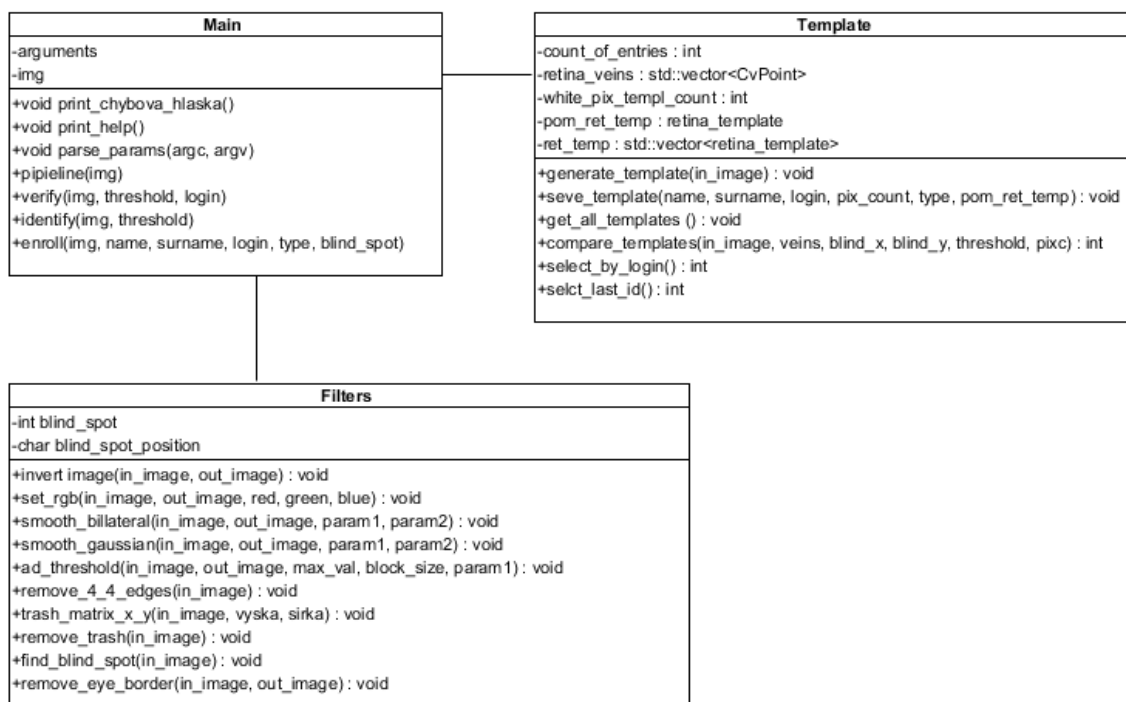
3.3.1 Diagram tříd

Nejprve byl vytvořen konceptuální diagram tříd 3.2. Ten ukazuje přibližnou strukturu programu. Podle něj byly vytvořeny další dva diagramy tříd. První diagram 3.3 nám ukazuje, jak by měl implementovaný program vypadat. Třetím diagramem 3.4 je již sofistikovaný program s plnou podporou rozhraní, balíčků umožňuje i využití relační či objektové databáze.

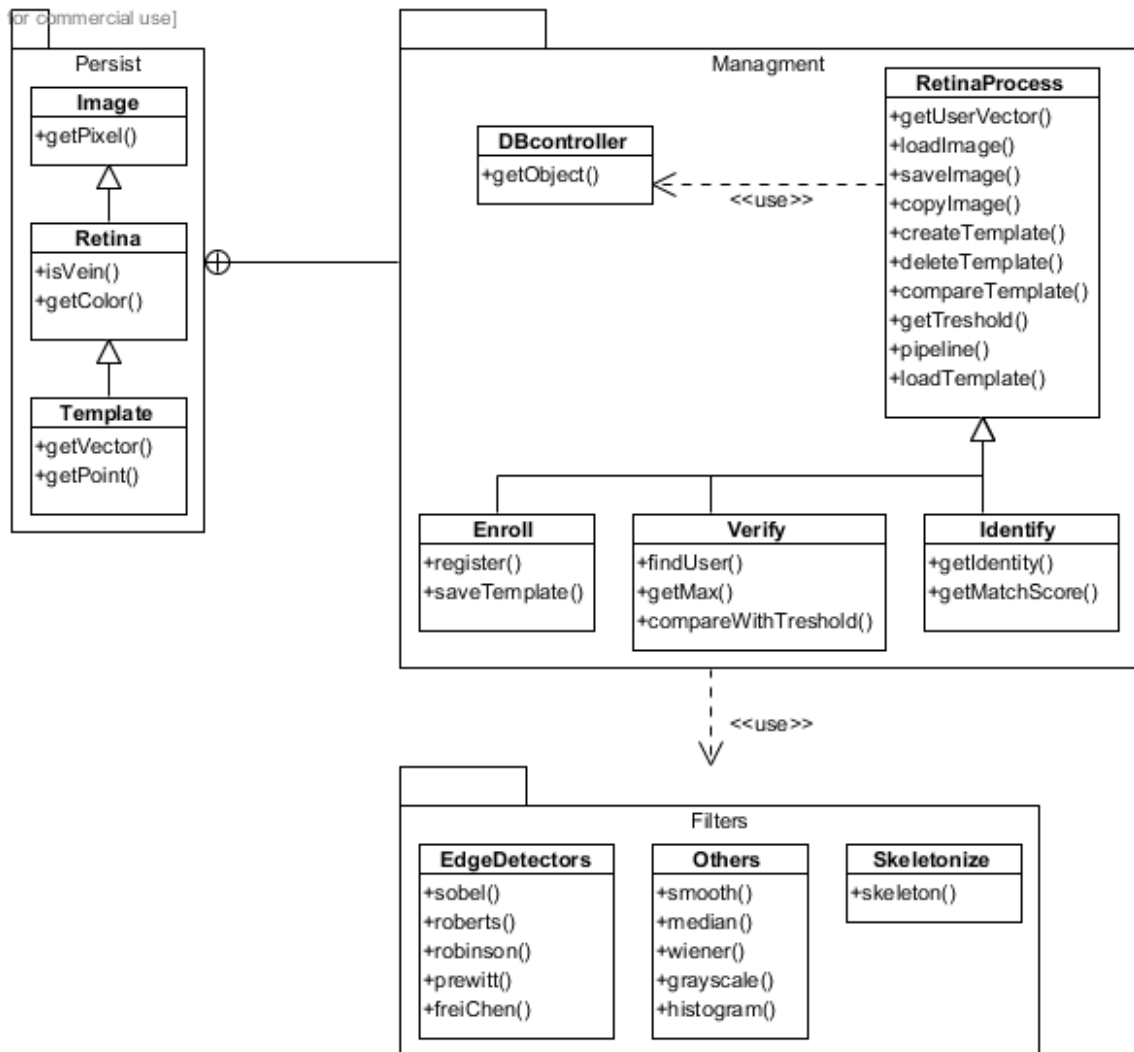
Nyní si přiblížíme použitý diagram tříd. Třída `Main` obsahuje metody, které slouží k zaregistrování do systému (`enroll`), verifikaci (`verify`), identifikaci (`identify`), zpracování obrazu (`pipeline`) a další pomocné metody. Třída `Template` se stará o vytvoření šablony (`generate_template`), porovnávání šablon (`compare_templates`), ukládání (`save_template`) a nahrávání (`get_all_templates`) šablon. Pro zpracování obrazu se použije třída `Filters`. Ta obsahuje metody pro jednotlivé filtry použité k extrakci.



Obrázek 3.2: Konceptuální diagram tříd



Obrázek 3.3: Použitý diagram tříd

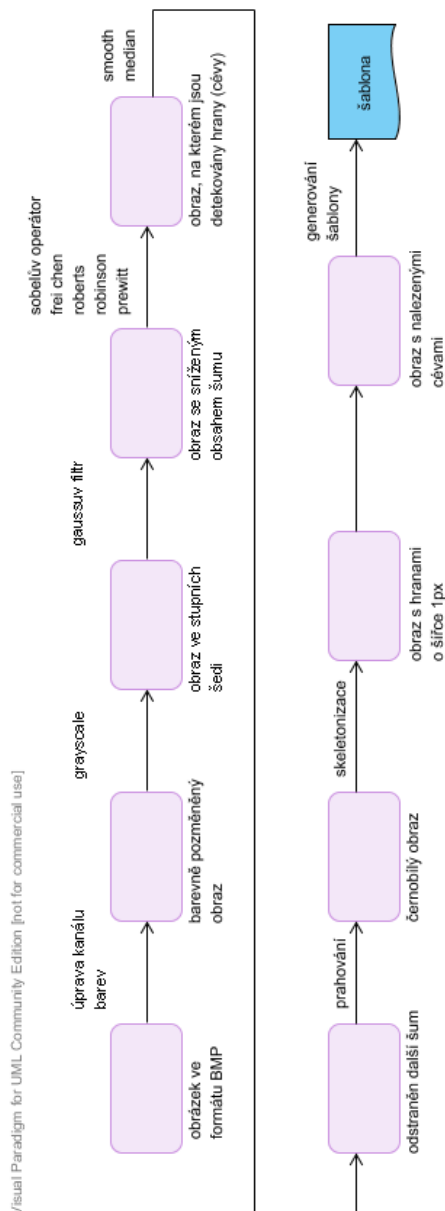


Obrázek 3.4: Sofistikovaný diagram tříd

Kapitola 4

Obrazové filtry

V této části práce se budeme věnovat digitálním obrazovým filtrům. Ty jsou použity k extrakci cévního řečiště v obrazu sítnice. Existuje celá řada filtrů, které umožňují různé operace s obrazem. Pro tuto práci jsou důležité zejména filtry odstraňující šum z obrazu, filtry pro detekci hran v obraze, filtry pro převod do stupňů šedi, binarizační filtry. Dále je potřeba vytvořit algoritmus skeletonizace. Ten ztenčí velikost vyhledaných hran na 1px. Je třeba nalézt vhodnou funkci, které je schopna proložit křivky obrazem. Důležité je ovšem tyto filtry ve správném pořadí použít a tím vytvořit „pipeline“ filtrů vedoucí k šabloně. Předpokládané pořadí filtrů je patrné z obrázku [4.6](#).



Obrázek 4.1: Postup aplikace filtrů

4.1 Detekce hran

Detekce hran je postup v digitálním zpracování obrazu, sloužící k nalezení oblastí bodu, ve kterých se podstatně mění jas. Slouží k nalezení takových míst obrazu, které lidské oko vnímá jako hranice objektu nebo různých tvarů, např. rozhraní světla a stínu. K provedení hranové detekce se používají hranové detektory (edge detector).

Hranových detektorů existuje celá řada, liší se jak v metodách nalezení hrany, její reprezentaci, tak v citlivosti rozpoznání hrany a s ní související odolnosti proti šumu a jiným nežádoucím efektům zpracovávaného obrazu.

V základu se hranové detektory dají rozdělit na detektory využívající první derivaci

nebo druhou derivaci jasové funkce. Při použití první derivace je výsledný hranový gradient porovnán s prahem, který určuje, jedná-li se o hrany či nikoli. U metod druhé derivace je výskyt hrany detekován, je-li prostorová změna v polaritě druhé derivace dostatečně významná.

Pokud je hrana definována jako relativně velká změna hodnoty jasové funkce, bude v místě hrany velká hodnota derivace této funkce. Hodnota této derivace (gradientu) bude maximální v kolmém směru ku hraně. V praxi se však detekce provádí jen ve dvou, resp. ve čtyřech směrech. K výpočtu gradientu můžeme přistupovat jako ke konvolučnímu filtrování obrazu. Jednotlivé hranové detektory se pak liší jádrem konvolučního filtru. Jádro filtru (matice užitá ke konvoluci) udává, které body se pro výpočet gradientu použijí a jaké budou mít váhy. Velikost a vlastnosti filtru významně ovlivňují výsledek detekce. Obecně platí, že čím je matice větší, tím je detektor odolnější proti šumu, protože okolí použité k aproximaci je větší. Konvolucí mezi jádrem filtru a zpracovávaným obrazem získáme tzv. gradientní obraz. Konvoluce se provádí pro detekci v jednom směru (vodorovně nebo svisle) a pro druhý směr se použije matice transponovaná [1].

Ukázky některých významných filtrů:

- Robertsův operátor $G_x = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $G_y = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

- Operátor Prewittové $G_x = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$ $G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$

- Sobelův operátor $G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$ $G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

- Robinsonův operátor $G_x = \begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$ $G_y = \begin{bmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

- Frei-Chenův operátor $G_x = \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & \sqrt{-2} \\ 1 & 0 & -1 \end{bmatrix}$ $G_y = \begin{bmatrix} -1 & \sqrt{-2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$

Z výsledku detekce hran v horizontálním a vertikálním směru G_x a G_y lze spočítat sílu hrany:

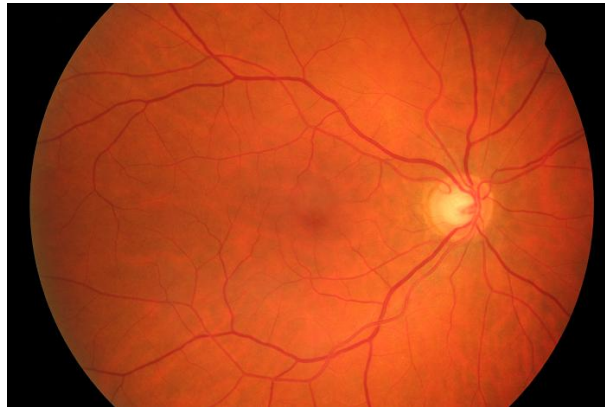
$$G = \sqrt{G_x^2 + G_y^2}$$

nebo můžeme hodnotu následně aproximovat:

$$|G| = |G_x| + |G_y|$$

Orientace gradientu vzhledem k vodorovné ose lze vypočítat jako:

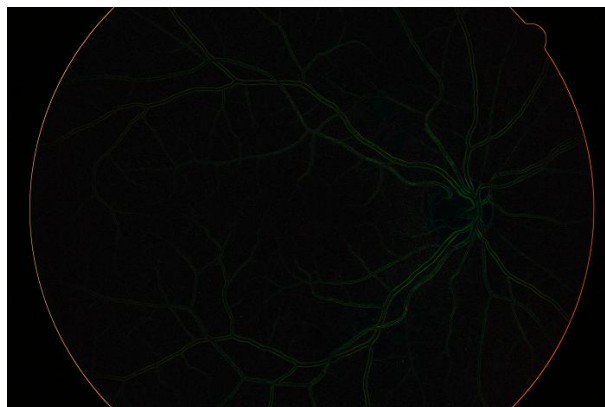
$\theta = \arctan \frac{G_x}{G_y}$
Aplikace těchto filtrů je vidět na následujících obrázcích:



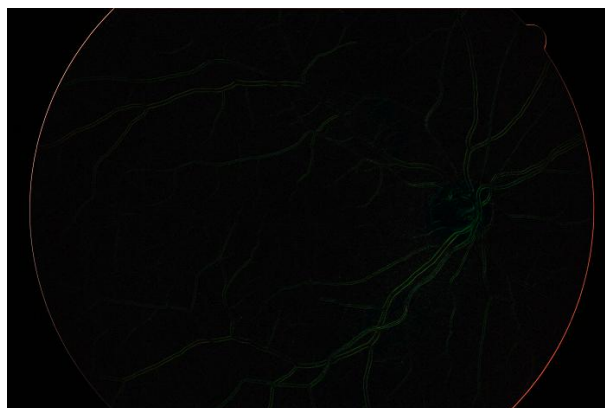
Obrázek 4.2: Originální obrázek



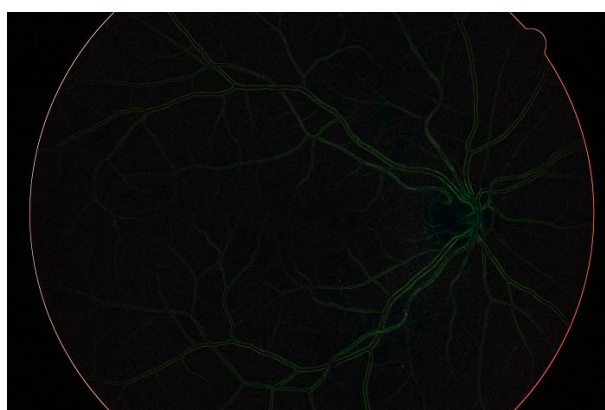
Obrázek 4.3: Robertsův filtr



Obrázek 4.4: Prewittův filtr



Obrázek 4.5: Robinsonův filter



Obrázek 4.6: Sobelův filter

4.2 Odstranění šumu

Šum je každý nežádoucí signál v obraze. Jde většinou o vysokofrekvenční signál s nulovou střední hodnotou, ale může mít velký rozptyl. Na jeho potlačení nebo úplné odstranění se používají různé druhy filtrů. Pro detekci hran je důležité minimalizovat šum v obrazu sítnice. Mohlo by dojít k nalezení neexistující hrany a tím k vytvoření chybné šablony. V digitálním obraze rozeznáváme nejčastěji dva typy šumu:

- Náhodný šum (nezávislý šum) – je způsoben například vadnými CCD elementy (příkladem tohoto typu šumu je šum typu „sůl a pepř“)
- Gaussův šum (závislý šum) – každý pixel obrazu mírně pozměněn

4.2.1 Lineární metody vyhlazování

Wienerův filtr

Tento filtr se používá na obrázky, které jsou degradované aditivním šumem a které jsou rozmazané. Při Wienerově filtraci (s využitím nejmenších čtverců) je důležité rozpoznat charakter poruchy a statistické vlastnosti šumu. Ve frekvenční oblasti hledá vysoké frekvence s vysokým rozptylem. Je založena na minimalizaci rozdílu mezi odhadem f' a původním obrazem f s nejmenší střední hodnotou čtverce chyby e^2 . Z obrázku neodstraní šum, ale rozmáže ho:

$$e^2 = \epsilon \{ [f(i, j) - f'(i, j)]^2 \}$$

Kde ϵ představuje střední hodnotu. Jestliže neexistují žádné podmínky, tak optimální odhad f' je průměrná střední hodnota ideálního obrazu f za podmínky pozorované obrazu g . Většinou ovšem není známá podmíněná pravděpodobnost správného obrazu f za podmínky, že je k dispozici pozorovaný obraz g . Optimální odhad f' je obvykle nelineární funkcí pozorovaného obrazu g .

Gaussův filtr

Další lineární metoda odstranění šumu je konvoluce s maskou, která se skládá z elementů určených Gaussovou funkcí. Tato metoda vede k rozmazání obrázku, což může být pro další zpracování obrazu problém (například pro detekci hran). Je to efektivní technika k potlačení Gaussova šumu. 2D Gaussova funkce:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

4.2.2 Nelineární metody vyhlazování

Mediánový filtr

Nejjednodušším a ve zpracování obrazu je nejčastěji používán mediánový filtr. Vezme pro každý pixel obrazu jeho okolí. Ze všech těchto pixelů vybere medián, který se stává novou hodnotou zpracovávaného pixelu. Medián leží uprostřed posloupnosti hodnot, které jsou seřazeny podle velikosti. Při lichém počtu je roven hodnotě uprostřed. Při sudém počtu hodnot je roven aritmetickému průměru hodnot ležících uprostřed. Nejpoužívanější je okolí 3×3 . Ovšem při použití většího okolí než 7×7 dochází ke zkreslování výsledného obrazu.

Konzervativní vyhlazení

Dalším nelineárním filtrem je tzv. „konzervativní vyhlazení“. Jedná se o rychlou metodu určenou k odfiltrování izolovaných pixelů s výjimečně vysokou nebo nízkou intenzitou (například šum typu sůl a pepř). Tato metoda se snaží nejprve detekovat vadný pixel. Tedy nejdříve nalezne minimální a maximální hodnotu intenzity z okolí pixelu, jehož hodnotu chceme spočítat. Jestliže je hodnota tohoto centrálního pixelu mezi minimem a maximem, je pixel předán do výstupního obrazu beze změny. V opačných případech by měl být pixel nahrazen novou hodnotou, jež je odvozena z nepostížených sousedních pixelů. Například v případě, že je jeho hodnota menší než minimum, je nahrazen tímto minimem. Naopak, je-li jeho hodnota větší než maximum, je nahrazen maximem.

4.3 Ostatní filtry a funkce

4.3.1 Grayscale

První možností je redukce plně barevného modelu na stupně šedi. Stupně šedi jsou většinou vzorkovány na 8 bitů, což dává standardních 256 úrovní šedi. Pro převod se používá empirický vztah [17]:

$$I = 0.299R + 0.587G + 0.114B$$

4.3.2 Prahování

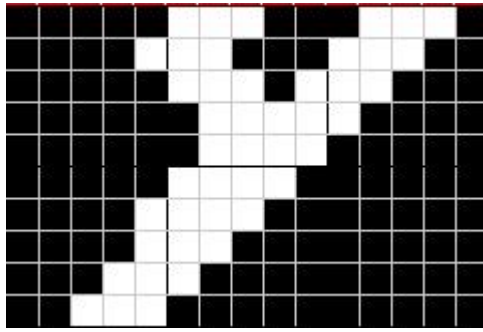
Prahování se používá k převedení obrazu s více úrovněmi jasu (šedotónového obrazu) na obraz, ve kterém se vyskytují dvě jasové úrovně (černá a bílá). Bodům s hodnotou jasu větší než určitá hodnota - práh (threshold) je přiřazena hodnota 1, ostatním bodům je přiřazena 0. Zda budeme považovat popředí za 1 a pozadí za 0, stanovíme na základě toho, co nás v obraze zajímá. Prahování může být globální nebo lokální. U globálního prahování je nalezena nebo zvolena určitá hodnota prahu T a hodnotám jasu v obraze, pro které platí $f(x, y) \geq T$, je přiřazena 1, jinak 0 tato hodnota prahu T je použita pro celý obraz. U lokálního prahování je hodnota prahu T měněna (adaptována) - v různých částech obrazu se používá různá hodnota prahu. Prahování můžeme zapsat jako:

$$g(x, y) = \begin{cases} 1, & \text{když } f(x, y) \geq T \\ 0 & \text{jinak} \end{cases}$$

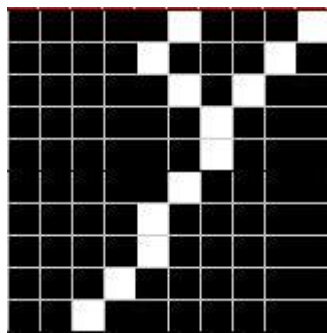
Práh je možno stanovit různými způsoby. Na základě výběru hodnot jasu v obraze nebo z histogramu je možno vybrat práh tak, že zkoušením několika různých hodnot prahů najdeme hodnotu, která dává nejlepší výsledek. Tento přístup může být úspěšný v interaktivním prostředí, kde uživatel může nastavovat práh a okamžitě vidět výsledek [17].

4.3.3 Skeletonizace

Tento algoritmus slouží ke ztenčení hran na 1 px. Nejlépe pracuje v prahovaném obraze, kdy kontroluje pouze dvě barvy. Je založen na průchodu obrazem a kontrole následujícího pixelu za aktuálním pixelem podle jeho barevné hodnoty určí, zda se má tento pixel zanést jako hrana nebo ne. Pokud máme bílé hrany, které chceme ztenčit, a černé pozadí viz. obr. 4.7, funguje algoritmus takto. Procházíme horizontálně od kraje pixely, pokud je černý zanes do nového obrazu tento pixel. Pokud narazí na první bílý pixel, zanes je také do obrazu. Nyní přeskoč všechny bílé pixely, dokud nenarazíš znovu na černý, který se v novém obraze zanes za bílý pixel. Takto se projde celý obraz viz. obr. 4.8.



Obrázek 4.7: Původní obraz



Obrázek 4.8: Skeletonizovaný obraz

Kapitola 5

Implementace

V této kapitole je popsána samotná implementace systému na rozpoznávání sítnice oka. Jsou zde popsány použité technologie pro vytvoření tohoto programu. Dále jsou zde diskutována nastavení jednotlivých funkcí z knihovny OpenCV. Jsou zde uvedeny přednosti, omezení programu a řešení některých problémů a rozdíly oproti návrhu.

5.1 Použité nástroje

Program byl z důvodů přenositelnosti a použití ve vestavěných systémech psán v jazyce C++. Kromě vlastních zdrojových souborů je použita knihovna OpenCV, konkrétně verze 1.0¹

Tato knihovna je využívána zejména pro své výborné vlastnosti v oboru zpracování obrazu. Knihovna poskytuje řadu důležitých funkcí, které usnadní extrakci cévního systému. Její použití je velmi jednoduché a přímočaré. O některých použitých funkcích si povíme v následující části. Jako programovací prostředí bylo zvoleno Visual Studio 2005. To umožňuje jednoduché použití knihoven OpenCV.

5.2 OpenCV

V této části se podrobněji seznámíme s knihovnou OpenCV a jejími funkcemi. Jelikož potřebujeme z obrazu sítnice vyextrahovat cévní systém, je potřeba na obraz aplikovat ve správném pořadí a se správnými parametry funkce této knihovny. Nebudeme se zabývat funkcemi, které jsou pomocné a slouží k načítání, kopírování obrázku atd.

5.2.1 cvCvtColor

Většina funkcí z knihovny OpenCV může být aplikována pouze na obraz ve stupních šedi, proto je nutné aplikovat na obraz funkci `cvCvtColor()`. Ta převádí obraz z jednoho barevného prostoru na druhý. Prvními dvěma parametry jsou vstupní a výstupní obraz, třetím a posledním parametrem je typ převodu. Typů převodu existuje celá řada. Pro tento případ je použit typ převodu `CV_BGR2GRAY`, ten převede obraz z RGB kanálu do stupňů šedi.

¹Dostupná na adrese: http://sourceforge.net/projects/opencvlibrary/files/opencv-win/1.0/OpenCV_1.0.exe/download

5.2.2 cvSmooth

Dalším důležitým filtrem je filtr na odstranění šumu v obraze, protože obraz sítnice obsahuje chyby způsobené při snímání, nebo jde o samotné ostré přechody na očním pozadí, je nutné se těchto nedokonalostí zbavit. Pro tento účel byla vybrána funkce `cvSmooth`. Opět jde funkci, která obsahuje několik parametrů. První dva jsou vstupní a výstupní obraz, třetí a nejdůležitější je typ filtru pro odstranění šumu. Posledními parametry jsou hodnoty, které slouží pro nastavení filtrů. Na obraz je aplikován typ filtru `CV_GAUSSIAN` s parametry 11 a 3.

5.2.3 cvThreshold

Tento filtr v programu slouží k nalezení slepé skvrny a k nalezení okraje sítnice. Samotné parametry `cvThreshold` jsou následující: vstupní obraz, výstupní obraz, práh, maximální hodnota a typ prahovací funkce. Těch je opět celá řada. Pro nalezení slepé skvrny i okraje je použita funkce binárního prahování `CV_THRESH_BINARY`. Ten nastaví podle prahu barvu pixelu na bílou nebo černou.

Pro nalezení slepé skvrny bylo po dlouhém testování a zkoušení zvolena hodnota 149. Tato hodnota je dostatečně vysoká k tomu aby odhalila velmi světlou skvrnu. Při nižších hodnotách byly rozpoznány i jiné části obrazu. Naopak při vyšších hodnotách docházelo k nenalezení skvrny.

Dále je prahování využito k nalezení kraje sítnice. Ten je při aplikování algoritmu pro detekci hran zaznamenán jako hrana a je zbytečně jej ukládat do databáze. Pro tuto funkci v této části algoritmu byla zvolena hodnota 35. Jedná se opět o otestovanou hodnotu, která vyhovuje většině snímků. Pro detekci okraje je využita také následující funkce z knihovny OpenCV.

5.2.4 cvAdaptiveThreshold

Tento prahový modifikátor má prahovací funkci jako jeden ze svých parametrů. Důležitým parametrem je velikost bloku. V tomto bloku se obraz prochází pixel po pixelu a podle výběru adaptivní metody se počítá prahování.

5.2.5 cvCanny

Hranových detektorů existuje v knihovně OpenCV celá řada. Pro naši situaci je využit cannyho operátor `cvCanny`. Ten dosahuje při správných parametrech výborných výsledků narozdíl od ostatních hranových funkcí jako např. `cvSobel`. Cannyho metoda detekce hran je podobná Laplaceově algoritmu. První derivace je počítána podle x a y a poté je zkombinovaná do 4 různých směrů. Body, které se nacházejí v těchto směrech, jsou lokální maxima a jsou kandidáty na převedení do hran.

Významným rozměrem Cannyho algoritmu je, že se snaží spojovat jednotlivé pixely aspirující na hranu do kontur. Tyto kontury jsou formovány aplikací hysterezní prahovací funkce na pixely. To znamená, že existují dva prahy – spodní a horní. Jestliže pixel má gradient větší než vyšší práh, potom je přijat jako hrana. Naopak má-li hodnotu gradientu menší než nižší práh, je odmítnut. Pokud se hodnota gradientu pixelu nachází mezi těmito prahy, je přijat pouze v případě, že je spojen s pixelem, který je přijatý. Poměr mezi horním a dolním prahem by měl být přibližně 2:1 nebo 3:1.

Pro aplikaci byl horní práh nastaven na hodnotu 6 a nízký na 5. Při nastavení vyšších hodnot dochází k nenadetekování některých žil, případně k velkému počtu mezer mezi nimi. Při nižších hodnotách ovšem detekuje hrany i v černém okraji okolo sítnice.

5.3 Databáze

Nezbytnou součástí systému je i databáze obsahující informace o uživatelích a samozřejmě šablony sítnice. Existuje celá řada formátů a možností, které lze pro uložení šablony zvolit. Můžeme využít nějaký existující databázový systém (MySQL, Oracle, atd.) dále můžeme použít filesystem. Zde se dá využít dvou formátů txt a xml. V podkapitolách se zaměříme na zmíněné možnosti uložení.

5.3.1 MySQL

MySQL je databázový systém, vytvořený švédskou firmou MySQL AB. Jde o multiplatformní databázi. Komunikace s ní probíhá, jak už název napovídá, pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními.

Pro svou snadnou implementovatelnost (lze jej instalovat na Linux, MS Windows, ale i další operační systémy), výkon a především díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl v současné době používaných databázích.

MySQL bylo od počátku optimalizováno především na rychlost, a to i za cenu některých zjednodušení: má jen jednoduché způsoby zálohování, a až donedávna nepodporovalo pohledy, trigger, a uložené procedury. Tyto vlastnosti jsou doplňovány teprve v posledních letech, kdy začaly nejčastějším uživatelům produktu, programátorům webových stránek, již poněkud scházet.

Každé klientské připojení dostane uvnitř serverového procesu vlastní vlákno (thread). Dotazy tohoto připojení se vykonávají uvnitř tohoto jediného vlákna, které zase sídlí na jednom jádru nebo CPU. Protože server udržuje vlákna v cache, nemusejí se vytvářet a likvidovat pro každé nové připojení. Autentizace je založena na uživatelském jménu, hostiteli, odkud pocházejí a heslu. Dají se také používat certifikáty X509 přes připojení SSL. Jakmile se klient připojí, server pro každý dotaz vydaný klientem ověřuje, zdali má patřičná oprávnění pro akci, kterou chce vykonat [43].

Pokud se tedy rozhodneme použít tento databázový systém, můžeme aplikaci použít víceméně odkudkoliv. Tím může mít do databáze přístup více instalací tohoto programu. Díky tomu se do databáze dostane více dat. Vzniká ovšem problém, že musíme být neustále připojeni k databázovému serveru. Tudíž jakýkoliv výpadek sítě znemožní použití našeho systému. Dalším nebezpečím je nabourání databáze. Pokud by došlo k tomuto bezpečnostnímu incidentu, ztratila by databáze i systém svou důvěru. Je proto lepší použít buď lokální databázi, nebo databázi jako filesystem.

5.3.2 XML

Extensible Markup Language je obecný značkovací jazyk, který byl vyvinut a standardizován konsorciem W3C. Je zjednodušenou podobou staršího jazyka SGML. Umožňuje snadné vytváření konkrétních značkovacích jazyků pro různé účely a různé typy dat. Používá se pro serializaci dat, v čemž soupeří např. s JSON či YAML. Zpracování XML je podporováno řadou nástrojů a programovacích jazyků.

Jazyk je určen především pro výměnu dat mezi aplikacemi a pro publikování dokumentů, u kterých popisuje strukturu z hlediska věcného obsahu jednotlivých částí, nezabývá se vzhledem. Prezentace dokumentu může být definována pomocí kaskádových stylů. Další možností zpracování je transformace do jiného typu dokumentu, nebo do jiné aplikace XML.

Vznikla potřeba vytvořit nějaký jednoduchý otevřený formát, který není úzce svázán s nějakou platformou nebo proprietární technologií. Tím může být právě XML, který je založen na jednoduchém textu a je zpracovatelný (v případě potřeby) libovolným textovým editorem.

Specifikace XML konsorcia W3C je zdarma přístupná všem. Každý tak může bez problémů do svých aplikací implementovat podporu XML. To je velký rozdíl oproti firemním formátům, k nimž není k dispozici žádná dokumentace a navíc se jedná v porovnání s XML o velice složité, často binární, formáty.

Pomocí XML značek (tagů) vyznačujeme v dokumentu význam jednotlivých částí textu. Dokumenty tak obsahují více informací, než kdyby se používalo značkování zaměřené na prezentaci – definice písma, odsazení a podobně. XML dokumenty jsou informačně bohatší. To lze samozřejmě s výhodou využít v mnoha oblastech. Největší přínos bude samozřejmě pro prohledávání, kdy můžeme určit i jaký význam má mít hledaný text [44].

Abychom mohli tento formát použít, je nutné mít knihovnu pro práci s tímto formátem. Jako ideální se nabízí libxml2² nebo Expat³.

5.3.3 TXT

Textový soubor je soubor pro uchování prostého elektronického textu. Text je zde reprezentován prostou sekvencí znaků, textový soubor není vnitřně strukturovaný. Tento základní formát nepotřebuje pro svou práci speciální knihovny. Proto byl zvolen jako vhodný formát pro uložení dat. Každý uživatel má v systému uložené následující informace:

- ID
- Jméno a Příjmení
- Login
- Typ oka (levé, pravé)
- Počet záznamů šablony
- Šablona
- Pozice relativního středu slepé skvrny

5.4 Popis metod

Nyní budou podrobněji popsány důležité naimplementované třídy, metody a funkce. Výsledná implementace se mírně liší od návrhu uvedeného v kapitole 3.

²Dostupná verze na: <http://xmlsoft.org/>

³Dostupná verze na: <http://expat.sourceforge.net/>

5.4.1 Kontrola parametrů

Aby byla vůbec nějaká metoda volána a použita, je nutné zjistit, s jakými parametry uživatel program spustil. Pokud dojde ke spuštění se špatnými parametry, je o tom uživatel informován chybovou hláškou a program je ukončen. Jednotlivé parametry jsou popsány v sekci 5.5. Parametry jsou kontrolovány pomocí funkce `strcmp()` z knihovny `string.h`. Dále je testován správný počet parametrů pro jednotlivé hlavní parametry. Tato kontrolující část je aplikována i na dávkový soubor, se kterým je aplikace spuštěna, pokud nejsou zadány parametry. Ten je po řádcích rozdělen na jednotlivé parametry a uložen do této struktury viz 5.1.

Listing 5.1: Struktura argumentů

```
struct arguments_struct
{
    string arg_1;
    string arg_2;
    string arg_3;
    string arg_4;
    string arg_5;
    int arg_count;
}temp_arguments;

std::vector<arguments_struct> arguments;
```

Uložení do struktury je nasnadě. Dávkový soubor může obsahovat několik příkazů, které se musí provést. Proto se v cyklu načítají jednotlivé parametry, které jsou kontrolovány a následně prováděny.

5.4.2 Získání šablony

Aby bylo možné uložit šablonu ze vzoru, je potřeba nejprve tuto šablonu získat. Jak bylo popsáno dříve, využívají se zde zejména funkce z knihovny OpenCV. Správné pořadí tvoří tzv. pipeline. Ta je pro všechna spuštění stejná. Pořadí se nesmí změnit, mohlo by dojít ke špatné detekci nebo dokonce k žádné, a tím by funkce systému jako takového selhala. Pro pipeline není vytvořena žádná speciální třída. Je součástí hlavní třídy `main`. Metody, které jsou v pipeline použity, jsou naimplementovány ve třídě `Filters` viz 5.2. Tato třída se stará o veškeré filtry, které jsou aplikovány na obraz.

Listing 5.2: Třída Filters

```
class Filters
{
public:
    int blind_spot [2][2];
    char* blind_spot_position;

    void invert_image(CvArr* in_image, CvArr* out_image);
    void set_rgb(int height, int width, int step, int channels,
                uchar *data, int red, int green, int blue);
    void smooth_bilateral(CvArr* in_image, CvArr* out_image,
```

```

        int param1, int param2);
void smooth_gaussian(CvArr* in_image, CvArr* out_image,
                    int param1, int param2);
void ad_threshold(CvArr* in_image, CvArr* out_image,
                 double max_val, int block_size, double param1);
void remove_4_4_edges(IplImage* in_image);
void trash_matrix_x_y(IplImage* in_image, int vyska, int sirka);
void remove_trash(IplImage* in_image);
void find_blind_spot(IplImage* in_image);
void remove_eye_border(IplImage* in_image,
                       IplImage* in_image_2);
}

```

Prvním úkolem pipeline je nastavení barevných kanálů. Protože obrázek je převážně červený viz. obr. 5.1, je dobré stáhnout červený kanál, a naopak zvýšit zelený. Jako ideální se ukázaly hodnoty -50 pro červený kanál a 50 pro zelený. OpenCV umí přistupovat k jednotlivým barevným složkám pixelu jednotlivě. Nastavení je tudíž jednoduché. Ve třídě `filters` jej implementuje metoda `void set_rgb()`.

Následuje převod do stupňů šedi. K tomu slouží metoda `void invert_image()`. Ta využívá funkci `cvCvtColor()`. Parametry jsou popsány v sekci 5.2.1. Parametry funkce jsou vstupní a výstupní obrázek. Zde si musíme dát pozor, výstupní obrázek musí mít pouze 1 kanál nikoliv 3.

Po převodu je nutné odstranit šum z obrázku. Toho dosáhneme aplikací gaussovského a bilaterálního filtru `smooth_gaussian``smooth_bilateral`. Jejich aplikace je popsána v sekci 5.2.2

Nyní si v obrázku nadetekujeme slepou skvrnu viz. obr. 5.2. Pomocí ní si určíme relativní střed. Na tento bod budeme zarovnávat šablony, abychom jej mohli porovnat. Nelze provést porovnání dvou šablon, jestliže nejsou zarovnané. Dojde totiž s největší pravděpodobností k nulové shodě těchto šablon. Také nám tento bod prozradí, jedná-li se o pravé nebo levé oko. V třídě `Filters` je k tomuto nalezení určena metoda `find_blind_spot()`. Ta potřebuje pouze jeden parametr. Tím je vstupní obrázek.

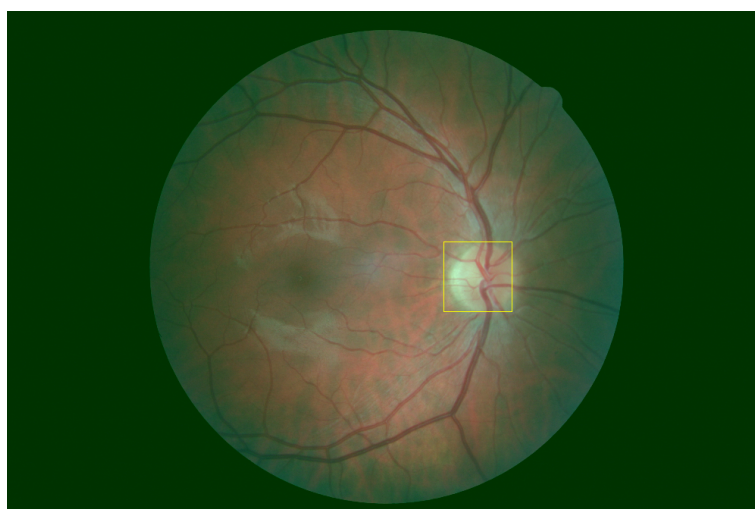
Teď už můžeme pomocí hranového operátoru nadetekovat žíly v obraze viz. obr. 5.3. Zde je operátor použit přímo a nikoliv přes třídu `Filters`.

Poté co jsme nadetkovali všechny hrany v obraze, musíme odstranit veškeré přebytečné a malé hrany a okraj sítnice viz. obr.5.4. K tomu složí metody `remove_eye_border()` a `remove_trash()`. První jmenovaná používá funkce `cvTheshold` a `cvAdaptiveThreshold` z knihovny OpenCV. Jakmile máme kraj nadetekován, dochází k jeho odstranění. To probíhá následovně. Z obrázku, kde je nadetekován kraj jsou vybrány souřadnice `x` a `z` černých bodů. Podle nich se prochází obraz s nadetekovanými hranami. A pro každý tento bod je změněno aplikována matice 5x5, kde bod je středem matice, a toto okolí je nastaveno na černou barvu včetně středového pixelu. Obdobně funguje i odstranění malých pixelů. Obraz se prochází různě velkými maticemi. Pokud jsou kraje matice černé a střed matice bílý, je tento střed začerněn. Tím dojde k odstranění malých shluků pixelů. Pokud bychom tam tyto pixely nechali, byla by velikost šablony pro jednoho uživatele přibližně 0.5 MB. Což je pro nás velmi nevýhodné. Po těchto úpravách je velikost šablony přibližně 0.1 MB.

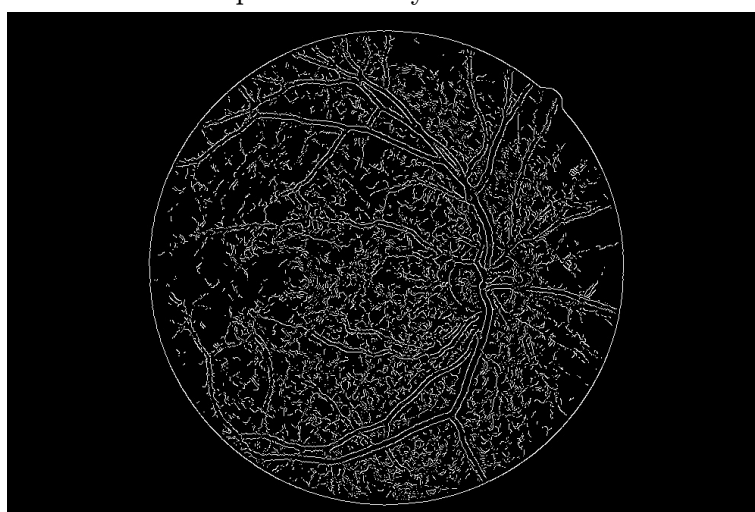
Před generováním šablony jsou ukázány průběžné výsledky úpravy obrazu.



Obrázek 5.1: Původní sítnice



Obrázek 5.2: Upravené kanály a nadetekovaná skvrna



Obrázek 5.3: Nadetekované hrany



Obrázek 5.4: Předloha pro šablonu

Nyní se již z takto upraveného obrazu dá vygenerovat šablona. K tomu je využita třída `Template`. Ta vypadá takto viz [5.3](#).

Listing 5.3: Třída `Template`

```
class Template
{
public:
    int count_of_entries;
    std::vector<CvPoint> retina_veins;
    std::vector<CvPoint> retina_veins2;
    int white_pix_count;

    struct retina_template
    {
        int id;
        string name;
        string login;
        int white_pix_count;
        string eye_type;
        std::vector<CvPoint> pixels;
        int blind_x;
        int blind_y;
    }pom_ret_temp;

    std::vector<retina_template> ret_temp;

    void generete_template(IplImage* in_image);
    void save_template(string name, string surname, string login,
        int pix_c, string type, std::vector<CvPoint> ret_vet,
        int spot_x, int spot_y);
```



```

void get_all_templates ();
int compare_templates(IplImage* in_image ,
                      std::vector<CvPoint> veins , int in_blind_x ,
                      int in_blind_y , int temp_blind_x , int temp_blind_y ,
                      int temp_pix_count , int treshold );
int select_by_login(string login);
int get_last_id ()
}

```

Pro generování šablony je z této třídy použita funkce `genereta_template()`. Princip fungování je jednoduchý. Algoritmus prochází obraz a hledá bílé pixely. Ty se ukládají do `retina_veins`, což je pomocné pole struktur typu `cvPoint`. Při každém nalezení bílého pixelu se zvedá čítač pixelu `white_pix_tmpl_count`. Tento počet je důležitý, jak pro ukládání šablony, tak i pro porovnání. Ukázka šablony viz [5.4](#).

Metoda `save_template` ukládá šablonu do souboru `retina_templates.txt`, který je naší databází. Jednotlivé body jsou odděleny : a ;.

Listing 5.4: Ukázka šablony

```

7
jaroslav struzka
xstruz
11092
righth
479:33;
480:33;
478:34;
.
.
.
538:629;
539:629;
661:306

```

Tato třída implementuje dále metodu na získání šablony `get_all_templates`. Ta prochází celý soubor s databází šablon a ukládá jednotlivé položky do pole struktur `ret_temp`.

Další metody slouží jako databázové operace např. nalezení posledního vloženého záznamu, nalezení záznamu podle loginu apod.

Poslední a nejdůležitější metodou je `compare_templates`. Ta porovnává šablonu s vstupujícím obrazem. Nejprve se musí obraz zarovnat podle relativního středu. Toto zarovnání nemusí být vždy dokonalé. Proto nemůžeme porovnávat pixel ze šablony s jedním pixelem z obrazu. Musíme okolí rozšířit o 1-2 pixely ze všech stran. Takto se porovnají všechny pixely ze šablony. Při shodě se zvedá čítač shodných pixelů. Po průchodu celou šablonou je spočítána procentuální shoda šablony s obrazem.

5.5 Parametry

Spojení GUI systému, které poskytuje Visual Studio 2005, s knihovnou OpenCV je nekompatibilní. OpenCV sice poskytuje také určitou možnost tvorby uživatelského rozhraní, ale

i u toho použití dochází k nepříjemnostem. Proto je program koncipován jako konzolová aplikace. Pro spuštění existuje několik parametrů, které jsou popsány dále.

- bez parametru – V případě, že je program spuštěn bez parametru. Spustí se příkazy v dávkovém souboru `config.txt`. Jednotlivé příkazy musí být na řádcích a oddělené středníkem.

```
enroll c:\\IMG_6565.jpg petr dufek xdufek;
printDB;
```

- `help` – Program vypíše nápovědu obsahující popis parametrů a potřební informace o spuštění.
- `printDB` – V tomto případě program vypíše přehlednou tabulku s jednotlivými uživateli, kteří jsou zaregistrováni v systému.
- `enroll` – Tento parametr umožňuje zaregistrovat uživatele do systému. Za tímto parametrem musí dále následovat umístění obrazu se sítnicí. Dále jméno a příjmení. Posledním parametrem je login, pod kterým bude uživatel znám. Dochází ke kontrole loginu s ostatními loginy v databázi, aby nevznikla nekonzistence existencí dvou stejných loginů. Příklad příkazu:

```
enroll c:\\IMG_6565.jpg petr dufek xdufek
```

- `identify` – Při tomto parametru prochází program celou databází sítnic a porovnává ji se vzorem vytvořeného z obrazu, který je zadán jako druhý parametr. Pokud je shoda uložené šablony a vzoru větší nebo rovna prahu, jež je posledním parametrem, je vypsán login, který odpovídá shodující se šabloně, a procentuální shoda. Příklad příkazu:

```
identify c:\\IMG_6565.jpg 85
```

- `verify` – Poslední parametr, se kterým lze program spustit. Za ním musí následovat umístění obrazu se sítnicí, login, se kterým bude vzor porovnán, a práh. Příklad příkazu:

```
verify c:\\IMG_6565.jpg xstruz 68
```

5.6 Shrnutí

Implementace celého systému byla relativně bezproblémová. Největší úskalí a časová náročnost byla zejména na nalezení vhodných funkcí z knihovny OpenCV a jejich následné správné nastavení.

Kapitola 6

Experimenty a testování

V této kapitole si rozebereme prováděné experimenty a testování programu pro rozpoznávání podle sítnice oka. Cílem je určit jednotlivé míry spolehlivosti uvedené v kapitole 2.4.9 a následně zpracovat ROC křivku.

6.1 Zařízení a zisk vzorků

Abychom mohli vůbec něco testovat, je zapotřebí mít určitý rozsah vzorků. Pro jejich získání bylo využito speciálního optického zařízení. Jedná se o Non-mydriatickou funduskameru Canon CR-1, která je blíže popsána v sekci 2.3.2.

Zisk obrazů sítnic probíhal ve dvou kolech. Kola byly oddělena větší časovou mezerou. Důvodem je pozorování, zda v tomto období dojde ke změnám, které se mohou projevit při použití rozpoznávacího systému.

První kolo proběhlo 1.12.2009, a sloužilo k zisku reprezentativního vzorku. Bylo nasnímano celkem 10 osob. Od každé osoby levé a pravé oko. Tyto data sloužila k vývoji aplikace a jako data testovací.

Druhé snímání proběhlo 11.5.2010. Bohužel se část lidí nemohla na toto druhé kolo dostavit. Přesto bylo množství získaných vzorků dostačující na provádění testování. U každé osoby bylo po časových intervalech nasnímano celkem 30 vzorků z každého oka. Časový interval a množství bylo zvoleno z bezpečnostních důvodů. Při každém snímání je vyfocena sítnice, proto musí být osvětlena bleskem. Každé takové focení je uživateli nepříjemné. Přibližně po pěti fotografiích se již začínají tvořit mžítka před okem, a to i přestože byla síla blesku na velmi nízké úrovni. Snímání bylo tedy omezeno na maximálně 5 snímků v jednom kole. A minimální pauza mezi dalším snímáním byla cca. 15 minut.

Snímání ovšem neprobíhalo ideálně. Přístroj po provedení přibližně 8 fotografií odmítal pořídit snímek sítnice. Bylo nutné jej vypnout a znova zapnout. V některých případech bylo nutné přístroj restartovat dokonce 6x po sobě, než byl pořízen vzorek sítnice. Tato prodleva by byla u reálného nasazení pravděpodobně velkou překážkou.

6.2 Výsledky

Dosažené výsledky jsou prezentovány tabulkami a grafy. Pro jednotlivé míry spolehlivosti je to tabulka 6.1 a graf 6.3. K hodnotám FMR a FNMR je třeba dopočítat míru FTE. Míru FTA zanedbáváme, jelikož jsou testy prováděny s nasnímanými sítnicemi. Z celkového počtu 148 sítnic se u 4 nedokázala vytvořit šablona, proto $FTE = 100 * 4/148 = 2,7\%$.

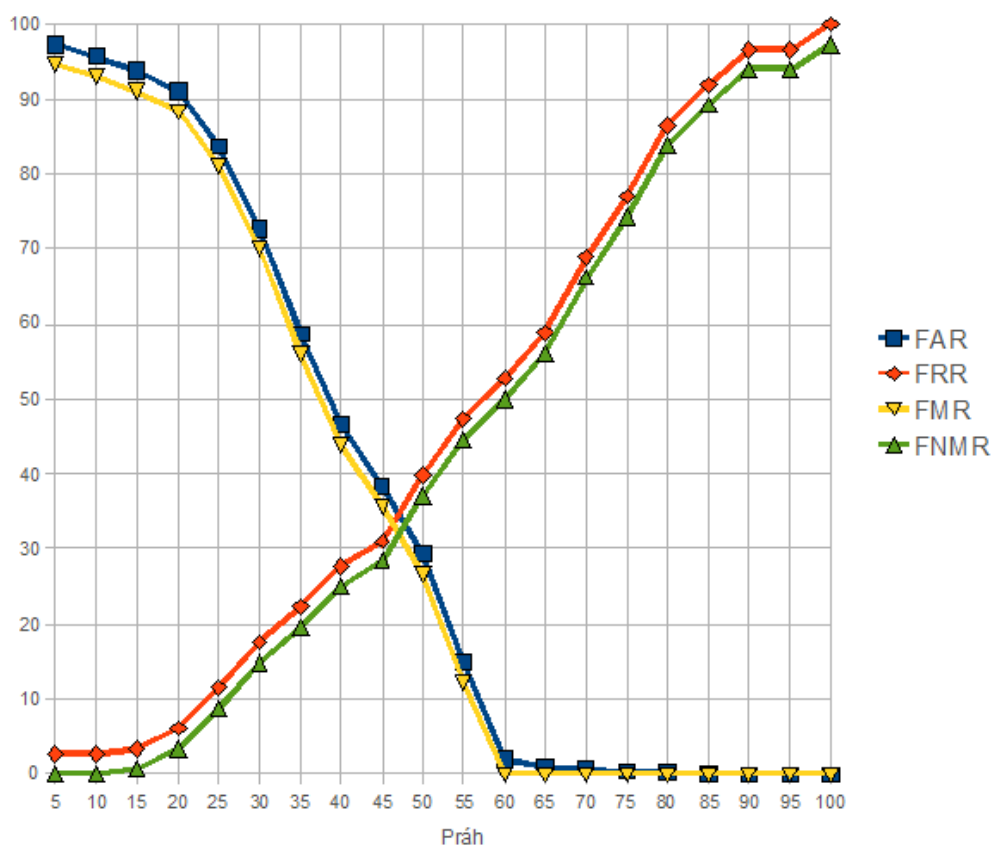
Tím dostanem míry FMR a FNMR. Z nich byla vytvořena tabulka pro ROC křivku 6.2 a následně ROC křivka .

PRÁH	FAR	FRR	FMR	FNMR
5	97,3	2,7	94,6	0
10	95,7	2,7	93	0
15	93,73	3,38	91,03	0,68
20	91,03	6,08	88,33	3,38
25	83,72	11,49	81,02	8,79
30	72,73	17,57	70,03	14,87
35	58,72	22,3	56,02	19,6
40	46,56	27,7	43,86	25
45	38,33	31,08	35,63	28,38
50	29,42	39,86	26,72	37,16
55	14,93	47,3	12,23	44,6
60	2,03	52,7	0	50
65	0,98	58,78	0	56,08
70	0,8	68,92	0	66,22
75	0,37	77,03	0	74,33
80	0,31	86,49	0	83,79
85	0,06	91,89	0	89,19
90	0	96,62	0	93,92
95	0	96,62	0	93,92
100	0	100	0	97,3

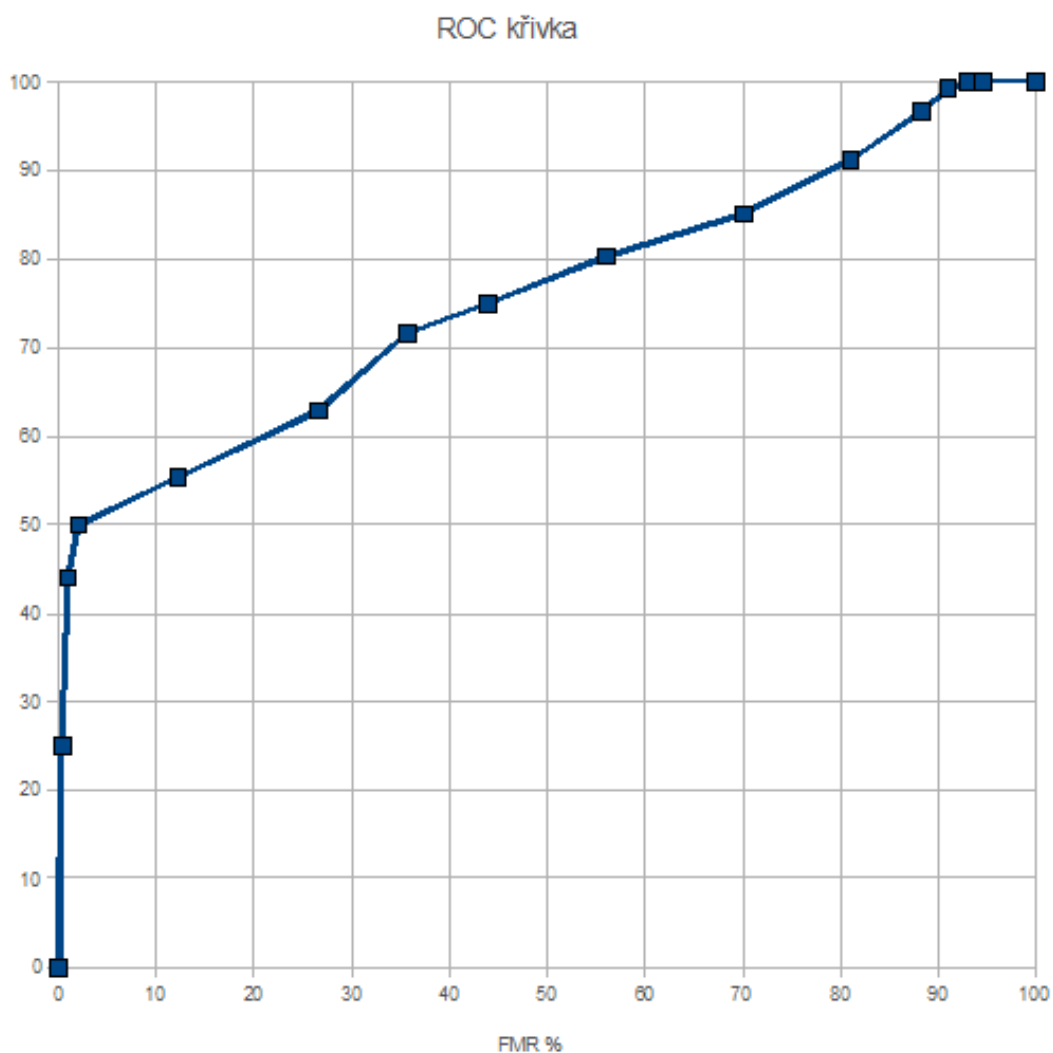
Obrázek 6.1: Tabulka s mírami

FMR	1-FNMR
0	0
0,4	25
1	44
2	50
12,23	55,4
26,72	62,84
35,63	71,62
43,86	75
56,02	80,4
70,03	85,13
81,02	91,21
88,33	96,62
91,03	99,32
93	100
94,6	100
100	100

Obrázek 6.2: Tabulka pro ROC křivku



Obrázek 6.3: Průběh statistických koeficientů



Obrázek 6.4: ROC křivka

Dosažené výsledky nejsou ideální. Může to být způsobeno mnoha faktory: špatný nebo nekvalitní snímek sítnice. Chyba v algoritmu pro získání šablony např. špatné nalezení relativního středu, chybná detekce hran.

Větší problém nastává u osob, které mají více jak $\pm 7D$. V těchto případech byly snímky neostré. Je to způsobeno buď špatnou obsluhou přístroje, nebo neschopností přístroje takto postižené oko zaostřit.

Bohužel nebyla možnost experimentovat se sítnicemi, které jsou postiženy některou z chorob popsaných v kapitole 2.2. Tyto testy by mohli být klíčové v nastavování parametrů funkcí pro zpracování.

Kapitola 7

Závěr

V současnosti se spolehlivá identifikace a verifikace osob stává nutností. Jde například o různé přístupové systémy, hraniční kontroly apod. V těchto případech už je znalost přihlašovacíh údajů nedostačující. Prolomení hesla není problém. Proto se v některých případech přistoupilo k testování biometrických vlastností.

Tato práce se zabývá málo používanou formou testování prostřednictvím sítnice oka. Nejde o vytvoření komplexního systému pro rozpoznávání sítnice oka, ale pouze o zpracování již nasnímaného obrazu. Vytvoření takto komplikovaného systému by mohlo být pro svou obtížnost a náročnost zvoleno jako téma disertační práce.

Z pohledu zadání diplomové práce byly splněny všechny požadavky, které byly na tuto práci kladené. Počínaje prostudováním problematikou lidského oka a metodami rozpoznávání. Dalším bodem zadání bylo vytvořit návrh implementovaného systému. Ten je popsán pomocí jazyka UML v kapitole 3. Podle něj byl následně systém implementován. Součástí práce je také vytvoření rozsáhlejší databáze sítnic. Ta je potřebná k otestování správné funkce systému. Průběh snímání a dosažené výsledky jsou popsány v předposlední kapitole. V této kapitole jsou také podrobně popsány výsledky implementovaného systému.

Jako rozšíření se nabízí možnost připojení k databázovým systémům, vytvoření příznivějšího GUI dále využít umělou inteligenci pro simulaci poškození sítnic.

Literatura

- [1] Beránek, J.: *Metody detekce a reprezentace hran v obraze [BP]*. FIT VUT v Brně, 2007.
- [2] Bospene, E. B.: *Eye Cancer Research Progress*. Nova Publishers, 2008, iSBN 978-1-60456-045-9.
- [3] Drahanský, M.: *Biometrické systémy*. FIT VUT v Brně, 2006, dokument dostupný na URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/BIO-IT/texts/08_Rozpoznavani_oka.pdf (prosinec 2009).
- [4] Drahanský, M.: *Biometrické systémy*. FIT VUT v Brně, 2006, dokument dostupný na URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/BIO-IT/texts/04_Hodnoceni_spolehlivosti.pdf (prosinec 2009).
- [5] Drahanský, M.: *Biometrické systémy*. FIT VUT v Brně, 2006, dokument dostupný na URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/BIO-IT/texts/02_Uvod_do_biom_systemu.pdf (prosinec 2009).
- [6] Grehn, F.; Stamper, R.: *Glaucoma Essentials in Ophthalmology*. Springer, 2006, iSBN 3-5402-6220-2.
- [7] Kittner, O.; Langmeier, M.; Marešová, D.; aj.: *Základy lékařské fyziologie*. Univerzita Karlova v Praze, 1.lékařská fakulta, Fyziologický ústav, 2009, iSBN 978-80-247-2526-0.
- [8] Kolář, P.; Beránek, M.; Došková, H.; aj.: *Věkem podmíněná makulární degenerace*. Grada Publishing, 2008, iSBN 978-80-247-2605-2.
- [9] Lalonde, M.; Gagnon, L.; Boucher, M.-C.: *Non-recursive paired tracking for vessel extraction from retinal images*. 2000, dokument dostupný na URL <http://www.crim.ca/perso/langis.gagnon/articles/vi2000.pdf> (prosinec 2009).
- [10] Lodrová, D.: *Rozpoznávání živosti otisků prstů [DP]*. FIT VUT v Brně, 2006.
- [11] Merkunová, A.; Orel, M.: *Anatomie a fyziologie člověka*. Grada Publishing, 2008, iSBN 978-80-247-1521-6.
- [12] Odstrčilík, J.: *Analýza barevných snímků sítnice se zaměřením na segmentaci cévního řečiště [DP]*. FEKT VUT v Brně, 2008.
- [13] Rhee, D. J.: *Glaucoma: color atlas & synopsis of clinical ophthalmology*. McGraw-Hill Professional, 2003, iSBN 0-07-137597-X.

- [14] Rosypal, S.: *Nový přehled biologie*. Scientia, 2007, iISBN 80-86960-23-4.
- [15] Sukkaew, L.; Uyyanonvara, B.; Barman, S. A.; aj.: *Automated Vessels Detection on Infant Retinal Images*. ISSAC, 2004, dokument dostupný na URL <http://www.siit.tu.ac.th/bunyarit/publications/2004%20-%2005%20-%20ICCAS04%20Automated%20Vessels%20Detection%20on%20Infant%20Retinal%20Images.pdf> (prosinec 2009).
- [16] Valešová, L.; Hycl, J.: *Diabetická retinopatie*. Triton, 2002, iISBN 80-7254-233-3.
- [17] Vrbová, B.: *Základy počítačové grafiky - Studijní opora*. FIT VUT v Brně, 2006, dokument dostupný na URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/IZG-IT/texts/izg_opora.pdf (prosinec 2009).
- [18] WWW stránky: Canon. http://www.usa.canon.com/html/industrial_medeq/cf1.html.
- [19] WWW stránky: Eye cancer network. <http://www.eyecancer.com/Patient/Condition.aspx?nID=62&Category=Choroidal+Tumors&Condition=Choroidal+Melanoma>.
- [20] WWW stránky: Healthy sight institute. http://www.healthysightinstitute.com/Obstacles/2.3.3.0_macular_degeneration.htm.
- [21] WWW stránky: Lekari online. <http://www.lekari-online.cz/ocni-lekarstvi/novinky/degenerace-zlute-skvrny>.
- [22] WWW stránky: Masarykův onkologický ústav. <http://www.mou.cz/cz/co-jsou-nadory/article.html?id=174>.
- [23] WWW stránky: Neusser augenarzt. <http://www.neusser-augenarzt.de/glaukom.html>.
- [24] WWW stránky: Ordinace. <http://www.ordinace.cz/clanek/diabeticka-retinopatie/>.
- [25] WWW stránky: Simple Anatomy of the Retina. <http://webvision.med.utah.edu/sretina.html>.
- [26] WWW stránky: Visionář. <http://www.visionar.cz/?p=29>.
- [27] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Eye>.
- [28] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/Lidskéoko>.
- [29] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Pupil>.
- [30] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/Rohovka>.
- [31] WWW stránky: Wikipedia. [http://en.wikipedia.org/wiki/Iris\(anatomy\)](http://en.wikipedia.org/wiki/Iris(anatomy)).
- [32] WWW stránky: Wikipedia. [http://en.wikipedia.org/wiki/Lens\(anatomy\)](http://en.wikipedia.org/wiki/Lens(anatomy)).
- [33] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Vitreoushumour>.

- [34] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Retina>.
- [35] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/%C5%BDlut%C3%A1skvrna>.
- [36] WWW stránky: Wikipedia. [http://en.wikipedia.org/wiki/Blind_spot_\(vision\)](http://en.wikipedia.org/wiki/Blind_spot_(vision)).
- [37] WWW stránky: Wikipedia. http://en.wikipedia.org/wiki/Macular_degeneration.
- [38] WWW stránky: Wikipedia. http://en.wikipedia.org/wiki/Diabetic_retinopathy.
- [39] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/Glaukom>.
- [40] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Toxoplasmosis>.
- [41] WWW stránky: Wikipedia. <http://en.wikipedia.org/wiki/Biometrics>.
- [42] WWW stránky: Wikipedia. http://cs.wikipedia.org/wiki/Unified_Modeling_Language.
- [43] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/Mysql>.
- [44] WWW stránky: Wikipedia. <http://cs.wikipedia.org/wiki/Xml>.
- [45] Zendulka, J.; Bartík, V.; Šárka Květoňová: *Analýza a návrh informačních systémů - Studijní opora*. FIT VUT v Brně, 2006, dokument dostupný na URL <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/AIS-IT/texts/AIS.pdf> (prosinec 2009).
- [46] Říhová, E.: *Uveitidy*. Grada Publishing, 2009, iISBN 978-80-247-2897-1.

Dodatek A

Obsah CD

- text této diplomové práce ve formátu pdf a ve formě zdrojových souborů pro \LaTeX
- zdrojové soubory programu
- uživatelský manuál
- soubor readme.txt s popisným seznamem obsahu CD