



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLGIÍ**

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

POKROČILÉ METODY SEGMENTACE CÉVNÍHO ŘEČIŠTĚ NA FOTOGRAFIÍCH SÍTNICE

ADVANCED RETINAL VESSEL SEGMENTATION METHODS IN COLOUR FUNDUS IMAGES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. ONDŘEJ SVOBODA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JAN ODSTRČILÍK

BRNO 2013



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav biomedicínského inženýrství

Diplomová práce

magisterský navazující studijní obor
Biomedicínské inženýrství a bioinformatika

Student: Bc. Ondřej Svoboda

ID: 115117

Ročník: 2

Akademický rok: 2012/2013

NÁZEV TÉMATU:

Pokročilé metody segmentace cévního řečiště na fotografiích sítnice

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s principem snímání očního pozadí pomocí fundus kamery a s vlastnostmi takto získaných obrazových dat. Zejména se zaměřte na struktury, které na snímcích reprezentují cévní řečiště. 2) Proveďte literární rešerši prací, zabývajících se segmentací těchto struktur. 3) Dle pokynů vedoucího vybrané metody nastudujte a proveďte jejich implementaci s využitím vhodných programovacích nástrojů. 4) Implementované algoritmy detailně otestujte na dostupných obrazových datech. 5) Diskutujte dosažené výsledky a zhodnoťte účinnost a využitelnost aplikovaného řešení. 6) K vytvořeným algoritmům a programovým funkcím sepište přehledný návod k obsluze.

DOPORUČENÁ LITERATURA:

[1] ODSTRČILÍK, J., et al. Improvement of Vessel Segmentation by Matched Filtering in Colour Retinal Images. IFMBE Proceedings of World Congress on Medical Physics and Biomedical Engineering, Munich 2009.

[2] SEZGIN, M., et al. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging, 13(1), pp. 146-165, January 2004.

Termín zadání: 11.2.2013

Termín odevzdání: 24.5.2013

Vedoucí práce: Ing. Jan Odstrčilík

Konzultanti diplomové práce:

prof. Ing. Ivo Provazník, Ph.D.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Segmentace cévního řečiště je častým krokem při zpracování retinálních obrazů. V dnešní době existuje řada automatických metod segmentace cévního řečiště. Tyto metody jsou založeny na mnoha přístupech. Od přizpůsobené filtrace, přes metody využívající rozpoznávání vzorů, až po algoritmy využívající klasifikace obrazu. Použití automatických metod při zpracování retinálních snímků výrazně urychluje a zjednodušuje diagnostiku retinálních onemocnění. Při zpracování automatickými segmentačními algoritmy je jednou ze stěžejních částí prahování obrazu, a právě prahování fundus snímků se věnuje tato práce. Je zde popsána řada prací využívajících globální a lokální prahovací metody, a zejména metody klasifikace obrazu pro segmentaci cévního řečiště ze snímků sítnice. Následně byla na výsledky dvou metod segmentace cévního řečiště použita metoda klasifikace obrazu s učením. Z dosažených výsledků byla posléze stanovena schopnost daných metod segmentovat cévní řečiště. Použitím klasifikace obrazu namísto globálního prahování došlo u první metody na zdravé části databáze k poklesu sensitivity na 63,32 % a přesnosti na 94,99 %. Naopak u specifacity byl zaznamenán nárůst na 95,75 %. U druhé metody bylo dosaženo sensitivity 69,24 %, specifacity 98,86 % a přesnosti 95,29 %. Kombinací výsledků obou metod bylo dosaženo sensitivity 72,48 %, specifacity 98,59 % a výsledné přesnosti 95,75 %. Tímto nebyl s použitím daného klasifikátoru potvrzen předpoklad, že klasifikace obrazu s učením je oproti prostému prahování efektivnější. Zároveň bylo však prokázáno, že rozšíření příznakového vektoru kombinací výsledků z obou metod došlo k nárůstu sensitivity, specifacity i přesnosti.

Klíčová slova: segmentace cévního řečiště, fundus snímky, retinální snímky, segmentace obrazu, prahování obrazu, klasifikace obrazu, klasifikace fundus snímků, klasifikace obrazu s učením

Abstract

Segmentation of vasculature tree is an important step of the process of image processing. There are many methods of automatic blood vessel segmentation. These methods are based on matched filters, pattern recognition or image classification. Use of automatic retinal image processing greatly simplifies and accelerates retinal images diagnosis. The aim of the automatic image segmentation algorithms is thresholding. This work primarily deals with retinal image thresholding. We discuss a few works using local and global image thresholding and supervised image classification to segmentation of blood tree from retinal images. Subsequently is to set of results from two different methods used image classification and discuss effectiveness of the vessel segmentation. Use image classification instead of global thresholding changed statistics of first method on healthy part of HRF. Sensitivity and accuracy decreased to 62,32 %, respectively 94,99 %. Specificity increased to 95,75 %. Second method achieved sensitivity 69.24 %, specificity 98.86% and 95.29 % accuracy. Combining the results of both methods achieved sensitivity up to 72.48%, specificity to 98.59% and the accuracy to 95.75%. This confirmed the assumption that the classifier will achieve better results. At the same time, was shown that extend the feature vector combining the results from both methods have increased sensitivity, specificity and accuracy.

Keywords: vessel segmentation, fundus images, retinal images, image segmentation, image thresholding, image classification, fundus image classification, supervised image classification

SVOBODA, O. Pokročilé segmentační techniky a prahování objektů na fotografiích sítě. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2013. 69 s. Vedoucí diplomové práce Ing. Jan Odstrčilík.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma Pokročilé segmentační techniky a prahování objektů na fotografiích sítnice jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

V Brně dne 24. května 2013

.....
podpis autora

Poděkování

Děkuji vedoucímu Ing. Janu Odstrčilíkovi za účinnou pomoc, cenné rady, pedagogické vedení i trpělivost během celé tvorby diplomové práce. Dále děkuji Ing. Jiřímu Sekorovi za spolupráci při využívání CML serveru.

V Brně, dne 24. května 2013

.....
podpis autora

Obsah

1. ÚVOD	11
2. SNÍMÁNÍ OČNÍHO POZADÍ FUNDUS KAMEROU	13
2.1. Digitální fundus kamera	13
2.2. Princip snímání očního pozadí pomocí fundus kamery	15
2.3. Vlastnosti obrazových dat z fundus kamery a struktury reprezentující cévní řečiště	16
2.4. Využití charakteristických struktur ve fundus snímcích k diagnostice patologií	18
3. SEGMENTAČNÍ TECHNIKY	20
3.1. Obecný úvod	20
3.2. Metody segmentace struktur reprezentujících cévní řečiště	23
3.3. Detailní popis zvolených metod segmentace cévních struktur	27
4. IMPLEMENTACE ZVOLENÉHO PŘÍSTUPU	29
PRAHOVÁNÍ SNÍMKŮ	29
4.1. Popis klasifikátorů	29
4.2. Software WEKA	33
4.3. Implementace klasifikačního modelu	37
4.3.1. Hlavní funkce <i>main.m</i>	39
4.3.2. Klasifikační rámec.....	41
5. VÝSLEDKY A JEJICH DISKUZE	43
6. ZÁVĚR.....	52
7. SEZNAM POUŽITÉ LITERATURY	55
8. POUŽITÉ ZKRATKY A SYMBOLY	60
9. SEZNAM PŘÍLOH.....	61

Seznam obrázků

Obr. 1: Schéma fundus kamery: 1 – zobrazovací jednotka; 2-4 – poziční jednotka; 5 – operační panel pro lékaře; 6 – objektiv; 7 – počítač [19].....	13
Obr. 2: a) Promítnutí FOV na sítnici; vliv FOV na množství zobrazené sítnice a) 20°, b) 40°, c) 60°[21].....	13
Obr. 3: Digitální fundus kamera Canon CF-60 UVi [26]	14
Obr. 4: Barevný obrázek oka s diabetickou retinopatií z digitální fundus kamery	16
Obr. 5: a) zobrazení tepen a žil na fundus snímku; b) zobrazení centrálního reflexu.....	17
Obr. 6: Diabetický snímek v jednotlivých obrazových kanálech: a) zelený; b) červený; c) modrý	17
Obr. 7: Fundus snímek levého oka pacienta s diabetickou retinopatií a) RGB prostor ; b) zelený obrazový kanál	18
Obr. 8: Fundus snímek pravého oka pacienta s glaukomem a) RGB prostor ; b) zelený obrazový kanál	19
Obr. 9: Příklad zlatého standardu levého zdravého oka.....	20
Obr. 10: Výstupy z metody 1: a) přizpůsobené filtry; b) binární reprezentace snímku a).....	27
Obr. 11: Klasifikátory založené na rozhodovacích stromech: a) klasický rozhodovací strom; b) stejný rozhodovací strom interpretovaný jako střídavý strom; c) obecný střídavý strom [69]	31
Obr. 12: Klasifikátor Naive Bayes interpretován jako Baesova síť [61]	32
Obr. 13: Úvodní obrazovka softwaru WEKA 3, výběr pracovního módu.....	33
Obr. 14: Pracovní mód Explorer v softwaru WEKA	34
Obr. 15: Příklad struktury souboru arff.....	35
Obr. 16: Příklady výstupů MFR z metody [15]: a) snímek levého zdravého oka; b) snímek pravého oka postiženého diabetickou retinopatií.....	37
Obr. 17: Příklady výstupů jednotlivých vrstev Gaussovy pyramidy z metody [16] převzorkované na původní velikost: a) první vrstva; b) druhá vrstva; c) třetí vrstva; d) čtvrtá vrstva	38
Obr. 18: Základní blokové schéma funkce mainJOAB.m.....	40
Obr. 19: a) histogram původní trénovací sady; b) histogram zmenšené trénovací sady.....	41
Obr. 20: Základní blokové schéma klasifikačního rámce	42
Obr. 21: a) MFR; b) klasifikovaný snímek s původní maskou; c) klasifikovaný snímek se zmenšenou maskou	44

Obr. 22: Snímek 07-image_0964: a) výsledek varianty 1; b) zlatý standard z databáze	45
Obr. 23: Snímek 06-image_0888: a) výsledek varianty 2; b) zlatý standard z databáze	46
Obr. 24: Snímek 08-image_0980: a) výsledek varianty 3; b) zlatý standard z databáze	48
Obr. 25: Výsledky jednotlivých kombinací výsledků na snímku 07-image_0964: a) varianta 1; b) varianta 3; c) varianta 2; d) zlatý standard	49
Obr. 26: Snímek 06-ote_R_1 zatížený diabetickou retinopatií: a) výsledek kombinace metody 1 + 2; b) zlatý standard z databáze	50
Obr. 27: Snímek 06-image_0688 zatížený glaukomem: a) výsledek kombinace metody 1 + 2; b) zlatý standard z databáze.....	51

Seznam tabulek

Tab. 1: Úspěšnosti jednotlivých klasifikátorů na výřezu ze snímku 06-image_0888.jpg.....	29
Tab. 2: Výsledky jednotlivých klasifikátorů na snímku 06-image_0888.jpg ze zdravé části databáze HRF	43
Tab. 3: Vliv upravené masky na sensitivitu, specifickou a přesnost	44
Tab. 4: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 1 – výsledky metod 1 + 2	46
Tab. 5: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 2 – pouze výsledky metody 1	47
Tab. 6: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 3 – pouze výsledky metody 2	48
Tab. 7: Výsledky zvoleného klasifikátoru na snímcích z retinopatické části databáze, varianta metoda 1 + 2.....	50
Tab. 8: Výsledky zvoleného klasifikátoru na snímcích z glaukomatické části databáze, varianta metoda 1 + 2.....	51

1. Úvod

Diagnostika onemocnění očního aparátu a některých chorob cévního řečiště je v dnešní době značně usnadněna pomocí neinvazivního snímání očního pozadí digitálními fundus kamerami. Z fundus snímků lze diagnostikovat řadu onemocnění, můžeme uvést například glaukom [1], makulární degeneraci [2], [3], diabetickou retinopatii [4], případně i arteriosklerózu [5]. U pacientů, patřících do skupin s vyšší prevalencí onemocnění očního aparátu, je nezbytné provádět preventivní screeningová vyšetření např. pomocí fundus kamer, neboť ve chvíli, kdy již pacient pozoruje zhoršené vidění, bývá již často poškození nevratné.

Velkou mírou přispívá k podpoře diagnostiky automatické zpracování, klasifikace a diagnostika snímků. Existují algoritmy pro automatickou lokalizaci optického disku [6], [7], detekci nervových vláken [8], segmentaci cév a struktur náležících k cévnímu řečišti [9], [10], [11], případně jsou dostupné algoritmy pro automatickou klasifikaci snímků na zdravé a patologické [12].

O principech snímání digitální fundus kamerou a vlastnostech takto sejmutých snímků pojednává druhá kapitola. Dále jsou zde popsány charakteristické a případné patologické struktury, které lze na fundus snímcích pozorovat. Následně je popsána možnost využití patologických struktur k diagnostice onemocnění očního aparátu a zásobovacích cév.

V této práci je jako stěžejní část segmentace cévního řečiště z retinálních snímků diskutováno prahování obrazu. A to zejména z důvodu, že při nesprávně nastaveném prahu se může ze snímku odstranit diagnosticky relevantní informace. Ve třetí kapitole jsou proto nejprve obecně popsány segmentační techniky cévního řečiště z fundus snímků. Poté je uveden příklad rozdělení segmentačních technik. Následně je provedena rešerše prací, které se zabývají segmentací cévního řečiště z retinálních snímků. V literární rešerši jsou práce z hlediska prahování rozděleny do tří částí. První část se zabývá metodami segmentace cévního řečiště využívajícími globálního prahování, druhá část se pak zabývá metodami využívajícími lokálního prahování a ve třetí části jsou popsány techniky využívající k segmentaci cévního řečiště klasifikace obrazu. Dle práce [13] by měla klasifikace obrazu s učením dosahovat ve srovnání s globálním a lokálním prahováním lepších výsledků. Tento předpoklad je dán tím, že algoritmus vytvoří (naučí) na základě předložených vzorů požadovaný klasifikátor. Vzory obsahují jistou „vlastnost“, na jejíž predikci je daný klasifikátor naučen. Daná „vlastnost“ může být v obraze více či méně patrná. Pro schopnost klasifikovat i snímky, ve kterých je „vlastnost“ méně patrná, je vhodné vytvořit požadovaný

klasifikátor za pomoci několika snímků, které tuto „vlastnost“ obsahují. Tím, že je klasifikátor vytvořen na dostatečně velké množině trénovacích snímků, lze zajistit správnou klasifikaci i méně kvalitních snímků. Existují práce, jako například [14], ve kterých byl klasifikátor trénován na množině několika set tisíc vzorů. Takto vytvořený klasifikátor již můžeme považovat za velmi kvalitní [13]. V návaznosti na tyto předpoklady byla v této práci pro prahování zvolena právě klasifikace obrazu s učením.

Ve čtvrté kapitole je poté popsán způsob, jakým byl implementován postup využívající k segmentaci cévního řečiště klasifikaci obrazu s učením. Rovněž jsou zde popsány testované klasifikátory a klasifikační software WEKA, jehož knihovny byly importovány a následně využívány ke klasifikaci vstupních dat v programovém prostředí Matlab. Na konci kapitoly je dále uveden stručný princip funkce vytvořeného klasifikačního rámce (v textu se rovněž můžeme setkat s názvem klasifikační model, případně anglickým názvem *framework*), včetně základních vývojových schémat.

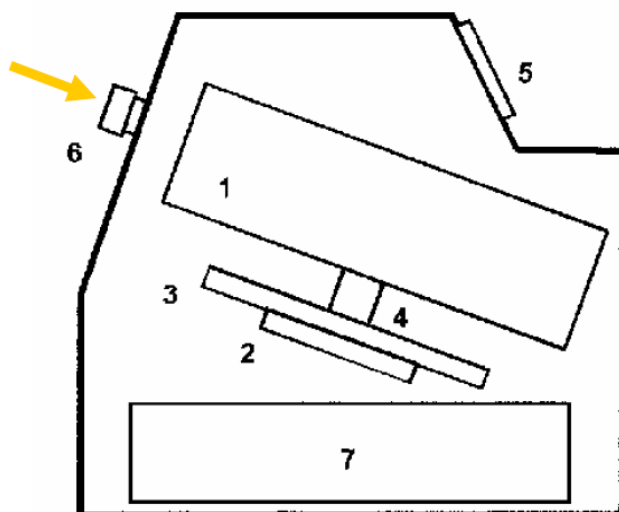
Pro trénování i testování klasifikátorů byla v této práci dostupná sada výsledků ze dvou různých metod segmentace cévního řečiště. Konkrétně se jedná o metodu JO [15] (v textu se můžeme setkat také s označením „metoda 1“) a metodu AB [16] (v textu se můžeme setkat i s označením „metoda 2“). Pro klasifikaci byly zvoleny tři kombinace vstupních dat. Nejprve byla aplikována klasifikace obrazu na každou sadu výsledků samostatně. Jelikož byly obě sady výsledků vytvořeny na databázi HRF [17], bylo možné zkombinovat obě výsledkové sady. Vznikla tak nová sada výsledků, kterou bylo možné následně testovat jako samostatnou vstupní proměnnou. Jelikož je touto kombinací vytvořen 6-ti rozměrný příznakový vektor (každý vzorek obsahuje více „vlastností“ a příslušnost do dané skupiny v obraze je proto silnější a pro klasifikátor lépe rozpoznatelná), předpokládáme, že tato kombinace bude dosahovat nejlepších výsledků. Tímto postupem byla získána rozsáhlá sada výsledků. Dosažené výsledky jsou uvedeny a diskutovány v páté kapitole.

V první příloze této práce jsou detailně okomentovány vytvořené ovládací funkce *main.m*. Ve druhé příloze je následně popsán vytvořený klasifikační rámec, včetně principu všech funkcí a možných uživatelských úprav a zásahů. Dále je zde uveden způsob implementace klasifikačního rámce pro využití při klasifikaci různých vstupních dat.

2. Snímání očního pozadí fundus kamerou

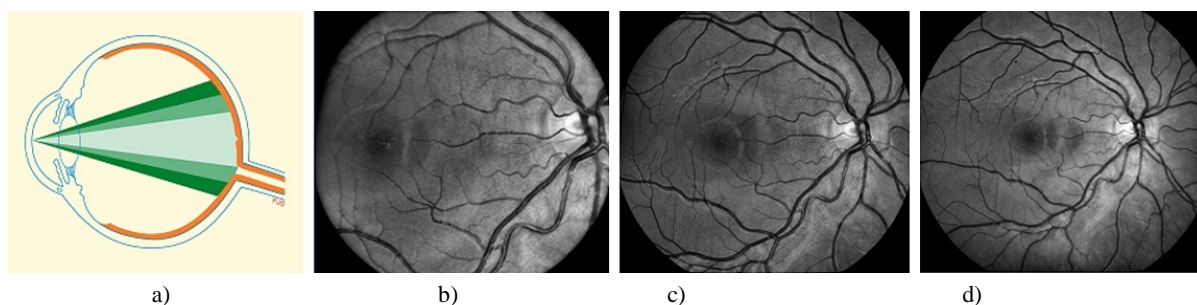
2.1. Digitální fundus kamera

Fundus kamera je speciální nízkoenergetický mikroskop, vybavený kamerou s CCD čipem, používaný v oftalmologii jako jeden z přístrojů umožňujících zobrazit sítnici. Optický princip fundus kamery je založen na nepřímé oftalmoskopii. Více o nepřímé oftalmoskopii lze nalézt v [18]. Přístroj je svým optickým uspořádáním schopen korigovat i velmi vysokou ametropii oka [19]. Schematické uspořádání digitální fundus kamery vidíme na Obr. 1.



Obr. 1: Schéma fundus kamery: 1 – zobrazovací jednotka; 2-4 – poziční jednotka; 5 – operační panel pro lékaře; 6 – objektiv; 7 – počítač [19]

Zobrazovací extrémní fundus kamery jsou charakterizovány tzv. FOV, neboli optickým úhlem, pod kterým je čočka schopna snímat sítnici. Tzn., jakou část sítnice promítne čočka na detektor [20]. Standardní úhel FOV je 30° a čočka tak vytváří obraz 2.5krát zvětšený. Širokoúhlé objektivy mají úhel FOV od 45° až po 140° a jejich zvětšení je úměrně nižší. Úzkopásmové objektivy mají úhel FOV menší než 20° a mají zvětšení úměrně vyšší. Promítnutí úhlu FOV na sítnici a vliv velikosti FOV na množství zobrazené sítnice vidíme na Obr. 2.



Obr. 2: a) Promítnutí FOV na sítnici; vliv FOV na množství zobrazené sítnice a) 20° , b) 40° , c) 60° [21]

V dnešní době se již téměř bezvýhradně vyvíjí digitální fundus kamery, jejíž výhody jsou zejména okamžitá znalost kvality obrazu, snížení nákladů na vyšetření způsobené odstraněním nutnosti používat film, urychlení času potřebného pro měření i diagnostiku, či možnost vytváření diagnostické databáze [22]. Používání digitálních fundus kamer hraje významnou roli ve včasné diagnostice patologií oka i srdečního systému [23]. Více k těmto onemocněním i jejich příznakům lze nalézt v kapitole 2.4.

Pro získání informace o 3D struktuře sítnice je možné ji zobrazit pomocí stereofundus kamer, které využívají pro každé oko oddělenou optiku a vytváří tak pseudo 3D obraz. Tyto kamery využívají faktu, že každé oko pozoruje přes optiku daný preparát z jiného úhlu a mozek pak z těchto obrazů skládá výsledný snímek, ve kterém můžeme vnímat hloubky. Stereofundus kamery využívají stejně jako klasické fundus kamery odraženého, nikoli prošlého světla. Více informací o principech stereomikroskopie můžeme nalézt v [24]. V souvislosti s fundus stereomikroskopií můžeme zmínit práci Tsuru a kol. [25], kteří se již v roce 1996 zabývali principy a návrhem digitální stereofundus kamery.

Snímky, jejichž deriváty jsou v této práci dostupné jako sady výsledků z daných metod, byly pořízeny pomocí digitální fundus kamery Canon CF-60 UVi s vestavěným fotoaparátem Canon EOS-20D s nastaveným FOV 30-60°. Tuto kameru můžeme vidět na Obr. 3. Canon CF-60 UVi je vybavena přesnou měřicí optikou, nastavitelnou pracovní vzdáleností, 8.2 Mpix fotoaparátem s CMOS snímačem a automatickou expozicí na 35 mm barevné fotografie. Více technických specifikací této kamery lze nalézt v [26] nebo [27].



Obr. 3: Digitální fundus kamera Canon CF-60 UVi [26]

2.2. Princip snímání očního pozadí pomocí fundus kamery

Fundus kamery využívají, stejně jako mikroskopy nebo optická zrcátka faktu, že sítnice oka je jediné místo na těle, kde lze neinvazivně pozorovat cévní řečiště. Fundus fotografie zobrazují sítnici, neurosenzitivní vrstvu oka, která převádí viditelný obraz na elektrické impulzy. Tyto elektrické impulzy je posléze mozek schopn patřičně zpracovat.

Pacient při vyšetření sedí, bradu má opřenou ve speciální kolébce a čelo má opřené o čelní panel, oči má otevřené a hledí přímo před sebe. V místnosti je ztlumeno osvětlení. Optika fundus kamery je vystředěna a zaostřena na optický disk. Některé fundus kamery obsahují systémy, které provedou vystředění a zaostření kamery automaticky. Následuje osvětlení sítnice zdrojem bílého světla a sejmutí obrazu kamerou. Automatické stření kamery může být prováděno např. využitím zaměřovacího kříže [21]. Více o principech zaostřování fundus kamery je uvedeno v [23], vliv rozostřených fotografií na diagnostiku je diskutován v práci [28]. Celá optická soustava fundus kamery využívá soustavy čoček a zrcadel a její popis převyšuje rozsah této práce. Více informací o optické soustavě fundus kamer lze však nalézt pracích [21], případně [23].

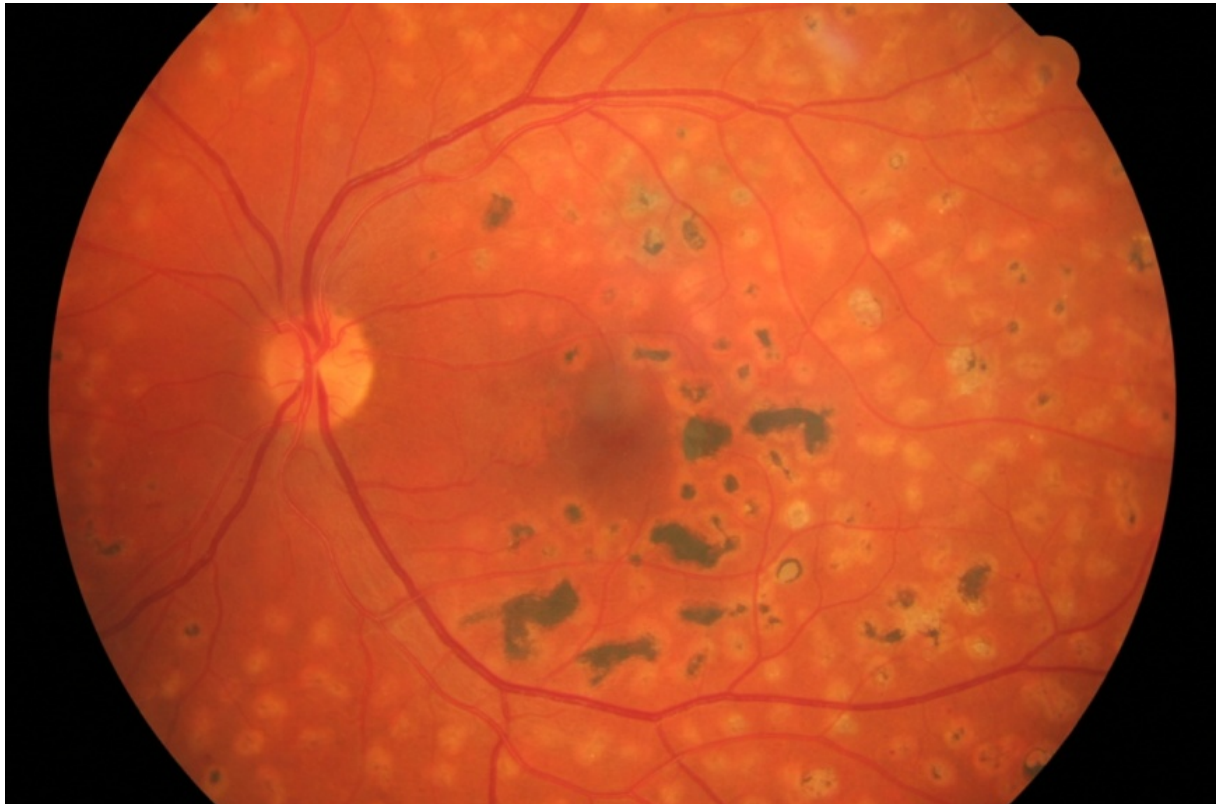
Při snímání fundus fotografií mohou být také použity barevné filtry, které přímo odstraní požadovanou část spektra (tohoto se využívá zejména při angiografickém režimu snímání [22]), nebo speciální barviva včetně fluorescenčních a indocianínové zeleně. Filtrování spektra se využívá zejména při fotografování nervových vláken sítnice.

Fundus kamera může pracovat v několika režimech snímání obrazu [22]:

1. **Barevný snímek** – sítnice je osvětlena bílým světlem a obraz je reprezentován barevnou fotografií, jako můžeme vidět např. na Obr. 4.
2. **GB režim** (v originále *Red-free* režim) – v tomto režimu je použito filtrů k odstranění červené (*red*) složky spektra. Tohoto režimu se využívá ke zvýraznění kontrastu cévního řečiště a dalších struktur sítnice. Více o těchto strukturách v kapitole 2.3.
3. **Angiografie** – při tomto režimu je kontrast cévního řečiště zvýšen použitím fluorescenčních barviv. Odfiltrováním části spektra generujícího excitaci, lze získat velmi kontrastní snímek cévního řečiště. V tomto režimu lze také snímat sekvenci snímků a získat tak informaci o časové distribuci barviva do cévního řečiště. Jedná se o velmi specifický mód. Jedním z velmi často používaných kontrastních látek jsou sodíková fluorescenční barviva.

2.3. Vlastnosti obrazových dat z fundus kamery a struktury reprezentující cévní řečiště

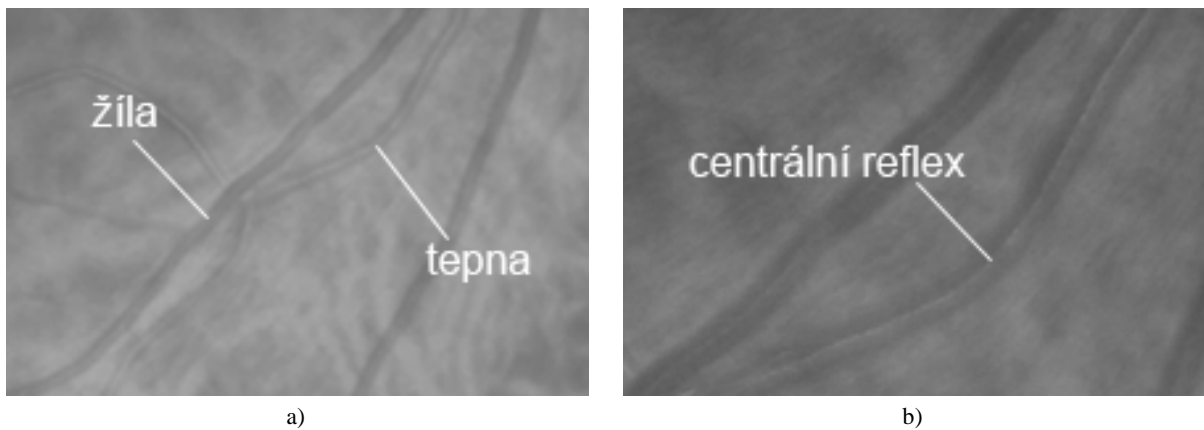
Snímky z digitální fundus kamery jsou nejčastěji uloženy v barevném kódování RGB, v případě použití speciálních režimů snímání (viz předchozí kapitola) mohou být také specificky barevné, závislé od daného režimu. Příklad snímku levého oka postiženého diabetickou retinopatií uloženého v RGB barevném kódování je uveden na Obr. 4.



Obr. 4: Barevný obrázek oka s diabetickou retinopatií z digitální fundus kamery

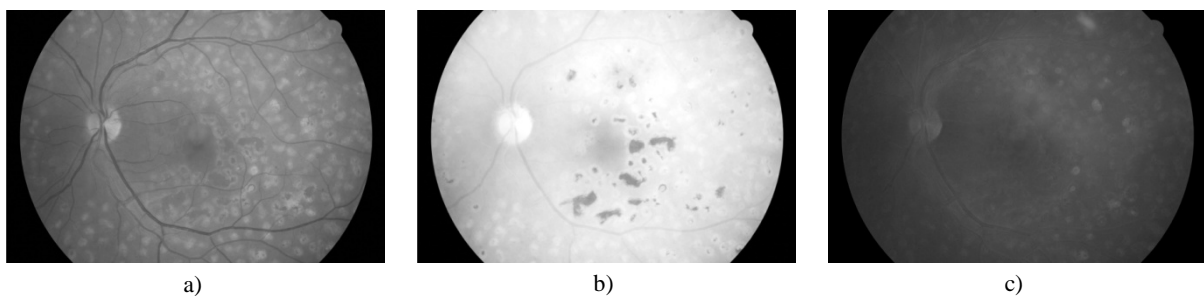
Charakteristické červené zabarvení fundus snímků je způsobeno bohatě prokrvenou sítnicí (retinou). Dále na snímcích můžeme nalézt zásobovací cévy tvořící cévní řečiště, ty se dělí na tepny a žíly. Rozdíly mezi nimi jsou patrné z obrázku Obr. 5 a). U širších cév můžeme pozorovat také centrální reflex, viz Obr. 5 b). Cévní řečiště se sbíhá v místě nejsvětlejšího místa snímku, tzv. optického disku, neboli terče zrakového nervu (papilla). Optický disk je patrný na Obr. 4.

Nejlepšího kontrastu mezi pozadím a cévami je dosaženo v zeleném kanálu [11], jak také vidíme na Obr. 6. Proto řada segmentačních algoritmů, diskutovaných v kapitole 3, využívá pro další zpracování právě zelené složky fundus snímků. Na snímcích je také patrná žlutá skvrna (fovea). Je to místo neostřejšího vidění, neboť se zde sbíhají nervová vlákna a



Obr. 5: a) zobrazení tepen a žil na fundus snímku; b) zobrazení centrálního reflexu

opouští zde oko [29]. Nervová vlákna jsou charakteristická svým světlým žháním a v RGB barevném režimu příliš viditelná nejsou, nejlépe je můžeme pozorovat v okolí zrakového terče. Při větším přiblížení jsou nervová vlákna viditelná na Obr. 6 c). Pro jejich zvýraznění se využívají charakteristické barevné filtry.

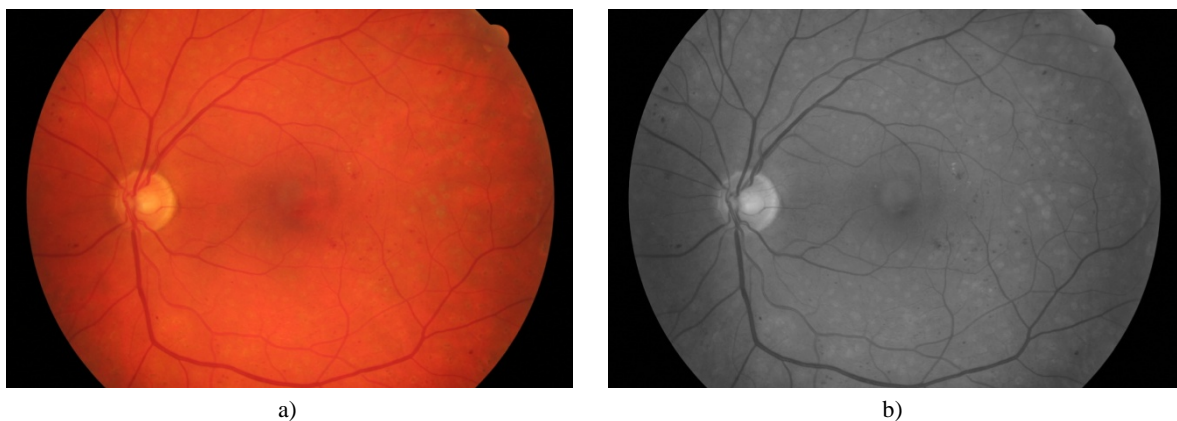


Obr. 6: Diabetický snímek v jednotlivých obrazových kanálech: a) zelený; b) červený; c) modrý

Na patologických snímcích, jako je např. Obr. 4, můžeme pozorovat charakteristické změny červeného zbarvení sítnice, případně tmavé skvrny způsobené v tomto případě důsledkem diabetické retinopatie. U patologických snímků můžeme rovněž pozorovat změny tloušťek cévního řečiště, opět způsobené diabetickou retinopatií, dále může být cévní řečiště rozvětveno do mnoha nitkovitých cévek [29]. Využití přítomnosti těchto patologií ve snímcích k diagnostice onemocnění je popsáno v následující kapitole.

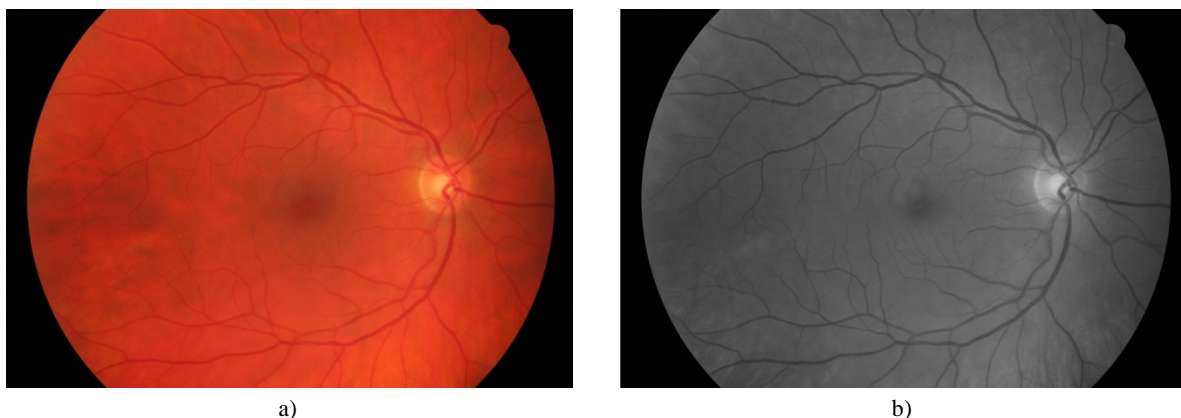
2.4. Využití charakteristických struktur ve fundus snímcích k diagnostice patologií

Fundus fotografie reflektují stav cévního řečiště. Automatickou analýzou pozice a tvaru optického disku se zabývá řada prací, můžeme zmínit např. [6], [7], [30]. Detekcí a analýzou nervových vláken se zabývají práce [8] a [31]. Dále je fundus fotografií využíváno k diagnostice řady onemocnění. Například při stanovení diabetické retinopatie, jejímž vlivem je cévní řečiště a zrakový nerv poškozen zvýšeným krevním tlakem. Z tohoto důvodu se u diabetických pacientů pravidelně provádí preventivní screeningové vyšetření k vyloučení tohoto degenerativního poškození, jelikož ve chvíli, kdy pacient pozoruje zhoršené vidění, bývá již často poškození nevratné, případně vratné s trvalými následky. Diagnostikou tohoto onemocnění se zabývají Sinthanayothin a kol. [4], případně Jomier a kol. [32]. Snímek levého oka pacienta s diabetickou retinopatií vidíme na Obr. 7. V jeho části b) je zobrazený zelený obrazový kanál. Vidíme, že cévní řečiště je zde lépe odlišitelné od pozadí. Potvrzují se tak teoretické předpoklady uvedené v kapitole 2.3.



Obr. 7: Fundus snímek levého oka pacienta s diabetickou retinopatií a) RGB prostor ; b) zelený obrazový kanál

Dalším z onemocnění, které můžeme diagnostikovat pomocí fundus kamery, je zelený zákal neboli glaukom. Glaukom je ve většině případů charakterizován zvýšeným nitroočním tlakem a následnou progresivní degenerací zrakového nervu, jejímž vlivem dochází k výpadkům zorného pole. Tyto jevy jsou způsobeny zvýšeným nitroočním tlakem. V průběhu onemocnění se vlivem degenerace snižuje tloušťka nervové vrstvy sítnice a dochází ke ztrátě spojení mezi fotoreceptory a mozkovou kůrou. Glaukom je v současné době jednou z nejčastějších celosvětových příčin slepoty. U všech věkových skupin má prevalenci více než 2,4 % a u skupiny starší 75 let dokonce 4,7 % [33]. Automatickou detekcí glaukomu z fundus snímků se zabývá např. Bock a kol. [1]. Na Obr. 8 vidíme snímek pravého oka pacienta s glaukomem.



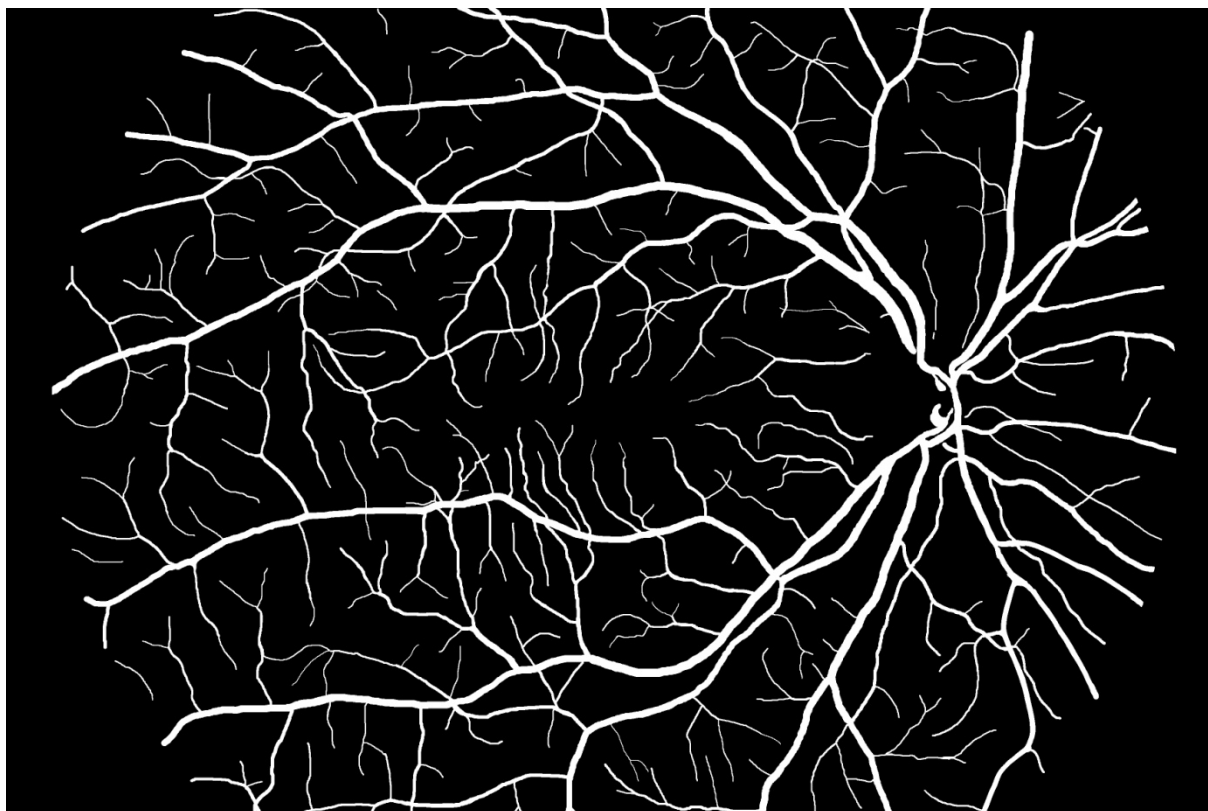
Obr. 8: Fundus snímek pravého oka pacienta s glaukomem a) RGB prostor ; b) zelený obrazový kanál

Z fundus fotografií lze diagnostikovat také přítomnost věkem podmíněné makulární degenerace (AMD), což je onemocnění postihující centrální zrakovou ostrost. Základními příznaky tohoto onemocnění jsou pozorovatelné deformace obrazu, výpadky zorného pole a pokles centrální zrakové ostrosti [2]. AMD se dělí na vlhkou (výpotkovou) a suchou (bezvýpotkovou). V západním světě postihuje vlhká forma AMD až 35 % populace starší 50-ti let, právě vlhká forma AMD je vůbec nejčastější příčina ztráty zraku u populační skupiny 50 a více let [29]. Výpotkové formě AMD, která postihuje místo nejostřejšího vidění - žlutou skvrnu, je v posledních letech věnována značná pozornost a v současné době ji lze poměrně úspěšně léčit. Existují i případy léčby, kdy se podařilo stav již rozvinuté AMD zvrátit a pacientovi zcela vrátit zrakovou ostrost. Více informací o AMD i způsobech léčby lze nalézt v práci [2]. Detekcí makulární degenerace a využitím pro diagnostické účely se zabývají Rapantzikos a kol. [3]. Metodu automatické detekce makulární degenerace publikovali také Köse a kol. [34].

3. Segmentační techniky

3.1. Obecný úvod

Segmentace obrazu je postup digitálního zpracování obrazu, při němž se snažíme rozčlenit obraz do skupin se společnými vlastnostmi, majícími obvykle nějaký společný, charakteristický význam. Jinými slovy, cílem segmentace obrazu je zachycení významného prvku sejmuté scény [35]. V našem případě se jedná o detekci cévního řečiště v retinálním snímku a oddělení jej od pozadí. Segmentace vychází z globální znalosti obrazu a skládá se z několika bodů. Jejich posloupnost záleží na zvoleném přístupu [36]. Obvykle se jedná o předzpracování obrazu s následným aplikováním daného prahovacího algoritmu, poté dochází k dalšímu zpracování a následně ke zobrazení výsledků. Výstupem segmentace je binární obraz, kde barevná reprezentace segmentované složky a pozadí závisí na zvolené aplikaci [37]; např. binárními 1 jsou reprezentovány cévy, binárními 0 pozadí. Jan [38] se ve své publikaci velmi podrobně zabývá úvodem do technik automatického zpracování obrazů. V knize jsou rovněž popsány možnosti využití daných konceptů při zpracování medicínských obrazů. Tato kniha společně s publikací [13] čtenáře podrobně zavede do problematiky automatického zpracování, nejen medicínských, obrazů.



Obr. 9: Příklad zlatého standardu levého zdravého oka

Jak již bylo řečeno dříve, je segmentace jedním z kritických bodů automatického zpracování obrazu a je proto nutné věnovat značnou pozornost volbě použité metody. Stěžejní částí segmentace je prahování [39]. Při prahování dochází nejprve k nalezení optimálního prahu a následně k rozdělení množiny bodů do vhodně zvolených skupin. Práh, případně prahy, může určit uživatel jako vstupní proměnnou. Alternativně může být automaticky vypočítán algoritmem. Tento postup je znám jako automatické prahování [40]. Ideálním prahem je takový práh, který rozdělí všechny body tak, aniž by nějaký bod připadl do nesprávné skupiny - 100% sensitivní [13]. Existuje mnoho technik prahování, můžeme zmínit například metody globálního a lokálního prahování obrazu, regionově či hranově orientované metody, případně znalostní a hybridní metody [36]. Sezgin a Sankur rozdělili metody prahování do 6 skupin. Popis těchto skupin a souhrn čtyřiceti různých prahovacích funkcí náležících do jednotlivých kategorií, je uveden v práci [37]. Alternativní přístupy dělení, jako jsou obrazově či modelově řízené metody prahování, můžeme nalézt v práci [39].

Hodnocení schopnosti dané metody segmentovat cévní řečiště se provádí porovnáním dosaženého výsledku s tzv. zlatým standardem. Jedná se o datový soubor (obraz, matice, atd.), ve kterém jsou přesně zařazeny jednotlivé elementy do správných skupin. Opět ilustrujeme na našem příkladu: při segmentaci cévního řečiště je zlatý standard snímek, ve kterém je správně označeno cévní řečiště a pozadí. Příklad zlatého standardu levého zdravého oka vidíme na Obr. 9. Vzhledem k požadavku, aby byly zlaté standardy co nejkvalitnější, jsou většinou vytvářeny ručně odborníky z daného oboru. Pro hodnocení dosažených výsledků, potažmo schopnosti jednotlivých metod segmentovat cévní řečiště, byly v této práci dostupné snímky zlatých standardů z nové databáze retinálních snímků HRF [17]. Každému výsledku, zpracovávanému v této práci, náleží odpovídající zlatý standard z databáze. Nutno podotknout, že v případě zpracování dat z metody 2 a případné kombinaci výsledků obou metod, náleží jeden zlatý standard ke všem příznakům. Ilustrujme na příkladu: čtyřem snímkům *01_h_seg.m.png_backsampled0.png* až *01_h_seg.m.png_backsampled3.png*, vytvořeným metodou [16] na zdravé části databáze, odpovídá jeden snímek zlatého standardu, konkrétně snímek *01-image_0246.bmp*.

Samotné hodnocení úspěšnosti segmentace cévního řečiště u dílčích výsledků je v této práci provedeno pomocí koeficientů sensitivity (SE), specificity (SP) a přesnosti (ACC). Tyto koeficienty porovnávají vytvořený binární výsledek se zlatým standardem. K tomuto porovnání je využito proměnných TP, TN, FP, FN. Proměnná TP vyjadřuje v našem případě počet bodů cévního řečiště, které jsou dle zlatého standardu očekávány a skutečně se na

daných místech ve vytvořeném snímku nacházejí. Obdobně je tomu u proměnné TN, avšak s obměnou pro pozadí. Proměnná FP vyjadřuje počet bodů, které jsou označeny jako cévní řečiště, i když náleží do pozadí. FN vyjadřuje počet bodů, které sice náleží do cévního řečiště, ale byly označeny jako pozadí. Z těchto čtyř proměnných se dle vzorce 1 pro SE, vzorce 2 pro SP a vzorce 3 pro ACC vypočítají dané statistiky [41].

$$SE = \frac{TP}{TP+FN} \quad (1)$$

$$SP = \frac{TN}{TN+FP} \quad (2)$$

$$ACC = \frac{TP+TN}{TP+FN+TN+FP} \quad (3)$$

Čím vyšší dosahuje metoda sensitivity, tím vyšší je její schopnost správně určit body náležící cévnímu řečišti. Obdobně je tomu u specificity s pozadím. Přesnost poté udává chybovost dané metody. Všechny tři tyto hodnoty požadujeme u kvalitní metody co nejvyšší.

V následující kapitole jsou popsány práce, využívající prahování jako finální části segmentace cévního řečiště z retinálních snímků. Práce jsou z hlediska prahování rozděleny na metody využívající globálního prahování, lokálního prahování a metody využívající klasifikace obrazu. Právě posledně zmíněná segmentace obrazu pomocí klasifikátorů je ve čtvrté kapitole implementována do Matlabu s použitím klasifikačního softwaru Weka.

3.2. Metody segmentace struktur reprezentujících cévní řečiště

Prvním z popisovaných přístupů k segmentaci cévního řečiště z fundus snímků jsou metody založené na globálním prahování obrazu. Tyto metody vypočtou pro celý obraz jedno globální kritérium, dle kterého následně celý obraz rozdělí do dvou kategorií - cévy a pozadí. Metody globálního prahování ovšem nejsou při výpočtu prahovacího kritéria schopny zohledňovat lokální vlastnosti obrazu. Byla publikována celá řada metod výpočtu globálního prahu, např. výpočet pomocí konvexní obálky histogramu nebo pomocí shlukového prahování. Dále Otsu nebo Kittlerova shlukovací funkce, případně Yenova či Kapurova entropická funkce [37]. Metody globálního prahování jsou hojně používané pro svou relativní jednoduchost [37] a při vhodně zvoleném výpočtu prahu, mohou dosahovat poměrně uspokojivých přesností, viz následující odstavce.

Al-Rawi a kol. [9] použili metodu přizpůsobené filtrace a upravili původní parametry pro přizpůsobené filtry. Tyto parametry byly odvozeny ze snímků databáze DRIVE. Al-Rawi a kol. dosáhli při použití globálního prahu na databázi DRIVE přesnosti 95,32 %, přičemž optimální hodnota globálního prahu byla hledána v rozmezí 0-1 s krokem 0,05.

Cinsdikici a Aydin [42] publikovali hybridní metodu segmentace cévního řečiště. Použili metodu přizpůsobených filtrů a rozšířili ji o optimalizaci pomocí mravenčí kolonie, která využívá tzv. feromonové stopy. Jako globální práh je volena aktuální hodnota feromonové stopy. Pro jednotlivé mravence procházející obrazem se mění úroveň feromonové stopy od 10^{-10} až po 100 s krokem 0,1 a sleduje se dosažená přesnost na daném snímku. Pro výsledné prahování využívají globálního prahu, vypočteného jako průměr z prahů získaných na 20 snímcích databáze DRIVE. S tímto algoritmem dosáhli na databázi DRIVE přesnosti 92,93 %.

Poli a Valli [43] publikovali metodu segmentace cévního řečiště založenou na filtraci sadou lineárních filtrů, zohledňujících různou orientaci i tloušťku cév. Výstupem filtrace je sada několika snímků s možnými tloušťkami a orientacemi cév, tyto snímky jsou následně sečteny (integrovány) a výsledný snímek je poté zpracován pomocí hysterezního prahování.

Zana a Klein [44] založili svou metodu segmentace vzorů náležících k cévnímu řečišti na matematické morfologii a analýze zakřivení. Jedná se o poměrně rozsáhlou metodu, vytvářející pro cévní řečiště modely a využívající Laplaceovy a Gaussovy transformace. Navrženými filtry je opět zpracován obraz a pomocí globálního, algoritmem vypočteného prahu, jsou poté odezvy filtrů binarizovány. Ve výsledném snímku je patrná řada nežádoucích zakřivení, přerušení cév i nadbytečných struktur. Autoři srovnávají výsledky své metody

s jinými dostupnými metodami přizpůsobených filtrů a segmentací pomocí manuálně zvoleného prahu. Pro testování algoritmu byly využity snímky z retinální angiografie a barevné fundus fotografie od zdravých i nemocných pacientů.

Olumi a kol. [45] použili pro detekci cévního řečiště banku směrově citlivých Gaborových filtrů pro několik průměrů cév a dále také prodlužovacích parametrů. Tato metoda dosáhla na databázi DRIVE plochy pod ROC křivkou 0,96.

Metoda Mendoçi a Campilha [46] provádí segmentaci kombinací detektoru centrálního reflexu a morfologické rekonstrukce. Nejprve dochází k extrakci centrálních reflexů z obrazu a následně je pomocí iteračního regionového narůstání oblastí získán výsledný obraz. Při regionovém narůstání oblastí dochází k integraci několika binárních obrazů získaných z morfologických filtrů. Typ filtru je závislý na tloušťce cév. Globální práh pro binarizaci odezvy filtru je získán z histogramu dané odezvy, obsahující kandidáty na přiřazení k cévnímu řečišti. Na databázi DRIVE dosáhla tato metoda přesnosti 94,63 %, na databázi STARE 94,69 %.

Dalším z popisovaných algoritmů využívajících globálního prahování je práce Ricci a Perfetti [47]. Tato metoda využívá liniového operátoru, dříve používaného v mamografii, jehož výstup je prahován globálním prahem a použit jako vstup klasifikátoru bez učení. Při použití tohoto postupu bylo dosaženo na databázi STARE přesnosti 95,84 % a 95,63 % na databázi DRIVE. Autoři dále rozvinuli výše popsanou metodu o dvouortogonální liniový detektor, jehož výstup je použit ke konstrukci příznakového vektoru pro klasifikaci s učením. Po provedení těchto modifikací dosahuje metoda na databázi STARE přesnosti 96,46 % a na DRIVE 95,45 %.

Chaudhuri a kol. [48] publikovali metodu segmentace cévního řečiště založenou na přizpůsobené filtraci. Tato metoda využívá filtraci přizpůsobenými filtry a korelaci s lokálními maskami. Pro prahování výstupů filtrů, je využita prahovací technika Otsu [49].

Nyní se dostáváme k lokálním metodám prahování. Jejich výhodou oproti globálním je fakt, že upravují hodnoty prahu pro každý pixel dle aktuálních charakteristik okolí. Tímto je dosaženo zlepšením výsledků ve srovnání s globálními metodami. Z lokálních prahovacích metod můžeme zmínit White - Roher prahovací funkci, využívající lokálního kontrastu v obraze. Případně Niblackovu metodu přizpůsobující práh dle lokálního průměru a směrodatné odchylky [37]. Více informací o lokálních prahovacích funkcích i jejich matematický zápis můžeme nalézt v [37]. Popis několika metod využívajících lokálního prahování je uveden níže.

Hoover a kol. [10] používají ve své metodě prahování snímku po částech, přičemž pro segmentaci cévního řečiště využívají lokálních i globálních vlastností obrazu. Pro každý podobraz je z pěti definovaných parametrů vypočten práh, pomocí kterého je pak daná část obrazu segmentována. Výsledkem této metody je segmentace téměř $\frac{3}{4}$ cévního řečiště a několikanásobně nižší počet FP bodů, než u dříve publikovaných metod.

Chanwimaluang a Fan [50] a Kande a kol. [51] používají k prahování lokální entropie, které velmi dobře zohledňují strukturu vaskulárních řečišť. Obě metody využívají k filtrování obrazu přizpůsobené filtry a jejich výstupy jsou poté prahovány výše zmíněnými entropiemi. Robustnost algoritmu Chanwimalunga a Fana není příliš velká, neboť u patologických snímků nedochází k detekci všech lézí. Naopak metoda Kande a kol. vykazuje na databázi STARE přesnost 94,85 % a na databázi DRIVE 94,37 %.

Metoda Jiang a Mojon [52] je založena na víceprahovém ověřovacím schématu. Prvotní zařazení do skupin je provedeno pomocí binarizace použitím hypotetického prahu. Navržené rozdělení do skupin je následně potvrzeno nebo zamítnuto pomocí ověřovací procedury. Ověřovací procedura je závislá na typu použití algoritmu – v tomto případě se jedná o znalostní metodu lokálního prahování. Tato ověřovací metoda vychází z lokálních znalostí intenzit v obraze a vytváří několik prahů pro detekci objektu v právě zpracovávané části obrazu. Provedením ověřovací procedury s použitými prahy se objekt v obraze buďto zachová, nebo odstraní z dané skupiny.

Posledním z přístupů, popisovaných v této práci, jsou metody založené na učení. Jsou to metody, u nichž je nutné algoritmu nejprve předložit snímek, na němž se naučí predikovat danou vlastnost, a poté takto vytvořený model testovat na neznámých datech. Tyto metody využívají k rozdělení snímku na významnou složku a pozadí klasifikátory. Klasifikátory jsou matematické operátory, které jsou na základě předem získaných znalostí schopny zařadit objekt do určité skupiny. Zařazení do skupiny je zatíženo určitou mírou nejistoty (pravděpodobnosti chyby) [13]. Vyžadují pro naučení předložení vektoru příznaků [11]. Tento vícerozměrný vektor, skládající se z vlastností vypočtených z trénovacího obrazu, může mít i několik desítek prvků. Jako příklad můžeme uvést práci Lupascu a kol. [14], ve které je příznakový vektor složen ze 41 prvků, jedná se např. regionové a hranicové vlastnosti v Gausově prostoru, nebo transformace pomocí Gaborových vlnek. Více o klasifikátorech a jejich učení můžeme nalézt v publikaci [13]. V posledních letech došlo v této oblasti k velkému pokroku [11]. Byla publikována celá řada prací zabývajících využitím klasifikátorů. Jejich souhrn můžeme nalézt dále v této kapitole.

Sinthanayothin a kol. [7] používají ve své práci k detekci cévního řečiště výstupy vícevrstvého perceptronu a detekci hran z první komponenty PCA. Vstupy vícevrstvého perceptronu byly odvozeny z analýzy hlavních komponent obrazu. Při detekci cévního řečiště se jim podařilo dosáhnout na dostupných datech sensitivity 83,3 % a specificity 91,0 %.

Staal a kol. [11] publikovali metodu založenou na detekci hřebenů, které úzce korespondují s centrálním reflexem a značí přítomnost cévního řečiště. Pro každý pixel je zkonstruován příznakový vektor, jehož součástí jsou například liniové elementy. Tyto příznakové vektory jsou poté předloženy kNN klasifikátoru a snímek je poté rozřazen do skupin pozadí a cévního řečiště. Přesnost této metody je na 40 testovaných snímcích 94,4 % pro první sadu zlatých standardů a 94,7 % pro druhou sadu. Výsledky jejich metody dosahují na stejných sadách snímků lepších výsledků, než dříve publikované metody Hoover a kol. [10] a Jiang a Mojon [52].

Conduchare a kol. publikovali v roce 2005 [53], a v roce 2010 aktualizovali [54], metodu klasifikace obrazu založenou na hysterezi. Tato metoda přistupuje k dané problematice, jako ke klasifikování pomocí vzorů. K segmentaci využívá percentilového klasifikátoru. Navržený klasifikátor využívá specifických doménových znalostí k odstranění problematických jevů během klasifikace. Můžeme zmínit například překrytí spekter cév a pozadí. Na databázi, kterou vytvořil Hoover a kol. [10], bylo touto metodou v roce 2005 dosaženo přesnosti 89,34%, respektive 95,56 % v roce 2010.

Metoda Soarese a kol. [55] má příznakový vektor sestavený z intenzit obrazových bodů a odezvy spojených 2D Morletových vlnek v různých směrech. Pro klasifikaci je využito Bayesova klasifikátoru s třídně závislou pravděpodobnostní distribuční funkcí. Na databázi DRIVE bylo dosaženo přesnosti 94,67 %, na databázi STARE 94, 74 %.

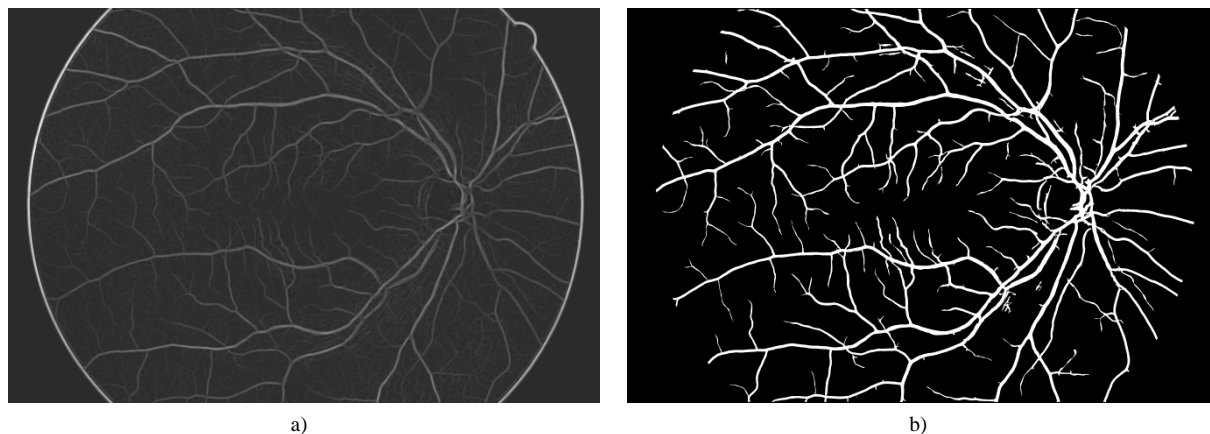
Marín a kol. [56] publikovali v roce 2010 metodu využívající šedotónový obraz a vlastnosti momentových invariantností. Jako vstup klasifikátoru je zkonstruován 7-mi dimenzionální příznakový vektor sestavený právě z šedotónové reprezentace obrazu a zmíněných invariantností. Obraz je následně klasifikován pomocí neuronové sítě s využitím daného příznakového vektoru. Tento algoritmus dosahuje na databázi DRIVE přesnosti 94,52 % a na databázi STARE 95,26 %.

Lupascu a kol. [14] vytvořili pro vstup AdaBoost klasifikátoru 41-rozměrný vektor, složený z prvků popsaných v úvodu této kapitoly. Klasifikátor byl naučen na 789 914 snímcích zlatých standardů a testován na databázi DRIVE, na které dosáhl přesnosti klasifikace 95,97 %. Tyto výsledky jej řadí mezi nejpřesnější segmentační algoritmy vůbec.

3.3. Detailní popis zvolených metod segmentace cévních struktur

V této práci jsou dostupné výstupy z metod [15] a [16], proto jsou zde detailněji popisovány právě tyto metody. Obě metody byly mimo jiné testovány na databázi HRF a pro získání binárních výsledků využívají globálního prahování. Výsledky z obou metod byly zpracovány jednak samostatně a jednak jako kombinace příznaků. Postupy, jakými byla data zpracována a testována jsou popsány v následujících kapitolách.

Odstrčilík a kol. [15] vytvořili metodu založenou na přizpůsobených filtrech. Pro nalezení cévního řečiště je zde využito filtrace 2D přizpůsobenými filtry a korelace s 2D lokálními maskami, vytvořenými v několika směrech. K získání binárních snímků je využita prahovací technika Otsu [49]. Metoda byla vytvořena a testována na snímcích s vysokým rozlišením z databáze HRF i na veřejně dostupné obrazové databázi DRIVE. Na databázi DRIVE bylo dosaženo sensitivity 70,60 %, specificity 96,93 % a přesnosti 93,4%. Na databázi HRF poté sensitivity 78,61%, specificity 97,50 % a přesnosti 95,39 %. Vytvořená metoda poměrně spolehlivě detekuje zásobovací cévy s širším profilem a také optický disk. Naproti tomu cévy s užším profilem nejsou detekovány příliš spolehlivě. Na Obr. 10 a) je uveden výstup z přizpůsobených filtrů. Na snímku b) je poté daný výstup zpracovaný pomocí globálního prahu a upravený pomocí zmenšení masky pro odstranění výrazného bílého ohraničení kolem celého FOV.



Obr. 10: Výstupy z metody 1: a) přizpůsobené filtry; b) binární reprezentace snímku a)

Budai a kol. [16] ve své metodě nejprve vytváří Gausovu pyramidu složenou ze tří úrovní. Každá úroveň obsahuje černobílý snímek o poloviční šířce a výšce předchozí úrovně. Tímto jsou získány pravděpodobnosti příslušnosti pixelu k cévním strukturám. Následně je provedena analýza sousedů pomocí vlastních hodnot Hessianovy matice druhých derivací. Toto je provedeno na každé vrstvě Gaussovy pyramidy. Poté jsou všechny úrovně Gaussovy pyramidy převzorkovány na původní rozlišení a prahovány pomocí hysterezního prahování

pomocí dvou prahů. Výsledný obraz je vytvořen sloučením jednotlivých binárních obrazů pomocí *pixel-wise OR* operátoru. Tento postup generuje ve výsledném binárním obrazu velmi dobře tenké cévy i jejich jemná zakončení. Metoda Budai a kol. [16] dosáhla na veřejných databázích přesnosti 93,8 % (STARE), respektive 94,9 % (DRIVE). Výstupy jednotlivých vrstev Gaussovy pyramidy před prahováním a následným sečtením jsou uvedeny na Obr. 17.

4. Implementace zvoleného přístupu prahování snímků

4.1. Popis klasifikátorů

V rámci předešlého semestrálního projektu [57] byly v důsledku paměťových omezení testovány jednotlivé klasifikátory pouze na výřezu ze snímku 06-image_0888.jpg. Výsledky jednotlivých klasifikátorů jsou uvedeny v Tab. 1. Na základě těchto výsledků byly zvoleny čtyři klasifikátory s nejlepšími výsledky (v Tab. 1 podbarveny modře), které byly v úvodu navazující diplomové práce otestovány na celém tomto snímku. Výsledky těchto klasifikátorů jsou uvedeny v Tab. 2. Jak je patrné z této tabulky, původní klasifikátory byly doplněny ještě o jeden klasifikátor - Naive Bayes. Důvod, proč byl doplněn tento klasifikátor, je popsán v kapitole 5.

Tab. 1: Úspěšnosti jednotlivých klasifikátorů na výřezu ze snímku 06-image_0888.jpg

Typ klasifikátoru	Sensitivita[%]	Přesnost klasifikace [%]
Bayesova logistická regrese [58]	96,10	96,00
Bayesova síť [59]	95,70	95,90
DMNBtext [60]	85,04	72,90
Naive Bayes [61]	95,50	95,80
Logistická funkce [62]	96,00	96,00
Vícevrstvý perceptron [59]	95,60	95,80
RBF síť [59]	91,70	94,00
Lineární logistická regrese [63]	96,00	96,00
Voted Perceptron [64]	94,60	95,20
AdaBoost M1 [64]	92,10	94,20
Atributově selektivní klasifikátor [65]	95,70	95,90
LogitBoost [66]	94,80	95,40
MultiBoost AB [67]	93,40	94,70
Prahovací selektor [59]	94,20	94,20
Konjunktivní rozh. Pravidlo [59]	92,10	94,20
ZeroR pravidlo [68]	85,40	72,90
AD rozh. strom [69]	96,10	96,10
J48 [70]	95,70	95,90
Random Forest [71]	93,10	93,70

V následujících odstavcích budou stručně popsány čtyři zvolené klasifikátory tak, jak jsou uvedeny v Tab. 2 odshora. Pátý, dodatečně přidáný, klasifikátor je popsán na konci této kapitoly.

Prvním z popisovaných je klasifikátor využívající Bayesovy logistické regrese. Tento model využívá matematické statistiky k odhadu pravděpodobnosti jistého jevu (příslušnosti prvku do dané třídy). Tato pravděpodobnost je dána na základě určitých známých skutečností (nezávisle proměnných). Řešením daného modelu dochází k hledání takové kombinace nezávisle proměnných, která co nejlépe reprezentuje míru pravděpodobnosti daného jevu (závislou proměnnou). Pravděpodobnost výskytu zkoumaného jevu je modelována pomocí náhodné veličiny nabývající hodnot 0 (jev nenastal) a 1 (jev nastal) [58]. Model předpokládá, že za podmínek určených vektorem ρ bude s pravděpodobností, jejíž závislost na ρ lze vyjádřit rovnicí 4, náhodná veličina $Y(\rho)$ rovna jedné. V rovnici 4 je vyjádřen člen logistické funkce, který z celého modelu vytváří logistický regresní klasifikátor [72].

$$\psi(\rho) = \frac{\exp(\beta_0)}{1 + \exp(\beta_0)} \quad (4)$$

kde vektor β je vektorem neznámých parametrů, ψ je výstupní proměnná a ρ je vektor vstupních parametrů. Celý model je poté reprezentován rovnicí 5, jejímž výstupem je vektor příslušností jednotlivých vzorků vstupního vektoru dodaných tříd [58].

$$p[y = +1 | \beta, \rho_0] = \psi(\beta^T \rho_0) = \psi(\sum_j \beta_j, \rho_{i,j}) \quad (5)$$

kde vektor β je vektorem neznámých parametrů, ψ je vstupní proměnná, ρ je vektor vstupních parametrů, i a j jsou indexy, p je odhadovaná pravděpodobnost. Odhadem tohoto vektoru dochází ke stanovení pravděpodobnosti výskytu hledaného jevu.

Druhý z popisovaných klasifikátorů využívá pro klasifikaci mnohočlenného logistického regresního modelu s odhadem průběhu logistické funkce [68]. V tomto případě platí rovněž postup popsany u předchozího klasifikátoru. Ve stručnosti dochází k hledání ideální kombinace vstupních parametrů (nezávislých proměnných) tak, aby závislá proměnná co nejlépe reprezentovala míru pravděpodobnosti, že nastane hledaný jev. Model lze matematicky vyjádřit rovnicí 6, případně ekvivalentní rovnicí 7 [62].

$$p(X_i) = \frac{\exp(\sum_{j=1}^n \beta_j X_{ij})}{1 + \exp(\exp(\sum_{j=1}^n \beta_j X_{ij}))} \quad (6)$$

$$\text{logit}\{p(X_i)\} = \sum_{j=1}^n \beta_j X_{ij} \quad (7)$$

kde vektor β je vektorem neznámých parametrů, i a j jsou indexy, p je odhadovaná pravděpodobnost.

Třetí z použitých klasifikátorů využívá pro klasifikaci lineárního logistického regresního modelu. I v tomto případě platí dříve popsané postupy pro hledání ideální

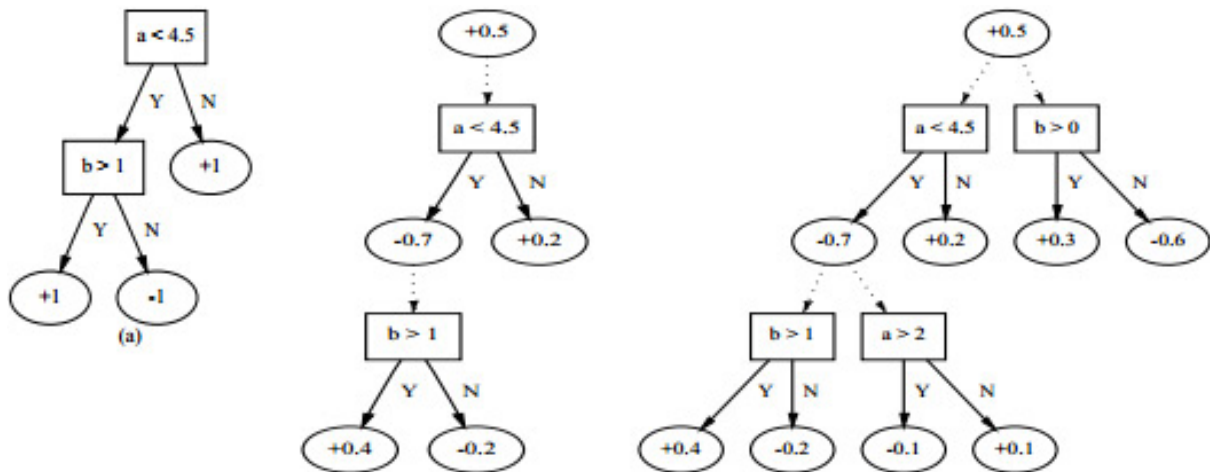
kombinace vstupních parametrů. Pro vytvoření klasifikátoru je použito LogitBoost algoritmu, který se řadí do kategorie strojového učení, v originále *machine learning*. Učení LogitBoost klasifikátoru je provedeno za pomoci logistické regresní funkce, viz rovnice 8 [72]. Výstup z klasifikátoru lze poté vyjádřit rovnicí 9 [63].

$$\alpha(\rho) = \frac{\exp(\beta_0)}{1 + \exp(\beta_0)} \quad (8)$$

$$H(x) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(x)) \quad (9)$$

kde vektor β je vektorem neznámých parametrů, t je index, α je logistická proměnná dosazovaná do vztahu, H je výstupní vektor.

Čtvrtý klasifikátor využívá střídavého rozhodovacího stromu. V prostředí WEKA je tento klasifikátor založen na metodě Freuda a Maysona [69], kteří ve své práci vytvořili nový, zjednodušený, typ rozhodovacího stromu. Tento strom je založený na posilování vah (původní název *boosting*). Blokové schéma algoritmu a jeho srovnání s klasickým rozhodovacím stromem můžeme vidět na Obr. 11. Celý algoritmus střídavého rozhodovacího stromu je detailně popsán v práci [69]. V současné době je tohoto u klasifikátoru v prostředí WEKA podporována pouze varianta predikované třídy se dvěma skupinami (např. 0 a 1). Dále je nutné upravit počet zesilovacích iterací tak, aby vyhovovali dané datové sadě a zároveň aby klasifikátor dosahoval dostatečně kvalitních výsledků. Více o těchto nastaveních lze najít v [68].



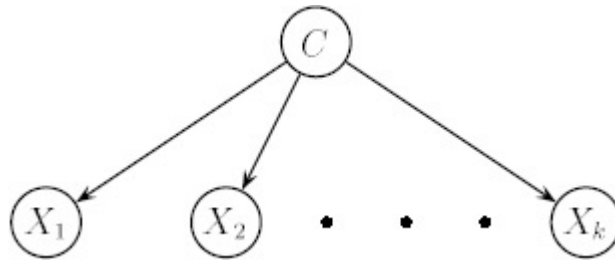
Obr. 11: Klasifikátory založené na rozhodovacích stromech: a) klasický rozhodovací strom; b) stejný rozhodovací strom interpretovaný jako střídavý strom; c) obecný střídavý strom [69]

Posledním, dodatečně přidaným, je klasifikátor ze skupiny Bayesovských funkcí - Naive Bayes. Tento klasifikátor využívá odhadu třídy. Algoritmus pro odhad výstupní třídy je v tomto případě odvozen z analýzy trénovacích dat. Celý tento klasifikátor vychází v programovém prostředí WEKA z práce [61], kde byl Naive Bayes klasifikátor použit na

spojitá data, předzpracovaná pomocí metod pokročilé statistiky. Tímto byla vyřešena do té doby problematická analýza spojitych dat pomocí Naive Bayes klasifikátoru. Pro lepší představu o principu funkce lze Naive Bayes klasifikátor reprezentovat ve formě sítě, kde každý z predikovaných výstupních atributů (X) je odvozen z dané třídy C . Výstupní atributy X jsou na sobě navzájem nezávislé. Ukázkou klasifikátoru ve formě Baesovy sítě vidíme na Obr. 12. Bayesova klasifikátoru bylo využito zejména proto, že poskytuje ve srovnání s jinými klasifikátory jednoduchý přístup k dané problematice s dobře srozumitelným zápisem procesu učení. Rovněž je velmi dobře popsán princip jeho používání.

$$p(C = c | X = x) = \frac{p(C=c)p(X=x|C=c)}{p(X=x)} \quad (10)$$

kde C je náhodná veličina vyjadřující třídu a X je vektor náhodných proměnných vyjadřujících zkoumaný atribut.



Obr. 12: Klasifikátor Naive Bayes interpretován jako Baesova síť [61]

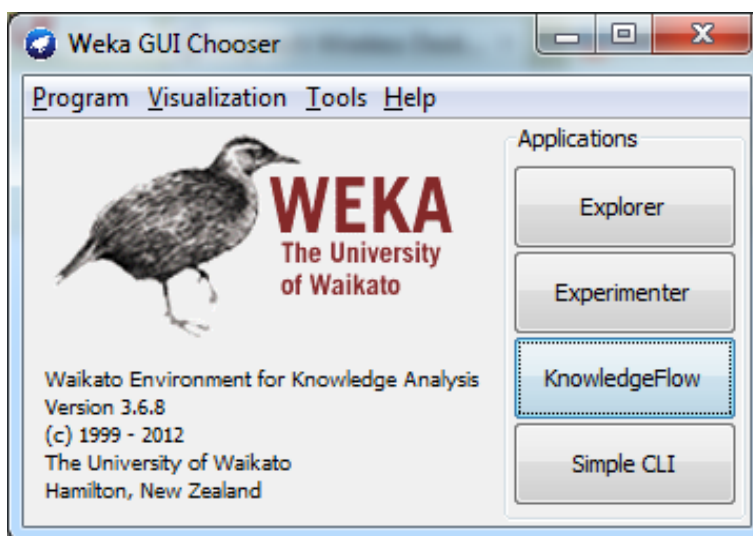
4.2. Software WEKA

V této práci byla jako implementovaná metoda zvolena klasifikace obrazu s učením. Dle teoretických předpokladů by měl tento postup dosahovat nejlepších výsledků, neboť se nejprve na zlatých standardech vytvoří klasifikační model a s jeho pomocí se následně zpracovává testovací sada snímků. Jelikož programové prostředí Matlab neumožňuje jednoduché využívání klasifikátorů (je nutné využívat speciální nástavby), byl pro klasifikaci obrazu zvolen volně dostupný *datamining* software WEKA [68]. Programové rozhraní tohoto softwaru bylo implementováno do Matlabu. Tím bylo umožněno komplexní řešení klasifikace obrazu s učením pouze za použití jednoho softwaru.

V této části jsou shrnuty hlavní vlastnosti programu WEKA. Detailní informace o používání, seznam filtrů, klasifikátorů, shlukovacích algoritmů a funkcí obsažených v programových knihovnách, jsou uvedeny v kompletním manuálu k softwaru [59].

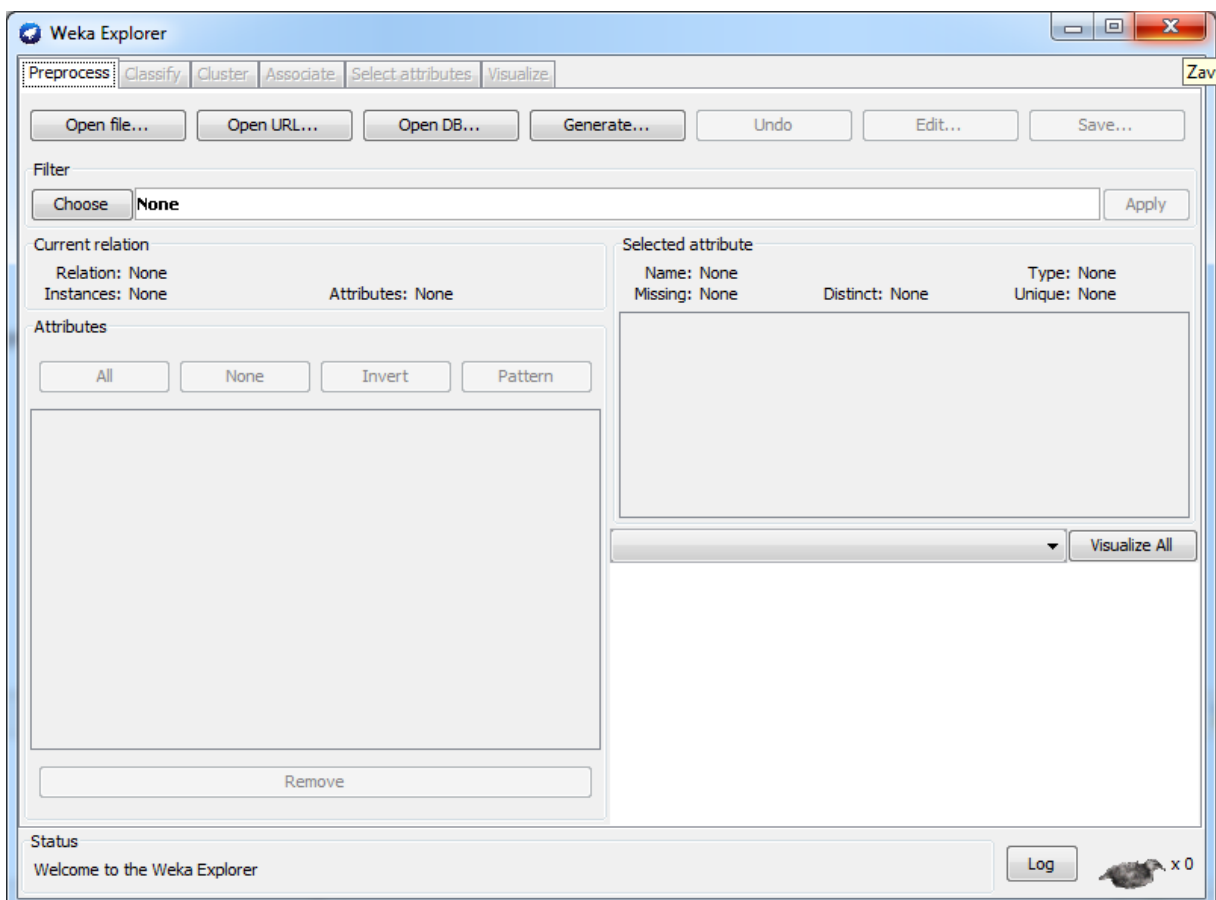
WEKA je *open source* software vyvíjený univerzitou Waikato na Novém Zélandu. Celý program běží na platformě *Java*, je tedy spustitelný na většině dnešních PC. WEKA umožňuje nejen klasifikaci vstupních dat využitím řady klasifikátorů, ale také jejich předzpracování použitím mnoha filtrů. Dále umožňuje shlukovou analýzu a vizualizaci vstupních dat i výsledků. Vše záleží na použitém rozhraní, viz dále.

Pravděpodobně v důsledku ne zcela odladěného kódu softwaru a jeho běhu na platformě *Java*, lze při práci narazit na občasné chyby či samovolné vypnutí programu. Některé chyby, se kterými jsme se při práci setkali, jsou popsány dále v této kapitole, po seznámení s pracovním prostředím.



Obr. 13: Úvodní obrazovka softwaru WEKA 3, výběr pracovního módu

Po spuštění programu lze volit ze čtyř módů, v nichž lze pracovat. Příklad úvodní obrazovky je uveden na Obr. 13. První z pracovních módů je *Explorer*. V tomto módu je dostupná většina funkcí, které WEKA obsahuje. Pracovní prostředí módu *Explorer* vidíme na Obr. 14. Dalším z pracovních módů je *Experimenter*. Tento mód slouží k pokročilému zpracování a analýze navržených modelů. Lze zpracovávat i více modelů najednou a posléze statisticky hodnotit kvalitu dílčích částí. Tento mód nebyl v této práci používán. Třetím z módů je prostředí *KnowledgeFlow*. Jedná se o grafické rozhraní, ve kterém lze model seskládat z jednotlivých bloků a ty posléze mezi sebou spojovat datovými spojnicemi. V tomto módu jsou dostupné všechny funkce, které software obsahuje. Slouží především k většímu uživatelskému přizpůsobení a umožňuje způsoby zpracování i zobrazení dat, které nejsou dostupné v režimu *Explorer*. Tento mód ovšem vyžaduje od uživatele pokročilé znalosti tohoto softwaru. Posledním z módů je *Simple CLI*. Jedná se o příkazový řádek, ze kterého lze volat všechny dostupné funkce obsažené tímto programovým prostředím. Pracovní mód *Simple CLI* vyžaduje rozsáhlou znalost funkcí softwaru WEKA i jazyka *Java*. Pro zpracování dat byl v této práci využit mód *Simple CLI*, ovládaný z programového prostředí Matlab.



Obr. 14: Pracovní mód Explorer v softwaru WEKA

Jako vstupní data umí WEKA zpracovat řadu formátů. Nejrozšířenějším formátem, který umí WEKA zpracovat, je formát *Comma Separated Value*, tedy csv. Tento formát je vhodný pro zejména pro použití *cross-validation* nebo rozdělení souboru na trénovací a testovací využitím integrovaných funkcí programu. Dále je tento formát vhodný také pro shlukovou analýzu. Doporučeným, avšak méně rozšířeným formátem pro práci, je *Attribute-Relation File Format*, neboli *ARFF*. Tento formát byl vyvinut primárně pro využití v programu WEKA. Jedná se o textový datový formát skládající se ze dvou částí. První částí je hlavička, ve které je definován seznam atributů (počet sloupců) a jejich typ. Ve druhé, datové, části jsou ve sloupcích oddělenými čárkami číselné hodnoty pro jednotlivé atributy. Příklad souboru *ARFF* vidíme na Obr. 15. Formát *ARFF* lze otevřít i v poznámkovém bloku, případně internetovém prohlížeči.

```
@relation 400x400-training-data-v1-(sorted)-weka.filters.unsupervised.attribute.NumericToBinary

@attribute 0_binarized {0,1}
@attribute 4.6697 numeric

@data
0,7.4277
0,9.7153
0,11.47
0,12.819
1,14.856
1,17.375
1,19.674
```

Obr. 15: Příklad struktury souboru arff

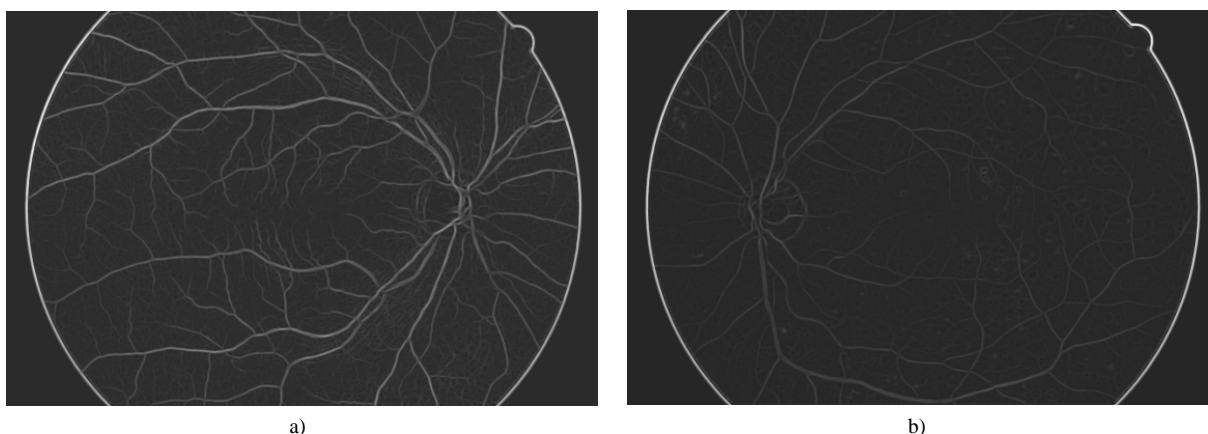
WEKA umí, jak vidíme na Obr. 15, zpracovávat nominální i numerické proměnné. Pro nominální je ovšem dostupné větší množství klasifikačních funkcí, jako jsou např. Bayesovské nebo logistické funkce, případně vícevrstvý perceptron.

Jak již bylo uvedeno, WEKA pracuje primárně se specifickým formátem dat *ARFF*. Při vytváření testovacích a trénovacích souborů předpokládá rozdělení jednoho velkého datového souboru do těchto dvou skupin. Pokud testovací soubor neobsahuje stejný počet sloupců jako trénovací, vyskytne se chyba nekompatibilních datových souborů. Dalším z neduhů, na které jsme při práci narazili, je omezení použité paměti RAM maximálně na ¼ celkové paměti. Pokud načítáme velké datové soubory (řádově miliony vzorků), software soubor nenačte a ukončí se. Nelze tedy jednoduše pracovat s velkými datovými soubory. Poslední z popsaných chyb znemožňuje původně zamýšlenou klasifikaci celých snímků pouze za použití prostředí WEKA. Soubory pro klasifikaci jsou totiž uloženy v rozlišení 3504×2336 pixelů. V každém snímku se tedy jedná o více než 8 milionů vzorků a uvážíme-li fakt, že je nutné ještě přidat příslušný příznakový vektor, dostáváme výsledný datový soubor s několika miliony vzorků. Bylo tedy nutné nalézt jinou možnost zpracování těchto dat. Řešením se ukázalo propojení

programových prostředí WEKA a Matlab a využití výpočetního CML serveru, kde lze pro tyto výpočty využít až 12 Gb paměti RAM. Postup implementace je popsán na následujících stranách.

4.3. Implementace klasifikačního modelu

V této práci byl jako hlavní programová část vytvořen klasifikační rámec implementující programové prostředí WEKA do Matlabu. Tento rámec umožňuje společně s nadřazenou ovládací funkcí předzpracování, klasifikaci i rekonstrukci obrazů pouze za použití programového prostředí Matlab. Jelikož jsou obě prostředí založena na platformě *Java*, je jejich propojení založeno na importu základní knihovny WEKA do Matlabu a následným voláním požadovaných funkcí obdobně, jako v pracovním módu *Simple CLI*. Implementace odstranila nutnost používat pracovní mód *Explorer*, jenž vyžadoval rozsáhlé ruční předzpracování vstupních dat i následnou rekonstrukci výstupních dat. Jednalo se o poměrně složitý a časově náročný postup. Propojením obou prostředí bylo dosaženo výrazného zrychlení pracovního postupu a zvýšení uživatelského komfortu. Dále implementace umožnila jednodušší využití výpočetního serveru CML a s ním spojené navýšení v Matlabu dostupné paměti RAM. Konkrétně se jedná o její část používanou pro *Java* výpočty. Na CML serveru je pro dané výpočty možno vyčlenit až na 12 Gb. Toto navýšení umožnilo zpracování snímků ve vysokém rozlišení (obsahující až několik desítek milionů vzorků).



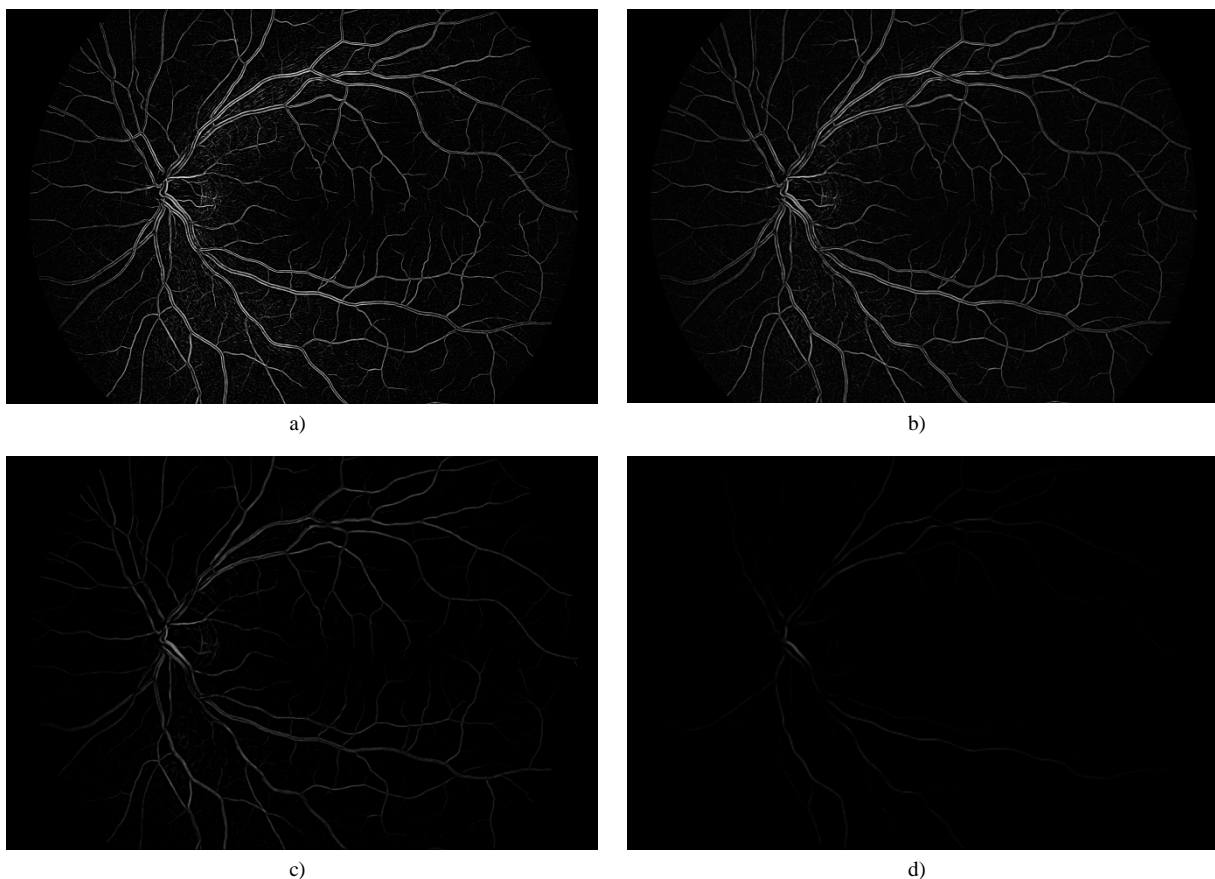
Obr. 16: Příklady výstupů MFR z metody [15]: a) snímek levého zdravého oka; b) snímek pravého oka postiženého diabetickou retinopatií

Pro testování vytvořeného klasifikačního rámec jsou dostupné dvě sady výsledků. Jednak jsou to výsledky metody [15], u níž jsou to odezvy přizpůsobených filtrů (MFR) před prahováním, uložené ve formátu *mat*. Každému snímku z databáze HRF přísluší jeden soubor *mat*. Data jsou uložena v šedotónovém barevném režimu. Ukázky dostupných výsledků z této metody jsou uvedeny na Obr. 16.

Výsledky metody [16] jsou uloženy ve formátu *png*, před prahováním a slučováním. Pro každý snímek z databáze HFR jsou dostupné čtyři (metody byla rozšířena na čtyřúrovňovou Gausovu pyramidu) šedotónové snímky. Příklad těchto snímků pro pravé oko

zdravého pacienta je uveden na Obr. 17. Vidíme zde, že nejvíce patrné jsou zásobovací cévy na prvním snímku a postupně se, se zvyšujícími se vrstvami pyramidy, vytrácí.

Testování klasifikačního rámce bylo rozděleno na tři části. V první části, byla zpracována pouze data z metody 1 společně se snímkem, obsahujícím příslušnost k cévnímu řečišti, tedy zlatým standardem. Druhou částí bylo zpracování dat z metody [16] obdobně. Jednalo se tedy o čtyři snímky odvozené z Gaussovy pyramidy (pro každý snímek z databáze jsou dostupné čtyři snímky) a odpovídající zlatý standard. Poslední, třetí částí, bylo vytvoření kombinace výsledků z obou metod. Vznikl tak 6-ti dimenzionální příznakový vektor. První příznak je snímek MFR z metody 1, druhý až pátý příznak je odvozen z metody 2 a poslední, šestý, příznak je snímek zlatého standardu. Každá z těchto částí má vytvořenou samostatnou ovládací funkci *main.m*, pojmenovanou dle principu. Ostatní funkce, včetně vytvořeného klasifikačního rámce, jsou pro všechny tři postupy společné. Dané přístupy byly následně otestovány na klasifikátoru popsaném v kapitole 4.1. a na zdravých i patologických snímcích. Dosažené výsledky jsou uvedeny a diskutovány v následující kapitole.



Obr. 17: Příklad výstupů jednotlivých vrstev Gaussovy pyramidy z metody [16] převzorkované na původní velikost: a) první vrstva; b) druhá vrstva; c) třetí vrstva; d) čtvrtá vrstva

Pro správnou implementaci klasifikačního rámce do Matlabu, je nutné splnit několik požadavků, které umožní správný běh celého skriptu. V první řadě se jedná instalaci

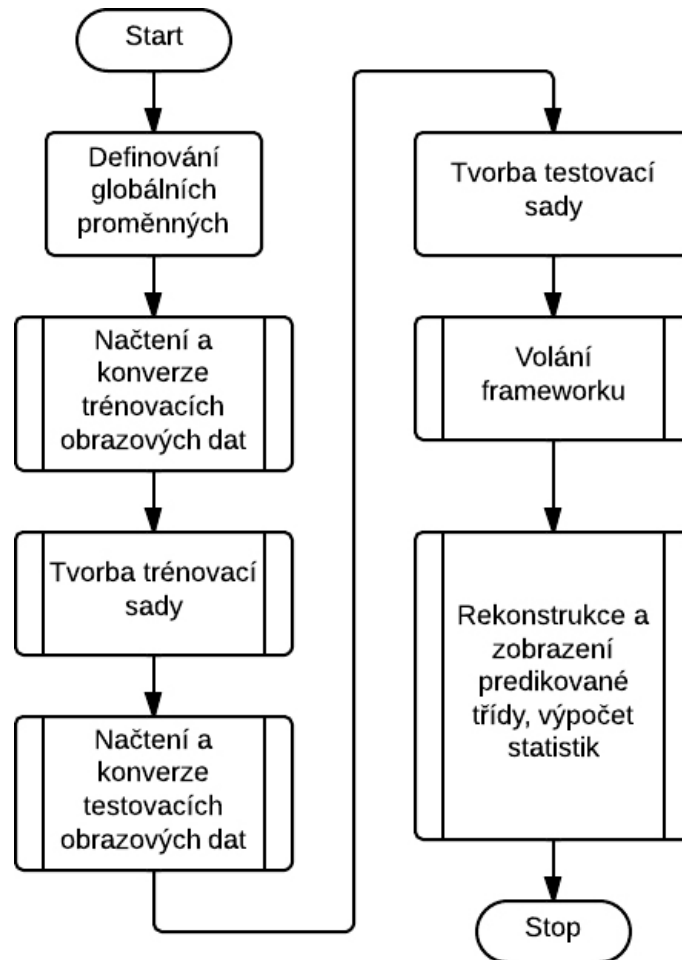
programového prostředí WEKA minimálně ve verzi 3.6.x. Dále je nutné mít v souboru knihoven Matlab umístěný odkaz na knihovnu *weka.jar*. Tato knihovna se nachází v kořenové složce prostředí WEKA. Tento odkaz je nutné vložit do textového souboru, který lze v Matlabu otevřít příkazem *edit classpath.txt*. Po uložení souboru je nutné restartovat Matlab. V případě práce s objemnými datovými soubory, je nutné v Matlabu navýšit paměť pro *Java* výpočty. Tato položka je v Matlabu od verze R2010 umístěna v menu: *Preferences » General » Java Heap Memory*. Po uložení nastavení je nutné restartovat Matlab. Při použití starších verzí Matlabu je nutné vytvořit speciální konfigurační soubor *java.opts*, umístěný v kořenové složce Matlabu. Jedná se o poměrně složité nastavení, které vyžaduje pokročilejší uživatelské znalosti operačního systému, Matlabu i administrátorská práva zápisu. Postup vytvoření konfiguračního souboru a jeho aplikaci lze nalézt v [73].

Samotná implementace klasifikačního modelu je rozdělena na dvě části. V první části byl vytvořen klasifikační rámec, jehož základní princip a funkce jsou uvedeny v kapitole 4.3.2. Detailně jsou poté popsány v příloze 2. Druhou částí bylo vytvoření nadřazené ovládací funkce *main.m*. Základní popis této funkce je uveden v kapitole 4.3.1. Detailně je tato funkce popsána v příloze 1. Každá z částí obsahuje množství dalších funkcí, nutných pro správný běh. Jejich seznam i podrobný popis je uveden v jednotlivých přílohách této práce.

4.3.1. Hlavní funkce *main.m*

Hlavní, ovládací, funkcí celé práce je funkce *main.m* (dále pojmenována dle kombinace vstupních dat). Je vytvořena ve třech provedeních, přičemž tyto se od sebe liší pouze prací se vstupními daty. Jedná se o funkce *mainJOAB.m*, *mainJO.m* a *mainAB.m*. Důvodem vytvoření těchto tří variant ovládací funkce, byl požadavek na jednoduchou možnost otestování různých kombinací vstupních dat tak, jak byly popsány dříve. Jelikož by úprava kódu pro jednotlivé kombinace byla zdlouhavá a pro uživatele nebyla nijak přínosná, byly rovnou vytvořeny tyto tři varianty. Důvodem vytvoření rámci nadřazené ovládací funkce byl fakt, že je celý rámec koncipován jako obecný klasifikační blok a vstupní data musí být předkládána v určitém formátu. A právě konverzi a následnou rekonstrukci obrazových dat, včetně hodnocení úspěšnosti segmentace cévního řečiště provádí funkce *main.m*. Ve stručnosti lze říci, že provádí automaticky to, co bylo v původní semestrální práci nutné provádět zdlouhavě ručně a za použití dílčích programových prostředí. Základní princip hlavní funkce je zobrazen v blokovém schématu na obrázku Obr. 18. Jedná se o definování globálních proměnných a adres k použitým souborům. Poté načtení obrazových dat pro trénovací sadu, konverzi dat na sloupcové vektory a tvorbu trénovací sady. Dále načtení obrazových dat pro testovací sadu,

konverzi dat na sloupcové vektory a tvorbu testovací sady. Následně je volán klasifikační rámeček a dochází k rekonstrukci predikované třídy, výpočtu statistik a zobrazení výsledků.



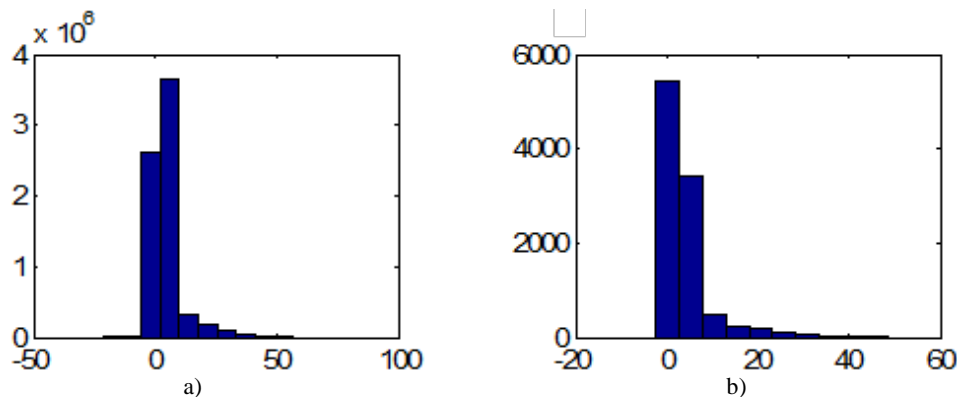
Obr. 18: Základní blokové schéma funkce mainJOAB.m

Detailní popis ovládacích funkcí, jejich vnořených funkcí i požadované vstupní formáty jsou popsány v příloze . Základní pokyny a informace jsou také vloženy jako komentáře do jednotlivých funkcí.

Jelikož se při zpracování velkých datových souborů jedná o časově velmi náročnou operaci, jsou uživatelům pro lepší představu o prováděných výpočtech zobrazeny jednotlivé hotové kroky spolu s časovými značkami.

Ve zcela specifickém případě využití klasifikačního rámce, jako bylo v této práci, tedy ke zpracování velmi velkých vstupních dat, bylo nutné omezit trénovací množinu na menší počet vzorků. Od velikosti trénovací sady se totiž odvíjí doba potřebná pro trénování klasifikátoru. Ta byla v případě původní velikosti 2×8 milionů vzorků neúměrně vysoká (cca 48 hodin pro trénování klasifikátoru). Bylo proto přistoupeno k úpravě funkce *main.m* tak, že pět snímků ze zdravé části databáze je převedeno do sloupcových vektorů, a poté jsou seskládány pod sebe. Vznikne tak jeden velmi dlouhý vektor, obsahující více než 30 milionů

vzorků (jsou totiž vybírány pouze body uvnitř FOV). Poté je pomocí funkce generující náhodná čísla s normálním rozložením vybráno tisíc vzorků, které by měly dostatečně reprezentovat původní trénovací sadu. Tento předpoklad se potvrdil při porovnání histogramů původní a zmenšené trénovací sady. Tyto histogramy jsou uvedeny na Obr. 19.

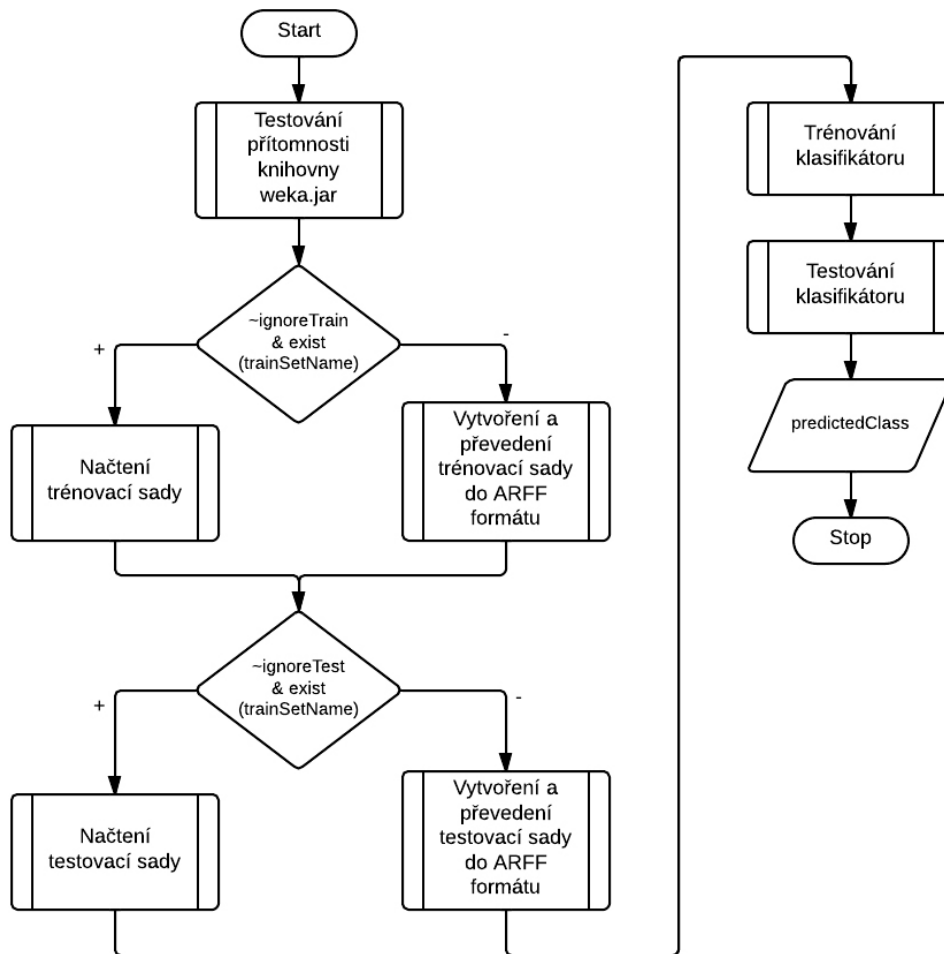


Obr. 19: a) histogram původní trénovací sady; b) histogram zmenšené trénovací sady

Po prvním spuštění každé ovládací funkce se automaticky vytvoří a uloží trénovací a testovací sady. V případě opětovného spuštění záleží na uživatelském nastavení, zda se budou datové sady vytvářet znovu, případě jestli se budou pouze načítat již vytvořené soubory. Více o možnostech nastavení se lze dočíst v příloze 2.

4.3.2. Klasifikační rámec

Nyní se dostáváme ke druhé části implementace – vytvoření klasifikačního rámce. Ten je složen z jedné hlavní funkce *framework.m* a sedmi dalších funkcí. Jedná se zjednodušeně řečeno o klasifikační část prostředí WEKA importovanou do Matlabu. Celý blok funkcí je koncipován jako klasifikační model, lze jej tedy za předpokladu správné implementace a použití daných programových prostředí využít pro klasifikaci jakýchkoli vstupních dat. Výstupem modelu je predikovaná třída. Jelikož klasifikační rámec vyžaduje specifický formát vstupních dat, je vhodné jej využít jako funkci vnořenou určitě nadřazené ovládací funkci, tak jak tomu je v případě této práce. Nemusí tomu ovšem být ve všech případech. Vytvoření klasifikačního rámce je na ovládací funkci *main* zcela nezávislý a lze jej volat přímo z příkazového řádku Matlabu (předpokladem jsou v paměti načtená potřebná data). Princip funkce klasifikačního rámce je zobrazen v základním blokovém schématu uvedeném na Obr. 20 a lze jej popsat následovně: otestování přítomnosti knihovny *weka.jar* v knihovnách Matlabu. Dále načtení, případně vytvoření, trénovací a testovací sady v závislosti na splnění daných podmínek, následované trénováním klasifikátoru. Nakonec dochází k jeho otestování a tím k vytvoření výstupů.



Obr. 20: Základní blokové schéma klasifikačního rámce

Při implementaci klasifikačního rámce je nutné vložit do zvolené složky veškeré funkce, začínající písmenem "F". Jedná se celkem o 8 funkcí (jejich názvy i popis jsou uvedeny v příloze 2). Poté již stačí v daném formátu vstupních dat zavolat funkci *framework.m*. Snahou bylo, vytvořit klasifikační rámec tak, aby uživatel nemusel nijak zasahovat do zdrojového kódu, pouze zavolat danou funkci s danými vstupy a ostatní se provedlo automaticky. Nelze tak ovšem učinit ve všech případech, proto jsou v příloze 2 popsány a ve zdrojovém kódu okomentovány všechny funkce tak, aby se uživatel v klasifikačním rámci dokázal rychle zorientovat a mohl případné úpravy provést rychle a jednoduše. Dobré orientaci napomáhá také vhodné pojmenování jednotlivých dílčích funkcí.

5. Výsledky a jejich diskuze

Veškeré výpočty byly provedeny na školním CML serveru. Využití výpočetního serveru si vynutily velmi vysoké výpočetní nároky celého procesu, zejména náročnost na paměť RAM. Pro případné testování mimo CML server je doporučeno používat PC s minimálně 8Gb paměti RAM a dvou-jádrovým procesorem. V tomto případě je však před spuštěním skriptu nutné splnit požadavky popsané v kapitole 4.3. Doporučuje se však využít CML serveru, kde jsou již nainstalované potřebné knihovny WEKA a v uživatelském profilu stačí pouze navýšit *Java Heap Memory*, viz kapitola 4.3.

Pro klasifikaci obrazu byl po úvodním testování zvolen klasifikátor Naive Bayes, popsáný v kapitole 4.1. Všechny klasifikátory byly natrénovány na sadě 1 000 vzorků, získaných z pěti snímků ze zdravé části databáze HRF [17]. Výběr reprezentativního vzorku z těchto snímků byl proveden pomocí náhodné funkce s normálním rozložením hodnot. Testování bylo poté provedeno na celém snímku 06-image_0888. Celý proces probíhal v kombinaci výsledků z metody 1 a metody 2, jelikož by teoretických předpokladů měl dosahovat nejlepších výsledků.

Tab. 2: Výsledky jednotlivých klasifikátorů na snímku 06-image_0888.jpg ze zdravé části databáze HRF

Typ klasifikátoru	Kombinace metod	Sensitivita [%]	Specifická [%]	Přesnost [%]	Doba výpočtu
bayes.BayesianLogisticRegression	metoda 1 + metoda 2	73,59	97,60	94,70	18° 55'
functions.SimpleLogistic		71,26	99,29	96,31	23° 57'
functions.Logistic		68,83	99,39	95,69	32° 12'
trees.ADTTree		52,90	99,79	94,12	32° 16'
bayes.NaiveBayes		85,12	97,38	95,90	32° 13'

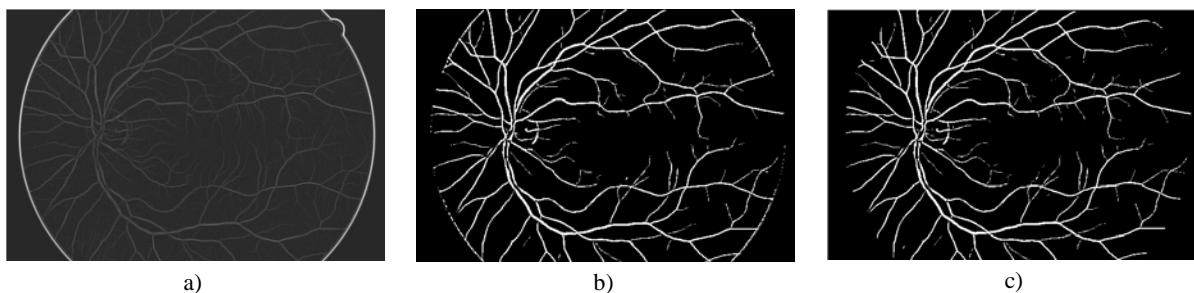
V Tab. 2 jsou uvedeny výsledky pěti zvolených klasifikátorů. První čtyři plynou z Tab. 1, jako klasifikátory, s nimiž bylo na výřezu ze snímku 06-image_0888 dosaženo nejvyšších přesností. Jelikož však byly dosažené výsledky na celém snímku 06-image_0888.jpg, při kombinaci vstupních dat z metody 1 a metody 2, poměrně nízké. Byl přidán ještě jeden klasifikátor, který je dle WEKA manuálu [59] pro svou univerzalitu a relativní jednoduchost v tomto prostředí jedním z nejpoužívanějších. Jak můžeme pozorovat i v tomto případě (viz Tab. 2), dosáhl na celém snímku nejlepších výsledků. Z tohoto důvodu byl pro další testování zvolen právě Naive Bayes klasifikátor.

Jelikož snímky z metody 1 po celém svém okraji obsahují vysoké funkční hodnoty (nežádoucí artefakt způsobený pravděpodobně nedokonalým nastavením v metodě

přizpůsobených filtrů, tento artefakt vzniká při přechodu mezi vnitřním a vnějším obsahem masky), jejichž vliv se promítnul i do výsledného snímku (viz Obr. 21 b)), bylo přistoupeno ke zmenšení masky tak, aby se tyto artefakty odstranily. Zmenšení masky bylo provedeno příkazem *imerode*, který pro úpravu využívá morfologických filtrů. Masky, definující FOV, byla u každého snímku zmenšena v poloměru o 20 obrazových bodů. Experimentálně bylo zjištěno, že tato velikost je pro odstranění nežádoucích okrajových artefaktů postačující. Zmíněná úprava navýšila u snímku 06-image_0888.jpg sensitivitu o téměř 0,7 %. Rovněž došlo k nárůstu přesnosti, a to o 0,2 %. Výsledky před úpravou a po úpravě jsou uvedeny v Tab. 3. Vliv zmenšení masky na okrajové artefakty je možné pozorovat také na Obr. 21.

Tab. 3: Vliv upravené masky na sensitivitu, specificku a přesnost

Název snímku	Kombinace metod	Typ masky	Sensitivita [%]	Specificku [%]	Přesnost [%]	Doba výpočtu
06-image_0888	pouze metoda 1	původní	84,44	97,50	95,70	21° 40'
		zmenšená	85,12	97,38	95,90	32° 13'



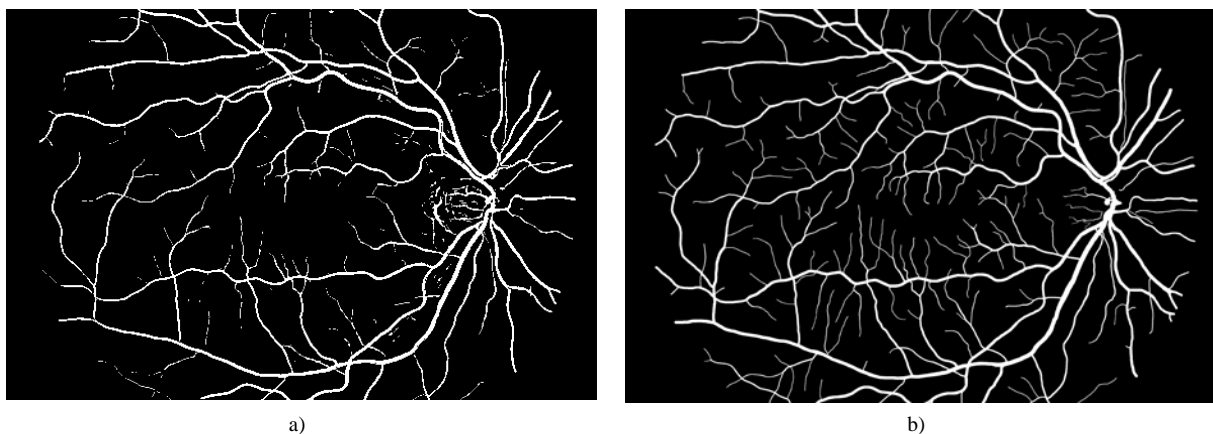
Obr. 21: a) MFR; b) klasifikovaný snímek s původní maskou; c) klasifikovaný snímek se zmenšenou maskou

V Tab. 4 jsou uvedeny výsledky dosažené se zvoleným klasifikátorem a zdravé části databáze HRF v kombinaci výsledků obou metod. Z této tabulky je patrné, že sensitivita jednotlivých snímků se pohybuje od 67,82% až 85,12%, s průměrem ze všech snímků 72,48%, mediánem 69,79% a směrodatnou odchylkou 5,5%. Vidíme, že se průměr a medián liší od sebe o necelá tři procenta. Důležitá je v tomto případě hodnota směrodatné odchylky, která je pro sensitivitu poměrně vysoká – 5,5%. To svědčí o relativně velké variabilitě dílčích hodnot. Tato variabilita je pravděpodobně způsobena rozdílnými parametry jednotlivých snímků a jejich obtížnější klasifikací. Jinými slovy se klasifikátor nedokázal zcela přesně naučit na klasifikaci bodů náležících cévnímu řečišti. Bylo by tedy vhodné zvážit rozšíření trénovací množiny o další snímky a reprezentativní vzorek 1000 vzorků tak vybírat z větší množiny trénovacích dat. Případně zvolit jiný klasifikátor, který by dosahoval lepších parametrů.

Průměrná specificku dosahuje ve srovnání se sensitivitou uspokojivých 98,59 % a v případě mediánu 98,80 %. Směrodatná odchylka dosahuje také velmi nízkých 0,58 %. Lze

tedy říci, že variabilita specificity je v tomto případě poměrně nízká. Klasifikátor je tedy při klasifikaci objektů patřících k pozadí úspěšný.

Celková přesnost dané kombinace testovacích dat dosahuje v průměru 95,75 %, její medián je 95,95 % a rozptyl nízkých 0,53 %. Což svědčí o poměrně vysoké schopnosti kombinace obou metod segmentovat cévní řečiště z retinálních snímků. Jak můžeme pozorovat z Tab. 4, Tab. 5 a Tab. 6, dosahují výsledky dosažené pomocí kombinace obou metod nejlepších výsledků. Tento fakt byl očekáván již od samého počátku celé práce – proto byla kombinace výsledků vytvořena. Kombinací došlo k vytvoření trénovacího vektor, který obsahuje pro každý bod výsledného obrazu více tzv. příznaků a klasifikátor byl tak schopen u testovacího snímku lépe určit příslušnost obrazových bodů do dané skupiny. Jinak řečeno díky většímu počtu příznaků získal klasifikátor lepší „pohled“ na daný snímek, což mu umožnilo přesněji klasifikovat jednotlivé body a následně dosáhnout lepších výsledků. Každá z testovaných metod je schopna úspěšně segmentovat jinou část cévního řečiště. Zatímco metoda 2 lépe segmentuje tenké cévy, metoda 1 je vhodná spíše pro tlustší zásobovací cévy. Na Obr. 22 a) vidíme výsledek dosažený kombinací obou metod, na snímku b) je poté odpovídající zlatý standard z databáze. Je zde patrné, že rekonstruovaný obraz postrádá i ve variantě metoda 1 + metoda 2 velké množství drobných cévek a jemných zakončení. Absence těchto cévek výrazně snižuje sensitivitu celého modelu.



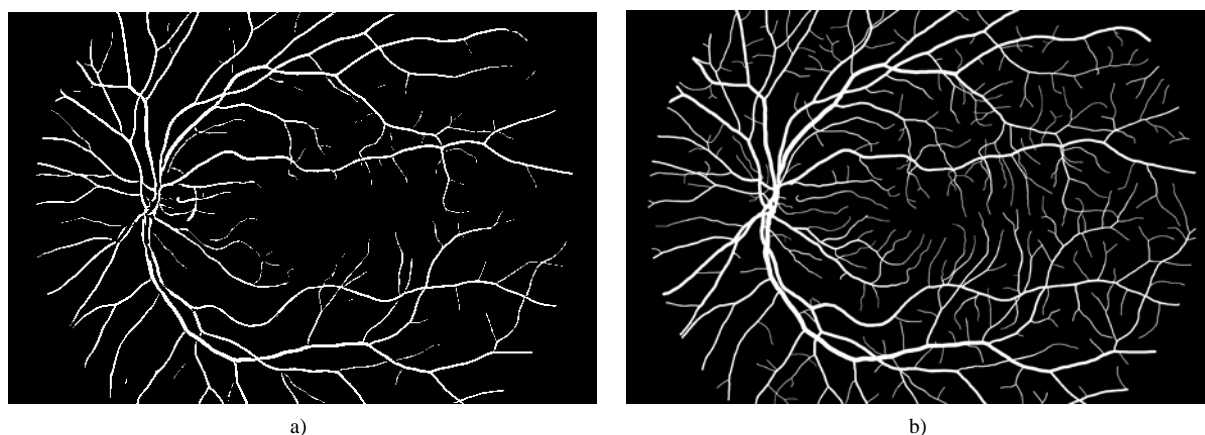
Obr. 22: Snímek 07-image_0964: a) výsledek varianty 1; b) zlatý standard z databáze

U varianty dvě, tedy samostatné výsledky metody 1, lze dosažené výsledky srovnat s původními výsledky dosaženými pomocí globálního prahování. Dosažená průměrná sensitivita je na zdravé části databáze HRF nižší téměř o 15 %. Dosažená průměrná sensitivita je na zdravé části databáze HRF nižší téměř o 15 %. Ve srovnání s variantou jedna je sensitivita nižší téměř o 9 %. Zároveň dosahuje vysoké hodnoty směrodatné odchylky, která je způsobena velmi nízkou sensitivitou u snímku 08-image_0980. Jinak jsou si hodnoty

Tab. 4: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 1 — výsledky metod 1 + 2

Název snímku	Sensitivita [%]	Specifická [%]	Přesnost [%]	Doba výpočtu
06-image_0888	85,12	97,38	95,90	32° 13'
07-image_0964	75,22	98,92	96,39	28° 57'
08-image_0980	67,95	99,36	95,55	33° 50'
09-image_1012	67,99	99,08	96,20	34° 00'
10-image_1028	70,84	98,94	96,05	32° 55'
11-image_1154	76,26	98,45	95,93	33° 35'
12-image_1165	77,01	98,67	95,77	34° 02'
13-image_1204	67,87	98,06	94,66	33° 52'
14-image_1213	68,73	98,02	94,93	40° 28'
15-image_1365	67,82	99,02	96,10	40° 27'
Průměr	72,48	98,59	95,75	34° 52'
Medián	69,79	98,80	95,92	33° 27'
Směrodatná odchylka	5,50	0,58	0,53	3° 17'

sensitivitu u ostatních snímků velmi blízké. Varianta dvě však dosáhla ze všech zkoumaných kombinací nejvyšší hodnoty specifické společně s nejnižší hodnotou směrodatné odchylky. Lze tedy říci, že metoda 1 nejlépe zařazuje body náležící k pozadí. Celková dosažená přesnost varianty 2 je ve srovnání s variantou 1 nižší přibližně o 0,8 % a ve srovnání s variantou 3 nižší o 0,3 %. Zároveň dosahuje varianta 2 u celkové přesnosti nejvyšší hodnoty směrodatné odchylky. Tato je způsobena velkými rozdíly mezi sensitivitou a specifickou jednotlivých snímků. Na Obr. 23 a) je uveden výsledek snímku 06-image_0888, dosažený variantou 2. Na snímku b) je poté odpovídající zlatý standard. Na rekonstruovaném snímku postrádáme velké množství tenkých cév, což snižuje hodnotu sensitivity.



Obr. 23: Snímek 06-image_0888: a) výsledek varianty 2; b) zlatý standard z databáze

Tab. 5: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 2 — pouze výsledky metody 1

Název snímku	Sensitivita [%]	Specifická [%]	Přesnost [%]	Doba výpočtu
06-image_0888	64,09	99,39	95,12	33° 40′
07-image_0964	66,31	99,46	95,93	23° 28′
08-image_0980	28,18	99,98	91,36	23° 34′
09-image_1012	68,83	98,75	95,96	40° 29′
10-image_1028	67,13	98,97	95,67	33° 40′
11-image_1154	65,89	99,01	94,79	32° 12′
12-image_1165	68,21	98,64	95,41	35° 54′
13-image_1204	69,11	99,36	95,45	33° 24′
14-image_1213	66,93	98,93	94,78	31° 05′
15-image_1365	68,54	99,27	95,43	33° 15′
Průměr	63,32	99,18	94,99	31° 55′
Medián	67,03	99,14	95,42	33° 21′
Směrodatná odchylka	11,80	0,38	1,27	4° 52′

Jelikož u metody [16] neexistují publikované výsledky na databázi HRF, nelze přesně vyjádřit, o kolik se hodnoty použitím klasifikace obrazu namístoglobálního prahování změnily. Jak je patrné z Tab. 6 dosáhla varianta 3 na zdravé části databáze HRF průměrné sensitivity 66,24 %, což je o více než 6 % méně, než kombinace výsledků obou metod (varianta 1). Medián je nižší o téměř 4,5 % a směrodatná odchylka je téměř shodná s variantou 1. U specifickosti dosahuje tento přístup ve srovnání s variantou 1 ve všech ukazatelích vyšších hodnot. Avšak díky výrazně nižší sensitivitě je celková dosažená přesnost metody ve srovnání s první variantou nižší o 0,4 %.

Ve srovnání s variantou 2, dosahuje třetí varianta lepších výsledků, a to téměř o 3 %. Toto je způsobeno nízkou kvalitou snímku 08-image_0980 a zároveň také celkově nižšími dílčími hodnotami varianty 2. Nižší hodnoty jsou u varianty 2 pravděpodobně způsobené menším počtem příznaků. Tato varianta totiž obsahuje pro každý vzorek (bod, který se klasifikuje) pouze dvě hodnoty příznaků, zatímco varianta 3 obsahuje příznaků pět. Zároveň jsou výsledky varianty 2 ovlivněny celkovou schopností metody 1 segmentovat spíše tlustší cévy. Rovněž má na tyto hodnoty vliv zvolený klasifikátor, a to u všech použitých variant. Druhá varianta dosáhla lepší specifickosti téměř o 0,5 %.

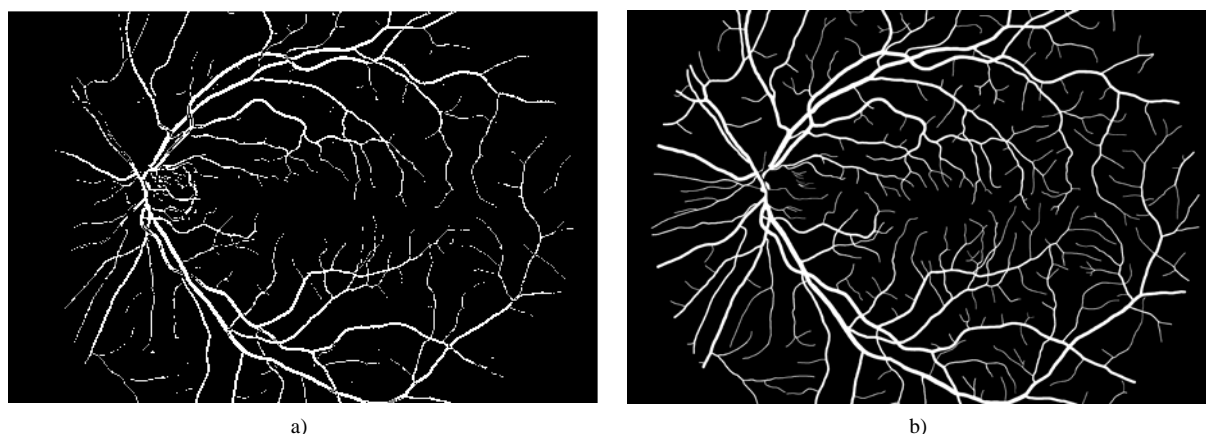
Celkově lze při porovnání metody 1 a metody 2 tvrdit následující: metoda 1 samostatně, ve srovnání s výsledky metody 2, lépe klasifikuje snímky patřící k pozadí. Celkově však

samostatná metoda AB dosáhla při segmentaci lepší přesnosti a to o 0,3 %. Tento lepší výsledek můžeme opět odvodit z parametrů uvedených v Tab. 6.

Pro subjektivní hodnocení je na Obr. 24 uveden výsledek dosažený na snímku 08-image_0980. I zde je patrné, že ve snímku chybí vůči zlatému standardu řada tenkých cévek. V případě detailnějšího zkoumání lze zjistit i absenci některých tlustších zásobovacích cév, což potvrzuje předpoklad, že metoda 2 lépe segmentuje spíše tenčí cévy.

Tab. 6: Výsledky zvoleného klasifikátoru na snímcích ze zdravé části databáze, varianta 3 – pouze výsledky metody 2

Název snímku	Sensitivita [%]	Specifita [%]	Přesnost [%]	Doba výpočtu
06-image_0888	64,09	99,39	95,12	28° 40'
07-image_0964	57,77	99,44	94,94	23° 42'
08-image_0980	65,66	99,35	95,31	23° 25'
09-image_1012	62,11	99,29	95,84	23° 32'
10-image_1028	70,25	98,87	95,92	23° 23'
11-image_1154	74,98	98,35	95,70	23° 27'
12-image_1165	75,52	98,60	95,51	22° 59'
13-image_1204	60,35	98,40	94,12	23° 06'
14-image_1213	66,46	97,90	94,58	23° 34'
15-image_1365	65,17	99,02	95,85	23° 35'
Průměr	66,24	98,86	95,29	23° 17'
Medián	65,42	98,95	95,41	23° 20'
Směrodatná odchylka	5,55	0,50	0,57	1° 32'



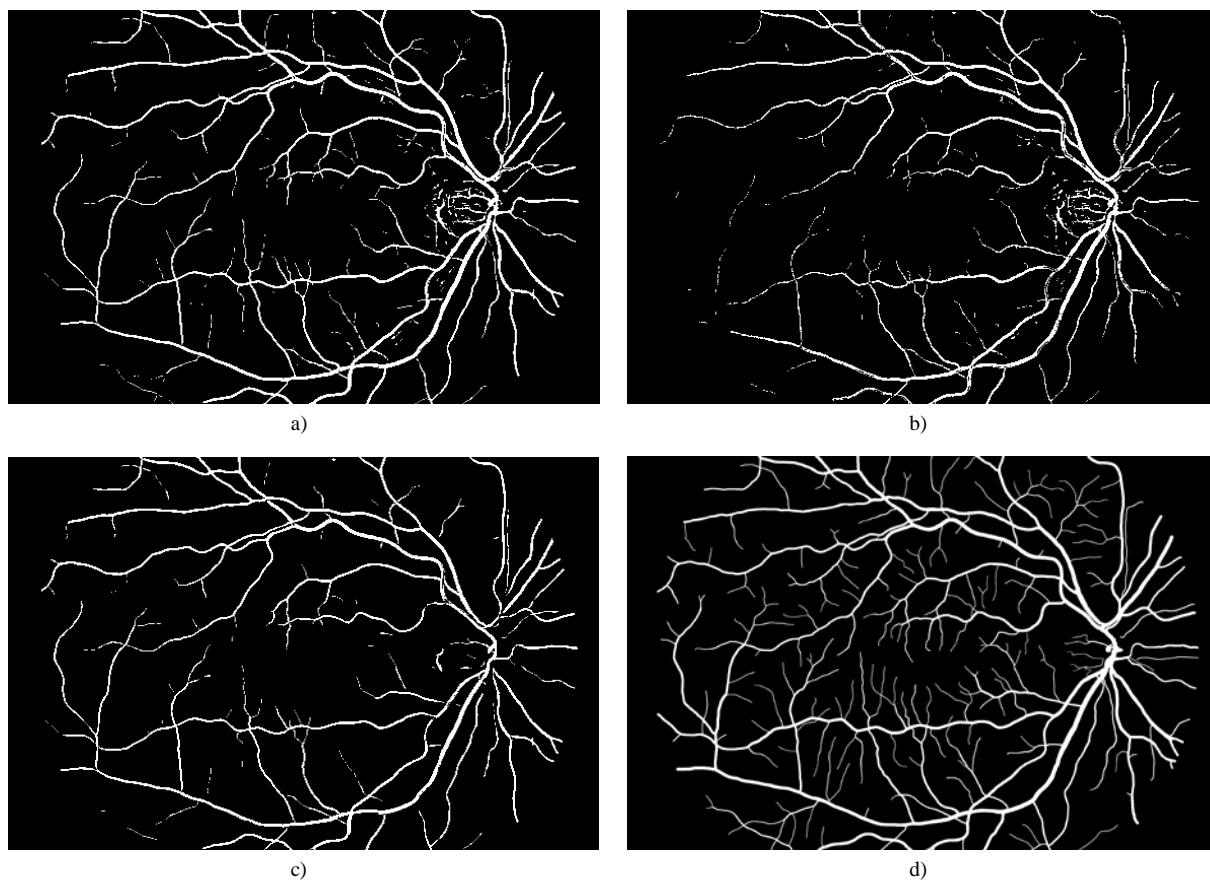
Obr. 24: Snímek 08-image_0980: a) výsledek varianty 3; b) zlatý standard z databáze

Co se týče časové náročnosti, je ze všech tří tabulek (Tab. 4, Tab. 5 a Tab. 6) patrné, že celý proces trénování a testování klasifikátoru je časově velmi náročná záležitost. Přičemž načtení datových sad, trénování klasifikátoru, rekonstrukce obrazu a následný výpočet statistik zabírá z celkového výpočetního času cca 0,01 %. Nejvíce času (99,9 %) zabere

testování klasifikátoru, tedy klasifikace vstupního obrazu. U kombinace výsledků metody 1 + metody 2 je střední doba výpočtu 33 hodin a 27 minut s rozptylem 3 hodin a 17 minut. Velký rozptyl je pravděpodobně způsoben nárazovým zatížením CML serveru ze strany ostatních uživatelů a následným snížením výpočetního výkonu pro právě probíhající klasifikaci. Poměrně velký rozptyl výpočetní doby je patný také v Tab. 5 a Tab. 6. Důvod je obdobný jako u Tab. 4. Varianta 2, pouze výsledky metody 1, provádí hodnocení klasifikace se střední dobou výpočtu 33 hodin a 21 minut. Rozptyl je poté poměrně velký – 4 hodiny a 52 minut. Třetí z variant, pouze data z metody 2, má střední dobu výpočtu 23 hodin a 20 minut s rozptylem výpočetní doby 1 hodina a 32 minut a její rozptyl je ze všech tří tabulek nejmenší.

Kombinace výsledků (metoda 1 + 2) je varianta s nejdelší průměrnou výpočetní dobou. To je způsobeno zejména faktem, že je zpracováváno více dat, než u ostatních přístupů. Ve srovnání s dobou výpočtu varianty dvě má průměrnou dobu výpočtu téměř o dvě a půl hodiny delší. Při porovnání s variant 1 a 3 je doba výpočtu varianty 1 téměř o deset hodin delší.

Pro srovnání vizuální úspěšnosti jednotlivých variant jsou na Obr. 25 uvedeny dosažené výsledky snímku 07-image_0964 společně s odpovídajícím zlatým standardem. Subjektivně je patrné, že varianta 1 ve srovnání s ostatními dosahuje nejlepších výsledků.

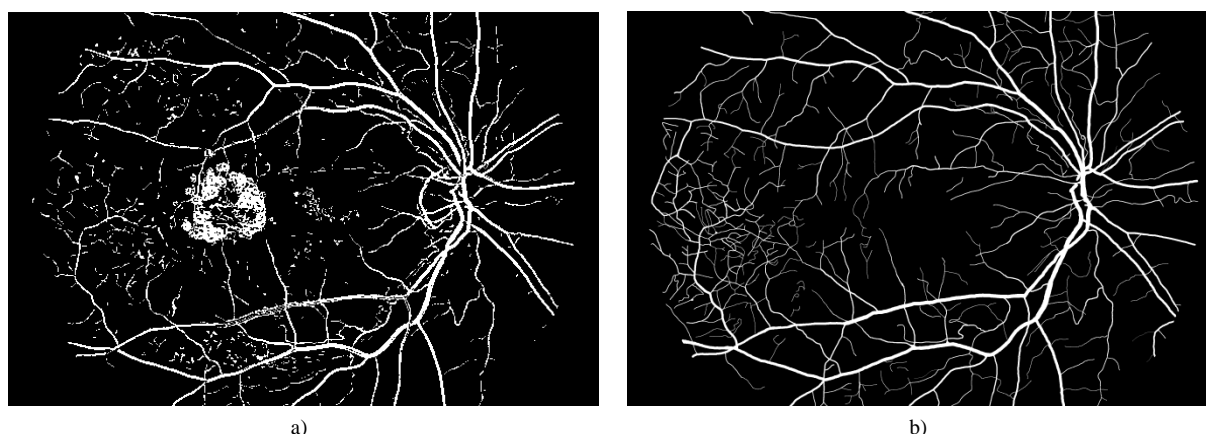


Obr. 25: Výsledky jednotlivých kombinací výsledků na snímku 07-image_0964: a) varianta 1; b) varianta 3; c) varianta 2; d) zlatý standard

Jelikož časové a výpočetní kapacity nedovolily provedení klasifikace na všech snímcích z obou patologických částí databáze HRF, byly provedeny pouze ilustrační výpočty na jednom snímku z každé z částí. V obou případech byla zvolena kombinace výsledků metody 1 + metody 2, neboť tato kombinace dosahovala na zdravé části databáze nejlepších výsledků. V Tab. 7 vidíme výsledky snímku 06-ote_R_1 postiženého diabetickou retinopatií. Je patrné, že klasifikátor dosáhl na daném snímku poměrně dobré přesnosti, i když byl trénován na zdravé části databáze. Specificita je poněkud nižší, než v případě zdravé části databáze, toto můžeme opět přiřknout tomu, že byl klasifikátor trénován pouze na zdravých snímcích. Dosažený výsledek je možné také vizuálně porovnat s odpovídajícím zlatým standardem, viz Obr. 26.

Tab. 7: Výsledky zvoleného klasifikátoru na snímcích z retinopatické části databáze, varianta metoda 1 + 2

Název snímku	Sensitivita [%]	Specificita [%]	Přesnost [%]	Doba výpočtu
06-ote_R_1	69,57	94,60	92,35	23° 48'

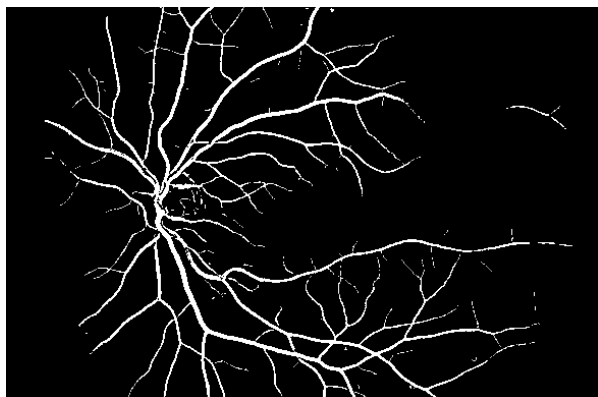


Obr. 26: Snímek 06-ote_R_1 zatížený diabetickou retinopatií: a) výsledek kombinace metody 1 + 2; b) zlatý standard z databáze

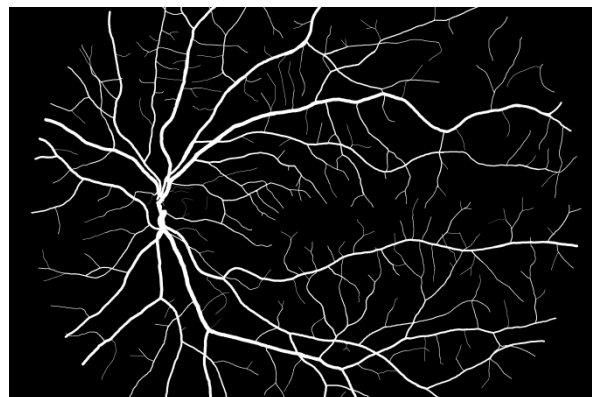
V Tab. 8 jsou uvedeny výsledky dosažené na patologickém snímku z databáze HFR. Jedná se o snímek oka postiženého glaukomem. Klasifikátor dosáhl sensitivity 71,02 %, specificity 98,31 % a přesnosti 96,08 %, což je ve srovnání se zdravou částí databáze poměrně uspokojivý výsledek. Z hodnocení jednoho snímku na patologické části databáze však není možné určit, zda je klasifikátor celkově u patologických snímků úspěšný. Pro vizuální porovnání je výsledek dosažený na glaukomatickém snímku uveden na Obr. 27. Ani zde nejsou patrná nitkovitá zakončení zásobovacích cév. Rovněž zde chybí i části tlustších zásobovacích cév, i přesto dosáhl klasifikátor na glaukomatickém snímku lepších výsledků než na retinopatickém.

Tab. 8: Výsledky zvoleného klasifikátoru na snímcích z glaukomatické části databáze, varianta metoda 1 + 2

Název snímku	Sensitivita [%]	Specifická [%]	Přesnost [%]	Doba výpočtu
06-image_0688	71,02	98,31	96,08	24° 24'



a)



b)

Obr. 27: Snímek 06-image_0688 zatížený glaukomem: a) výsledek kombinace metody 1 + 2; b) zlatý standard z databáze

6. Závěr

V první části této práce byla popsána digitální fundus kamera a shrnuty principy snímání očního pozadí. Následně byly popsány fundus snímky a vlastnosti charakteristických struktur cévního řečiště, které lze na fundus snímcích rozlišit. Rovněž je popsán vliv patologických změn cévních struktur na podporu diagnostiky očních onemocnění, i onemocnění kardiovaskulárního systému. Dále byl shrnut úvod do metod automatické segmentace obrazu, společně s příkladem rozdělení segmentačních algoritmů. V návaznosti na tento úvod byla poté sepsána literární rešerše, obsahující přehled aktuálních metod segmentace cévního řečiště z retinálních snímků. Metody byly dle přístupu k prahování rozděleny do tří skupin.

Následně byla zvolena metoda ze skupiny klasifikace obrazu s učením a pomocí programových prostředí Matlab a WEKA byla implementována do klasifikačního rámce. Vytvořený klasifikační rámec byl posléze otestován na dostupných datech. Trénování i testování klasifikačního rámce bylo provedeno na nově vzniklé databázi HRF a z ní vytvořených výsledků.

Pomocí vytvořeného klasifikačního rámce byl na jednom snímku ze zdravé části databáze HRF nalezen nejúspěšnější klasifikátor, s jehož pomocí byla následně důkladně otestována celá zdravá část databáze HRF. Testování bylo provedeno ve třech variantách. Tyto varianty byly odvozeny z dostupné výsledkové sady. Ta je složena z výsledků dvou metod segmentace cévního řečiště. Byla proto testována každá výsledková sada samostatně a posléze byla testována i jejich kombinace. Pro ilustraci vytvořeného klasifikačního rámce klasifikovat také patologické snímky, byl otestován jeden snímek zatížený glaukomem a jeden zatížený diabetickou retinopatií. Oba tyto snímky jsou z patologické části databáze HRF.

Byla získána rozsáhlá sadavýsledků, přičemž nejvyšší přesnosti dosáhl zvolený klasifikátor (Naive Bayes) při kombinaci výsledků z metody 1 + 2, a to 95,75 %. Následuje přístup využívající pouze výsledků z metody 2 s 95,29 % přesnosti. Nejnižší přesnosti bylo dosaženo při použití samostatných výsledků z metody 1, a to 94,99 %.

V rámci teoretické části práce, byly stanoveny jisté předpoklady, které byly následně v praktické části ověřeny. První z teoretických předpokladů – zvyšující se přesnost metody společně s vyšším počtem příznaků, byl jednoznačně potvrzen. Při srovnání výsledků na zdravé části je patrné, že se přesnost metody v kombinaci více příznaků zvyšuje a může dosahovat rozdílu téměř jednoho procenta, což je v případě dnes publikovaných metod

poměrně výrazná hodnota. V případě dalšího testování by tedy mohlo být jedním z možných postupů, pro zlepšení dosažených přesností, přidání dalších příznaků popsanych například v práci [14]. Samozřejmě je nutné brát ohled na jejich vhodnost.

Druhý teoretický předpoklad – dosažení lepších výsledků pomocí klasifikace obrazu s učením, se daným klasifikátorem nepotvrdil, neboť dosažené průměrné výsledky sensitivity jsou ve srovnání s původními, získanými globálním prahováním, nižší téměř o 6 %. Celková přesnost metody se poté snížila o 0,4 %. Toto snížení může být ovlivněno zejména použitým klasifikátorem, jelikož volba byla provedena pouze za pomoci jednoho snímku, na kterém bylo testováno několik klasifikátorů. Ovšem tímto mohl být zvolen klasifikátor, který je pro daný snímek velmi přesný, neboť obsahuje pro klasifikátor ideální podmínky, avšak u dalších snímků databáze již nevykazuje tak přijatelné výsledky. Vhodnější by proto bylo otestovat více klasifikátorů např. na pěti snímcích, a ze získaných statistik poté zvolit nejlepší klasifikátor. Vzhledem k velmi velké časové náročnosti procesu testování klasifikátoru byla však volba provedena pouze na základě analýzy jednoho snímku. V návaznosti na tuto část se tedy nabízí varianta důkladnějšího otestování dalších klasifikátorů a případná změna klasifikátoru, spojená s novým testováním.

Dále se pro zlepšení výsledků nabízí úprava trénovací sady, respektive volba reprezentativního vzorku z více snímků. V tomto případě bylo voleno 1 000 vzorků z pěti snímků (z téměř 31 milionů vzorků). Tyto snímky ovšem nemusí obsahovat veškeré „vlastnosti“, které by byly pro správné natrénování a následnou klasifikaci obrazu potřebné. Proto by bylo vhodné vybírat vzorky z více snímků, například pro trénování klasifikátoru použít celou sadu zdravých snímků z databáze HRF – tedy použít všech 15 snímků. Tento postup však vyžaduje velmi výkonnou pracovní stanici, extrémní požadavky jsou kladeny zejména na paměť RAM. Pro představu můžeme uvést, že při výběru ze zmiňovaných 5 snímků bylo u varianty výsledky metody 1 + 2 využito 28 Gb paměti RAM. Tato náročnost je způsobena tím, že pro práci s daty je nutné je nejprve všechna načíst do paměti RAM a náročné zpracování pomocí prostředí WEKA. Pokud pracujeme s kombinací výsledků z obou metod je třeba každý snímek vynásobit počtem příznaků a zlatého standardu. Do paměti RAM je tak nutné načíst téměř 155 000 000 různých vzorků a z nich poté vybrat pouze 1 000.

Možným řešením paměťové i výpočetní náročnosti se jeví také přímá implementace klasifikačních bloků do programového prostředí Matlab. Jedná se ovšem o poměrně složitý postup, vyžadující speciálních nástaveb a velmi dobrých znalostí jednotlivých klasifikátorů. I při provedení této implementace se však naskýtá jedna velká nevýhoda, programové prostředí

Matlab totiž velice špatně podporuje více-jádrové procesory. Nelze tedy jednoduše rozdělit výpočty mezi více jader a využít tak plných možností výkonných pracovních stanic. Nabízí se tak možnost, vytvořit celý klasifikační rámec v jiném programovém prostředí, které plně podporuje více-jádrové procesory, například C++. Tento přístup však od uživatele navíc, oproti stávajícím řešení, vyžaduje velmi dobré znalosti klasifikace obrazu i programového prostředí C++ a pokročilé znalosti algoritmizace úloh.

Ještě před změnou klasifikátoru by bylo vhodné za použití stávajícího dokončit testování obou patologických sad, jelikož by klasifikátor na patologických sadách mohl dosahovat uspokojivých přesností. Testováním jednoho snímku z každé patologické části databáze toto nelze stanovit.

Celý pracovní postup by bylo vhodné testovat i na jiných databázích. Jelikož databáze HRF, na které byl celý postup navrhnout a otestován může obsahovat jistou „vlastnost“, která nemusí být v běžném snímku tolik patrná, a která může uměle zvyšovat úspěšnost celého procesu. Na databázích jiného typu by tak mohla být přesnost daných metod výrazně odlišná.

Celkově lze zkonstatovat, že bylo v práci otestováno několik snímků v různé kombinaci metod. Dále byl s daným klasifikátorem vyvrácen předpoklad dosažení lepších výsledků použití klasifikace obrazu. Tento přístup však nelze zcela zamítnat, jelikož použitý klasifikátor nemusel být zvolen správně. Nabízí se tedy hlubší analýza jiných klasifikátorů a rovněž také testování na různých typech zdravých i patologických snímků.

7. Seznam použité literatury

- [1] Bock R., Meier J., Nyúl G. L., Hornegger J., Michelson G. Glaucoma risk index: Automated glaucoma detection from color fundus images. Medical Image Analysis . Frankfurt : Elsevier B.V., 2009.
- [2] Kousal B., Dubská Z. Věkem podmíněná makulární degenerace - principy a léčba. Postgraduální medicína. [Online] E15, 3. Květen 2010. [Citace: 25. listopad 2012.] <http://zdravi.e15.cz/clanek/postgradualni-medicina/vekem-podminena-makularni-degenerace-principy-a-lecba-450134>.
- [3] Detection and segmentation of drusen deposits on human retina: Potential in the diagnosis of age-related macular degeneration. Rapantzikos K., Zervakis M., Balas K. Thessaloniki : Elsevier B.V., 2001. Medical Image Analysis 7. stránky 95-108.
- [4] Sinthanayothin C., Boyce F. J., Williamson T. H., Cook H. L., Mensah E., Lal S., Usher D. Automated detection of diabetic retinopathy on digital fundus images. Diabetes UK. Diabetic Medicine. 2002, 19, stránky 105-112.
- [5] Wong T. Y., Klein D., Couper D. J., Cooper L. S., Shahar E., Hubbard L. D., Wofford M. R. Retinal microvascular abnormalities and the incident stroke the Atherosclerosis Risk in Communities Study. The Lancet. 358, 2001.
- [6] Wisaeng K., Hiransakolwong N., Ekkarat Pothiruk. Automatic Optic Disc Detection from Low Contrast Retinal Images. Applied Mathematical Sciences. 6. edice. 2012, stránky 5127-5136.
- [7] Sinthanayothin C., Boyce J. F., Cook H. L., Williamson T. H. Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images. Br F Ophthalmol. 1999, stránky 902-910.
- [8] Weinreb R. N., Dreher A. W., Coleman A., Quigley H., Shaw B., Reiter K. Histopathologic Validation of Fourier-Ellipsometry Measurements of Retinal Nerve Fiber Layer Thickness. Arch Ophthalmol. 1990, stránky 557-560.
- [9] Al-Rawi M., Qutaishat M., Mohammed A. An improved matched filter for blood vessel detection of digital retinal images. Computers in Biology and Medicine 37. Vol. 37. 2007.
- [10] Hoover A., Kouznetsova V., Goldbaum M. Locating Blood Vessels in Retinal Images by Piecewise Threshold Probing of a Matched Filter Response. Transactions on Medical Imaging. Vol. 19. 2000, stránky 203-210.
- [11] Staal J., Abramoff M. D., Niemeijer M., Viergever M. A., Ginneken B. Ridge-Based Vessel Segmentation in Color Images of the Retina. TRANSACTIONS ON MEDICAL IMAGING. 2004.
- [12] Lu S., Liu J., Lim J. H., Zhang z., Meng T. N., Wong W. K., Li H., Wong T. Y. Automatic Fundus Image Classification for Computer-Aided Diagnosis. 31st Annual International Conference of the IEEE EMBS. Minneapolis : IEEE, 2009.
- [13] Duda R.O., Hart P.E., Stork D.G. Pattern classification. 2. edice. 2005. str. 738. 0471056693.
- [14] Lupascu C. A., Tegolo D., Trucco E. Retinal Vessel Segmentation Using AdaBoost. IEEE Transaction on Informarion Technology in Biomedicine. 2010.

- [15] Odstrčilík J., Jan J., Gazárek J., Kolář R. Improvement of Vessel Segmentation by Matched Filtering in Colour Retinal Images. [autor knihy] Dössel O. and Schlegel W.C. IFMBE Proceedings. 2009, stránky 327-330.
- [16] Budai A., Michelson G., Hornegger J. Multiscale Blood Vessel Segmentation in Retinal Color Images. [pdf] Erlangen-Nürnberg : Friedrich-Alexander-Universität, 2010.
- [17] Odstrčilík J., Svoboda O.; Budai A., Kolář R., Jan J., Gazárek J. Fundus gold standard database; Gold standard database for evaluation of fundus image segmentation algorithms. [ZIP] Erlangen-Nuremberg; Brno : Friedrich-Alexander University a Vysoké učení technické v Brně, FEKT, Ústav biomedicínského inženýrství, 2011.
- [18] D., Volk A. Diagnostic indirect ophthalmoscopy contact lens system. 5200773 Ohio, 6. Duben 1993.
- [19] Kolář R. Fundus kamery, konfokální mikroskopie - přednáška. Diagnostika bio a eko systémů. [Online] [Citace: 23. listopad 2012.] http://www.ubmi.feec.vutbr.cz/~kolarr/mdbe/Kolar_MDBE_3.pdf.
- [20] Walter D. Reviewing the Clinical Importance of Field of View. Optos. [Online] [Citace: 24. listopad 2012.] http://www.optos.com/Global/documents/Optomeric_Mgnt_May_2010_supplement.pdf.
- [21] Fundus Photography Overview. Ophthalmic Photographers' Society. [Online] Ophthalmic Photographers' Society, Inc. [Citace: 24. listopad 2012.] <http://www.opsweb.org/?page=fundusphotography&terms=what+and+is+and+a+and+fundus+and+camera>.
- [22] Patrick J. S., Marshall E. T. Ophthalmic Photography: Retinal Photography, Angiography, and Electronic Imaging. 2 edice. Oxford : Butterworth-Heinemann Medical.
- [23] Haishuy Y., Zhishan G., Ting L., Yang H. Optical configuration of fundus camera based on inner focusing manner. 2010.
- [24] Introduction to Stereomicroscopy. MICROSCOPYU:COM. [Online] Nikon. [Citace: 27. listopad 2012.] <http://www.microscopyu.com/articles/stereomicroscopy/stereointro.html>.
- [25] Tsuru K., Inoue T., Origo N., Matuyama M., Nanjo T., Yano N., Hino T. A study and development for stereo fundus camera sytem using Stereo Matching Technique. Internatiol Archives of Protogrammetry ans Remove Sensing. Vol. 31. 1996.
- [26] Ltd, MedWOW. Manufacturer Specifications - CF-60UVi Fundus, Canon. MedWOW. [Online] [Citace: 23. listopad 2012.] <http://www.medwow.com/med/retinal-camera/canon/cf-60uvi-fundus/32233.model-spec>.
- [27] Lantham & Phillips Ophthalmic. Fundus camera CANON CF-60 UVi specifications. [Online] [Citace: 24. listopad 2012.] <http://lpoproducts.com/Products/LPO13.asp>.
- [28] Saine J. P. Focusing The Fundus Camera: A Clinical Approach. [Online] [Citace: 25. listopad 2012.] http://c.ymcdn.com/sites/www.opsweb.org/resource/resmgr/boc_resources_pdf/14-1-02.pdf.

- [29] Ho A. C., Brown C. G., McNamara A. J., Recchia M. F., Regillo D. C., Vander F. J. Color atlas & synopsis of clinical optalmology. 2003. 0-07-137596-1.
- [30] Dehghani A., Moghaddam H. A., Moin M. S. Optic disc localization in retinal images using histogram matching. Journal on Image and Video Processing. [Online] 2012. [Citace: 25. listopad 2012.] <http://jivp.eurasipjournals.com/content/pdf/1687-5281-2012-19.pdf>.
- [31] Mohammadi K., Bowd Ch., Weinreb R. N., Medeiros F. A., Sample P. A, Zangwill L. M. Retinal nerve fiber layer thickness measurements with scanning laser polarimetry predict glaucomatous visual field loss. American Journal of Ophthalmology. 2004, stránky 592-601.
- [32] Jomier J., Wallace K. D., Aylward R. S. Quantification of Retinopathy of Prematurity via Vessel Segmentation. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003. Chapel Hill USA : Springer Berlin Heidelberg, 2003.
- [33] Prevalence of glaucoma. The Beaver Dam Eye Study. Klein B.E., Klein R., Sponsel W.E., Franke T., Cantor L.B., Martone J., Menage M.J. Madison : Research Support, U.S. Gov't, P.H.S., 1992.
- [34] Köse C., Şevik U., Gençalioglu O. Automatic segmentation of age-related macular degeneration in retinal fundus images. Computers in Biology and Medicine. St. Louis : Elsevier B.V., 2008, stránky 611-619.
- [35] Pal R. N., Pal S. K. A review on imagesegmentation techniques. Vol. 26. 1993, stránky 1277-1294.
- [36] Ilona Kalová. Segmentace a detekce geometrických primitiv. Počítačové vidění. [Online] [Citace: 27. listopad 2012.] <http://www.uamt.feec.vutbr.cz/vision/TEACHING/MPOV/05%20-%20Segmentace%20a%20detekce%20geometrickych%20primitiv.pdf>.
- [37] Sezgin M., Sankur B. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging 13. 2004, stránky 146-165.
- [38] Jan J. Medical image processing, reconstruction, and restoration; concepts and methods. Boca Raton : Taylor & Francis, 2006. 978-0-8247-5849-3.
- [39] A Review on Image Segmentation Techniques with Remote Sensing Perspective. Dey V., Zhang Y., Zhong M. 2010.
- [40] Haralick R. M., Shapiro L. G. Image segmentation techniques. [editor] Ann Arbor. Computer Vision, Graphics, and Image Processing. Vol. 29. Michigan 48104, USA : Machine Vision International, 1985, stránky 100-132.
- [41] Provazník I. Úvod do biomedicínské informatiky, Pravděpodobnost v rozhodování. [PDF] Brno : ÚBEMI VUT, 2009.
- [42] Cinsdikici M. G., Aydin D. Detection of blood vessels in ophthalmoscope images using MF/ant (matched filter/ant colony) algorithm. Computer methods and programs in biomedicine. 96. 2009, stránky 85-95.
- [43] Poli R., Valli G. An algorithm for Real-time Vessel Enhancement and Detection. [PDF] Birmingham : School of Computer Science The University of Birmingham, Edgbaston.

- [44] Zana F., Klein J. C. Segmentation of Vessel-Like Patterns Using Mathematical Morphology and Curvature Evaluation. Transactions on Image Processing. Vol. 10. 2001, stránky 1010-1019.
- [45] Detection of Blood Vessels in Fundus Images of the Retina using Gabor Wavelets. Oloumi Fa., Rangayyan R. M., Oloumi Fo., Eshgtaled-Zanjani P., Ayres F. J. 2007. stránky 6451-6454.
- [46] Mendonça A. M., Campilho A. Segmentation of Retinal Blood Vessels by Combining the Detection of Centerlines and Morphological Reconstruction. Transactions on Medical Imaging. Vol. 25. 2006, stránky 1200-1213.
- [47] Ricci E., Perfetti R. Retinal Blood Vessel Segmentation Using Line Operators and Support Vector Classification. Transactions on Medical Imaging. Vol. 26. 2007, stránky 1357-1365.
- [48] Chaudhuri S., Shankar Ch., Katz N., Nelson M., Goldbaum M. Detection of Blood Vessels in Retinal Images Using Two-Dimensional Matched Filters. Transactions on Medical Imaging. Vol. 9. 1989, stránky 263-269.
- [49] Otsu N. A Threshold Selection Method. Transaction on System, man, and cybernetics. Vol. 9. 1979, stránky 62-66.
- [50] Chanwimaluang T., Fan G. An Efficient Blood Vessel Detection Algorithm for Retinal Images. [PDF] Stillwater : School of Electrical and Computer Engineering, Oklahoma State University, 2003.
- [51] Kande G. B., Savithri T. S., Subbaiah P. V. Retinal Vessel Segmentation using Local Relative Entropy Thresholding. [PDF] 2008.
- [52] Jiang X., Mojon D. Transactions on Pattern Analysis and Machine Intelligence. 2003, stránky 131-137.
- [53] Conduchare A. P., Mertins A., Aach T. Supervised, hysteresis-based segmentation of retinal images using the linear-classifier percentile. [PDF] Luebeck, Aachen : autor neznámý, 2005.
- [54] Condurache A. P., Müller F., Mertins A. An LDA-based relative hysteresis classifier with application to segmentation of retinal vessels. [PDF] Lubeck : Institute for Signal Processing, University of Luebeck, 2010.
- [55] Soares J. V. B., Landro J. J. G., Caesar R. M. Jr., Jelinek H. F., Cree M. J. Retinal Vessel Segmentation Using the 2-D Morlet Wavelet and Supervised Classification. [PDF] Sao Paulo, Hamilton, Charles Sturt : IEEE, 2006.
- [56] Marín D., Aquino A., Gegúndez-Arias E. M., Bravo J. M. A New Supervised Method for Blood Vessel Segmentation in Retinal Images by Using Gray-Level and Moment Invariants-Based Features. [PDF] místo neznámé : IEEE, 2010.
- [57] Svoboda O. Pokročilé segmentační techniky a prahování objektů na fotografiích sítnice: semestrální práce. [PDF] Brno : Vysoké učení technické v Brně, FEKT, Ústav biomedicínského inženýrství, 2013.
- [58] Genkin A., Lewis D.D., Madigan D. Large-Scale Bayesian Logistic Regression for Text Categorization. American Statistical Association and the American Society for Quality Technometrics. 49, 2007, Sv. 3.

- [59] M., Eibe F, Holmes G., Pfahringer B., Reutemann P., Witten I. H. A comprehensive Weka 3.7.7 manual. Weka 3: Data Mining Software in Java. [Online] [Citace: 9. prosinec 2012.] [http://sourceforge.net/projects/weka/files/documentation/3.7.x/WekaManual-3-7-7.pdf/download?use_mirror=freefr&download=.](http://sourceforge.net/projects/weka/files/documentation/3.7.x/WekaManual-3-7-7.pdf/download?use_mirror=freefr&download=)
- [60] Proceedings of the 25th international conference on Machine learning: Discriminative parameter learning for Bayesian networks. Su J., Zhang H., Ling Ch. X., Matwin S. New York : autor neznámý, 2008. 978-1-60558-205-4.
- [61] Estimating Continuous Distributions in Bayesian Classifiers. John G. H., Langley P. San Mateo : Morgan Kaufmann Publishers, 1995.
- [62] Cessie Le S., Houwelingen van J. C. Ridge Estimators in Logistic Regression. Royal Statistical Society. 1992.
- [63] Friedman J., Hastie T., Tibshirani R. Additive Logistic Regression: a Statistical View of Boosting. [PDF] Stanford : Department of Statistic, Stanford University, 1998.
- [64] Freund Y., Schapire R. E. Large Margin Classification Using the Perceptron Algorithm. Machine Learning. Kluwer Academic Publishers : autor neznámý, 1999.
- [65] Experiments with a new boosting algorithm. Freund Y., Schapire R. E. 1996 : autor neznámý, San Francisco.
- [66] Friedman J., Hastie T., Tibshirani R. Additive logistic regression: a statistical view of boosting. Annals of Statistics. 28, 2000, 2.
- [67] Webb G. I. MultiBoosting: A Technique for Combining Boosting and Wagging. Machine Learning. 40, 2000, 2.
- [68] Hall M., Eibe F, Holmes G., Pfahringer B., Reutemann P., Witten I. H. Weka 3: Data mining software in Java. [Online] The University of Waikato, Waikato, New Zealand. [Citace: 9. prosinec 2012.] [http://www.cs.waikato.ac.nz/ml/weka/.](http://www.cs.waikato.ac.nz/ml/weka/)
- [69] Freund Y., Mason L. The alternating decision tree learning algorithm. [PDF] Florham Park; Craneberry : autor neznámý, 1999.
- [70] Quinlan R. J. Programs for Machine Learning. San Mateo : Morgan Kaufmann Publishers, 1993.
- [71] Breiman L. Random Forests. Machine Learning. 45, 2001.
- [72] Agresti A. Categorical Data Analysis. New York : Wiley-Interscience, 2002. ISBN 0-471-36093-7.
- [73] The MathWorks, Inc. How do I increase the heap space for the Java VM in MATLAB 6.0 (R12) and later versions? [Online] The MathWorks, Inc., 26. duben 2012. [Citace: 18. března 2013.] [http://www.mathworks.com/support/solutions/en/data/1-18I2C/.](http://www.mathworks.com/support/solutions/en/data/1-18I2C/)

8. Použité zkratky a symboly

FOV	Fieldofview
CCD	Charge-coupled device
CMOS	Complementary Metal Oxide Semiconductor
TP	True positive
FP	False positive
TN	True negative
FN	False negative
MFR	Matched Filter Response

9. Seznam příloh

Příloha 1 – Popis ovládacích funkcí main.m	62
Příloha 2 – Manuál ke klasifikačnímu modelu	64

Příloha 1 – Popis ovládacích funkcí *main.m*

Hlavní funkce *main.m* je pro snadné testování jednotlivých postupů vytvořena ve třech verzích (důvod vytvoření těchto verzí byl popsán v kapitole 4). Každá verze zpracovává vstupní data z jiné metody, případně jejich kombinaci. Jednotlivé verze se ovšem od sebe liší pouze minimálně. Jako referenční funkci berme *mainJOAB.m*, neboť pracuje se všemi proměnnými. Zbývající dvě funkce mají omezené využití některých proměnných. Jako příklad můžeme uvést funkci *mainAB.m*, která nevyužívá proměnnou *MFRList*. Dále se funkce od sebe liší pouze v proměnných *adressesData*, *train{i}.features*, *trainDataPom2* a *testDataPom*. Všechny ostatní proměnné i podfunkce jsou společné. Z tohoto titulu je na následujících stranách popsán princip funkce *main.m*, podfunkce i proměnné pouze jednou. Funkce *main.m* využívá čtyř dalších podfunkcí. Tyto jsou popisovány v pořadí tak, jak jsou postupně volány v hlavní funkci.

Main.m

Jedná se o hlavní ovládací funkci celé práce. Její základní blokové schéma je uvedeno na Obr. 18. Ve funkci jsou nejprve definovány globální proměnné a adresy ke všem potřebným souborům. Poté jsou vytvořena dvě struktury obsahující názvy trénovací a testovací množiny. Následuje dle nastavení na řádcích 19 a 20 definování pomocných proměnných, nebo vytvoření trénovacích a testovacích dat společně s vykreslením histogramů trénovací sady. Vliv zmíněného nastavení na celý průběh je popsán v příloze 2. Vytvoření trénovací sady se od prostého načtení a překódování testovací specificky liší a je popsáno v kapitole 4.3.1. Po načtení obrazových dat dochází také k morfologickému menšení masky. Důvod tohoto zmenšení i vliv na dosažené výsledky je popsán v kapitole 5. Po vytvoření datových sad je následně volána funkce *framework.m*, jejíž výstupní predikovaná třída ve formě sloupcového vektoru je následně převedena zpět do maticové formy a vizualizována. Dochází také k výpočtu dosažené SE, SP a přesnosti.

Určení snímku (předposlední znak na daném řádku), který bude použit pro testování klasifikátoru, se provádí na řádku 75 funkce *mainJOAB.m*, potažmo 73 u funkce *mainAB.m*, respektive 74 u funkce *mainJO.m*. Volba se provádí uvedením čísla daného řádku ve struktuře, která obsahuje adresy k testovacím souborům z databáze HRF. Standardně jsou testovací snímky v této práci voleny ze zdravé části databáze a začínají od snímku s počátečním číslem 06-xxx, neboť prvních pět snímků ze zdravé části databáze je použito pro trénování klasifikátoru. V případě, že uživatel zvolí jiný snímek, než byl původně použitý, je

vhodné změnit název a adresu testovací sady uložené v proměnné *testSetName*, případně přepsat proměnnou *ignoreTest* na *true*, nebo alespoň smazat původní testovací souboru *ARFF*.

LoadImg.m

Je funkce použita načtení veškerých obrazových dat. Je v ní ošetřena nedokonalá komprese masky i chybějící MFR u funkce *mainAB.m*.

MatrixToVector.m

Tato funkce převádí za použití příkazu *reshape* vstupní snímky po sloupcích do sloupcového vektoru. Převod se provádí pro každý vstupní parametr zvlášť a data se skládají vedle sebe. Vznikne tak až 6-ti dimenzionální sloupcový vektor. Každý vzorek je zároveň převeden na typ *cell*, jelikož nepřítomnost tohoto datového typu ovlivní další zpracování dat zpracování klasifikačním rámcem. Tato funkce má rovněž ošetřen chybějící soubor příznaků *features*.

VectorToMatrix.m

Slouží k převodu predikované sloupcové třídy do maticového tvaru, který lze reprezentovat jako obrázek. Tato rekonstrukce je prováděna za použití masky pomocí dvou cyklů *for*. Vždy, když je v masce na daném místě hodnota „1“, umístí cyklus do nového snímku na danou pozici bod z predikované třídy. Po rekonstrukci sou ve vytvořeném snímku pomocí příkazu *bwareaopen* odfiltrovány oblasti menší než 100 px. Experimentálně bylo zjištěno, že tato velikost je vzhledem k sensitivitě nejlepší. Se zvětšující se velikostí této oblasti se může sensitivita měnit až o jednotky procent. Následně jsou vypočítány statistiky pro daný snímek a zobrazeny výsledky.

CountRatios.m

Funkce, která pomocí podmíněného indexování spočte TP, FP, TN, FN a z těchto údajů následně určí dosaženou SE, SP a přesnost klasifikátoru pro daný snímek. Dosažené výsledky poté zobrazí uživateli.

Příloha 2 – Manuál ke klasifikačnímu modelu

Tato příloha slouží jako manuál k prezentovanému klasifikačnímu rámci. Je v ní popsán jeho princip, způsob implementace, veškeré důležité funkce a v závěru i možné uživatelské změny. Je určen pro uživatele s pokročilejšími znalostmi programových prostředí Matlab i WEKA. Detailní informace o programovém prostředí WEKA můžeme nalézt v kompletním manuálu k programu [59]. Jednoduchou nápovědu, jak spustit a otestovat klasifikační rámec na dostupných datech, skrývá také soubor *README.txt* uložený na dvouvrstevném DVD, které je přiloženo k této práci.

Vytvořený rámec je určen ke klasifikaci vstupních dat, vstupem mohou být data ve sloupcových vektorech, minimálně o dvou sloupcích. Výstupem je poté jeden sloupcový vektor s příslušností do dané třídy. Tříd, do kterých se budou vstupní data zařazovat, může být v predikovaném vektoru samozřejmě několik.

Klasifikační rámec kombinuje programové prostředí Matlab a WEKA, přičemž celý skript je napsaný v Matlabu s voláním potřebných funkcí z importované knihovny WEKA. K možnosti vytvořit toto propojení přispěl také fakt, že jsou obě prostředí založena na jazyce *Java*. Nejprve je pro správný běh celého modelu nutné importovat do knihoven Matlabu hlavní knihovnu *weka.jar*. Toto je možné provést zadáním příkazu *edit classpath.txt* do příkazového řádku Matlabu a následným vložením celé cesty ke knihovně *weka.jar*. Jako příklad můžeme uvést: *C:/Program Files/Weka-3-6/weka.jar*. Po upravení a uložení tohoto souboru je nutné restartovat Matlab. U uživatelů s nižšími, než administrátorskými právy je nutné brát v potaz zápis souborů na disk C.

Poznámka: V případě, že se chystáme zpracovávat objemné datové soubory (řádově statisíce, až miliony vzorků v každém sloupci vstupních dat) je nutné navýšit v Matlabu paměť pro *Java* výpočty. Tato položka je v Matlabu od verze R2010 umístěna v položce *File » Preferences » General » Java Heap Memory*. Posuvník je nutné umístit zcela vpravo, maximální hodnota této paměti je $\frac{1}{4}$ paměti RAM. Po uložení je nutné opět restartovat Matlab. U starších verzí Matlabu je dostupnou RAM možné nastavit pouze pomocí speciálního konfiguračního souboru *java.opts*. Jedná se o poměrně náročný postup, který od uživatele požaduje pokročilé znalosti programovacího jazyka *Java* i velmi dobrou orientaci v operačním systému. Celý postup vytvoření tohoto souboru i jeho umístění a zprovoznění můžeme nalézt v [73].

Po provedení potřebných prvotních nastavení je možné následně plně vyžít klasifikační rámec. Ten obsahuje 8 funkcí, které jsou popsány níže v tomto návodu a na příloženém DVD, jsou ve složce *Code* označeny počátečním písmenem F (všechny ostatní funkce slouží ke konverzi dostupných výsledků do sloupcových vektorů a následnou rekonstrukci obrazových dat). Klasifikační rámec je nutné volat ve specifickém formátu vstupních proměnných. Tento formát je popsán níže.

Nyní přistupme k popisu jednotlivých funkcí, na hlavní ovládací funkci si zároveň popíšeme princip funkce celého klasifikačního rámce. Funkce jsou popisovány v pořadí, ve kterém jsou volány v hlavní ovládací funkci *framework.m*

Framework.m

Jedná se hlavní ovládací (spouštěcí) funkci celého rámce. Tato funkce nejprve vytvoří trénovací a testovací sadu, následně vytvoří a natrénuje klasifikátor, který si uživatel zvolil a nakonec daný klasifikátor otestuje a vytvoří predikovanou třídu. Tento postup se může lišit v případě, že uživatel zvolí místo vytváření nových datových sad, načtení již dříve vytvořených sad ze souboru *ARFF*. Při tomto nastavení se datové sady nevytváří znovu, ale pouze se načtou. Toto nastavení může u velkých souborů urychlit dobu běhu celého skriptu až o několik hodin. Proto je doporučeno si již vytvořené sady ukládat (v případě defaultního nastavení provádí model automaticky, viz uživatelské nastavení) a při další práci je pouze načítat. Avšak je nutné brát v potaz, že WEKA primárně pracuje se specifickým formátem *ARFF*, je tedy nutné data ukládat do tohoto formátu.

Jak již bylo řečeno, funkci *framework.m* je nutné volat ve specifickém formátu. Má osm vstupních a jednu výstupní proměnnou. Volání této funkce vidíme zde:

```
predictedClass = Framework(trainDataPom2, testDataPom, featureNames,
ignoreTrain, ignoreTest, classifierType, trainSetName, testSetName)
```

Nyní popíšme význam jednotlivých vstupních i výstupních proměnných:

predictedClass – Je klasifikátorem vytvořená predikovaná třída. Jedná se o výstupní jedno-sloupcový vektor, o stejné délce jako testovací sada. Obsahuje celočíselné příslušnosti každého vzorku (řádku) z testovací sady do dané skupiny. Počet predikovaných skupin je dán počtem skupin v posledním sloupci trénovací sady (na predikci této třídy je trénován klasifikátor, lze změnit, viz Možné uživatelské změny a nastavení klasifikačního rámce). Příklad predikované třídy pro datovou sadu se třemi skupinami (0-2) vidíme v Tabulka P2 č. 1.

Tabulka P2 č. 1: Ukázka predikované třídy se třemi skupinami

0
0
1
0
2
0
2

`trainDataPom2` – Je trénovací množina pro klasifikátor. Jak již bylo řečeno, jedná se o sloupcový vektor s minimálně dvěma sloupci. Délka není omezena, avšak vzhledem k tomu, že jsou délka filtrování dat i trénování klasifikátoru exponenciálně závislé na počtu vzorků, doporučuje se snížit počet trénovacích vzorků na minimum. Optimální počet vzorků se pohybuje mezi jedním až deseti tisíci vzorky. Správnost zmenšené trénovací lze ověřit porovnáním histogramů původní a zmenšené datové sady. V Tabulka P2 č. 2 vidíme příklad trénovací sady se šesti sloupci (příznaky)

Tabulka P2 č. 2: Ukázka trénovací sady se šesti sloupci

0.458221	3	4	0	0	0
0.840284	0	2	0	0	0
1.530141	1	7	0	0	0
1.601894	0	4	0	0	0
1.244326	0	6	0	0	0
1.517082	0	9	0	0	0
1.664654	2	5	0	0	0

`testDataPom` – Testovací sada pro klasifikátor. U testovací sady bychom logicky předkládali klasifikátoru sadu zmenšenou o sloupec obsahující příslušnosti do dané třídy (poslední sloupec v datové sadě). WEKA ovšem požaduje, aby trénovací i testovací sada měly stejný počet sloupců. Tato sada obsahuje všechny body, které uživatel požaduje klasifikovat, může jich být až několik milionů. V tom případě je ovšem nutné opět zohlednit časovou náročnost celého procesu.

Tabulka P2 č. 3: Ukázka testovací sady se dvěma sloupci

26.155661	1
0.328192	0
2.244031,0	0
4.800005	0
1.21757	0

U testovací i trénovací sady je důležité, aby každý vzorek ve sloupcových vektorech byl uložen ve formátu *cell*, nikoli *number*.

`featureNames` – Pole buněk (*cell*) o stejné délce, jako je počet sloupců v testovací a trénovací sady. Obsahuje názvy jednotlivých sloupců. Názvy jsou typu *string*.

`ignoreTrain` a `ignoreTest` – Proměnné typu *boolean*. Ovlivňují ignoraci vytvořené trénovací nebo testovací sady. Jinými slovy, pokud jsou nastavené na *false* a trénovací nebo testovací sada existuje, tak ji skript pouze načte. Pokud jsou nastavené na *true*, pak skript vytvoří a uloží novou testovací sadu, bez ohledu na to, zda již existuje v nějakém souboru.

`classifierType` – Jedna z nejdůležitějších vstupních proměnných. Obsahuje řetězec s názvem a typem zvoleného klasifikátoru. Zápis je ve specifickém formátu a v proměnné typu *string*. WEKA obsahuje velké množství klasifikátorů a jejich popis přesahuje rámec této práce. Námi testovaný klasifikátor je popsán v kapitole 4.1. Další klasifikátory, včetně jejich zápisu v programovém prostředí WEKA lze nalézt v[59]. Příklad zápisu klasifikátoru:

```
classifierType='bayes.BayesianLogisticRegression'
```

`trainSetName` a `testSetName` – Proměnné typu *string* obsahující celou cestu a název trénovací a testovací sady.

FWekaPathCheck.m

Jedná se o velmi důležitou funkci, neboť kontroluje přítomnost knihovny *weka.jar* v seznamu knihoven Matlab. Pokud není daná knihovna přítomna, celý skript se ukončí a vypíše se text s požadavkem na přidání dané knihovny do seznamu.

FLoadARFF.m

Funkce načte data ze souboru *ARFF* do objektu *wekaOBJ* a nastaví predikovanou třídu na poslední sloupec.

FDataFilter.m

Jelikož je pro řadu klasifikátorů nutné, aby predikovaná třída (tedy poslední sloupec v datové sadě) byla v nominálním, nikoli numerickém formátu, je nutné daný sloupec přefiltrovat do daného typu. K tomu slouží právě funkce s názvem *FDataFilter.m*. Filtrace je provedena za použití filtru s názvem *NumericToNominal*, který je obsažen v knihovně funkcí *weka.jar* pod položkou *filters.unsupervised.attribute*. Celý proces filtrace je poměrně časově náročný a převedení posledního sloupce do nominálního tvaru je nutné provést u trénovací i testovací sady. Proto je vhodné, aby alespoň velikost trénovací sady byla co nejmenší. Pro ilustraci přefiltrování 16 milionů vzorků trvá přibližně 4 hodiny 30 minut. Doba filtrace je také závislá na výkonnosti pracovní stanice.

Funkce pracuje tak, že nejprve převede vstupní datovou sadu do formátu *ARFF*, poté vytvoří filtr, u něhož definuje typ a sloupec k filtraci a ten následně aplikuje na vytvořenou datovou sadu. Výstupem funkce je přefiltrovaná datová sada v podporovaném formátu *ARFF*, se kterou je dále pracováno v dalším postupu. Funkce je univerzální pro trénovací i testovací sadu.

FMatlabToWeka.m

Funkce *FMatlabToWeka.m* je podružnou, avšak velmi důležitou částí celého rámce, neboť provádí konverzi vstupních dat do formátu *ARFF*, jehož struktura je popsána v kapitole 4.2. Ve funkci je také definována třída *class*, na jejíž predikci je trénován klasifikátor. Standardně se jedná o poslední sloupec v datové sadě.

Funkce jako taková neuloží data přímo do *ARFF* souboru, ale převede je do objektu, ve který lze uložit do daného formátu. Tedy obsahují hlavičku s definicí názvů sloupců a poté datovou část, ve které jsou uloženy vzorky. Takto vytvořený objekt je poté zpracován nadřazenou funkcí *FDataFilter.m* tak, jak je popsáno výše.

FSaveARFF.m

Funkce ukládající objekt obsahující data ve formátu *ARFF* do souboru s názvem a adresou určenou vstupní proměnnou.

FTrainWekaClassifier.m

Tato funkce má jako vstupní proměnnou trénovací množinu dat a typ klasifikátoru, který si uživatel zvolil jako vstupní proměnnou klasifikačního modulu. Na základě těchto dat je jako výstupní objekt vytvořen daný klasifikátor.

FWekaClassify.m

Funkce FWekaClassify.m provádí ověření klasifikátoru na testovacích datech, neboli vytvoří ze vstupních testovacích dat predikovanou třídu, která je reprezentována celočíselnými příslušnostmi do jednotlivých tříd. Počet a typ tříd je dán datovým sloupcem v trénovací sadě, který je označen jako *class*.

Možné uživatelské změny a nastavení klasifikačního rámce

Uživatel má možnost celé řady změn, které se projeví na funkčnosti celého modelu. Zde budou popsány základní úpravy:

1. Změna proměnných *ignoreTrain* a *ignoreTest* – standardně jsou nastaveny na *false*, pokud tedy již existuje trénovací nebo testovací sada, rámec ji znovu nevytvoří, ale pouze ji načte z adresy definované jako *trainSetName*, respektive *testSetName*. Pokud bude nastavena na *true*, bude vždy vytvořena a uložena nová datová sada.
2. Změna názvu testovací a trénovací sady – tyto názvy jsou uloženy v proměnných *trainSetName* a *testSetName*. Tyto proměnné obsahují jak název, tak celou cestu k těmto souborům. Lze tedy měnit nejen názvy, ale i umístění.
3. Změna klasifikátoru – uživatel má možnost měnit typ použitého klasifikátoru, stačí změnit obsah vstupní proměnné *classifierType*.
4. Změna predikované třídy *class* – Je možno změnit třídu označenou jako *class*. Změna záleží na tom, zda vytváříme novou datovou sadu, v tomto případě je nutno změnit ve funkci FMatlabToWeka.m řádek 5 – *targetIndex*. Pokud pouze načítáme již hotovou sadu, mění se tato třída ve funkci FLoadARFF.m na řádku 9 – *wekaOBJ.setClassIndex()*.
5. Uložení trénovací nebo testovací sady – pokud si uživatel nepřeje z jakéhokoli důvodu ukládat vytvořené datové sady, stačí ve funkci *framework.m* zakódovat nebo odstranit řádky 18, respektive 29.