



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

SEGMENTACE CÉVNÍHO ŘEČIŠTĚ V RETINÁLNÍCH OBRAZOVÝCH DATECH

BLOOD VESSEL SEGMENTATION IN RETINAL IMAGE DATA

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Johana Vančurová

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Odstrčilík, Ph.D.

BRNO 2019

Diplomová práce

Magisterský navazující studijní obor **Biomedicínské inženýrství a bioinformatika**

Ústav biomedicínského inženýrství

Studentka: Johana Vančurová

ID: 174536

Ročník: 2

Akademický rok: 2018/19

NÁZEV TÉMATU:

Segmentace cévního řečiště v retinálních obrazových datech

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s principem snímání očního pozadí pomocí experimentálního video oftalmoskopu a s vlastnostmi takto získaných obrazových dat. Zejména se zaměřte na charakteristické struktury, které na snímcích tvoří cévní řečiště. 2) Provedte literární rešerši aktuálních prací zabývajících se segmentací cévního řečiště na snímcích sítnice. Zaměřte se zejména na metody, které jsou robustní vůči šumu a artefaktům v obraze. 3) Vybranou metodu blíže nastudujte a proveďte její implementaci ve vybraném programovém prostředí. 4) Implementované algoritmy následně otestujte a vhodně vyhodnoťte na dostupných obrazových datech z video oftalmoskopu. 5) Provedte diskusi dosažených výsledků a zhodnoťte účinnost a využitelnost aplikovaného řešení. 6) K vytvořeným programovým funkcím sepište přehledný návod k obsluze.

DOPORUČENÁ LITERATURA:

[1] ALMOTIRI, J. et al. Retinal Vessels Segmentation Techniques and Algorithms: A Survey, vol. 155, no. 8, pp. 1-331, 2018.

[2] JIANG, Z., et al. Retinal blood vessel segmentation using fully convolutional network with transfer learning. Computerized Medical Imaging and Graphics, vol. 68, Sep., pp. 1-15, 2018.

Termín zadání: 4.2.2019

Termín odevzdání: 17.5.2019

Vedoucí práce: Ing. Jan Odstrčilík, Ph.D.

Konzultant:

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.

Fakulta elektrotechniky a komunikačních technologií, Vysoké učení technické v Brně / Technická 3058/10 / 616 00 / Brno

Abstrakt

Tato diplomová práce se zabývá segmentací cévního řečiště v retinálních obrazových datech. Teoretická část je zaměřena na základní popis anatomie a fyziologie oka a na metody snímání očního pozadí. Dále jsou popsány principy klasických a konvolučních neuronových sítí a segmentační techniky, které se využívají k segmentaci cévního řečiště v retinálních snímcích. V praktické části je implementována segmentační metoda pomocí konvoluční neuronové sítě U-net. Tato síť je natrénována na třech datových sadách. Dvě datové sady obsahují snímky z experimentálního video oftalmoskopu. Z důvodů nemožnosti porovnat výsledky této metody na těchto dvou datových sadách, byla použita třetí datová sada HRF, která zahrnuje snímky pořízené fundus kamerou. Výsledky testování na této datové sadě slouží k porovnání s jinými metodami segmentace cévního řečiště.

Klíčová slova

Experimentální video oftalmoskop, neuron, neuronové sítě, konvoluce, segmentace, U-net, fundus snímky

Abstract

This master's thesis deals with blood vessel segmentation in retinal image data. The theoretical part is focused on the basic description of anatomy and physiology of the eye and methods of observing the back of the eye. This thesis also describes the principles of classical and convolutional neural networks and segmentation techniques that are used to segment blood vessel in retinal images. In the practical part, a segmentation method using convolutional neural network U-net is implemented. This neural network is trained on the three datasets. Two datasets include images from experimental video ophthalmoscope. Because it is impossible to compare the results of these two datasets with any other methods of retinal blood vessel segmentation, U-net is trained on other dataset that is HRF database. This dataset includes fundus images. The results of testing on this dataset serve for comparing results with other methods of retinal blood vessel segmentation.

Keywords

Experimental video ophthalmoscope, neuron, neural network, convolution, segmentation, U-net, fundus images

Bibliografická citace:

VANČUROVÁ, Johana. *Segmentace cévního řečiště v retinálních obrazových datech*. Brno, 2019. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/118346>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství. Vedoucí práce Jan Odstrčilík.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma „Segmentace cévního řečiště v retinálních obrazových datech“ jsem vypracovala 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 závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědoma 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.

V Brně dne 17. května 2019

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce Ing. Janu Odstrčilíkovi Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce. Dále bych také ráda poděkovala Ústavu biomedicínského inženýrství, který mi poskytl možnost využít grafickou kartu, která zjednodušila praktickou část mé diplomové práce.

Obsah

Úvod	13
1 Oko	15
1.1 Anatomie sítnice	16
2 Metody snímání očního pozadí	17
2.1 Oftalmoskopie	17
2.1.1 Přímá oftalmoskopie	17
2.1.2 Nepřímá oftalmoskopie	18
2.2 Fundus kamera	18
2.3 Experimentální video oftalmoskop	20
2.3.1 Vlastnosti výstupu z video oftalmoskopu	20
3 Umělá neuronová síť	22
3.1 Umělý neuron	22
3.2 Topologie sítě	24
3.3 Učení sítě	24
3.4 Chybové funkce	25
3.5 Optimalizační algoritmy	26
4 Konvoluční neuronové sítě	29
4.1 Vrstvy konvolučních neuronových sítí	29
4.2 Přeučení neuronové sítě	32
4.2.1 Augmentace dat	33
4.2.2 Dropout	33
4.2.3 Early stopping	34
5 Segmentace obrazu	35
5.1 Metody segmentace cévního řečiště v retinálních snímcích	35
5.2 U-NET	40
6 Vyhodnocovací metody	42
6.1 Senzitivita	42
6.2 Specificita	42
6.3 AUC	43
6.4 ACC	44
6.5 Dice koeficient	44
7 Implementace U-net	45
7.1 Datové sady z experimentálního video oftalmoskopu	46
7.1.1 Experimentální sada A	46
7.1.2 Experimentální sada B	47
7.2 Datová sada HRF	48
8 Výsledky a diskuze	50
8.1 Experimentální datová sada A	50

8.2	Experimentální datová sada B.....	55
8.3	Porovnání výsledků experimentálních datových sad	57
8.4	Datová sada HRF.....	58
8.5	Porovnání výsledků datové sady HRF s jinými metodami	62
9	Návod k obsluze programů.....	63
9.1	Příprava.....	63
9.2	Trénování	64
9.3	Testování	64
	Závěr.....	65
	Literatura	67
	Použité zkratky	72
	Seznam příloh.....	73
	Příloha 1: Obsah přiloženého CD	74

Seznam obrázků

Obrázek 1: Řez okem, převzato z [5]	15
Obrázek 2: Princip přímé oftalmoskopie, převzato z [8]	17
Obrázek 3: Princip nepřímé oftalmoskopie, převzato z [9]	18
Obrázek 4: Schéma fundus kamery, převzato z [10]	19
Obrázek 5: Snímek pořízený pomocí fundus kamery	19
Obrázek 6: Schéma experimentálního video oftalmoskopu, převzato z [11]	20
Obrázek 7: Zprůměrovaný snímek z experimentálního video oftalmoskopu	21
Obrázek 8: Umělý neuron, převzato z [15]	22
Obrázek 9: Příklady aktivčních funkcí, převzato z [15]	23
Obrázek 10: Topologie dopředné neuronové sítě, převzato z [15]	24
Obrázek 11: Princip konvoluce vstupního obrazu s kernel filtrem, převzato z [24]	29
Obrázek 12: Aktivační funkce ReLU, převzato z [25]	30
Obrázek 13: „Max-pooling“, převzato z [25]	31
Obrázek 14: Plně propojená vrstva, převzato z [28]	32
Obrázek 15: Přetrénování sítě	33
Obrázek 16: Architektura sítě navržená Jinnanem Guo a spol. [39]	38
Obrázek 17: Výřezy vstupního obrazu (a) a jejich „ground truth“ (b) [40]	39
Obrázek 18: Architektura CNN podle Zhongwei Feng a spol. [40]	39
Obrázek 19: Architektura U-net, převzato z [35]	40
Obrázek 20: ROC křivka, převzato z [44]	43
Obrázek 21: Příklad snímků a jejich příslušných binárních masek z experimentální sady A	46
Obrázek 22: Příklad uměle rozšířené experimentální sady A rotací snímků o 30°. 47	
Obrázek 23: Příklady původních snímků z experimentální sady B, jejich příslušných původních masek a masky převedené na binární	47
Obrázek 24: RGB snímek z fundus kamery a jeho zobrazení v jednotlivých kanálech.	48
Obrázek 25: Příklad z datové sady HRF	49
Obrázek 26: Průběh trénování sítě na experimentální sadě A po 100 epochách	51
Obrázek 27: Průběh chybové funkce při trénování sítě na experimentální sadě A po 100 epochách	51
Obrázek 28: Testování na experimentální sadě A	52
Obrázek 29: Průběh trénování sítě na augmentované experimentální sadě A po 20 epochách	53
Obrázek 30: Průběh chybové funkce při trénování sítě na experimentální sadě A po 20 epochách	54

Obrázek 31: Testování na augmentované experimentální sadě A.....	55
Obrázek 32: Průběh trénování na experimentální sadě B po 30 epochách	56
Obrázek 33: Průběh chybové funkce při trénování sítě na experimentální sadě B po 30 epochách.....	56
Obrázek 34: Testování na experimentální sadě B	57
Obrázek 35: Průběh trénování na datové sadě HRF po 50 epochách.....	58
Obrázek 36: Průběh chybové funkce při trénování sítě na datové sadě HRF po 50 epochách.....	59
Obrázek 37: Testování na datové sadě HRF	59
Obrázek 38: Průběh trénování na augmentované datové sadě HRF po 50 epochách	60
Obrázek 39: Průběh chybové funkce při trénování sítě na augmentované datové sadě HRF po 50 epochách	61
Obrázek 40: Testování na augmentované datové sadě HRF	62

Seznam tabulek

Tabulka 1: Statistické pojmy	42
Tabulka 2: Hodnocení kvality testu podle AUC	43
Tabulka 3: Výsledky trénování pro experimentální datovou sadu A.....	50
Tabulka 4: Výsledky testování pro experimentální datovou sadu A.....	52
Tabulka 5: Výsledky trénování pro augmentovanou experimentální datovou sadu A	53
Tabulka 6: Výsledky testování pro augmentovanou experimentální datovou sadu A	54
Tabulka 7: Výsledky trénování pro experimentální datovou sadu B.....	55
Tabulka 8: Výsledky testování pro experimentální datovou sadu B.....	57
Tabulka 9: Testování natrénované sítě na experimentální sadě B pro testovací snímky z experimentální sady A	58
Tabulka 10: Výsledky trénování pro datovou sadu HRF	58
Tabulka 11: Výsledky testování pro datovou sadu HRF	59
Tabulka 12: Výsledky pro augmentovanou datovou sadu HRF	60
Tabulka 13: Výsledky testování pro augmentovanou datovou sadu HRF	61
Tabulka 14: Porovnání výsledků datové sad HRF s jinými metodami	62

ÚVOD

Oči jsou nepostradatelnou součástí lidského těla. Díky očím jsme schopni vnímat světlo, barvy, okolí a také se orientovat v prostoru. Díky možnosti vidět všechno kolem sebe, jsme schopni samostatně manipulovat s různými věcmi. Pro člověka, který má schopnost vidění, může být těžké si představit s jakými problémy se potýká člověk, který bohužel takové štěstí nemá. Proto jsou důležitá všechna vyšetření, která vedou ke správné diagnostice onemocnění očí a všech dalších nemocí, které jsou s očima spojené.

Segmentace cévního řečiště v retinálních snímcích je nezbytným krokem před další analýzou snímků z fundus kamery nebo jiného přístroje, který je schopen zachytit oční pozadí. Tato segmentace může pomoci při diagnostice onemocnění oka jako je glaukom nebo diabetická retinopatie a dále také při diagnostice kardiovaskulárních onemocnění [1]. Proto je velmi důležité se touto segmentací cévního řečiště v retinálních snímcích zabývat.

Hlavním cílem této diplomové práce je implementace a testování segmentační metody cévního řečiště v retinálních snímcích pořízených experimentálním video oftalmoskopem. Diplomová práce je rozdělená do 9 kapitol.

První kapitola se zabývá základní anatomií a fyziologií oka a je zde blíže popsána anatomie sítnice. V druhé kapitole jsou popsány metody snímání očního pozadí, jako je přímá a nepřímá oftalmoskopie. Dále je zde uvedený princip fundus kamery a experimentálního video oftalmoskopu s popisem výstupních dat z toho přístroje.

Třetí kapitola je zaměřena na teorii neuronových sítí, kde je popsán umělý neuron, učení sítě a také různé topologie neuronových sítí. Dále jsou zde zmíněny chybové funkce pro hodnocení odchylky neuronové sítě a optimalizační algoritmy, které slouží k minimalizaci chybové funkce. Čtvrtá kapitola se zabývá konvolučními neuronovými sítěmi, jejich architekturou a funkčností. Je zde taky popsán syndrom přeučení sítě a jak tomu to syndromu předejít.

V páté kapitole je obecně popsána segmentace obrazových dat a jsou zde také uvedeny metody segmentace cévního řečiště v retinálních snímcích. Nejběžnějšími metodami segmentace cévního řečiště jsou prahování a v posledních pár letech také rychle se rozvíjející konvoluční neuronové sítě. Z nastudovaných metod byla vybrána konvoluční neuronová síť U-net, která byla vytvořena pro segmentaci biomedicínských dat. Vybraná metoda byla blíže nastudována a v této kapitole popsána.

Šestá kapitola se zabývá statistickým hodnocením výsledků testování neuronových sítí. Jsou zde popsány vybrané statistické parametry jako je senzitivita,

specifická, přesnost, ROC křivka a plocha pod křivkou. Posledním zmíněným parametrem je Dice koeficient, který se v poslední době hojně využívá.

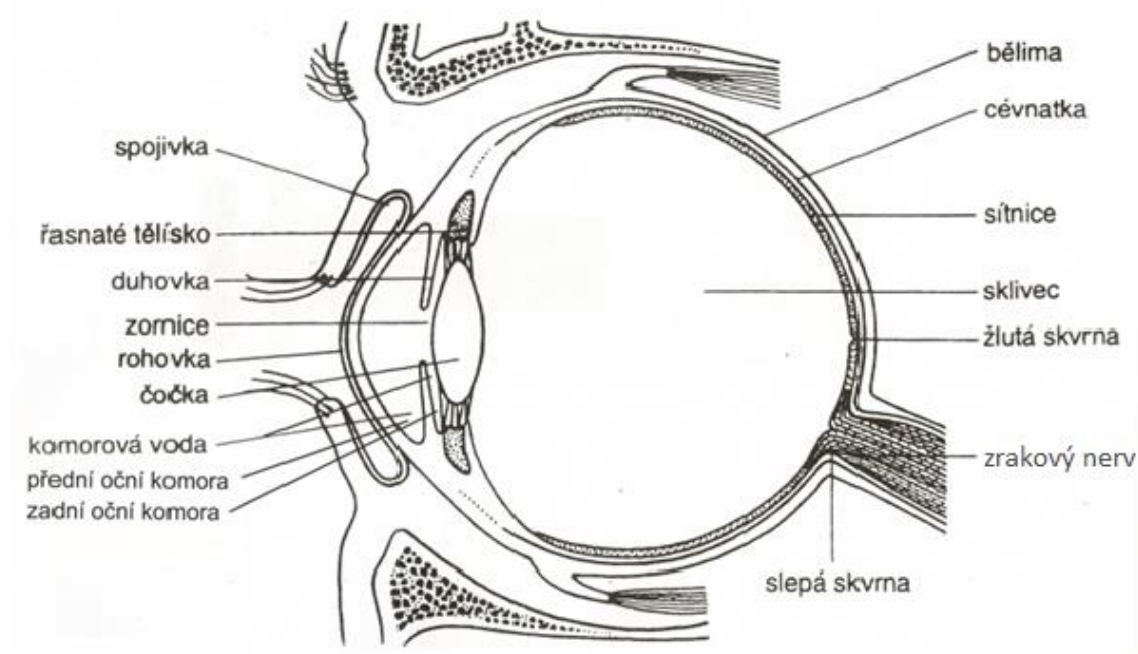
Následující kapitola popisuje implementaci U-net v programovém jazyce Python. Dále jsou zde popsány všechny použité datové sady, které byly využity jak k trénování, tak k testování sítě. Dvě z těchto datových sad obsahují snímky získané zprůměrováním výstupu z experimentálního video oftalmoskopu a třetí sada zahrnuje snímky z fundus kamery. Třetí datová sada slouží k porovnání výsledků z ostatními metodami zabývajícími se segmentací cévního řečiště v retinálních snímcích.

V osmé kapitole jsou popsány výsledky trénování a testování implementované konvoluční neuronové sítě U-net. Dále jsou zde porovnány výsledky prvních dvou vytvořených datových sad a třetí datová sada je porovnána s vybranými metodami segmentace cévního řečiště. V poslední deváté kapitole je uveden stručný návod k obsluze všech programů a skriptů, které byly pro účely diplomové práce vytvořeny.

1 OKO

Lidské oko je párový orgán, který se skládá z oční koule a přídatných očních orgánů. Jelikož je oko zrakový orgán, jsme schopni díky němu vnímat světlo, barvy a také se orientovat v prostoru. Lidským okem je možné detekovat úzké rozmezí světelného spektra, přibližně od 400 do 750 nm vlnové délky. Nejdelší vlnovou délku vnímáme jako červenou barvu a nejkratší jako modrou barvu. [2] [3]

Oční koule má téměř kulovitý tvar o průměru zhruba 23 mm a je uložena v očníci na tukovém polštáři, který ji chrání před otřesy. Řez okem je vidět na obrázku 1, kde jsou popsány všechny části oka. [3] [4]



Obrázek 1: Řez okem, převzato z [5]

Oční kouli tvoří tři vrstvy, kterými jsou vnější, střední a vnitřní vrstva. Vnější vrstvu tvoří bílá a neprůhledná bělima a rohovka, která je průhledná a představuje jedno z optických médií oka. Střední vrstvu tvoří cévnatka, která je bohatě prokrvena, dále řasnaté tělísko, na kterém je zavěšena čočka, a duhovka, jejímž středem je zornice, kterou procházejí paprsky světla dovnitř oční koule. Poslední vrstvou je vnitřní vrstva, která je tvořena sítnicí.

Uvnitř oční koule se nachází průhledné rosolovité těleso zvané sklivec. Dále pak čočka bikonvexního tvaru, která díky změně poloměru svého zakřivení zajišťuje vidění zblízka i do dálky. Čočka je uložena v tekutině, které se nazývá komorový mok. [3] [6]

Přídavné oční orgány tvoří okohybné svaly, víčka, spojivka a slzný aparát. Okohybné svaly umožňují pohyb oční koule, víčka chrání oči před vnějším prostředím a jsou opatřeny řasami a žlázkami, které produkují maz. Vnitřní plochu víček kryje tenká blána nazývaná spojivka, v jejímž vaku se hromadí slzy. Tyto slzy produkují slzné žlázy, které jsou součástí slzného aparátu. [3]

1.1 Anatomie sítnice

Sítnice neboli retina představuje vnitřní část oční koule. Vystýlá celou dutinovou část oční koule a naléhá na její prostřední vrstvu. Sítnice je tvořena dvěma oddíly, kterými jsou přední a zadní optická část. Přední část sítnice se nazývá slepá, protože neobsahuje žádné světločivné buňky, ale pouze pigmentové. Na rozdíl od přední části sítnice, zadní část obsahuje světločivné buňky, kterými jsou tyčinky a čípky. Sítnice obsahuje asi 120 miliónů tyčinek a 6 - 7 miliónů čípků. [3] [6]

Tyčinky reagují na tlumené světlo a obsahují zrakové barvivo rhodopsin, který při dopadu světelných paprsků mění barvu (bledne) a tím vyvolává nervové vzruchy.

Čípky reagují na jasné osvětlení a obsahují tři různá zraková barviva, která mají různou citlivost na světlo o různé vlnové délce. Tímto vzniká diferencovaná citlivost ke třem barvám, kterými jsou červená, zelená a modrá. Skládáním těchto tří barev vzniká barevný obraz. Místo, kde se nachází pouze čípky, se nazývá žlutá skvrna. Tato skvrna je místem nejostřejšího vidění. [2] [3] [6]

Zadní optická část sítnice obsahuje kromě tyčinek a čípků také vrstvu bipolárních a gangliových buněk, kterými se šíří nervových vzruch z tyčinek a čípků. Místo, kde se sbíhají nervová vlákna sítnice a nevyskytují se zde žádné světločivné buňky, je slepá skvrna neboli optický disk. Optickým diskem vstupují do oka tepny a vystupují z oka žíly. [3] [6]

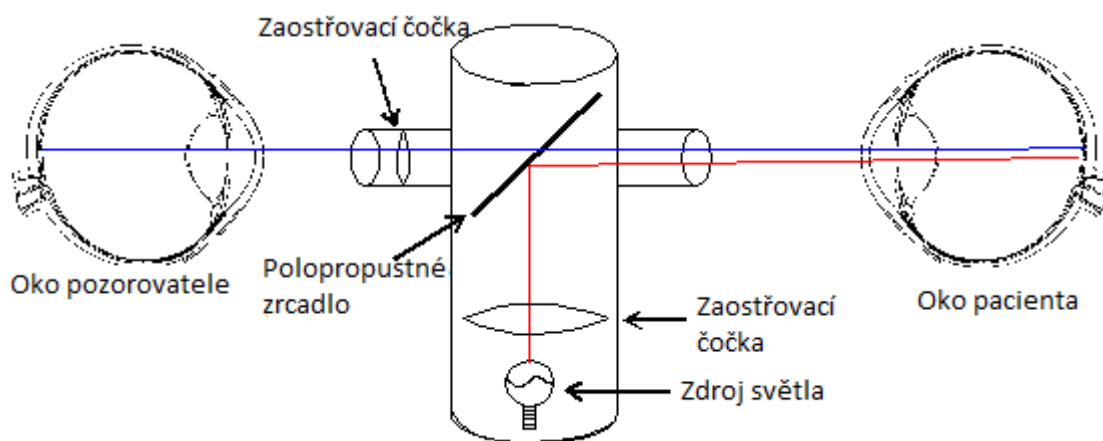
2 METODY SNÍMÁNÍ OČNÍHO POZADÍ

2.1 Oftalmoskopie

Oftalmoskopie je vyšetřovací metoda zadního segmentu oka, což znamená především vyšetření sítnice a terče zrakového nervu neboli optického disku. Tato metoda využívá osvětlení nitra oka světlem o dostatečné intenzitě a následné pozorování světelných paprsků, které se odráží od sítnice. Odražené paprsky jsou pozorovatelné jen tehdy, když je osa pozorování stejná nebo velmi blízká ose osvětlení. Pokud je daná podmínka splněna, lze v zornici pozorovat tzv. červený reflex, který vzniká prokrvením cévnatky. Oftalmoskopie se dělí na přímou a nepřímou. [7]

2.1.1 Přímá oftalmoskopie

Přímá oftalmoskopie využívá přístroj (oftalmoskop), který je přenosný a lze s jeho pomocí vyšetřit sítnici. Princip přímého oftalmoskopu je na obrázku 2. Základem je polopropustné zrcadlo, od kterého se světlo jdoucí ze zdroje odráží přímo do oka pacienta a tam dopadá na sítnici. Světlo odražené od sítnice poté putuje zpět k polopropustnému zrcadlu, které jej propouští a přes zaostřovací čočku dopadají odražené paprsky do oka pozorovatele. [7]

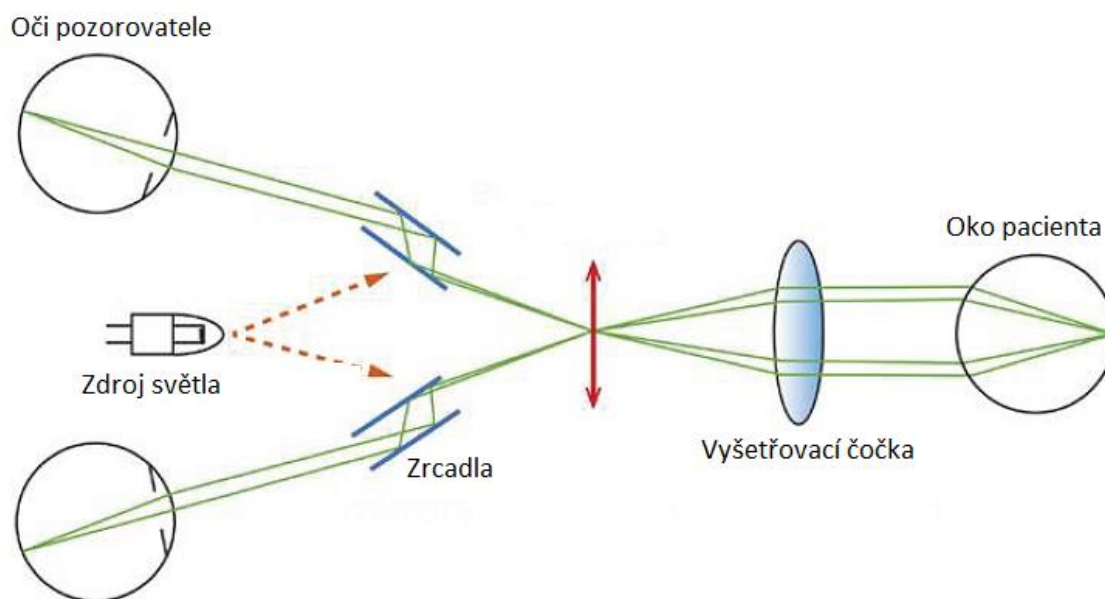


Obrázek 2: Princip přímé oftalmoskopie, převzato z [8]

Výsledný obraz sítnice je pozorovatelem sledován jedním okem z krátké vzdálenosti, která je cca 2 cm. Při daném sklonu oftalmoskopu jsme schopni pozorovat oblast sítnice velkou zhruba 2 mm. Při potřebě vyšetření různých oblastí sítnice je nutné změnit úhel náklonu oftalmoskopu. Výhodou přímého oftalmoskopu je detailní vyšetření sítnice. Nevýhodou ovšem je ztráta prostorového vjemu obrazu sítnice z důvodů pozorování jedním okem. [7]

2.1.2 Nepřímá oftalmoskopie

Nepřímá oftalmoskopie umožňuje, za použití vyšetřovací čočky, vyšetření z větší vzdálenosti a to zhruba z 50 cm. Vyšetřovací čočka je nejčastěji asférická spojka, která má lámavost 28D. Princip nepřímého oftalmoskopu je na obrázku 3. Čočka je držena vyšetřujícím před okem pacienta ve vzdálenosti, která odpovídá ohniskové vzdálenosti čočky. Ze zdroje světla vychází paprsky, které přes vyšetřující čočku osvětlují zadní segment oka a vyšetřující tak pozoruje obraz sítnice. Tento obraz se přes čočky zobrazí jako reálný a převrácený. [7]

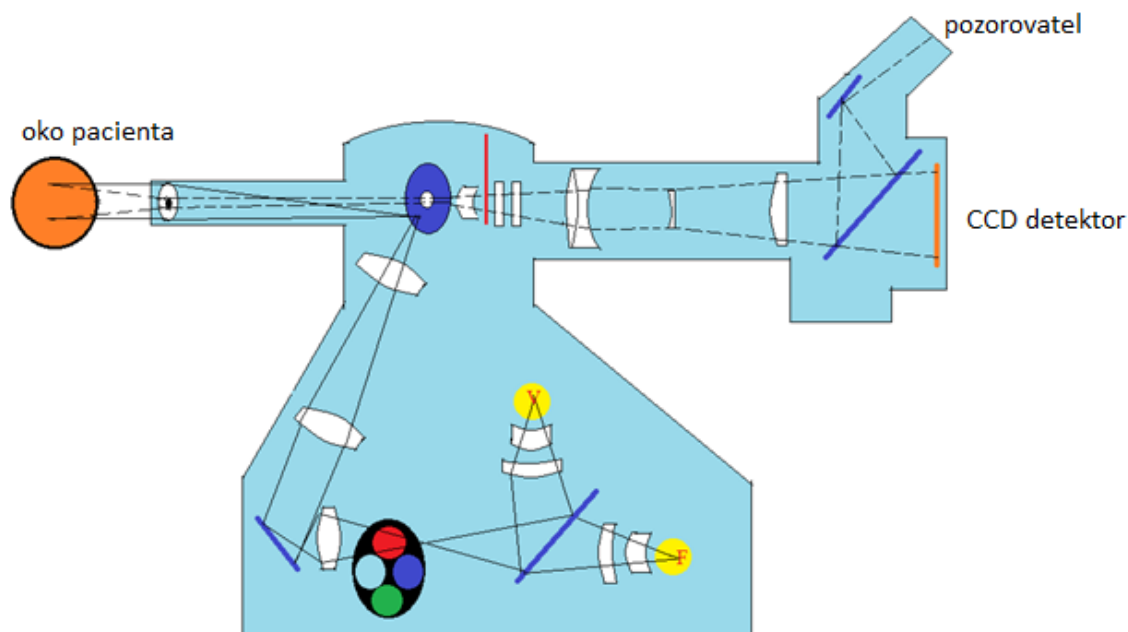


Obrázek 3: Princip nepřímé oftalmoskopie, převzato z [9]

U nepřímé oftalmoskopie dochází k menšímu zvětšení výsledného obrazu sítnice. Ovšem díky použití spojky lze pozorovat větší část sítnice, takže je vyšetření přehlednější. Další výhodou oproti přímé oftalmoskopii je prostorový vjem výsledného obrazu díky pozorování obrazu oběma očima. [7]

2.2 Fundus kamera

Fundus kamera je komplexní optický systém používaný pro zobrazování sítnice oka. Díky tomu, že osvětlovací i zobrazovací paprsky procházejí přes zornici, je možné sítnici pozorovat přímo. Tato kamera je specializovaný mikroskop s připojenou digitální kamerou, který dokáže uložit obraz sítnice ve formě digitálního obrazu. Schéma fundus kamery je na obrázku 4.



Obrázek 4: Schéma fundus kamery, převzato z [10]

Funkce kamery je založena na nepřímé oftalmoskopii. Běžný zobrazovací úhel sítnice je 30° , při kterém je obraz sítnice až 2,5x větší než reálný obraz. Širokoúhlé fundus kamery dokáží zachytit obraz od 45° do 140° , ale poskytují menší zvětšení sítnice.

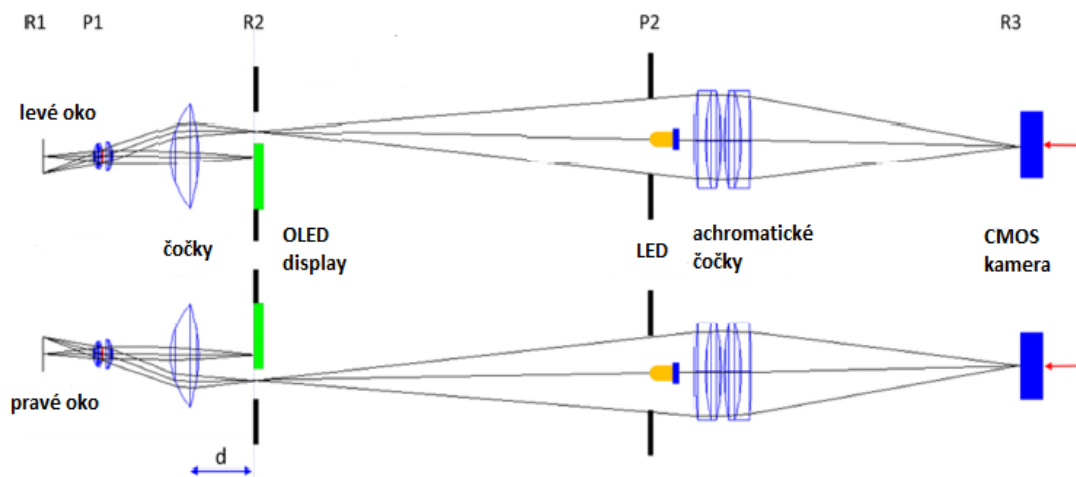
Světlo generované pomocí lampy je promítáno přes sadu filtrů na zrcadlo. Toto zrcadlo odráží světelné paprsky na sadu čoček, které tyto paprsky fokusují na polopropustné zrcadlo, ze kterého se světlo odráží a dopadá na sítnici. Poté se světlo odráží od sítnice a vrací se zpět přes polopropustné zrcadlo, soustavy čoček a filtrů do CCD detektoru a do okuláru, kterým scénu sleduje vyšetřovatel. Výsledný digitální snímek získaný pomocí fundus kamery je na obrázku 5. [10]



Obrázek 5: Snímek pořízený pomocí fundus kamery

2.3 Experimentální video oftalmoskop

Experimentální video oftalmoskop je založen na principu fundus kamery. Schéma tohoto přístroje je na obrázku 6.

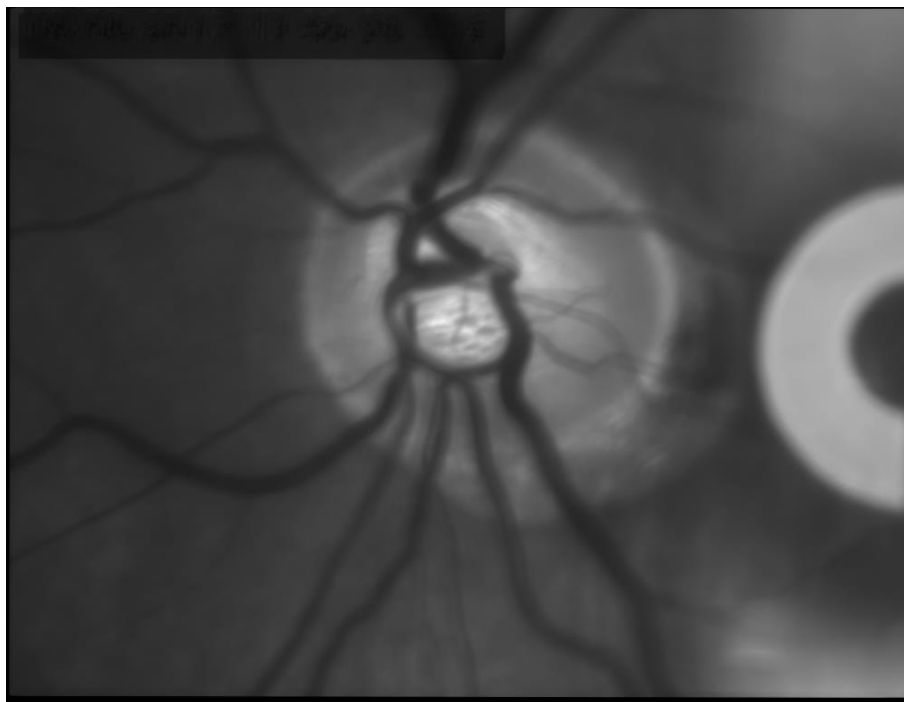


Obrázek 6: Schéma experimentálního video oftalmoskopu, převzato z [11]

Tento oftalmoskop se skládá z čočky, která vytváří obraz sítnice v obrazové rovině mezi touto čočkou a čočkou následující. Tento obraz je následně zpracován systémem dvou achromatických čoček s ohniskovou vzdáleností 12 mm do roviny senzorů CCD nebo CMOS kamery. Zorné pole tohoto oftalmoskopu je $20^\circ \times 15^\circ$ a obraz je vycentrovaný na optický disk. Hlavním rozdílem oproti fundus kameře je osvětlení, kdy pro osvětlení je využita centrální oblast zornice. Pro snímání je pak využit její okraj. Pro minimalizaci expozice světla okem a pro minimalizaci počtu rušivých odrazů je snímána pouze obdélníková část okolí optického disku. Osvětlení zajišťuje LED, která je umístěna v obrazové rovině zornice. Jako detektor obrazu se používá CCD nebo CMOS kamera. Tento přístroj umožňuje snímat obraz sítnice s frekvencí 25 snímků za sekundu. Výstupem video oftalmoskopu jsou videosekvence. Přístroj je napájen přes USB rozhraní počítače. [12] [13]

2.3.1 Vlastnosti výstupu z video oftalmoskopu

Výstupem z experimentálního video oftalmoskopu je videosekvence očního pozadí. Pro analýzu se nevyužívá videosekvence, ale jen jednoho snímku. Tento snímek je vytvořen zprůměrováním snímků z jednoho video záznamu (obrázek 7).



Obrázek 7: Zprůměrovaný snímek z experimentálního video oftalmoskopu

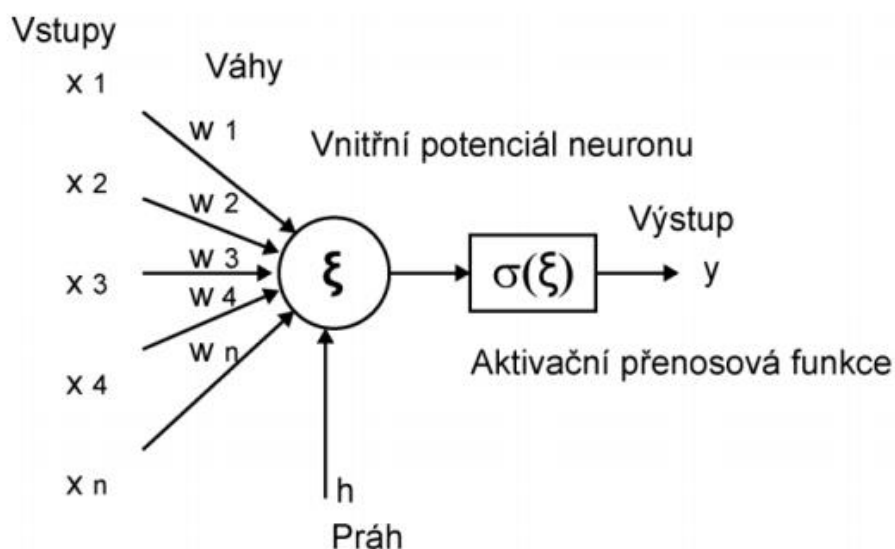
Na snímku lze vidět cévní řečiště, které je tmavší oproti pozadí a optický disk, který se vyskytuje uprostřed snímku. Snímek je šedotónový, což může vést k znesnadnění segmentace, kdy u RGB snímků se využívá jednoho nebo kombinaci dvou barevných kanálů, ve kterých má snímek nejlepší vlastnosti. U dat získaných z video oftalmoskopu možnost výběru kanálu odpadá. Dalším problémem je zašumění obrazu, které vede k neostrosti. U cévního řečiště to způsobuje rozmazání hranice mezi stěnou cév a pozadím, což vede také k znesnadnění segmentace. Vlivem šumu v obraze zanikají také některé tenké cévy, které jsou automatickým algoritmem těžko detekovatelné.

3 UMĚLÁ NEURONOVÁ SÍŤ

Umělá neuronová síť byla inspirována biologickým nervovým systémem. Je složena z velkého množství propojených procesních prvků (neuronů), které pracují společně tak, aby dospěly k řešení konkrétního problému. Umělé neuronové sítě jsou navrhovány pro specifické aplikace jako je klasifikace nebo segmentace dat. V biologických systémech učení zahrnuje úpravy synaptických vazeb mezi biologickými neurony. Učení umělé neuronové sítě funguje stejně, kdy se upravují váhy jednotlivých vstupů do umělého neuronu. [14]

3.1 Umělý neuron

Umělý neuron je matematickým modelem biologického neuronu a lze jej rozdělit na několik částí, kterými jsou vstup, váhy, potenciál neuronu, aktivační přenosová funkce a výstup. Schéma umělého neuronu je na obrázku 8.



Obrázek 8: Umělý neuron, převzato z [15]

Neuron má obecně n vstupů reprezentovaných vektorem $x = (x_1, \dots, x_n)$. Každá vstupní hodnota je ohodnocena vahou. Váhy reprezentuje vektor $w = (w_1, \dots, w_n)$. Do neuronu také vstupuje konstantní hodnota h , kterou je práh. Tento práh může být modelován jako jedna ze vstupních hodnot, kdy do vektoru vstupů přidáme hodnotu $x_0 = 1$ a do vektoru vah hodnotu $w_0 = -h$.

Vektor vstupů, vektor vah a práh vstupují do těla neuronu. Tělo neuronu je tvořeno vnitřním potenciálem neuronu ξ , který je dán vztahem:

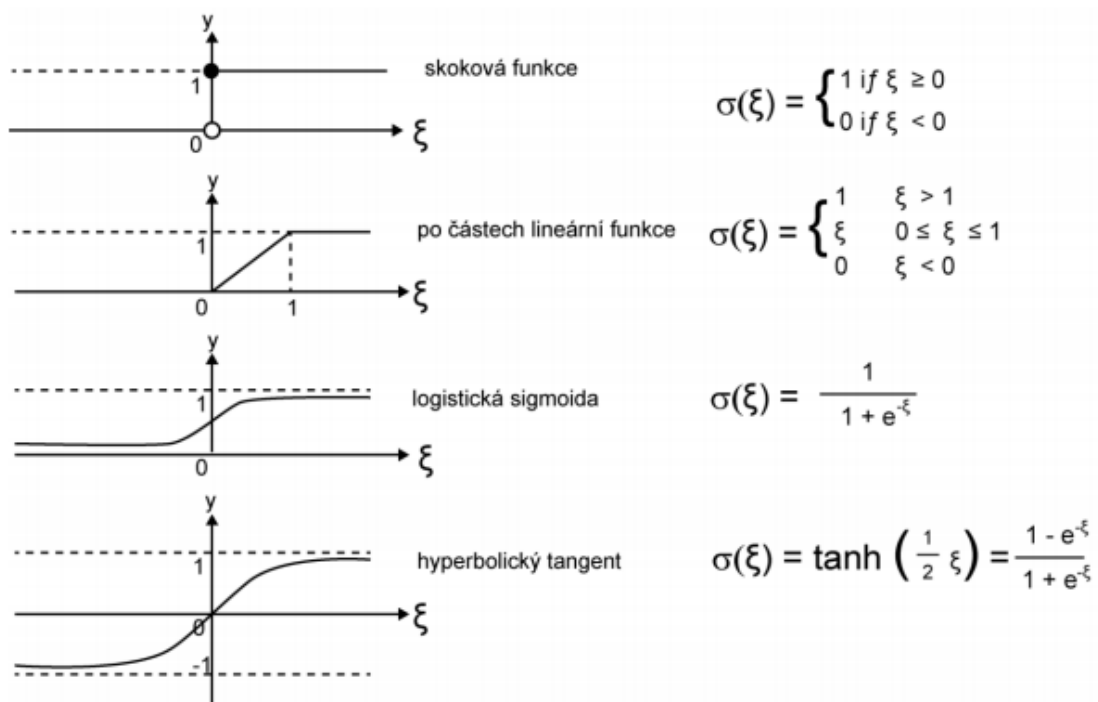
$$\xi = \sum_{i=1}^n w_i \cdot x_i - h \quad (3.1)$$

Každý prvek vstupního vektoru x_i je násoben jeho příslušnou vahou w_i a následně je vypočítána suma všech těchto součinů. Od této sumy je poté odečten práh h . Výsledná hodnota vnitřního potenciálu je argumentem obecně nelineární aktivační přenosové funkce σ . Výstup z neuronu y je pak dán vztahem:

$$y = \sigma(\xi) = \sigma\left(\sum_{i=0}^n w_i \cdot x_i\right), \quad (3.2)$$

kde hodnota prahu je součástí vstupních hodnot.

Druhů aktivačních přenosových funkcí existuje velké množství. Příklady nejčastěji používaných funkcí pro dopředné neuronové sítě jsou na obrázku 9. [15]

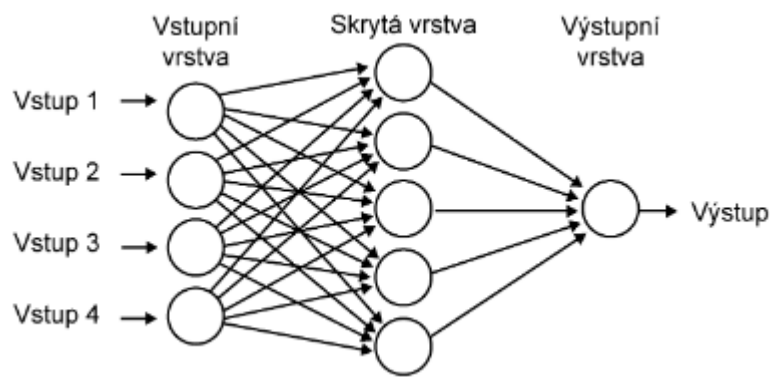


Obrázek 9: Příklady aktivačních funkcí, převzato z [15]

3.2 Topologie sítě

Topologie neuronové sítě se týká způsobu propojení neuronů a je důležitým faktorem pro fungování a učení sítě.

Nejčastější topologií je plně propojená třívrstvá dopředná síť (obrázek 10). U tohoto druhu sítě jsou všechny vstupní hodnoty do sítě připojeny ke všem neuronům ve skryté vrstvě (skrytá vrstva, protože není viditelná na vstupu ani na výstupu). Výstupy skrytých neuronů jsou následně propojeny se všemi neurony ve výstupní vrstvě a aktivace výstupních neuronů tvoří výstup celé sítě. Výhodou této topologie je robustnost a odolnost vůči poškození, kdy často síť dokáže poskytovat správné výstupy i když jsou některé elementy sítě poškozeny. [15] [16]



Obrázek 10: Topologie dopředné neuronové sítě, převzato z [15]

3.3 Učení sítě

Procesem učení neuronových sítí je myšlena úprava parametrů neuronu tak, aby prováděl požadovanou transformaci. Parametry, které jsou v průběhu upravovány, jsou zpravidla váhy vstupních dat a práh. Učení neuronových sítí se dělí dvou základních kategorií, kterými jsou učení s učitelem a bez učitele. [15]

Kontrolované učení neboli **učení s učitelem** zahrnuje externího učitele. Během tréninku je vstupní vektor prezentován do sítě, která následně produkuje výstupní vektor. Tento výstupní vektor je porovnán s požadovaným výstupním vektorem a generuje se chybový signál. Na základě tohoto chybového signálu jsou upravovány váhy tak, aby skutečný výstup odpovídal požadovanému výstupu.

Nekontrolované učení neboli **učení bez učitele** nezahrnuje externího učitele. Při tréninku v rámci nekontrolovaného učení jsou vstupní vektory podobného typu kombinovány a vytvářejí klastry. Po vstupu nového vstupního vektoru poskytuje neuronová síť výstupní odezvu, která určuje třídu, do které vstupní vektor patří. Při učení bez učitele není k dispozici požadovaný výstup, proto samotná síť musí objevit vzory nebo rysy ze vstupních dat a vztah vstupních hodnot s výstupními. [14] [17]

„**Backpropagation**“ je nejčastější metoda učení neuronových sítí, které se říká algoritmus zpětného šíření chyby. Je to metoda učení s učitelem dopředných sítí, kdy se řešení porovnává s očekávaným výstupem a zjišťuje jak moc se neuronová síť „spletla“. Poté se zpětně upravují hodnoty vah, tak aby odchylka od správného řešení byla co nejmenší. [18]

3.4 Chybové funkce

Chybová funkce se používá k optimalizaci hodnot parametrů v modelu neuronové sítě. Tato funkce mapuje sadu hodnot parametrů sítě na skalární hodnotu. Ta udává, jak dobře tyto parametry dosáhnou úlohy, kterou má síť provádět. Existuje několik chybových funkcí, které často měří čtvercovou nebo absolutní chybu mezi skutečným výstupem sítě a požadovaným výstupem. Dále budou uvedeny jen příklady některých chybových funkcí. [19]

Křížová entropie (Cross entropy)

Křížová entropie se běžně používá v binární klasifikaci, kdy se předpokládá, že masky obsahují pouze hodnoty 0 a 1. Tato chybová funkce se počítá pomocí vzorce:

$$L = -\frac{1}{n} \cdot \sum_{i=1}^n [y^{(i)} \cdot \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \cdot \log(1 - \hat{y}^{(i)})], \quad (3.3)$$

kde y je požadovaná výstupní hodnota a $\hat{y}$ je skutečná výstupní hodnota. Křížová entropie měří rozdíly mezi dvěma distribuce pravděpodobnosti. Pokud je křížová entropie velká znamená to, že rozdíl mezi dvěma distribucemi je velký. Naopak když je křížová entropie malá jsou si pravděpodobnostní distribuce navzájem podobné.

Střední kvadratický chyba (MSE)

Střední kvadratická chyba se využívá v rámci lineární regrese. Základním principem je minimalizovat kvadratickou sumu rozdílu požadované výstupní hodnoty a skutečné výstupní hodnoty. Výpočet je realizován pomocí vzorce:

$$L = \frac{1}{n} \cdot \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2 \quad (3.4)$$

Pokud použijeme jako aktivační funkci sigmoidu, dojde k pomalé rychlosti učení. Pro jiné aktivační funkce tento problém nenastává.

Odchylka Kullback Leibler (KL)

Tato odchylka je známá také jako relativní entropie, kdy zisk nebo odchylka je měřítkem podobnosti pravděpodobnostního rozdělení skutečného výstupu s pravděpodobnostním rozdělením očekávaného výstupu. KL odchylka se vypočítá podle vzorce:

$$L = \frac{1}{n} \cdot \sum_{i=1}^n (y^{(i)} \cdot \log(y^{(i)})) - \frac{1}{n} \cdot \sum_{i=1}^n (y^{(i)} \cdot \log(\hat{y}^{(i)})), \quad (3.5)$$

kde první část je entropie a druhá část je křížová entropie. V nejjednodušším případě je KL odchylka rovna 0. To znamená, že můžeme očekávat podobné nebo stejné chování dvou rozdílných rozdělení. Zatímco pokud je KL odchylka rovna 1, znamená to pravý opak. [19]

3.5 Optimalizační algoritmy

Optimalizační algoritmy jsou používány pro aktualizaci parametrů sítě, která vede ke snížení chybové funkce. Optimalizační algoritmy rozdělujeme do dvou základních kategorií:

- Algoritmy optimalizace prvního řádu
Tyto algoritmy minimalizují nebo maximalizují chybovou funkci pomocí gradientu, nejpoužívanější algoritmus prvního řádu je Gradient Descent
- Algoritmy optimalizace druhého řádu
Tyto metody používají derivát druhého řádu, který se nazývá Hessián, aby minimalizoval nebo maximalizoval chybovou funkci. [20] [21]

Gradient Descent

Gradient Descent je nejpoblárnější optimalizační algoritmus používaný při optimalizaci neuronové sítě. Tato metoda počítá gradient pro celou datovou množinu a aktualizuje hodnoty vah v opačném směru ke gradientu, dokud nenajde místní minima. Jednoduše řečeno metoda najde minima, zkontroluje odchylku a následně aktualizuje parametry modelu, které vedou ke konvergenci. Aktualizace parametrů modelu probíhá podle následujícího vzorce:

$$\theta = \theta - \eta \cdot \nabla J(\theta), \quad (3.6)$$

kde η je rychlost učení, $\Delta J(\theta)$ je gradient chybové funkce. Výpočet této metody je pomalý a výpočetně náročný. To je způsobeno pouze jednou aktualizací vah v rámci celé množiny dat. [20] [21]

Stochastic Gradient Descent (SGD)

Metoda Stochastic Gradient Descent se liší od předchozí metody v tom, že provádí aktualizaci parametrů pro každý příklad tréninku na rozdíl od Gradient Descent, který provádí pouze jednu aktualizaci. Z toho vyplývá, že je mnohem rychlejší. Vzorec pro výpočet této metody je:

$$\theta = \theta - \eta \cdot \nabla J(\theta; x(i); y(i)), \quad (3.7)$$

kde $\{x(i), y(i)\}$ jsou trénovací příklady. Z důvodů těchto častých aktualizací parametrů mají aktualizace velký rozptyl, což způsobuje kolísání chybové funkce na různé hodnoty. Výhodou toho je, že může nalézt nová a lepší místní minima. Nevýhodou jsou časté výkyvy kvůli častým aktualizacím, což vede ke zpomalení konvergence. [21]

Momentum

Tato metoda byla navržena k urychlení metody SGD, kdy kolísání rozptylu znesnadňuje dosažení konvergence. Urychlení je realizováno přidáním momentu hybnosti γ do aktualizacího vektoru z předchozího kroku do aktuálního aktualizacího vektoru:

$$V(t) = \gamma V(t-1) + \eta \nabla J(\theta) \quad (3.8)$$

Hodnota momentu hybnosti se obvykle nastavuje na hodnotu 0,9 nebo na hodnotu blízkou této hodnotě. Přidáním momentu hybnosti docílíme rychlejší a stabilnější konvergence. Výsledný výpočet aktualizace parametrů modelu je pak dán vzorcem [21]:

$$\theta = \theta - V(t) \quad (3.9)$$

Adam

Adam je metoda adaptivního učení, což znamená, že počítá individuální míru učení pro různé parametry. Tato metoda používá odhady prvního a druhého momentu gradientu tak, aby se přizpůsobila rychlost učení pro každou váhu neuronové sítě.

K odhadnutí těchto momentů využívá metoda Adam exponenciálně klesající průměr předcházejících čtvercových gradientů. Samotné odhady prvního a druhého momentu se vypočítají jako:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad (3.10)$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}, \quad (3.11)$$

kde $\hat{m}_t$ je odhad prvního momentu, m_t je první moment, $\hat{v}_t$ je odhad druhého momentu, v_t je druhý moment a β_1^t a β_2^t jsou hyperparametry, které řídí exponenciální rychlost poklesu v odhadu momentu. Pro výpočet aktualizace parametrů slouží vzorec [21]:

$$\theta_t = \theta_{t-1} - \eta \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (3.12)$$

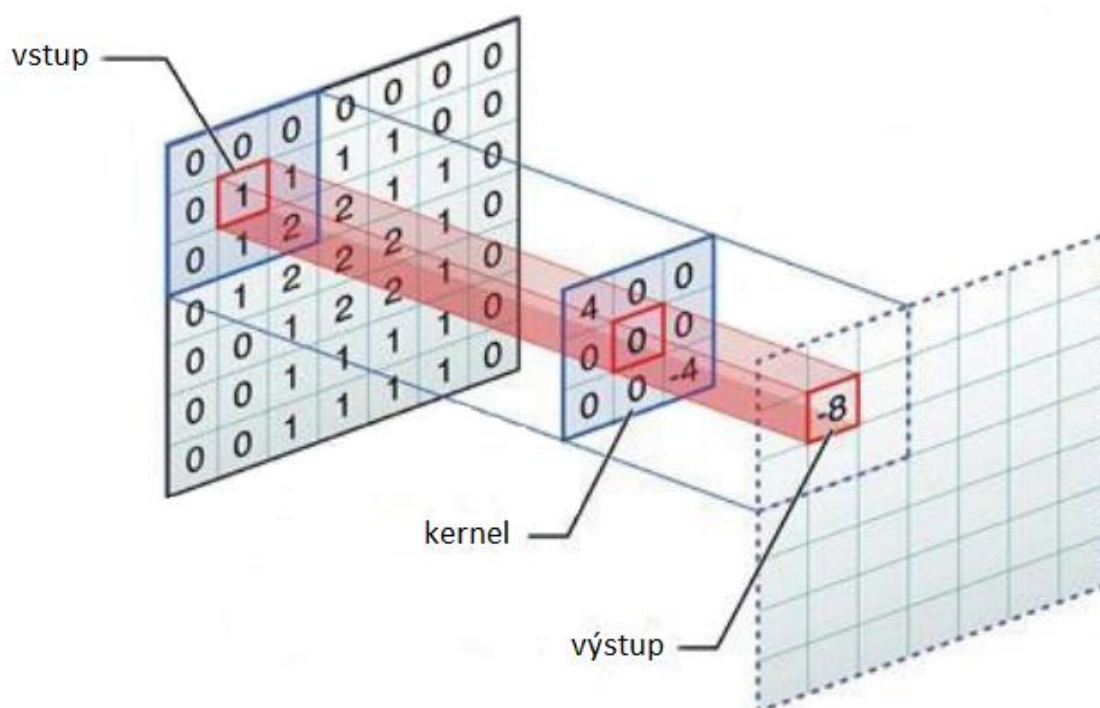
4 KONVOLUČNÍ NEURONOVÉ SÍTĚ

Konvoluční neuronové sítě (CNN) jsou velmi podobné běžným neuronovým sítím, které se ukázaly jako velmi účinné v oblastech rozpoznávání nebo klasifikace snímků. Skládají se z neuronů, které mají naučené váhy. Konvoluční neuronové sítě předpokládají, že vstupy jsou obrazy, které umožní kódovat určité vlastnosti do architektury sítě.

4.1 Vrstvy konvolučních neuronových sítí

Konvoluční neuronové sítě se skládají z několika vrstev, kde každá vrstva může být jiného typu. Existují tři typy vrstev, kterými jsou konvoluční, podvzorkovací a plně propojená vrstva. [22] [23]

Konvoluční vrstva je hlavní prvkem konvolučních neuronových sítí a její primárním účelem je získávání vlastností ze vstupního obrazu. Tato vrstva využívá konvoluční filtr (kernel), který detekuje v obraze struktury, jako jsou například hrany. Filtrem se rozumí matice hodnot, jejíž hodnoty jsou vycvičeny na detekci specifických vlastností. Filtre se posunuje po obraze a hledá struktury, které mají být detekovány a provádí konvoluci s vybranou částí obrazu. Princip konvoluce je na obrázku 11. [22] [24]



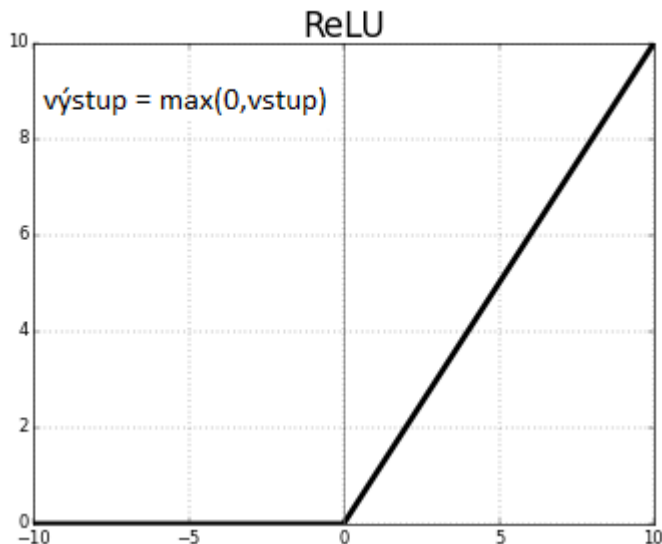
Obrázek 11: Princip konvoluce vstupního obrazu s kernel filtrem, převzato z [24]

Nejčastěji se používá konvoluční filtr o velikosti 3x3, kdy celkový počet prvků filtru je 9. Filtr se pohybuje po obraze a provádí konvoluci s hodnotami vstupního obrazu pod maskou. Jedná se o násobení hodnot filtru s příslušnými hodnotami pod maskou a následný výpočet sumy těchto součinů.

Velikost výstupu z konvoluční vrstvy určují tři parametry, kterým se říká hyperparametry. Prvním hyperparametrem je počet konvolučních filtrů, z nichž každý hledá jiné vlastnosti na vstupu. Druhým hyperparametrem je krok, který definuje posunutí filtru po vstupní obraze. Pokud je krok 1, prochází filtr vstupní obraz pixel po pixelu. Pokud je krok nastaven na hodnotu 2, tak se filtr posune o 2 pixely. To je důvodem vzniku menších výstupních obrazů. Posledním hyperparametrem je přidávání nul („zero-padding“) do vstupní matice obrazu. Tímto se docílí stejné velikosti výstupu jako vstupu, čehož se využívá při navrhování CNN, kde je potřeba zachovat velikost vstupu na výstupu. [23]

Aby se CNN mohla naučit správné hodnoty konvolučního filtru, musí filtr projít nelineárním mapováním, což zajišťuje aktivační funkce. [24]

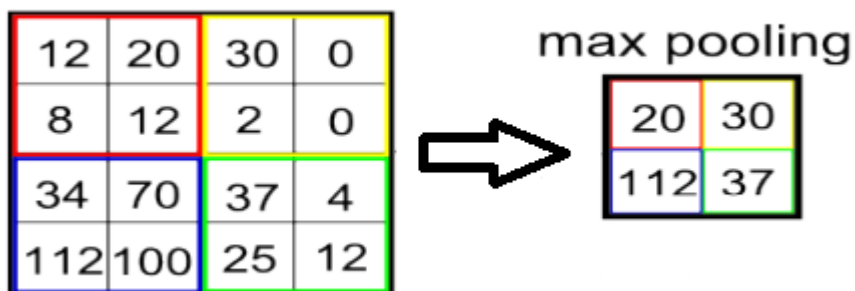
Aktivační funkcí u konvolučních neuronových sítí je ReLU (retifikovaná lineární jednotka), což je nelineární operace. Účelem ReLU je zavést do sítě nelinearitu, protože většina dat v reálném světě jsou nelineární. ReLU je operace, která nahrazuje všechny záporné hodnoty v obraze nulou (obrázek 12). [25]



Obrázek 12: Aktivační funkce ReLU, převzato z [25]

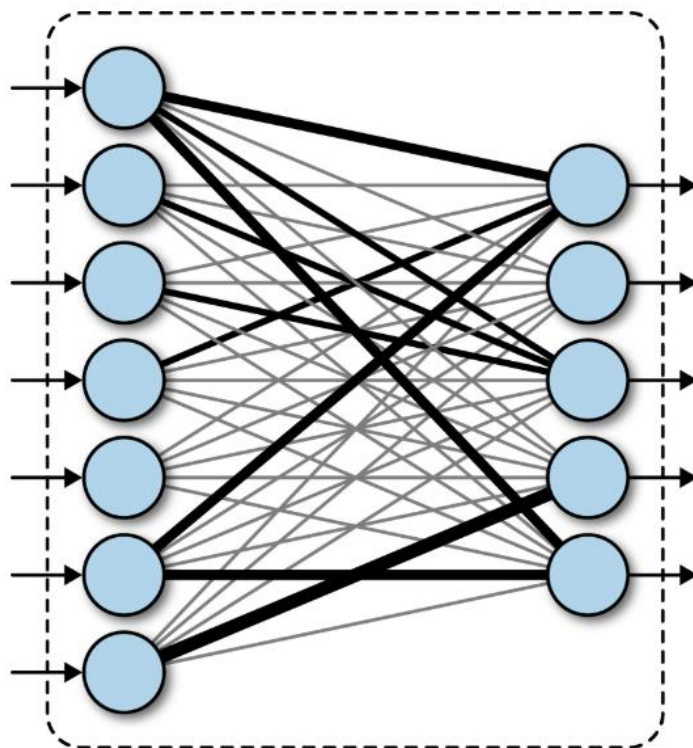
Podvzorkovací vrstva může následovat po každé konvoluční vrstvě a slouží k postupnému snižování prostorové velikosti vstupního obrazu, aby se snížil počet parametrů a výpočtů v síti. Tato vrstva přebírá malé obdélníkové bloky z konvoluční vrstvy. Z toho to bloku následně vytvoří jeden výstup, který může být vypočítán například jako průměrná nebo maximální hodnota vybraného obdélníkové bloku.

Nejčastěji se používá maximální hodnota daného bloku („max-pooling“). Nejběžnější velikost filtrů ve sdružovací vrstvě je 2x2. Princip sdružování je na obrázku 13, kdy v okně o velikosti prvků 4 je vybráno maximum, které je uloženo do výstupu. Výstupem je pak obraz o menší velikosti než obraz, který do této vrstvy vstupoval. [23]



Obrázek 13: „Max-pooling“, převzato z [25]

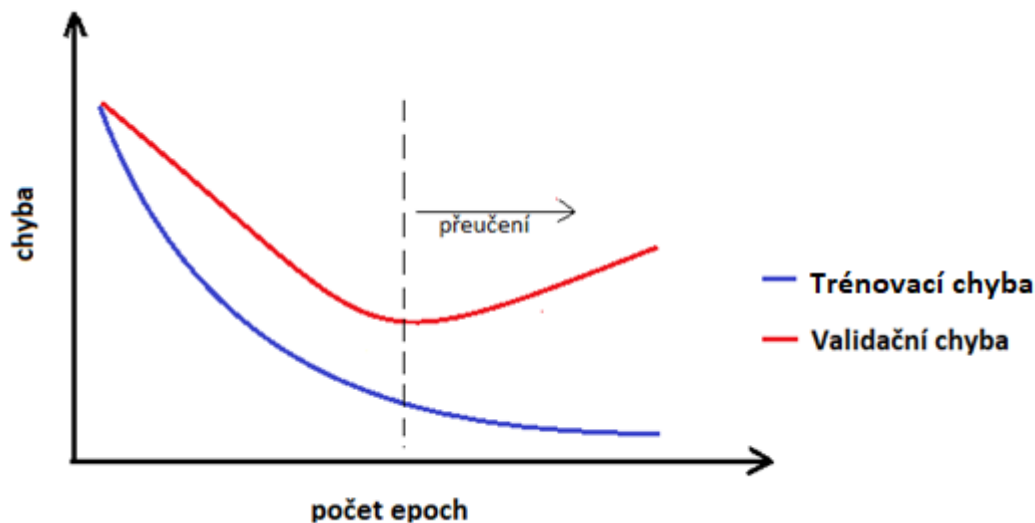
Plně propojená vrstva je poslední vrstvou konvoluční neuronové sítě. Jejím úkolem je převést vstupní reprezentaci obrazu do vektoru vlastností, které předpovídají pravděpodobnosti výstupu. Každý vstup je připojen ke všem výstupům, kdy každé spojení má svoji váhu, která určuje důležitost jednotlivých vstupních neuronů. Ukázka plně propojené vrstvy je na obrázku 14, kde silnější čáry značí vyšší důležitost než tenké čáry. [14] [27]



Obrázek 14: Plně propojená vrstva, převzato z [28]

4.2 Přeučení neuronové sítě

Hluboké neuronové sítě s velkým počtem parametrů jsou velmi výkonné systémy strojového učení. Díky tomu, že využívají nelineární „skryté“ vrstvy, dokáží se naučit velmi složité vztahy mezi vstupy a výstupy sítě. Jejich problémem je však přetrénování sítě, označováno jako „overfitting“. Po dokončení trénovací epochy obecně platí, že je natrénovaná síť ověřena na validační sadě dat. Přetrénování nastane v momentě, kdy trénovací úspěšnost stále roste, ale validační úspěšnost se snižuje. Neboli trénovací chyba se snižuje, ale validační chyba roste (obrázek 15). [29]



Obrázek 15: Přetrénování sítě

Existují metody, které se snaží přeučení sítě předejít. Dále budou popsány metody, které jsou použity v praktické části této práce.

4.2.1 Augmentace dat

Augmentace dat je nejběžnější a nejjednodušší způsob, jak předejít přeučení neuronové sítě. Augmentace spočívá v umělém zvětšení sady dat pomocí různých geometrických transformací obrazu. K těmto transformacím patří například rotace a posunutí obrazu. Dále je pak možné přidání šumu do obrazu. Augmentace dat se využívá především tehdy, když je k dispozici malá sada trénovacích snímků. [30]

4.2.2 Dropout

Dropout znamená vyřazení skrytých či viditelných jednotek v neuronové síti. Vyřazením jednotky je myšleno dočasné odstranění jednotky ze sítě, spolu se všemi vstupními i výstupními spojeními. Volba jednotky, která má být vyřazena je náhodná. V nejjednodušším případě je každá jednotka zachována s pravděpodobností p , která není závislá na jiných jednotkách. Pravděpodobnost p lze nastavit podle validační sady nebo jednoduše nastavit na hodnotu 0,5, což je optimální hodnota pro celou řadu úkolů. Pro vstupní jednotky se však hodnota pravděpodobnosti obvykle volí blíže hodnotě 1. [29]

4.2.3 Early stopping

Metoda „Early stopping” neboli včasné zastavení je široce používána, protože je velice jednoduchá. Tato metoda funguje na principu porovnávání validační chyby během učení. Po každé trénovací epoše, při které se provádí validace dat, je určena validační chyba. Pokud je tato chyba větší, než byla poslední určená validační chyba, je trénování sítě zastaveno.

Problémem ale je, že při trénování reálné neuronové sítě validační chyba nemusí při stoupajícím počtu epoch jenom klesat, ale při jedné může narůst a při další zase klesnout. Neexistuje žádné pravidlo, které by určovalo, kdy přesně má být učení zastaveno. [31]

5 SEGMENTACE OBRAZU

Segmentace obrazu je základní metodou v analýze obrazu, která vymezuje oblasti obrazu představující různé objekty, jako jsou například různé typy tkání, orgány, kosti aj. Formálně se jedná o rozdělení celé oblasti vstupního obrazu R do množiny dílčích oblastí $\{R_1, R_2, \dots, R_s\}$, které se nepřekrývají. Tyto oblasti pokrývají celou plochu vstupního obrazu. Z toho vyplývá, že výstupem segmentace je obraz s odlišenými oblastmi. Odlišení bývá realizováno například barvou. Segmentace je většinou postupně iterativní proces pro zlepšení výsledné segmentace tak, aby odpovídala co nejlepší interpretaci obrazu. Existuje mnoho různých metod segmentace obrazu, z nich ale nelze jednoznačně určit tu nejlepší. Segmentaci obrazu lze rozdělit do několika tříd [32]:

- Segmentace podle homogenity oblastí
- Regionově-orientovaná segmentace
- Segmentace metodou rozvodí
- Hranově orientovaná segmentace
- Pružné a aktivní kontury
- Jiné přístupy

5.1 Metody segmentace cévního řečiště v retinálních snímcích

Segmentací cévního řečiště se věnuje mnoho vědeckých článků a bylo také vymyšleno mnoho způsobů, jak tuto segmentaci cévního řečiště realizovat. Cílem této kapitoly není popsat všechny existující metody, ale nastínit alespoň některé z těchto metod. Budou zde popsány nejnovější metody, z nichž většina je založena na prahování a na hodně rozvíjejících se konvolučních neuronových sítích a hlubokém učení.

Akyas T. U. Ahamed a spol. [33] navrhli segmentaci cévního řečiště pomocí hysterezního prahování. Nejprve provedli předzpracování obrazu, kdy z RGB obrazu vybrali pouze kanál pro zelenou barvu, kde je nejlepší kontrast mezi cévami a pozadím. Dále upravili kontrast obrazu pomocí adaptivní ekvalizace histogramu (CLAHE) a spojili s metodou detekce vícenásobné linky, která detekuje hranice cév. Pro samotnou segmentaci použili metodu hysterezního prahování, která využívá dvě operace. Dvěma operacemi je myšleno prahování intenzity obrazu pro dva rozsahy šedých hodnot, z nichž jeden je zahrnut do druhého. Silný práh odděluje s velkou spolehlivostí cévy od okolí a slabý práh odděluje pixely pozadí od okolních struktur. Výsledná segmentace je následně vylepšena morfologickými operacemi,

kterými jsou vyplnění mezer mezi pixely v nalezených cévách a odstranění izolovaných pixelů, které byly nesprávně určeny jako cévy. Přesnost této metody je 95,92 %.

Metodu segmentace cévního řečiště pomocí Otsu prahování s analýzou hlavních komponent navrhli Jyotiprava Dash a Nilamani Bhoi [34]. Jedná se o třístupňovou prahovací metodu, kdy se v prvním kroku využívá kombinace metody hlavních komponent (PCA) a adaptivní ekvalizace histogramu. Kombinace těchto metod se používá pro zkvalitnění obrazu. Následně se ve druhém kroku segmentují cévy pomocí globálního Otsu prahování. V poslední kroku je pak aplikováno morfologické čištění k odstranění nedokonalostí segmentace. Tato metoda se vyznačuje jednoduchou implementací a výpočetní nenáročností. Dosahuje průměrné přesnosti 95,6 % pro veřejně dostupnou databázi DRIVE.

Další metodu navrhli P.T. Karule a Shilpa Joshi [35], kdy pro segmentaci cévního řečiště použili místní prahování. Jak u všech předchozích metod provedli před samotnou segmentací předzpracování obrazu. Součástí předzpracování obrazu je normalizace barev, úprava kontrastu, zvýraznění cév a odstranění optického disku. Před samotným prahováním použili konvoluci dvou lineárních filtrů, která zlepšují citlivost a specifičnost segmentačního algoritmu. Vlastní segmentace je založena na místním prahování. Kdy se prochází obraz pixel po pixelu a určuje se, zda daný pixel patří k cévě nebo k pozadí. Předpokládá se, že pozadí má nižší úroveň šedé než cévy. Výstupem je binární obraz, kde 0 reprezentuje pozadí a 1 cévy. Tento algoritmus dosahuje průměrné přesnosti 96 %.

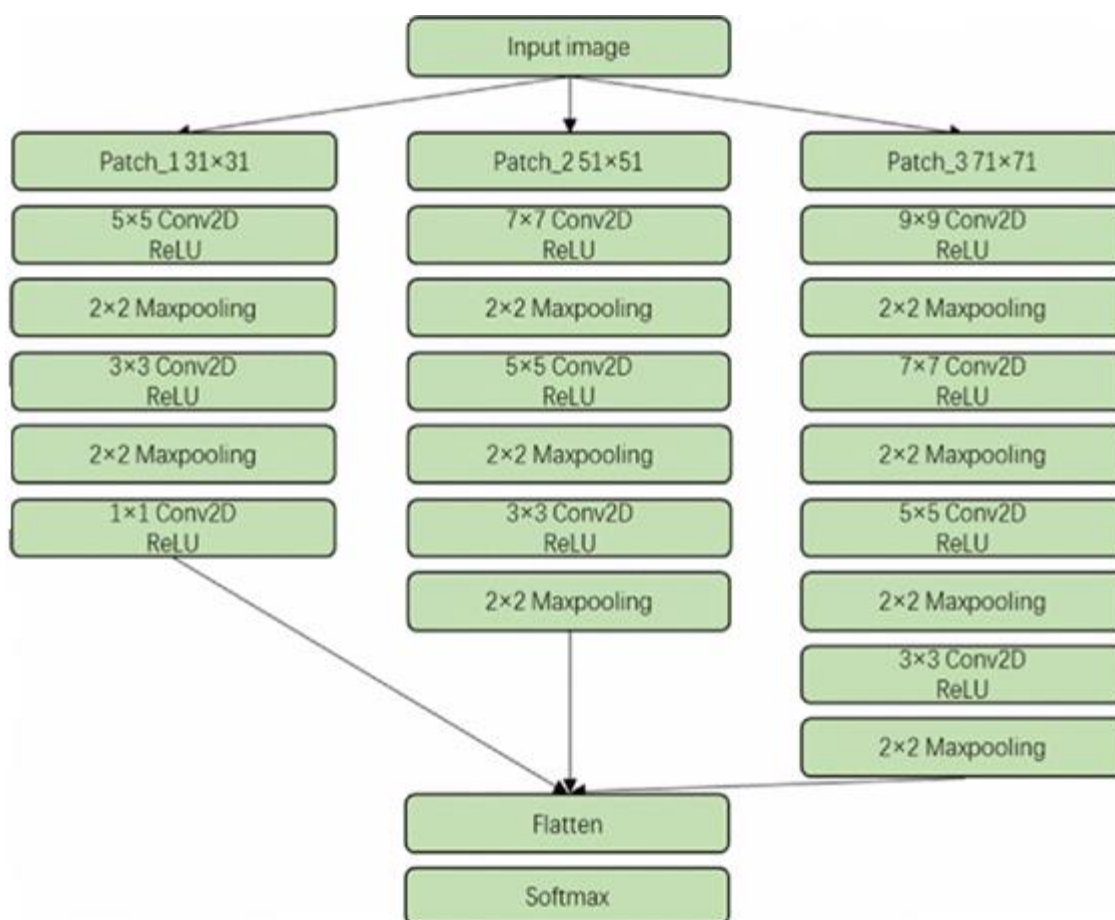
Toufique A. Soomro a spol. [36] využili pro segmentaci cévního řečiště metodu založenou na základních filtracích. Tato metoda je rozdělena do tří částí, kdy v první části dochází k úpravě vstupního obrazu. Úprava obrazu spočívá ve vybrání zeleného kanálu a převedením do šedotónového obrazu. V druhé části se pomocí morfologické filtrace odstraňuje nerovnoměrné osvětlení obrazu a pomocí adaptivního Wienerova filtru odstraňuje šum. V poslední třetí části dochází k samotné detekci cév. Je aplikován Lapacián gausiánu druhého řádu. Takto jsou detekovány silné cévy, problém ale nastává u tenkých cév, které jsou těžko detekovatelné. Pro přesnější segmentaci se využívá orientovaného difúzního filtru. Výsledný obraz je binarizován pomocí metody narůstání oblastí se dvěma prahy. Přesnost tohoto algoritmu je 95,15 % při testování na databázi DRIVE.

Filtraci využili také Hanung Adi Nugroho a spol. [37], kteří pro segmentaci cévního řečiště použili Frangi filtr a morfologické operace. Algoritmus je rozdělen do dvou oddělených částí. V první části se aplikuje Frangi filtr pro segmentaci silných cév. Tento filtr využívá vlastních čísel Hessianovy matice a detekuje spojitě okraje cév. Po aplikaci toho filtru jsou použity morfologické operace a výstupem je obraz se segmentovanými silnými cévami. První část algoritmu segmentuje cévy jasněji, ale tenké cévy nejsou detekovány, proto byla vytvořena druhá část, kde tenké cévy detekovány jsou. Obraz je v druhé části předzpracován pomocí výběru zeleného kanálu, zvýraznění osvětlení cév, adaptivní ekvalizace histogramu a mediánového filtru. Poté jsou aplikovány morfologické operace a výstupem této části je obraz se segmentovanými cévami. Výstupem celého algoritmu je kombinace výstupů z obou částí. Tato metoda dosahuje přesnosti 94,50 % na databázi DRIVE.

Avijit Dasgupta a Sonam Singh [38] použili pro segmentaci cévního řečiště konvoluční neuronovou síť. Vstupní data do neuronové sítě byli nejdříve předzpracováni, kdy byl vybrán zelený kanál, normalizována data, upraven kontrast pomocí adaptivní ekvalizace histogramu a gamma korekce. Takto upravený obraz vstupuje do neuronové sítě. Navržená architektura sítě má osm vrstev. První dvě vrstvy jsou konvoluční s velikostí filtru 32x32. Třetí vrstva je podvzorkovací vrstva s oknem 2x2, kdy velikost vstupního obrazu, je zmenšena na polovinu. Dále následují dvě konvoluční vrstvy s filtrem o velikosti 64x64. Šestá vrstva je vzorkovací, kdy je vstupní obraz zvětšen na původní velikost. Poslední dvě vrstvy jsou konvoluční s filtrem o velikosti 32x32. Aktivační funkcí je v celém modelu lineární jednotka ReLU, kromě poslední vrstvy, kde je aktivační funkcí „softmax“. Tato metoda dosahuje přesnosti 95,60 %.

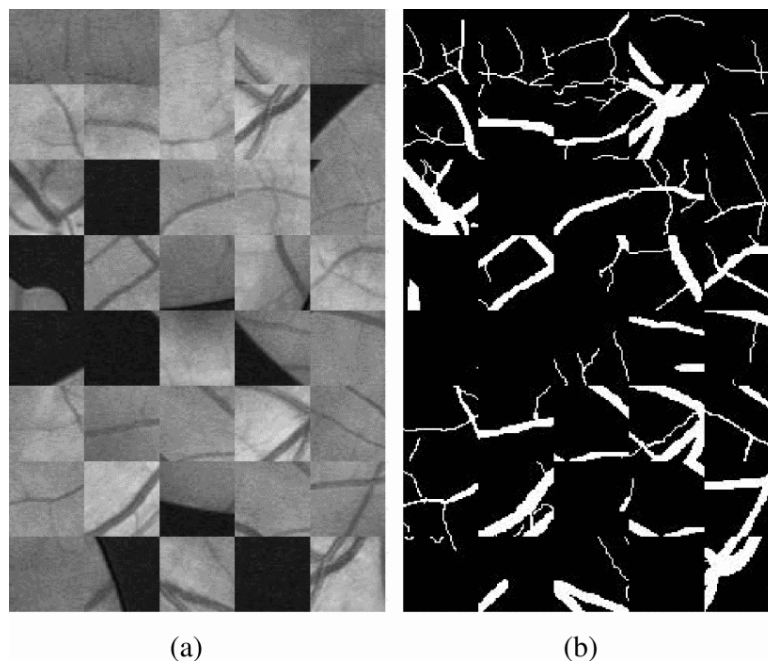
Další metoda je také založena na CNN a vymysleli ji Jinnan Guo a spol. [39]. Navrhli víceúrovňovou CNN, která má tři vstupy. Předzpracovaný vstupní obraz je rozdělen do tří výřezů (patch), kde středový pixel ve všech výřezích je bodem, který se klasifikuje. Menší výřez podává informaci o lokálních a podrobných informacích o daném pixelu (např. vztah mezi sousedními pixely). Větší výřezy poskytují více globálních informací o daném pixelu. Pro každý výřez je vytvořena vlastní větev a zároveň je síť trénována na každém výřezu odděleně. Proto se také větve výřezů liší, jak je možné vidět na obr 16. Tyto tři větve jsou nakonec spojeny do jedné větve a následuje aktivační funkce softmax, díky které se získá výsledná klasifikace každého pixelu. Použitou optimalizační funkcí je SGD s počáteční rychlostí učení 0,001 a poté sníženou o faktor 10 po každých 1000 iterací. V této metodě je taky použit „dropout“ pro předejití přeučení sítě s hodnotou $p = 0,3$. Výstup sítě je následně naprahován,

jsou vymazány izolované pixely a spojeny rozpojené cévy. Metoda byla testována na databázi DRIVE s výslednou přesností 96,2 %.



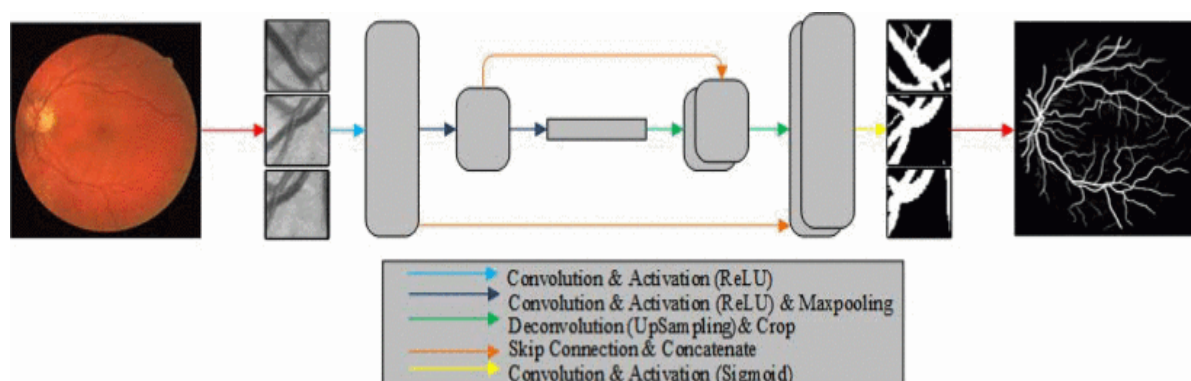
Obrázek 16: Architektura sítě navržená Jinnanem Guo a spol. [39]

Zhongwei Feng a spol. [40] použili k segmentaci cévního řečiště také CNN, jejíž trénování je založeno také na výřezech vstupního obrazu. Samotnému trénování sítě předchází předzpracování, jako je převedení RGB obrazů na šedotónové, normalizace vstupních obrazů, odstranění šumu pomocí mediánového filtru a upravením kontrastu. Poslední krok předzpracování spočívá v normalizaci hodnot pixelů obrazu na rozsah (0,1). Protože datová sada, kterou použili pro trénování a testování sítě obsahuje 40 snímků, využili metodu lokální entropie vzorkování. Tímto získali výřezy původních snímků (obrázek 17).



Obrázek 17: Výřezy vstupního obrazu (a) a jejich „ground truth“ (b) [40].

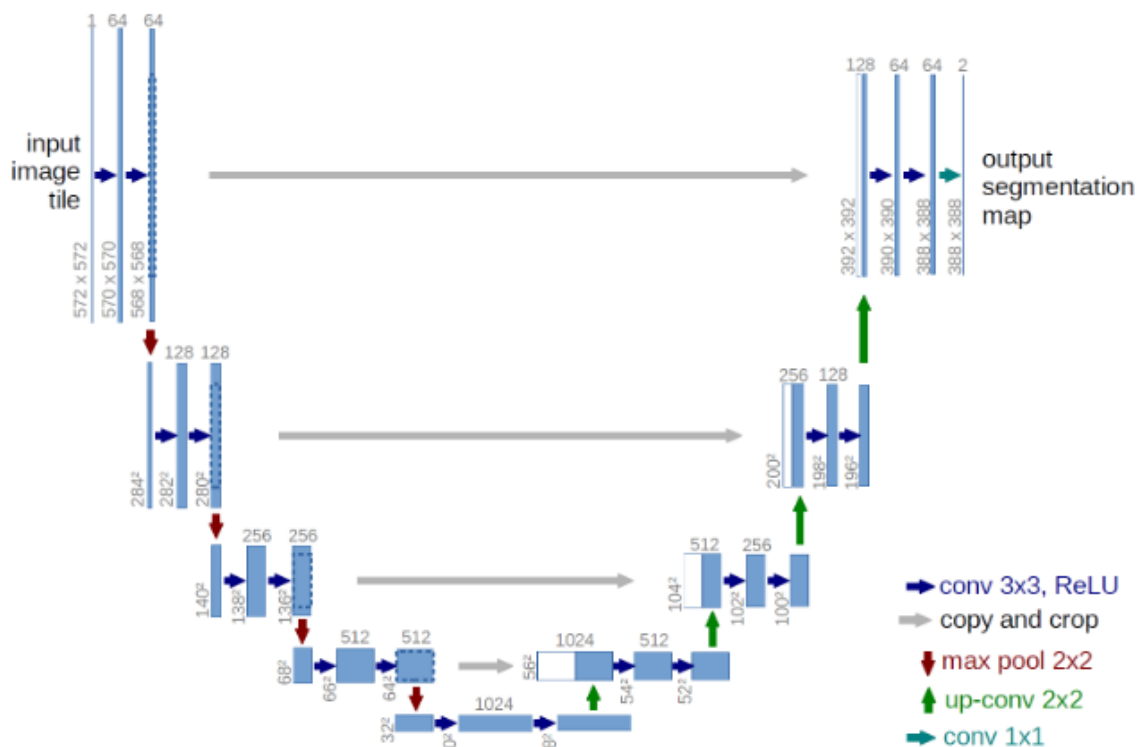
Tyto snímky jsou vstupem do CNN, jejíž architektura je na obrázku 18. Vrstvy této CNN odpovídají klasickým konvolučním neuronovým sítím, které se skládají z opakovaných aplikací 3x3 konvolucí následované jednotkou ReLU a operací „maxpooling“. Poté je aplikována dekonvoluce pro převzorkování, následuje zřetězení a ReLU jednotka. Poslední vrstva obsahuje 1x1 konvoluci. Jako aktivační funkci použili sigmoidu a chybovou funkci „class-balancing loss“. Pro předejití přeučení sítě je aplikován „dropout“ s parametrem $p=0,2$. Tato metoda dosahuje přesnosti 95,6 %. [40]



Obrázek 18: Architektura CNN podle Zhongwei Feng a spol. [40]

5.2 U-NET

U-net je konvoluční neuronová síť, která byla vytvořena speciálně pro segmentaci biomedicínských dat. Pro mnohé neuronové sítě je potřeba mít velké množství trénovacích dat, které ale často nejsou dostupné. Tato síť využívá trénovací data mnohem efektivněji a není jich potřeba takové množství. Architektura této sítě je na obrázku 19.



Obrázek 19: Architektura U-net, převzato z [35]

U-net se skládá ze dvou cest, a to z cesty kontraktilní a expanzivní. Obě tyto cesty jsou téměř symetrické a výsledná architektura je ve tvaru písmene U, proto se také nazývá tato síť U-net.

Kontraktilní cesta je v levé části obrázku a skládá se z opakovaných aplikací dvou 3x3 konvolucí, kdy pokaždé konvoluci následuje ReLU. Po každé dvojici konvolucí je aplikováno podvzorkování s oknem velikosti 2x2 a krokem 2. Takto dojde ke zmenšení obrazu na polovinu velikosti obrazu vstupujícího do této vrstvy.

Expanzivní cesta se skládá z opakování 2x2 konvoluce a ze zřetězení s odpovídajícími oříznutými mapami z kontraktilní cesty. Dále pak následuje 3x3 konvoluce a v poslední vrstvě je použita 1x1 konvoluce. Celkově má síť 23 konvolučních vrstev. Architektura U-net neobsahuje žádnou plně propojenou vrstvu.

Pro trénování této sítě je potřeba mít k dispozici trénovací obrazy. Trénovací sada obrazů se sestává z originálních obrazů a k nim příslušných segmentačních map („ground truth“).

U-net byla trénovaná na snímcích získaných ze světelného mikroskopu. Trénovací sada, obsahující snímky glioblastom-astrocytom U373 buňky, byla získaná pomocí fázově-kontrastního mikroskopu. Na této sadě dosahuje síť průměrné IOU („intersection over union“) hodnoty 92 %. Druhým souborem trénovacích dat byly snímky buněk HeLa, které byly získány pomocí mikroskopu využívajícího diferenciální interferenční kontrast. U-net na této sadě dosáhla průměrné IOU hodnoty 77,5 %. [41]

6 VYHODNOCOVACÍ METODY

V této kapitole budou popsány metody vyhodnocování úspěšnosti testování neuronových sítí. Tyto metody budou využity v praktické části pro hodnocení úspěšnosti vybrané metody a pro srovnání s ostatními metodami. Při výpočtu jednotlivých statistických skóre se využívá pojmů, které jsou vysvětleny následující tabulkou [42]:

Tabulka 1: Statistické pojmy

Výsledky	Cévy v ground truth	Pozadí v ground truth
Cévy	True Positive (TP)	False Positive (FP)
Pozadí	False Negative (FN)	True Negative (TN)

6.1 Senzitivita

Citlivost neboli sensitivita je poměr správně predikovaných pozitivních případů ku všem skutečně pozitivním případům [43]. Při segmentaci to znamená poměr správně predikovaných pixelů cév v obraze, ku všem pixelům patřícím k cévám v ground truth.

$$\text{Senzitivita} = \text{TPR} = \frac{TP}{TP + FN} \quad (6.1)$$

6.2 Specifická

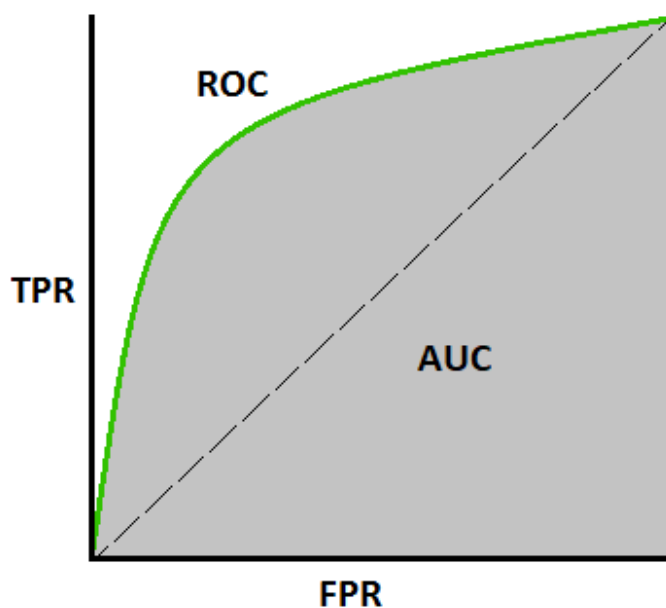
Specifická je poměr správně predikovaných negativních případů ku všem skutečně negativním případům [43]. V našem případě je to tedy poměr správně predikovaných pixelů pozadí v obraze, ku všem pixelům patřícím k pozadí v ground truth.

$$\text{Specifická} = \text{TNR} = \frac{TN}{TN + FP} \quad (6.2)$$

6.3 AUC

Pro určení hodnoty AUC neboli plochy pod křivkou je nutné definovat ROC křivku (obrázek 19). Tato křivka slouží pro hodnocení a optimalizaci klasifikačních systémů [44]. ROC křivka je závislost sensitivity na „fall-out“ (1 - specificita) označovaném jako FPR. Výpočet FPR je dán vzorcem [45]:

$$FPR = \frac{FP}{TN + FP} \quad (6.3)$$



Obrázek 20: ROC křivka, převzato z [44].

AUC určuje míru přesnosti testu, v našem případě míru přesnosti segmentace. Čím vyšší je hodnota AUC, tím model lépe odlišuje segmentovaný objekt od pozadí. Pokud je AUC rovno 100 %, predikovaný obraz se rovná ground truth. Hodnocení kvality testu (segmentace) je v tabulce 2.

Tabulka 2: Hodnocení kvality testu podle AUC

Hodnoty AUC	Význam
0,5 – 0,75	nedostatečný
0,75 – 0,92	dobrý
0,92 – 0,97	velmi dobrý
0,97 - 100	vynikající

6.4 ACC

Přesnost, označovaná podle anglického accuracy (ACC), je definována pomocí sensitivity a specificity. Jak už název napovídá ACC určuje přesnost testu (segmentace) a její výpočet je dán vztahem:

$$ACC = \frac{TP + TN}{TP + FN + TN + FP} \quad (6.4)$$

6.5 Dice koeficient

Dice koeficient je statistická metoda pro porovnání shodnosti dvou souborů, v tomto případě obrazů. Určuje procento podobnosti ground truth s výslednými predikovanými snímky. Vzorec pro dva obecné soubory dat je daný jako [46]:

$$D(X, Y) = 2 \cdot \frac{|X \cap Y|}{|X| + |Y|}, \quad (6.5)$$

kde X a Y představují dva soubory dat. V případě neuronových sítí představují ground truth a predikované obrazy. Čím je hodnota Dice koeficientu bližší 1, tím jsou si ground truth a predikovaný obraz podobnější.

Dice koeficient lze také vyjádřit pomocí termínů z tabulky 1 jako:

$$Dice\ koeficient = \frac{2 \cdot TP}{(TP + FP) + (TP + FN)} \quad (6.7)$$

7 IMPLEMENTACE U-NET

Vybraná metoda konvoluční neuronové sítě U-net byla realizovaná pomocí programovacího jazyka Python verze 3.6.8 používaného v integrovaném vývojovém prostředí PyCharm. Skript pro úpravu fundus snímků byl vytvořen v programovacím prostředí Matlab R2018b. K trénování a testování sítě byla využita grafická karta © ASUS GeForce GTX 1070Ti, která se nachází na Ústavu biomedicínského inženýrství. Jelikož samotná implementace architektury sítě U-net je velmi náročná, byly využity dostupné knihovny, které práci s neuronovými sítěmi zjednodušují. [47]

Tensorflow je open source softwarová knihovna pro vysoce náročné numerické výpočty. Tato knihovna byla vyvinuta výzkumnými pracovníky a inženýry z týmu Google Brain v rámci organizace Google AI. Architektura této knihovny umožňuje počítání výpočtů na různých platformách (procesory, GPU, TPU). Tensorflow byla naprogramována v C++, ale je podporována i v jazyce Python. Výhodou této knihovny je rozhodně to, že umožňuje výpočet na grafické kartě (GPU), kde je výpočet několikanásobně rychlejší než na centrálním procesoru (CPU). [48]

Keras je knihovna, která byla vyvinuta pro programování konvolučních neuronových sítí. Byla vytvořena na základě umožnění rychlého experimentování, aby bylo možné dojít od nápadu k výsledku co nejrychleji. Knihovna Keras byla napsána v jazyce Python a je podporována Tensorflow a Theano. Stejně jako Tensorflow podporuje výpočty na CPU a GPU. [47]

Přestože U-net byla testována na jiných datech než na datech, které jsou obsahem této práce, architektura této sítě zůstala téměř nezměněna. Architektura U-net byla převzata z [49], kde byla přidána do 4 a 5 konvolučního bloku vrstva „dropout“ s parametrem $p=0,5$ pro předejití přeučení sítě.

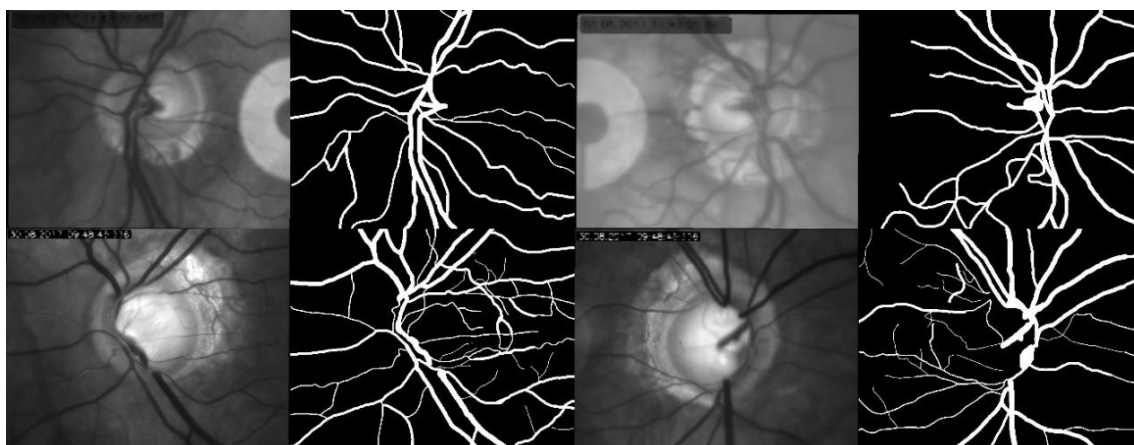
Byly vytvořeny skripty pro načítání dat, trénování, testování a validaci výsledků. Implementovaná metoda je natrénovaná na různých datových sadách snímků z experimentálního video oftalmoskopu. Z důvodu malého rozšíření těchto datových sad v člancích je vybraná metoda natrénovaná také na datové sadě HRF, která obsahuje snímky z fundus kamery. Tato datová sada je veřejně dostupná, a proto ji mnoho experimentátorů využilo ve svých experimentech. Datová sada HRF slouží pro porovnání výsledků vybrané metody s výsledky jiných metod segmentace cévního řečiště pro danou sadu snímků.

7.1 Datové sady z experimentálního video oftalmoskopu

Tato datová sada byla získána pomocí experimentálního video oftalmoskopu, kdy z každého video záznamu (po registraci) byl vytvořen průměrný snímek. Trénovací a testovací snímky bylo nutné předzpracovat. Nejprve bylo potřeba změnit jejich velikost ze 1000x770 na 512x512 z důvodů snížení výpočetní náročnosti. Jelikož jsou vstupní obrazy šedotónové nebylo je potřeba převádět do odstínů šedi. Rozsah šedotónového obrazu byl normalizován na hodnoty od 0 do 1. Z této datové sady byly vytvořeny dvě samostatné sady, které se liší pouze v kvalitě binárních masek.

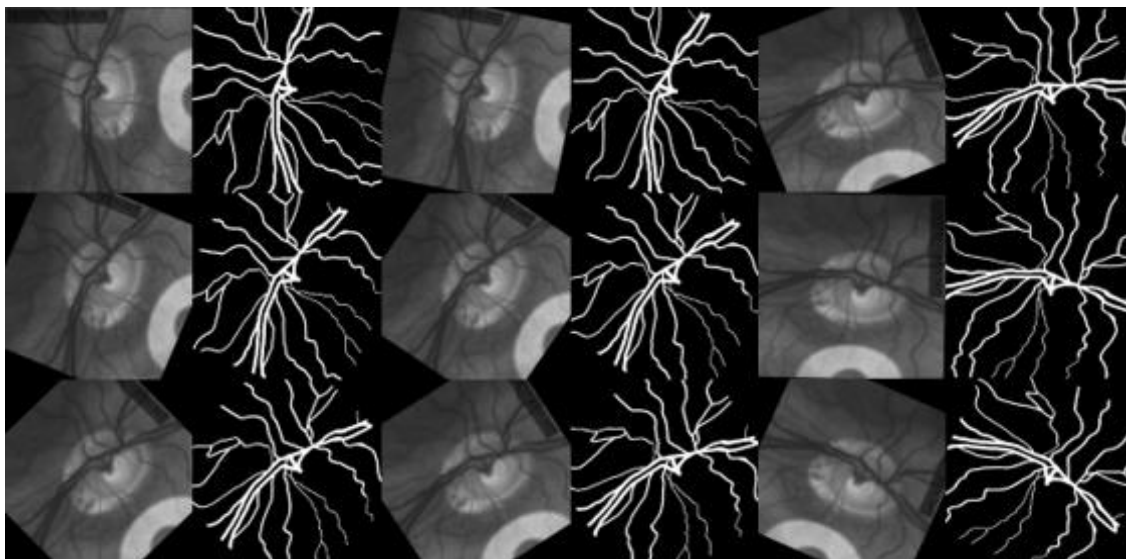
7.1.1 Experimentální sada A

Experimentální sada snímků A obsahuje 20 snímků, které byly rozděleny na 15 trénovacích a 5 testovacích snímků. Binární masky („ground truth“) byly získány základní segmentační metodou a ručně dokresleny pro co největší přesnost. Příklady testovacích snímků a jejich příslušných binárních masek jsou na obrázku 21.



Obrázek 21: Příklad snímků a jejich příslušných binárních masek z experimentální sady A

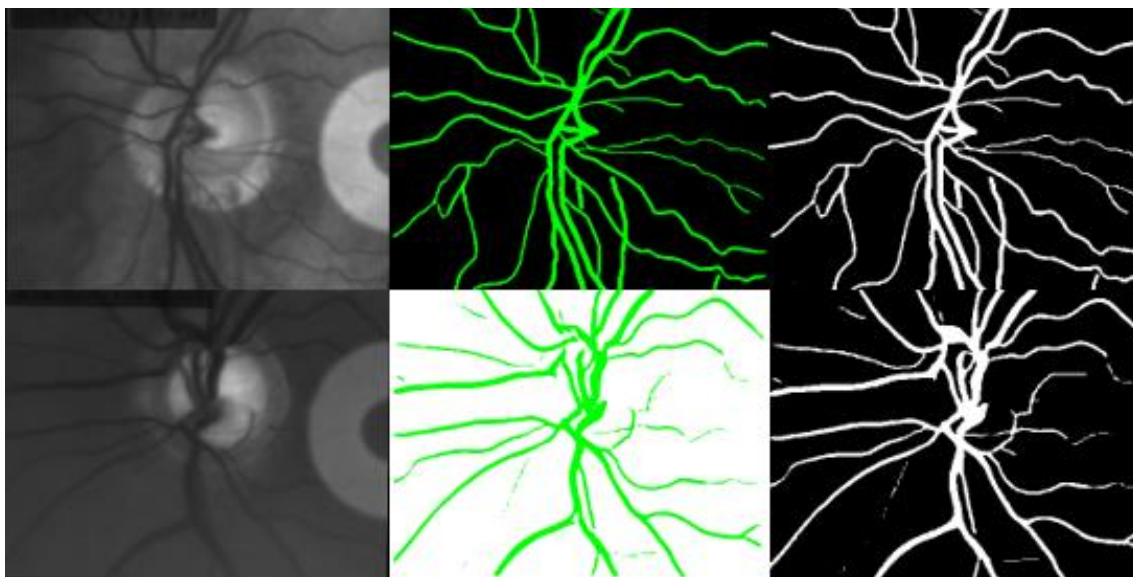
Z důvodu malého počtu snímků v této sadě byla využita také augmentace neboli umělé rozšíření datové sady. Augmentace byla provedena pomocí rotace snímků po 30°. Po této úpravě obsahuje sada 180 snímků (obrázek 22). Původní i uměle zvětšená experimentální sada A byla využita při trénování implementované metody.



Obrázek 22: Příklad uměle rozšířené experimentální sady A rotací snímků o 30°

7.1.2 Experimentální sada B

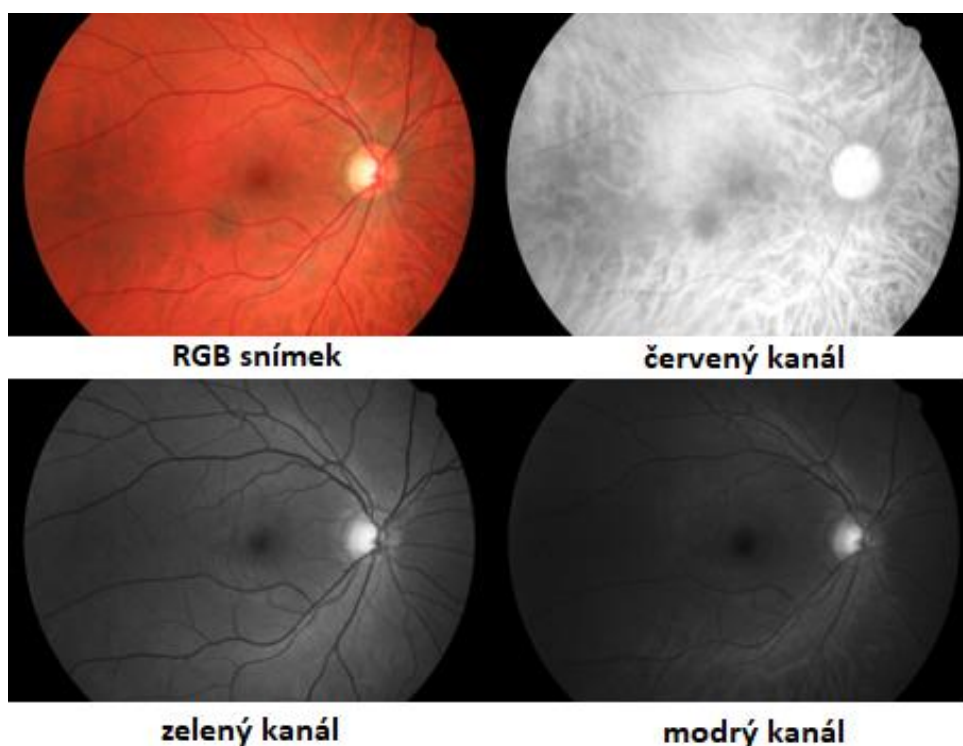
Experimentální sada snímků B obsahuje 146 snímků, které byly rozděleny na 122 trénovacích a 24 testovacích snímků. Masky k této datové sadě byly původně zelenočerné nebo zelenobílé (obrázek 23), proto bylo potřeba je převést na binární. Rozdíl v maskách sady B a sady A je pouze v tom, že v sadě B nebyly masky tak precizně dokresleny. Přesnost byla dodržena jen v okolí optického disku.



Obrázek 23: Příklady původních snímků z experimentální sady B, jejich příslušných původních masek a masky převedené na binární

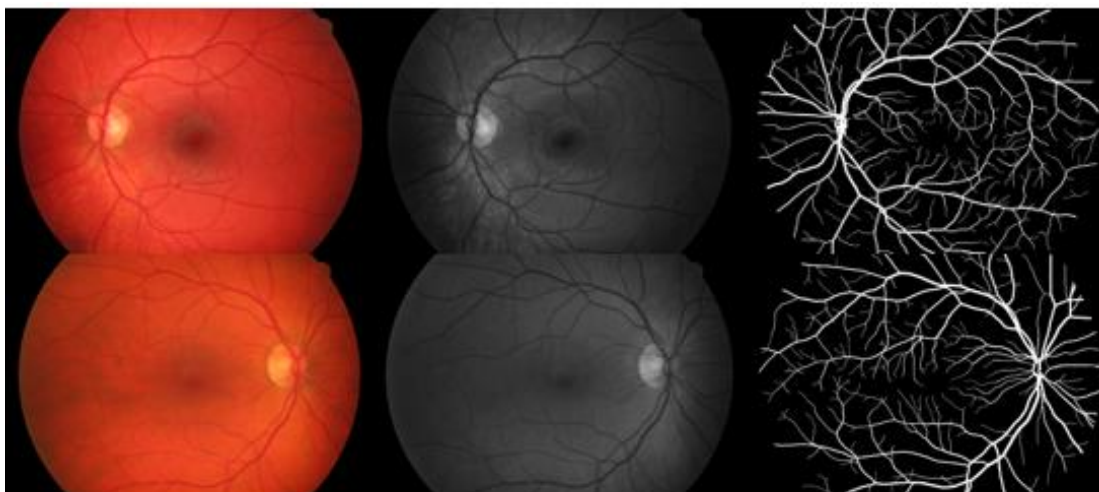
7.2 Datová sada HRF

Tato datová sada obsahuje snímky získané z fundus kamery. Přesně se jedná o snímky z databáze HRF. Tato datová sada obsahuje 14 snímků rozdělených na 10 trénovacích a 4 testovací snímky. U snímků z fundus kamery je nutné obrazy před vstupem do neuronové sítě předzpracovat. Fundus snímek je barevný RGB obraz. Z literatury vyplývá, že pro segmentaci není vhodné převést RGB do šedotónového obrazu, ale je lepší vybrat jeden z RGB kanálů, kde jsou cévy nejlépe vidět. Na obrázku 24 lze vidět snímky z jednotlivých kanálů RGB snímku. V červeném kanále cévy nejsou téměř vidět, v modrém kanále jsou lépe viditelné ale tento kanál má špatný kontrast. Proto je pro segmentaci vybrán zelený kanál, kde jsou cévy viditelné nejlépe.



Obrázek 24: RGB snímek z fundus kamery a jeho zobrazení v jednotlivých kanálech.

Jako u předchozích datových sad byla velikost snímků upravena na 512x512 a rozsah hodnot na (0,1). Příklad ze sady snímků z fundus kamery je na obrázku 25.



Obrázek 25: Příklad z datové sady HRF

U datové sady HRF byla kvůli malému množství dat také využita augmentace. Podobně jako u experimentální sady A byly u této datové sady snímky rotovány po 30° a datová sada se tak zvětšila na 120 snímků. Původní i uměle zvětšená datová sada HRF byla využita k trénování sítě.

8 VÝSLEDKY A DISKUZE

Trénování a testování sítě probíhalo na všech vytvořených datových sadách. Architektura sítě zůstala stejná, jak bylo popsáno v teoretické části.

Z optimalizačních funkcí, které bylo možno použít vyšla jako nejlepší optimalizační funkce Adam, která pro daný problém vykazovala nejlepší výsledky. Rychlost učení optimalizační funkce byla experimentálně určena na 0,0001. Nižší hodnota rychlosti učení způsobovala pomalejší učení sítě, které mohlo být kompenzováno zvýšením počtu epoch. Zvýšením počtu epoch se ale také zvýší celkový čas učení sítě. Nízká hodnota rychlosti učení také může způsobit setrvání hodnoty úspěšnosti trénování v lokálním minimu, a to může být důvodem k nenatrénování sítě. Zvýšení rychlosti učení vede k přeučení sítě už po pár epochách.

Jako chybová funkce byla použita binární křížová entropie, která se hojně využívá pro segmentaci obrazu do dvou klasifikačních skupin, kterými jsou v našem případě cévy a pozadí (bílá a černá).

Trénování sítě bylo provedeno pro různý počet epoch, tak aby se dospělo k co nejlepšímu možnému výsledku následného testování sítě. Sít' byla natrénována a otestována na každé datové sadě zvlášť. Pro experimentální datovou sadu A a datovou sadu HRF bylo využito umělého zvětšení těchto datových sad pro lepší výsledky jak trénování, tak testování.

8.1 Experimentální datová sada A

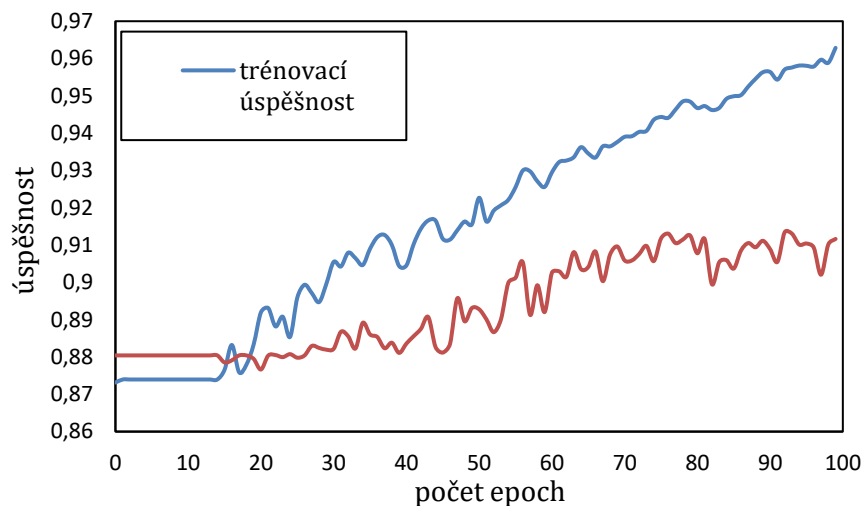
Z důvodu malého množství snímků v experimentální sadě A bylo nutné volit větší počet epoch učení sítě. Pro malý počet epoch se sít' nenatrénovala. Výsledky trénování jsou v tabulce 3, kde lze vidět, že se zvyšujícím se počtem epoch roste jak trénovací, tak validační úspěšnost. S větším počtem epoch roste také čas trénování.

Tabulka 3: Výsledky trénování pro experimentální datovou sadu A

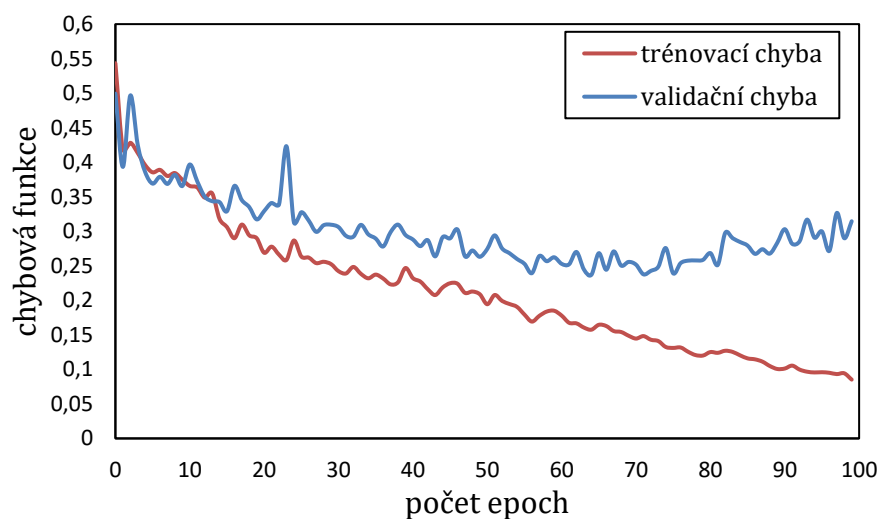
počet epoch	trénovací úspěšnost	validační úspěšnost	čas trénování [hh:mm:ss]
40	0,9160	0,8889	0:02:39
50	0,9215	0,8964	0:03:18
60	0,9308	0,9011	0:03:52
70	0,9322	0,9076	0:04:31
80	0,9442	0,9088	0:04:51
90	0,9482	0,9125	0:05:44
100	0,9629	0,9126	0:06:23

Zvyšující se trénovací a validační úspěšnost ale nezaručuje lepší výsledky testování. Pokud se blíže podíváme na jedno určité trénování, například trénování

pro 100 epoch a vykreslíme průběh úspěšnosti a chybové funkce v jednotlivých epochách (obrázek 26 a 27) lze vidět, že došlo k přetrénování sítě. Zhruba po 75 epochách dochází ke snižování validační úspěšnosti a ke zvyšování validační chyby, což indikuje právě přeučení sítě.



Obrázek 26: Průběh trénování sítě na experimentální sadě A po 100 epochách



Obrázek 27: Průběh chybové funkce při trénování sítě na experimentální sadě A po 100 epochách

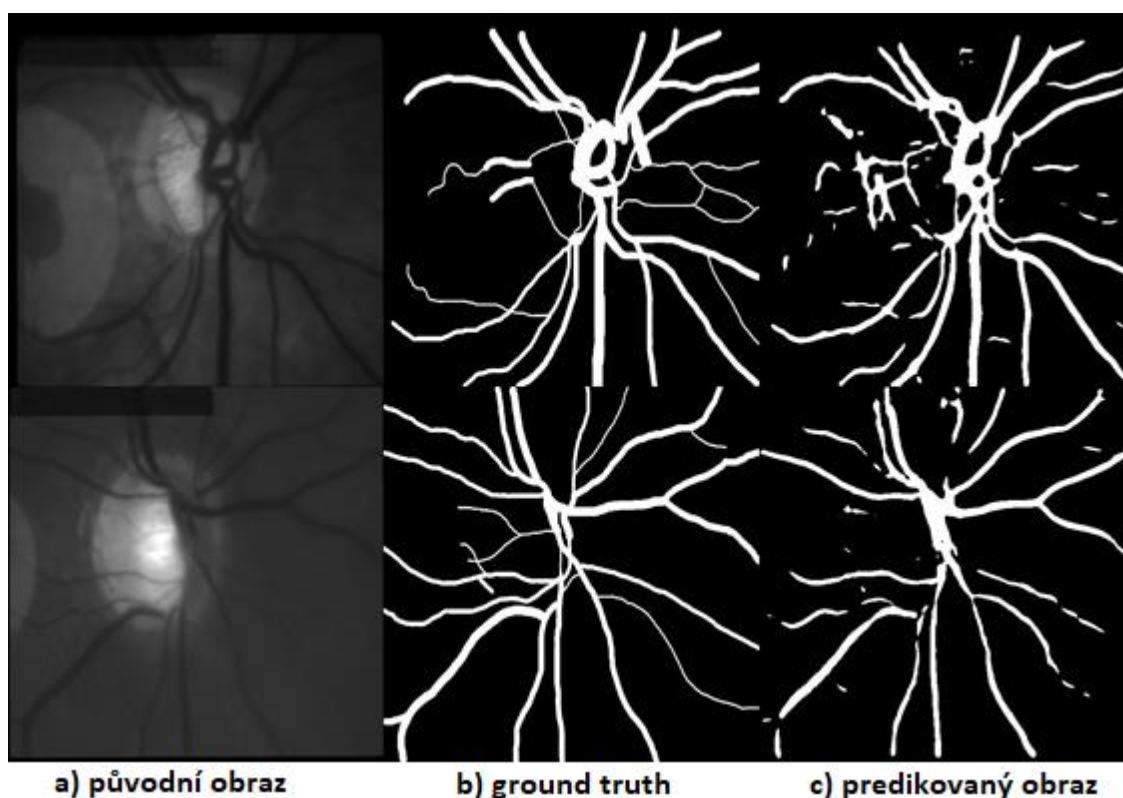
Síť natrénovaná na experimentální datové sadě A byla otestována na testovací sadě snímků, která obsahovala 5 snímků a k nim příslušné binární masky. Testování je hodnoceno pomocí pěti parametrů, kterými jsou senzitivita, specificita, přesnost (ACC), plocha pod ROC křivkou (AUC) a Dice koeficient. Tyto parametry jsou počítány jako průměr hodnot jednotlivých parametrů pro každý testovací snímek zvlášť. Výsledky testování jsou uvedeny v tabulce 4, kde je uveden počet epoch, po kterých byla síť natrénována a pět hodnot parametrů uvedených výše.

Tabulka 4: Výsledky testování pro experimentální datovou sadu A

počet epoch	SE	SP	ACC	AUC	Dice koeficient
40	0,5242	0,9741	0,9167	0,9251	0,6138
50	0,5408	0,9724	0,9173	0,9235	0,6230
60	0,5705	0,9726	0,9215	0,9246	0,6469
70	0,5483	0,9788	0,9239	0,9365	0,6454

Nejlepšího výsledku dosahovala natrénovaná síť po 70 epochách, kdy hodnota přesnosti sítě dosahuje 92,39 %. Nižší hodnota úspěšnosti je způsobena malým množstvím trénovacích dat, na kterých se síť nedokáže dobře natrénovat.

Příklady výstupu testování, kterými jsou predikce segmentovaných cév, jsou na obrázku 28. Z obrázku lze vidět, že síť docela dobře segmentuje silné cévy, ale problémem jsou cévy tenké, které síť segmentuje velice málo nebo vůbec. Také se do segmentovaných struktur dostávají hranice optického disku, což je v našem případě nežádoucí. Proto také Dice koeficient, který ukazuje podobnost „ground truth“ a predikovaného obrazu, dosahuje pouze hodnoty 64,54 %.



Obrázek 28: Testování na experimentální sadě A

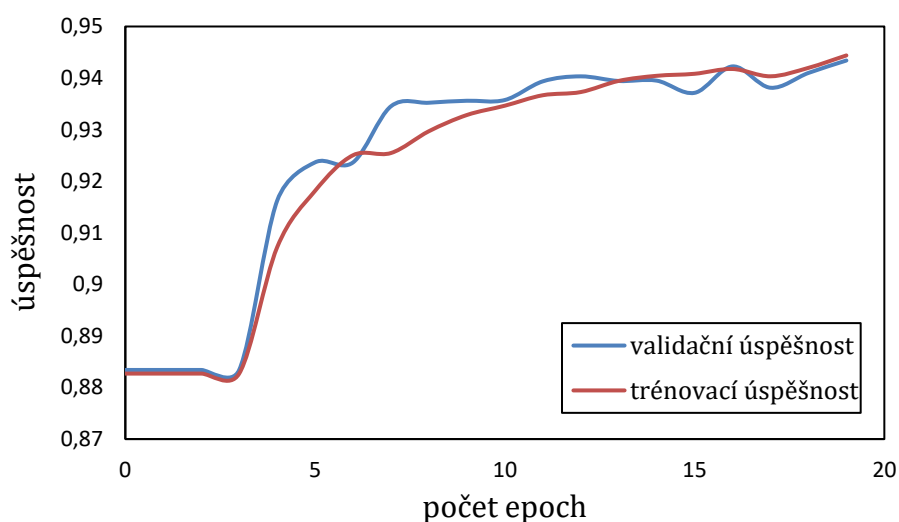
Výsledky testování na experimentální datové sadě nemůžeme považovat za dostačující, proto byla experimentální datová sada uměle zvětšena pomocí augmentace. Počet trénovacích snímků vzrostl z 15 na 180 snímků. U-net byla natrénovaná také na této nově vytvořené datové sadě pro různý počet epoch.

Počet epoch byl volen menší než v předchozím případě, protože na vstup sítě bylo přivedeno větší množství trénovacích dat. Výsledky trénování sítě na augmentované experimentální sadě A jsou v tabulce 5. Jako v předchozím případě se zvyšujícím se počtem epoch roste trénovací i validační úspěšnost. V porovnání s trénováním na původní experimentální sadě A dosahuje úspěšnost trénování vyšší hodnoty pro augmentovanou datovou sadu. Ovšem stejně jako v prvním případě i zde došlo k přetrénování sítě, a to pro více jak 30 epoch.

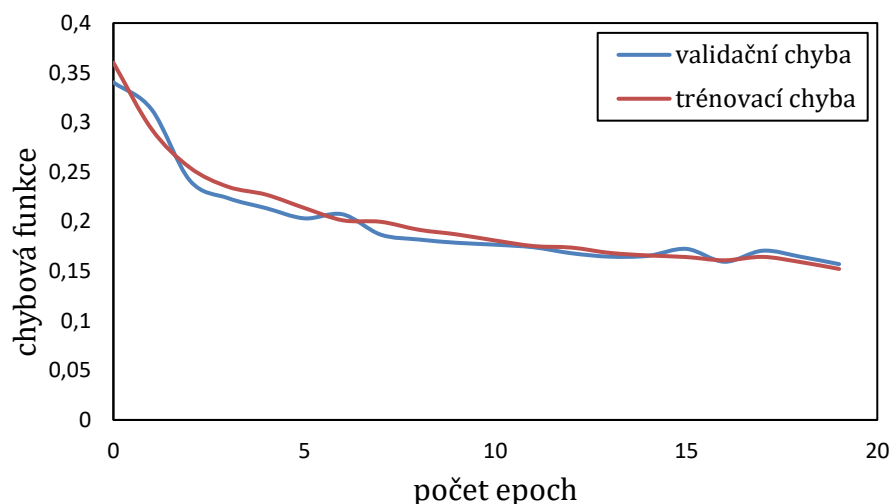
Tabulka 5: Výsledky trénování pro augmentovanou experimentální datovou sadu A

počet epoch	trénovací úspěšnost	validační úspěšnost	čas trénování [hh:mm:ss]
10	0,9310	0,9346	0:07:31
20	0,9444	0,9434	0:14:49
30	0,9536	0,9456	0:22:13
40	0,9647	0,9454	0:29:30
50	0,9718	0,9444	0:36:55

Na obrázku 29 a 30 jsou vidět průběhy trénování a chybové funkce po 20 epochách, kde lze názorně vidět, jak tyto průběhy vypadají při dobrém trénování. Stagnace úspěšnosti po prvních pár epoch je způsobena optimalizačním algoritmem, který nejprve hledá optimální „cestu“, kterou se má vydat a poté teprve začíná zlepšovat trénovací i validační úspěšnost.



Obrázek 29: Průběh trénování sítě na augmentované experimentální sadě A po 20 epochách



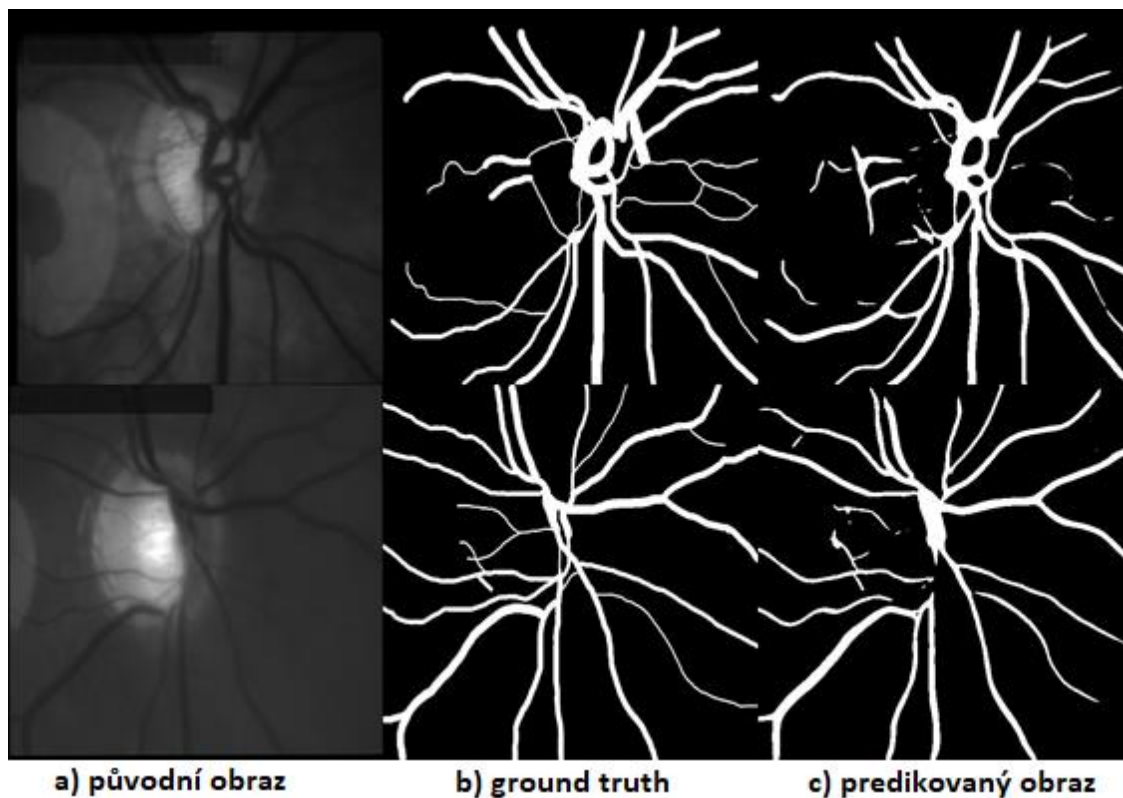
Obrázek 30: Průběh chybové funkce při trénování sítě na experimentální sadě A po 20 epochách

Síť natrénovaná na augmentované sadě snímků byla otestována na stejných testovacích snímcích jako síť natrénovaná na původní datové sadě A. V tabulce 6 jsou výsledky tohoto testování, kdy nejlepší přesnosti dosahuje po 30 epochách a to hodnoty 93,34 %. To znamená, že augmentací datové sady jsme dosáhli o něco lepších výsledků než při testování na původní experimentální sadě A.

Tabulka 6: Výsledky testování pro augmentovanou experimentální datovou sadu A

počet epoch	SE	SP	ACC	AUC	Dice koeficient
10	0,6415	0,9718	0,9296	0,9442	0,7000
20	0,6989	0,9663	0,9320	0,9527	0,7248
30	0,7044	0,9667	0,9334	0,9573	0,7303

Příklad predikovaných snímků takto natrénované sítě je na obrázku 31. Z obrázku si lze všimnout, že silné cévy jsou segmentovány s velikou přesností, problém ovšem nastává u tenkých cév. Tenké cévy jsou segmentovány o něco lépe než v předchozím případě, ale pořád ne dostatečně přesně. Proto, i když je Dice koeficient zhruba o 4 % vyšší než při testování na původní sadě dat, dosahuje pouze hodnoty 73,03 %.



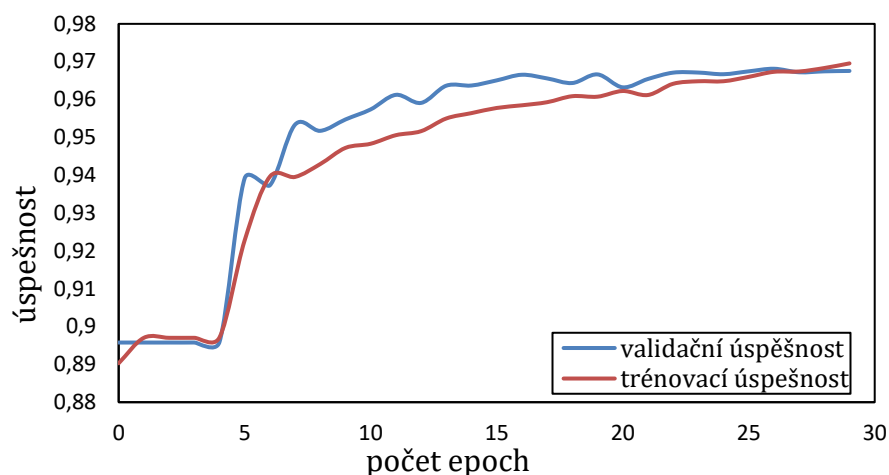
Obrázek 31: Testování na augmentované experimentální sadě A

8.2 Experimentální datová sada B

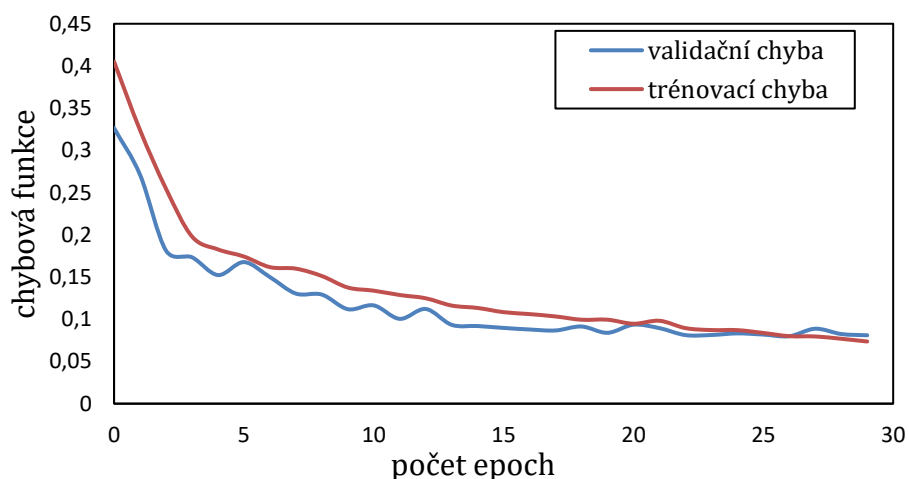
K trénování na experimentální datové sadě B nebylo potřeba vysoký počet epoch, protože tato datová sada obsahuje dostatečné množství trénovacích snímků. Výsledky trénování jsou uvedeny v tabulce 7. Trénovací úspěšnost dosahovala po 30 epochách 96,95 %. Při větším počtu epoch došlo k přetrénování sítě. Vývoj úspěšnosti a chybové funkce pro 30 epoch je na obrázcích 32 a 33.

Tabulka 7: Výsledky trénování pro experimentální datovou sadu B

počet epoch	trénovací úspěšnost	validační úspěšnost	čas trénování [hh:mm:ss]
10	0,9476	0,9603	0:05:06
20	0,9593	0,9645	0:10:01
30	0,9695	0,9676	0:14:00



Obrázek 32: Průběh trénování na experimentální sadě B po 30 epochách

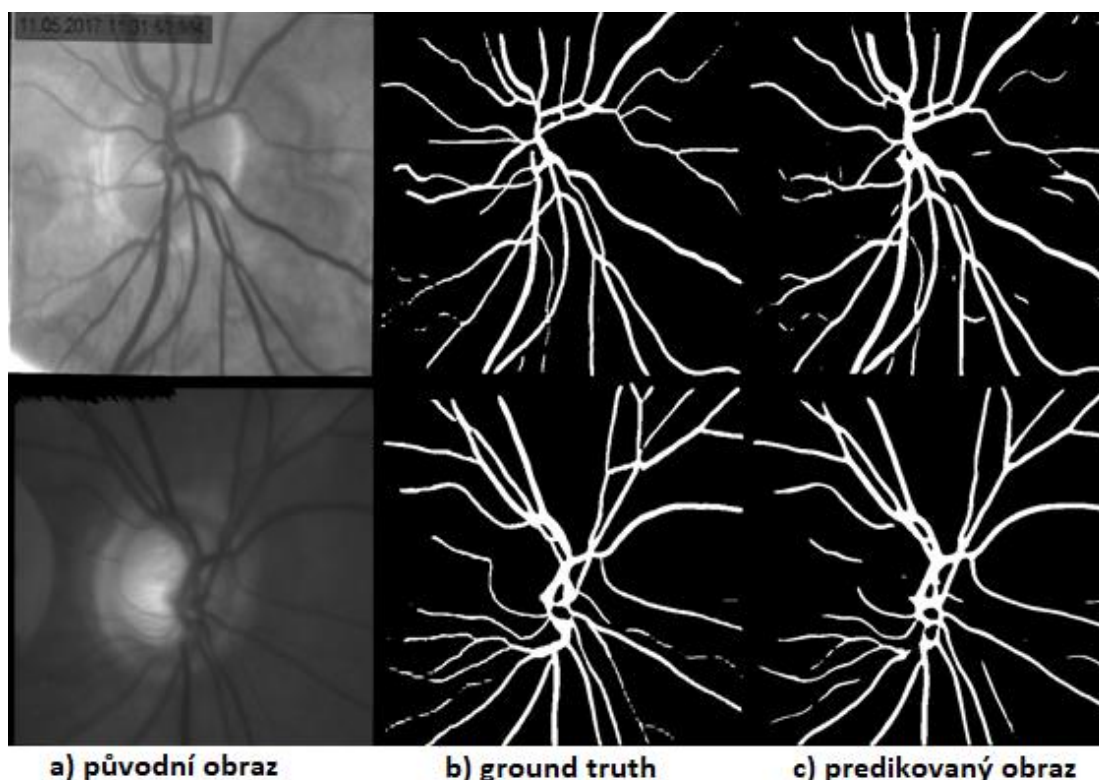


Obrázek 33: Průběh chybové funkce při trénování sítě na experimentální sadě B po 30 epochách

Testování natrénované sítě na experimentální sadě B probíhalo na 24 snímcích. V tabulce 8 jsou výsledky pro různý počet epoch. Nejlepší úspěšnosti segmentace cévního řečiště dosáhla síť po 30 epochách. Hodnota přesnosti dosáhla 96,07 %. Výsledky testování pro experimentální datovou sadu B jsou na obrázku 34. Výsledné predikce se opět setkávají s problémem tenkých cév, kdy část tenkých cév je segmentována správně, ale část cév není segmentována vůbec. To je také částečně způsobeno nedokonalostí binárních masek, které jsou využity pro trénování sítě.

Tabulka 8: Výsledky testování pro experimentální datovou sadu B

počet epoch	SE	SP	ACC	AUC	Dice koeficient
10	0,6906	0,9818	0,9506	0,9627	0,7456
20	0,7321	0,9850	0,9575	0,9746	0,7856
30	0,8208	0,9778	0,9607	0,9818	0,8180



Obrázek 34: Testování na experimentální sadě B

8.3 Porovnání výsledků experimentálních datových sad

Z výsledků se může zdát, že síť natrénovaná a otestovaná na experimentální datové sadě B dosahuje lepších výsledků než na experimentální datové sadě A. To ale jednoznačně nelze říct. Pokud jsou na vstup sítě přivedeny snímky s nedokonalými binárními maskami dochází k tomu, že síť se naučí také tyto nedokonalosti v binárních maskách, což pro testování na nedokonalých datech není problém. Pokud jsou ale na vstup testování takto natrénované sítě přivedena kvalitní data (to znamená s kvalitními binárními maskami), dochází ke snížení testovací přesnosti této sítě na hodnotu 93,97 % (tabulka 9), což je hodnota, které bylo dosaženo při trénování a testování na experimentální sadě A.

Z výsledku vyplývá, že je důležité, jak kvalitní data jsou přivedena na vstup trénování sítě. Ani větší množství nedokonalých dat nezaručí lepší výsledek segmentace oproti malému počtu kvalitních trénovacích dat.

Tabulka 9: Testování natrénované sítě na experimentální sadě B pro testovací snímky z experimentální sady A

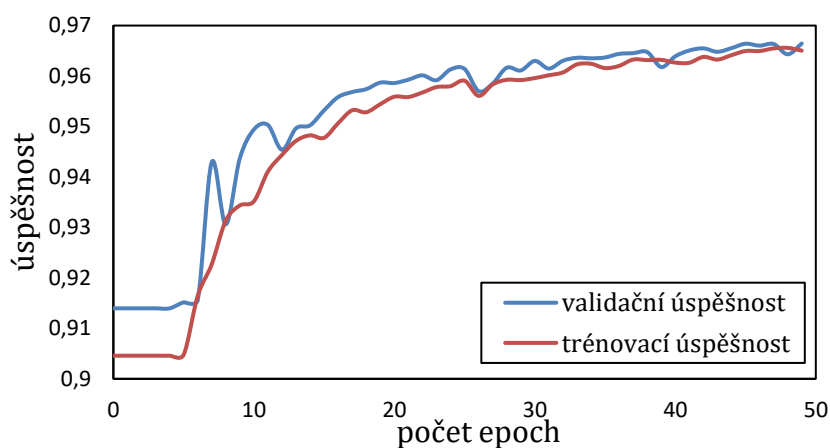
počet epoch	SE	SP	ACC	AUC	Dice koeficient
10	0,5903	0,9759	0,9267	0,9312	0,6726
20	0,5940	0,9823	0,9324	0,9503	0,6933
30	0,6884	0,9772	0,9397	0,9607	0,7477

8.4 Datová sada HRF

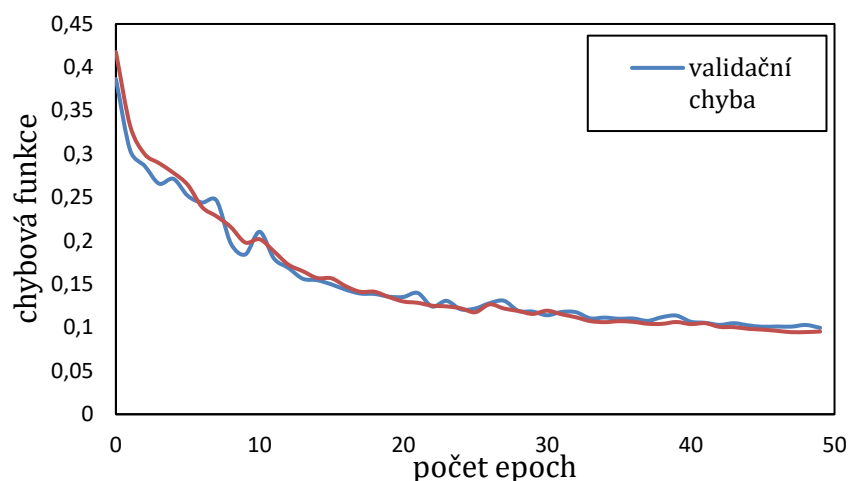
Datová sada HRF neobsahuje velké množství snímků, ale i přesto se síť dokáže natrénovat s úspěšností 96,50 % (tabulka 10). Tato hodnota je o něco vyšší než při trénování experimentálních datových sad. Z toho vyplývá, že daná architektura sítě si mnohem lépe dokáže poradit se snímky z fundus kamery než se snímky z experimentální video oftalmoskopu. To to je způsobeno tím, že snímky z fundus kamery jsou ostřejší a cévy jsou kontrastnější oproti pozadí. Průběh trénování a chybové funkce na datové sadě HRF je na obrázcích 35 a 36.

Tabulka 10: Výsledky trénování pro datovou sadu HRF

počet epoch	trénovací úspěšnost	validační úspěšnost	čas trénování [hh:mm:ss]
30	0,9545	0,9572	0:01:27
40	0,9607	0,9643	0:01:52
50	0,9650	0,9664	0:02:17



Obrázek 35: Průběh trénování na datové sadě HRF po 50 epochách



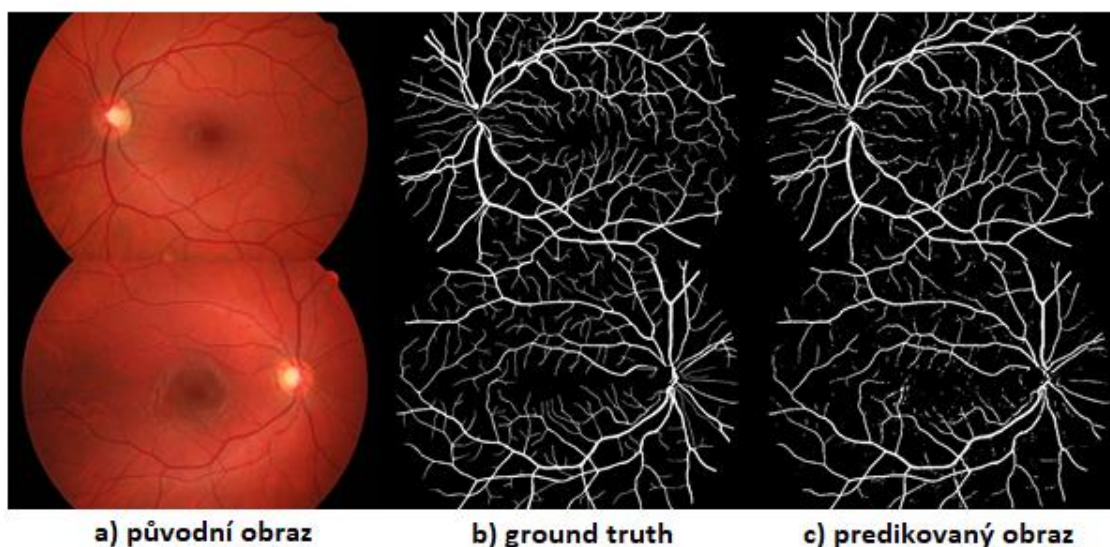
Obrázek 36: Průběh chybové funkce při trénování sítě na datové sadě HRF po 50 epochách

Testování sítě natrénované na datové sadě HRF dosahuje docela dobré úspěšnosti i při malém počtu snímků. Po 50 epochách dosahuje přesnost hodnoty 96,74 % (tabulka 11). Takové hodnoty se nám při testování předchozích datových sad nepodařilo dosáhnout. Příklad výsledné predikce je na obrázku 37.

Tabulka 11: Výsledky testování pro datovou sadu HRF

počet epoch	SE	SP	ACC	AUC
30	0,6042	0,9933	0,9591	0,9713
40	0,6833	0,9909	0,9640	0,9771
50	0,7457	0,9886	0,9674	0,9831

Predikce natrénované sítě na HRF sadě je velice přesná pro silné cévy. Ovšem s tenkými cévami si síť nedokáže úplně dobře poradit, přesněji s jejich ukončeními. Kdy tenké cévy nejsou plně propojeny a objevují se zde osamocené bílé pixely.

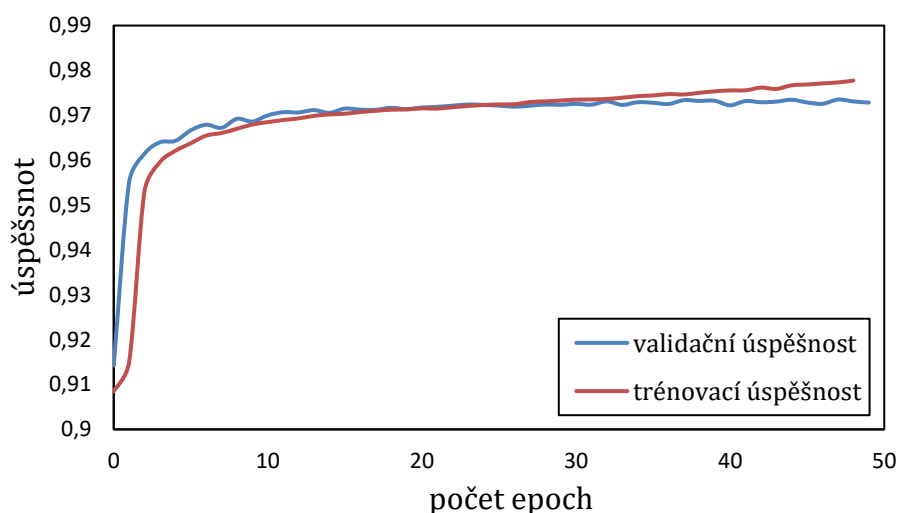


Obrázek 37: Testování na datové sadě HRF

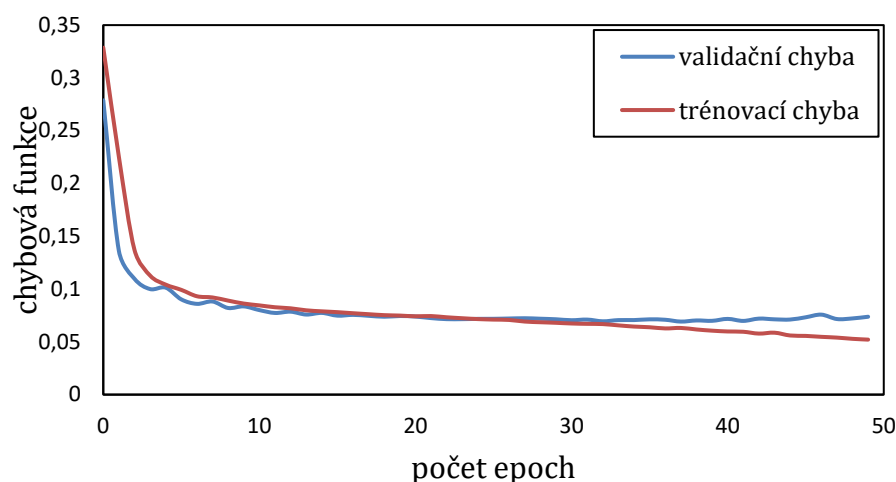
Z důvodu malého množství trénovacích snímků byla také u datové sady HRF využita augmentace dat. Sada byla zvětšena z 10 snímků na 120 snímků. Augmentace dat pomohla ke zlepšení trénovací úspěšnosti s 96,50 % na hodnotu 97,80 % (tabulka 12). Průběhy trénování a chybové funkce jsou na obrázcích 38 a 39, kde si lze všimnout, že zhruba po 35 epochách začíná validační úspěšnost a validační chyba pozvolna narůstat, to znamená, že došlo k přetrénování sítě. Nejlépe se síť tedy natrénovala po 30 epochách s úspěšností 97,26 %.

Tabulka 12: Výsledky pro augmentovanou datovou sadu HRF

počet epoch	trénovací úspěšnost	validační úspěšnost	čas trénování [hh:mm:ss]
10	0,9678	0,9691	0:05:09
20	0,9705	0,9706	0:10:05
30	0,9726	0,9721	0:14:58
40	0,9741	0,9723	0:19:54
50	0,9780	0,9728	0:24:50



Obrázek 38: Průběh trénování na augmentované datové sadě HRF po 50 epochách



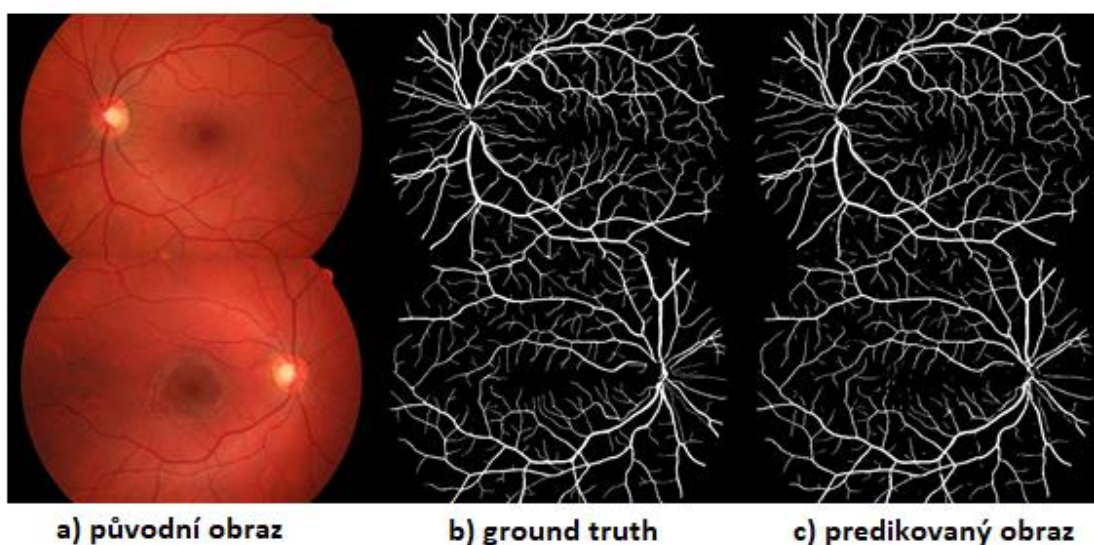
Obrázek 39: Průběh chybové funkce při trénování sítě na augmentované datové sadě HRF po 50 epochách

K testování sítě natrénované na uměle zvětšené datové sadě HRF byly použity stejné snímky jako pro testování na původní HRF sadě. V tabulce 13 jsou uvedeny výsledky tohoto testování. Lze si všimnout, že pomocí augmentace dat jsme schopni dosáhnout vyšší přesnosti segmentace cévního řečiště. Hodnota přesnosti segmentace dosahuje 97,50 %, což je o necelé procento lepší než testování na původní HRF sadě.

Tabulka 13: Výsledky testování pro augmentovanou datovou sadu HRF

počet epoch	SE	SP	ACC	AUC
10	0,8030	0,9876	0,9714	0,9880
20	0,8382	0,9863	0,9733	0,9878
30	0,8336	0,9887	0,9750	0,9913

Na obrázku 40 jsou uvedeny příklady výsledné predikce sítě natrénované na augmentované datové sadě HRF. V porovnání s obrázkem 37 jsou zde tenké cévy segmentovány s větší přesností. Objevuje se také menší množství osamocených bílých pixelů.



Obrázek 40: Testování na augmentované datové sadě HRF

8.5 Porovnání výsledků datové sady HRF s jinými metodami

Datová sada HRF je využita pro porovnání výsledků implementované metody s jinými metodami zabývajícími se segmentací cévního řečiště. Jako hodnoty ke srovnání s ostatními metodami byly vybrány nejlepší výsledky, kterých bylo implementovanou metodou dosaženo. To znamená, že byly použity výsledky dosažené na augmentované datové sadě HRF. V tabulce 14 jsou uvedeny hodnoty senzitivity, specifity a přesnosti u jednotlivým vybraných metod a metody navržené a otestované v této diplomové práci. K porovnání jsou využity výsledky metod segmentace cévního řečiště na databázi HRF. V porovnání s ostatními metodami dosahuje implementovaná metoda nejvyšší přesnosti a to 97,50 %.

Tabulka 14: Porovnání výsledků datové sad HRF s jinými metodami

metoda	SE	SP	ACC
Bantan, 2016 [50]	0,7000	0,9700	0,9500
Kumar et al., 2016 [51]	0,7485	-	0,9520
Odstrčilík et al., 2015 [1]	0,7887	0,9931	0,9581
Annunziata et al., 2015 [52]	0,6820	0,9935	0,9587
Budai et al., 2013 [53]	0,6620	0,9920	0,9610
Cheng et al., 2014 [54]	0,7889	0,9865	0,9647
trénování na HRF, 2019	0,8336	0,9887	0,9750

9 NÁVOD K OBSLUZE PROGRAMŮ

V této kapitole bude popsán stručný postup, jak spustit vytvořené programy a funkce, které k nim přísluší. Celkově byly vytvořeny dva programy pro trénování (training.py) a testování (testing.py) neuronové sítě a pět skriptů obsahující funkce načítání dat (data_generator.py), statistické výpočty (statistic_metrics.py), výpočet ROC křivky a AUC (ROC_curve_AUC.py), architekturu U-net (model.py) a úpravu fundus snímků (fundus_preprocessing.m). Návod je rozdělen do tří částí a těmi jsou příprava dat a skriptů, trénování a testování.

9.1 Příprava

Před samotným otevřením programů je nutné provést několik kroků:

1. Všechny programy a skripty uložte do jedné složky
2. Vytvořte 4 složky, kde uložte trénovací snímky, trénovací masky, testovací snímky a testovací masky
3. Upravte název snímků na čísla od 0 do celkového počtu snímků (takto je potřeba upravit jak trénovací a testovací snímky, tak jejich příslušné masky a to tak, aby maska měla stejné číslo jako k ní příslušný snímek)
4. Pokud chcete použít k trénování a testování neuronové sítě snímky s fundus kamery využijte pro jejich úpravu skript fundus_preprocessing.m (otevřete v matlabu), kde zadejte cestu ke snímkům, dále pak cestu, kam se mají upravené snímky uložit a počet všech snímků, které vaše sada obsahuje
5. Otevřete Pycharm a jako projekt otevřete složku, ve které jsou uloženy programy a skripty (file → open → složka s programy a skripty)
6. Nainstalujte všechny potřebné knihovny, které jsou uvedeny v seznamu knihoven

Seznam použitých knihoven:

- | | |
|------------------|----------------|
| • numpy | • matplotlib |
| • tensorflow-gpu | • scikit-learn |
| • keras | • cv2 |
| • time | • os |
| • pandas | • skimage |
| • xlswriter | • imgaug |

9.2 Trénování

1. V Pycharmů otevíráte program training.py
2. Zadejte cestu k trénovacím snímkům (train_path)
3. Zadejte cestu k trénovacím maskám (mask_path)
4. Zadejte počet obrázků, které trénovací sada obsahuje (number_of_images)
5. Vyberte generátor dat:
 - a) bez augmentace dat
 - b) s augmentací dat
6. Ve funkci model.fit nastavte počet epoch trénování (epochs) a jaké procento snímků má být použito k validaci (validation_split)
7. Spustíte program

Výsledky trénování se uloží do souboru s příponou .hdf5. Dále se do excelu uloží hodnoty dosažené trénovací a validační úspěšnosti a chyby po každé epoše. Poslední hodnotou, co se uloží do textového souboru je čas trénování.

9.3 Testování

1. V Pycharmů otevíráte program testing.py
2. Zadejte cestu k trénovacím snímkům (test_path)
3. Zadejte cestu k trénovacím maskám (mask_path)
4. Zadejte cestu, kam se mají predikované snímky uložit (save_path)
5. Zadejte počet obrázků, které testovací sada obsahuje (number_of_images)
6. Ve funkci model.load_weights zadejte název souboru s příponou .hdf5, kde jsou uloženy natrénované váhy sítě.
7. Spustíte program

Výstupem programu testing.py jsou predikované snímky (binární) a statistické hodnocení testování sítě jako je přesnost, senzitivita, specificita, Dice koeficient a plocha pod ROC křivkou.

ZÁVĚR

Tématem diplomové práce byla segmentace cévního řečiště v retinálních obrazových datech. V teoretické části byla popsána základní anatomie a fyziologie oka, principy metod pozorování očního pozadí, ke kterým se řadí také experimentální video oftalmoskop. Následně byly popsány principy klasických a konvolučních neuronových sítí. Byla provedena rešerše ohledně metod segmentace cévního řečiště v retinálních obrazových datech a vybrána jedna metoda, která byla realizována.

Vybranou metodou byla konvoluční neuronová síť U-net, která byla implementována v programovacím jazyce Python, za použití knihoven Tensorflow a Keras. Následně byla tato síť natrénována a otestována.

Trénování a testování probíhalo na třech datových sadách. Dvě datové sady obsahovaly snímky z experimentálního video oftalmoskopu (experimentální datové sady A a B) a jedna datová sada (HRF) obsahovala snímky získané pomocí fundus kamery.

Trénování na experimentální datové sadě A dosáhlo trénovací úspěšnosti 93,22 % po 70 epochách. Pro větší počet epoch došlo k přetrénování sítě. Hodnota přesnosti testování je rovna 92,32 %. Výsledné predikce jsou nedokonalé a potýkají se s problémem segmentace tenkých cév, které jsou v původní obraze špatně viditelné, a proto také špatně predikovány. Pro zlepšení výsledků experimentální sady A byla využita augmentace dat pro umělé zvětšení této datové sady. Augmentace dat dokázala výsledky o něco málo vylepšit. Trénovací úspěšnost se zlepšila z hodnoty 93,22 % na hodnotu 95,36 % a testovací přesnost dosáhla hodnoty 93,34 %, což znamená zlepšení segmentace o 1 %. Výsledné predikce jsou také o něco lepší než u původní datové sady A, kdy část tenkých cév je segmentována správně. Stále se ale objevují části tenkých cév, které segmentovány nejsou. Silné cévy jsou segmentovány přesně.

Úspěšnost trénování na experimentální datové sadě B, která obsahuje větší množství trénovacích dat s nedokonalými binárními maskami, dosáhla hodnoty 96,95 %. Přesnost testování dosáhla hodnoty 96,07 %. I u tohoto testování se objevuje problém segmentace tenkých cév, což je dáno nedokonalostí binárních masek, kde nejsou tenké cévy přesně anotovány a také kvalitou vstupních snímků.

Z výsledků testování na experimentálních sadách může vyplývat, že testování na sadě B bylo úspěšnější, ale není tomu tak. Síť trénovaná na datech s nedokonalými binárními maskami se dokáže naučit i tyto nepřesnosti v anotaci masek. Proto byla síť natrénovaná na experimentální datové sadě B otestována na

kvalitních datech z experimentální sady A s výslednou přesností 93,97 %. Tato hodnota je téměř stejná jako u trénování a testování na experimentální sadě A. Z toho vyplývá, že ani větší množství nedokonalých trénovacích dat nedokáže zlepšit výslednou segmentaci cévního řečiště.

Trénování a testování na sadě HRF obsahující snímky s fundus kamery bylo provedeno stejně jako u předchozích datových sad, jen bylo potřeba data jinak předzpracovat. Testovací úspěšnost dosáhla hodnoty 96,50 % a přesnost trénování 96,74 %. Pro zlepšení výsledků byla využita augmentace této datové sady. Trénovací úspěšnost se zlepšila na 97,80 % a ACC testování na 97,50 %. Výsledné predikce jsou docela přesné, kdy silné cévy jsou segmentovány s velkou přesností, avšak tenké cévy nejsou úplně dokonale segmentovány a objevují se osamocené bílé pixely.

V porovnání výsledků testování na datové sadě HRF s jinými metodami segmentace cévního řečiště v retinálních snímcích na této datové sadě vyšla implementovaná metoda nejlépe v přesnosti segmentace, kdy dosáhla hodnoty 97,50 %, což je o 1-2 % vyšší hodnota než u ostatních metod.

Implementovaná metoda U-net je kvalitní metodou pro segmentaci cévního řečiště v retinálních snímcích. Síť se dokáže natrénovat tak, aby segmentovala silné cévy s velkou přesností. Tato metoda funguje lépe pro snímky z fundus kamery, které jsou oproti snímkům z experimentálního video oftalmoskopu ostřejší. Problémem segmentace tenkých cév je, že jsou ve snímcích špatně viditelné, tudíž také těžko detekovatelné, ale i s tím si implementovaná metoda dokáže částečně poradit. Pro zlepšení výsledků segmentace implementované metody u datových sad obsahujících snímky z experimentálního video oftalmoskopu je potřeba zamyslet se nad předzpracováním těchto dat. Je možné, že pokud se ze snímků odstraní šum a eliminuje se nízká kvalita snímků dosáhne metoda pro dané data lepších výsledků.

Literatura

- [1] ODSTRČILÍK, J.; KOLÁŘ, R.; HARABIŠ, V.; TORNOW, R. Classification-Based Blood Vessel Segmentation in Retinal Images. In Computational Vision and Medical Image Processing V. London, UK: CRC Press Taylor and Francis Group, 2015. s. 95-100. ISBN: 978-1-138-02926-2.
- [2] WARD, Jeremy P. T a R. W. A LINDEN. *Základy fyziologie*. Praha: Galén, c2010. ISBN 978-80-7262-667-0.
- [3] HOLIBKOVÁ, Alžběta a Stanislav LAICHMAN. *Přehled anatomie člověka*. 5. vyd. Olomouc: Univerzita Palackého v Olomouci, 2010. ISBN 978-80-244-2615-0.
- [4] SYNEK, Svatopluk a Šárka SKORKOVSKÁ. *Fyziologie oka a vidění*. 2., dopl. a přeprac. vyd. Praha: Grada, 2014. ISBN 978-80-247-3992-2.
- [5] NOVOTNÝ, Ivan a Michal HRUŠKA. *Biologie člověka*. 3. rozš. a upr. vyd. Praha: Nakladatelství Fortuna, 1995. ISBN 80-7168-819-3.
- [6] ČIHÁK, Radomír. *Anatomie*. Třetí, upravené a doplněné vydání. Ilustroval Ivan HELEKAL, ilustroval Jan KACVINSKÝ, ilustroval Stanislav MACHÁČEK. Praha: Grada, 2016. ISBN 978-80-247-5636-3.
- [7] HLOŽÁNEK, Martin a Blanka BRŮNOVÁ. *Přístrojová technika v oftalmologii*. Praha: Univerzita Karlova, 2. lékařská fakulta, 2006. ISBN 80-902160-9-9.
- [8] K. KAISER, Peter. *The Joy of Visual Perception* [online]. York: York University, c1996-2005 [cit. 2016-12-28]. Dostupné z: <http://www.yorku.ca/eye/ophthal.htm>
- [9] CORDERO, Ismael, 2016. *Understanding and caring for an indirect ophthalmoscope*. 2016. ISSN 09536833.
- [10] SAINE, Patrick J. a Marshall E. TYLER. *Ophthalmic Photography: Retinal Photography, Angiography, and Electronic Imaging. 2nd Edition*. Butterworth-Heinemann, 2002. ISBN 0750673729.
- [11] TORNOW, Ralf-Peter, Jan ODSTRČILÍK a Radim KOLAR, 2018. Time-resolved quantitative inter-eye comparison of cardiac cycle-induced blood volume changes in the human retina. *Biomedical Optics Express* [online]. ISSN 2156-7085. Dostupné z: doi:10.1364/boe.9.006237
- [12] TORNOW, Ralf-Peter, Radim KOLAR a Jan ODSTRČILÍK, 2015a. Ophthalmoscope measures fast temporal changes in the human retina. *SPIE Newsroom* [online]. Dostupné z: doi:10.1117/2.1201509.006061
- [13] TORNOW, Ralf P., Radim KOLÁŘ a Jan ODSTRČILÍK, 2015b. Non-mydratic video ophthalmoscope to measure fast temporal changes of the human retina. In: *Novel Biophotonics Techniques and Applications III* [online]. Dostupné z: doi:10.1117/12.218118
- [14] STERGIOU, Christos a Dimitrios SIGANOS. *Neural networks* [online]. [cit. 2018-12-23]. Dostupné z:

- https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html#What%20is%20a%20Neural%20Network
- [15] INSTITUT BIOSTATISTIKY A ANALÝZ MASARYKOVY UNIVERZITY. *Matematická biologie* [online]. [cit. 2018-12-23]. Dostupné z: <http://portal.matematickabiologie.cz/>
- [16] Miikkulainen R. (2011) Topology of a Neural Network. In "Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA
- [17] Artificial Neural Network Tutorial. Tutorials point simply easy learning [online]. [cit. 2018-12-23]. Dostupné z: https://www.tutorialspoint.com/artificial_neural_network
- [18] CHALUPNÍK, Vitalij. Biologické algoritmy: *Neuronové sítě* [online]. 2012 [cit. 2018-12-29]. Dostupné z: <https://www.root.cz/clanky/biologicke-algoritmy-5-neuronove-site/>
- [19] CHANGHAU, Isaac. *Loss Functions in Neural Networks* [online]. 2017 [cit. 2018-12-29]. Dostupné z: https://isaacchanghau.github.io/post/loss_functions/
- [20] AGRAWAL, Apoorva. *Loss Functions and Optimization Algorithms*. Demystified. Data Science Group, IITR [online]. 2017 [cit. 2018-12-29]. Dostupné z: <https://medium.com/data-science-group-iitr/loss-functions-and-optimization-algorithms-demystified-bb92daff331c>
- [21] SINGH WALIA, Anish. *Loss Functions and Optimization Algorithms*. Demystified. Towards Data Science [online]. 2017 [cit. 2018-12-29]. Dostupné z: <https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-95ae5d39529f>
- [22] The data science blog: *An Intuitive Explanation of Convolutional Neural Networks* [online]. 2016 [cit. 2018-12-23]. Dostupné z: <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>
- [23] KARPATY, Andrej. *Neural Networks Part 1: Setting Up the Architecture*. Notes for CS231n Convolutional Neural Networks for Visual Recognition, Stanford University. 2015. [online]. [cit. 2018-12-27]. Dostupné z: <http://cs231n.github.io/convolutional-networks/>
- [24] ESCONTELA, Alejandro. *Convolutional Neural Networks from the ground up*. Towards Data Science [online]. 2018 [cit. 2018-12-28]. Dostupné z: <https://towardsdatascience.com/convolutional-neural-networks-from-the-ground-up-c67bb41454e1>
- [25] SINGHAL, Harsh. *Convolutional Neural Network with TensorFlow implementation*. Data Science Group, IITR [online]. 2017 [cit. 2018-12-28]. Dostupné z: <https://medium.com/data-science-group-iitr/building-a-convolutional-neural-network-in-python-with-tensorflow-d251c3ca8117>
- [26] SARKAR, Kanchan. *ReLU : Not a Differentiable Function: Why used in Gradient Based Optimization? and Other Generalizations of ReLU*. Data Science Group, IITR [online]. 2018 [cit. 2018-12-28]. Dostupné z:

- <https://medium.com/@kanchansarkar/relu-not-a-differentiable-function-why-used-in-gradient-based-optimization-7fef3a4cecec>
- [27] HOLLEMANS, Matthijs. Convolutional neural networks on the iPhone with VGGNet. 2016. [Online]. [cit.2018-12-28]. Dostupné z: <http://machinethink.net/blog/convolutional-neural-networks-on-the-iphone-with-vggnet/>
- [28] RAMSUNDAR, Bharath a Reza Bosagh ZADEH. *TensorFlow for deep learning: from linear regression to reinforcement learning*. Beijing: O'Reilly Media, 2018. ISBN 1491980451.
- [29] SRIVASTAVA, Nitish, Geoffrey HINTON, Alex KRIZHEVSKY, Ilya SUTSKEVER a Ruslan SALAKHUTDINOV, 2004. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* [online]. ISSN 13871811. Dostupné z: doi:10.1016/j.micromeso.2003.09.025
- [30] KRIZHEVSKY, Alex, Ilya SUTSKEVER a Geoffrey E HINTON, 2012. ImageNet Classification with Deep Convolutional Neural Networks. In: *ImageNet Classification with Deep Convolutional Neural Networks*.
- [31] PRECHELT, Lutz, 2012. Early stopping - But when? *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* [online]. ISSN 03029743. Dostupné z: doi:10.1007/978-3-642-35289-8
- [32] JAN, Jiří. *Medical image processing, reconstruction, and restoration: concepts and methods*. Boca Raton, FL: Taylor & Francis, 2006. ISBN 0-8247-5849-8.
- [33] AHAMED, Akyas T U, Abhin JOTHISH, Geo JOHNSON a Santhosh B V KRISHNA. Automated System for Retinal Vessel Segmentation. In: *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* [online]. IEEE, 2018, 2018, s. 717-722 [cit. 2018-12-23]. DOI: 10.1109/ICICCT.2018.8473333. ISBN 978-1-5386-1974-2. Dostupné z: <https://ieeexplore.ieee.org/document/8473333/>
- [34] DASH, Jyotiprava a Nilamani BHOI, 2018. Retinal blood vessel segmentation using Otsu thresholding with principal component analysis. In: *Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018* [online]. ISBN 9781538608074. Dostupné z: doi:10.1109/ICISC.2018.8398938
- [35] KARULE, P. T. a Shilpa JOSHI, 2018. Blood vessels segmentation using thresholding approach for fundus image analysis. In: *Proceedings of 2017 International Conference on Intelligent Computing and Control, I2C2 2017* [online]. ISBN 9781538603741. Dostupné z: doi:10.1109/I2C2.2017.8321965
- [36] SOOMRO, Toufique A., Manoranjan PAUL, Junbin GAO a Lihong ZHENG, 2018. Retinal blood vessel extraction method based on basic filtering schemes. In: *Proceedings - International Conference on Image Processing, ICIP* [online]. ISBN 9781509021758. Dostupné z: doi:10.1109/ICIP.2017.8297118
- [37] NUGROHO, Hanung Adi, Rezty Amalia ARAS, Tri LESTARI a Igi ARDIYANTO,

2017. Retinal vessel segmentation based on Frangi filter and morphological reconstruction. In: *ICCREC 2017 - 2017 International Conference on Control, Electronics, Renewable Energy, and Communications, Proceedings* [online]. ISBN 9781538616673. Dostupné z: doi:10.1109/ICCEREC.2017.8226686
- [38]DASGUPTA, Avijit a Sonam SINGH, 2017. A fully convolutional neural network based structured prediction approach towards the retinal vessel segmentation. In: *Proceedings - International Symposium on Biomedical Imaging* [online]. ISBN 9781509011711. Dostupné z: doi:10.1109/ISBI.2017.7950512
- [39]J. Guo, S. Ren, Y. Shi and H. Wang. "Automatic Retinal Blood Vessel Segmentation Based on Multi-Level Convolutional Neural Network." *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Beijing, China, 2018, pp. 1-5. doi: 10.1109/CISP-BMEI.2018.8633115
- [40]Z. Feng, J. Yang and L. Yao. Patch-based fully convolutional neural network with skip connections for retinal blood vessel segmentation. *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, 2017, pp. 1742-1746. doi: 10.1109/ICIP.2017.8296580
- [41]RONNEBERGER, Olaf, Philipp FISCHER a Thomas BROX, 2015. U-net: Convolutional networks for biomedical image segmentation. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* [online]. ISBN 9783319245737. Dostupné z: doi:10.1007/978-3-319-24574-4_28
- [42]GAO, Xurong, Yiheng CAI, Changyan QIU a Yize CUI, 2018. Retinal blood vessel segmentation based on the Gaussian matched filter and U-net. In: *Proceedings - 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2017* [online]. ISBN 9781538619377. Dostupné z: doi:10.1109/CISP-BMEI.2017.8302199
- [43]POWERS, David M. W. a AILAB, 2007. Evaluation : From Precision , Recall and F-Factor to ROC , Informedness , Markedness & Correlation. *Journal of Machine Learning Technologies*.
- [44]NARKHEDE, Sarang. Understanding AUC - ROC Curve. *Towards Data Science* [online]. 26.6.2018 [cit. 2019-04-20]. Dostupné z: <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
- [45]CHICCO, Davide, 2017. *Ten quick tips for machine learning in computational biology* [online]. 2017. ISSN 17560381. Dostupné z: doi:10.1186/s13040-017-0155-3
- [46]OCO, Nathaniel, Leif Romeritch SYLIONGKA, Rachel Edita ROXAS a Joel ILAO, 2013. Dice's coefficient on trigram profiles as metric for language similarity. In: *2013 International Conference Oriental COCOSDA Held Jointly with 2013 Conference on Asian Spoken Language Research and Evaluation, O-*

- COCOSDA/CASLRE 2013* [online]. ISBN 9781479923786. Dostupné z: doi:10.1109/ICSDA.2013.6709892
- [47]Keras Documentation. [online]. [cit. 2018-12-28]. Dostupné z: <https://keras.io/>
- [48]TensorFlow. An open-source software library for Machine Intelligence. [online]. [cit. 2018-12-28]. Dostupné z: <https://www.tensorflow.org>
- [49]ZHI XU, Hao. Implementation of deep learning framework – Unet, using Keras. [online]. [cit. 2019-05-15]. Dostupné z: <https://github.com/zhixuhao/unet>
- [50]BANTAN, Malak T. Auto-segmentation of retinal blood vessels using image processing. In: *2016 4th Saudi International Conference on Information Technology (Big Data Analysis) (KACSTIT)* [online]. IEEE, 2016, 2016, s. 1-6 [cit. 2019-05-09]. DOI: 10.1109/KACSTIT.2016.7756068. ISBN 978-1-4673-8956-3.
- [51]KUMAR, Debamita, Ankita PRAMANIK, Sudeshna Sil KAR a Santi P. MAITY. Retinal blood vessel segmentation using matched filter and Laplacian of Gaussian. In: *2016 International Conference on Signal Processing and Communications (SPCOM)* [online]. IEEE, 2016, 2016, s. 1-5 [cit. 2019-05-09]. DOI: 10.1109/SPCOM.2016.7746666. ISBN 978-1-5090-1746-1
- [52]ANNUNZIATA, Roberto, Andrea GARZELLI, Lucia BALLERINI, Alessandro MECOCCI a Emanuele TRUCCO. Leveraging Multiscale Hessian-Based Enhancement With a Novel Exudate Inpainting Technique for Retinal Vessel Segmentation. *IEEE Journal of Biomedical and Health Informatics* [online]. 2016, 20(4), 1129-1138 [cit. 2019-05-13]. DOI: 10.1109/JBHI.2015.2440091. ISSN 2168-2194
- [53]BUDAI, A., R. BOCK, A. MAIER, J. HORNEGGER a G. MICHELSON. Robust Vessel Segmentation in Fundus Images. *International Journal of Biomedical Imaging* [online]. 2013, 1-11 [cit. 2019-05-15]. DOI: 10.1155/2013/154860
- [54]CHENG, Erkang, Liang DU, Yi WU, Ying J. ZHU, Vasileios MEGALOOIKONOMOU a Haibin LING. Discriminative vessel segmentation in retinal images by fusing context-aware hybrid features. *Machine Vision and Applications* [online]. 2014, 25(7), 1779-1792 [cit. 2019-05-15]. DOI: 10.1007/s00138-014-0638-x. ISSN 0932-8092

POUŽITÉ ZKRATKY

ACC	Accuracy
Adam	Adaptive Moment Estimation
AUC	Area Under the ROC curve
CCD	charge-coupled device
CLAHE	Contrast-limited adaptive histogram equalization
CMOS	Complementary Metal–Oxide–Semiconductor
CNN	Convolutional Neural Network
CPU	central processing unit
GPU	graphics processing unit
KL	Kullback Liebler divergence
LED	Light-Emitting Diode
MSE	mean squared error
ReLU	rectified linearity unit
RGB	barevný model červená-zelená-modrá, který slouží pro aditivní míchání barev
ROC	Receiver Operating Characteristic
SGD	Stochastic Gradient Descent

SEZNAM PŘÍLOH

Příloha 1: Obsah přiloženého CD

PŘÍLOHA 1: OBSAH PŘILOŽENÉHO CD

1. Diplomová práce (formát .pdf)
2. Složka „Programy“
 - tranning.py
 - testing.py
 - model.py
 - data_generator.py
 - statistic_metrics.py
 - ROC_curve_AUC.py
 - fundus_preprocessing.m