



Dokumentace k projektu BIO

Detekce krevních sraženin ve snímku sítnice oka

Autoři: Bc. Lukáš Pelánek, xpelan03

Bc. Hana Prostředníková, xprost01

Datum odevzdání: 20. 11. 2017

1 Zadání

Cílem tohoto projektu je vytvořit program, který automaticky detekuje krevní sraženiny ve snímku sítnice oka. V tomto projektu jsme se zaměřili konkrétně na detekci hemoragií a mikroaneuryzmat. Snímky sítnic byly čerpány z databáze Diaret [1].

2 Studium problematiky

Diabetická retinopatie je označení pro patologické změny sítnice a jejích cév v souvislosti s cukrovkou. Jedná se o nejčastější cévní onemocnění sítnice. Vzniká v důsledku specifických morfologických změn způsobených chronickou hyperglykemií. Jedná se o závažnou komplikaci diabetu, která může vyústit až v slepotu.

Toto onemocnění lze pozorovat jako degenerativní cévní změny na sítnici. Projevuje se v podobě mikroaneuryzmat, hemoragií či nahromaděných shluků tuků a proteinů. Včasná detekce těchto změn může přispět k potlačení projevů a ke zpomalení postupu onemocnění [5].

2.1 Mikroaneuryzmata

Mikroaneuryzmata jsou prvním prokazatelným projevem diabetické retinopatie. Mikroaneuryzmata se jeví jako červené tečky, které jsou velké 10 - 100 mikrometrů. Jde o dilatace nebo výchlípky ve středních vrstvách kapilár sítnice. Vznik souvisí s odumíráním pericytů [5].

2.1 Hemorágie

Hemorágie se objevují jako následek prasknutí mikroaneuryzmat, vlásečnic a nebo žil. Rozlišujeme více druhů hemoragií. Jde o drobné kulaté červené tečky, které se zpravidla vyskytují na úrovni vlásečnic. Mohou také vypadat jako světle červená až tmavá skvrna [5].

3 Použité technologie

Jako implementační jazyk jsme zvolili Python pro jeho širokou škálu dostupných knihoven. Pro zpracování vstupního obrazu jsme využili knihovnu OpenCV [2]. Jde o otevřenou multiplatformní knihovnu pro práci s obrazem. Pro interní reprezentaci obrazu jsme využili knihovnu NumPy [3], která s OpenCV úzce souvisí.

4 Implementace

Prvním krokem implementace bylo zvolit postup řešení. Problematiku jsme se rozhodli řešit pomocí matematických morfologických operací.

Dalším krokem bylo najít relevantní vědecké články, které nám pomohou se s problematikou důkladně seznámit a navést k řešení. Bohužel drtivá většina článků obsahovala pouze teoretické poznatky, a tak jsme byli nuceni experimentovat. Avšak výsledky nebyly příliš uspokojivé.

Nakonec jsme se rozhodli vycházet z bakalářské práce [4], která byla zaměřena na toto téma. Výsledky snažení jsou popsány v kapitole 4.1. S těmito výsledky jsme ale nebyli zcela spokojeni, a proto jsme se rozhodli zkusit jiný přístup, který je popsán v kapitole 4.2 a 4.3.

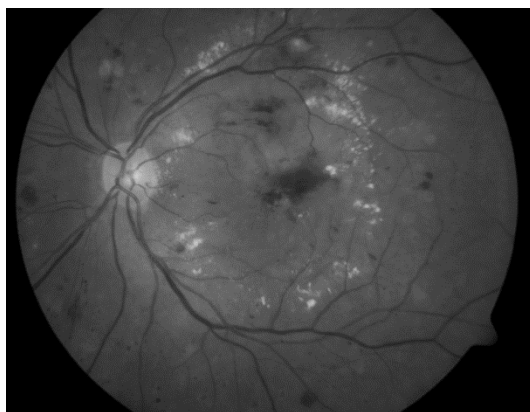
4.1 Algoritmus 1



Obr. 1 Vstupní obraz

První algoritmus je implementován v souboru `hemorrhages2.py`. Jako vstup očekává obrázek s názvem `image.png`. Výstup je pak generován do souboru `result.png`. Vstupní obraz (obr. 1) bylo potřebné před samotnou detekcí upravit, aby byly patologické změny co nejvíce výrazné.

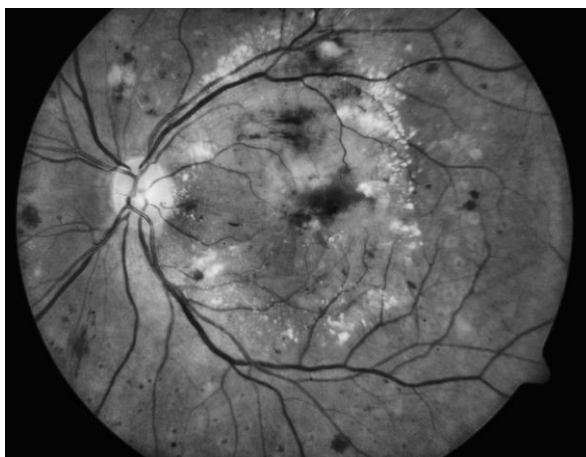
Vstupní obraz byl tedy vhodně předzpracován. První úpravou byla extrakce zeleného



kanálu vstupního obrazu. Na zeleném kanálu lze nejlépe pozorovat strukturu sítnice (obr. 2).

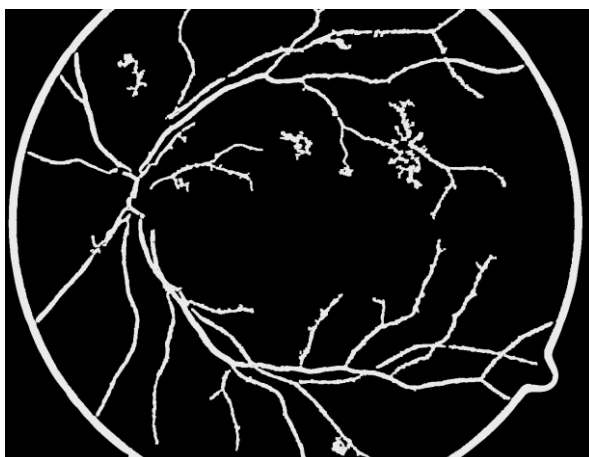
Obr. 2 Extrahovaný zelený kanál

Na takto upravený obraz bylo aplikováno Gaussovo rozmazání. Díky této úpravě dojde ke splynutí velmi malých oblastí s pozadím. Pokud by zůstaly takové oblasti v obrazu, byly by nesprávně vyhodnoceny jako samostatné oblasti v algoritmu *region growing* a nepříznivě by ovlivnily detekci. Dále je potřeba počítat se skutečností, že každý snímek sítnice byl pořízen při jiných světelných podmínkách. Proto je potřeba použít vhodnou ekvalizaci histogramu, která normalizuje jas a zvýší kontrast obrazu (obr. 3). V projektu jsme pro ekvalizaci použili CLAHE.



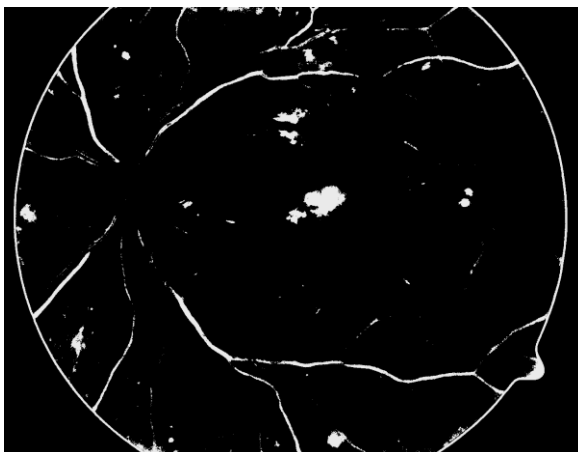
Obr. 3 Obraz po zvýšení kontrastu

Po této úpravě jsou hemoragie o poznání zřetelnější. Dále je potřeba extrahovat z obrazu žíly. Extrakce byla provedena použitím algoritmu *region growing* (obr. 4). Jak lze vidět, kromě žil bylo nesprávně detekováno i oční pozadí. Experimentováním jsme však došli k tomuto výsledku, který se jevil jako nejlepší, tedy došlo k detekci většiny žil a jen k poměrně malému množství chybně detekovaného šumu.



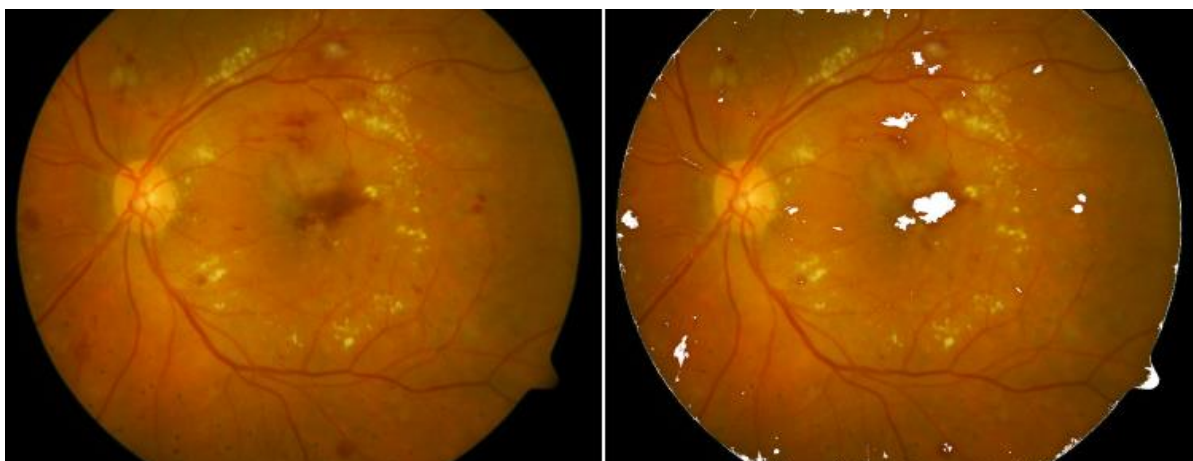
Obr. 4 Extrahované žíly

Nyní již zbývá detekovat hemorágie. K tomu jsme opět využili algoritmus region growing, pomocí kterého jsme detekovali oblasti s barvou tmavší než určitý práh. Tento práh byl nastaven na $\frac{1}{5}$ nejsvětlejší barvy v obrazu. Výsledek detekce je na obr. 5.



Obr. 5 Detekované hemoragie

Jak je vidět, algoritmus nesprávně detekoval některé žíly jako hemorágie. Proto z tohoto obrazu odstraníme již předem detekované žíly. Výsledek zakreslíme do původního obrazu. Tím je detekce krevních sraženin u konce (obr. 6).



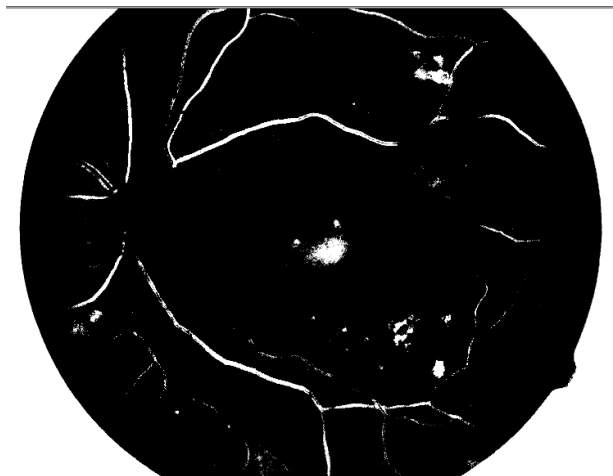
Obr. 6 Porovnání vstupního obrazu a výstupu detekce.

Z obrázku 6 je patrné, že při detekci byly nesprávně detekovány okraje obrazu, které se jevily dostatečně tmavé na to, aby byly rozeznány jako hemoragie. Výsledek byl uspokojivý, no přesto jsme se rozhodli vyzkoušet i jiný přístup.

4.2 Algoritmus 2

Druhý algoritmus je implementován v souboru `hemorrhages_final.py`. Jako vstup očekává obrázek s názvem `image.png`. Výstup je pak generován do souboru `hemorrhages.png`.

Postupovali jsme velmi podobně jako v prvním algoritmu. Rozdíl oproti algoritmu 1 spočívá také v tom, že tento algoritmus detekuje pouze hemorágie. Provedli jsme extrakci zeleného kanálu, který jsme podrobili ekvalizaci histogramu. Obraz jsme následně podrobili prahování, kdy pixely s `thresholdem` větším než 40 byly odstraněny (takto světlé objekty nemohou být hemorágie) (obr. 7).



Obr. 7 Výsledek prahování

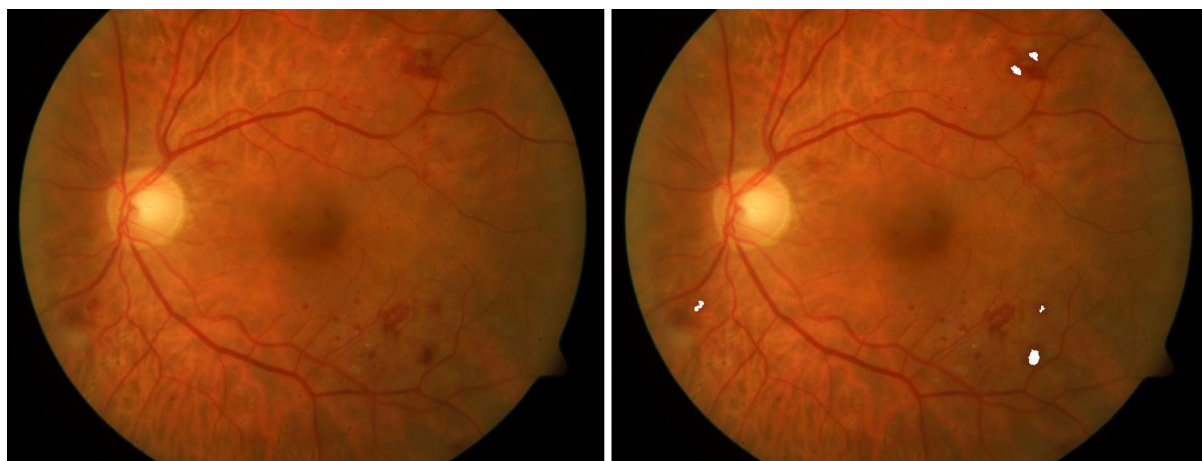
Následně jsme provedli morfologický opening, abychom se zbavili některých žil. Dalším krokem bylo najít kontury a tyto kontury následně vyfiltrovat a zakreslit do obrazu. Kontury byly filtrovány dle následujících kritérií:

- Poměr šířky a výšky byl mezi 0,6 a 1,4
- Výška byla mezi 15px a 70px
- Šířka byla mezi 15px a 70px



Obr. 8 Vyfiltrované kontury

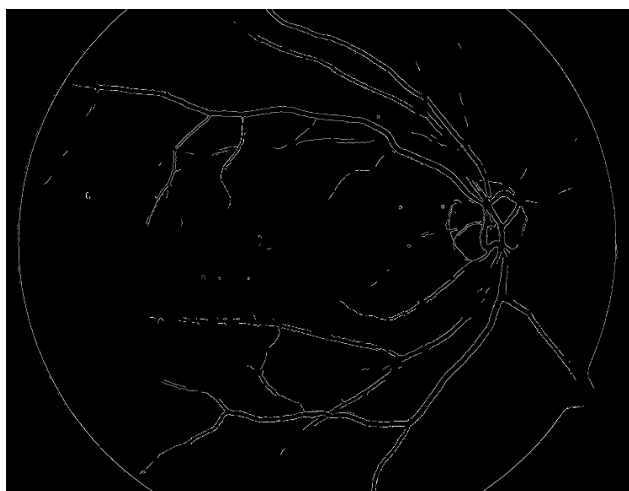
Mezi nalezenými konturami jsou patrné i stopy po některých žilách (Obr. 8). Proto provedeme detekci žil z předchozího algoritmu a odstraníme je.



Obr. 9 Porovnání vstupního obrazu a výsledku detekce

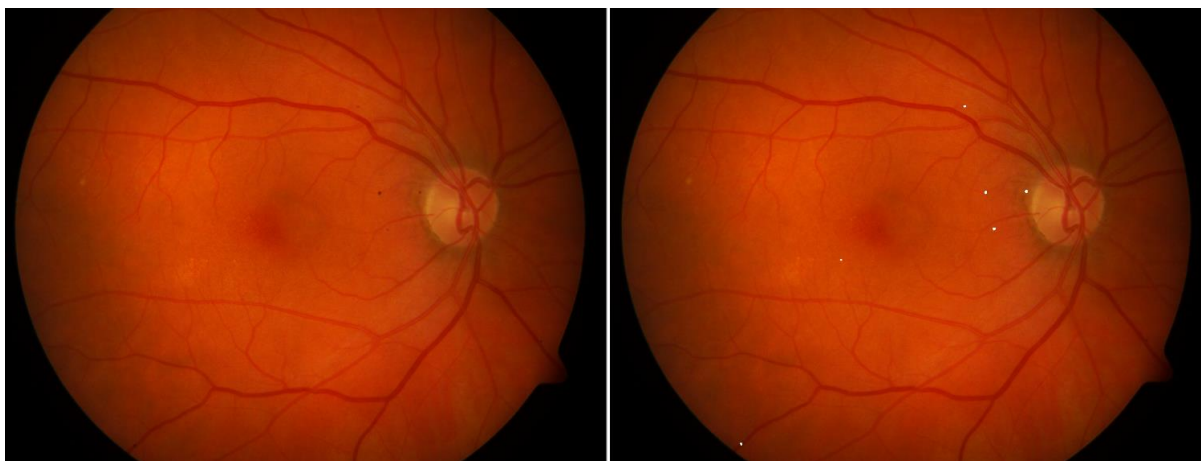
4.3 Algoritmus 3

Třetí algoritmus je zaměřen na detekování mikroaneuryzmat. Je implementován v souboru `microaneurysm_final.py`. Jako vstup očekává obrázek s názvem `image.png`. Výstup je pak generován do souboru `result.png`. Vstupní obrázek je nejprve konvertován do odstínu šedi a následně je aplikováno mediánové rozmazání s velikostí jádra 5x5. Jako další část je detekce hran metodou Canny (obr. 10).



Obr. 10 Detekce hran metodou Canny

Jako další krok je detekce kontur a jejich vyplnění. Všechny kontury, které jsou spojené, tzn. tvoří uzavřený objekt a jejich velikost je menší než 30 jsou vyplněny. Posledním krokem je morfologický opening, který nám zajistí, že v obrazu zůstanou pouze tyto vyplněné kontury (nalezené mikroaneuryzmy).



Obr. 11 Porovnání vstupního obrazu a výsledku detekce

4.4 Region growing

Region growing je jednoduchý algoritmus sloužící k segmentaci obrazu. Algoritmus vychází z výchozího bodu a sdružuje podobné pixely dle určité vlastnosti do regionu. Následně je obraz rozdělen do regionů, kde každý pixel musí patřit do určitého regionu, ale nesmí patřit do více než jednoho regionu.

5 Vyhodnocení

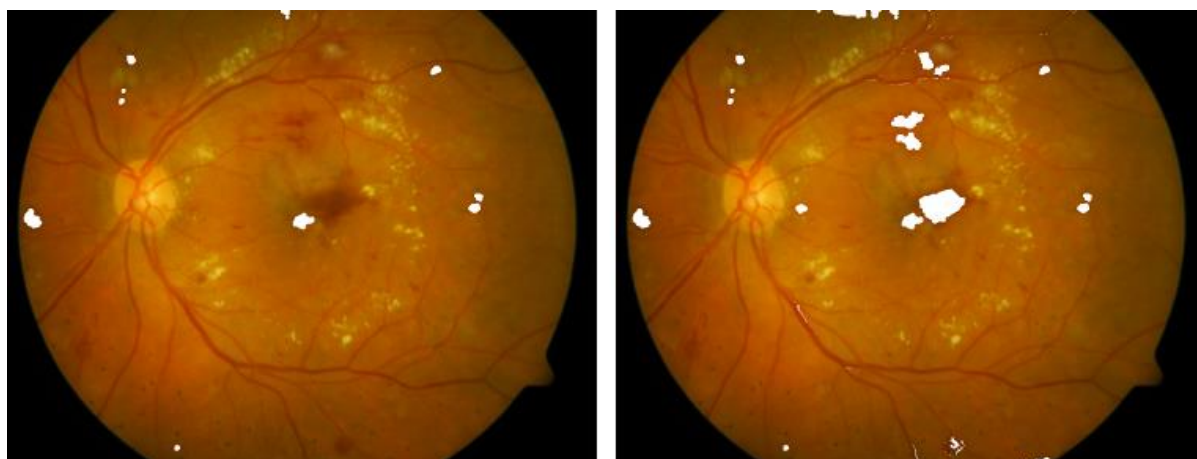
Na obrázku 13 a 14 jsou zobrazeny výsledky použití všech uvedených algoritmů na dvou různých snímcích sítnice.

Algoritmus 1 byl implementován tak aby detekoval jak mikroaneuryzmata tak hemoragie. Z obrázků je patrné, že dosahuje poměrně uspokojivé výsledky. Nevýhodou však je poměrně velký šum, který vzniká chybnou detekcí tmavších míst. Chybná detekce je nejvýraznější v oblasti okrajů obrazu, kde je oční pozadí rozeznáno jako krevní sraženina.

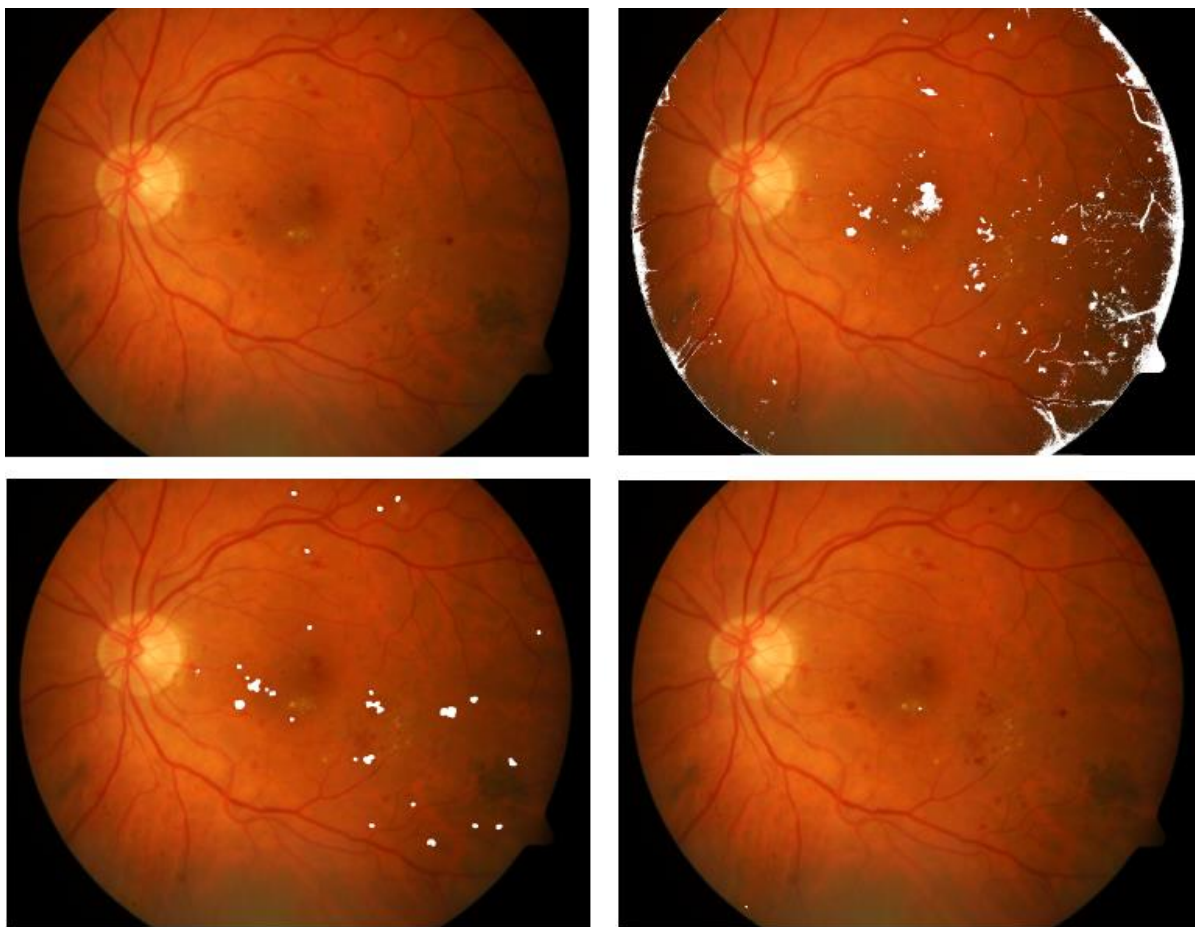
Algoritmus 2 byl implementován pouze pro detekci hemoragií. Chybná detekce byla výrazně potlačena. Změnou parametrů, které byly nastaveny pro velikost hemorágie lze dosáhnout ještě lepších výsledků (obr. 12), ale za cenu toho, že budou nesprávně detekovány i jiné objekty.

Algoritmus 3 byl implementován pro detekci mikroaneuryzmat. Především na obrázku 14 je patrná chybná detekce světlých lézí.

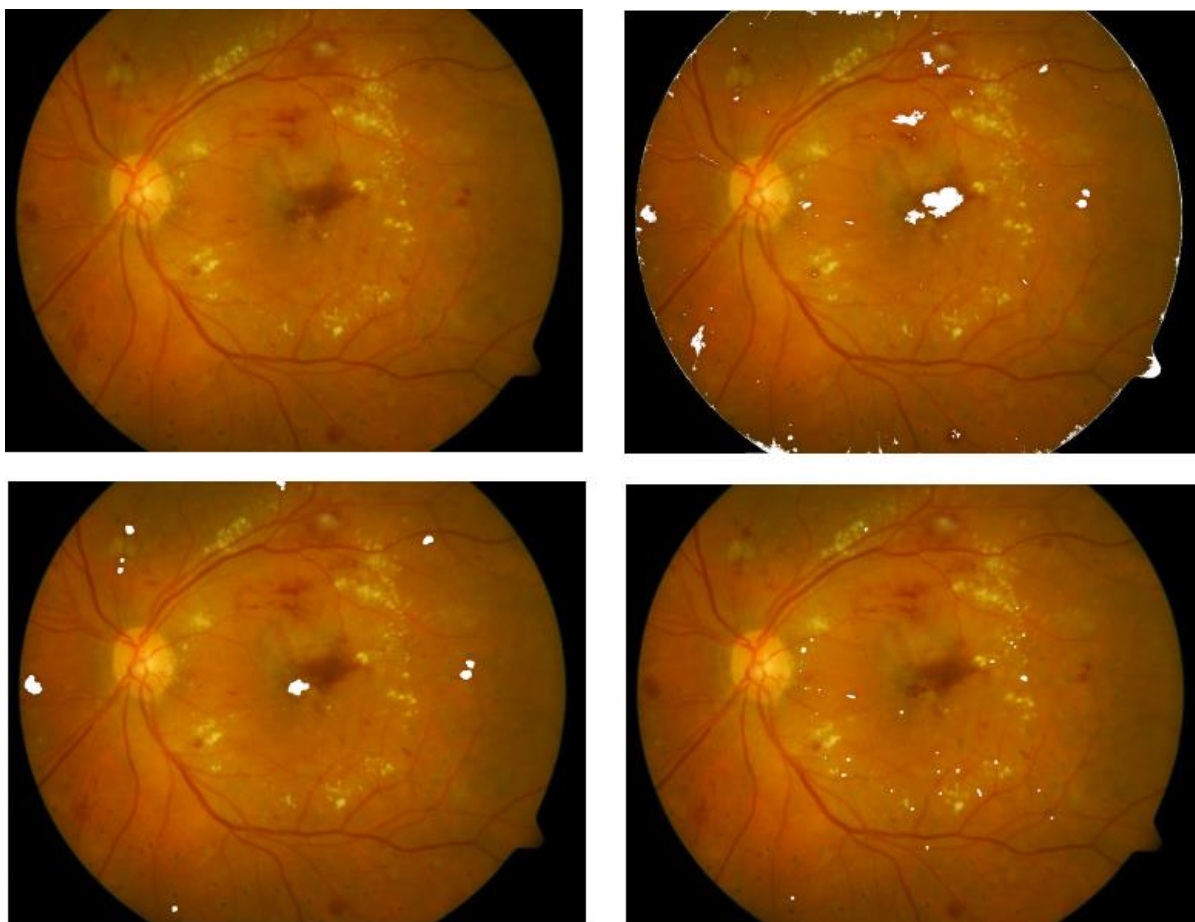
Každý přístup má své klady a své zápory. Detekce není v žádném z přístupů zcela stoprocentní, no i přes to dosahuje uspokojivých výsledků. Chybná detekce se ve většině případů týká pouze okrajových oblastí. Pravdou je, že zůstávají nedetekované některé sraženiny. Důvodem je různá barevná intenzita snímků a tedy pro každý platí různé hodnoty filtrovacích parametrů. Vzhledem k časovým možnostem, který jsme na daný problém měli, jsme s výsledky převážně spokojeni.



Obr. 12 Porovnání detekce hemoragií při změně parametrů



Obr. 13 V prvním řádku porovnání vstupního obrazu a detekce algoritmem 1.
V druhém řádku detekce hemoragií algoritmem 2 a detekce mikroaneuryzmat algoritmem 3.



Obr. 14 V prvním řádku porovnání vstupního obrazu a detekce algoritmem 1. V druhém řádku detekce hemoragií algoritmem 2 a detekce mikroaneuryzmat algoritmem 3.

Bibliografie

- [1] *IMAGERET: Optimal Detection and Decision-Support Diagnosis of Diabetic Retinopathy* [online]. [cit. 2017-11-14]. Dostupné z: <http://www.it.lut.fi/project/imageret/>
- [2] *OpenCV* [online]. [cit. 2017-11-18]. Dostupné z: <https://opencv.org/>
- [3] *NumPy* [online]. [cit. 2017-11-18]. Dostupné z: <http://www.numpy.org/>
- [4] KOŠTIALIK, Daniel. *Detekce chorob v snímku sítnice oka*. Brno, 2017. Dostupné z: <http://www.fit.vutbr.cz/study/DP/BP.php?id=19892>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. 2017-06-14. Vedoucí práce Maruniak Lukáš.
- [5] TOBIÁŠOVÁ, N. *Detekce mikroaneuryzmat a hemoragií ve snímcích sítnice*. Brno: