

1 Úvod

Projekt je zaměřený na detekci krevních sraženin ve snímku sítnice oka. Na snímku sítnice hledá krevní sraženiny (hemoragie a aneurysma) a ve výsledném snímku je graficky označí. Snímky sítnice jsme získali z veřejně dostupných databází.

2 Rozbor tématu a použitých prostředků

Krevní sraženiny na sítnici vznikají často u nemocných lidí, např. při diabetes mellitus. Automatická detekce těchto známek by tedy mohla být účinným prostředkem pro včasné odhalení podobných nemocí, pokud by se prováděla při každé oční prohlídce, kde lékař dané jevy nevyhledává.

Všechna testovací data jsme čerpali z veřejně dostupných databází, jako jsou projekty IMAGERET¹, DRIVE² a STARE³. V nich lze nalézt obrázky sítnic s krevními sraženinami a dalšími poškozeními, např. světlými lézemi.

Program jsme vytvořili v jazyce C++ za použití standardních knihoven a knihovny OpenCV⁴, která slouží k práci s obrazem (počítačové vidění) a usnadňuje tak úpravu snímků sítnic a detekci různých komponent v nich.

Většinu použitých informací a algoritmů jsme čerpali z článků dostupných v IEEE digital library⁵, které byly doporučeny v zadání projektu. Postupy použité pro úpravy obrázku před detekcí jsme čerpali především z článků [1], [2] a [3]. Části těchto dokumentů však nebyly pro naše účely dostatečně konkrétně popsány nebo jejich náročnost (nejen časová) silně převyšovala rozsah projektu, proto jsme se těchto postupů nedrželi striktně a nahradili libovolné části vlastními kroky nebo se inspirovali v jiných člancích.

Články [1] a [2] se zabývají nejen detekcí krevních sraženin, ale také světlých lézí, optického disku (slepé skvrny), fovea centralis (žluté skvrny) a krevního řečiště. Pro účely projektu jsme detekci těchto částí neimplementovali.

3 Postup a implementace algoritmu

Snímek sítnice nejprve načteme v barevném spektru RGB. Jelikož část vzorových snímků obsahuje nedokonalosti v kvalitě obrazu kvůli chybám či nevhodným podmínkám při nasnímání, snažíme se tyto chyby minimalizovat úpravami obrazu.

Nejprve obraz převedeme z RGB do formátu LAB, kde složka L reprezentuje údaj o intenzitě/jasu daného pixelu. Poté na tuto složku aplikujeme variantu ekvalizace histogramu, metodu CLAHE (contrast limited adaptive histogram equalization). Tímto se docílí toho, že se srovnají nerovnoměrnosti v světlosti/tmavosti hůře nasnímaných částí obrazu a zároveň se zvýší kontrast mezi tmavými a světlými místy. To vede k snadnější viditelným krevním sraženinám a cévám.

¹ <http://www.it.lut.fi/project/imageret/>

² <https://www.isi.uu.nl/Research/Databases/DRIVE/>

³ <http://cecas.clemson.edu/~ahoover/stare/>

⁴ <https://opencv.org/>

⁵ <https://www.vutbr.cz/uk/eiz/database/ieee>

Poté obrázek převedeme zpět do spektra RGB. Snímky sítnic jsou obecně převážně červené barvy, což zastírá rozdíly mezi různými částmi obrazu. Zaměřili jsme se tedy na zelené spektrum obrazu, kde jsou rozdíly jasněji viditelné. Proto RGB složku pro červenou a modrou vynulujeme a zůstane pouze požadovaná zelená verze obrazu.

Na tento snímek dále aplikujeme úpravu sytosti barvy (modifikace saturace), aby se ještě zvětšily rozdíly mezi světlými (sítnice, světlé léze, slepá skvrna) a tmavými částmi (krevní sraženiny, krevní řečiště, žlutá skvrna).

Snímek poté převedeme z RGB formátu na černobílý obraz, aby se s výsledkem snáze pracovalo. Obraz nyní obsahuje pouze jeden barevný kanál, který udává hodnotu černé barvy (0 – černá, 255 - bílá).

Díky úvodní aplikaci CLAHE algoritmu je sice kontrast mezi dílčími oblastmi vysoký, ale některé hrany jsou příliš ostré a místy se vyskytují pixely neodpovídající svému okolí. Uplatnili jsme proto mírné rozmazání obrazu tzv. box blur filtrem, který průměruje hodnoty okolních pixelů. Zkoušeli jsme také úpravu rozmazání Gaussovým filtrem, ale nedařilo se nám dosáhnout dobrých výsledků.

Následně černobílý obraz upravíme pomocí operací „erosion“ a „dilation“. Erosion prochází obraz a zvýrazňuje tmavé části tak, že pro každý pixel nahradí jeho hodnotu nejtmavší hodnotou z lokálního okolí daného velikostí šablony. Provádí se dvakrát, a tím silně zvýrazní krevní sraženiny a cévy. Dilation funguje jako opak operace erosion, utlumí tedy méně výrazné rozdíly na pozadí sítnice, které by mohly vést k falešné detekci.

Výsledné krevní sraženiny nalézáme pomocí třídy SimpleBlobDetector z knihovny OpenCV, která umožňuje detekci objektů v obraze. Tyto objekty mají různé tvary a vlastnosti, které specifikujeme pomocí podle intenzity barvy (minimální a maximální hodnota černé), velikosti jejich plochy, tvarů (kruh, elipsa) a jejich parametrů (konvexnost, zaoblení, atp.).

Pozice nalezených útvarů jsou poté uloženy a nakonec vyznačeny pomocí žlutých kruhů na vstupní snímek.

4 Testování, výsledky a zhodnocení

Program jsme testovali na vzorcích ze sad obrázků zmíněných v úvodu. Algoritmus dokáže detekovat značné množství krevních sraženin. Nejlépe nalezne tmavé aneurysma, avšak stále je problematická lokalizace menších světlých hemoragií, popřípadě libovolných sraženin, které jsou světlejšího charakteru a bez viditelných hranic (okraje mizí pozvolně a tzv. do ztracena).

Problematické jsou také okraje některých snímků, kde kvalita bývá horší a obraz ztmavený a nejasný. Pozitivních výsledků paradoxně program dosahuje v některých tmavých oblastech, kde lidským okem sraženina na první pohled není kvůli snížené viditelnosti patrná, ale automatická detekce vadu úspěšně detekuje.

Algoritmus také označuje do značné míry tzv. falešné detekce (false-pozitive) v krevním řečišti v místech, kde se cévy větví nebo jsou výrazné oproti okolí. Tuto vadu je možné odstranit tak, že se napřed detekuje krevní řečiště a z výsledného seznamu nálezů se odstraní ty, které leží v oblasti cév. Tímto bychom ovšem ztratili možnost detekovat sraženiny v krevním řečišti, proto jsme tuto funkcionalitu ve výsledném programu nezahrnuli.

5 Návod k použití

Program lze přeložit na školním serveru merlin pomocí příkazu „make“ v kořenovém adresáři projektu, tedy na stejné úrovni, jako je soubor Makefile. Výsledek kompilace je umístěn ve složce bin/. Po kompilaci je možné spustit demonstraci aplikace, a to použitím příkazu „make demo“. Pro spuštění program vyžaduje jeden parametr, jenž označuje cestu ke vstupnímu souboru obsahující obrazová data (například ve formátu PNG). Příklad použití:

```
./Main <cesta k souboru>.png
```

Aplikace se ukončí stiskem libovolné klávesy po zobrazení výsledku.

Závěr

Z výsledků projektu vyplynulo, že nezanedbatelnou roli pro detekci hraje kvalita snímku a také že program je někdy schopen detekovat i krevní sraženiny v místech, kde na první pohled nejsou okem zřejmé.

Zdroje

[1] SINTHANAYOTHIN, C., JF BOYCE, TH WILLIAMSON, HL COOK, S LAL, E MENSAH a D USHER. *Automated detection of diabetic retinopathy on digital fundus images*. 2002. DOI: 11874425.

[2] SINTHANAYOTHIN, C., JF BOYCE, TH WILLIAMSON a HL COOK. Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images. *Br J Ophthalmol*. 1999. DOI: PMC1723142.

[3] JYOTHIS, Jose a Kuruvilla JINSA. Detection of Red Lesions and Hard Exudates in Color Fundus Images. *International Journal Of Engineering And Computer Science*. 2014, **3**(10), 8583-8588. DOI: PMC1723142. ISSN 2319-7242.