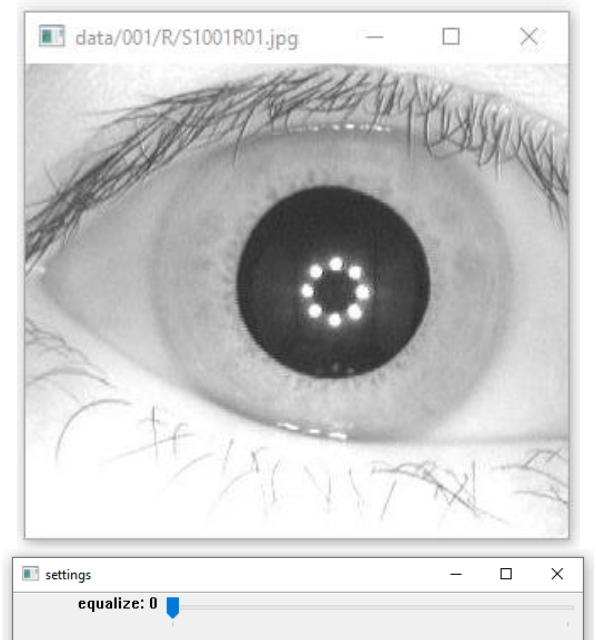
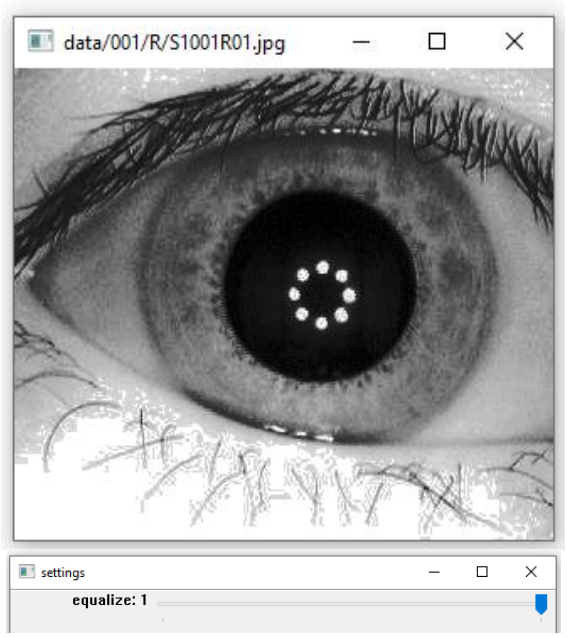
I-BIOM: Zadanie č.1

HĽADANIE BIOMETRIKY NA OBRAZE I

Prepínanie histogramovej ekvalizácie

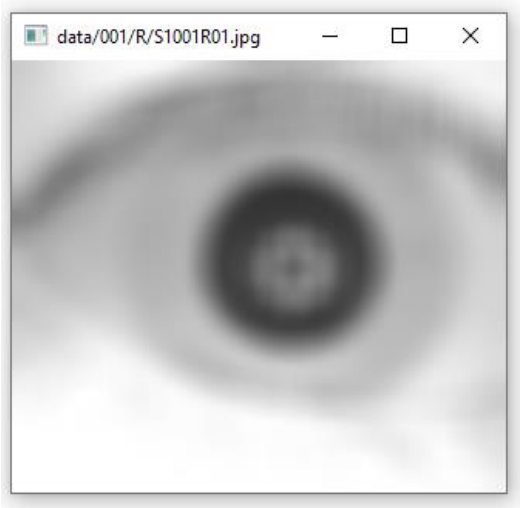
Prepínanie je riešené pomocou slidera, ktorý ma hodnoty 0 a 1.





Aplikáciu gaussovského rozmazania. Stupeň rozmazania nech je možné interaktívne meniť (veľkosť jadra aj parameter sigma)

Veľkosť jadrá musí byť kladná a nepárna. Zapínanie je taktiež riešené pomocou slidera. Guassovsské rozmazanie funguje aj bez ekvalizácie. zdroj



Veľkosť jadra = 25
Parameter sigma = 21
Gaussovské rozmazanie = 1 (zapnuté)

```
if (ksize %  2 ) != 0 and gauss == 1:
    gauss_iris = cv2.GaussianBlur(main_iris,(ksize,ksize), sigma)
    cv2.imshow(windowName, gauss_iris)
    gauss_defined = True
else:
    gauss_defined = False
```

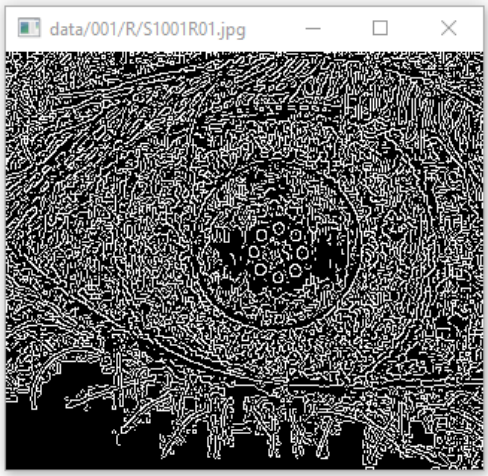
Dávid Gavenda 98533

slidery trackbary

<u>Aplikáciu Cannyho algoritmu na detekciu hrán. Treshold nech je možné interaktívne meniť.</u>

V prípade že je gaussovské rozmazanie aktívne, využije sa to. Rovnako platí aj v prípade ekvivalizácie.

Nastaviť sa dajú hodnoty *threshold1*, *threshold2* a *aperture*. zdroj



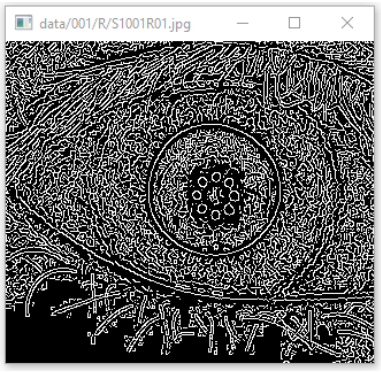
Bez gaussovského rozmazania

S gaussovským rozmazaním

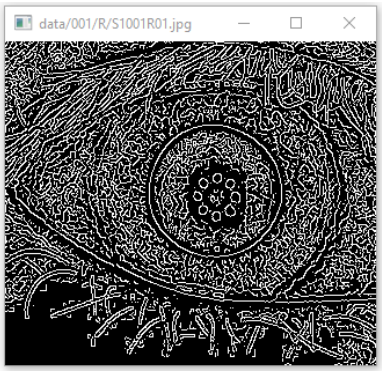
```
if canny:
    if gauss_defined:
        if aperture == 3 or aperture == 5 or aperture == 7:
            canny_iris = cv2.Canny(gauss_iris,threshold1,threshold2,apertureSize=aperture)
        else:
            canny_iris = cv2.Canny(gauss_iris,threshold1,threshold2)
        cv2.imshow(windowName, canny_iris)
    else:
        if aperture == 3 or aperture == 5 or aperture == 7:
            canny_iris = cv2.Canny(main_iris,threshold1,threshold2,apertureSize=aperture)
        else:
            canny_iris = cv2.Canny(main_iris,threshold1,threshold2)
        cv2.imshow(windowName, canny_iris)
```

Hodnoty pre *aperture* musia byť 3, 5 alebo 7. V inom prípade sa nepoužijú a sú teda defaultne 3.

aperture = 3



aperture = 5



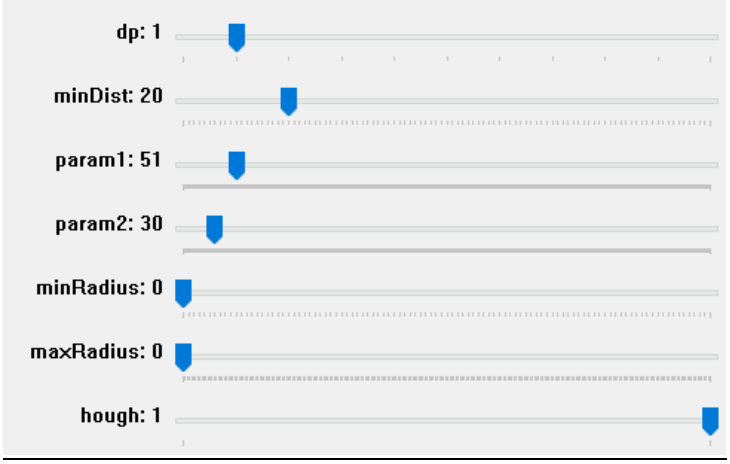
aperture = 7

Dávid Gavenda 98533

slidery trackbary

Aplikáciu Houghovej transformácie na nájdenie kruhov. Parametre nech je možné interaktívne meniť. Pri tomto algoritme je potrebné aby bolo možné nastaviť parametre rozlíšenie akumulátora, minimálna vzdialenosť centier kružníc, minimálny polomer/priemer kružníc, maximálny polomer/priemer kružníc (a prípadne aj parameter pre Cannyho detektor hrán a hranicu akumulátora pre detekciu kruhov)

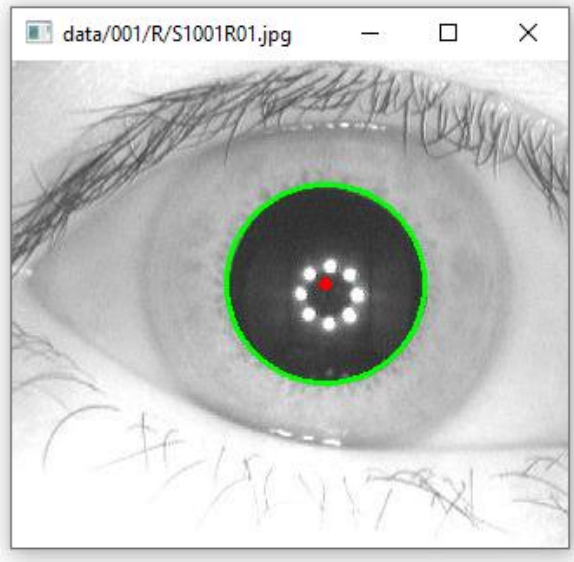
Dané hodnoty sa dajú nastaviť . zdroj



dp : Inverzný pomer k rozlíšeniu akumulátora
minDist : Minimálna vzdialenosť medzi stredmi kružníc
param1 : V tomto prípade, keďže využívam HOUGH_GRADIENT, je to threshold pre Canny edge detector
param2 : Accumulator threshold, čím je nižší, tým viac falošných kruhov sa detekuje
minRadius : Minimálny priemer kruhu
maxRadius : Maximálny priemer kruhu
hough : Vypínanie a zapínanie

Hodnoty dp, minDist, param1 a param2 musia byť kladné.

Veľkosť jadra = 5
Parameter sigma = 2
Threshold1 = 100
Threshold2 = 200
Aperture = 3
Dp = 1
MinDist = 20
Param1 = 50
Param2 = 30
Gauss, Canny a Hough



Taktiež bolo pridaný slider *spoiler*, ktorý nakreslí kružnice tak ako majú byť.

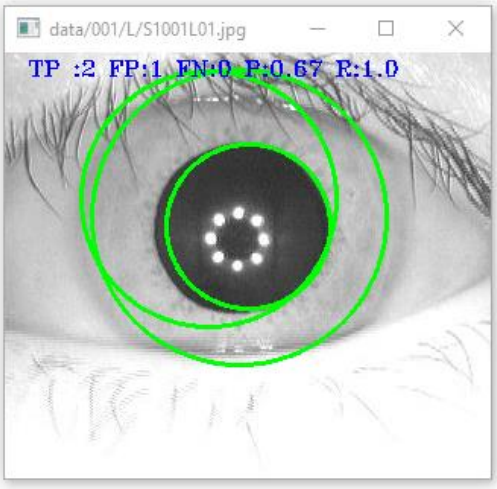
Dávid Gavenda 98533

slidery trackbary

<u>Vaším cieľom je nájsť vhodnú kombináciu parametrov, aby ste našli ohraničujúce kruhy pre dúhovku a zreničku. Doplňte do programu vyhodnocovanie úspešnosti tejto detekcie – porovnajte nájdené kruhy s anotáciami pomocou Intersection-over-Union 1.5b. Spočítajte Precision a Recall (threshold pre IoU=0.75) 1.5b. Vyskúšajte vaše programy na pár obrázkoch z dodanej databázy, vyhodnoťte, ktoré parametre by mohli viesť k dobrým výsledkom.</u>

IoU som vypočítaval pomocou štvorcov, ktorým bola kružnica vpísaná (zdroj). Ako správne určená sa brala len tá, ktorá presiahla IoU threshold. Pokiaľ malá priemer nad 70 tak sa kontrolovalo s dúhovkou, v opačnom prípade so zreničkou.

Equalize
dp = 1
minDist = 7
param1 = 148
param2 = 71
minRadius = 14
maxRadius = 155



Po dlhšom testovaní som sa k najlepšiemu výsledku dostal takémuto. Výsledok nie je bohužiaľ ideálny ale postačuje. Parametre spomenuté vyššie viedli k obstojnému výsledku.

Dávid Gavenda 98533

slidery trackbary

Aplikujte získané skúsenosti – postupne prejdite celú databázu, aplikujte (min.) Gaussovo rozmazanie a Hough. kružnice, optimálne parameter najdite pomocou mriežkového vyhľadávania a okolo hodnôt z predošlého bodu zadania (optimálne znamena s najlepšími hodnotami Precision a Recall pri IoU = 0.75). Má zmysel hľadať zreničku a dúhovku v samostatných cykloch (s inými nastaveniami Hough. transformácie) 2b.

Gridsearch bežal pre 27000 kombinácii a testoval sa na 10 obrázkoch z databázy. Obrázky sa vždy najskôr načítali, equalizovali, následne sa aplikovalo Gaussovo rozmazanie a potom Hough. kružnice, ktoré využívajú aj Canny edge.

```
for generated_maxRadius in range(0,201,100):
    for generated_minRadius in range(0,101,30):
        for generated_param2 in range(1,500,50):
            for generated_param1 in range(1,500,100):
                for generated_minDist in range(1,100,20):
                    for generated_sigma in range(2,9,3):
                        for generated_ksize in range(1,10,4):
                            for i in range(10): #Nejdem to bezat pre cely dataset kokso
```

Z údajov nám najlepší Recall a Precision vyšiel pre dané parametre

ksize	sigma	minDist	param1	param2	minRadius	maxRadius	Precision	Recall
9	2	1	101	51	0	0	1	0.9

Bohužiaľ medzi výsledkami boli taktiež priemerné najlepšie hodnoty pre zreničku a dúhovku.

averageBestIris	averageBestPupil
0.08	0.91

V prípade že boli dané parametre spustené pre celý dataset, výsledky boli nasledovné.

ksize	sigma	minDist	param1	param2	minRadius	maxRadius	averagePrecision	averageRecall	averageIoU	averageBestIris	averageBestPupil
9.0	2.0	1.0	101.0	51.0	0.0	0.0	0.82	0.81	0.83	0.3	0.79

A táto situácia sa opakuje pre väčšinu hodnôt. Najlepšie, kde boli zrenička aj dúhovka nad rozumnú IoU a zároveň je Precision a Recall v rozumnej úrovni. Môžeme si všimnúť že hodnoty pre Gaussovo rozmazanie ostali rovnaké, tak ako aj param1 a param2.

ksize	sigma	minDist	Param1	Param2	minRadius	maxRadius	Precision	Recall
9	5	1	101	51	30	200	0.79	1

averageBestIris	averageBestPupil
0.93	0.94

Celkovo ale po skontrolovaní si nemyslím že je ideálne zisťovať zároveň oboje a je lepšie zisťovať každé osobitne. Prišlo by mi to viac efektívne. Hlavne tieto hodnoty IoU boli pod 0.75

ksize	sigma	minDist	param1	param2	minRadius	maxRadius	averagePrecision	averageRecall	averageIoU	averageBestIris	averageBestPupil
9.0	5.0	1.0	101.0	51.0	30.0	200.0	0.95	0.75	0.87	0.55	0.52

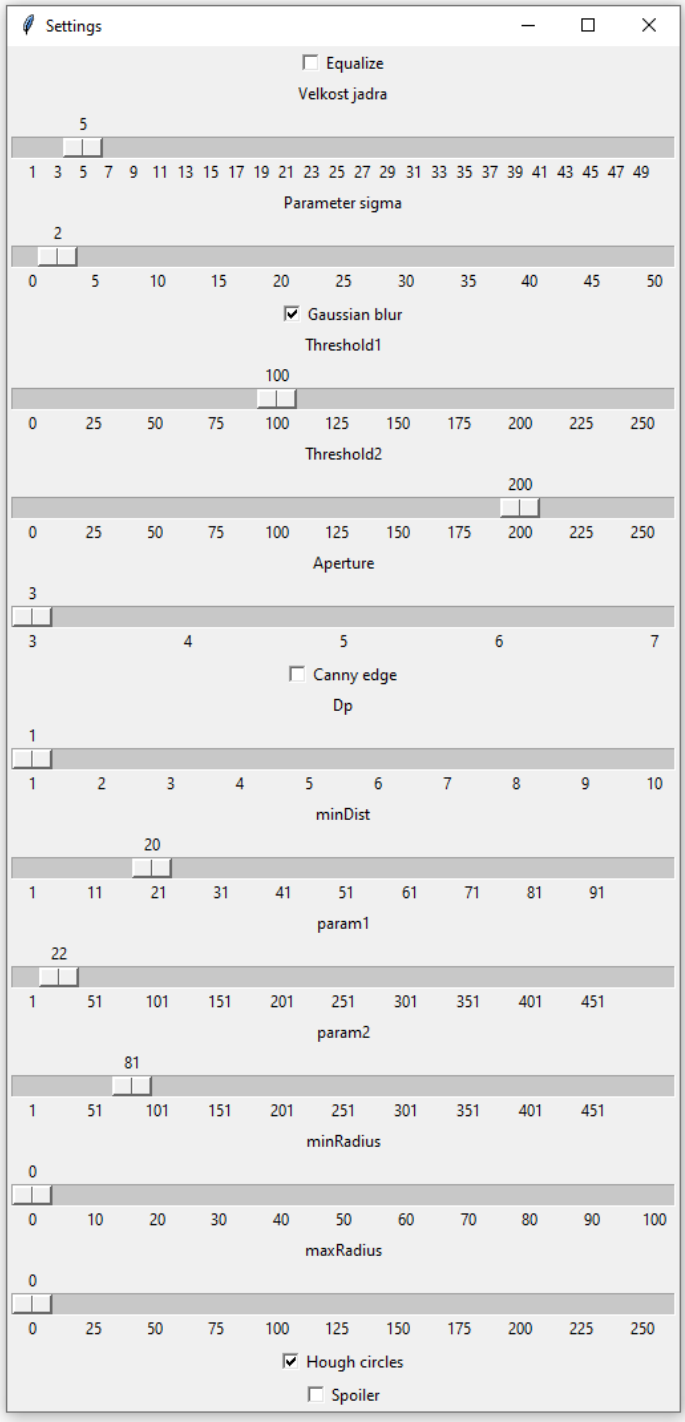
Dávid Gavenda 98533

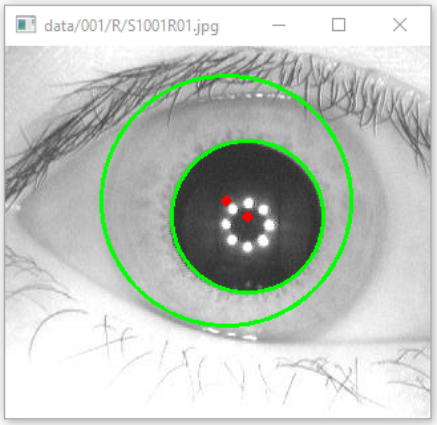
slidery trackbary

BONUS

Implementujte GUI inak ako prostredníctvom OpenCV. 1-3b

GUI som implementoval pomocou taktiež ***tkinter*** knižnice. Na vykresľovanie obrázka slúži stále OpenCV. Pre zmenu sa tu dajú správne využívať checkboxy, tak som to využil, taktiež hodnoty majú správne limitácie a celkovo to budí lepší dojem.

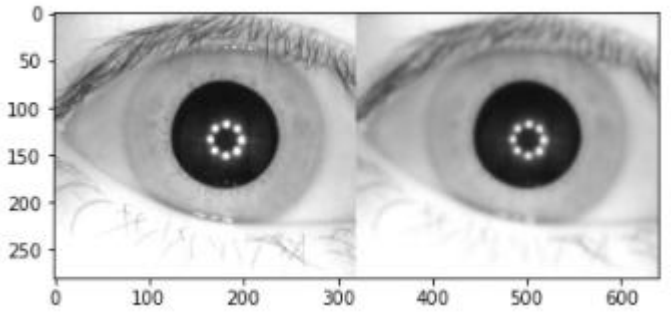
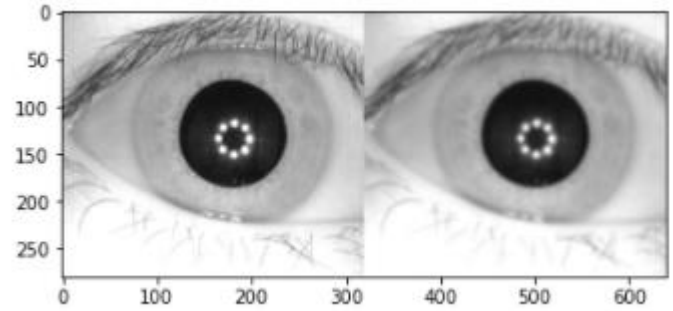
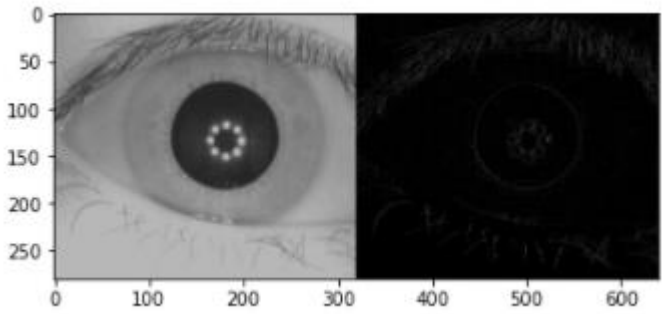
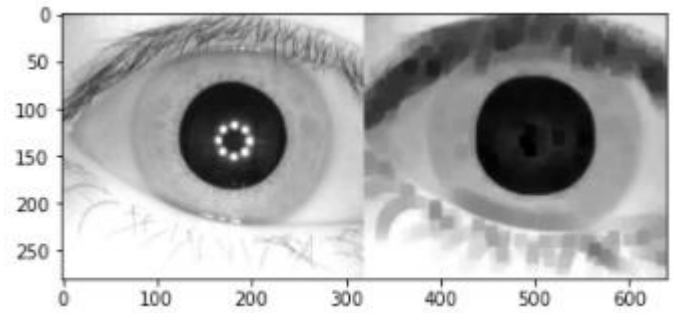




Dávid Gavenda 98533

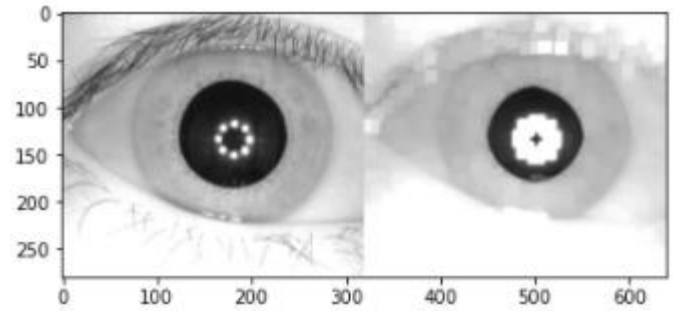
slidery trackbary

Vyskúšajte aj iné filtre na úpravy obrazu v OpenCV 1-2b

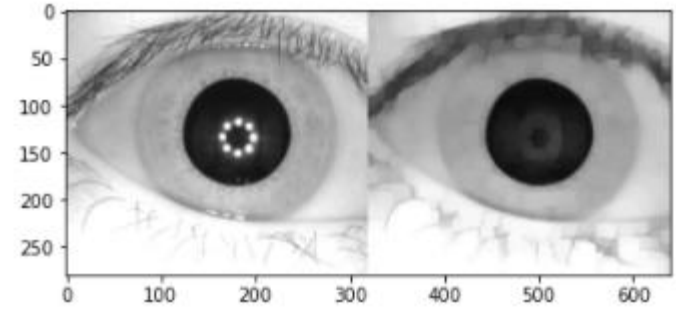
originál	nové	filter
		Bilateral Filter (9, 75 ,75)
		Blur(5,5)
		Laplacian(2)
		Kernel = 5x5 Erode(Kernel,iterations=3)

Dávid Gavenda 98533

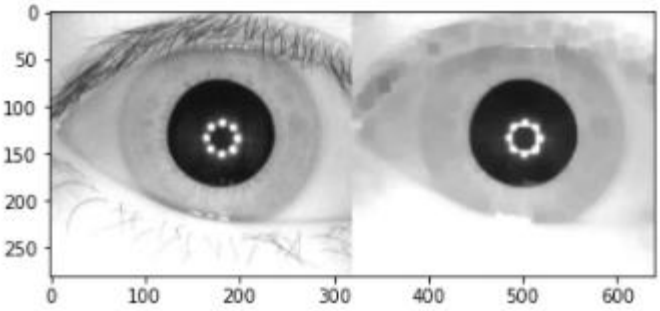
slidery trackbary



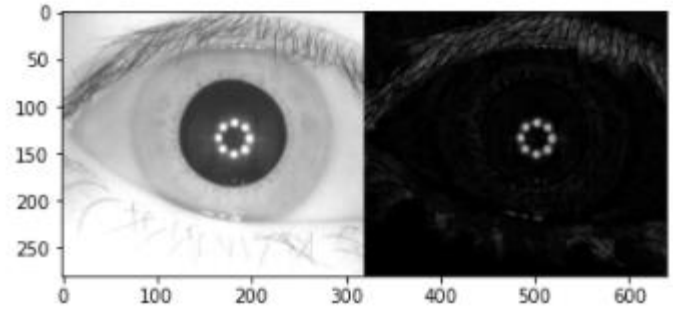
Kernel = 5x5
Dilate(Kernel,iterations=3)



Kernel = 11x11
MorphologyEx(MORPH_OPEN,Kernel)



Kernel = 11x11
MorphologyEx(MORPH_CLOSE,Kernel)



Kernel = 11x11
MorphologyEx(MORPH_TOPHAT,Kernel)

Dávid Gavenda 98533

slidery trackbary

Nájdite a analyzujte problémové obrazy v databáze. 1b

```
for i in range(len(images)):
    try:
        gray_iris = cv2.imread(images[i],cv2.COLOR_BGR2GRAY)
        imshow(gray_iris)
    except:
        print(iris_df['image'][i])
```

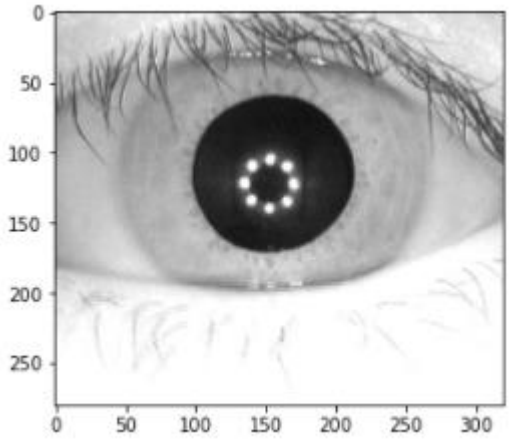
Pomocou jednoduchého cyklu si prejdeme celé .csv a pokúsime sa otvoriť súbory. Pokiaľ sa taký súbor otvoriť nedá, vypíšeme si názov.

008/R/S1008R09.jpg
008/R/S1008R10.jpg
011/L/S1011L10.jpg
024/R/S1024R03.jpg
024/R/S1024R04.jpg
024/R/S1024R05.jpg
026/R/S1026R04.jpg
026/R/S1026R05.jpg
037/R/S1037R04.jpg
037/R/S1037R05.jpg
038/L/S1038L05.jpg
043/R/S1043R10.jpg
053/L/S1053L14.jpg
156/L/S1156L10.jpg
158/R/S1158R06.jpg
162/R/S1162R11.jpg
181/R/S1181R01.jpg
182/L/S1182L10.jpg
223/R/S1223R04.jpg

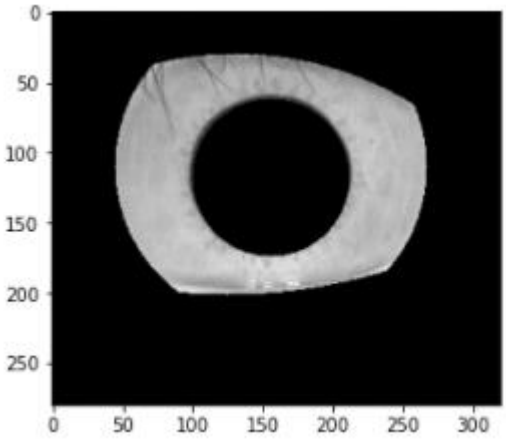
Rýchla kontrola nám ukáže, že majú síce cestu zapísanú ale žiadny súbor s daným označením nie je.

Podľa kružníc dúhovku segmentujte (vytvorte nový obrázok, kde budú všetky pixely, ktoré nepatria dúhovke, nastavené na nulu; ostatné na jednotku). 1b

Pomocou masiek a vypnutia stredu som dosiahol segmentáciu podľa údajov z .csv (zdroj).



vstup



výsledok

Dávid Gavenda 98533

slidery trackbary