

Informatyka w Medycynie

Projekt 2-Wykrywanie naczyń dna siatkówki oka

Raport

1. Skład grupy:

Dominik Rolewski, 136792, grupa L4
Kamil Stróżyk, 136805, grupa L4

2. Zastosowany język programowania oraz dodatkowe biblioteki

Zastosowaliśmy język Python, z dodatkowymi bibliotekami:

- OpenCV
- Numpy
- Matplotlib
- Os
- Skimage
- Scipy
- Sklearn

Z Sklearn wzięliśmy implementację klasyfikatora odległościowego kNN, pozostałe służyły nam do przetwarzania obrazu i przygotowywania zbioru testowego.

3. Opis zastosowanych metod:

A. Przetwarzanie obrazów

I. Kroki przetwarzania

1. Rozdzielenie obrazu na kanały i wybranie kanału zielonego do dalszej obróbki
2. Zastosowanie filtra CLAHE
3. Wielokrotne operacje otwierania i zamykania obrazu ze zwiększającą się maską i odjęcie wyników tej operacji od obrazu sprzed operacji
4. Ponowne zastosowanie CLAHE oraz threshold
5. Wyszukanie konturów na obrazie oraz zaznaczenie na biało tych wystarczająco dużych.
6. Ponowny threshold

II. Uzasadnienie dla poszczególnych kroków

1. Po przeanalizowaniu wyglądu wszystkich składowych oraz programowej zmiany kolorów na odcienie szarości, stwierdziliśmy, że kanał zielony posiada najwięcej potrzebnych nam szczegółów
2. Chcieliśmy zwiększyć kontrast zdjęcia
3. Taka operacja pozwoliła nam usunąć większość tła niebędącego naczyniami- wielokrotne otwieranie i zamykanie powoduje powstanie "rozmytego" obrazu oka, na którym nie widać naczyń
4. CLAHE znowu służy zwiększeniu kontrastu, threshold sprawia, że zdjęcie jest niejako już maską bitową odpowiedzi
5. Wyszukiwanie konturów i odsiewanie mniejszych z nich służy pozbyciu się "artefaktów"- drobnych elementów nie będących naczyniami a nie odsianymi thresholdem
6. Ponowny threshold uwydatnia naczynia oraz zapewnia 100% pewności, że wygenerowana maska jest binarna

B. Uczenie maszynowe (4.0)

I. Przygotowanie danych

Do obróbki braliśmy jedynie kanał zielony z obrazów. Wycinki były wyznaczane na podstawie indeksu danego piksela w tablicy- potem wyznaczaliśmy kwadrat o rozmiarze 10x10 pikseli (w przypadku pikseli na krańcach obrazu pomijamy indeksy nieistniejących pikseli, wokół środkowego jest zbudowany prostokąt) i obliczaliśmy dla niego momenty H_u , które były naszymi danymi podawanymi na wejście klasyfikatora, jako przewidywany wynik podawaliśmy binarną maskę odpowiedzi wykonaną przez ekspertów.

II. Zastosowane metod uczenia maszynowego wraz z informacją o przyjętych parametrach

Zastosowaliśmy prosty klasyfikator kNN na podstawie gotowej implementacji z pakietu scikit-learn. Liczba sąsiadów = 5.

III. Wyniki wstępnej oceny zbudowanego klasyfikatora

Test hold-out na zbiorze testowym będącym połową innego obrazka (nie prezentujemy go w wynikach) wykazały accuracy, mierzone za pomocą metody knn.score z pakietu sklearn o wartości 0.9217557624945258, w przybliżeniu 92%

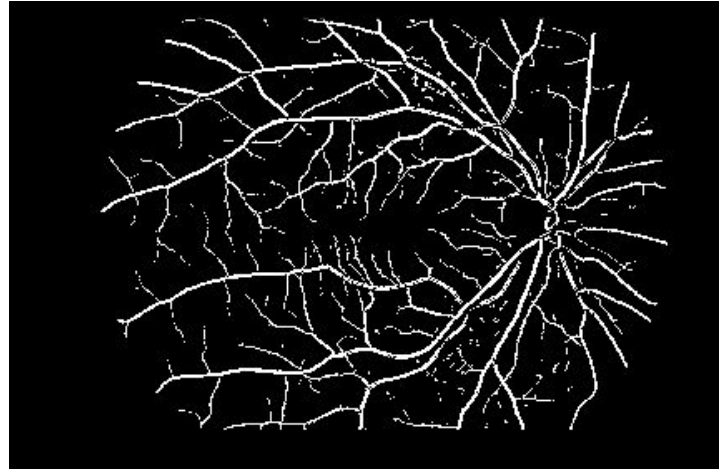
IV. Krótkie uzasadnienie zastosowanego rozwiązania

Z racji braku większego doświadczenia w machine learningu zdecydowaliśmy nie rzucać się na głęboką wodę i poeksperymentować z prostym klasyfikatorem. Eksperymentowaliśmy z rozmiarem wycinka oraz ilością sąsiadów w klasyfikatorze. Stąd inny niż sugerowany rozmiar wycinka i ilość sąsiadów- zwiększanie rozmiaru i ilości dawało lepsze wyniki, naszą granicą zwiększania był czas przetwarzania i ograniczenia pamięciowe.

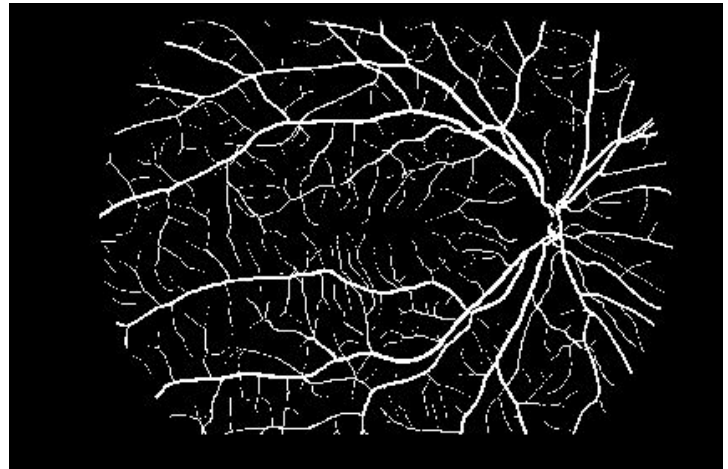
4. Wizualizacja wyników działania programu dla wybranych obrazów

1. Obraz 1

Obraz bazowy i maska ekspercka:



A. Przetwarzanie obrazu



confusion matrix:

622440 , 162458

211448 , 5917437

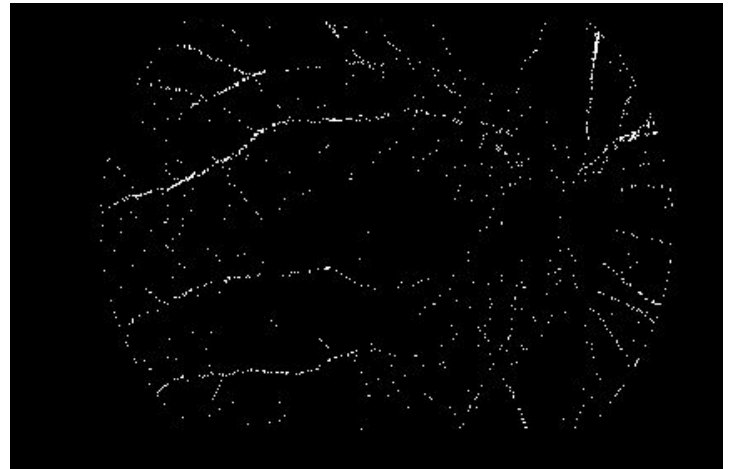
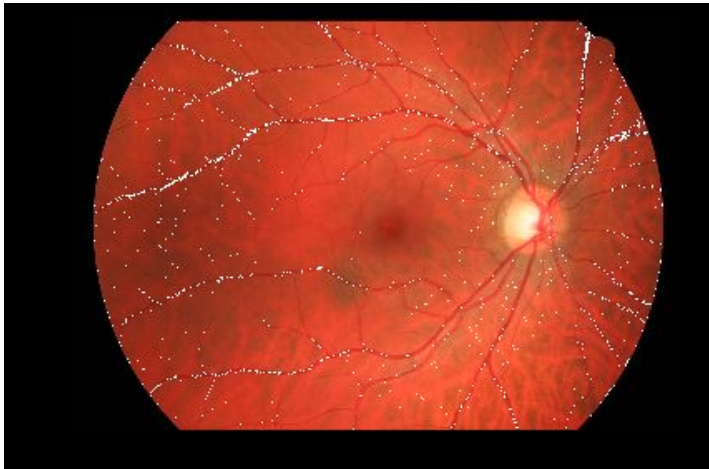
accuracy: 0.9459187538862588

sensitivity: 0.7464311754096473

specificity: 0.9732794727540525

geometric mean of sensitivity and specificity 0.8523415634884228

B. Klasyfikator



confusion matrix:

69747 , 31192

764141 , 6048703

accuracy: 0.8849641361321291

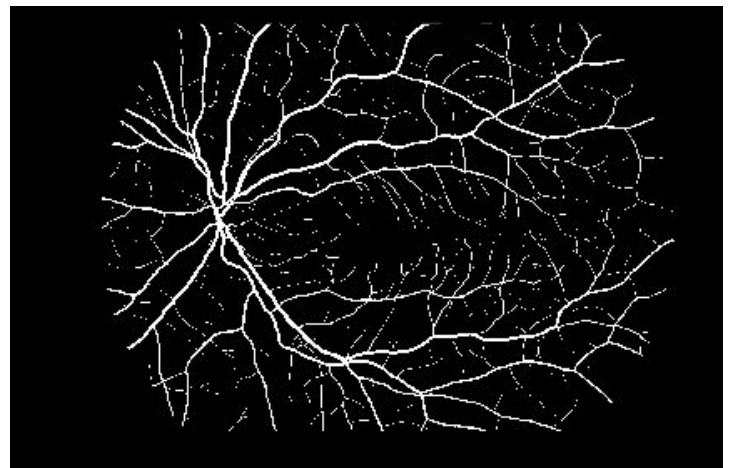
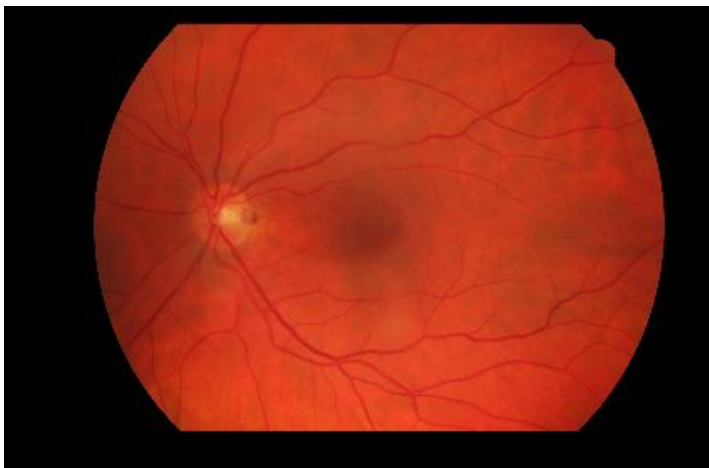
sensitivity: 0.08364072873095668

specificity: 0.9948696482422805

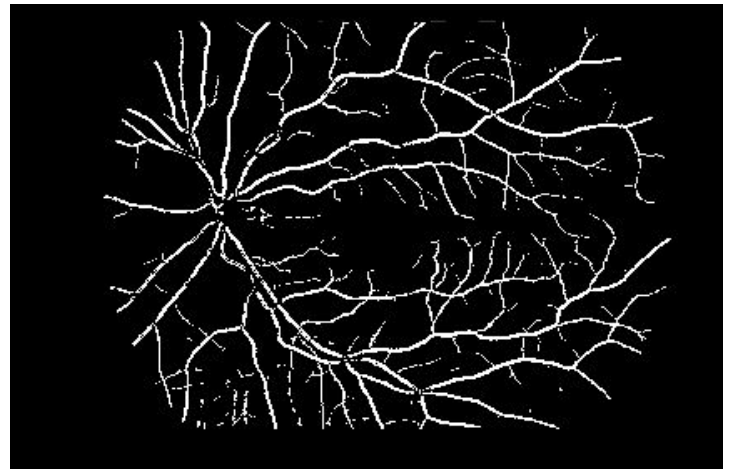
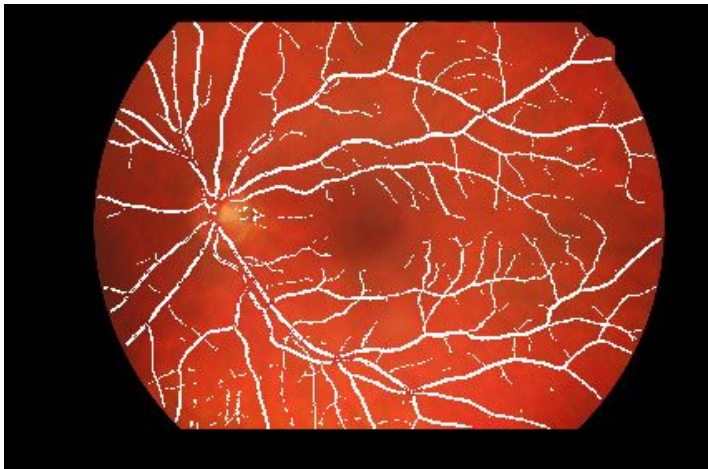
geometric mean of sensitivity and specificity 0.28846424799495496

2. Obraz 2

Obraz bazowy i maska ekspercka:



A. Przetwarzanie obrazu



confusion matrix:

470501 , 243274

139032 , 6062923

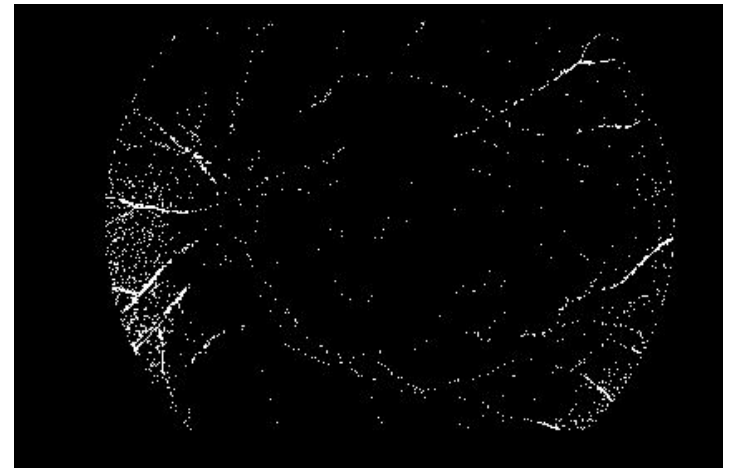
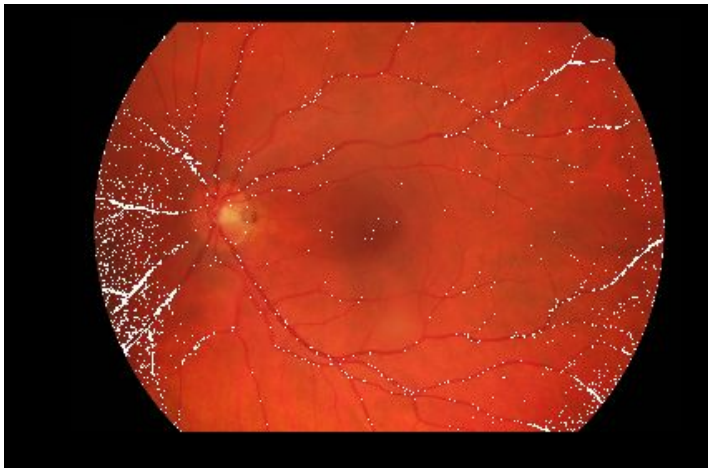
accuracy: 0.9447193571755982

sensitivity: 0.7719040642590311

specificity: 0.9614230256365286

geometric mean of sensitivity and specificity 0.8614675507301776

B. Klasyfikator



confusion matrix:

81572 , 90389

527961, 6215808

accuracy: 0.9105878916614732

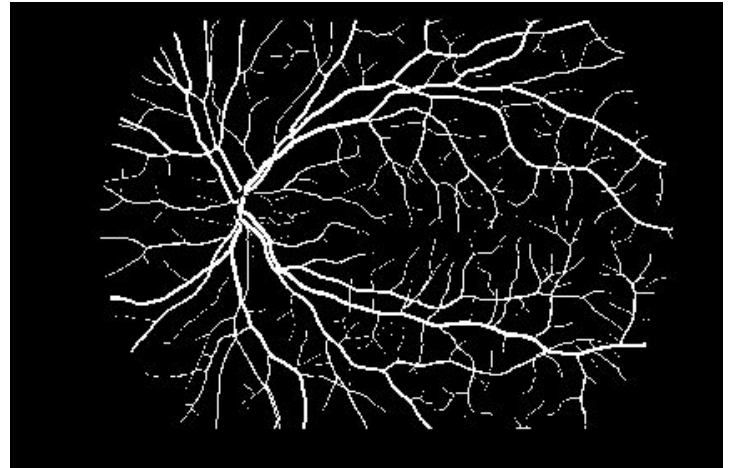
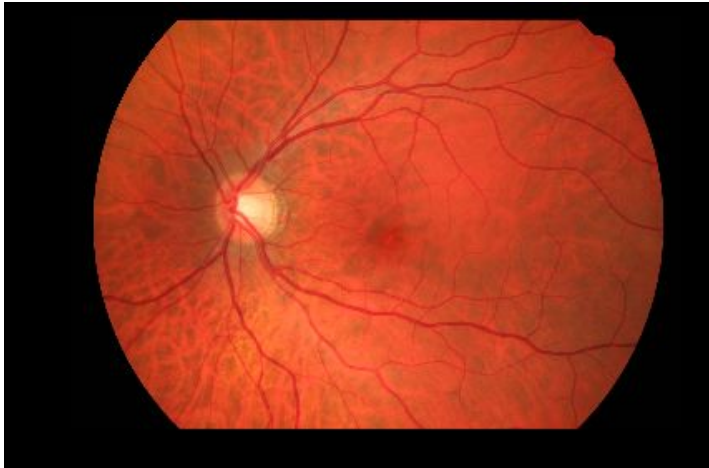
sensitivity: 0.13382704463909256

specificity: 0.9856666387047535

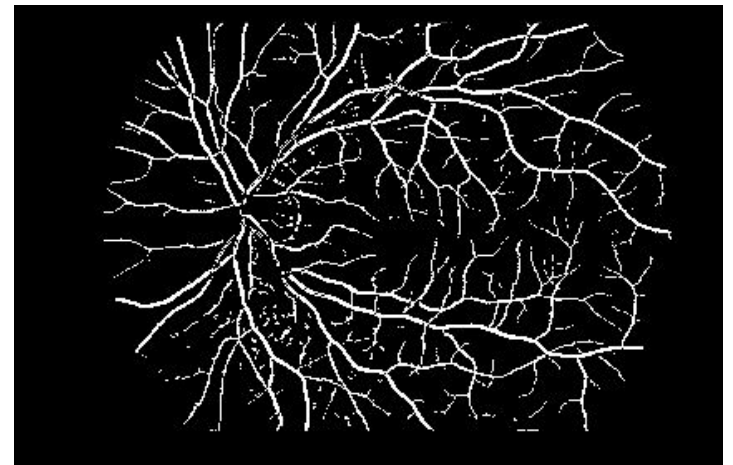
geometric mean of sensitivity and specificity 0.3631925842541466

3. Obraz 3

Obraz bazowy i maska ekspercka:



A. Przetwarzanie obrazu



confusion matrix:

667660 , 194763

140762, 5911184

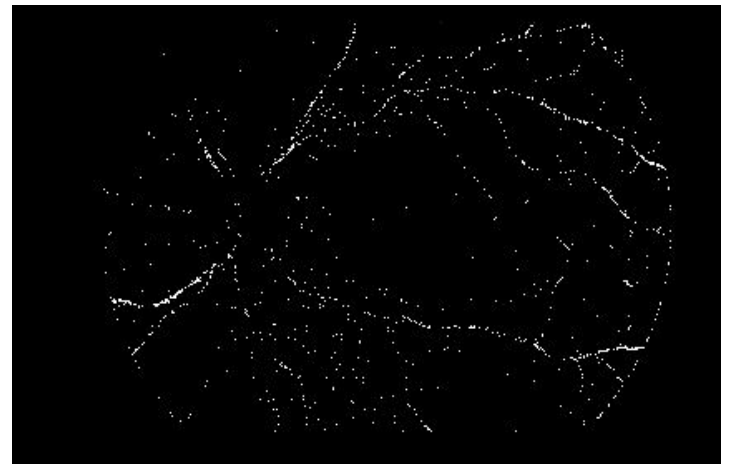
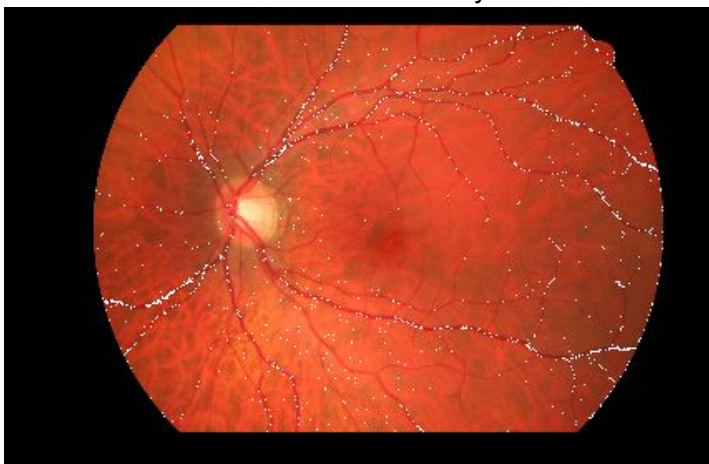
accuracy: 0.9514742415396112

sensitivity: 0.8258805425879058

specificity: 0.968102736561585

geometric mean of sensitivity and specificity 0.8941684479740483

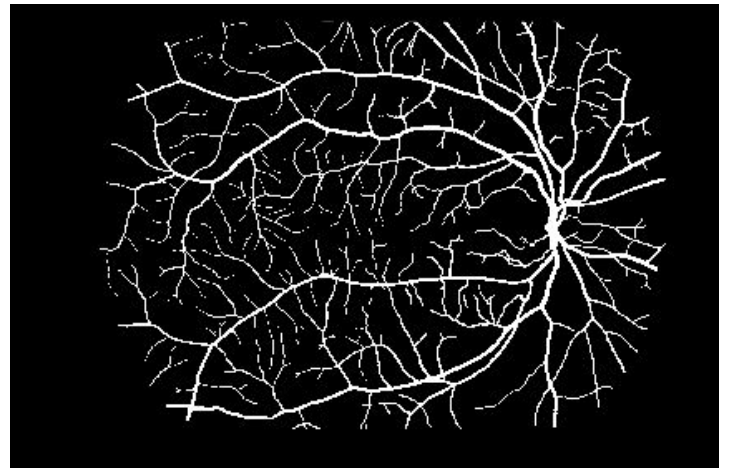
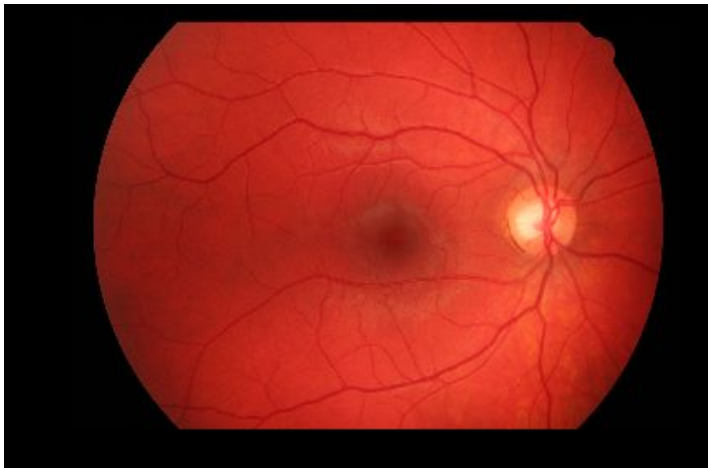
B. Klasyfikator



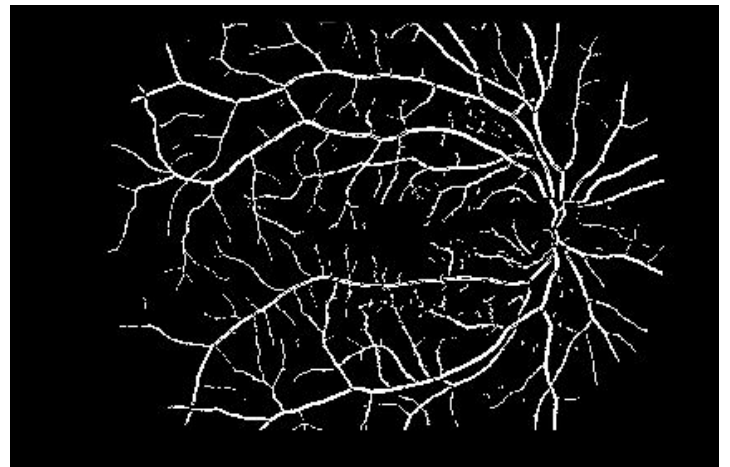
confusion matrix:
69567 , 27125
738855, 6078822
accuracy: 0.8892191030013006
sensitivity: 0.08605282884434119
specificity: 0.9955576096549806
geometric mean of sensitivity and specificity 0.29269531699076007

4. Obraz 4

Obraz bazowy i maska ekspercka:

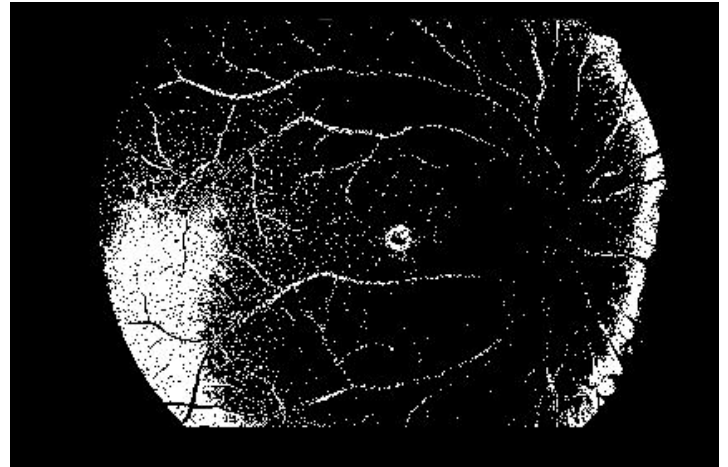
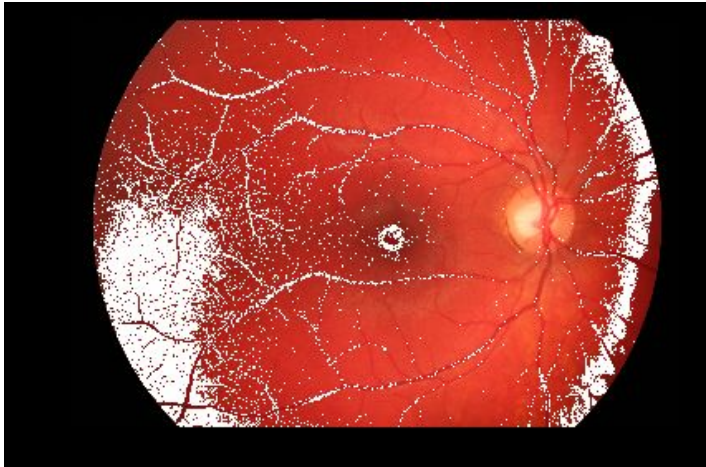


A. Przetwarzanie obrazu



confusion matrix:
605787 , 127257
258339 , 5920689
accuracy: 0.9442141227695545
sensitivity: 0.7010401260927226
specificity: 0.978958641495807
geometric mean of sensitivity and specificity 0.828425789961769

B. Klasyfikator



confusion matrix:

221536 , 912023

642590 , 5135923

accuracy: 0.7750872676094809

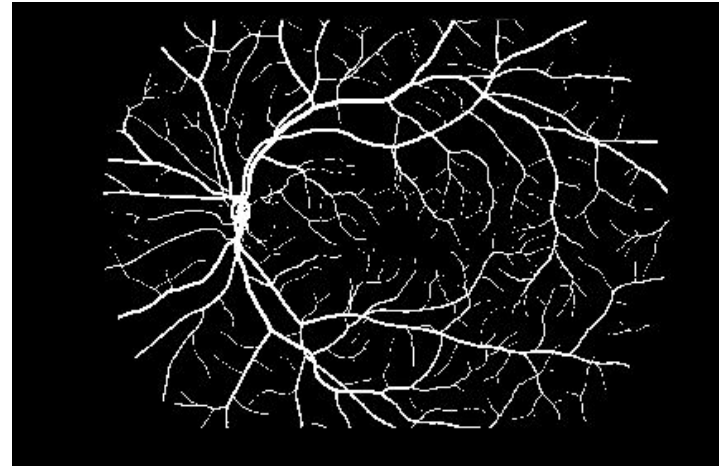
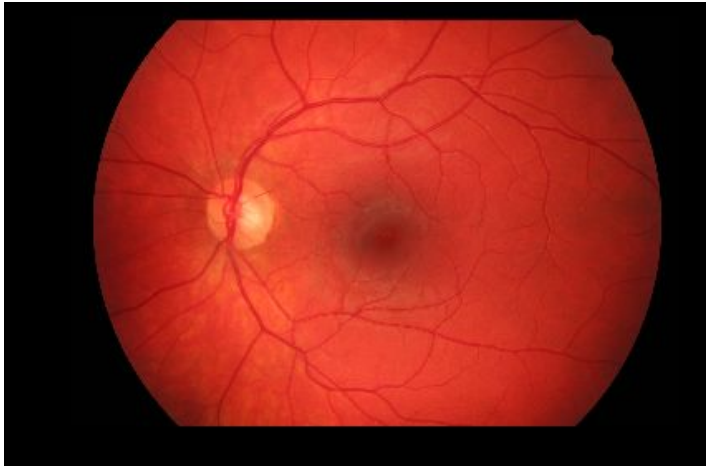
sensitivity: 0.25637002011280763

specificity: 0.8492011998784381

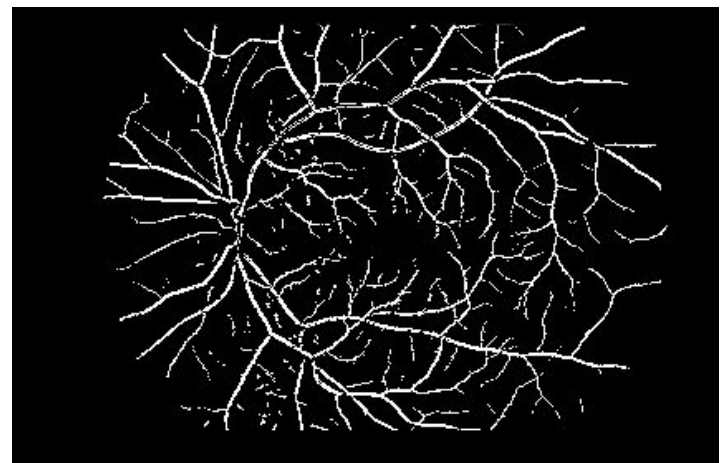
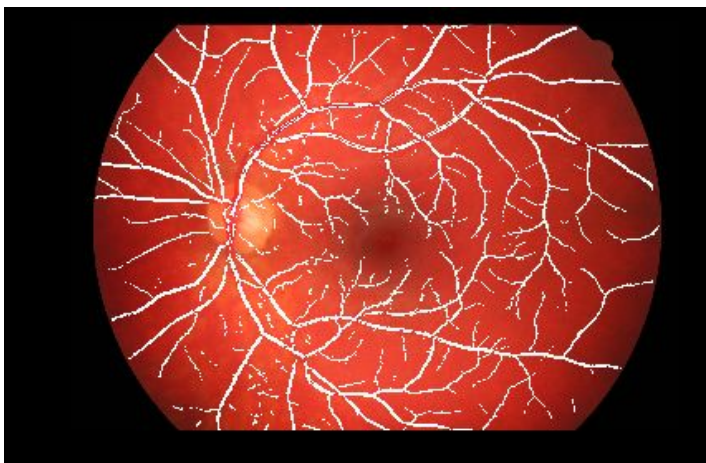
geometric mean of sensitivity and specificity 0.46659375123618574

5. Obraz 5

Obraz bazowy i maska ekspercka:

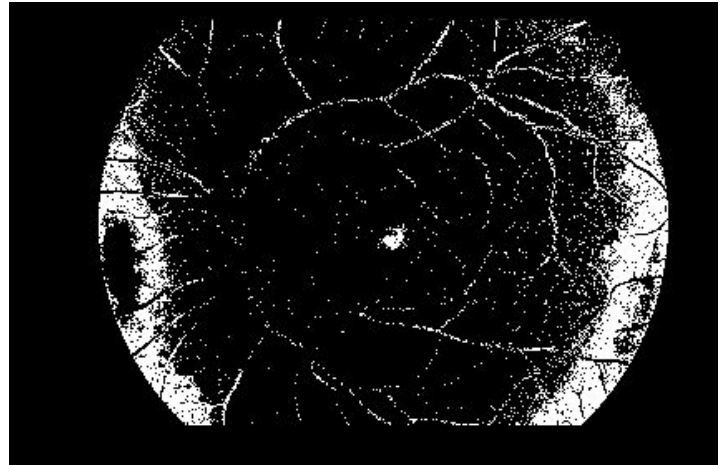
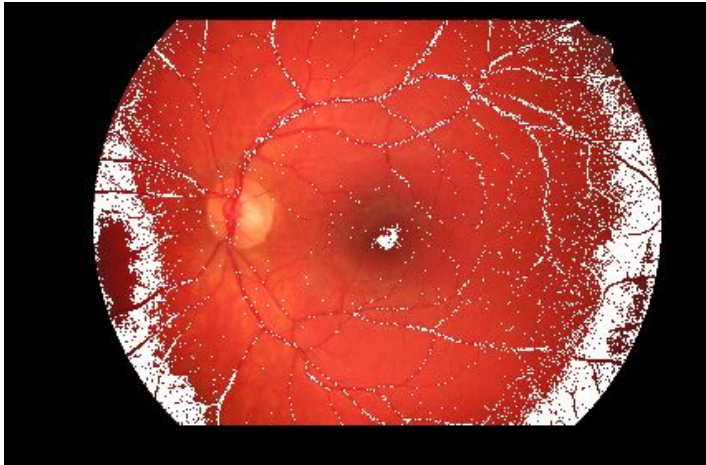


A. Przetwarzanie obrazu



confusion matrix:
576031 , 151573
195564, 5989844
accuracy: 0.9497849851844609
sensitivity: 0.7465457915097946
specificity: 0.9753195394483065
geometric mean of sensitivity and specificity 0.8532998872333245

B. Klasyfikator



confusion matrix:
176041 , 801594
595554, 5339823
accuracy: 0.7978959099159671
sensitivity: 0.22815207459872083
specificity: 0.8694773535162976
geometric mean of sensitivity and specificity 0.4453909092262085

5. Analiza wyników działania programu dla wybranych obrazów

1. Obraz 1

W przypadku przetwarzania obrazu największą wartość macierzy pomyłek ma true negative, "na oko" widać, że żyły zostały dość poprawnie wykryte, potwierdza to trafność na poziomie 94,5%, wyższa niż dla klasyfikatora, Czułość i swoistość ma jednak od klasyfikatora niższą, za to wartość ich średniej geometrycznej jest niemal 4 razy większa. Klasyfikator również najwięcej rozpoznał pikseli niebędących naczyniem, nawet więcej niż przetwarzanie obrazu, jednak całościowo był gorszy, jednak wynik i tak jest w miarę zadowalający na tle pozostałych.

2. Obraz 2

Wizualnie sytuacja jest podobna- przetwarzania obrazu poradziło sobie świetnie, klasyfikator-trochę gorzej. O dziwo, trafność obydwu jest dość zbliżona 94% do 91% dla przetwarzania. Obydwa mają największą wartość

True Negative w macierzy pomyłek. Czułość przetwarzania jest blisko 7 razy większa niż klasyfikatora, mają natomiast zbliżoną swoistość. Znowu średnia geometryczna dwóch wyżej wymienionych wartości jest dużo mniejsza dla klasyfikatora.

3. Obraz 3

Wizualnie oraz w temacie macierzy pomyłek sytuacja jest analogiczna do poprzednich obrazków. Znowu występuje sytuacja, w której obraz przetworzony ma wielokrotnie większą czułość niż klasyfikator, tym razem dziesięciokrotnie, swoistość jest podobna, a średnia geometryczna niższa dla klasyfikatora. W temacie wizualizacji dla klasyfikatora, naszą subiektywną opinią jest to najlepiej wyglądająca maska bitowa odpowiedzi- mimo, że nie ma najwyższej trafności ze wszystkich, znalazła najwięcej naczyń i zaznaczyła najmniej dodatkowych elementów.

4. Obraz 4

Porównanie macierzy pomyłek oraz wizualizacji wypada tak jak dla pozostałych obrazów. Wizualnie oraz w pod względem trafności jest to najgorsza z masek odpowiedzi wygenerowanych przez klasyfikator. Mieliśmy pokazać zarówno sukcesy jak i porażki- ten możemy uznać za porażkę- doszło do sytuacji, że w niektórych miejscach klasyfikator zaznaczył naczynia i tło zupełnie odwrotnie. Czułość znowu jest dużo wyższa dla przetwarzania obrazu, swoistość natomiast nie jest już na zbliżonym poziomie tylko wyraźnie mniejsza dla klasyfikatora. Występuje tutaj jedna z najmniejszych różnic między średnimi geometrycznymi.

5. Obraz 5

Porównanie macierzy pomyłek oraz wizualizacji wypada tak jak dla pozostałych obrazów. Wizualnie oraz w pod względem trafności jest to druga najgorsza z masek odpowiedzi wygenerowanych przez klasyfikator, jest niewiele lepsza od obrazu 4- również uznajemy ją za porażkę. Ponownie, czułość jest dużo wyższa dla przetwarzania obrazu i swoistość jest wyraźnie mniejsza dla klasyfikatora. Różnica między średnimi geometrycznymi również jest tutaj dość niska.