Sprawozdanie –

retina vessel segmentation

1. **Skład grupy**

Stanisław Główczewski 151877  
Rafał Szubert 151792

1. **Zastosowany język programowania oraz dodatkowe biblioteki**

Program zaimplementowano w języku python wraz ze standardową biblioteką tkinter do stworzenia GUI. Wykorzystano bibliotekę numpy do wspomagania obliczeń, PIL w celu obróbki obrazów oraz sklearn i tenserflow do stworzenia modli uczenia maszynowego.

1. **Opis zastosowanych metod:**
   1. przetwarzanie obrazów (3.0)

Początkowo zdjęcie poddano preprocesingowi. Usunięto kanały czerwony i niebieski oraz wyostrzono obraz. Następnie został nałożony filtr ze zmienną wartością graniczną. Na koniec, wykorzystano erozję i dylatację.

Zauważono, że wykorzystanie jednie zielonego koloru w przetwarzaniu obrazu daje co najmniej tak dobre wyniki, jak połącznie kilku albo wykorzystanie innego koloru. Rodzaj filtra, wartość graniczna oraz wielkość filtra zostały wybrane doświadczalnie poprzez wykonanie wielu testów z różnymi wartościami powyższych parametrów. Otrzymany w ten sposób obraz składał się z wielu kropek, więc w celu zwiększenia dokładności predykcji popularnej sztuczki, łączącej kropki, polegającej na zastosowaniu dylatacji a następnie erozji.

* 1. uczenie maszynowe (4.0)

Ta, jak i każda procedura jest napisana jako fragment modułu. Wykonanie danego pliku spowoduje przejście do części uczenia, która składa się z 3 etapów – przygotowania, uczenia i ewaluacji. Wpierw przygotowane są dane, poprzez otwarcie plików uczących, podzielenie ich na odpowiednią ilość mniejszych obrazków 5x5.

Dla każdego obrazka liczone są miary: value, variance, minimum, maximum. Każdy obrazek dostaje też binarny opis, czy środkowy element jest naczynkiem, czy nie. W tym momencie odbywa się też podział obrazków na zestaw treningowy i testowy.

Zgodnie z założeniami algorytmu „na 4” wykorzystano metodę KNN   
k-nearest neighbors, która tworzy n wymiarową przestrzeń ze znanymi z procesu uczenia punktami. W procesie ewaluacji algorytm znajduje k najbliższych sąsiadów wielowymiarowej przestrzeni i przyjmuje wartość najczęściej pojawiającego się stanu.

* 1. uczenie maszynowe (5.0)
* przygotowanie danych - wyznaczanie wycinków obrazu, ekstrakcja cech z wycinków (jeśli zastosowano)
  + - wstępne przetwarzanie zbioru uczącego (jeśli zastosowano)
    - zastosowane metody uczenia maszynowego wraz z informacją o przyjętych parametrach
    - wyniki wstępnej oceny zbudowanego klasyfikatora (testy *hold-out*)
    - krótkie uzasadnienie zastosowanego rozwiązania

1. **Wizualizacja wyników działania programu dla wybranych obrazów (warto pokazać zarówno sukcesy jak i porażki). Dla porównania należy zamieścić maskę ekspercką (*ground truth*). Wyniki wizualizacji należy zaprezentować dla co najmniej 5 obrazów (w przypadku metod uczenia maszynowego nie mogą to być obrazy, które zostały wykorzystane do nauczenia klasyfikatora).**
2. **Analiza wyników działania programu dla wybranych obrazów (tych samych, które wykorzystano w punkcie 4) z wykorzystaniem odpowiednich miar oceny (omawianych wcześniej). Analizę należy przeprowadzić indywidualnie dla każdego z obrazów. W przypadku realizacji zadań na 4.0/5.0 należy dokonać porównania miar oceny osiąganych przez metody przetwarzania obrazów oraz uczenia maszynowego.**