

# **Informatyka w Medycynie**

Wykrywanie naczyń dna oka

**Krzysztof Pasiewicz, 132302**

**Mikołaj Frankowski, 132220**

## 1. Metoda wykorzystująca przetwarzanie obrazu

### 1.1. Opis algorytmu

Nasze rozwiązanie problemu wykorzystujące przetwarzanie obrazu oparliśmy o wykrywanie koloru naczyń dna oka. Algorytm, który napisaliśmy jest dosyć prosty.

Najpierw na obrazie w formacie BGR zwiększamy delikatnie zawartość koloru zielonego proporcjonalnie do jego poprzedniej zawartości w kolorze piksela. Powoduje to zwiększenie kontrastu pomiędzy kolorem bardziej pomarańczowym a kolorem czerwonym.

Następnie przy użyciu funkcji `inRange`, zawartej w OpenCV, staramy się wygenerować jak najlepiej filtrującą maskę, poprzez odpowiedni dobór zakresu kolorów, który zostanie odfiltrowany. Potem maska zostaje zastosowana do obrazu przy użyciu funkcji `bitwise_and`.

Na końcu dla większej klarowności w obrazie, kolor naczynek zostaje zamieniony na kolor biały.

### 1.2. Otrzymane wartości metryk dla klasyfikatora

#### 1.2.1. Macierze pomyłek:

Dla najwyższej dopuszczalnej wartości Hue równej 6:

		Predykcja	
		Nie	Tak
Prawidłowa wartość	Nie	30421696	137564
	Tak	2319680	505034

Dla najwyższej dopuszczalnej wartości Hue równej 7:

		Predykcja	
		Nie	Tak
Prawidłowa wartość	Nie	30421696	649668
	Tak	2319680	777303

#### 1.2.2. Obliczone miary:

Dla najwyższej dopuszczalnej wartości Hue równej 6:

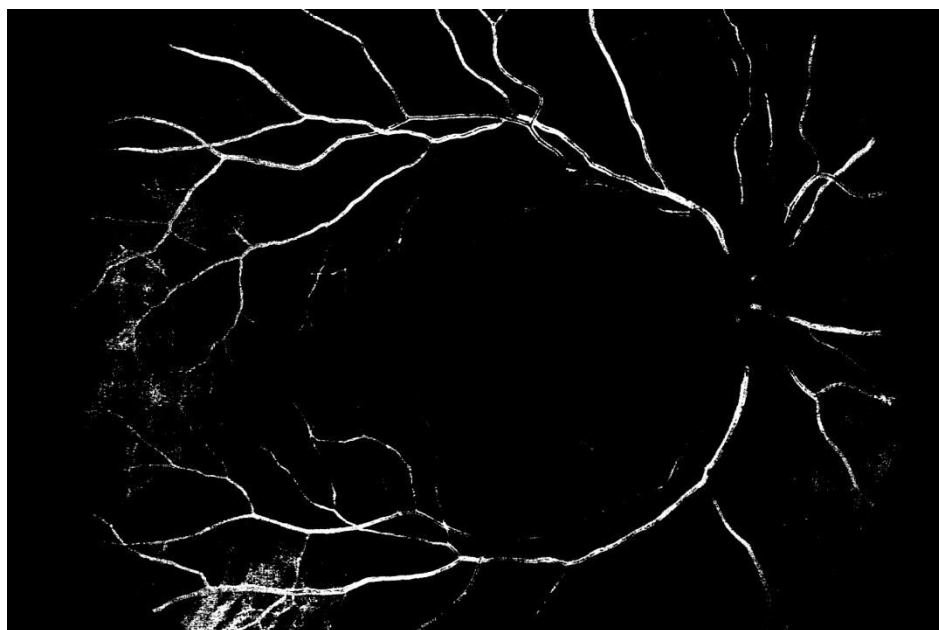
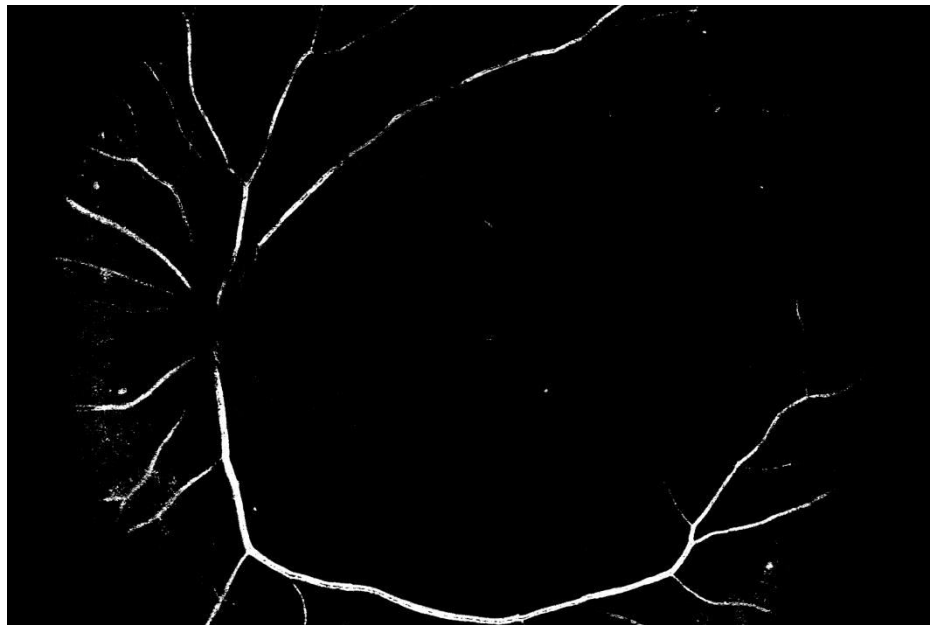
Accuracy	Precision	Recall/Sensitivity	Specificity	F1-Score
0,926395	0,78592526	0,178791198	0,995498	0,291312

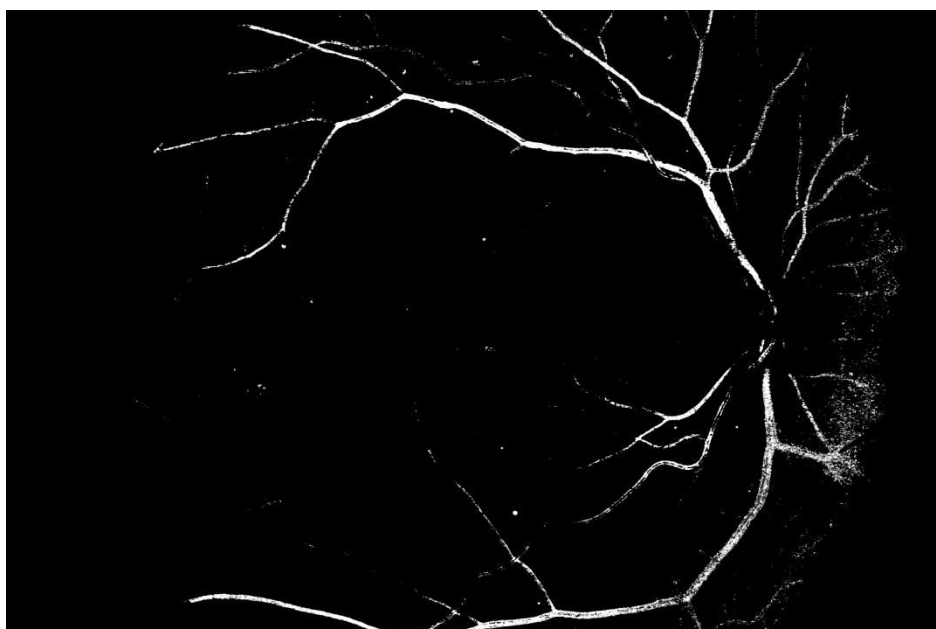
Dla najwyższej dopuszczalnej wartości Hue równej 7:

Accuracy	Precision	Recall/Sensitivity	Specificity	F1-Score
0,913097	0,544722352	0,25098717	0,979091	0,343639

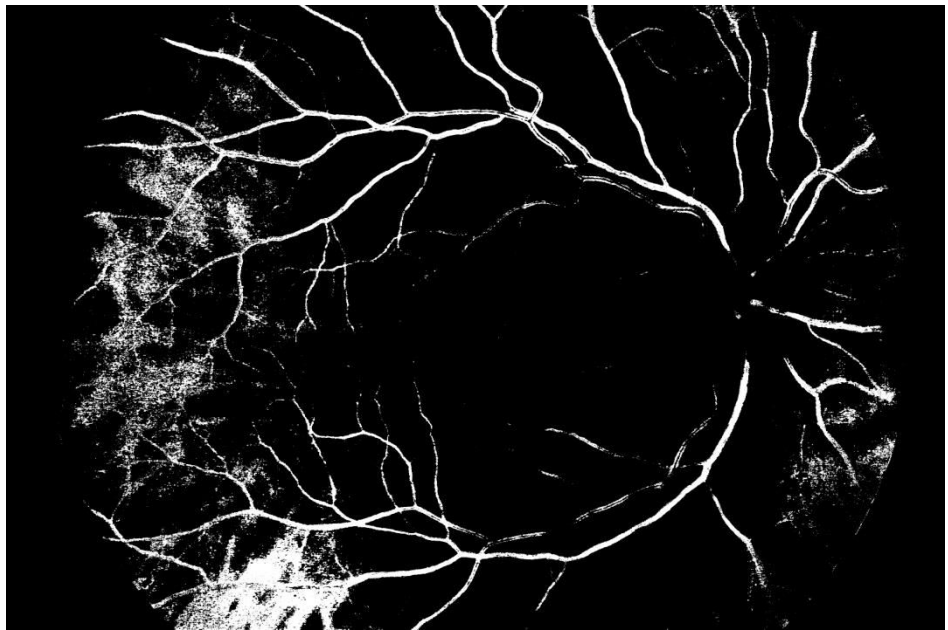
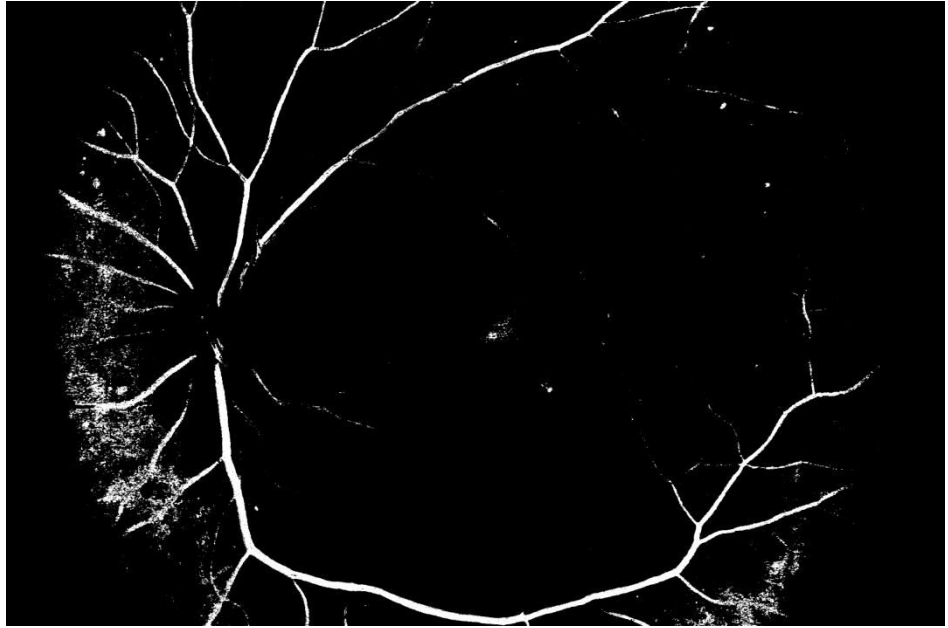
1.2.3.Zdjęcia z wykrytymi naczyniami:

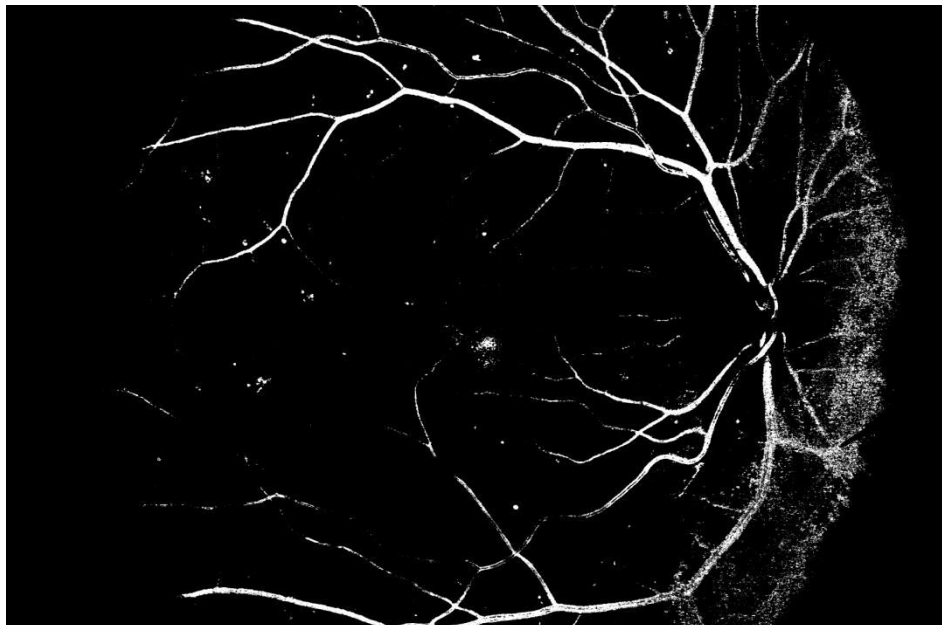
Dla najwyższej dopuszczalnej wartości Hue równej 6:





Dla najwyższej dopuszczalnej wartości Hue równej 7:





## 2. Metoda wykorzystująca algorytmy uczenia maszynowego

### 2.1. Opis wykorzystanego algorytmu uczenia maszynowego

W naszym rozwiązaniu wykorzystaliśmy algorytm losowego lasu. Jest to algorytm korzystający z algorytmu drzewa decyzyjnego i implementujący go wielokrotnie. Algorytm drzewa decyzyjnego dzieli zbiór danych uczących na grupy podobnych do siebie danych korzystając z entropii, a następnie tworzy drzewo decyzyjne w celu określenia grupy do której należy nowa obserwacja.

Natomiast algorytm losowego lasu implementuje zadaną ilość drzew decyzyjnych, z których każde uczone jest na losowo wybranym podzbiórze cech uczących (zarówno ilość drzew jak i maksymalna ilość cech w danym drzewie są określane odpowiednimi parametrami). Każda

nowa obserwacja jest klasyfikowana przez każde drzewo i ostateczna decyzja podejmowana jest poprzez „głosowanie” poszczególnych drzew decyzyjnych.

## 2.2. Opis rozwiązania

W celu wykrycia naczynek dna oka na zdjęciach pochodzących ze zbioru zdjęć dostępnego pod adresem TODO wykorzystaliśmy algorytm Losowego Lasu (RandomForestClassifier), który dla danego wycinka obrazu ( 7x7 pikseli ) podejmuje decyzje dla piksela znajdującego się na środku tego wycinka.

Pierwszym krokiem, jest podzielenie obrazu na fragmenty o rozmiarach 7x7 pikseli. Dla takiego fragmentu liczona są następujące wartości:

- Wariancja koloru pikseli
- Momenty danego wycinka
- Momenty Hu danego wycinka

Po zebraniu powyższych danych są one umieszczane w strukturze danych wraz ze zmienną decyzyjną dla danego piksela. Ze względu na znaczną różnicę w ilości pikseli przedstawiających naczynka i pozostałą część oka usuwamy nadmierną część rekordów tak, aby otrzymać równą ilość przykładów uczących dla decyzji pozytywnej i negatywnej.

Zbiór danych dzielony jest na zbiór treningowy i testowy w proporcji 0,8 – 0,2.

Wstępna analiza naszego zbioru danych ujawniła znaczącą różnicę zakresów naszych obliczonych cech wycinków obrazu (na przykład wartość wariancji potrafi osiągać wartości 100 czy 200, a momenty Hu wartości mniejsze niż 1. Aby zapobiec zdominowaniu modelu przez wariancję przeskalowaliśmy dane przy użyciu obiektu StandardScaler dostępnego w bibliotece Scikit-Learn.

Za pomocą tak przygotowanych danych rozpoczęliśmy budowę najlepszego możliwego klasyfikatora. W tym celu wykorzystaliśmy metodę przeszukiwania siatki (GridSearchCV), aby dobrać najlepsze możliwe parametry klasyfikatora. Przetwarzanie zwróciło następujące wyniki i na ich podstawie stworzyliśmy klasyfikator:

```
{ 'criterion': 'gini',  
  'max_depth': 8,  
  'max_features': 'sqrt',  
  'n_estimators': 10 }
```

Następnie przeprowadziliśmy wstępne testy jakościowe przy użyciu k-fold cross validation, oraz końcowo również przeprowadziliśmy testy na stworzonym wcześniej zbiorze testowym.

## 2.3. Otrzymane wartości metryk dla klasyfikatora

### 2.3.1. Macierz pomyłek (k-fold cross validation):

		Predykcja	
		Nie	Tak
idlo	wart	9577	4031

		2540	11058
	Tak		

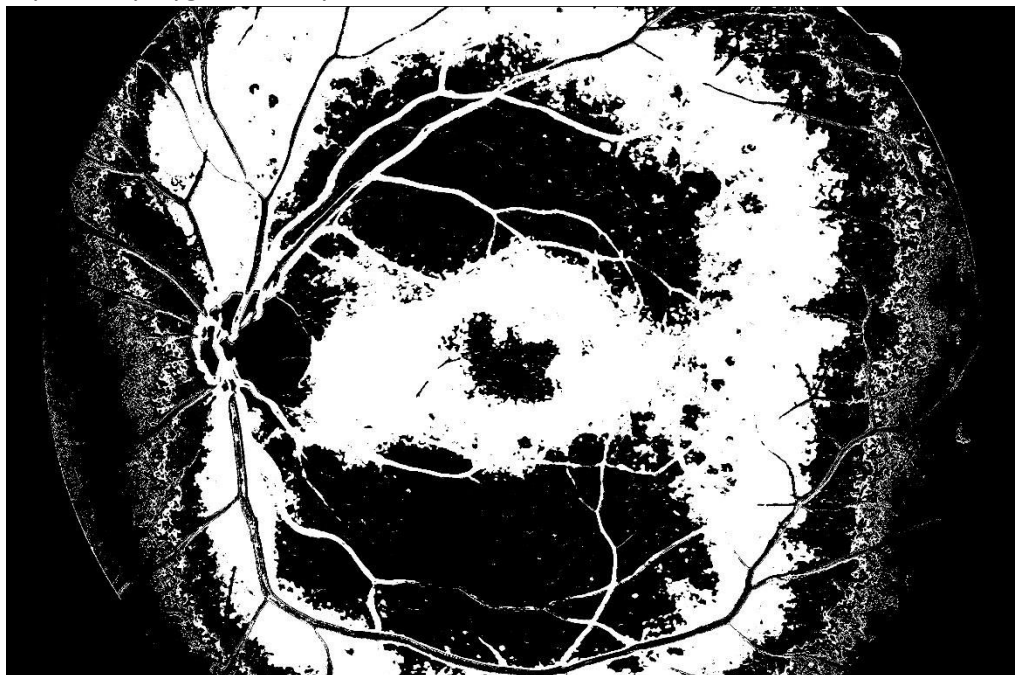
### 2.3.2. Macierz pomyłek (zbiór testowy):

		Predykcja	
		Nie	Tak
Prawidłowa wartość	Nie	2297	1099
	Tak	559	2847

### 2.3.3. Obliczone miary:

	Accuracy	Precision	Recall/Sensitivity	Specificity	F1-Score
training set	0,758472	0,732851746	0,813207825	0,703777	0,770942
test set	0,756248	0,721490117	0,835877863	0,676384	0,774483

### 2.3.4. Przykładowy wygenerowany obraz



## 3. Porównanie obu metod:

Wstępne porównanie jakości klasyfikacji pikseli dla obu metod pokazuje, że klasyfikator wykorzystujący Las Losowy nie okazał się lepszy od samego przetwarzania obrazu.

Trafność okazują się być bardzo wysoka tylko dla przetwarzania obrazu, natomiast uczenie maszynowe okazało się być gorsze. Jednak na pozostałych obliczonych metrykach przetwarzanie obrazu traci swoją przewagę. Precyzja (wartość pozytywnych przewidywań) jest całkiem wysoka



przy przetwarzaniu obrazu jak i algorytmie lasu losowego. Najgorzej klasyfikacja przez przetwarzanie obrazu wypada przy kolejnej metryce czyli Wrażliwość (czyli stosunek prawdziwie pozytywnych) wynika to z tego, że przetwarzanie obrazu zaproponowane przez nas wykrywa tylko dobrze widoczne naczynka pomijając te znacznie mniej widoczne na obrazie. Stąd tak wysoka wartość fałszywie negatywnych. Znacznie lepiej wypada metryka jaką jest specyficzność, gdyż ilość fałszywie pozytywnych jest znacznie mniejsza. W algorytmie przetwarzania obrazu wynika to z balansu między zbyt dużym, a zbyt małym stopniem wykrywania naczynek (odpowiada za to parametr). Jest to największy problem tego algorytmu. W celu porównania kombinacji miar obliczamy również F1 score – czyli średnią harmoniczną precyzji i wrażliwości. Dzięki temu możemy łatwo zauważyć, że Las Losowy jest znacznie lepszy w naszym zadaniu.

Obie metody są natomiast wrażliwe na dane wejściowe. Przetwarzanie obrazu działa na zasadzie nasycania i wykrywania koloru, więc inne kolorystycznie zdjęcia mogą okazać się bardzo problematyczne. Tak samo może się zdarzyć przy metodzie uczenia maszynowego, co może doprowadzić do wykonania błędnych predykcji przez model.

Po próbie wygenerowania ostatecznego obrazu okazało się, że próba rozpoznania naczynek na całym zdjęciu przez model uczenia maszynowego nie spisała się tak dobrze jakbyśmy tego oczekiwali. Wynika to zapewne z wybranego modelu uczenia maszynowego (sieci neuronowe zwykle lepiej sprawdzają się w zadaniach analizy i przetwarzania obrazu, jednak ich dostrojenie i nauczanie zajmuje zdecydowanie ogromną ilość czasu.).

Jednak przetwarzanie obrazu ma jedną znaczącą przewagę nad uczeniem maszynowym. Jest to szybkość działania – uczenie maszynowe wymaga przygotowania danych o każdym wycinku obrazu, przeanalizowaniu tych danych, wybraniu najlepszych parametrów dla modelu, nauczania go. Cały proces jest bardzo czasochłonny i pracochłonny. Natomiast druga metoda wymaga jedynie dostrojenia parametrów i bardzo szybko wykonuje wszystkie operacje.

Podsumowując metoda uczenia maszynowego osiąga lepsze wyniki co potwierdzają wartości niektórych obliczonych metryk, lecz radzi sobie gorzej przy ostatecznym rozwiązaniu naszego problemu. Jednak analizując całość pracy algorytmów łatwo zauważyć, że przetwarzanie obrazu jest znacznie szybsze i mniej czasochłonne.