

Sprawozdanie

Informatyka w medycynie

Projekt: Wykrywanie naczyń krwionośnych dna siatkówki oka

Wykonanie: Bartosz Górką, index. 127228

Spis treści

Spis treści	2
Cel projektu	3
Charakterystyka aplikacji okienkowej	3
Bazowa wersja przetwarzania	4
Statystyka	7
Czulość (recall, sensitivity)	7
Specyficzność (specificity)	7
Częstość fałszywych alarmów (false positive rate)	7
Częstość fałszywych odkryć (false discovery rate)	7
Precyzja	7
Efekt częstości występowania	7
Dokładność (accuracy)	7
Współczynnik korelacji Matthews'a	7
Klasifikatory	8
Klasifikator własny	8
Klasifikator k-najbliższych sąsiadów	8
Klasifikator Random Forest	8
Walidacja skrośna	8
Rezultaty przetwarzania	10
Obraz 01	10
Obraz 02	14
Obraz 03	18
Obraz 04	22
Obraz 05	26
Obraz 06	30
Obraz 07	34
Obraz 08	38
Obraz 09	42
Obraz 10	46
Obraz 11	50
Obraz 12	54
Obraz 13	58
Obraz 14	62
Obraz 15	66

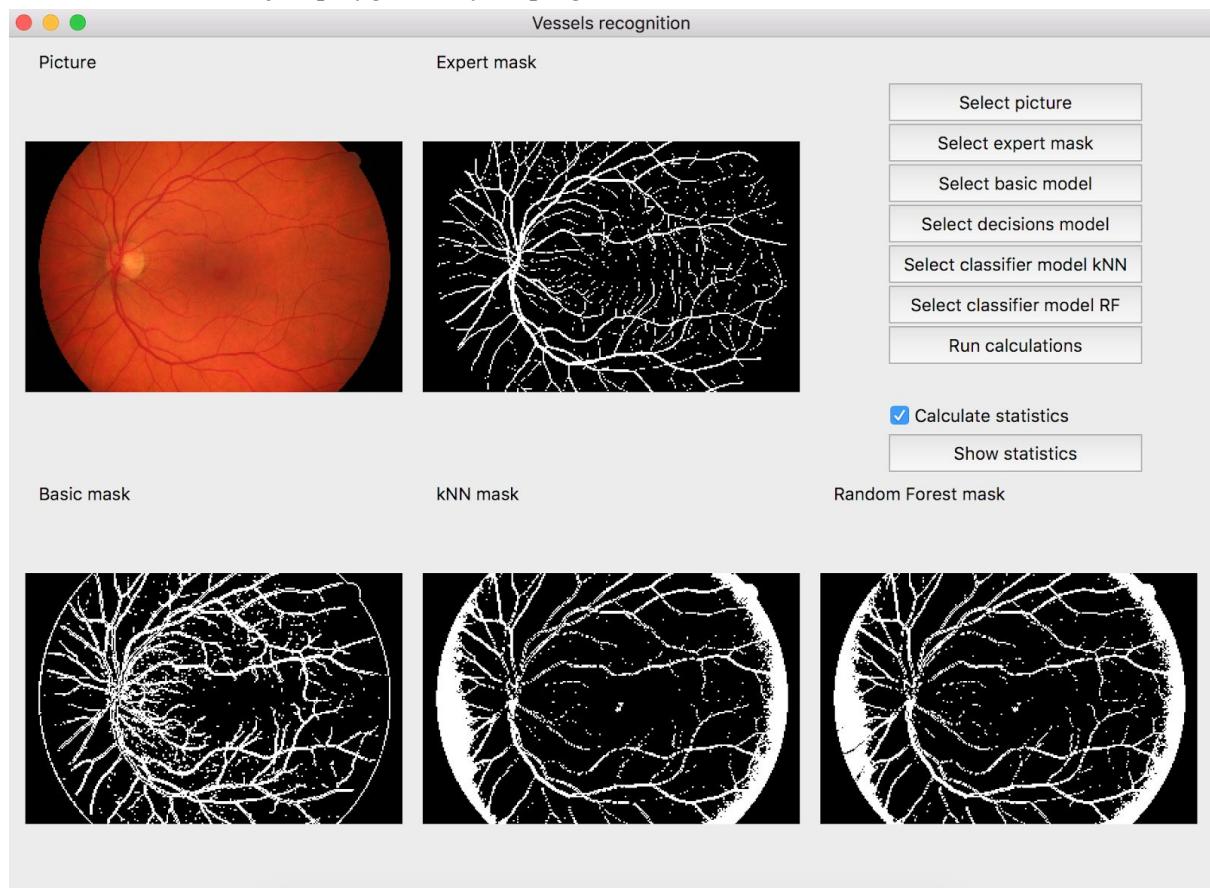
Cel projektu

W ramach przygotowywanego projektu należało dokonać implementacji programu komputerowego umożliwiającego wykrywanie naczyń krwionośnych w wybranym przez użytkownika obrazie wejściowym przedstawiającym dno siatkówki oka.

W wyniku działania aplikacji okienkowej, użytkownik powinien otrzymać binarną maskę odpowiedzi (każdy piksel ma charakteryzować czy jest tam, bądź nie ma naczynia krwionośnego).

Charakterystyka aplikacji okienkowej

Aplikację wykonano z wykorzystaniem biblioteki PyQT 5, w oparciu o język programowania Python. Dla wygody użytkownika przygotowano intuicyjny graficzny interfejs, umożliwiający kontrolowaną interakcję z przygotowanym oprogramowaniem.



Rysunek 1: Główne okno programu z przykładowym rezultatem przetwarzania.

W celu eliminacji problemu z określeniem jakości otrzymanego rezultatu, użytkownik może zdecydować o liczeniu statystyk dla rezultatów działania programu. Po wybraniu opcji "Calculate statistics" program analizuje otrzymane maski, porównując wyniki z maską eksperta.

	Algorithm	Sensitivity	Specificity	FPR	PDR	Accuracy
1	Basic alg.	0.71374	0.89267	0.10733	0.57133	0.87453
2	kNN (3) alg.	0.63117	0.87204	0.12796	0.64246	0.84762
3	RandomFore...	0.67047	0.87020	0.12980	0.63180	0.84995

Rysunek 2: Przykładowe statystyki przetwarzania.

Bazowa wersja przetwarzania

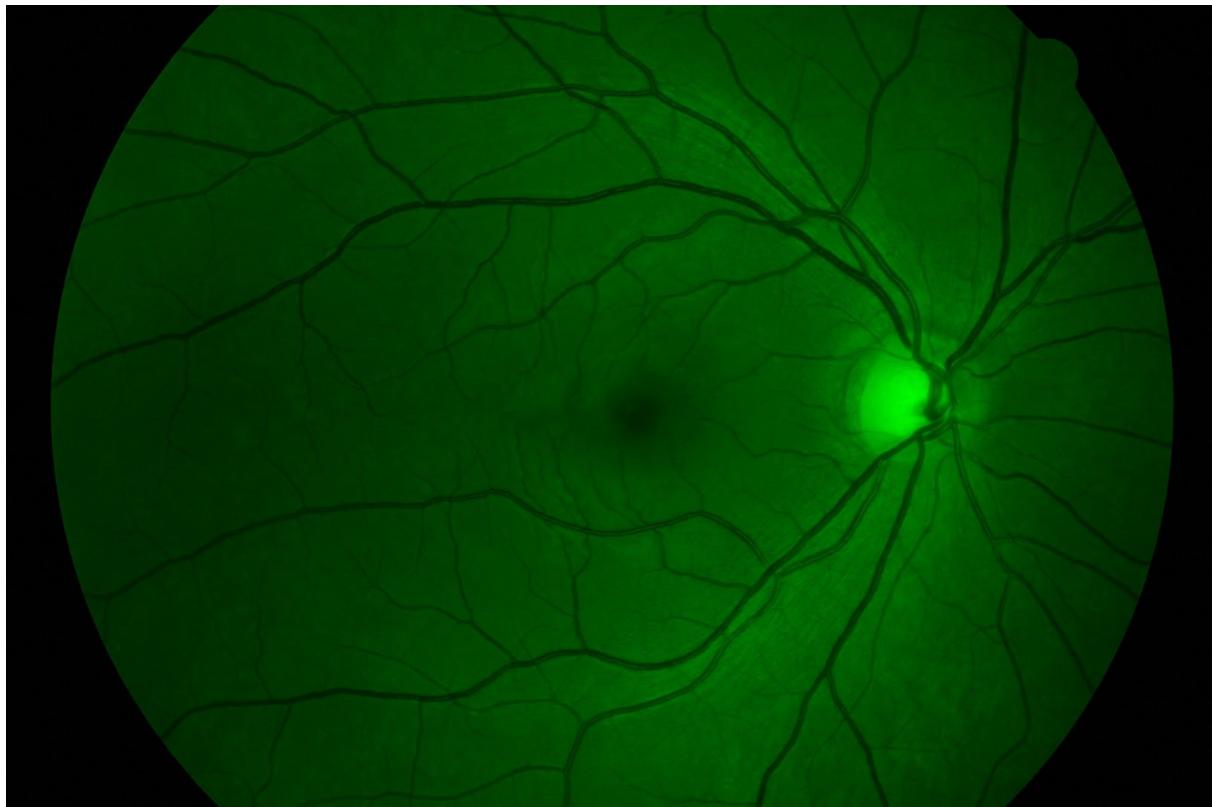
Algorytm w podstawowej wersji oparto o techniki przetwarzania obrazów poznane w ramach przedmioty Komunikacja Człowiek-Komputer. W ramach wstępniego przetworzenia następuje wyodrębnienie kanału zielonego, który zawiera najbardziej istotne informacje z punktu dalszego przetwarzania. Ponadto, w celu poprawy jakości przetwarzania zastosowano kontrast oraz rozjaśnienie obrazu. Dzięki wstępemu przetworzeniu obrazu możliwe było wyeliminowanie dodatkowych artefaktów tła - przy zachowaniu informacji o cieńszych naczyniach, bardzo łatwych do pominięcia przez program.

Po wczesnej obróbce materiału graficznego, dane poddawane są działaniu filtru *Canny*, który umożliwia wyekstrahowanie krawędzi. Jako progi ustawiono wartości 150 oraz 255. Użyto również aperture dla operatora *Sobel* z wartością 5. Następnym krokiem jest transformacja morfologiczna w postaci *dylatacji* z maską 3x3 piksele w kształcie elipsy. W celu oczyszczania nadmiarowych szumów, zastosowano *medianBlur* o rozmiarze 5 pikseli. Ostatnie czynności w przetwarzaniu to ponowna *dylatacja*, tym razem z maską 9x9 pikseli. W celu zagwarantowania binarnej odpowiedzi - zastosowano progowanie.

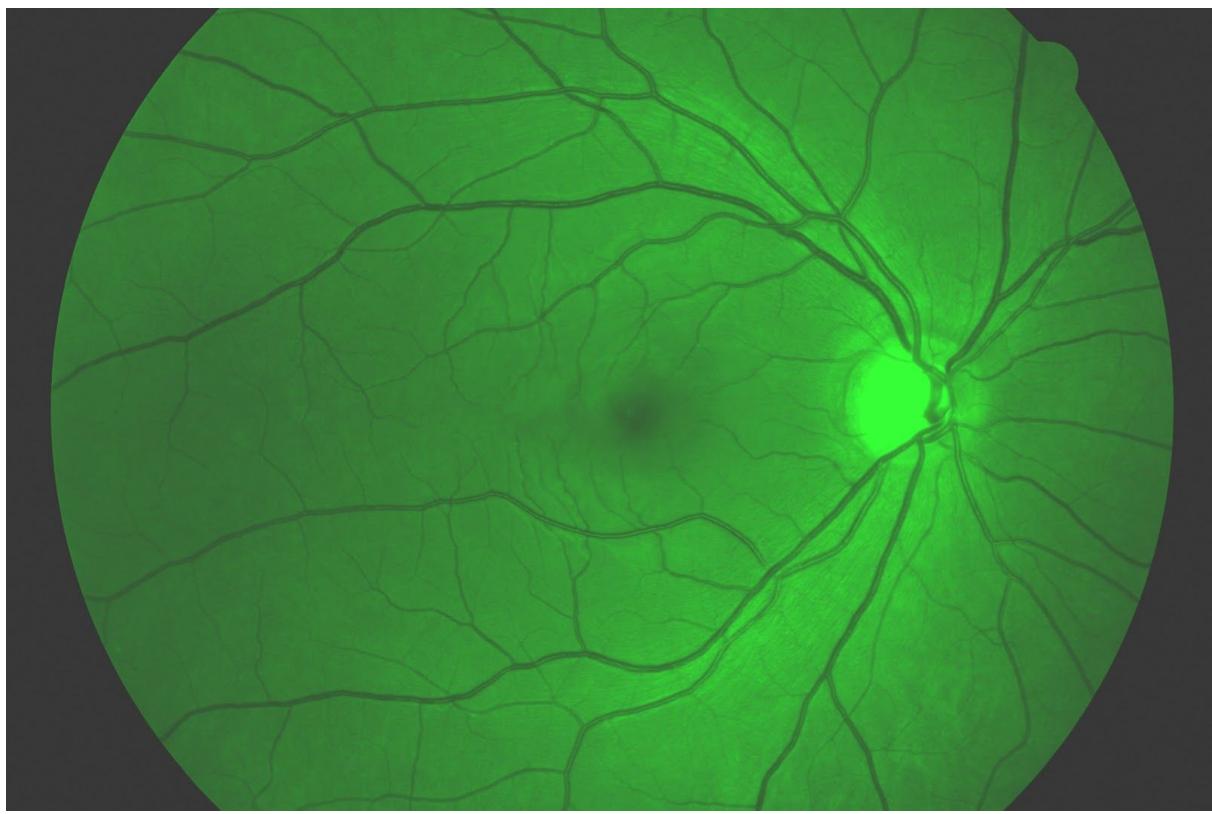
W przebiegu doświadczeń z algorytmem zastosowano również filtr *Frangi'ego* - lecz po przeanalizowaniu wyników jego przetwarzania, zrezygnowano z użycia. Wydłuża on czas przetwarzania, a dodatkowo powoduje zwielokrotnienie naczyń - wąskie naczynia stają się szersze, pomyłki są zwielokrotniane co skutkuje pogorszeniem statystyk przetwarzania i samego efektu wizualnego.



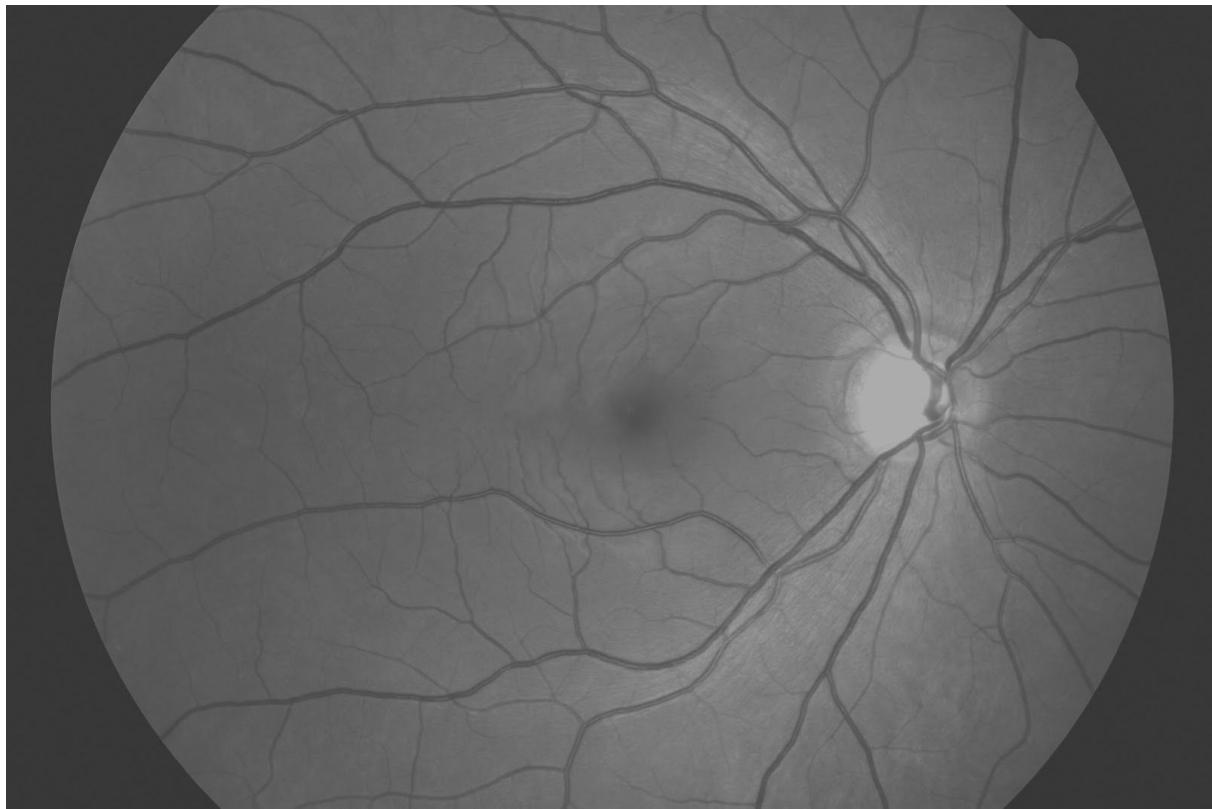
Rysunek 3: Oryginalny obraz dna siatkówki oka



Rysunek 4: Obraz z wyszczególnionym kanałem zielonym



Rysunek 5: Materiał graficzny po aplikacji kontrastu oraz rozjaśnienia w celu eliminacji szumów



Rysunek 6: Materiał graficzny do dalszej analizy - po przejściu na skalę szarości.

Statystyka

Aby móc porównywać niezależnie od względów użytkownika jakość otrzymanych masek binarnych, wprowadzono szereg statystyk, które ułatwiają ten proces.

Czułość (*recall, sensitivity*)

Prawdopodobieństwo, że klasyfikacja będzie poprawna pod warunkiem, że przypadek jest pozytywny.

Specyficzność (*specificity*)

Prawdopodobieństwo, że klasyfikacja będzie poprawna pod warunkiem, że przypadek jest negatywny.

Częstość fałszywych alarmów (*false positive rate*)

Wskazanie, jak dużą część braku naczyń krwionośnych program uzna za naczynie.

Częstość fałszywych odkryć (*false discovery rate*)

Wskazanie, jak dużą część wyników pozytywnych będzie fałszywa.

Precyzja

Wskazanie jakie jest prawdopodobieństwo, że analizowany piksel odpowiada naczyniu mimo że jest tłem (*positive predictive value*), bądź jak duże prawdopodobieństwo, że jednak piksel nie zawiera naczynia, mimo że faktycznie zawiera (*negative predictive value*).

Efekt częstości występowania

W problemie mamy do czynienia z sytuacją, kiedy jedna z klas (brak naczynia krwionośnego) znacznie przewyższa liczebnością klasę naczyń krwionośnych. Skutkuje to zaburzeniem podstawowych miar oceny jakości - trzeba też analizować je jednocześnie. Dzięki obliczaniu ponadto dokładności (*accuracy*) klasyfikacji możemy za pomocą jednej wartości wyrazić ocenę działania algorytmów.

Dokładność (*accuracy*)

Prawdopodobieństwo poprawnej klasyfikacji.

Współczynnik korelacji Matthews'a

Współczynnik uwzględnia wyniki zarówno prawdziwie jaki i fałszywie pozytywne i negatywne. Jest traktowany jako zrównoważona miara, co ma znaczenie w naszym przypadku. Jest współczynnikiem korelacji pomiędzy obserwowanymi i przewidywanymi klasyfikacjami binarnymi. Współczynnik I odpowiada idealnej klasyfikacji, 0 nie lepiej niż losowe przypisanie wyniku i -1 oznacza całkowitą niezgodę między klasyfikacją i stanem faktycznym.

Klasyfikatory

W początkowej fazie projektu w ramach rozpoznawania używano momentów Hu. Niestety stanowią one znaczne obciążenie obliczeniowe, co skutkuje znacznym wydłużeniem procesu przetwarzania obrazu dna siatkówki oka w postać maski bitowej naczyń krwionośnych. W zmienionej wersji oprogramowania postanowiono wykorzystać wprost informację z wyciętego obszaru obrazu czyli wartości kolorów.

Klasyfikator własny

Pierwszy z klasyfikatorów, bardzo naiwny opiera się na podobieństwo kosinusowym (*cosine similarity*) między analizowanym obszarem obrazu a całą bazą wiedzy dla klasyfikatora z procesu przedstawionego w poprzednim paragrafie. Wybierany jest pierwszy z elementów o największej wartości podobieństwa. Taki sposób implementacji powoduje ogromną złożoność obliczeniową $O(n*m*k)$ gdzie n - wysokość analizowanego obrazu, m - szerokość obrazu, k - liczba zebranych próbek w procesie uczenia (pominięto kwestie obliczania samego podobieństwa).

Klasyfikator k-najbliższych sąsiadów

Drugi klasyfikator zbudowano na podstawie pierwszego, lecz zastosowano klasyfikator k-najbliższych sąsiadów (*KNeighborsClassifier*) dostarczany przez bibliotekę SciKit Learn. Po procesie uczenia, można wykorzystać nauczony klasyfikator do oceny całego wiersza obrazu co znacząco przyspiesza proces przetwarzania. Program analizuje trzech najbliższych sąsiadów, każdy z jednostkową (równe) wagą.

Klasyfikator Random Forest

Ostatnia implementacja, już bardziej rozbudowanego klasyfikatora oparta została na klasyfikatorze *RandomForest* również dostarczanego przez bibliotekę SciKit Learn. Klasyfikator ten jest dokładniejszy i pozwala wyszukiwać bardziej złożone wzorce w obrazie - co jest przydatne w przypadku wykrywania naczyń krwionośnych czyli ciągłości.

Ponadto, klasyfikator ten jest znacznie szybszy w obliczeniach, co pozwala w krótkim czasie dostarczyć użytkownikowi informację zwrotną w postaci maski binarnej naczyń krwionośnych.

Walidacja skrośna

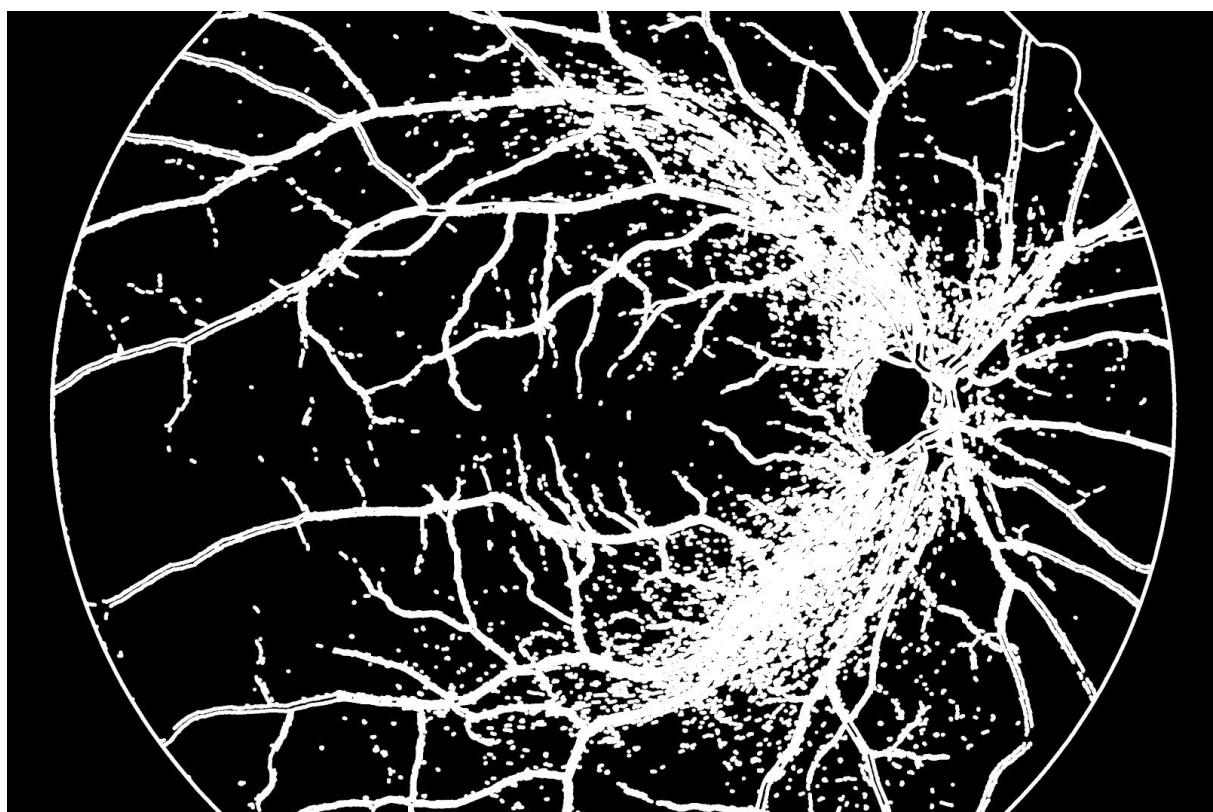
W projekcie przygotowano również dwie wersje dla 10-krotnej walidacji skrośnej (*10-fold cross validation*). Pierwsza została zrealizowana w oparciu o bibliotekę SciKit Learn i funkcję *cross_validation*. Druga to własna implementacja pętli losującej 90% obiektów jako elementy ze zbioru uczącego oraz 10% elementów (pozostałych) jako elementy ze zbioru testowego. Cały proces powtarzany jest dziesięciokrotnie, a na koniec obliczane są statystyki *min*, *max*, *mean* wartości predykcji modelu.

```
{  
    'fit_time': array([  
        144.86081314, 146.55803013, 140.85327601,  
        147.30300117, 147.31020212, 143.93635726,  
        120.62959790, 114.68320704, 136.96049190,  
        148.46707702  
    ]),  
    'score_time': array([  
        0.23089290, 0.22734785, 0.22550011, 0.23215413,  
        0.22035313, 0.22998691, 0.20539331, 0.20657802,  
        0.22864914, 0.24200392  
    ]),  
    'test_score': array([  
        0.96643333, 0.96673333, 0.96903333, 0.96700000,  
        0.96746667, 0.96713333, 0.96666667, 0.96783333,  
        0.96736667, 0.96826667  
    ]),  
    'train_score': array([  
        0.99673333, 0.99675926, 0.99653333, 0.99661481,  
        0.99664074, 0.99659630, 0.99677407, 0.99654074,  
        0.99670370, 0.99645926  
    ])  
}
```

Rysunek 7: Dane z walidacji skrośnej dotyczącej klasyfikatora RandomForest, maska 21x21 pikseli oraz 300 tys. elementów w zbiorze.

Rezultaty przetwarzania

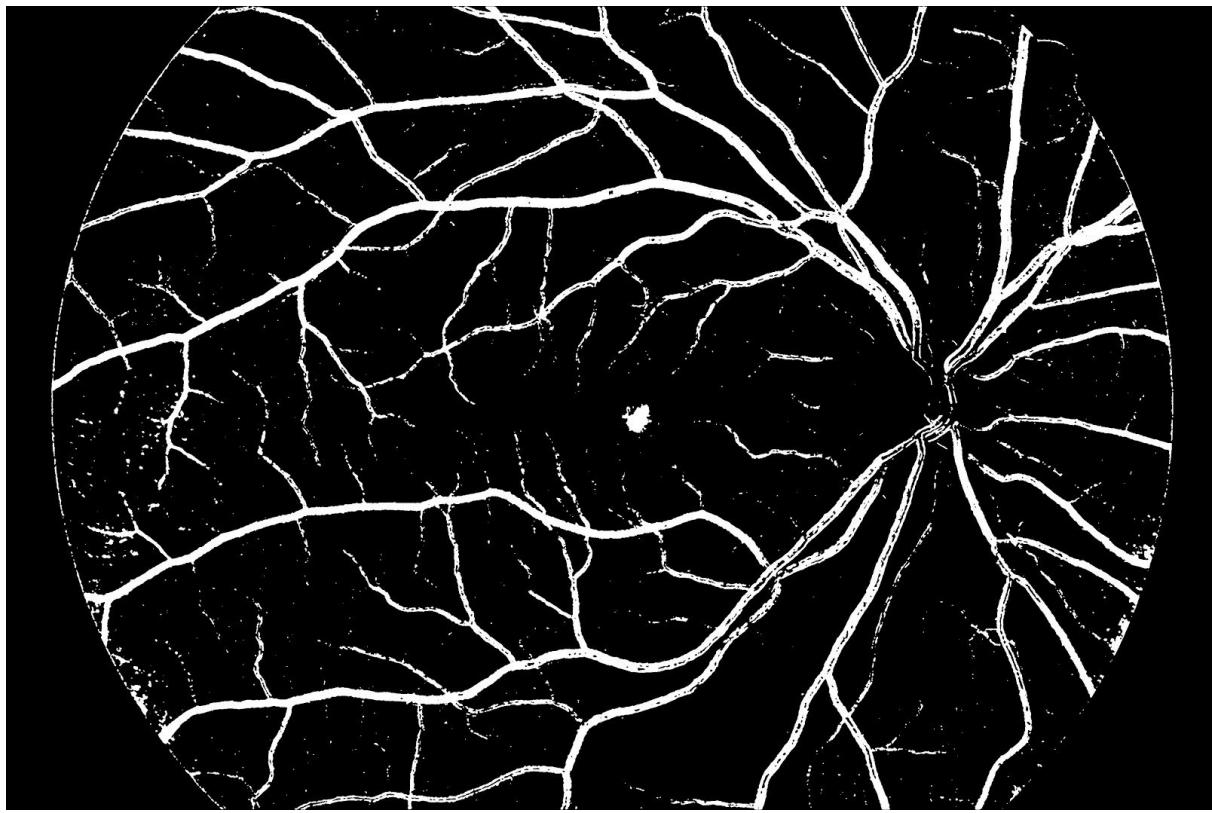
Obraz 01



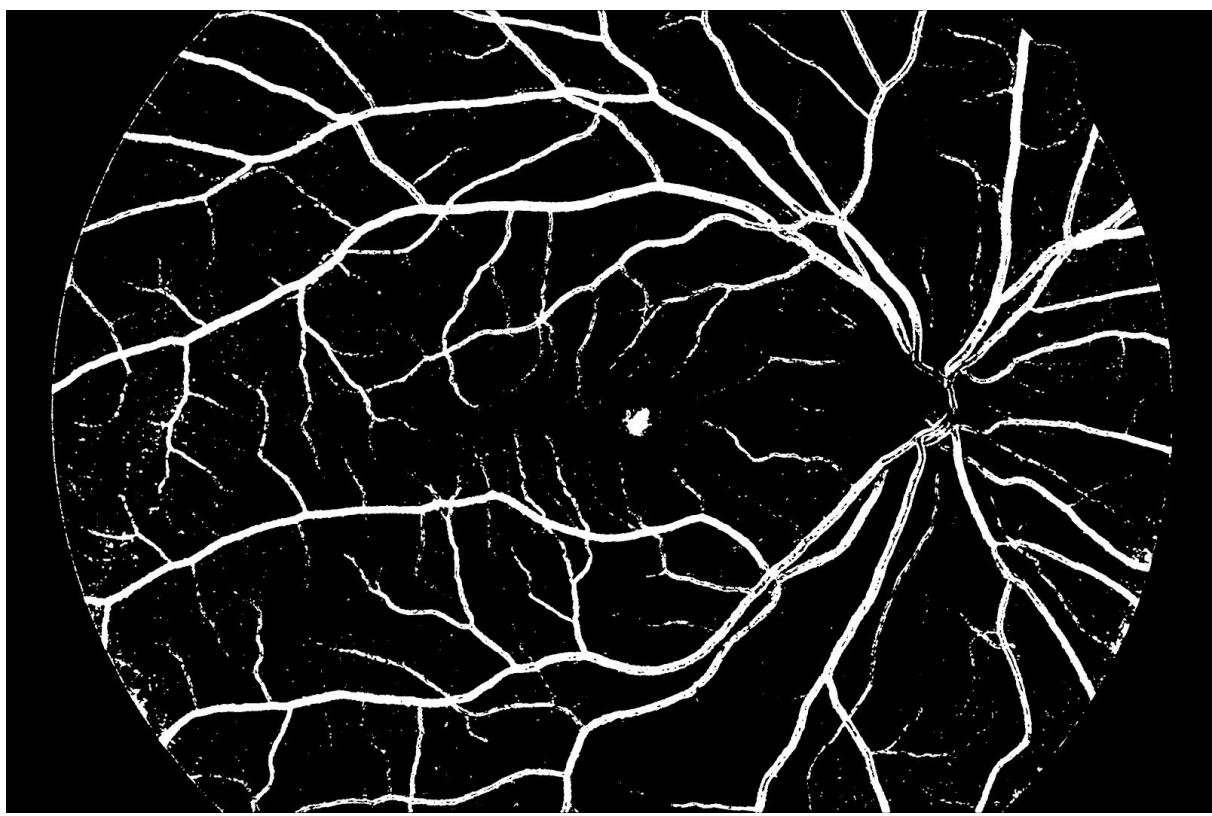
Rysunek 8: Bazowe przetwarzanie



Rysunek 9: Przetwarzanie dla klasyfikatora RandomForest, maska 5×5 pikseli, 1 tys. elementów w zbiorze



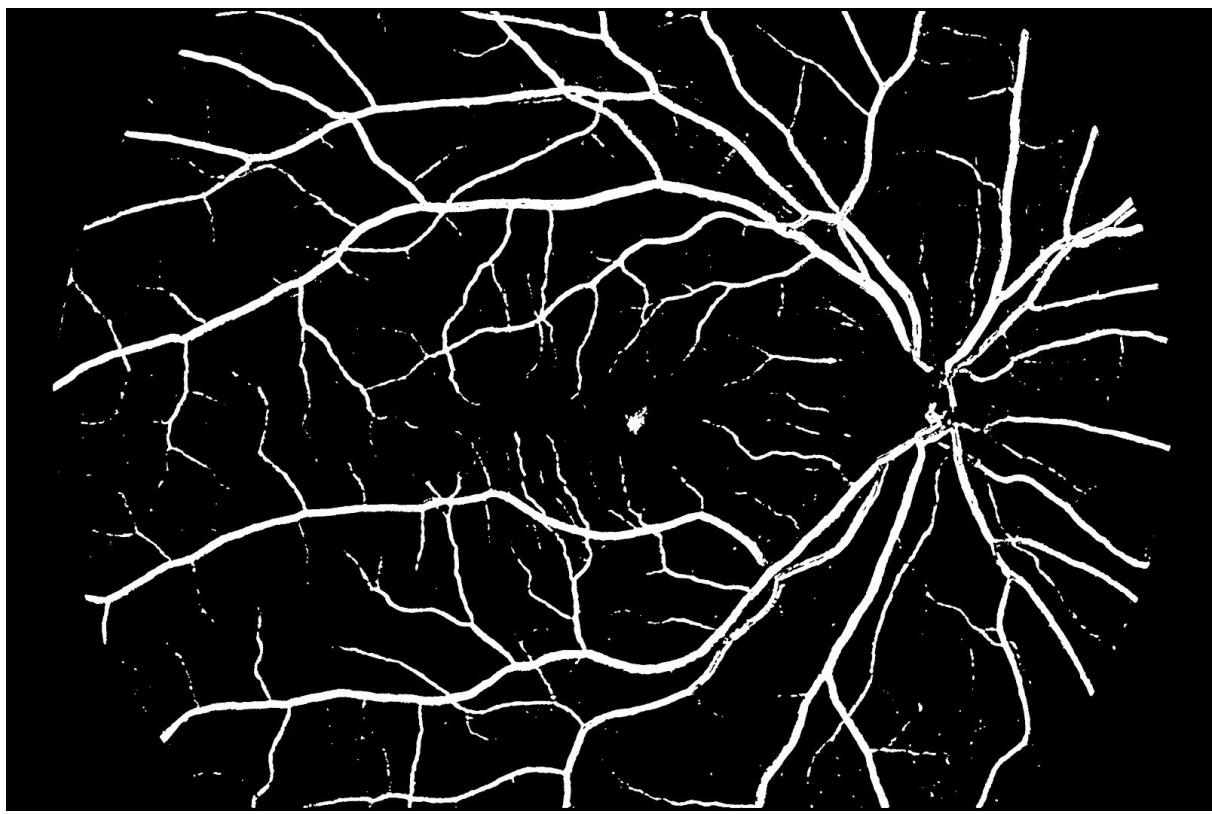
Rysunek 10: Przetwarzanie dla klasyfikatora RandomForest, maska 5×5 pikseli, 10 tys. elementów w zbiorze



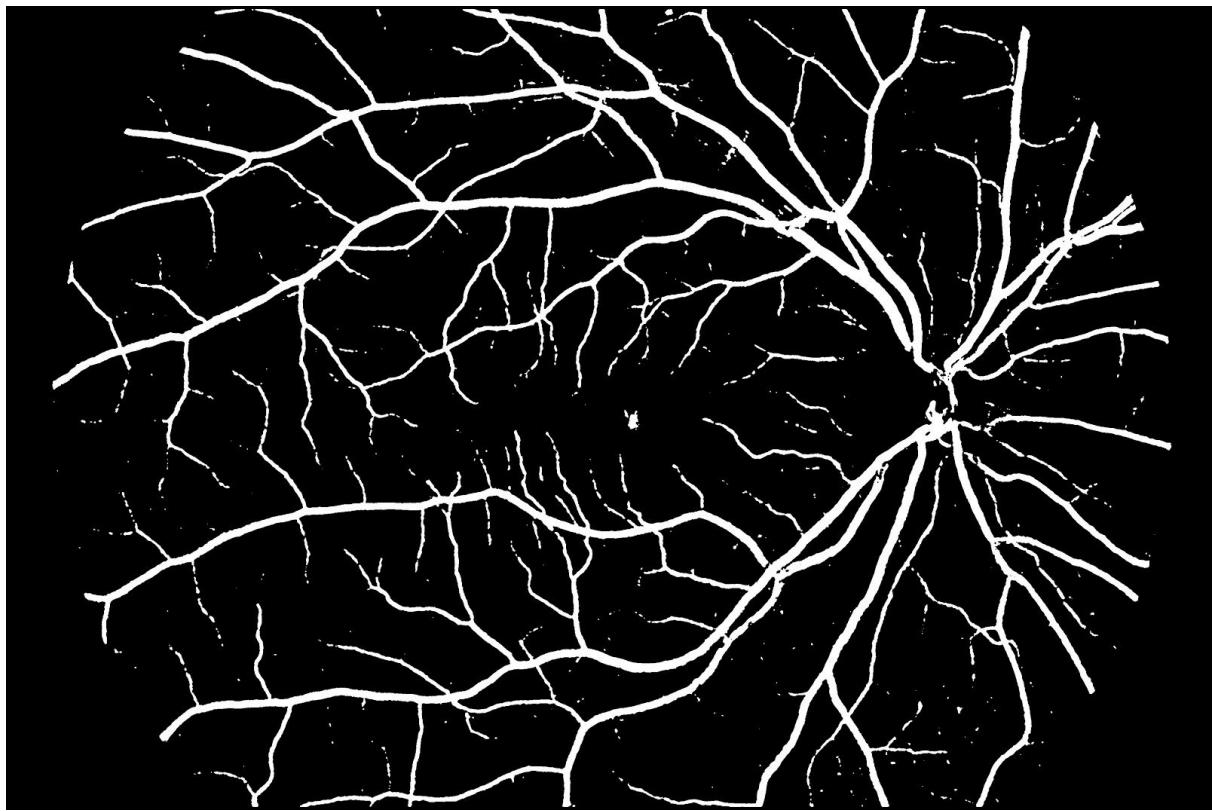
Rysunek 11: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 12: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

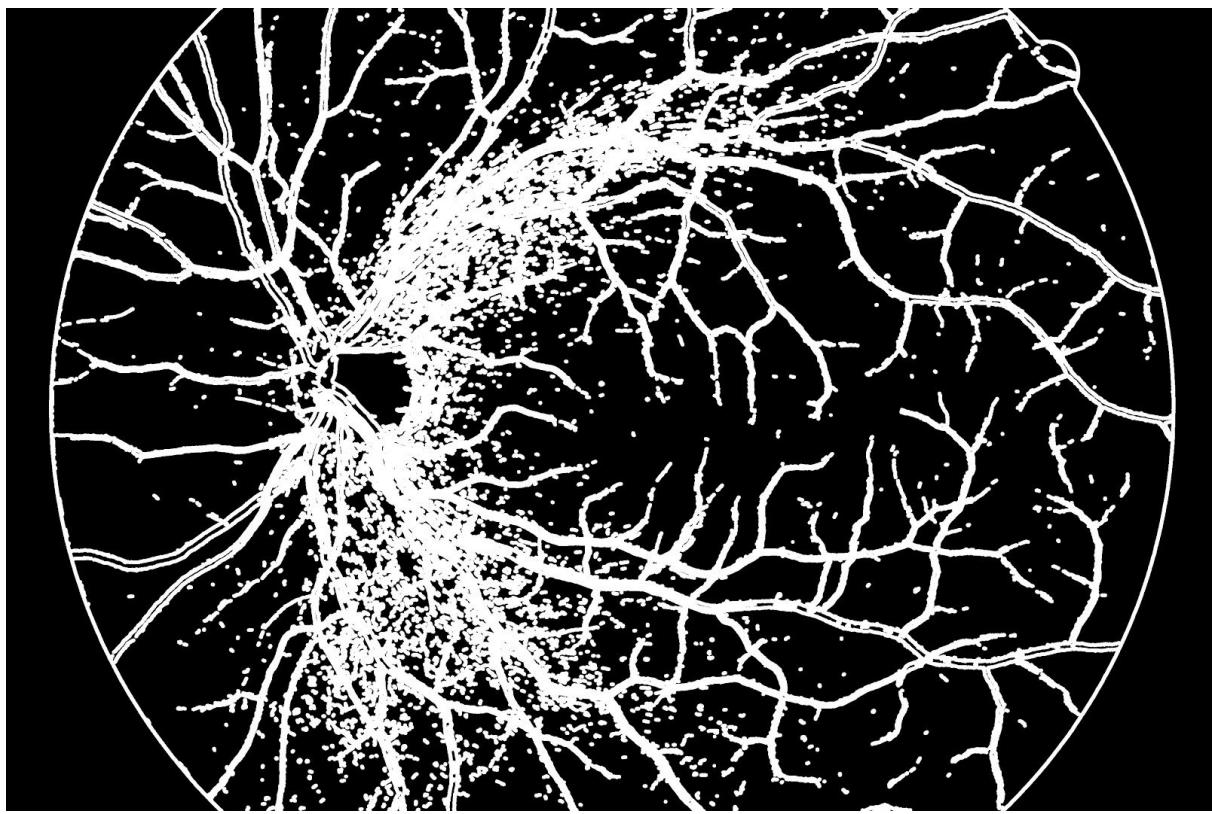


Rysunek 13: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

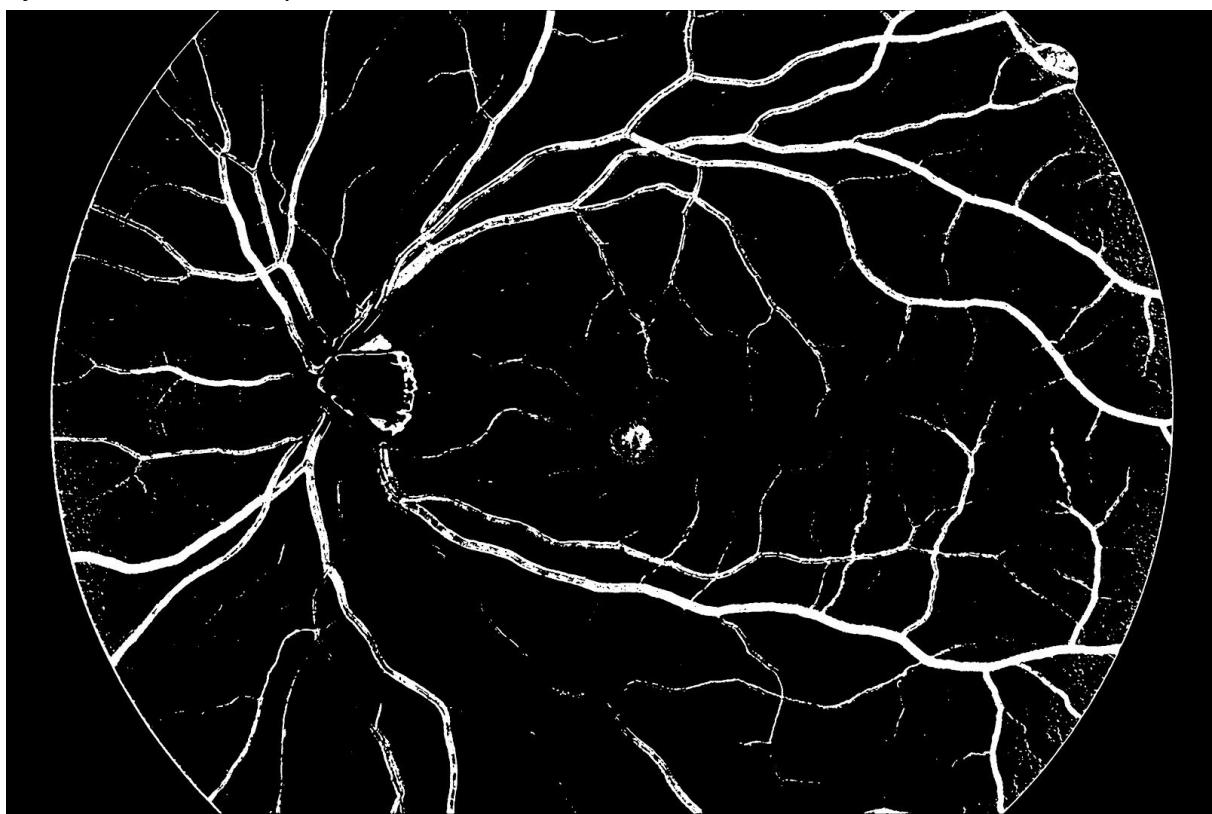


Rysunek 14: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

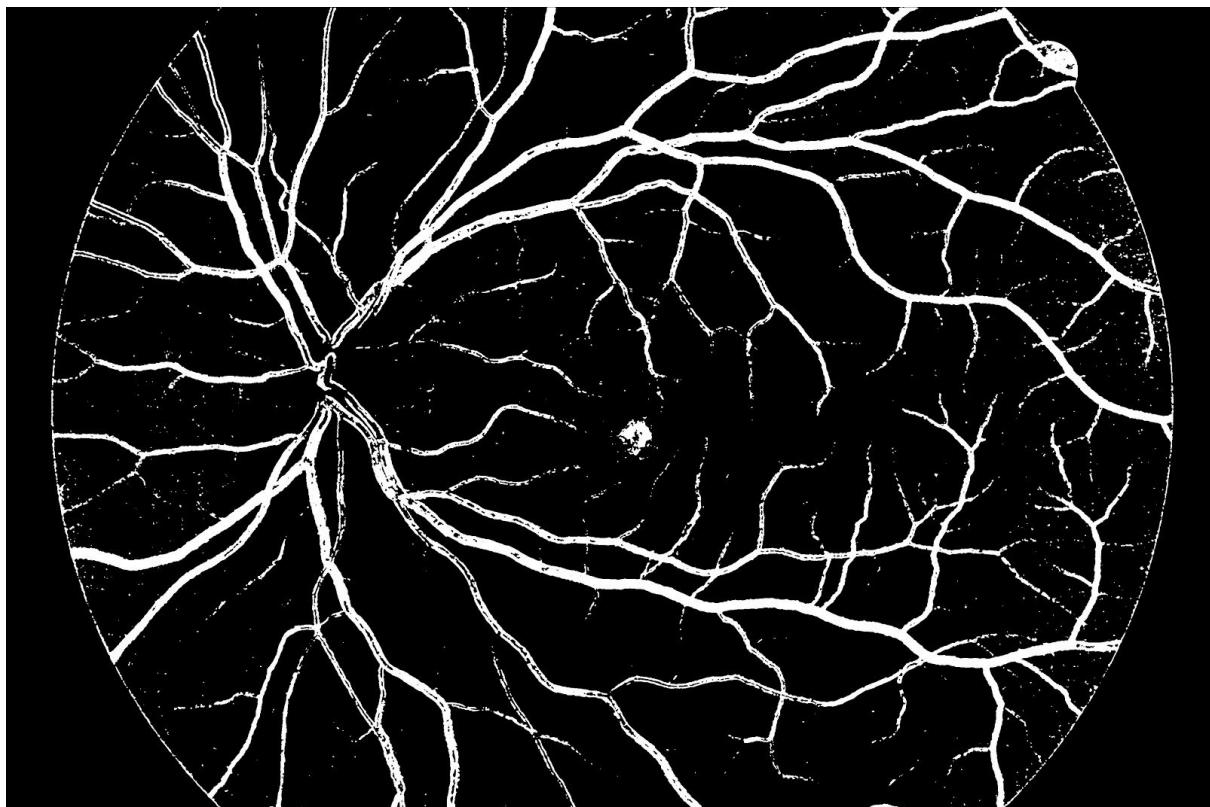
Obraz 02



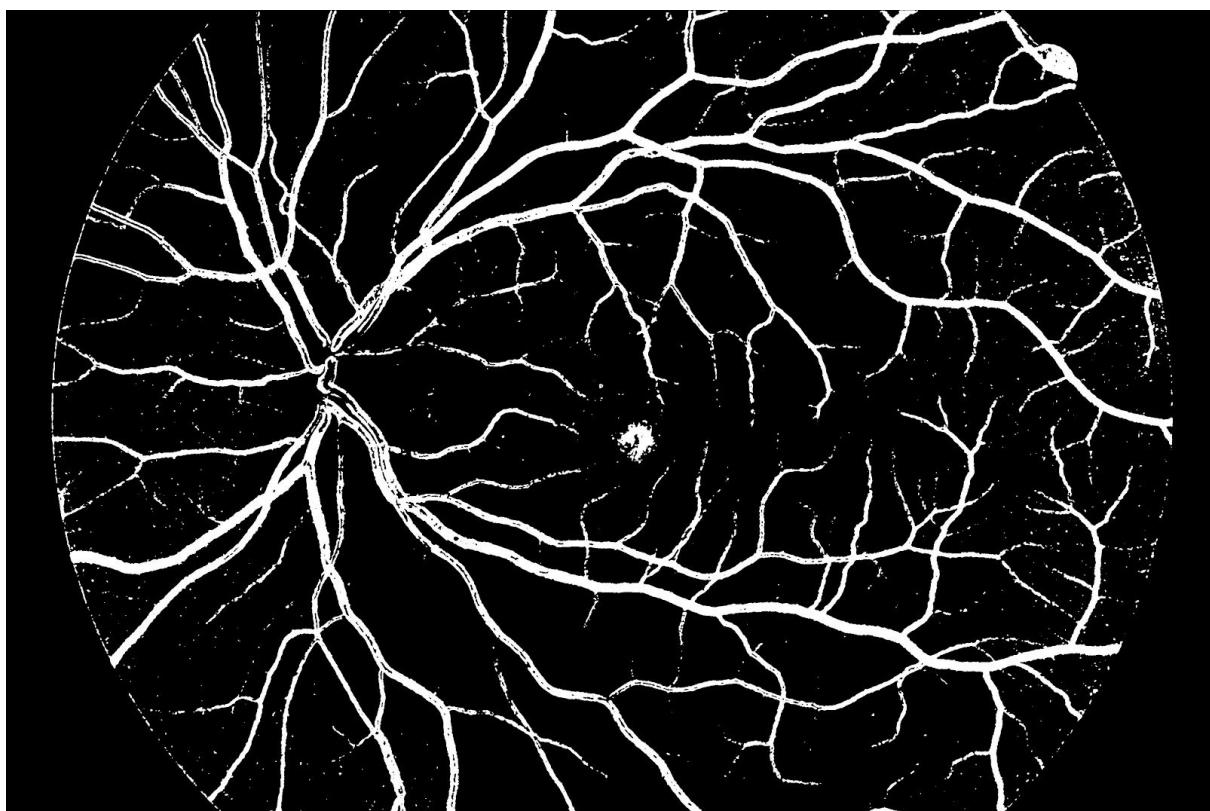
Rysunek 15: Bazowe przetwarzanie



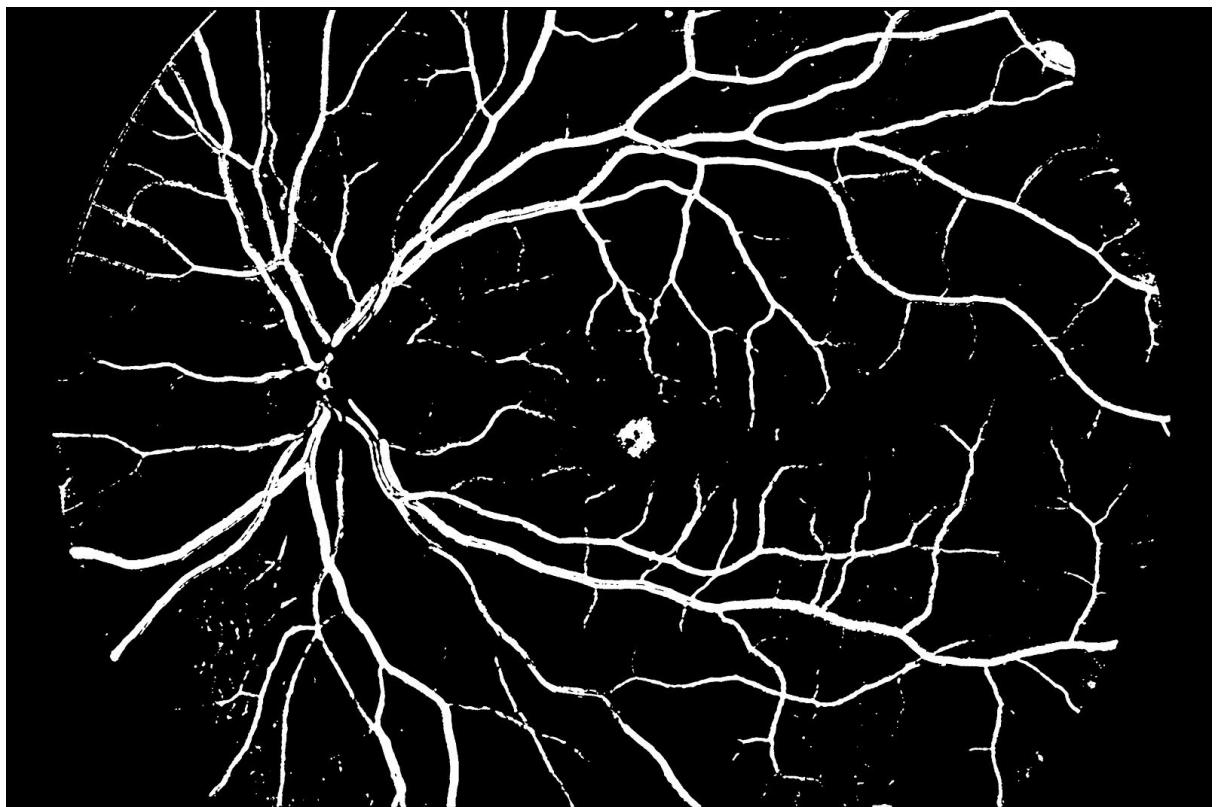
Rysunek 16: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



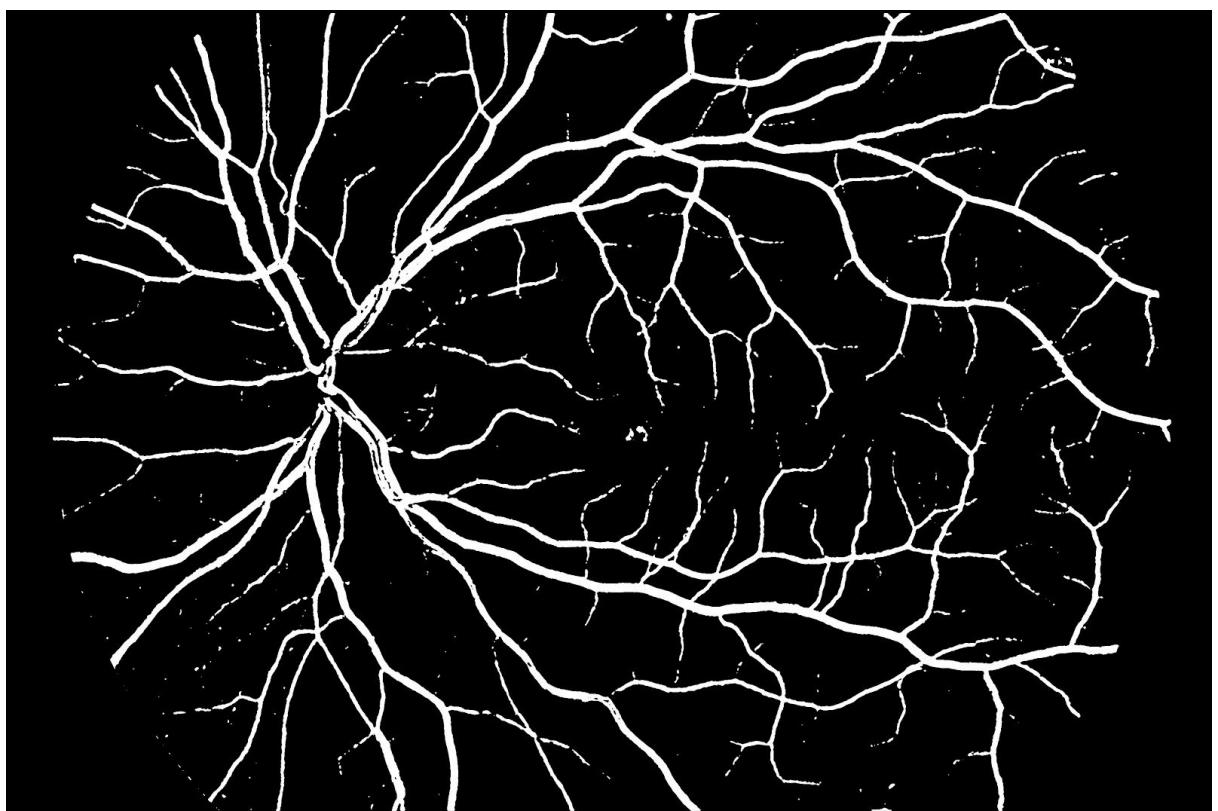
Rysunek 17: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



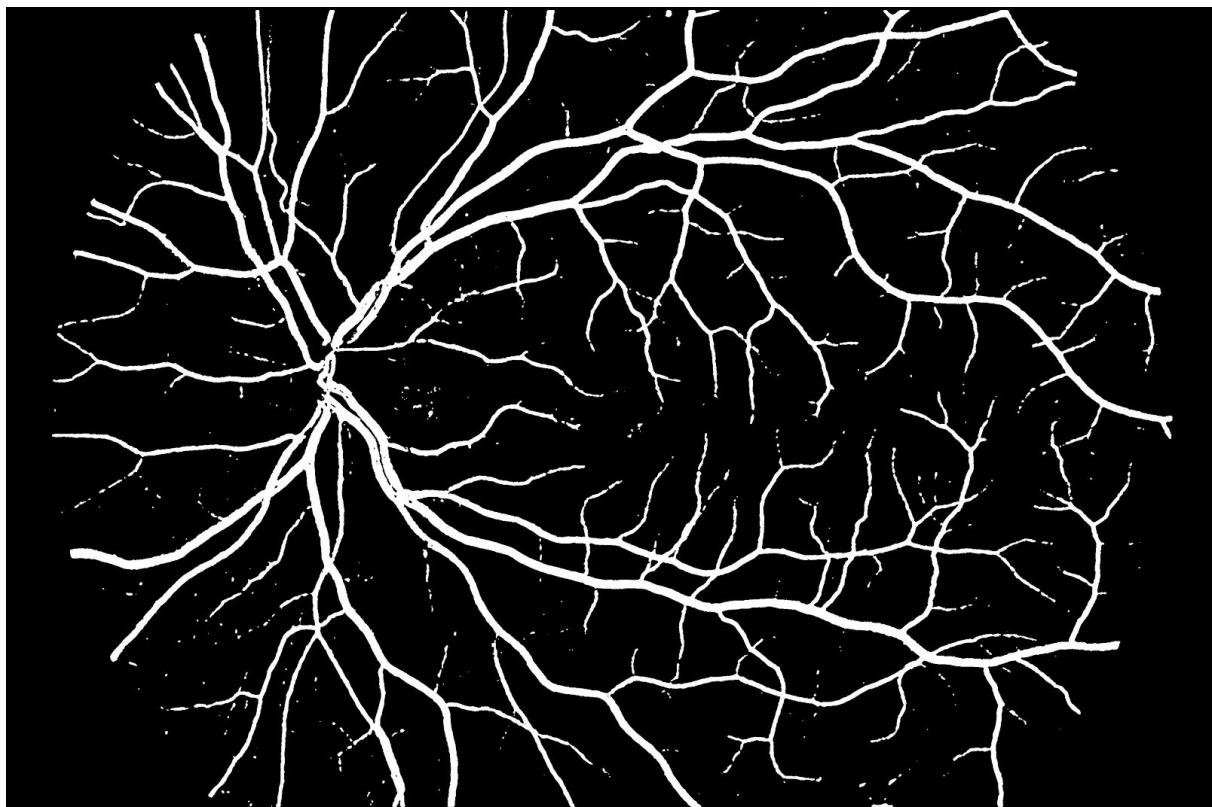
Rysunek 18: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 19: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

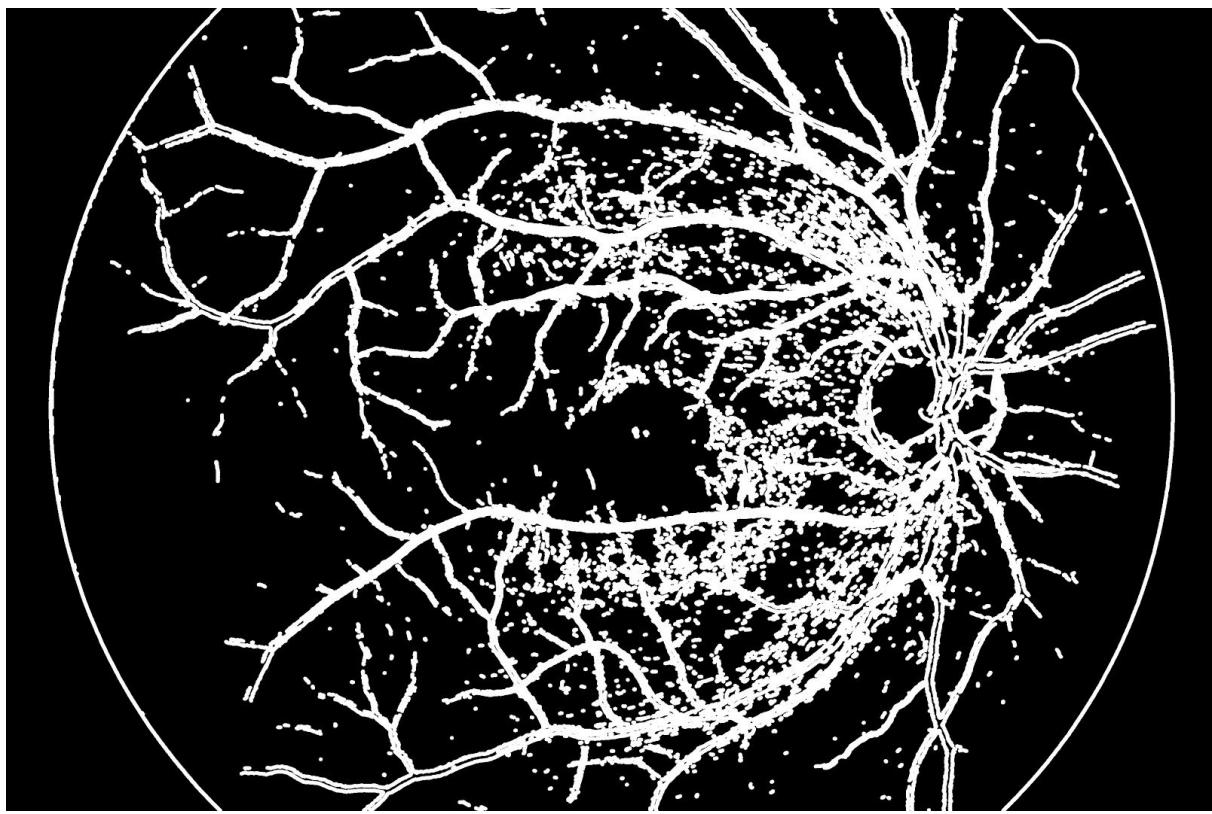


Rysunek 20: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

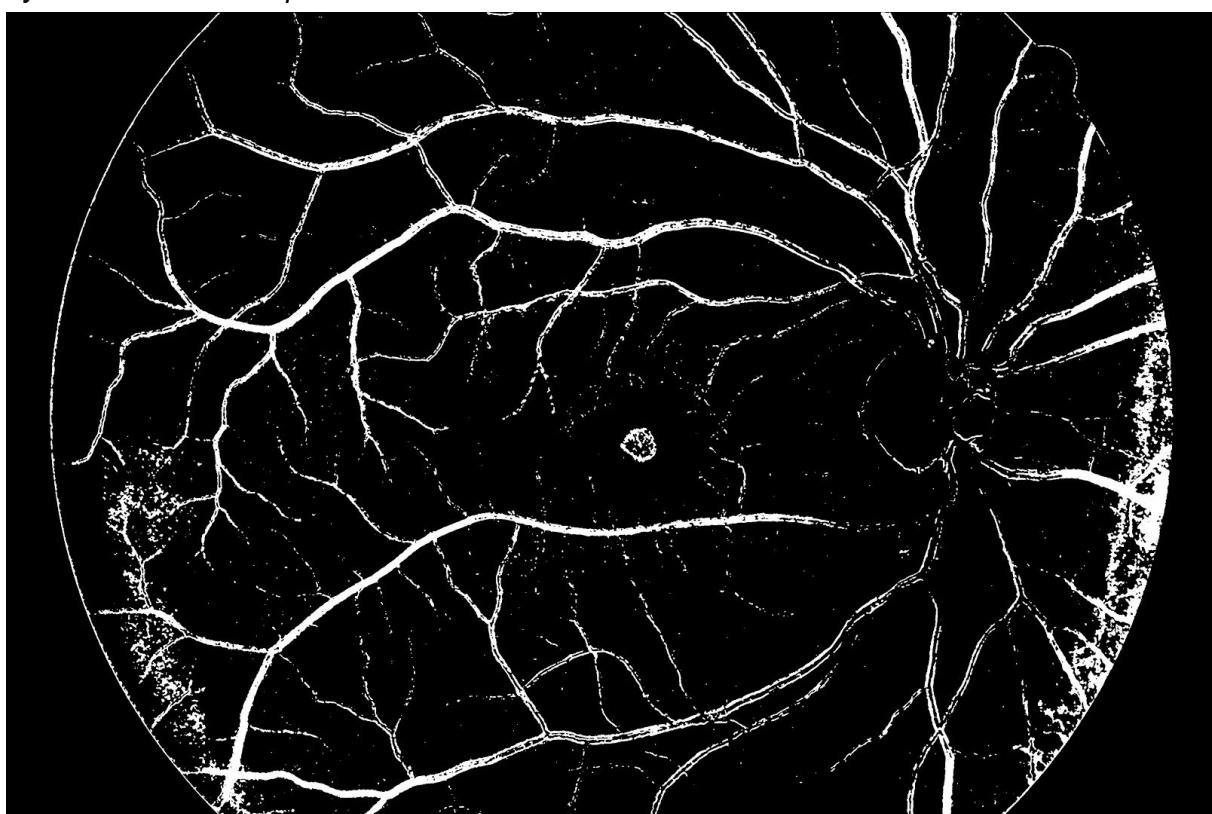


Rysunek 21: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

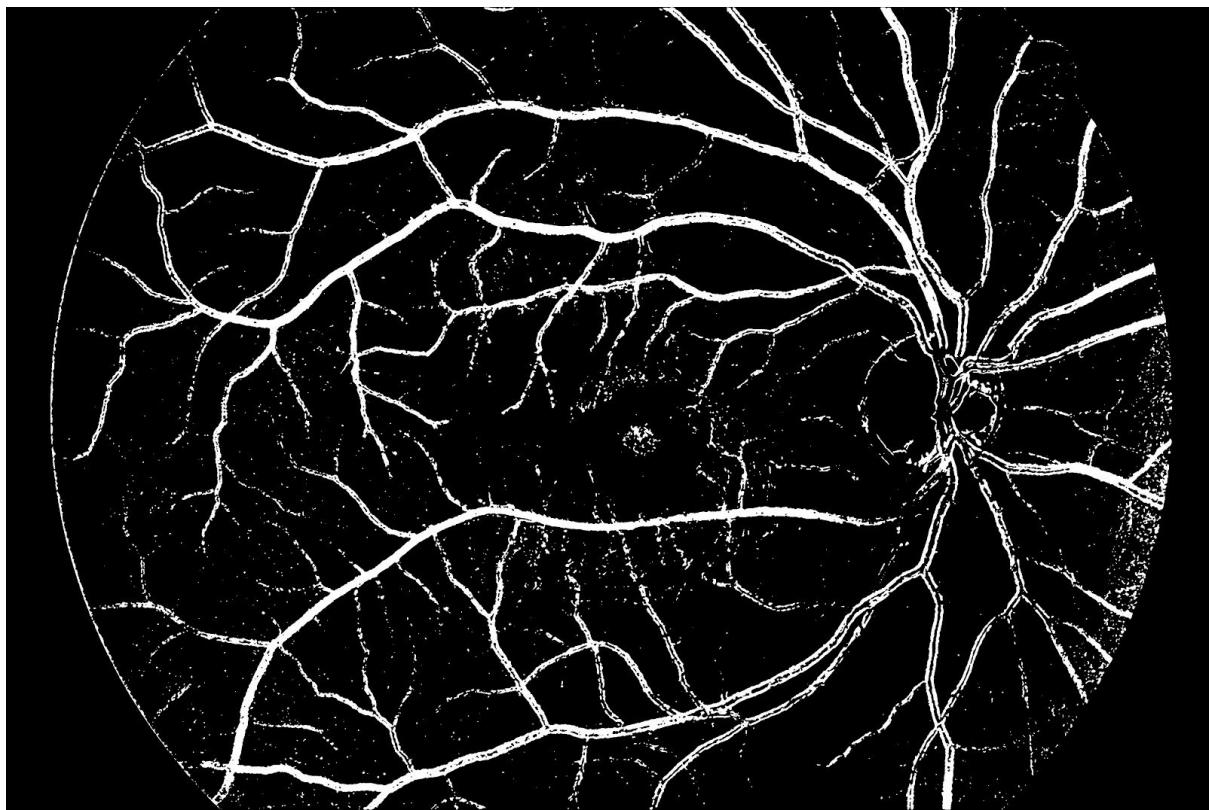
Obraz 03



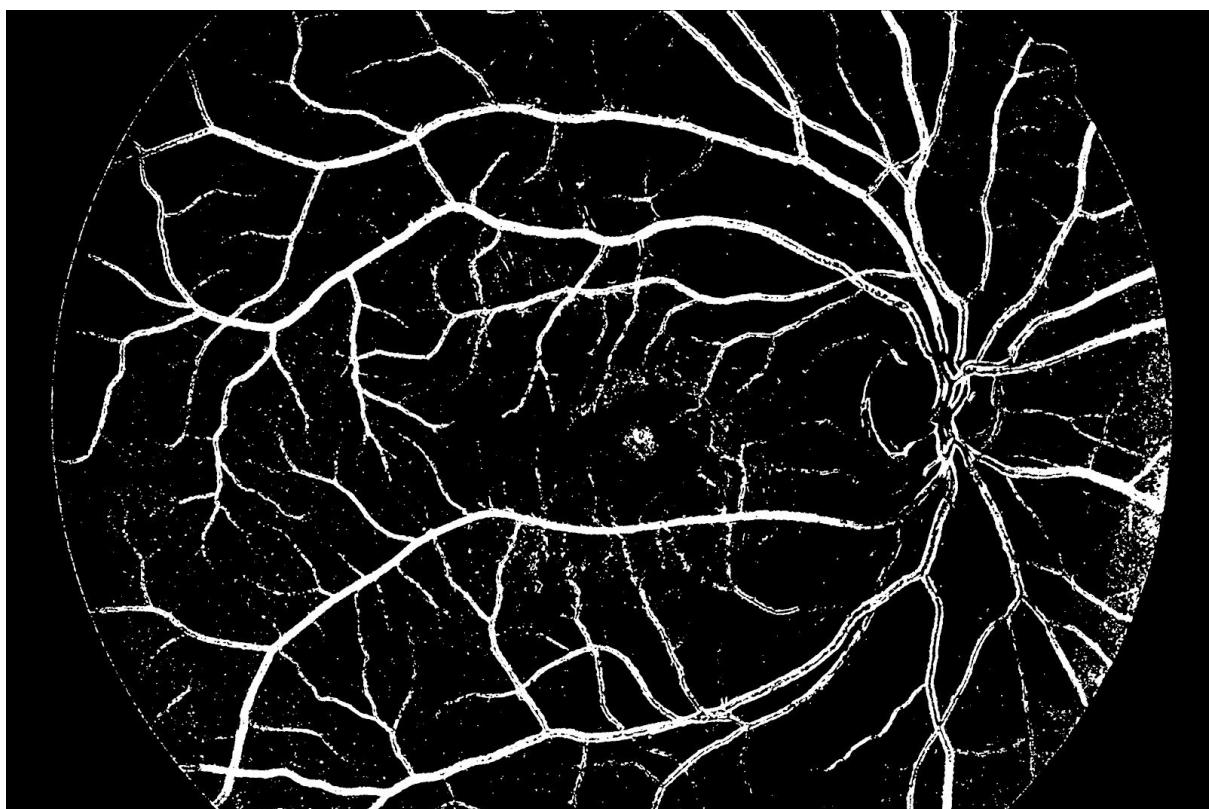
Rysunek 22: Bazowe przetwarzanie



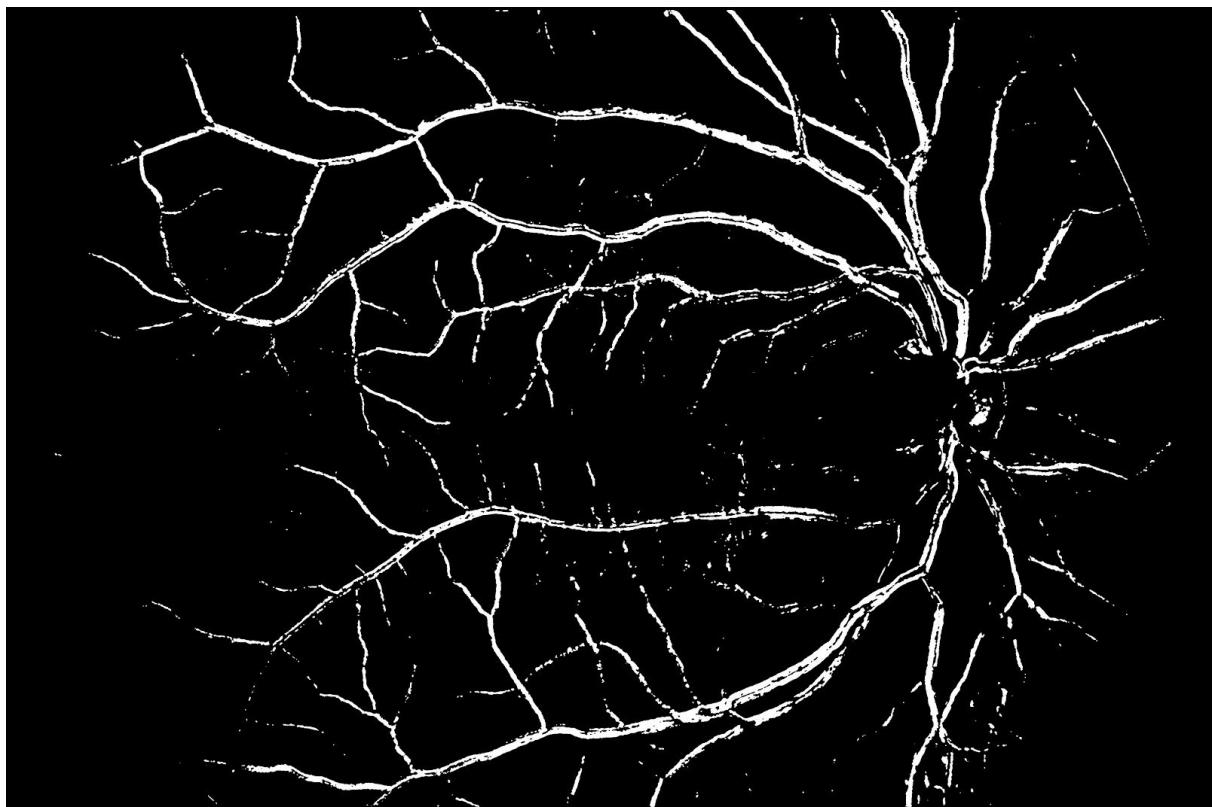
Rysunek 23: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



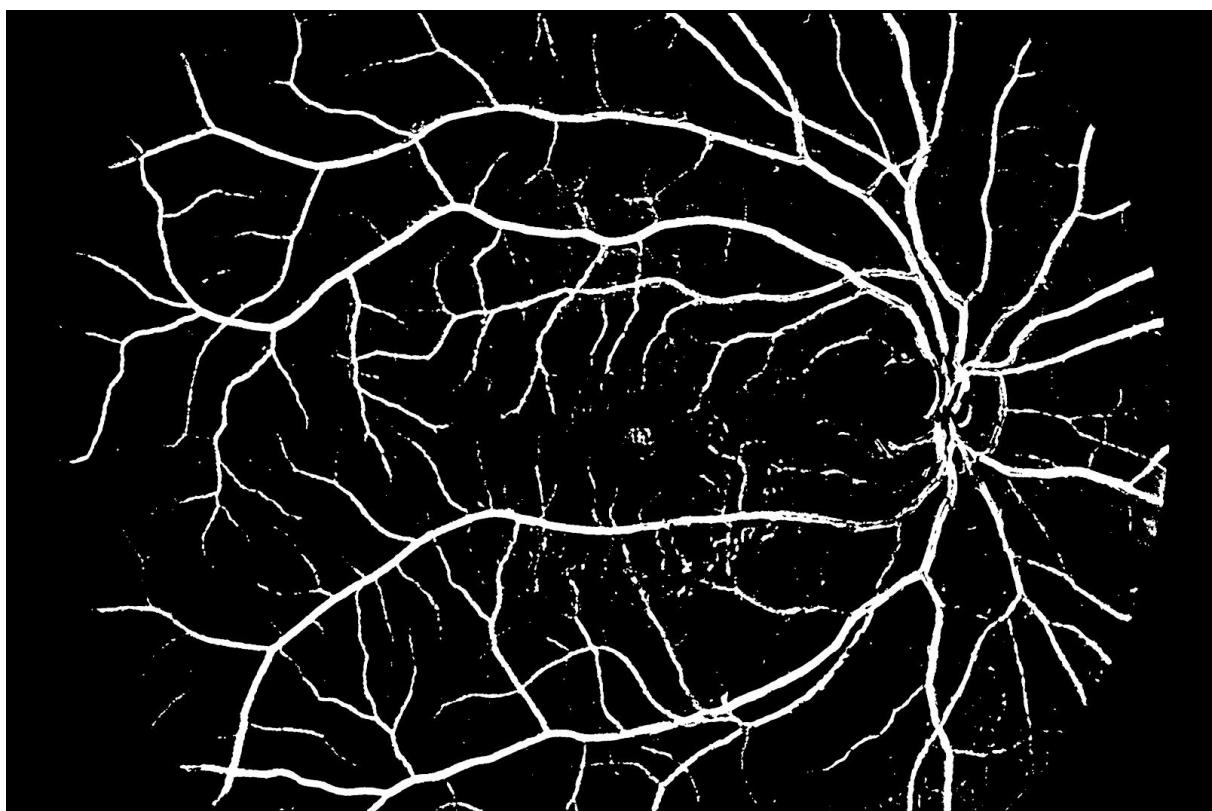
Rysunek 24: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



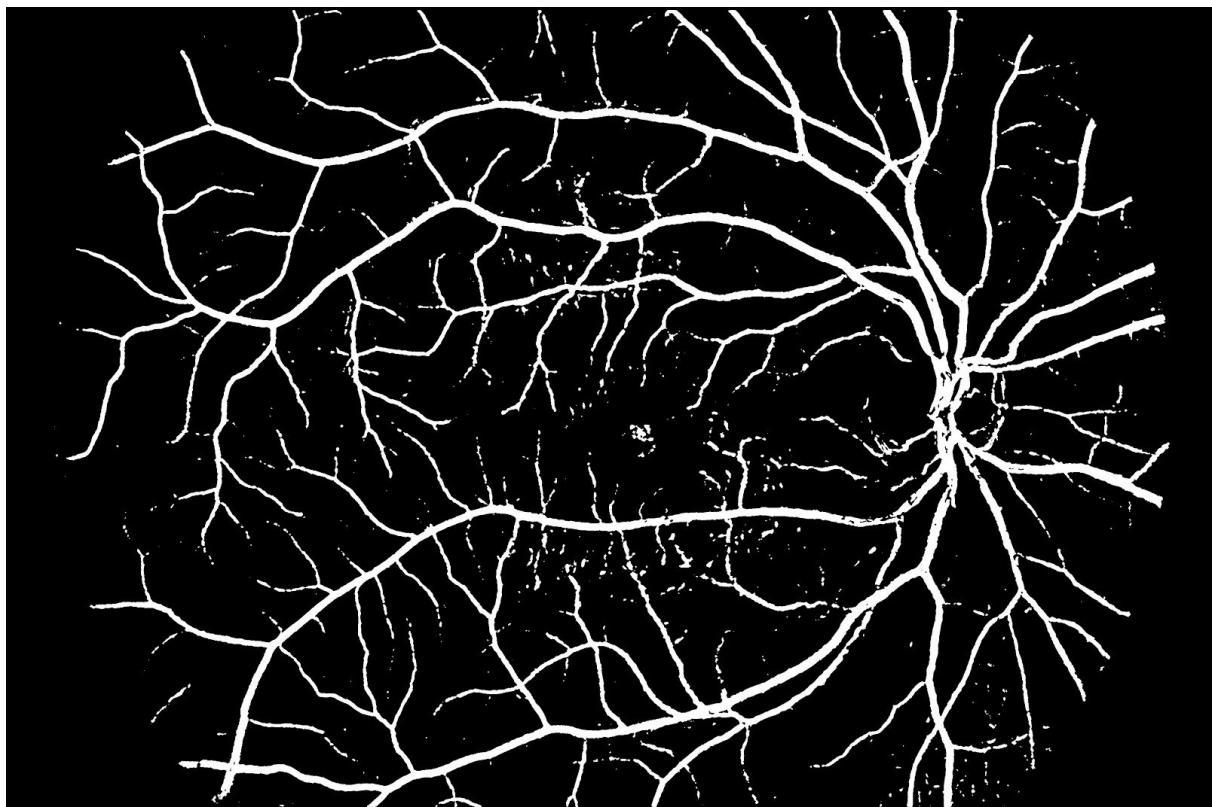
Rysunek 25: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 26: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

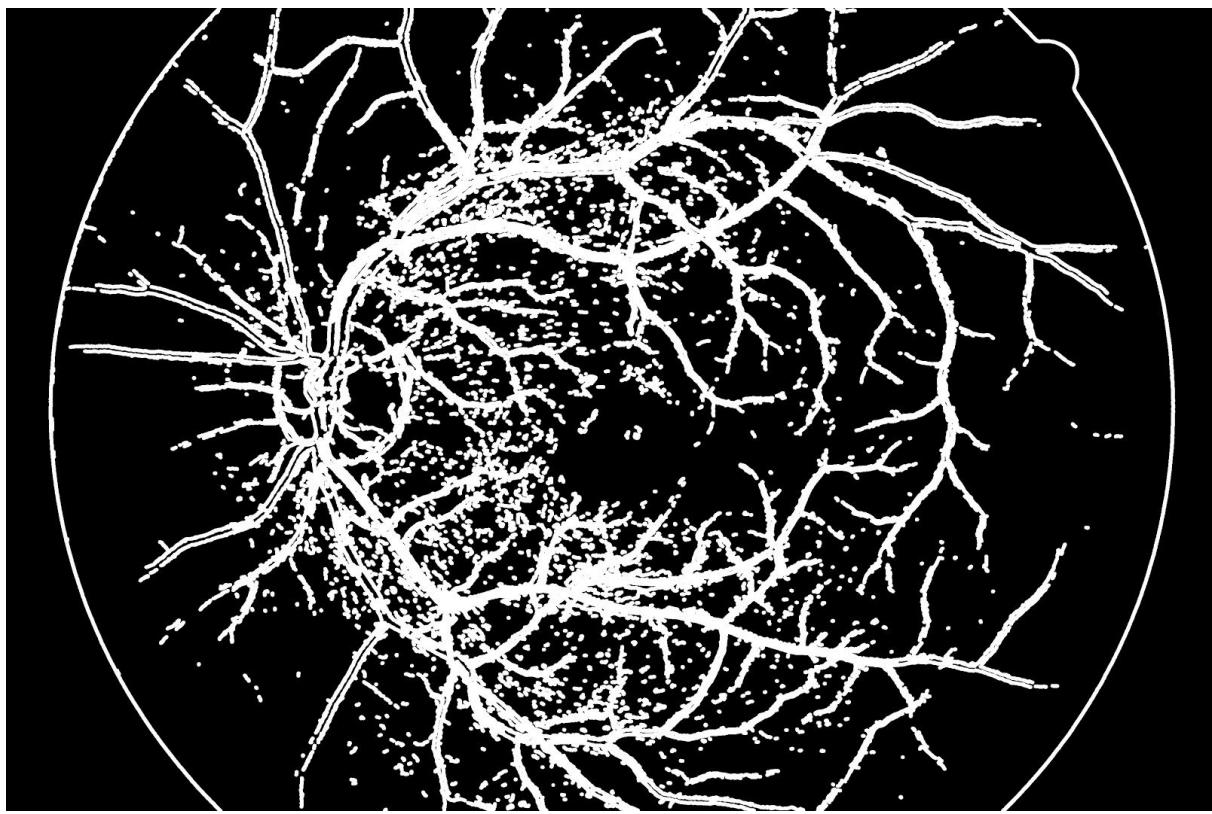


Rysunek 27: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

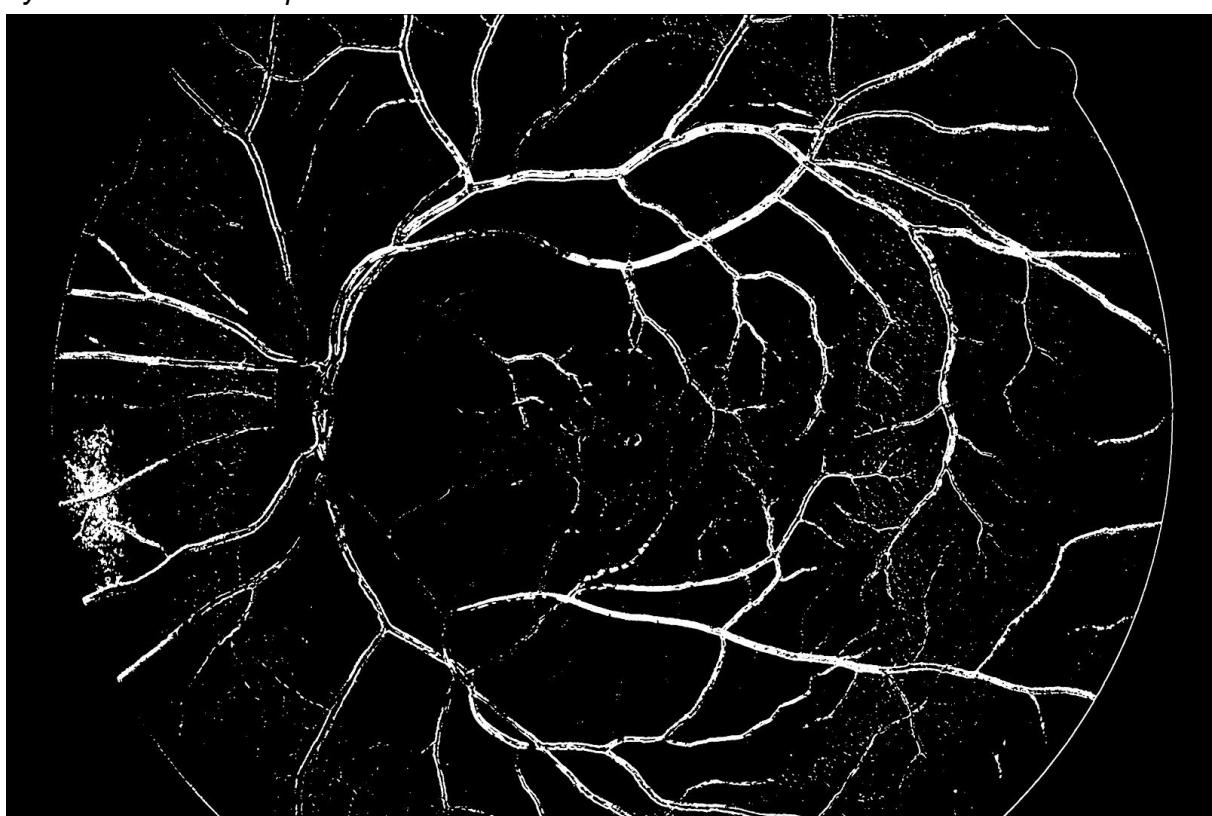


Rysunek 28: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

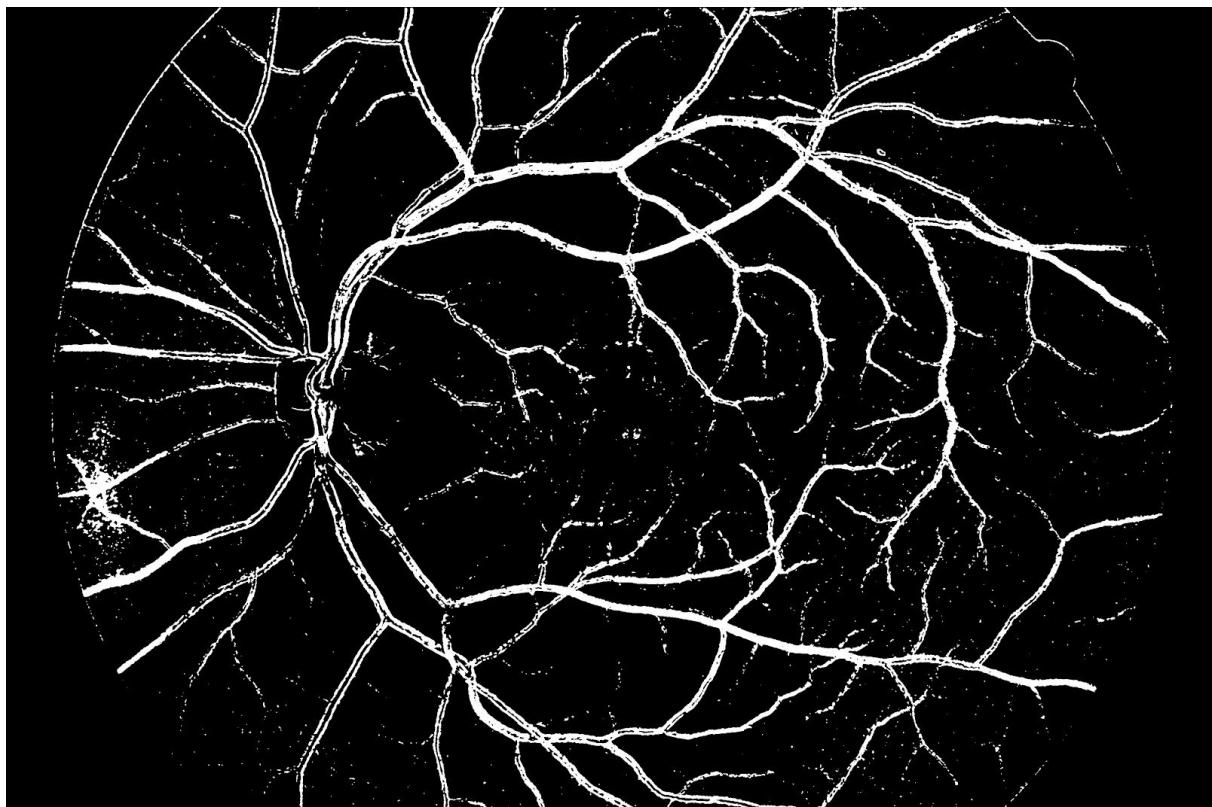
Obraz 04



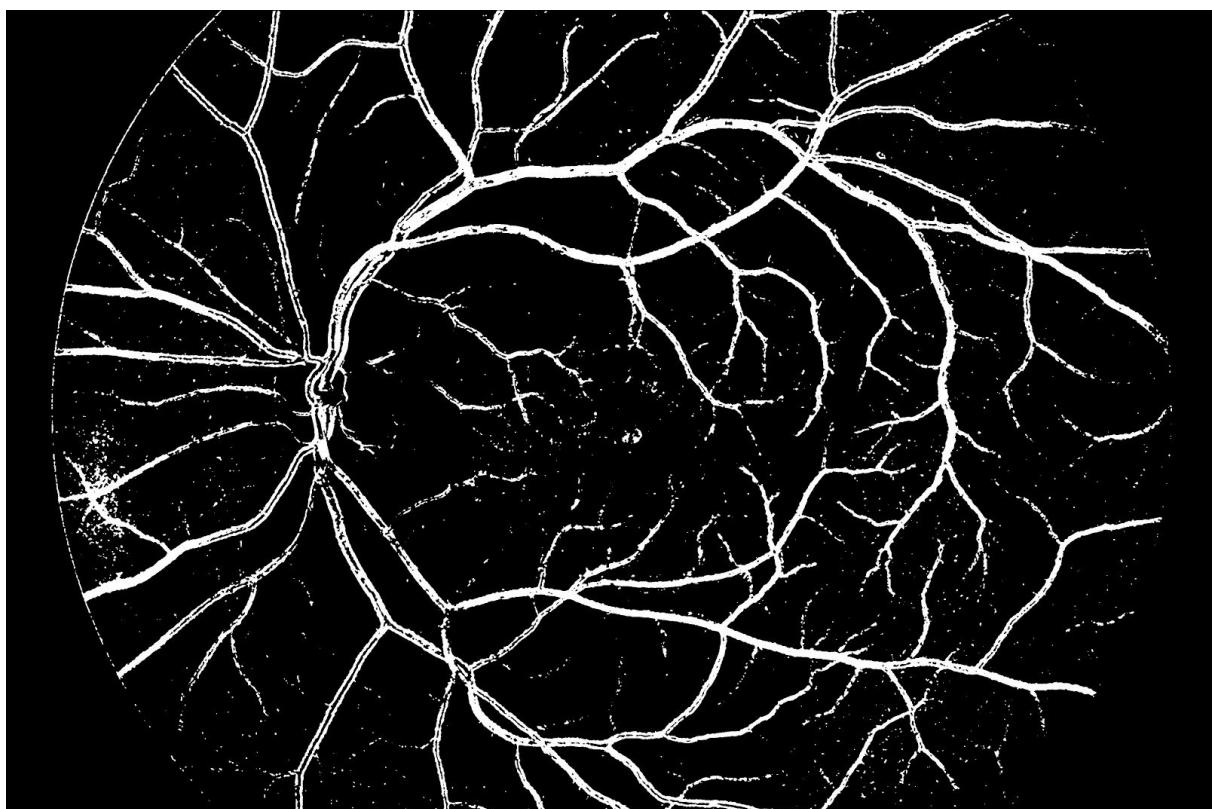
Rysunek 29: Bazowe przetwarzanie



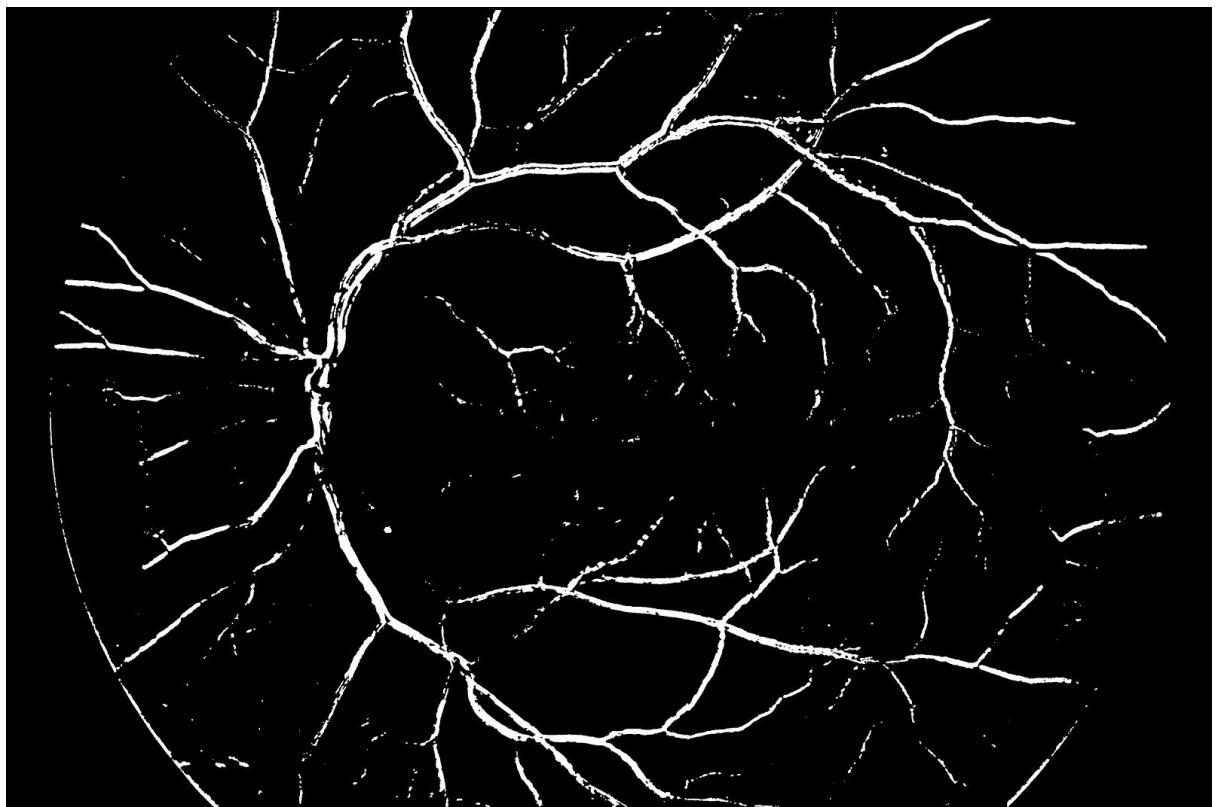
Rysunek 30: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



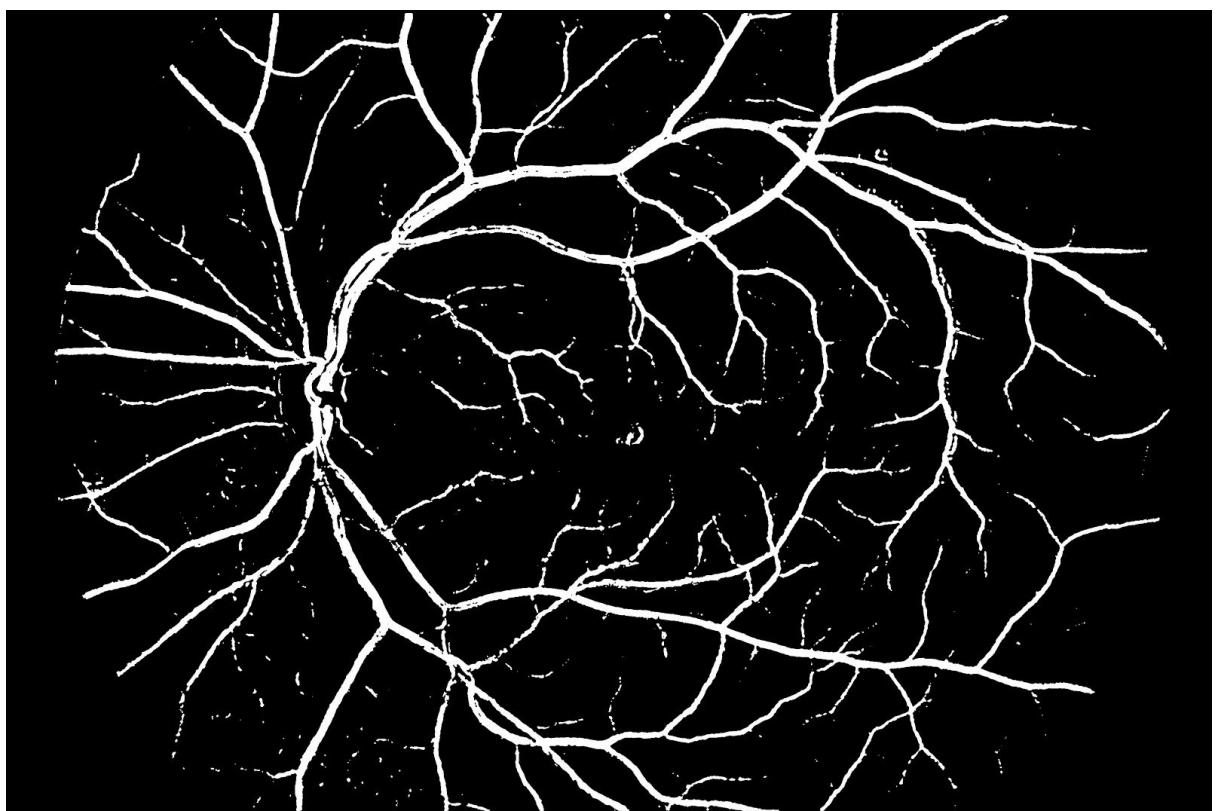
Rysunek 31: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



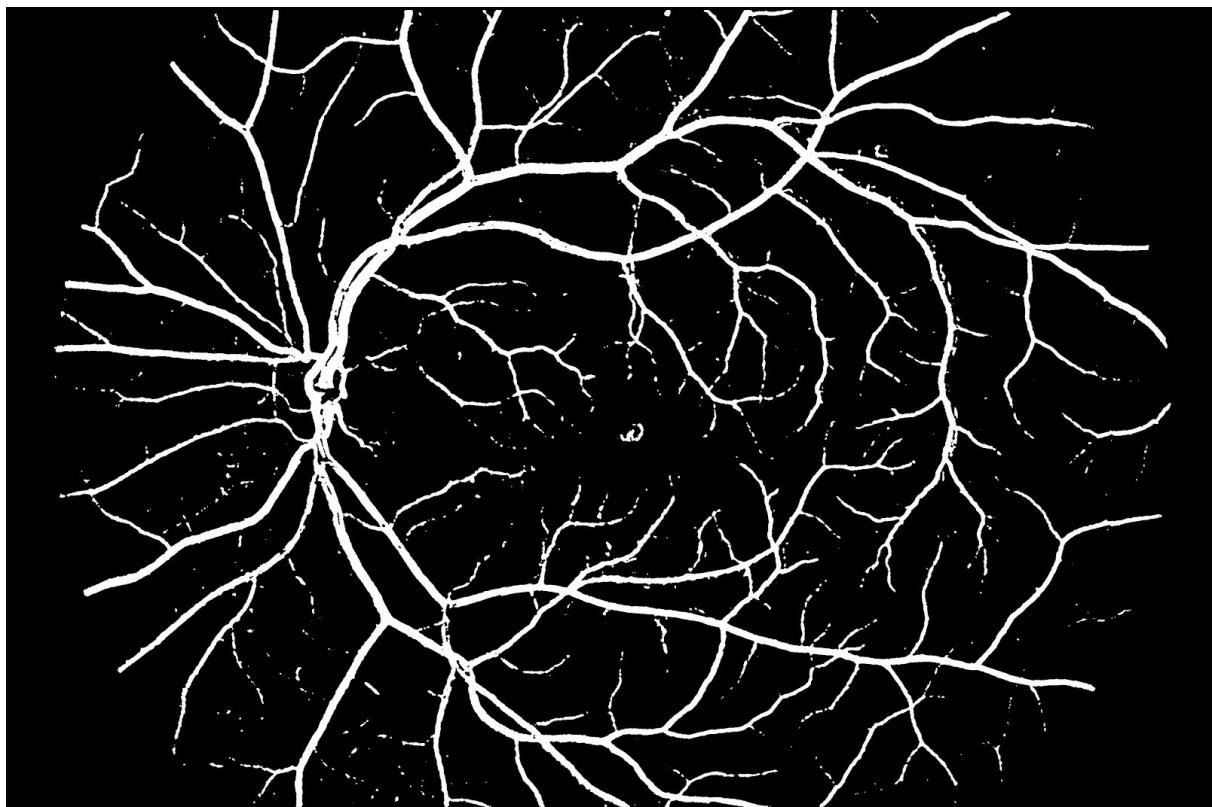
Rysunek 32: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 33: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

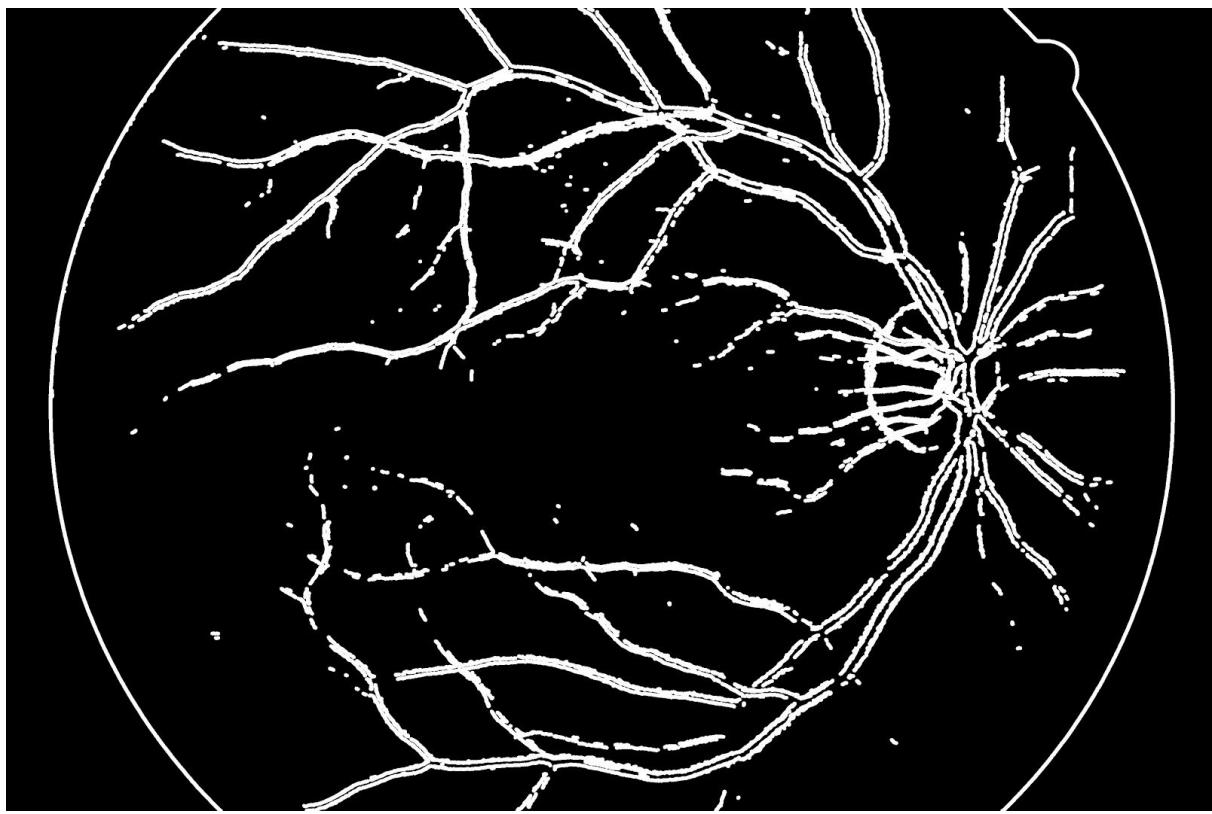


Rysunek 34: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

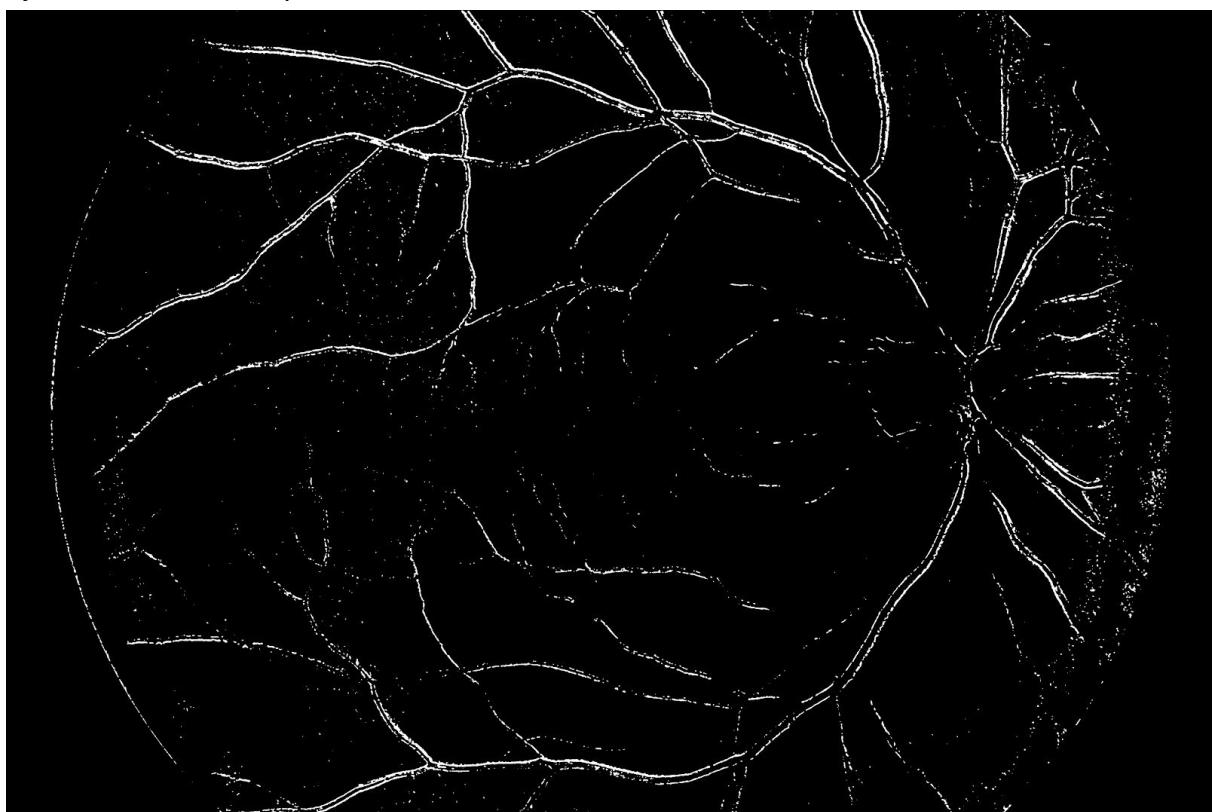


Rysunek 35: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

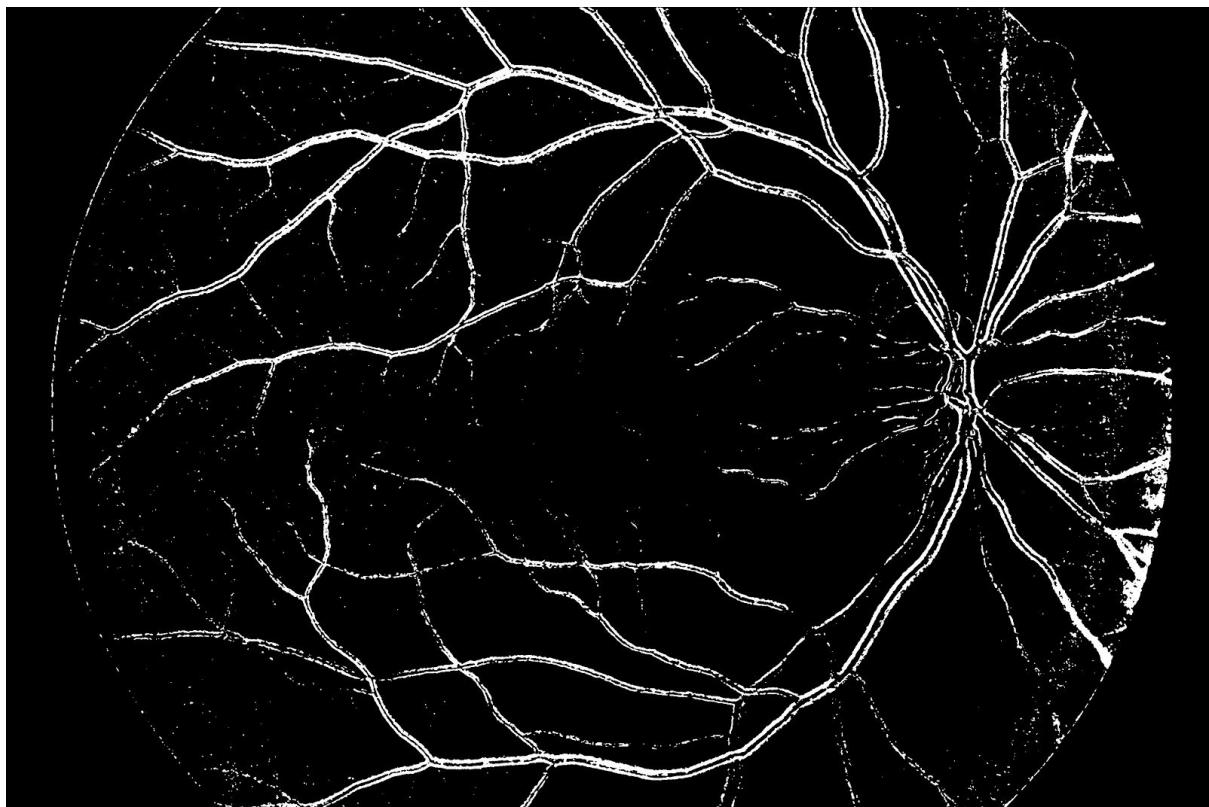
Obraz 05



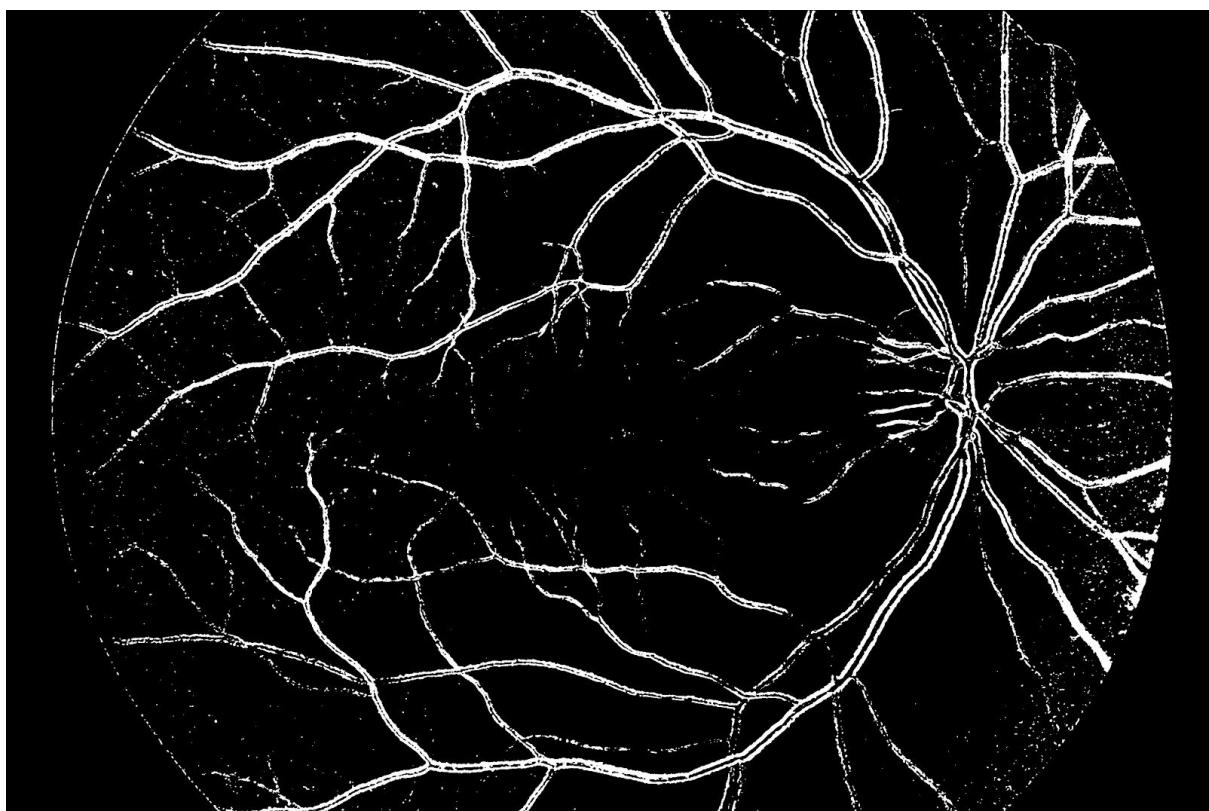
Rysunek 36: Bazowe przetwarzanie



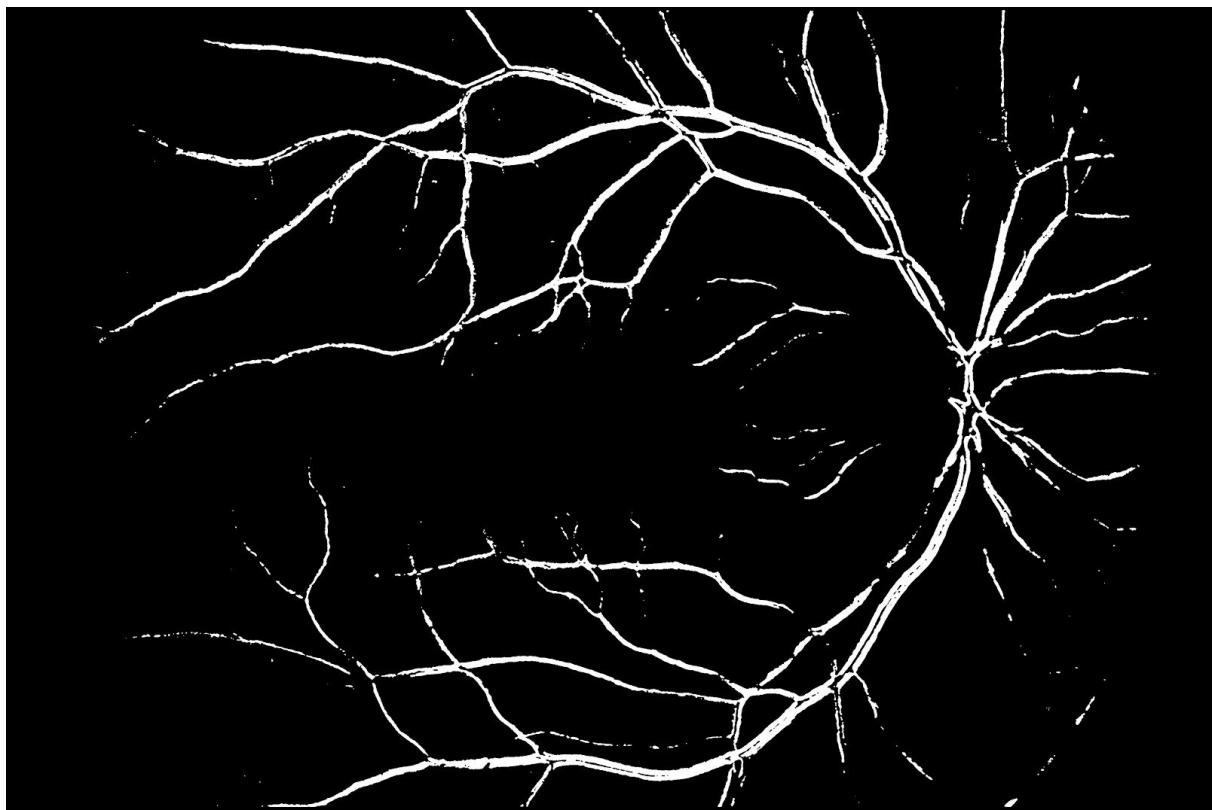
Rysunek 37: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



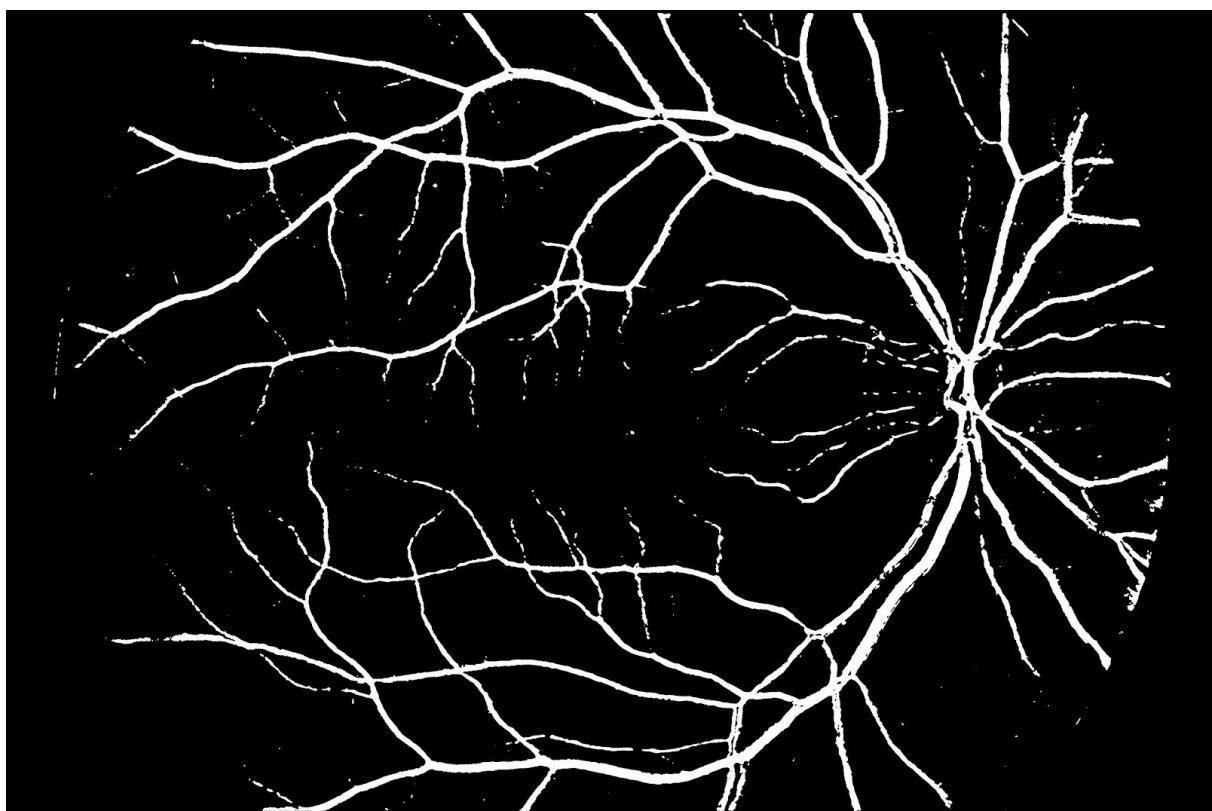
Rysunek 38: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



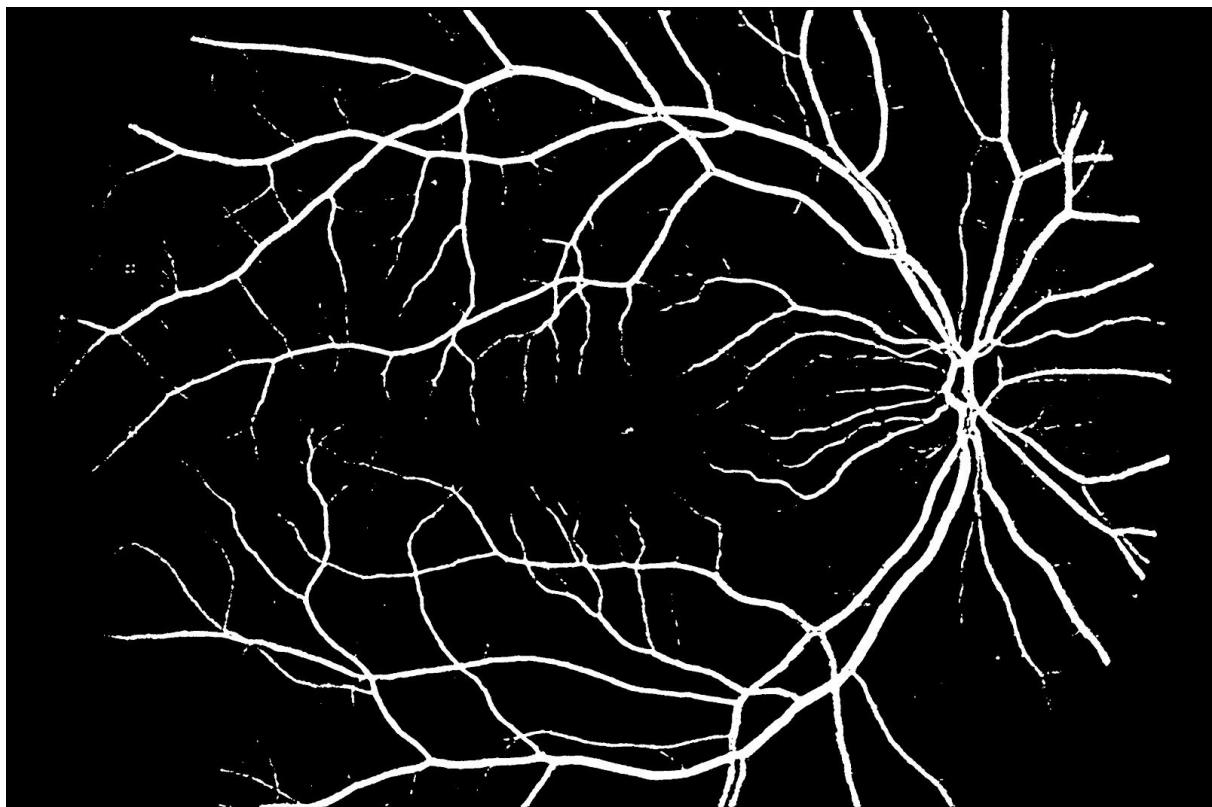
Rysunek 39: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 40: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

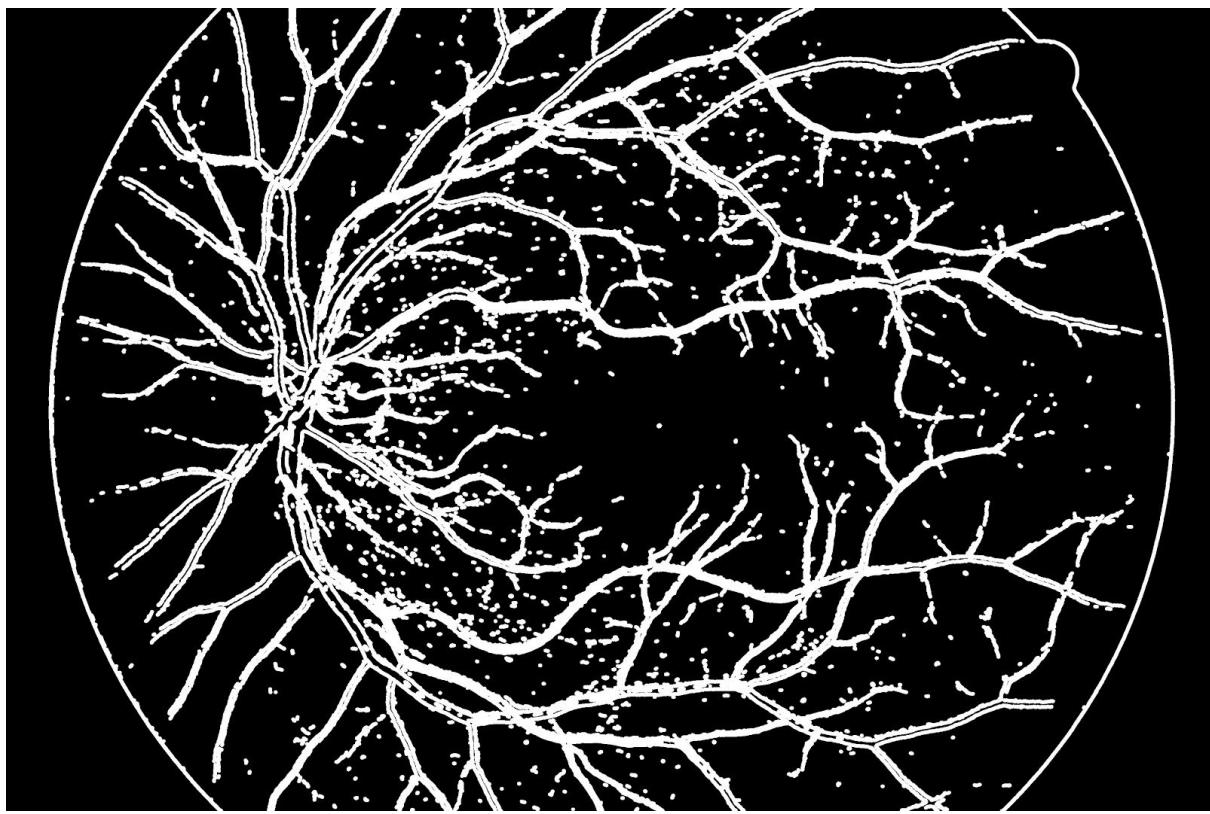


Rysunek 41: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

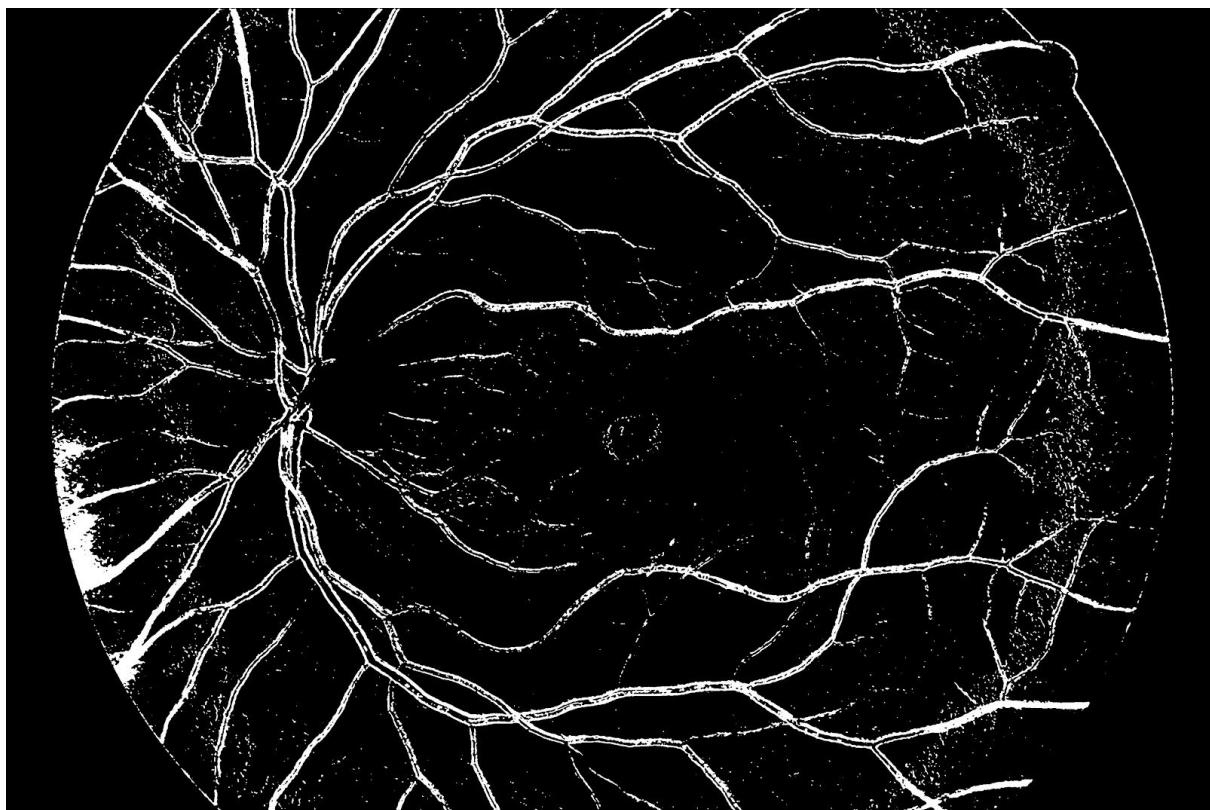


Rysunek 42: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

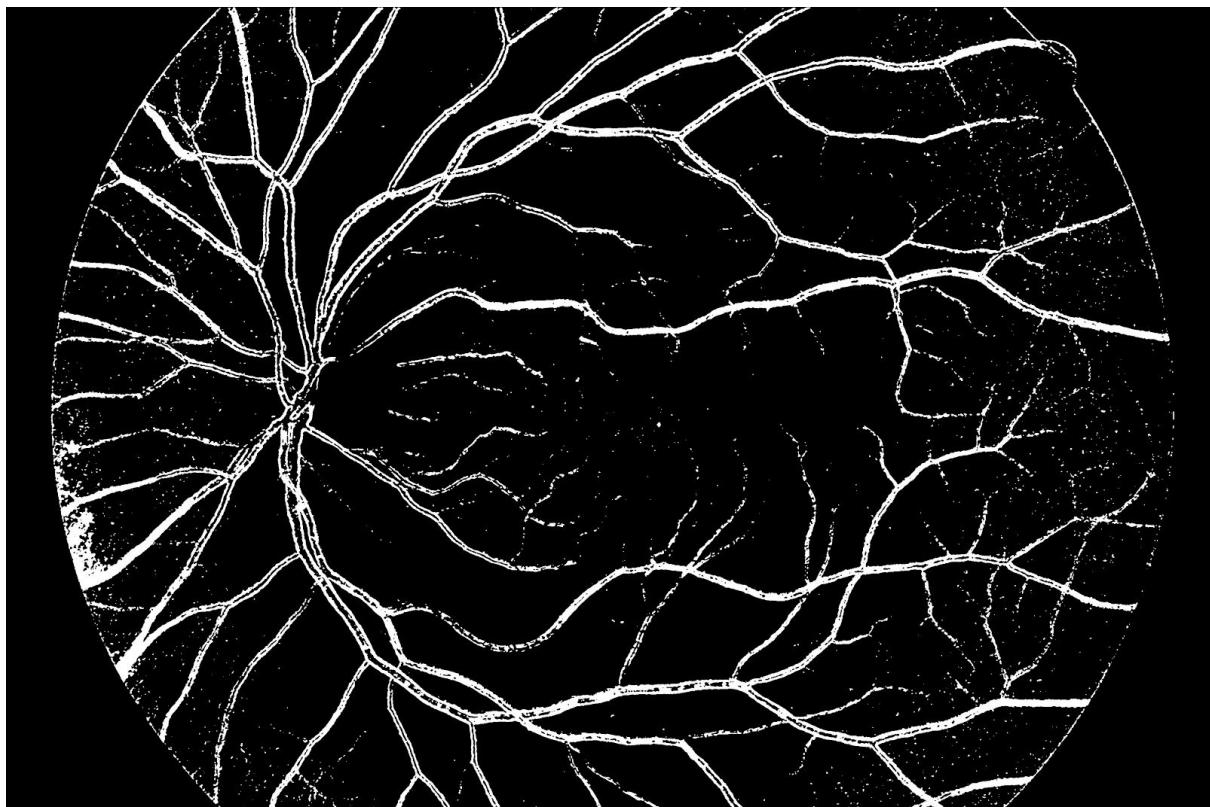
Obraz 06



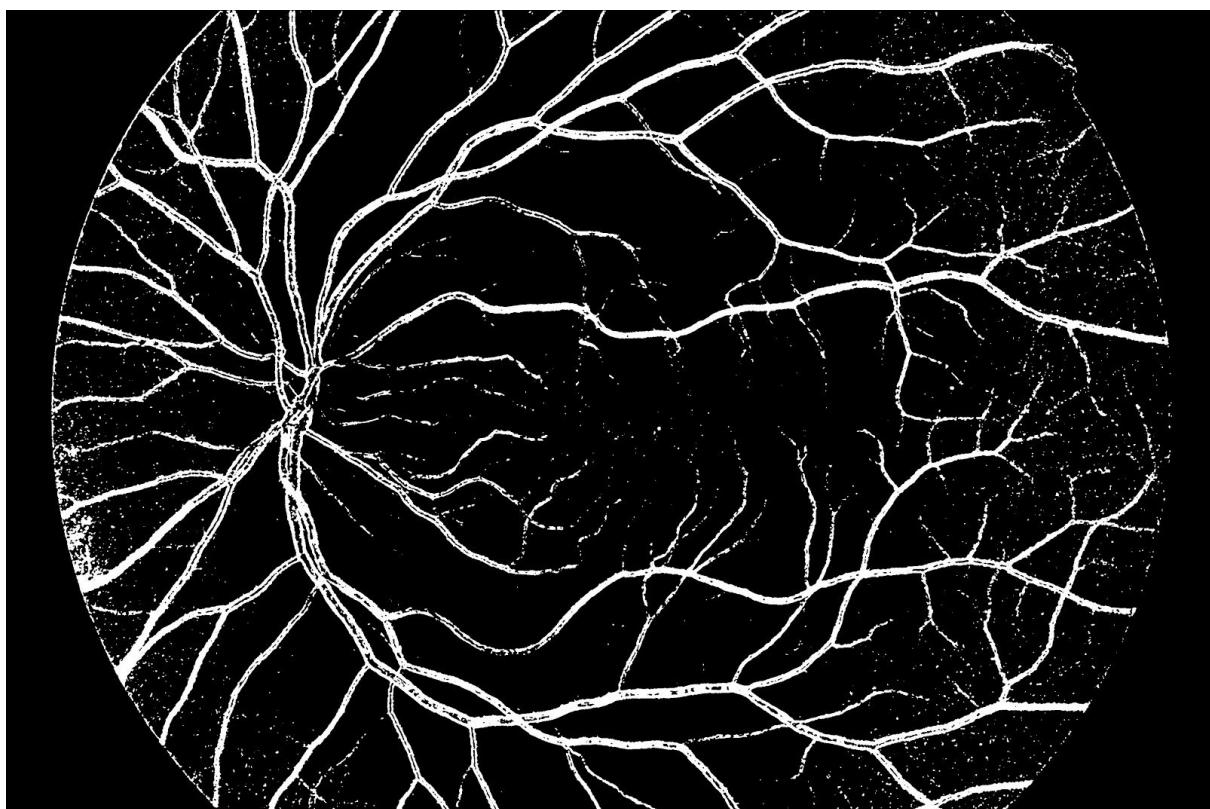
Rysunek 43: Bazowe przetwarzanie



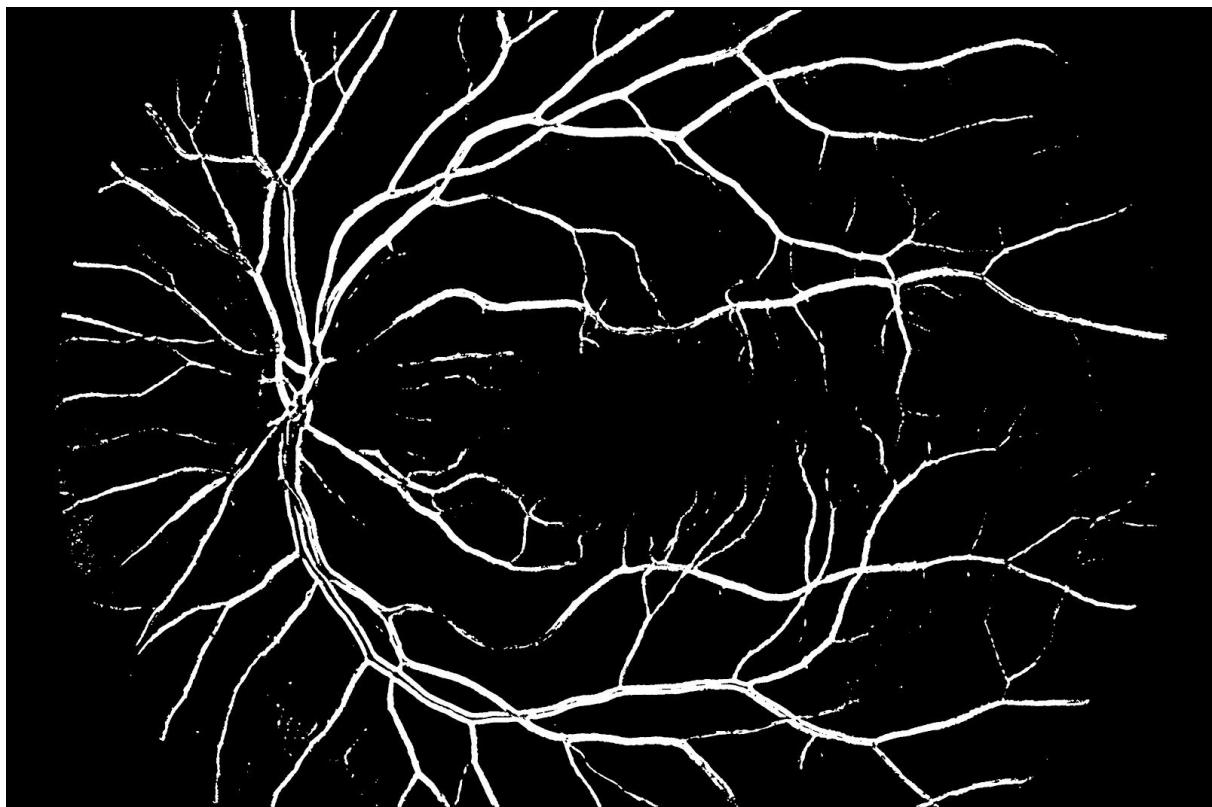
Rysunek 44: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



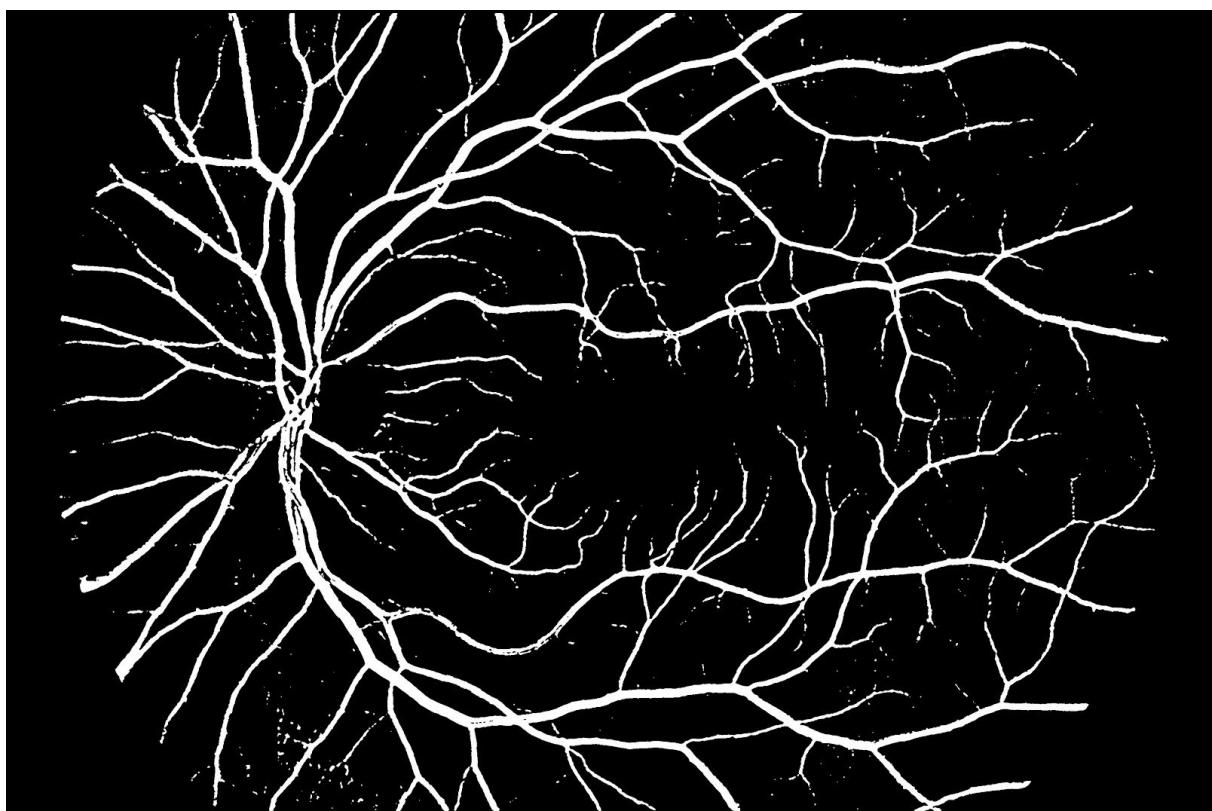
Rysunek 45: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



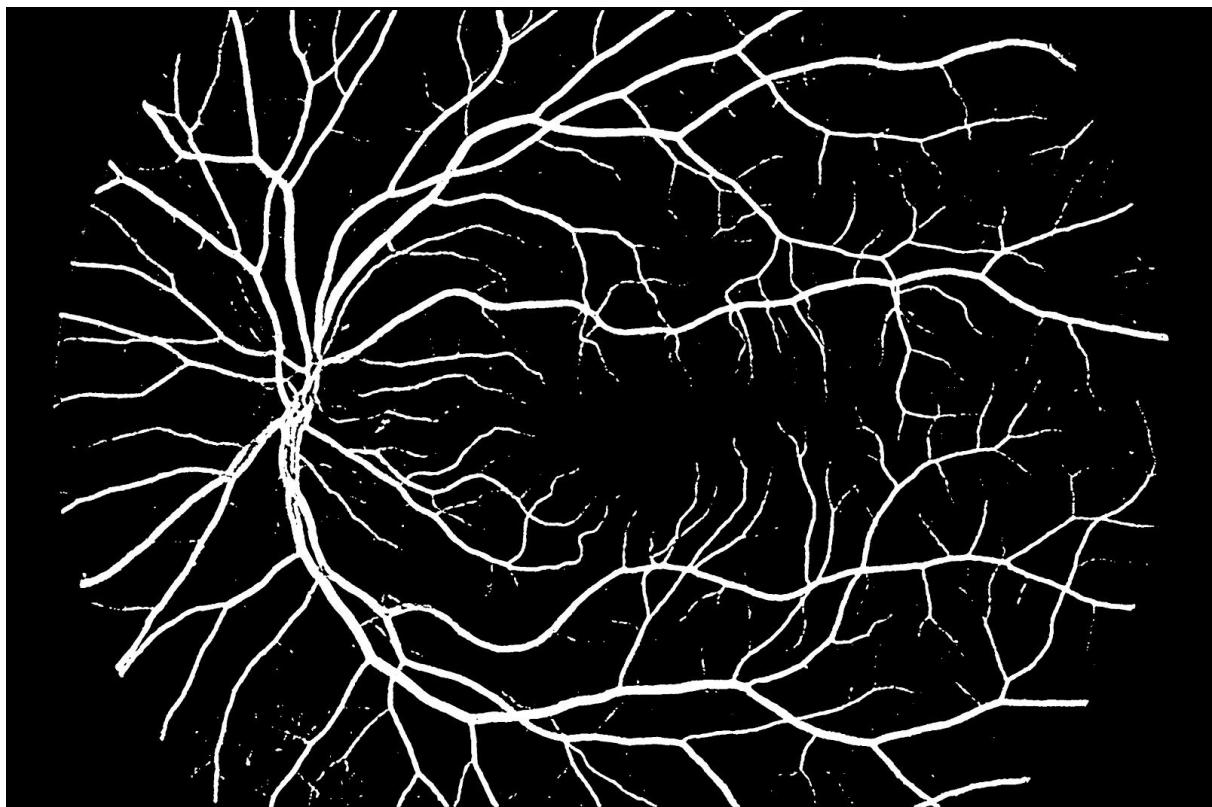
Rysunek 46: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 47: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

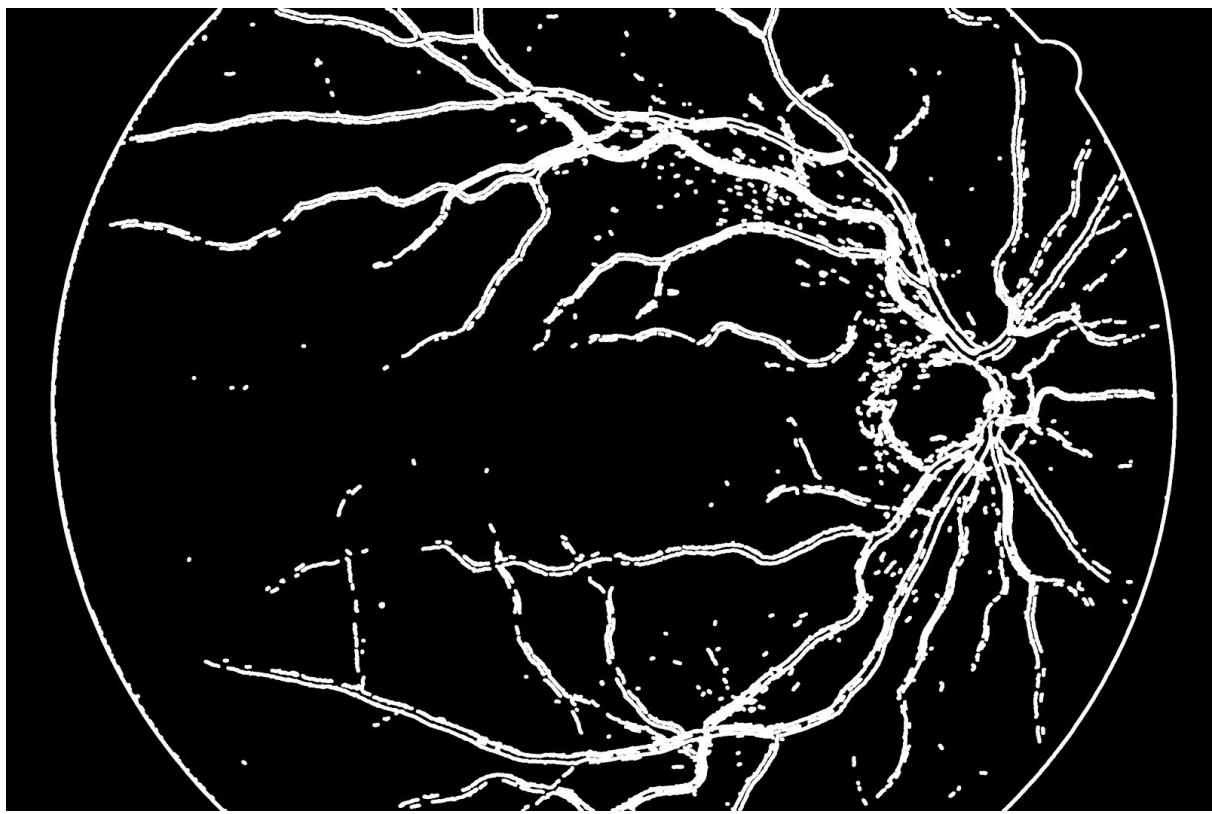


Rysunek 48: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

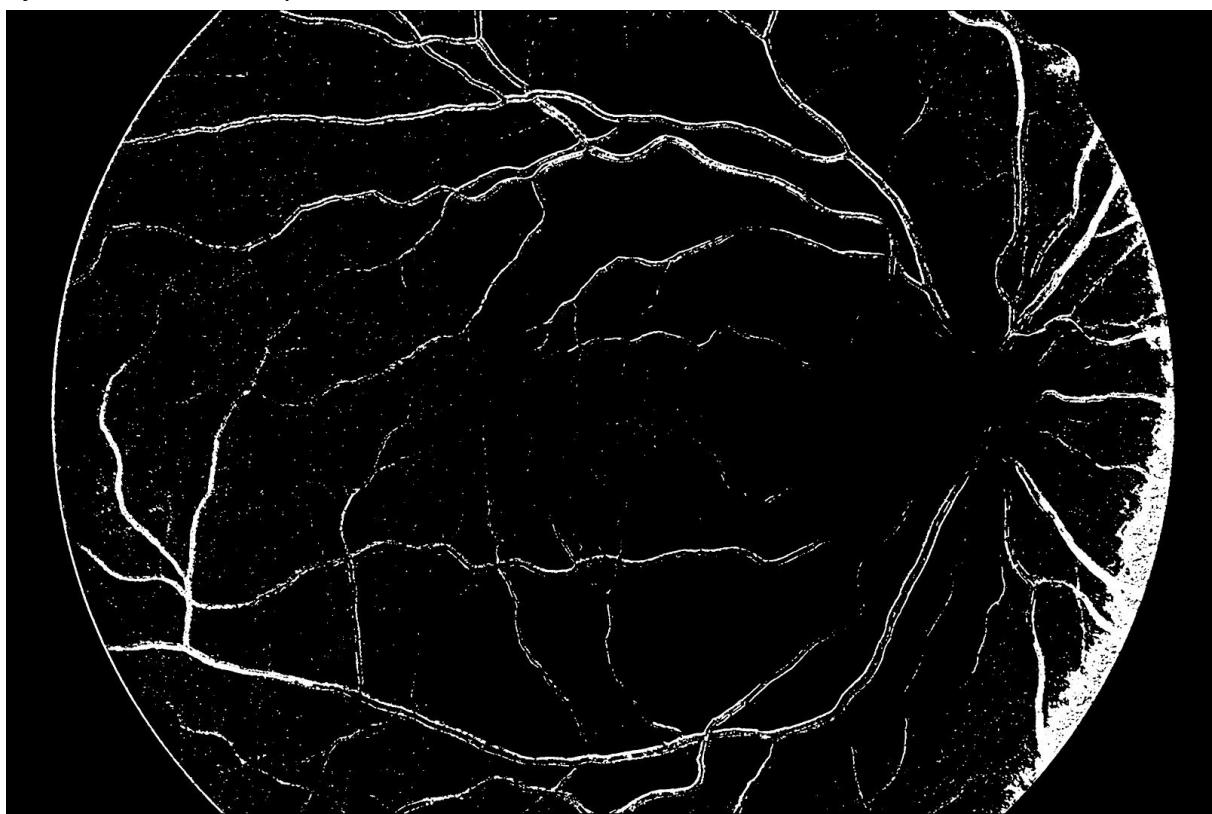


Rysunek 49: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

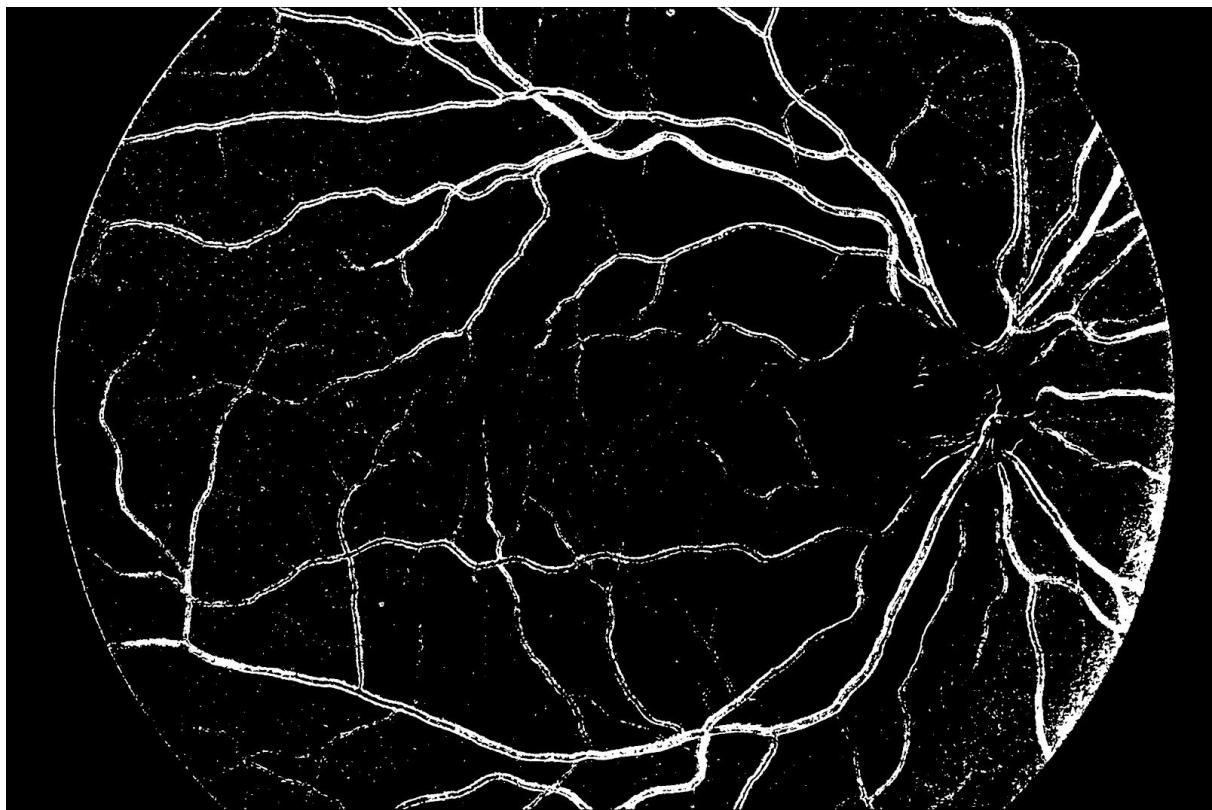
Obraz 07



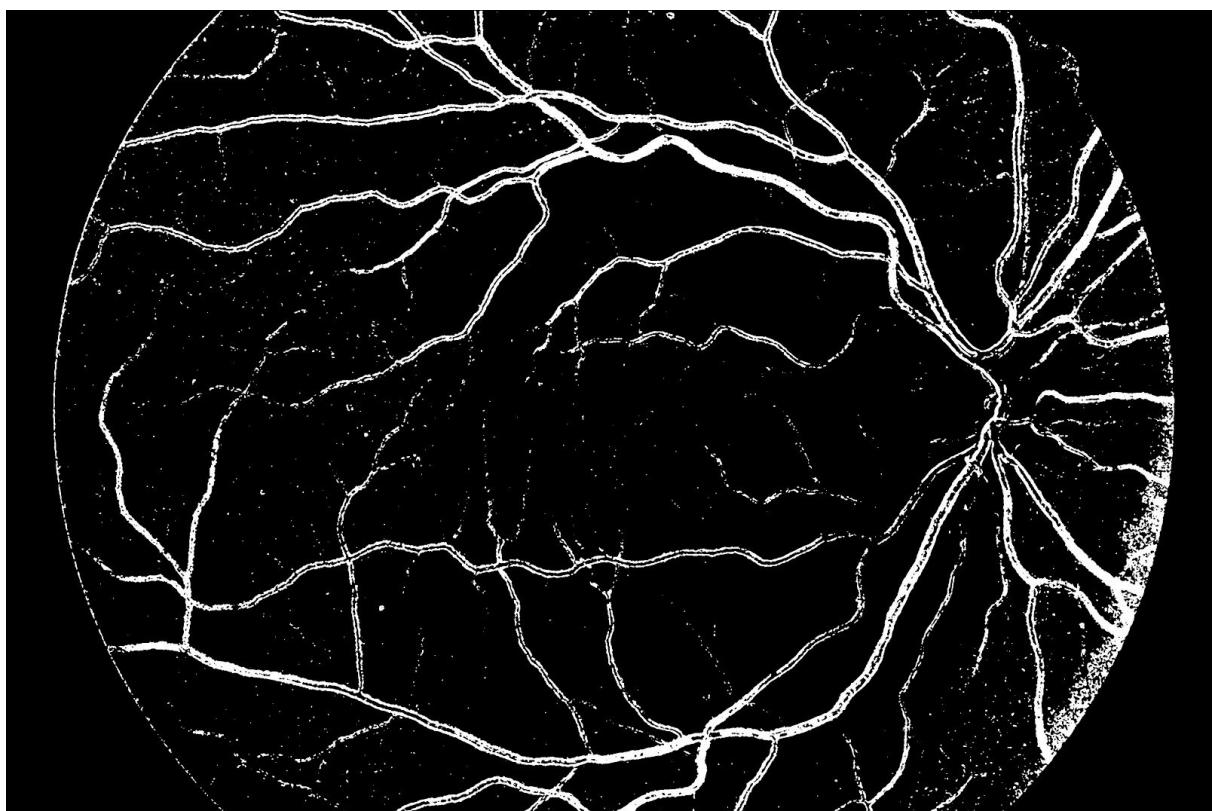
Rysunek 50: Bazowe przetwarzanie



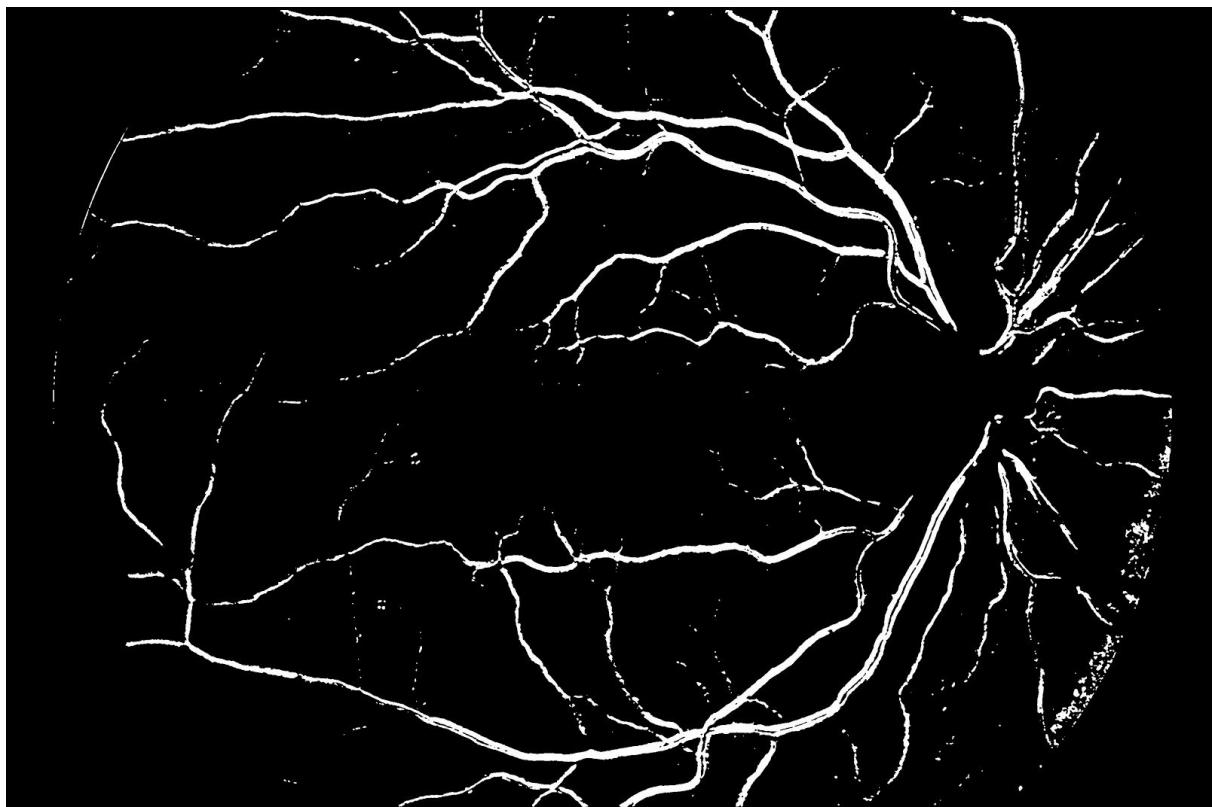
Rysunek 51: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



Rysunek 52: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



Rysunek 53: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 54: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

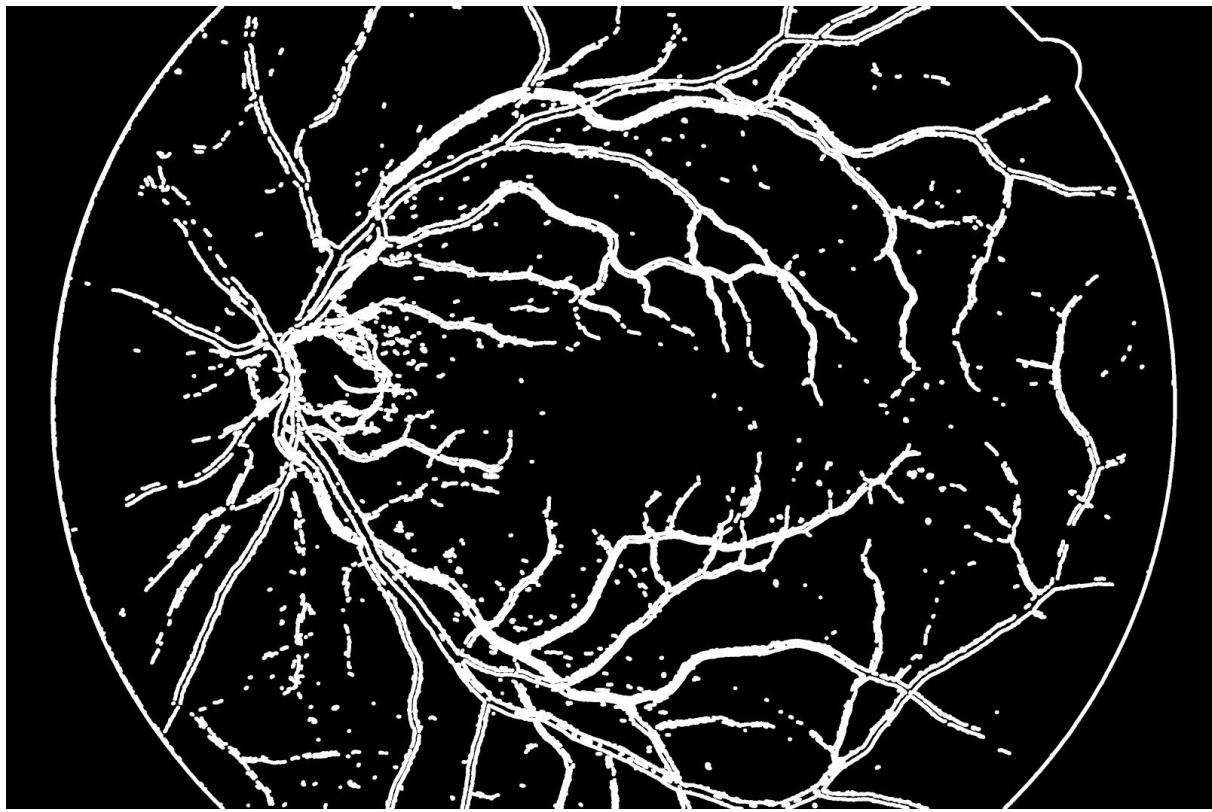


Rysunek 55: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

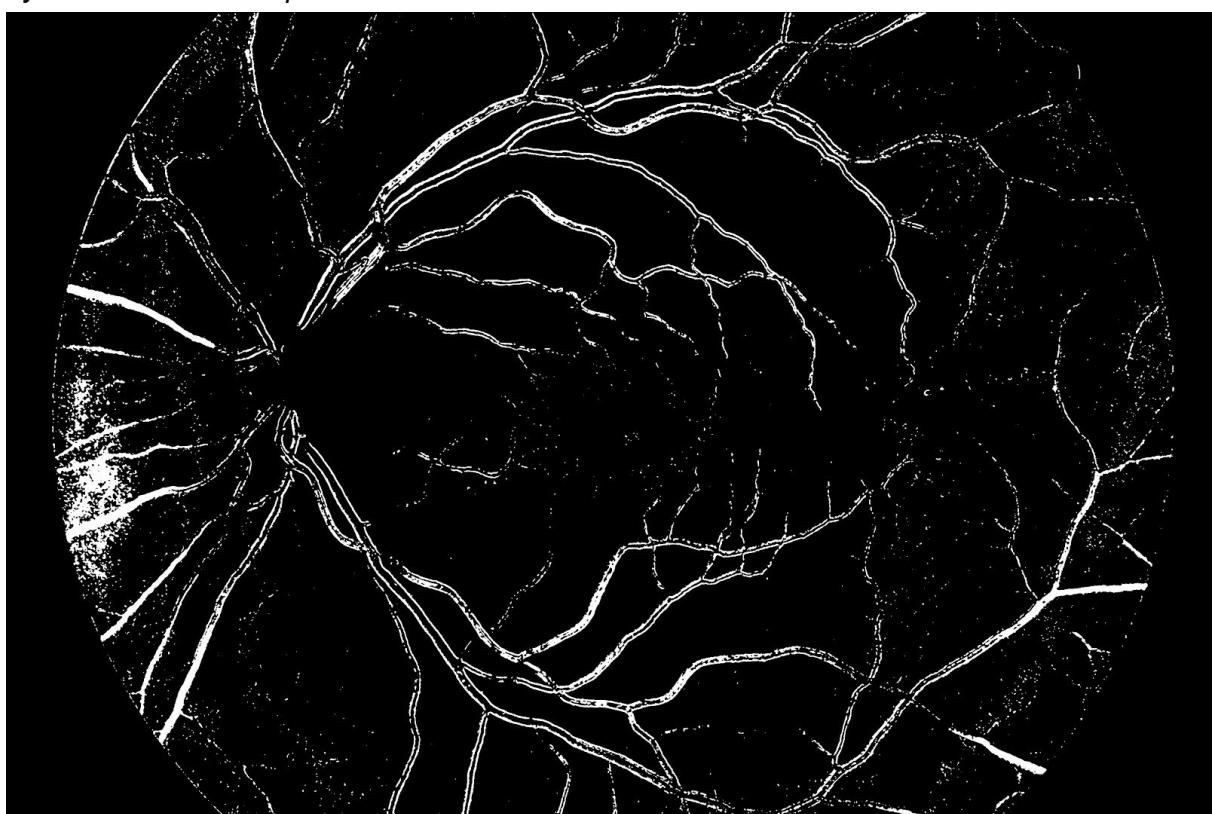


Rysunek 56: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

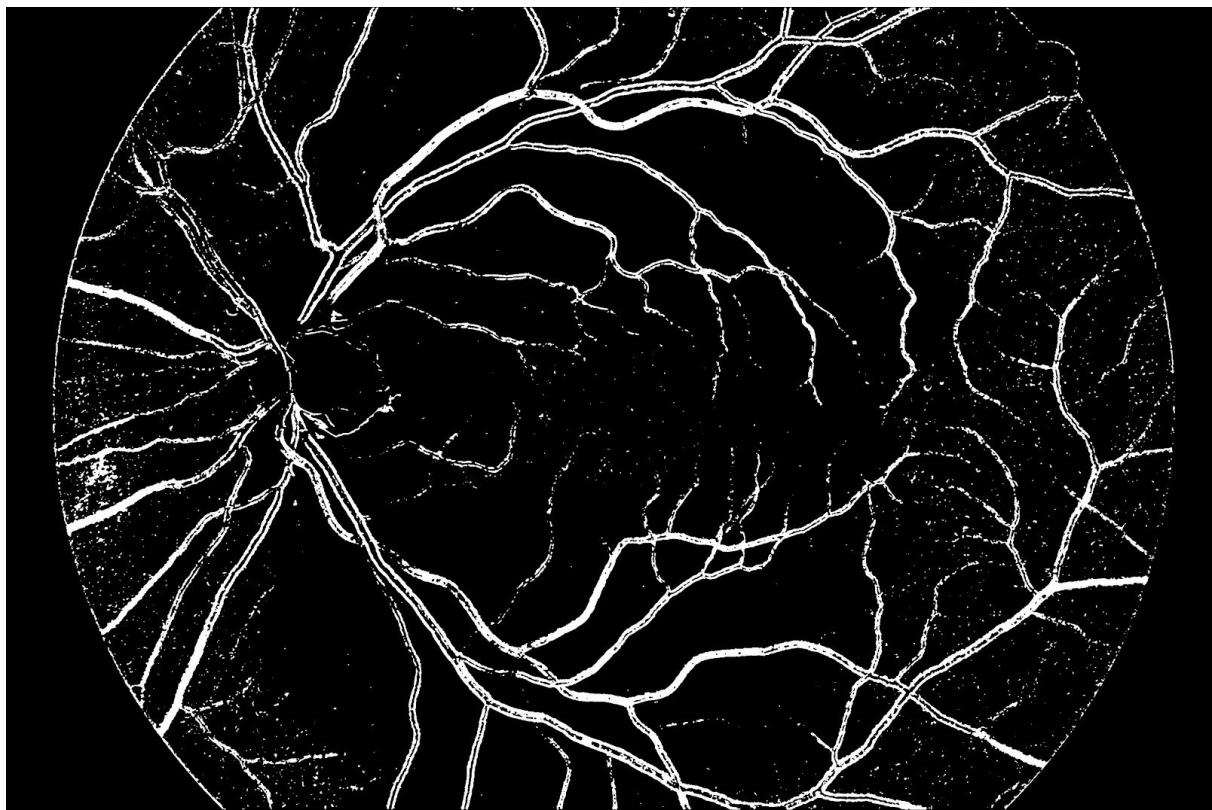
Obraz 08



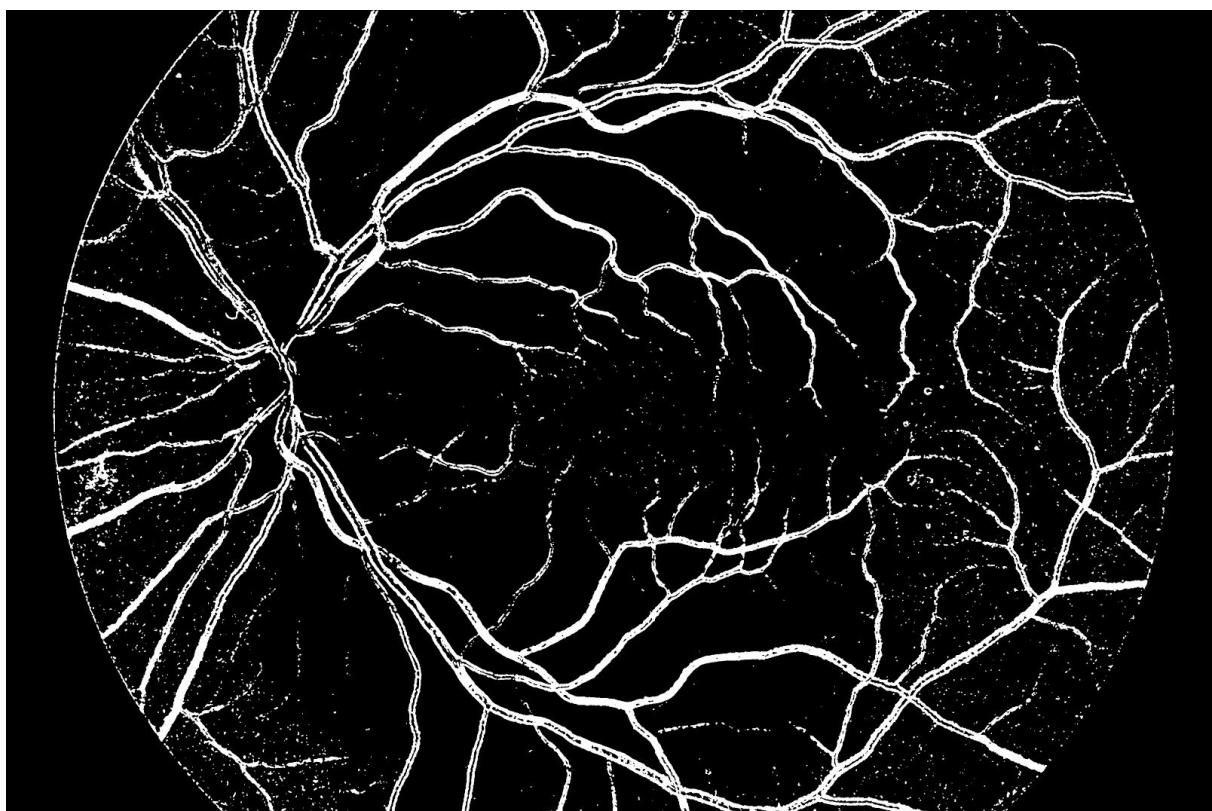
Rysunek 57: Bazowe przetwarzanie



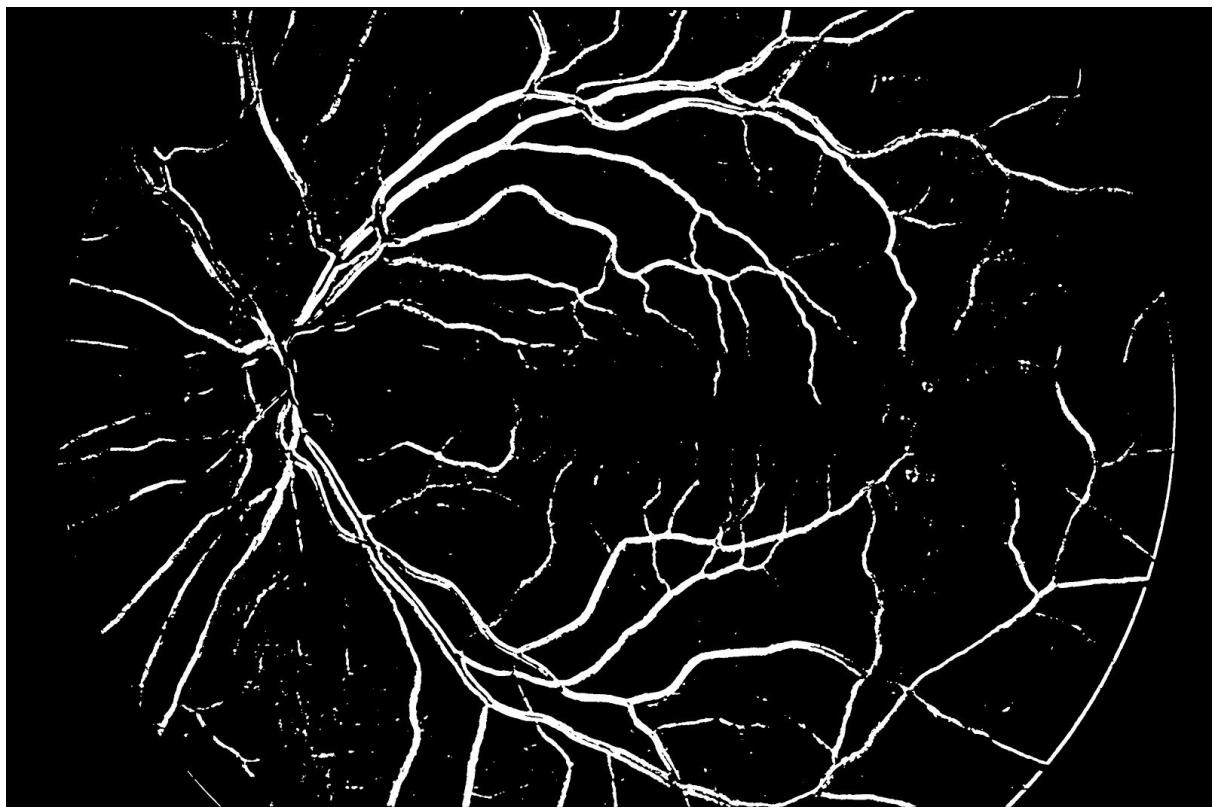
Rysunek 58: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



Rysunek 59: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



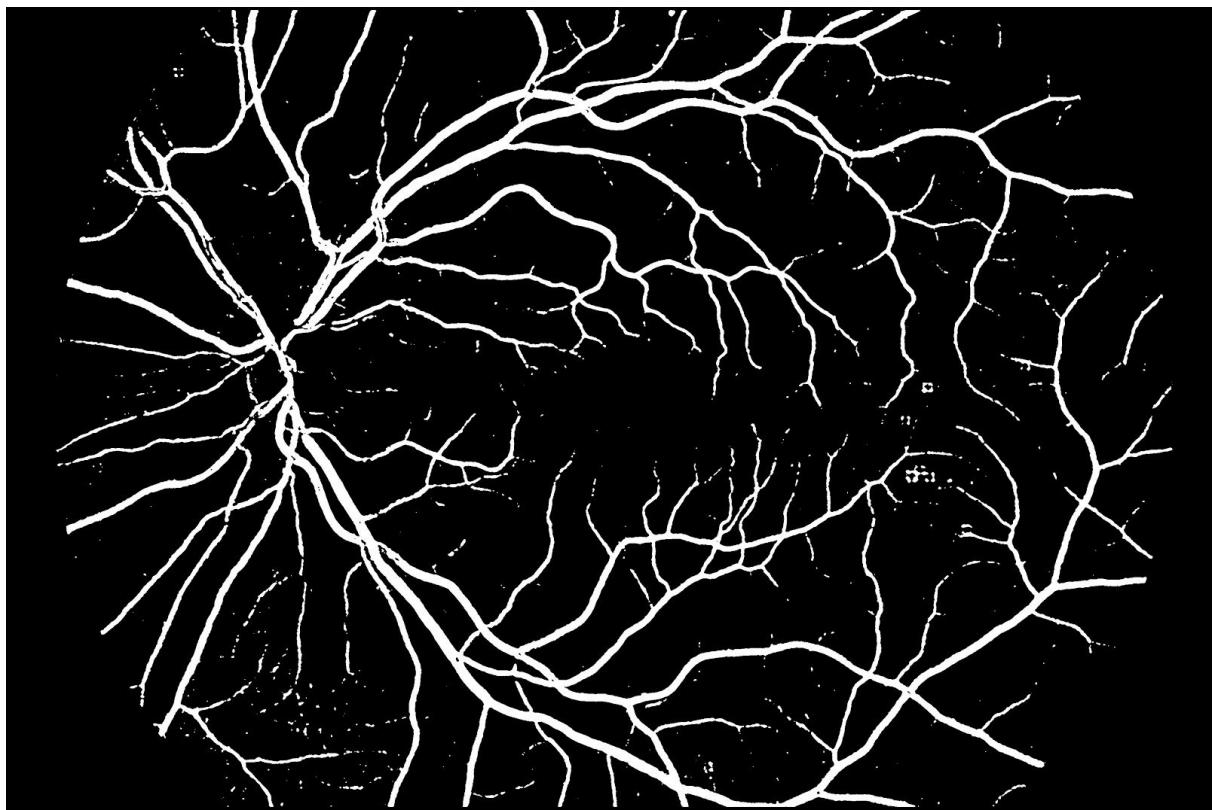
Rysunek 60: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 61: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

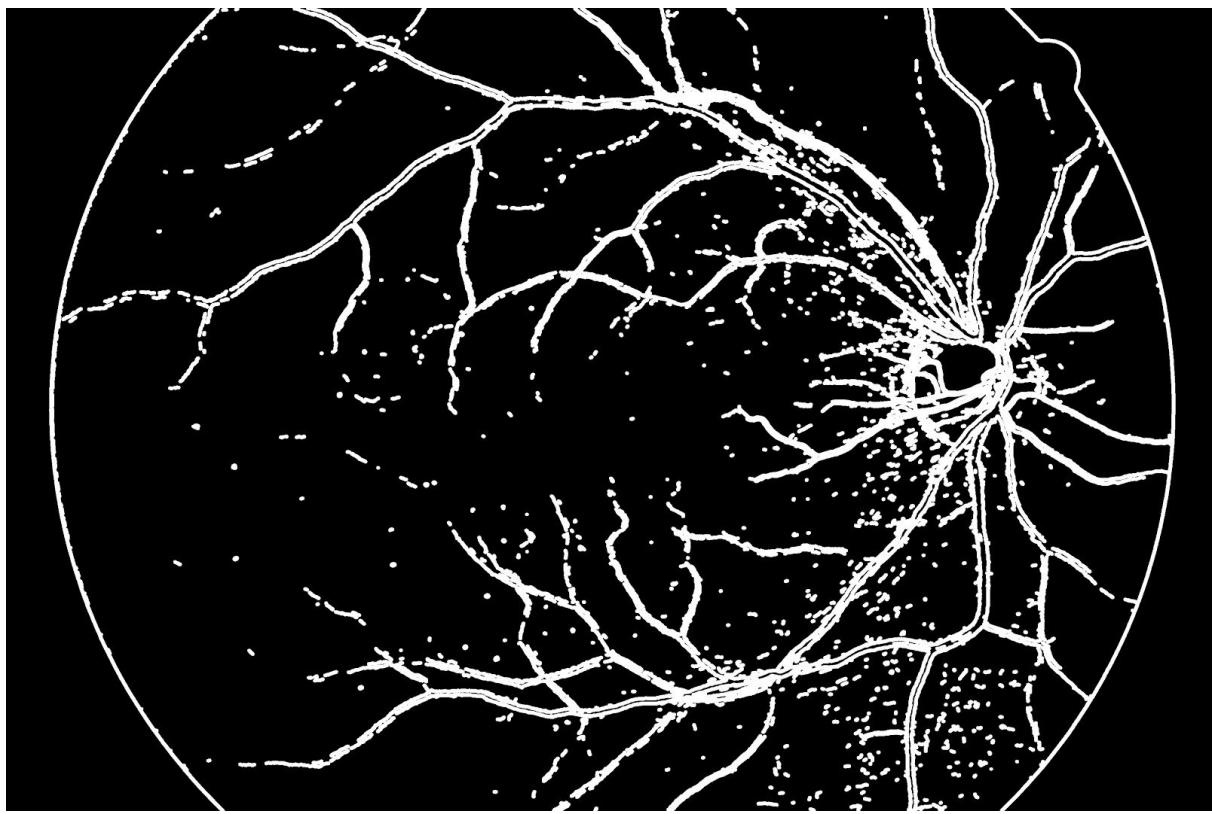


Rysunek 62: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze



Rysunek 63: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

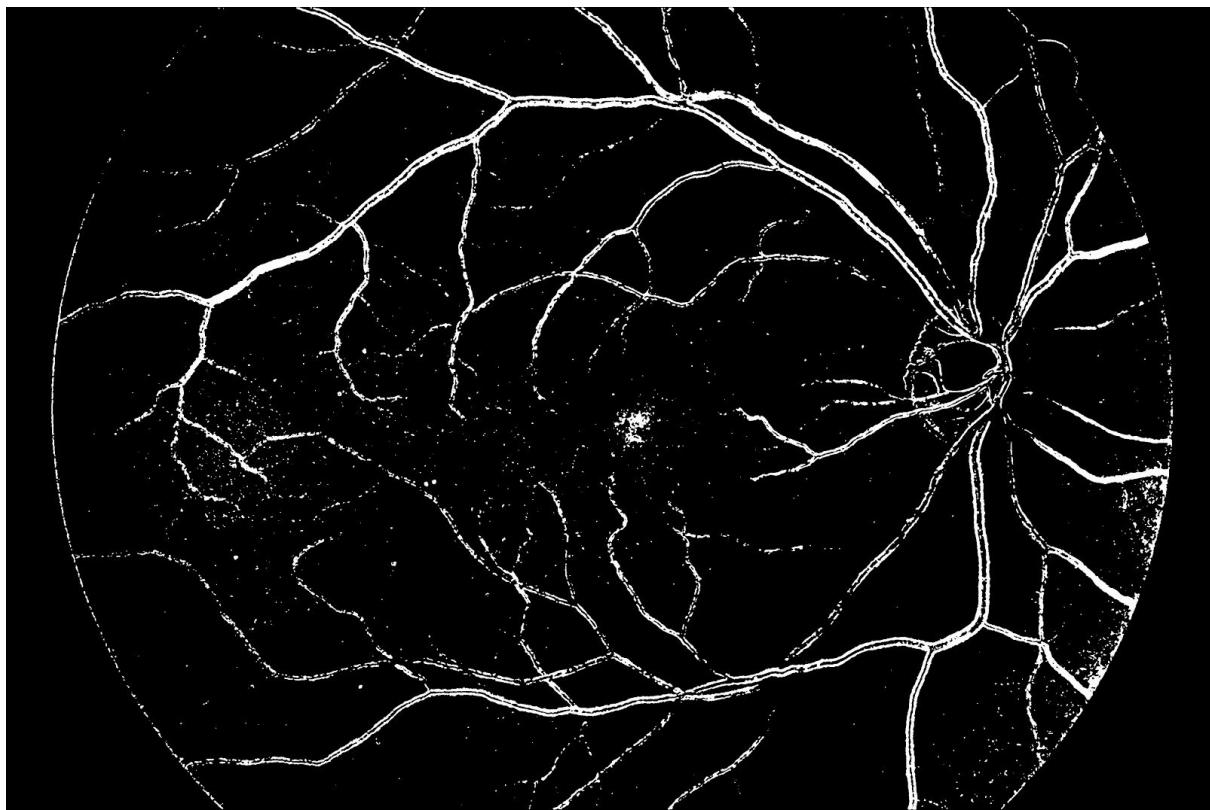
Obraz 09



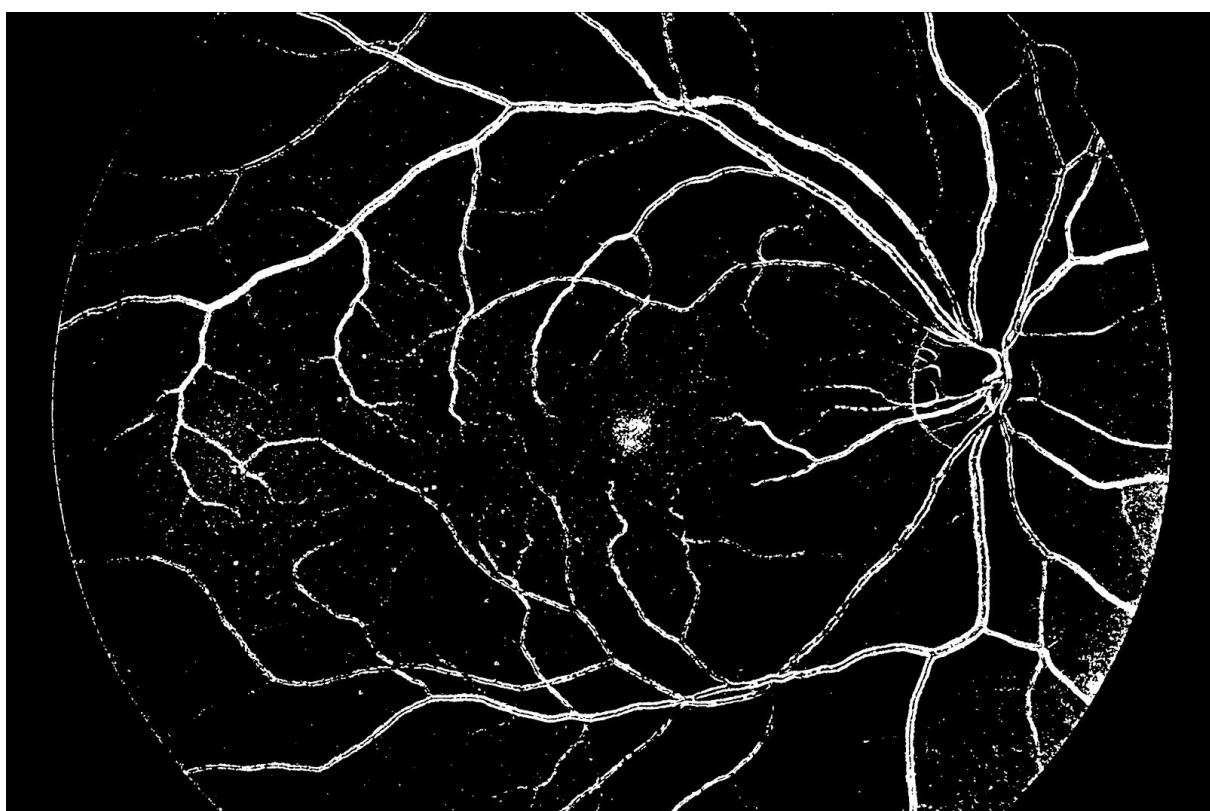
Rysunek 64: Bazowe przetwarzanie



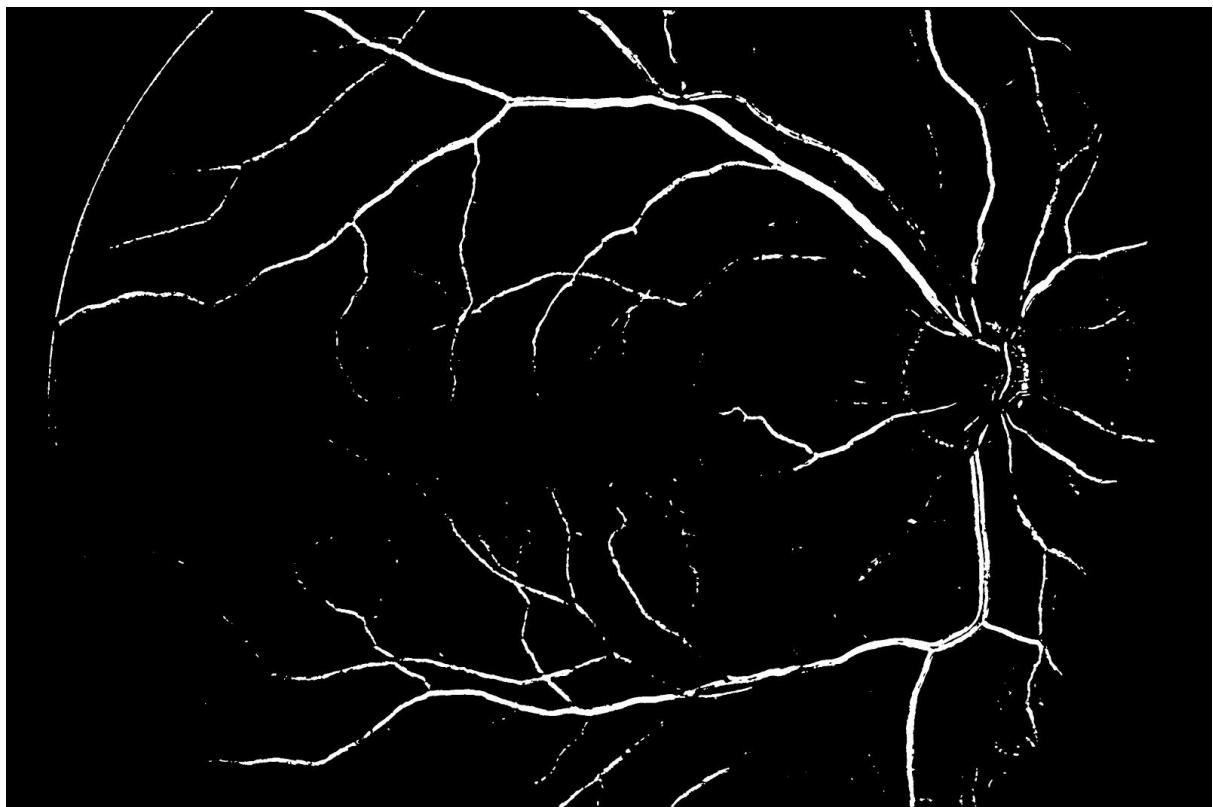
Rysunek 65: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



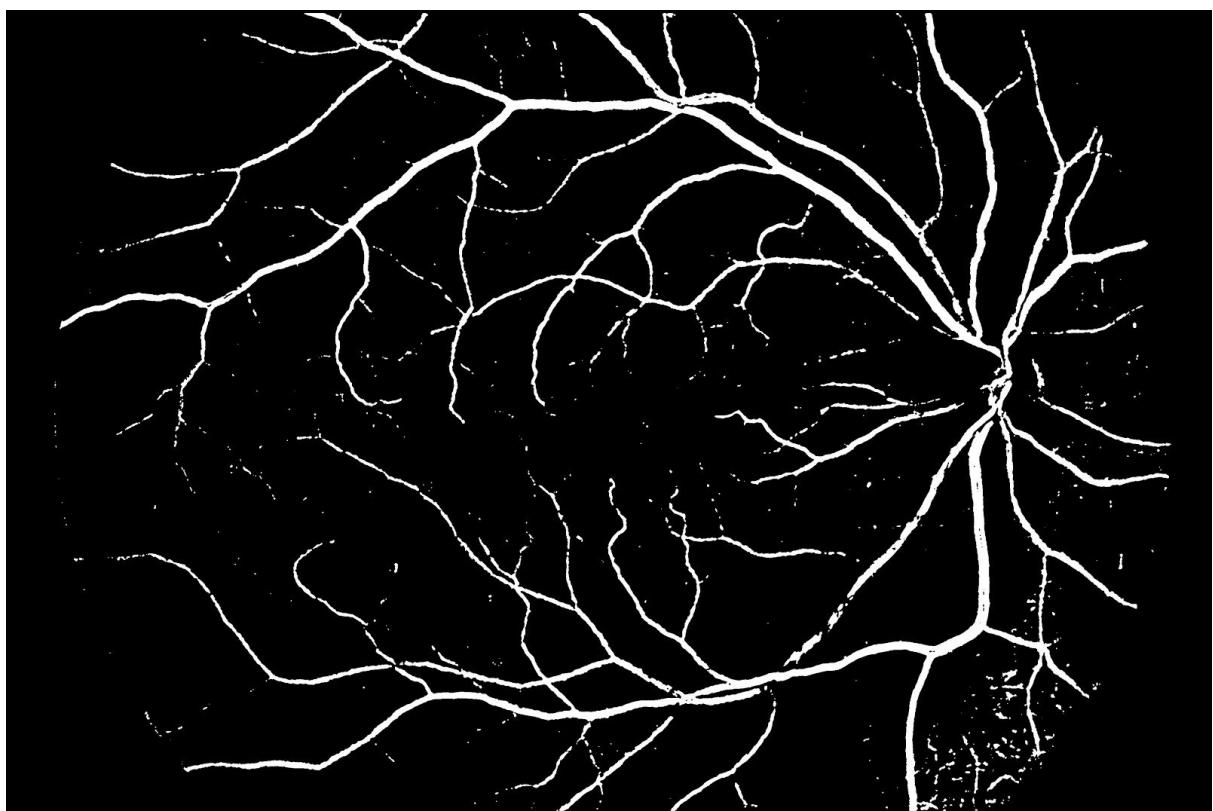
Rysunek 66: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



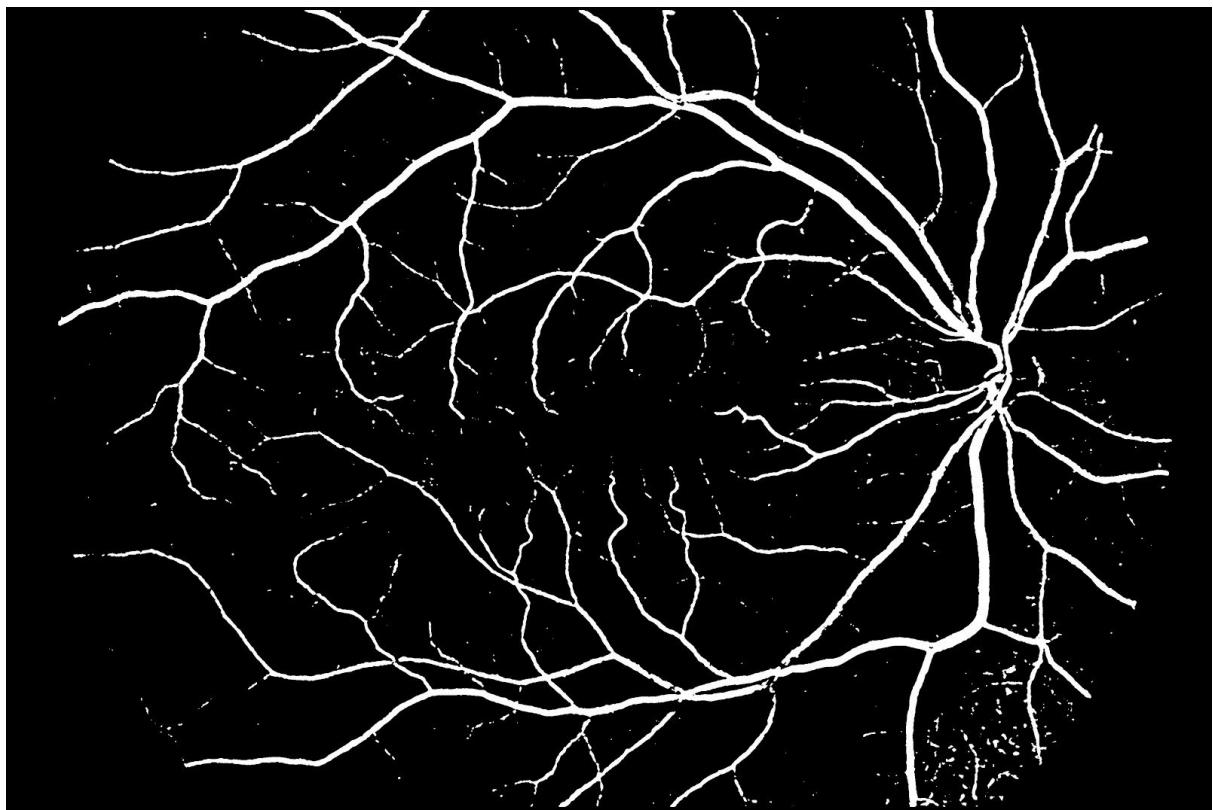
Rysunek 67: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 68: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

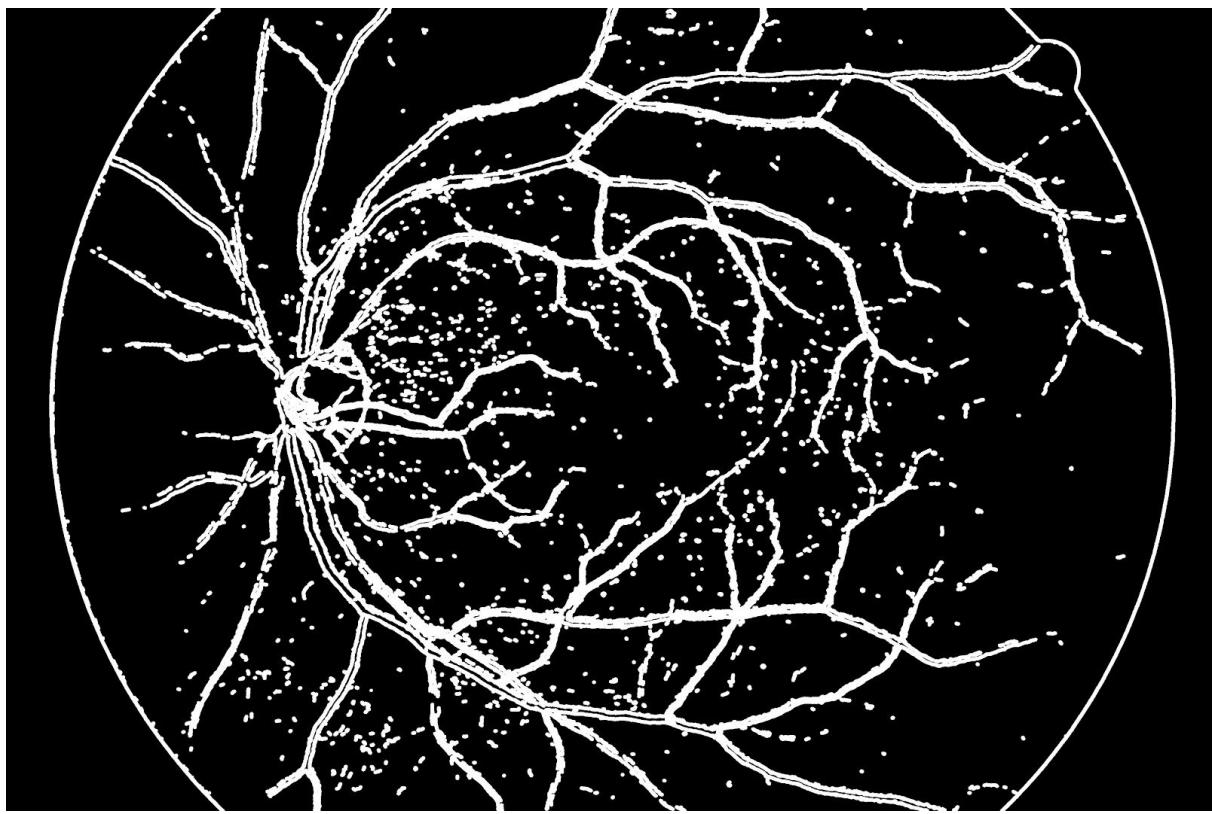


Rysunek 69: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

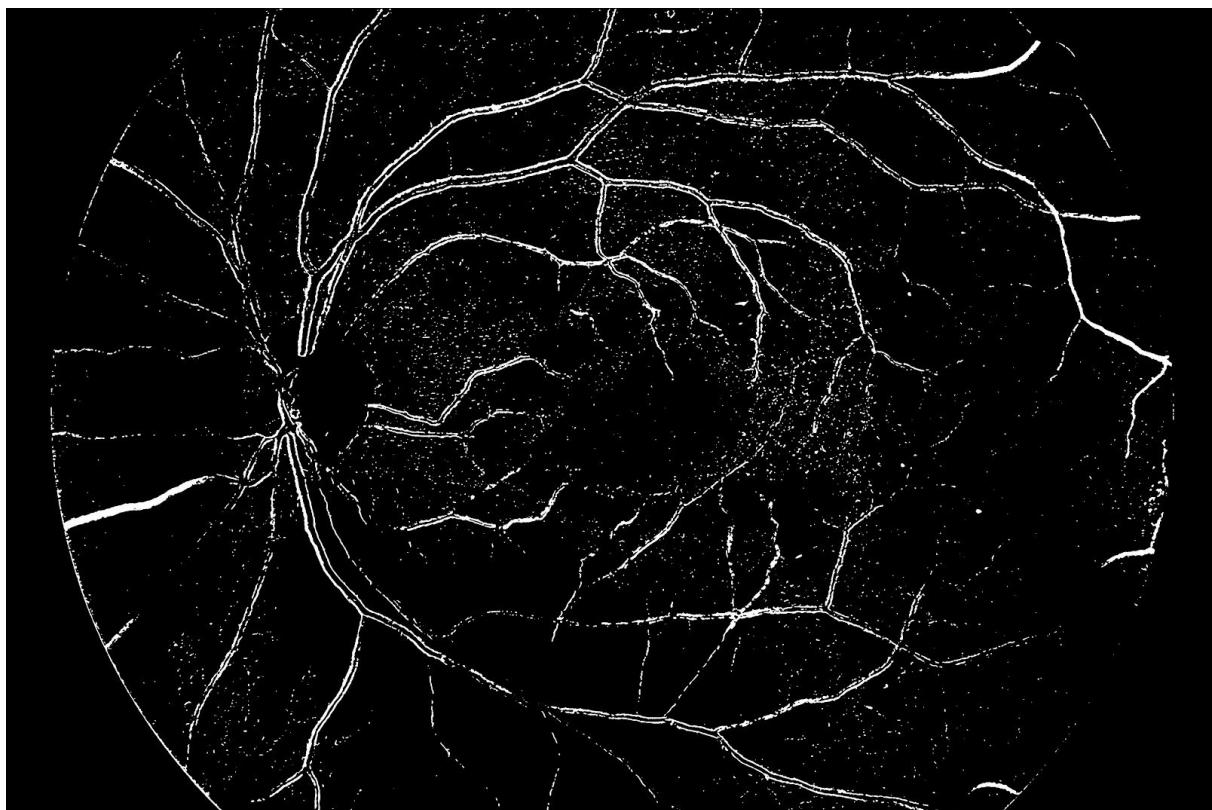


Rysunek 70: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

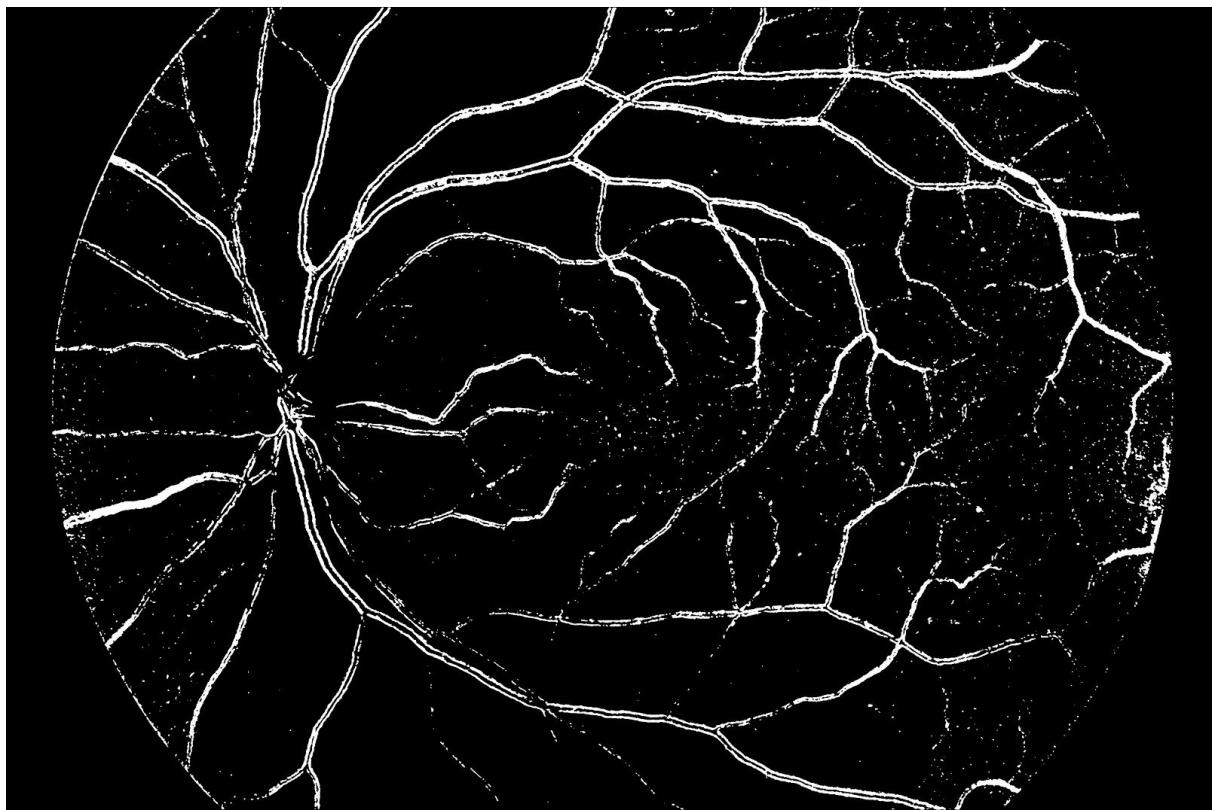
Obraz 10



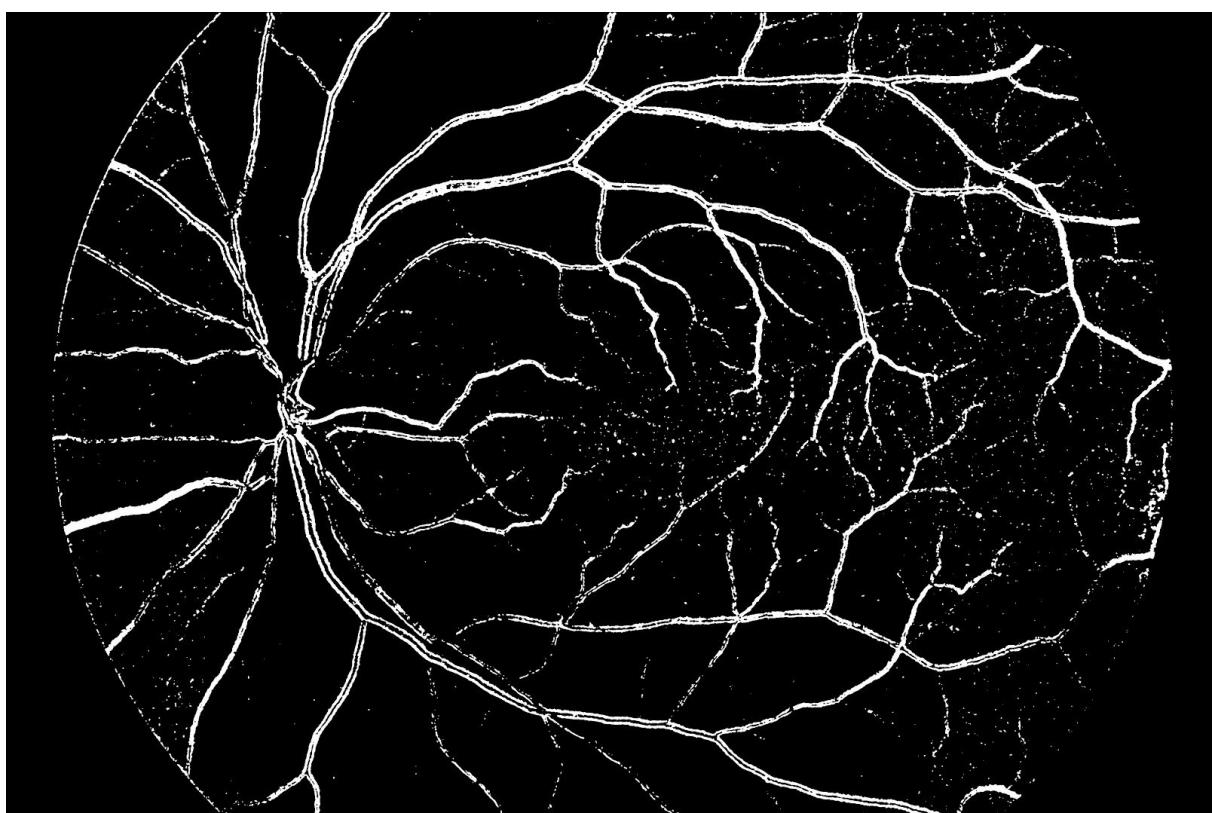
Rysunek 71: Bazowe przetwarzanie



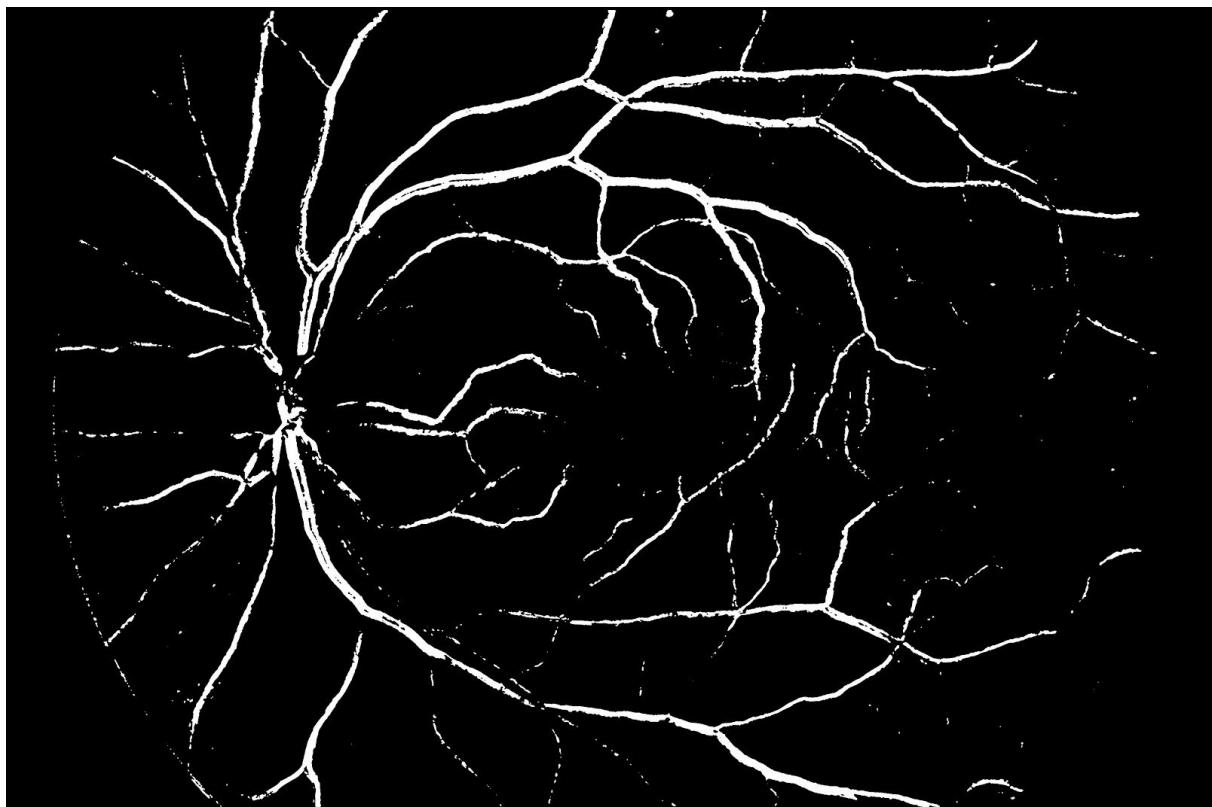
Rysunek 72: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



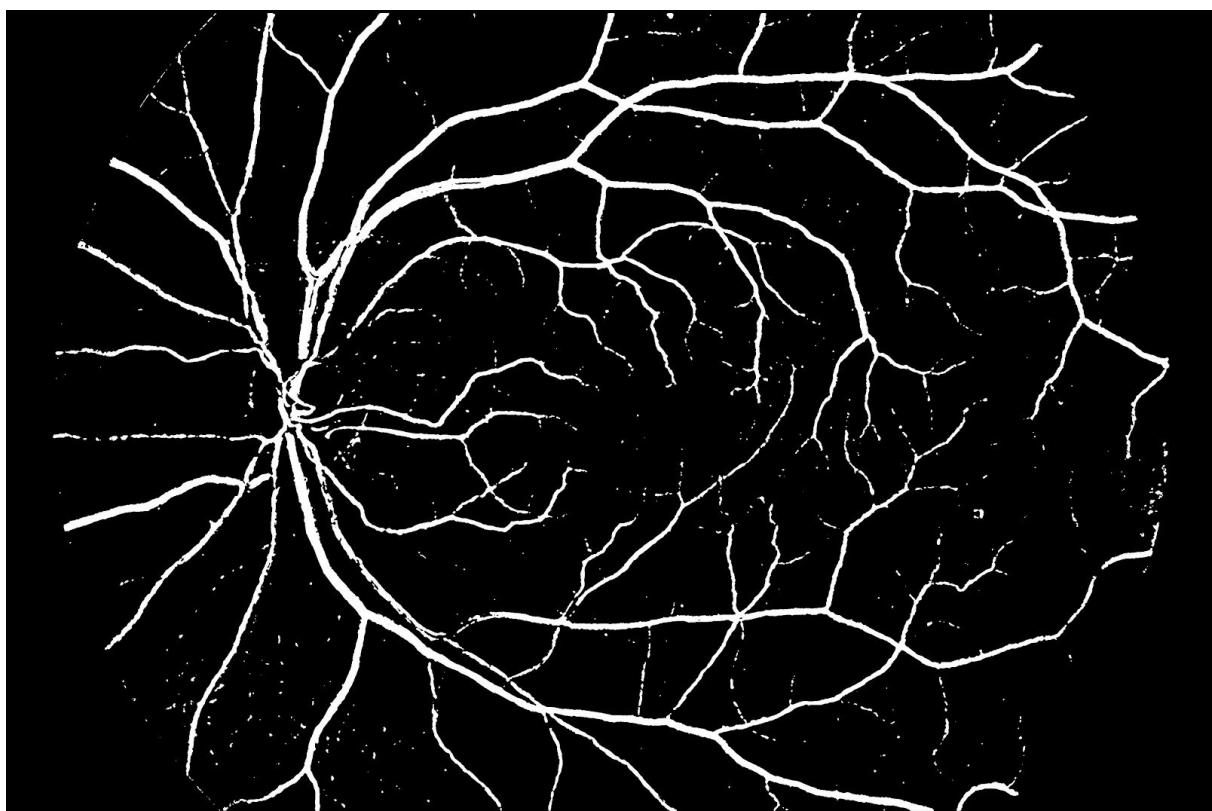
Rysunek 73: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



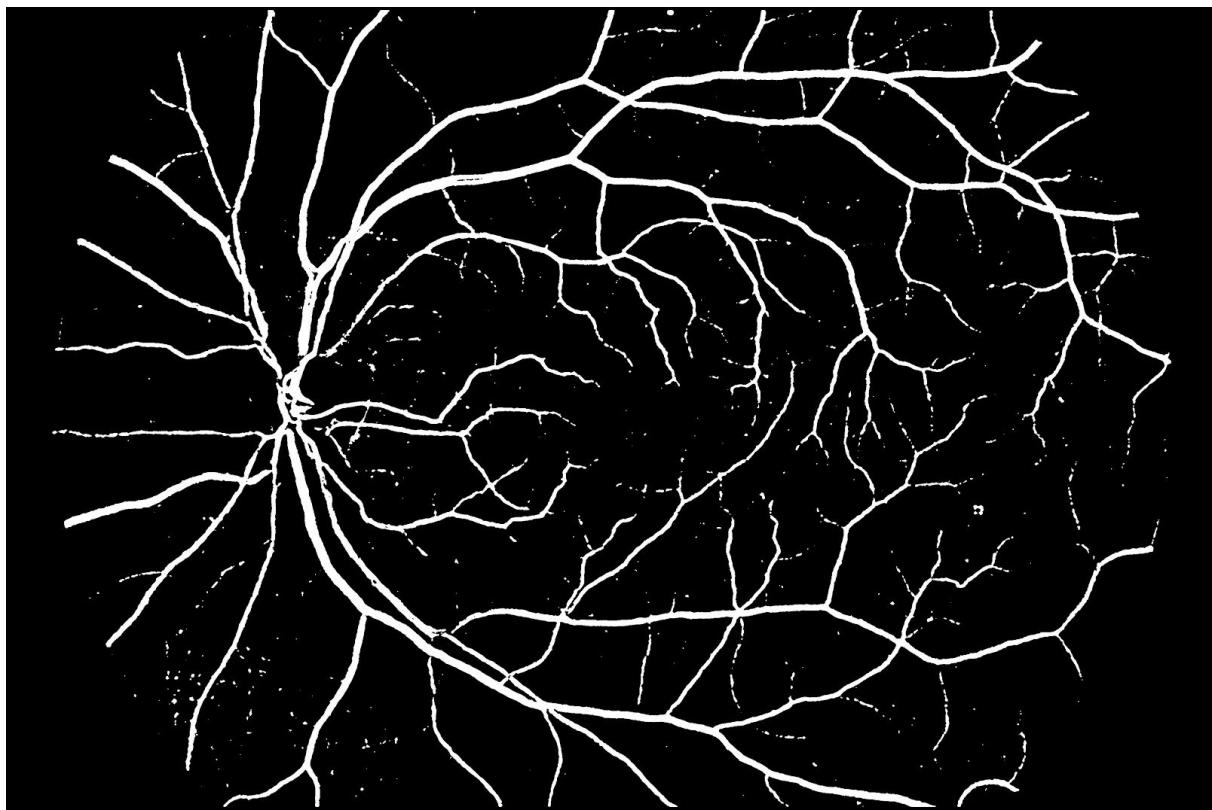
Rysunek 74: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 75: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

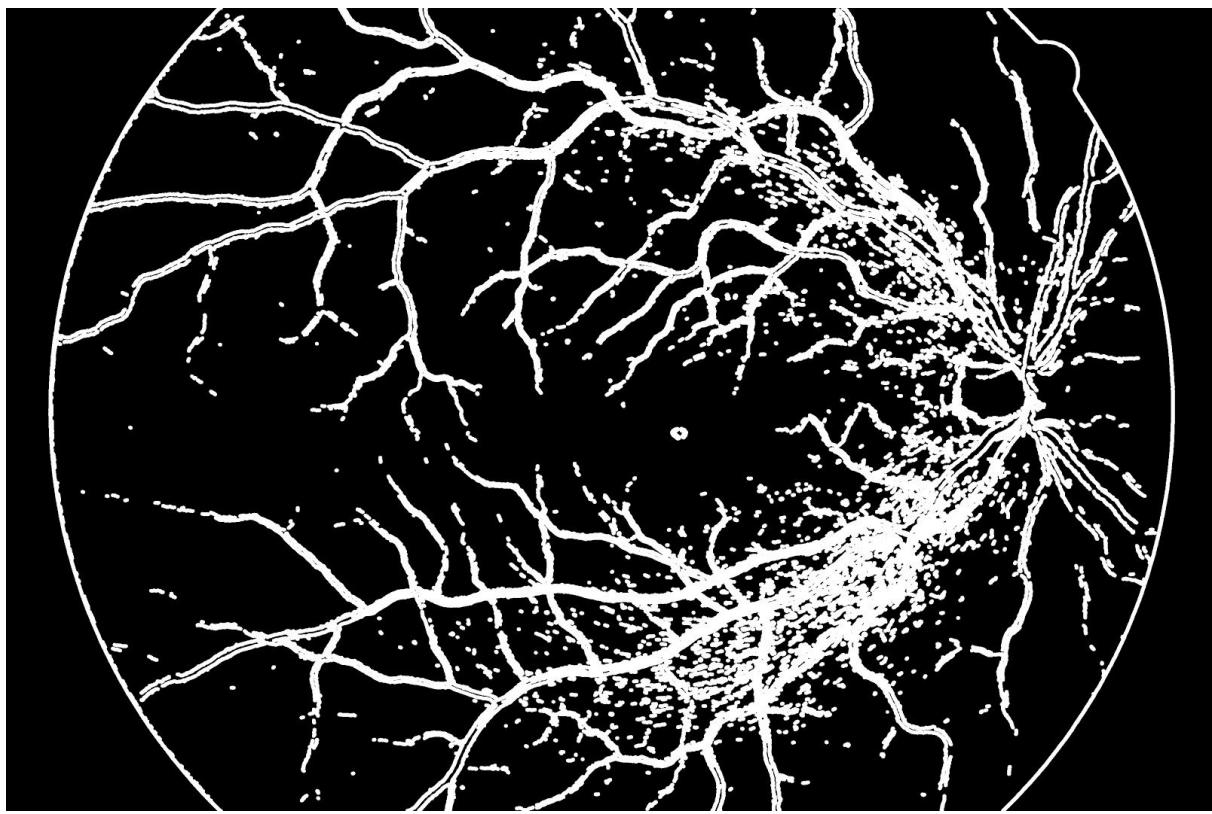


Rysunek 76: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

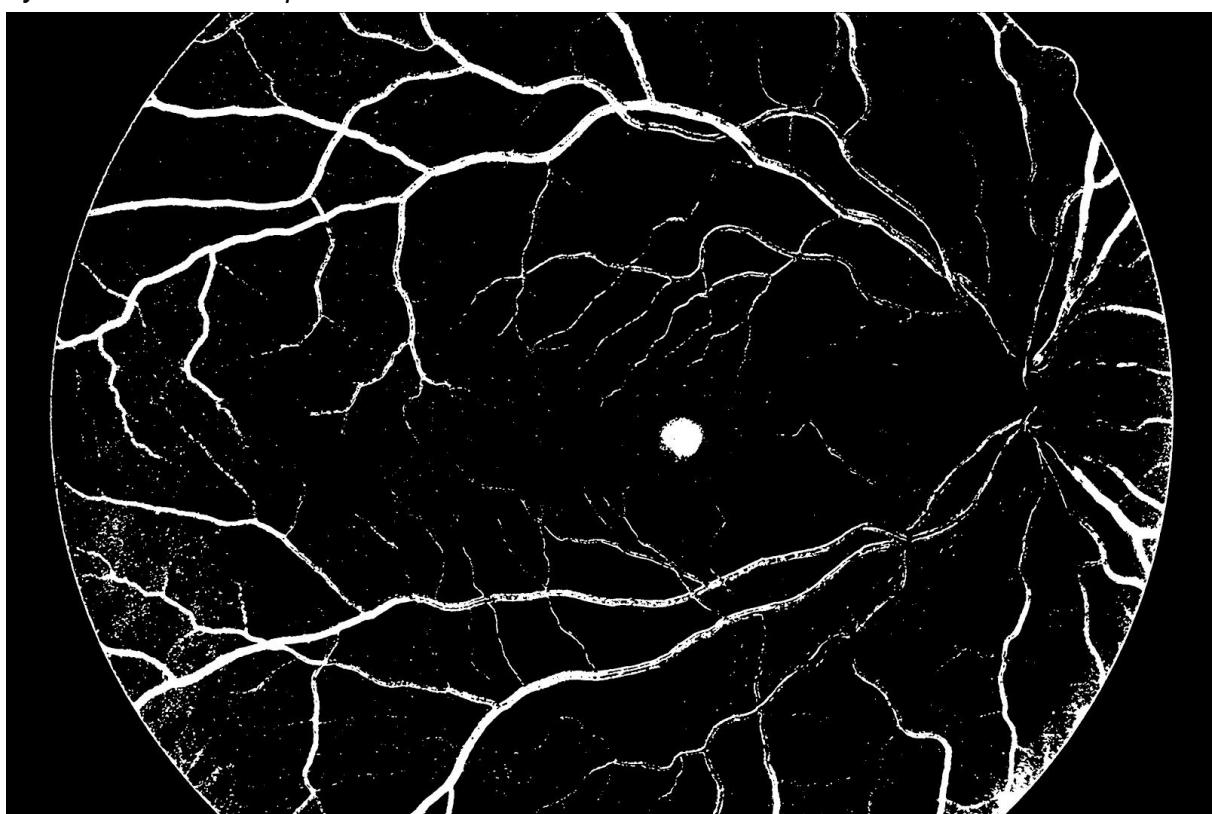


Rysunek 77: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

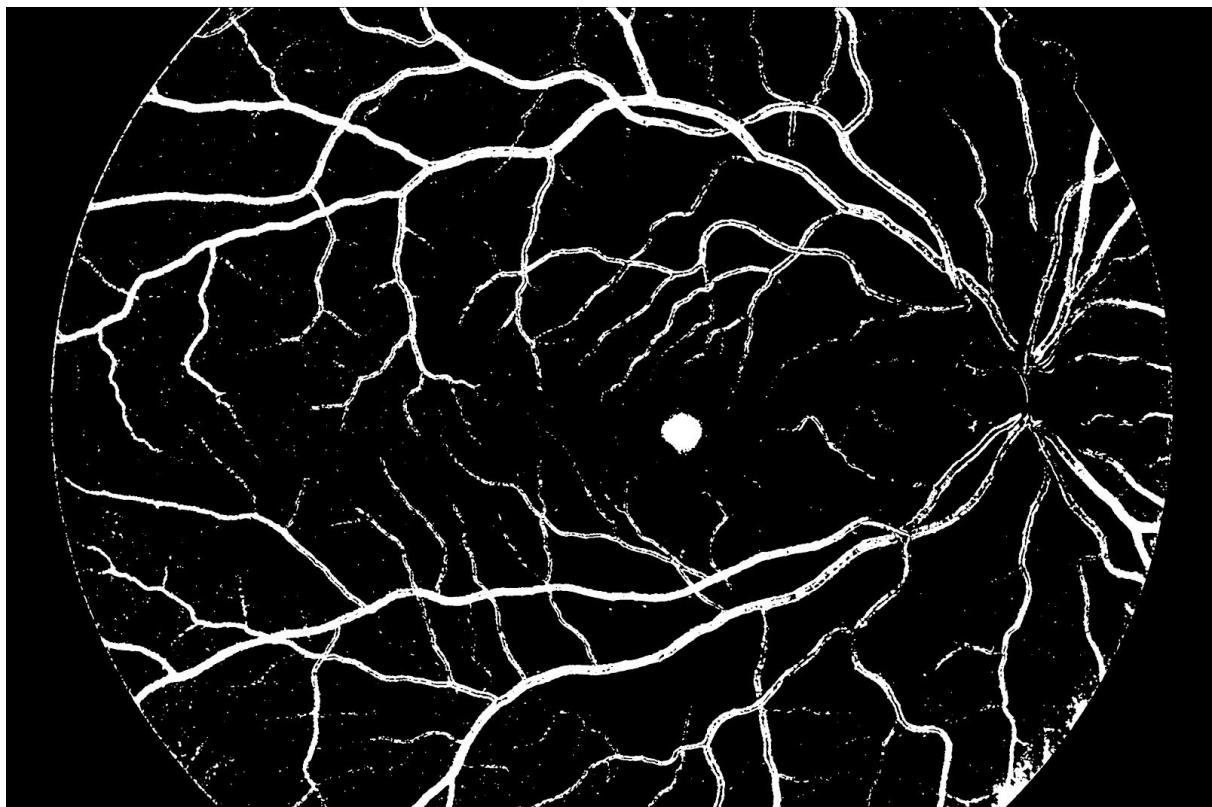
Obraz 11



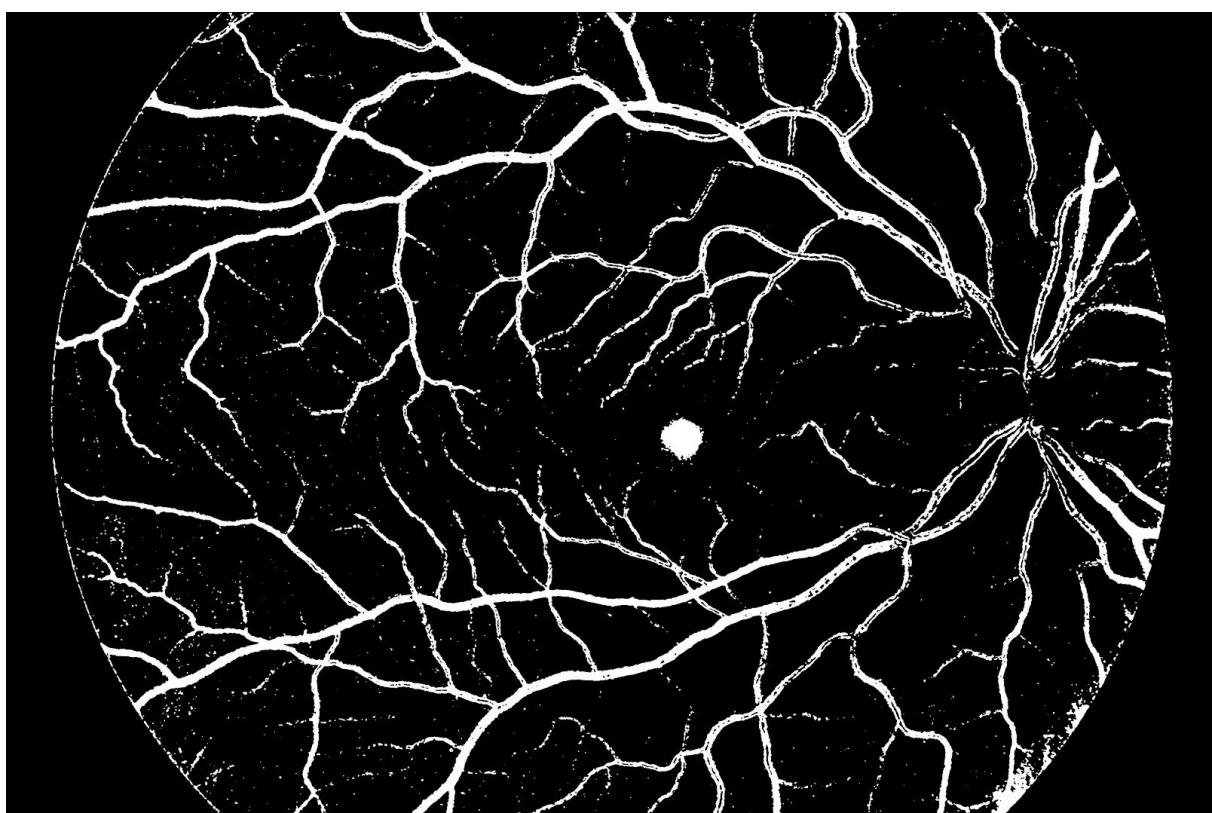
Rysunek 78: Bazowe przetwarzanie



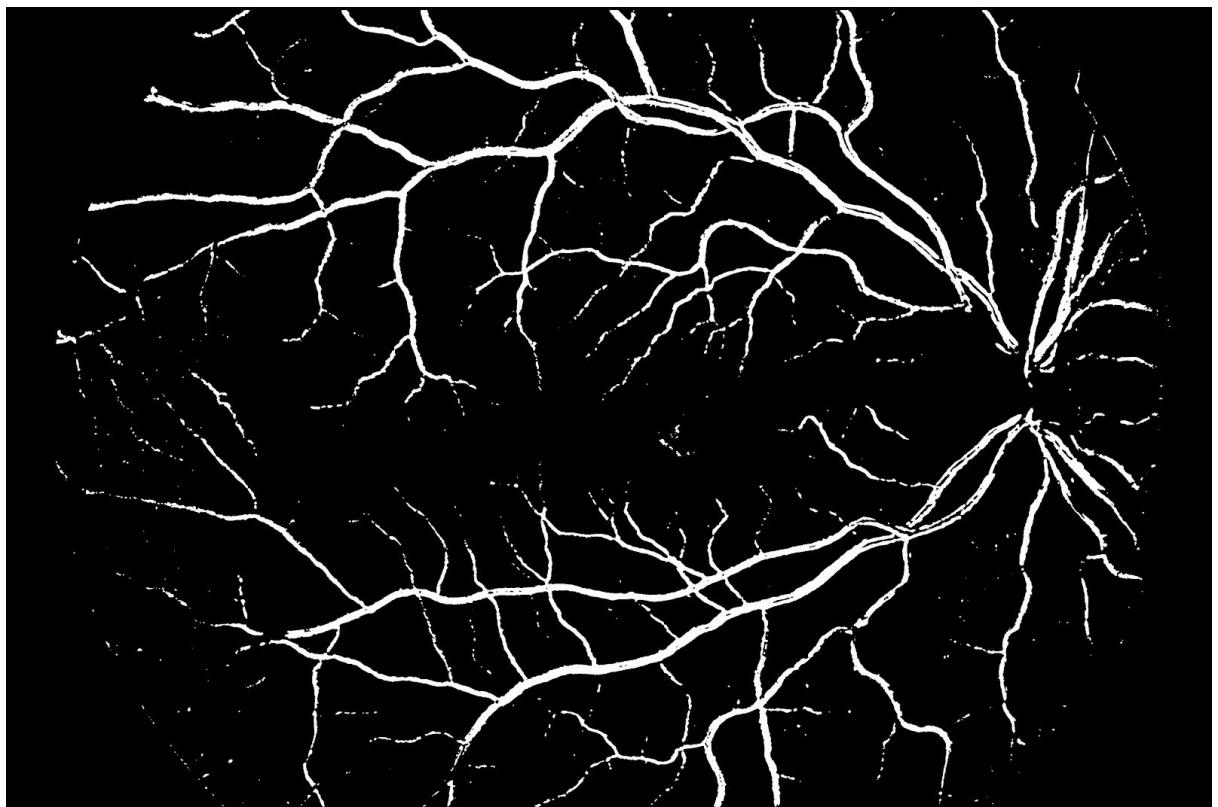
Rysunek 79: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



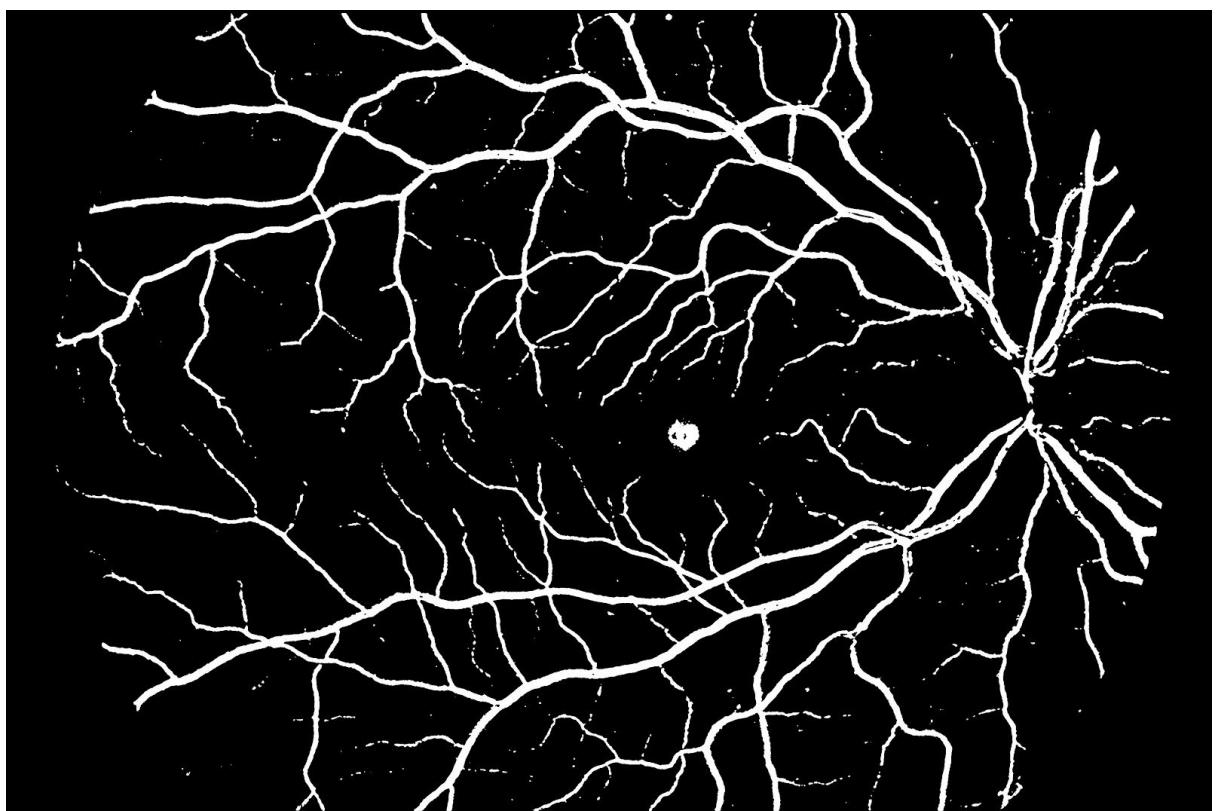
Rysunek 80: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



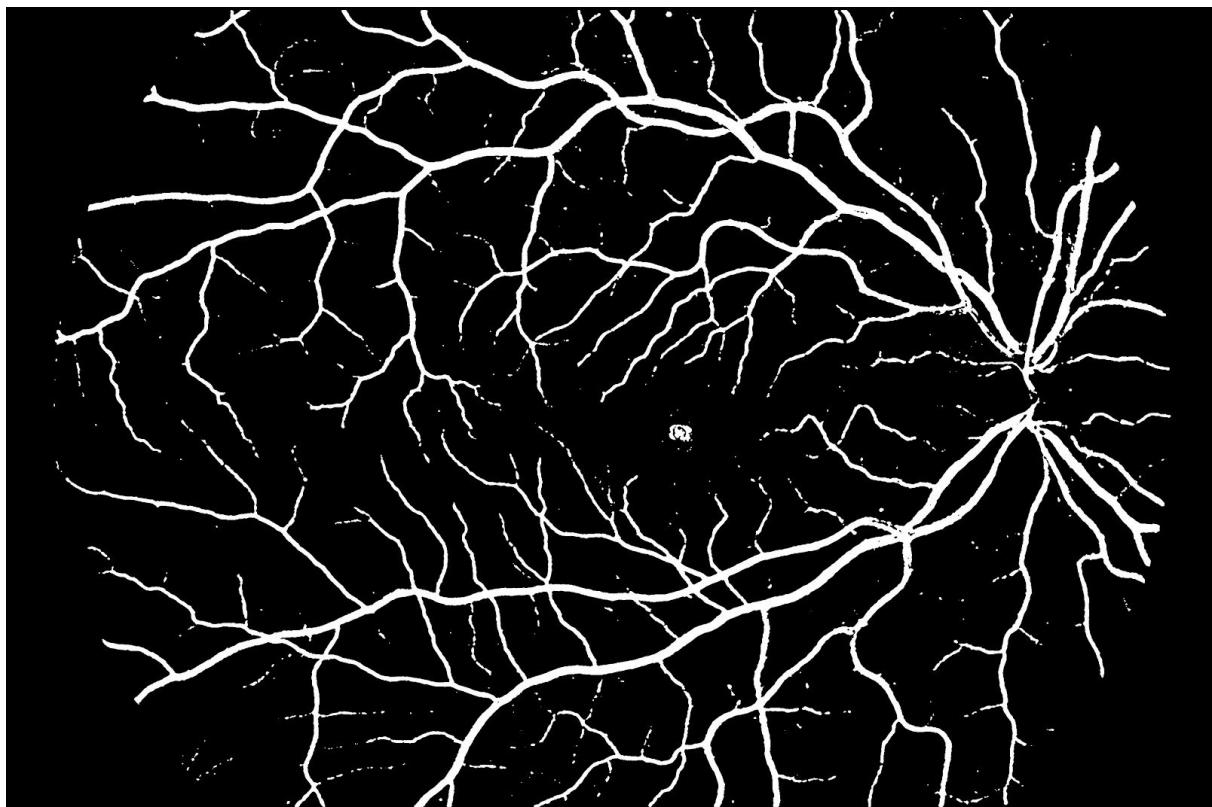
Rysunek 81: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 82: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

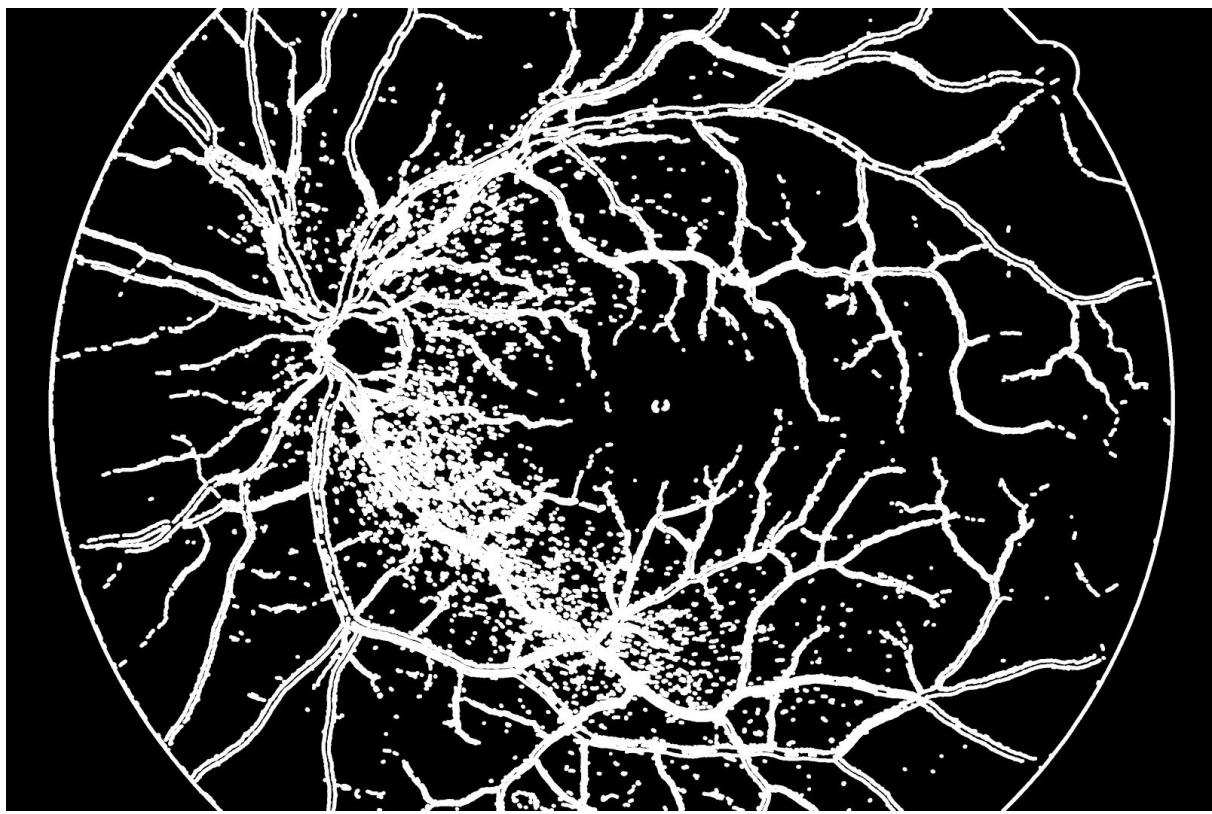


Rysunek 83: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

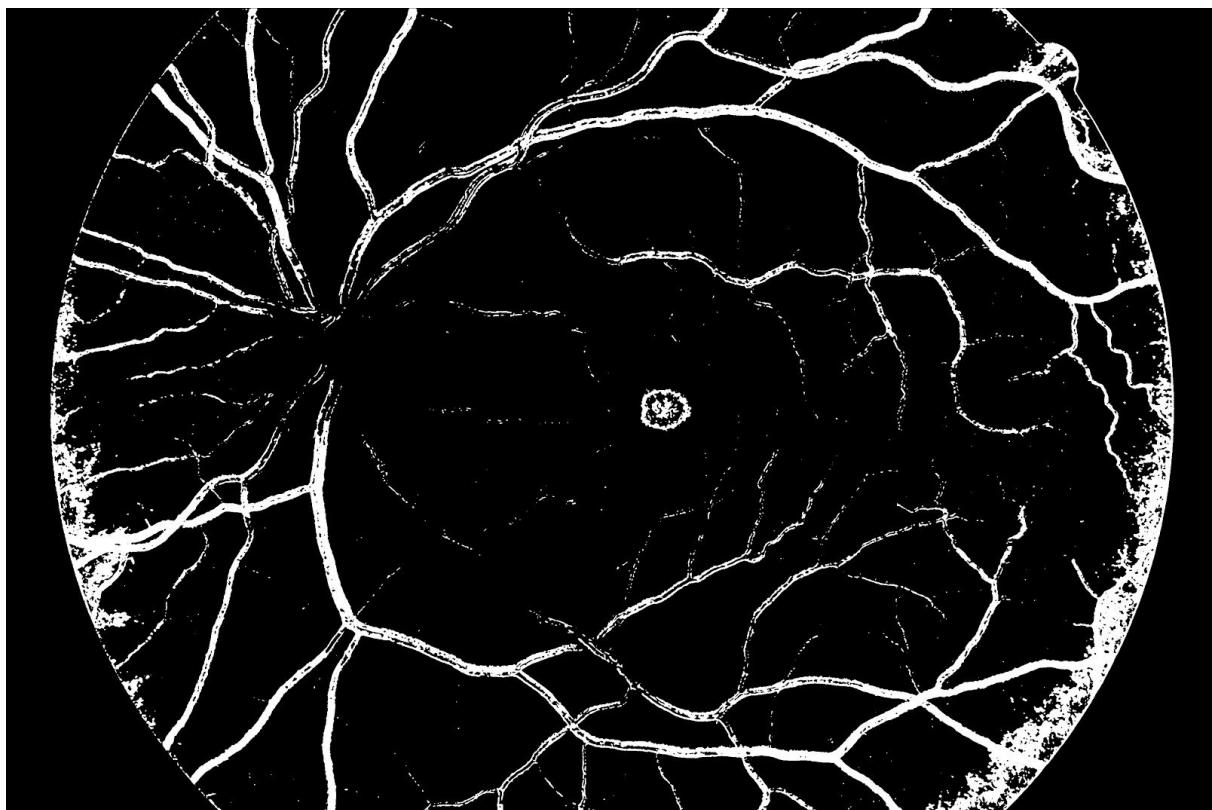


Rysunek 84: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

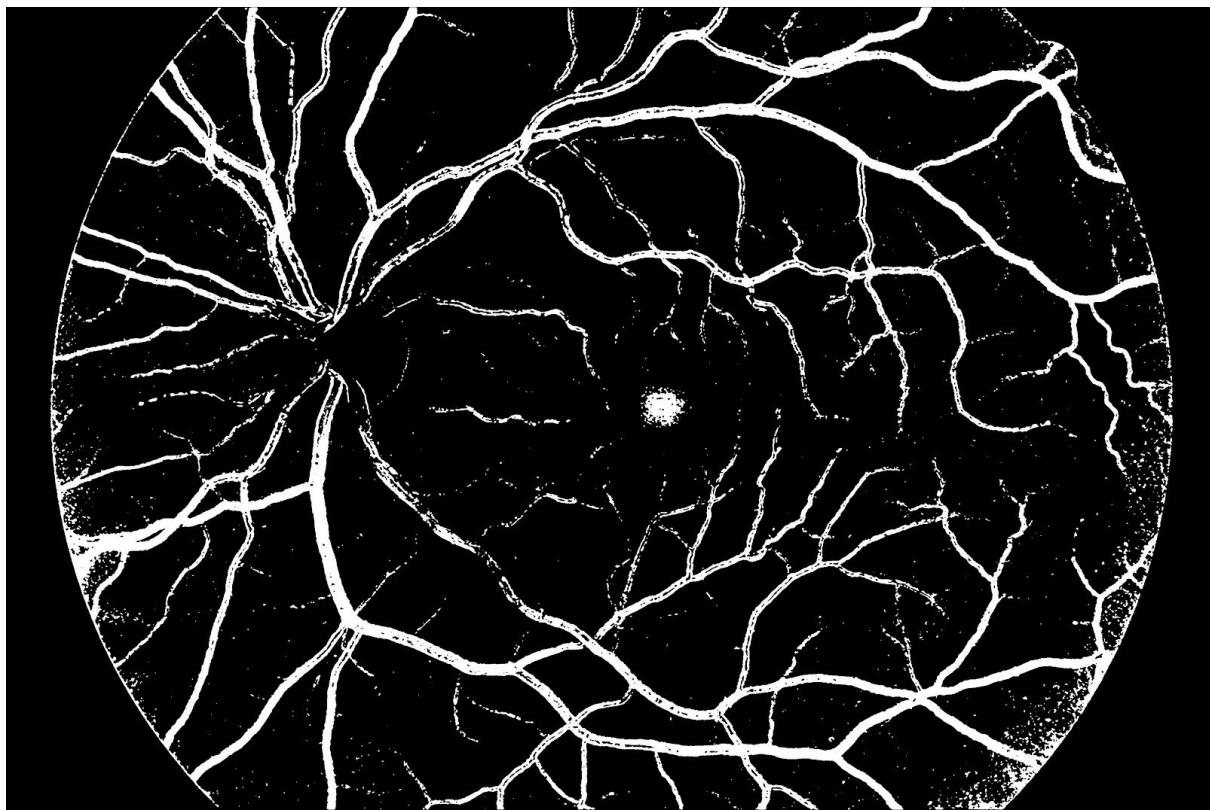
Obraz 12



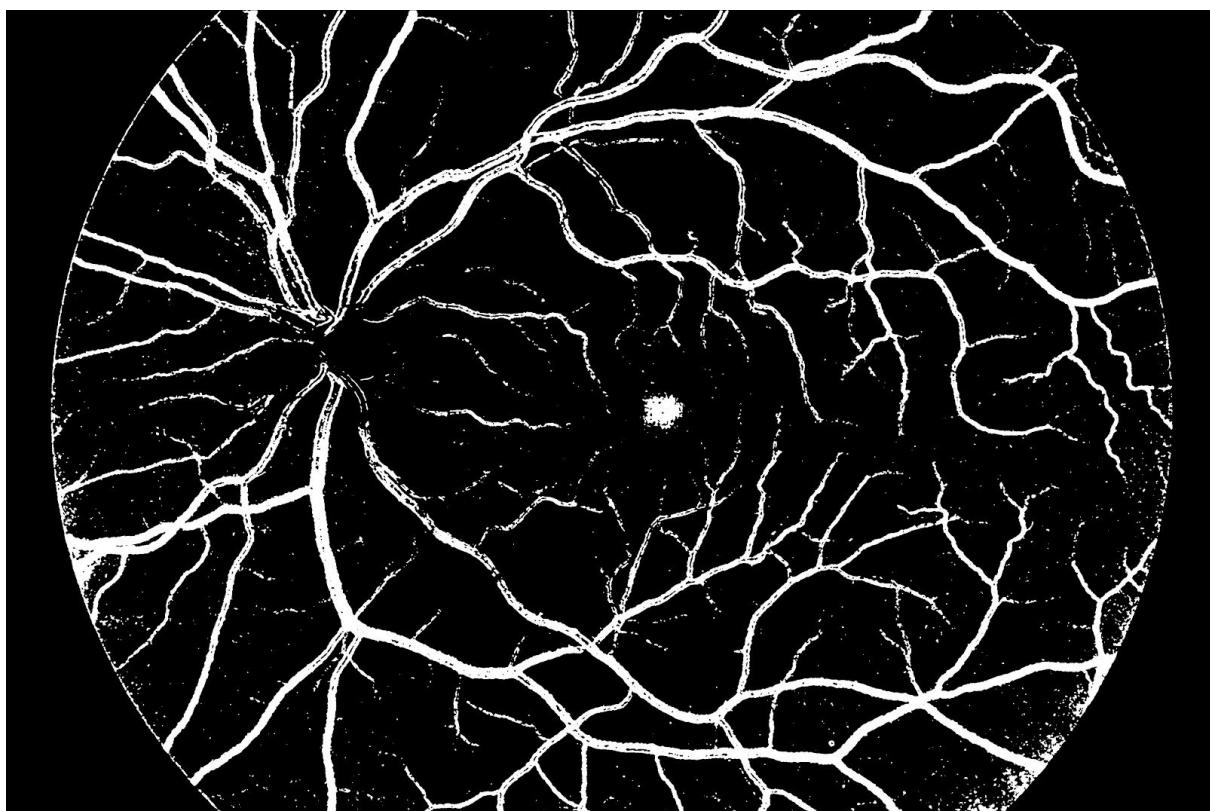
Rysunek 85: Bazowe przetwarzanie



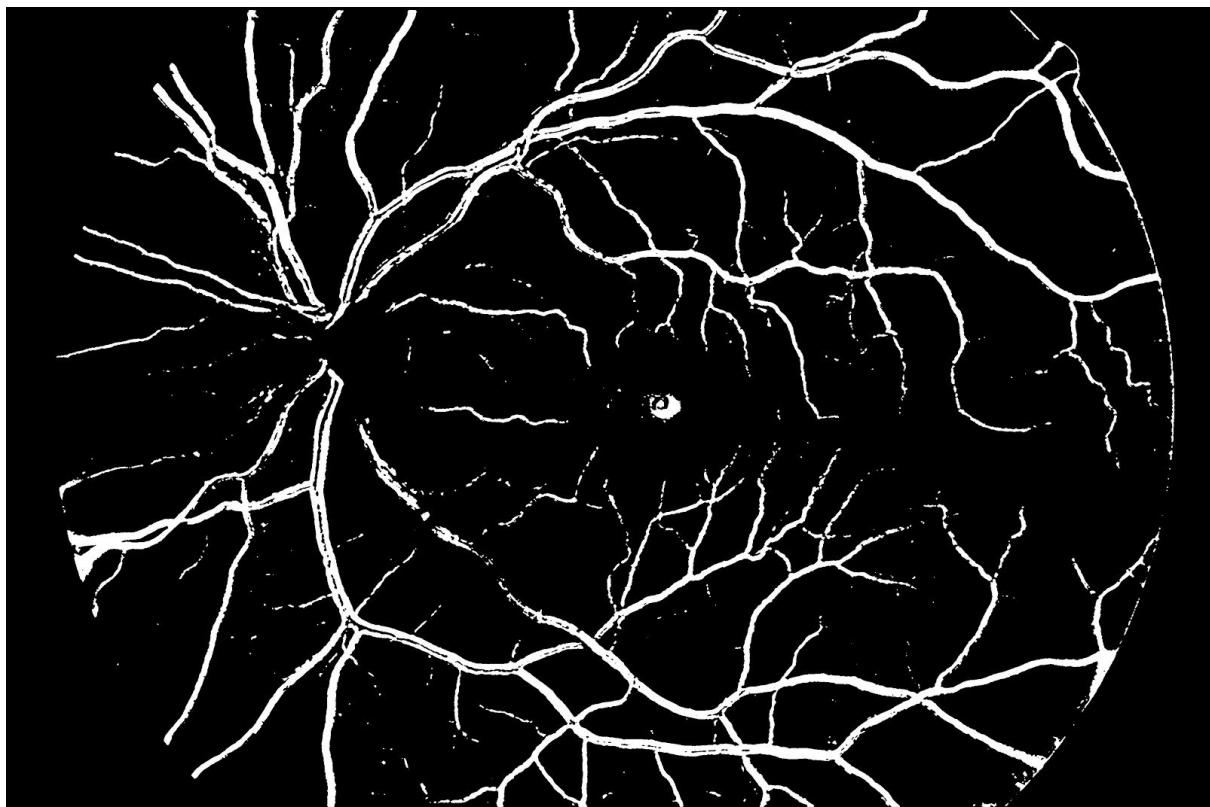
Rysunek 86: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



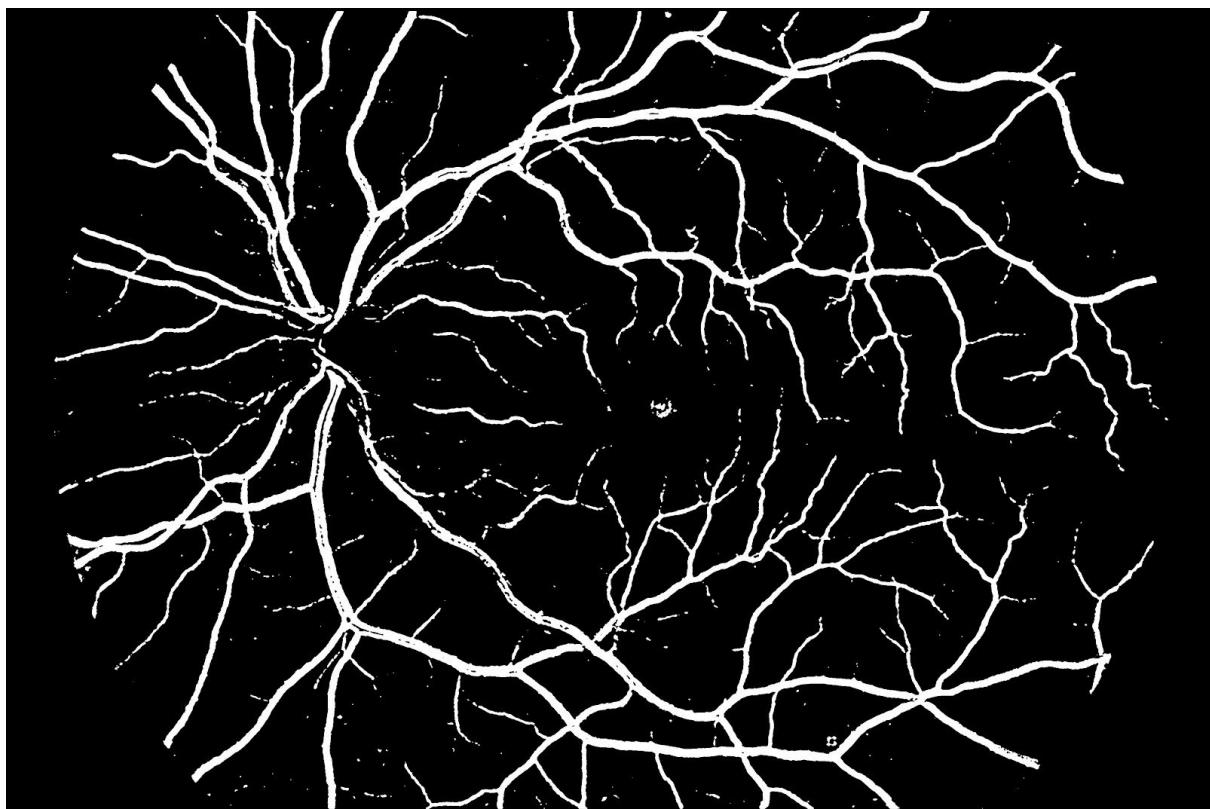
Rysunek 87: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



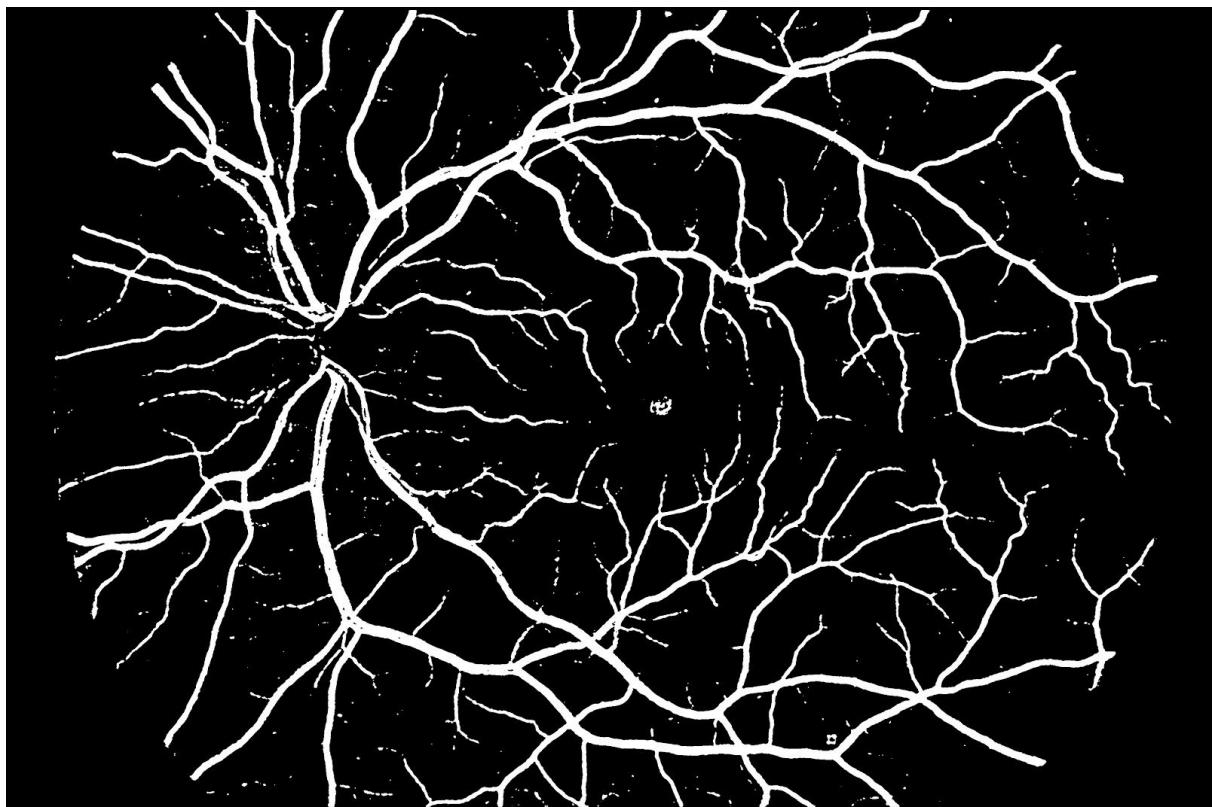
Rysunek 88: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 89: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

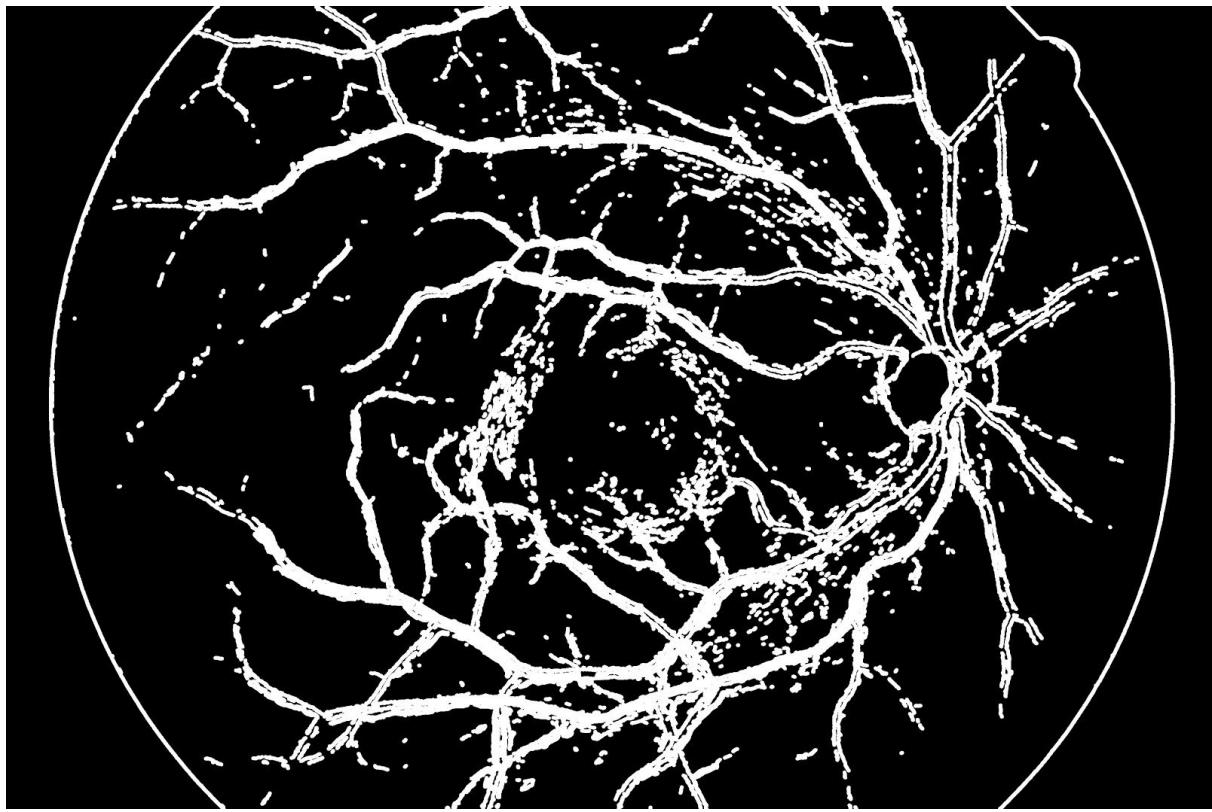


Rysunek 90: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

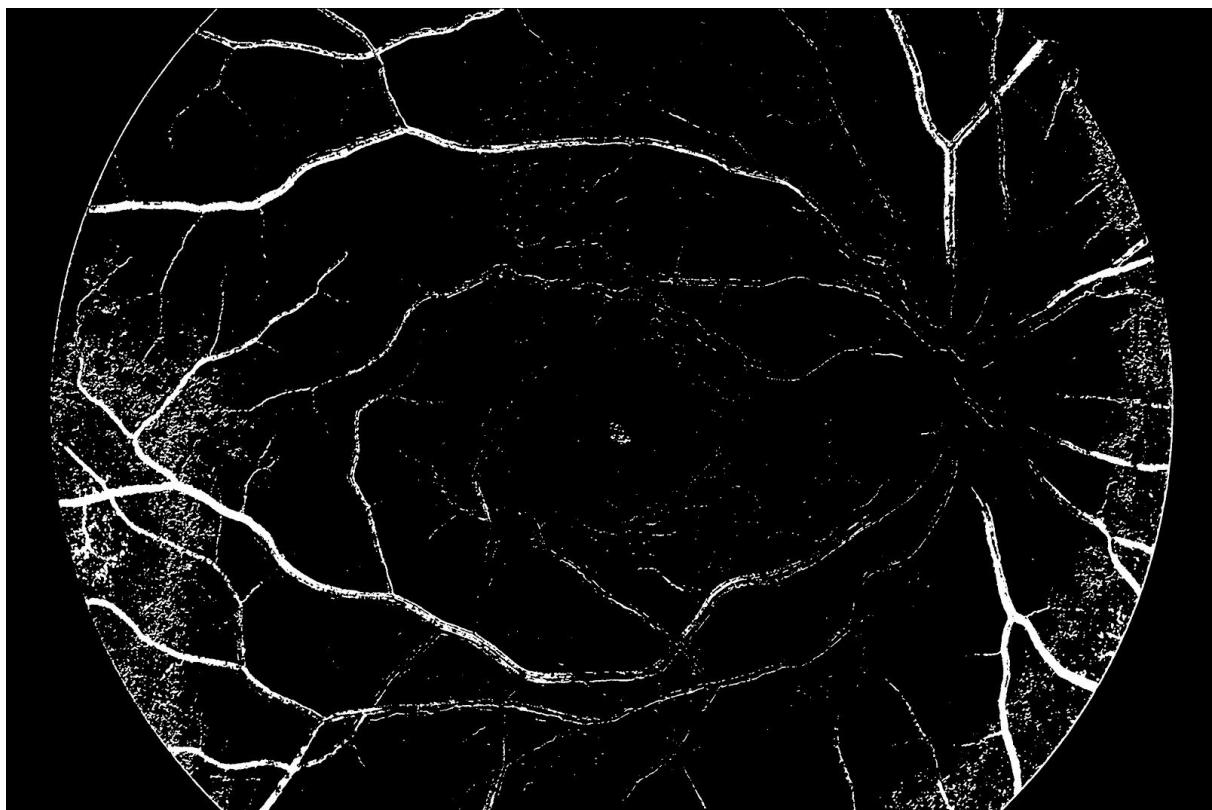


Rysunek 91: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

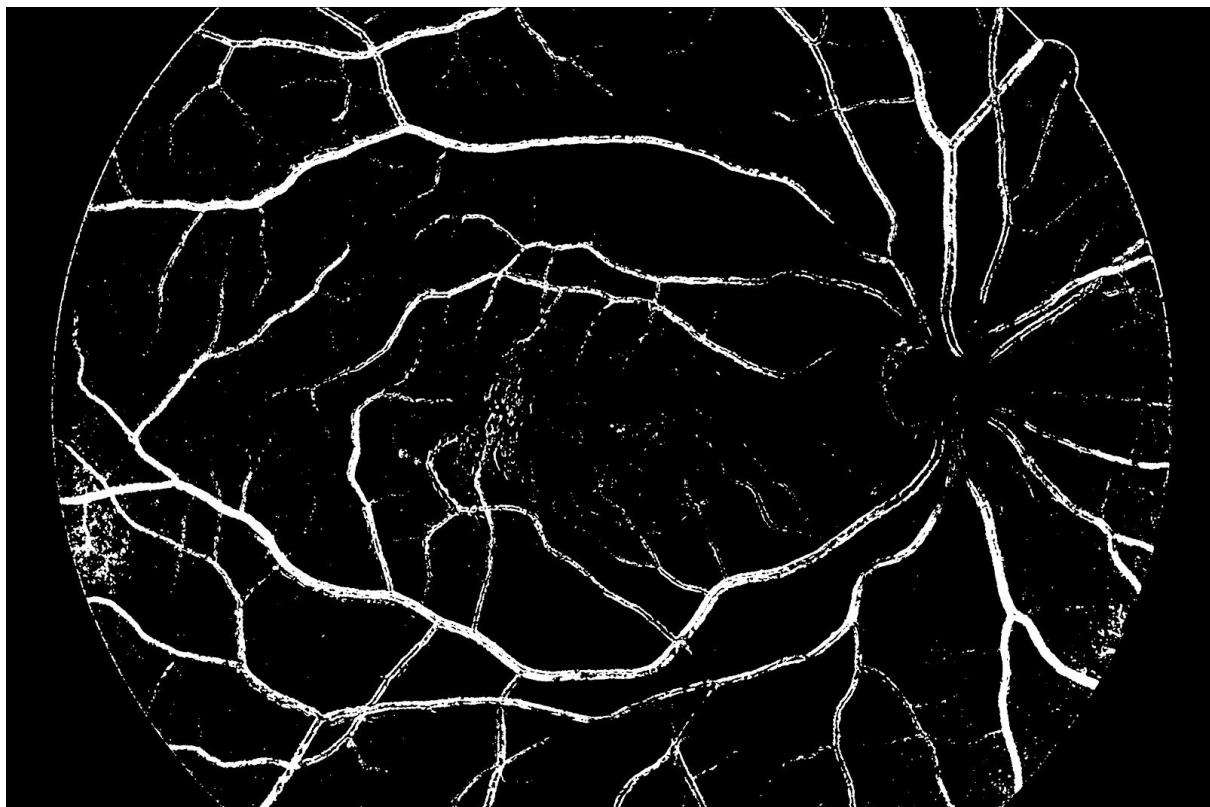
Obraz 13



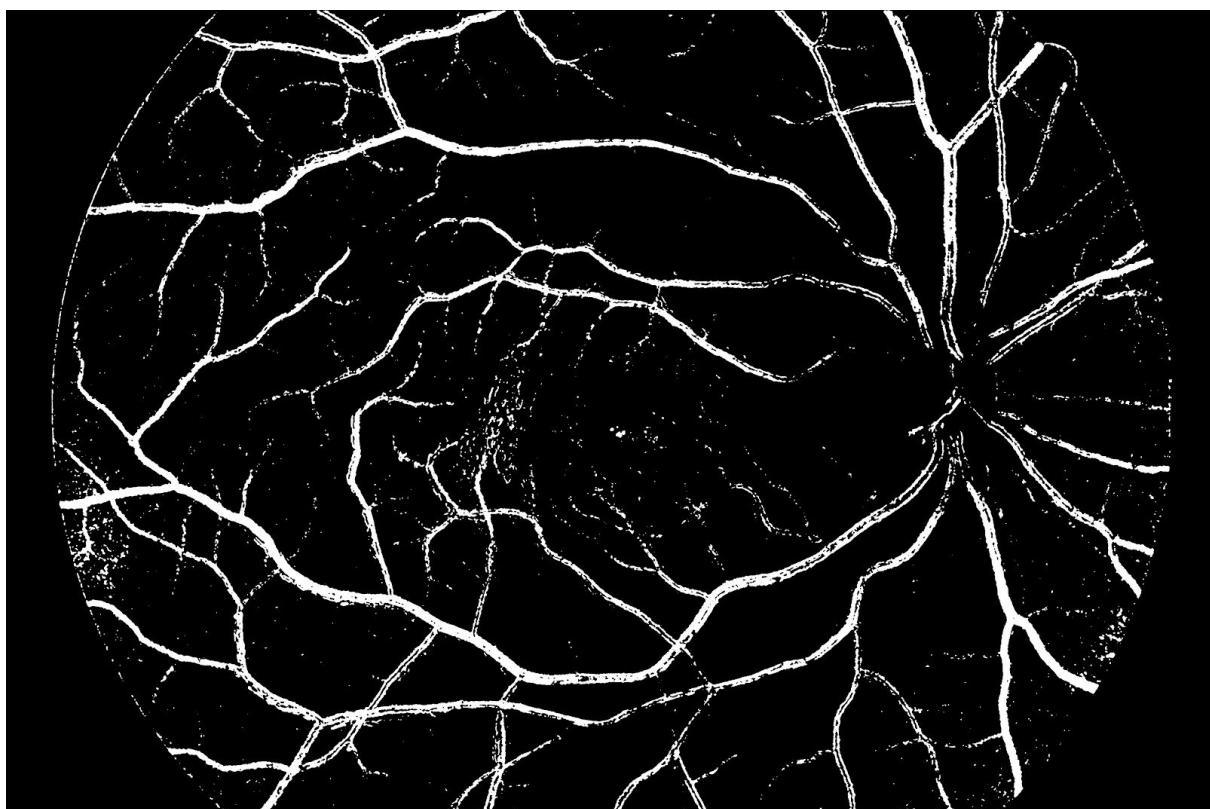
Rysunek 92: Bazowe przetwarzanie



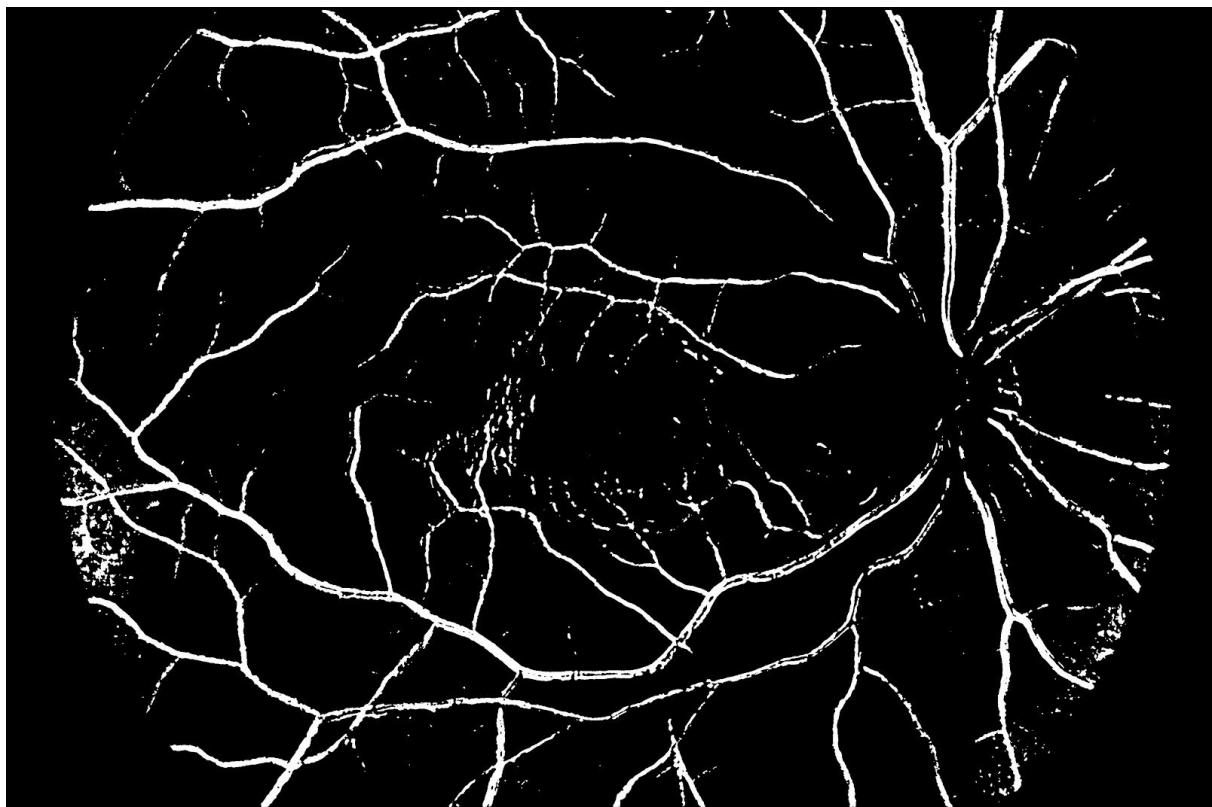
Rysunek 93: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



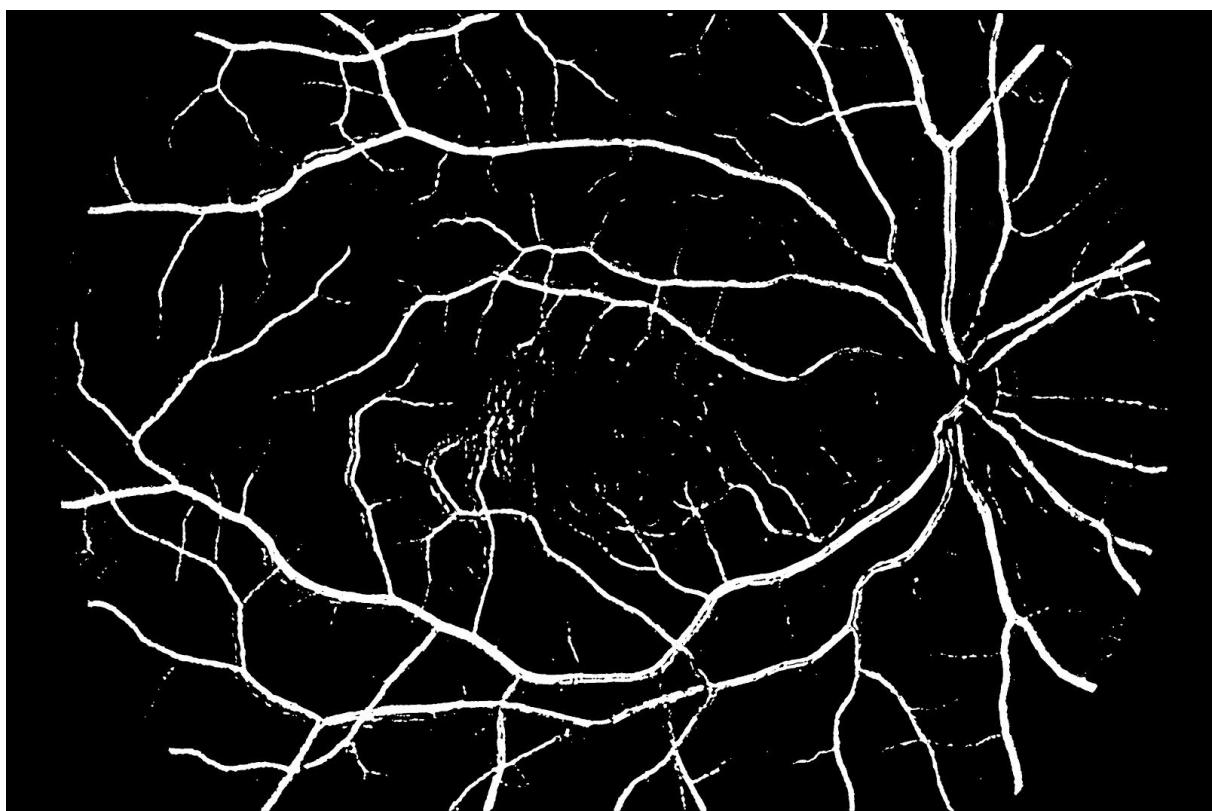
Rysunek 94: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



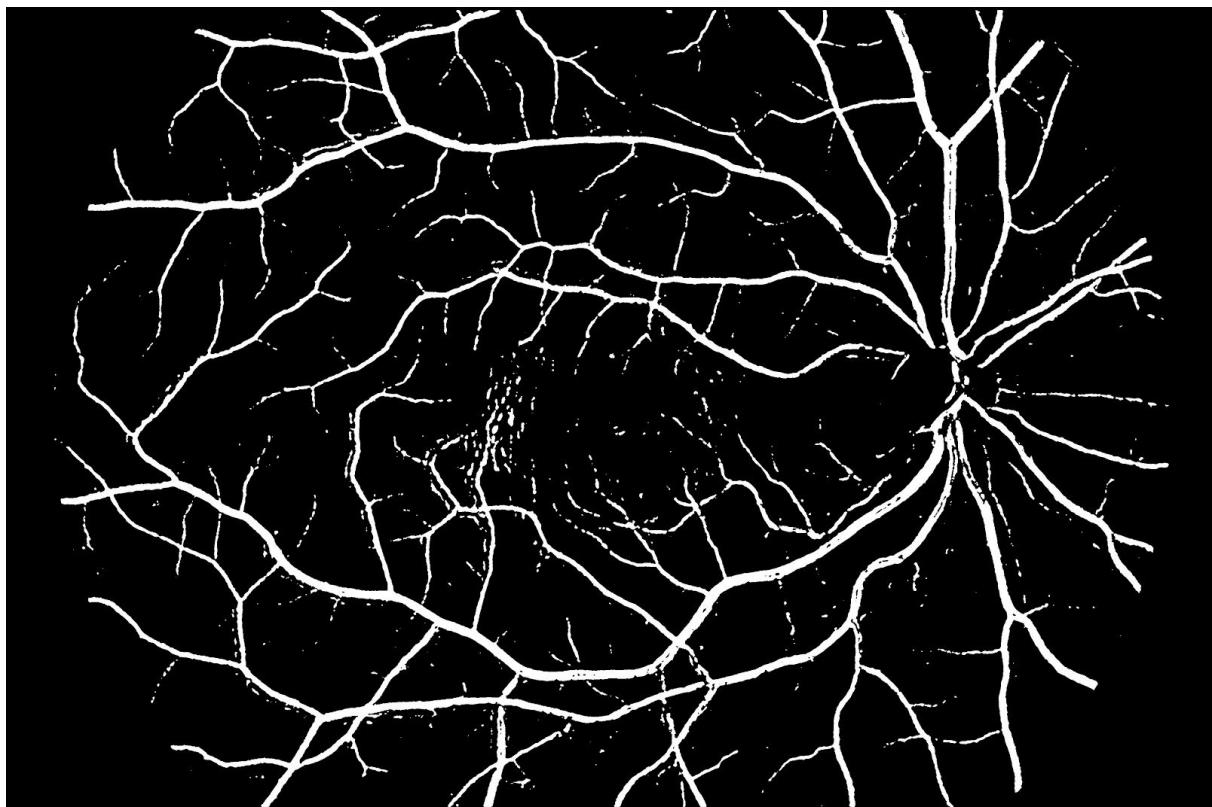
Rysunek 95: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 96: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

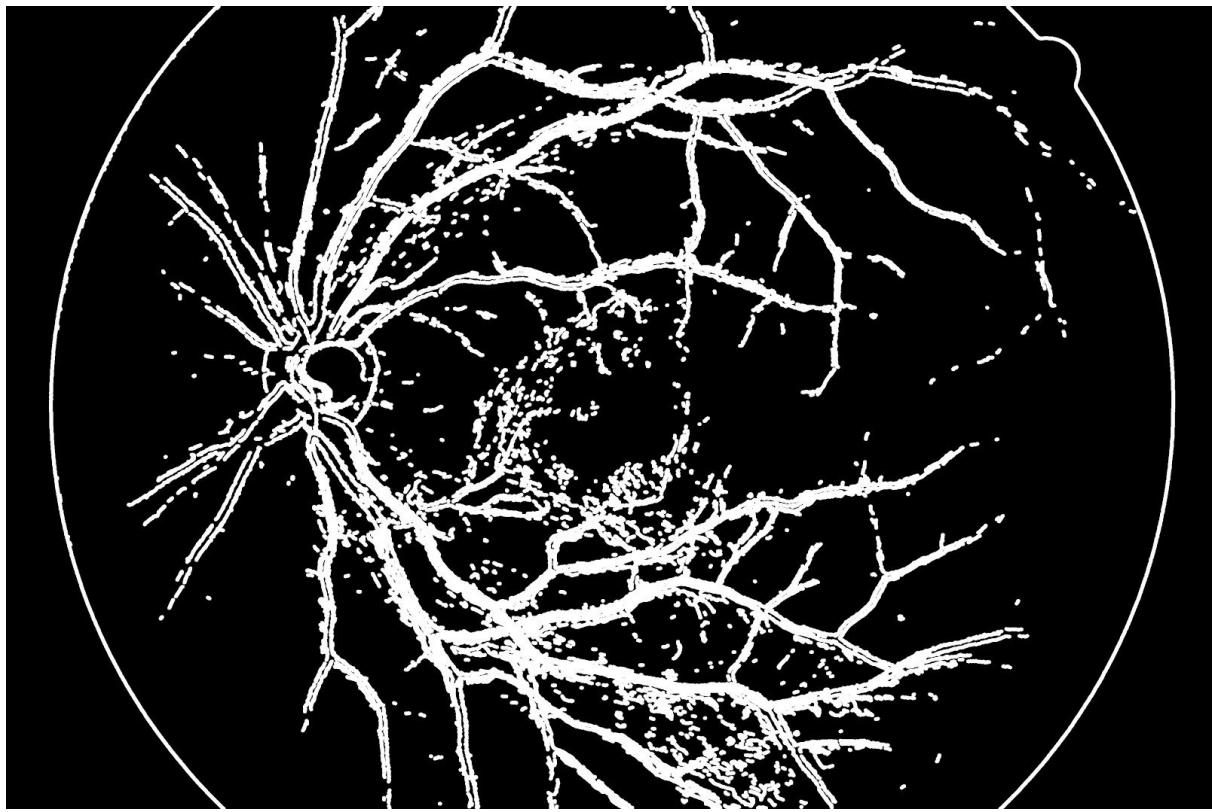


Rysunek 97: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

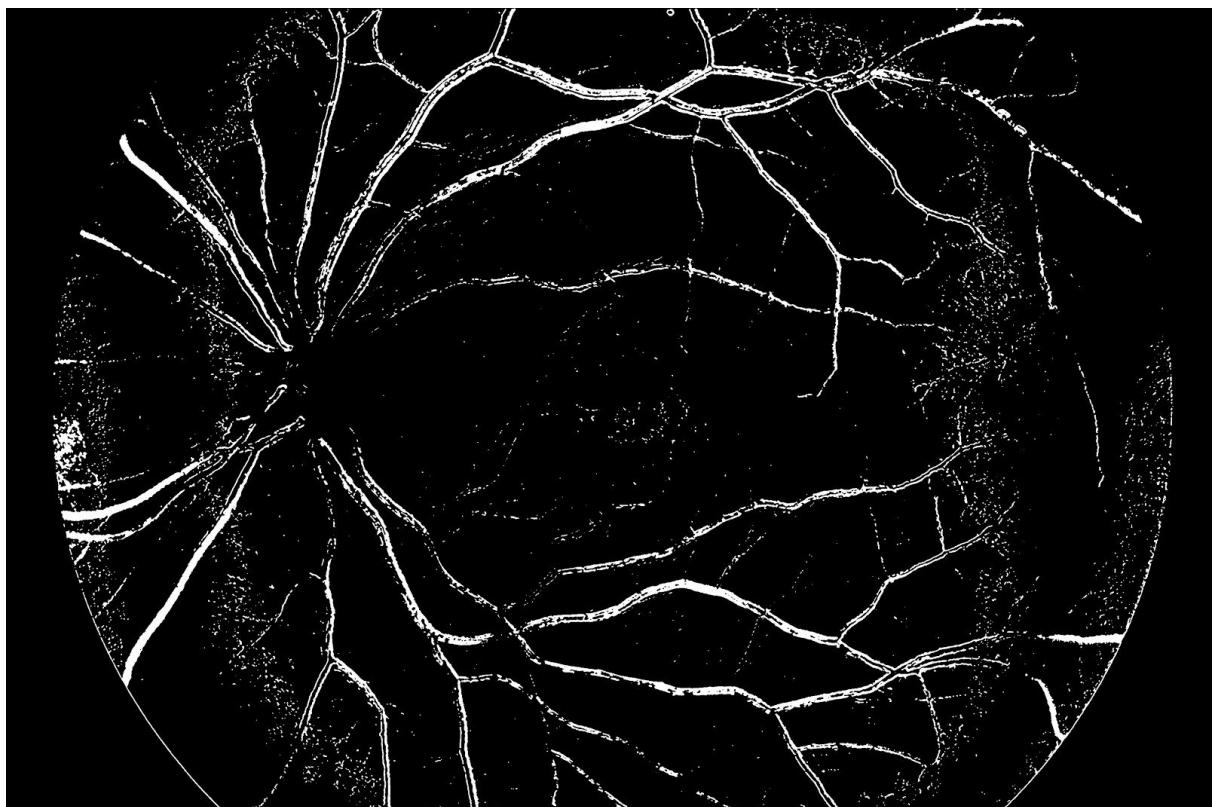


Rysunek 98: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

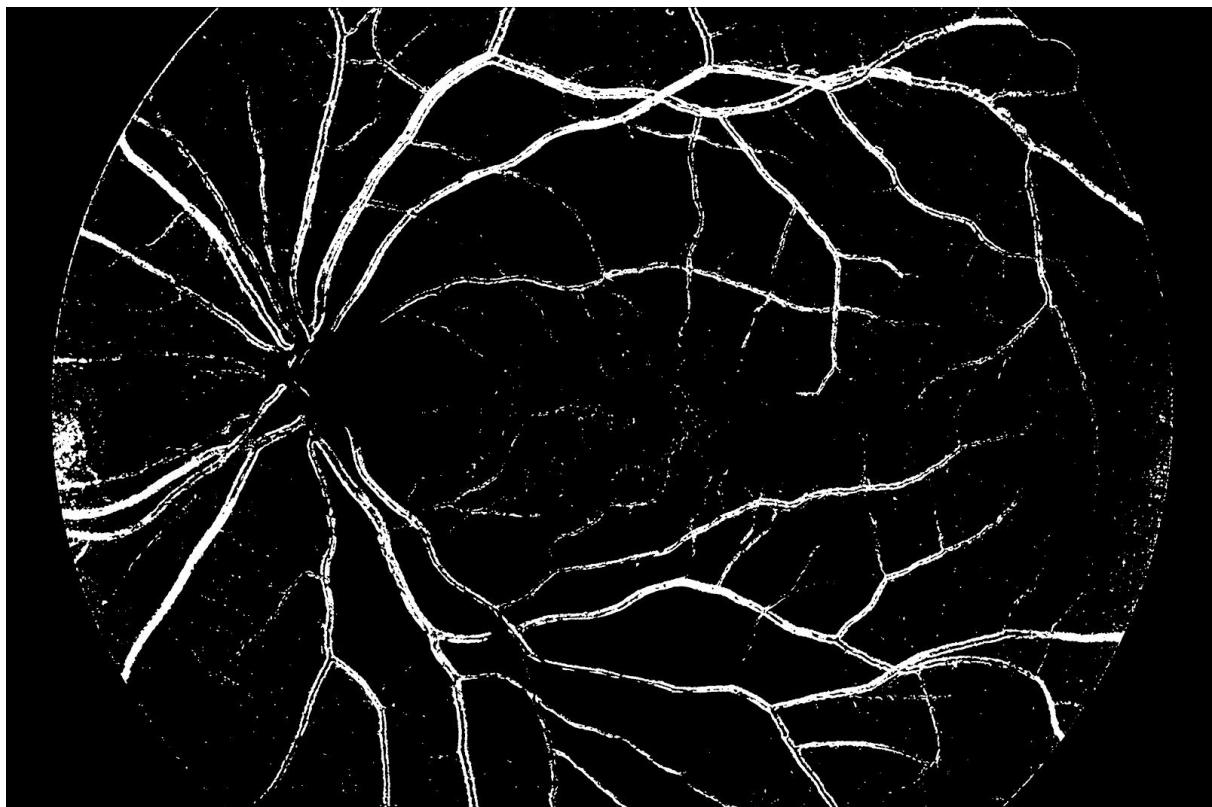
Obraz 14



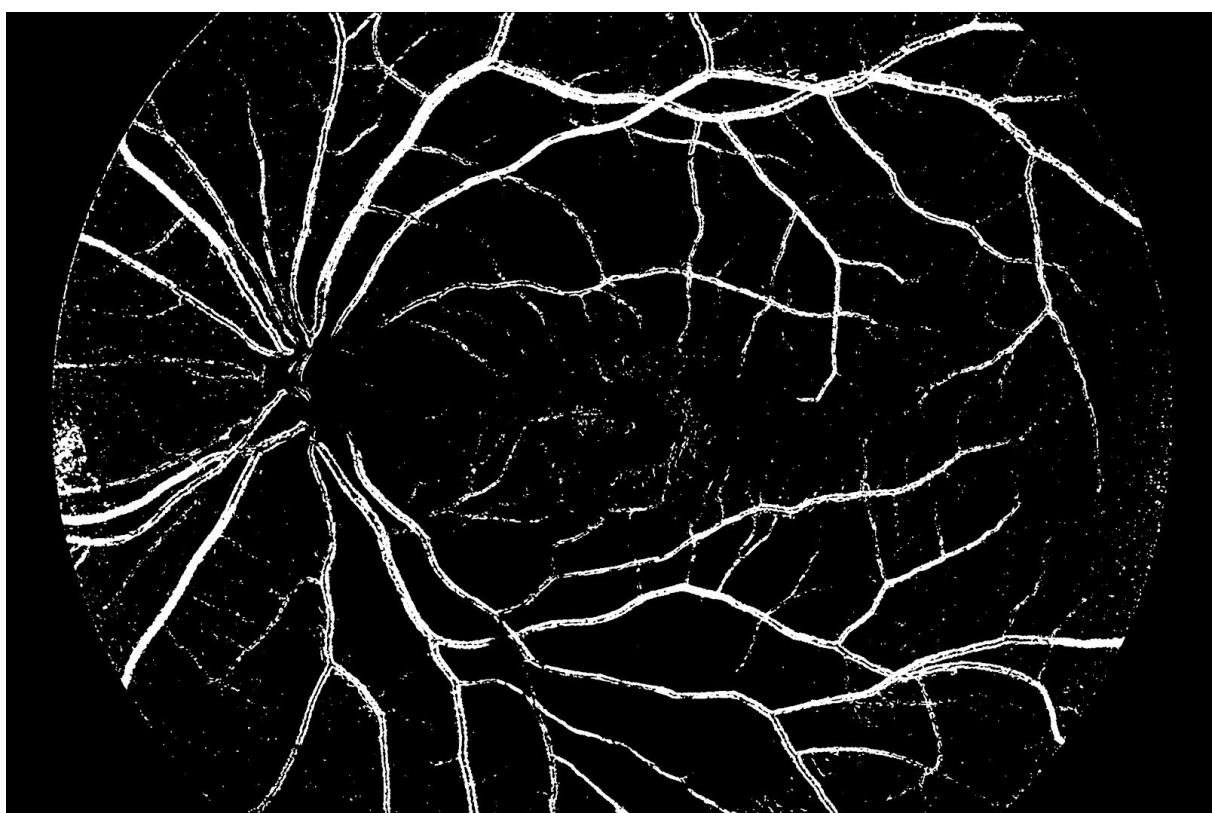
Rysunek 99: Bazowe przetwarzanie



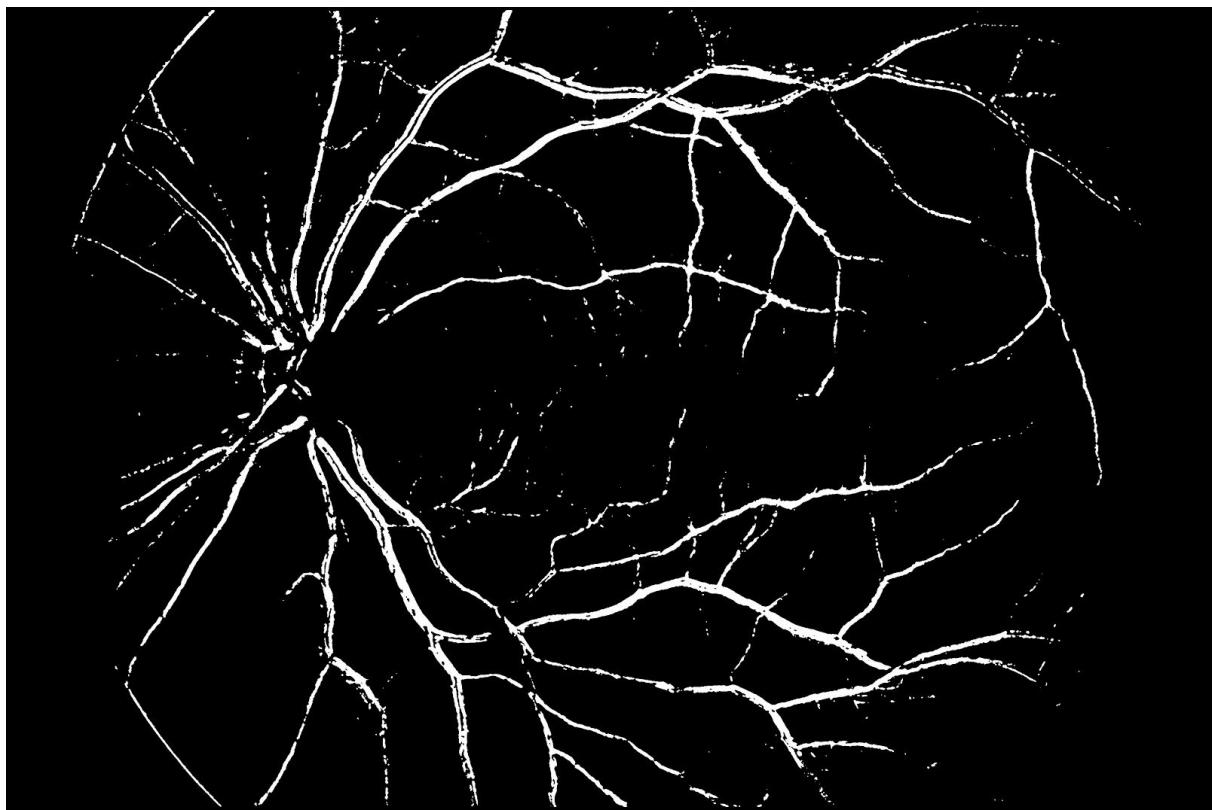
Rysunek 100: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



Rysunek 101: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



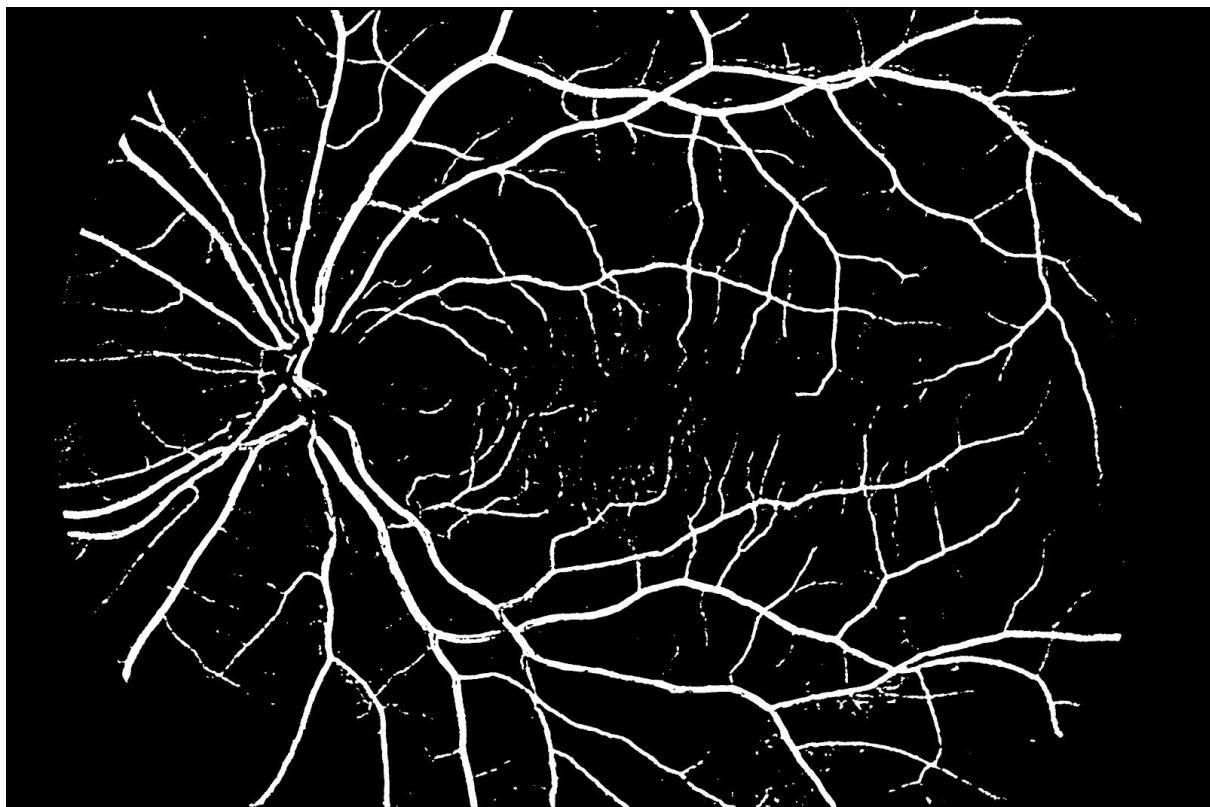
Rysunek 102: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 103: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze

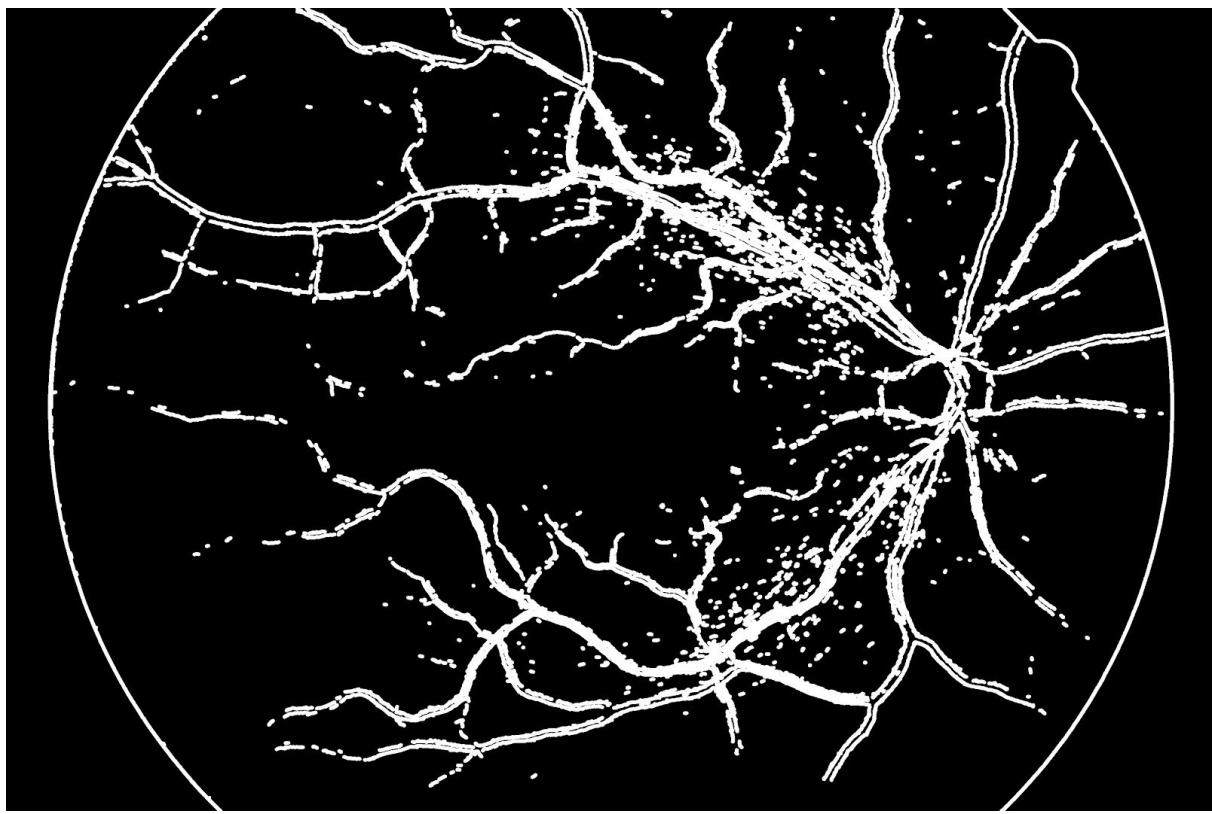


Rysunek 104: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze

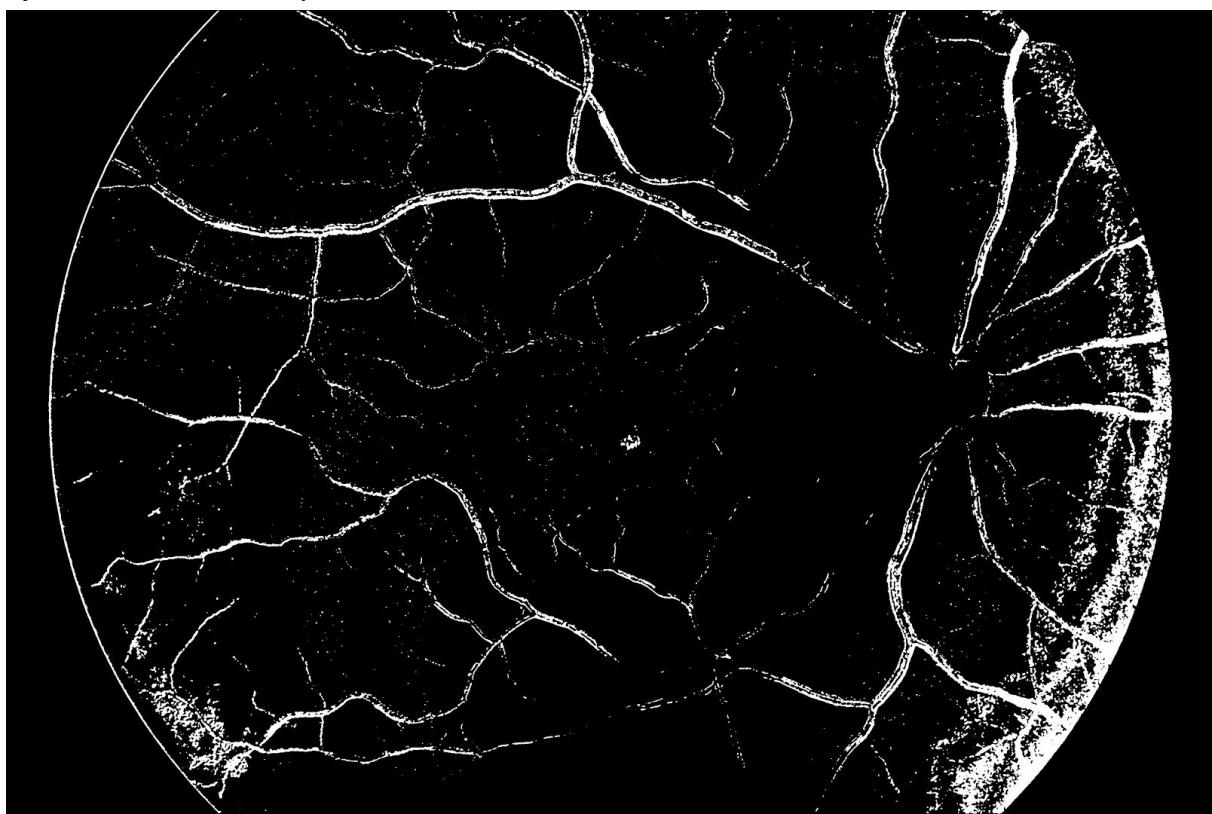


Rysunek 105: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze

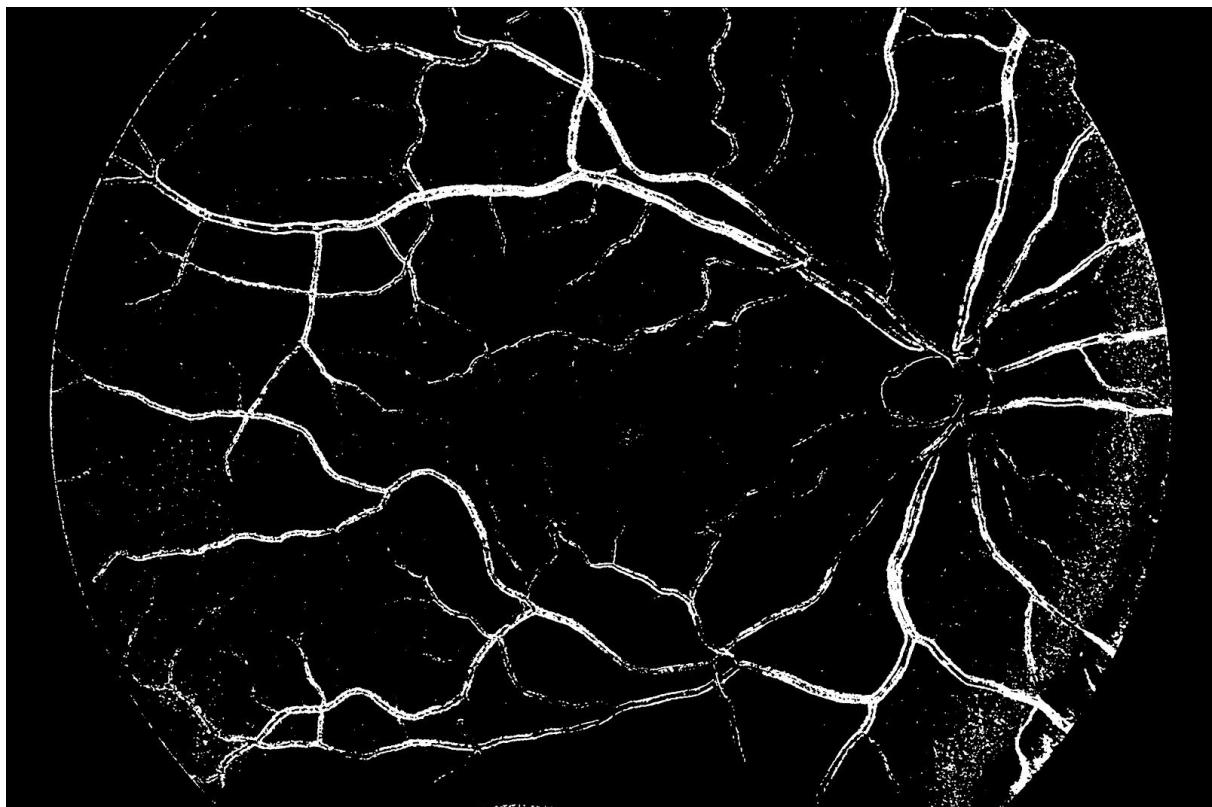
Obraz 15



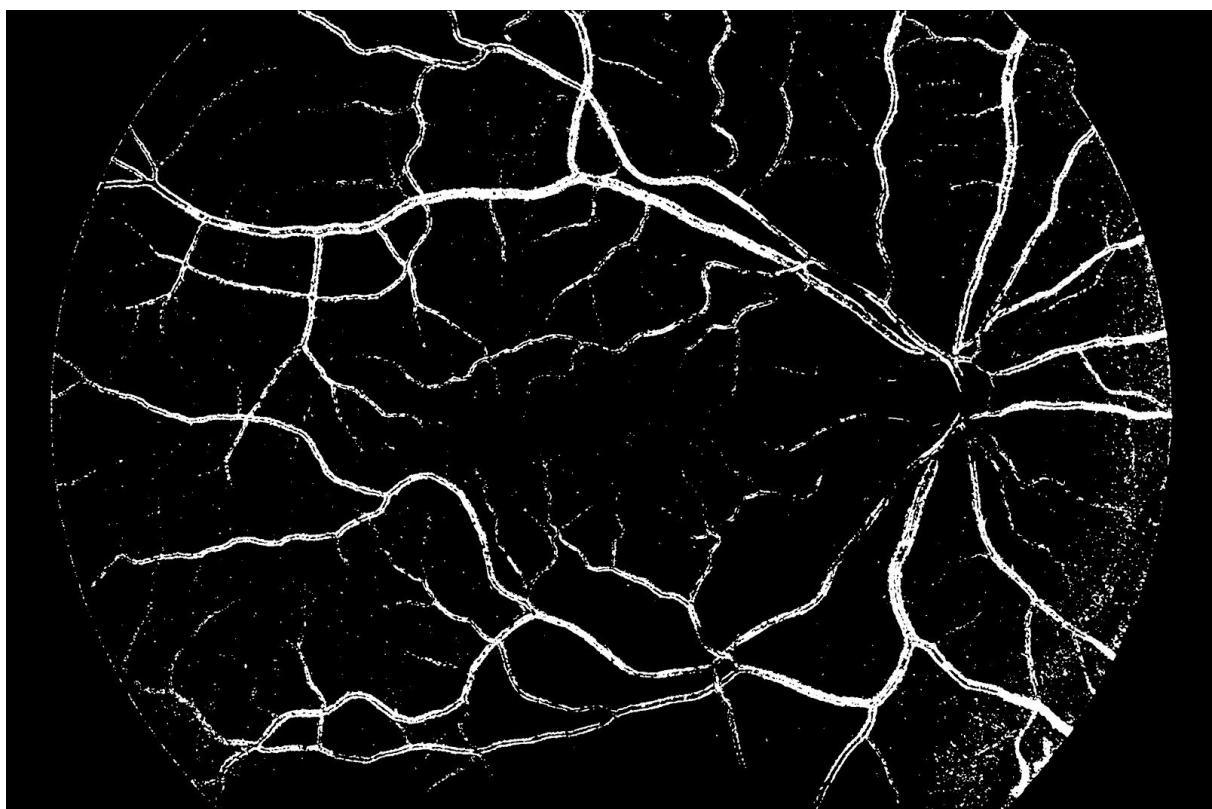
Rysunek 106: Bazowe przetwarzanie



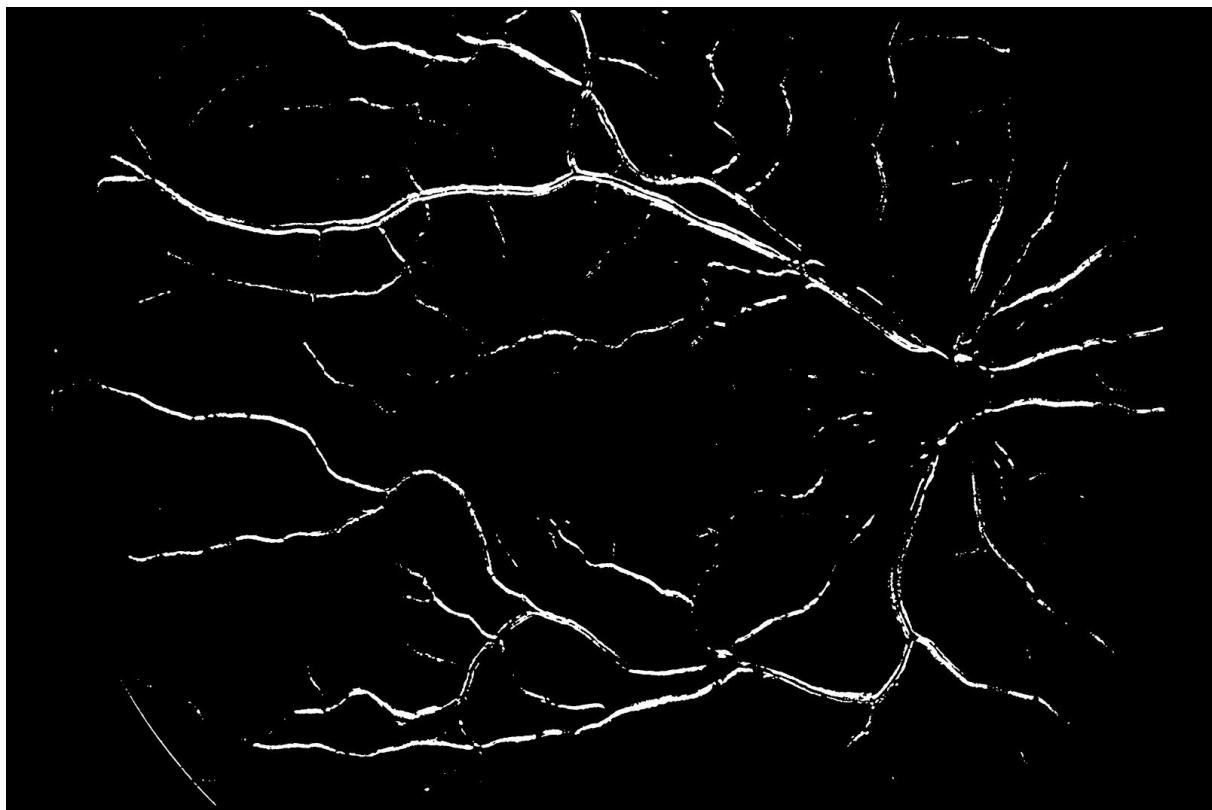
Rysunek 107: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 1 tys. elementów w zbiorze



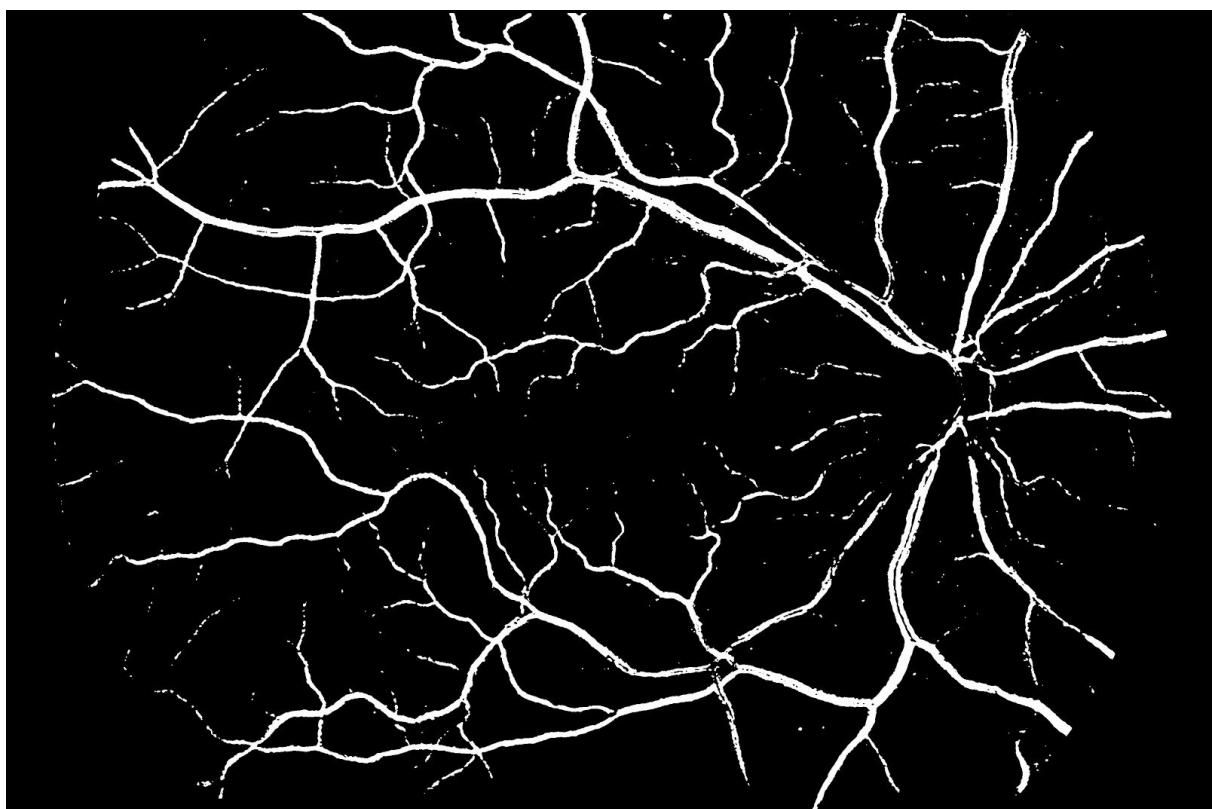
Rysunek 108: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 10 tys. elementów w zbiorze



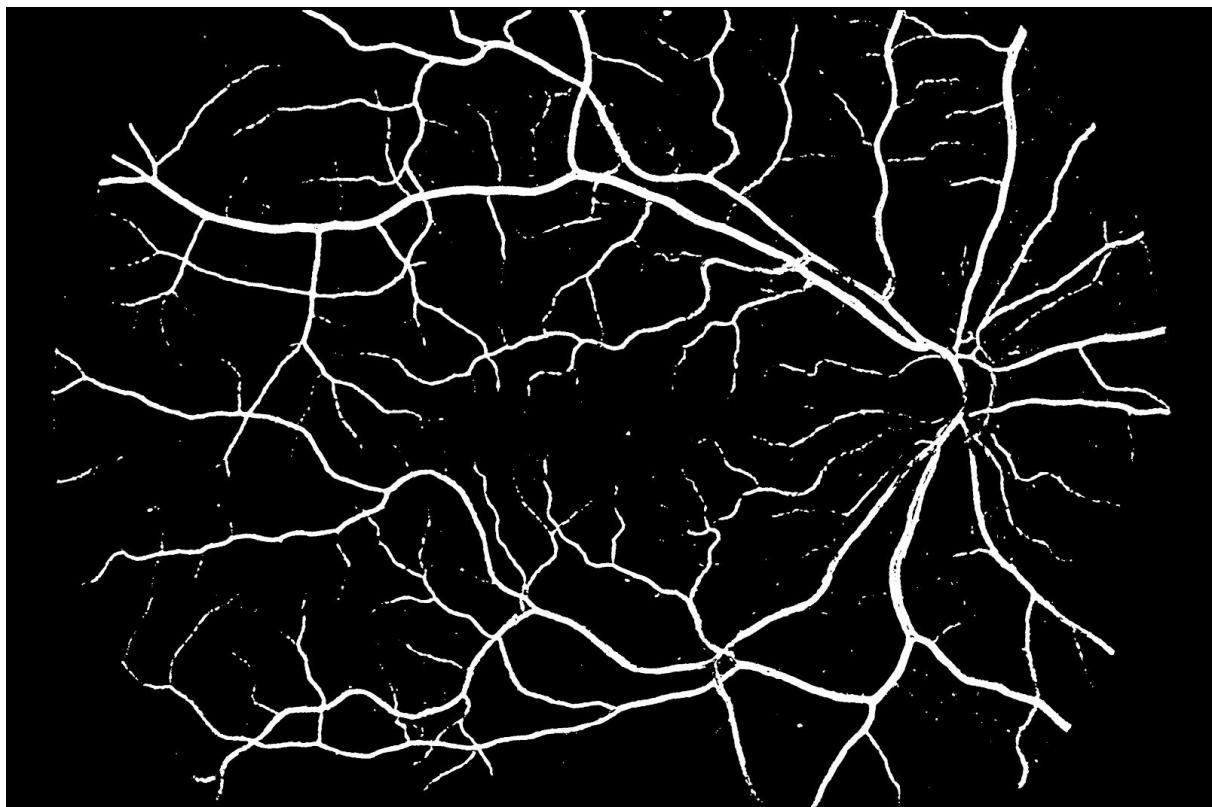
Rysunek 109: Przetwarzanie dla klasyfikatora RandomForest, maska 5x5 pikseli, 100 tys. elementów w zbiorze



Rysunek 110: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 1 tys. elementów w zbiorze



Rysunek 111: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 10 tys. elementów w zbiorze



Rysunek 112: Przetwarzanie dla klasyfikatora RandomForest, maska 21x21 pikseli, 100 tys. elementów w zbiorze