

# Wyznacznik pochodzenia człowieka

## Analiza obrazów

Krzysztof Żelazny  
Maksym Kazhaiev  
Darya Haidukevich

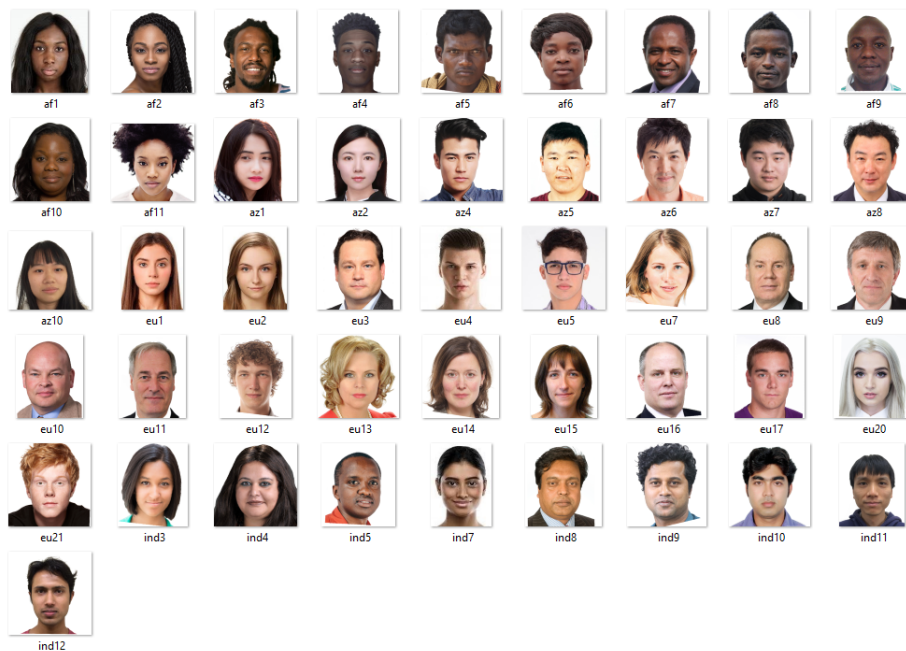
22 stycznia 2023

Program ma na celu ustalić etniczne pochodzenie na podstawie koloru skóry człowieka. Jako wynik otrzymujemy procentowe prawdopodobieństwo pochodzenia z czterech regionów, takich jak Europa, Afryka, Indie oraz Azja Wschodnia.

## 1 Dane

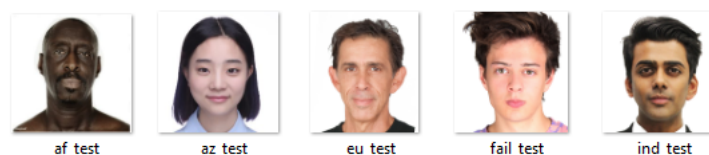
Wybraliśmy 51 zdjęcie twarzy ludzi na białym tle, aby tło nie zakłócało działania algorytmu. W programie korzystamy z sieci neuronowej, więc podzieliliśmy obrazy na dwie grupy: dane trenujące oraz testowe.

Zestaw danych trenujących składa się z 46 obrazów, których nazwy wskazują na pochodzenie osób na zdjęciach (Rysunek 1).



Rysunek 1: Zestaw danych trenujących

Również utworzyliśmy zestaw danych testowych, zawierający 5 obrazów. Nazwa każdego ze zdjęć pokazuje na oczekiwany przez nazwę wynik podczas testowania (Rysunek 2).



Rysunek 2: Zestaw danych testowych

## 2 Opis algorytmu

Głównym celem naszego algorytmu jest przygotowanie danych wejściowych do sieci neuronowej oraz jej trenowanie. Tym się zajmuję funkcja *nowyalg*.

Do przedstawienia obrazów testujących jako tablice wartości średnich kolorów służy funkcja *algorytm* (jest wykorzystana wewnątrz funkcji *nowyalg*), która składa się z kilku etapów:

1. przyjmuje jako argument nazwę obrazu,
2. odczytuje obraz za pomocą funkcji *imread* i konwertuje go na skalę szarości za pomocą funkcji *rgb2gray*, konwertując go na podwójną precyzję oraz dzieląc obraz przez 255 (aby wartości pikseli obrazu mieściły się w przedziale od 0 do 1),
3. tworzy binarne obrazy kanałów czerwonego, zielonego i niebieskiego oryginalnego obrazu przy użyciu funkcji *imbinarize* z progiem 0.95,
4. sumuje wszystkie białe piksele obrazów binarnych każdego z kanałów (liczba białych pikseli pokazuje, w jakim stopniu dany kolor przeważa na obrazie),
5. pobiera szerokość i wysokość obrazu za pomocą funkcji *size*,
6. oblicza średnie wartości kolorów kanałów czerwonego, zielonego i niebieskiego obrazu,
7. zwraca obliczone wartości jako 3-elementowy wektor.

Czyli po przetworzeniu obrazów mamy 46 wektorów, z których robimy cztery macierze o wymiarze *liczba obrazów z danego regionu*  $\times$  3, odpowiadające każdemu regionu. Jako dane wejściowe do sieci neuronowej przyjmujemy macierz składającą się z tych czterech transponowanych macierzy.

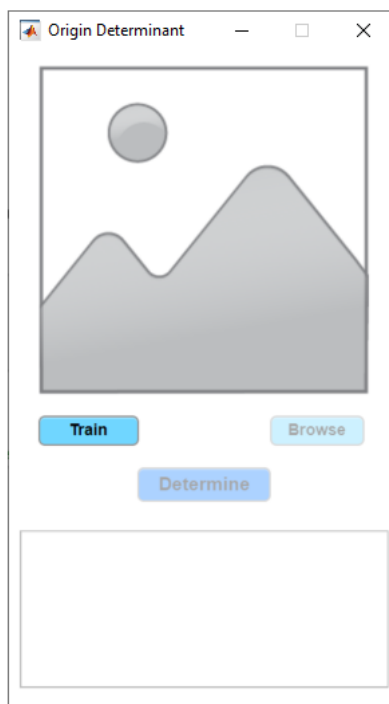
Jako dane wyjściowe przyjmujemy macierz o wymiarze  $3 \times 46$  składającą się z jedynek i zer. Taka macierz ma 18 kolumn  $[1, 0, 0, 0]$  (Europa), 10 kolumn  $[0, 1, 0, 0]$  (Afryka), 10 kolumn  $[0, 0, 1, 0]$  (Indie) oraz 8 kolumn  $[0, 0, 0, 1]$  (Azja Wschodnia). Każda kolumna reprezentuje idealny wynik obrazu po przejściu przez sieć neuronową.

Aby zdefiniować sieć neuronową, korzystamy z funkcji *feedforwardnet*. Trenowanie odbywa się za pomocą funkcji *train*, do której jako argumenty przekazujemy sieć neuronową, dane wejściowe oraz dane wyjściowe.

Funkcja *nowyalg* zwraca nauczoną sieć neuronową, która jest wykorzystana w graficznym interfejsie. Jeżeli przekazać do sieci 3-elementowy wektor reprezentujący pewien obraz, dostaniemy wektor o długości cztery, w którym każdy element pokazuje, w jakim stopniu dana osoba pochodzi z pewnego regionu. Im bliżej jest ta wartość do jedynki, im większe jest prawdopodobieństwo.

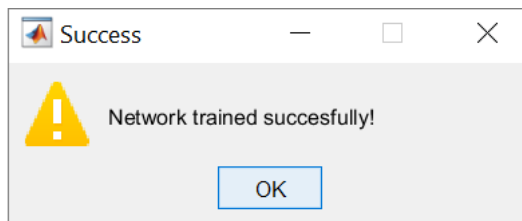
### 3 Graficzny interfejs

Aby uruchomić graficzny interfejs użytkownika, należy otworzyć w MATLABie plik *app1.mlapp* i uruchomić program. Od razu po uruchomieniu pojawi się okno przedstawione poniżej (Rysunek 3).



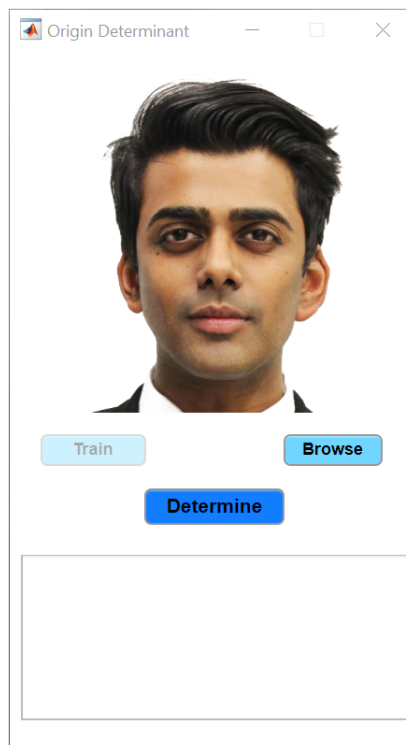
Rysunek 3: Aplikacja od razu po uruchomieniu

Po naciśnięciu jedynego odblokowanego przycisku "*Train*" się wywoła funkcja *nowyalg*, która przetwarza dane wejściowe i trenuje sieć neuronową. Po zakończeniu trenowania powinniśmy otrzymać komunikat o zakończeniu nauczania się (Rysunek 4).



Rysunek 4: Komunikat o zakończeniu trenowania

Po komunikacie "*Success*" odblokuje się przycisk "*Browse*", który daje możliwość przekazania obrazu do testowania. Jeżeli zdjęcie zostało wybrane, aplikacja odblokuje przycisk "*Determine*" (Rysunek 5).



Rysunek 5: Aplikacja po wprowadzeniu obrazu testowego

Wciśnięcie przycisku "*Determine*" wyświetla wyniki.

## 4 Wyniki

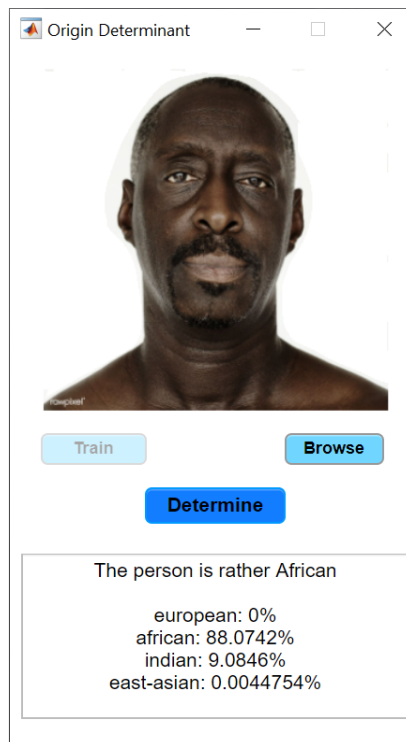
W większości przypadków aplikacja wyświetla poprawne wyniki dla czterech z pięciu obrazów testujących. Poprawnym liczy się wynik, kiedy prawdopodobieństwo pochodzenia z jednego regionu jest większe od 50% i zgadza się z oczekiwaniami, natomiast prawdopodobieństwo z innych regionów jest mniejsze od 50%.

Jednak możemy otrzymać niesatysfakcjonujące wyniki dwóch typów:

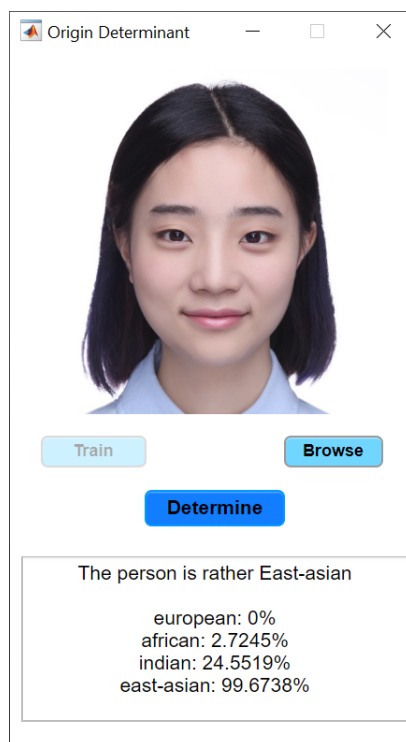
- trudno wyznaczyć pochodzenie człowieka (żadne prawdopodobieństwo nie przekracza 50% oraz kilka wyników są blisko tego progu),
- wynik nie zgadza się z oczekiwaniami (jest wyznaczony jeden region, jednak on się nie zgadza z rzeczywistością).

### 4.1 Sukces

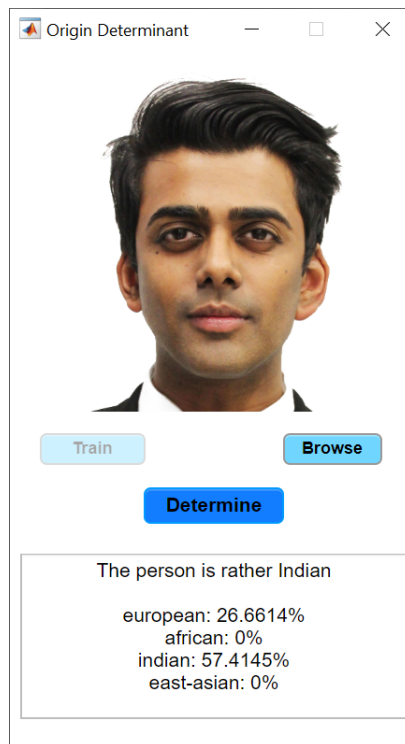
Poniżej są przedstawione sukcesywne wyniki dla czterech ludzi pochodzących z różnych regionów (Rysunek 6-9).



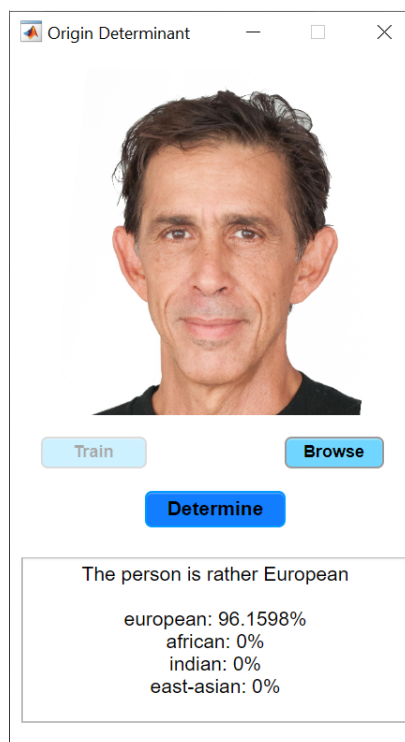
Rysunek 6: Sukcesywny wyniki dla osoby pochodzącej z Afryki



Rysunek 7: Sukcesywny wyniki dla osoby pochodzącej z Azji Wschodniej



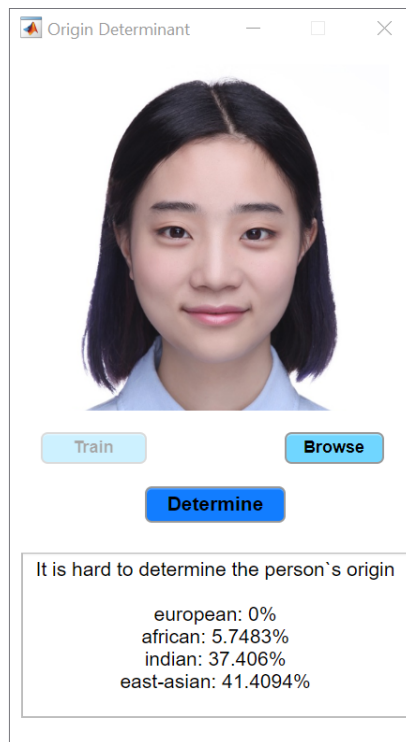
Rysunek 8: Sukcesywne wyniki dla osoby pochodzącej z Indii



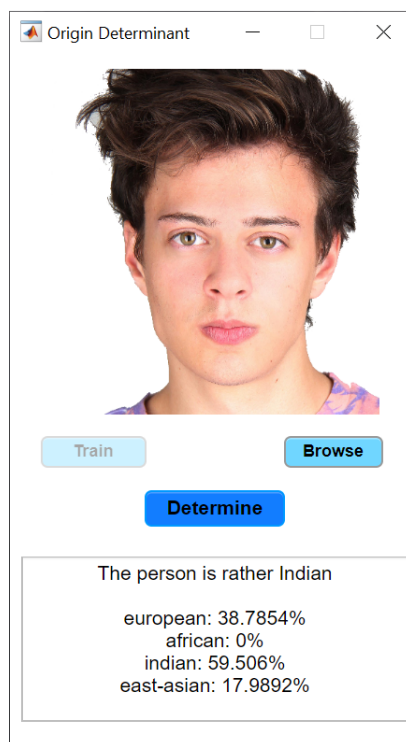
Rysunek 9: Sukcesywne wyniki dla osoby pochodzącej z Europy

## 4.2 Niepowodzenie

Poniżej są przedstawione wyniki, których nie spodziewaliśmy się (Rysunek 10-11).



Rysunek 10: Aplikacja nie jest w stanie określić pochodzenie człowieka



Rysunek 11: Aplikacja wyświetla niepoprawne wyniki

## 5 Dodatkowa informacja

### 5.1 Podział pracy

- Algorytm: Krzysztof Żelazny

- Interfejs graficzny: Maksym Kazhaiev
- Dane & dokumentacja: Darya Haidukevich

## 5.2 Link do kodu

<https://github.com/dar144/Analiza-Obrazow>