

AGH

Akademia Górniczo-Hutnicza
im. Stanisława Staszica
w Krakowie

Analiza Obrazów

Dokumentacja projektu

Aplikacja do wykrywania znaków drogowych
i określania ich typu

Bartosz Mikołajczyk, Piotr Matiaszewski, Jakub Perlak

Wydział Fizyki i Informatyki Stosowanej

2020 / 2021

Opis i założenia projektu

Celem naszego projektu było napisanie aplikacji, która jest w stanie wykrywać znaki drogowe i przypisywać je do jednej z kategorii znaków – informacyjnych, nakazu, zakazu lub ostrzegawczych – na podstawie przesłanego jej zdjęcia.

Założeniem projektu było wykorzystanie sieci neuronowej w celu rozpoznawania odpowiednich kształtów. Dzięki tej informacji oraz znajomości koloru obiektu można jednoznacznie zakwalifikować dany obiekt jako znak przynależący do jednej z wcześniej określonych kategorii.

Narzędzia wykorzystane do realizacji projektu

Program został napisany w całości przy użyciu oprogramowania MATLAB®. W projekcie użyto również dodatków Deep Learning Toolbox oraz Image Processing Toolbox, które umożliwiły operacje na obrazach oraz implementację sieci neuronowej.

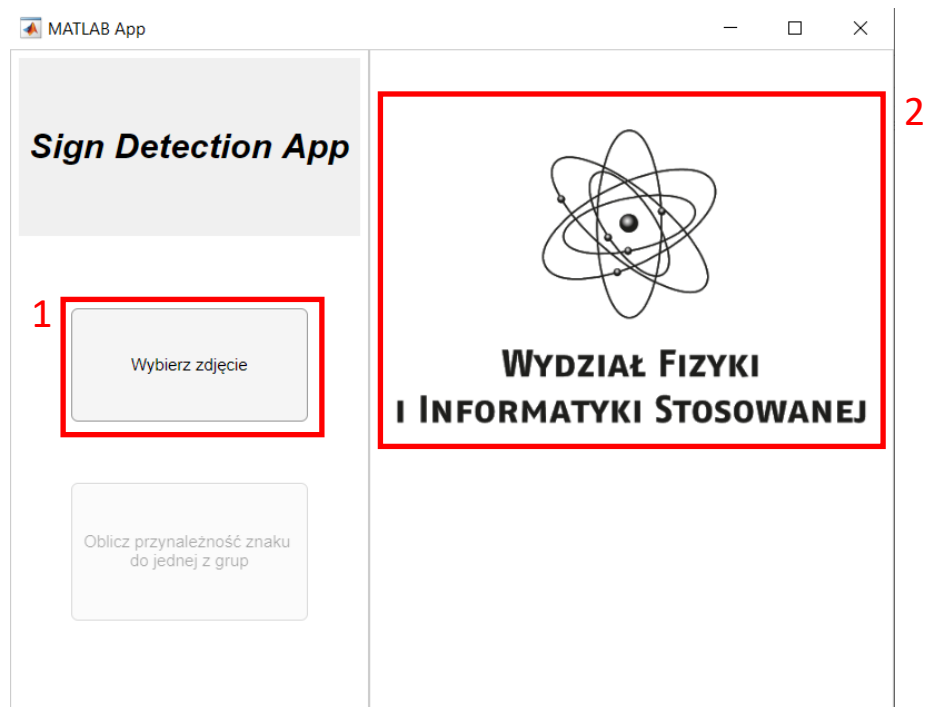
Do zaprojektowania interfejsu użytkownika użyto narzędzia MATLAB App Designer.

Podział pracy w zespole

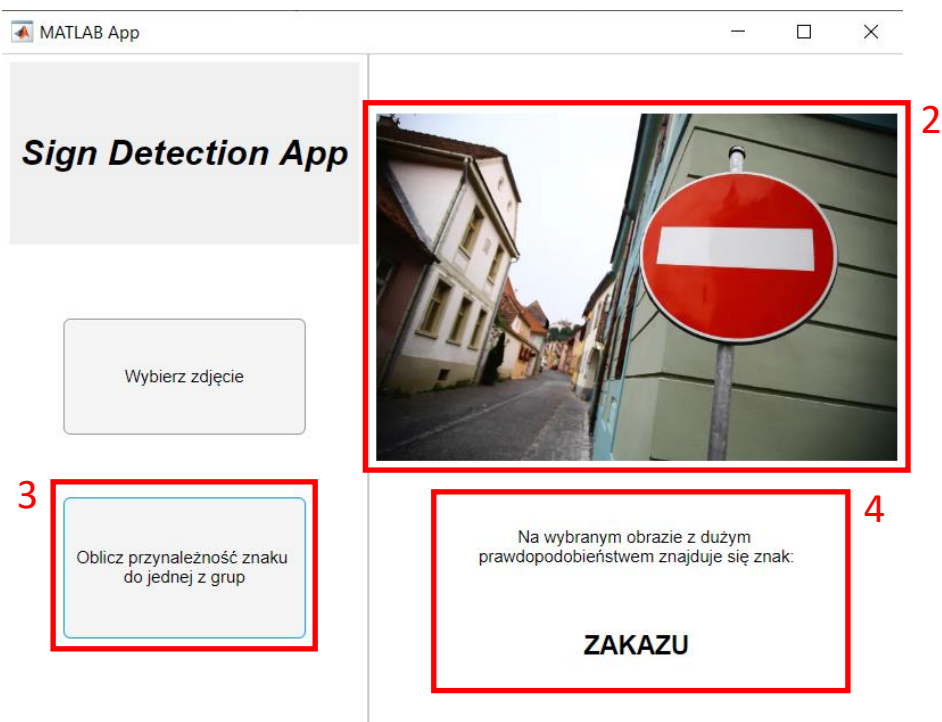
- Bartosz Mikołajczyk – preprocessing danych (wyodrębnienie obiektów z obrazu), analiza wyników zwracanych przez sieć neuronową.
- Piotr Matiaszewski – opracowanie współczynników geometrycznych, interfejs użytkownika.
- Jakub Perlak – przygotowanie danych uczących i testowych, obsługa sieci neuronowej.

Należy zaznaczyć, że powyższe przypisanie zadań studentom to jedynie określenie osoby nadzorującej dany segment projektu. Sama realizacja zadań była przeprowadzana grupowo.

Interfejs użytkownika



Obraz 1. Wygląd aplikacji przed wczytaniem obrazu (po uruchomieniu programu).

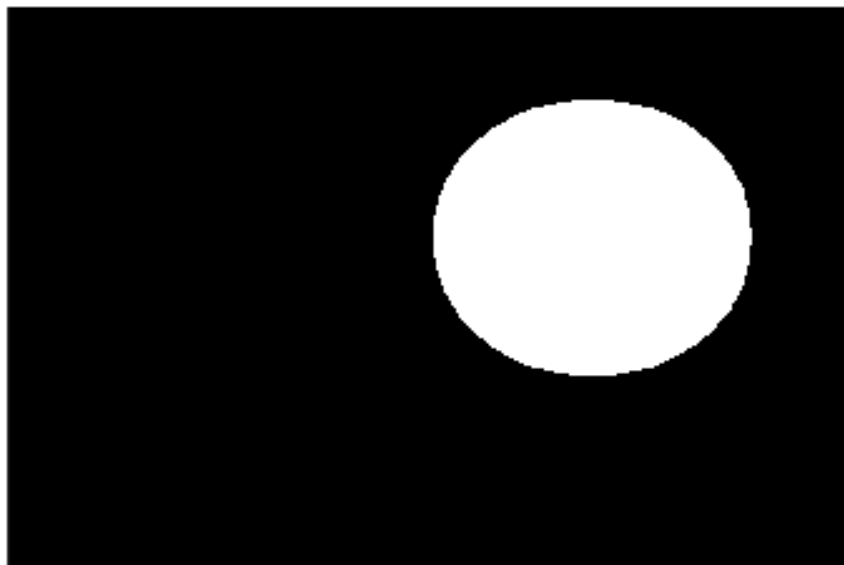


Obraz 2. Wygląd aplikacji po wczytaniu obrazu.

1. Przycisk do obsługi danych wejściowych. Po jego kliknięciu wywołuje się menu wyboru pliku w formacie .jpg lub .png.
2. Na obrazie 1. widoczny jest obszar, w którym pojawi się wczytane przez użytkownika zdjęcie. Domyślnie jest to logo WFiIS. Na obrazie 2. widoczny jest obraz wczytany przez użytkownika, który będzie poddawany analizie.
3. Przycisk uruchamiający program.
4. Wynik działania programu.

Opis działania programu

1. Preprocessing przekazanego obrazu (wyodrębnienie obiektów).
 - Binaryzacja obrazu za pomocą maski utworzonej poprzez rozkład obrazu na kanały RGB:
 - maski są tworzone dla znaków typowych dla każdej z czterech kategorii znaków drogowych rozważanych w projekcie. W związku z tym, że obiekty typowe dla każdej kategorii mają niebieskie (znaki informacyjne i nakazu) lub czerwone (znaki ostrzegawcze i zakazu) elementy, to zaimplementowane w projekcie maski obejmują takie kolory.
 - Redukcja szumu i pozbycie się małych obiektów (ze względu na rozmiar i stosunek osi obiektu).
 - Domknięcie oraz wypełnienie obiektów.



Przykład 1. Obiekt z obrazu 2. po preprocessingu.

2. Wyznaczenie kryterium klasyfikacji podziału obiektów na grupy pod kątem kształtu (prostokąty, koła i trójkąty).

- Wyznaczenie współczynników geometrycznych dla każdego z wyznaczonych obiektów.
- Dodanie współczynnika kolorystycznego do danych opisujących każdy obiekt.

```
test_coeffs =  
  
    9.8281  -0.0927  0.8232  0.9966  28.5554
```

Przykład 2. Współczynniki geometryczne obiektu z obrazu 2.

3. Sieć neuronowa.

- Załadowanie wcześniej wyuczonej sieci neuronowej.
- Przekazanie sieci neuronowej współczynników geometrycznych obiektu.

```
result =  
  
    0.0000  
    1.0000  
   -0.0000  
    0
```

Przykład 3. Wynik sieci neuronowej (znaleziono obiekt w kształcie koła) z dodanym współczynnikiem kolorystycznym.

4. Interpretacja wyników

- Wyniki sieci neuronowej, wraz z wcześniej przygotowanymi współczynnikami kolorystycznymi, porównywane są z rzeczywistą formą znaków (kształtem i kolorem).
- Zinterpretowany wynik przekazywany jest do interfejsu użytkownika w postaci odpowiedniego komunikatu.

Potencjalne błędy

1. W programie przeprowadzane jest odsumianie obrazu oraz usuwanie małych obiektów. Jednakże, niektóre obiekty, które nie zostały wstępnie usunięte, mogą zostać zinterpretowane jako potencjalne znaki.

2. Ze względu na małą liczbę obiektów w zbiorze uczącym i ich mocne podobieństwo do siebie (w obrębie kształtu), sieć neuronowa może źle interpretować obiekty na przekazanym przez użytkownika obrazie.

Potencjał rozwoju

1. Do odpowiedniego wytrenowania sieci neuronowej potrzebna jest spora liczba obiektów uczących. W związku z tym, można rozważyć powiększenie bazy obrazów uczących w celu zwiększenia dokładności klasyfikacji obiektów przez sieć neuronową.
2. Dobrym pomysłem jest również poprawa obsługi szumu i małych, zbędnych obiektów, które w nielicznych przypadkach mogą zostać nadmiarowo sklasyfikowane.
3. W projekcie jest przeprowadzany rozkład obrazu na kanały RGB. Można spróbować rozważyć rozkład obrazu na kanały HSL, aby uzyskać mniejszą wrażliwość klasyfikacji na jasność obrazu.