

# Wykrywanie tablic rejestracyjnych przy pomocy algorytmów głębokiego uczenia

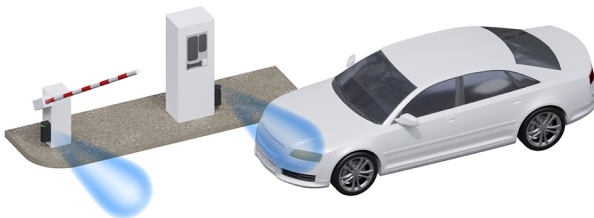
## Warsztaty z technik uczenia maszynowego

Jakub Dmitruk i Julita Kulesza

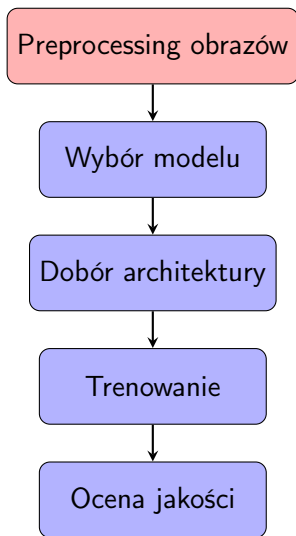
Wydział Matematyki i Nauk Informatycznych

07.06.2024

Celem naszego projektu było stworzenie systemu zdolnego do **rozpoznawania prostokątnych tablic** na różnorodnych zdjęciach aut. Łączył on w sobie techniki przetwarzania obrazu z zaawansowanymi metodami uczenia maszynowego, aby zapewnić dobrą dokładność detekcji.



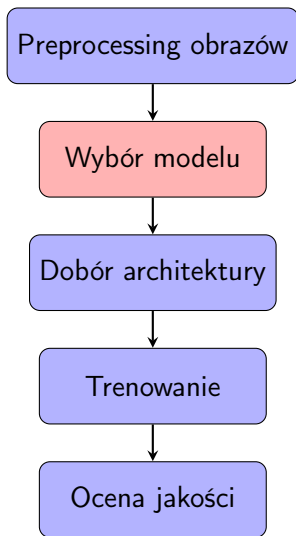
źródło:  
[https://www.optex.co.jp/e/products/vehicle-detection/app\\_license-plate-recognition/](https://www.optex.co.jp/e/products/vehicle-detection/app_license-plate-recognition/)



Przeskalowaliśmy każde zdjęcie do rozmiaru  $224 \times 224$  pikseli z wygładzeniem przy pomocy filtra `Image.LANCZOS`, aby ujednolicić rozmiary zdjęć. Oczywiście, konieczne było też dostosowanie etykiet położenia tablic do nowych rozmiarów zdjęć.

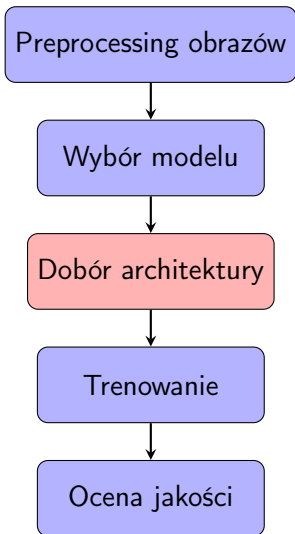
Następnie zdjęcia przekształciliśmy do formatu `numpy.array`, uzupełniliśmy o szum gaussowski i znormalizowaliśmy piksele do zakresu  $[0, 1]$ .

Aby zwiększyć objętość zbioru testowego, uzupełniliśmy go o odbicia lustrzane każdego z elementów.

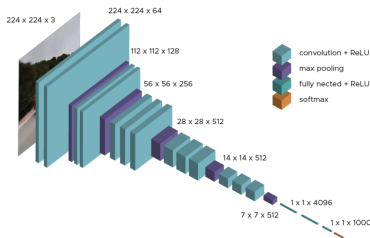


Używając podejścia *transfer learning*, zdecydowaliśmy się wykorzystać wcześniej wytrenowany model VGG16 do zadania klasyfikacji obrazów. Miało to pozwolić na skrócenie czasu trenowania.

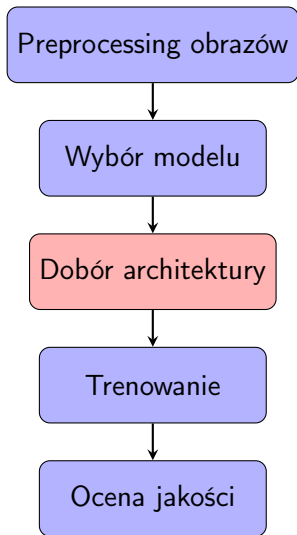
Punktem odniesienia była zbudowana przez nas sieć neuronową. Cel? Lepsze zrozumienie i kontrola architektury naszego modelu.



**Model 1:** VGG16 to sieć neuronowa wytrenowana na ogromnym zbiorze danych ImageNet. Wzbogaciliśmy go o dwie dodatkowe warstwy, warstwę w pełni połączoną z funkcją aktywacji *relu* oraz warstwę wyjściową.

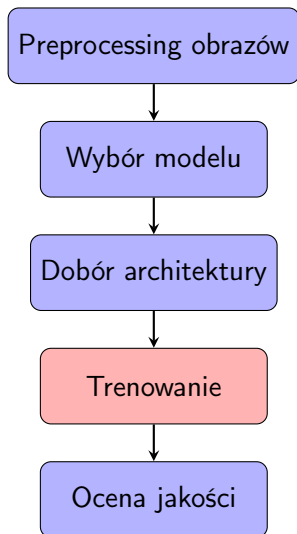


Rysunek: Architektura modelu VGG16

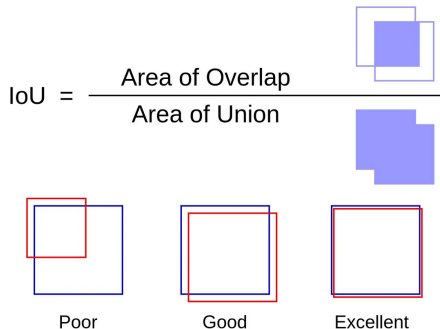


**Model 2:** Nasza sieć neuronowa składała się z różnego typu warstw:

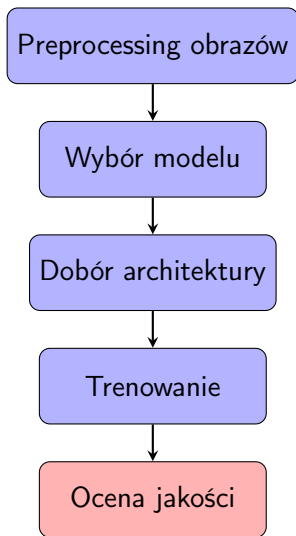
- za pomocą warstwy `BatchNormalization` normalizowaliśmy wejścia do warstw. Zazwyczaj czyni to algorytm stabilniejszym;
- warstwa `MaxPooling2D` pozwoliła nam na redukcję rozmiaru danych w procesie;
- warstwa `Dropout` wyłączała losowo część neuronów, regularyzując model.



Podczas trenowania modeli użyliśmy metryki IoU (ang. *Intersection over Union*) do oceny jakości przewidywań.



Rysunek: Definicja metryki IoU

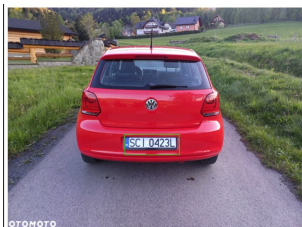


Ocena działania algorytmu była stosunkowo prosta i szybka, ponieważ wystarczyło zastosować go do predykcji na kilku obrazach i wizualnie ocenić wyniki.

Dzięki temu można było łatwo zidentyfikować ewentualne problemy i ewentualnie dokonać niezbędnych korekt (już nie tak łatwo).



Poniżej prezentujemy wyniki działania lepszego z algorytmów, czyli tego opartego na metodzie *transfer learning*.



- Uzyskane wyniki na zbiorze testowym były przyzwoite (rejestracja zazwyczaj znajdowała się w zaznaczonym zakresie).

- Uzyskane wyniki na zbiorze testowym były przyzwoite (rejestracja zazwyczaj znajdowała się w zaznaczonym zakresie).
- Średnie IoU w procesie walidacji wynosiło 0.48, istnieje więc znaczne pole do poprawy dokładności.

- Uzyskane wyniki na zbiorze testowym były przyzwoite (rejestracja zazwyczaj znajdowała się w zaznaczonym zakresie).
- Średnie IoU w procesie walidacji wynosiło 0.48, istnieje więc znaczne pole do poprawy dokładności.
- Możliwe dalsze kierunki rozwoju: zmiana architektury modelu, optymalizacja hiperparametrów, czy zwiększenie rozmiaru zbioru treningowego.
- Gotowe modele, takie jak VGG16, zapewniają solidne wyniki już na starcie i są trudne do zrównania pod względem skuteczności.

- Uzyskane wyniki na zbiorze testowym były przyzwoite (rejestracja zazwyczaj znajdowała się w zaznaczonym zakresie).
- Średnie IoU w procesie walidacji wynosiło 0.48, istnieje więc znaczne pole do poprawy dokładności.
- Możliwe dalsze kierunki rozwoju: zmiana architektury modelu, optymalizacja hiperparametrów, czy zwiększenie rozmiaru zbioru treningowego.
- Gotowe modele, takie jak VGG16, zapewniają solidne wyniki już na starcie i są trudne do zrównania pod względem skuteczności.
- Zadanie detekcji obiektów jest trudne, czasochłonne i wymaga precyzyjnej znajomości różnych architektur.

# Wykrywanie tablic rejestracyjnych przy pomocy algorytmów głębokiego uczenia

## Warsztaty z technik uczenia maszynowego

Jakub Dmitruk i Julita Kulesza

Wydział Matematyki i Nauk Informacyjnych

07.06.2024