

**Akademia Górniczo-Hutnicza im. Stanisława Staszica
Wydział Automatyki, Elektroniki Informatyki i Inżynierii
Biomedycznej**



Automatyczna segmentacja nerek

**Projekt realizowany w ramach przedmiotu Techniki Obrazowania
Medycznego**

**Kuba Chrobociński, Adrian Bieliński, Agata Machałek
Kraków, 8 czerwca 2020**

1 Zaimplementowane rozwiązanie

Pomysł na rozwiązanie problemu klasyfikacji nerek i guza z obrazu tomografii komputerowej zakłada zaimplementowanie trzech głównych etapów: preprocessingu, treningu modelu SVM na zbiorze treningowym oraz walidacji na zbiorze testowym.

1. Preprocessing - dla każdego przekroju kolejnych case'ów tworzymy maskę binarną za pomocą, której wytniemy część obrazu. Najpierw tworzymy kopię oryginalnego obrazu, przeprowadzamy adaptacyjne wyrównanie histogramu, przepuszczamy przez filtr medianowy, dokonujemy poprawy kontrastu poprzez funkcję gamma. Po tych procedurach dokonujemy pierwszej operacji wielokrotnego progowania metodą Otsu, wycinamy z oryginalnego obrazu obszar nas interesujący, następnie przeprowadzamy ponownie operację wyrównania histogramu oraz wielokrotnego progowania. Tak otrzymaną maskę poddajemy zamknięciu oraz usuwamy małe czarne punkty. Wyrównujemy histogram danych oryginalny, a następnie wycinamy interesujący nas obszar za pomocą otrzymanej maski. Przed przekazaniem danych w celu uczenia modelu należy również przekształcić je w formę jaką model może przyjąć, tzn. konwertujemy trójwymiarową tablicę do dwuwymiarowej tablicy której wiersze są kolejnymi punktami, a w kolumnach znajdują się opisujące je cechy(zmienne objaśniające): współrzędne x,y oraz z, a także wartość tej cechy. W związku z różnymi wielkościami planowane jest skalowanie danych.
2. Trening SVM - początkowo problemem było trenowanie modelu ze względu na ograniczoną ilość pamięci RAM. Dzięki zastosowaniu wersji algorytmu wykorzystującego metodę najszybszego spadku możliwe jest uczenie modelu za pomocą mniejszych porcji danych, co pozwala na wykorzystanie całego zbioru. Przeprowadzone zostało póki co w celu sprawdzenia działania modelu trenowanie go na dwóch case'ach. Planowane jest dalsze optymalizowanie parametrów modelu oraz zwiększenie ilości danych podanej do trenowania modelu oraz poprawa wydajności skryptu.
3. Ewaluacja - wykonana jest przy pomocy funkcji z biblioteki „sklearn.linear_model” score - która zwraca średnią dokładność (*accuracy*) testowanych danych i nadanych im etykiet. Wyznaczony zostanie również indeks Jaccarda oraz wynik F1.