

Raport - wykrywanie naczyń dna siatkówki oka

1 Skład grupy

- Zuzanna Piniarska 136782
- Mateusz Kałamoniak 136730

2 Zastosowany język programowania i biblioteki

Język programowania: **Python**

Dodatkowo zastosowane biblioteki

- tensorflow
- numpy
- PIL
- matplotlib
- datetime
- IPython

3 Opis zastosowanych metod

W implementacji posłużyliśmy się w głównej mierze biblioteką Tensorflow. Posłużyła ona do wstępnego przetwarzania obrazu jak i tworzenia sieci neuronowej. Zastosowaliśmy techniki augmentacji takie jak modyfikacja saturacji, kontrastu czy odcienia. Wszystkim z tych funkcji podaliśmy losowe wartości w określonych przedziałach. Dodatkowo obracaliśmy obraz. Z użyciem tak przetworzonych obrazów wytrenowaliśmy sieć neuronową. Zaimplementowaliśmy sieć będącą modyfikacją U-Net i MobileNetV2. Wykorzystaliśmy architekturę z U-Net, natomiast encoder został zaimplementowany

na wzór MobileNetV2. Do optymalizacji zastosowaliśmy algorytm Adam z współczynnikiem uczenia 0.001. Funkcja kosztu składa się z sumy indeksu Jaccarda z wagą 0.3 oraz binary cross entropy z wagą 0.7. Przy tworzeniu sieci sugerowaliśmy się przede wszystkim artykułem M2U-Net: Effective and Efficient Retinal Vessel Segmentation for Real-World Applications (<https://arxiv.org/pdf/1811.07738.pdf>).

Uzupełnione informacje

Nasza sieć uczyła się przez około 3 godziny, uczyła się na 200 epokach. Ze względu na ilość obrazów wejściowych podzieliliśmy nasz zbiór na 15 obrazów treningowych i 30 obrazów walidujących. Obrazy pochodziły ze zbioru HRF.

Wykorzystaliśmy augmentację 15 obrazów składającą się ze zmiany saturacji, kontrastu i odcienia oraz orientacji.

Rozmiary zdjęć są zmienione na 544x544. Pomijając powyższe zmiany resztę algorytmu staraliśmy się zaimplementować sugerując się pracą naukową.

4 Wyniki parametrów przetwarzania

Name	Accuracy	Precision	Sensitivity	Specificity	F1 score
01_h.jpg	0,9514	0,8387	0,6479	0,9858	0,7311
12_dr.jpg	0,9583	0,7706	0,5526	0,9880	0,6436
14_h.jpg	0,9576	0,8039	0,6797	0,9842	0,7366
13_h.jpg	0,9577	0,8107	0,6707	0,9851	0,7341
10_g.jpg	0,9624	0,7173	0,7056	0,9804	0,7114
01_dr.jpg	0,9574	0,5605	0,7552	0,9683	0,6435
03_h.jpg	0,9450	0,8120	0,6247	0,9829	0,7061
15_dr.jpg	0,9576	0,6767	0,6738	0,9775	0,6753
11_h.jpg	0,9587	0,8296	0,7021	0,9851	0,7605
04_g.jpg	0,9630	0,7082	0,7285	0,9792	0,7182
15_h.jpg	0,9631	0,8529	0,6394	0,9906	0,7309
09_h.jpg	0,9599	0,7712	0,6876	0,9828	0,7270
09_dr.jpg	0,9558	0,7361	0,5430	0,9858	0,6250
07_h.jpg	0,9600	0,7961	0,7407	0,9814	0,7674
04_dr.jpg	0,9666	0,7253	0,6588	0,9850	0,6905
13_g.jpg	0,9561	0,6746	0,6856	0,9759	0,6801
05_dr.jpg	0,9576	0,6272	0,7891	0,9688	0,6989
10_dr.jpg	0,9488	0,7036	0,7102	0,9715	0,7069
05_g.jpg	0,9653	0,7313	0,7440	0,9808	0,7376
02_dr.jpg	0,9571	0,6476	0,7219	0,9731	0,6827
08_g.jpg	0,9603	0,6889	0,7587	0,9750	0,7221
01_g.jpg	0,9558	0,6471	0,7507	0,9706	0,6950
13_dr.jpg	0,9618	0,8079	0,5762	0,9900	0,6727
02_g.jpg	0,9566	0,6945	0,7396	0,9739	0,7164
04_h.jpg	0,9539	0,7793	0,7109	0,9791	0,7435
12_h.jpg	0,9558	0,8413	0,7119	0,9843	0,7712
15_g.jpg	0,9552	0,6791	0,7296	0,9729	0,7035
07_g.jpg	0,9632	0,7111	0,7339	0,9792	0,7223
14_dr.jpg	0,9547	0,7522	0,6057	0,9835	0,6710
11_g.jpg	0,9578	0,7207	0,7336	0,9764	0,7271

5 Wizualizacja i analiza wyników działania programu

W poniżej przedstawionych wynikach po lewej stronie znajduje się oryginalne zdjęcie, po środku predykcja, a po prawej maska.





