

Laboratorium 4

Segmentacja obrazów – część 2

Zadanie 1 – *Random Walker*

Wczytać obraz *lung_lesions.bmp*. Dokonać segmentacji obrazu metodą Random Walker (`skimage.segmentation.random_walker`) w oparciu o zestaw pikseli źródłowych wskazanych kolejno w plikach *lungs_lesion_seeds1.bmp*, *lungs_lesion_seeds2.bmp* oraz *lungs_lesion_seeds3.bmp*. Zbadać wpływ parametru *beta* na wynik działania metody. Przetestować zakres *beta* od 0 do 100000 z wybranym krokiem. Każdorazowo, wyświetlić wynik w trzech panelach jak poniżej.



Zadanie 2 – *Random Walker 2*

Korzystając z metody Random Walker dokonać segmentacji czerwonego jabłka z obrazu *apples.jpg*. Zaproponować automatyczny sposób wyboru pikseli źródłowych dla tła i obiektu.



Przydatne funkcje: `skimage.segmentation.random_walker`, `cv2.threshold`, `cv2.THRESH_BINARY`, `numpy.nonzero`

Zadanie 3 – *Aktywny kontur*

Wczytać obraz *cat.jpg*. Dokonać segmentacji obrazu metodą aktywnego konturu (`skimage.segmentation.active_contour`) kolejno motyla i kota (por. rys. poniżej, gdzie czerwona przerywana linia oznacza kontur początkowy, a niebieska ciągła linia – wynik segmentacji). Zbadać podatność metody na inicjalizację oraz wpływ parametrów wpływających na energię konturu na końcowy wynik segmentacji. Sprawdzić możliwość dokładnego dopasowania konturu do kształtu kota oraz jednoczesnej segmentacji obu

obiektów. Przetestować metodę na innych obrazach, w szczególności *objects1.jpg*, *objects2.jpg* oraz *objects3.jpg*.

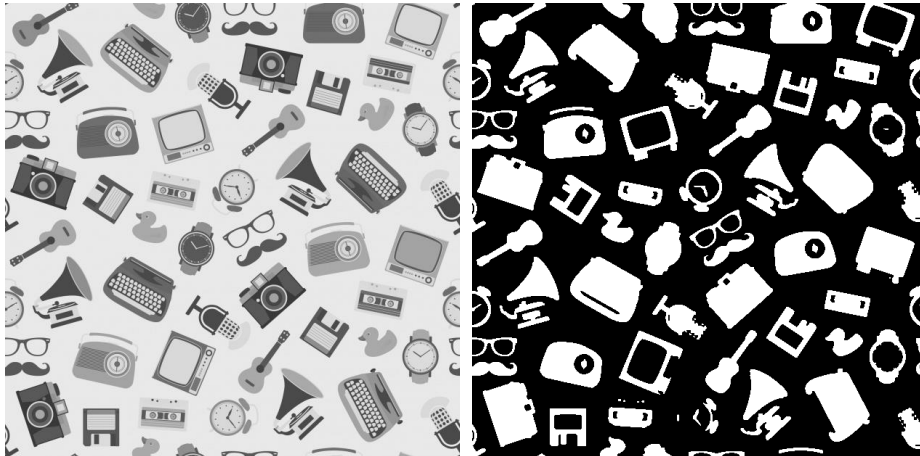


Przykład, jak wygenerować kontur początkowy dostępny jest pod adresem:

http://scikit-image.org/docs/dev/auto_examples/edges/plot_active_contours.html

Zadanie 4 – Metoda zbiorów poziomicowych

Wykonać segmentację obrazów *objects1.jpg*, *objects2.jpg* oraz *objects3.jpg* z wykorzystaniem metody zbiorów poziomicowych, na przykład metody Chan-Vese (`skimage.segmentation.chan_vede`). Wcześniej należy przekonwertować obrazy wejściowe do skali szarości. Przetestować wpływ liczby iteracji na końcowy wynik segmentacji. Dodać do obrazu szum (`numpy.random.normal`, `numpy.random.poisson`, `numpy.random.randn`) oraz powtórzyć eksperyment. Wyznaczyć eksperymentalnie maksymalny poziom szumu, dla którego możliwa jest poprawna segmentacja obiektów. Przykładowy obraz oraz wynik segmentacji przedstawia rysunek poniżej.



Obrazy do testów dostępne pod adresem: <http://an-fab.iis.p.lodz.pl/PSIO>