

Sprawozdanie z projektu 1 z analizy obrazów

Miałem wykonać 4 przekształcenia:

- Normalizacja w zadanych 3 punktach x,y
- Filtracja odchyleniem std + normalizacja
- Zamknięcie obiektem kołowym
- Efekt wypukłego otoczenia

Normalizacja:

Pętla przechodzi przez każdy pixel i sprawdza pixel do którego przedziału się kwalifikuje czyli x_1, x_2, x_3 . Po wybraniu przedziału mnożymy odpowiednim mnożnikiem. Wykonuje się to dla każdego R, G i B.

Wejście:



Normalizacja(1, 1, 100, 200, 255, 255);



Filtracja odchyleniem std + normalizacja:

Na start zmieniam rgb do chromo, a następnie wykonuje pętle w której dla każdego pixela przelatuje kolejną pętlę dla całej maski. Zliczam elementy, obliczam odchylenie standardowe i zapisuje do pixela. Na końcu normalizuje wg wzoru:

The linear normalization of a grayscale digital image is performed according to the formula

$$I_N = (I - \text{Min}) \frac{\text{newMax} - \text{newMin}}{\text{Max} - \text{Min}} + \text{newMin}$$

Przykład:

Wejscie:



Mono:



Filtr + normalizacja:

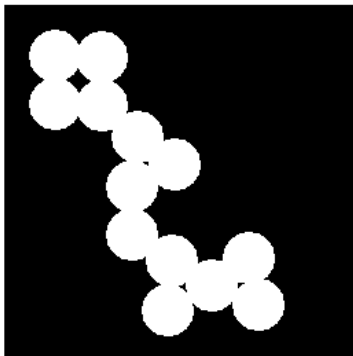


Zamknięcie obiektem kołowym:

Algorytm tworzy macierz dynamiczna z białym kołem w środku macierzy. Na start wykonuje dylatację. Dla każdego spotkanego białego pixela wykonuje zmienienie pixeli w otoczeniu jego dokładnie z macierzy pomocniczej. Wykonuje kopie obrazu z dylatacją. Następnie wykonuje erozję. Algorytm poszukuje sprzeczności w otoczeniu pixela porównując ją z pomocniczą macierzą. Gdy jest kolizja po porównaniu następuje zamiana pixela na przeciwny czyli czarny.

Przykład:

Wejście:



Zamknięcie o promieniu 5pix:



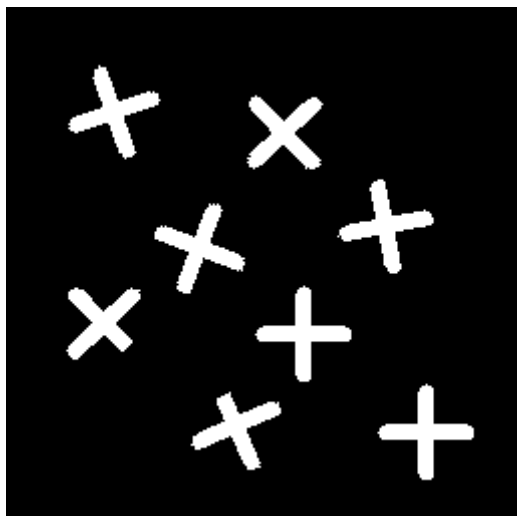
Efekt wypukłego otoczenia:

Algorytm wykorzystuje 8 pomocniczych macierzy.

Algorytm poszukuje kolizji w obrębie pixeli i gdy taka następuje to zmieniamy pixel na przeciwny. Wynik czyli kilka białych pixeli dodajemy do oryginalnego obrazka. Tak wykonujemy dla każdej maski i powtarzamy algorytm do uzyskania zbieżności.

Przykład:

Wejsice:



Wynik:

