

Rozciąganie / wyrównywanie histogramu

BW obraz $g(i, j) \in \{0, \dots, 255\}$. Rozciąganie histogramu:

$$g_2(i, j) = \frac{g(i, j) - \min}{\max - \min}, \quad \text{gdzie } \min = \min_{i,j} g(i, j), \max = \max_{i,j} g(i, j).$$

Wyrównywanie/dopasowywanie histogramu. Załóżmy, że obraz g ma histogram $p(k)$, $k = 0, \dots, 255$ oraz “dystrubuantę” $G(n) = \sum_{k=0}^n p_2(k)$, podobnie obraz h ma histogram $q(k)$ oraz “dystrubuantę” $T(k)$. Zamieniamy każdy pixel obrazka g o wartości r na wartość s , tak, aby

$$G(s) = T(r), \quad \text{czyli } s = G^{-1}(T(r)).$$

(w przypadku, gdy $T(r)$ odpowiada rozkładowi jednostajnemu mówimy o wyrównywaniu histogramu.

1. Dla wybranych obrazków z katalogu http://www.math.uni.wroc.pl/~lorek/image_analysis/images/hist/ zastosuj wyrównywanie histogramu i/lub dopasowanie. Narysuj też histogramy przed i po przekształceniach.
2. *Filtr sobela.* Zastosuj do wybranych obrazów filt Sobela. Wyświetl np. $M(i, j) = \sqrt{S_x^2(i, j) + S_y^2(i, j)}$. Poeksperymentuj na różnych obrazkach. Potestuj późniejsze progowanie z różnymi progami (cel: wykrywanie krawędzi). Przykład jak można stosować ten filtr w Pythonie:

```
from scipy import signal

def sobel_filt(img):

    fil_y = np.array([[ -1,  0,  1], [-2,  0,  2], [-1,  0,  1]], dtype = np.float)
    fil_x = np.array([[ 1,  2,  1], [ 0,  0,  0], [-1, -2, -1]], dtype = np.float)

    dx = signal.convolve2d(img, fil_y)
    dy = signal.convolve2d(img, fil_x)

    sob = np.sqrt(dx**2 + dy**2)

    #Zastosowanie filtru Sobela:
    im_sob= sobel_filt(img)
```

3. *Transformata fouriera*. Przykład pokazujący jak można wyliczyć transformatę fouriera w Pythonie

```
from scipy import fftpack

#Policzenie fft (wynik: macierz liczb zespolonych)
img_fft = fftpack.fft2(img)

#transformata odwrotna (dla pewnosci bierzemy czesc rzeczywista)
img_recovered = fftpack.ifft2(img_fft2).real
```

Uwaga: Wyświetaląc transf. fouriera (lub jakąś jej funkcję) najlepiej jeszcze zamienić odpowiednie “ćwiartki”, tak, by punkty (założmy, że obrazek ma rozmiar (h, w)): $(0, 0)$, (h, w) , $(0, w)$, $(h, 0)$ były w środku.

Poeksperymentuj na różnych obrazkach: a) policz fft; b) zmień (niektóre) wartości; c) policz ifft.

Typowe zastosowanie: usuwanie szumu. W http://www.math.uni.wroc.pl/~lorek/image_analysis/images/fourier/ przygotowanych jest kilka obrazków, w szczególności dla `clown.jpg` jest przygotowana maska `clown_mask_blurred.png`

Potestuj na różnych obrazkach (tj. na transf.) np.

- Usuń dużo wartości oprócz “środką”
- Usuń środek
- Usun duże / małe wartości.. wszędzie lub poza środkiem
- “Wymaskuj” (“ręcznie”) duże “peaki”

Czym skutkują powyższe operacje?