

LAB 5 Statystyki obrazu, histogram, porównywanie obrazów. Pobieranie i mieszanie kanałów w trybie RGB, Matplotlib.

1. Statystyki

from PIL import ImageStat as stat

Ciekawsze statystyki:

1. MEAN Średnia wartość pikseli – większa wartość średnie intuicyjnie oznacza jaśniejszy obraz
2. STDDEV Odchylenie standardowe wartości pikseli - większa wartość intuicyjnie mówi o większym kontraście obrazu

$$stddev = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \text{średnia})^2}$$

3. RMS Pierwiastek średniokwadratowy pikseli (wartość skuteczna) - większa wartość intuicyjnie mówi o większej „energii” obrazu

$$rms = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}$$

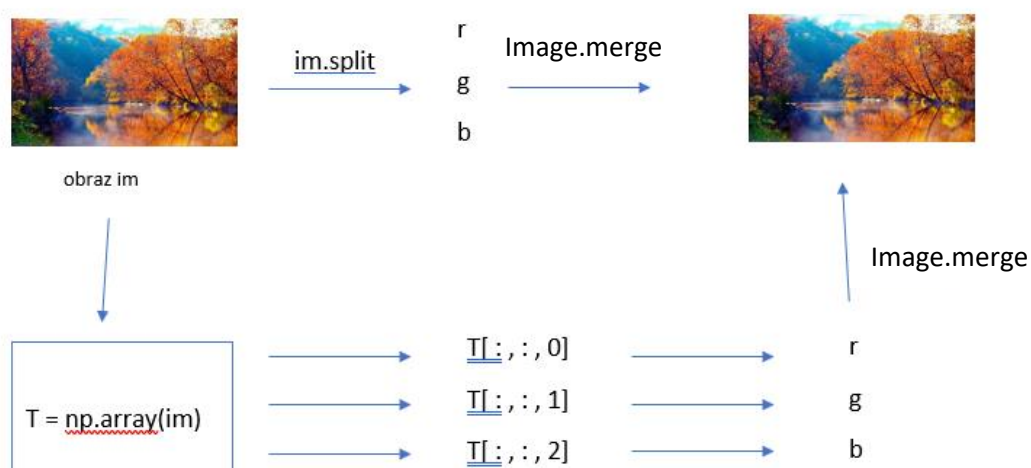
2. Histogram

Histogram w trybie 'L' : lista długości 256, gdzie element o indeksie i jest j liczbą pikseli o tej wartości i.

Histogram w trybie 'RGB' : lista długości 3*256, powstała jako konkatencja histogramów kanałów r, g, b.

Podobnie powstają histogramy w innych trybach.

Pobieranie i mieszanie kanałów w trybie RGB



Zadania

Wyszukaj w sieci dowolny obraz w trybie RGB (**obraz.png**) i wczytaj jako **im**.

1. Pobierz statystyki obrazu **im** i je skomentuj
 - a. przedstaw histogram obrazu **im** na diagramie plt, przedstaw histogramy wszystkich jego kanałów na diagramach plt.
 - b. Ile jest pikseli o wartości 155 na każdym z kanałów?
 - c. Napisz funkcję **zlicz_piksele(obraz, kolor)**, która zlicza, ile jest pikseli w danym kolorze. Ile jest pikseli o wartości [155,155,155] w obrazie **im**?
2. Zapisz obraz **im** w formacie jpg a potem wczytaj jako **im_jpg**.
 - a. Porównaj statystyki obrazów **im** oraz **im_jpg**. Dla czego te obrazy się różnią?
 - b. Zastosuj **ImageChops.difference**, aby otrzymać różnicę tych obrazów. Pobierz statystyki różnicy i je skomentuj
 - c. Jak zmieniają się statystyki, gdy jeszcze dwa razy zapiszesz obraz **im_jpg** w formacie jpg?
3. Wykonaj następujące polecenia dla obrazu **im**
 - a. Wczytaj tablicę obrazu i pobierz kanały **t_r**, **t_g**, **t_b** obrazu z tablicy obrazu, zapisz jako obrazy **im_r**, **im_g**, **im_b**
 - b. Utwórz obraz **im1** przez scalenie metodą **merge** obrazów **im_r**, **im_g**, **im_b** i zastosuj **ImageChops.difference(im, im1)** do porównania otrzymanego obrazu z obrazem wejściowym.
 - c. Umieść na jednej figurze plt (**fig1.png**) obrazy **im**, **im1** i wynik porównania
 - d. Czy są jakieś różnice?
4. Napisz funkcję **mieszaj_kanaly(obraz)**, która w sposób losowy miesza kanały **r**, **g**, **b** pobrane metodą **split** z danego obrazu oraz ich negatywy **nr**, **ng**, **nb**. Dopuszczalne jest losowanie z powtórzeniami tzn. może być **b**, **r**, **g** ale też **b**, **b**, **g**.
 - a. Zastosuj tę funkcję do obrazu **im**. Otrzymany obraz nazwij **mix** i zapisz w formacie png.
 - b. Napisz funkcję **rozpoznaj_mix(obraz, mix)**, która dla danych obrazów w trybie RGB **obraz** i **mix** rozpoznaje w jaki sposób **mix** powstał z **obrazu** (zmiana kolejności kanałów).
5. Dlaczego polecenie **r, g, b = im.split()** nie działa, gdy **im = Image.open('beksinski1.png')**?