

Grafika rastrowa i fotografia

Instrukcja do laboratorium 2

inż. Andrii Shekhovtsov

Listopad 2021

1 Zasady oceniania

Oceny będą wystawiane na podstawie sprawozdań wykonywanych z każdej instrukcji. W treści instrukcji są podane **Zadania** wraz przypisaną **[liczbą punktów]**. W tej instrukcji jest **7** zadań, łącznie za **10** punktów. Ocena zależy od liczby punktów za wykonane zadania.

Liczba punktów	Ocena
5	3,0
6 – 7	3,5
8	4,0
9	4,5
10	5,0

Termin oddania sprawozdania jest ustawiony w systemie moodle. W przypadku nie oddania zadania w terminie, uzyskana ocena będzie zmniejszana o 0,5 za każdy zaczęty tydzień opóźnienia.

W przypadku pracy w środowisku Jupyter Notebook proszę o przesłanie tylko wygenerowanego pliku pdf.

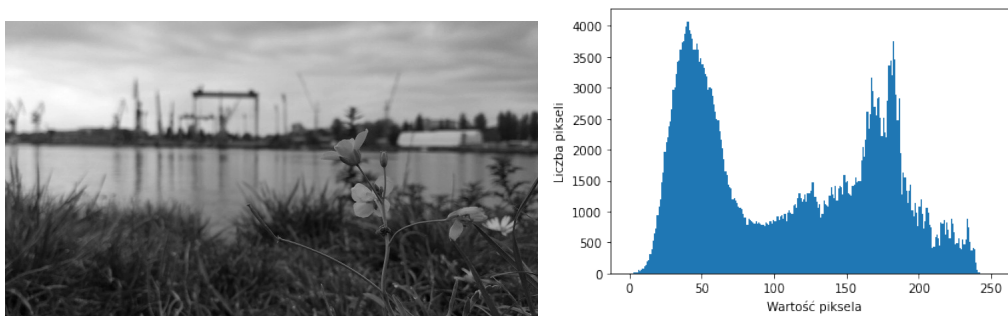
W przypadku pracy w innych środowiskach proszę o przesłanie sprawozdania w formacie pdf zawierającego wykonane zadania (w zależności od zadania to mogą być odpowiedzi na pytania, kod funkcji, obrazek (screenshot) i wyjaśnienie), **a także całości kodu wykonanego w ramach przygotowania tego sprawozdania.**

2 Materiał pomocniczy

2.1 Histogram

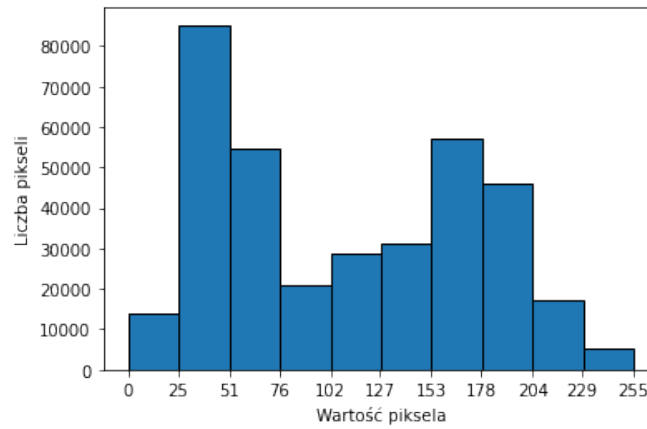
Histogram obrazu pokazuje rozkład wartości pikseli w tym obrazie. Na Rysunku 1 jest pokazane zdjęcie i histogram wyznaczony dla tego zdjęcia. Oś X oznacza wartość piksela, a oś Y pokazuje ile pikseli o takiej wartości zawiera się w naszym obrazie.

Za pomocą histogramu można ocenić rozkład wartości pikseli w obrazie. Przykładowo, pik w histogramie po lewej stronie wskazuje na dużą liczbę ciemnych pikseli na zdjęciu.



Rysunek 1: Obrazek i jego histogram

Jak już było powiedziane, histogram to liczba pikseli o poszczególnych wartościach. Żeby wyznaczyć histogram, należy po prostu zliczyć wystąpienia poszczególnych wartości pikseli. Możliwe też jest wyznaczenie histogramu dla mniejszej liczby przedziałów, w tym wypadku zlicza się liczbę pikseli trafiających do każdego z zakresów wartości, jak to jest pokazane na Rysunku 2.



Rysunek 2: Przykład histogramu o 10 przedziałach.

Źródło: https://en.wikipedia.org/wiki/Image_histogram

2.1.1 Normalizacja obrazu

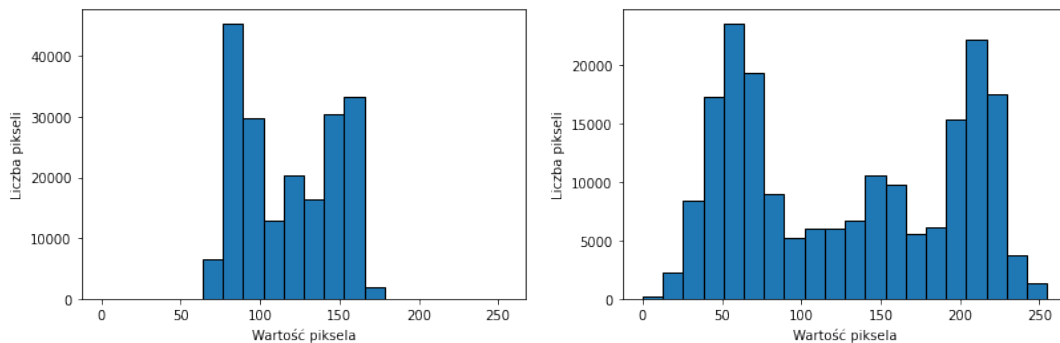
Normalizacja obrazu (inaczej rozciąganie histogramu) jest operacją zmieniającą zakres wartości pikseli. Zwykle stosuje się tej operacji do poprawienia kontrastu zdjęć.

Zakładając, że mamy obraz I w odcieniach szarości o zakresie wartości $[Min, Max]$ i chcemy go znormalizować do zakresu wartości $[newMin, newMax]$. Wynikowy obrazek otrzymamy za pomocą wzoru (1).

$$I_N = (I - Min) \frac{newMax - newMin}{Max - Min} + newMin \quad (1)$$

W przypadku normalizacji do pełnego zakresu wartości $[0, 255]$ stosujemy wzór (2).

$$I_N = 255 \cdot \frac{I - Min}{Max - Min} \quad (2)$$



Rysunek 3: Histogram pewnego obrazu do (po lewej) i po normalizacji (po prawej).

Źródło: [https://en.wikipedia.org/wiki/Normalization_\(image_processing\)](https://en.wikipedia.org/wiki/Normalization_(image_processing))

2.1.2 Wyrównywanie histogramu

Wyrównywanie histogramu jest operacją na histogramie obrazku, pozwalającą na wypłaszczenie histogramu, co skutkuje poprawą kontrastu zdjęcia.

Dla wyrównywania histogramu należy przedstawić go w postaci rozkładu prawdopodobieństwa, czyli podzielić każdą wartość histogramu na liczbę pikseli w obrazie (suma histogramu).

$$p_i = \frac{h_i}{n} \quad (3)$$

W równaniu (3) powyżej p_i oznacza rozkład prawdopodobieństwa wystąpienia poszczególnych pikseli, h_i to kolejne wartości histogramu, a n to liczba wszystkich pikseli w obrazie.

Następnie jest liczona dystrybuenta (4).

$$d_i = \sum_{j=0}^i p_j \quad (4)$$

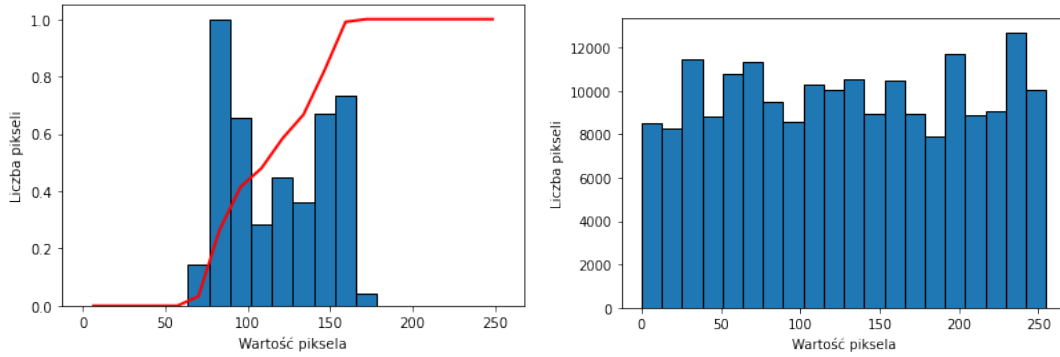
Używając wartości dystrybuenty, uzyskujemy nowy obrazek:

$$y_{i,j} = d(x_{i,j}), \quad (5)$$

gdzie $y_{i,j}$ to piksel nowego obrazku, $x_{i,j}$ to wartość piksela obrazku oryginalnego, a $d(x_{i,j})$ to wartość dystrybuenty pod indeksem $x_{i,j}$.

Należy zwrócić uwagę że po przekształceniu wartości pikseli w obrazie zawierają się w przedziale $[0, 1]$. Żeby wrócić do pierwotnego zakresu $[0, 255]$ na leży wykonać przekształcenie (6) i zamienić wynik na `int`.

$$y' = \lceil 256 \cdot y \rceil - 1. \quad (6)$$



Rysunek 4: Histogram pewnego obrazu do i jego dystrybuenta (po lewej), histogram po wyrównywaniu (po prawej).

Źródło: https://en.wikipedia.org/wiki/Histogram_equalization

2.2 Porównywanie obrazków za pomocą histogramu

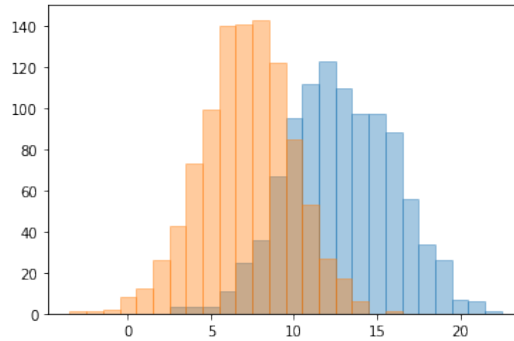
Używając histogramu obrazku, można zaimplementować bardzo prosty algorytm służący do porównywania obrazków. Załóżmy że mamy dwa pokrywające się histogramy $H^{(1)}$ i $H^{(2)}$. Przykładowo, mogą być położone względem siebie w sposób pokazany na Rysunku 5.

Uwaga! Histogramy muszą być znormalizowane przed użyciem w tych wzorach! Można do tego użyć przykładowo wzoru (7), w którym x_i oznacza wartość przed znormalizowaniem, x_{min} i x_{max} oznaczają odpowiednio najmniejszą i największą wartość spośród x_i , a r_i oznacza wartość po znormalizowaniu:

$$r_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (7)$$

Część wspólna dwóch histogramów może być wyznaczona w następujący sposób:

$$IS = \sum_{i=0}^{n-1} \min(H_i^{(1)}, H_i^{(2)}), \quad (8)$$



Rysunek 5: Przykład intersekcji dwóch histogramów

gdzie IS_i to część wspólna histogramów, n to liczba wartości w każdym histogramie, a $H_i^{(1)}$ i $H_i^{(2)}$ to są poszczególne wartości obu histogramów. Uzyskiwana w ten sposób wartość nie może być jednak wprost użyta do porównywania obrazków, należy dodatkowo podzielić ją poprzez sumę wartości pierwszego histogramu:

$$IS = \frac{\sum_{i=0}^{n-1} \min(H_i^{(1)}, H_i^{(2)})}{\sum_{i=0}^{n-1} H_i^{(1)}} \quad (9)$$

Wynikiem tego wzoru będzie znormalizowana intersekcja dwóch histogramów IS , na podstawie której można wnioskować o podobieństwie dwóch obrazków. Im bliżej ta wartość jest do 1, tym bardziej podobne są obrazy. Dla dwóch jednakowych obrazków wartość ta wynosi 1.

Źródło: <https://mpatacchiola.github.io/blog/2016/11/12/the-simplest-classifier-histogram-intersection.html>

2.3 Wczytywanie i wyświetlanie obrazków

Przy wczytywaniu obrazku najlepiej od razu skonwertować go do typu `int`, żeby nie doprowadzić do przepełnienia zmiennej przy modyfikowaniu obrazku w dalszym ciągu. Należy też sprawdzić czy wartości pikseli dla wczytanego obrazku zawierają się w granicach $[0, 1]$ czy $[0, 255]$. W przypadku zakresu $[0, 1]$ należy skonwertować go do zakresu $[0, 255]$.

```
1 # Wczytywanie obrazu i konwersja go na int
2 img = plt.imread('flower.jpg').astype(int)
```

Przy wyświetlaniu obrazu należy też podać dodatkowe argumenty, które spowodują brak normalizacji przy wyświetlaniu. Jest to konieczne żeby móc obserwować wyniki operacji na histogramie. Do wyświetlania można też posłużyć się poniższą funkcją.

```
1 def imshow(img):
2     plt.imshow(img, cmap='gray', vmin=0, vmax=255)
3
4 # Przykład użycia
5 imshow(img)
6 plt.show()
```

3 Zadania do wykonania

3.1 Histogram i operacje na histogramie

Uwaga: W tej części używamy obrazku pod nazwą `lake.jpg`.

[2 pkt.] Zadanie 1: Korzystając z informacji zawartych w Materiałach pomocniczych, zaimplementować własną funkcję do obliczenia histogramu z jednokanałowego obrazka (w odcieniach szarości lub jeden z kanałów RGB). W

implementacji należy uwzględnić możliwość podania różnej ilości przedziałów (jako drugi argument funkcji). Zakładamy, że zakres możliwych wartości pikseli to $[0, 255]$. W wyniku działania funkcja ma zwracać rozkład wartości pikseli (histogram) oraz granice słupków. Do wygenerowania zakresów można posłużyć się funkcją `np.linspace()`.

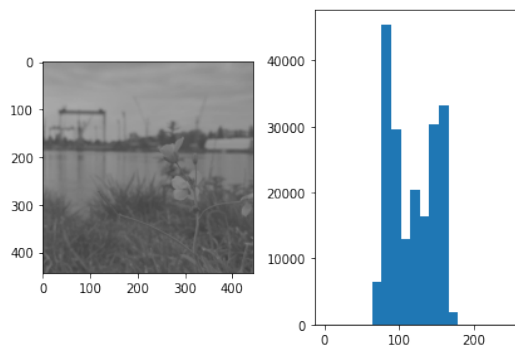
Wyniki działania muszą być zgodne z wywołaniem funkcji `np.histogram()` na tym samym obrazie i dla takiej samej liczby przedziałów.

Przykładowo, dla 10 przedziałów będzie to wyglądać w ten sposób:

```
1 hist, bins = np.histogram(img, bins=10, range=(0, 255))
```

[1 pkt.] Zadanie 2: Napisać funkcję, która wyświetli podany jako argument obraz wraz z jego histogramem o zadanej liczbie przedziałów. Przy wyświetlaniu obrazka należy podać również argumenty `vmin=0` i `vmax=255`, lub skorzystać z funkcji podanej na końcu sekcji 2. Do rysowania słupków histogramu należy użyć funkcji `plt.bar()` (jej opis jest podany w dokumentacji).

Przykładowy wynik działania takiej funkcji jest podany na Rysunku 6.



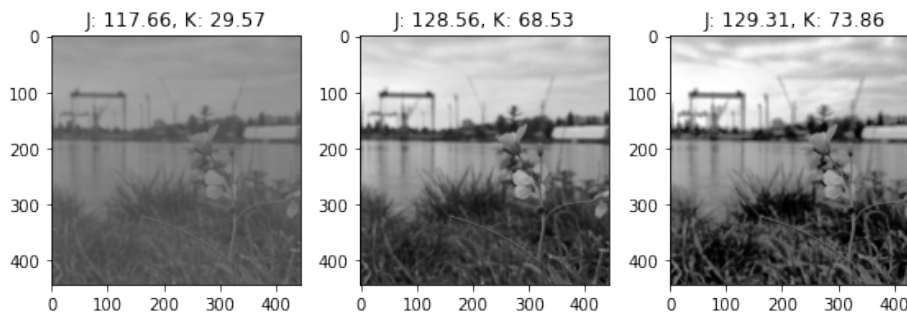
Rysunek 6: Przykład działania funkcji do wyświetlenia obrazku i jego histogramu

[1 pkt.] Zadanie 3: Używając równania (2) napisać funkcję do znormalizowania obrazku. Funkcja ta ma normalizować wartości pikseli do zakresu $[0, 255]$. Wyświetlić przetworzony obrazek i jego histogram.

[1 pkt.] Zadanie 4: Używając równania opisane w sekcji 2.1.2 napisać funkcję do wyrównywania histogramu. Wyświetlić przetworzony obrazek i jego histogram.

[2 pkt.] Zadanie 5: Używając funkcji `plt.subplot` i kodu opracowanego na poprzednim laboratorium, wyświetlić obok siebie oryginalny obrazek, obrazek po normalizacji, obrazek po wyrównywaniu histogramu. Nad każdym obrazkiem jako tytuł podać jasność i kontrast. Przykład jest podany na Rysunku 7.

Słownie opisać, w jaki sposób operacje normalizacji oraz wyrównywania histogramu wpływają na obrazek i jego histogram (przydadzą się też wykresy z poprzednich zadań).



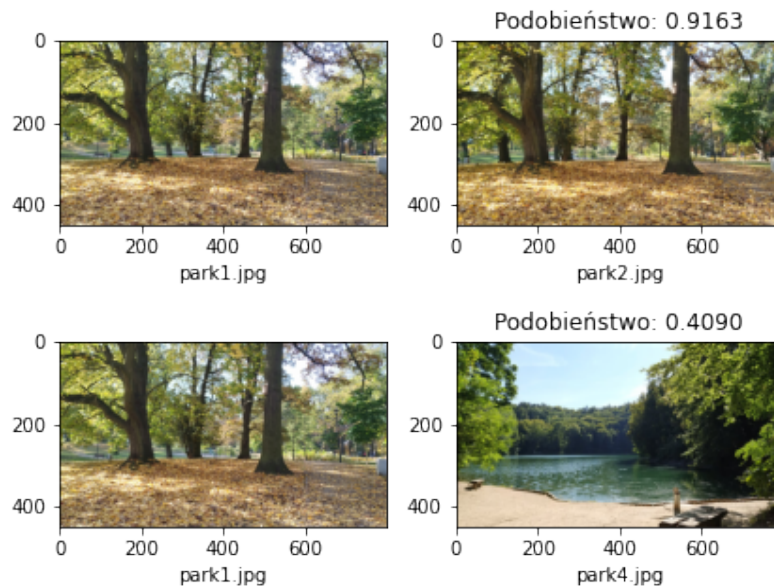
Rysunek 7: Przykład wyświetlenia oryginalnego i przetworzonych obrazków.

3.2 Porównywanie obrazków za pomocą histogramu

[1 pkt.] **Zadanie 6:** Używając informacji, podanych w sekcji 2.2, napisać funkcję zwracającą znormalizowaną intersekcję (część wspólną) dwóch histogramów.

[2 pkt.] **Zadanie 7:** Używając informacji, podanych w sekcji 2.2 i wskazówek poniżej, napisać funkcję, która obliczy współczynnik podobieństwa dwóch **kolorowych** obrazów, bazując się na znormalizowanej intersekcji histogramów dwóch obrazów.

Zademonstrować działanie funkcji na wykresach, pokazujących dwa obrazki obok siebie wraz ze stopniem ich podobieństwa. Do porównania można użyć obrazki z folderu `park/` lub `flowers/`. Przykład demonstracji pokazany na Rysunku 8.



Rysunek 8: Przykład działania funkcji

Wskazówka 1: Kolorowy obraz składa się z 3 warstw (kanałów). Dla każdego kanału jesteśmy w stanie obliczyć histogram.

Wskazówka 2: Współczynnik podobieństwa dla obrazu kolorowego możemy wyliczyć jako średnia arytmetyczna 3 wartości intersekcji histogramów obu obrazków dla poszczególnych kanałów w tych obrazkach.