

# Podstawowe metod analizy i przetwarzania obrazów.

## Wprowadzenie

Przetwarzanie obrazu to fascynująca dziedzina informatyki, która zajmuje się analizą, manipulacją i interpretacją danych wizualnych. W dzisiejszym świecie, w którym obrazy stanowią znaczną część informacji, metody przetwarzania obrazu odgrywają kluczową rolę w wielu dziedzinach życia głównie w medycynie.

Podstawą sukcesu w przetwarzaniu obrazu jest zdolność do ekstrahowania istotnych informacji z obrazów, co często może być trudne ze względu na różnorodność danych i ich skomplikowaną naturę. Dlatego metody przetwarzania obrazu, takie jak segmentacja, rozpoznawanie wzorców, analiza tekstur czy detekcja obiektów, stają się nieodzownymi narzędziami w rozwiązywaniu problemów związanych z obrazami. Wiele z tych metod wymaga wstępnego przetworzenia analizowanego obrazu.

## Ćwiczenia laboratoryjne

### Zadanie 1. Eksplorator histogramów

**Kontekst.** Wyrównanie zakresu jasności często zaczyna się od zrozumienia histogramu. Chcemy porównać rozkłady tonalne w RGB i w skali szarości oraz zademonstrować rozciąganie zakresu na percentylach.

#### Zadanie.

1. Wczytaj obraz (RGB lub Gray) z `skimage.data` i/lub własny.
2. Oblicz i narysuj `histogram(y)` dla:
  - o skali szarości (zbudowanej z RGB jako luminancja **lub** średnia kanałów),
  - o każdego kanału R, G, B (jeśli obraz jest kolorowy).
3. Zaimplementuj „stretching” intensywności z odcięciem percentyli  $p_{low}$  i  $p_{high}$  (domyślnie 2–98%).
4. Wyświetl: oryginał, stretched (gray i/lub RGB), odpowiadające histogramy (te same ustawienia osi/bins).
5. Dodaj możliwość zmiany  $p_{low}$ ,  $p_{high}$  oraz wyboru sposobu konwersji do Gray ('lum', 'avg', 'r'/'g'/'b').

**Wymagania minimalne.** Jedna funkcja, która dla danego obrazu rysuje mozaikę: `obraz(y) + histogram(y)` przed/po; możliwość ustawienia percentyli.

**Wynik.** Krótka notatka (3–5 zdań): co się dzieje z histogramem i kontrastem przy 2–98% vs 1–99%?

**Wskazówki.** `skimage.exposure.histogram`, `exposure.rescale_intensity`, `matplotlib` (wspólne osie), normalizacja danych do  $[0,1]$ .

## Zadanie 2. Normalizacja vs wyrównanie histogramu vs gamma

**Kontekst.** Trzy popularne operacje globalne: liniowe rozciąganie, wyrównanie histogramu oraz korekcja gamma — mają różne skutki i artefakty.

### Zadanie.

1. Dla wybranego obrazu (ciemnego/kontrastowo słabego) wygeneruj trzy wersje:
  - o **Normalizacja:** `rescale_intensity` (percentyle 1–99 lub 2–98),
  - o **Wyrównanie histogramu:** `equalize_hist`,
  - o **Gamma:** `adjust_gamma` (np.  $\gamma \in \{0.6, 0.8, 1.2\}$ ).
2. Złóż siatkę 2×2: oryginał + trzy przetworzenia (opisz podpisami, jakie parametry).
3. Krótko skomentuj, kiedy która metoda działa lepiej (np. zachowanie świateł/cieni, „przepalenia”, banding).

**Wymagania minimalne.** Jedna funkcja, która przyjmuje obraz i zwraca figurę 2×2 oraz słownik parametrów użytych metod.

**Wynik.** Siatka 2×2 + komentarz (4–6 zdań) o różnicach i potencjalnych artefaktach.

**Wskazówki.** Porównaj histogramy każdej wersji. Dla gamma pamiętaj:  $\gamma < 1$  rozjaśnia cienie,  $\gamma > 1$  przyciemnia.

## Zadanie 3. Lokalny kontrast (CLAHE)

**Kontekst.** Globalne wyrównanie potrafi przepalać światła. CLAHE (Contrast Limited Adaptive Histogram Equalization) wzmacnia kontrast lokalnie z ograniczeniem wzmocnienia.

### Zadanie.

1. Zastosuj `equalize_adapthist` (CLAHE) z różnymi parametrami:
  - o **clip\_limit:** np. 0.01, 0.03, 0.05,
  - o **kernel\_size** (rozmiar płytki): mały vs duży (np. 8×8 vs 32×32).
2. Porównaj wynik z globalnym `equalize_hist`.
3. Oceń efekty na obrazie o nierównym oświetleniu lub z drobnymi detalami w cieniach (wybierz adekwatny obraz).
4. Zadeemonstruj co najmniej 4 warianty (różne kombinacje parametrów).

**Wymagania minimalne.** Funkcja `clahe_demo(img, clip_limit, kernel_size)` + skrypt porównujący 4 warianty i globalne wyrównanie.

**Wynik.** Mozaika wyników + notatka (5–8 zdań): wpływ `clip_limit` i rozmiaru płytki na: szum, halo, „plastikowy” wygląd, ochronę świateł.

**Wskazówki.** CLAHE najlepiej stosować na luminancji (Gray/Y/L), a potem łączyć z kolorami, jeśli obraz jest RGB.

## Zadanie 4. Rozkład informacji w przestrzeniach barw (HSV, LAB, YUV)

**Kontekst.** Różne przestrzenie barw rozdzielają luminancję i chrominancję w różnym stopniu, co wpływa na filtrację i segmentację.

### Zadanie.

1. Dla obrazu RGB oblicz reprezentacje w: HSV, LAB, YUV.
2. Wyświetl obok siebie kanały:
  - HSV: H, S, V
  - LAB: L, a, b
  - YUV: Y, U, V
3. Opisz jakościowo, które kanały najlepiej niosą informację o jasności (krawędzie, faktura), a które o kolorze (odcień vs nasycenie).
4. Zaznacz różnice wizualne (np. granice obiektów lepiej widoczne w L/Y/V).

**Wymagania minimalne.** Funkcja rysująca wiersz/kolumnę kanałów dla każdej przestrzeni (wspólna skala intensywności, czytelne tytuły).

**Wynik/oddanie.** Jedna plansza z 9 obrazkami ( $3 \times 3$ ) + krótka notatka (5–8 zdań) „który kanał do czego” i dlaczego.

**Wskazówki.** Upewnij się, że kanały o zakresie ujemnym (np. a/b w LAB) są poprawnie przeskalowane do wizualizacji.