

# PRZETWARZANIE OBRAZÓW CYFROWYCH

## ĆWICZENIE 4

### REDUKCJA LICZBY BARW W OBRAZIE - DYFUZJA BŁĘDÓW

#### WPROWADZENIE

Typowym zadaniem z zakresu przetwarzania obrazów barwnych omawianym w ćwiczeniu 3 jest kwantyzacja barwy. Skupiono się tam na doborze optymalnej palety barw i metodach kwantyzacji z wykorzystaniem różnych przestrzeni barwnych.

W omawianym ćwiczeniu zapoznamy się z metodami kwantyzacji wykorzystującymi dithering (czyli rozpraszanie) w celu symulacji kolorów przy ograniczonej palecie barw. Dithering pozwala nie tylko na uzyskanie "dodatkowych" kolorów, ale przede wszystkim powoduje złudzenie bardziej płynnego przejścia od jednego koloru do drugiego. Widać to bardzo wyraźnie dla ośmiokolorowych grafik przedstawionych poniżej (źródło: [wikipedia.pl](http://wikipedia.pl)).



1. Pełna paleta barw (24 bity)



2. Obraz po kwantyzacji na 8 barw



3. Obraz po kwantyzacji na 8 barw z wykorzystaniem metody Floyda-Steinberga

Jednym z najbardziej znanych algorytmów ditheringu jest Algorytm rozpraszania błędów Floyda-Steinberga. Algorytm Floyda-Steinberga stosuje się przy dostosowywaniu obrazu do formatu zapisu lub medium o ograniczonej palecie barw.

Zasada działania algorytmu Floyda-Steinberga opiera się o rozproszenie błędu kwantyzacji piksela na „sąsiadów” zgodnie z maską:

$$\frac{1}{16} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 7 \\ 3 & 5 & 1 \end{bmatrix}$$

Piksele obrazu przetwarzane są w kolejnych rzędach, od lewej do prawej. Dla każdego przetworzonego piksela obliczany jest błąd kwantyzacji (w przypadku monochromatycznym jest to pojedyncza liczba, dla przestrzeni RGB - trzy, dla CMYK - cztery). Wartość tej różnicy jest następnie dodawana z odpowiednimi współczynnikami do wartości jeszcze nie przetworzonych pikseli sąsiadujących.



Poniżej przedstawiony jest pseudo kod opisujący działanie algorytmu Floyda-Steinberga:

```
for each y from top to bottom
  for each x from left to right
    oldpixel := pixel[x][y]
    newpixel := find_closest_palette_color(oldpixel)
    pixel[x][y] := newpixel
    quant_error := oldpixel - newpixel
    pixel[x+1][y] := pixel[x+1][y] + 7/16 * quant_error
    pixel[x-1][y+1] := pixel[x-1][y+1] + 3/16 * quant_error
    pixel[x][y+1] := pixel[x][y+1] + 5/16 * quant_error
    pixel[x+1][y+1] := pixel[x+1][y+1] + 1/16 * quant_error
```

## PROGRAM ĆWICZENIA

1. Napisz skrypt redukujący liczbę barw w obrazie wykorzystujący algorytm dyfuzji błędów Floyda-Steinberga
2. Wybierz 5 różnorodnych obrazów barwnych o rozdzielczości przestrzennej co najmniej 512x512
3. Dla każdego z obrazów przeprowadź kwantyzację na 2, 16 oraz 256 barw przy pomocy skryptu Matlaba oraz dostępnego programu graficznego np. Irfanview, GIMP
4. Wyznacz wartości PSNR dla wszystkich obrazów po kwantyzacji i porównaj z je z subiektywną oceną barwy
5. Sformułuj wnioski wynikające z ćwiczenia.

## MATERIAŁY DODATKOWE

- [http://en.wikipedia.org/wiki/Floyd%E2%80%93Steinberg\\_dithering](http://en.wikipedia.org/wiki/Floyd%E2%80%93Steinberg_dithering)
- [http://en.literateprograms.org/Floyd-Steinberg\\_dithering\\_%28C%29](http://en.literateprograms.org/Floyd-Steinberg_dithering_%28C%29)
- <http://www.algorytm.org/przetwarzanie-obrazow/algorytm-floyda-steinberga.html>