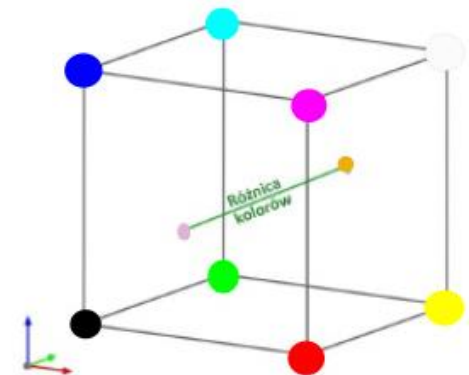


# REDUKCJA KOLORÓW

## (KWANTYZACJA KOLORÓW)

- Proces redukcji liczby kolorów dla obrazu
- Dostępne kolory – paleta kolorów
- **Zastosowania:**
  - wyświetlenie obrazu na urządzeniu o ograniczonej liczbie dostępnych kolorów
  - konwersja do formatów z ograniczoną liczbą różnych kolorów (np. GIF)
  - kompresja obrazów i zdjęć
- **Dwa podstawowe kroki podczas redukcji kolorów:**
  - wyznaczenie palety kolorów
  - dla każdego piksela zamiana jego koloru na kolor z palety
    - najczęściej, ale nie zawsze jest to kolor najbliższy w paletcie – odległość w  $R^3$  (metryka euklidesowa, metryka miejska)
    - odległość w modelu Lab



# METODY REDUKCJI KOLORÓW

## Metody wyznaczania palety kolorów

### 1. Paleta niezależna od obrazu

- np. Jednorodny podział sześciangu RGB
- Wada: występujące w obrazie artefakty

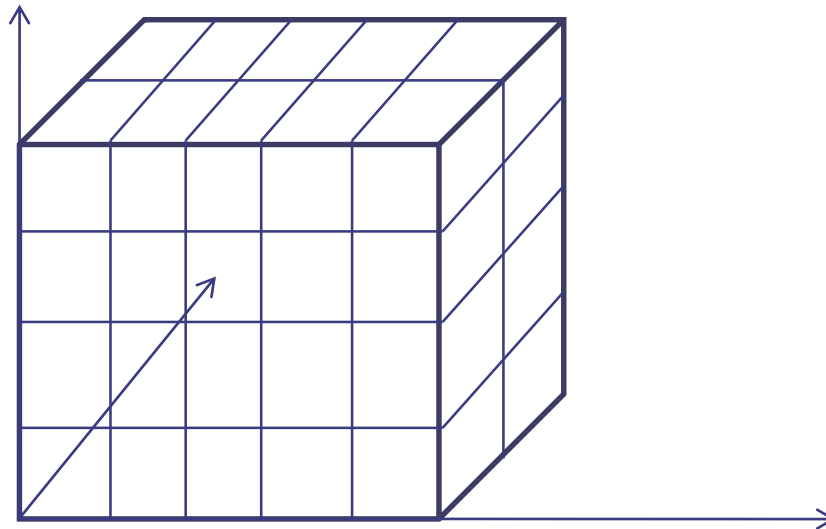
### 2. Metoda adaptacyjna

- Znacznie lepsza jakość obrazów, ale metoda czasowo kosztowna



# 1. PALETA NIEZALEŻNA OD OBRAZU

- Paleta wybierana w sposób arbitralny, niezależny od obrazu
- Jednorodny podział sześcianu RGB dla każdej współrzędnej
- Wynikowa liczba kolorów:  $k_R * k_G * k_B$
- Jako paleta wynikowa – środki prostopadłościanów



# DITHERING (DRŻENIE, ROZPRASZANIE)

(NIEZALEŻNIE DLA KAŻDEGO KANAŁU RGB)

- **Dithering** umożliwia uzyskanie wrażenia wielu odcieni szarości (lub koloru) na urządzeniu posiadającym zdolność wyświetlania jedynie k-poziomów
- **K=2** – dostępne jedynie dwa kolory: czarny i biały
- **K=4** – dostępne poziomy: czarny, biały i dwa odcienie szarości 33% i 66%



# ROZPRASZANIE ŚREDNIE

## THE AVERAGE DITHERING, UNIFORM QUANTIZATION

- Najprostsza metoda sprowadzająca się do dyskretyzacji obrazu wejściowego.
- Jako próg wykorzystywana jest średnia wartość w przedziale.
- Metoda prosta w implementacji, ale niezadawalające wyniki



Obraz oryginalny  
256 odcieni



2 poziomy B&W



4 poziomy



# ROZPRASZANIE LOSOWE

## THE RANDOM DITHERING

- Dla każdego punktu generowana jest wartość losowa
- Jeśli jest ona większa od wartości piksela, wybierany jest górny poziom, w przeciwnym przypadku dolny



Obraz oryginalny  
256 odcieni



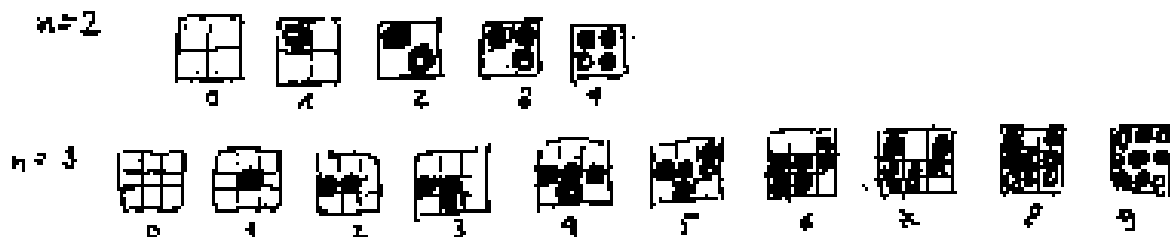
2 poziomy B&W



4 poziomy

# METODA UPORZĄDKOWANEGO DRŻENIA

## THE ORDERED DITHERING



**Macierze drżenia:**

$$D_2 = \begin{bmatrix} 0 & 2 \\ 3 & 1 \end{bmatrix} \quad D_3 = \begin{bmatrix} 6 & 8 & 4 \\ 1 & 0 & 3 \\ 5 & 2 & 7 \end{bmatrix}$$

$$D_{2n} = \begin{bmatrix} 4D_n & 4D_n + 2U_n \\ 4D_n + 3U_n & 4D_n + 1U_n \end{bmatrix}$$

gdzie

$U_n$  – macierz samych jedynek

a)  $k=2$

Aby wyświetlić intensywność  $I$  za pomocą wzoru  $n \times n$  należy włączyć te wszystkie piksele, których wartość w macierzy  $D_n$  jest mniejsza niż  $I$

b)  $k>2$

$I(\text{image})$  is in range  $\langle 0, n^2 * (k-1) \rangle$

$\text{col} = I(\text{image}) \div n^2$

$\text{re} = I(\text{image}) \bmod n^2$

Aby wyświetlić intensywność  $I$  za pomocą wzoru  $n \times n$  należy użyć poziomu  $\text{col}$  w tych pikselach, których wartość w macierzy  $D_n$  jest mniejsza niż  $\text{re}$ , w przeciwnych przypadkach wartości  $\text{col}+1$

# METODA UPORZĄDKOWANEGO DRŻENIA

```
Ii = GetPixel(x,y); // Ii is in range <0,n2*(k-1)>  
col = Ii div n2  
re = Ii mod n2  
i = x mod n  
j = y mod n  
if re > Dn[i,j] col++  
PutPixel(x,y,col);
```



Obraz oryginalny  
256 odcieni



2 poziomy B&W



4 poziomy



# METODA PROPAGACJI BŁĘDU

## ERROR DIFFUSION DITHERING

Błąd (różnica pomiędzy wartością piksela a jego aproksymacją) jest dodawany do kolejnych pikseli (z różnymi wagami)

```
K = Approximate(S[x][y]);    //Approximate S to the nearest displayable
    intensity
I[x][y] = K;                  //Draw the pixel at (x,y)
error = S[x][y]-K;            //Error - must be of type float

//error is diffused over several pixels in the image array using filter
    array
// float filter[2*f_x+1][2*f_y+1]

for (i=-f_x; i<+f_x; i++)
    for (j=-f_y; j<+f_y; j++)
        S[x+i][y+j] += error*filter[i][j];
```



# METODA PROPAGACJI BŁĘDU

## Filtry (wagi)

Floyd and Steinberg Filter	0	0	0		
	0	0	7/16		
	3/16	5/16	1/16		
Burkes Filter	0	0	0	0	0
	0	0	0	8/32	4/32
	2/32	4/32	8/32	4/32	2/32
Stucky Filter	0	0	0	0	0
	0	0	0	0	0
	0	0	0	8/42	4/42
	2/42	4/42	8/42	4/42	2/42
	1/42	2/42	4/42	2/42	1/42



Obraz oryginalny

256 odcieni

Grafika komputerowa 1



2 poziomy B&W



4 poziomy



# DITHERING – ROZPRASZANIE ŚREDNIE



Obraz oryginalny 256 odcieni



2 poziomy B&W

# DITHERING - ROZPRASZANIE



Obraz oryginalny 256 odcieni



2 poziomy B&W  
(metoda uporządkowanego drżenia)





# DITHERING - ROZPRASZANIE



Obraz oryginalny 256 odcieni

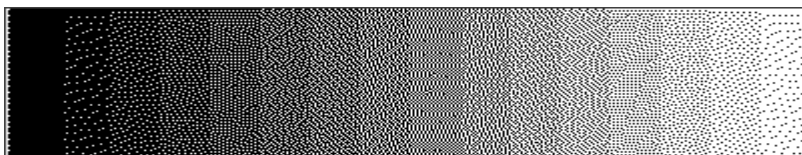
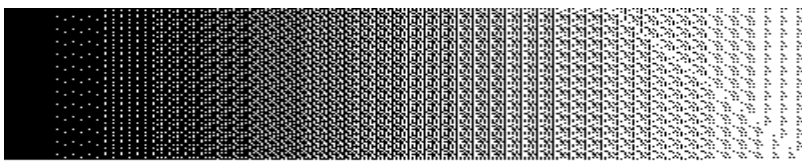
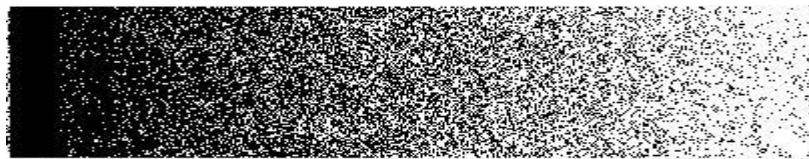
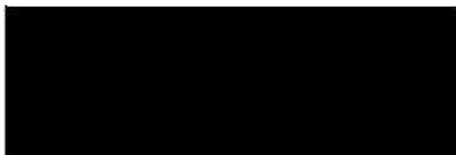


2 poziomy B&W  
(metoda propagacji błędów)



# PORÓWNANIE RÓŻNYCH METOD DITHERINGU

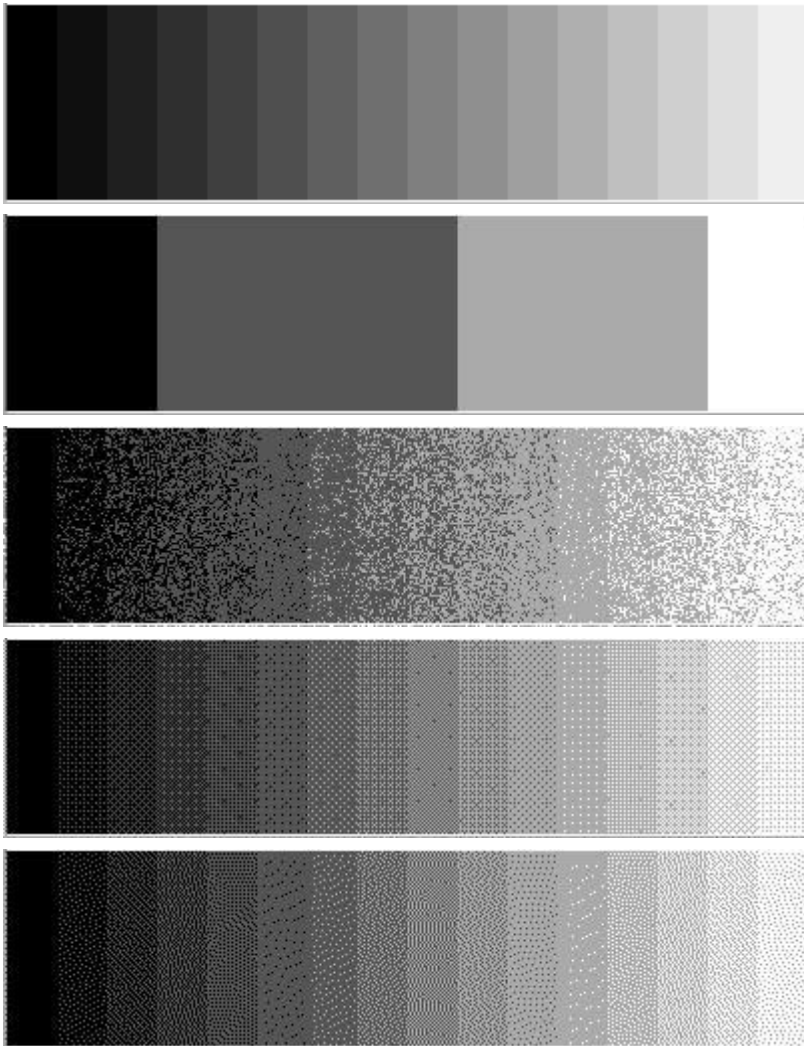
K=2 (DWA POZIOMY NA URZĄDZENIU)



- 16 poziomów szarości
- Rozpraszanie średnie
- Rozpraszanie losowe
- Uporządkowane drzenie
- Metoda propagacji błędu

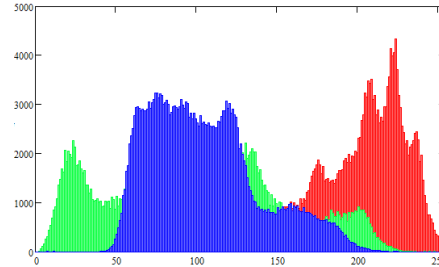
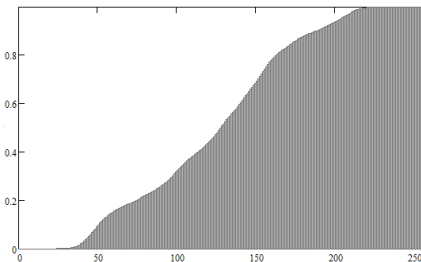
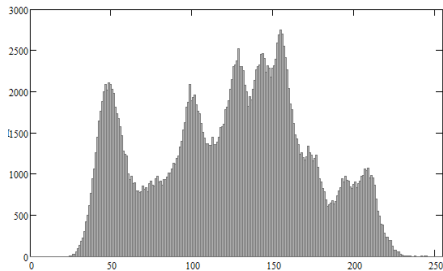
# PORÓWNANIE RÓŻNYCH METOD DITHERINGU

K=4 (CZTERY POZIOMY NA URZĄDZENIU )



- 16 poziomów szarości
- Rozpraszanie średnie
- Rozpraszanie losowe
- Uporządkowane drzenie
- Metoda propagacji błędu

# LENA



- Równomierne histogramy.
- Wiele różnorodnych detali: jednolite obszary, cieniowanie, tekstury – bardzo dobrze nadające się do testowania różnorodnych algorytmów przetwarzania obrazów.
- Symetryczna twarz.
- Piękna dziewczyna



# LENA SÖDERBERG



## 1972-1973

Zdjęcie z rozkładówki magazynu Playboy z listopada 1972 zostało użyte przez Alexandra Sawchuka w 1973 do badań naukowych (University of Southern California Signal and Image Processing Institute - SIPI).

Magazyn Playboy, kiedy zorientował się, że nastąpiło naruszenie praw autorskich do zdjęcia, próbował zakazać używania go bezprawnie, jednak kiedy okazało się to niemożliwe, postanowiono wykorzystać popularność zdjęcia i jednocześnie „przymknąć oko” na jego rozpowszechnianie.

## 1997

Lena mieszka niedaleko Sztokholmu. Pracuje dla agencji rządowej zajmującej się nadzorowaniem zatrudniania osób niepełnosprawnych. Nazywana jest „pierwszą damą internetu”. Jej zdjęcie jest ciągle popularne. Lena została zaproszona na konferencję IS&T's



## 2. PALETA WYZANACZANA ADAPTACYJNIE

- Algorytm popularnościowy
- Algorytm drzewa ósemkowego (Octree)
- Algorytm medianowy
- Algorytm k-średnich





# ALGORYTM POPULARNOŚCIOWY

## POPULARITY ALGORITHM

- Metoda adaptacyjna
- Jako paletę wynikową wybieramy K najczęściej występujących kolorów
  - reprezentantów można również wyszukać w maksimach lokalnych histogramu wygenerowanego na podstawie obrazu
  - ewentualnie zastosować wstępne jednorodne (podobnie jak w metodzie jednorodnego podziału sześciangu RGB) grupowanie kolorów – w tym przypadku poszukujemy nie koloru a prostopadłościanu z największą liczbą punktów
- Struktury danych (Słownik):
  - Lista – duża złożoność czasowa, nie polecana
  - Tablica indeksowana kolorami – duża złożoność pamięciowa
  - Drzewo RST (binarne lub ósemkowe)

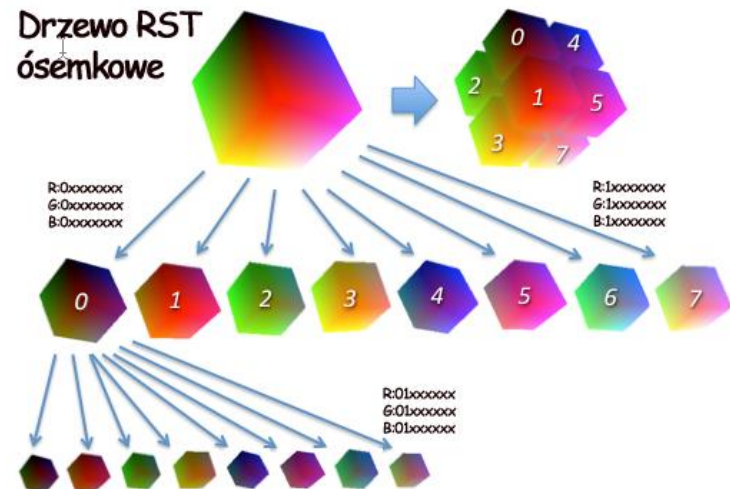


# ALGORYTM DRZEWA ÓSEMkowego

## OCTREE ALGORITHM

- a) Budowane jest drzewo RST kolorów (drzewo ósemkowe).  
Liście znajdują się na 8 poziomie

```
void InsertTree (Octree tree, Color RGB)
{
    if (Tree=NULL) NewAndInit(tree);
    if (Leaf)
    {
        tree->color_count++;
        AddColors(tree->RGB, RGB);
    }
    else
        InsertTree (Next [Branch (RGB) ], RGB);
}
```



b) Redukcja drzewa, aż liczba liści osiągnie zadaną liczbę kolorów

○ **Kryteria redukcji:**

- Redukcję zaczynamy od najniższych poziomów. Liście zastępujemy ich bezpośrednim wspólnym przodkiem.
- Jeśli istnieje kilka węzłów do redukcji na najniższym poziomie – wybieramy ten węzeł, który reprezentuje najmniejszą liczbę pikseli



### c) Tworzenie palety wynikowej

- Wynikowe kolory = liście (liczba liści  $\leq K$ )

### d) Mapowanie kolorów do wynikowej palety

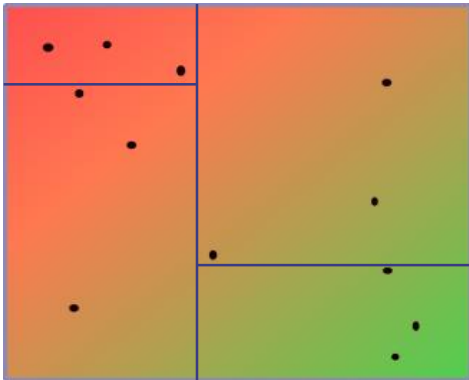
- Obraz jest czytany ponownie. Dla każdego koloru wejściowego jest poszukiwany wynikowy kolor (liść) w zredukowanym drzewie Octree.



# ALGORYTM MEDIANOWY

## THE MEDIAN CUT ALGORITHM

- *Tworzony jest taki podział sześciangu RGB na  $k$  prostopadłościów, aby każdy z nich, zawierał w przybliżeniu taką samą liczbę pikseli obrazu wejściowego*

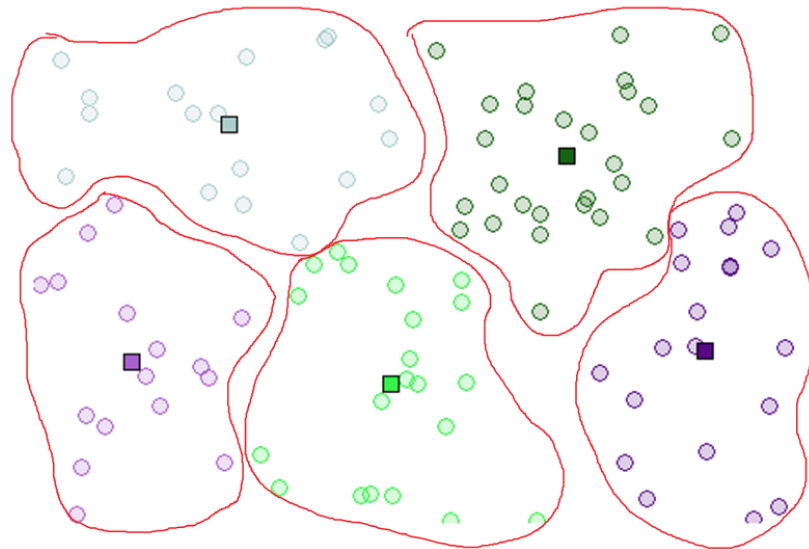




# ALGORYTM K-ŚREDNICH (CENTROIDÓW)

## K-MEANS ALGORITHM

- Tworzenie (np. losowo)  $k$  centroidów, wstępnie reprezentujących skupiska,
- Przyporządkowywanie do nich najbliższych im kolorów z obrazu wejściowego.
- W każdej nowopowstałej grupie wyznaczany jest, na podstawie środka ciężkości, nowy centroid.
- Dla tak określonych środków nowych grup ponownie przyporządkowywane są kolory.
- Proces ten jest powtarzany do chwili, gdy centroidy w kolejnej iteracji nie zmieniają się.



# REDUKCJA KOLORÓW VS. DITHERING

- Wynikowa liczba kolorów = 8
- 2 poziomy na każdy kanał

Obraz oryginalny (TrueColor)



Redukcja kolorów (met. Octree)



Dithering (met. Propagacji błędów)





# REDUKCJA KOLORÓW VS. DITHERING

- Wynikowa liczba kolorów = 256
- 6R/7G/6B – liczby poziomów

Obraz oryginalny (TrueColor)



Redukcja kolorów (met. Octree)



Dithering (met. Propagacji błędów)

