

# Joint Photographic Experts Group

Artur Drozd

Uniwersytet Jagielloński

14 maja 2010

- 1 Co to jest JPEG? Dlaczego powstał?
- 2 Transformata Fouriera
- 3 Dyskretna transformata kosinusowa (DCT-II)
- 4 Kodowanie Huffmana
- 5 Algorytm JPEG
  - Transformacja kolorów i redukcja rozdzielczości
  - Rozłożenie obrazu na bloki
  - DCT-II
  - Kwantyzacja
  - Przetastawienie zygzakowate macierzy i kodowanie Huffmana
- 6 Dekompresja
- 7 Przykład

## Co to jest JPEG? Dlaczego powstał?

## Co to jest JPEG?

**JPEG** - format **kompresji stratnej** statycznych obrazów rastrowych, przeznaczony głównie do przetwarzania obrazów naturalnych (pejzaży, portretów itp.), charakteryzujących się płynnymi przejściami barw oraz brakiem lub małą ilością ostrych krawędzi i drobnych detali.

## Dlaczego powstał?

Motywacją do powstania tego standardu było ujednoczenie algorytmów kompresji obrazów monochromatycznych i kolorowych

Prace nad standardem rozpoczęły się w kwietniu 1983 roku w organizacji ISO. W 1986 roku z inicjatywy ISO oraz CCITT powstał zespół ekspertów nazwany **Joint Photographic Experts Group** (ang. Połączona Grupa Ekspertów Fotograficznych), który zjednoczył wysiłki różnych niezależnych grup.

Standard o nazwie ISO/IEC IS 10918-1 ITU-T Recommendation T.81 został opublikowany w dwóch częściach w 1991 roku i definiował podstawowy, sekwencyjny tryb kompresji stratnej, oparty na dyskretnej transformacji kosinusowej (DCT) oraz jego różne rozszerzenia.

## Co to jest JPEG?

**JPEG** - format **kompresji stratnej** statycznych obrazów rastrowych, przeznaczony głównie do przetwarzania obrazów naturalnych (pejzaży, portretów itp.), charakteryzujących się płynnymi przejściami barw oraz brakiem lub małą ilością ostrych krawędzi i drobnych detali.

## Dlaczego powstał?

Motywacją do powstania tego standardu było ujednoczenie algorytmów kompresji obrazów monochromatycznych i kolorowych

Prace nad standardem rozpoczęły się w kwietniu 1983 roku w organizacji ISO. W 1986 roku z inicjatywy ISO oraz CCITT powstał zespół ekspertów nazwany **Joint Photographic Experts Group** (ang. Połączona Grupa Ekspertów Fotograficznych), który zjednoczył wysiłki różnych niezależnych grup.

Standard o nazwie ISO/IEC IS 10918-1 ITU-T Recommendation T.81 został opublikowany w dwóch częściach w 1991 roku i definiował podstawowy, sekwencyjny tryb kompresji stratnej, oparty na dyskretnej transformacji kosinusowej (DCT) oraz jego różne rozszerzenia.

## Co to jest JPEG?

**JPEG** - format **kompresji stratnej** statycznych obrazów rastrowych, przeznaczony głównie do przetwarzania obrazów naturalnych (pejzaży, portretów itp.), charakteryzujących się płynnymi przejściami barw oraz brakiem lub małą ilością ostrych krawędzi i drobnych detali.

## Dlaczego powstał?

Motywacją do powstania tego standardu było ujednoczenie algorytmów kompresji obrazów monochromatycznych i kolorowych

Prace nad standardem rozpoczęły się w kwietniu 1983 roku w organizacji ISO. W 1986 roku z inicjatywy ISO oraz CCITT powstał zespół ekspertów nazwany **Joint Photographic Experts Group** (ang. Połączona Grupa Ekspertów Fotograficznych), który zjednoczył wysiłki różnych niezależnych grup.

Standard o nazwie ISO/IEC IS 10918-1 ITU-T Recommendation T.81 został opublikowany w dwóch częściach w 1991 roku i definiował podstawowy, sekwencyjny tryb kompresji stratnej, oparty na dyskretnej transformacji kosinusowej (DCT) oraz jego różne rozszerzenia.

## Co to jest JPEG?

**JPEG** - format **kompresji stratnej** statycznych obrazów rastrowych, przeznaczony głównie do przetwarzania obrazów naturalnych (pejzaży, portretów itp.), charakteryzujących się płynnymi przejściami barw oraz brakiem lub małą ilością ostrych krawędzi i drobnych detali.

## Dlaczego powstał?

Motywacją do powstania tego standardu było ujednoczenie algorytmów kompresji obrazów monochromatycznych i kolorowych

Prace nad standardem rozpoczęły się w kwietniu 1983 roku w organizacji ISO. W 1986 roku z inicjatywy ISO oraz CCITT powstał zespół ekspertów nazwany **Joint Photographic Experts Group** (ang. Połączona Grupa Ekspertów Fotograficznych), który zjednoczył wysiłki różnych niezależnych grup.

Standard o nazwie ISO/IEC IS 10918-1 ITU-T Recommendation T.81 został opublikowany w dwóch częściach w 1991 roku i definiował podstawowy, sekwencyjny tryb kompresji stratnej, oparty na dyskretnej transformacie kosinusowej (DCT) oraz jego różne rozszerzenia.

## Transformata Fouriera



## Transformata Fouriera

Niech  $f: \mathbb{R}^n \rightarrow \mathbb{R} \in L^1(\mathbb{R}^n)$ , definiujemy **transformatę Fouriera** jako

$$F(\xi) = \int_{-\infty}^{+\infty} f(x)e^{-2\pi i x \xi} dx$$

oraz odwrotną **transformatę Fouriera**

$$f(x) = \int_{-\infty}^{+\infty} F(\xi)e^{2\pi i x \xi} d\xi$$

Jeżeli zmienna  $x$  oznacza czas (s) to zmienna  $\xi$  oznacza częstotliwość (Hz).

## Transformata Fouriera

Niech  $f: \mathbb{R}^n \rightarrow \mathbb{R} \in L^1(\mathbb{R}^n)$ , definiujemy **transformatę Fouriera** jako

$$F(\xi) = \int_{-\infty}^{+\infty} f(x)e^{-2\pi i x \xi} dx$$

oraz odwrotną **transformatę Fouriera**

$$f(x) = \int_{-\infty}^{+\infty} F(\xi)e^{2\pi i x \xi} d\xi$$

Jeżeli zmienna  $x$  oznacza **czas** (s) to zmienna  $\xi$  oznacza **częstotliwość** (Hz).

Dla przypadku dyskretnego mamy i sygnału jednowymiarowego:

### Jednowymiarowa DFT

Dla ciągu liczb rzeczywistych  $f_0, \dots, f_{N-1}$  **1D DFT** nazywamy  $N$  liczb  $F_0, \dots, F_{N-1}$ , gdzie

$$F_k = \sum_{j=0}^{N-1} f_j e^{-\frac{2i\pi jk}{N}}, \quad k = 0, \dots, N-1$$

oraz **1D IDFT**

$$f_k = \frac{1}{N} \sum_{j=0}^{N-1} F_j e^{\frac{2i\pi jk}{N}}, \quad k = 0, \dots, N-1$$

## Dwuwymiarowa DFT

Dla sygnału dwuwymiarowego reprezentowanego przez macierz rzeczywistą  $f \in \mathcal{M}(M \times N)$  mamy **2D DFT**

$$F_{xy} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f_{xy} e^{\frac{-2i\pi mx}{M} + \frac{-2i\pi ny}{N}}$$

oraz **2D IDFT**

$$f_{xy} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F_{xy} e^{\frac{2i\pi mx}{M} + \frac{2i\pi ny}{N}}$$

## Dyskretna transformata kosinusowa (DCT-II)

**DCT-II** jest szczególnym przypadkiem **2D DFT**. Transformata ta nie zajmuje się składowymi sinusowymi.

### DCT-II

Dla  $f \in \mathcal{M}(N \times N)$  mamy **DCT-II**

$$F_{uv} = \alpha_u \alpha_v \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f_{xy} \cos\left(\frac{(2x+1)\pi u}{2N}\right) \cos\left(\frac{(2y+1)\pi v}{2N}\right)$$

oraz **IDCT-II**

$$f_{xy} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha_u \alpha_v F_{uv} \cos\left(\frac{(2x+1)\pi u}{2N}\right) \cos\left(\frac{(2y+1)\pi v}{2N}\right)$$

gdzie

$$\alpha_t = \begin{cases} \sqrt{\frac{1}{N}}, & t = 0 \\ \sqrt{\frac{2}{N}}, & t \neq 0 \end{cases}$$



- Większość współczynników jest zwykle **bliska 0**.
- Przekształcenie wielowymiarowe DCT-II jest **separowalne** tzn. możliwe do uzyskania poprzez odpowiednie przekształcenia na poszczególnych wymiarach.

- $F_{00}$  jest składową stałą sygnału
- DCT ma **cechę skupiania energii** tzn. Współczynniki  $F_{uv}$  odpowiadające amplitudom wolnych zmian jasności znajdują się w lewej górnej części macierzy  $F$  (czyli nad przekątną), natomiast współczynniki odpowiadające szybkim zmianom jasności znajdują się w prawej dolnej części (pod przekątną).



- Większość współczynników jest zwykle **bliska 0**.
  - Przekształcenie wielowymiarowe DCT-II jest **separowalne** tzn. możliwe do uzyskania poprzez odpowiednie przekształcenia na poszczególnych wymiarach.
- 
- $F_{00}$  jest składową stałą sygnału
  - DCT ma **cechę skupiania energii** tzn. Współczynniki  $F_{uv}$  odpowiadające amplitudom wolnych zmian jasności znajdują się w lewej górnej części macierzy  $F$  (czyli nad przekątną), natomiast współczynniki odpowiadające szybkim zmianom jasności znajdują się w prawej dolnej części (pod przekątną).



## Kodowanie Huffmana

Dany jest alfabet źródłowy  $S = \{x_1, \dots, x_n\}$ , niech prawdopodobieństwo wystąpienia symbolu  $x_j$  wynosi  $p_j$ . **Kodowanie Huffmana** polega na utworzeniu słów kodowych (ciągów bitowych), których długość jest odwrotnie proporcjonalna do prawdopodobieństwa (tj. im częściej symbol występuje, tym krótszy jest jego kod).

#### Własności

- jest kodem prefikсовym; oznacza to, że żadne słowo kodowe nie jest początkiem innego słowa
- średnia długość kodów jest najmniejsza spośród kodów prefikсовych
- jeśli prawdopodobieństwa są różne, tzn.  $p_j > p_i$ , to długość kodu dla symbolu  $x_j$  jest nie mniejsza od kodu dla symbolu  $x_i$
- słowa kodu dwóch najmniej prawdopodobnych symboli mają równą długość
- dwa słowa kodowe o tej samej długości różnią się tylko jednym, ostatnim bitem

Dany jest alfabet źródłowy  $S = \{x_1, \dots, x_n\}$ , niech prawdopodobieństwo wystąpienia symbolu  $x_j$  wynosi  $p_j$ . **Kodowanie Huffmana** polega na utworzeniu słów kodowych (ciągów bitowych), których długość jest odwrotnie proporcjonalna do prawdopodobieństwa (tj. im częściej symbol występuje, tym krótszy jest jego kod).

### Własności

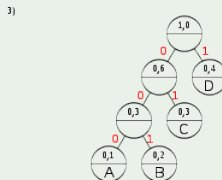
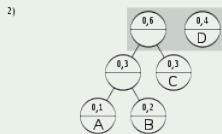
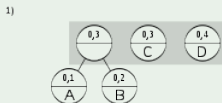
- jest kodem prefiksowym; oznacza to, że żadne słowo kodowe nie jest początkiem innego słowa
- średnia długość kodów jest najmniejsza spośród kodów prefiksowych
- jeśli prawdopodobieństwa są różne, tzn.  $p_j > p_i$ , to długość kodu dla symbolu  $x_j$  jest nie mniejsza od kodu dla symbolu  $x_i$
- słowa kodu dwóch najmniej prawdopodobnych symboli mają równą długość
- dwa słowa kodowe o tej samej długości różnią się tylko jednym, ostatnim bitem

## Algorytm Huffmana

- 1 Określ prawdopodobieństwo (lub częstość występowania) dla każdego symbolu ze zbioru  $S$ .
- 2 Utwórz listę drzew binarnych, które w węzłach przechowują pary: symbol, prawdopodobieństwo. Na początku drzewa składają się wyłącznie z korzenia.
- 3 Dopóki na liście jest więcej niż jedno drzewo, powtarzaj:
  - 1 Usuń z listy dwa drzewa o najmniejszym prawdopodobieństwie zapisanym w korzeniu.
  - 2 Wstaw nowe drzewo, w którego korzeniu jest suma prawdopodobieństw usuniętych drzew, natomiast one same stają się jego lewym i prawym poddrzewem. Korzeń drzewa nie przechowuje symbolu.

Drzewo, które pozostanie na liście, jest nazywane **drzewem Huffmana**.  
Prawdopodobieństwo zapisane w korzeniu jest równe 1, natomiast w liściach drzewa zapisane są symbole.

## Przykład



Mamy następujące elementy:  
 $(A; 0, 1)$ ,  $(B; 0, 2)$ ,  $(C; 0, 3)$ ,  $(D; 0, 4)$ , wykonujemy następujące kroki:

- 1 Łączymy A i B
- 2 Łączymy drzewo AB z C
- 3 Łączymy drzewo ABC z D

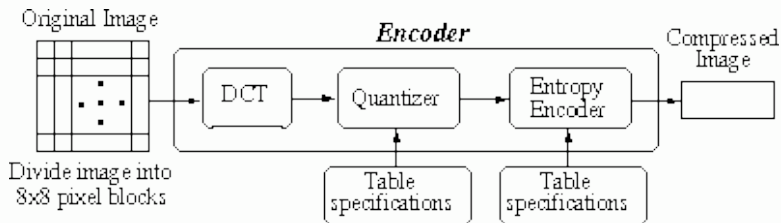
Otrzymaliśmy następujące kody:

- A = 000
- B = 001
- C = 01
- D = 1

## Algorytm JPEG

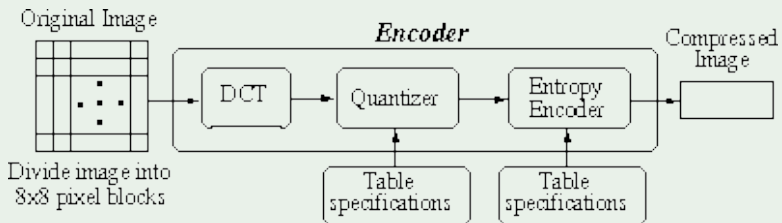
## Schemat algorytmu

- 1 Transformacja kolorów i redukcja rozdzielczości
- 2 Rozłożenie obrazu na bloki
- 3 DCT-II na każdym bloku
- 4 Kwantyzacja
- 5 Przetastawienie zygzakowate macierzy i kodowanie Huffmana



## Schemat algorytmu

- 1 Transformacja kolorów i redukcja rozdzielczości
- 2 Rozłożenie obrazu na bloki
- 3 DCT-II na każdym bloku
- 4 Kwantyzacja
- 5 Przetworzenie zygzakowate macierzy i kodowanie Huffmana





Obraz  $\Leftrightarrow$  macierz pikseli.

### Grayscale

- 8 bitów na piksel
- Wyrównanie względem zera poprzez odjęcie 128 (dostajemy liczby z  $[-128, 127]$ )

### RGB

- 24 bity na piksel (po 8 na kolor)
- Konwersja  $RGB \rightarrow YCbCr$  (ponieważ ludzkie oko jest bardziej wrażliwe na zmiany jasności niż zmiany kolorów) zgodnie ze wzorem:

$$\begin{cases} Y = 0,299R + 0,587G + 0,114B \\ Cb = 128 + 0,5R - 0,418688G - 0,081312B \\ Cr = 128 - 0,168736R - 0,331264G - 0,53B \end{cases}$$

gdzie  $Y$  jest składową **luminacji**, a  $Cb$  i  $Cr$  to składowe barwy (**chrominacji**) [odpowiednio antyniebieska i antyczerwona].

- Zmiana próbkowania składowych **chrominacyjnych**
  - format 4:4:4 - składowe chrominacji bez zmian.
  - format 4:2:2 - rozdzielczość pozioma  $Cb$  i  $Cr$  zmniejszana jest o połowę
  - format 4:2:0 - rozdzielczość  $Cb$  i  $Cr$  zmniejszana jest o połowę

Obraz  $\Leftrightarrow$  macierz pikseli.

### Grayscale

- 8 bitów na piksel
- Wyrównanie względem zera poprzez odjęcie 128 (dostajemy liczby z  $[-128, 127]$ )

### RGB

- 24 bity na piksel (po 8 na kolor)
- Konwersja  $RGB \rightarrow YCbCr$  (ponieważ ludzkie oko jest bardziej wrażliwe na zmiany jasności niż zmiany kolorów) zgodnie ze wzorem:

$$\begin{cases} \mathbf{Y} = 0,299\mathbf{R} + 0,587\mathbf{G} + 0,114\mathbf{B} \\ \mathbf{Cb} = 128 + 0,5\mathbf{R} - 0,418688\mathbf{G} - 0,081312\mathbf{B} \\ \mathbf{Cr} = 128 - 0,168736\mathbf{R} - 0,331264\mathbf{G} - 0,53\mathbf{B} \end{cases}$$

gdzie  $\mathbf{Y}$  jest składową **luminacji**, a  $\mathbf{Cb}$  i  $\mathbf{Cr}$  to składowe barwy (**chrominacji**) [odpowiednio antyniebieska i antyczerwona].

- Zmiana próbkowania składowych **chrominacyjnych**
  - format **4:4:4** - składowe chrominacji bez zmian.
  - format **4:2:2** - rozdzielczość pozioma  $\mathbf{Cb}$  i  $\mathbf{Cr}$  zmniejszana jest o połowę
  - format **4:2:0** - rozdzielczość  $\mathbf{Cb}$  i  $\mathbf{Cr}$  zmniejszana jest o połowę

## Grayscale

- Podział obrazka na bloki  $8 \times 8$  pikseli.

## RGB

- Każda składowa **Y**, **Cb**, **Cr** dzielona jest na bloki  $8 \times 8$  pikseli.
- W zależności od formatu (4:x:y) składowe **Cb** i **Cr** mogą zajmować  $8 \times 8$ ,  $16 \times 8$ ,  $16 \times 16$  oryginalnego obrazka.

## Uwaga

Wymiary obrazka nie muszą być wielokrotnościami 8 (16), wtedy uzupełniamy odpowiednio z prawej strony i z dołu do wielokrotności 8 (16) na kilka sposobów:

- uzupełnienie zerami (najprościej, ale najgorzej)
- uzupełnienie kolorem ostatniego piksela
- uzupełnienie takimi pikselami aby **IDCT** zwróciła wartości jak najbliższe wartością oryginalnym.

## Grayscale

- Podział obrazka na bloki  $8 \times 8$  pikseli.

## RGB

- Każda składowa **Y**, **Cb**, **Cr** dzielona jest na bloki  $8 \times 8$  pikseli.
- W zależności od formatu (4:x:y) składowe **Cb** i **Cr** mogą zajmować  $8 \times 8$ ,  $16 \times 8$ ,  $16 \times 16$  oryginalnego obrazka.

## Uwaga

Wymiary obrazka nie muszą być wielokrotnościami 8 (16), wtedy uzupełniamy odpowiednio z prawej strony i z dołu do wielokrotności 8 (16) na kilka sposobów:

- uzupełnienie zerami (najprościej, ale najgorzej)
- uzupełnienie kolorem ostatniego piksela
- uzupełnienie takimi pikselami aby **IDCT** zwróciła wartości jak najbliższe wartościom oryginalnym.

## Grayscale

- Podział obrazka na bloki  $8 \times 8$  pikseli.

## RGB

- Każda składowa **Y**, **Cb**, **Cr** dzielona jest na bloki  $8 \times 8$  pikseli.
- W zależności od formatu (4:x:y) składowe **Cb** i **Cr** mogą zajmować  $8 \times 8$ ,  $16 \times 8$ ,  $16 \times 16$  oryginalnego obrazka.

## Uwaga

Wymiary obrazka nie muszą być wielokrotnościami 8 (16), wtedy uzupełniamy odpowiednio z prawej strony i z dołu do wielokrotności 8 (16) na kilka sposobów:

- uzupełnienie zerami (najprościej, ale najgorzej)
- uzupełnienie kolorem ostatniego piksela
- uzupełnienie takimi pikselami aby **IDCT** zwróciła wartości jak najbliższe wartości oryginalnym.

- Na każdym bloku  $8 \times 8$  wykonujemy DTC
- Otrzymujemy 16 bitową macierz  $F \in \mathcal{M}(8 \times 8)$
- $F_{00}$  nazywany jest współczynnikiem DC
- Pozostałe 63 współczynniki nazywane są współczynnikami AC

- Współczynniki macierzy  $F_{uv}$  poddawane są kwantyzacji liniowej przy użyciu macierzy kwantyzacji

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

- Otrzymujemy macierz współczynników  $B_{uv} = \text{round} \left( \frac{F_{uv}}{Q_{uv}} \right)$ ,  $u, v = 0, \dots, 7$  po kwantyzacji, gdzie  $\text{round}(x)$  oznacza najmniejszą liczbę całkowitą większą bądź równą liczbie rzeczywistej  $x$ .
- Kwantyzacja zeruje większość współczynników odpowiedzialnych za wysokie częstotliwości (pod przekątną), ponieważ oko ludzkie jest bardziej czułe na niskie częstotliwości (mała wrażliwość na amplitudę szybkich zmian jasności).

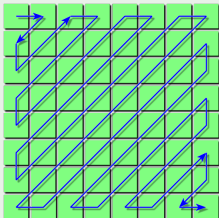
## Ustalanie poziomu kompresji

- W JPEG można ustalić poziom jakości  $QL$  (*quality level*) - liczba naturalna od 1 do 100.
- Mniejszy  $QL \Leftrightarrow$  gorsza jakość  $\Leftrightarrow$  lepszy współczynnik kompresji
- Zmiana współczynnika jakości polega na manipulacji macierzą kwantyzacji.
- $QL = 50$  - najlepsze wyważenie pomiędzy jakością obrazu, a stopniem kompresji
- Niech  $Q_{QL}$  oznacza macierz kwantyzacji przy  $QL$ . Przyjmujemy
  - $Q_{50} = Q$
  - $Q_{100} = ones(8, 8)$
  - $QL > 50 : Q_{QL} = \frac{(100-QL)}{50} Q_{50}$
  - $QL < 50 : Q_{QL} = \frac{50}{QL} Q_{50}$

Po wymnożeniu współczynniki macierzy  $Q_{QL}$  są obcinane do zakresu  $[1, 255]$



- Ustawiamy współczynniki w wektor (mniej więcej) rosnąco względem częstotliwości.

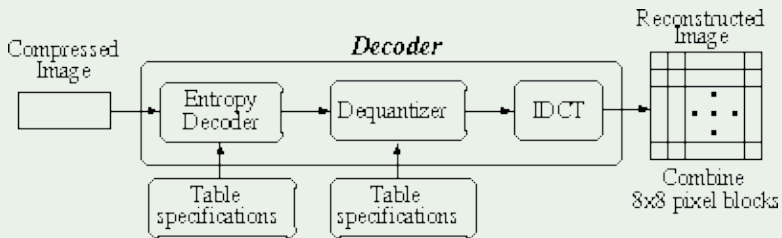


$$\begin{bmatrix} 0 & 1 & 5 & 6 & 14 & 15 & 27 & 28 \\ 2 & 4 & 7 & 13 & 16 & 26 & 29 & 42 \\ 3 & 8 & 12 & 17 & 25 & 30 & 41 & 43 \\ 9 & 11 & 18 & 24 & 31 & 40 & 44 & 53 \\ 10 & 19 & 23 & 32 & 39 & 45 & 52 & 54 \\ 20 & 22 & 33 & 38 & 46 & 51 & 55 & 60 \\ 21 & 34 & 37 & 47 & 50 & 56 & 59 & 61 \\ 35 & 36 & 48 & 49 & 57 & 58 & 62 & 63 \end{bmatrix}$$

- W procesie kwantyzacji większość współczynników odpowiedzialnych za wysokie częstotliwości została zaokrąglona do zera. Dzięki temu nasz wektor od pewnego momentu będzie zawierał tylko zerowe elementy, zastępujemy niepotrzebne zera specjalnym symbolem EOB.
- Tak obcięty wektor jest następnie kompresowany metodą Huffmana.

# Dekompresja

Dekompresja polega na wykonaniu tych samych kroków co w kompresji tylko w odwrotnej kolejności, czyli



Przykład

→ Octave

### Użyte materiały

- [http://pl.wikipedia.org/wiki/Kodowanie\\_Huffmana](http://pl.wikipedia.org/wiki/Kodowanie_Huffmana)
- <http://pl.wikipedia.org/wiki/Jpeg>
- <http://en.wikipedia.org/wiki/JPEG>
- [www.cmmsigma.eu/download/mat\\_dsp/kompresja\\_jpeg.pdf](http://www.cmmsigma.eu/download/mat_dsp/kompresja_jpeg.pdf)

Dziękuję za uwagę.