

# Kompresja – kodowanie transformacyjne (*transform coding*)

Przemysław Śliwiński

May 24, 2018

## 1 Zadanie

- Zaprojektować własny kodek (koder-dekoder) **kompresji bezstratnej** obrazów rastrowych wykorzystując wybrane narzędzia:
  - transformatę kolorów ( $\mathbf{RGB} \iff \mathbf{Y} \mathbf{C}_B \mathbf{C}_R$ )
  - transformatę kosinusową (o wybranym rozmiarze bloku, np. **8x8**, **16x16**, **32x32**)
  - transformatę falkową (z wybraną falką, np. **Haara**, **LeGalla 5/3**, **CDF 9/7**)
  - transformatę Walsha-Hadamarda (o wybranym rozmiarze bloku, np. **8x8**, **16x16**, **32x32**)
  - kodowanie arytmetyczne (**adaptacyjne** lub nie)
  - kodowanie Huffmana (**adaptacyjne** lub nie)
  - kodowanie **LZW**
- Opracować format pliku z rozszerzeniem **.JXX** (**XX** - inicjały Autora) przechowującego skompresowany obraz.
- Porównać efektywność kompresji (rozmiar pliku wyjściowego) z oryginałem i rozmiarami plików innych studentów.<sup>1</sup>

---

<sup>1</sup>Autor najlepszego kodeka (dla którego plik konkursowy będzie najkrótszy – wyznaczy sobie nagrodę – ocenę z laboratorium).



Figure 1: Oryginalny obraz konkursowy znajduje się pod adresem: <http://diuna.iiar.pwr.wroc.pl/sliwinski/dydaktyka/2017-2018/Lato/ZAPDC/Wiosna-winniczka.tif>

## 2 Sugestie i podpowiedzi

### 2.1 DCT + kodowanie Huffmana (á la JPEG)

- Na podanym obrazie rastrowym (4096x2048) dokonać następujących operacji

- transformacja każdego z pikseli z przestrzeni RGB do YCbCr
- podział na bloki 8x8
- Dla każdego bloku i składowej koloru
  - \* transformata DCT 2D w każdym z bloków
  - \* kwantyzacja współczynników transformacji

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

- \* konwersja współczynników AC do wektora za pomocą algorytmu *zig-zag*
- \* kompresja współczynników kodem Huffmana (prawdopodobieństwa poszczególnych wartości współczynników wyznaczone z częstości ich występowania w całym obrazie)
- zastosowanie kodu Huffmana do kompresji współczynników DC każdego z bloków

### 2.2 FWT + kodowanie arytmetyczne (á la JPEG2000)

- Na wybranym obrazie rastrowym (4096x2048) dokonać następujących operacji

- transformacja kolorów pikseli (z RGB do YCbCr)
- pełna transformacja falkowa (falki LeGalla 5/3) 2D całego obrazu dla każdej składowej koloru
- podział przetransformowanego obrazu na bloki
- Dla każdego bloku i składowej koloru
  - \* kompresja współczynników za pomocą kodera arytmetycznego

### 2.3 Dekompresja

- Odtworzyć obrazy wejściowe odwracając kolejność działań
- Porównać obrazy wynikowe z wejściowymi