

# Przetwarzanie obrazów 2

- Handout

SZ 2024/2025

Wydział Matematyki, Fizyki i Informatyki



Dr.-Ing. A. Rowińska-Schwarzsteller  
[agnieszka.rowinska-schwarzsteller@ug.edu.pl](mailto:agnieszka.rowinska-schwarzsteller@ug.edu.pl)

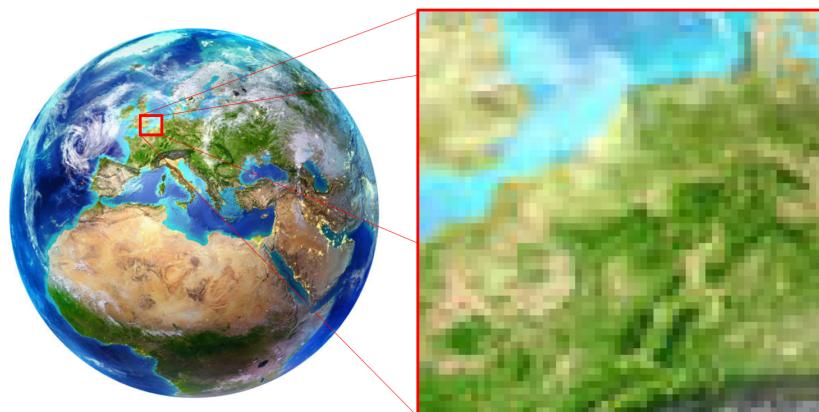
7. Oktober 2024

# Spis treści

<b>4 Digitalizacja obrazów</b>	<b>2</b>
4.1 Podstawowe pojęcia . . . . .	2
4.2 Próbkowanie sceny ( <i>Sampling</i> ) . . . . .	3
4.2.1 Dystrybucja delta . . . . .	3
4.2.2 Funkcja grzebieniowa $\Delta$ (m) . . . . .	4
4.2.3 Sampling: sformułowanie formalne . . . . .	4
4.2.4 Twierdzenie Nyquista–Shannona . . . . .	5
4.2.5 Aliasing . . . . .	5
4.2.6 Digitalizacja w przestrzeni: przykład i podsumowanie . . . . .	5
4.3 Kwantyzacja obrazu . . . . .	7
4.3.1 Prawo Webera-Fechnera . . . . .	7
4.3.2 Kwantyzacja obrazu: przykład i podsumowanie . . . . .	9
4.4 Charakterystyka jakościowa obrazów . . . . .	11
4.5 Kompresja danych, formaty plików . . . . .	11
4.5.1 JPEG . . . . .	11
<b>5 Dyskretyzacja w teorii sygnałów i systemów</b>	<b>21</b>
5.1 Sygnały i systemy . . . . .	21
5.2 Transformacja Fouriera . . . . .	23
5.2.1 Dyskretna transformacja Fouriera, DFT ( <i>Discrete Fourier Transform</i> ) . . . . .	28
5.2.2 Transformata Fouriera w przetwarzaniu obrazu (2D DFT) . . . . .	29
5.2.3 Interpretacja transformaty obrazu . . . . .	39
5.2.4 Niektóre własności transformaty Fouriera . . . . .	41
5.2.5 Filtrowanie obrazu w obszarze widma . . . . .	41
5.2.6 Fast Fourier Transform FFT . . . . .	43
5.2.7 Dyskretna transformacja kosinusowa DCT ( <i>Discrete Cosine Transform</i> ) . . . . .	46

# 4

## Digitalizacja obrazów



Grafika 4.1: Digitalizacja obrazu

### 4.1 Podstawowe pojęcia

#### Digitalizacja

→ ograniczenie wielkości ciągłej (analogowej) do skońzonego (dyskretnego) zbioru wartości.

W systemach przetwarzania obrazów realizowana jest w dwóch formach  
( $\rightarrow$  grafika 4.2):

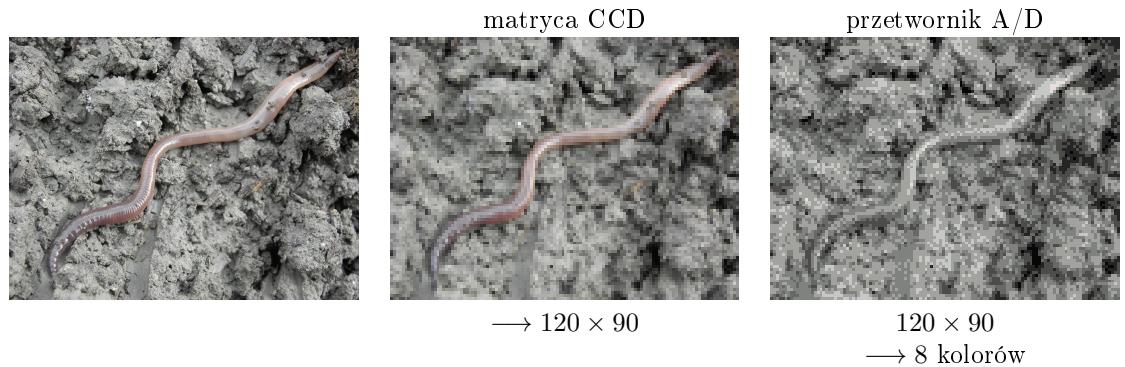
1. Digitalizacja miejsca: próbkowanie, dyskretyzacja, kwantowanie w przestrzeni (*sampling*)  
Matryca CCD/CMOS nie dostarcza sygnału intensywności, który jest ciągły w

przestrzeni, ale raczej wartości dla skończonej liczby pozycji, środków pikseli CCD.

## 2. Digitalizacja intensywności: kwantyzacja

Przetwornik analogowo-cyfrowy (A/D) nie dostarcza ciągłych wartości intensywności na piksel, ale skończony zestaw poziomów szarości.

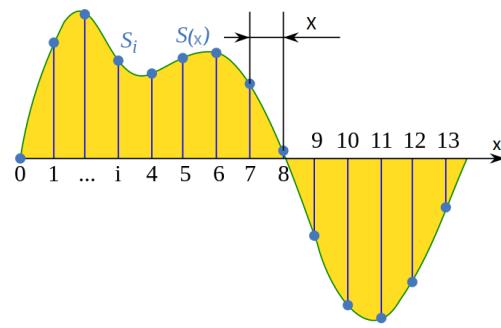
Skrajny przypadek to „binaryzacja” z tylko dwiema wartościami intensywności.



Grafika 4.2: Digitalizacja obrazu

## 4.2 Próbkowanie sceny (*Sampling*)

- Jasność obrazu nie jest mierzona w sposób ciągły w każdym punkcie  $(x, y)$ .
- Pomiar tylko w określonych punktach lub całkowanie w określonych obszarach



- Wyrażenie matematyczne: mnożenie sygnału ciągłego przez ciąg impulsów
- Impuls  $\delta \rightarrow$  delta Diraca, dystrybucja

### 4.2.1 Dystrybucja delta

Dystrybucja delta przypisuje każdej funkcji  $f$ , którą można różniczkować dowolną liczbę razy, liczbę rzeczywistą lub zespoloną, która jest wartością funkcji  $f$  dla argumentu

0:

$$\delta(f) = \int_{\mathbb{R}} \delta(x) f(x) dx = f(0)$$

W fizyce zazwyczaj definicje

$$\delta(x) = \begin{cases} 0 : & x \neq 0 \\ +\infty : & x = 0 \end{cases}$$

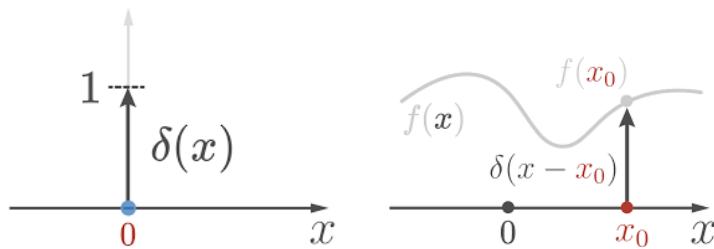
lub

$$\int_{-\infty}^{+\infty} \delta(x) dx = 1$$

$\delta$  jest matematycznym modelem nieskończego wąskiego impulsu dla argumentu  $x = 0$  o nieskończonym dużej amplitudzie i polu równym 1.

Z definicji  $\delta$  wartość funkcji  $f$  dla argumentu  $x_0$  ( $\rightarrow$  grafika 4.3)

$$\int_{-\infty}^{+\infty} \delta(x) f(x - x_0) dx = \int_{-\infty}^{+\infty} \delta(x - x_0) f(x) dx = f(x_0)$$



Grafika 4.3: Dystrybucja delta

#### 4.2.2 Funkcja grzebieniowa $\Delta$ (III)

$\rightarrow$  funkcja grzebieniowa  $\Delta$ , funkcja sza (III), ciąg impulsów Diraka, Dirac comb, sha function ( $\rightarrow$  grafika 4.4)

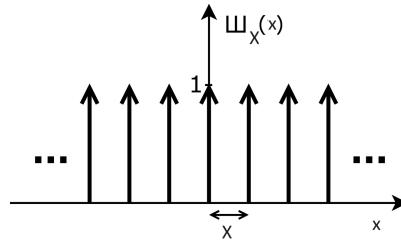
$$\Delta_X(x) = \sum_{n=-\infty}^{\infty} \delta(x - nX)$$

#### 4.2.3 Sampling: sformułowanie formalne

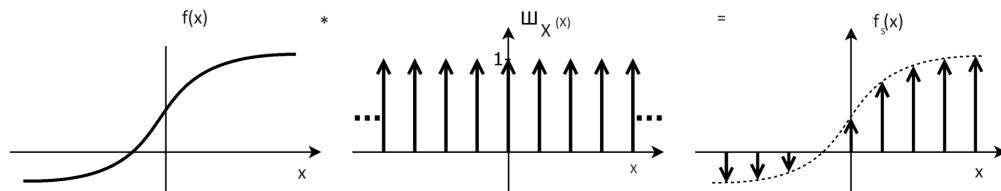
$\rightarrow$  grafika 4.5

Funkcja  $f_s(x)$ : funkcja  $f(x)$  próbkowana w punktach  $nX$  tzn. z częstotliwością  $f_p = \frac{1}{X}$ :

$$\begin{aligned} f_s(x) &= f(x) \cdot \Delta_X(x) \\ &= f(x) \cdot \sum_{n=-\infty}^{\infty} \delta(x - nX) \end{aligned}$$



Grafika 4.4: Funkcja grzebieniowa



Grafika 4.5: Sampling

#### 4.2.4 Twierdzenie Nyquista–Shannona

##### Twierdzenie o próbkowaniu:

Sygnal o maksymalnej składowej częstotliwości  $f_{\max}$  musi być próbkowany z co najmniej dwukrotnie większą częstotliwością  $f_p = 2f_{\max}$ .

→ grafika 4.6

#### 4.2.5 Aliasing

- Nieprawidłowy sygnał spowodowany próbkowaniem o zbyt małej częstotliwości  
→ grafika 4.7

- Częstotliwość sygnału aliasingowego  $f_{\text{alias}}$   
= różnica pomiędzy częstotliwością sygnału oryginalnego i częstotliwością próbkowania

$$f_p < 2f_{\max} \rightarrow f_{\text{alias}} = |f_p - f_{\max}|$$

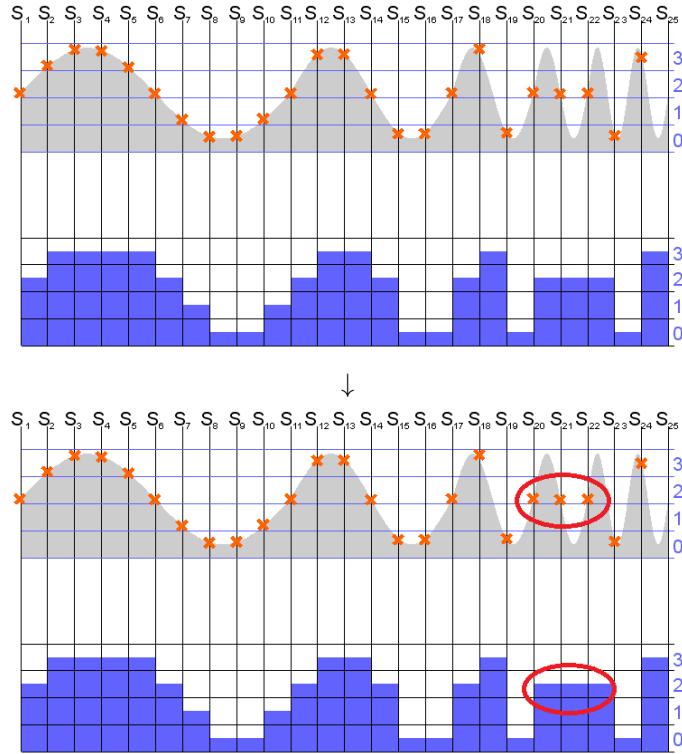
##### Aliasing w obrazach cyfrowych

→ efekt Moiré, grafiki 4.8, 4.9, 4.10

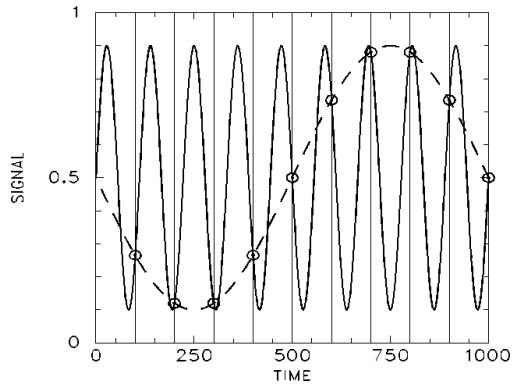
#### 4.2.6 Digitalizacja w przestrzeni: przykład i podsumowanie

→ grafika 4.11

- Człowiek jest stosunkowo wrażliwy na niedopróbkowanie
- Rozdzielcość jest ograniczona przez układ CCD



Grafika 4.6: Próbkowanie sygnału ze zbyt niską częstotliwością  $f_p$ . Źródło: Computergrafik 2: Digitale Bilder & Punktoperationen, M. Rohs



Grafika 4.7: Nieprawidłowy sygnał spowodowany próbkowaniem o zbyt małej częstotliwości

- Nadpróbkowanie nie poprawia wrażenia obrazu
- Zapotrzebowanie na pamięć wzrasta kwadratowo wraz z częstotliwością próbkowania (piksele/długość)



Grafika 4.8: Aliasing w obrazach cyfrowych. Źródła: wikipedia, tvpis.org



Grafika 4.9: Aliasing w obrazach cyfrowych

### 4.3 Kwantyzacja obrazu

→ grafika 4.12 Obrazy cyfrowe są zwykle kwantowane do 256 poziomów szarości.

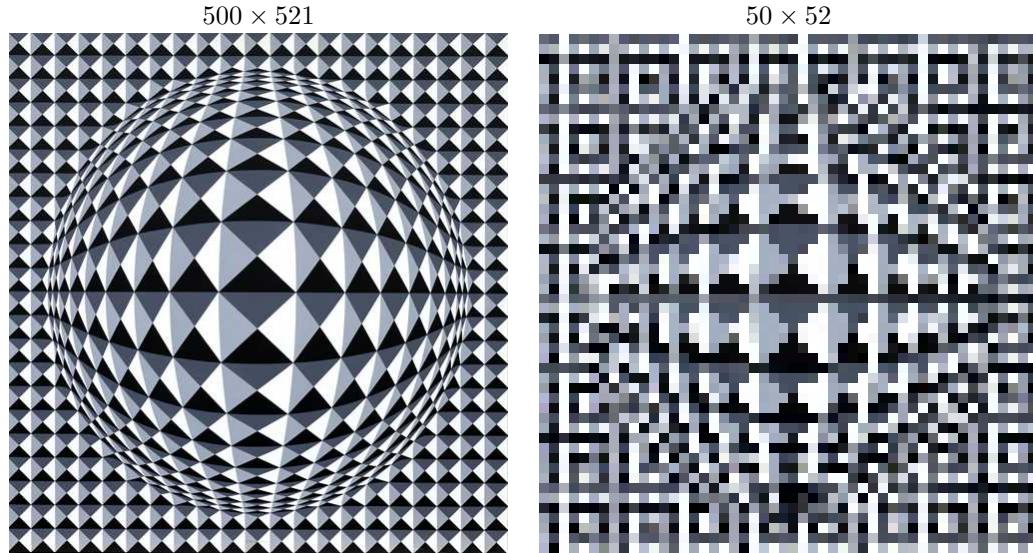
#### 4.3.1 Prawo Webera-Fechnera

→ prawo wyrażające relację pomiędzy fizyczną miarą bodźca a reakcją zmysłów, tu różnicowanie jasności

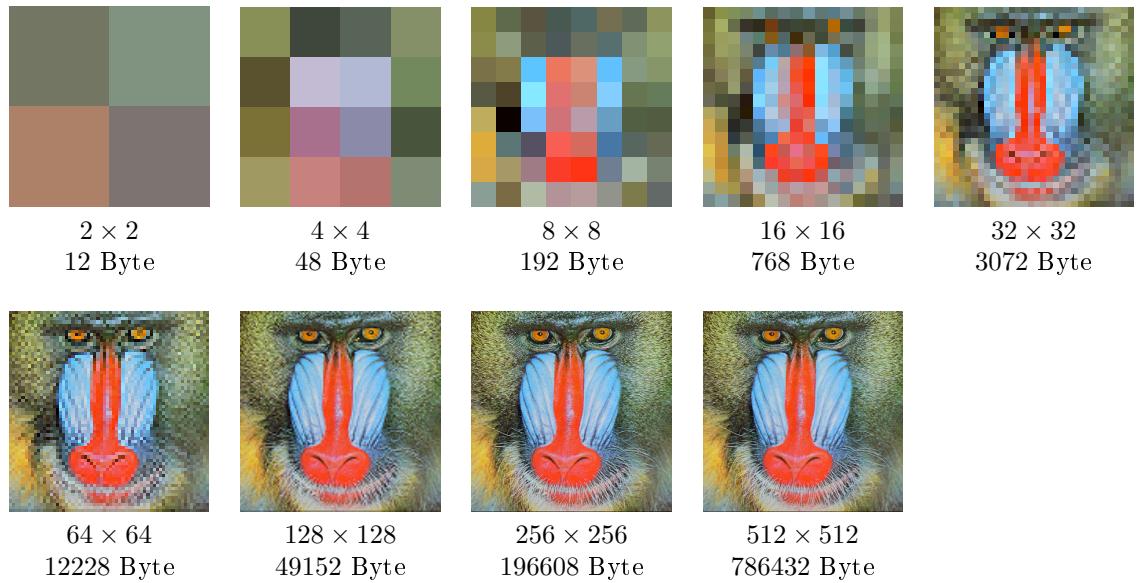
→ grafika 4.13

Próg percepcji różnicowej (just noticeable difference, JND)  $k$ :

$$k = \frac{\Delta I}{I}, k = 1 - 2\% \text{ dla natężenia światła}$$



Grafika 4.10: Aliasing w obrazach cyfrowych

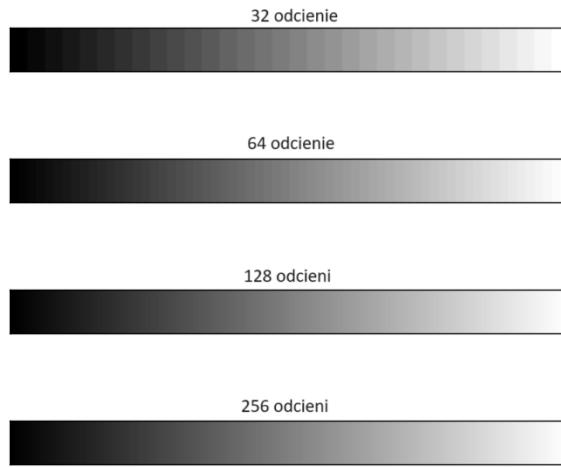


Grafika 4.11: Przykłady digitalizacji w przestrzeni

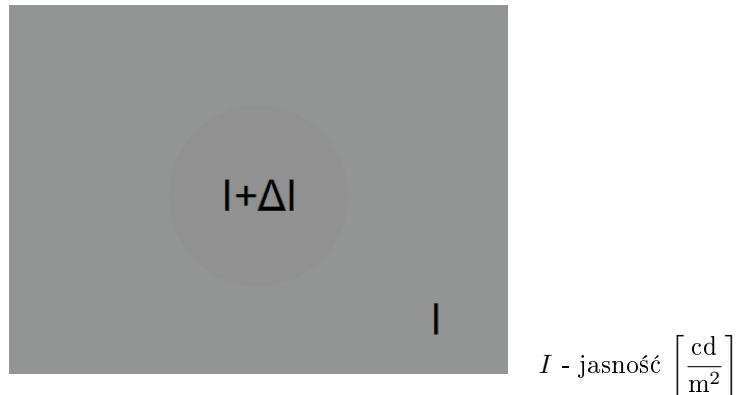
### Jasność w percepkcji wzrokowej

Według ludzkiej percepkcji jasność nie jest wielkością rosnącą liniowo.  
 → grafika 4.14

Jeśli sygnał jasności urządzenia wyświetlającego, na przykład monitora, ma być



Grafika 4.12: Różnicowanie odcieni szarości. Źródło: Computergrafik 2: Digitale Bilder & Punktoperationen, M. Rohs



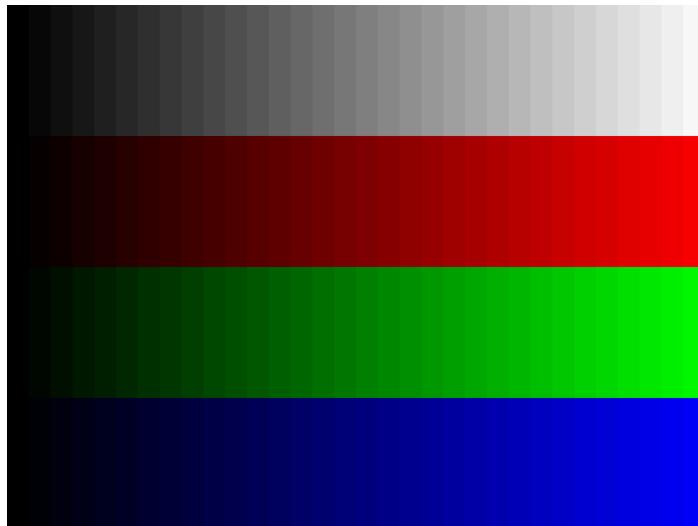
Grafika 4.13: Różnicowanie jasności - prawo Webera-Fechnera

odbierany liniowo, musi zostać nieliniowo skorygowany (np. korekcja  $\gamma$ , rozdział "Statystyka obrazu")  
 → grafika 4.15

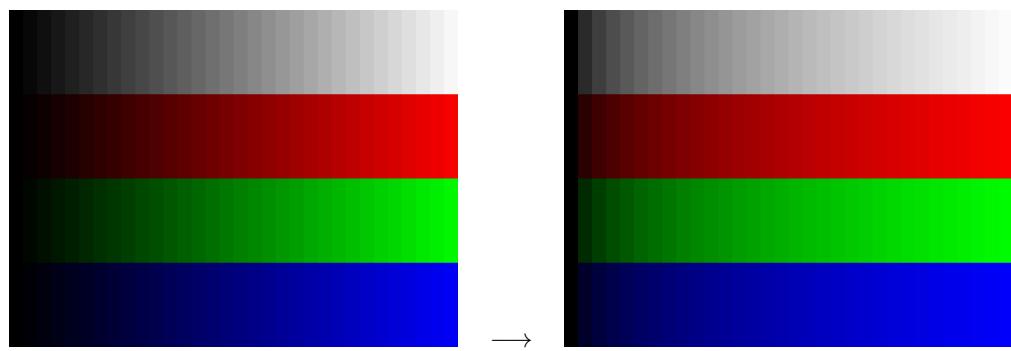
#### 4.3.2 Kwantyzacja obrazu: przykład i podsumowanie

→ grafika 4.16

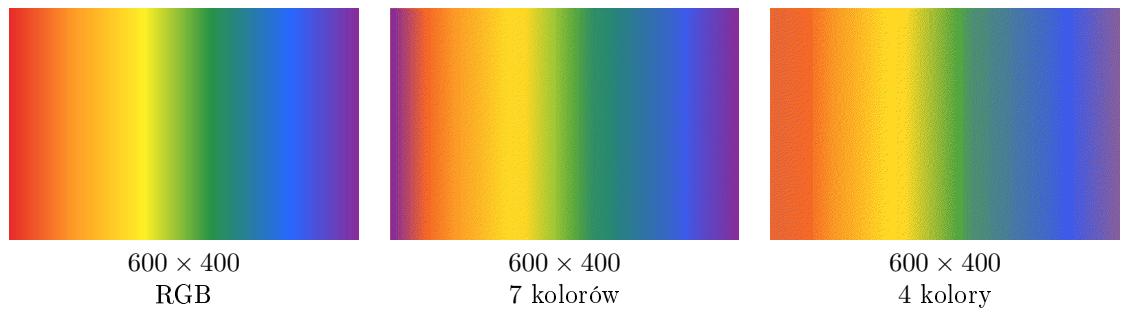
- Ludzie nie są zbyt wrażliwi na zgrubną kwantyzację → diethering
- Rozdzielcość jest zasadniczo ograniczona dostępną pamięcią



Grafika 4.14: Różnicowanie jasności - klin w skali szarości i kliny w kolorach RGB, każdy posiadający 32 pola o liniowo rosnącej jasności



Grafika 4.15: Różnicowanie jasności - korekta wartości jasności.



Grafika 4.16: Kwantyzacja obrazów.

- Typowe rozdzielczości to 24 bity (8 bitów na kolor) na piksel

#### 4.4 Charakterystyka jakościowa obrazów

→ grafika 4.17

- **Kontrast globalny**

Największa różnica wartości szarości na obrazie:

$$c_{\text{global}} = \frac{\max_{mn} f(m, n) - \min_{mn} f(m, n)}{g_{\text{range}}},$$

$g_{\text{range}}$  zakres wartości szarości.

- **Kontrast lokalny**

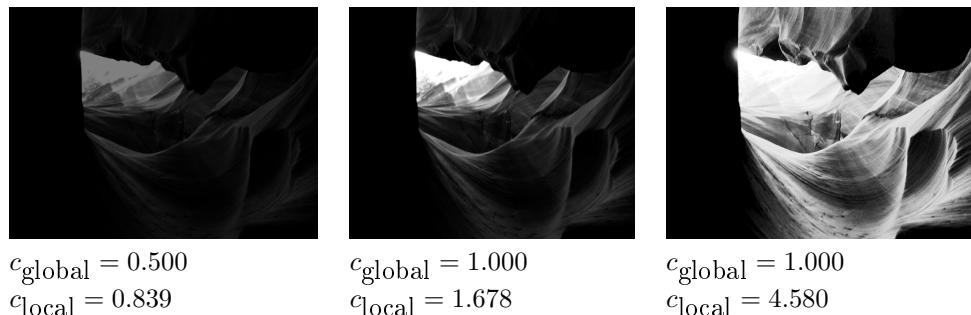
np. średnia różnica wartości szarości pomiędzy sąsiednimi pikselami

$$c_{\text{local}} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |f(m, n) - f_{nb}(n, m)|,$$

$f_{nb}(n, m)$  średnia wartość szarości w obszarze sąsiedztwa  $f(m, n)$

- entropia

→ średnia ilość informacji przypadająca na ciąg pikseli o długości  $N$



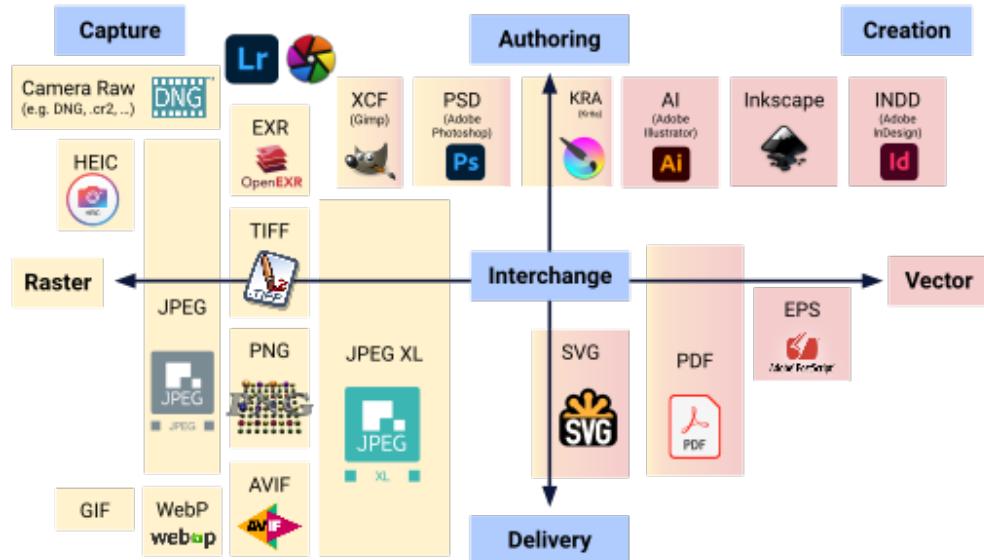
Grafika 4.17: Globalny i lokalny kontrast. Źródło: Computergrafik 2: Digitale Bilder & Punktoperationen, M. Rohs

#### 4.5 Kompresja danych, formaty plików

→ grafika 4.18

##### 4.5.1 JPEG

- norma ISO/IEC 10918-1 oraz CCITT Recommendation T.81
- nazwa "JPEG" wywodzi się od *Joint Photographic Experts Group*, która opracowała standard JPEG



Grafika 4.18: Formaty plików. Źródło: wikipedia

- JPEG oferuje różne metody kompresji i kodowania, w tym kompresję strażną i bezstratną, różne głębie kolorów oraz tryby sekwencyjne lub progresywne (powszechnie stosowana jest kompresja strażna w trybie sekwencyjnym lub progresywnym oraz 8-bitowe kanały kolorów)
- standard JPEG opisuje jedynie metody kompresji obrazu, ale nie określa, w jaki sposób powinny być zapisywane dane wynikowe, "Pliki JPEG" lub "pliki JPG" są zazwyczaj używane w odniesieniu do plików w formacie graficznym *JPEG File Interchange Format (JFIF)*. Jednak JFIF to tylko jeden ze sposobów przechowywania danych JPEG; SPIFF i JNG to inne, choć mniej popularne opcje.
- JPEG/JFIF obsługuje maksymalny rozmiar obrazu  $65\,535 \times 65\,535$  pikseli.
- w starszych wersjach systemu Windows słowo JPG było używane głównie jako skrót JPEG (system pierwotnie obsługiwał tylko trzyliterowe rozszerzenia plików, z tego samego powodu nadal istnieją identyczne formaty TIF i TIFF, MPG i MPEG itp.)
- następca: JPEG 2000  
→ JPEG 2000 jest niekompatybilny z JPEG i mało rozpowszechniony (obawy związane z prawami własnością intelektualnej, brak kompatybilności wstępnej ze starszymi programami i urządzeniami)

Kompresja danych odbywa się w kilku etapach przetwarzania, z których cztery są strażne:

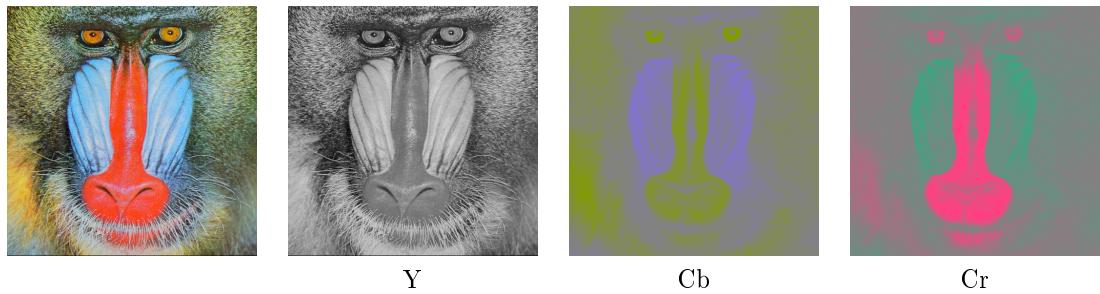
1. konwersja palety barw z (zazwyczaj) przestrzeni barw RGB na model kolorów YCbCr (analogicznie do CCIR 601) (teoretycznie bezstratny, zgodnie ze CCIR 601 strażny).

2. filtrowanie dolnoprzepustowe i podpróbkowanie sygnałów Cb i Cr (stratne).
3. podział na bloki  $8 \times 8$  i dyskretna transformacja kosinusowa tych bloków (teoretycznie bezstratna, ale obarczona stratą na skutek błędów zaokrągleń).
4. kwantyzacja (stratna)
5. przegrupowanie
6. kodowanie entropijne

#### **Konwersja palety barw do modelu różnicowego YCbCr**

→ grafika

- Y – składowa luminancji (jasności),
- Cb – składowa różnicowa chrominancji Y-B,
- Cr – składowa chrominancji Y-R



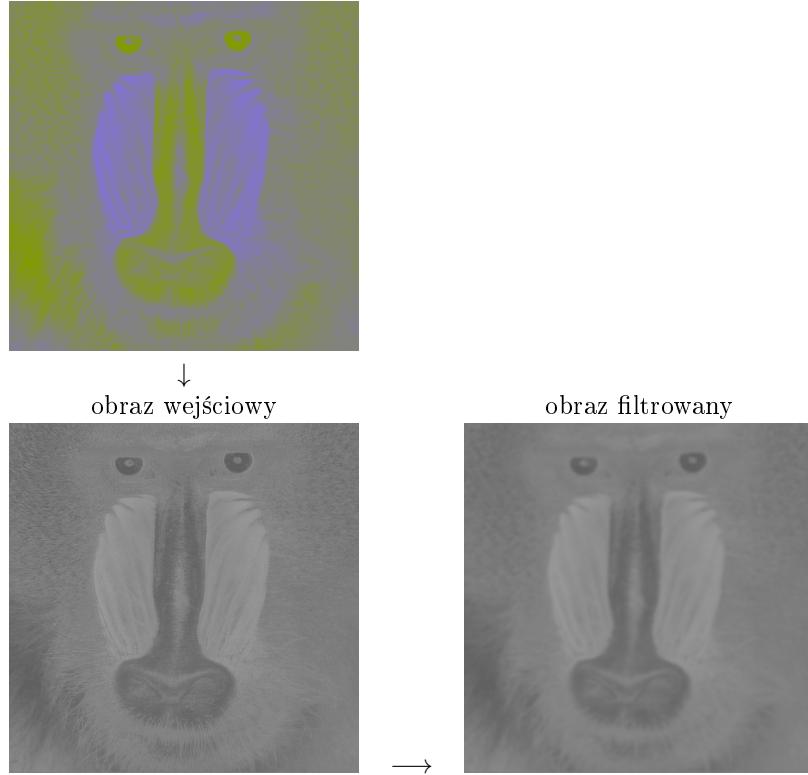
Grafika 4.19: Model przestrzeni kolorów YCbCr.

Podczas konwersji palety barw powstają typowe błędy zaokrągleń ze względu na ograniczoną dokładność obliczeń.

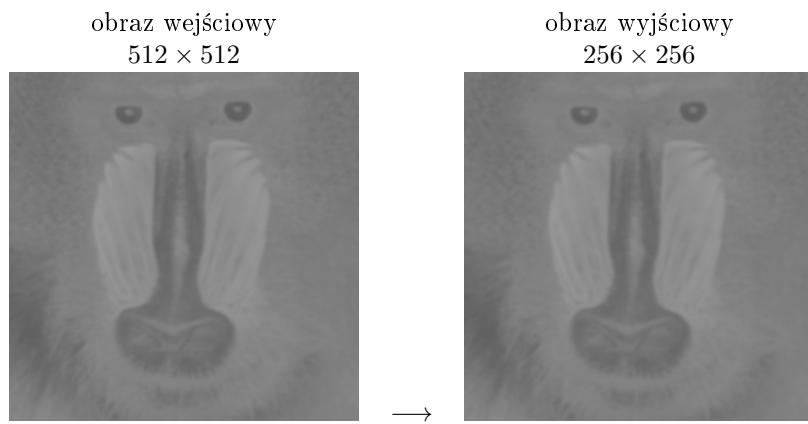
#### **Filtrowanie dolnoprzepustowe sygnałów różnicowych Cb i Cr**

Konwersja wykorzystuje fakt, że rozdzielcość przestrzenna ludzkiego oka w przypadku kolorów jest znacznie niższa niż w przypadku przejść jasności:

- sygnały różnicowe Cb i Cr są zwykle przechowywane w zmniejszonej rozdzielcości:
  1. filtracja dolnoprzepustowa (w najprostszym przypadku uśrednianie wartości pikseli)  
→ grafika 4.20
  2. podpróbkowanie (zmniejszenie rozdzielcości)  
zwykle podpróbkowanie pionowe i poziome o faktor 2 (YCbCr 4:2:0), co zmniejsza ilość danych o faktor 4  
→ grafika 4.21



Grafika 4.20: Uśrednienie wartości pikseli kanału  $Cb$  (filtracja dolnoprzepustowa)

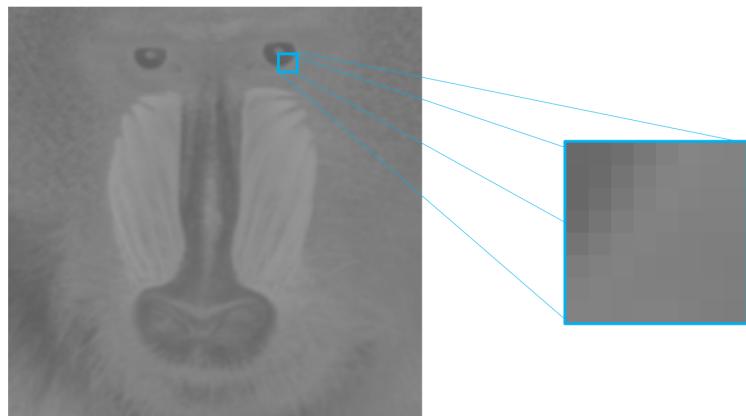


Grafika 4.21: Zmniejszenie rozdzielczości uśrednionego kanału  $Cb$

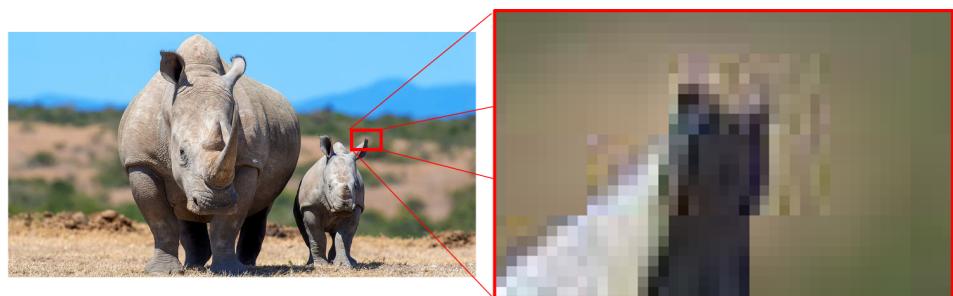
### Tworzenie bloków i dyskretna transformacja kosinusowa

→ grafiki 4.22, 4.23

- każda współrzędna (Y, Cb i Cr) obrazu jest podzielona na bloki  $8 \times 8$  pikseli

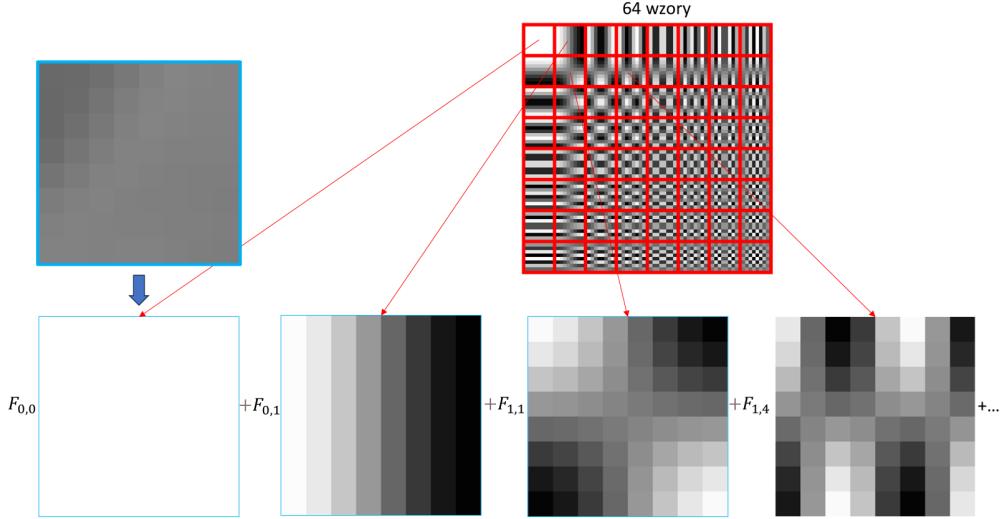


Grafika 4.22: Blok pikseli  $8 \times 8$  (kanał  $Cb$ )



Grafika 4.23: Bloki pikseli  $8 \times 8$  widoczne na powiększonym obrazie w formacie JPEG

- każdy blok  $8 \times 8$  obrazu jest reprezentowany jako kombinacja liniowa 64 wzorów pikseli
- grafika 4.24
- współczynniki rozkładu obliczone poprzez dyskretną transformację kosinusową (DCT)



Grafika 4.24: Reprezentacja bloku pikseli poprzez kombinację liniową 64 wzorów pikseli (DCT)

- dwuwymiarowa dyskretna transformata kosinusowa (DCT)<sup>1</sup> dla bloku pikseli  $8 \times 8$

$$F(k, l) = \frac{1}{4} C_k C_l \sum_{m=0}^7 \sum_{n=0}^7 f(m, n) \cos \frac{(2m+1)k\pi}{16} \cos \frac{(2n+1)l\pi}{16} \quad \text{dla } k = 0, \dots, 7 \\ \text{oraz } l = 0, \dots, 7$$

gdzie

$$C_k, C_l = \begin{cases} \frac{1}{\sqrt{2}} & : k, l = 0 \\ 1 & : w \text{ innym przypadku} \end{cases}$$

W szczególności element  $F(0, 0)$  (oznaczany zazwyczaj przez  $DC$ <sup>2</sup>) jest proporcjonalny do średniej jasności obrazu:

$$F(0, 0) = \frac{1}{8} \sum_{m=0}^7 \sum_{n=0}^7 f(m, n) \cos \frac{(2m+1)k\pi}{16} \cos \frac{(2n+1)l\pi}{16}$$

- transformację kosinusową oblicza się korzystając z szybkiej transformaty Fouriera (FFT)
- DCT ma dobre właściwości kompresji
- istnieje transformacja odwrotna IDCT (co oznacza również, że DCT jest bezstratna; żadna informacja nie zostaje utracona, ponieważ dane zostają po prostu przekształcone w bardziej korzystną formę do dalszego przetwarzania)

<sup>1</sup>Z wyjątkiem stałego współczynnika 2, DCT odpowiada DFT ciągu liczb rzeczywistych o długości  $4N$  o parzystej symetrii, gdzie wszystkie elementy o parzystym indeksie mają wartość 0.

<sup>2</sup>Direct Current oznaczenie dla prądu stałego

## Kwantyzacja

→ powoduje rzeczywistą redukcję danych (i pogorszenie jakości)

- współczynniki DCT są dzielone przez macierz kwantyzacji  $Q$  (element po elemencie), a następnie zaokrąglane do najbliższej liczby całkowitej

$$F^Q(k, l) = \text{round} \left( \frac{F(k, l)}{Q(k, l)} \right)$$

- macierz kwantyzacji  $Q$  odpowiada zarówno za jakość, jak i stopień kompresji
  - $Q$  jest zapisywana w nagłówku plików JPEG (znacznik DQT)
  - $Q$  jest optymalna, jeśli w przybliżeniu przedstawia wrażliwość oka na odpowiednie częstotliwości przestrzenne. Oko jest bardziej wrażliwe na gruboziarniste struktury, dlatego wartości kwantyzacji dla tych częstotliwości są mniejsze niż dla wysokich częstotliwości.
  - przykład macierzy kwantyzacji  $Q$ :

$$Q = \begin{pmatrix} 10 & 15 & 25 & 37 & 51 & 66 & 82 & 100 \\ 15 & 19 & 28 & 39 & 52 & 67 & 83 & 101 \\ 25 & 28 & 35 & 45 & 58 & 72 & 88 & 105 \\ 37 & 39 & 45 & 54 & 66 & 79 & 94 & 111 \\ 51 & 52 & 58 & 66 & 76 & 89 & 103 & 119 \\ 66 & 67 & 72 & 79 & 89 & 101 & 114 & 130 \\ 82 & 83 & 88 & 94 & 103 & 114 & 127 & 142 \\ 100 & 101 & 105 & 111 & 119 & 130 & 142 & 156 \end{pmatrix}$$

**Przykład:** zastosowanie  $Q$  do bloku  $8 \times 8$  współczynników DCT

$$F = \begin{pmatrix} 782.91 & 44.93 & 172.52 & -35.28 & -20.58 & 35.93 & 2.88 & -3.85 \\ -122.35 & -75.46 & -7.52 & 55.00 & 30.72 & -17.73 & 8.29 & 1.97 \\ -2.99 & -32.77 & -57.18 & -30.07 & 1.76 & 17.63 & 12.23 & -13.57 \\ -7.98 & 0.66 & 2.41 & -21.28 & -31.07 & -17.20 & -9.68 & 16.94 \\ 3.87 & 7.07 & 0.56 & 5.13 & -2.47 & -15.09 & -17.70 & -3.76 \\ -3.77 & 0.80 & -1.46 & -3.50 & 1.48 & 4.13 & -6.32 & -18.47 \\ 1.78 & 3.28 & 4.63 & 3.27 & 2.39 & -2.31 & 5.21 & 11.77 \\ -1.75 & 0.43 & -2.72 & -3.05 & 3.95 & -1.83 & 1.98 & 3.87 \end{pmatrix}$$

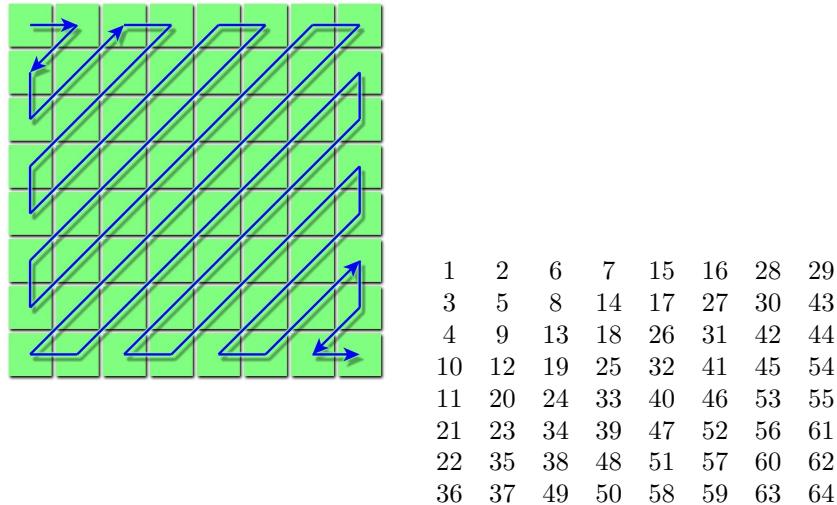
Obliczenie współczynników  $F^Q(k, l)$ :

$$\begin{aligned} F^Q(0, 0) &= \text{round} \left( \frac{F(0, 0)}{Q(0, 0)} \right) = \text{round} \left( \frac{782.91}{10} \right) = 78 \\ F^Q(0, 1) &= \text{round} \left( \frac{F(0, 1)}{Q(0, 1)} \right) = \text{round} \left( \frac{-122.35}{15} \right) = -8 \\ &\dots \end{aligned}$$

$$F^Q = \begin{pmatrix} 78 & 3 & 7 & -1 & 0 & 1 & 0 & 0 \\ -8 & -4 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

### Przegrupowanie i kodowanie różnicowe składowej DC

- 64 współczynniki dyskretnej transformaty kosinusowej są sortowane<sup>3</sup> według częstotliwości, zaczynając od stałej składowej o częstotliwości 0 (DC).  
→ grafika 4.25 Współczynniki o wysokich wartościach są teraz zwykle pierwsze,



Grafika 4.25: Zyzgakowata kolejność komponentów zkwantyzowanej transformaty obrazu

a małe współczynniki dalej. Optymalizuje to wejście dla kolejnego kodowania.

---

<sup>3</sup>Zyzgakowe przegrupowanie współczynników DCT wchodziło w zakres ochrony patentu US4698672. Jednakże w 2002 r. stwierdzono, że zastrzegana metoda nie jest nowością w stanie techniki, zatem wszelkie roszczenia byłyby trudne do wyegzekwowania.

**Przykład:** przegrupowana kolejność współczynników  $F^Q$  w bloku  $8 \times 8$

$$F^Q = \begin{pmatrix} 78 & 3 & 7 & -1 & 0 & 1 & 0 & 0 \\ -8 & -4 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

78 3 -8 0 -4 7 -1 0 -1 0 0 0 -2 1 0 1 1 -1 0 ...

- kodowanie różnicowe współczynników DC:  
składowa DC kodowana jest różnicowo ze składową DC bloku znajdującej się po lewej stronie - w ten sposób uwzględniane są zależności pomiędzy sąsiednimi blokami:

$$\begin{array}{ccccccccccccccc} 119 & & \dots & & & & & & & & & & & & & & & \dots \\ 78 & 3 & -8 & 0 & -4 & 7 & -1 & 0 & -1 & 0 & \dots \\ 102 & 5 & -5 & 0 & 3 & -4 & 2 & -1 & 0 & 0 & \dots \\ 75 & -19 & 2 & -1 & 0 & -1 & 1 & -1 & 0 & 0 & \dots \\ & & & & & & & & & & & & & & & & \downarrow \\ & \dots & & & & & & & & & & & & & & & \\ -41 & 3 & -8 & 0 & -4 & 7 & -1 & 0 & -1 & 0 & \dots \\ 24 & 5 & -5 & 0 & 3 & -4 & 2 & -1 & 0 & 0 & \dots \\ -27 & -19 & 2 & -1 & 0 & -1 & 1 & -1 & 0 & 0 & \dots \end{array}$$

### Kodowanie entropijne

- zazwyczaj kodowanie Huffmana (*Huffman coding*) —→ grafika 4.26
- norma JPEG umożliwia także kodowanie arytmetyczne (generuje pliki, które są od 5 – 15% mniejsze)  
—→ rzadko używane ze względów patentowych, a także znacznie wolniejsze

1. "A\_DEAD\_DAD\_CEDED\_A\_BAD\_BABE\_A\_BEADED\_ABACA\_BED"
2. 

```

graph LR
    C2["C: 2"] --- M1(( ))
    B6["B: 6"] --- M1
    E7["E: 7"] --- M1
    D10["D: 10"] --- M1
    A11["A: 11"] --- M1
    M1 --- ECB15["ECB: 15"]
    style M1 fill:none,stroke:none
    style ECB15 fill:none,stroke:none
    
```
3. 

```

graph LR
    E7["E: 7"] --- M2(( ))
    CB8["CB: 8"] --- M2
    D10["D: 10"] --- M2
    A11["A: 11"] --- M2
    M2 --- ECB15["ECB: 15"]
    style M2 fill:none,stroke:none
    style ECB15 fill:none,stroke:none
    
```
4. 

```

graph LR
    D10["D: 10"] --- M3(( ))
    A11["A: 11"] --- M3
    ECB15["ECB: 15"] --- M3
    M3 --- D20["D: 20"]
    style M3 fill:none,stroke:none
    style D20 fill:none,stroke:none
    
```
5. 

```

graph LR
    A11["A: 11"] --- M4(( ))
    ECB15["ECB: 15"] --- M4
    D20["D: 20"] --- M4
    M4 --- AEBCB26["AEBCB: 26"]
    style M4 fill:none,stroke:none
    style AEBCB26 fill:none,stroke:none
    
```
6. 

```

graph LR
    AEBCB26["AEBCB: 26"] --- M5(( ))
    D20["D: 20"] --- M5
    D10["D: 10"] --- M5
    M5 --- DAEBCB46["DAEBCB: 46"]
    style M5 fill:none,stroke:none
    style DAEBCB46 fill:none,stroke:none
    
```
7. 

```

graph LR
    DAEBCB46["DAEBCB: 46"] --- M6(( ))
    A11["A: 11"] --- M6
    ECB15["ECB: 15"] --- M6
    M6 --- Root["Root"]
    style M6 fill:none,stroke:none
    style Root fill:none,stroke:none
    
```
8. "10000111010010001100100111011001100100100011111001001111101111110  
0010001111110100111001001011111011101000111111001"

Grafika 4.26: Kod Huffmmana. Źródło: wikipedia commons

# 5

## Dyskretyzacja w teorii sygnałów i systemów

- **Sygnal**

↪ proces zmian (w czasie, w przestrzeni) wielkości fizycznej lub stanu obiektu fizycznego

- **Modelowanie matematyczne sygnału**

- funkcje rzeczywiste
- funkcje zespolone
- dystrybucje

→ grafika 5.1

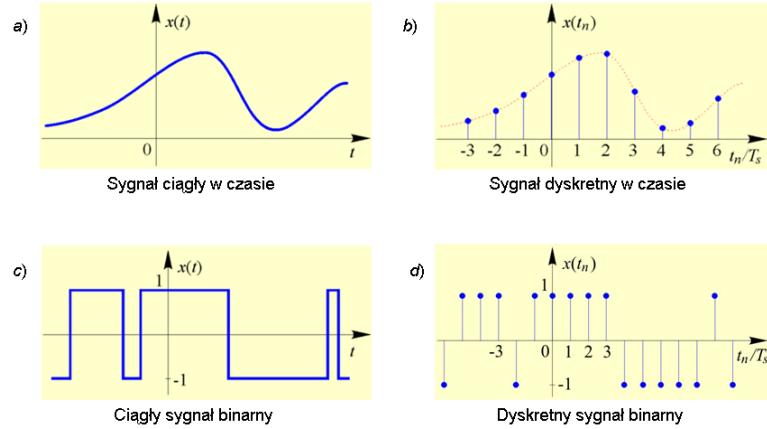
### 5.1 Sygnały i systemy

→ grafika 5.2

Teoria sygnałów systemów zajmuje się matematycznym opisem i obliczaniem systemów fizycznych na poziomie abstrakcyjnym. Takimi układami fizycznymi mogą być na przykład filtry lub pętle sterowania.

- Sygnał to zmienna wielkość lub funkcja reprezentująca informację (np. napięcie elektryczne, fala dźwiękowa, obraz lub cena akcji).
- System to opis abstrakcyjny, czyli model (matematyczny) rzeczywistego procesu przetwarzającego takie sygnały.

$$f_1 \xrightarrow{\text{system}} f_2$$
$$f_1 \xrightarrow{T\{\cdot\}} f_2$$



Grafika 5.1: Przykładowe rodzaje sygnałów w teorii systemów. Źródło: Materiały do wykładu "Przetwarzanie sygnałów", Instytut Informatyki i Centrum Otwartej Multi-medialnej Edukacji Uniwersytetu Warszawskiego, <http://wazniak.mimuw.edu.pl/>

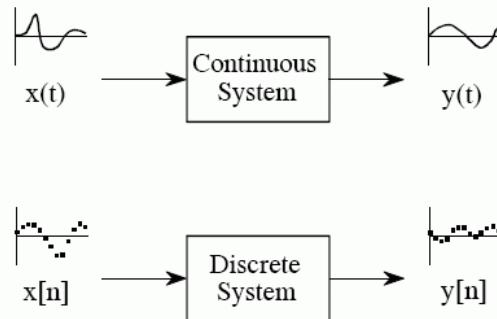


FIGURE 5-1  
Terminology for signals and systems. A system is any process that generates an output signal in response to an input signal. Continuous signals are usually represented with parentheses, while discrete signals use brackets. All signals use lower case letters, reserving the upper case for the frequency domain (presented in later chapters). Unless there is a better name available, the input signal is called:  $x(t)$  or  $x[n]$ , while the output is called:  $y(t)$  or  $y[n]$ .

Grafika 5.2: Sygnały i systemy. Źródło: <https://www.dspsguide.com/ch5/1.htm>

### Narzędzia opisu systemów LTI

System LTI - system liniowy niezmienniczy w czasie (linear time-invariant system)  
System liniowy:

$$T\{af_1 + bf_2\} = aT\{f_1\} + bT\{f_2\}$$

→ transformacja Fouriera

→ transformacja Laplace'a

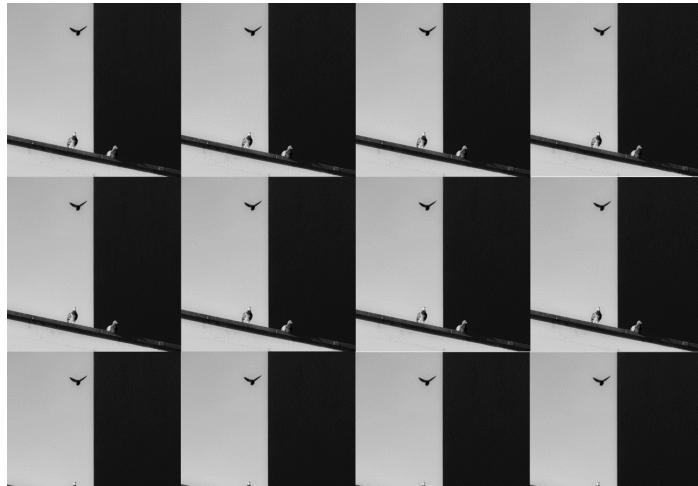
→ transformacja Z

→ ...

### Twierdzenie Fouriera

**Każdą funkcję okresową można przedstawić jako sumę funkcji sin i cos o różnych częstotliwościach.**

- Jeśli funkcja nie jest okresowa, ale jest ograniczona do określonego obszaru argumentów, można skopiować ten obszar (kontynuować go okresowo) i w ten sposób uzyskać funkcję okresową (grafika 5.3)
- Wiersze i kolumny obrazu można postrzegać jako nieokresowe funkcje dyskretne. Można więc także przekształcić obraz transformacją Fouriera.



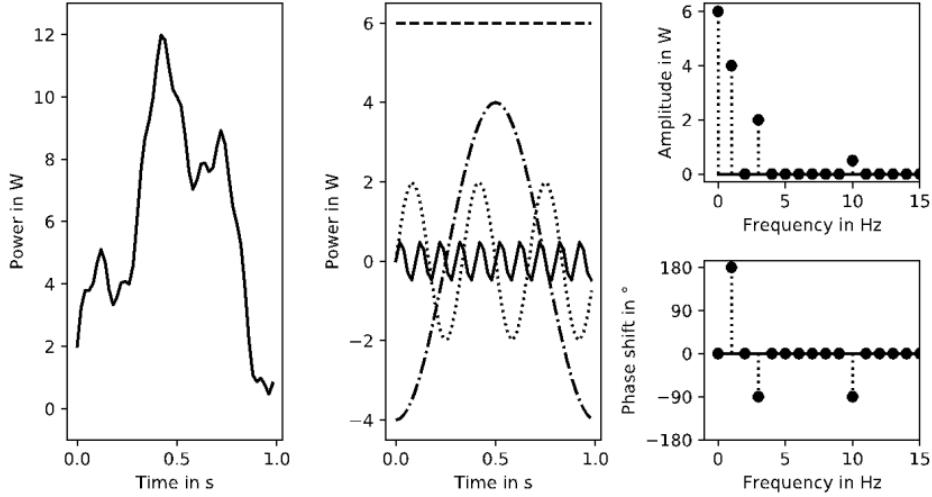
Grafika 5.3: Obraz jako dwuwymiarowa funkcja okresowa

## 5.2 Transformacja Fouriera

**Cel:**

Rozkład funkcji wyjściowej na funkcje trygonometryczne o różnych częstotliwościach (spektrum, widmo).

Przykład → grafika 5.4



Grafika 5.4: Zasada transformacji Fouriera: sygnał wejściowy (po lewej), który można opisać poprzez superpozycję różnych oscylacji (w środku), a także reprezentacja tego sygnału w dziedzinie częstotliwości (po prawej). Źródło: "Optimal linking of heat flows based on the discrete Fourier transform", C. Sejkora et al., 2019

### Rozkład funkcji okresowej o okresie $2\pi$

$$f(x) = f(x + k \cdot 2\pi), k \in \mathbb{Z}$$

Jean Baptiste Joseph Fourier, 1822:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} [a_k \cos(kx) + b_k \sin(kx)]$$

gdzie

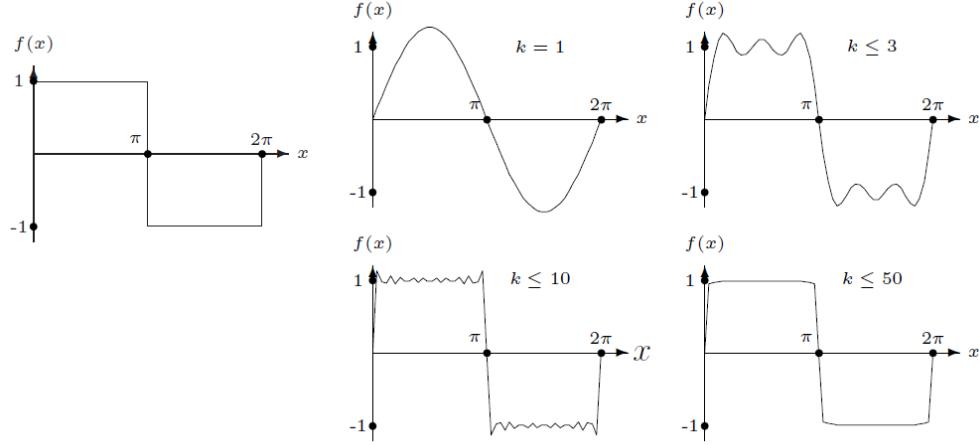
$$\begin{aligned} a_0 &= 1/\pi \int_0^{2\pi} f(x) dx \\ a_k &= 1/\pi \int_0^{2\pi} f(x) \cos(kx) dx \\ b_k &= 1/\pi \int_0^{2\pi} f(x) \sin(kx) dx \end{aligned}$$

Jeśli  $\sum_{k=1}^{\infty}$  zostanie zastąpione przez  $\sum_{k=1}^{k_{\max}}$ , funkcja wyjściowa  $f(x)$  zostanie przybliżona.  $\rightarrow$  grafika 5.5.

### Zapis w postaci zespolonej

#### Liczby zespolone

Liczby zespolone ( $\mathbb{C}$ ) stanowią rozszerzenie liczb rzeczywistych. Celem rozszerzenia jest umożliwienie rozwiązania równań algebraicz-



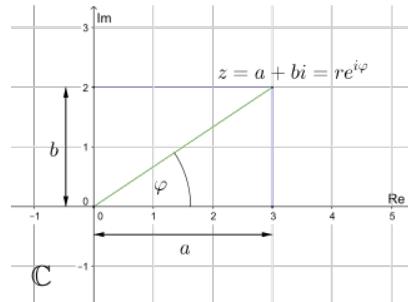
Grafika 5.5: Przybliżenie funkcji okresowej poprzez skończony szereg Fouriera

nych takich jak  $x^2 + 1 = 0$  lub  $x^2 = -1$ .

Zapis:

$$z = a + i \cdot b \quad \text{gdzie } a, b \in \mathbb{R}, i^2 = -1$$

$\Re\{z\} = a$	część rzeczywista
$\Im\{z\} = b$	część urojona
$\varphi$	argument
$ z  = \sqrt{a^2 + b^2}$	moduł



Postać trygonometryczna:

$$z = |z| \cdot (\cos \varphi + i \cdot \sin \varphi)$$

Postać wykładnicza:

$$\text{z wzorze Eulera } e^{i\varphi} = \cos \varphi + i \cdot \sin \varphi$$

$$z = |z| \cdot e^{i\varphi}$$

Wzory Eulera:

$$\begin{aligned}
e^{ikx} &= \cos(kx) + i \cdot \sin(kx) \\
e^{-ikx} &= \cos(kx) - i \cdot \sin(kx) \\
\rightarrow f(x) &= \frac{a_0}{2} + \sum_{k=1}^{\infty} [a_k \cos(kx) + b_k \sin(kx)] \\
&= \frac{a_0}{2} + \sum_{k=1}^{\infty} \left[ a_k \frac{e^{ikx} + e^{-ikx}}{2} + b_k \frac{e^{ikx} - e^{-ikx}}{2i} \right] \\
&= \frac{a_0}{2} + \sum_{k=1}^{\infty} \left[ a_k \frac{e^{ikx} + e^{-ikx}}{2} - ib_k \frac{e^{ikx} - e^{-ikx}}{2} \right] \\
&= \frac{a_0}{2} + \sum_{k=1}^{\infty} \left[ \frac{a_k - ib_k}{2} e^{ikx} + \frac{a_k + ib_k}{2} e^{-ikx} \right] \\
&= \frac{a_0}{2} + \sum_{k=1}^{\infty} \left[ \frac{a_k - ib_k}{2} e^{ikx} \right] + \sum_{k=-\infty}^{-1} \left[ \frac{a_{-k} + ib_{-k}}{2} e^{ikx} \right]
\end{aligned}$$

Rozkład:

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx}$$

gdzie

$$\begin{aligned}
c_k &= \frac{1}{2\pi} \int_0^{2\pi} f(x) e^{-ikx} dx \\
&= \begin{cases} a_0/2 & : k = 0 \\ 1/2 \cdot (a_k - ib_k) & : k > 0 \\ 1/2 \cdot (a_{-k} + ib_{-k}) & : k < 0 \end{cases}
\end{aligned}$$

Rozszerzenie na dowolne sygnały okresowe (okres  $L$ ):

$$\cos(kx) \longrightarrow \cos\left(kx \cdot \frac{2\pi}{L}\right) = \cos\left(\frac{2\pi kx}{L}\right) \text{ Szereg Fouriera:}$$

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{i\frac{2\pi k}{L}x}$$

gdzie

$$c_k = \frac{1}{L} \int_0^L f(x) e^{-i\frac{2\pi k}{L}x} dx$$

**Dyskretny szereg Fouriera w przetwarzaniu (okresowych) sygnałów ciągłych**

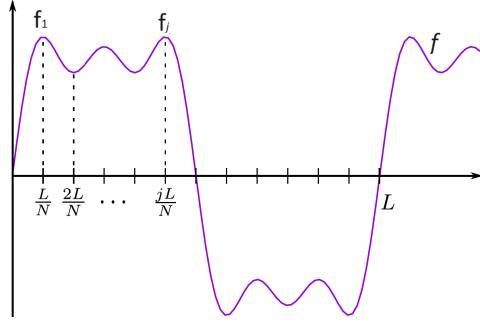
$c_k \longrightarrow F(k)$  widmo częstotliwościowe sygnału (współczynniki Fouriera, spektrum)

Szereg Fouriera:

$$f(x) = \sum_{k=-\infty}^{\infty} F(k) e^{i\frac{2\pi k}{L}x}$$

gdzie

$$F(k) = \frac{1}{L} \int_0^L f(x) e^{-i\frac{2\pi k}{L}x} dx$$



Grafika 5.6: Dyskretne wartości funkcji  $f(x)$  (próbkowanie, sampling)

Dla (dyskretnych) wartości  $x = nT, n \in \{0, \dots, N-1\}$ ,  $T = \frac{L}{N}$  ( $\rightarrow$  grafika 5.6)

$$\begin{aligned} f_n = f(nT) &= \sum_{k=-\infty}^{\infty} F(k) e^{i \frac{2\pi k}{L} n \frac{L}{N}} \\ &= \sum_{k=-\infty}^{\infty} F(k) e^{i \frac{2\pi}{N} kn} \end{aligned}$$

Łatwo zauważyc, że term  $e^{i \frac{2\pi}{N} kn}$  jest identyczny dla wszystkich wartości  $k$ , które różnią się o  $N$ :

$$\begin{aligned} e^{i \frac{2\pi}{N} k \cdot (n+rN)} &= e^{i \frac{2\pi}{N} kn} \cdot e^{i \frac{2\pi}{N} krN} \\ &= e^{i \frac{2\pi}{N} kn} \cdot e^{i 2\pi kr} \\ &= e^{i \frac{2\pi}{N} kn} \cdot (\cos(2\pi kr) + i \sin(2\pi kr)) \\ &= e^{i \frac{2\pi}{N} kn} \end{aligned}$$

$\rightarrow \sum_{k=-\infty}^{\infty}$  można zastąpić przez  $\sum_{k=0}^{N-1}$

Jednocześnie współczynniki Fouriera  $F(k)$  można przybliżyć sumą skończoną:

$$\begin{aligned} F(k) &= \frac{1}{L} \int_0^L f(x) e^{-i \frac{2\pi k}{L} x} dx \\ &\approx \frac{1}{NT} \sum_{k=0}^{N-1} T f(nT) e^{-i \frac{2\pi k}{NT} nT} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} f(nT) e^{-i \frac{2\pi k}{N} n} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} f_n e^{-i \frac{2\pi}{N} kn} \end{aligned}$$

$\rightarrow$  dyskretna transformacja Fouriera (rozdział 5.2.1)

### Transformacje Fouriera

Transformację Fouriera można zdefiniować dla sygnałów, które są dyskretne lub ciągłe w czasie i mają skończony lub nieskończony czas trwania.  $\rightarrow$  grafika 5.7

Time Duration		
Finite	Infinite	
Discrete FT (DFT) $X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\omega_k n}$ $k = 0, 1, \dots, N - 1$	Discrete Time FT (DTFT) $X(\omega) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}$ $\omega \in (-\pi, +\pi)$	discr. time $n$
Fourier Series (FS) $X(k) = \int_0^P x(t)e^{-j\omega_k t} dt$ $k = -\infty, \dots, +\infty$	Fourier Transform (FT) $X(\omega) = \int_{-\infty}^{+\infty} x(t)e^{-j\omega t} dt$ $\omega \in (-\infty, +\infty)$	cont. time $t$
discrete freq. $k$	continuous freq. $\omega$	

Grafika 5.7: Transformata Fouriera.  $\omega_k = 2\pi \frac{k}{N}$ . Źródło: <https://ccrma.stanford.edu/>

### 5.2.1 Dyskretna transformacja Fouriera, DFT (*Discrete Fourier Transform*)

**Dyskretna transformata Fouriera  $F(k)$  dla ciągu  $f(n)$ :**

$$f(n) \text{ ciąg, } n \in \{0, \dots, N - 1\}$$

$$F(k) = \sum_{k=0}^{N-1} f(n)e^{-i\frac{2\pi}{N}kn} \text{ dla } k = 0, \dots, N - 1$$

Wynikiem transformacji<sup>1</sup> jest rozkład ciągu na składowe harmoniczne (sinusoidalne) oraz stałą  $F(0)$ , który odpowiada wartości średniej ciągu wejściowego.

**Przekształcenie odwrotne (IDFT, *Inverse transform*):**

$$f(n) = \frac{1}{N} \sum_{k=0}^{N-1} F(k)e^{i\frac{2\pi}{N}kn} \text{ dla } n = 0, \dots, N - 1$$

### Splot ciągów

$f(n), g(n)$  dwa (okresowe) ciągi długości  $N$

Definicja splotu  $f(n) * g(n)$ :

---

<sup>1</sup> Wybrane tutaj współczynniki normalizacji mnożące DFT i IDFT (1 i  $\frac{1}{N}$ ) i znaki wykładników są najczęstszymi konwencjami. Jedynymi rzeczywistymi wymaganiami dla konwencji jest to, że DFT i IDFT mają wykładniki o przeciwnych znakach i że iloczyn ich współczynników normalizacji wynosi  $\frac{1}{N}$ . Możliwa jest np. normalizacja  $\frac{1}{\sqrt{N}}$  zarówno dla DFT, jak i IDFT lub współczynniki  $\frac{1}{N}$  dla DFT i 1 dla IDFT.

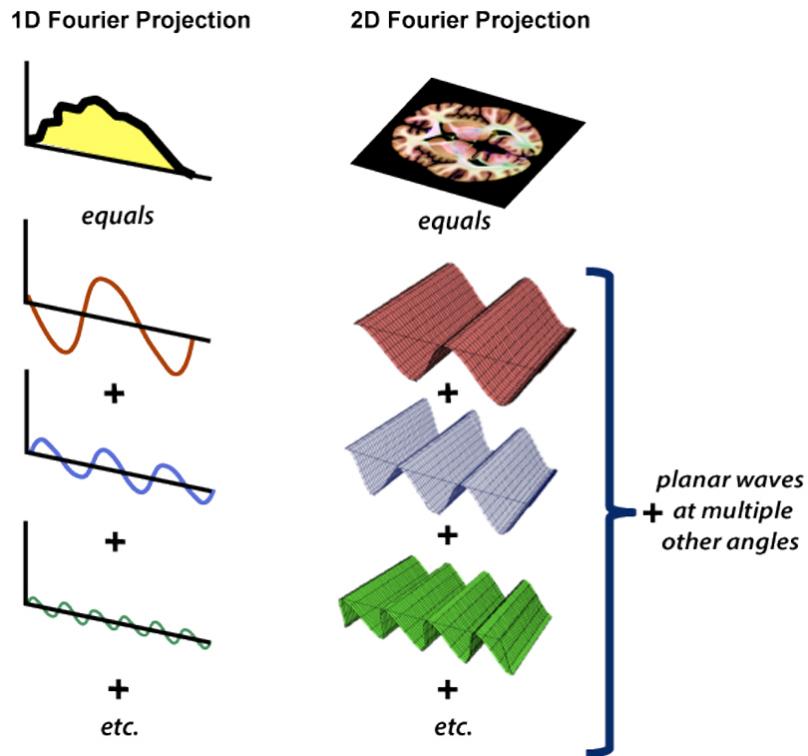
$$w(j) = \sum_{n=0}^{N-1} f(n)g(j-n), j \in \mathbb{Z}$$

DFT dla ciągu  $w(j)$ :

$$\begin{aligned} W(k) &= \sum_{k=0}^{N-1} w(j)e^{-i\frac{2\pi}{N}kj} \\ &= \sum_{k=0}^{N-1} \sum_{n=0}^{N-1} f(n)g(j-n)e^{-i\frac{2\pi}{N}kj} \\ &= \sum_{n=0}^{N-1} f(n)e^{-i\frac{2\pi}{N}kn} \sum_{k=0}^{N-1} g(j-n)e^{-i\frac{2\pi}{N}k(j-n)} \\ &= F(k) \cdot G(k) \end{aligned}$$

### 5.2.2 Transformata Fouriera w przetwarzaniu obrazu (2D DFT)

→ grafika 5.8

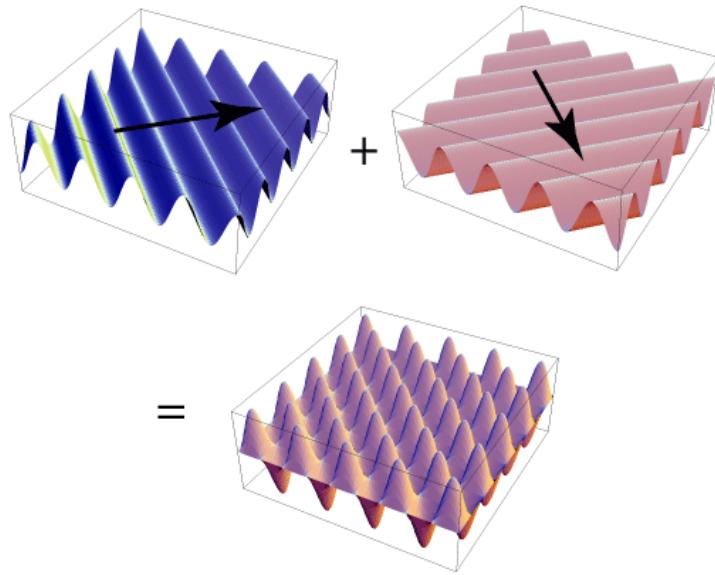


Grafika 5.8: Transformacja Fouriera w 2D. Źródło: <https://mriquestions.com/what-are-2d—3d-ft.html>

Obraz można rozłożyć na płaskie fale o różnych amplitudach i częstotliwościach i odwrotnie: suma fal płaskich może zostać wykorzystywana do tworzenia płaskiego

obrazu.

Suma dwóch fal płaskich pod różnymi kątami  $\rightarrow$  grafika 5.9



Grafika 5.9: Suma dwóch fal płaskich pod różnymi kątami. Źródło:  
<https://mriquestions.com/what-are-2d—3d-ft.html>

#### Dwuwymiarowe przekształcenie Fouriera (2D DFT):

$$F(k, l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-i \frac{2\pi}{M} km} e^{-i \frac{2\pi}{N} ln} \quad \text{dla } k = 0, \dots, M-1 \\ \text{oraz } l = 0, \dots, N-1$$

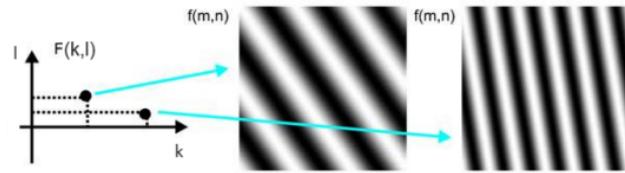
#### Przekształcenie odwrotne (2D IDFT):

$$f(m, n) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} F(k, l) e^{i \frac{2\pi}{M} km} e^{i \frac{2\pi}{N} ln} \quad \text{dla } m = 0, \dots, M-1 \\ \text{oraz } n = 0, \dots, N-1$$

Interpretacja  $\rightarrow$  grafika 5.10

#### Przykład $M, N = 4$

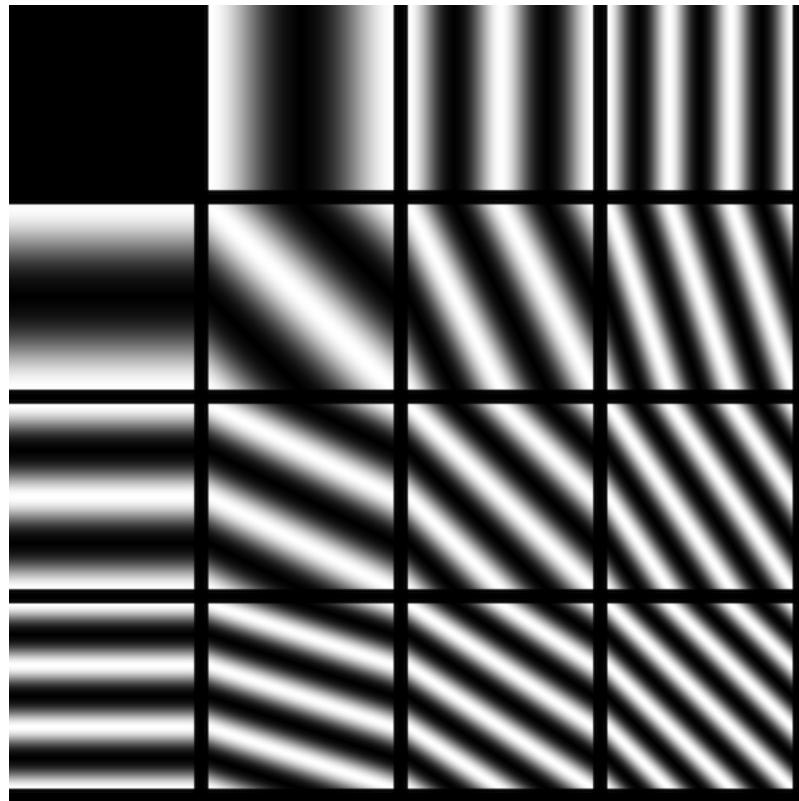
$$f(m, n) = \frac{1}{16} \sum_{k=0}^3 \sum_{l=0}^3 F(k, l) e^{i \frac{\pi}{2} km} e^{i \frac{\pi}{2} ln} \quad \text{dla } m = 0, \dots, 3 \\ \text{oraz } n = 0, \dots, 3$$



Grafika 5.10: Interpretacja przekształcenia odwrotnego 2D DFT

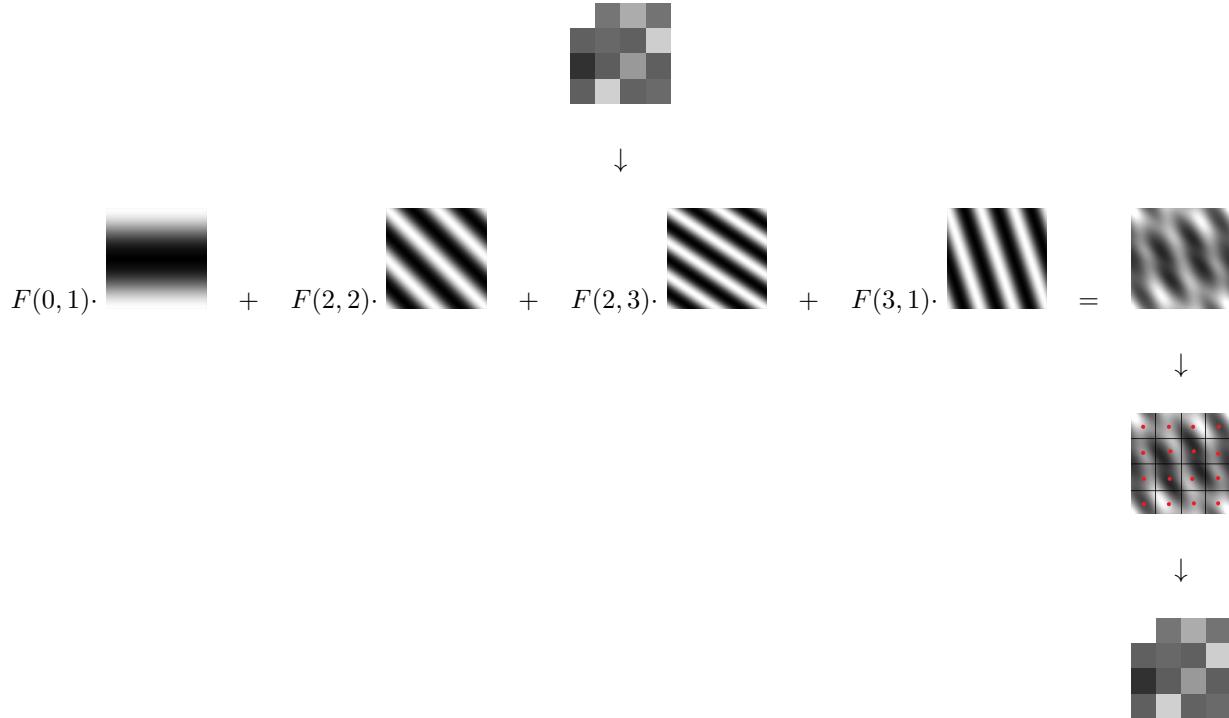
Obraz jest superpozycją fal płaskich (obrazów bazowych, grafika 5.11)  
 $e^{i\frac{\pi}{2}km}e^{i\frac{\pi}{2}ln} = e^{i\frac{\pi}{2}(km+nl)}$ .

$$\begin{aligned} k = 0, l = 1 &\longrightarrow e^{i\frac{\pi}{2}n} = \cos\left(\frac{\pi}{2}n\right) \\ k = 3, l = 1 &\longrightarrow e^{i\frac{\pi}{2}(3m+n)} \end{aligned}$$



Grafika 5.11: Obrazy bazowe 2D-DFT o rozmiarze  $M, N = 4$ . Źródło: Visualizing DFT and DCT Basis, DPVIEW

Przykładowy obraz i jego złożenie: grafika 5.12

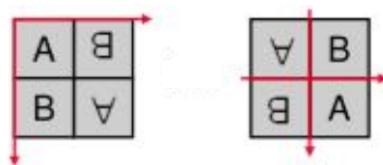


Grafika 5.12: Przykład: obraz \$4 \times 4\$ jako superpozycja fal płaskich (2D-DFT)

### Amplituda i faza

$$\begin{aligned}
 F(k, l) &= |F(k, l)| e^{i\Phi(k, l)} \\
 |F(k, l)| &= \sqrt{(\Re\{F(k, l)\})^2 + (\Im\{F(k, l)\})^2} \quad \text{amplituda widma (spektrum)} \\
 \Phi(k, l) &= \arctan \left[ \frac{\Im\{F(k, l)\}}{\Re\{F(k, l)\}} \right] \quad \text{faza} \\
 P(k, l) &= |F(k, l)|^2 \quad \text{moc (power spectrum)}
 \end{aligned}$$

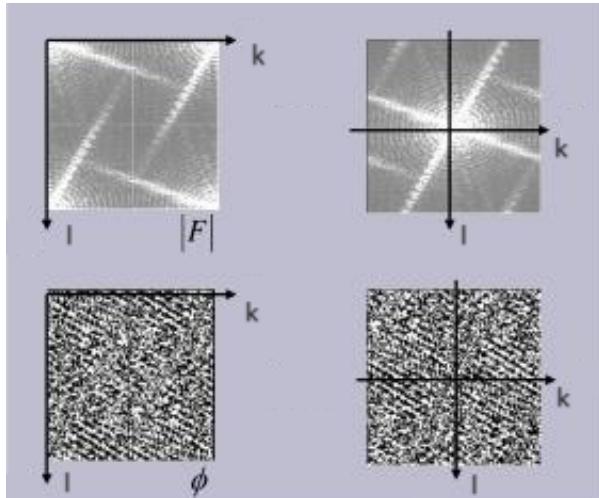
\$|F(k, l)|\$ i \$\Phi(k, l)\$ są punktowo symetryczne, dlatego wystarczy obliczenie tylko połowy współczynników:



Konwencje przedstawiania \$|F(k, l)|\$ i \$\Phi(k, l) \rightarrow\$ grafika 5.13.

### Ograniczenia 2D DFT

- Widmo DFT jest wartością zespoloną \$\rightarrow\$ amplituda, faza.
- Nie zawiera informacji o czasie tzn. o zmianach sygnału czy obrazu.



Grafika 5.13: Konwencje dla  $|F(k, l)|$  i  $\Phi(k, l)$ . Źródło: Diskrete Fouriertransformation. Ziven.

### Wizualizacja widma (spektrum) 2D DFT - przykłady

→ grafiki 5.14, 5.15, 5.16

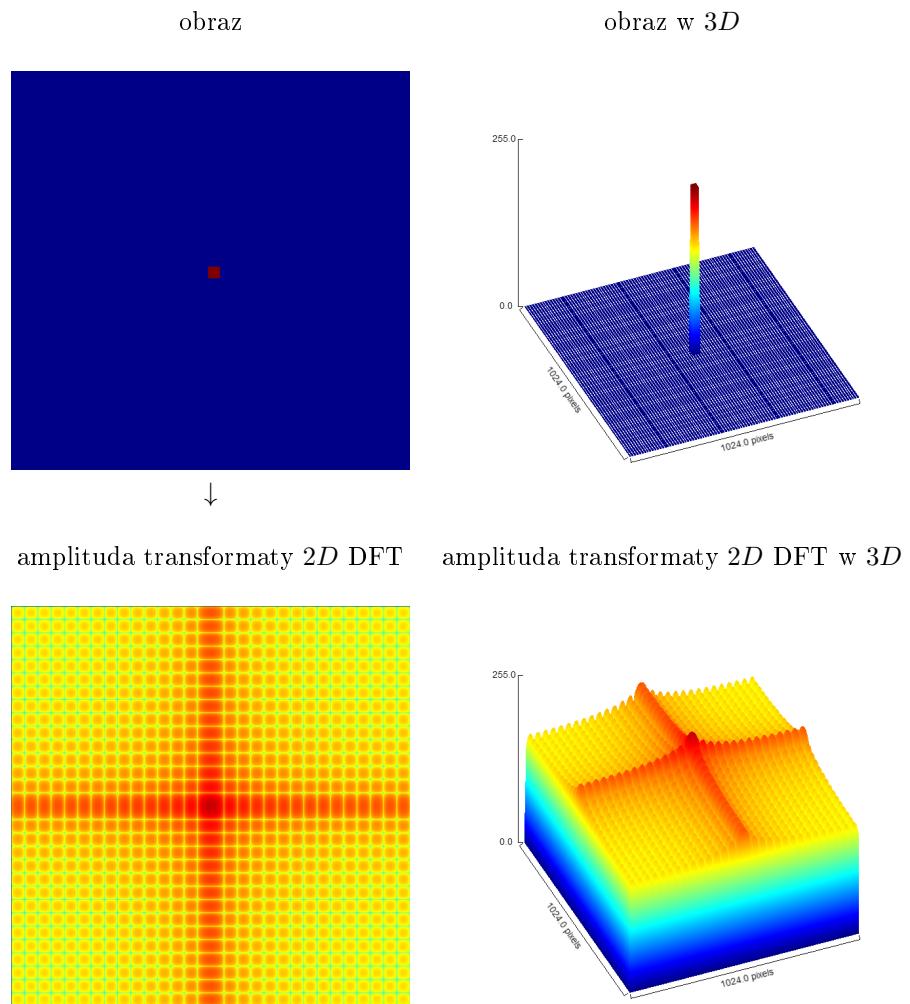
#### Przykład: szum na obrazie

→ szum w obrazie można traktować jak sumę ważonych, przesuniętych względem środka obrazu impulsów  $\delta$

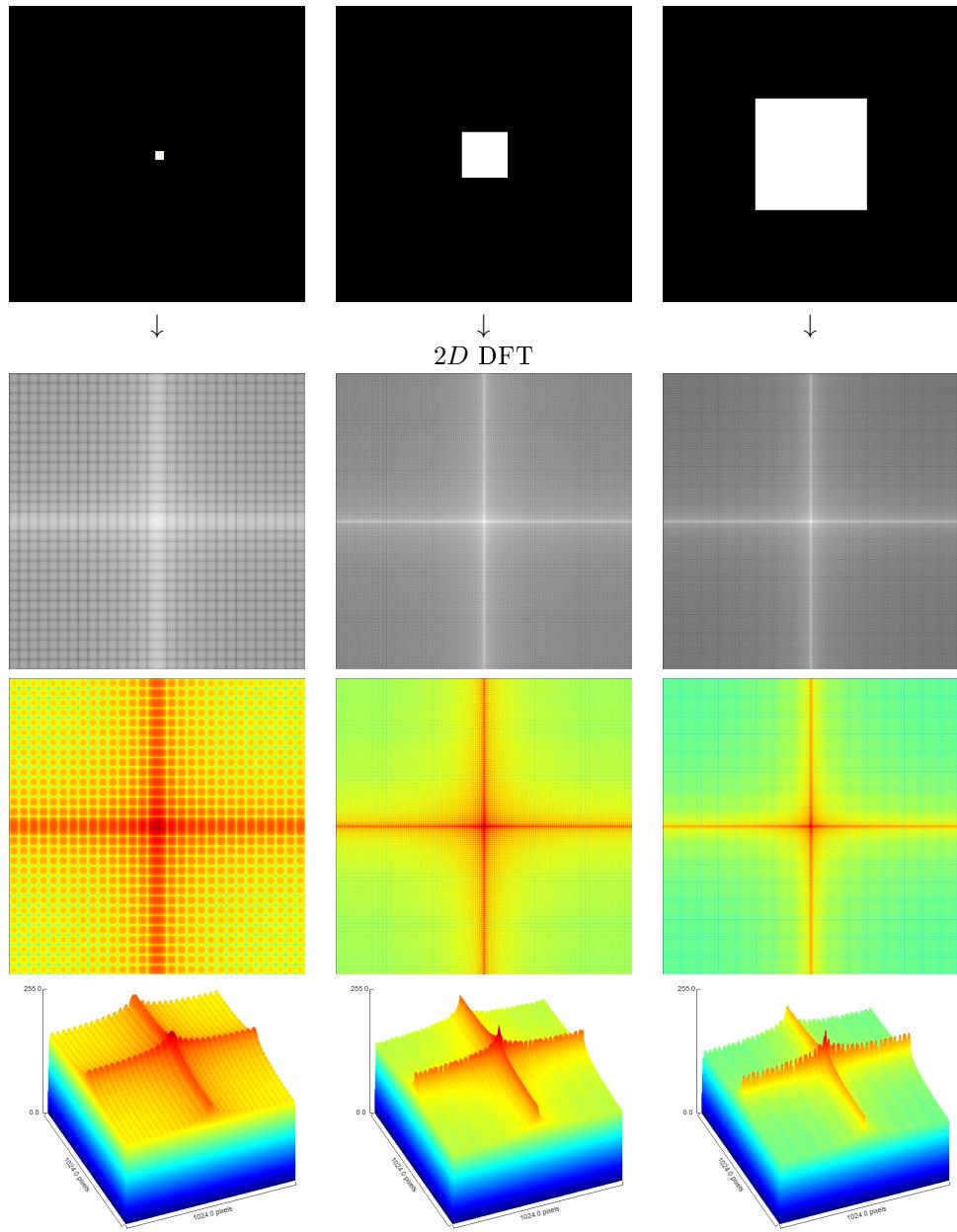
→ grafika 5.17

#### 2D DFT - przykład w skali logarytmicznej

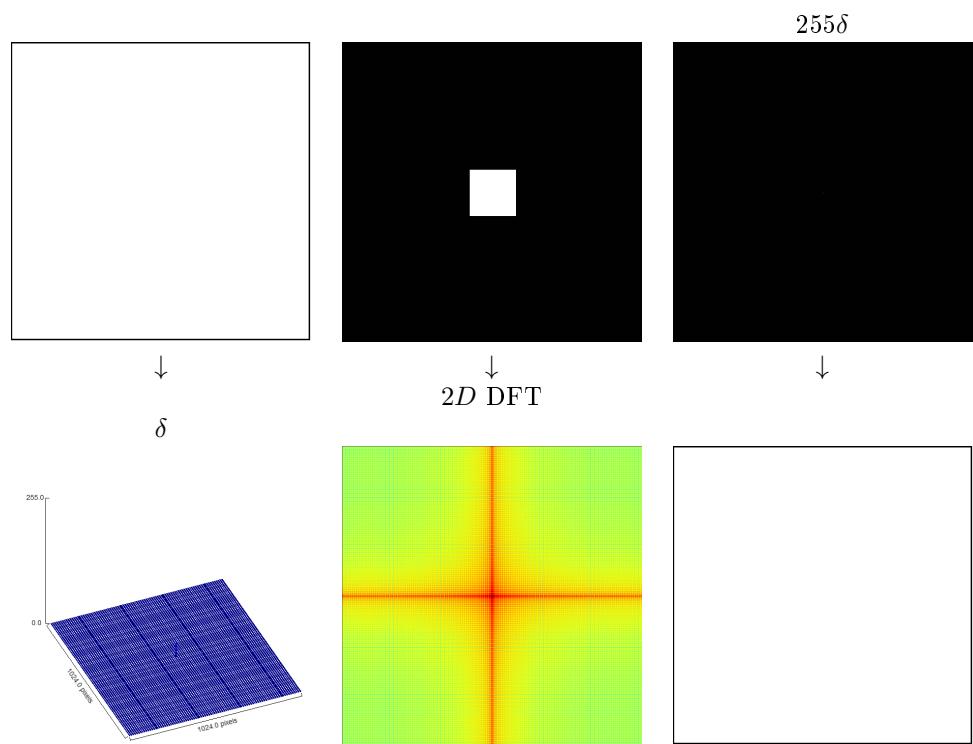
→ grafika 5.18



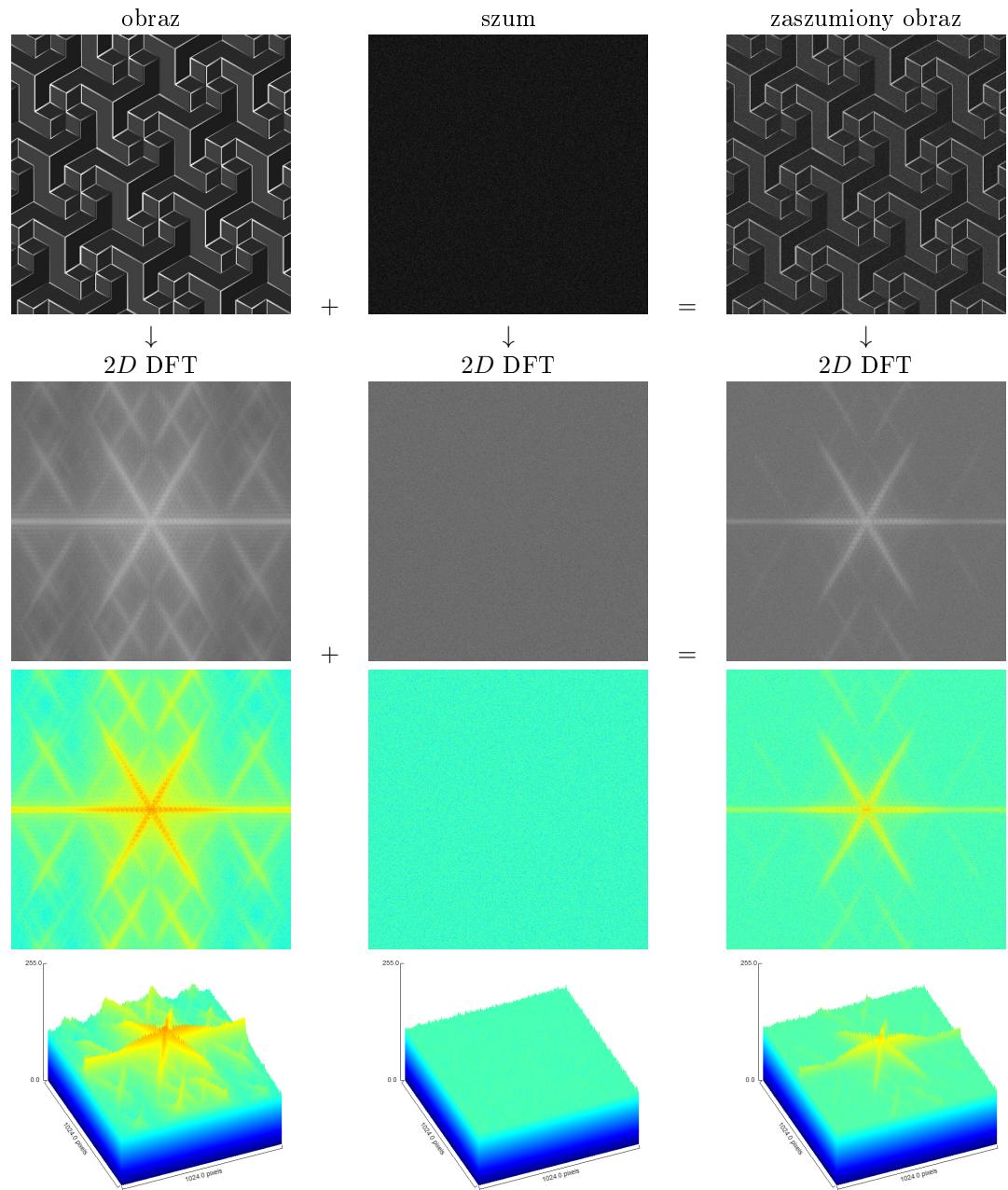
Grafika 5.14: Obraz kwadratu i jego spektrum (wizualizacja CLUT).



Grafika 5.15: 2D DFT dla obrazów kwadratów różnych wielkości.



Grafika 5.16: 2D DFT dla obrazów kwadratów o maksymalnych wymiarach (tu:  $1024 \times 1024$ ), o wymiarach  $150 \times 150$  i minimalnych wymiarach ( $1 \times 1$  tzn.  $255\delta$ )

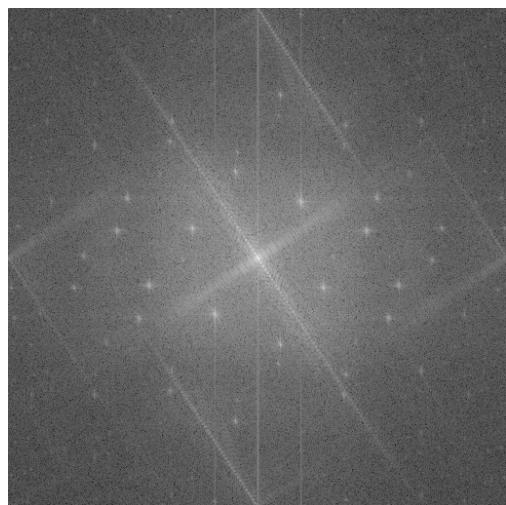


Grafika 5.17: 2D DFT zaszumionego obrazu

obraz



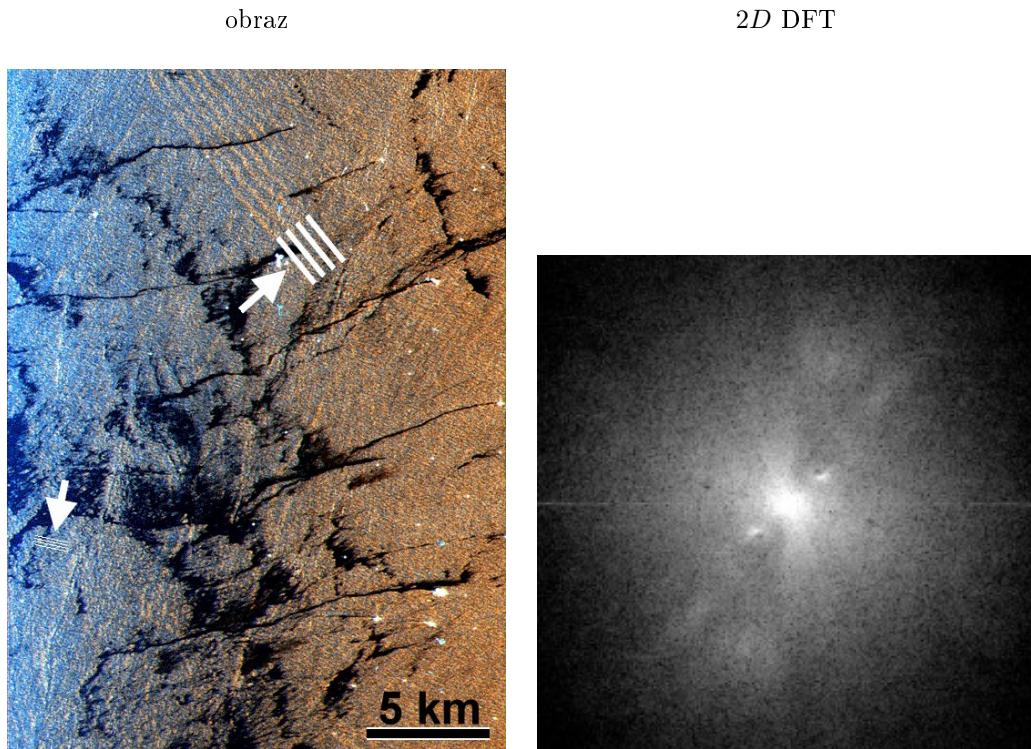
2D DFT



Grafika 5.18: Obraz i jego 2D DFT w skali logarytmicznej.

### 5.2.3 Interpretacja transformaty obrazu

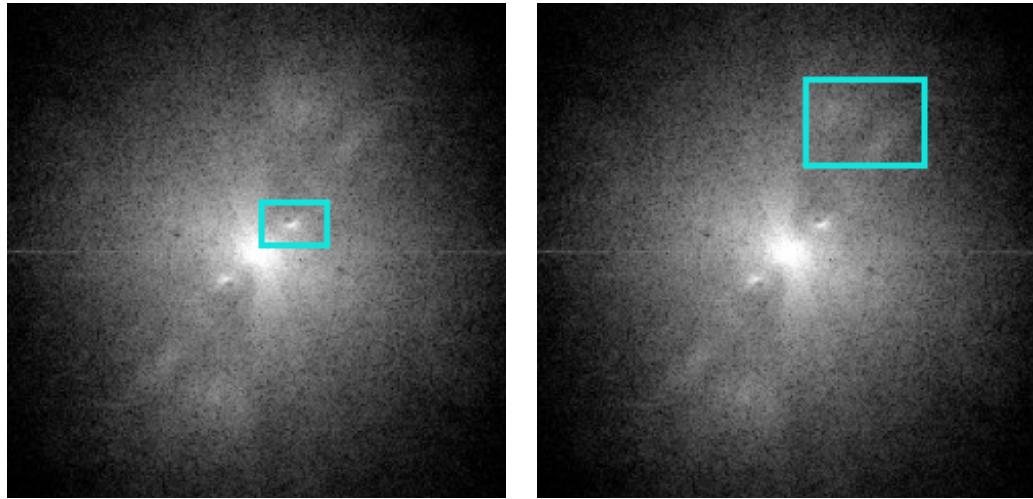
Przykład: obraz ze strukturami powtarzającymi się okresowo: → grafiki 5.19, 5.20  
 Obraz SAR (Synthetic Aperture Radar, radar z syntetyczną aperturą) Oceanu Indyjskiego z falami wodnymi o różnych długościach fal.



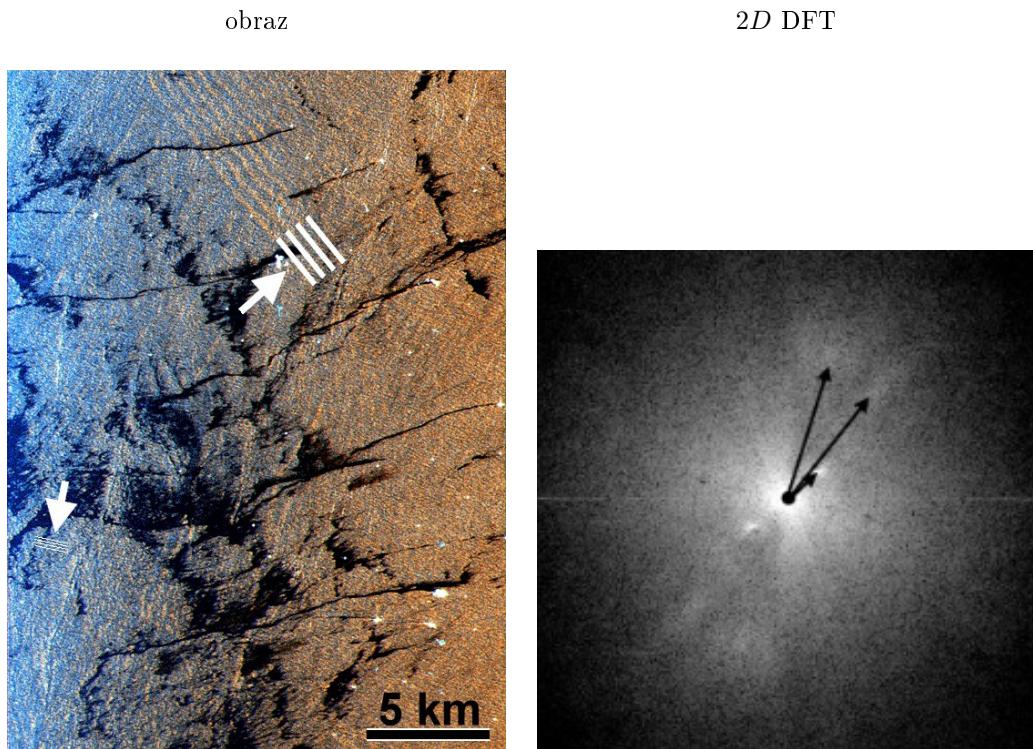
Grafika 5.19: Zdjęcie SAR Oceanu Indyjskiego i jego 2D DFT. Źródło: wikipedia

Na grafice 5.20 dwa jasne odbicia wskazują zarówno kierunek, jak i średnią długość fali regularnych, długookresowych fal wodnych (długość fali ok. 500m, fale wewnętrzne). Długości fal powierzchniowych są bardziej zróżnicowane, dlatego nie tworzą one ostrych zarysów w obrazie 2D DFT. Istnieją dwa podstawowe kierunki propagacji fal, które są słabo widoczne na obrazie satelitarnym. Długości fal wynoszą około 150 m i 160 m ).

Na grafice 5.21 obszary odpowiadające falom wodnym w 2D DFT oznaczone są strzałkami (krótką strzałką wskazuje obszar odpowiadający falom wewnętrznym, długie strzałki odpowiadają falom powierzchniowym).



Grafika 5.20: 2D DFT z grafiki 5.19 z zaznaczonymi obszarami wysokiej amplitudy widma. Źródło: wikipedia



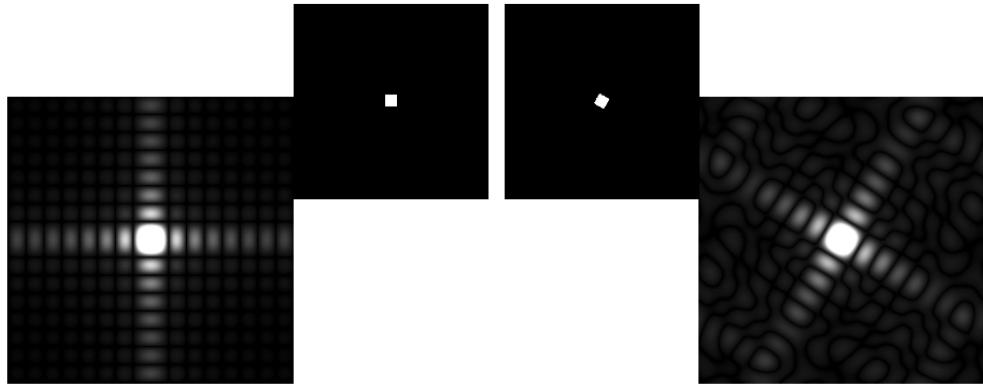
Grafika 5.21: Zdjęcie SAR Oceanu Indyjskiego i jego 2D DFT - strzałkami oznaczone są obszary odpowiadające widocznym na zdjęciu falom. Źródło: wikipedia

### 5.2.4 Niektóre własności transformaty Fouriera

#### Rotacja

→ grafika 5.22

Spektrum  $F(k, l)$  rotuje o ten sam kąt jak obraz  $f(m, n)$ .



Grafika 5.22: Rotacja obrazu.

#### Przesunięcie

→ grafika 5.23

Przesunięcie obrazu  $f(m, n)$  powoduje zmianę fazy spektrum  $F(k, l)$ .

#### Wartość średnia (DC coefficient)

$$F(0, 0) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)$$

#### Skalowanie

$$\begin{aligned} T\{a \cdot f(m, n)\} &= a \cdot F(k, l) \\ T\{f(a \cdot m, b \cdot n)\} &= \frac{1}{ab} F\left(\frac{k}{a}, \frac{l}{b}\right) \end{aligned}$$

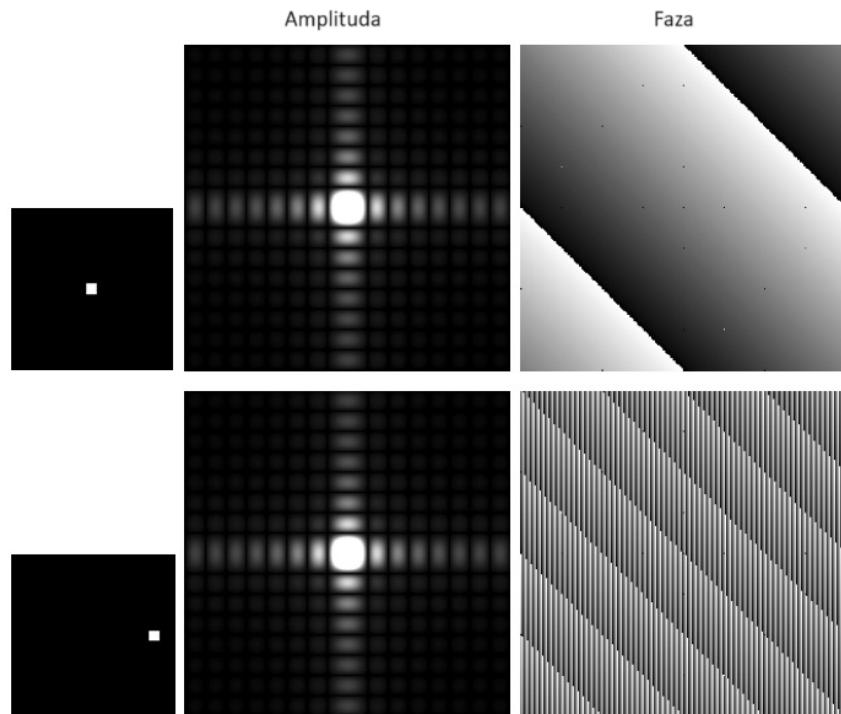
#### Odpowiedź impulsowa (odpowiedź punktowa)

$$T\{\delta(m, n)\} = 1$$

### 5.2.5 Filtrowanie obrazu w obszarze widma

#### Usuwanie artefaktów

→ grafika 5.24



Grafika 5.23: Przesunięcie obrazu.

### Znaczenie niskich/wysokich częstotliwości w obrazie

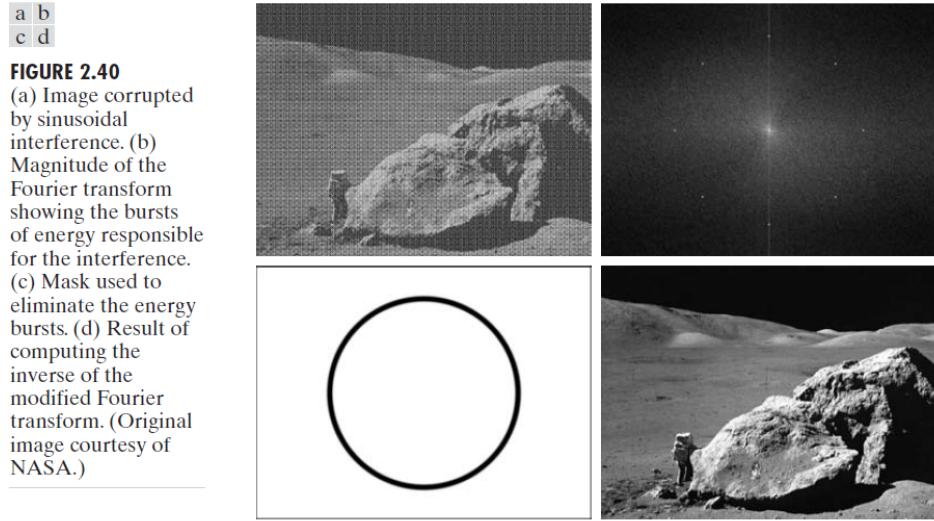
→ grafika 5.25

### Filtrowanie sygnałów - efekt Gibbsa

Rozwinięcie nieciągłej (okresowej) funkcji w szereg Fouriera (tzn. aproksymacja nieciągłej funkcji przez skończone szeregi sinusów / fal płaskich) prowadzi do oscylacji w punktach nieciągłości, które nie zmniejszają się nawet przy próbie aproksymacji funkcji za pomocą dodatkowych wyrazów sumarycznych. Dodanie dodatkowych wyrazów sumarycznych powoduje jedynie, że częstotliwość oscylacji wzrasta, a czas ich trwania maleje, jednak ich maksymalne odchylenie na krótko przed lub po punkcie nieciągłości pozostaje stałe. → grafiki 5.26, 5.27

Efekt Gibbsa jest przyczyną powstawania wysokoczęstotliwościowych oscylacji przy zastosowaniu w obszarze spektrum filtrów o nieciągłych oknach.

→ wybór odpowiedniej charakterystyki filtra w dziedzinie częstotliwości! (filtr Czebyszewa, filtr Gaussa, filtr Butterwortha, ...)



Grafika 5.24: Usuwanie artefaktów. Źródło: Digital Image Processing, Second Edition, R. Gonzalez, Z. Faisal (2019)

### Filtryle liniowe

→ Rozdział "Filtryle liniowe"

Przykład: filtr Gaussa (grafika 5.28)

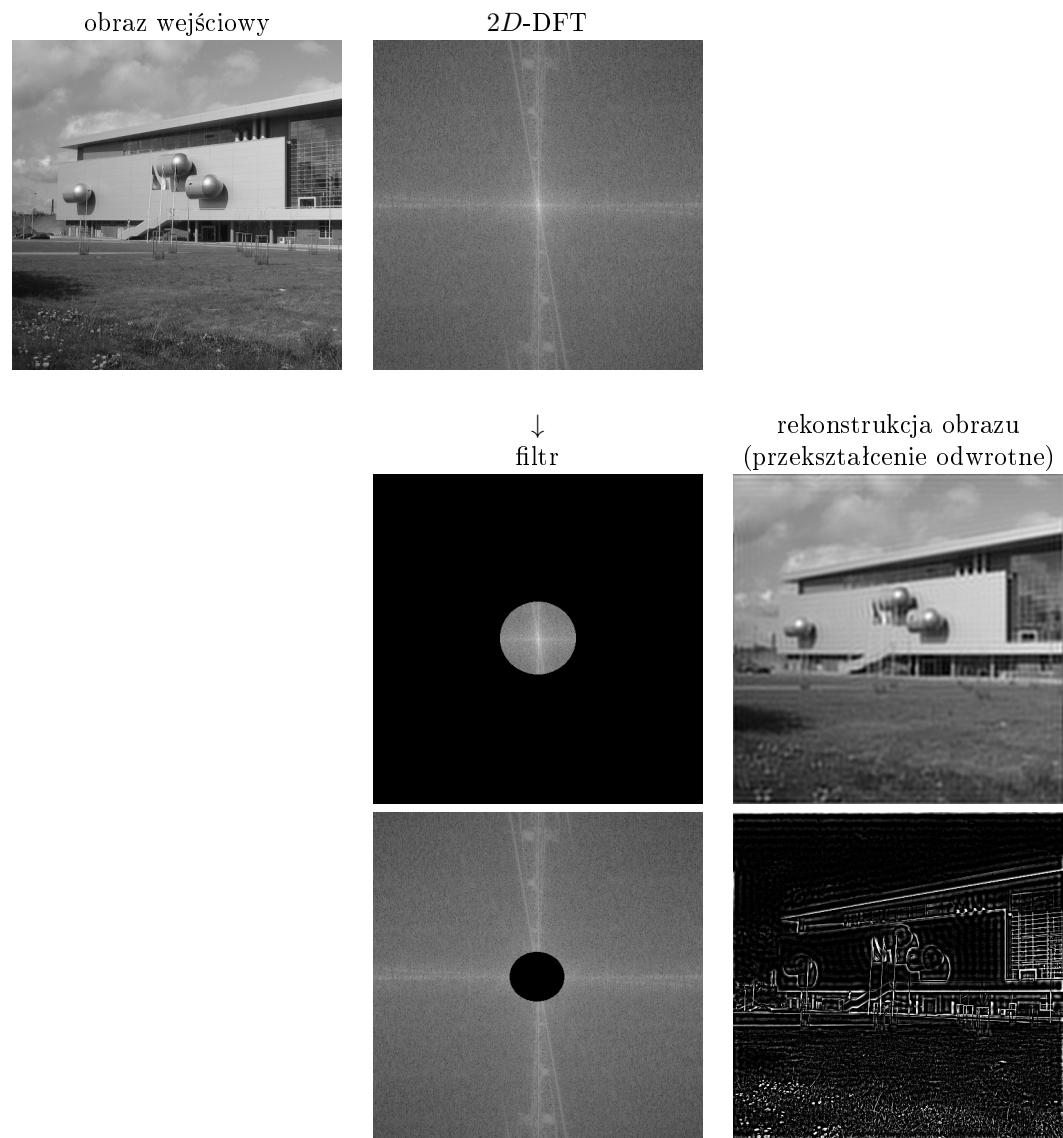
### 5.2.6 Fast Fourier Transform FFT

→ modyfikacja algorytmu DFT

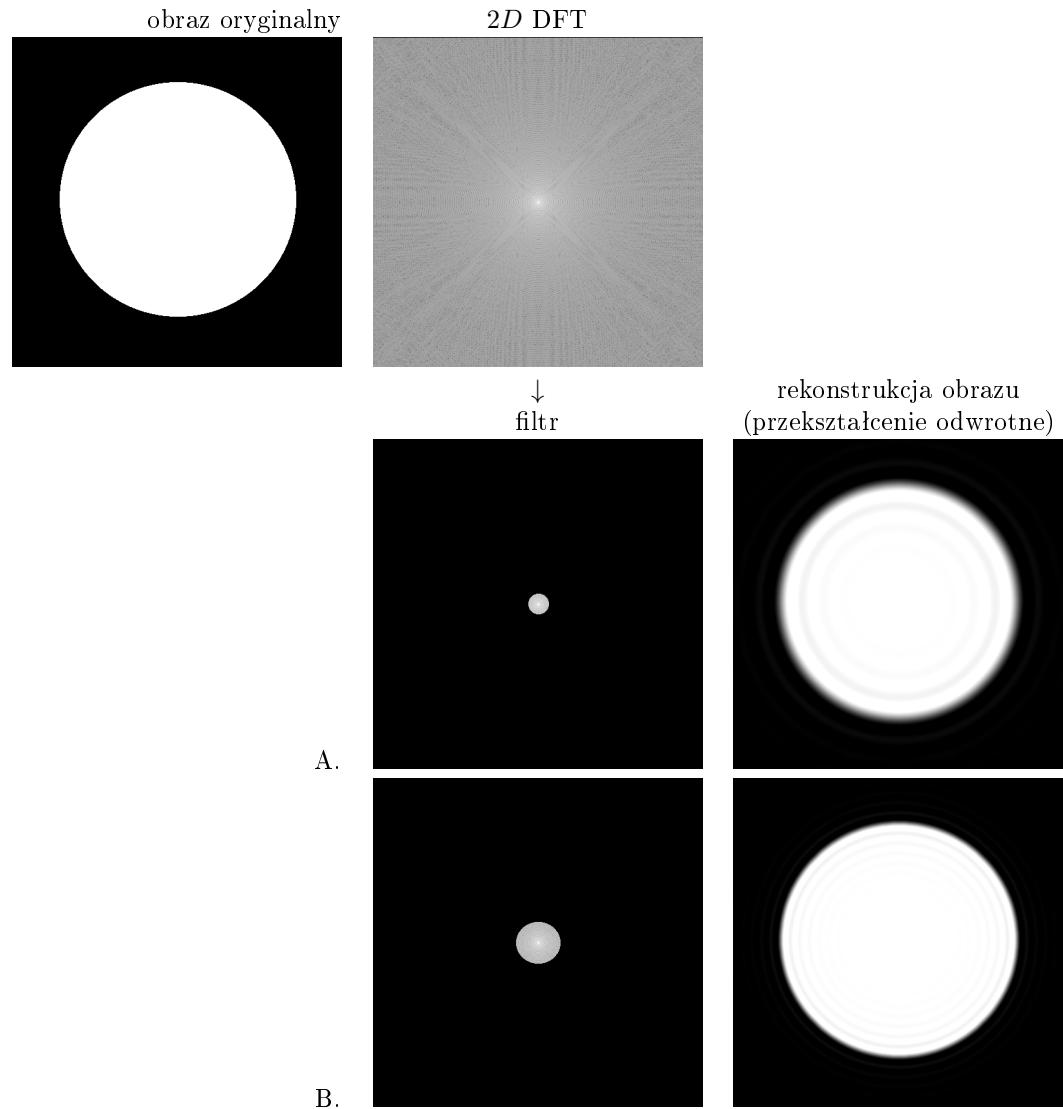
- Założenie upraszczające algorytm:  $N = 2^k, 1 < k \in \mathbb{N}$
- FFT opiera się na technice "dziel i zwyciężaj":  
 ↳ DFT sygnału obliczana jest rekurencyjnie z dwóch składowych sygnału.

Algorytm Cooley i Tukey:

```
function FFT( $n, \vec{f}$ )
  if  $n = 1$  then return  $f$ 
  else
     $\vec{g} := \text{FFT}\left(\frac{n}{2}, (f_0, f_2, \dots, f_{n-2})\right);$ 
     $\vec{u} := \text{FFT}\left(\frac{n}{2}, (f_1, f_3, \dots, f_{n-1})\right);$ 
    for  $k = 0$  to  $\frac{n}{2} - 1$  do {
       $c_k := g_k + u_k \cdot e^{-2\pi ik/n}$ 
       $c_{k+\frac{n}{2}} := g_k - u_k \cdot e^{-2\pi ik/n}$ 
    }
```



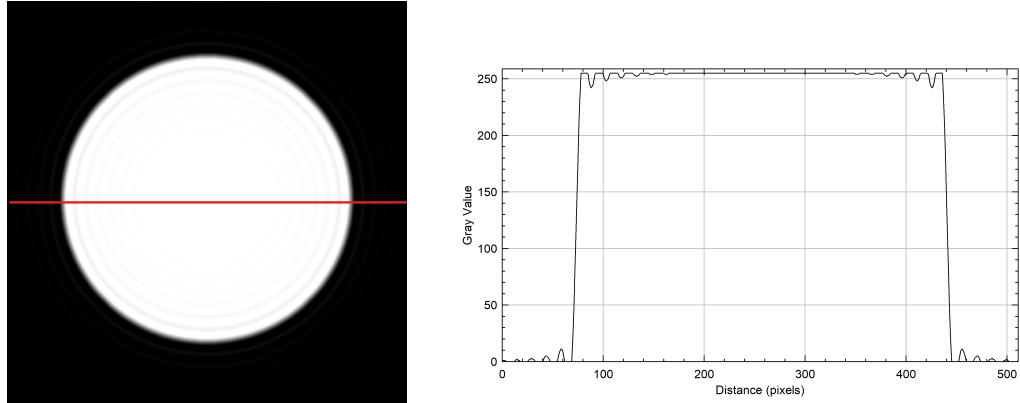
Grafika 5.25: Znaczenie niskich i wysokich częstotliwości w obrazie



Grafika 5.26: Efekt Gibbsa - oscylacje przy punktach nieciągłości obrazu (krawędziach obiektu)  $\rightarrow$  idealny filtr dolnoprzepustowy.

```
    };
    return c;
```

- W przetwarzaniu obrazów rzad złożoności 2D DFT to  $O(N^4)$ , podczas gdy FFT  $O(N^2 \log N)$ .



Grafika 5.27: Efekt Gibbsa - profil liniowy wzdłuż zaznaczonej linii na zrekonstruowanym obrazie B. z grafiki 5.26

### 5.2.7 Dyskretna transformacja kosinusowa DCT (*Discrete Cosine Transform*)

Chociaż w praktyce wiele sygnałów ma wartości rzeczywiste, dyskretna transformata Fouriera jest transformacją o wartościach zespolonych. W przeciwieństwie do dyskretnej transformaty Fouriera, DCT jest transformacją rzeczywistą i daje rzeczywiste współczynniki.

Podobnie jak inne dyskretne transformaty częstotliwości, dyskretna transformata kosinusowa wyraża skończoną sekwencję sygnałów wejściowych jako skończoną sumę ważonych funkcji trygonometrycznych o różnych częstotliwościach. W transformacji kosinusowej używane są tylko funkcje kosinus.

W przypadku dyskretnej transformaty Fouriera założona jest kontynuacja okresowa ciągu, a w przypadku dyskretnej transformaty kosinusowej to parzysta<sup>2</sup> kontynuacja ciągu wejściowego. Najczęściej stosowana konwencja to DCT-II. (→ grafika 5.29)

#### DCT (DCT-II)

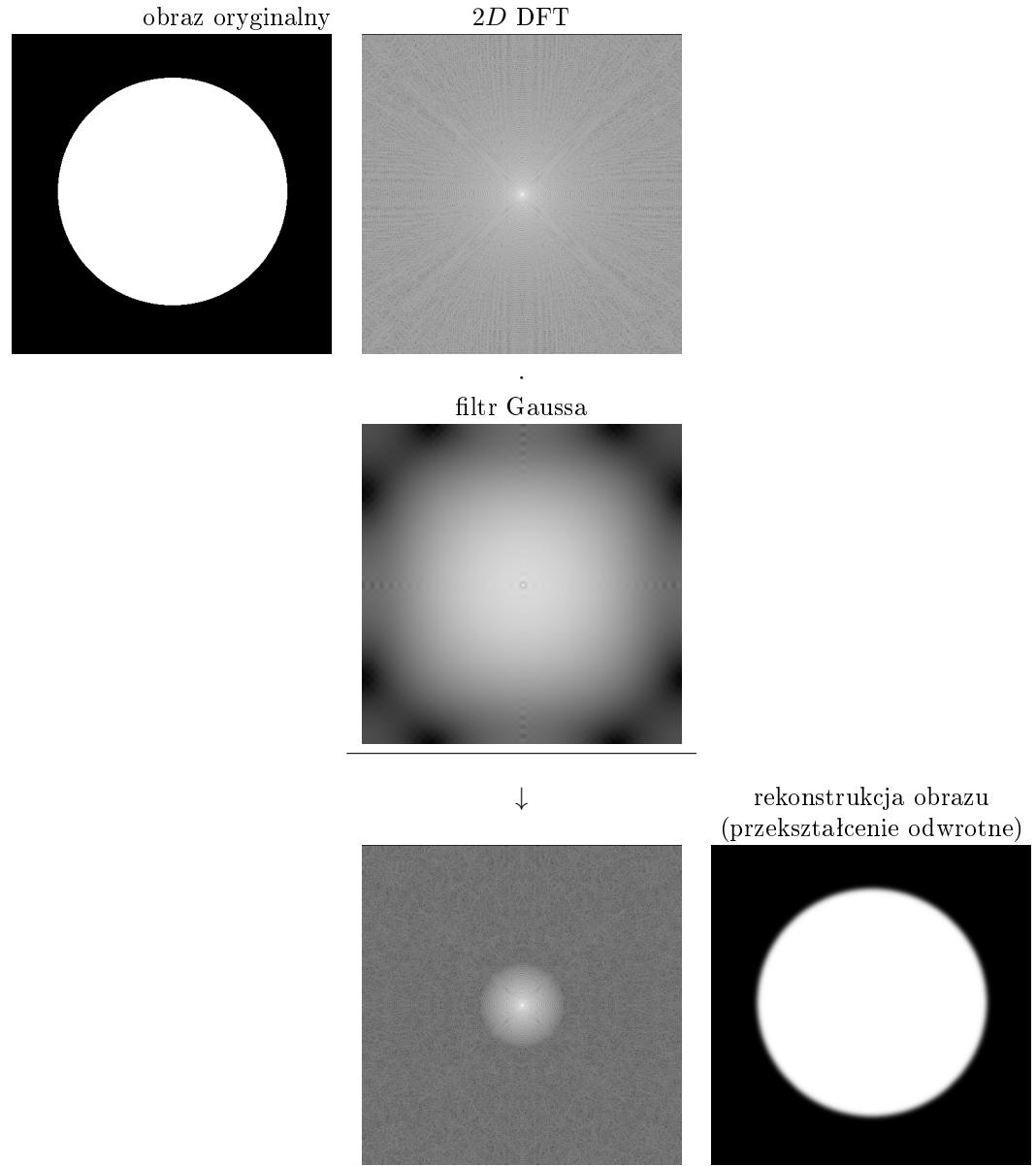
$$f(n) \quad \text{ciąg, } n \in \{0, \dots, N-1\}$$

$$F(k) = \sum_{n=0}^{N-1} f(n) \cos \left[ \frac{\pi}{N} k(n+1) \right] \text{ dla } k = 0, \dots, N-1$$

Z wyjątkiem stałego współczynnika<sup>3</sup> 2, DCT odpowiada DFT dla ciągu liczb rzeczywistych o długości  $4N$  o parzystej symetrii, gdzie wszystkie elementy o parzystym indeksie mają wartość 0.

<sup>2</sup>Dla parzystej kontynuacji ciągu możliwych jest osiem form warunków wartości brzegowych. Przewodzi to do różnych form dla DCT, które są określone w literaturze jako DCT-I do DCT-VIII.

<sup>3</sup>Stale współczynniki DCT nie są konsekwentnie zdefiniowane w literaturze.

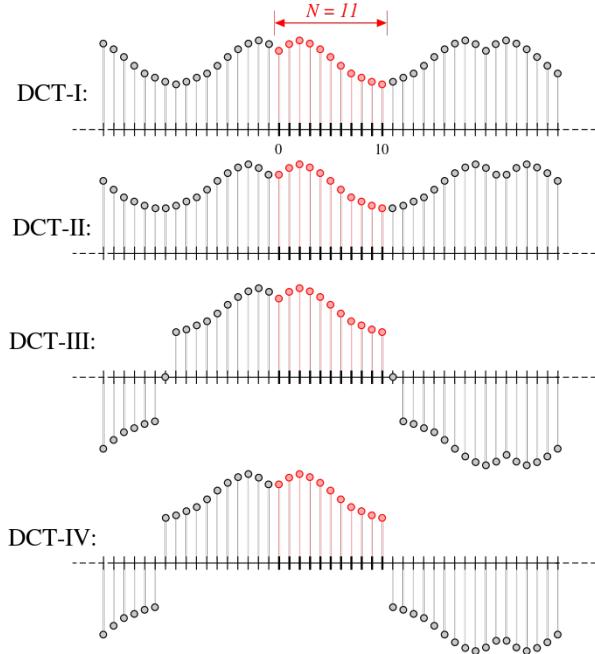


Grafika 5.28: Filtr Gaussa

### Przekształcenie odwrotne (IDCT, *Inverse transform*)

(w przypadku DCT-II przekształcenie odwrotne odpowiada DCT-III)

$$f(n) = \frac{2}{N} \left( \frac{F(0)}{2} + \sum_{k=1}^{N-1} F(k) \cos \left[ \frac{\pi}{N} n(k+1) \right] \right) \text{ dla } n = 0, \dots, N-1$$



Grafika 5.29: Parzysta kontynuacja wartości ciągu na przykładzie wejściowej sekwencji danych z 11 wartościami (na czerwono) w ramach czterech standardowych wariantów DCT (DCT-I do DCT-IV).

Źródło: [https://commons.wikimedia.org/wiki/Category:Discrete\\_cosine\\_transform](https://commons.wikimedia.org/wiki/Category:Discrete_cosine_transform)

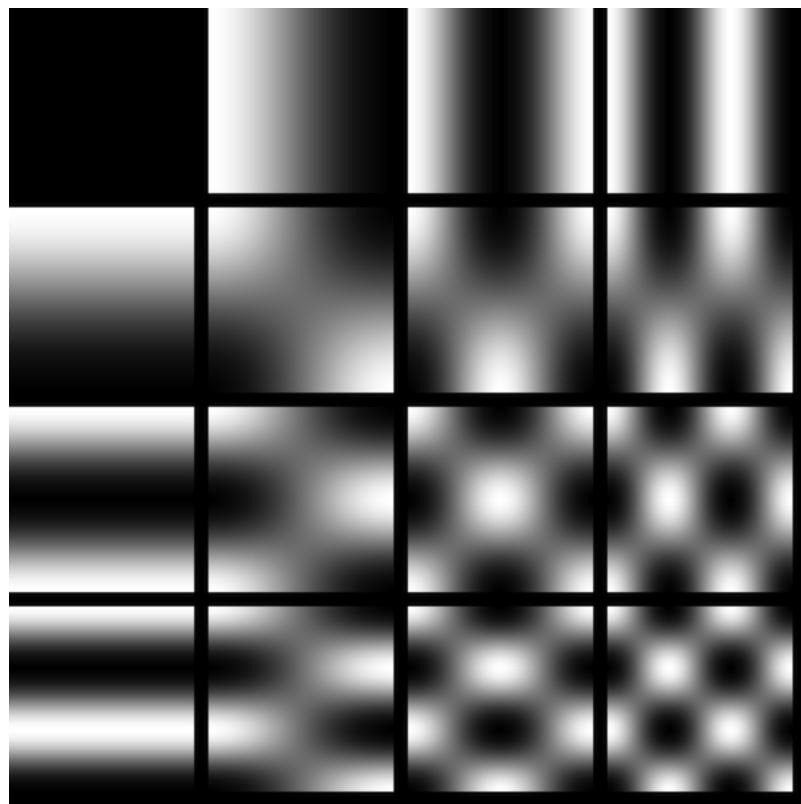
## 2D - DCT

$$F(k, l) = \sum_{m=0}^M \sum_{n=0}^N f(m, n) \cos \left[ \frac{\pi}{M} \left( m + \frac{1}{2} \right) k \right] \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) l \right] \quad \text{dla } k = 0, \dots, M \\ \text{oraz } l = 0, \dots, N$$

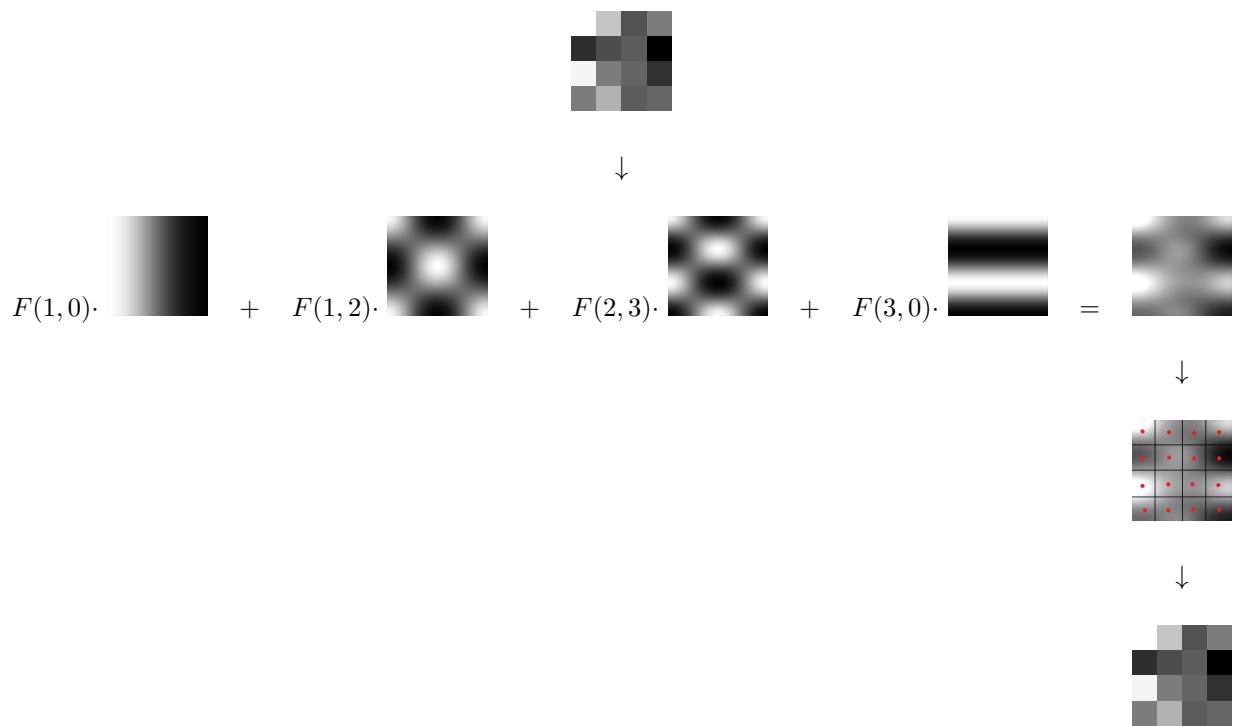
### Przykład $M, N = 4$

Obraz jest superpozycją obrazów bazowych (grafika 5.30)

Przykładowy obraz i jego złożenie: grafika 5.31



Grafika 5.30: Obrazy bazowe 2D-DCT o rozmiarze  $M, N = 4$ . Źródło: Visualizing DFT and DCT Basis, DPReview



Grafika 5.31: Przykład: obraz  $4 \times 4$  jako superpozycja funkcji bazowych (2D-DCT)