Przetwarzanie informacji obrazowej

Temat 5

Piotr Syga

2022

Intuicja

Załóżmy, że mamy sygnał sygnał sinusoidalny S. Zadziałajmy na S filtrem (o charakterystyce impulsowej) $\psi(x)$, dostaniemy wtedy sinusoidę S', taką, że $S'(x) = S(X) * \psi(x) = e^{i\omega x} * \psi(x) = Ae^{i\omega x+\phi}$.
Zbierzmy wartości amplitudy i przesunięcia fazowego dla różnych częstotliwości ω:

$$\mathcal{F}\{\psi(x)\} = Ae^{i\phi} \ .$$

Zatem transformata Fouriera, to odpowiedź układu na zespoloną sinusoidę przepuszczoną przez filtr $\phi(x)$.

Definicje

Transformata Fouriera:

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

Transformata odwrotna:

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

Dyskretna transformata:

$$\hat{f}(p) = \sum_{k=0}^{n-1} f(x) e^{-i \frac{2\pi pk}{n}}$$

Dyskretna transformata (2D):

$$\hat{f}(p,q) = \sum_{j=0}^{m-1} \sum_{k=0}^{n-1} f(j,k) e^{-2\pi i (\frac{pj}{n} + \frac{qk}{m})}$$

Uwaga: Dla dyskretnych transformat często wprowadza się normalizację.

Obliczanie DFT

$$\hat{f}(p) = \sum_{k=0}^{n-1} f(x) e^{-i\frac{2\pi pk}{n}}$$

$$f(x) = \frac{1}{n} \sum_{k=0}^{n-1} \hat{f}(k) e^{i\frac{2\pi kx}{n}}$$

Wyliczenie bezpośrednie zarówno $\hat{f}(p)$ jak i $f(x)$ wymaga $O(n^2)$ operacji. Wykorzystanie FFT pozwala zredukować złożoność do $O(n \log n)$.

FFT

Istnieje wiele algorytmów FFT różniących się założeniami, każdy ze znanych algorytmów ma taką samą złożoność asymptotyczną.

CT FFT

Założenia: n – potęga 2.
Idea: Obliczenie FFT dla próbek parzystych $f(2 \cdot l)$ oraz $f(2 \cdot l + 1)$, wykorzystanie rekurencji i połączenie wyników

$$\hat{f}(p) = \sum_{l=0}^{\frac{n}{2}-1} f(2l) e^{-(2lp)\frac{2\pi i}{n}} + \sum_{l=0}^{\frac{n}{2}-1} f(2l+1) e^{-((2l+1)p)\frac{2\pi i}{n}}$$

$$\hat{f}(p) = \sum_{l=0}^{\frac{n}{2}-1} f(2l) e^{-(lp)\frac{2\pi i}{n/2}} + e^{\frac{-2\pi p \cdot i}{n}} \sum_{l=0}^{\frac{n}{2}-1} f(2l+1) e^{-(lp)\frac{2\pi i}{n/2}}$$

Zatem możemy rozpisać wyliczenie $\hat{f}(p) = E(p) + e^{-\frac{2\pi ip}{n}} O(p)$.
Korzystamy z okresowości funkcji trygonometrycznych:
$\hat{f}(p + \frac{n}{2}) = \cdots = E(p) - e^{-\frac{2\pi ip}{n}} O(p)$.
Ostatecznie obliczamy DFT rozmiaru 2.

CT FFT

Obliczenie FFT rozmiaru $n = n_1 n_2$ rozkładamy na

1. Obliczenie n_1 DFT rozmiaru n_2.
2. Wykonanie mnożeń przez pierwiastek z jedności.
3. Wykonanie n_2 DFT rozmiaru n_1.

Zapamiętujemy wartości pośrednie. Możemy wykorzystać fakt, że dla sygnałów rzeczywistych, mamy symetrię $\hat{f}(n-p) = \hat{f}(p)^*$

Specjalne przypadki

- Istnieją algorytmy pozwalające na obliczenie DFT istotnie poniżej progu $O(n \log n)$ kosztem błędu zwiększającego się wraz z przyspieszeniem.
- Jeśli wiemy, że tylko M częstotliwości jest obecnych w funkcji, możemy obliczyć DFT w czasie $O(M \log n \log(\frac{n}{M}))$. Poprzez estymację M możemy uzyskać szybki algorytm obliczania aproksymacji DFT.

Częstotliwość przestrzenna

Spatial frequency

Miara określająca liczbę cykli sinusoidy na jednostkę odległości.

Transformata Fouriera obrazu będzie zawierała liczbę punktów odpowiadającym liczbie częstotliwości przestrzennych na oryginalnym obrazie. Położenie punktów po FT zależne jest od częstotliwości przestrzennych obrazu, natomiast ich intensywność oznacza kontrast (a więc kształt sygnału sinusa) na oryginalnym obrazie.

Częstotliwość przestrzenna

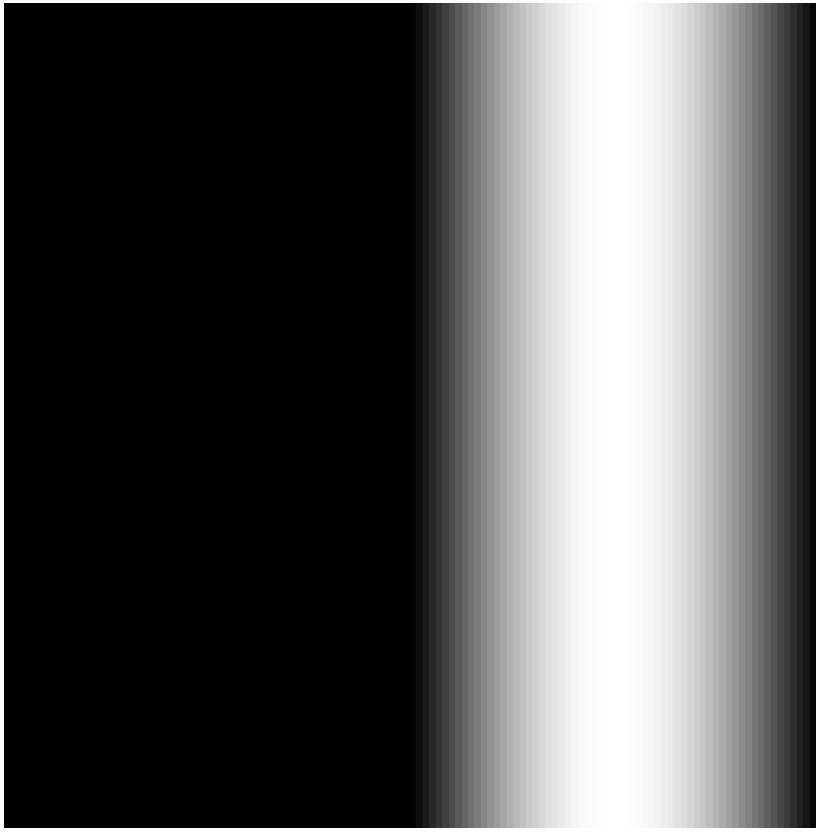
Przykład do policzenia na palcach:

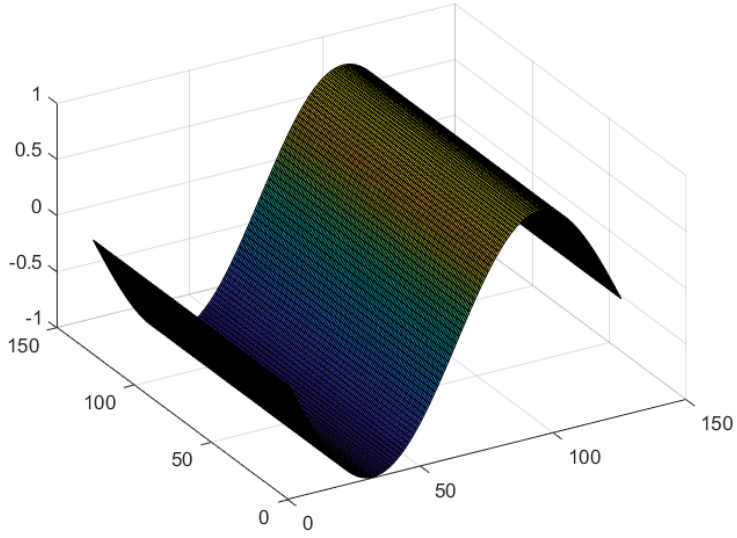
$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 8 & 0 \end{bmatrix}$$

$$\hat{f}(A) = \begin{bmatrix} 29.0000 + 0.0000i & -7.0000 - 5.1962i & -7.0000 + 5.1962i \\ -5.5000 - 6.0622i & 8.0000 + 1.7321i & -5.5000 - 6.0622i \\ -5.5000 + 6.0622i & -5.5000 + 6.0622i & 8.0000 - 1.7321i \end{bmatrix}$$

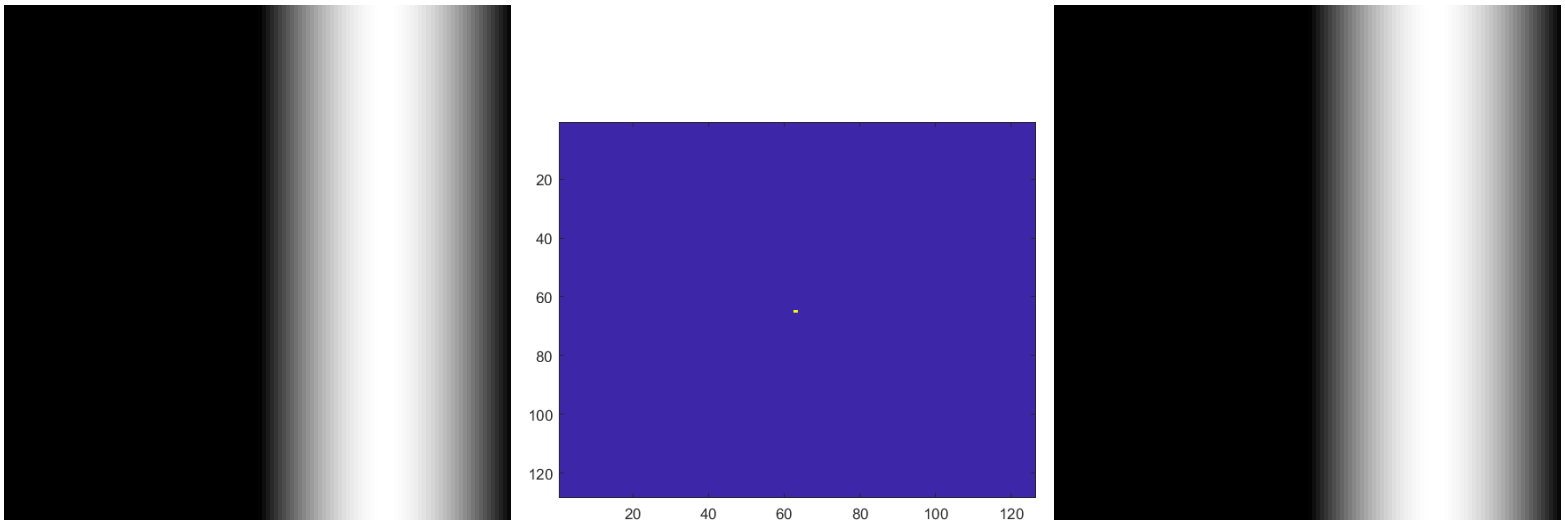
$$\hat{f}^{-1}(\hat{f}(A)) = \begin{bmatrix} 1.0000 & 2.0000 & 3.0000 \\ 4.0000 & 5.0000 & 6.0000 \\ 0 & 8.0000 & 0 \end{bmatrix}$$

Przykłady





Przykłady

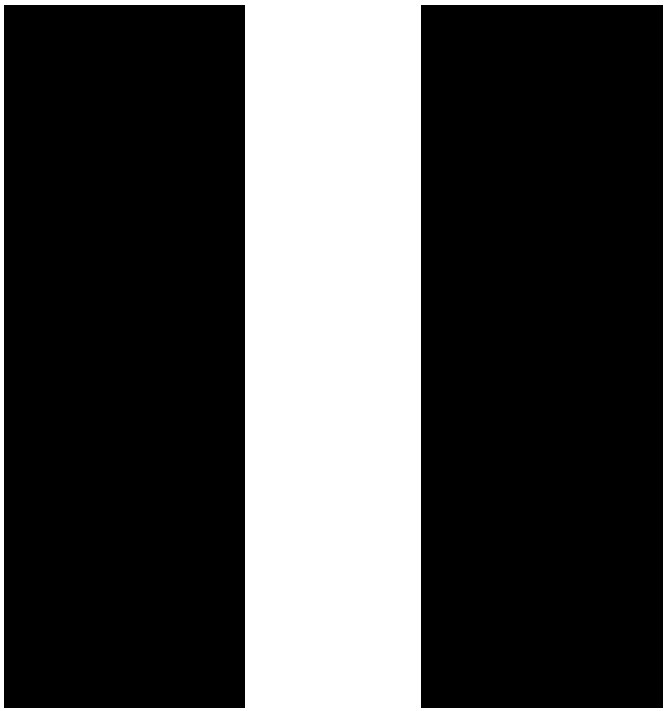


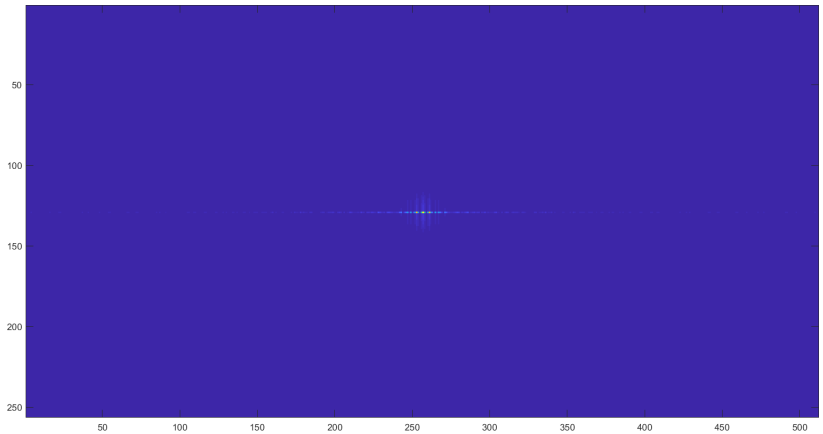
Przykłady



Uwaga: Podatność na błędu numeryczne, np. $-1.2032e-13$ zamiast 0.

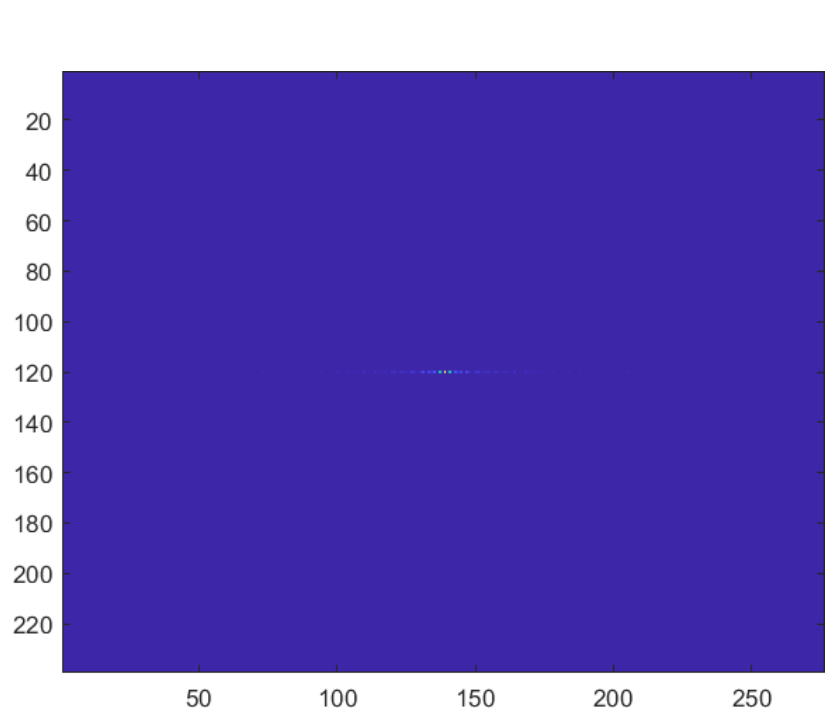
Przykłady





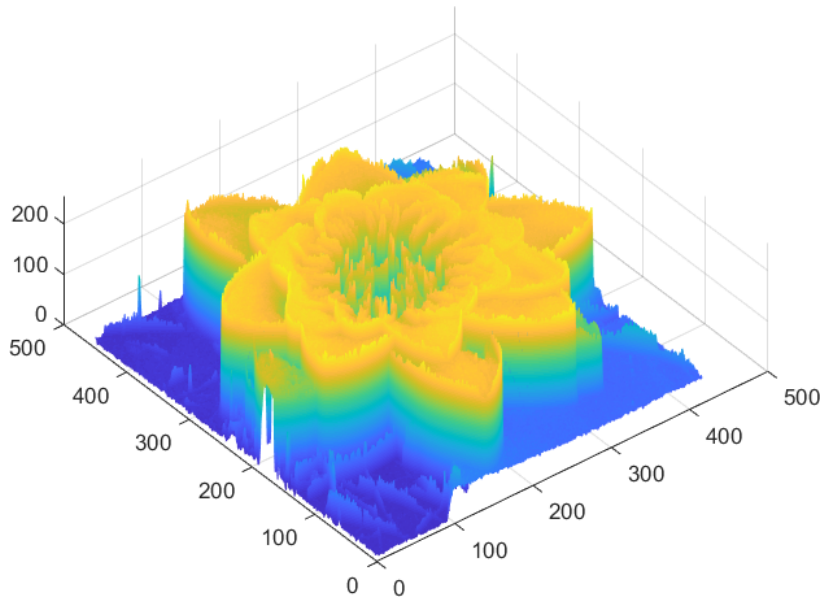
Przykłady



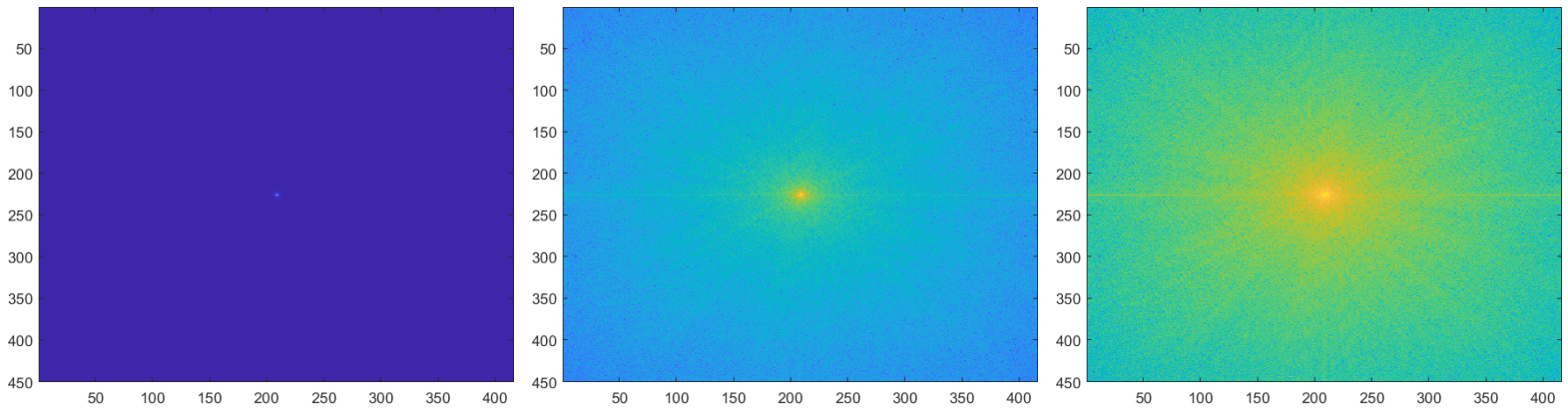


Przykłady

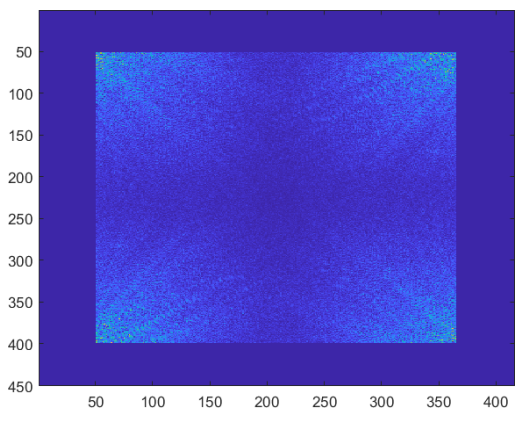


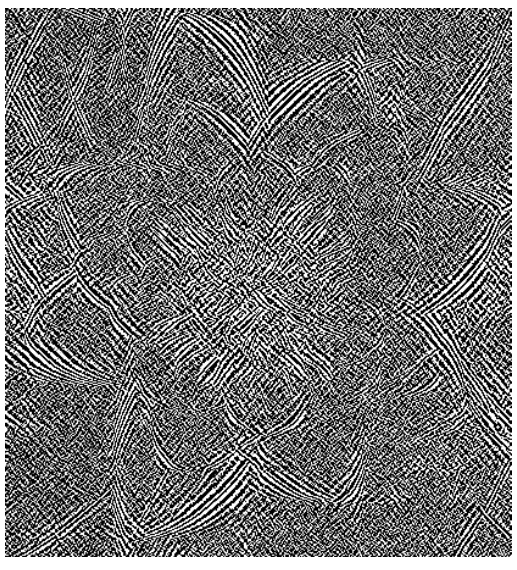


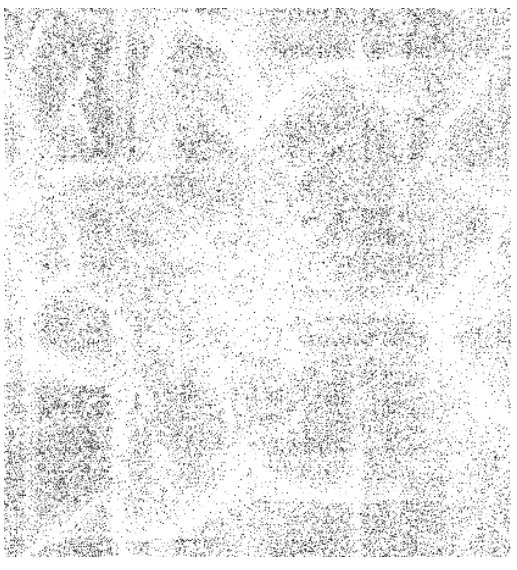
Przykłady



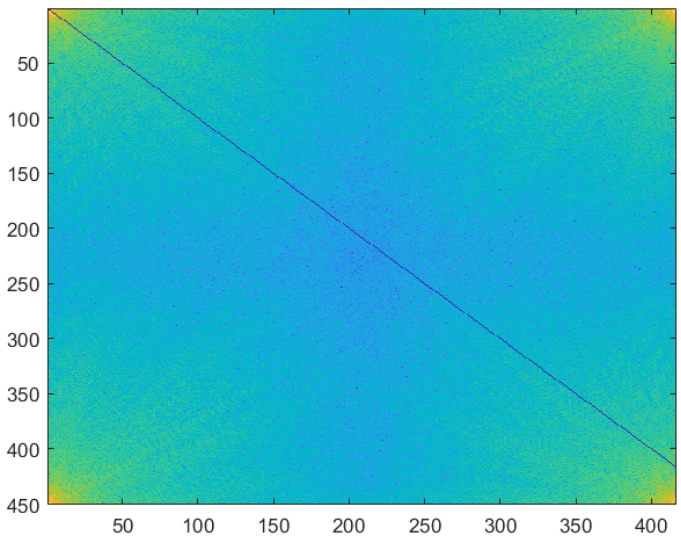
Przykłady

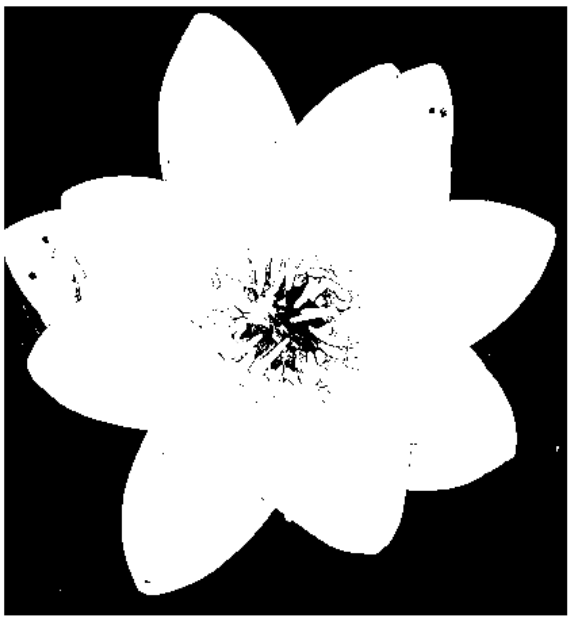




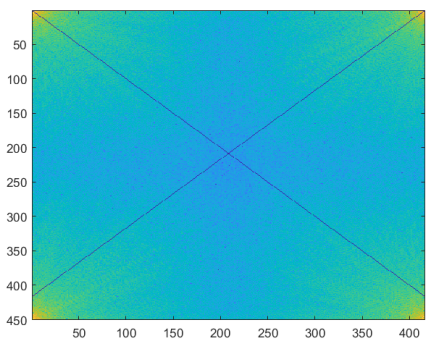


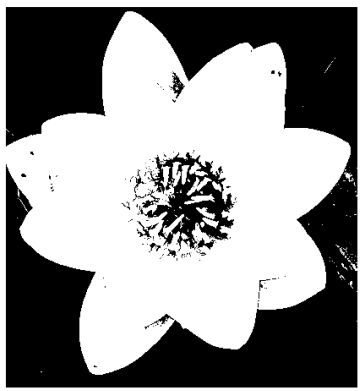
Przykłady

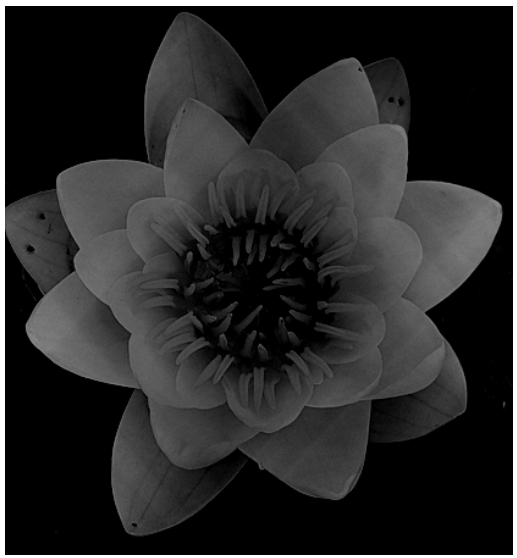




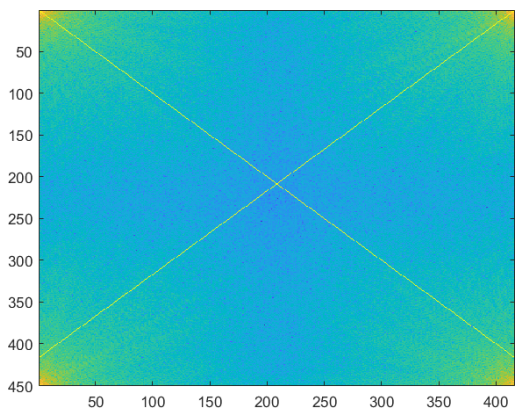
Przykłady

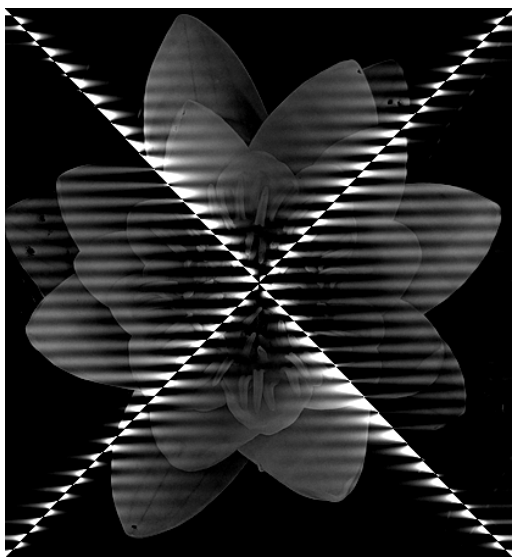


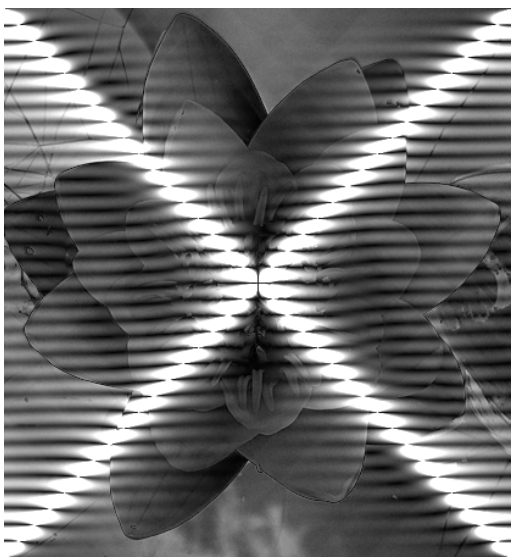




Przykłady







Funkcje i ich FT

$$\delta(x) \ : \ 1$$

$$\delta(x+\phi) \ : \ e^{-i\omega\phi}$$

$$\text{box}(\tfrac{x}{a}) \ : \ a\text{sinc}(a\omega) = \begin{cases} \frac{1}{\omega}\sin(a\omega) & \omega \neq 0 \\ a & \omega = 0 \end{cases}$$

$$G(x;\sigma) \ : \ \frac{\sqrt{2\pi}}{\sigma} G(\omega;\sigma^{-1})$$

$$LoG(x;\sigma) \ : \ -\frac{\sqrt{2\pi}}{\sigma}\omega^2 G(\omega;\sigma^{-1})$$

$$\text{triangle}(\tfrac{x}{a}) \ : \ a(\text{sinc}(a\omega))^2$$

Obliczanie 2D FFT

We wcześniejszym algorytmie podstawiamy za $f(x)$ macierz dwuwymiarową o wymiarach $n \times m$.
Obliczamy najpierw FFT dla kolumn, a następnie wierszowo zapamiętując pośrednie wyniki.

FT – Własności

- liniowość
- symetria
- sprzężenie
- przesunięcie
- twierdzenie Rayleigha o energii
- downsampling
- korelacja
- moc
- ...

Wybrane własności

Liniowość

Dla dowolnych $x, y \in \mathbb{C}^n$ oraz dowolnych skalarów $a, b \in \mathbb{C}$ zachodzi $\hat{f}(ax + by) = a\hat{f}(x) + b\hat{f}(y)$.

$\hat{f}_p(ax + by) = \sum_{k=0}^{n-1} f(ax_k + by_k)e^{-2\pi ip\frac{k}{n}} =$

$a\sum_k f(x_k)e^{-2\pi ip\frac{k}{n}} + b\sum_k f(y_k)e^{-2\pi ip\frac{k}{n}} = a\hat{f}_p(x) + b\hat{f}_p(y).$

Wybrane własności

Sprzężenie – odwrócenie

Dla dowolnego $x \in \mathbb{C}^n$ zachodzi $\hat{f}(\bar{x}) = \mathrm{Flip}(\overline{\hat{f}(x)})$ oraz $\mathrm{Flip}(\hat{f}(\bar{x})) = \overline{\hat{f}(x)}$.

$\hat{f}_p(\bar{x}) = \sum_{k=0}^{n-1} f(\bar{x}) e^{-2\pi i p \frac{k}{n}} = \sum_k \overline{f(x) e^{2\pi i p \frac{k}{n}}} = \overline{\sum_k f(x) e^{-2\pi i k \frac{-p}{n}}} = \mathrm{Flip}(\overline{\hat{f}_p(x)})$

Wybrane własności

Przesunięcie

Dla dowolnego $x \in \mathbb{C}^n$ oraz $\phi \in \mathbb{Z}$ zachodzi:
$\hat{f}_p(\text{Shift}(x, \phi)) = e^{-i\omega_p \phi} \hat{f}_p(x)$.
$\hat{f}_p(\text{Shift}(x, \phi)) = \sum_{k=0}^{n-1} f(x_k - \phi) e^{-2\pi i p \frac{k}{n}} = \sum_{k=-\phi}^{n-1-\phi} f(x_k) e^{-2\pi i p \frac{k+\phi}{n}} =$
$e^{-2\pi i p \frac{\phi}{n}} \sum_k f(x_k) e^{-2\pi i p \frac{k}{n}} = e^{-i\omega_p \phi} \hat{f}_p(x)$

Wybrane własności

Korelacja

Korelacją dwóch sygnałów x, y nazywam $(x \star y)_n = \sum_{m=0}^{n-1} \overline{x(m)} y(m+n)$.

Dla dowolnych x, $y \in \mathbb{C}^n$ zachodzi $\hat{f}(x \star y) = \overline{\hat{f}(x)} \cdot \hat{f}(y)$.

$(x \star y)_n = \sum_{m=0}^{n-1} \overline{x(m)} y(m+n) = \sum_m \overline{x(-m)} y(n-m) = \overline{\hat{f}(x)} \cdot \hat{f}(y)$

Convolution Theorem

Twierdzenie o konwolucji

Dla dowolnych $x, y \in \mathbb{C}^n$ zachodzi $\hat{f}(x * y) = \hat{f}(x) \cdot \hat{f}(y)$.
$\hat{f}_p(x * y) = \sum_{k=0}^{n-1} f(x_k * y_k) e^{-2\pi i p \frac{k}{n}} =$
$\sum_{k=0}^{n-1} \sum_{m=0}^{n-1} f(x_m) f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} = \sum_m f(x_m) \sum_k f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} =$
$\sum_m f(x_m) e^{-2\pi i p \frac{m}{n}} \hat{f}(y_k) = \hat{f}(x_k) \hat{f}(y_k)$.

Convolution Theorem

Twierdzenie o konwolucji

Dla dowolnych $x, y \in \mathbb{C}^n$ zachodzi $\hat{f}(x * y) = \hat{f}(x) \cdot \hat{f}(y)$.
$\hat{f}_p(x * y) = \sum_{k=0}^{n-1} f(x_k * y_k) e^{-2\pi i p \frac{k}{n}} =$
$\sum_{k=0}^{n-1} \sum_{m=0}^{n-1} f(x_m) f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} = \sum_m f(x_m) \sum_k f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} =$
$\sum_m f(x_m) e^{-2\pi i p \frac{m}{n}} \hat{f}(y_k) = \hat{f}(x_k) \hat{f}(y_k)$.

Odwrotne twierdzenie o konwolucji

Dla dowolnych $x, y \in \mathbb{C}^n$ zachodzi $\hat{f}(x \cdot y) = \hat{f}(x) * \hat{f}(y)$.

Convolution Theorem

Twierdzenie o konwolucji

Dla dowolnych $x, y \in \mathbb{C}^n$ zachodzi $\hat{f}(x * y) = \hat{f}(x) \cdot \hat{f}(y)$.
$\hat{f}_p(x * y) = \sum_{k=0}^{n-1} f(x_k * y_k) e^{-2\pi i p \frac{k}{n}} =$
$\sum_{k=0}^{n-1} \sum_{m=0}^{n-1} f(x_m) f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} = \sum_m f(x_m) \sum_k f(y_{k-m}) e^{-2\pi i p \frac{k}{n}} =$
$\sum_m f(x_m) e^{-2\pi i p \frac{m}{n}} \hat{f}(y_k) = \hat{f}(x_k) \hat{f}(y_k)$.

Odwrotne twierdzenie o konwolucji

Dla dowolnych $x, y \in \mathbb{C}^n$ zachodzi $\hat{f}(x \cdot y) = \hat{f}(x) * \hat{f}(y)$.

Wniosek: Złożoność bezpośredniego algorytmu $x * y$ wynosi $O(n^2)$, złożoność $\mathcal{F}^{-1}\{\mathcal{F}\{x\} \cdot \mathcal{F}\{y\}\}$ wynosi $O(n \log n)$.

Okienkowana transformata Fouriera

- zamiast transformaty całego obrazu, mogą nas interesować lokalne zmiany spektrum (względy mocy obliczeniowej lub sposób dostarczenia sygnału)
- lokalna korekcja szumów
- wykorzystujemy okienko, które jest niezerowe tylko na pewnym odcinku

$\mathcal{F}_{\mathrm{ST}}\{x[n]\}(m,\omega) = \hat{f}(m,\omega) = \sum_{m=-\infty}^{\infty} x[n]w[n-m]e^{-i\omega n}$, dla sygnały x, okienka w.
$|\hat{f}(m,\omega)|^2$ nazywamy gęstością mocy spektralnej.

Poza STFT

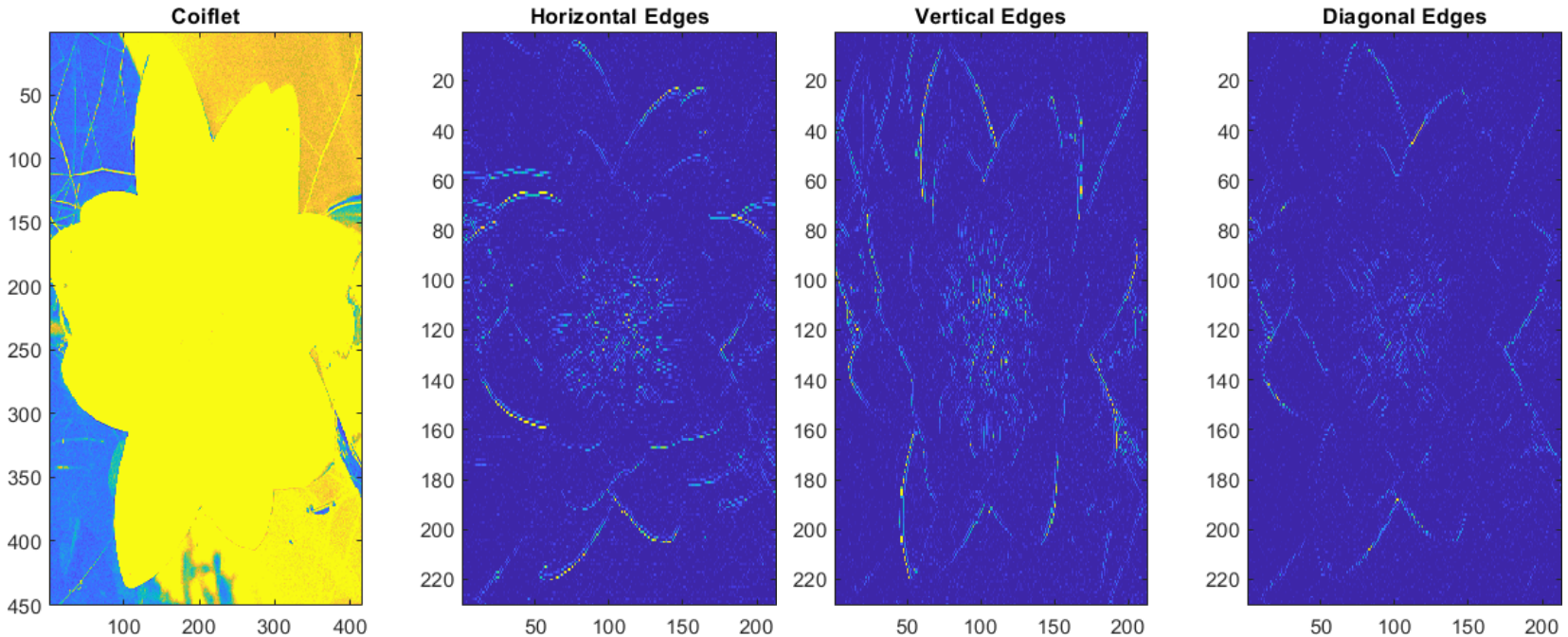
- analizując konkretny obraz może nam zależeć na większej dokładności niż przy STFT
- możemy rozważać inne niż sinusowe składowe sygnału
- chcemy by dyskretny odpowiednik modyfikacji wciąż był względnie szybki do wyliczenia, by był odwracalnym operatorem liniowym

Transformaty falkowe:
$\Psi_{ab}(t) = \frac{1}{\sqrt{|a|}}\psi(\frac{t-b}{a})$, dla funkcji falkowej ψ, przesunięcia b i reskalowania a.
DWT:
$\phi(j,k) = \frac{1}{\sqrt{M}}\sum_x f(x)\phi_{j,k}(x)$, dla funkcji f, ϕ

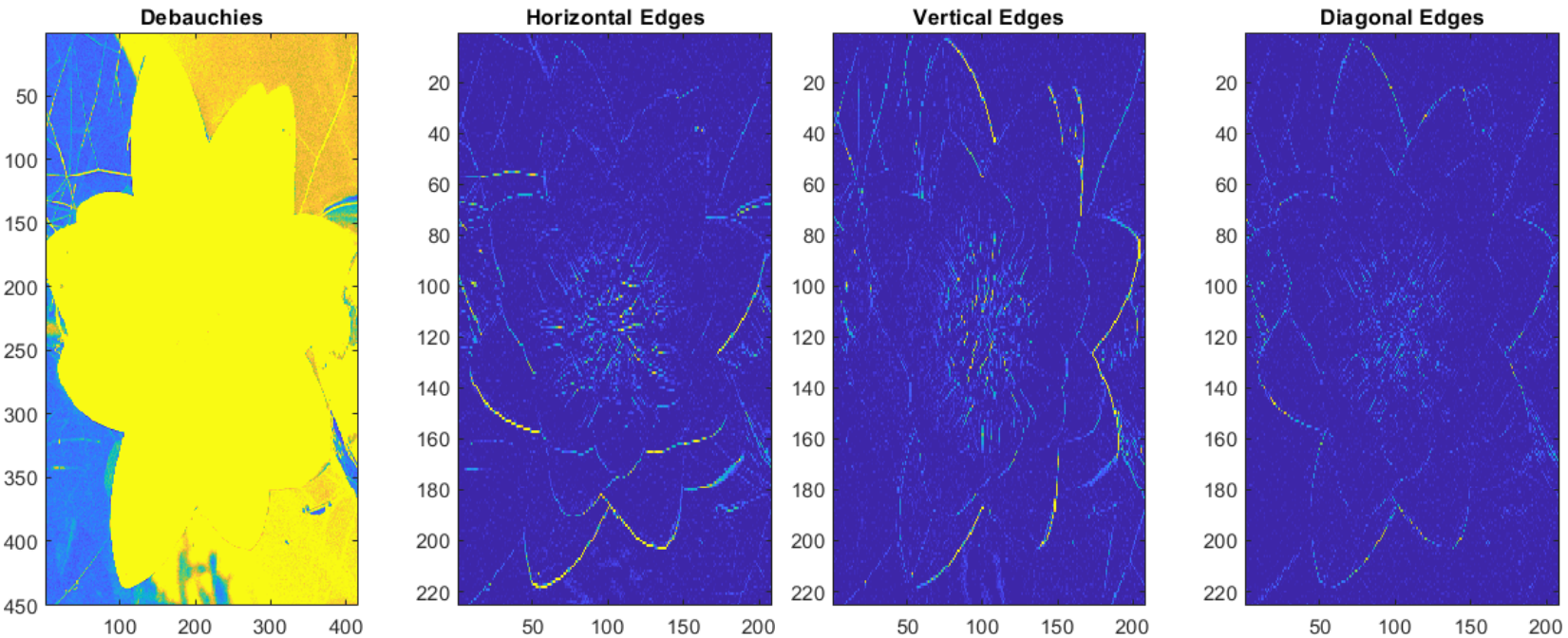
Zastosowania

- wykrywanie osobliwości
- statystyka
- kompresja obrazów
- biometria
- CRM
- modelowanie zjawisk

Przykłady



Przykłady


Debauchies
Horizontal Edges
Vertical Edges
Diagonal Edges