Akademia Górniczo-Hutnicza w Krakowie

Wydział Fizyki i Informatyki Stosowanej

Analiza Obrazów

Sprawozdanie nr 1 - Podstawowe właściwości obrazu, filtracja, analiza w dziedzinie częstotliwościowej

Marcin Knapczyk

20.11.2024

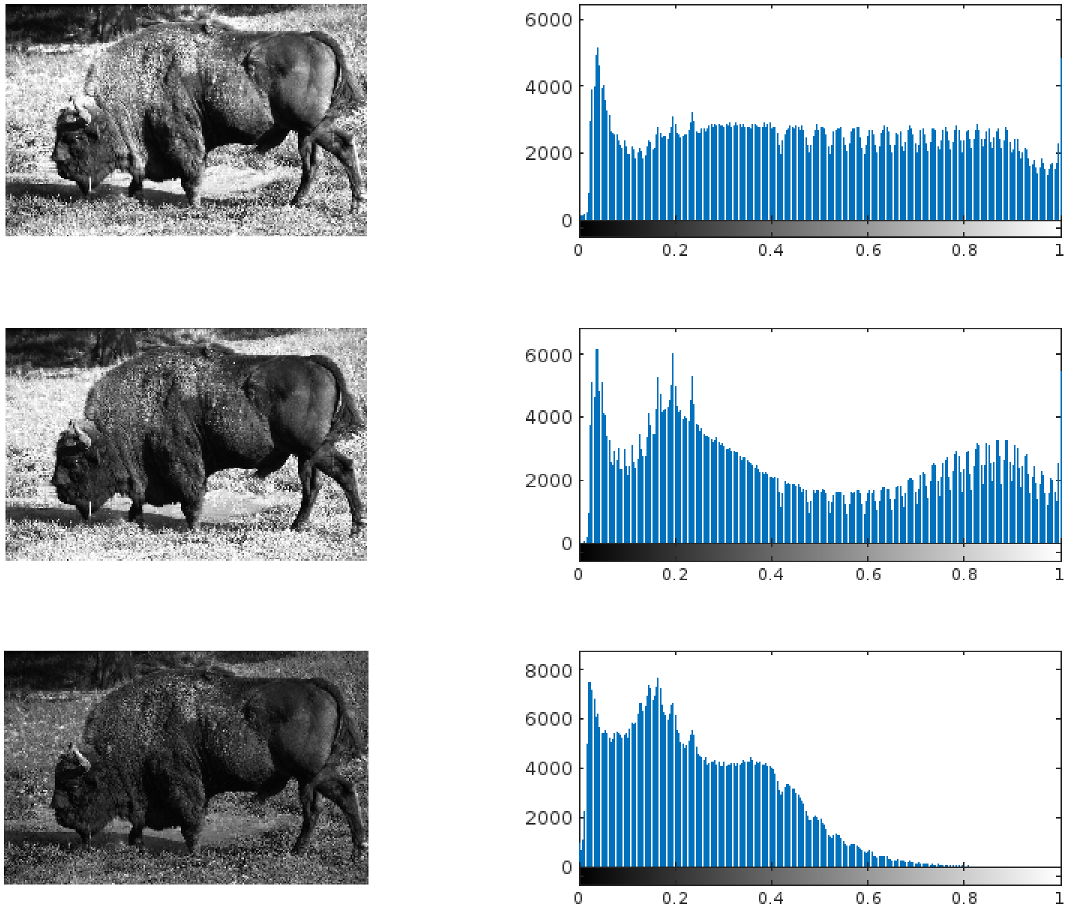
1 Podstawowe właściwości obrazu

1.1 Sposób zapisu i reprezentacja obrazu rastrowego

Obraz rastrowy składa się z siatki pikseli o różnych kolorach. Kolor piksela może być wyrażony na wiele różnych sposobów, takich jak model RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), YUV i inne.
Do załadowania obrazu do programu MATLAB służy wbudowana funkcja $imread()$, do której przekazujemy ścieżkę do pliku. W MATLABie obrazy przechowywane są w postaci trójwymiarowej macierzy, której dwa pierwsze wymiary (o rozmiarach $wys \times szer$) zawierają informacje o koordynatach pikseli, a trzeci wymiar przechowuje informacje o kanałach kolorów (domyślnie) w modelu RGB.

Rysunek 1: Załadowany obraz żubra

Możliwe jest wyodrębnienie kolorów obrazu poprzez selekcję wartości interesujących nas kanałów, np. dla koloru zielonego poprzez wybranie wartości z drugiego kanału: $g = im(:,:,2)$. Histogram w programie MATLAB można wygenerować za pomocą polecenia $imhist()$, do którego przekazujemy znormalizowaną macierz obrazu.



Rysunek 2: Wizualizacja wartości pikseli w poszczególnych kanałach (kolejno) RGB wraz z histogramem

Na powyższych obrazkach większym wartościom koloru dla danego piksela odpowiada jaśniejszy kolor. W ten sposób łatwo można dostrzec większą reprezentację koloru czerwonego na futrze żubra. Domyślnie wartości macierzy są całkowite, z zakresu [0, 255], ale w praktyce przekształca się je do wartości rzeczywistych z zakresu [0, 1]. Normalizacji tej można dokonać w następujący sposób: $im = double(im)/255$.

Do analizy obrazu najczęściej używa się jego reprezentacji w odcieniach skali szarości. Reprezentację taką można obliczyć ręcznie, np. licząc średnią z wartości RGB lub średnią ważoną standardu YUV (wagi RGB kolejno 0.299, 0.587, 0.144 pozwalają uzyskać reprezentację bardziej naturalną dla ludzkiego oka), albo korzystając z funkcji $rgb2gray()$, przekazując jako argument znormalizowaną macierz obrazu.

Rysunek 3: Porównanie sposobów przejścia do skali szarości: na lewo uzyskanego ze średniej, po prawej w modelu YUV

Obraz w standardzie YUV ma widocznie lepiej zachowany kontrast i szczegóły są na nim bardziej widoczne.

1.2 Przekształcenia kolorów obrazu

Na macierzy można dokonać różnych przekształceń, modyfikując wartości kanałów kolorów za pomocą różnych funkcji.
Jeżeli wartość funkcji przekroczy zakres [0, 1], należy ją sprowadzić do najbliższej wartości granicznej.

1.2.1 Jasność

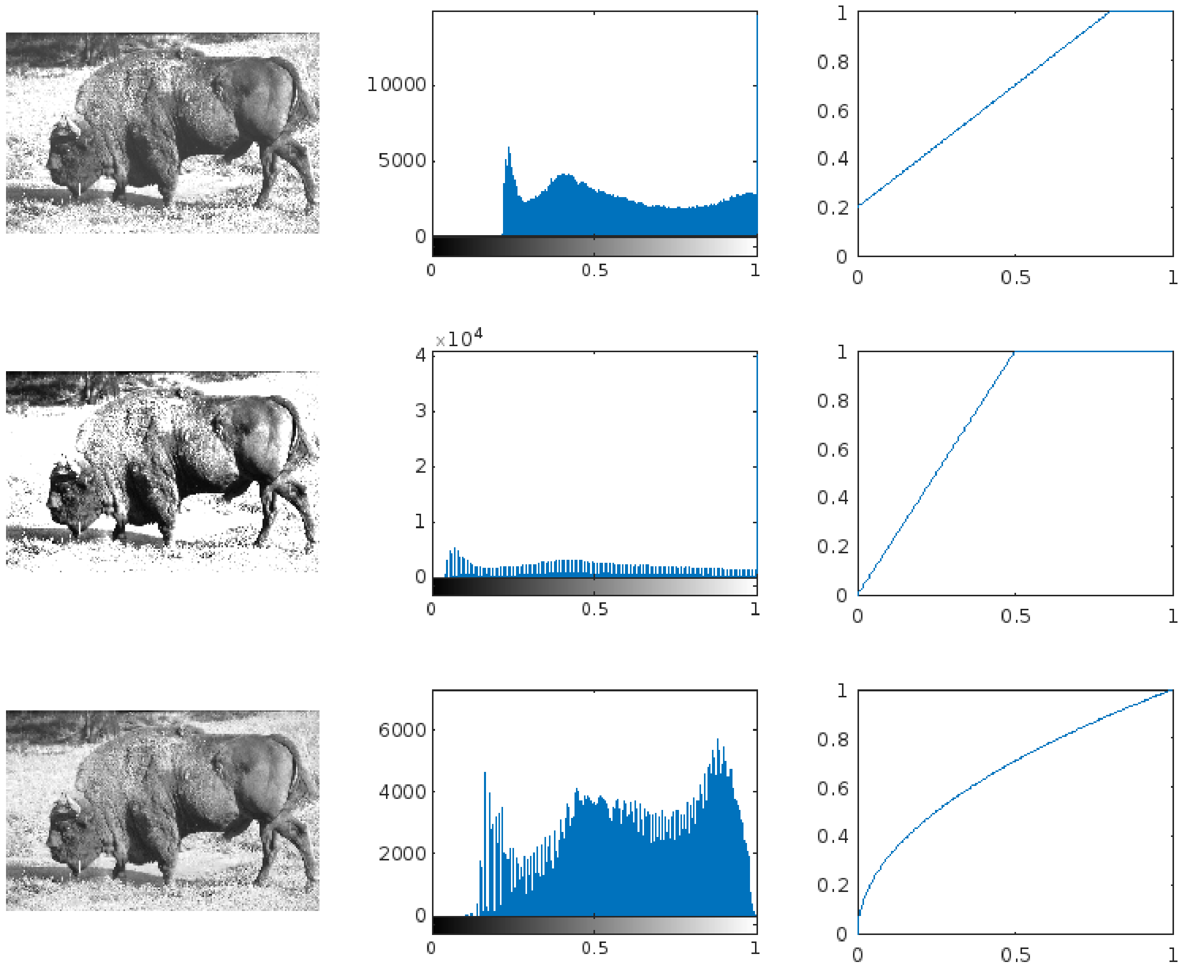
Zmiany jasności obrazu można dokonać poprzez dodanie stałej wartości do każdego piksela obrazu. Wówczas funkcja przyjmnie postać $y = x + b$.
Parametr b przyjmuje wartość (0-1] w przypadku rozjaśniania lub [-1, 0) w przypadku przyciemniania. Dla wartości $b = 0$ nie zajdzie żadna zmiana.

1.2.2 Konstrast

Kolejnym sposobem na przekształcenie obrazu jest pomnożenie każdej z wartości pikseli przez konkretną liczbę. Funkcja $y = c \cdot x$ służy do zmiany kontrastu. Parametr c należy do zakresu (0, +inf). Dla wartości (0, 1) funkcja zmniejszy kontrast, zaś dla c z zakresu (1, +inf) kontrast zostanie zwiększony.

1.2.3 Gamma

Inną przydatną funkcją jest $y = x^g$, która realizuje przekształcenie gamma. Wartości parametru g (z zakresu (0, +inf)) powyżej jeden rozciągają wartości niskie na szersze spektrum, ciemne spychają do spektrum węższego. Dla $g < 1$ efekt jest odwrotny.



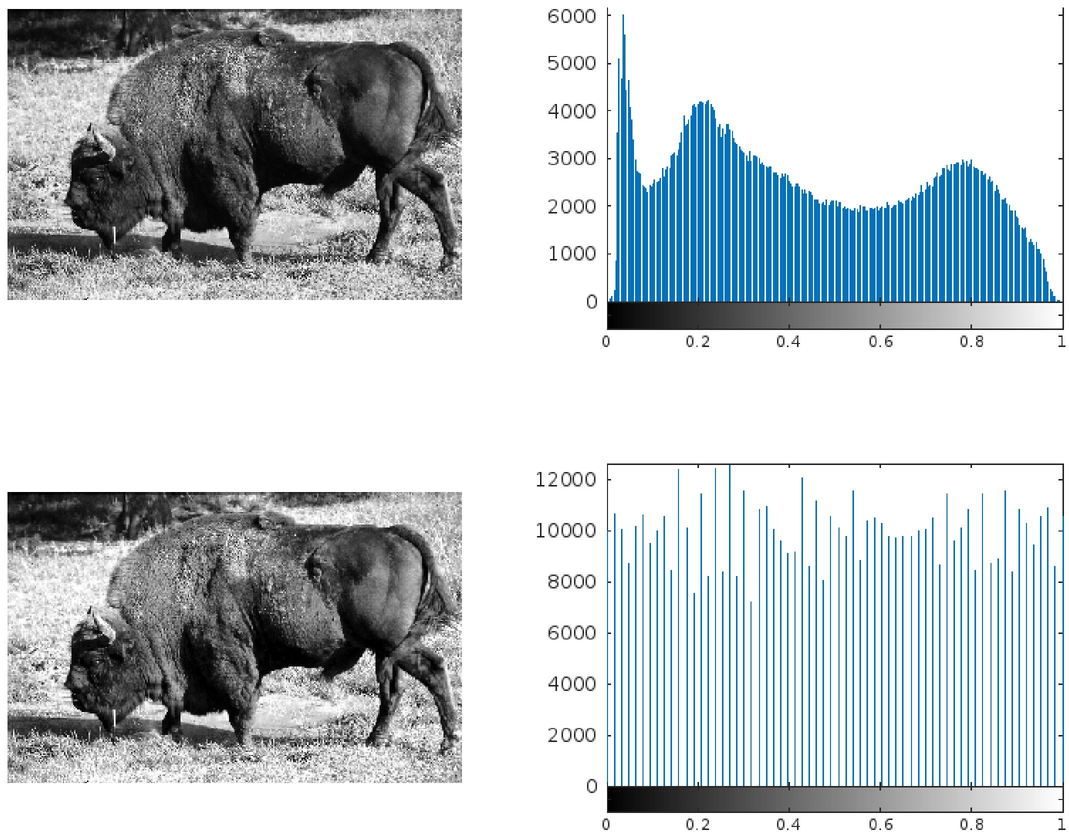
Rysunek 4: Porównanie efektów przekształceń kolejno dla jasności (b=0.2), kontrastu (c=2) i gamma (g=0.5). Wykresy funkcji

Jak widać, operacja zmiany jasności skutkuje przesunięciem histogramu o wektor [0, b], zmiana kontrastu rozciągnięcem histogramu w lewo lub w prawo, a gamma rozciągnięciem niskich lub wysokich wartości na szersze spektrum.

Przekształcenia te można łączyć, uzyskując funkcję $y = c \cdot x^g + b$.
Funkcja służąca do przekształceń może przyjąć dowolną postać, która zwraca wartości z zakresu [0, 1].

1.2.4 Wyrównanie histogramu

Innym przekształceniem jest wyrównanie histogramu, które, jak sama nazwa wskazuje, ma na celu sprawienie, by dla każdego przedziału wykresu histogramu wartości były możliwie podobne. W programie MATLAB służy do tego funkcja *histeq*().



Rysunek 5: Porównanie oryginalnego obrazka z efektem wyrównania histogramu

Po wyrównaniu histogramu widać, że poziomy jasności są lepiej rozłożone w całym zakresie.

2 Filtracja

Filtr składa się z macierzy kwadratowej o nieparzystych wymiarach. Filtrowanie obrazu polega na obliczeniu sumy wartości sąsiadujących pikseli przemnożonych przez odpowiednią wartość w macierzy filtru dla każdego piksela obrazu.
W programie MATLAB filtracji mozemy dokonać przy uzyciu funkcji $imfilter()$, jako argumenty podając macierz obrazu i macierz filtru.

2.1 Filtr dolnoprzepustowy

Jednym z prostszych filtrów jest filtr rozmywający (blur). Jego macierz składa się z wartości $1/n$, gdzie n to liczba wartości w macierzy (np. macierz filtru 3x3 wypełniona będzie wartościami 1/9) tak, by suma wartości w macierzy była równa 1.

Rysunek 6: Porównanie oryginalnego obrazka z obrazkiem przefiltrowanym (blur o macierzy 15x15)

Macierz filtru można konstruować także na inne sposoby. Dla uzyskania bardziej naturalego efektu rozmycia, wartości macierzy filtru można wyznaczyć na podstawie rozkładu Gaussa (gaussian blur).

Rysunek 7: Obrazek żubra rozmyty filtrem Gaussa

2.2 Filtr medianowy

Filtracja medianowa każdemu pikselowi przypisuje medianę pikseli z określonego otoczenia. Powoduje to rozmycie z jednoczesnym zachowaniem krawędzi. Jest on bardzo dobry do odszumiania. Filtracji tej mozna dokonać przy pomocy funkcji $medfilt2()$, jako argumenty podając macierz obrazu i wektor z wymiarami sąsiedztwa branego pod uwagę w trakcie filtrowania.

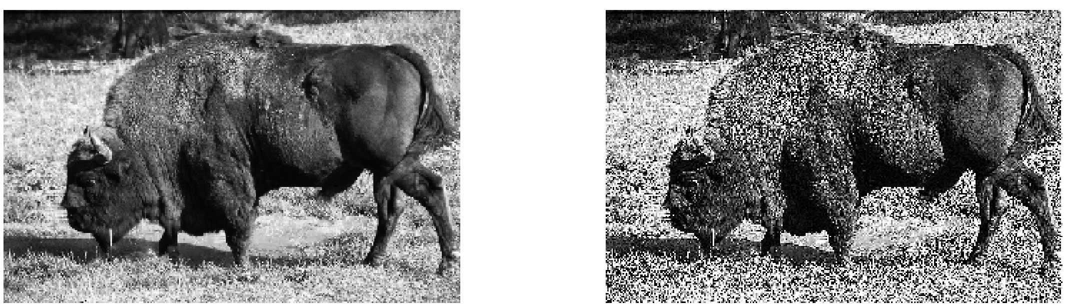
Rysunek 8: Obrazek żubra rozmyty filtrem medianowym 3x3

Rysunek 9: Obrazek żubra rozmyty filterm medianowym 21x21

Jak widać, nawet przy dużym rozmyciu krawędzie żubra zostały dobrze zachowane.

2.3 Filtr górnoprzepustowy

Gdy do konstrukcji filtru uzyjemy wartości ujemnych, uzyskamy efekt uwypuklenia różnic na obrazie (wyostrzanie). Suma wszystkich elementów w macierzy może być równa zero (filtr krawędziowy) lub jeden.



Rysunek 10: Porównanie oryginalnego obrazka z efektem filtru górnoprzepustowego 3x3

2.4 Binaryzacja

Binaryzacja polega na nadaniu każedmu pikselowi o wartości mniejszej od ustalonego progu wartości 0 lub 1, gdy jego wartość jest większa lub równa progowi.
Wartość progu może być wybrana dowolnie, jednak przy wyborze wartości warto posłużyć się histogramem, który pozwoli wyodrębnić części obrazu o różnych jasnościach.

Rysunek 11: Obrazek żubra po binaryzacji z progiem równym 0.6

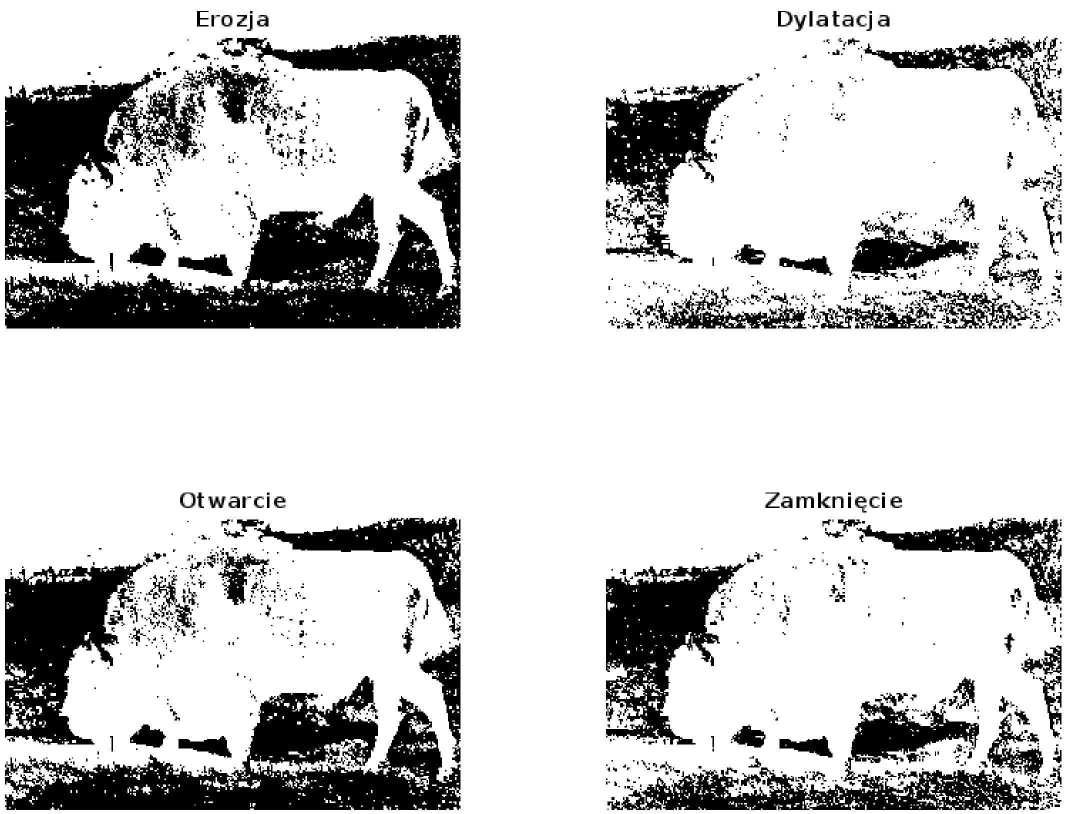
Próg mozna tez dobrać automatycznie za pomocą funkcji $greytresh()$, z argumentem w postaci macierzy obrazu.

Rysunek 12: Obrazek żubra po binaryzacji z progiem dobranym automatycznie

Jak widać, najlepszym sposobem doboru wartości progu zazwyczaj jest dobranie ręczne.

2.5 Erozja i dylatacja

Erozja i dylatacja należą do operacji morfologicznych (skupiających się na kształcie obiektu). Mogą one służyć do pozbycia się widocznych dziur lub kropek w niepożądanych miejscach.
Erozja (pomniejszenie) polega na polega na odejmowaniu, a dylatacja (powiększenie) na dodawaniu skrajnych pikseli kształtu.
Operacja dylatacji (erozji) działa tak, że przypisuje pikselowi wartość 1 (0) na podstawie wartości pikseli znajdujących się w jego sąsiedstwie (gdy mają wartości 1 (0)). Otwarcie morfologiczne polega na erozji, a następnie dylatacji obrazu. Zamkniecie morfologiczne polega zaś na dylatacji, a następnie erozji obrazu.
W programie MATLAB operacje te można wykonać za pomocą funkcji $imerode()$, $imdilate()$, $imopen()$ i $imclose()$.



Rysunek 13: Wynik różnych operacji morfologicznych wykonanych na zbinaryzowanym obrazie żubra (sąsiedstwo 3x3)

Można dokonać próby wycięcia kształtu poprzez przemnożenie obrazka przez macierz będącą wynikiem domknięcia jego zbinaryzowanej formy.

Rysunek 14: Wynik wycięcia żubra (domknięcie z sąsiedstwem 7x7)

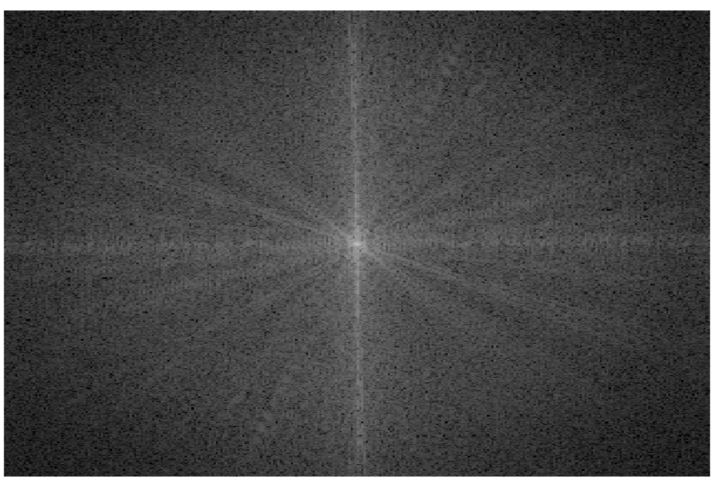
Wycięcie jest dalekie od ideału - pozostało bardzo dużo tła, pojawiły się też dziury w wycinanym kształcie.

3 Analiza obrazu w dziedzinie częstotliwościowej

3.1 Szybka transformacja Fouriera

Różnych operacji na obrazie można dokonywać także w dziedzinie częstotliwości. Pozyskanie takiej reprezentacji obrazu umożliwia dyskretna transformacja Fouriera. Do przeprowadzenia transformacji można użyć algorytmu szybkiej transformacji Fouriera (FFT - Fast Fourier Transform), w programie MATLAB to funkcja $fft2()$.
Uzyskane przez transformację wartości w dziedzinie zespolonej można przedstawić za pomocą amplitudy/modułu ($A \in [0, +\infty)$) lub fazy/kąta ($\phi \in [-\pi, \pi)$).
Amplitudę obrazu możemy uzyskać przez użycie funkcji $abs()$, a fazę za pomocą funkcji $angle()$, do których przekazujemy macierz uzyskaną przez transformację obrazu.

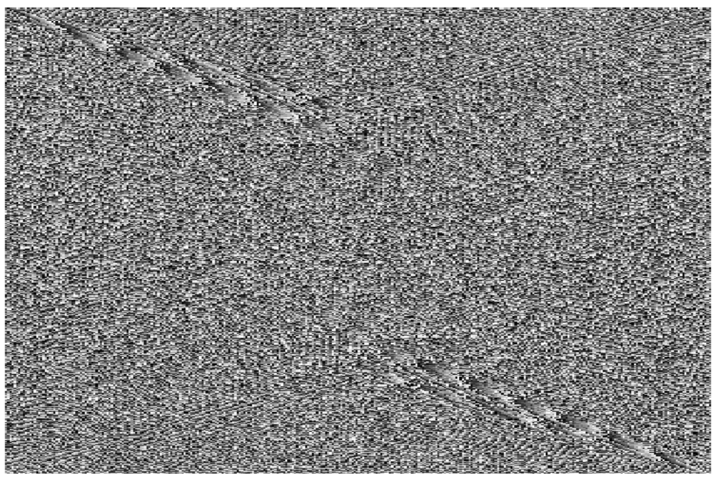
Rysunek 15: Obraz przedstawiający operę



Rysunek 16: Amplitudy przekształconego obrazu opery (ćwiartki przestawione za pomocą funkcji $fftshift()$ tak, by rozbłysk był na środku)

Zmiana nawet jednej wartości amplitudy powoduje powstanie zmian w obrazie oryginalnym (uzyskanym poprzez odwrotną transformację - funkcja $ifft2()$). Na obrazie pojawiają się fale (ich ilość zależy bezpośrednio od tego, którą z apmlitud zmienimy. Ustawienie dużej wartości $A(3, 4) = 10^5$ spowoduje, że na obrazie powstaną fale, które 2 razy przejdą przez krawędź boczną i 3 razy przejdą przez krawędź górną).

Rysunek 17: Obraz opery ze zmienioną amplitudą. Widoczne fale

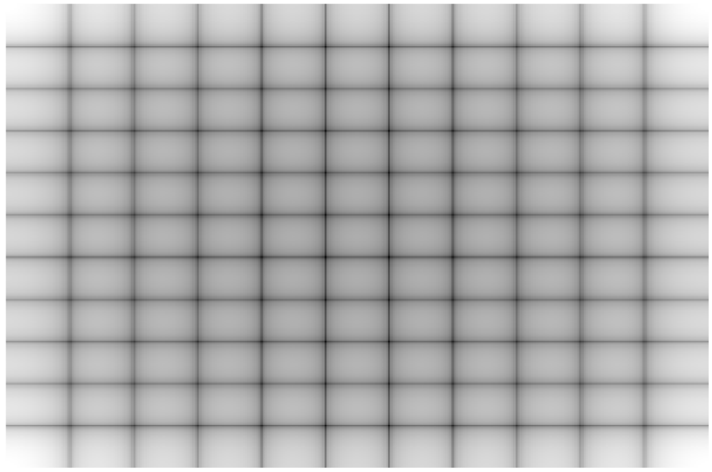


Rysunek 18: Fazy przekształconego obrazu opery

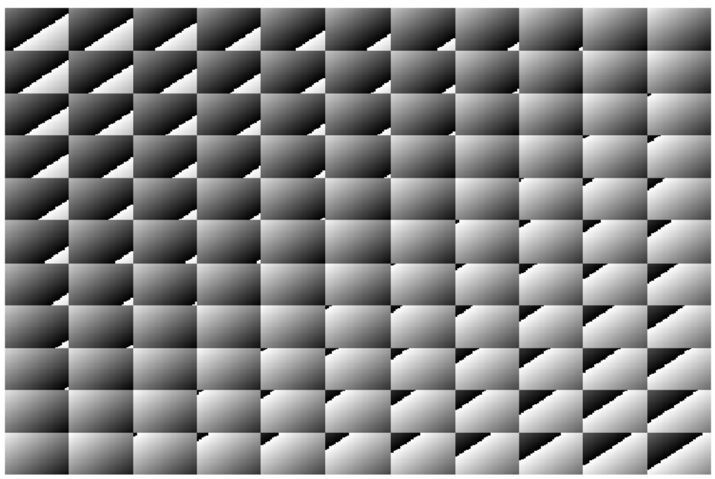
Faza nieregularnych obrazów (naturalne) przypomina biały szum.

3.2 Filtracja w dziedzinie częstotliwości

W dziedzinie częstotliwości możliwa jest takze filtracja. Wystarczy wówczas przemnożyć amplitudę obrazu przez amplitudę filtra. Pozwala to na zmniejszenie ilości potrzebnych obliczeń, jako iż operacja ta ma mniejszą złożoność obliczeniową.



Rysunek 19: Amplituda filtra dolnoprzepustowego (blur) 11x11. Wizualizacja przyjmuje postać 11x11 prostokątów



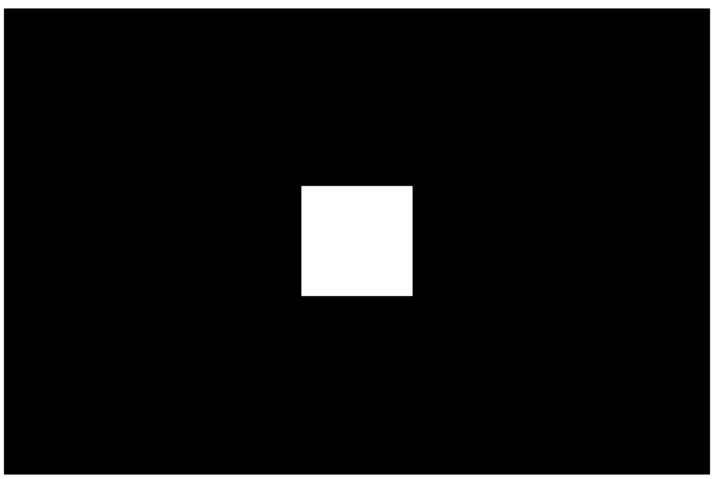
Rysunek 20: Faza filtra dolnoprzepustowego (blur) 11x11. Wizualizacja przyjmuje postać 11x11 prostokątów z cyklicznymi gradientami

Jak widać, amplituda i faza "sztucznych"obrazów (proste, regularne kształty) są regularne.

Rysunek 21: Przefiltrowany obraz opery

3.3 Kompresja obrazu

Operując na amplitudzie można dokonać kompresji obrazu. Jako iż jasne wartości amplitudy odpowiadają za najważniejsze informacje obrazu, obcinając wartości na obrzeżach amplitudy można małym kosztem zmniejszyć ilość danych reprezentujących obraz.



Rysunek 22: Maska służąca do "obcięcia"ramek amplitudy: 1 dla ważnych wartości, 0 dla mniej ważnych

Kompresji możemy dokonać poprzez przemnożenie amplitudy obrazu przez amplitudę maski.

Rysunek 23: Skompresowany obraz opery

Jak widać, nawet dla bardzo "przyciętej"amplitudy zostało zachowane dużo szczegółów oryginalnego obrazu.