

Zachodniopomorski Uniwersytet Technologiczny w Szczecinie
Wydział Elektryczny
Katedra Przetwarzania Sygnałów i Inżynierii Multimedialnej

Sprawozdanie z laboratorium

Przetwarzanie obrazów

Temat ćwiczenia: Filtracja medianowa ważona oraz filtracje częstotliwościowe DCT i FFT.

Data: 17.05.2021

Skład zespołu: Daniel Siutkowski

Sprawozdanie wykonał(a): Daniel Siutkowski

Nr albumu: 47035

Kierunek: Teleinformatyka

Semestr nr: 4

1. Opis skryptu:

1.1. Skrypt główny – ‘sd47035_lab4.m’

- Maski użyte do filtracji medianowej ważonej:

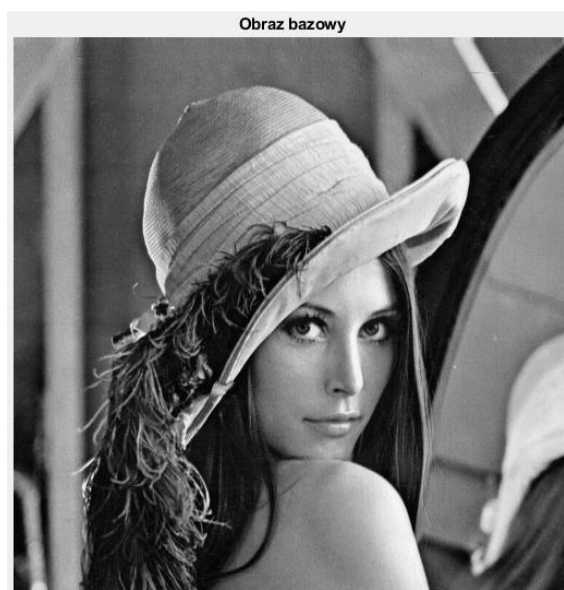
```
M1 = [0 1 0; 1 3 1; 0 1 0];  
M2 = [1 1 1; 1 3 1; 1 1 1];  
M3 = [3 0 1 0 3; 0 3 0 3 0; 1 0 9 0 1; 0 3 0 3 0; 3 0 1 0 3];  
listOfMasks = {M1,M2,M3};
```

Wszystkie maski zostały umieszczone w jednej zmiennej (cell array), która jest swego rodzaju listą.

- Wygląd kodu wywołującego funkcję odpowiedzialną za dokonanie filtracji medianowej ważonej:

```
if x == 1  
    imFMW = filtracjaMedianowaWazona(im,listOfMasks);  
  
    imshow(im);  
    title('Obraz bazowy');  
    pause(0.5);  
    figure;  
  
    imshow(imFMW);  
    title('Obraz wynikowy');  
end
```

Wygląd obrazu, na którym będą wykonywane operacje:



- Wygląd kodu wywołującego funkcję odpowiedzialną za dokonanie filtracji FFT lub DCT:

```

if x == 2
    fftORDct = 'Wybierz żadaną filtrację:\n1) FFT,\n2) DCT.\n> ';
    choice = input(fftORDct);

    if choice == 1
        % filtracja fft
        im = imread('PictureBW.bmp');

        fFFT = filtracjaFFT(im);
        imshow(fFFT);
        title('obraz wynikowy');
    end

    if choice == 2
        % filtracja dct
        im = imread('PictureBW.bmp');

        fdCT = filtracjaDCT(im);
        imshow(fdCT);
        title('obraz wynikowy');
    end
end
end

```

1.2. Pierwsza funkcja – ‘filtracjaMedianowaWazona.m’

- Szczytanie od użytkownika wyboru maski oraz przygotowanie zmiennych do późniejszych operacji:

```

ex1choice = 'Wybierz maskę dla filtracji medianowej:\n1)          2)          3)\n';
x = input(ex1choice);

[rows, cols] = size(im); %zczytanie wielkości obrazu do dwóch zmiennych
mask = length(M{x}); %długości maski, dla maski 3x3 przyjmie wartość 3
im = double(im); %zamiana typu wczytanego obrazu na double

```

- Mózg funkcji – podwójna pętla for:

```
[k,l] = deal(0,0);
for i = ceil(mask/2) : rows - floor(mask/2)
    k=k+1;
    for j = ceil(mask/2) : cols - floor(mask/2)
        l=l+1;
        tabWzOM = reshape(im(i-(floor(mask/2)) : i+(floor(mask/2)), j-(floor(mask/2)) : j+(floor(mask/2))),1,[]);
        reshaped_mask = reshape(M{x},1,[]); %zmiana kształtu macierzy maski (1 wiersz)
        mltpl_tabWzOM = repelem(tabWzOM,reshaped_mask); %mnozenie liczb z obszaru maski przez wartości z maski (p
        mltpl_tabWzOM = sort(mltpl_tabWzOM); %sortowanie
        result = median(mltpl_tabWzOM); %wybranie wartości środkowej
        imFMW(k,l) = result; %wpisanie wartości środkowej z posortowanego zestawu liczb
    end
    l=0;
end
```

- Normalizacja:

```
MIN = min(min(imFMW));
MAX = max(max(imFMW));
imFMW = uint8(255 .* ((imFMW-MIN) / (MAX-MIN)));
```

1.3. Druga funkcja – ‘filtracjaFFT.m’

- Deklaracja zmiennych:

```
function fFFT = filtracjaFFT(im)
    %im = imread('PictureBW.bmp');
    imF=fft2(im); %transformata fft
    imFshift = fftshift(imF);

    [rows, cols] = size(imF);
    usrInput = 'Wybierz filtr:\n1) DP,\n2) GP,\n3) PP.\n> ';
    x = input(usrInput);

    N = rows;
    R = ceil(N/10); %mozna zmienic wartosc dzielnika liczby N, np. z 10 na 20
    R2 = ceil(N/5);
    M = zeros(N);
    ii = ceil(N/2);
    M(ii,ii) = 1;
```

- Stworzenie macierzy dla filtrów dp, gp oraz pp:

```
dp = double(bwdist(M) <= R); %filtr dolnoprzepustowy
gp=1-dp; %filtr gornoprzepustowy
pp = double(xor(bwdist(M) <= R, bwdist(M) <= R2)); %filtr pasmowoprzepustowy
```

- Tworzenie macierzy wynikowych oraz wykonywanie operacji odwrotnych do fft2 i fftshift:

```
if x == 1
    Y=imFshift.*dp;
    imshow(dp);
    title('dp');
    figure;
end

if x == 2
    Y=imFshift.*gp;
    imshow(gp);
    title('gp');
    figure;
end

if x == 3
    Y=imFshift.*pp;
    imshow(pp);
    title('pp');
    figure;
end

YFshift=ifftshift(Y);
imFFT=abs(ifft2(YFshift));
```

- Normalizacja oraz wyświetlanie obrazów transformaty FFT (bez przesunięcia widma oraz z jego przesunięciem):

```
fFFT=255*mat2gray(imFFT); % Normalizacja
fFFT=uint8(abs(fFFT));
imshow(fFFT);

imshow(log(abs(imF)),[], colormap(jet), colorbar; %Wykres transformaty FFT
title('obraz transformaty FFT bez przesuniecia widma'); pause(0.5); figure;
imshow(log(abs(imFshift)),[], colormap(jet), colorbar;
title('obraz transformaty FFT z przesuniecieciem widma'); pause(0.5); figure;
```

1.4. Trzecia funkcja – *'filtracjaDCT.m'*

- Stworzenie transformaty DCT dla całego obrazu, odczytanie wielkości obrazu bazowego oraz czytanie wyboru użytkownika odnośnie filtru:

```
imD=dct2(im);  
[rows, cols] = size(imD);  
  
usrInput = 'Wybierz filtr:\n1) DP,\n2) GP,\n3) PP.\n> ';  
x = input(usrInput);
```

- Stworzenie macierzy masek dla filtrów dp, gp, pp:

```
[columnsInImage, rowsInImage] = meshgrid(1:rows, 1:cols);  
centerX = 1;  
centerY = 1;  
radius = 100;  
dp = double((rowsInImage - centerY).^2 + (columnsInImage - centerX).^2 <= radius.^2); %macierz masek dla filtru gornoprzepustowego dla filtracji DCT  
radius2 = 60;  
dpDR = double((rowsInImage - centerY).^2 + (columnsInImage - centerX).^2 <= radius2.^2);  
gp = 1 - dp; %macierz pelniaca funkcje filtru gornoprzepustowego dla filtracji DCT  
  
pp = xor(dp,dpDR); %macierz pelniaca funkcje filtru pasmowoprzepustowego dla filtracji DCT
```

- Tworzenie macierzy wynikowych, wykonanie operacji odwrotnej do dct2 oraz normalizacja:

```
if x == 1  
    Y=imD.*dp;  
end  
  
if x == 2  
    Y=imD.*gp;  
end  
  
if x == 3  
    Y=imD.*pp;  
end  
  
WY=abs(idct2(Y));  
  
%normalizacja  
fDCT = 255*mat2gray(WY);  
fDCT = uint8(abs(fDCT));
```

- Wyświetlanie obrazu transformaty DCT:

```
imshow(log(abs(imD)),[], colormap(jet), colorbar;  
title('obraz transformaty DCT');  
pause(0.5);  
figure;
```


2. Korzystanie ze skryptu:

By uruchomić skrypt, wystarczy odpalić plik główny ('sd47035_lab4.m').

Skrypt posiada intuicyjne menu wyświetlające się w Command Window, które w prosty sposób przeprowadzi użytkownika przez wszystkie opcje, aż do ujrzeniażądanego rezultatu.

W razie wyrażenia naglej chęci przerywania działania programu, wystarczy użyć skrótu klawiszowego Ctrl + C.

Przy pojedynczym uruchomieniu skryptu wyświetlany jest jedynie efekt działania wybranego filtru, więc by przejrzeć wyniki każdego z filtrów trzeba uruchomić program kilkakrotnie.

3. Wyniki realizacji poszczególnych zadań:

3.1. Procedura filtracji medianowej ważonej z wagami całkowitymi dla obrazów w skali szarości:

Maska:

```
0 1 0
1 3 1
0 1 0
```

Wynik filtracji:



Maska:

1 1 1
1 3 1
1 1 1

Wynik filtracji:



Maska:

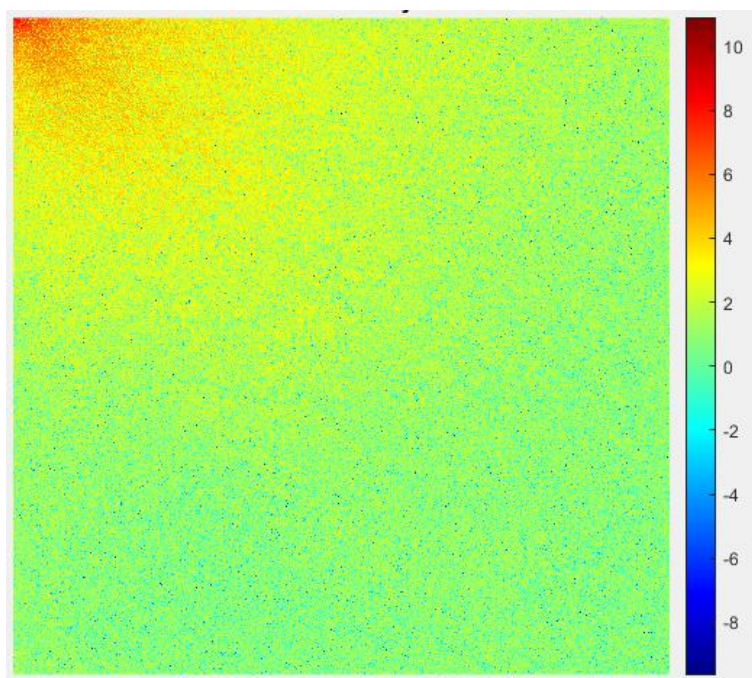
```
3 0 1 0 3
0 3 0 3 0
1 0 9 0 1
0 3 0 3 0
3 0 1 0 3
```

Wynik filtracji:



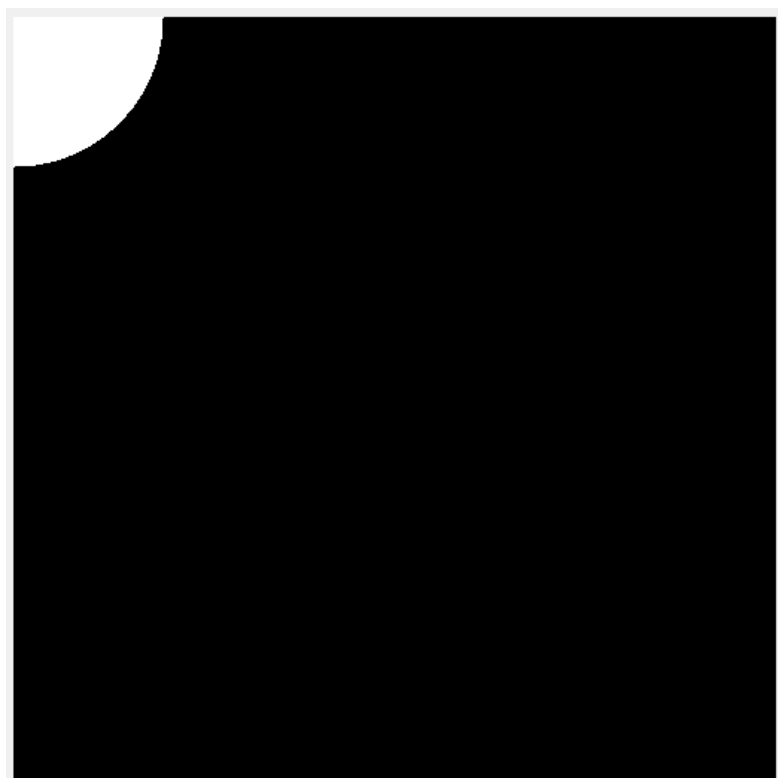
3.2. Procedura filtracji częstotliwościowej opartej na DCT dla obrazów w skali szarości:

Obraz transformaty DCT:



- Filtr dolnoprzepustowy:

Wizualizacja macierzy dla filtracji dolnoprzepustowej:

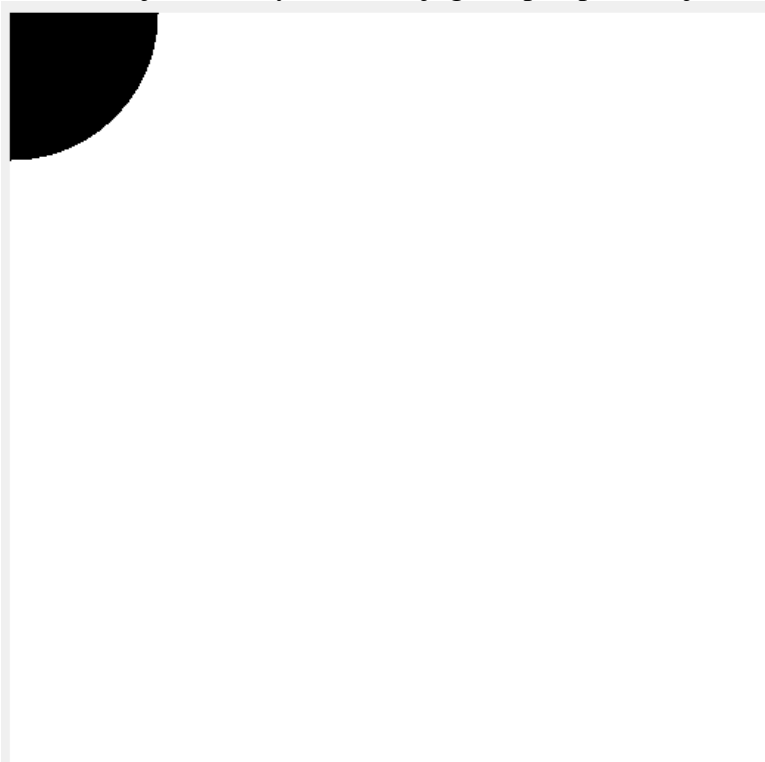


Obraz wynikowy:

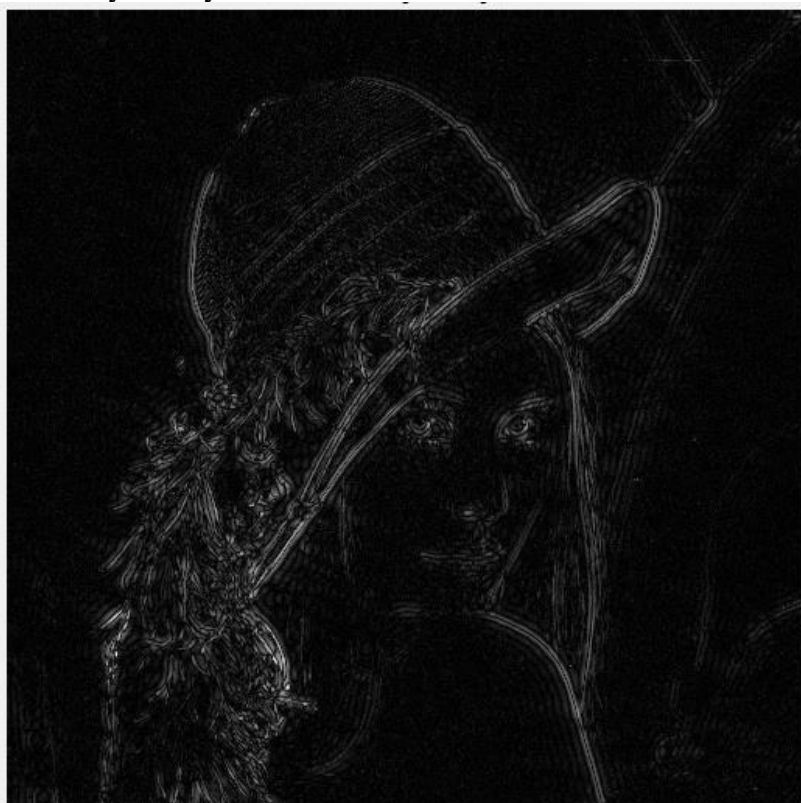


- Filtr górnoprzepustowy:

Wizualizacja macierzy dla filtracji górnoprzepustowej:



Obraz wynikowy:

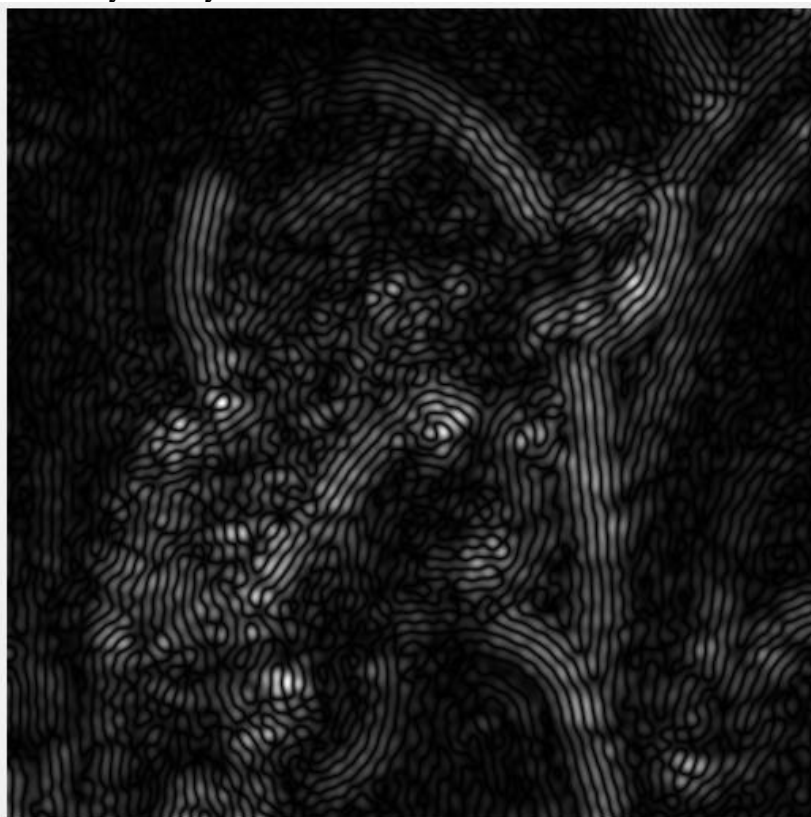


- Filtr pasmowoprzepustowy:

Wizualizacja macierzy dla filtracji pasmowo przepustowej:

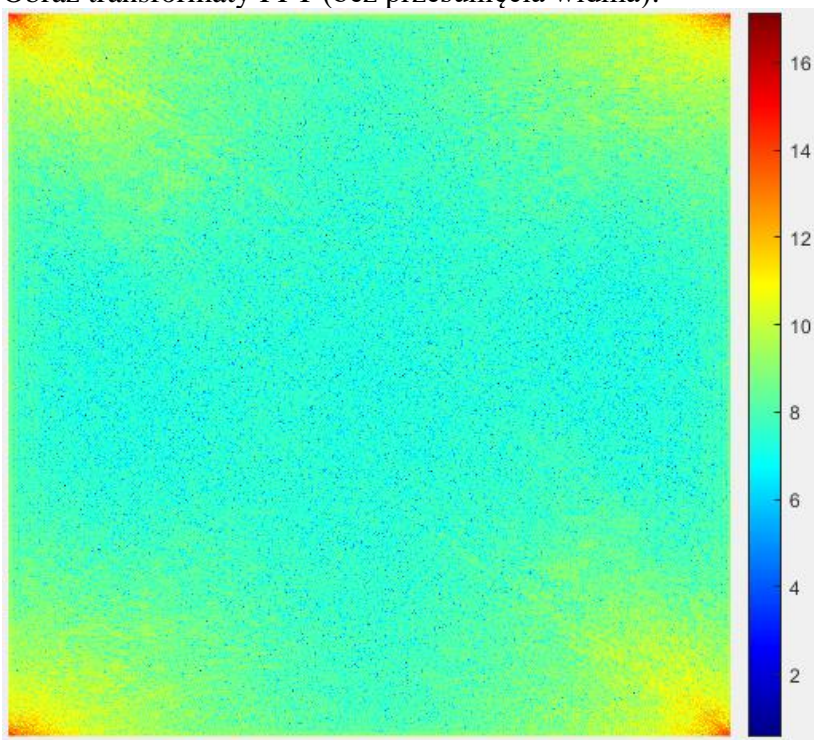


Obraz wynikowy:

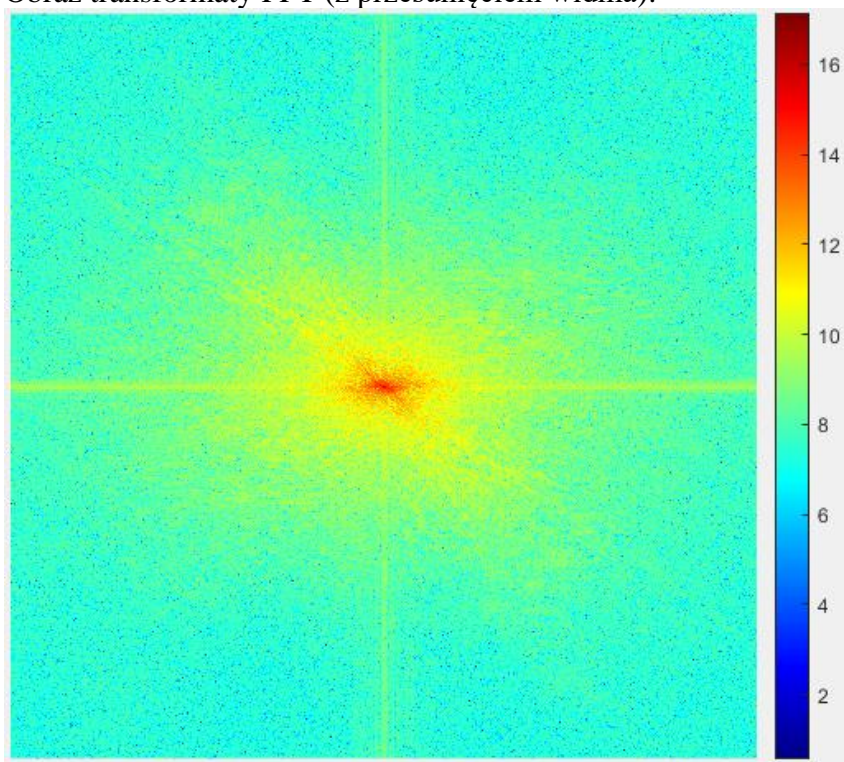


3.3. Procedura filtracji częstotliwościowej opartej na FFT dla obrazów w skali szarości:

Obraz transformaty FFT (bez przesunięcia widma):

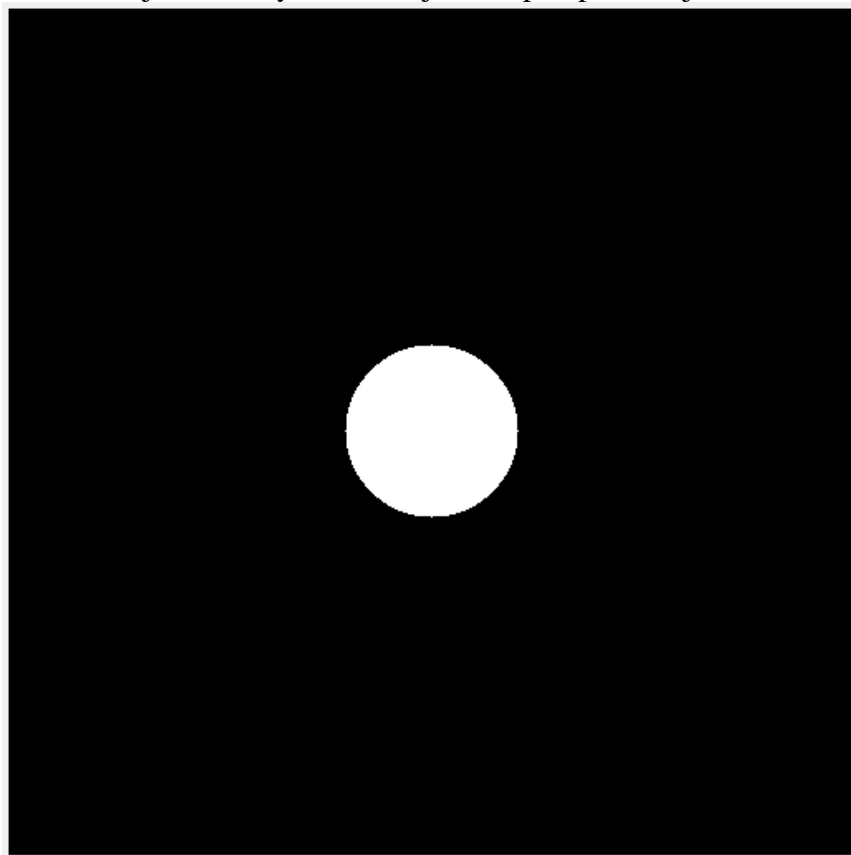


Obraz transformaty FFT (z przesunięciem widma):



- Filtr dolnoprzepustowy:

Wizualizacja macierzy dla filtracji dolnoprzepustowej:

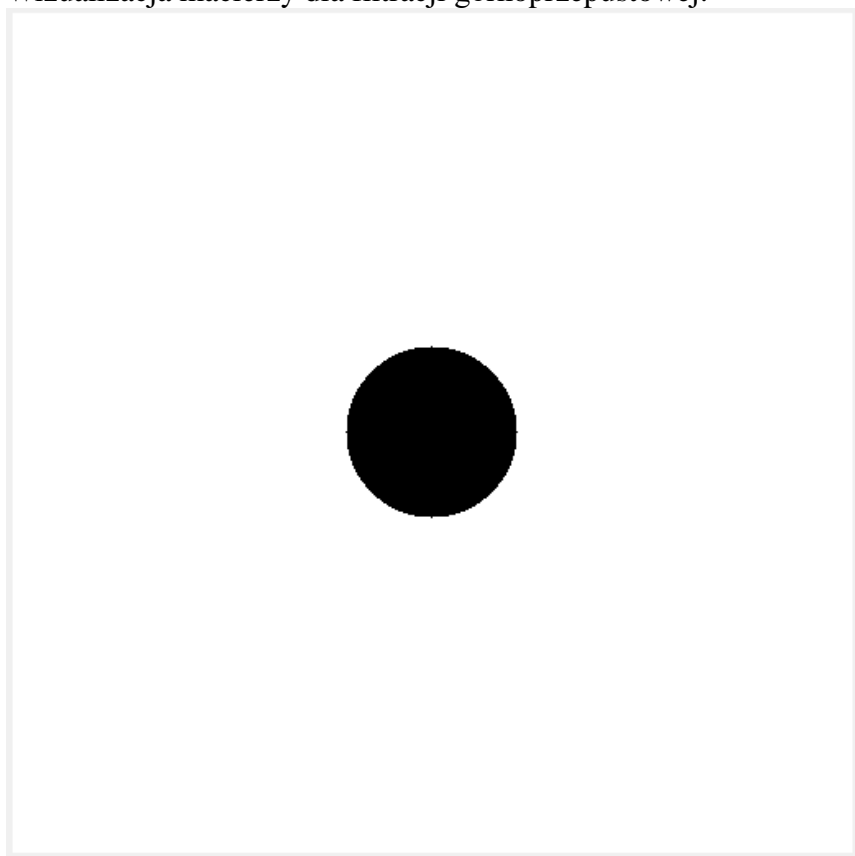


Obraz wynikowy:



- Filtracja górnoprzepustowa:

Wizualizacja macierzy dla filtracji górnoprzepustowej:

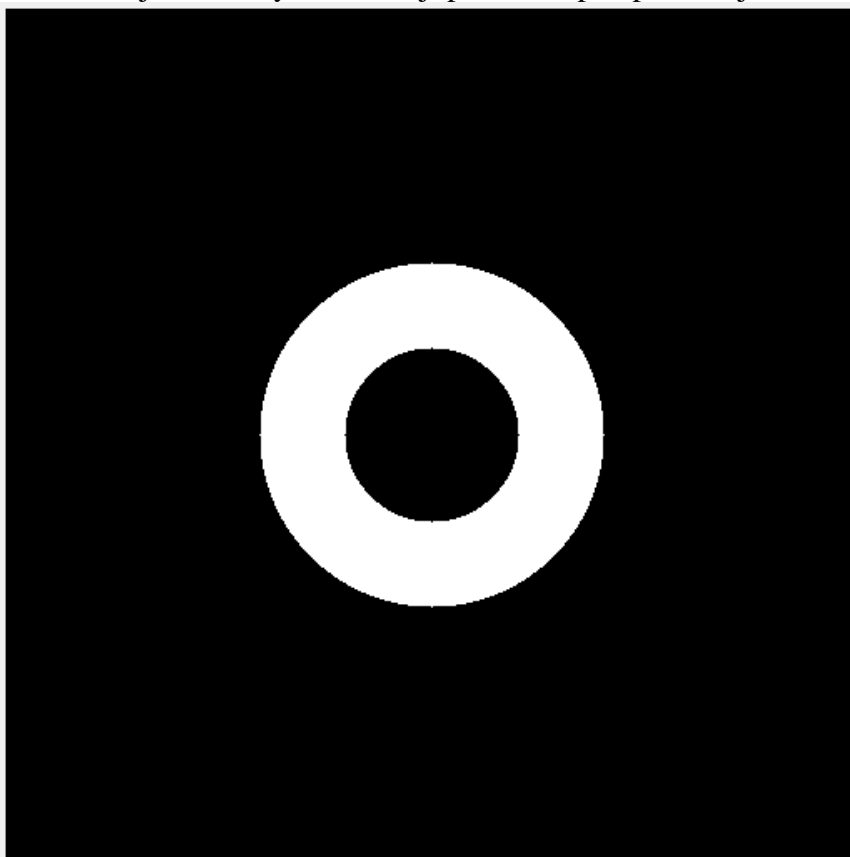


Obraz wynikowy:



- Filtracja pasmowoprzepustowa:

Wizualizacja macierzy dla filtracji pasmowo przepustowej:



Obraz wynikowy:



4. Wnioski:

Można dostrzec, iż obraz dobrej jakości, pozbawiony szumów po poddaniu procesowi filtracji medianowej ważonej nabiera specyficznego wyglądu. Obrazy wynikowe sprawiają wrażenie jakby dostały odrobinę kształtów, obłości czy też nawet wypukłości. Dodatkowo, jeżeli waga centralna w porównaniu do wag z pozostałych pól macierzy filtru ma wartość dwu lub trzykrotnie mniejszą od sumy z reszty pól, to obraz wynikowy nabiera pewnego rodzaju pastelowości, przynajmniej w moim odczuciu.

Po filtracji dolnoprzepustowej w dziedzinie częstotliwości możemy odnotować w obrazie wynikowym falowe zniekształcenia wokół krawędzi/zarysów postaci jak i wokół sąsiadujących bardziej wyrazistych elementów tła. Większe natężenie tego efektu możemy zaobserwować przy filtracji DCT, w filtracji FFT jest to odrobinę mniej intensywne zjawisko. Przy filtracji splotowej takiego zjawiska nie byliśmy w stanie odnotować, ogólna jakość rozmazanego obrazu była lepsza niż w przypadku filtracji w dziedzinie częstotliwości.

Filtry górnoprzepustowe w dziedzinie częstotliwości mają o wiele silniejsze działanie, niż w przypadku takich samych filtrów w dziedzinie przestrzennej. Przy filtracji splotowej, gdy korzystaliśmy z masek filtrów 3x3 lub 5x5 mogliśmy zauważyć, że dla pierwszej z masek otrzymany wynik prezentował obraz, który w głównej mierze posiadał bardziej jednolite, szare odcienie. Dla drugiej z ww. masek rezultatem był obraz w negatywie. Widać, że potrzebny byłby jeszcze większy rozmiar macierzy maski, by uzyskać podobny efekt jak w przypadku zwykłego filtru w dziedzinie częstotliwości, którego produktem jest czarny obraz z białymi konturami postaci.