

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

Sprawozdanie z laboratorium

Przetwarzanie obrazów

Temat ćwiczenia: Filtracja splotowa i medianowa obrazów	Data: 21.04.2021
Skład zespołu: Daniel Siutkowski	
Sprawozdanie wykonał(a): Daniel Siutkowski	Nr albumu: 47035
Kierunek: Teleinformatyka	Semestr nr: 4

1. Opis skryptu:

- Główny plik – *sd47035_lab3.m*:

Wczytanie obrazów:

```
clear all
im1 = imread('PictureBW.bmp');
im2 = imread('sp_img.bmp');
```

Utworzenie masek potrzebnych do przeprowadzenia filtracji:

```
Mdp3 = double(ones(3)); %maska dp uśredniająca 3x3
Mdp5 = double(ones(5)); %maska dp uśredniająca 5x5
Mdp7 = double(ones(7)); %maska dp uśredniająca 7x7
MGauss = double([1 2 1; 2 4 2; 1 2 1]); %maska dla filtru Gaussa
Mgp3 = double([-1 -1 -1; -1 9 -1; -1 -1 -1]); %maska gp 3x3
Mgp5 = double([0 -1 -1 -1 0; -1 2 -4 2 -1; -1 -4 13 -4 -1; -1 2 -4 2 -1; 0 -1 -1 -1 0]); % maska gp 5x5
MLaplace = double([0 -1 0; -1 4 -1; 0 -1 0]); %maska dla filtru Laplace'a
MPrewitt = double([-1 -1 -1; 0 0 0; 1 1 1]); %maska dla filtru Prewitta
MSobel = double([-1 -2 -1; 0 0 0; 1 2 1]); %maska dla filtru Sobela
MEMboss = double([-1 -1 -1; 0 1 0; 1 1 1]); %maska Emboss
Mdl1 = double([-1 -1 -1; 2 2 2; -1 -1 -1]); %detekcja linii, 1-szy przykład
Mdl2 = double([0 -1 0; 0 1 0; 0 0 0]); %detekcja linii, 2-gi przykład
```

Utworzenie list, w których będą przechowywane macierze masek:

```
userInput = {Mdp3, Mdp5, Mdp7, MGauss, Mgp3, Mgp5, MLaplace, MPrewitt, MSobel, MEMboss, Mdl1, Mdl2}; %lista przechowująca wszystkie maski potrzebne dla filtracji splotowej
userInputFM = {Mdp3, Mdp5, Mdp7}; %lista przechowująca wszystkie maski potrzebne dla filtracji medianowej
```

Wysłanie zapytania do konsoli, by użytkownik mógł wskazać, który sposób filtracji go interesuje:

```
wyberzFiltracje = 'Wyberz filtrację:\n1) Splotowa,\n2) Medianowa.\n> ';
wfiltr = input(wyberzFiltracje);
```

W zależności od dokonanego wyboru wykona się następująca część kodu:

Dla wybranej opcji związanej z filtracją splotową:

```
% ---- Filtracja Splotowa ----
if wfiltr == 1
    userChoice = '\nWyberz sposób traktowania krawędzi:\n1) Przepisanie wierszy i kolumn do obrazu wynikowego,\n2) Powiększenie obrazu oryginalnego.\n> ';
    choice = input(userChoice); %zmienna determinująca wybór sposobu traktowania krawędzi
    imFS = filtracjaSplotowa(im1,userInput,choice); %przypisanie rezultatu funkcji do zmiennej oraz wywołanie funkcji

    imshow(im1);
    title('Obraz bazowy');
    pause(0.5);
    figure;

    imshow(imFS);
    title('Obraz wynikowy');
end
```

Wewnątrz powyższej instrukcji warunkowej if użytkownik zostanie zapytany o to, w jaki sposób chce by krawędzie były traktowane, później następuje wywołanie funkcji, wewnątrz której użytkownik zostanie poproszony o wybór maski potrzebnej do dokonania filtracji. Później zostaną wyświetlone dwa obrazy: początkowy oraz wynikowy.

Po wybraniu opcji związanej z filtracją medianową:

```
% ---- Filtracja Medianowa ----  
if wfiltr == 2  
    imFM = filtracjaMedianowa(im2,userInputFM);  
  
    imshow(im2);  
    title('Obraz bazowy');  
    pause(0.5);  
    figure;  
  
    imshow(imFM);  
    title('Obraz wynikowy');  
end
```

Wewnątrz powyższej instrukcji warunkowej if po wywołaniu funkcji i interakcji użytkownika (wybór maski), zostaną wyświetlone dwa obrazy: początkowy oraz wynikowy.

• Funkcja – *filtracjaSplotowa.m*:

W zależności od wyboru dokonanego wcześniej przez użytkownika (sposób traktowania krawędzi), wykona się inny fragment kodu.

W przypadku wyboru pierwszej opcji (zamieszczam kod ze szczegółowymi komentarzami, przez które niestety cały kod może być mało czytelny w tym pliku, aczkolwiek w razie problemów z odczytaniem chociażby struktury kodu, odsyłam do pliku filtracjaSplotowa.m):

```
% -- przepisywanie wierszy i kolumn do obrazu wynikowego --  
if choice == 1  
    prompt1 = '\nPodaj żadaną maskę:\n1) DP 3x3,\n2) DP 5x5,\n3) DP 7x7,\n4) Gauss 3x3,\n5) GP 3x3,\n6) GP 5x5,\n7) Lapace,\n8) Prewitt,\n9) Sobel,\n10) Emboss,\n11) Detekcja linii: 1-szy filtr,\n12) Detekcja linii: 2-gi filtr.\n> ';  
    x = input(prompt1); %zczytanie wyboru użytkownika  
    [rows, cols] = size(im); %zczytanie wielkości obrazu do dwóch zmiennych  
    imFS = zeros(rows,cols); %stworzenie macierzy dla obrazu wynikowego  
    wypełnionej zerami  
    mask = length(userInput{x}); %długości maski, dla maski 3x3 przyjmie  
    wartość 3  
    im = double(im); %zamiana typu wczytanego obrazu na double  
  
    k = floor(length(userInput{x})/2); %wiersze obrazu wynikowego  
    l = floor(length(userInput{x})/2); %kolumny obrazu wynikowego  
    for i = ceil(mask/2) : rows - floor(mask/2) %podwójna pętla for wewnątrz  
    której następuje wpisanie zsumowanych wartości z obszaru maski do obrazu  
    wynikowego  
        k=k+1; %inkrementacja wierszy obrazu wynikowego  
        for j = ceil(mask/2) : cols - floor(mask/2)  
            l=l+1; %inkrementacja kolumn obrazu wynikowego  
            imFS(k,l) = sum(sum((im(i-(floor(mask/2)) : i+(floor(mask/2))),  
j-(floor(mask/2)) : j+(floor(mask/2)))).*userInput{x}));  
        end  
        l=floor(length(userInput{x})/2); %ustawienie wartości zmiennej do  
    początkowej w celu późniejszego przepisywania wartości imFS(k,l) w odpowiedniej  
    kolumnie  
    end
```

```

        %normalizacja
        MIN = min(min(imFS)); %uzyskanie wartości minimalnej z obrazu wynikowego
imFS
        MAX = max(max(imFS)); %uzyskanie wartości maksymalnej z obrazu
wynikowego imFS
        imFS = uint8(255 .* ((imFS-MIN) / (MAX-MIN))); %proces normalizacji +
zamiana typu na uint8

        for i = 1 : floor(mask/2) %pętla for odpowiedzialna za przepisanie
pierwszych wierszy
            for j = 1 : cols
                imFS(i,j) = im(i,j);
            end
        end

        for i = rows - floor(mask/2) : rows %pętla for odpowiedzialna za
przepisanie ostatnich wierszy
            for j = 1 : cols
                imFS(i,j) = im(i,j);
            end
        end

        for i = 1 : rows
            for j = 1 : floor(mask/2) %pętla for odpowiedzialna za przepisanie
pierwszych kolumn
                imFS(i,j) = im(i,j);
            end
        end

        for i = 1 : rows
            for j = cols - floor(mask/2) : cols %pętla for odpowiedzialna za
przepisanie ostatnich kolumn
                imFS(i,j) = im(i,j);
            end
        end
    end
end

```

W przypadku wyboru drugiej opcji:

```
% -- powiększenie obrazu początkowego --
if choice == 2
    prompt2 = '\nPodaj żadaną maskę:\n1) DP 3x3,\n2) DP 5x5,\n3) DP 7x7,\n4)
Gauss 3x3,\n5) GP 3x3,\n6) GP 5x5,\n7) Lapace,\n8) Prewitt,\n9) Sobel,\n10)
Emboss,\n11) Detekcja linii: 1-szy filtr,\n12) Detekcja linii: 2-gi filtr.\n> ';
    x = input(prompt2); %zczytanie wyboru użytkownika
    for i = 1 : floor(length(userInput{x})/2) %pętla for odpowiedzialna za
powielanie odpowiednią ilość razy wierszy i kolumn obrazu początkowego
        im(2:end+1,:) = im(:,,:); %zdublowanie pierwszego wiersza
        im(end+1,:) = im(end,:); %zdublowanie ostatniego wiersza
        im(:,2:end+1) = im(:,,:); %zdublowanie pierwszej kolumny
        im(:,end+1) = im(:,end); %zdublowanie ostatniej kolumny
    end

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

    [k,l] = deal(0,0); %ustawienie wartości wierszy i kolumn
    for i = ceil(mask/2) : rows - floor(mask/2) %podwójna pętla for wewnątrz
której następuje wpisanie zsumowanych wartości z obszaru maski do obrazu
wynikowego
        k=k+1; %inkrementacja wierszy obrazu wynikowego
        for j = ceil(mask/2) : cols - floor(mask/2)
            l=l+1; %inkrementacja kolumn obrazu wynikowego
            imFS(k,l) = sum(sum((im(i-(floor(mask/2)) : i+(floor(mask/2))),
j-(floor(mask/2)) : j+(floor(mask/2)))).*userInput{x}));
        end
        l=0;
    end

    MIN = min(min(imFS)); %uzyskanie wartości minimalnej z obrazu wynikowego
imFS
    MAX = max(max(imFS)); %uzyskanie wartości maksymalnej z obrazu
wynikowego imFS
    imFS = uint8(255 .* ((imFS-MIN) / (MAX-MIN))); %proces normalizacji +
zamiana typu na uint8
end
```

- **Funkcja – *filtracjaMedianowa.m*:**

Kod z ciała funkcji:

```
prompt1 = '\nPodaj żadaną maskę:\n1) 3x3,\n2) 5x5,\n3) 7x7\n> ';
x = input(prompt1); %zczytanie wyboru użytkownika
[rows, cols] = size(im); %zczytanie wielkości obrazu do dwóch zmiennych
mask = length(userInputFM{x}); %długości maski, dla maski 3x3 przyjmie
wartość 3
im = double(im); %zamiana typu wczytanego obrazu na double

[k,1] = 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,[]); %tabela wartości z obszaru maski
        tabWzOM = sort(tabWzOM); %sortowanie
        result = median(tabWzOM); %wybranie wartości środkowej
        imFM(k,l) = result; %wpisanie wartości środkowej z posortowanego
zestawu liczb
    end
    l=0;
end

%normalizacja
MIN = min(min(imFM));
MAX = max(max(imFM));
imFM = uint8(255 .* ((imFM-MIN) / (MAX-MIN)));

%przepisywanie wierszy i kolumn z obrazu bazowego
for i = 1 : floor(mask/2) %pętla for odpowiedzialna za przepisanie
pierwszych wierszy
    for j = 1 : cols
        imFM(i,j) = im(i,j);
    end
end

for i = rows - floor(mask/2) : rows %pętla for odpowiedzialna za
przepisanie ostatnich wierszy
    for j = 1 : cols
        imFM(i,j) = im(i,j);
    end
end

for i = 1 : rows
    for j = 1 : floor(mask/2) %pętla for odpowiedzialna za przepisanie
pierwszych kolumn
        imFM(i,j) = im(i,j);
    end
end

for i = 1 : rows
    for j = cols - floor(mask/2) : cols %przepisanie ostatnich kolumn
        imFM(i,j) = im(i,j);
    end
end
```

2. Korzystanie ze skryptu:

W celu uruchomienia skryptu, należy standardowo wcisnąć przycisk 'Run'.

W tym momencie w Command Window pojawia się pierwsze zapytanie do użytkownika:

```
Command Window
>> sd47035_lab3
Wybierz filtrację:
1) Splotowa,
2) Medianowa.
fx >
```

W przypadku wyboru filtracji Splotowej (1), ujrzymy kolejne zapytanie:

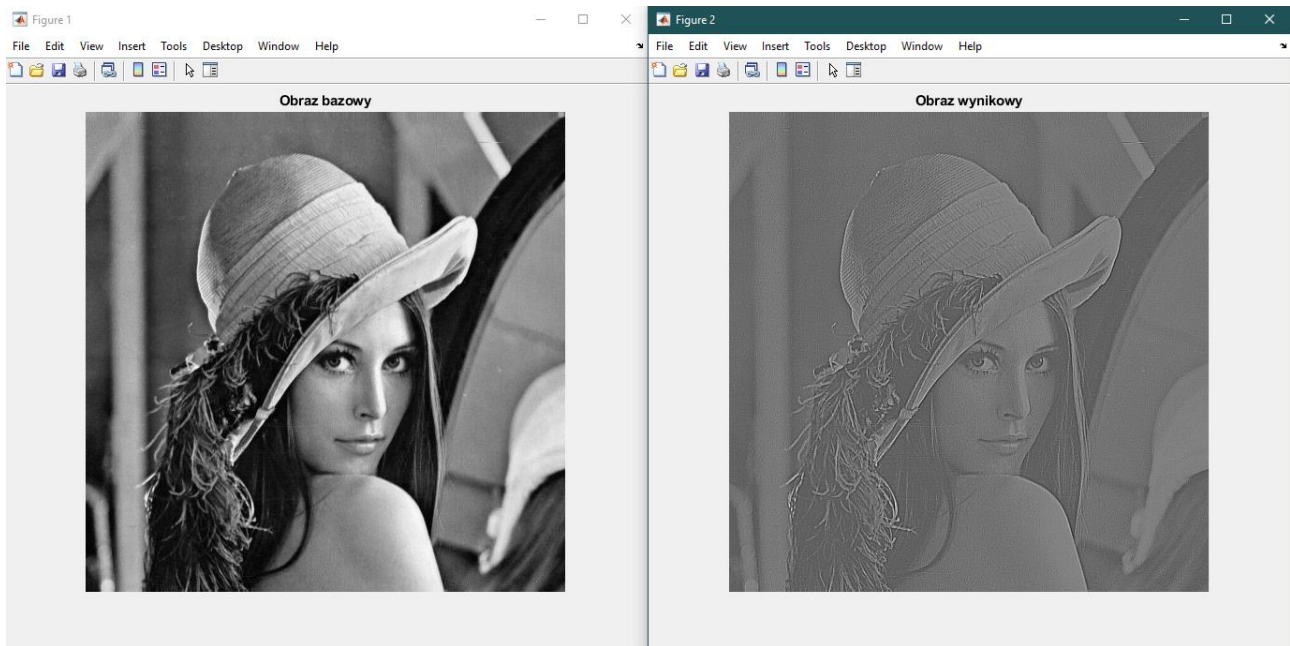
```
Wybierz sposób traktowania krawędzi:
1) Przepisanie wierszy i kolumn do obrazu wynikowego,
2) Powiększenie obrazu oryginalnego.
fx >
```

Na tym etapie, niezależnie od tego, czy wybierzemy opcję pierwszą, czy drugą, w następnym kroku będziemy poproszeni o wybór żądanej przez nas maski.

```
Podaj żadaną maskę:
1) DP 3x3,
2) DP 5x5,
3) DP 7x7,
4) Gauss 3x3,
5) GP 3x3,
6) GP 5x5,
7) Lapace,
8) Prewitt,
9) Sobel,
10) Emboss,
11) Detekcja linii: 1-szy filtr,
12) Detekcja linii: 2-gi filtr.
fx >
```

Po wybraniu maski, skrypt wygeneruje nam dwa podglądy obrazów: bazowego oraz wynikowego.

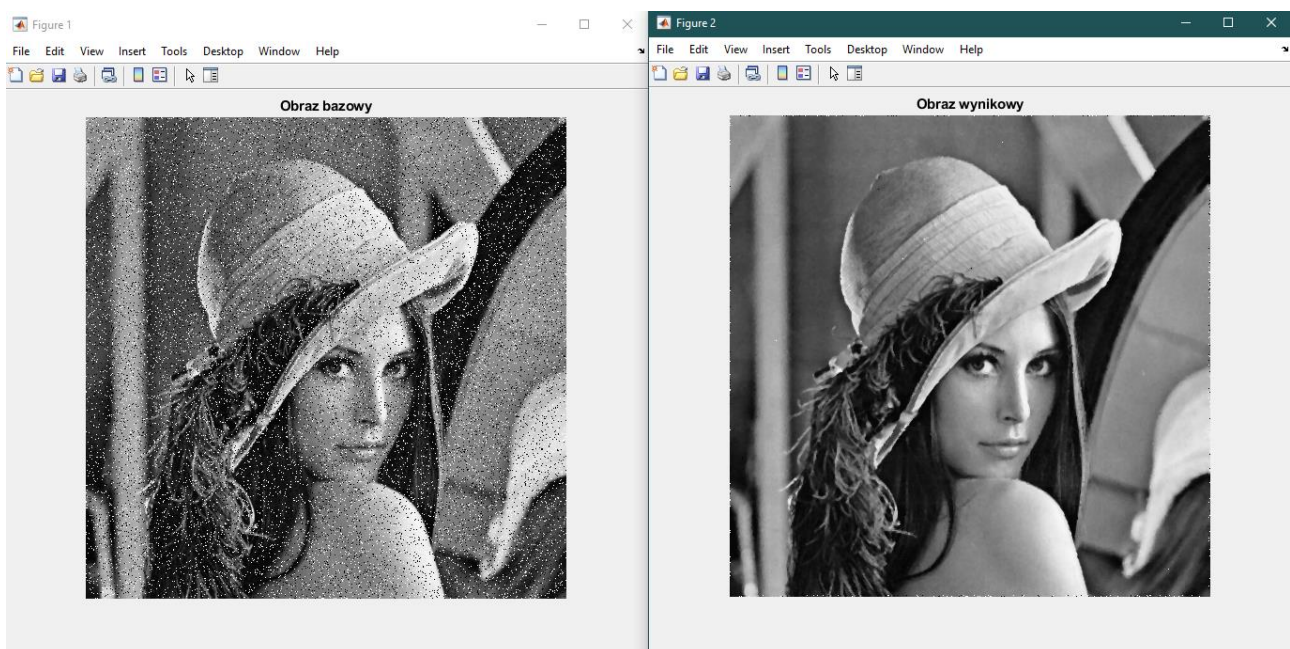
Przykładowe podglądy wygenerowane dla maski górnoprzepustowej 3x3 (opcja numer 5):



Jeżeli przy pierwszym zapytaniu wybrano jednak drugą opcję, to dla filtracji medianowej uzyskamy następujące opcje wyboru:

```
Podaj żadaną maskę:  
1) 3x3,  
2) 5x5,  
3) 7x7  
fx >
```

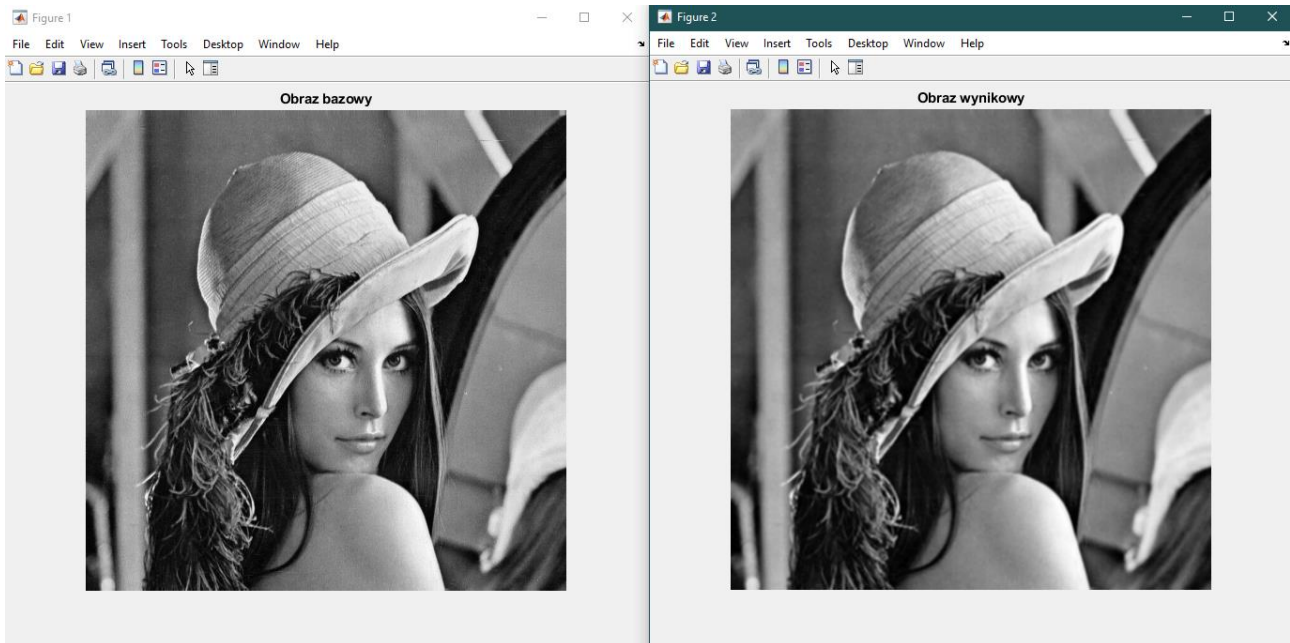
Podgląd wygenerowany dla pierwszej opcji (filtr 3x3):



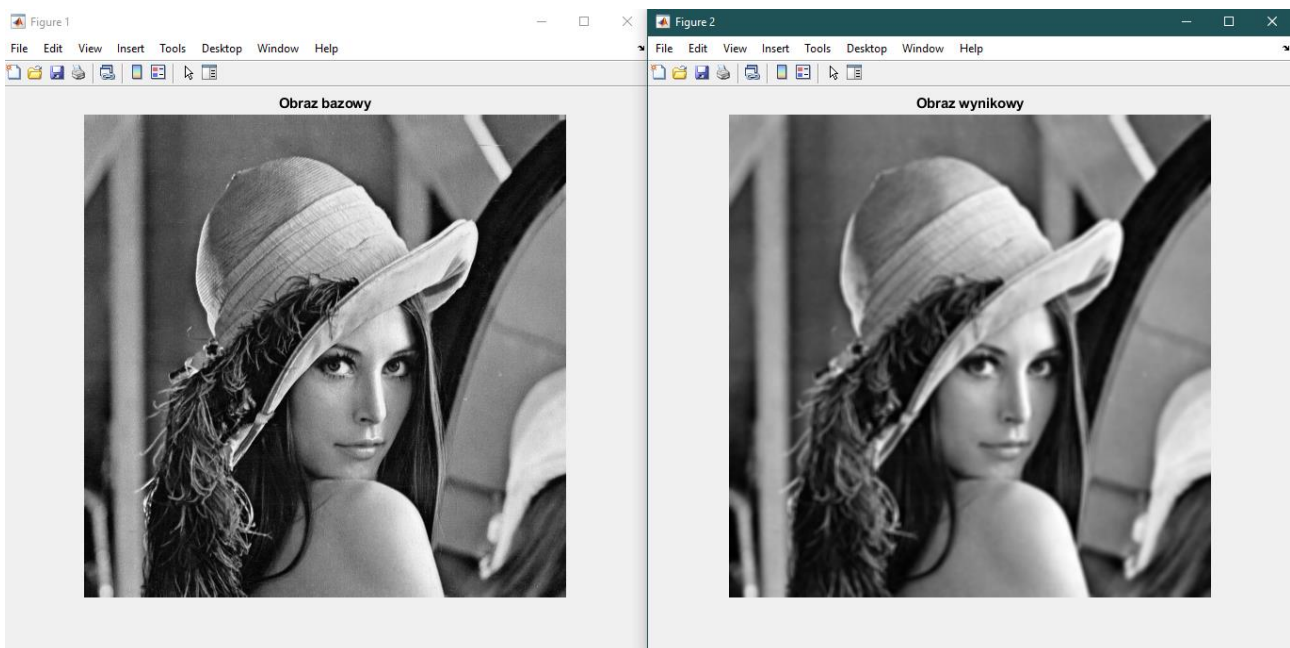
3. Wyniki filtracji dla poszczególnych masek:

Filtracja splotowa, powiększone obrazu bazowego:

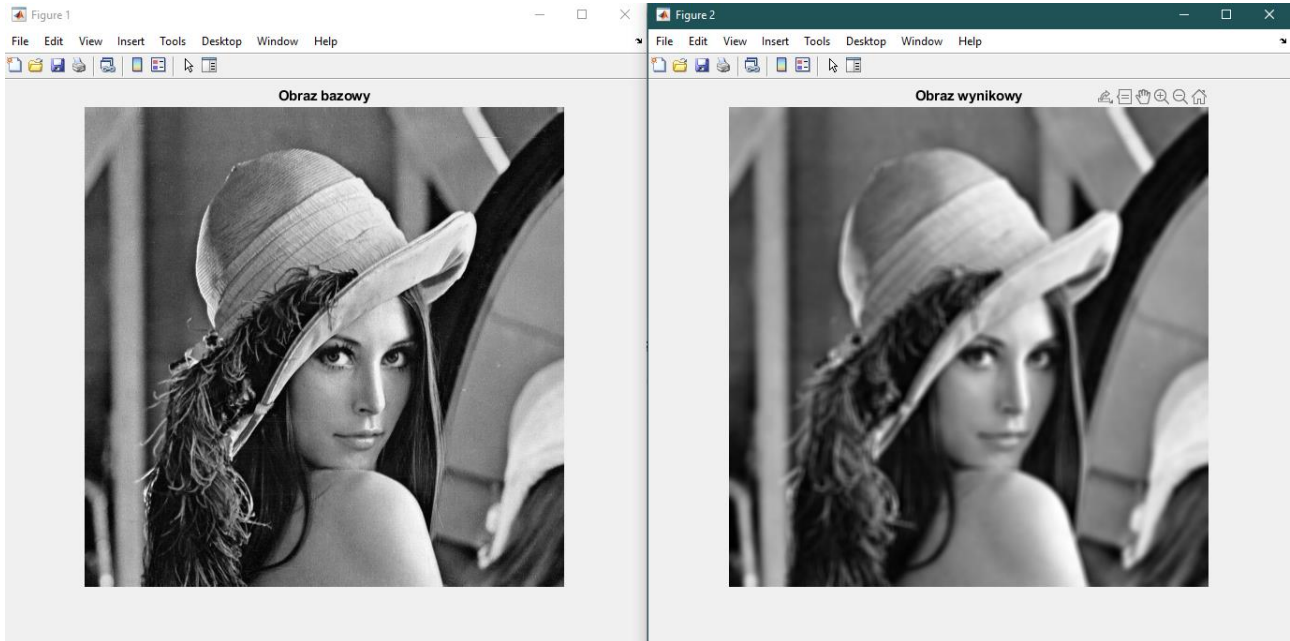
- Dolnoprzepustowa uśredniająca 3x3:



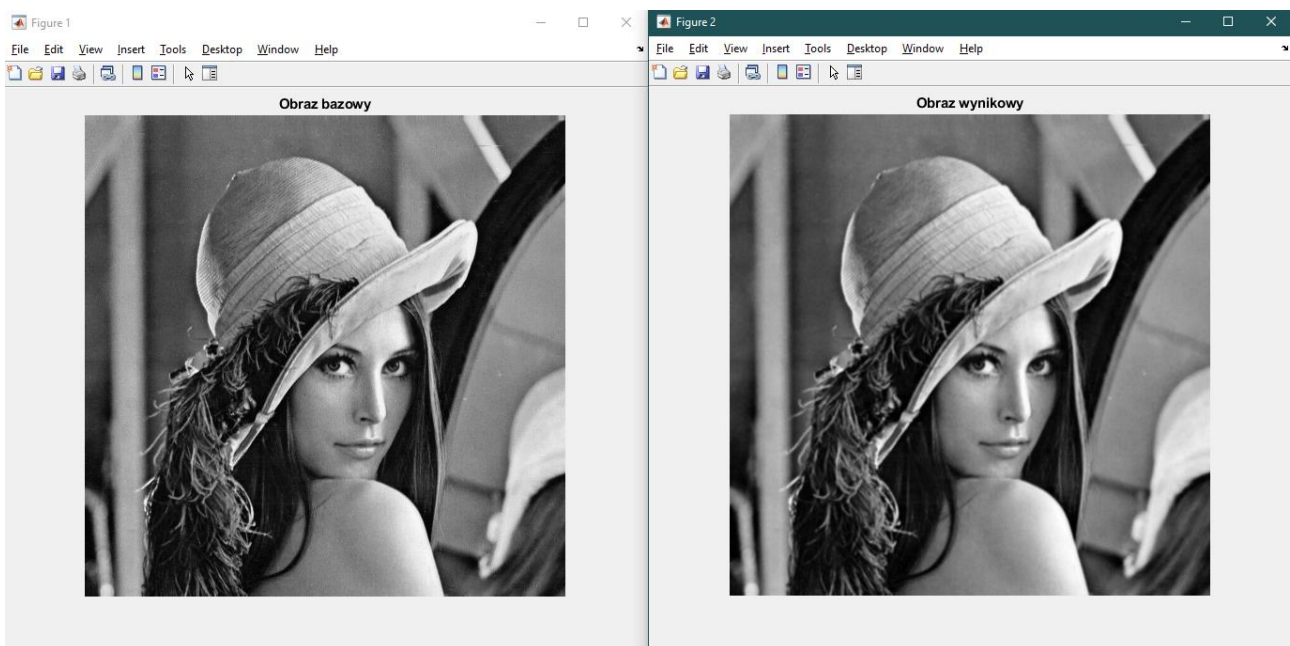
- Dolnoprzepustowa uśredniająca 5x5:



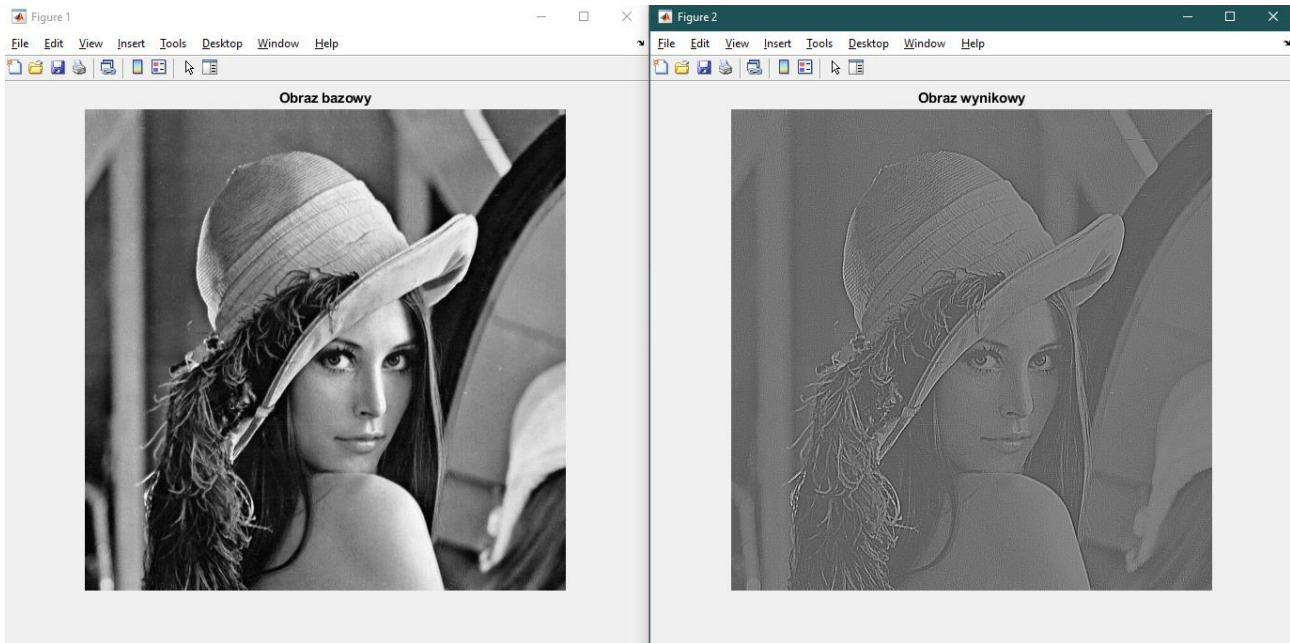
- Dolnoprzepustowa uśredniająca 7x7:



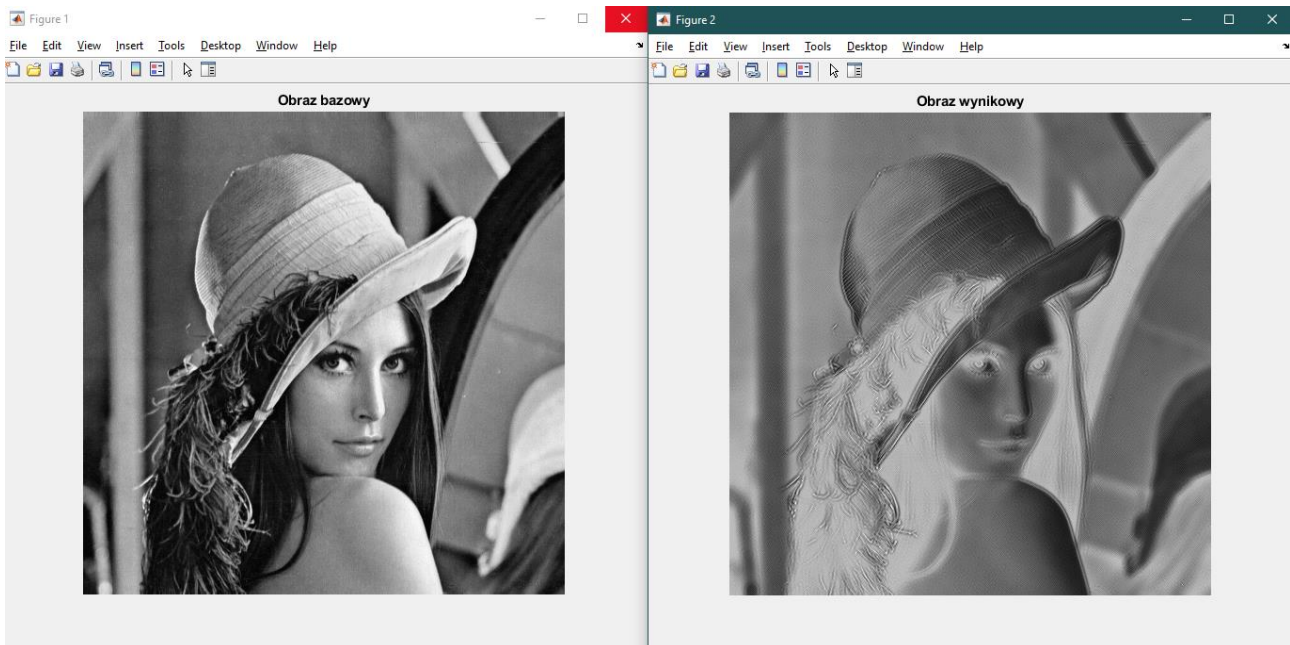
- Gaussa 3x3:



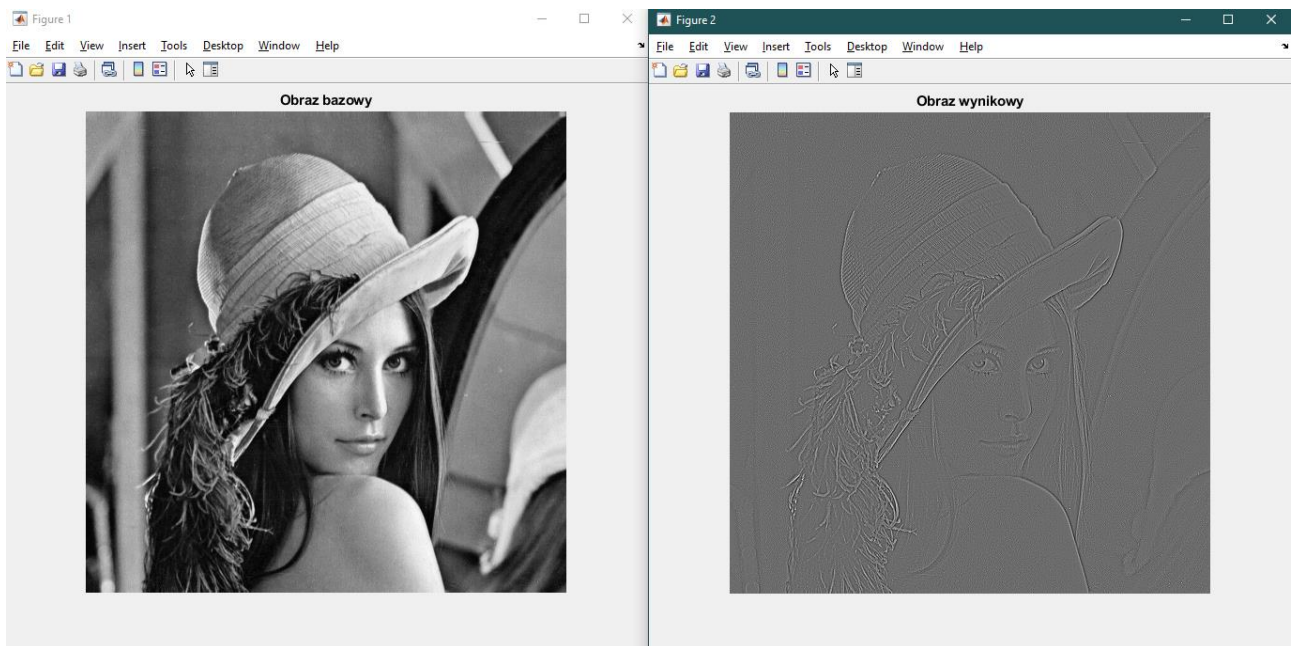
- Górnoprzepustowa 3x3:



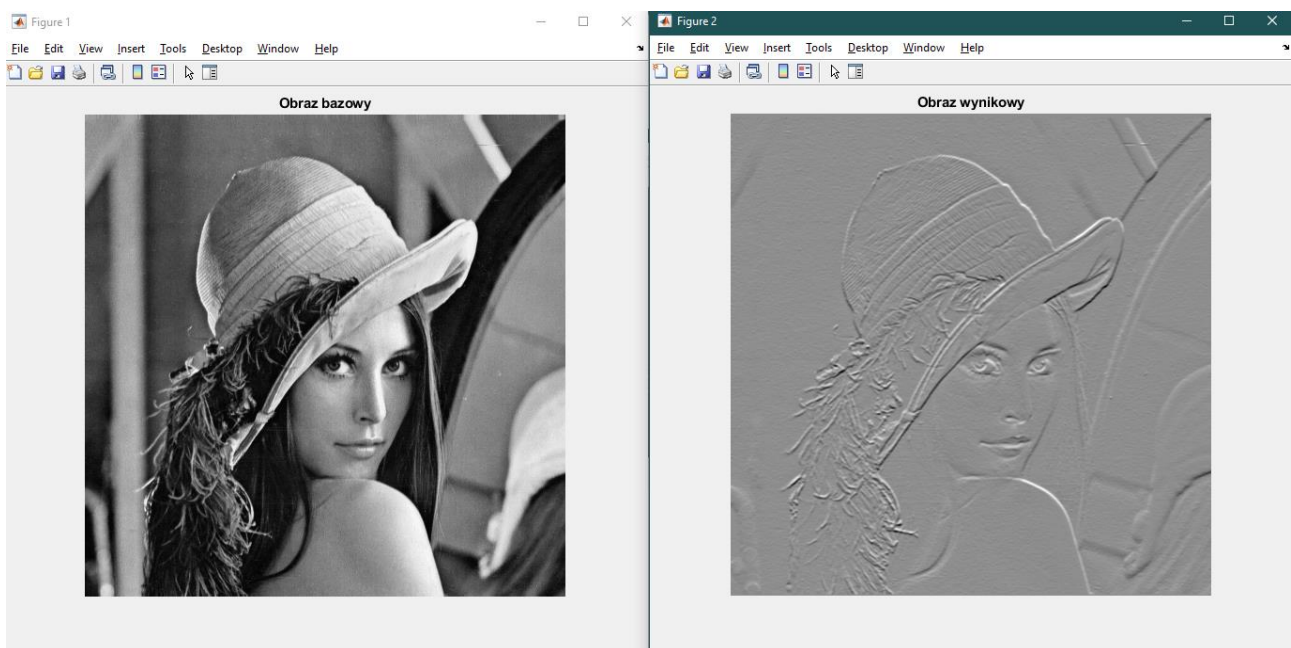
- Górnoprzepustowa 5x5:



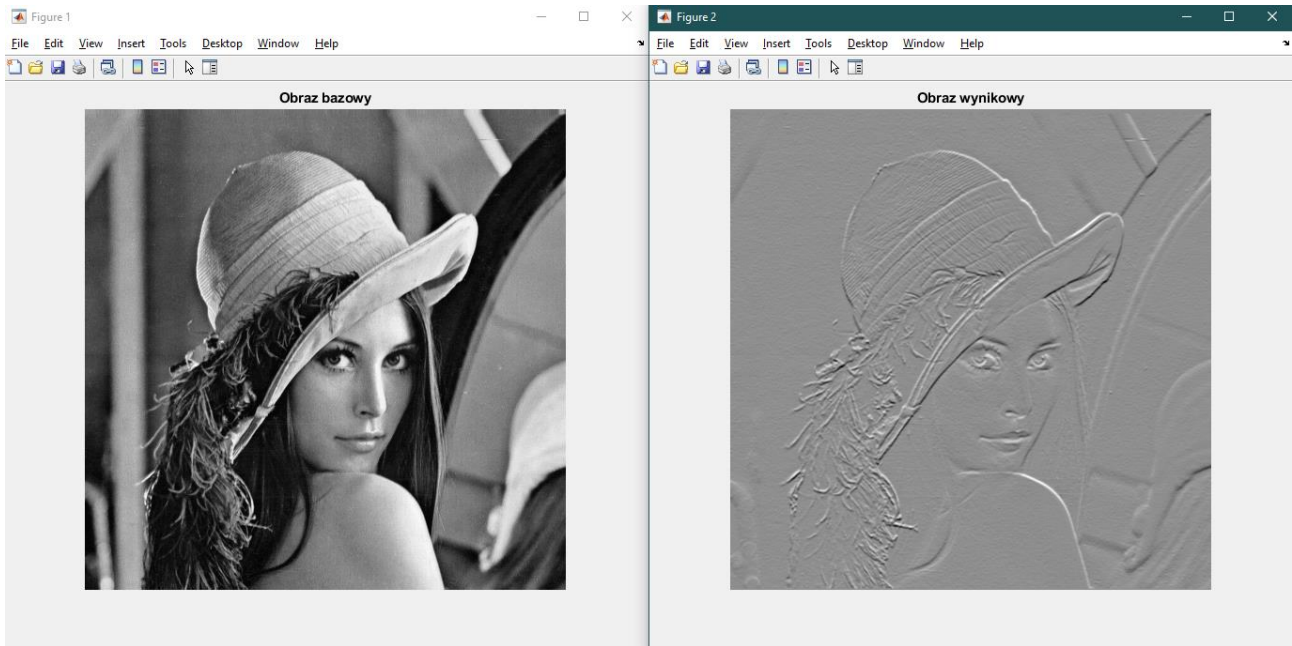
- Laplace'a:



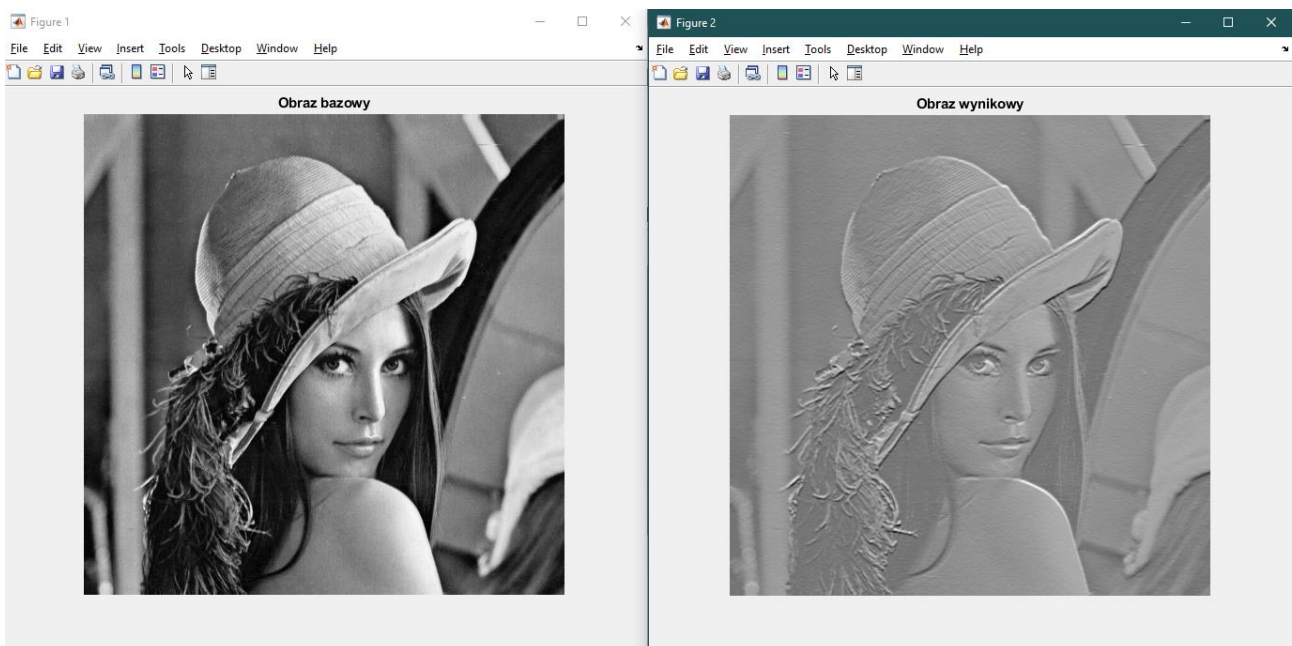
- Prewitta:



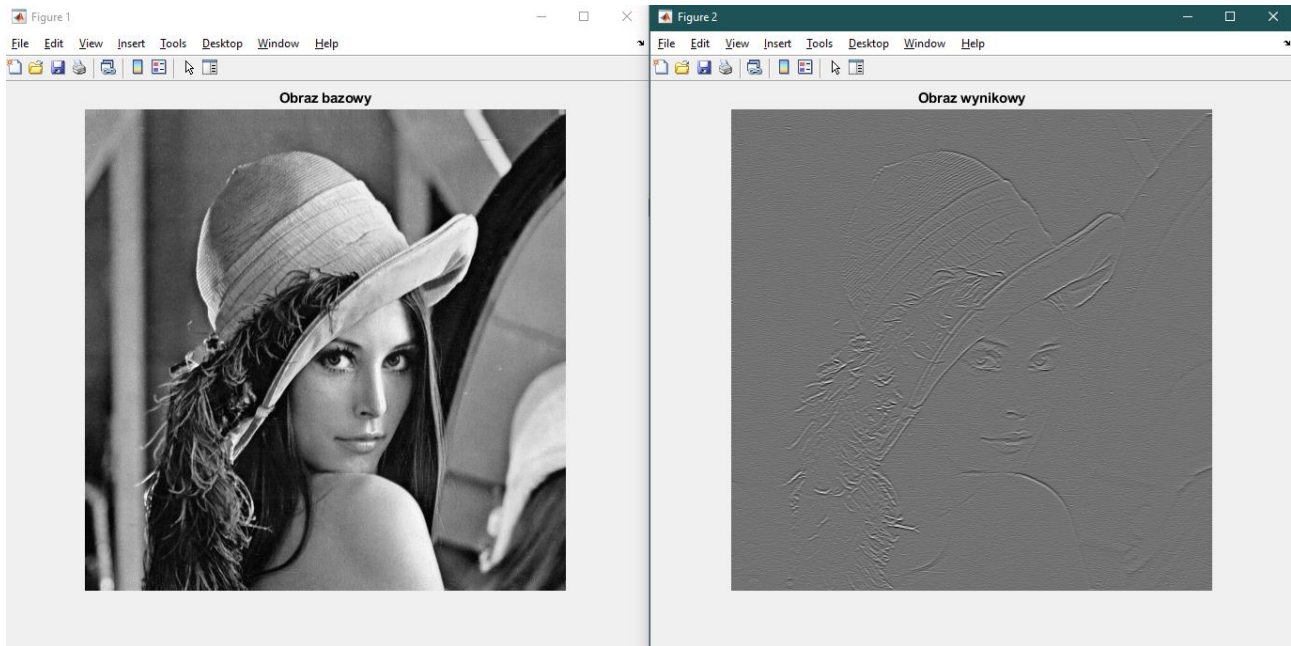
- Sobela:



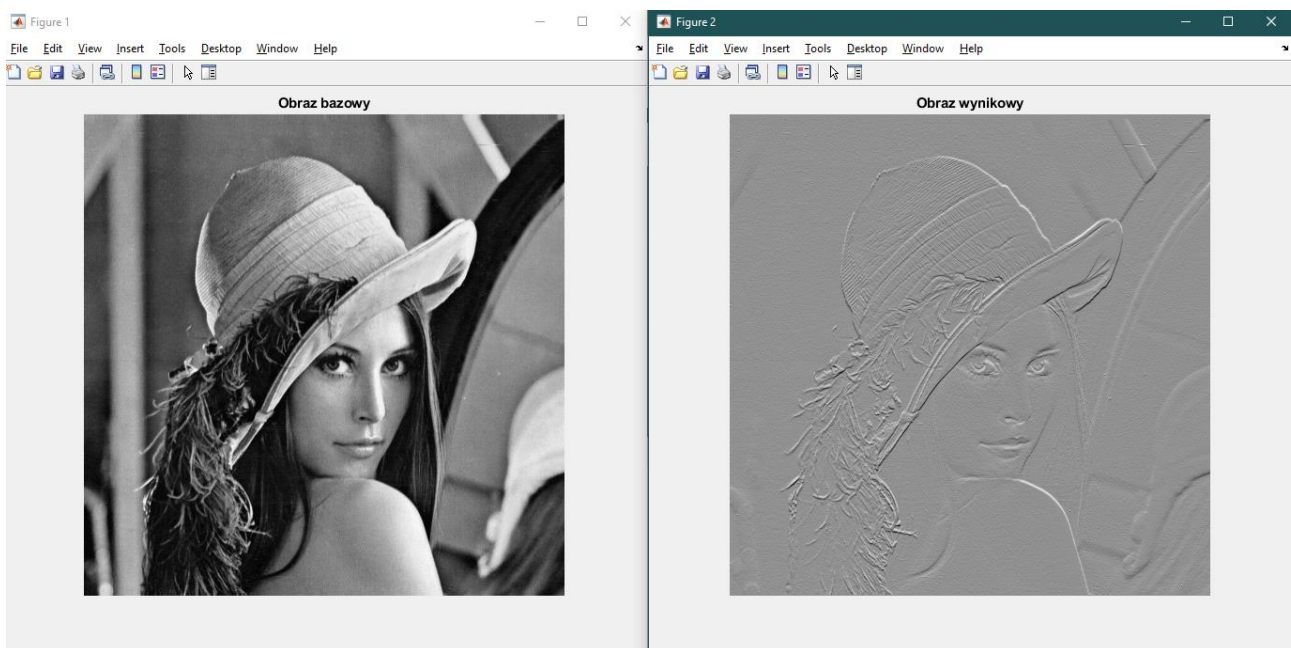
- Emboss:



- Detekcja linii (pierwsza maska):

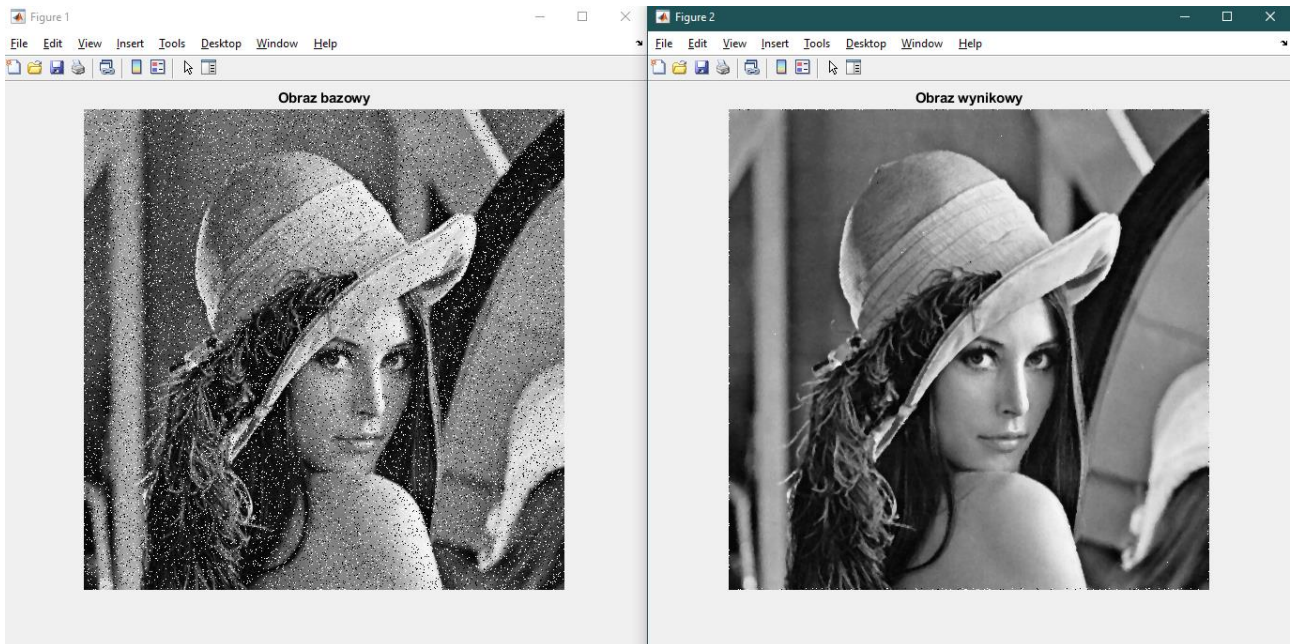


- Detekcja linii (druga maska):

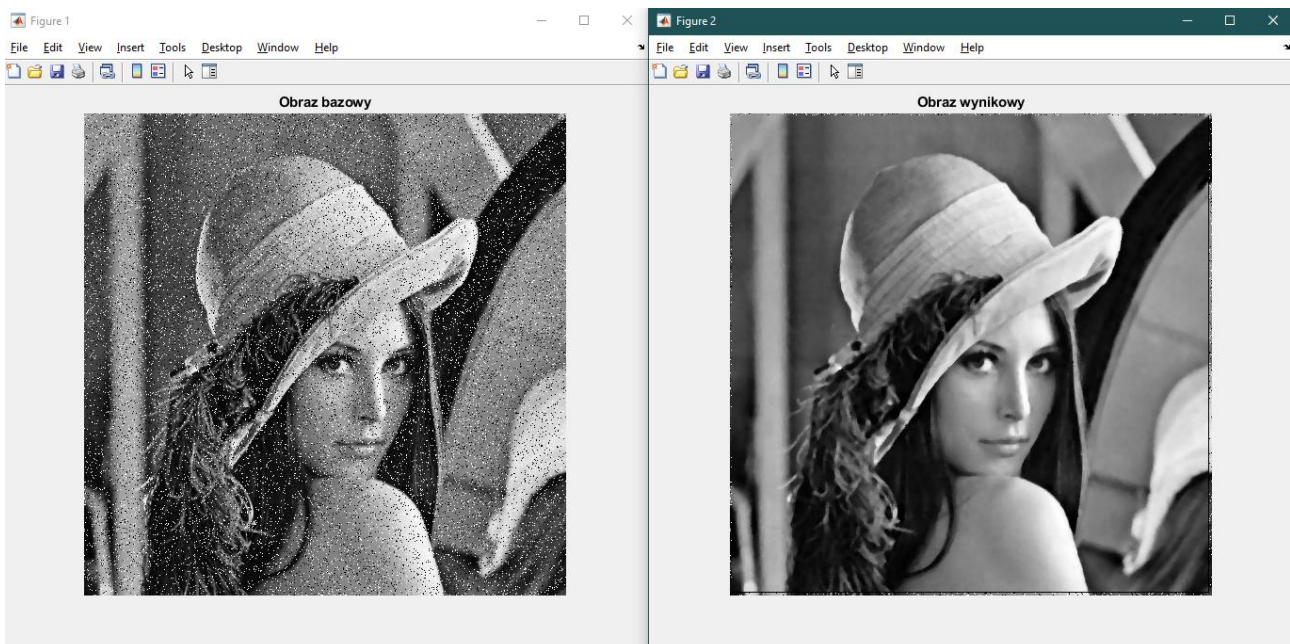


Filtracja medianowa:

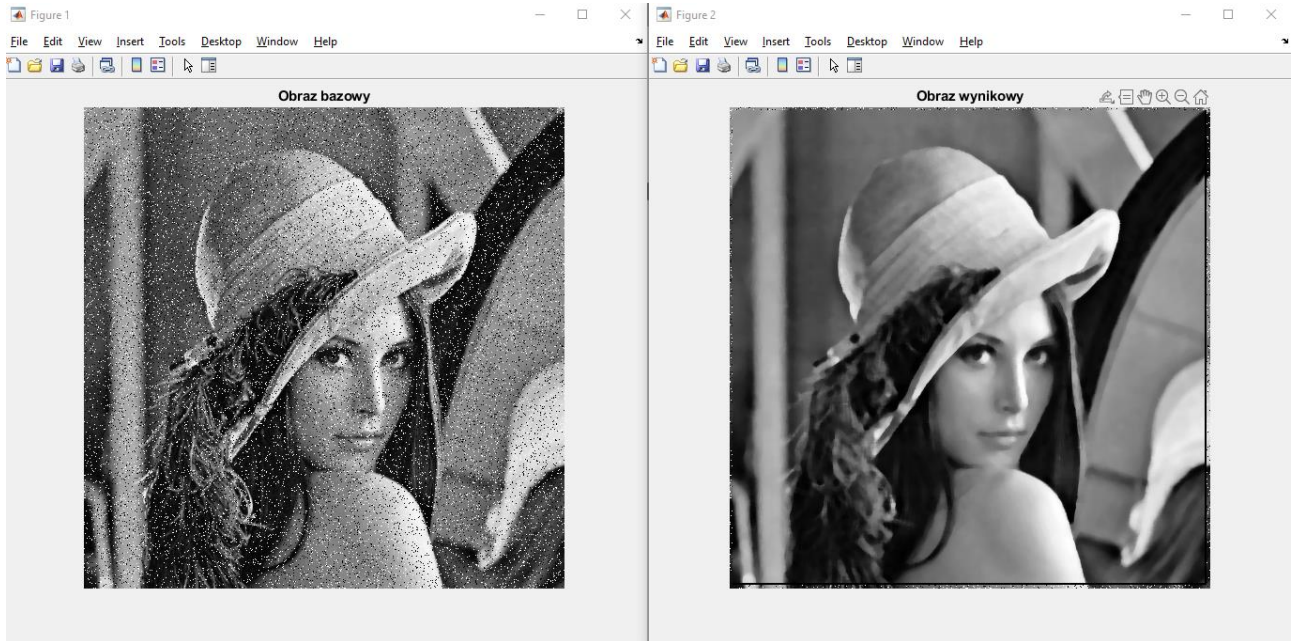
- Maska 3x3:



- Maska 5x5:



- Maska 7x7:



4. Wnioski:

Przy filtracji medianowej, najmniej stratne pozbycie się szumów ma miejsce przy zastosowaniu maski 3x3. Stosując maski dolnoprzepustowe uśredniające, w filtrach splotowych skutkuje rozmyciem obrazu. Jest ono tym mocniejsze im wyższa jest wielkość maski. Filtr górnoprzepustowy 5x5 daje wrażenie obrazu w negatywie.