



**Silesian
University
of Technology**

Faculty of Automatic Control,
Electronics and Computer Science

PRZETWARZANIE OBRAZÓW CYFROWYCH

Lab1. Filtry nieliniowe

Autor:

Michał Siedlaczek

Gliwice 2021

Contents

| | |
|---|-----------|
| 1 Przygotowanie obrazów testowych | 1 |
| 2 Zad1 | 2 |
| 3 Implementacja filtru medianowego | 4 |
| 4 Implementacja filtru LUM | 5 |
| 5 Porównanie działania zaimplementowanych filtrów | 6 |
| 6 Zad3a | 7 |
| 7 Zad3b | 8 |
| 8 Kod | 9 |
| 9 Kod3a | 15 |
| 10 Kod3b | 18 |
| 11 Porównanie działania zaimplementowanych filtrów | 22 |
| 12 Wnioski | 23 |
| Bibliography | 24 |

Chapter 1

Przygotowanie obrazów testowych



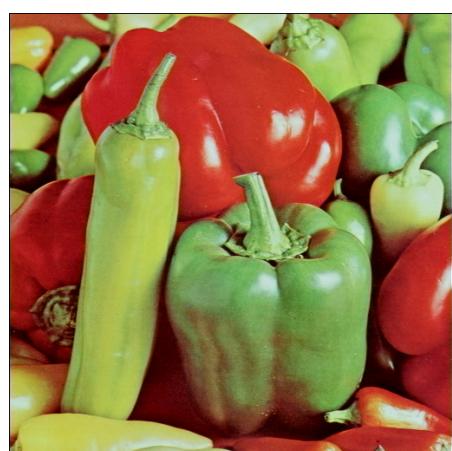
(a) Boats szare



(b) Papryka szare



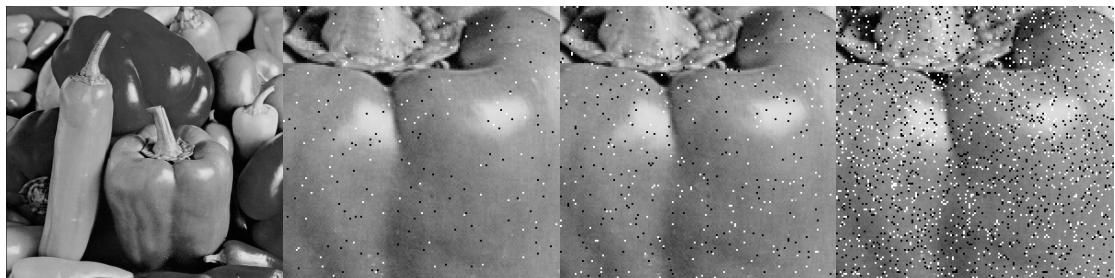
(c) Boats Kolor



(d) Papryka Kolor

Chapter 2

Zad1



(a) Papryka oryginał

(b) Papryka szum 1

(c) Papryka szum 2

(d) Papryka szum 10



(a) Boats oryginał

(b) Boats szum 1

(c) Boats szum 2

(d) Boats szum 10

Sprawdzenie dla obrazów barwnych



(a) Boats oryginał

(b) Boats szum 1

(c) Boats szum 2

(d) Boats szum 10



(a) Papryka gauss

(b) Papryka gauss10

(c) Papryka gauss20

(d) Papryka gauss30



(a) pomieszany

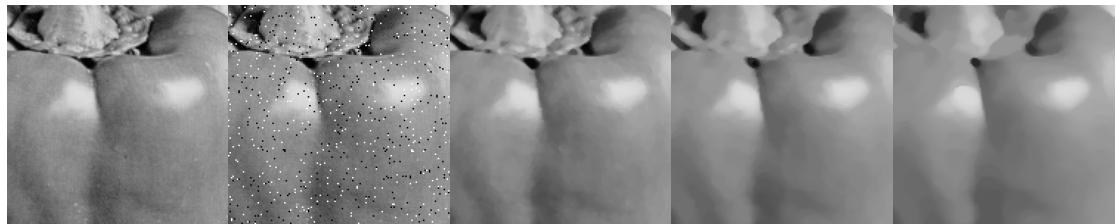
(b) pomieszany1

(c) pomieszany2

(d) pomieszany3

Chapter 3

Implementacja filtru medianowego



(a) Papryka oryginał
PSNR = 21,17
(b) Papryka szum1
PSNR = 21,17
(c) Mediana 3x3
PSNR = 31,42
(d) Mediana 5x5
PSNR=27,70
(e) Mediana 7x7
PSNR=25,11



(a) Boats oryginał
PSNR = 24,53
(b) Boats szum1
PSNR = 24,53
(c) Mediana 3x3
PSNR = 29,09
(d) Mediana 5x5
PSNR = 25,05
(e) Mediana 7x7
PSNR = 23,32

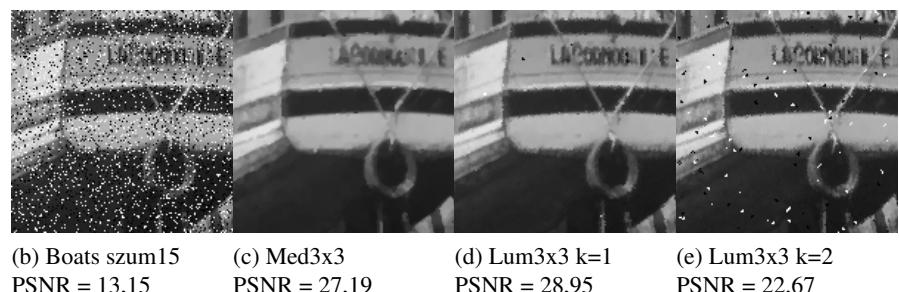
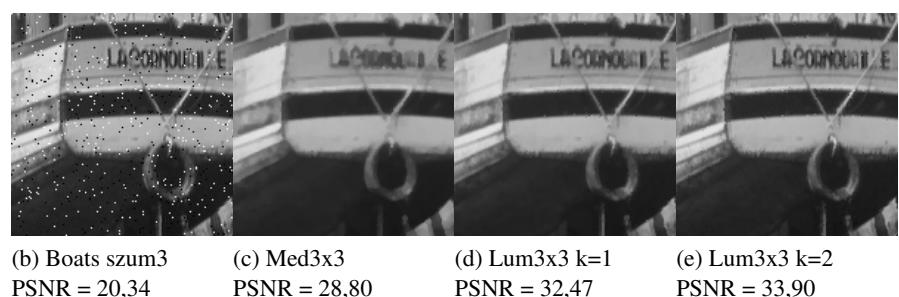
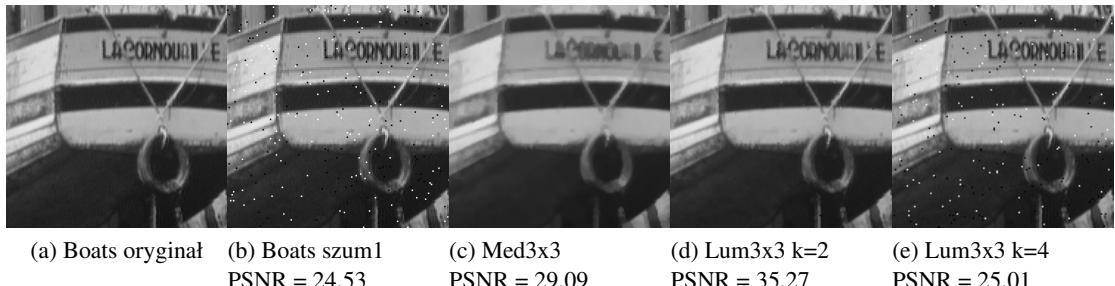
Chapter 4

Implementacja filtru LUM



Chapter 5

Porównanie działania zaimplementowanych filtrów

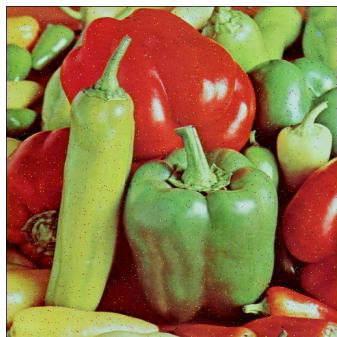


Chapter 6

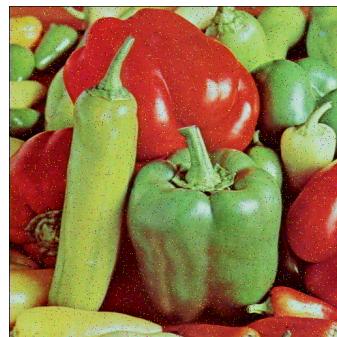
Zad3a



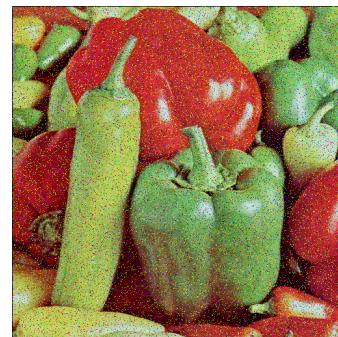
(a) oryginal



(b) szum1



(c) szum2



(d) szum10



(e) Y dla 1 szumu



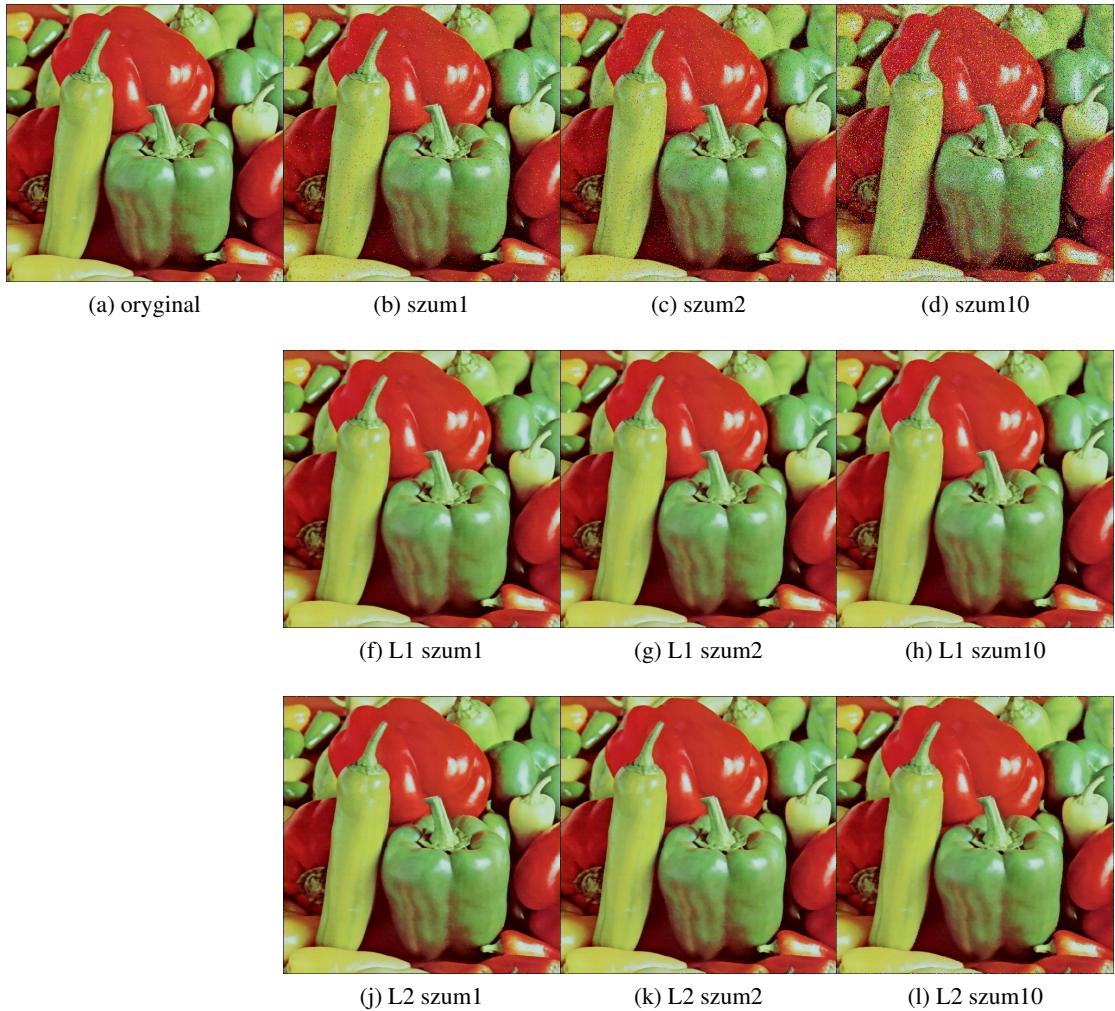
(f) Y dla 2 szumu



(g) Y dla 10 szumu

Chapter 7

Zad3b



Chapter 8

Kod

```
clear all;
clc;

Img = imread('boat512x512.bmp');

rows = size(Img,1);
columns = size(Img,2);

x1=imnoise(Img,'salt pepper',0.01);
x2=imnoise(Img,'salt pepper',0.03);
x10=imnoise(Img,'salt pepper',0.15);

xg=imnoise(Img, 'gaussian',0.01,0.01);
xg10=imnoise(Img, 'gaussian',0.01,0.10);
xg20=imnoise(Img, 'gaussian',0.01,0.20);
xg30=imnoise(Img, 'gaussian',0.01,0.30);

pomieszany=imnoise(Img, 'gaussian',0.01,0.01)+imnoise(Img,'salt pepper',0.01);
pomieszany1=imnoise(Img, 'gaussian',0.01,0.10)+imnoise(Img,'salt pepper',0.03);
pomieszany2=imnoise(Img, 'gaussian',0.01,0.20)+imnoise(Img,'salt pepper',0.15);
pomieszany3=imnoise(Img, 'gaussian',0.01,0.30)+imnoise(Img,'salt pepper',0.20);
imwrite(xg,'xg.png');
imwrite(xg10,'xg10.png');
imwrite(xg20,'xg20.png');
imwrite(xg30,'xg30.png');
```

```

imwrite(pomieszany,'pomieszany.png');
imwrite(pomieszany1,'pomieszany1.png');
imwrite(pomieszany2,'pomieszany2.png');
imwrite(pomieszany3,'pomieszany3.png');

ImgR=[Img(250:400,250:400,:),x1(250:400,250:400,:),x2(250:400,250:400,:),x10(250:400,250:400,:)];
ImgL=imresize(ImgR,3,"nearest");
imwrite(ImgL,'szumzad1SP.png');

szum1=x1(250:400,250:400,:);
szum2=x2(250:400,250:400,:);
szum10=x10(250:400,250:400,:);
ImgLargeS=imresize(szum1,3,"nearest");
imwrite(ImgLargeS,'szumik11.png');
ImgLarge1S=imresize(szum2,3,"nearest");
imwrite(ImgLarge1S,'szumik32.png');
ImgLarge2S=imresize(szum10,3,"nearest");
imwrite(ImgLarge2S,'szumik10.png');

Size=1; filtr=1; k=1;
Y = x1;
for i=(1+Size):(rows-Size)
for j=(1+Size):(columns-Size)
Minor = Y(i-Size:i+Size,j-Size:j+Size);
Minor = Minor(:);
Rosnaco = sort(Minor);
if (filtr==1)
if(Size==1)
Mediana = Rosnaco(5);
end
if(Size==2)
Mediana = Rosnaco(13);
end
if(Size==3)
Mediana = Rosnaco(25);

```

```

end
end
if (filtr==2)
if(Size==1)
Rosnaco2=sort([Rosnaco(5-k) Minor(5) Rosnaco(5+k)]);
Mediana = Rosnaco2(2);
end
if(Size==2)
Rosnaco2=sort([Rosnaco(13-k) Minor(13) Rosnaco(13+k)]);
Mediana = Rosnaco2(2);
end
if(Size==3)
Rosnaco2=sort([Rosnaco(25-k) Minor(25) Rosnaco(25+k)]);
Mediana = Rosnaco2(2);
end
end
Y(i,j) = Mediana;
end
end
Y1=Y;

Y = x2;
for i=(1+Size):(rows-Size)
for j=(1+Size):(columns-Size)
Minor = Y(i-Size:i+Size,j-Size:j+Size);
Minor = Minor(:);
Rosnaco = sort(Minor);
if (filtr==1)
if(Size==1)
Mediana = Rosnaco(5);
end
if(Size==2)
Mediana = Rosnaco(13);
end
if(Size==3)
Mediana = Rosnaco(25);

```

```

end
end
if (filtr==2)
if(Size==1)
Rosnaco2=sort([Rosnaco(5-k) Minor(5) Rosnaco(5+k)]);
Mediana = Rosnaco2(2);
end
if(Size==2)
Rosnaco2=sort([Rosnaco(13-k) Minor(13) Rosnaco(13+k)]);
Mediana = Rosnaco2(2);
end
if(Size==3)
Rosnaco2=sort([Rosnaco(25-k) Minor(25) Rosnaco(25+k)]);
Mediana = Rosnaco2(2);
end
end
Y(i,j) = Mediana;
end
end
Y2=Y;

```

```

Y = x10;
for i=(1+Size):(rows-Size)
for j=(1+Size):(columns-Size)
Minor = Y(i-Size:i+Size,j-Size:j+Size);
Minor = Minor(:);
Rosnaco = sort(Minor);
if (filtr==1)
if(Size==1)
Mediana = Rosnaco(5);
end
if(Size==2)
Mediana = Rosnaco(13);
end
if(Size==3)
Mediana = Rosnaco(25);

```

```

end
end
if (filtr==2)
if(Size==1)
Rosnaco2=sort([Rosnaco(5-k) Minor(5) Rosnaco(5+k)]);
Mediana = Rosnaco2(2);
end
if(Size==2)
Rosnaco2=sort([Rosnaco(13-k) Minor(13) Rosnaco(13+k)]);
Mediana = Rosnaco2(2);
end
if(Size==3)
Rosnaco2=sort([Rosnaco(25-k) Minor(25) Rosnaco(25+k)]);
Mediana = Rosnaco2(2);
end
Y(i,j) = Mediana;
end
end
Y10=Y;

```

```

ImgR=[Img(250:400,250:400,:);Y1(250:400,250:400,:);Y2(250:400,250:400,:);Y10(250:400,250:400,:)];
ImgL=imresize(ImgR,3,"nearest");
imwrite(ImgL,'wynikzad1LUMPK.png');

```

```

if(filtr==1)
mediana1=Y1(250:400,250:400,:);
mediana2=Y2(250:400,250:400,:);
mediana10=Y10(250:400,250:400,:);
Obraz=Img(250:400,250:400,:);
ImgLargeO=imresize(Obraz,3,"nearest");
imwrite(ImgLargeO,'Obrazek.png');
ImgLarge=imresize(mediana1,3,"nearest");
imwrite(ImgLarge,'mediana1.png');
ImgLarge1=imresize(mediana2,3,"nearest");
imwrite(ImgLarge1,'mediana2.png');

```

```
ImgLarge2=imresize(mediana10,3,"nearest");
imwrite(ImgLarge2,'mediana10.png');
end
```

```
if(filtr==2)
lum1=Y1(250:400,250:400,:);
lum2=Y2(250:400,250:400,:);
lum10=Y10(250:400,250:400,:);
Obraz=Img(250:400,250:400,:);
ImgLarge=imresize(lum1,3,"nearest");
imwrite(ImgLarge,'lum1.png');
ImgLarge1=imresize(lum2,3,"nearest");
imwrite(ImgLarge1,'lum2.png');
ImgLarge2=imresize(lum10,3,"nearest");
imwrite(ImgLarge2,'lum10.png');
end
```

Chapter 9

Kod3a

```
I=imread("peppers3512x512.bmp");
imshow(I);
figure;

x1=imnoise(I,'salt pepper',0.01);
x2=imnoise(I,'salt pepper',0.02);
x10=imnoise(I,'salt pepper',0.10);
imshow(x1);
figure;
imshow(x2);
figure;
imshow(x10);
figure;
[rows,columns,kanal] = size(I);
promienokna = 1;
rozmiarokna = (2 * promienokna + 1).^2;

Y=x1;
for i=(1+promienokna) : (rows - promienokna)
    for j = (1 + promienokna) : (columns - promienokna)
        vectorssetR = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 1), rozmiarokna, 1, 1);
        vectorssetG = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 2), rozmiarokna, 1, 1);
```

```

vectorssetB = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 3), rozmiarokna, 1, 1);
vecR = median(vectorssetR);
vecG = median(vectorssetG);
vecB = median(vectorssetB);
vec = [vecRvecGvecB];
Y(i, j, :) = vec;
end
end

```

```

Y2=x2;
for i=(1+promienokna) : (rows - promienokna)

for j = (1 + promienokna) : (columns - promienokna)
vectorssetR = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 1), rozmiarokna, 1, 1);
vectorssetG = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 2), rozmiarokna, 1, 1);
vectorssetB = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 3), rozmiarokna, 1, 1);
vecR = median(vectorssetR);
vecG = median(vectorssetG);
vecB = median(vectorssetB);
vec = [vecRvecGvecB];
Y2(i, j, :) = vec;
end
end

```

```

Y10=x10;
for i=(1+promienokna) : (rows - promienokna)

for j = (1 + promienokna) : (columns - promienokna)
vectorssetR = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 1), rozmiarokna, 1, 1);
vectorssetG = reshape(Y(i - promienokna : i + promienokna, j - promienokna :
j + promienokna, 2), rozmiarokna, 1, 1);
vectorssetB = reshape(Y(i - promienokna : i + promienokna, j - promienokna :

```

```

 $j + promien_{okna}, 3), rozmiar_{okna}, 1, 1);$ 
 $vec_R = median(vectors_{setR});$ 
 $vec_G = median(vectors_{setG});$ 
 $vec_B = median(vectors_{setB});$ 
 $vec = [vec_R vec_G vec_B];$ 
 $Y10(i, j, :) = vec;$ 
end
end

imshow(Y);
imwrite(Y,'YImage.png');
figure;
imshow(Y2);
imwrite(Y2,'Y2Image.png');
figure;
imshow(Y10);
imwrite(Y10,'Y10Image.png');

```

Chapter 10

Kod3b

```
I = imread('peppers3512x512.bmp');
imshow(I);
figure;
x1 = imnoise(I,'saltpepper',0.01);
x2 = imnoise(I,'saltpepper',0.02);
x10 = imnoise(I,'saltpepper',0.10);
imshow(x1);
figure;
imshow(x2);
figure;
imshow(x10);
figure;
L1Image = x1;
L2Image2 = x1;
L1Image2 = x2;
L2Image2x2 = x2;
L1Image10 = x10;
L2Image2x10 = x10;

[rows,columns,kanal] = size(I);
promienokna = 1;
rozmiarokna = (2 * promienokna + 1).^2;
for i = 1 + promienokna : rows - 2
    for j = 1 + promienokna : columns - 2
        vectorsset = reshape(L1Image(i - promienokna : i + promienokna, j - promienokna :
```

```

j + promienokna,:), rozmiarokna, 1, kanal);
vectorsset = reshape(permute(vectorsset,[2,3,1]),kanal,[])';
[n,d] = size(vectorsset);
dist = pdist(vectorsset,'cityblock');
dist = squareform(dist);
dist = sum(dist,2);
indx = find(dist == min(dist));
medianvec = vectorsset(indx,:);
L1Image(i,j,:) = medianvec(1,:);
dist2 = pdist(vectorsset,'euclidean');
dist2 = squareform(dist2);
dist2 = sum(dist2,2);
indx = find(dist2 == min(dist2));
medianvec = vectorsset(indx,:);
L2Image2(i,j,:) = medianvec(1,:);
end
end

for i = 1+promienokna : rows - 2
for j = 1 + promienokna : columns - 2
vectorsset = reshape(L1Imagex2(i-promienokna : i+promienokna, j-promienokna :
j + promienokna,:), rozmiarokna, 1, kanal);
vectorsset = reshape(permute(vectorsset,[2,3,1]),kanal,[])';
[n,d] = size(vectorsset);
dist = pdist(vectorsset,'cityblock');
dist = squareform(dist);
dist = sum(dist,2);
indx = find(dist == min(dist));
medianvec = vectorsset(indx,:);
L1Imagex2(i,j,:) = medianvec(1,:);
dist2 = pdist(vectorsset,'euclidean');
dist2 = squareform(dist2);
dist2 = sum(dist2,2);
indx = find(dist2 == min(dist2));
medianvec = vectorsset(indx,:);
L2Image2x2(i,j,:) = medianvec(1,:);

```

```

end
end

for i = 1+promien_o{kna : rows - 2
for j = 1 + promien_o{kna : columns - 2
vectors_set = reshape(L1Imagex10(i-promien_o{kna : i+promien_o{kna, j-promien_o{kna :
j + promien_o{kna,:}), rozmiar_o{kna, 1, kanal);
vectors_set = reshape(permute(vectors_set, [2, 3, 1]), kanal, [] )';
[n, d] = size(vectors_set);
dist = pdist(vectors_set, 'cityblock');
dist = squareform(dist);
dist = sum(dist, 2);
indx = find(dist == min(dist));
median_vec = vectors_set(indx, :);
L1Image10(i, j, :) = median_vec(1, :);
dist2 = pdist(vectors_set, 'euclidean');
dist2 = squareform(dist2);
dist2 = sum(dist2, 2);
indx = find(dist2 == min(dist2));
median_vec = vectors_set(indx, :);
L2Image2x10(i, j, :) = median_vec(1, :);
end
end

imshow(L1Image);
imwrite(L1Image,'L1Image.png');
figure;

imshow(L2Image2);
figure;
imwrite(L2Image2,'L2Image2.png');

imshow(L1Imagex2);
imwrite(L1Imagex2,'L1Imagex2.png');
figure;

```

```
imshow(L2Image2x2);
imwrite(L2Image2x2,'L2Image2x2.png');
figure;

imshow(L1Imagex10);
imwrite(L1Imagex10,'L1Imagex10.png');
figure;

imshow(L2Image2x10);
imwrite(L2Image2x10,'L2Image2x10.png');
```

Chapter 11

Porównanie działania zaimplementowanych filtrów



Chapter 12

Wnioski

Jak widać PSNR wychodzi najmniejsze dla największego zakłócenia, co powoduje że obraz jest mniej zbliżony do oryginalnego obrazu. Również im większe k tym gorsze PSNR dla filtra LUM. Jak widać dla modyfikacji mediany skalarnej jak i wektorowego filtru medianowego wyniki PSNR wyszły zadowalające.

Bibliography