Przetwarzanie Obrazów Cyfrowych

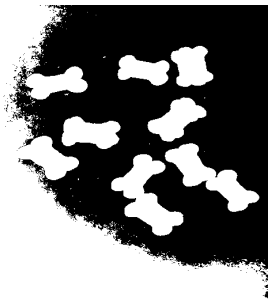
Raport z ćwiczenia nr. 3:
Metody segmentacji obrazów

Raport opracował:
Dawid Kania
Grupa 6 Semestr 7

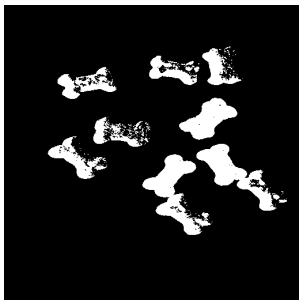
Data wykonania ćwiczenia: 28.11.2022

Zadanie 2

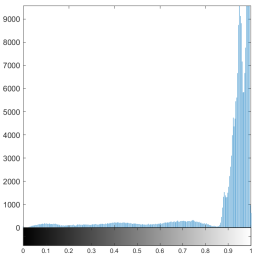
O_k

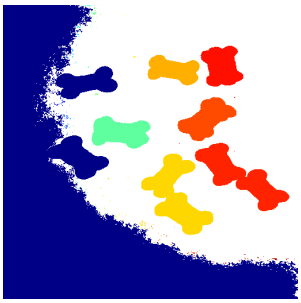


Wartość progu $t = 0.96078$



Wartość progu $t_o = 0.65882$

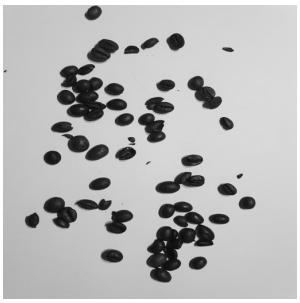




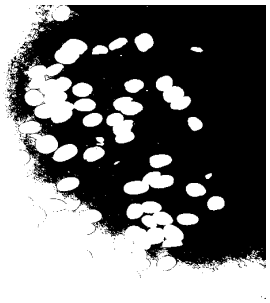
Liczba obszarów = 201

Liczba obszarów = 160

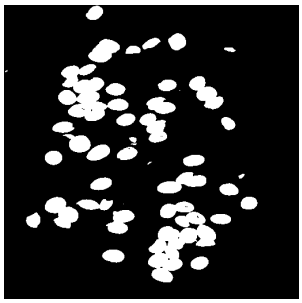
Figure 1: Porownanie



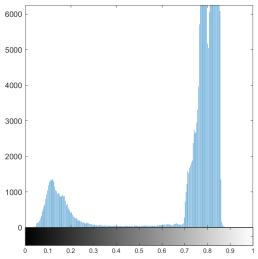
O_k

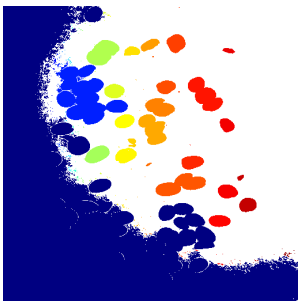


Wartość progu $t = 0.78824$

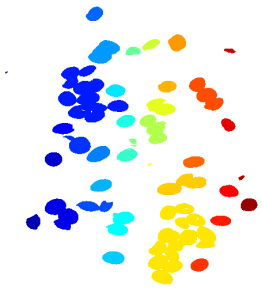


Wartość progu $t_o = 0.47451$





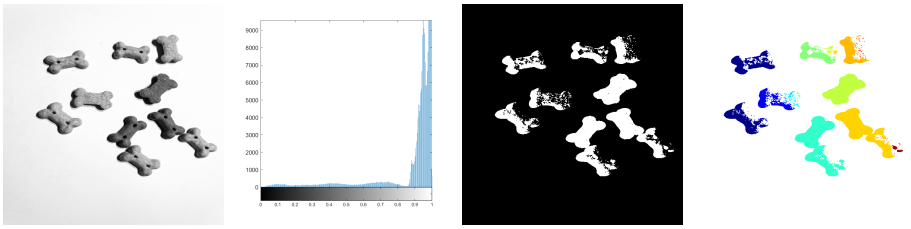
Liczba obszarów = 258



Liczba obszarów = 39

Figure 2: Porownanie

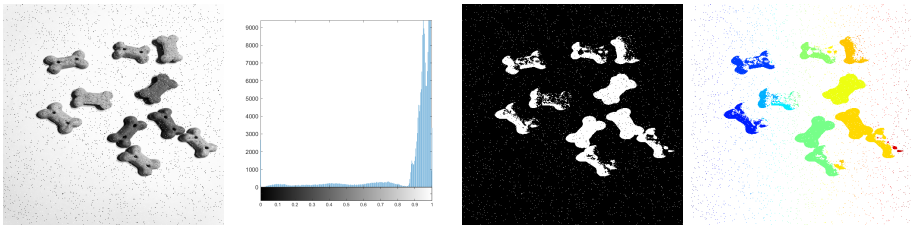
Zadanie 3



Oryginalny

Wartość progu $t = 0.65882$

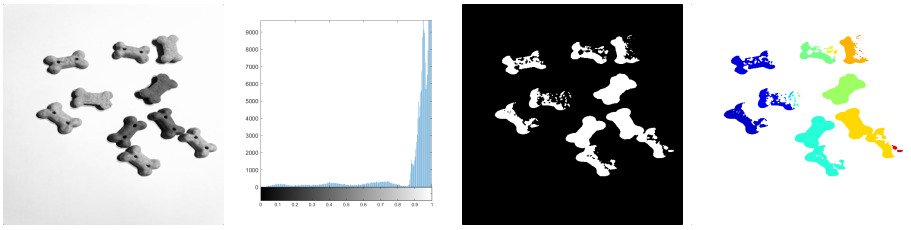
Liczba obszarów 160



Zakłócony

Wartość progu $t = 0.63137$

Liczba obszarów 2350

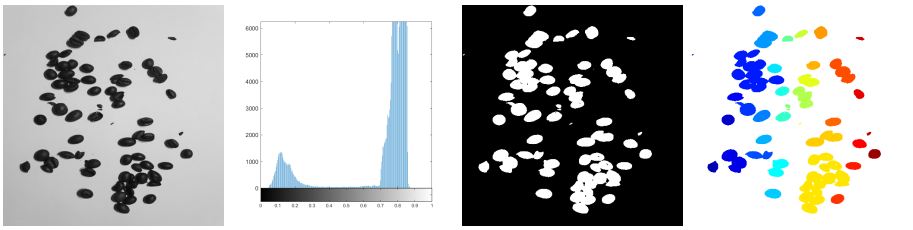


Filtrowany

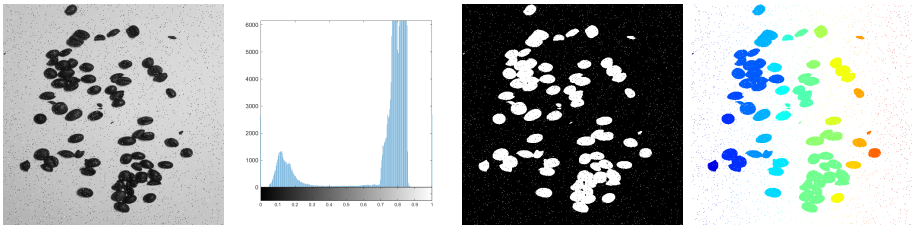
Wartość progu $t_2 = 0.66275$

Liczba obszarów 49

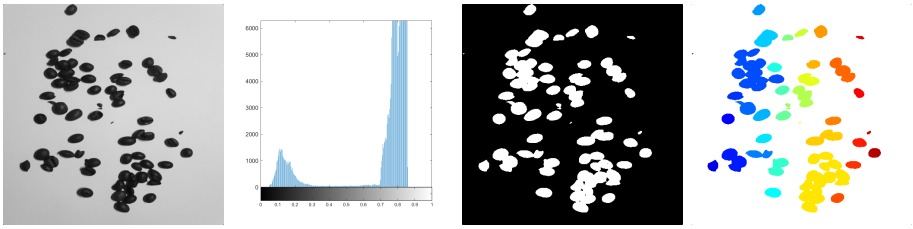
Figure 3: Porownanie



Oryginalny | Wartość progu $t = 0.47451$ | Liczba obszarów 39



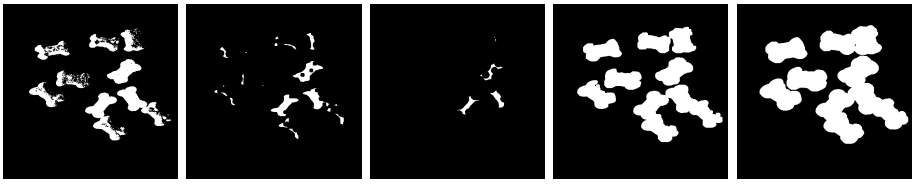
Zakłócony | Wartość progu $t = 0.47059$ | Liczba obszarów 2170



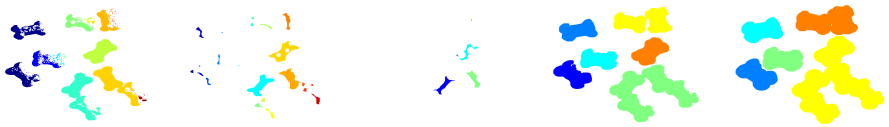
Filtrowany | Wartość progu $t_2 = 0.47451$ | Liczba obszarów 40

Figure 4: Porownanie

Zadanie 4

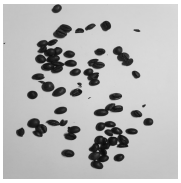


Wartość progu t = 160	s = offset-strel('ball',5,5); imerode(I, s);	s = offset-strel('ball',10,10); imerode(I, s);	s = offset-strel('ball',5,5); imdilate(I, s);	s = offset-strel('ball',10,10); imdilate(I, s);

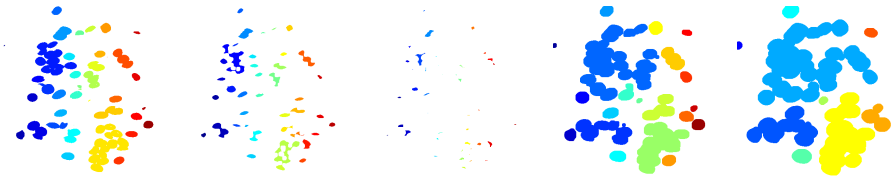


Liczba obszarów = 160	Liczba obszarów = 22	Liczba obszarów = 6	Liczba obszarów = 6	Liczba obszarów = 5

Figure 5: Porownanie



Wartość progu t = 39	s = offset-strel('ball',5,5); imerode(I, s);	s = offset-strel('ball',10,10); imerode(I, s);	s = offset-strel('ball',5,5); imdilate(I, s);	s = offset-strel('ball',10,10); imdilate(I, s);



Liczba obszarów = 39	Liczba obszarów = 42	Liczba obszarów = 42	Liczba obszarów = 20	Liczba obszarów = 9

Figure 6: Porownanie

Kody programów

zad2.m

```
clear all
close all


disp("Zad 2")

disp("   Img 1")
ProcessImages("../images/20191121_072040.jpg", "../zad2/Img1");

disp("   Img 2")
ProcessImages("../images/20191121_072227.jpg", "../zad2/Img2");




function ProcessImages(image_path, dest_folder)


    I = imread(image_path);
    I = rgb2gray(I);
    I = double(I) / 255;
    I = imresize(I, [512 512]);

    [I1, t1] = ImageBinarize(I, "hist");
    [I2, t2] = ImageBinarize(I, "otsu");

    I1 = -I1 + 1;
    I2 = -I2 + 1;

    [~, n1] = bwlabel(I1);
    [~, n2] = bwlabel(I2);

    I1_col = label2rgb(bwlabel(I1));
    I2_col = label2rgb(bwlabel(I2));

    % figure; imshow(I);
    % figure; imshow(I1); title("wartosc progu t1 = " + string(t1))
    % figure; imshow(I2); title("wartosc progu t2 = " + string(t2))
    %
    % figure; imhist(I);
    % figure; imshow(I1_col); title("Liczba óobszarw = " + string(
    n1))
    % figure; imshow(I2_col); title("Liczba óobszarw = " + string(
    n2))


    mkdir(dest_folder);

    imwrite(cropToSquare(I), dest_folder + "/I.png");
    imwrite(cropToSquare(I1), dest_folder + "/I1.png");
    imwrite(cropToSquare(I2), dest_folder + "/I2.png");

    HistSave(I, dest_folder + "/I_hist.png")
```

```
    imwrite(cropToSquare(I1_col), dest_folder + "/I1_col.png");
    imwrite(cropToSquare(I2_col), dest_folder + "/I2_col.png");


        Latex = [
    "\newcommand{\ww}{0.32} "
    "\begin{figure}[H] "
    "   \captionsetup[subfloat]{justification=raggedright,
    singlelinecheck=false, position=bottom,labelformat=empty} % "
    "   "
    "    \subfloat[ $O_k$ ]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I.png}}  \hfill% "
    "    \subfloat[ śćWarto progu $t$ = " + string(t1) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I1.png}}  \hfill% "
    "    \subfloat[ śćWarto progu $t_o$ = " + string(t2) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2.png}}  \hfill%"
    ""
    "    \subfloat[]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I_hist.png}}  \hfill% "
    "    \subfloat[Liczba óobszarw = " + string(n1) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I1_col.png}}  \hfill% "
    "    \subfloat[Liczba óobszarw = " + string(n2) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2_col.png}}  \hfill%"
    " "
    "\caption{Porownanie}"
    ""
    "\end{figure} "
    "\let\ww\undefined "
        ];

        Latex = join(Latex,[''],2);
        Latex = join(Latex,[newline],1);

        fid = fopen(dest_folder + "/result.tex",'wt');
        fprintf(fid,"%s", Latex);
        fclose(fid);


end



function HistSave(I, dest)
    f = figure;
    f.Position = [0,0,512,512];
    imhist(I);
    saveas(gcf, dest)
    close(gcf);
end
```

zad3.m

```
clear all
close all


disp("Zad 3")

disp("   Img 1")
ProcessImages("../images/20191121_072040.jpg", "../zad3/Img1");

disp("   Img 2")
ProcessImages("../images/20191121_072227.jpg", "../zad3/Img2");



function ProcessImages(image_path, dest_folder)


    I = imread(image_path);
    I = rgb2gray(I);
    I = double(I) / 255;
    I = imresize(I, [512, 512]);

    I1 = imnoise(I, "salt & pepper", .02);
    %I2 = imfilter(I, fspecial("gaussian", 3, 1));
    I2 = medfilt2(I,[3 3]);

    [I_b, t] = ImageBinarize(I, "otsu");
    [I1_b, t1] = ImageBinarize(I1, "otsu");
    [I2_b, t2] = ImageBinarize(I2, "otsu");

    I_b = -I_b + 1;
    I1_b = -I1_b + 1;
    I2_b = -I2_b + 1;

    [~, n] = bwlabel(I_b);
    [~, n1] = bwlabel(I1_b);
    [~, n2] = bwlabel(I2_b);

    I_col = label2rgb(bwlabel(I_b));
    I1_col = label2rgb(bwlabel(I1_b));
    I2_col = label2rgb(bwlabel(I2_b));


    mkdir(dest_folder);

    imwrite(cropToSquare(I),  dest_folder + "/I.png");
    imwrite(cropToSquare(I1), dest_folder + "/I1.png");
    imwrite(cropToSquare(I2), dest_folder + "/I2.png");

    HistSave(I,  dest_folder + "/H.png")
    HistSave(I1, dest_folder + "/H1.png")
    HistSave(I2, dest_folder + "/H2.png")

    imwrite(cropToSquare(I_b),  dest_folder + "/I_b.png");
    imwrite(cropToSquare(I1_b), dest_folder + "/I1_b.png");
```

```
    imwrite(cropToSquare(I2_b), dest_folder + "/I2_b.png");

    imwrite(cropToSquare(I_col),  dest_folder + "/I_col.png");
    imwrite(cropToSquare(I1_col), dest_folder + "/I1_col.png");
    imwrite(cropToSquare(I2_col), dest_folder + "/I2_col.png");


        Latex = [
    "\newcommand{\ww}{0.24} "
    "\begin{figure}[H] "
    "   \captionsetup[subfloat]{justification=raggedright,
    singlelinecheck=false, position=bottom,labelformat=empty} % "
    "   "
    "    \subfloat[ Oryginalny ]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I.png}}  \hfill% "
    "    \subfloat[]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/H.png}}  \hfill% "
    "    \subfloat[ śćWarto progu $t$ = " + string(t) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I_b.png}}  \hfill%"
    "    \subfloat[ Liczba óobszarw " + string(n) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I_col.png}}  \hfill%"
    "   "
    "    \subfloat[ łóZakcony ]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I1.png}}  \hfill% "
    "    \subfloat[]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/H1.png}}  \hfill% "
    "    \subfloat[ śćWarto progu $t$ = " + string(t1) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I1_b.png}}  \hfill%"
    "    \subfloat[ Liczba óobszarw " + string(n1) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I1_col.png}}  \hfill%"
    "   "
    "    \subfloat[ Filtrowany ]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2.png}}  \hfill% "
    "    \subfloat[]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/H2.png}}  \hfill% "
    "    \subfloat[ śćWarto progu $t_2$ = " + string(t2) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2_b.png}}  \hfill%"
    "    \subfloat[ Liczba óobszarw " + string(n2) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2_col.png}}  \hfill%"
    ""
    " "
    "\caption{Porownanie}"
    ""
    "\end{figure} "
    "\let\ww\undefined "
```

```
        ];

        Latex = join(Latex,['']',2);
        Latex = join(Latex,[newline],1);

        fid = fopen(dest_folder + "/result.tex",'wt');
        fprintf(fid,"%s", Latex);
        fclose(fid);




end



function HistSave(I, dest)
    f = figure;
    f.Position = [0,0,512,512];
    imhist(I);
    saveas(gcf, dest)
    close(gcf);
end
```

zad4.m

```
clear all
close all


disp("Zad 4")

disp("   Img 1")
ProcessImages("../images/20191121_072040.jpg", "../zad4/Img1");

disp("   Img 2")
ProcessImages("../images/20191121_072227.jpg", "../zad4/Img2");



function ProcessImages(image_path, dest_folder)


    I = imread(image_path);
    I = rgb2gray(I);
    I = double(I) / 255;
    I = imresize(I, [512, 512]);

    [I_b, t] = ImageBinarize(I, "otsu");

    I_b = -I_b + 1;

    I_b_u8 = uint8(I_b*255);

    Ib1 = imerode(I_b_u8, offsetstrel('ball',5,5)) > 127;
    Ib2 = imerode(I_b_u8, offsetstrel('ball',10,10)) > 127;
    Ib3 = imdilate(I_b_u8, offsetstrel('ball',5,5)) > 127;
    Ib4 = imdilate(I_b_u8, offsetstrel('ball',10,10)) > 127;

    [~, n] = bwlabel(I_b_u8);
    [~, n1] = bwlabel(Ib1);
    [~, n2] = bwlabel(Ib2);
    [~, n3] = bwlabel(Ib3);
    [~, n4] = bwlabel(Ib4);

    I_col = label2rgb(bwlabel(I_b_u8));
    I1_col = label2rgb(bwlabel(Ib1));
    I2_col = label2rgb(bwlabel(Ib2));
    I3_col = label2rgb(bwlabel(Ib3));
    I4_col = label2rgb(bwlabel(Ib4));


    mkdir(dest_folder);

    imwrite(cropToSquare(I), dest_folder + "/I.png");

    imwrite(cropToSquare(I_b), dest_folder + "/I_b.png");
    imwrite(cropToSquare(Ib1), dest_folder + "/Ib1.png");
    imwrite(cropToSquare(Ib2), dest_folder + "/Ib2.png");
    imwrite(cropToSquare(Ib3), dest_folder + "/Ib3.png");
    imwrite(cropToSquare(Ib4), dest_folder + "/Ib4.png");
```

```
56
57      imwrite(cropToSquare(I_col), dest_folder + "/I_col.png");
58      imwrite(cropToSquare(I1_col), dest_folder + "/I1_col.png");
59      imwrite(cropToSquare(I2_col), dest_folder + "/I2_col.png");
60      imwrite(cropToSquare(I3_col), dest_folder + "/I3_col.png");
61      imwrite(cropToSquare(I4_col), dest_folder + "/I4_col.png");
62
63
64
65          Latex = [
66      "\newcommand{\ww}{0.19} "
67      "\begin{figure}[H] "
68      "   \captionsetup[subfloat]{justification=raggedright,
        singlelinecheck=false, position=bottom,labelformat=empty} % "
69      "   "
70      "    \subfloat[]{"
71      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/I.png}}  \hfill% "
72      "    \subfloat[]{"
73      "        \includegraphics[width=\ww\linewidth]{../other/Empty.
        png}}  \hfill% "
74      "    \subfloat[]{"
75      "        \includegraphics[width=\ww\linewidth]{../other/Empty.
        png}}  \hfill%"
76          "    \subfloat[]{"
77      "        \includegraphics[width=\ww\linewidth]{../other/Empty.
        png}}  \hfill% "
78      "    \subfloat[]{"
79      "        \includegraphics[width=\ww\linewidth]{../other/Empty.
        png}}  \hfill%"
80      "   "
81      "    \subfloat[śćWarto progu t = " + string(n) + "]{"
82      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/I_b.png}}  \hfill% "
83      "    \subfloat[   s = offsetstrel('ball',5,5); \\ imerode(I, s)
        ;    ]{"
84      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/Ib1.png}}  \hfill% "
85      "    \subfloat[   s = offsetstrel('ball',10,10); \\ imerode(I,
        s);    ]{"
86      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/Ib2.png}}  \hfill%"
87      "    \subfloat[   s = offsetstrel('ball',5,5); \\ imdilate(I, s
        );    ]{"
88      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/Ib3.png}}  \hfill% "
89      "    \subfloat[   s = offsetstrel('ball',10,10); \\ imdilate(I,
         s);    ]{"
90      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/Ib4.png}}  \hfill%"
91      "   "
92      "    \subfloat[Liczba óobszarw = " + string(n) + "]{"
93      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/I_col.png}}  \hfill% "
94      "    \subfloat[Liczba óobszarw = " + string(n1) + "]{"
95      "        \includegraphics[width=\ww\linewidth]{" + dest_folder
        + "/I1_col.png}}  \hfill% "
```

```
    "    \subfloat[Liczba óobszarw = " + string(n2) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I2_col.png}}  \hfill%"
    "    \subfloat[Liczba óobszarw = " + string(n3) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I3_col.png}}  \hfill% "
    "    \subfloat[Liczba óobszarw = " + string(n4) + "]{"
    "        \includegraphics[width=\ww\linewidth]{" + dest_folder
    + "/I4_col.png}}  \hfill%"
    ""
    " "
    "\caption{Porownanie}"
    ""
    "\end{figure} "
    "\let\ww\undefined "
        ];

        Latex = join(Latex,[''],2);
        Latex = join(Latex,[newline],1);

        fid = fopen(dest_folder + "/result.tex",'wt');
        fprintf(fid,"%s", Latex);
        fclose(fid);




end



function HistSave(I, dest)
    f = figure;
    f.Position = [0,0,512,512];
    imhist(I);
    saveas(gcf, dest)
    close(gcf);
end
```

ImageBinarize.m

```
function [I1, level] = ImageBinarize(I, method)

    if method == "hist"
        level = HistogramHalf(I);
    elseif method == "otsu"
        level = OtsuTreshhold(I);
    end

    I1 = double( I > level );
end



function level = HistogramHalf(I)

    [counts,binLocations] = imhist(I);
    count_and_bin = [counts, binLocations]';

    all = numel(I);
    half = all/2;

    accumulator = 0;

    for cb = count_and_bin
        accumulator = accumulator + cb(1);

        if(accumulator >= half)
            level = cb(2);
            return;
        end
    end

    % function should never reach this
    level = 1;

end



function level = OtsuTreshhold(I)

    N = numel(I);

    [n_all, k_all] = imhist(I);


    p_all = n_all/N;


    selected_k = 0;
    max_sigma2_B = 0;

    for i = 1:numel(k_all)-1
```

```
56
57          k = k_all(i);
58
59          I_L = [1:i]';
60          I_H = [i+1:numel(p_all)]';
61
62          w0 = sum( p_all(I_L) );
63          w1 = sum( p_all(I_H) );
64
65          u0 = sum( I_L .* p_all(I_L) / w0 );
66          u1 = sum( I_H .* p_all(I_H) / w1 );
67
68          sigma2_B = w0*w1*(u1-u0)^2;
69
70          if( max_sigma2_B <  sigma2_B)
71              max_sigma2_B = sigma2_B;
72              selected_k = k;
73          end
74
75      end
76
77  level = selected_k;
78
79
80  end
```

run_all.m

```
clear all
close all
clc

zad2
zad3
zad4
```

cropToSquare.m

```
function I1 = cropToSquare(I)

    s1 = size(I,1);
    s2 = size(I,2);


    if s1 == s2
        I1 = I;

    elseif( s1 > s2 )

        diff2 = round((s1 - s2)/2);
        v1 = (1:s2) + diff2;
        v2 = 1:s2;

        I1 = I(v1,v2, :);

    elseif( s1 < s2 )

        diff2 = round((s2 - s1)/2);
        v1 = 1:s1;
        v2 = (1:s1) + diff2;

        I1 = I(v1,v2, :);

    end


end
```