

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Główny skrypt%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%% Przygotowanie banknotow i wzorców
```

```
banknot_we = imread('banknot_200_3.jpg'); %Wczytanie banknotu  
[m,n] = size(banknot_we); %Wyznaczenie rozmiaru obrazka
```

```
wzor_10 = imread('banknot_10.jpg' ); %Wczytanie wzorów  
wzor_20 = imread('banknot_20.jpg' ); %  
wzor_50 = imread('banknot_50.jpg' ); %  
wzor_100 = imread('banknot_100.jpg'); %  
wzor_200 = imread('banknot_200.jpg'); %
```

```
wzor_10 = imresize(wzor_10 , [m,n/3]); %Przeskalowanie wzorów, aby  
pasowały do banknotu  
wzor_20 = imresize(wzor_20 , [m,n/3]); %  
wzor_50 = imresize(wzor_50 , [m,n/3]); %  
wzor_100 = imresize(wzor_100, [m,n/3]); %  
wzor_200 = imresize(wzor_200, [m,n/3]); %
```

```
[banknot_we_R,banknot_we_G,banknot_we_B]=Przygotuj(banknot_we); %Wywołanie  
funkcji Przygotuj dla banknotu oraz wzorów  
[wzor_10_R,wzor_10_G,wzor_10_B]=Przygotuj(wzor_10); %  
[wzor_20_R,wzor_20_G,wzor_20_B]=Przygotuj(wzor_20); %  
[wzor_50_R,wzor_50_G,wzor_50_B]=Przygotuj(wzor_50); %  
[wzor_100_R,wzor_100_G,wzor_100_B]=Przygotuj(wzor_100); %  
[wzor_200_R,wzor_200_G,wzor_200_B]=Przygotuj(wzor_200); %
```

```
%% Odejmowanie wzorca 10 od banknotu w maskach RGB
```

```
outR=zeros(m,n/3);  
outG=zeros(m,n/3);  
outB=zeros(m,n/3);  
for i=1:m  
    for k=1:(n/3)  
        outR(i,k) = banknot_we_R(i,k) - wzor_10_R(i,k);  
        outG(i,k) = banknot_we_G(i,k) - wzor_10_G(i,k);  
        outB(i,k) = banknot_we_B(i,k) - wzor_10_B(i,k);  
    end  
end
```

```
out = sqrt( outR.^2 + outG.^2 + outB.^2); %Wyliczanie odległości  
euklidesowej  
val_10 = mean(mean(out)); %
```

```
%% Odejmowanie wzorca 20 od banknotu w maskach RGB
```

```
outR=zeros(m,n/3);  
outG=zeros(m,n/3);  
outB=zeros(m,n/3);  
for i=1:m  
    for k=1:(n/3)  
        outR(i,k) = banknot_we_R(i,k) - wzor_20_R(i,k);  
        outG(i,k) = banknot_we_G(i,k) - wzor_20_G(i,k);  
        outB(i,k) = banknot_we_B(i,k) - wzor_20_B(i,k);  
    end  
end
```

```
end
```

```
out = sqrt( outR.^2 + outG.^2 + outB.^2);  
val_20 = mean(mean(out));
```

```
%% Odejmowanie wzorca 50 od banknotu w maskach RGB
```

```
outR=zeros(m,n/3);  
outG=zeros(m,n/3);  
outB=zeros(m,n/3);  
for i=1:m  
    for k=1:(n/3)  
        outR(i,k) = banknot_we_R(i,k) - wzor_50_R(i,k);  
        outG(i,k) = banknot_we_G(i,k) - wzor_50_G(i,k);  
        outB(i,k) = banknot_we_B(i,k) - wzor_50_B(i,k);  
    end  
end
```

```
out = sqrt( outR.^2 + outG.^2 + outB.^2);  
val_50 = mean(mean(out));
```

```
%% Odejmowanie wzorca 100 od banknotu w maskach RGB
```

```
outR=zeros(m,n/3);  
outG=zeros(m,n/3);  
outB=zeros(m,n/3);  
for i=1:m  
    for k=1:(n/3)  
        outR(i,k) = banknot_we_R(i,k) - wzor_100_R(i,k);  
        outG(i,k) = banknot_we_G(i,k) - wzor_100_G(i,k);  
        outB(i,k) = banknot_we_B(i,k) - wzor_100_B(i,k);  
    end  
end
```

```
out = sqrt( outR.^2 + outG.^2 + outB.^2);  
val_100 = mean(mean(out));
```

```
%% Odejmowanie wzorca 200 od banknotu w maskach RGB
```

```
outR=zeros(m,n/3);  
outG=zeros(m,n/3);  
outB=zeros(m,n/3);  
for i=1:m  
    for k=1:(n/3)  
        outR(i,k) = banknot_we_R(i,k) - wzor_200_R(i,k);  
        outG(i,k) = banknot_we_G(i,k) - wzor_200_G(i,k);  
        outB(i,k) = banknot_we_B(i,k) - wzor_200_B(i,k);  
    end  
end
```

```
out = sqrt( outR.^2 + outG.^2 + outB.^2);  
val_200 = mean(mean(out));
```

```
%%
```

```
% ROZPOZNANIE
```

```

value = min([val_10 val_20 val_50 val_100 val_200]);           %Wybieranie
najmniejszej odległości euklidesowej

if value==val_10
    fprintf('Rozpoznano banknot 10zł\n');
elseif value==val_20
    fprintf('Rozpoznano banknot 20zł\n');
elseif value==val_50
    fprintf('Rozpoznano banknot 50zł\n');
elseif value==val_100
    fprintf('Rozpoznano banknot 100zł\n');
elseif value==val_200
    fprintf('Rozpoznano banknot 200zł\n');
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Funkcja Przygotuj%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [ y_R,y_G,y_B ] = Przygotuj( x )
%Przygotuj - przygotowuje banknot
%Argumenty
%   x- obrazek, który chcemy przygotować
%   y_R, y_G, y_B - poszczególne maski obrazka wejściowego
%               po odjęciu średniej oraz skalowaniu jasności

[y_R,y_G,y_B]=PodzialNaMaski(x);           %wywołanie funkcji PodzialNaMaski

y_R=OdejmijSrednia(y_R);                   %wywołanie funkcji OdejmijSrednia na
poszczególnych maskach
y_G=OdejmijSrednia(y_G);                   %
y_B=OdejmijSrednia(y_B);                   %

y_R=SkalowanieJasnosci(y_R);               %wywołanie funkcji SkalowanieJasnosci na
poszczególnych maskach
y_G=SkalowanieJasnosci(y_G);               %
y_B=SkalowanieJasnosci(y_B);               %

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Funkcja PodzialNaMaski%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [ y_R,y_G,y_B ] = PodzialNaMaski( x )
%Argumenty wejściowe
% x-obrazek do wyodrebnienia masek
%Argumenty wyjściowe
% y_r, y_G, y_B - poszczególne maski

y_R=x(:, :, 1);
y_G=x(:, :, 2);
y_B=x(:, :, 3);

end

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Funkcja OdejmijSrednia%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function [ y ] = OdejmijSrednia( x )
```

```
%Argumenty wejściowe
```

```
% x-obrazek
```

```
%Argumenty wyjściowe
```

```
% y - obrazek po odjęciu sredniej
```

```
x_avg=mean(mean(x));
```

```
y=double(x)-x_avg;
```

```
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Funkcja SkalujJasnosc%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function [ y ] = SkalowanieJasnosci( x )
```

```
%Argumenty wejściowe
```

```
% x-obrazek
```

```
%Argumenty wyjściowe
```

```
% y - obrazek po wyskalowaniu jasnosci, czyli odjęciu wartości maksymalnej
```

```
x_max=max(max(x));
```

```
y=x./x_max;
```

```
end
```