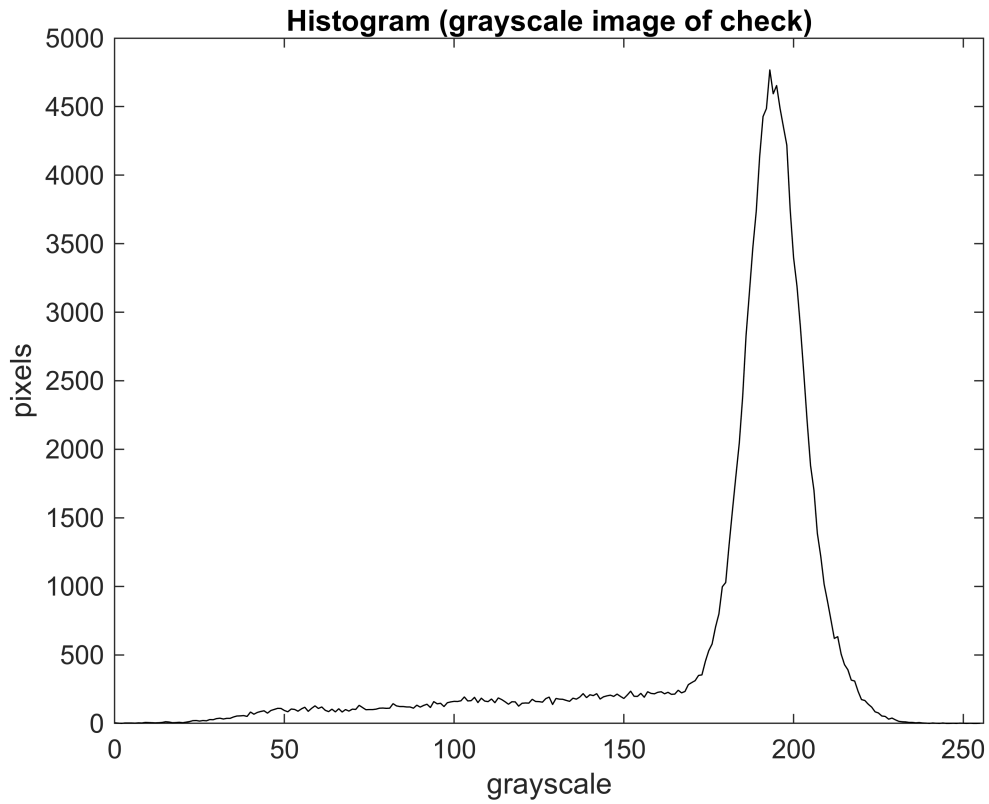
Activity 5: Feature Extraction Part 1 of 3: Image Segmentation

Exercise: Picking out handwritten text in grayscale image of a check

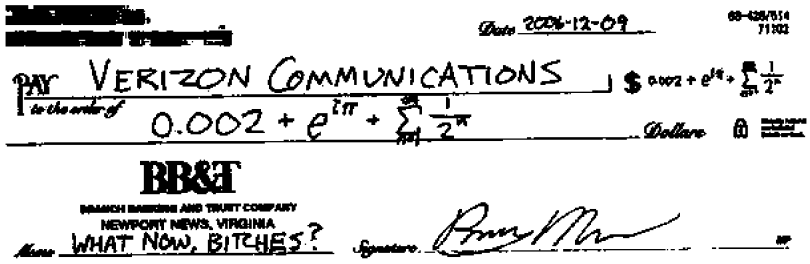
```
clear;close;
I = imread('cropped_grayscale_check.jpg');
[count, cells] = imhist(I, 256);
plot(cells, count, 'color', 'black'); ylabel('pixels'); xlabel('grayscale'); xlim([0 256]);
%hold on; xline(1, 'color', 'red'); xline(160,'color', 'red');
title('Histogram (grayscale image of check)');
```



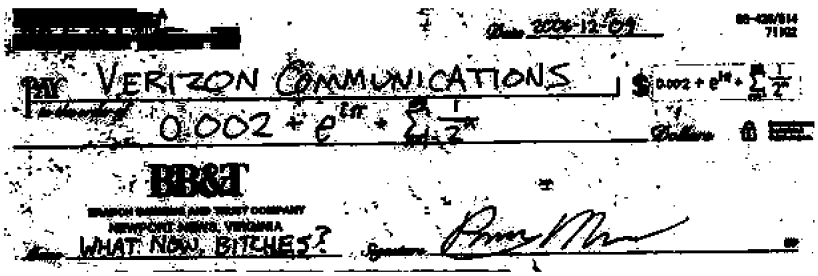
Note: The large peak in the historgam corresponds to the background pixels.

Threshold: 100

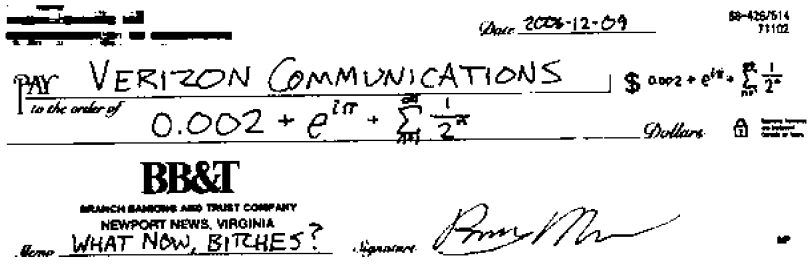
```
clear;close;
check(250, 160);
```



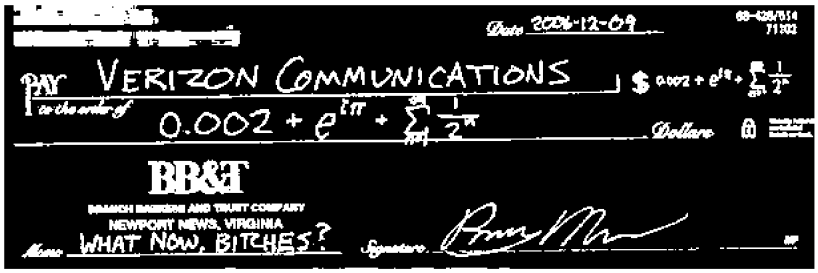

```
clear;close;
check(250, 180);
```



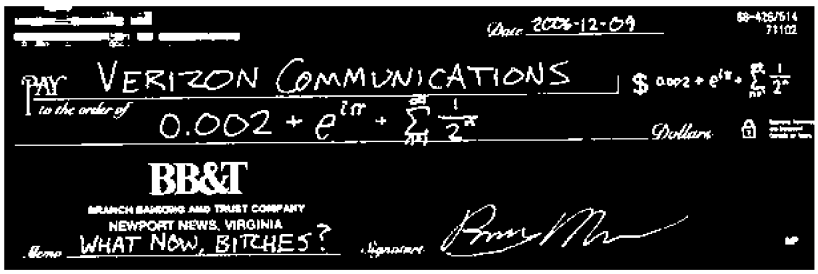

```
clear;close;
check(250, 130);
```



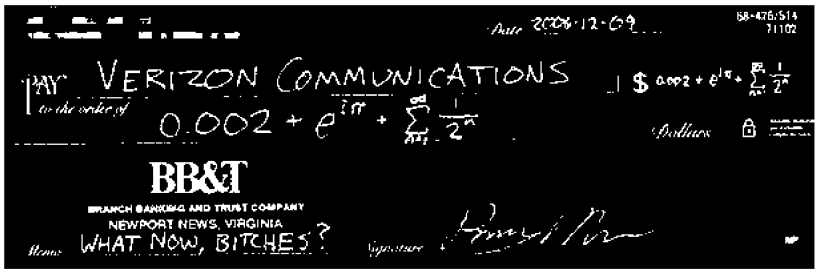

```
clear;close;
check(160,0);
```




```
clear;close;
check(130,1);
```



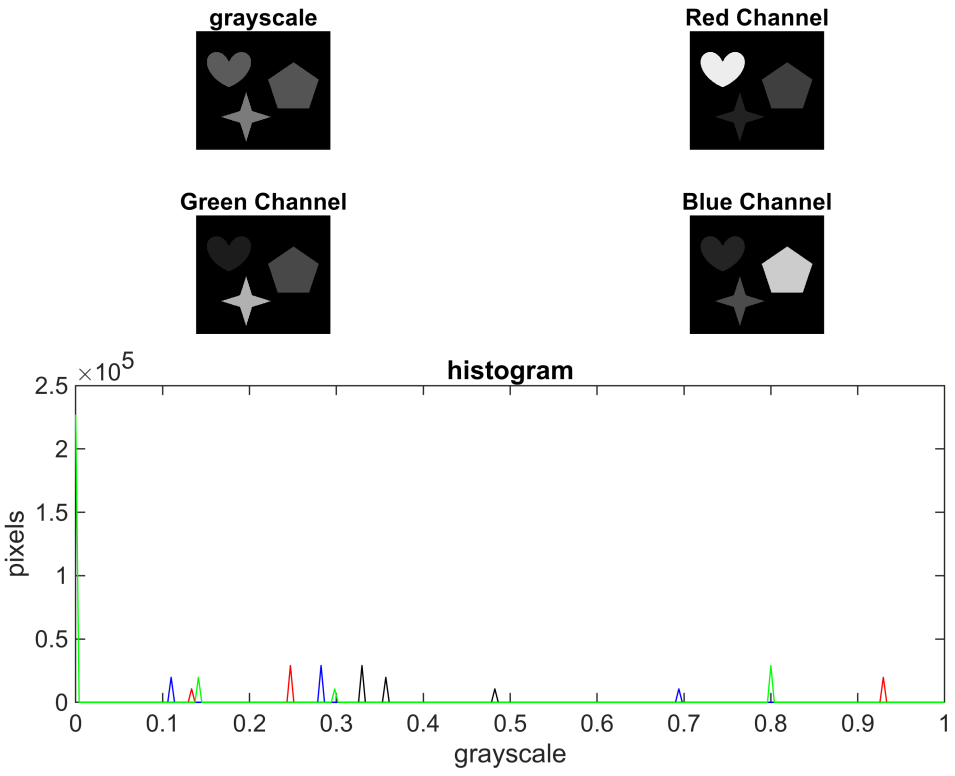

```
clear;close;
check(100,1);
```



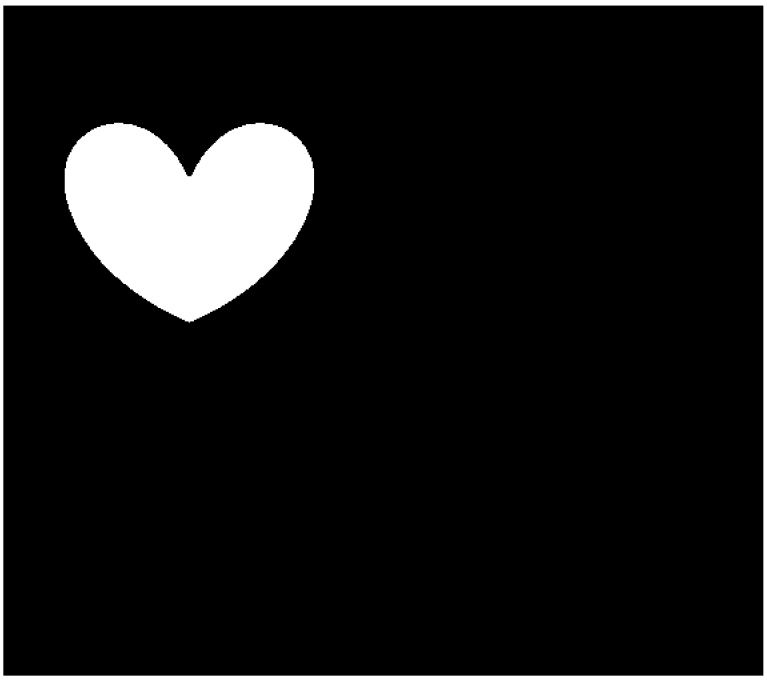
Thresholding in RGB

```
clear; close;
```

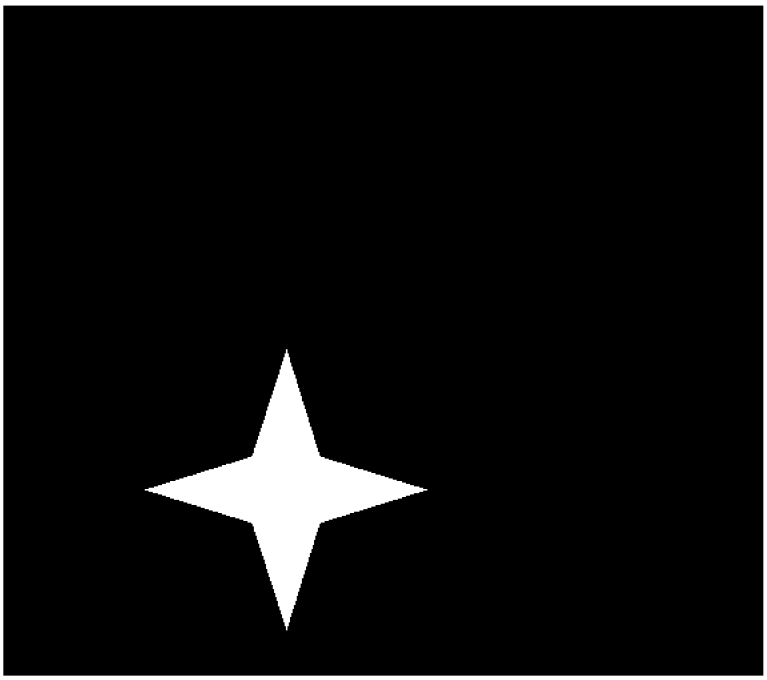
```
I = im2double(imread('rgb.png'));
[count, cells] = imhist(im2gray(I), 256);
R = I(:,:,1);
[count1, cells1] = imhist(R, 256);
G = I(:,:,2);
[count2, cells2] = imhist(G, 256);
B = I(:,:,3);
[count3, cells3] = imhist(B, 256);
subplot(4,2,1); imshow(im2gray(I)); title('grayscale');
subplot(4,2,2); imshow(R); title('Red Channel');
subplot(4,2,3); imshow(G); title('Green Channel');
subplot(4,2,4); imshow(B); title('Blue Channel');
subplot(4,2,[5 6 7 8]);plot(cells, count, 'color', 'black'); ylabel('pixels'); xlabel('grayscal
hold on;
plot(cells1, count1, 'color', 'red');
plot(cells2, count2, 'color', 'blue');
plot(cells3, count3, 'color', 'green');
title('histogram')
```



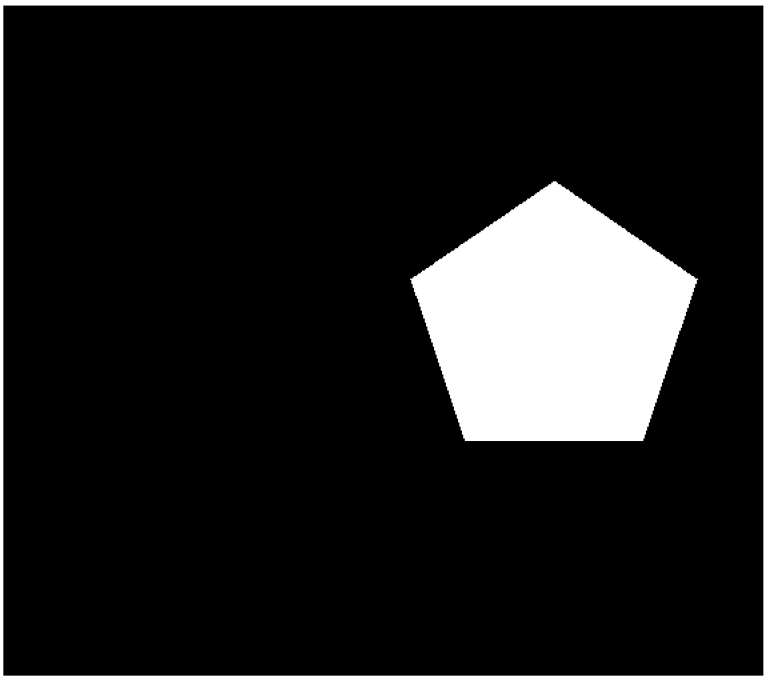

```
clear;close;
threshR(5,0.5, 'rgb.png');
```



```
clear;close;
threshG(5,0.5, 'rgb.png');
```

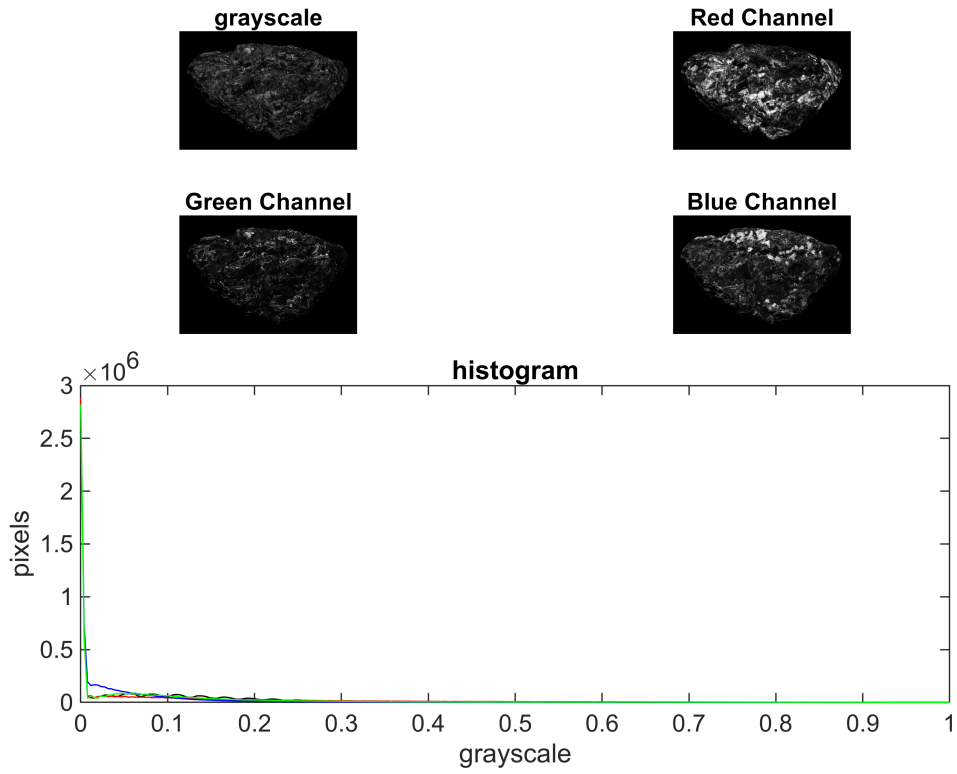


```
clear;close;
threshB(5,0.5, 'rgb.png');
```

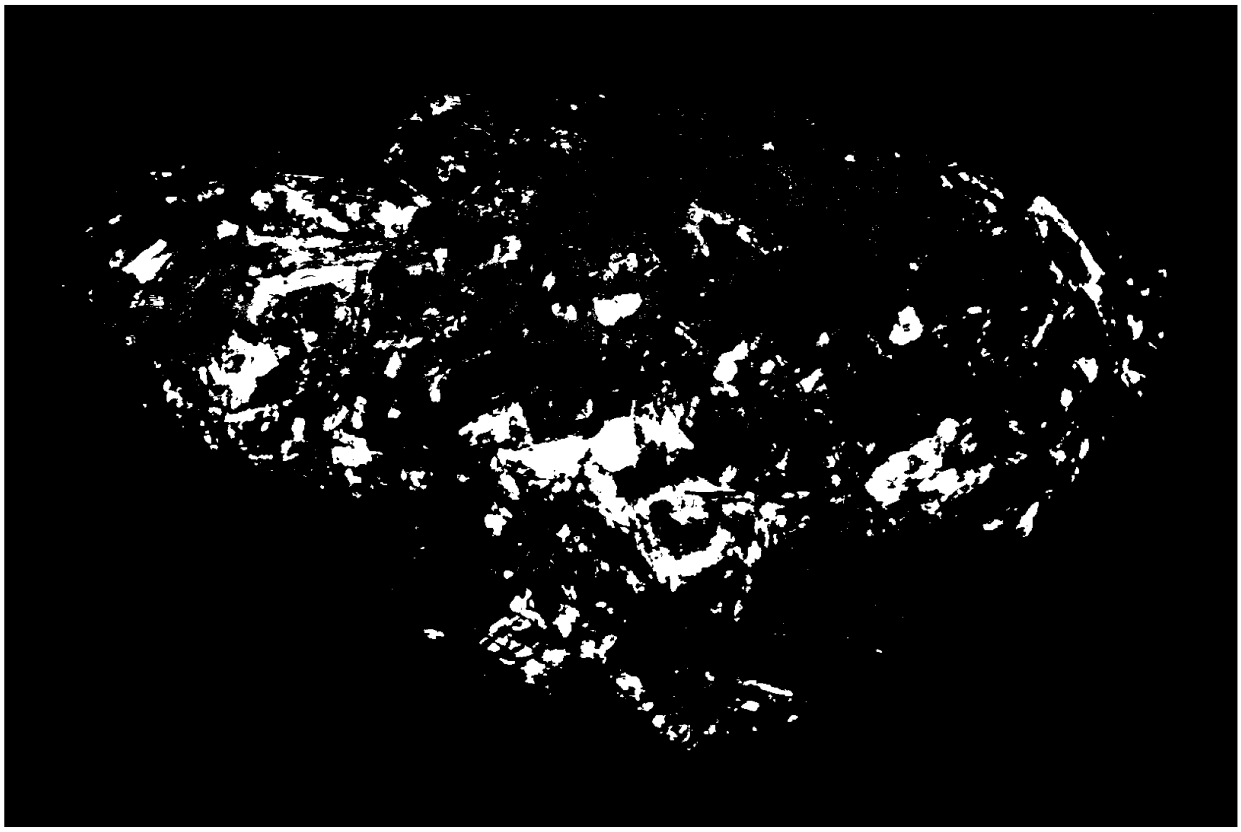


ANOTHER ONE RGB THRESHOLDING

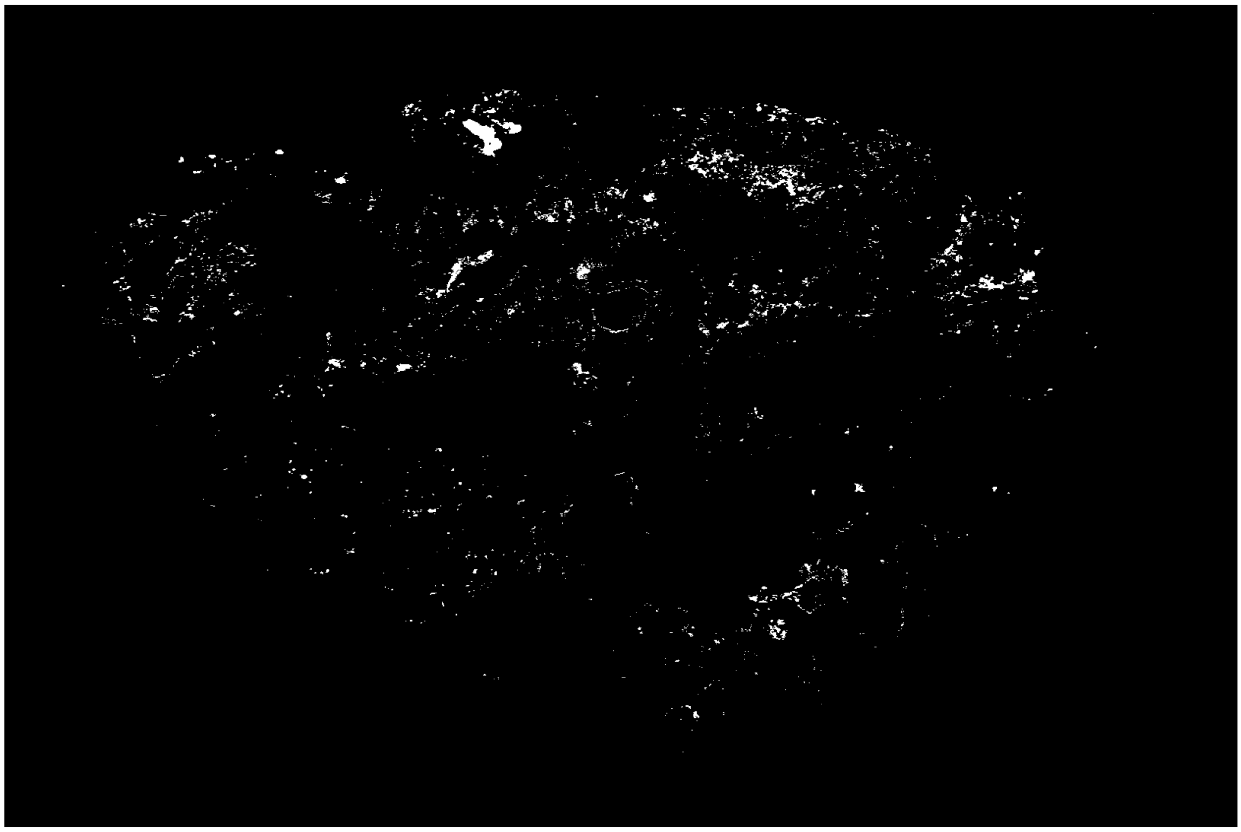
```
clear; close;
I = im2double(imread('rock.png'));
[count, cells] = imhist(im2gray(I), 256);
R = I(:,:,1);
[count1, cells1] = imhist(R, 256);
G = I(:,:,2);
[count2, cells2] = imhist(G, 256);
B = I(:,:,3);
[count3, cells3] = imhist(B, 256);
subplot(4,2,1); imshow(im2gray(I)); title('grayscale');
subplot(4,2,2); imshow(R); title('Red Channel');
subplot(4,2,3); imshow(G); title('Green Channel');
subplot(4,2,4); imshow(B); title('Blue Channel');
subplot(4,2,[5 6 7 8]);plot(cells, count, 'color', 'black'); ylabel('pixels'); xlabel('grayscal
hold on;
plot(cells1, count1, 'color', 'red');
plot(cells2, count2, 'color', 'blue');
plot(cells3, count3, 'color', 'green');
title('histogram')
```



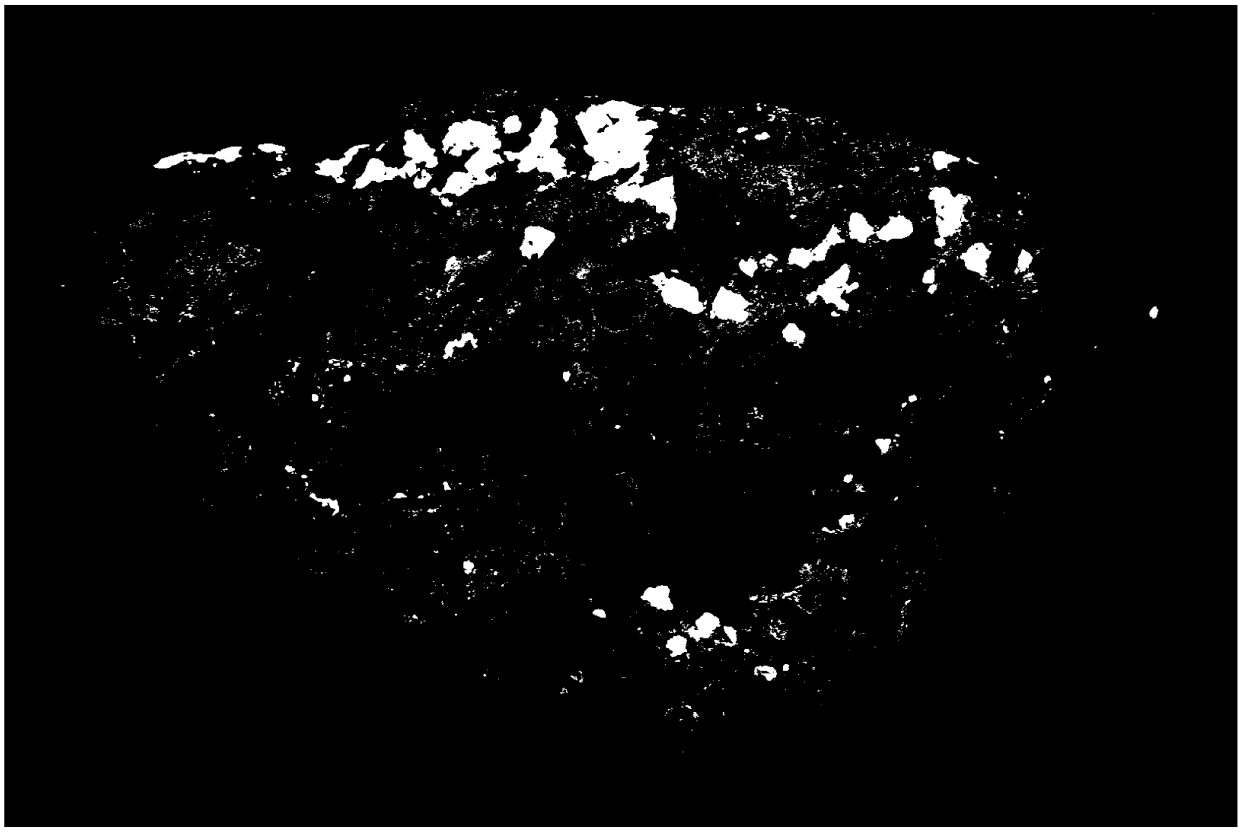

```
clear;close;
threshR(7,0.5, 'rock.png');
```



```
clear;close;
threshG(7,0.5, 'rock.png');
```



```
clear;close;
threshB(7,0.5, 'rock.png');
```

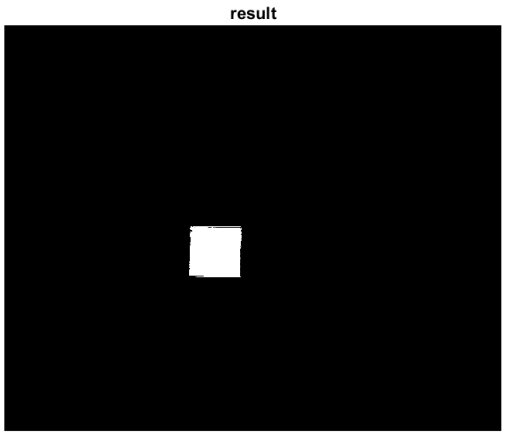


2D HISTOGRAM

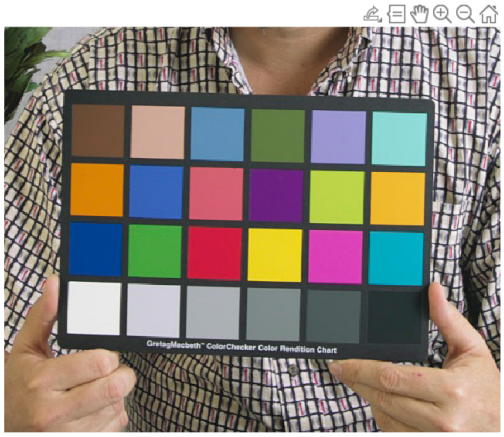
```
clear; close;
hist('Gretag-Macbeth_ColorChecker.jpg');
```

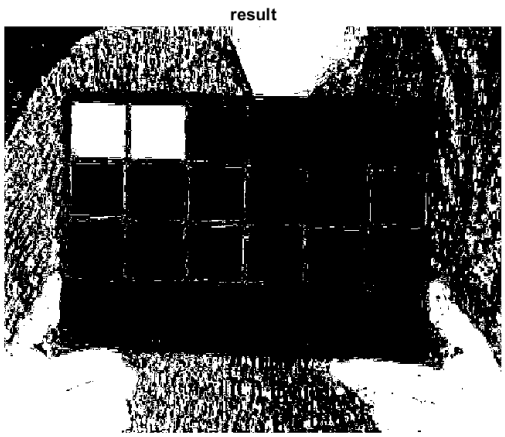
START

```
clear;close;
Param('Gretag-Macbeth_ColorChecker.jpg');
```



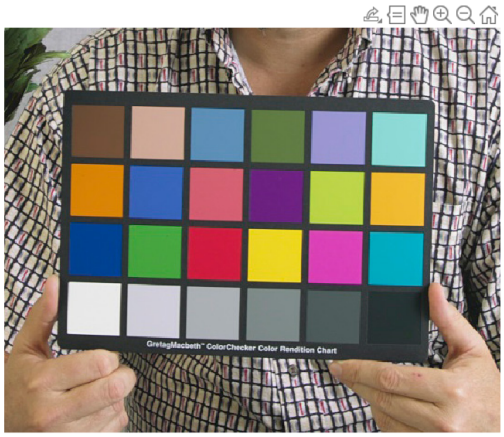

```
clear; close;
Nonparam(40, 'Gretag-Macbeth_ColorChecker.jpg');
```

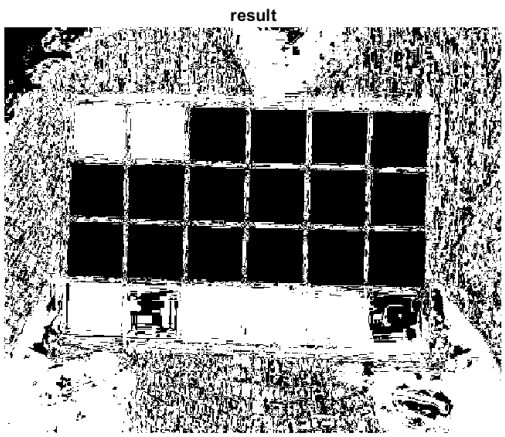




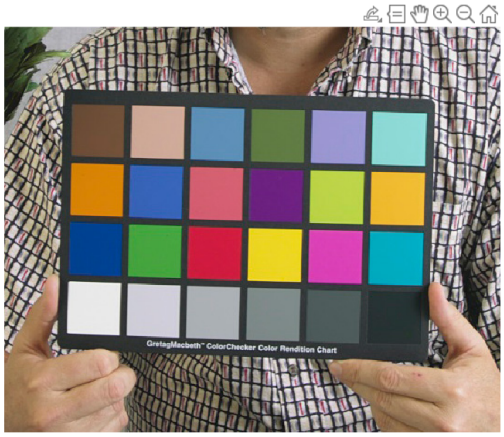
Different bin sizes

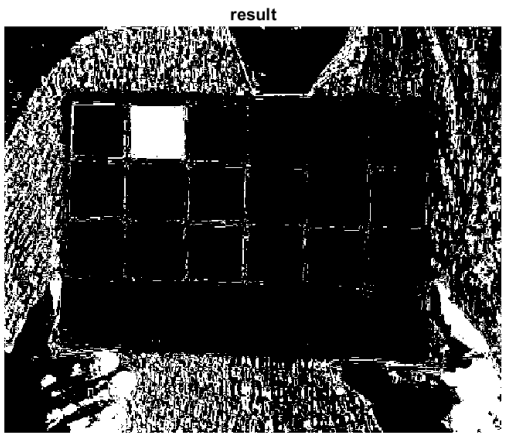
```
clear; close;
Nonparam(45, 'Gretag-Macbeth_ColorChecker.jpg');
```



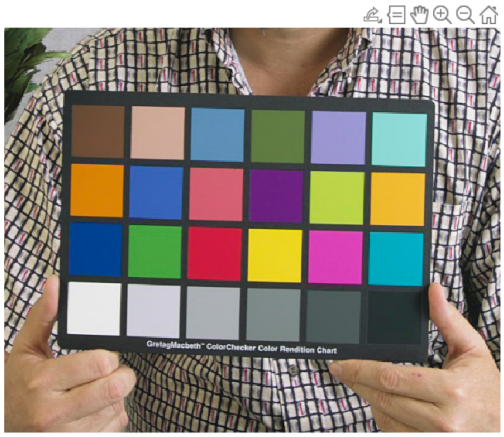


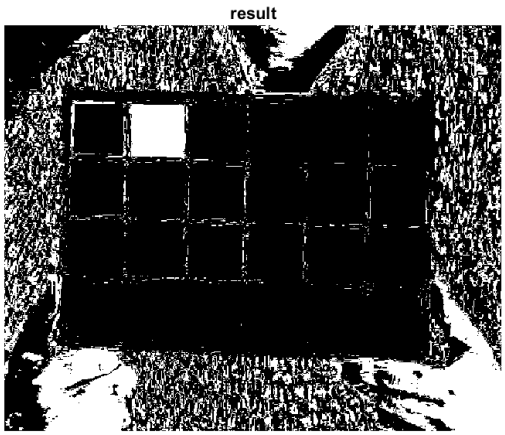

```
clear; close;
Nonparam(40, 'Gretag-Macbeth_ColorChecker.jpg');
```



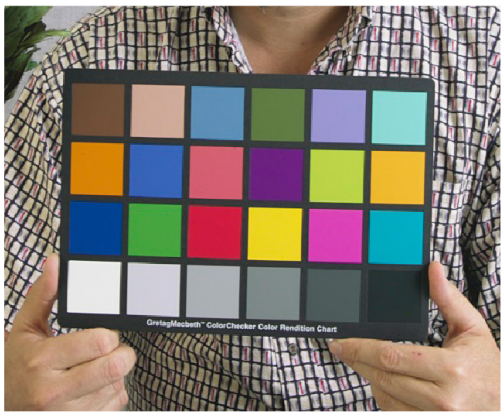


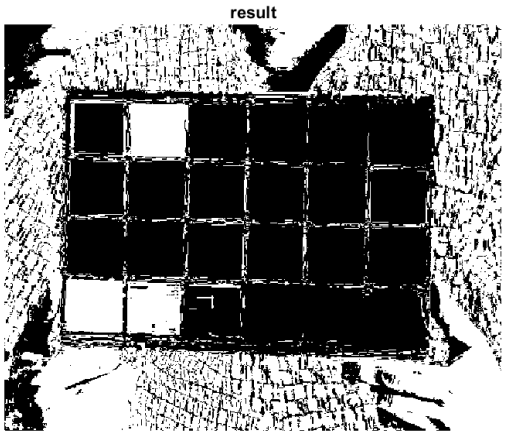

```
clear; close;
Nonparam(35, 'Gretag-Macbeth_ColorChecker.jpg');
```





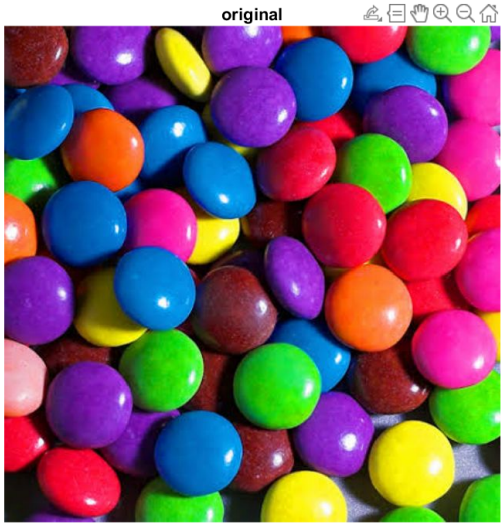

```
clear; close;
Nonparam(30, 'Gretag-Macbeth_ColorChecker.jpg');
```

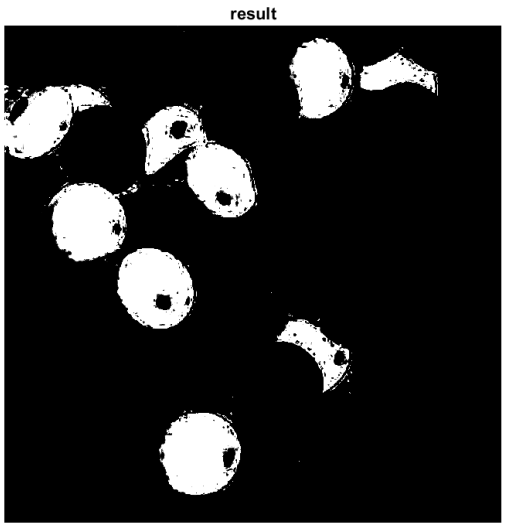




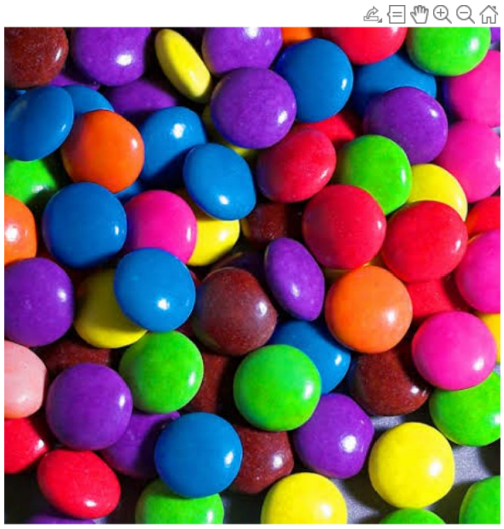
NIPS

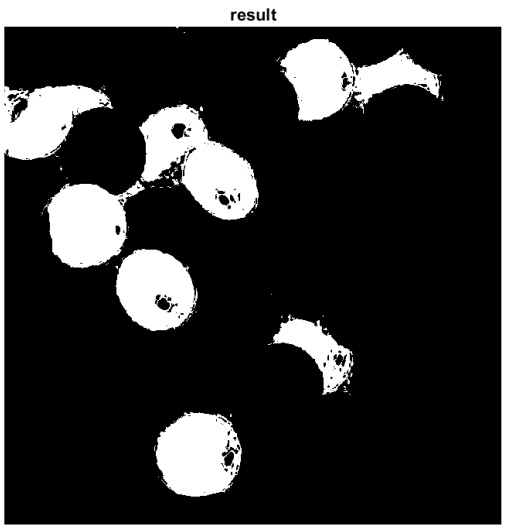
```
clear;close;
Param('nips.jpg');
```





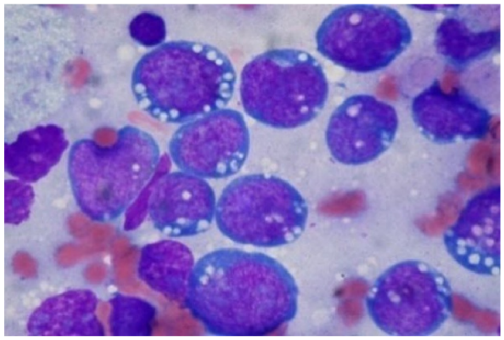

```
clear;close;
Nonparam(30, 'nips.jpg');
```

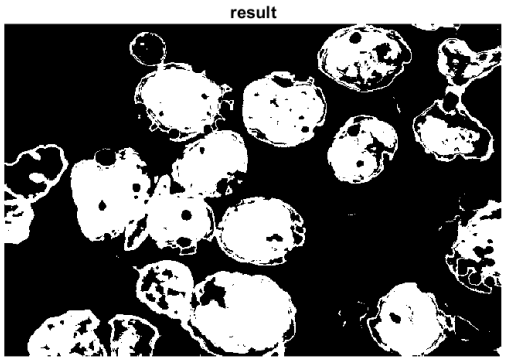




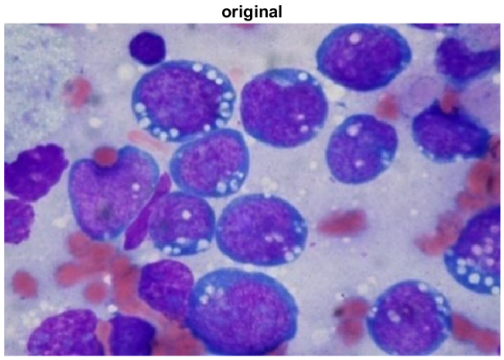
Application 01: Microscopy

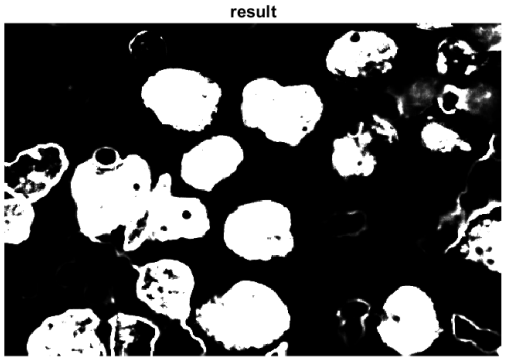
```
clear;close;
Nonparam(30, 'stained.jpg');
```






```
clear;close;
Param('stained.jpg');
```

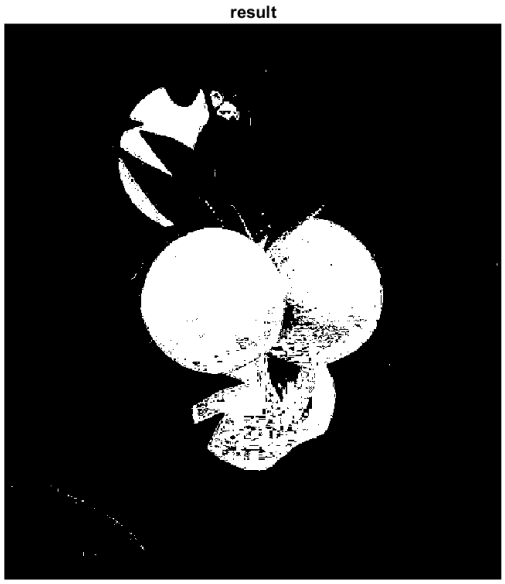




Application 02: Object Recognition

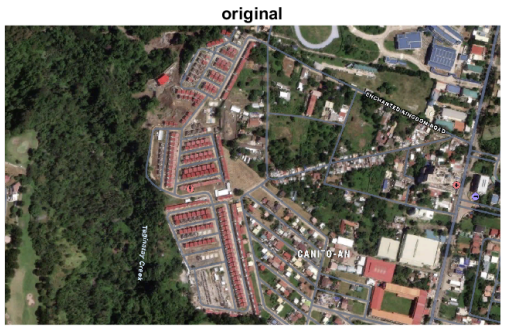
```
clear; close;
Param('basket.jpg');
```

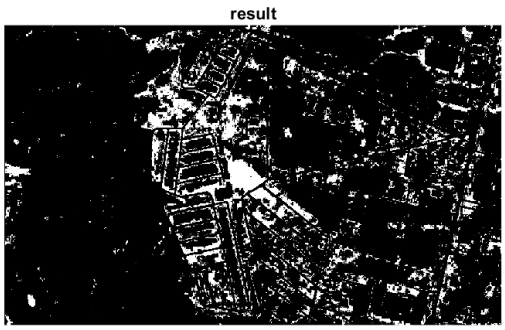

```
clear;close;
Nonparam(40, 'basket.jpg');
```



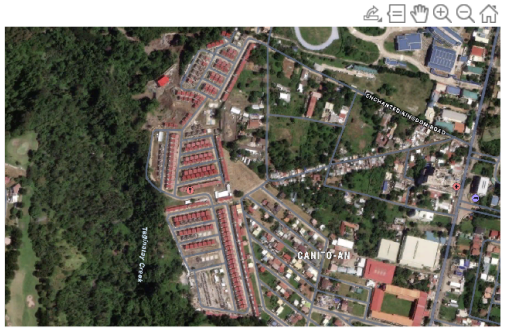
Application 03: Remote Sensing

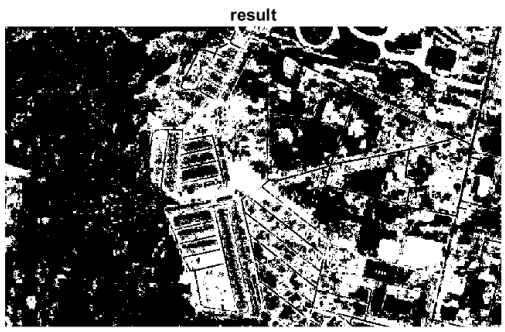
```
clear; close;
Param('map.png');
```






```
clear; close;
Nonparam(40 ,'map.png');
```






```
function [a,b] = hist(image)
    bins = 32;
    J = imread(image);
```

```
    go_on = 'y';
    while go_on == 'y'

        a = subplot(1,2,1);
        I = imcrop(J); %
        I = double(I); %I is the image of the region of interest
        R = I(:,:,1); G = I(:,:,2); B = I(:,:,3);
        Int = R + G + B;
        Int(Int == 0) = 1e5; %to prevent NaNs
        r = R./Int; g=G./Int;
        rint = round( r*(bins-1) + 1);
        gint = round( g*(bins-1) + 1);
        colors = gint(:) +  (rint(:)-1)*bins;
        hist = zeros(bins,bins);
        for row = 1:bins
            for col = 1:(bins-row+1)
                hist(row,col) = length(find(colors == (((col + (row-1)*bins)))));
            end
        end
        b = subplot(1,2,2);
        imagesc(imrotate((hist),90)); axis image; xlabel('r'); ylabel('g');
        colormap jet;

        go_on = input("Again? y/n", 's');
        if go_on == 'n'
            break
        end
    end
end

% thresholding in grayscale
function a = check(threshold1, threshold2)
    I = imread('cropped_grayscale_check.jpg');
    BW = and(I < threshold1, I > threshold2);
    a = imshow(BW);
end


% Thresholding in R, G, B
function a = threshR(threshold1, threshold2, image)
    I = im2double(imread(image));
    R = I(:,:,1);
    BW = and(R < threshold1, R > threshold2);
    a = imshow(BW);
end
function a = threshG(threshold1, threshold2, image)
    I = im2double(imread(image));
    G = I(:,:,2);
    BW = and(G < threshold1, G > threshold2);
    a = imshow(BW);
```

```
end
function a = threshB(threshold1, threshold2, image)
    I = im2double(imread(image));
    B = I(:,:,3);
    BW = and(B < threshold1, B > threshold2);
    a = imshow(BW);
end
%
function [a,b] = Param(image)
    I = im2double(imread(image));
    origR = double(I(:,:,1));
    origG = double(I(:,:,2));
    origB = double(I(:,:,3));
    [NCC_orig_r, NCC_orig_g] = NCC(origR,origG,origB);

    a = subplot(1,2,1); C = im2double(imcrop(I)); title('original');
    CR = double(C(:,:,1));
    CG = double(C(:,:,2));
    CB = double(C(:,:,3));

    [NCC_C_r, NCC_C_g] = NCC(CR,CG,CB);

    subplot(1,2,2);
    b = imshow(PDF(NCC_orig_r, NCC_C_r, NCC_orig_g, NCC_C_g)); title('result');
end
%joint PDF function for Parametric
function jointpdf = PDF(imager,croppedr, imageg, croppedg)
    mean_r = mean2(croppedr);
    mean_g = mean2(croppedg);
    std_r = std2(croppedr);
    std_g = std2(croppedg);
    pdf_r = (1/(std_r*sqrt(2*pi))) * (exp(-(((imager-mean_r).^2)/(2*(std_r).^2))));
    pdf_g = (1/(std_g*sqrt(2*pi))) * (exp(-(((imageg-mean_g).^2)/(2*(std_g).^2))));
    jointpdf = pdf_r .* pdf_g;
end

%NCC color space
function [r,g] = NCC(R, G, B)
    %per pixel
    I = R + G + B;

    I(I==0) = 1e5; %to avoid NANs

    r = R./I;
    g = G./I;
end

%
function [a,b] = Nonparam(BINS, image)
    % BINS = 40;
```

```
    % [filename,pathname] = uigetfile('nips.jpg');
    % J = imread([pathname,filename]);
    J = im2double(imread(image));
    a = subplot(1,2,1); imshow(J); title('original');
    [I, rect] = imcrop(J);
    %% Get the r g of the whole image
    R = J(:,:,1); G = J(:,:,2); B = J(:,:,3);

    [rJ, gJ] = NCC(R,G,B);

    %% Crop the region of interest in the rg space

    r = imcrop(rJ, rect);
    g = imcrop(gJ, rect);
    rint = round( r*(BINS-1) + 1);
    gint = round (g*(BINS-1) + 1);
    colors = gint(:) + (rint(:)-1)*BINS;

    %% Compute rg-histogram
    % This is the 1-d version of a 2-d histogram
    hist = zeros(BINS*BINS,1);
    for row = 1:BINS
        for col = 1:(BINS-row+1)
            hist(col+(row-1)*BINS) = length( find(colors==( ((col + (row-1)*BINS)))));
        end
    end
    %% Backproject histogram
    rJint = round( rJ*(BINS-1) + 1);
    gJint = round (gJ*(BINS-1) + 1);
    colorsJ = gJint(:) + (rJint(:)-1)*BINS;
    HB = hist(colorsJ);
    HBImage = reshape(HB,size(J,[1,2]));

    b = subplot(1,2,2); imshow(HBImage); axis image; axis off; title('result');
end
```