**Đinh Hoàng Sáng**

**BEBEIU17022**

**HomeWork9**

Ex1:

Code:

clc

close all

clear all

%auto clear, ez to test

rgb = imread('coloredChips.png');

figure,imshow(rgb), impixelinfo

rgb\_h = rgb2hsv(rgb);

figure,imshow(rgb\_h)

red = rgb\_h(:,:,1);

s = rgb\_h(:,:,2);

v = rgb\_h(:,:,3);

%change some chanel to find out differences between the color which you need dectect and other color.

%So, I saw the difference at rgb\_h(:,:,1); figure(3)

%Red color had index>0.9, other color have index <0.9;

figure,imshow(red)

figure,imshow(s)

figure,imshow(v)

[r c] =size(red);

%scan all pixel of figure(3)

for i = 1:r

for j= 1:c

if red(i,j) >= 0.9

out(i,j) = 255; %index>0.9 replaced by white

else

out(i,j) = 0; %other replaced by black

end

end

end

figure,imshow(out)

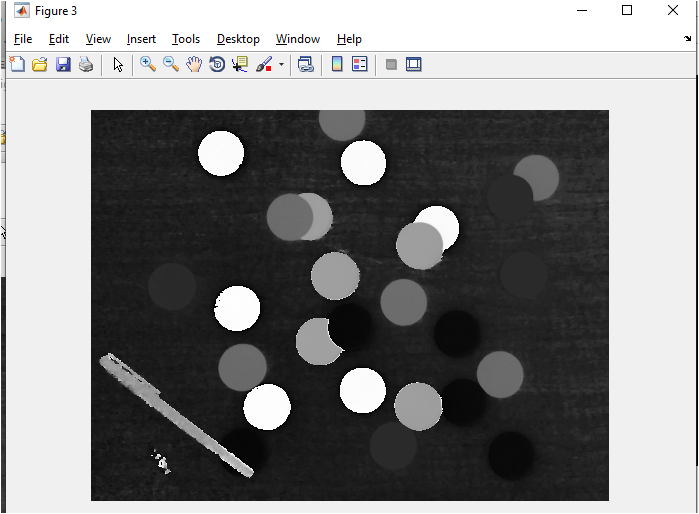
%out = bwareaopen(out,100); %remove noises if u need, I didn’t need at my case.

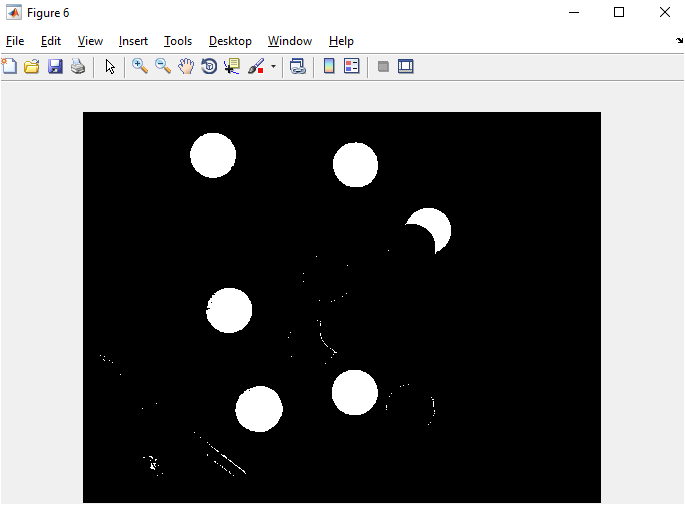
figure,imshow(rgb)

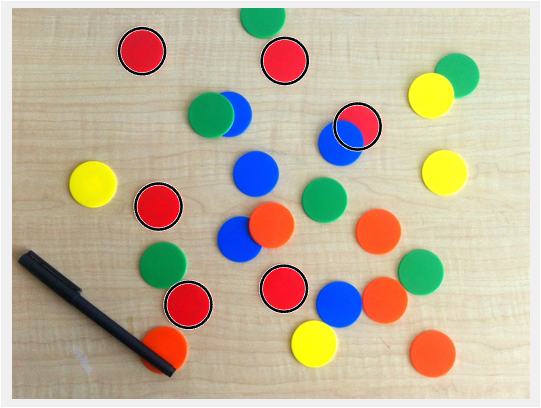
[centers, radii] = imfindcircles(out,[20 25], 'ObjectPolarity','bright', ...

'Sensitivity',0.92,'Method','twostage'); %create table to save centroid and radius.

h = viscircles(centers,radii,'Color','k');%overlap circles on newest figure







Ex1.

\*Detect shapes by color

clc

close all

clear all

%%

[X,cmap] = imread('shapes.gif');

rgb = ind2rgb(X,cmap);

figure,imshow(rgb),impixelinfo

gray=rgb2gray(rgb); %show "index"

figure,imshow(gray),impixelinfo

%%Scan pixel of image RGB

[r c] =size(gray);

for i = 1:r

for j= 1:c

%% satify condition => white

if gray(i,j) > 0.49 && gray(i,j) < 0.5 %màu cam ??m,hình vuông

out\_square(i,j) = 255;

%

elseif gray(i,j) > 0.72 && gray(i,j) < 0.74 %màu cam nh?t,elip

out\_elip(i,j)= 255;

%

elseif gray(i,j) > 0.8 && gray(i,j) < 0.81 %màu h??ng,hexagon

out\_hexagon(i,j)= 255;

%

elseif gray(i,j) > 0.47 && gray(i,j) < 0.48 %lá cây,hexagon

out\_circle(i,j)= 255;

%

elseif gray(i,j) > 0.16 && gray(i,j) < 0.17 %blue,hexagon

out\_rectangle(i,j)= 255;

%

elseif gray(i,j) > 0.88 && gray(i,j) < 0.89 %zang`,hexagon

out\_triangle(i,j)= 255;

%

elseif gray(i,j) > 0.48 && gray(i,j) < 0.49 %rose,octagon

out\_octagon(i,j) =255;

%%Other = Black

else

out\_square(i,j) = 0;

out\_elip(i,j) = 0;

out\_hexagon(i,j)= 0;

out\_circle(i,j)= 0;

out\_rectangle(i,j)= 0;

out\_triangle(i,j)= 0;

out\_octagon(i,j) =0;

end

end

end

%%Get BoundingBox and Area vs Centroid

[B\_square,L\_square] = bwboundaries(out\_square,'noholes');

[B\_elip,L\_elip] = bwboundaries(out\_elip,'noholes');

[B\_hexagon,L\_hexagon] = bwboundaries(out\_hexagon,'noholes');

[B\_circle,L\_circle] = bwboundaries(out\_circle,'noholes');

[B\_rectangle,L\_rectangle] = bwboundaries(out\_rectangle,'noholes');

[B\_triangle,L\_triangle] = bwboundaries(out\_triangle,'noholes');

[B\_octagon,L\_octagon] = bwboundaries(out\_octagon,'noholes');

%%

stats\_square = regionprops(L\_square,'Area','Centroid');

stats\_elip = regionprops(L\_elip,'Area','Centroid');

stats\_hexagon = regionprops(L\_hexagon,'Area','Centroid');

stats\_circle = regionprops(L\_circle,'Area','Centroid');

stats\_rectangle = regionprops(L\_rectangle,'Area','Centroid');

stats\_triangle = regionprops(L\_triangle,'Area','Centroid');

stats\_octagon = regionprops(L\_octagon,'Area','Centroid');

%%A empty Matrix M by 2

position=[];

%%Got T?a ?? c?a t?t c? centroid

%Square

for i = 1:length(stats\_square)

position = vertcat(position,stats\_square(i).Centroid);

end

%Elip

for i = 1:length(stats\_elip)

position = vertcat(position,stats\_elip(i).Centroid);

end

%Hexagon

for i = 1:length(stats\_hexagon)

position = vertcat(position,stats\_hexagon(i).Centroid);

end

%circle

for i = 1:length(stats\_circle)

position = vertcat(position,stats\_circle(i).Centroid);

end

%Hexagon

for i = 1:length(stats\_rectangle)

position = vertcat(position,stats\_rectangle(i).Centroid);

end

%Hexagon

for i = 1:length(stats\_triangle)

position = vertcat(position,stats\_triangle(i).Centroid);

end

%Octagon

for i = 1:length(stats\_octagon)

position = vertcat(position,stats\_octagon(i).Centroid);

end

%%overlap on orgiginal image vs Draw 'text' to regconize shapes

figure,imshow(rgb)

%Square

next\_F=length(stats\_square);

for i = 1:next\_F

text(position(i,1)-5,position(i,2),'S')

end

%Elip

next\_E=next\_F+length(stats\_elip);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'E')

end

next\_F=next\_E;

%Hexagon

next\_E=next\_F+length(stats\_hexagon);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'H')

end

next\_F=next\_E;

%Circle

next\_E=next\_F+length(stats\_circle);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'C')

end

next\_F=next\_E;

%Rectangle

next\_E=next\_F+length(stats\_rectangle);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'R')

end

next\_F=next\_E;

%Triangle

next\_E=next\_F+length(stats\_triangle);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'T')

end

next\_F=next\_E;

%Triangle

next\_E=next\_F+length(stats\_octagon);

for i = next\_F+1:next\_E

text(position(i,1)-5,position(i,2),'0')

end

next\_F=next\_E;

