Image processing

Coin counter

Dr: Neven Ibrahim

TA: Mahmoud Magdy

Members:

* Radwa Hamdy 20182893
* Maryam Emad 20182164
* Noureen Tarek 20182622
* Ahmed Abdelmaksoud 20184849

Coin counter

Let me talk about our project in details.

Our project is talking about how to sum number of coins in an image

We pass through many functions to get our output, and we faced many problems we try to solve them.

We will talk each function and what the problem we faced in it.

Firstly, we got a problem in MATLAB no one have it in PC, we downloaded it and start Our work...

we got another problem of image quality we asked the TA: Mahmoud Magdy to download the image in a file to download it in file that be able to download it with good quality.

Secondly, we start to write our code in MATLAB we start our code by entering our file location that MATLAB can detect and execute function on it.

Graphical user interface, text, application

Description automatically generated

We start to enter our image on MATLAB by function called “imread(‘imgname.PNG’);”

And we used function “figure,imshow(Img),title ('original image');”

This function is used to show our image to the user of MATLAB.



Output:

Graphical user interface, text, chat or text message

Description automatically generated

We used function “rgb2gray(I); ” ” figure,imshow(img that converted),title (' image in gray level');” to convert the RGB colormap to a grayscale colormap and redisplay the image.

We convert it for the complex “time and resource” as we work on one layer as the features doesn’t different as it come as an average between them.

Graphical user interface, text, application, website

Description automatically generated

Graphical user interface, text

Description automatically generated

After that we used threshold function to convert image to Black and white and redisplay it ” BW=im2bw(z,0.9);

figure,imshow(BW),title (' image in black and white');”

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

We inverse the image by using “BW = ~BW;

figure,imshow(BW),title('inverse');”

that inverse the image from black to white And vice versa.

Text

Description automatically generated

Graphical user interface

Description automatically generated

After that we use function “ fill=imfill(BW,'HOLES');

figure,imshow(fill),title('filling holes');”

to fill holes in the image to be one object.

Text, letter

Description automatically generated

Graphical user interface, application

Description automatically generated

We used function to convert an image to RGB that detect each object separately as we have 6 objects and background.

“[L, num] = bwlabel(BW);

RGB = label2rgb(L);

figure,imshow(RGB);”

Text

Description automatically generated

Chart, bubble chart

Description automatically generated

We use function to detect size of image “[h, w, ~] = size(I);

smallRatio = h\*w\*0.002;”

Text

Description automatically generated

We use label function to detect each object separately and we used for loop for the image to detect each object” for i=1:num

x = uint8(L==i);

f = sum(sum(x==1));

if(f < smallRatio)%% if the pixel is noise

continue;

end

d = zeros(size(I));

d(:,:,1) = uint8(x).\*I(:,:,1);

d(:,:,2) = uint8(x).\*I(:,:,2);

d(:,:,3) = uint8(x).\*I(:,:,3);

figure,imshow(uint8(d));

end

“

Text, timeline

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated