Name : Ankita Nag

OBJECT DETECTION AND TRACKING USING MATLAB

Aim: Red colour object detection and position tracking in Matlab

Theory:

In imaging science, image processing is processing of images using mathematical operations by using

any form of signal processing for which the input is an image, a series of images, or a video, such as a

photograph or video frame; the output of image processing may be either an image or a set of

characteristics or parameters related to the image. In this project we have a live video input coming

from a Logitech webcam where the aim is to identify red-colour present anywhere in that frame and

to show the centroid and boundary of this red-coloured object present in a multi-coloured

background.

The first task is to read the input as object from the webcam and give required values to parameters

like ‘FramesPerTrigger’ as ‘inf’ (infinity). Then we start the video and data is stored for only the just

elapsed 1 second ( 200\*5ms) to save memory. Each frame is taken at an interval of 5ms and the first

task which is done on this image is that the black and white parts are subtract from the image using

‘imsubtract’command. Next the ‘medfilt2’ command is used to find the median filtered image in the

3 x 3 neighbourhood of a pixel, and then at a distance of every 0.18 is checked in the binary image .

The actual red colour detection takes place in this line:

‘diff\_im=bwareaopen(diff\_im,300)’ , where 300 is the threshold given by us for detecting red colour.

The next line is also very important as it finds the connected region of red colour using ‘bwlabel’

command with a specified value of n=8 (default for 2-dimensions).

After this we get information relating to the boundary of the object (red colour) and its centroidal

location (of a rectangle). This information is then sent to the monitor for display by giving the

necessary inputs to ‘plot‘. Different font parameters are given and then the data stored till 200

frames ( 1 second) is cleared using ‘flushdata(vid);’.

Code:

vid = videoinput('winvideo');

set(vid,'FramesPerTrigger',Inf);

set(vid,'Returnedcolorspace','rgb')

vid.FrameGrabInterval=5;

start(vid)

while(vid.FramesAcquired<=200)

data=getsnapshot(vid);

diff\_im=imsubtract(data(:,:,1),rgb2gray(data));

diff\_im=medfilt2(diff\_im,[3 3]);

diff\_im=im2bw(diff\_im,0.18);

diff\_im=bwareaopen(diff\_im,300);

bw=bwlabel(diff\_im,8);

stats=regionprops(bw,'BoundingBox','Centroid');

imshow(data)

hold on;

for object=1:length(stats)

bb=stats(object).BoundingBox;

bc=stats(object).Centroid;

rectangle('Position',bb,'EdgeColor','r','LineWidth',2)

plot(bc(1),bc(2),'-m+')

a=text(bc(1)+15,bc(2),strcat('X:',num2str(round(bc(1))),'y:',num2str(round(bc(2)))));

set(a,'FontName','Arial','FontWeight','bold','FontSize',12,'color','yellow');

end

hold off;

end

stop(vid)

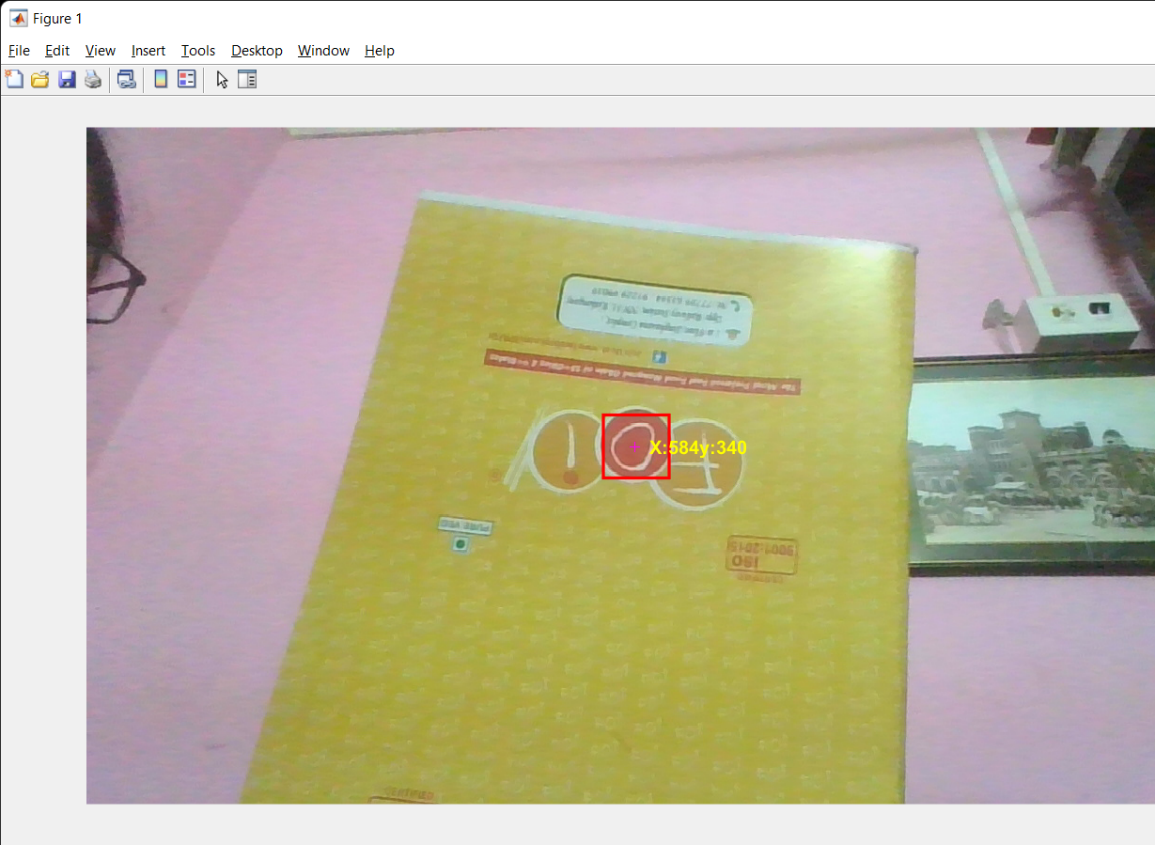
fiushdata(vid);

sprintf('%s','ImageTrackingcompleted')

Results:

Following are few of the snaps taken during or at the end of program run which detect the red

coloured part in a multicoloured background.



Conclusion:

In this project Matlab code is able to identify the red colour object in front of the camera and

encircle it by a red coloured rectangle and also giving the centroid of the object. It can be used if we

want to replace the red colour object by any other object of our desire. The application of this

project is widely used in modern gaming softwares, the latest and most famous being ‘Pokemon Go’

We see that there are many uses in navigation and position indicating in many flight vehicles.