//Python code in generating the CSV file from a Grayscale image to binary

import cv2

import numpy

img = cv2.imread('minirod.jpg')

ret,gray = cv2.threshold(img,127,256,cv2.THRESH\_BINARY)

cv2.imshow('Original Image',img)

cv2.imshow('Binary Image',gray)

with open('binaryfile.csv','w') as f:

for row in gray:

TMP = ''

for col in row:

TMP += '{0}, '.format(col[0])

TMP.rstrip(', ')

f.write(TMP+'\n')

f.close()

cv2.waitKey(0)

//Verilog code in counting the pixels hit from the binary image

// module declaration

module file\_readmemh#( parameter WIDTH = 208, HEIGHT = 175)( input clk, output reg out);

//image size 208 x175

reg [WIDTH:0] data [0:HEIGHT];

//read the binary image file

initial $readmemb("binaryfile-01.txt", data);

integer i, count,max;

/\* read and display the values from the text file on the compiler screen \*/

initial begin

count = 0;

max = 0;

$display("Image Data:");

for (i=0; i < HEIGHT; i=i+1)

begin

if (data[i] == 0)

begin

$display("Empty");

count = 0;

end

else if(data[i] != 0)

begin

count = count +1;

$display("White");

$display("%b",data[i]);

end

if (max < count)

begin

max = count; //height of the white stripe

$display("%d",max);

end

end

$display("The approximate length of the line is %d inches",max);

// $display("%d:%b",i,data[i]);

end

endmodule