Summer of Code :
Image Processing and Object Detection

Omkar Shirpure (22B0910)

June 25, 2023

Contents

1 Introduction

In this project we develop a Image processing model to predict the features of any image/object provided. This project will begin from basics of Python, learn libraries like numpy, OpenCV and matplotlib. We will work on two model to explore the field.

The Two models will focus on:

- Detecting number of objects present in the frame and area covered by them.
- Making a model to determine the porosity of the metal given its highly magnified surface image.

2 Methodology

The images/live images would be modified using various filtered settings to obtain the desired results for the specific purposes. Various types of blurs (Simple blur, Weighted Gaussian blur, Median filtering) to remove noise in the image, sharpening the image to make edges of objects more clear, then edge detection techniques like Canny, Sobel edge detection methods to get the edges of the objects.

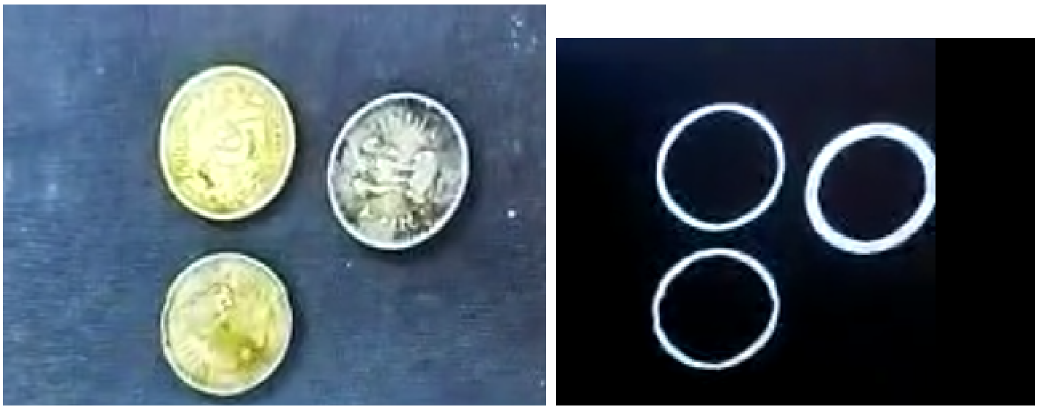
After that, using contours feature of openCV, saving the contours obtained contours after fine tuning these above settings to get fine edges and calculating the area of those contours to get the edge of the specific objects.

For the second part, same type of contours would be used to calculate the area of dark region (pores) and determine the overall porosity of that metal by the formula

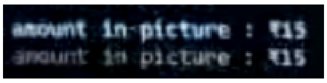
$$porosity = \frac{dark\ region\ of\ pores}{total\ area\ of\ surface} * 100$$

3 Object Detection

- Demo Project - Money Counter (To implement both edge detection and get no. of objects) (video of demo provided)
  - Used guassian blur and canny edge detection functions to convert the feed from camera of phone to get only the edges of coins.
  - Basically detects the number of objects/circles in the screen.
  - Then based upon the area of the circle(edge of the coin) (PS - which was calibrated before hand in code), it detects the type/value of the coin.
  - Then the value of the coins is added up after every refresh. (the video demo of the final working testing model is also provided in the GitHub Repositry).



The original snapshot of the video stream of the coins (left), and the processed image to obtain the area of contours(right)

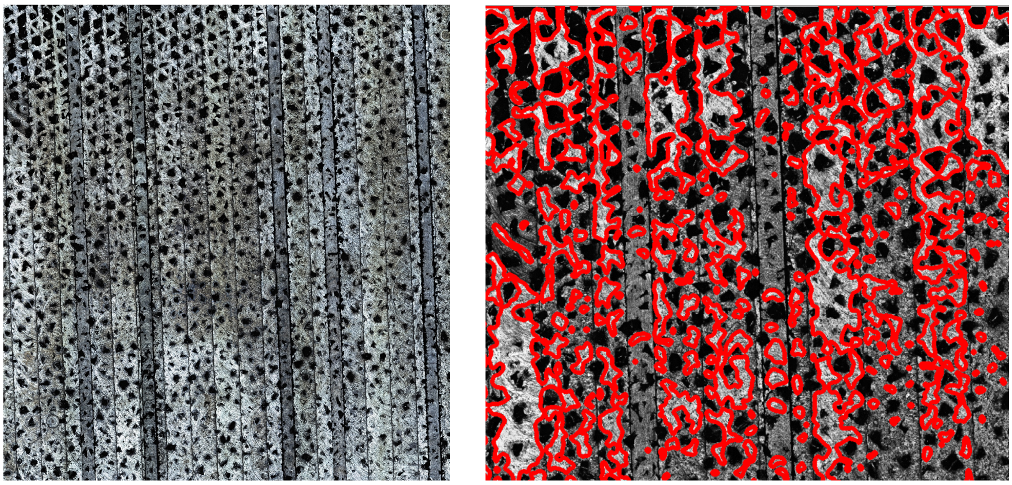


```
amount in picture : ₹15
amount in picture : ₹15
```

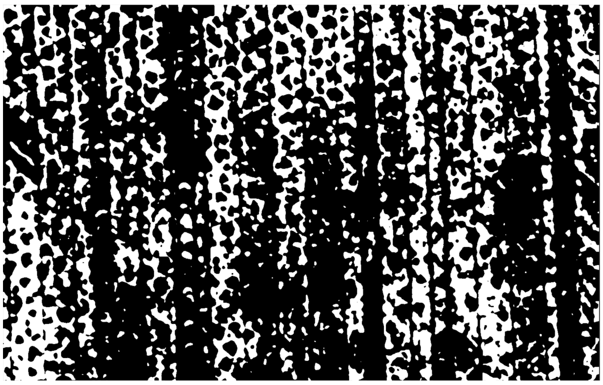
Total amount displayed in the terminal

4 Porosity Calculator

- The Porosity calculator (Use combination of image processing techniques such as blurs and different thresholding techniques)
  - *image_path* contains the path of the highly magnified image of the surface of metal.
  - The default *blur_rate* and *threshold_value* is set to the recommended value.
  - This can be adjusted using the live slider to get the best configuration to find the dark pores of the metal surface inside the contours and verify the threshold image matches the approximate pore structure.
  - Once the desired configuration is obtained press 'q' to exit the image recalibration and the value of porosity in percentage will be displayed.
  - if the desired value is not obtained just adjust the values to best cover all the pores(dark areas) in contours



Original image to be processed(left) and the processed image with contours around the pore region marked on the image(right).

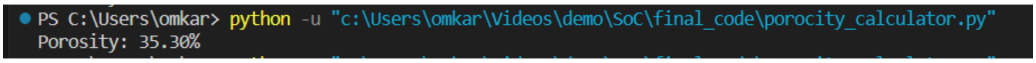


The processed binary image after the contours and applying proper value of threshold. (Dark part indicates the pores and white indicated the metal surface)

5 Results

The porosity of the metal sample was calculated using the highly magnified image. The image processing and object detection algorithm were applied to analyze the porosity characteristics of the metal.

The result for the given problem image after adjusting the *threshold* = 140 and *blur_rate* = 5 to cover all the contours of the pores was found to be 35.30%



```
● PS C:\Users\omkar> python -u "c:\Users\omkar\Videos\demo\SoC\final_code\porocity_calculator.py"
  Porosity: 35.30%
```

6 References

- Numpy Tutorial: https://www.youtube.com/watch?v=j31ah5Qa4QI
- OpenCV Tutorial: https://www.youtube.com/watch?v=oXlwWbU8l2o
- Image processing with OpenCV: https://youtu.be/kSqxn6zGE0c
- OpenCV Documentation: https://docs.opencv.org/4.x/
- Numpy Documentation: https://numpy.org/doc/