



# **SoC: Image Processing and object Detection**

## **Week 5 and 6**

NAME: B V S S Prabandh

Roll\_no: 210050037

**JULY 1 2023**

## Determine the % porosity of the surface

### Idea

- First read the image using openCV in grayscale
- Find total number of pixels using shape of image
- We have to convert image into black and white using appropriate threshold
- find number of pixels which are black i.e, equal to zero (number of void pixels)
- Hence, find % porosity of the surface

First import the necessary packages cv2 and numpy

```
1 # import packages
2 import cv2
3 import numpy as np
```

**Listing 1:** Import packages

Read image in gray-scale and convert it into a black-white image using a threshold value. We are using cv2.THRESH\_BINARY\_INVERSE this changes pores into white and remaining into black.

```
1 # threshold values
2 threshold = 5
3
4 # Read image in grayscale
5
6 image = cv2.imread('Image-detection/demo/Resources/Photos/Sample Image.jpg'
7                   , cv2.IMREAD_GRAYSCALE)
8
9 # Convert into black and white image
10 _, binary_image = cv2.threshold(image, threshold, 255, cv2.
11                                THRESH_BINARY_INV)
```

**Listing 2:** Image transform

Find total number of pixels and number of white pixels in the binary image and hence find porosity.

```
1 # find total pixels in the image
2 total_pixels = image.shape[0] * image.shape[1]
3
4 # find number of black pixels
5 pore_pixels = cv2.countNonZero(binary_image)
6
7 # Calculate porosity_percentage = (black pixels)/(total pixels)* 100
8 porosity_percentage = (pore_pixels / total_pixels) * 100
9
10 # print porosity upto 2 decimals
11 print(f"Porosity Percentage: {porosity_percentage:.2f}%")
```

**Listing 3:** Porosity

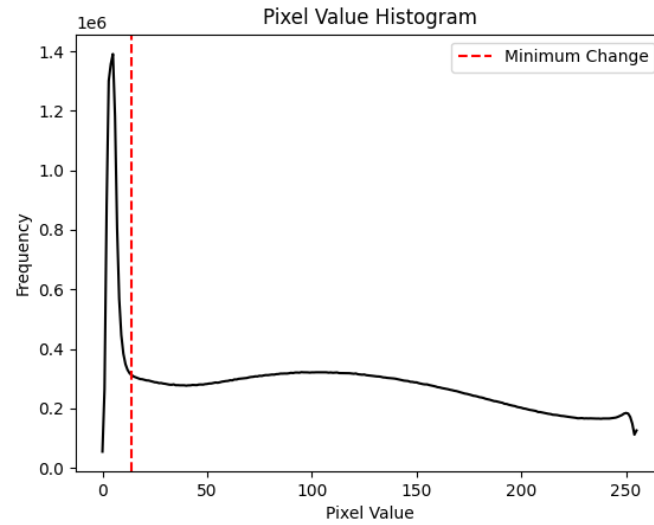
Close windows upon keyboard input.

```
1 # wait and destroy all windows upon keyboard input
2 cv2.waitKey(0)
3 cv2.destroyAllWindows()
```

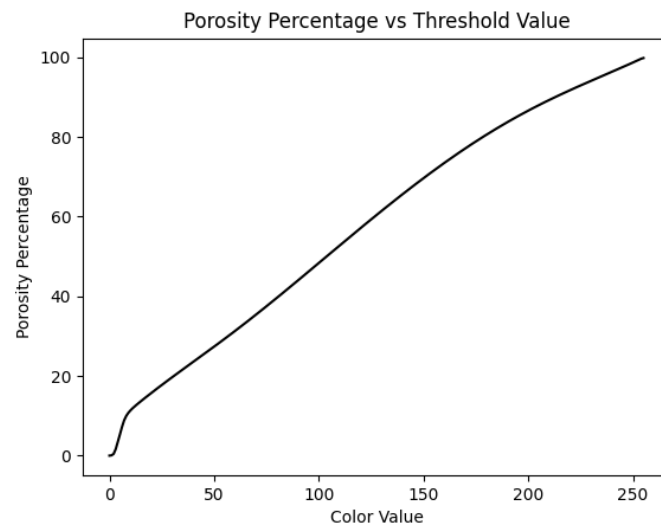
**Listing 4:** Delete windows

## **Tunable parameters that could affect accuracy of porosity**

We need to adjust the threshold value so that all the pore pixels have value less than threshold and white pixels more than threshold. How to find threshold? Lets look at a graph between colour\_value and number of pixels for that certain colour value



We can see that most of the pore pixels are under the color value of 14.  
 So if we set the threshold value as 14, we might get the best approximate value.  
 If we run our code for a threshold of 14, we obtain a porosity of **13.66%**.



Above is the graph between threshold and porosity percentage  
 The kink at the start is the most accurate value of porosity.

```

1 # import packages
2 import cv2
3 import numpy as np
4
5 # threshold values
6 threshold = 14
7
8 # Read image in grayscale
9
10 image = cv2.imread('Image-detection/demo/Resources/Photos/Sample Image.jpg'
11                    , cv2.IMREAD_GRAYSCALE)
12
13 # Convert into black and white image
14 _, binary_image = cv2.threshold(image, threshold, 255, cv2.
15                                THRESH_BINARY_INV)
16
17 # find total pixels in the image
18 total_pixels = image.shape[0] * image.shape[1]
19
20 # find number of black pixels
21 pore_pixels = cv2.countNonZero(binary_image)
22
23 # Calculate porosity_percentage = (black pixels)/(total pixels)* 100
24 porosity_percentage = (pore_pixels / total_pixels) * 100
25
26 # print porosity upto 2 decimals
27 print(f"Porosity Percentage: {porosity_percentage:.2f}%")
28
29 # wait and destroy all windows upon keyboard input
30 cv2.waitKey(0)
31 cv2.destroyAllWindows()

```

**Listing 5:** Complete script

## Using ADAPTIVE THRESHOLD to find threshold value

Most of script stays same , we need to add adaptive threshold instread of manual threshold

```
1 adaptive_threshold = cv2.adaptiveThreshold(image, 255, cv2.  
    ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY_INV, 11, 8)
```

**Listing 6:** adaptive threshold

If we run our code using adaptive threshold we obtain a porosity of **31.68%**.

```
1 # import packages  
2 import cv2  
3 import numpy as np  
4  
5 # threshold values  
6 threshold = 14  
7  
8 # Read image in grayscale  
9  
10 image = cv2.imread('Image-detection/demo/Resources/Photos/Sample Image.jpg',  
    , cv2.IMREAD_GRAYSCALE)  
11  
12 adaptive_threshold = cv2.adaptiveThreshold(image, 255, cv2.  
    ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY_INV, 11, 8)  
13 # find total pixels in the image  
14 total_pixels = image.shape[0] * image.shape[1]  
15  
16 # find number of black pixels  
17 pore_pixels = cv2.countNonZero(adaptive_threshold)  
18  
19 # Calculate porosity_percentage = (black pixels)/(total pixels)* 100  
20 porosity_percentage = (pore_pixels / total_pixels) * 100  
21  
22 # print porosity upto 2 decimals  
23 print(f"Porosity Percentage: {porosity_percentage:.2f}%")  
24  
25 # wait and destroy all windows upon keyboard input  
26 cv2.waitKey(0)  
27 cv2.destroyAllWindows()
```

**Listing 7:** complete script