CS669 Project Report
On
Surface Crack Detection

Rajan Shukla T21016

Vikas Singh T21017

1 SUMMARY

The aim of this project is to be able to develop an algorithm that will detect the Surface crack when an image input is given we have used Matlab for coding.

We have taken image data which helps us to make assumptions about various aspects of this algorithm and it also has improved the efficiency of this program

We have also discussed the working of the algorithm and included the test results in this report, and finally, we have developed a Graphical User Interface for using our algorithm.

2 PROBLEM

The cause of the Surface crack is due to the following reasons :

- The cause of Cracks in the concrete surface would be excess water. General Rule says that if you mix 1L of excess water in 50KG of Cement mix so it will decrease the strength by 1.4mm N/m^2.Ans this case to increase the strength and cause shrinkage because when water is evaporated it will create a void in structure.
- When Concrete is drying too fast because concrete reacts with water if temperate is high during summer the concrete will dry fast.
- Improper strength concrete poured on the job.
- Lack of Control Joint, Less curing Done, Settlement of soil.

Due to this reason, the concrete structure develops the crack and we need a way to identify them and repair them to avoid catastrophe.

3 SOLUTION

Manual inspection has considerable limitations in that it is time-consuming and complex. We need a way to automate this process, and this initiative certainly attempts to take the first step in that direction. When a raw image is put into the algorithm, it will recognize the crack and highlight the outline of the fracture. We will create an algorithm that will be based on MATLAB and will use the basics of Picture Processing that we learned in our CS669 course to recognize the crack in an image. This functionality can be expanded to incorporate real-time video, but that is beyond the scope of this project. Furthermore, the output image from our algorithm may be used by an expert civil engineer to determine the source and remedy.

3.1 ALGORITHM USED

Input the image using GUI.

We take a block size of 50 which is found by trying different images.

In the histogram, the count is the number of individuals in each bin. The count is also known as the frequency. By these counts, we can determine a percentage of individuals with a given interval of variable values.

Otsu method is clustering-based image thresholding, Which works when the histogram is bimodal. This method is used to minimize the within-class variance and at the same time, it maximizes the between-class variance because TOTAL variance = Within-class variance + Between-class variance.

We divide the image into small images so that the average filter can perform efficiently.

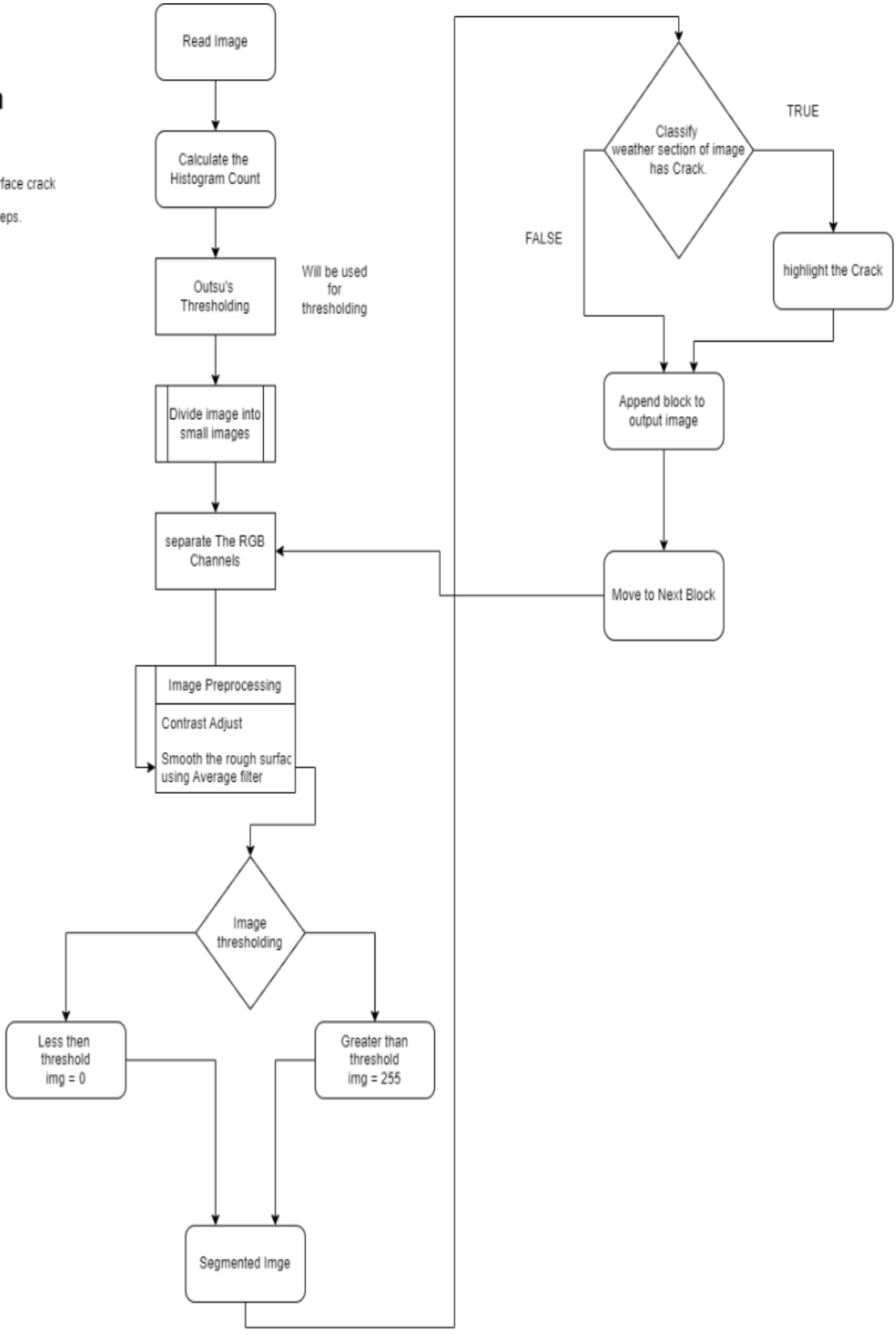
To create an RGB image with uninterrupted areas of red, green, blue and display the image separate the three color channels.

The Working of the Algorithm can be understood From the Below Flow chart.

Algorithm Working

For detecting the surface crack

we have used this steps.



Input the image using GUI.

We take a block size of 50 which is found by trying different images.

In the histogram, the count is the number of individuals in each bin. The count is also known as the frequency. By these counts, we can determine a percentage of individuals with a given interval of variable values.

Otsu method is clustering-based image thresholding, Which works when the histogram is bimodal. This method is used to minimize the within-class variance and at the same time, it maximizes the between-class variance because TOTAL variance = Within-class variance + Between-class variance.

We divide the image into small images so that the average filter can perform efficiently.

To create an RGB image with uninterrupted areas of red, green, blue and display the image separate the three color channels.

To adjust the contrast in an RGB image, we perform -

1. View the histogram of the image to determine the intensity value limits.
2. specify these limits as a fraction between 0 and 1.

Average or Mean filter reduces the amount of intensity variation between neighboring pixels. It works by moving the image pixel by pixel, replacing each value with the average value of neighboring pixels including itself.

Then we check the segmented image whether the section of the image has cracks or not, If yes highlights the image, and if not append the block to the output image. Then again perform all the processes for another separated RGB channel.

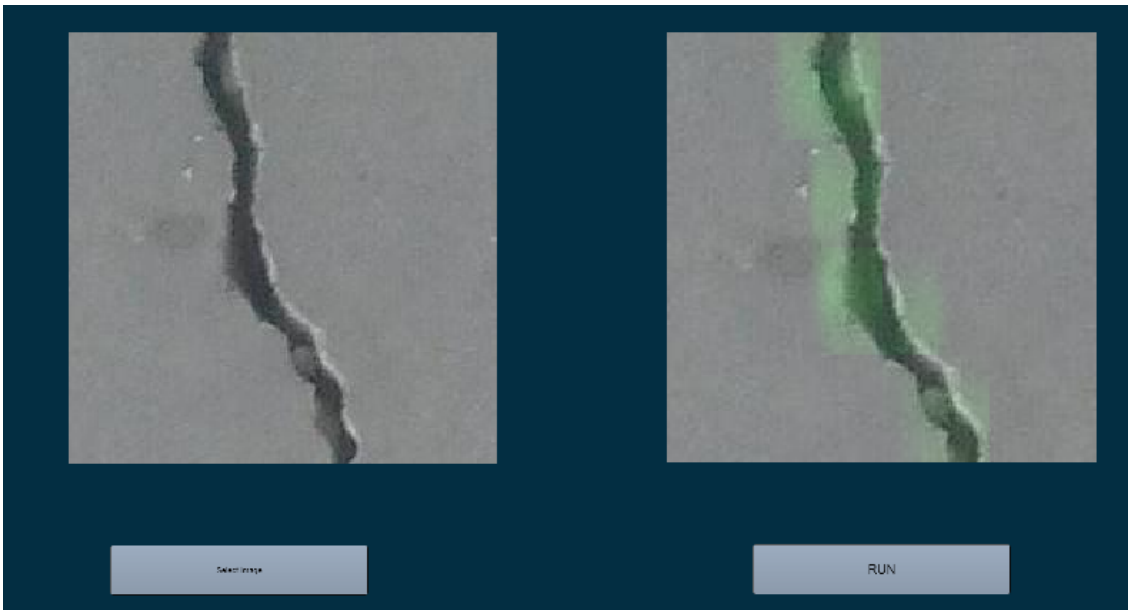
3.2 ASSUMPTION

This Algorithm assumes that that input image is of surface only as it will perform based on this assumption only.

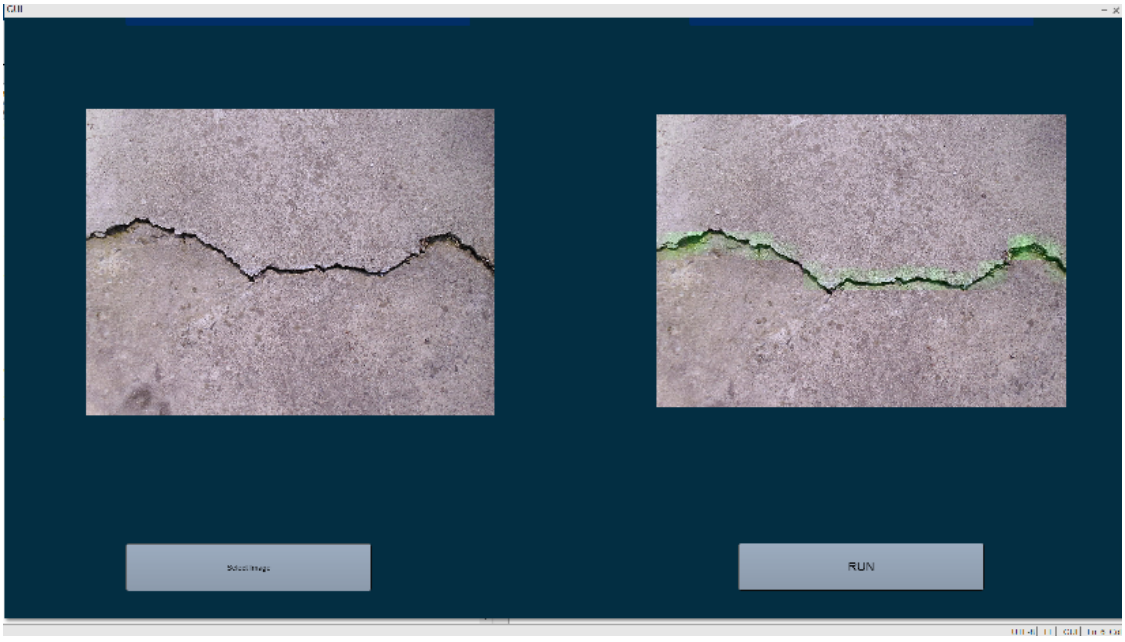
We have taken a block size of 50 which is found by trying different images.

4 Result and Analysis

Left side are input images and just right side are output imges respectively in GUI.

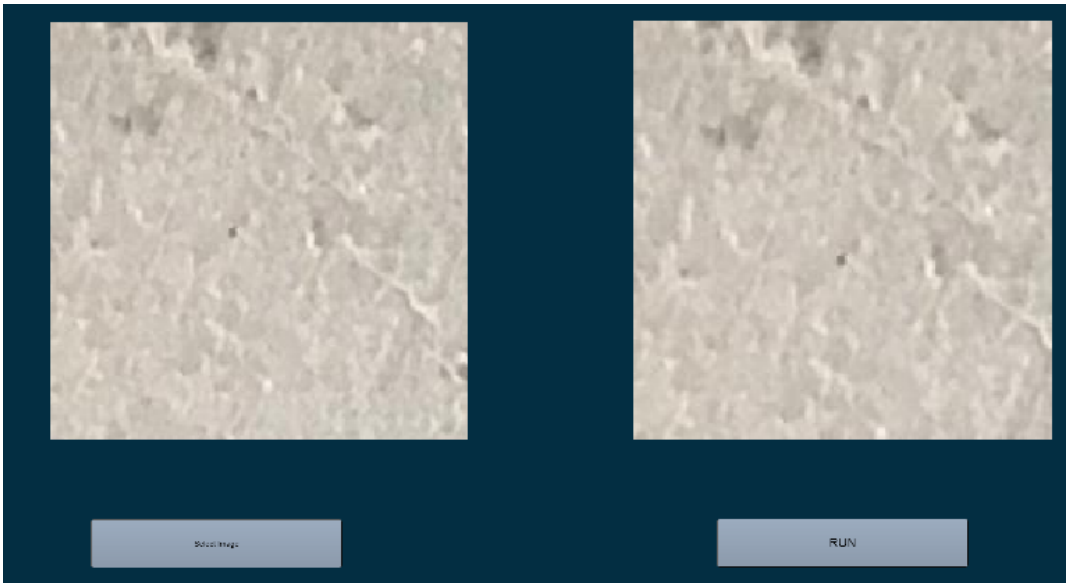


Test image source Kaggle



Test image source Google

cracks is detected by green marking.



Test image source Kaggle

Here roughness is there in the surface, Not cracks are present. So no green marking in the output. code is working fine.

Test image source Google

But for the above image, we can see that algorithm has detected the rough surface as the crack which is not correct although it has detected the crack very well.

5 Conclusion

This project helps us to apply the concept of Digital Image Processing and solve real-world problems and the used case we have taken is just a small part of the potential image processing have. Along with this, we have learned an equally important quality as working in a team sharing ideas, and having a discussion making improvements.

6 Project GitHub page

RAJAN SHUKLA-Link

VIKAS SINGH-Link