Identification and Comparative Analysis of Potholes Using Image Processing Algorithms

Submission Date: 17/4/2018 Tuesday

Submitted by:

Fahim Shahriare Anik-13221035

Asif Ahmed- 14121002

Avita Maheen-14121065

Puily Alam-14121054

Department of Electrical and Electronics Engineering

BRAC University

Supervisor:

Dr. Amitabha Chakrabarty

Assistant Professor

Department Of Computer Science and Engineering

Declaration

We, hereby declare that this thesis is based on the results found by ourselves. Materials of work found by other researcher are mentioned by reference. This Thesis, neither in whole or in part, has been previously submitted for any degree.

Signature of Supervisor

Dr. Amitabha Chakrabarty

Assistant Professor

Department Of Computer Science and Engineering

BRAC University

Signature of Author

Fahim Shahriare Anik-13221035

Signature of Author

Asif Ahmed-14121002

Signature of Author

Avita Maheen -14121065

Signature of Author

Puily Alam - 14121054

Abstract

Potholes have become a major havoc and are the leading reason for the damage of road transportation vehicles. This is why this paper was made to verify and understand whether a feasible commercial product can be made using the current algorithms to detect potholes, which can aid the driver of the vehicle. The topic selected uses different algorithms to test whether it can identify all types of potholes based on the intensity of the image and the shape of the pothole in any given condition. Performance analysis of the different algorithms is used in Matlab2015Ra image processing toolbox to find out the effectiveness of the algorithms for potholes in different environments of Bangladesh and the pothole photos from internet. And the following techniques that were worked on were Image Thresholding, canny edge detector, K-Means clustering and Fuzzy C-Means clustering along with other operations.

Acknowledgment

All thanks to Almighty ALLAH, the most merciful, beneficent and the most gracious; who provided us guidance, strength and abilities to complete this research. We are especially thankful to Dr. Amitabha Chakrabarty, our thesis supervisor, for his help, guidance and support in completion of our project. We are grateful to BRAC University, its faculty and Staffs and the department Electrical and Electronics and Engineering, who have been a light of guidance for us in the whole study period at BRAC University, particularly in building our base in education and enhancing our knowledge. Finally, we would like to thank and express our sincere gratefulness to our beloved parents, brothers and sisters for their love and care. We are grateful to all of our friends who helped us directly or indirectly to complete our thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Potholes became a major havoc and are the leading reason for the damage of road transportation vehicles. This is why this paper was made to verify and understand whether a feasible commercial product can be made using the current algorithms to detect potholes, which can aid the driver of the vehicle. Road transport means moving goods and personnel to different locations via roads. Simply speaking it is a medium between two destinations that is cemented or worked on to allow transportation by way of power-driven vehicles. There are many advantages of road transport compared to other means of transport. The cost of construction, operating cost and maintaining roads are cheaper than that of railways. The major advantage of road transport is that funding required is very less compared to other forms of transport such as railways and air transport also it can enable door-to-door delivery of goods and materials and can provide a very cost-effective means of cartage, loading and unloading. Sometimes road transport is the only way for carrying goods and people to and from rural areas which are not carried to by rail, water or air transport. Road transport allows transport of goods and services to any place. Every mode of transport has limitations even this one for instance, opportunities for accident by road is much higher than any other form of transport and so is not as safe. Organization of mode of travel is low compared to other forms and is irregular and undependable as well as being unstable. However the speed of travel is the biggest issue. Long distance transport of goods via roads can also be costly [1].

1.1 Bangladesh roads and road surface layers

Bangladesh grew rapidly over the course of its independence. The annual growth rate for freight roads is 8.2% and 8.4% for passenger transport [2].

Table 1.1- Projected demand and modal shares- Passenger and Freight [2]

Year	Passenger				Freight			
	Total Pass-Km	Mode Shares			Total Pass-Km	Mode Shares		
		Road	Rail	IWT		Road	Rail	IWT
1997	90 billion	72%	11%	17%	12 billion	65%	7%	28%
2002	152 billion	70%	12%	18%	19 billion	72%	8%	20%

The yearly rate of growth for transport demand with reference to the projected target year of 1997 are said to be 7.5 per cent and 8 per cent for passenger and cargo travel vehicles respectively for the year 2002. The expected transport demand and modal shares for the year 2002 is shown in Table 1.1 [2].

Figure 1.1: Typical highway roads in Bangladesh [3]

The increasing number of roads was due to the increasing number of vehicles being used throughout the country of Bangladesh to meet the demands of the increasing population. As of February 2017 2,948,906 cars were registered in the country which eventually meant that there would be more strain on the roads being used causing constant decay of the road surface [2], meaning that there would be potholes. A typical asphalt road constructed consists of at least 4-6 layers as shown in figure 1.2 below.

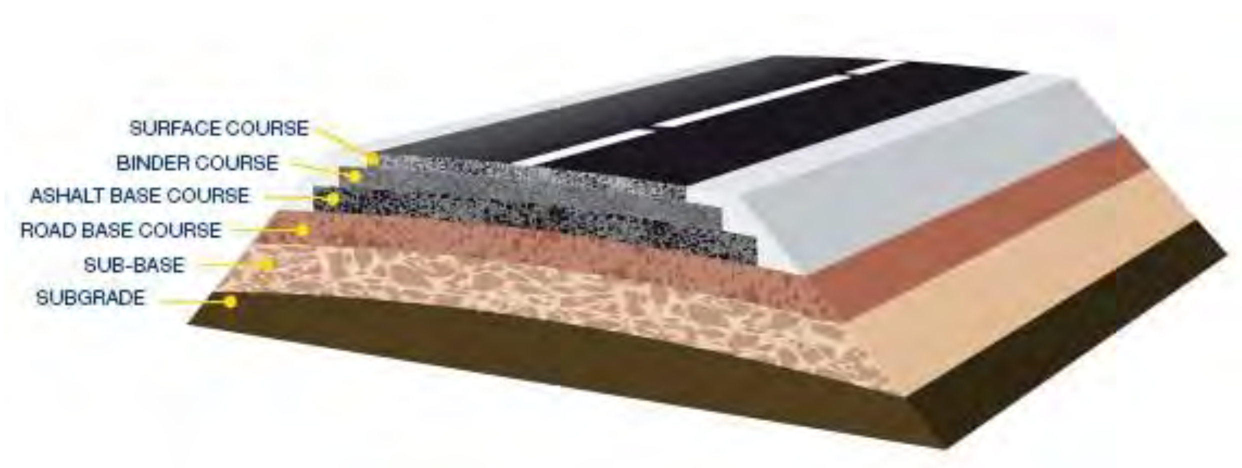

Figure 1.2- layers of an asphalt road [4]

Surface course is the part of the road in contact with vehicles and heavy load and extreme heat or cold or any other adversity of weather. Therefore it is made water proof and durable to heavy strain on the roads by the traffic. It is also made elastic so that it can expand in extreme heat to prevent it from cracking.

The layer immediately below the surface layer is the binder course. Its main function is to provide a structure and a base to evenly spread out the weight across the surface from the surface layer [5].

The function of the third layer and fourth layer, i.e. the base course (asphalt and road base course) is similar to that of the binder layer, which is to distribute the load evenly throughout the surface coming from the layer above. It is usually composed of crushed stones or crushed slag or anything that is untreated.

The sub-base consists of materials similar to the third and fourth layer and is essential to the lifespan of the road. The primary function of this surface is to provide structural support. Subgrade layer is made of just natural soil, it is the fixed base upon which the road is built upon all the other layers forms on this road to ensure that the road lasts long without decay the subgrade layer must be flat [5].

1.2 Potholes

Although the roads have a lifespan of up to 26-39 years [6] most roads do not last that long due to extreme environments and potholes. Potholes are formed due to wear and tear of road overtime due to environmental hazards and vehicle loads. Potholes can be of any shape and size and even depth. In Bangladesh, roughly 60.5% of the damage on transport vehicles via roads is due to potholes and hence a major issue for the vehicle owners. Moreover it enhances the chances of road accidents due to uneven road surface created by the pothole [7]. Figure 1.3 shows the formation of potholes and in which road layer it is usually associated with.

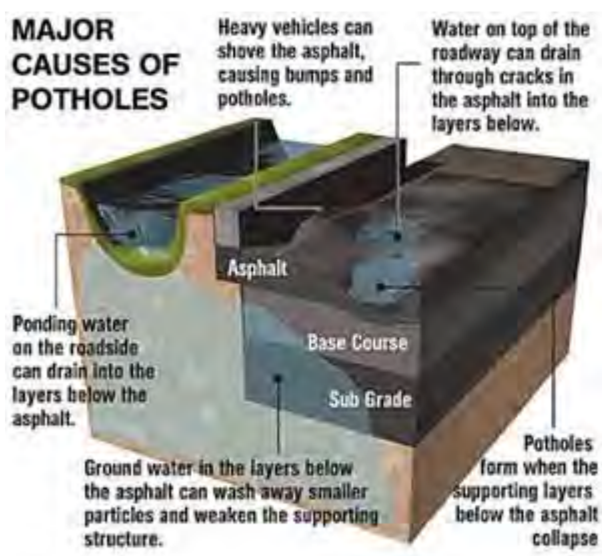

Figure 1.3- causes of potholes on pavement [8]

As briefly shown in the diagram above, there are several reasons for potholes which can be mainly summarized to pressure from vehicles, heavy rain or adversity in climate and poor drainage systems. Vehicles used for road transportation can weigh more than a ton. When they drive through the pressure on the pavement can be huge enough to displace the asphalt and gradually creating an uneven surface and road decay creating potholes. The drains on the side of the road if not properly isolated and built by brick and concrete can cause to soak the surrounding soil and layers of the road surface below causing the layers below to be distorted and displaced causing a pothole. Heavy rain can also create potholes when it flows through the cracks entering the layers below which can cause the top most layers to collapse.

1.3 Motivation

Road decay and potholes have been a constant major issue in Bangladesh and so this paper was made to analyze potholes using image processing. Before we look at how it is done using image processing an overview of image processing is required for general idea and to answer the question as to why image processing was used. Image processing is one of the most widely used operations at the moment because of its ability to convert analog to digital information which can be stored and formed into a database and which can be altered in many ways in order to extract results we seek. Image processing is an operation that converts an image into digital form so that different methods can be used to extract a result we are looking for. Since the thesis topic is about pothole detection, image processing was the best option to get an accurate result. It has many applications in many different areas such as remote sensing, intelligent transport system, defense surveillance, moving object tracking etc. It can be used not only for these purposes but for any problem solving situation in a digital platform. Hence why this topic was selected to detect potholes as it can be used in any way to produce the desired result, which in this case is detection of potholes.

1.4 Thesis Contribution Summary

The paper uses different algorithms applied on images in different environment and assessed to observe the results of the different images and cross matched with the results from other algorithms. This paper uses mainly a software based application that takes snaps after which it is applied to with different algorithms through MATLAB software to give the results which are then cross matched.

Four different types of algorithms are used in this paper for performance analysis for pothole detection which includes:

- Fuzzy C means algorithm for pothole detection
- K means algorithm for pothole detection

- Canny edge detection algorithm for pothole detection
- Image segmentation algorithm for pothole detection.

1.5 Thesis Outline

- Chapter 1 is introduction. Talks about the basics of potholes.
- Chapter 2 talks about background information such as image processing and the steps related to it in details as well gives a preview for the thesis and talks about the processes used and the morphological operations.
- Chapter 3 discusses the algorithms we used in details and how it is formed
- Chapter 4 elaborates the experimental results and how the results were evaluated in details
- Chapter 5 explains the conclusion and future goal of the system

Chapter 2

Technical review

For readers that are not aware of the terminology and how image processing is performed, this chapter provides an outlook and valuable information on the various tasks and operations related to image processing. Also, this chapter focuses on giving a brief idea of the algorithms and operations used and how it can be executed, this paper used four different algorithms along with many functions that allowed the detection of potholes possible such as structuring elements.

2.1 Image Pre-processing

Before image processing is done, basic image activities are executed to guarantee that the image is prepared to be utilized for image processing, the anomalies/blunders could be in resolution, brightness or contrast esteems along these lines making the image more moved forward. Image upgrade techniques are different and each unique in relation to the next that serves a particular capacity which has a definitive objective of making the image more profound and understandable to specialists or analysts. The other intention is to feature particular highlights of the image. Distinctive methods can be named as filtering, sharpening and so forth. Enhancing the image through improving it doesn't influence the measure of the information; it essentially makes a region of focus that guides the client for later stages. For this there are different methods such as:

- Image Thresholding
- Noise filtering
- RGB to grey image conversion
- Histogram equalization

These are a few processes amongst many others.

Threshold
Consists of image segmentation mostly, the main task of this process is to convert a grey scaled image into a Black and white image i.e. binary image. The feature is very useful for many operations especially for the thesis topic that we have discussed. What this method does basically is segmentation based on different intensities; there are only two options for the final image to be produced as it is a binary image, so it will be either zero or one. Each option is assigned a range of intensities, any value above or below the intensity is assigned with a number (either 0 or 1) to give us new co-ordinate values on a 2D plane which is plotted to give us pixels and which eventually forms a thresholded image [9]. For example, if a(b,c) is the final image, the new image co-ordinate formed would be:

$f(x,y)$ where, $a(b,c)=1$ if $f(x,y) \geq T$ and 0 otherwise [10].

An example of a threshold image is given in the figure below:

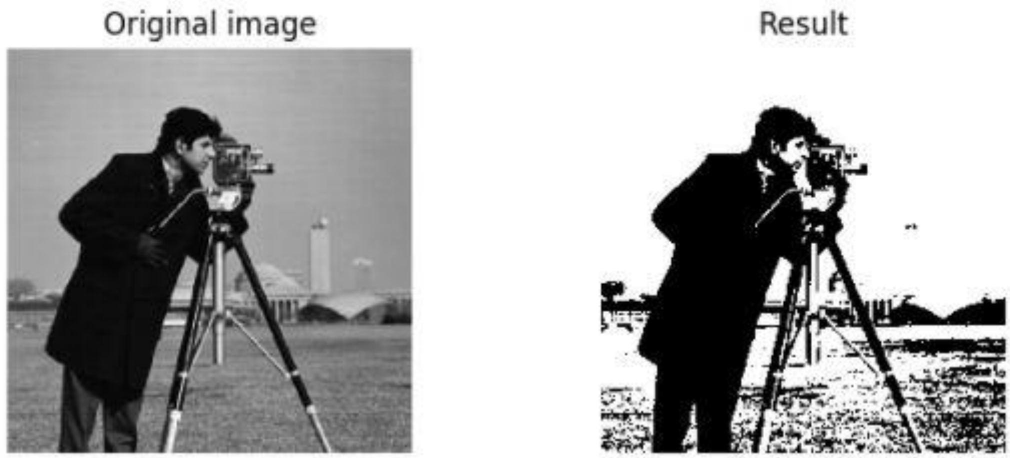

Figure 2.1-image before and after Thresholding [11]

Noise Filtering

Like noise in signals, noise in images also exists; noise in an image is excess or unwanted features of an image that is usually removed to amplify certain features for the user. Specific Filters are used to remove noise such as high or low or bandpass filters etc. noise is filtered using Gaussian distribution with parameter π [12].

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp(\frac{-(x-\mu)^2}{2\sigma^2}) \text{---------- (12)}$$

The figure below shows how noise filtering helps with enhancing the image.

Figure 2.2-Image before and after noise filtration [13]

RGB to Gray

All image operations are easier if the image is in grayscale and hence this step is very important step in image processing, this is usually the first step used before all the other steps are being carried out. The intensity of the image is converted from color domain to black and white domain, i.e. all intensities of color are converted to intensities of grey/black. For example pure blue is (0, 0, 255), pure Green is (0, 255, 0) and pure Red is (255, 0, 0) in terms of intensities of each color. The encoded value has a particular set of digits or a particular range assigned in the black and white domain, thus the representation follows that set of rules hence producing a black and white image [14]. The equation for the conversion is:

$$\text{Grayscale} = \frac{R+G+B}{3} \text{---------} (14)$$

And the perfect equation would be:

$$Grayscale\ Image = (0.3R + 0.59G + 0.11B) \text{--------}(14)$$

The following diagram shows the conversion of RGB to grayscale.

Figure 2.3-image before and after RGB to grayscale conversion [14]

Histogram Equalization

Its basic purpose is to adjust image intensities so that it is easier to calculate and perform operations on. If F is a given image represented by matrix of pixels of intensities ranging from 0 to L-1 [15], then:

$$Y = T(X) = (L-1) \int_0^X px(x)dx \text{ ---------------- } (15)$$

Where Y and X are variables. The figure below shows the result of histogram equalization:

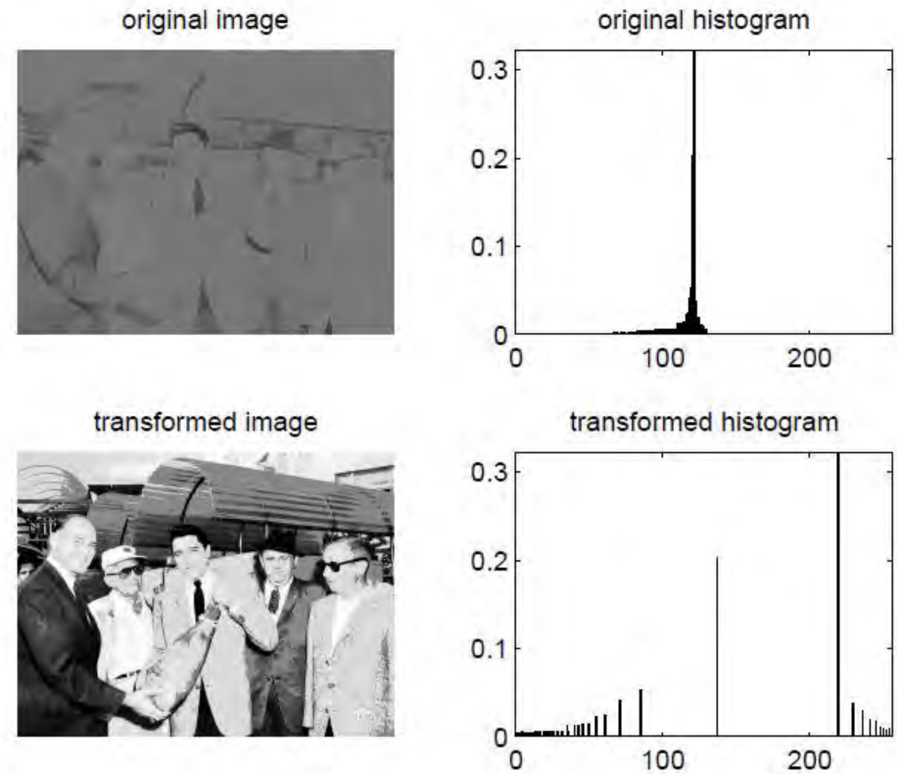

Figure 2.4- image before and after histogram equalization [15]

2.2 Image processing

It is basically a process of conversion of image into digital form with the intent of executing image operations to improve or modify it according to the needs. Due to technology, alteration of multi-dimensional signals with a variety of range allows image operations for various purposes [16]. It mainly is subdivided into three categories where the alterations take place:

• Image Processing

• Image Analysis

• Image Understanding

2.2.1 Digital image

Digital image is projected on a 2D plane where analog images are converted to digital images through sampling and quantization. The co-ordinates of the image such as a(C,D) is converted to a(R,S). The continuous image or analog image is divided into C columns and D rows and the individual co-ordinates are referred to as pixel which is assigned individual values to give a digital image. For the process of image conversion from analog to digital certain processes are required such as convolution, Fourier Analysis etc [16].

2.2.2 Fundamentals of image processing

There are various important steps for image processing as shown in the figure below:

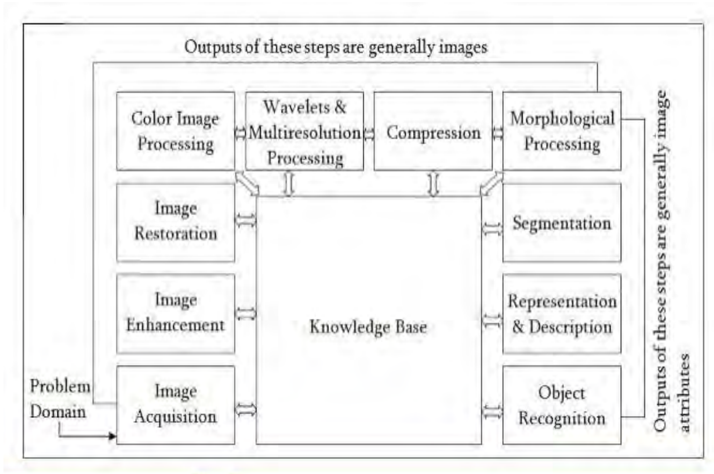

Figure 2.5- Fundamentals of image processing [17]

The steps that are usually required are explained below [17]:

- **Image Acquisition-** this is the preliminary step for digital image processing, it is basically a process of image conversion from analog to digital via scaling and quantization.
- **Image Enhancement-** the main task for this process is to make the image clearer and better and is also done to identify certain features or details of the image that were not seen before.
- **Image restoration-** it also deals with making the image better for example old pictures that decayed with time can be captured and converted into digital image and then this process is used to heighten the quality of the image. There are various ways in which image can be restored as well as a variety of algorithms.
- **Color Image Processing-** usually changes the contrast and the HSV or RGB configuration that is again used to modify and make the image more suitable for image processing to be applied.
- **Wavelets and Multi-Resolution Processing-** represents image in different resolution and is further divided into tiny portions for compression of data and represented in a pyramid.

- **Compression-**this handles technique required for memory reduction via changing the bandwith of the image.
- **Morphological Processing-** it extracts the features of an image that is advantageous to describe a certain feature of the image.
- **Segmentation-** separates the different parts of an image and groups the similar parts together and is considered a challenging phase for image processing and aids to formulate the solution of the image.
- **Representation and Description-** considered as the result of the segmentation stage, generally provides data on pixels of a certain region or the boundary of the region. Deals with obtaining features that gives quantative information.
- **Knowledge base-** involves modification of area of an image that contains the ROI [17].

2.3 MATLAB

MATLAB is the software used in this thesis paper for image analysis to detect potholes. MATLAB is a high-performance language used for technical computing. It integrates computation, visualization, and programming to create an easy-to-use environment where problems and solutions are showed in familiar mathematical notation [18]. Typical uses include:

- Math and computation
- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including Graphical User Interface building

MATLAB is an intuitive system whose fundamental information element is an array that does not require dimensioning. This enables you to take care of numerous specialized computing issues, particularly those with matrix and vector formulations, in a fraction of the time it would take to compose a program in a scalar non-interactive language, for example, C or Fortran.

The name MATLAB remains for matrix research facility. MATLAB was initially composed to give simple access to matrix software created by the LINPACK and EISPACK projects, which together speak to the best in class in software for matrix calculation.

MATLAB has advanced over a time of years with input from various clients. In university environments, it is the standard instructional tool for from the get-go and pushed courses in arithmetic, engineering, and science. In industry, MATLAB is the tool of choice for high-effectiveness research, change, and examination. MATLAB features a group of application-particular arrangements called toolboxes. Important to most clients of MATLAB, toolboxes enable you to learn and apply specific innovation. Toolboxes are far reaching accumulations of MATLAB capacities (M-records) that stretch out the MATLAB condition to tackle specific classes of issues. Zones in which toolboxes are accessible incorporate signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and numerous others [18].

2.4 Morphological Operations

Morphology is a method of image processing based on shapes. The estimation of every pixel in the output image depends on a comparison of the corresponding pixel in the input image with its neighbors. By picking the size and shape of the neighborhood, you can develop a morphological operation that is sensitive to particular shapes in the input image. This area portrays the Image Processing toolbox morphological functions. You can utilize these functions to perform basic image processing undertakings, for example, contrast enhancement, noise expulsion, thinning, skeletonization, filling, and segmentation.

Topics that are secured by this include Terminology, Dilation and Erosion, Watershed Segmentation, Morphological Reconstruction, Distance Transform, Region and Feature Measurement, Objects and Lookup Table Operations. Here, for this paper we have to know Dilation and Erosion, Structuring Elements, Dilating an Image, Eroding an Image. Morphological Reconstruction etc. concise introduction of each of these are given beneath [19].

Table 2.1: Terminology for morphological operations [20]

Terms	Definitions
Background	In a binary image, pixels that are off, i.e., set to the value 0, are considered the background. When you view a binary image, the background pixels appear black.
Connectivity	The criteria that describes how pixels in an image form a connected group. For example, a connected component is "8-connected" if diagonally adjacent pixels are considered to be touching, otherwise, it is "4-connected." The toolbox supports 2-D as well as multidimensional connectivities
Foreground	In a binary image, pixels that are on, i.e., set to the value 1, are considered the foreground. When you view a binary image, the foreground pixels appear white.
Global maxima	The highest regional maxima in the image. See the entry for regional maxima in this table for more information.
Global Minima	The lowest regional minima in the image. See the entry for regional minima in this table for more information.
Morphology	A broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. The most basic morphological operations are dilation and erosion.
Neighborhood	A set of pixels that are defined by their locations relative to the pixel of interest. A neighborhood can be defined by a structuring element or by specifying

	connectivity.
Object	A set of pixels in a binary image that form a connected group. In the context of this chapter, "object" and "connected component" are equivalent.
Packed Binary Image	A method of compressing binary images that can speed up the processing of the image.
Regional Maxima	A connected set of pixels of constant intensity from which it is impossible to reach a point with higher intensity without first descending; that is, a connected component of pixels with the same intensity value, *t*, surrounded by pixels that all have a value less than *t*.
Regional Minima	A connected set of constant intensity from which it is impossible to reach a point with lower intensity without first ascending; that is, a connected component of pixels with the same intensity value, *t*, surrounded by pixels that all have a value greater than *t*.
Structuring Element	A matrix used to define a neighborhood shape and size for morphological operations, including dilation and erosion. It consists of only 0's and 1's and can have an arbitrary shape and size. The pixels with values of 1 define the neighborhood.

2.5 Canny edge detection

Canny Edge Detector: An outstanding edge identification algorithm for its optimum outcomes is canny edge operator. It is basically an optimization problem with constraints. Three distinct criteria are tended to in this detector i.e. localization, low error rate and single response to the single edge. These parameters were implemented by utilizing the canny operator [21].

Steps in Canny Edge Detector:-

Convert to Grayscale:- First convert the image to grayscale using some type of RGB to grayscale conversion.

Noise Reduction:- Noise reduction implies some sort of blurring operations. Gaussian filter is usually used to reduce noise. Commonly used filter is 5 × 5 filter.

Compute Gradient Magnitude and Angle:- The derivatives (Dx(x, y) and Dy(x, y)) of the image in both x and y directions are calculated. Then the angle of the gradient and the gradient magnitude are also calculated [21]. The angle of gradient is computed as follows :-

$$\theta = arctan \frac{Dx(x,y)}{Dy(x,y)} \text{------------- (21)}$$

Non-Maximum Surpression :- The “non-maximal surpression” keeps only those pixels on a edge which have the maximum gradient magnitude. The maximal magnitudes should always occur right at the edge boundary, and then the gradient magnitude should always fall off with distance from the edge.So, three pixels in a 3 × 3 around pixel (x, y) are examined:

• If θ(x, y) = 0◦, then the pixels (x + 1, y), (x, y), and (x − 1, y) are to be considered.

• If θ(x, y) = 90, then the pixels (x, y + 1), (x, y), and (x, y − 1) are to be considered.

• If θ(x, y) = 45◦, then the pixels (x + 1, y + 1), (x, y), and (x − 1, y − 1) are to be considered.

• If θ(x, y) = 135◦, then the pixels (x + 1, y − 1), (x, y), and (x − 1, y + 1) are to be considered.

If pixel (x, y) has the highest gradient magnitude out of all the three pixels which are considered, it is kept as an edge. If one of the other two pixels has a higher gradient magnitude, then pixel (x, y) is not on the "center" of the edge, thus, should not be considered as an edge pixel [21].

2.6 Image Thresholding

Image Segmentation is to divide image into (continuous) regions or sets of pixels [22].

- Region Based
- Boundary Based
- Edge Based

This technique basically divides the image into continuous regions or series of pixels, Otsu's method was used in this algorithm to achieve the results. Grayscale images are transformed into Binary images. Image is broken down into foreground and background images. Intensities of regions are segmented such as light and dark regions. For thresholding of image [22], the following rule is used:

$$g(x,y) = \begin{cases} 1 \; if \; (x,y) \geq T \\ 0 \; otherwise \end{cases} \text{------------------} \; (22)$$

Where g(x, y) is the thresholded image and f(x, y) is the grayscale image.

2.7 K means Algorithm

This method groups data into a number of groups; the groups are represented by the variable K. The algorithm then assigns each group a data point based on its features and then clustered on the basis of its similarity [23]. In the results:

1) The centroids labels a new data
2) Every data point is allocated to a cluster that permits the analysis of groups formed naturally.

For grouping the following types of methods are used:

- Behavioral segmentation
- Inventory Categorization
- Sorting Sensor Measurements
- Detecting bots and anomalies [23]

2.8 Fuzzy C means Algorithm

It performs clustering by creating a set of fuzzy clusters and the center points of the clusters that signify the arrangement of the data efficiently. However the number of clusters must be specified before the data can be clustered. The function of the algorithm takes a set of data as a desired number of clusters and returns the optimal cluster center for all the data points which is used to create a fuzzy inference system by forming membership functions that symbolize the fuzzy characteristics of each cluster [24]. The equation below represents the objective function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \left\| x_i - c_j \right\|^2 \quad , 1 \leq \mathrm{m} < \infty \text{ ------ } (24)$$

where m is any real number greater than 1, u_{ij} the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimension center of the cluster, and the modulus is any norm expressing the similarity between any measured data and the center. The equation for the cluster center is:

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} * x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \text{----------------- } (24)$$

After the previous techniques were used morphological operations were used to extract the perimeter of the pothole. From the properties of the images bounding box method was used to place a rectangle over the perimeter of the pothole that was selected [24].

Chapter 3

System Implementation

This chapter outlines the arrangement and operations of the algorithms used in this paper. Each algorithm used uses a specific set of instructions that allows the pothole to be efficiently detected along with a sample to show the performance and operation of pothole detection. Before we move on to the morphological operations procedure here is a brief overview of the morphological operations used in this paper.

Dilation and Erosion

These are two fundamental morphological activities. Dilation adds pixels to the limits of objects in an image, while erosion evacuates pixels on object limits. The number of pixels taken or added from the objects in an image relies upon the size and shape of the organizing component used to process the image [25].

Concept of Dilation and Erosion

In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. The rule used to process the pixels defines the operation as dilation or erosion. This table lists the rules for both dilation and erosion [26].

Table 3.1: Rules for Grayscale and Dilation Erosion [26]

Operation	Rule
Dilation	The value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to the value 1, the output pixel is set to 1.
Erosion	The value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to 0, the output pixel is set to 0.

The accompanying figure explains the dilation of a binary image. The figure shows how the organizing element defines the neighborhood of the pixel of interest, which is circled. (See Structuring Elements for more data.) The dilation function applies the fitting guideline to the pixels in the neighborhood and assigns a value to the relating pixel in the output image. In the figure, the morphological dilation function sets the value of the output pixel to 1 since one of the elements in the neighborhood defined by the organizing element is on [26].

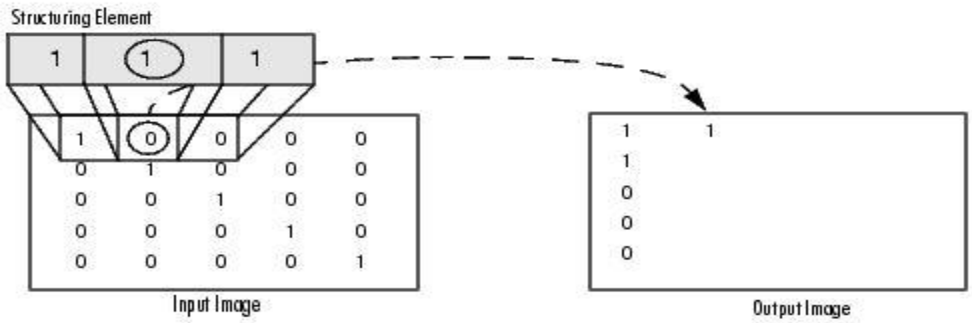

Figure 3.1- Morphological Processing of binary image [26]

The accompanying figure illustrates this processing for a grayscale image. The figure demonstrates the processing of a specific pixel in the input image. Note how the function applies the rule to the input pixel's neighborhood and utilizations the largest value of every pixels in the neighborhood as the value of the comparing pixel in the output image.

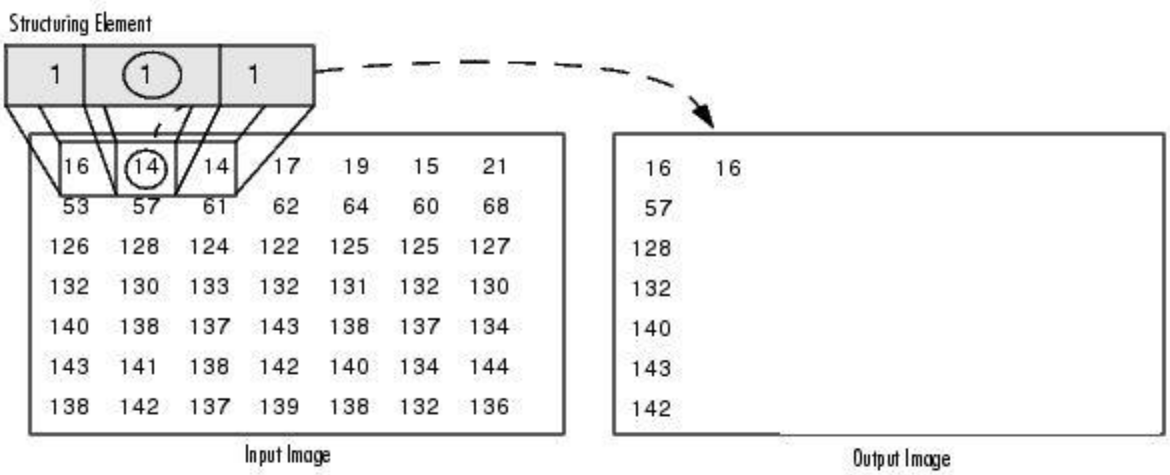

Figure 3.2: Morphological Processing for Gray Scale Images [26]

Processing Pixels at Image Borders

It positions the origin of the structuring element and its center element, over the area of interest in the input image. Pixels at the edge of the image that is defined by the structuring element can widen past the boundary of the image.

Morphological functions are assigned with a value to undefined pixels to process the pixels at the boundary of an image. The functions pads the image with additional rows and columns and the value of the ‘pads’ varies according to dilation and erosion properties. The table below shows the padding rules for dilation and erosion of both binary and grayscale images [26].

Table 3.2: Rules for Padding Images [26]

Operation	Rule
Dilation	Pixels beyond the image border are assigned the minimum value afforded by the data type. For binary images, these pixels are assumed to be set to 0. For grayscale images, the minimum value for uint8 images is 0.
Erosion	Pixels beyond the image border are assigned the maximum value afforded by the data type. For binary images, these pixels are assumed to be set to 1. For grayscale images, the maximum value for uint8 images is 255.

Structuring Elements

Structuring elements which are used to probe input images are an essential part of dilation and erosion, a matrix is formed of 0's or 1's that is two-dimensional in nature or flat. The center pixel of the structuring element is named the origin which is used to identify the pixel of interest i.e. the pixel being processed. The pixels in the structuring element containing 1's define neighborhood of the structuring element. To convert the 2D dimension to 3D of the structuring elements in the x and y plane, a z plane or height values also needs to be added [27].

The Origin of a Structuring Element

The equation or code used to find the co-ordinate values of the origin of structuring elements of any size and dimension is given by:

$$Origin = Floor * \frac{(size * nhood) + 1}{2} \text{ ----- (27)}$$

The following structure shows a diamond shaped element and its origin [27].

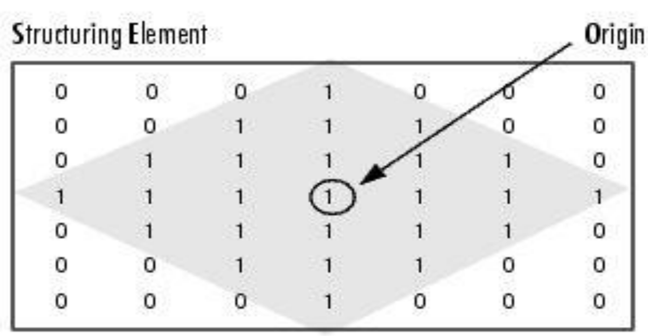

Figure 3.3: Origin of a diamond shaped structural element [27]

Forming a Structuring Element

The toolbox dilation and erosion functions acknowledge structuring component objects, called a STREL. You utilize the strel function to make STRELs of any arbitrary size and shape. The strel function likewise includes built-in support for some, basic shapes, for example, lines, jewels, plates, periodic lines, and balls [27].

Structuring Element Decomposition

To upgrade performance, the strel function may break structuring elements into little pieces, a system known as structuring element decomposition.

For instance, dilation by an 11-by-11 square structuring element can be expert by enlarging first with a 1-by-11 structuring element, and afterward with a 11-by-1 structuring element. These outcomes in a hypothetical speed change of a factor of 5.5, in spite of the fact that by and by the real speed change is to some degree less.

Structuring element decompositions utilized for the 'disk' and 'ball' shapes are approximations; every other decomposition are correct. Decomposition isn't utilized with arbitrary structuring elements, unless it is a flat structuring element whose neighborhood is every one of the 1's [27].

Dilating an Image

To dilate an image, utilize the imdilate function. The imdilate function acknowledges two essential arguments:

- The input image to be processed (grayscale, binary, or packed binary image)
- A structuring element object, returned by the strel function, or a binary matrix characterizing the area of a structuring element.

Imdilate additionally acknowledges two optional arguments: PADOPT and PACKOPT. The PADOPT contention influences the size of the output image. The PACKOPT contention recognizes the input image as packed binary [28].

The following Operations were also used during the course of this thesis.

Table 3.3- Morphological operations used in MATLAB [29]

Operation	Description
'bridge'	Bridges unconnected pixels, that is, sets 0-valued pixels to 1 if they have two nonzero neighbors that are not connected. For example: 1 0 0 1 1 0 1 0 1 becomes 1 1 1 0 0 1 0 1 1
'close'	Performs morphological closing (dilation followed by erosion).
'fill'	Fills isolated interior pixels (individual 0s that are surrounded by 1s), such as the center pixel in this pattern. 1 1 1 1 0 1 1 1 1
'open'	Performs morphological opening (erosion followed by dilation).
'thicken'	With n = Inf, thickens objects by adding pixels to the exterior of objects until doing so would result in previously unconnected objects being 8-connected. This option preserves the Euler number.

3.1 Canny Edge detection

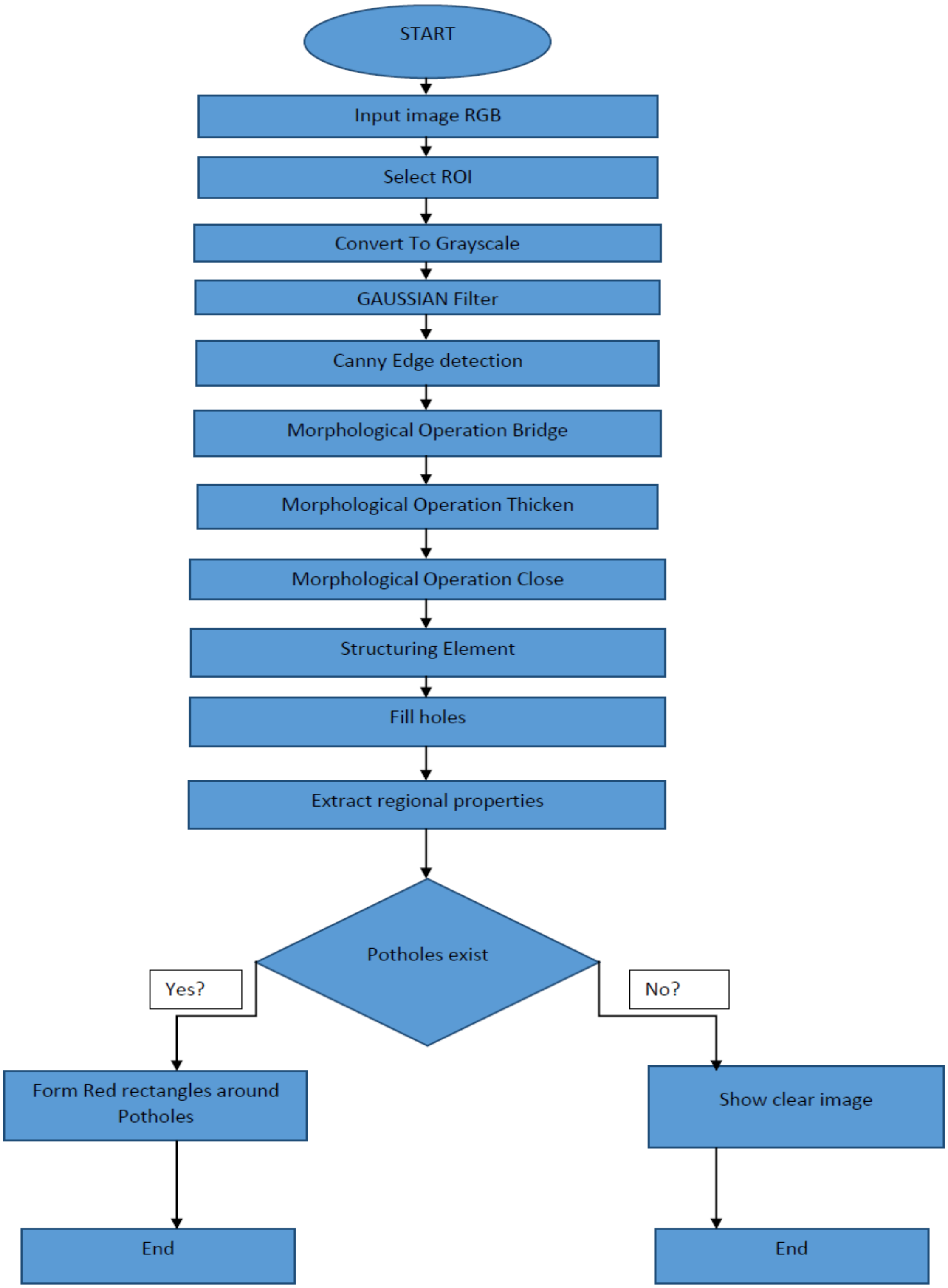

Figure 3.4 - Canny Edge detection Algorithm

This is the first algorithm used amongst the four. The tools used were MATLAB 2015Ra and image processing toolbox. After canny edge was applied, the methodology used was bridging, thickening, dilation and erosion and filling after structuring element and bounding box at the end to produce the pothole [29].

Firstly image pre-processing was done with the image as shown in the flowchart in fig 3.4. Region of interest (ROI) was selected from the input image and then was converted to grayscale from RGB. Then Gaussian filter was applied. The image below shows the result.

Figure 3.5- Original Image

Figure 3.6- Image after ROI and RGB to gray

Figure 3.7- Image after Gaussian filter is applied

After the Gaussian filter was applied, canny edge detection which is basically edge identification algorithm for its optimum outcomes is canny edge operator. It is basically an optimization problem with constraints. Three distinct criteria are tended to in this detector i.e. localization, low error rate and single response to the single edge. After it was carried out it was followed by Morphological Operation Bridge thicken and close. The bridge operation connects the binary values of zeros and ones of the pixels of the image connecting them together [29]; Thickening basically thickens objects by adding more pixels to the exterior of the objects and closing is used to get a smooth shape of the pothole which is done by dilation and erosion [29]. Structuring elements was then performed to recognize the potholes while filling was performed that fill the gaps in the pixels to get the complete circumference or region boundary identifying the potholes. After that bounding box was used from regional properties that places a rectangle around the pothole. The following images show the steps and operations in details.

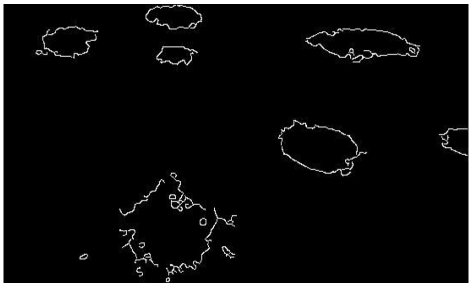

Figure 3.8- Image after canny edge detection was applied

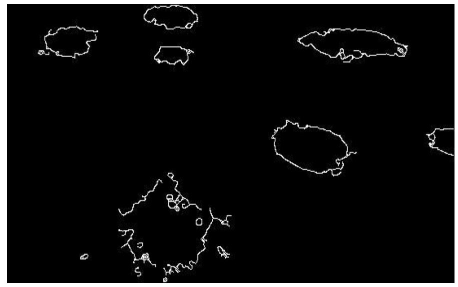

Figure 3.9- Image after bridge operation was applied

Figure 3.10- Image after thickening and closing was applied

Figure 3.11- Image after thickening and closing was applied

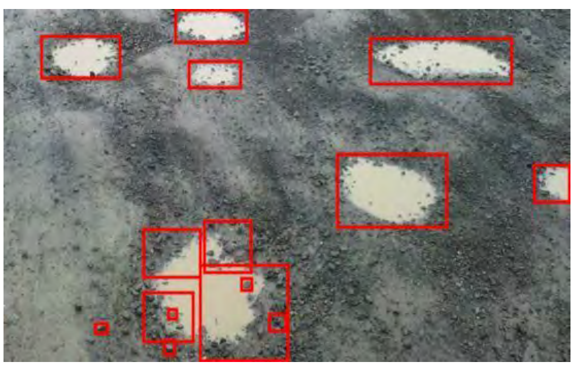

Figure 3.12- Final Output image after bounding box was applied

3.2 Image Thresholding for Pothole Detection

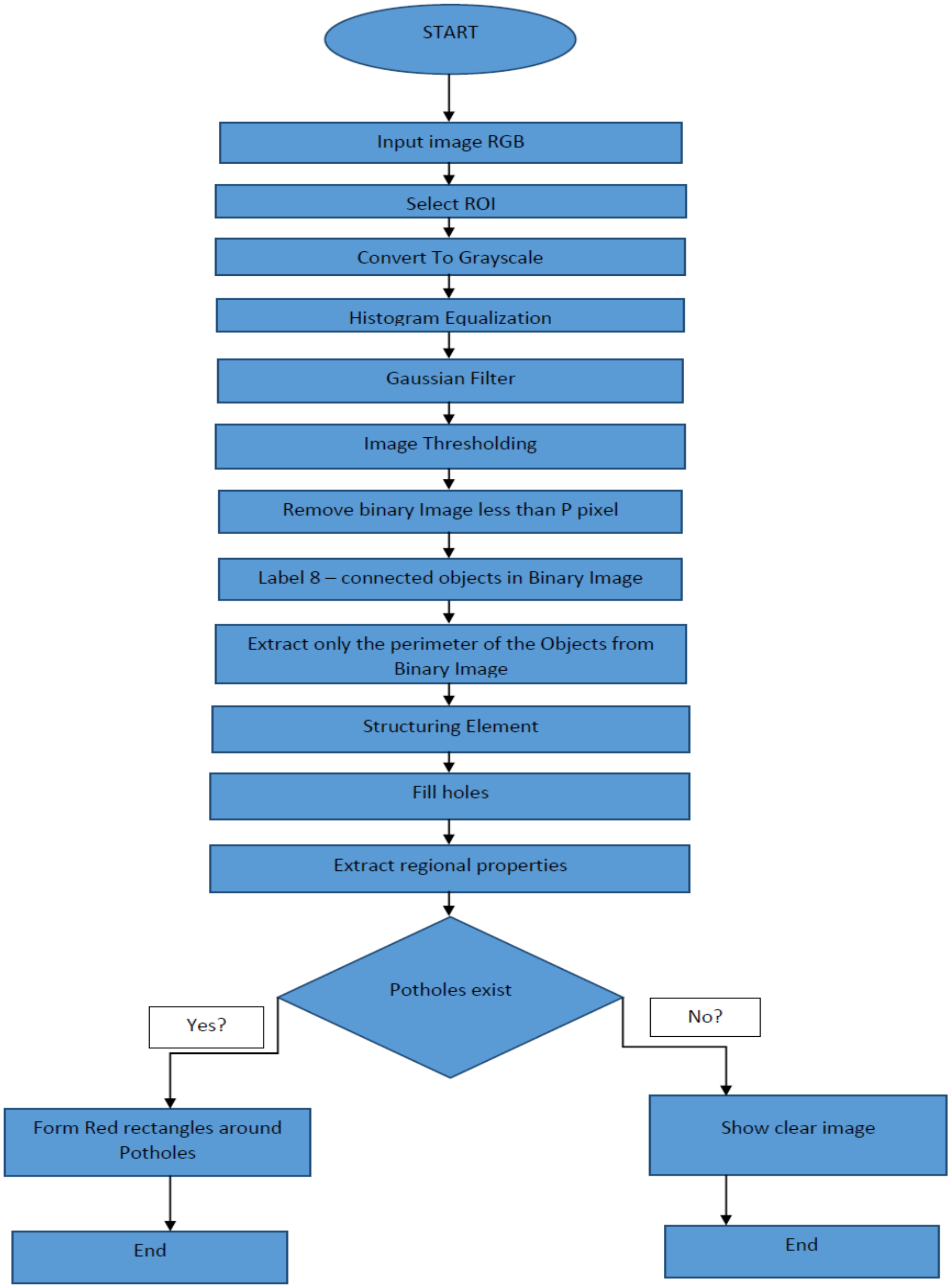

Figure 3.13 - Image Thresholding Algorithm

This is the second algorithm used using the same software (MATLAB 2015Ra) and image processing toolbox. After Image thresholding was applied, the operations used for this algorithm are binary image area open (represented by pixels), morphological operation label, Structuring element, fill operation, bounding box etc.

The flow chart for this algorithm is shows step by step techniques and operations used to understand and learn which process comes after which aiding the understanding of Image Thresholding algorithm.

Firstly image pre-processing was done after the image was taken as the input; after ROI was selected the image was converted to grayscale for better operation of image. After histogram equalization and Gaussian filter was applied the Image threshold algorithm was applied to extract the potholes. The following image shows the results of the operations described.

Figure 3.14- Original image

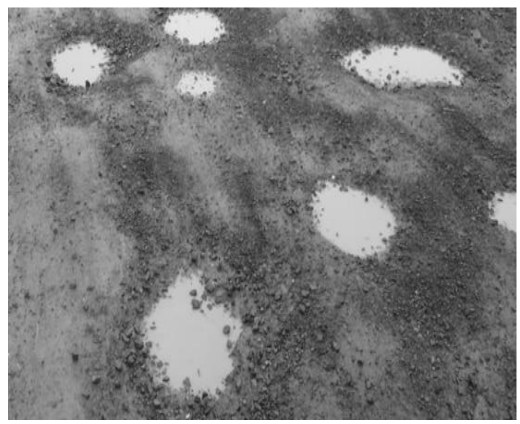

Figure 3.15- Image after selecting ROI and RGB to gray conversion

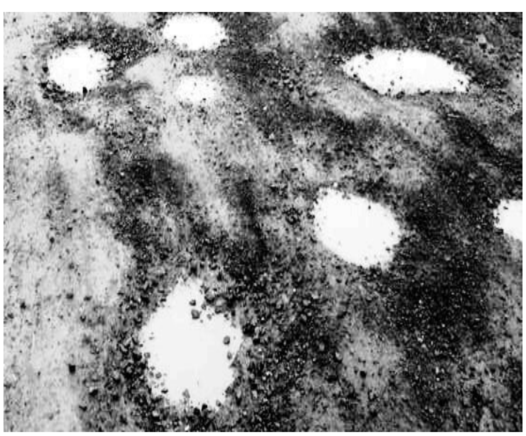

Figure 3.16- Image after histogram equalization

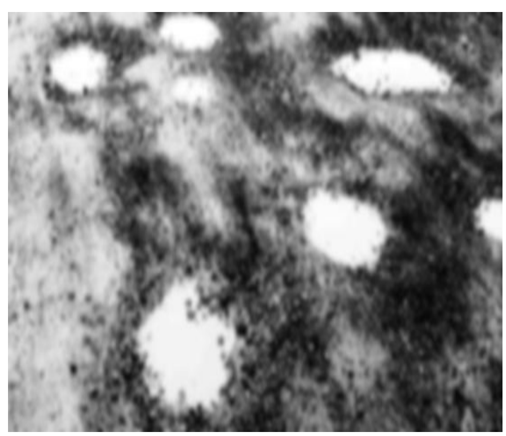

Figure 3.17- Image after Gaussian filter was applied

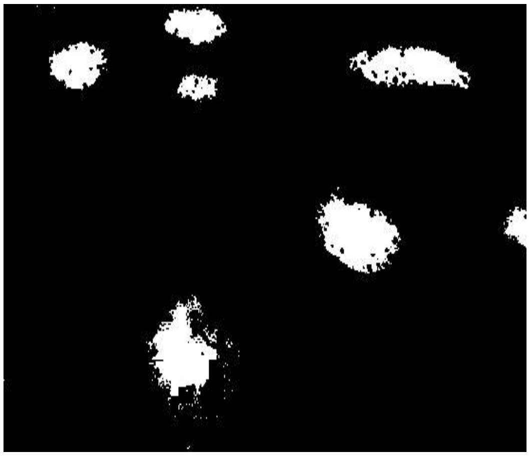

Figure 3.18- Image Thresholding result

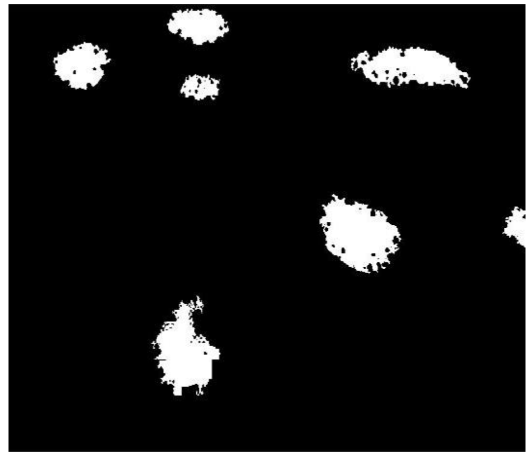

Figure 3.19- After filtering out noise

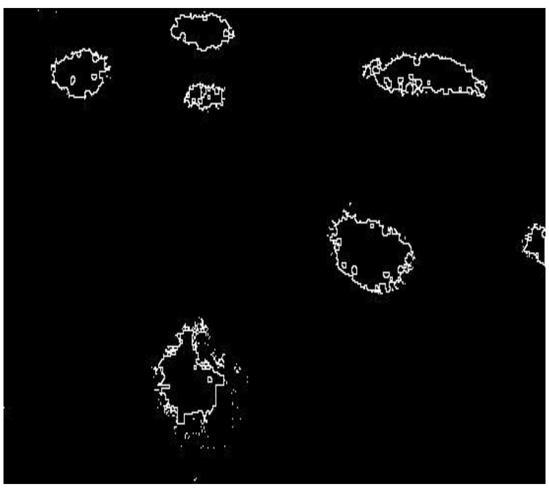

Figure 3.20- Image after labeling operation and extracting the perimeter of the pothole

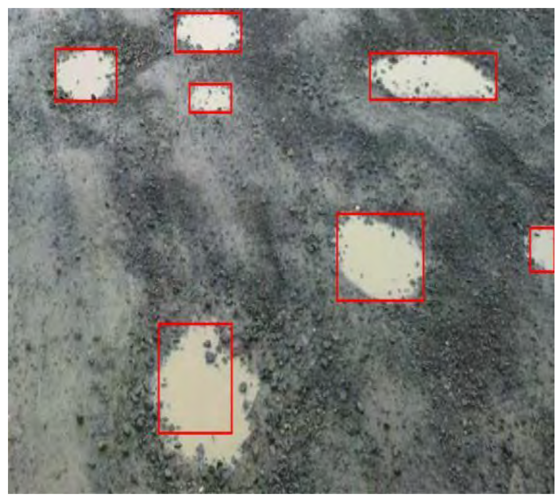

Figure 3.21- Final image result

Image thresholding converts grayscale images are transformed into Binary images. Image is broken down into foreground and background images. Intensities of regions are segmented such as light and dark regions.

After image thresholding was applied, binary image area open function [30] was applied that filters out noise and potholes detected below a specific set of pixels which were than labeled to give figure 3.20. This image coupled with the image from the ROI gives us the final image that successfully identifies the pothole. Using bounding box method we are able to place a rectangular box around the pothole aiding the visual understanding of the location of potholes.

3.3 K-means for Pothole Detection

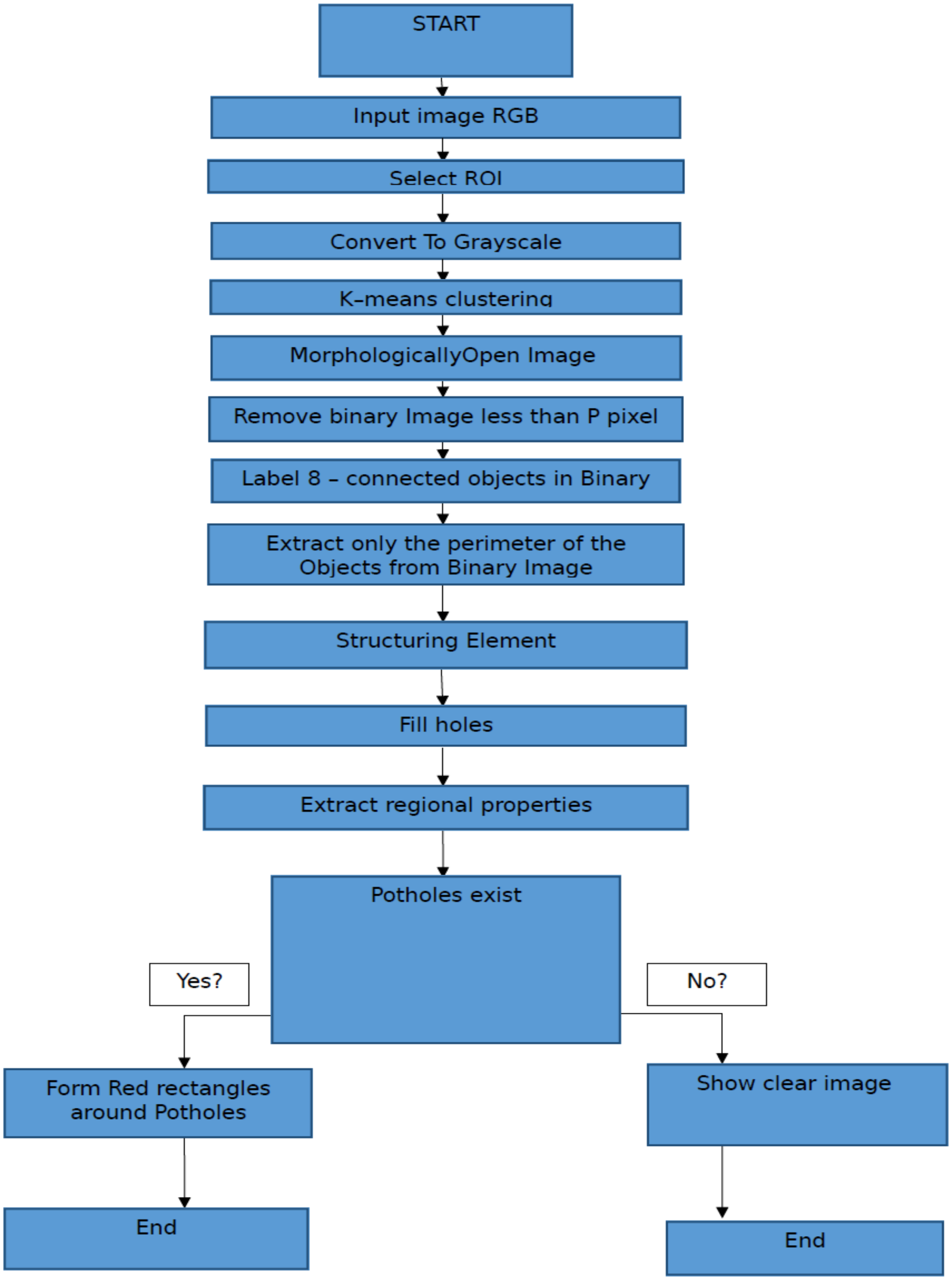

Figure 3.22 - K means clustering Algorithm

This is the third algorithm used using the same software (MATLAB 2015Ra) and image processing toolbox. After K means clustering was done, operations used for this algorithm was open image operation, binary image open area operation, labeling [31], structuring elements, filling, and bounding box.

Like previous methods, image pre-processing was performed where the image was converted to grayscale after ROI was selected. Then K means algorithm was applied followed by open image operation which is basically erosion followed by dilation, then open area function is applied to remove noise which is then recognized using structural element and then fill operation used to cut out only the shape of the pothole which is then identified using bounding box that places a rectangular shape onto the potholes for the users to identify it with comfort. The image of each steps are shown below:

Figure 3.23- Original image

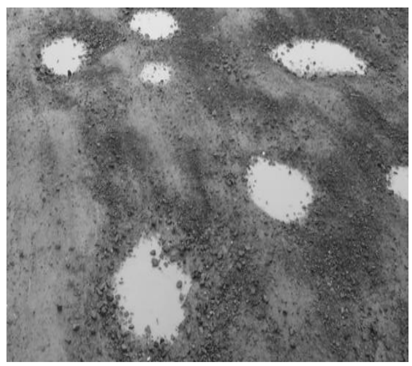

Figure 3.24- Image after ROI selection and Grayscale conversion

Figure 3.25-Image after K means algorithm performed

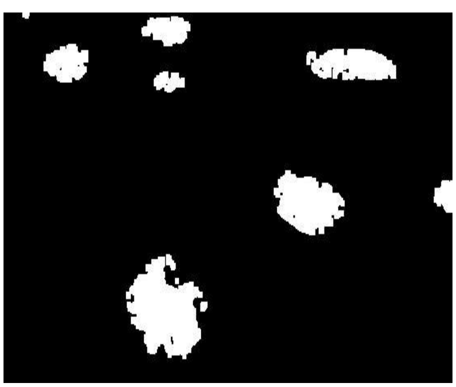

Figure 3.26- Image after Open image operation

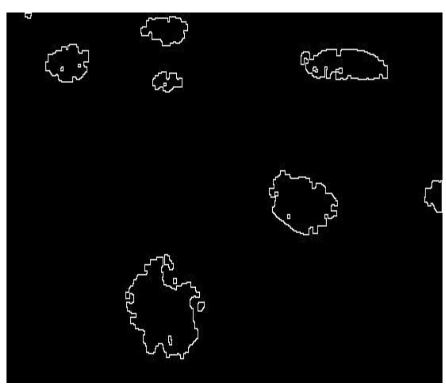

Figure 3.27- Image after labeling operation and fill operation

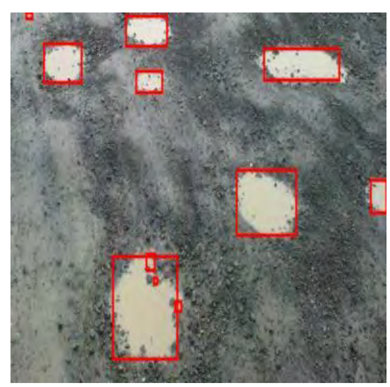

Figure 3.28- Final Image

3.4 Fuzzy C means for Pothole Detection

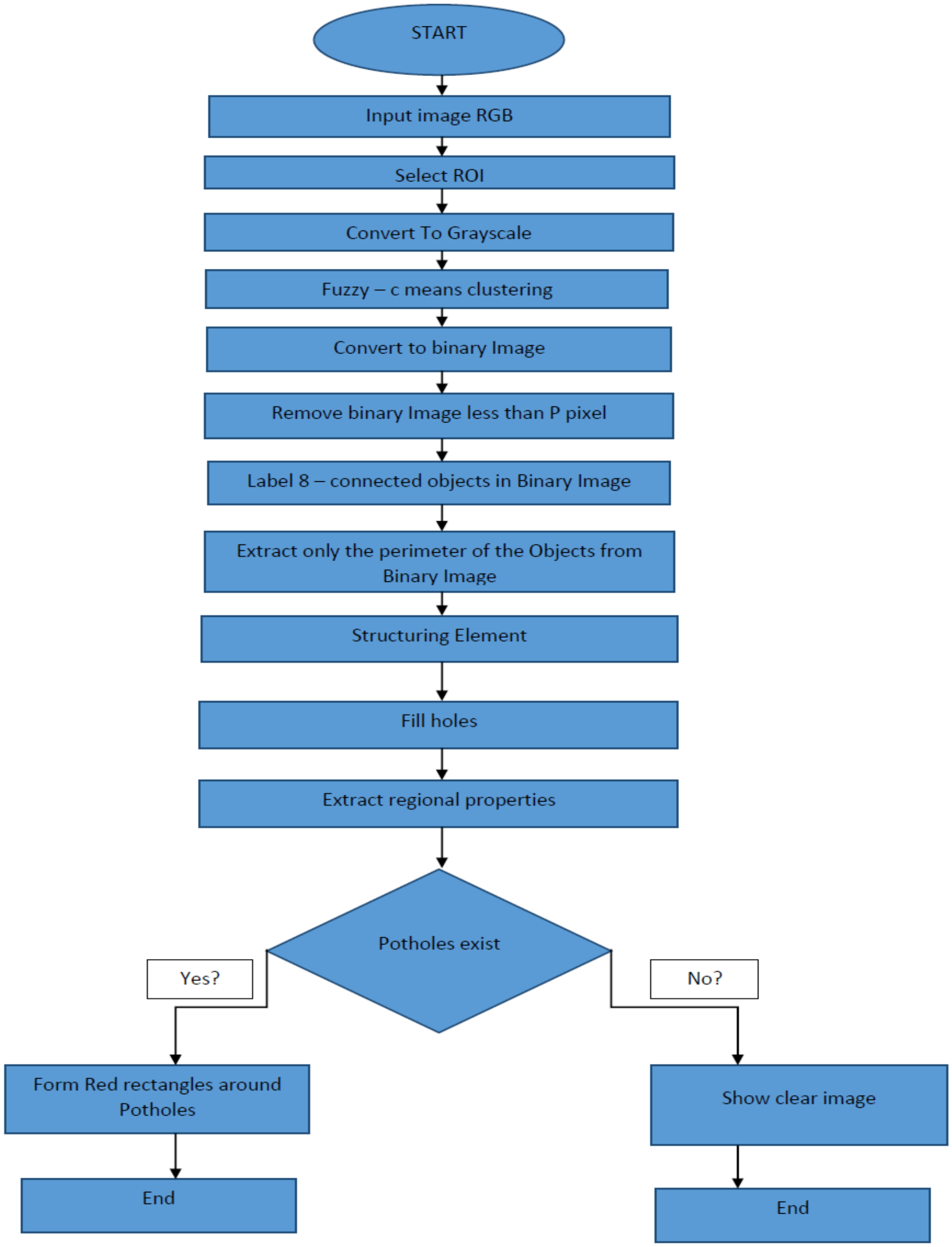

Figure 3.29 - Fuzzy C means Algorithm

This is the last algorithm used using the same software (MATLAB 2015Ra) and image processing toolbox. Binary image conversion, labeling, perimeter extraction, structuring element, filing and bounding box are the operations performed on it after the algorithm was applied.

The methodology follows a similar pattern to those of the previous algorithms, i.e. the image is taken and the ROI selected, then it is converted to grayscale after which the algorithm was applied. Afterwards all the operations mentioned above were applied. Labeling and perimeter extraction [31] helps cancel out noise and give a consistent shape to the area of the pothole. Structuring element and filing takes the boundary of the area of the potholes to be shown separately which allows pothole detection to take place with the aid of bounding box operations which draws a rectangle over the area of the pothole. The following diagrams show the following result step by step.

Figure 3.30- Original Image

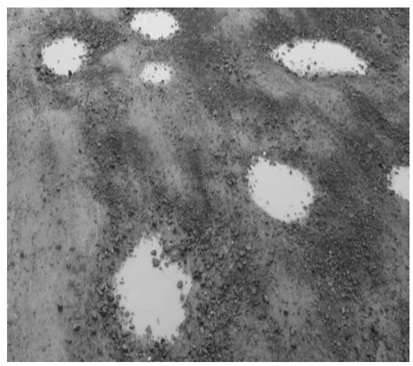

Figure 3.31- Grayscale Image with ROI selection

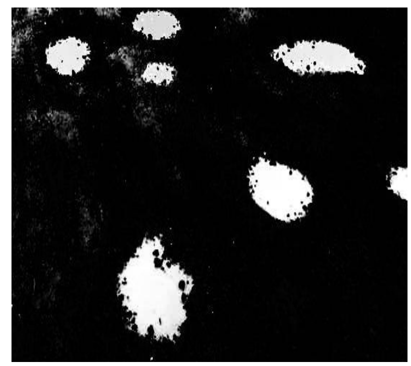

Figure 3.32- Fuzzy C means applied result

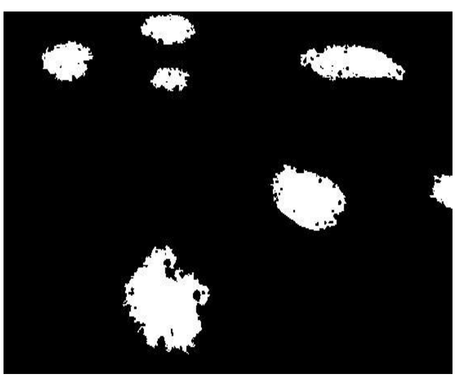

Figure 3.33- Converted Binary Image

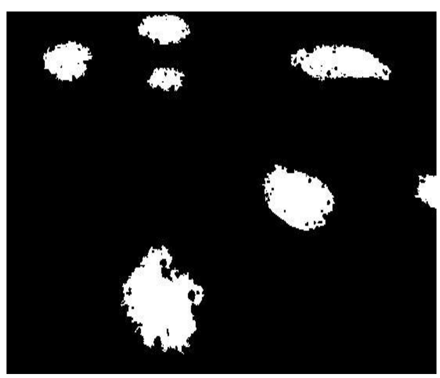

Figure 3.34- Image after labeling and perimeter extraction

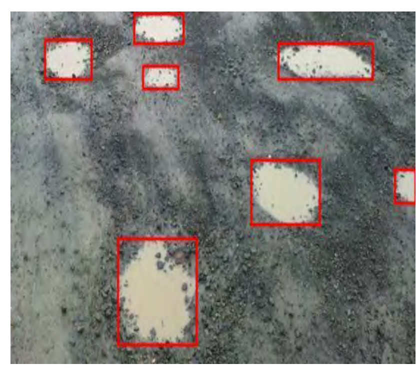

Figure 3.35- Final image of Fuzzy C means algorithm

Chapter 4

Experimental result

First 30 photos of potholes were taken from internet and the following techniques were tried out. Most of the photos from internet do not include situations or conditions that we regularly see in Bangladesh. As it was necessary to perform a detailed study and see how much effective the techniques are in those conditions. Hence another sample of 30 photos were captured which consisted of 4 different environments and the following algorithms were applied. The following flow charts below shows the algorithm of each of the methods used. Therefore each algorithm is tried on a total of 60 pictures, 30 from the internet images to represent ideal conditions and 30 from real life image showing non-ideal characteristics. The images below show the 4 different conditions of the potholes:

Figure 4.1- Potholes filled with sand or polyethene or any other disturbance.

Figure 4.2 - Potholes which are consist of bump or rocks.

Figure 4.3 - Potholes in muddy or earth road

Figure 4.4 - Potholes in wet road

In some of the images, lane detection was used to filter out the lines. The four different algorithms used in this paper were tested in terms of accuracy and precision and cross matched to see the difference in results between ideal situation and real life potholes. The following formulas were used to measure the performance of the potholes.

$$Accuracy(Acc) = \frac{(TP+TN)}{N} \text{------------------ (32)}$$

$$Precision\ (P) = \frac{TP}{(TP+FP\ )} \text{----------------- (32)}$$

Where,

TP-number of potholes that are correctly detected,

TN – number of potholes that are correctly rejected,

FP- number of potholes that are incorrectly identified,

FN-number of potholes that are incorrectly rejected,

N – Total number = TN+TP+FN+FP

In these paper it is considered that TN=0.

4.1 Image Thresholding

Image thresholding results for internet images and database are shown in the tables. It shows the number of potholes detected and not detected and the number of potholes that were falsely detected.

4.1.1 Internet images

Thirty pictures from the internet were used here on which image thresholding was performed along with the morphological operations as explained in chapter 3. The results for the potholes detected are shown below.

Table 4.1- Image Thresholding results from Internet images

No. of Potholes	Detected	False detected	**Not detected**
7	7	0	0
1	1	0	0
14	12	1	2

7	6	0	0
3	3	1	0
14	13	0	1
1	1	1	0
9	7	0	2
1	1	1	0
5	3	0	2
1	1	0	0
1	1	0	0
1	1	0	0
1	1	2	0
1	1	0	0
2	2	0	0
11	11	3	0
8	8	0	0
12	10	0	2
7	7	0	0
2	2	0	0
12	9	0	3
1	0	2	1
10	8	0	2

4	3	0	1
14	8	1	6
1	1	0	0
7	6	0	1

4.1.2 From database

Images taken using HD camera was used to obtain pictures of the roads that formed the database of our own which represents more realistic scenario of roads as opposed the internet images. Thresholding was then applied using the steps shown and described in chapter 3. The results of the potholes of each of the 30 image taken are shown below.

Table 4.2- Image thresholding results from images taken in Bangladesh

No. of potholes	Detected	False Detected	Not detected
2	2	12	0
1	1	3	0
2	2	0	0
11	11	1	0
10	10	1	0
15	12	4	3
15	12	4	3
6	5	8	1
3	2	4	1
7	7	0	0
1	0	4	1

1	1	4	0
3	1	2	2
1	0	6	1
1	1	6	0
1	0	7	1
1	1	7	0
1	1	0	0
4	4	0	0
2	0	4	2
3	3	6	0
4	4	4	0
4	4	5	0
1	0	8	1
1	1	5	0
3	0	4	3
8	6	6	2
7	5	8	2

4.1.3 Calculations and result

The following graph is displayed to compare analysis of internet images with database images to show the difference in accuracy and precision. The results are shown as Bar charts showing accuracy and precision, the method for calculations are shown below the graphs.

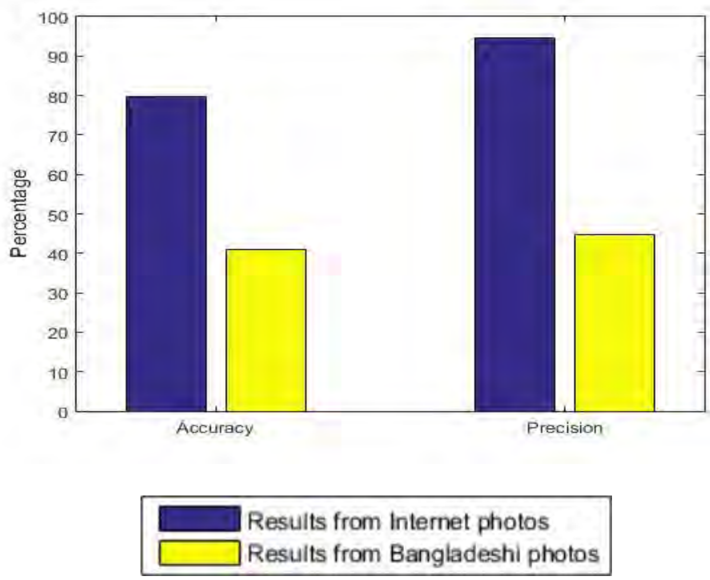

Figure 4.5- Difference in accuracy and precision for image thresholding

- **Calculations**

For **Internet images** i.e. images that show ideal conditions with no anomalies in the captured pictures, we have:

Table 4.3- Image thresholding final result (Internet image)

Number of Total Potholes	**158**
Total Number of potholes detected	138
Total number of potholes falsely detected	11
Total Number of potholes not detected	23

Therefore, using the formula below, the calculated accuracy and precision are:

$$Accuracy(Acc) = \frac{(TP+TN)}{N} \qquad Precision\ (P) = \frac{TP}{(TP+FP\ )}$$

Accuracy = 79.88%

Precision = 94.47%

For **Image from our database** i.e. images that have a lot of varying factors such as sand on the road along with water and oil etc, the results obtained from using the same algorithm on MATLAB gives us:

Table 4.4 – Image Thresholding final result (database)

Number of Total Potholes	**121**
Total Number of potholes detected	100
Total number of potholes falsely detected	123
Total Number of potholes not detected	21

Therefore, using the same formula, the calculated accuracy and precision are:

Accuracy = 40.98%

Precision = 44.84%

This shows us that the accuracy and precision as shown drops for pothole detection in real life conditions.

4.2 Canny Edge Detection

Canny Edge Detection results for internet images and database are shown in the tables. It shows the number of potholes detected and not detected and the number of potholes that were falsely detected.

4.2.1 Internet images

Thirty pictures from the internet were used here on which Canny edge Detection was performed along with the morphological operations as explained in chapter 3. The results for the potholes detected are shown below.

Table 4.5- Canny edge detection results for Internet images

No. of Potholes	Detected	False detected	Not detected
14	14	0	0
1	1	2	0

7	7	0	0
14	14	1	0
5	4	1	1
3	3	4	0
9	9	0	0
1	1	5	0
5	5	0	0
1	1	0	0
4	3	0	1
2	2	1	0
1	1	0	0
1	1	2	0
1	1	0	0
2	2	2	0
11	10	0	1
8	8	0	0
12	12	0	2
7	6	0	1
2	2	0	0
7	7	0	0
1	1	0	1

6	6	0	2
10	10	6	1
14	14	0	6
1	1	2	0
7	7	0	1
1	1	5	0

4.2.2 From Database

Images taken using HD camera was used to obtain pictures of the roads that formed the database of our own which represents more realistic scenario of roads as opposed the internet images. Canny Edge Detection was then applied using the steps shown and described in chapter 3. The results of the potholes of each of the 30 image taken are shown below.

Table 4.6- Canny Edge Detection results for images from Bangladesh

No. of potholes	Detected	False Detected	Not detected
2	2	8	0
1	1	6	0
2	2	4	0
11	11	0	0
10	10	0	0
15	10	0	5
15	10	0	5
6	6	4	0
3	3	1	0

7	2	0	5
1	1	1	0
1	1	0	0
3	3	0	0
1	1	3	0
1	1	0	0
1	1	0	0
1	1	9	0
1	1	0	0
4	4	0	0
2	2	5	2
3	3	7	0
4	4	7	0
4	4	2	0
1	1	4	0
1	1	0	0
3	3	1	0
8	8	4	0
7	7	10	0

4.2.3 Calculations and result

The following graph is displayed to compare analysis of internet images with database images to show the difference in accuracy and precision. The results are shown as Bar charts showing accuracy and precision, the method for calculations are shown below the graphs.

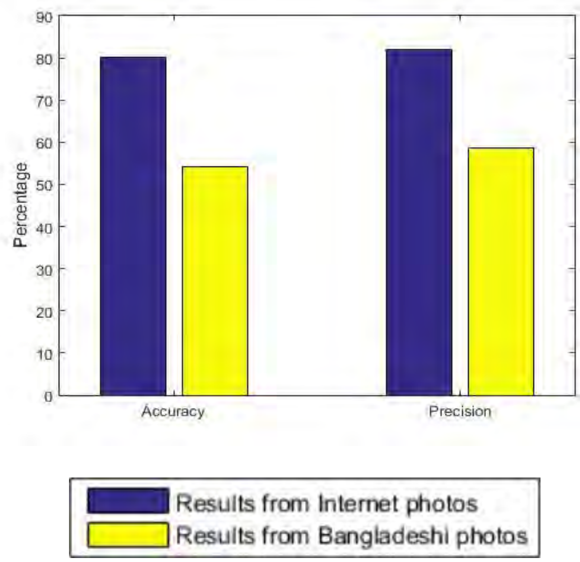

Figure 4.6- Difference in accuracy and precision for Canny Edge Detection

- **Calculations**

For **Internet images** i.e. images that show ideal conditions with no anomalies in the captured pictures, we have:

Table 4.7- Canny edge final result (Internet images)

Number of Total Potholes	**158**
Total Number of potholes detected	154
Total number of potholes falsely detected	34
Total Number of potholes not detected	4

Therefore, using the formulas shown, the calculated accuracy and precision are:

$Accuracy(Acc) = \frac{(TP+TN)}{N}$ $\quad Precision\ (P) = \frac{TP}{(TP+FP\ )}$

Accuracy = 80.21%

Precision = 81.92%

For **Image from our database** i.e. images that have a lot of varying factors such as sand on the road along with water and oil etc, the results obtained from using the same algorithm on MATLAB gives us:

Table 4.8 – Canny edge final result (database)

Number of Total Potholes	**121**
Total Number of potholes detected	108
Total number of potholes falsely detected	76
Total Number of potholes not detected	13

Therefore, using the same formulas above, the calculated accuracy and precision are as follows:

Accuracy = 54.27%

Precision = 58.70%

This shows us that the accuracy and precision as shown drops for pothole detection in real life conditions.

4.3 Fuzzy C means

Fuzzy C means Algorithm results for internet images and database are shown in the tables. It shows the number of potholes detected and not detected and the number of potholes that were falsely detected.

4.3.1 Internet Images

Thirty pictures from the internet were used here on which Fuzzy C means was performed along with the morphological operations as explained in chapter 3. The results for the potholes detected are shown below.

Table 4.9- Fuzzy C means result for Internet images

No. of Potholes	Detected	False detected	Not detected
14	14	0	0

1	1	2	0
7	7	0	0
14	13	1	1
5	4	1	1
3	2	4	1
9	8	0	1
1	1	5	0
5	5	0	0
1	1	0	0
4	3	0	1
2	2	1	0
1	1	0	0
1	1	2	0
1	1	0	0
2	2	2	0
11	8	0	3
8	8	0	0
12	11	0	1
7	5	0	2
2	2	0	0
7	7	0	0
1	1	0	0
6	6	0	0
10	6	6	4

14	14	0	0
1	1	0	0
7	7	0	0
1	1	0	0

4.3.2 From Database

Images taken using HD camera was used to obtain pictures of the roads that formed the database of our own which represents more realistic scenario of roads as opposed the internet images. Fuzzy C means was then applied using the steps shown and described in chapter 3. The results of the potholes of each of the 30 image taken are shown below.

Table 4.10- Fuzzy C means result for images taken from Bangladesh

No. of potholes	Detected	False Detected	Not detected
2	2	2	0
1	1	4	0
2	2	0	0
11	11	0	0
10	10	0	0
15	11	0	4
15	11	0	4
6	4	0	2
3	2	0	1
7	7	0	0
1	1	0	0

1	1	3	0
3	3	2	0
1	1	3	0
1	1	3	0
1	1	3	0
1	1	8	0
1	1	0	0
4	4	0	0
2	1	0	1
3	1	2	2
4	2	2	2
4	3	3	1
1	1	0	0
1	1	1	0
3	2	1	1
8	5	5	3
7	2	4	5

4.3.3 Calculations and result

The following graph is displayed to compare analysis of internet images with database images to show the difference in accuracy and precision. The results are shown as Bar charts showing accuracy and precision, the method for calculations are shown below the graphs.

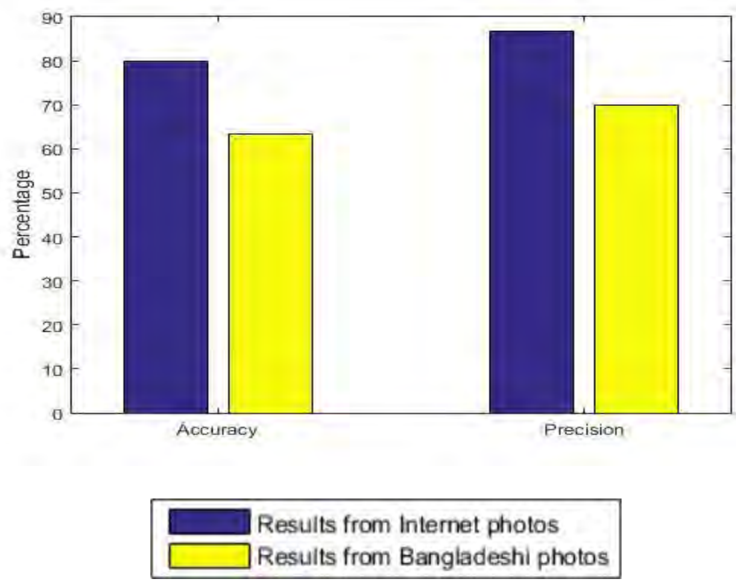

Figure 4.7- Difference in accuracy and precision for Fuzzy C means Algorithm

- **Calculations**

For **Internet images** i.e. images that show ideal conditions with no anomalies in the captured pictures, we have:

Table 4.11 – Fuzzy C means final result (Internet images)

Number of Total Potholes	**158**
Total Number of potholes detected	142
Total number of potholes falsely detected	20
Total Number of potholes not detected	16

Therefore, using the formulas shown, the calculated accuracy and precision are as follows:

$Accuracy(Acc) = \frac{(TP+TN)}{N}$ $Precision\ (P) = \frac{TP}{(TP+FP\,)}$

Accuracy = 79.78%

Precision = 86.71%

For **Image from our database** i.e. images that have a lot of varying factors such as sand on the road along with water and oil etc, the results obtained from using the same algorithm on MATLAB gives us:

Table 4.12 – Fuzzy C means final result (database)

Number of Total Potholes	**121**
Total Number of potholes detected	105
Total number of potholes falsely detected	45
Total Number of potholes not detected	16

Therefore, using the same formula above, the calculated accuracy and precision are as follows:

Accuracy = 63.25%

Precision = 70.0%

This shows us that the accuracy and precision as shown drops for pothole detection in real life conditions.

4.4 K means Algorithm

K means Clustering Algorithm results for internet images and database are shown in the tables. It shows the number of potholes detected and not detected and the number of potholes that were falsely detected.

4.4.1 Internet Images

Thirty pictures from the internet were used here on which K means Algorithm was performed along with the morphological operations as explained in chapter 3. The results for the potholes detected are shown below.

Table 4.13- K means Clustering result for Internet images

No. of Potholes	Detected	False detected	Not detected
14	12	0	2

1	1	0	0
7	7	0	0
14	14	1	0
5	4	1	1
3	3	0	0
9	8	0	1
1	1	0	0
1	1	1	0
5	5	1	0
2	2	0	0
1	1	0	0
1	1	0	0
1	1	0	0
2	2	3	0
11	11	0	0
8	8	0	0
12	9	0	3
7	6	1	1
3	2	0	1
7	7	1	0
1	1	0	0
4	4	1	0
6	4	1	2
10	6	0	4

14	10	0	4
7	5	0	2
1	1	0	0

4.4.2 From Database

Images taken using HD camera was used to obtain pictures of the roads that formed the database of our own which represents more realistic scenario of roads as opposed the internet images. K means Algorithm was then applied using the steps shown and described in chapter 3. The results of the potholes of each of the 30 image taken are shown below.

Table 4.14- K means Clustering result for images taken from Bangladesh

No. of potholes	Detected	False Detected	Not detected
2	2	3	0
1	1	3	0
2	2	0	0
11	11	0	0
10	10	0	0
15	9	0	6
15	9	0	6
6	6	6	0
3	2	4	1
7	7	0	0
1	1	0	0
1	1	5	0

3	1	1	2
1	1	3	0
1	1	4	0
1	1	5	0
1	1	7	0
1	1	3	0
4	4	3	0
2	2	0	0
3	3	6	0
4	4	5	0
4	4	3	0
1	1	1	0
1	1	3	0
3	2	7	1
8	4	1	4
7	4	10	3

4.4.3 Calculations and result

The following graph is displayed to compare analysis of internet images with database images to show the difference in accuracy and precision. The results are shown as Bar charts showing accuracy and precision, the method for calculations are shown below the graphs.

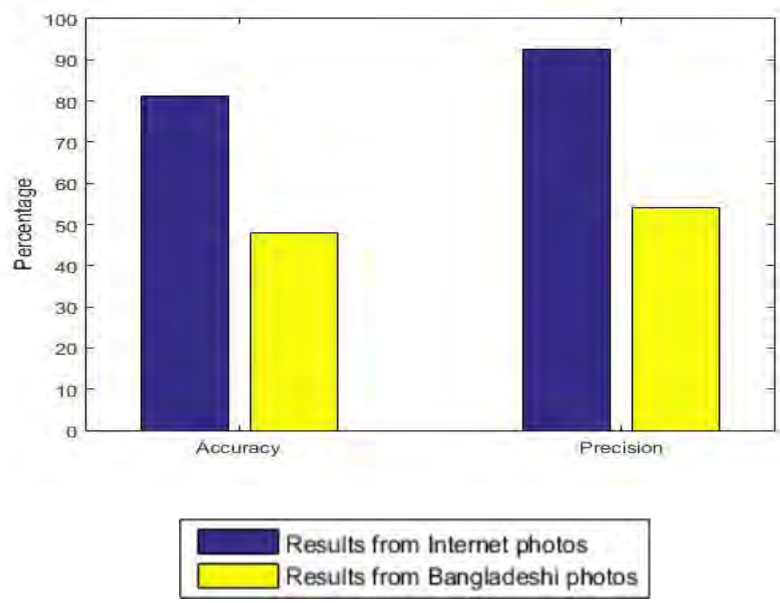

Figure 4.8- Difference in accuracy and precision for K means Clustering

- **Calculations**

For **Internet images** i.e. images that show ideal conditions with no anomalies in the captured pictures, we have:

Table 4.15 – K means final result (Internet images)

Number of Total Potholes	**158**
Total Number of potholes detected	137
Total number of potholes falsely detected	11
Total Number of potholes not detected	21

Therefore, using the formulas shown, the calculated accuracy and precision are as follows:

$Accuracy(Acc) = \frac{(TP+TN)}{N}$ $Precision\ (P) = \frac{TP}{(TP+FP\ )}$

Accuracy = 81.07%

Precision = 92.56%

For **Image from our database** i.e. images that have a lot of varying factors such as sand on the road along with water and oil etc, the results obtained from using the same algorithm on MATLAB gives us:

Table 4.16 – K means final result (database)

Number of Total Potholes	**121**
Total Number of potholes detected	98
Total number of potholes falsely detected	83
Total Number of potholes not detected	23

Therefore, using the same formulas above, the calculated accuracy and precision are as follows:

Accuracy = 48.04%

Precision = 54.14%

This shows us that the accuracy and precision as shown drops for pothole detection in real life conditions.

4.5 Comparison and analysis of algorithms

The data from the previous topic was made and formed into bar charts shown below to give a better understanding of which algorithm is more efficient and why. The following figure 4.9 shows the accuracy values of the four algorithms from the pictures that represent ideal conditions i.e. the internet images. Figure 4.10 shows the comparison of precision values for internet images between the four algorithms taken from pictures with ideal conditions to detect potholes i.e. from internet images. Figure 4.11 shows the accuracy results of the four images that represent more realistic conditions which are the images we took of the roads i.e. from our database. The final figure of this chapter, figure 4.12 shows the results of images that show precision of the more realistic conditions of potholes i.e. the images taken by our group. The diagrams are given below:

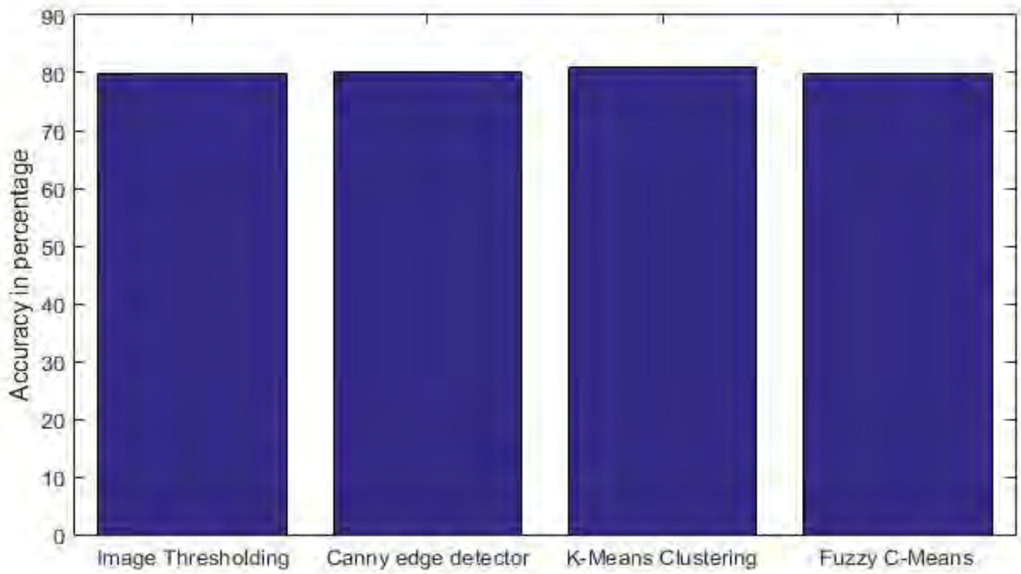

Figure 4.9- Comparison of accuracy values for different algorithms on Internet images

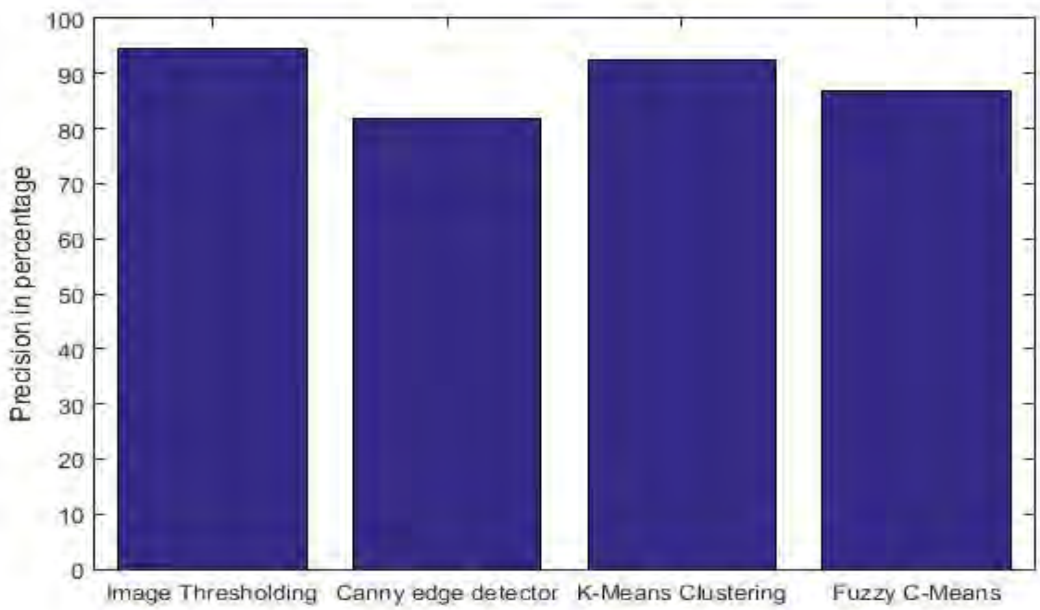

Figure 4.10- Comparison of precision values for different algorithms on Isnternet images

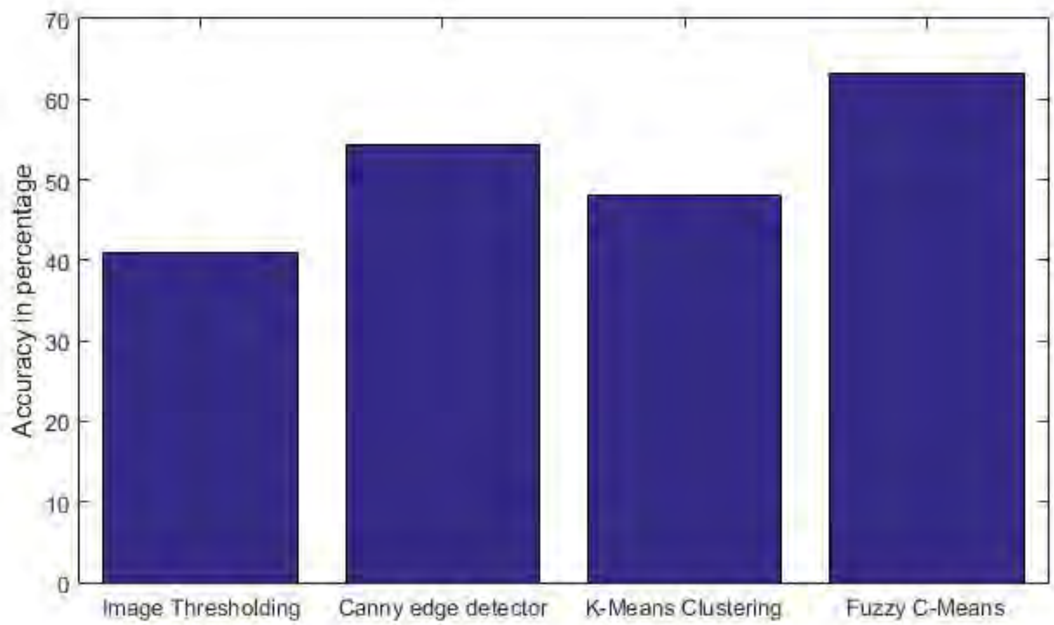

Figure 4.11- Comparison of accuracy values for different algorithms from our database.

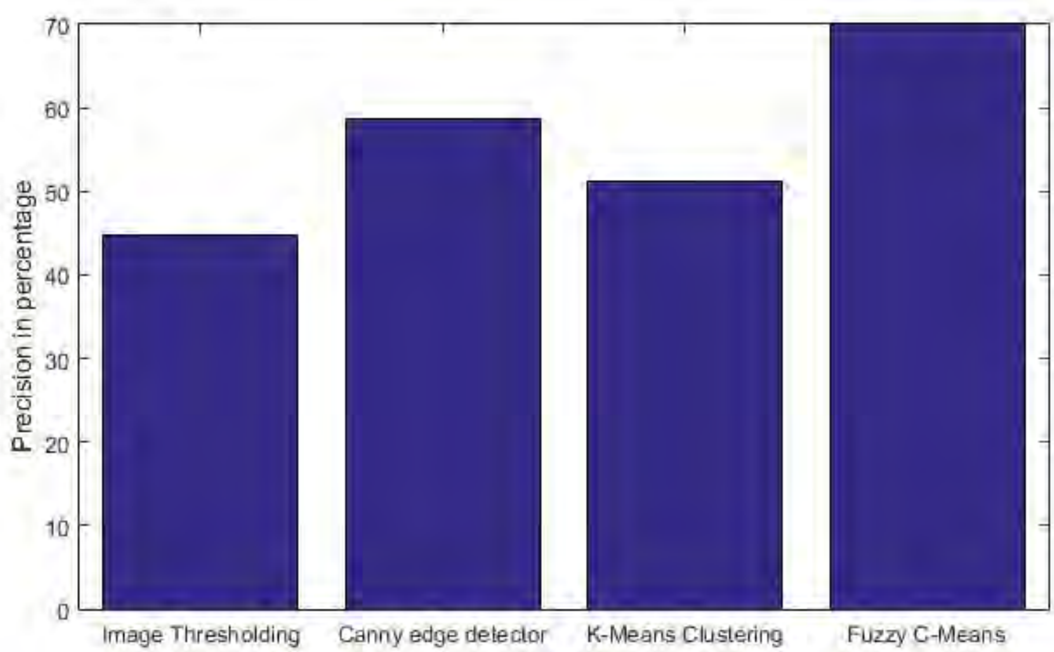

Figure 4.12 - Comparison of precision values for different algorithms from our database.

As we can see from these graphs there was considerable decrease in accuracy for almost all the algorithms, this show to say that the accuracy and precision drops immensely when the result was tried in real life conditions in various different environments as compared to ideal conditions. It can also be pointed out that this could be a main reason as to why this method is not being applied in mass production on automated vehicles.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this paper, it can be seen the reality of researches done and the outcome of the present image segmentation techniques. Most of the observation from internet can be understood that there were not much disturbance present around potholes and neither was it from the various environment which are present in real case scenarios. The images from the database showed more likely scenarios and also lead to a drastic degradation in terms of accuracy and precision which was the objective and main agenda for this thesis topic to highlight. So it can be said that the values obtained for internet image are not ideal and applicable in real life and therefore for the table below the final values in terms of percentage are shown for all the algorithms from the pictures from our database. The values below show us the final readings for the accuracy values for our database for the entire algorithms.

To further point out, amongst the results obtained in terms of accuracy and precision for the four algorithms it can be observed that the method of Fuzzy C means algorithm was the most efficient in detecting potholes accurately and precisely, as it had the best values combined (63.25% accuracy and 70% precision). Hence, the algorithms require debugging before it can be implemented with a camera in a car. In future can be worked with hybrid classifiers or neural network for better accuracy hence, looking forward to automated applications.

5.2 Future Scope

Just like every other research there is no end to it but within the given time frame and timeline. Although there were some limitations as to the methods of accuracy and precision, this is how far we have progressed so far. To explain it easier, we can think of breaking down our work onto 3 phrases. If this is considered as phase 1, phase 2 and 3 will be carried out in the future. An explanation is given out below of phase 2 and 3:

Phase 2

Just like any other working algorithm or software requires constant debugging, this also requires debugging. This is due to the reason that even though the techniques provided satisfactory result on the photos of pothole from the internet but none of them could show promising results in real case scenarios. Hence, to achieve better results more research needs to done. Maybe, a hybrid classifier or other machine learning techniques could be used, where we can make the device learn to differentiate potholes from different unconventional, noisy surroundings. Thus, better accuracy and precision can be achieved. Only then, we would be able to say which technique will be the best for detecting real time potholes.

Phase 3

After we can assure that the implementation will promise a satisfactory result, then we can think of using it as a commercial product. Thus, we can implement a system with a camera connected with a suitable processing device and install it on a car. Later, more work can be done focused on reducing computational time and increasing the limitation of the speed of the car.

References

[1] The Economic Times. (2018). *Definition of Road Transport | What is Road Transport? Road Transport Meaning - The Economic Times*. [online] Available at: https://economictimes.indiatimes.com/definition/Road-transport [Accessed 26 Mar. 2018].

[2] Ari.buet.ac.bd. (2018). *Transport System in Bangladesh: Issues and Options for Sustainable Development*. [online] Available at: http://ari.buet.ac.bd/Personal%20Websites/Sohel/Sohel%20Mahmud%20Web%20page/Publication_%20All%20paper%20and%20abstract%20for%20WEB/Regional%20&%20National%20Conferences/4b_Full%20Paper_%20Transport%20System%20in%20Bangladesh.pdf [Accessed 26 Mar. 2018].

[3] NTV Online. (2018). *Auto-rickshaws, easy-bikes banned on Comilla city roads*. [online] Available at: http://en.ntvbd.com/bangladesh/156297/Auto-rickshaws-easy-bikes-banned-on-Comilla-city-roads [Accessed 26 Mar. 2018].

[4] Twitter. (2018). *The basic ABC & BBC layers for asphalt road*. [online] Available at: https://twitter.com/highwayseng/status/610485608021032960 [Accessed 26 Mar. 2018].

[5] Nptel.ac.in. (2018). *Pavement design*. [online] Available at: http://nptel.ac.in/courses/105101087/19-Ltexhtml/p6/p.html [Accessed 26 Mar. 2018].

[6] Orange.nsw.gov.au. (2018). *How long do roads last?*. [online] Available at: http://www.orange.nsw.gov.au/site/index.cfm?display=329376 [Accessed 26 Mar. 2018].

[7] Cornwall.gov.uk. (2018). *Potholes - Cornwall Council*. [online] Available at: https://www.cornwall.gov.uk/transport-and-streets/roads-highways-and-pavements/highway-maintenance/potholes/ [Accessed 26 Mar. 2018].

[8] Pinterest. (2018). *Asphalt*. [online] Available at: https://www.pinterest.co.uk/pin/231724343302323796/ [Accessed 26 Mar. 2018].

[9] H. K. Anusuya Devi, *"Thresholding: A Pixel-Level Image Processing Methodology Preprocessing Technique for an OCR System for the Brahmi Script* ", Ancient Asia, Journal of the society of south Asian Archaeology.

[10] Bryan S. Morse*, Lecture 4: Thresholding, Brigham Young University, 1998–2000*. [Online] Available at:http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/threshold.Pdf [Accessed 26 Mar. 2018]

[11] Scikit-image.org. (2018). *Thresholding — skimage v0.14dev docs*. [online] Available at: http://scikit-image.org/docs/dev/auto_examples/xx_applications/plot_thresholding.html [Accessed 26 Mar. 2018].

[12] Jason Corso. *Linear Filtering and Image Processing.* [Online] Available at: web.eecs.umich.edu/~jjcorso/t/598F14 [Accessed 26 Mar.2018]

[13] Mathworks.com. (2018). *Toolox Image - A Toolbox for General Purpose Image Processing*. [online] Available at: https://www.mathworks.com/examples/matlab/community/29583-toolox-image-a-toolbox-for-general-purpose-image-processing [Accessed 26 Mar. 2018].

[14] www.tutorialspoint.com. (2018). *Grayscale to RGB Conversion*. [online] Available at: https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm [Accessed 26 Mar. 2018].

[15] Math.uci.edu. (2018). *Histogram Equalization*. [online] Available at: https://www.math.uci.edu/icamp/courses/math77c/demos/hist_eq.pdf [Accessed 26 Mar. 2018].

[16] Gonzalez, R. C., & Woods, R. E. (2008). *Digital image processing* (3rd ed.). New Jersey: Pearson Prentice Hall.

[17] Slideshare.net. (2018). *Image processing fundamentals*. [online] Available at: https://www.slideshare.net/abshinde/image-processing-fundamentals-47399733 [Accessed 26 Mar. 2018].

[18] Cimss.ssec.wisc.edu. (2018). *What is Matlab*. [online] Available at: http://cimss.ssec.wisc.edu/wxwise/class/aos340/spr00/whatismatlab.htm [Accessed 26 Mar. 2018].

[19] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph.html [Accessed 26 Mar. 2018].

[20] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph2.html [Accessed 26 Mar. 2018].

[21] *Comparative Analysis of Various Edge Detection Techniques* by Er. Komal Sharma, Er. NavneetKaur, Computer Science Engineering, Chandigarh University, India

[22] Mathworks.com. (2018). *Image Thresholding*. [online] Available at: https://www.mathworks.com/discovery/image-thresholding.html [Accessed 26 Mar. 2018].

[23] Trevino, A. (2018). *Introduction to K-means Clustering*. [online] Datascience.com. Available at: https://www.datascience.com/blog/k-means-clustering [Accessed 26 Mar. 2018].

[24] Mathworks.com. (2018). *Fuzzy C-Means Clustering- MATLAB & Simulink Example- MathWorks United Kingdom*. [online] Available at: https://www.mathworks.com/help/fuzzy/examples/fuzzy-c-means-clustering.html [Accessed 26 Mar. 2018].

[25] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph3.html [Accessed 26 Mar. 2018].

[26] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph4.html [Accessed 26 Mar. 2018].

[27] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph5.html [Accessed 26 Mar. 2018].

[28] Edoras.sdsu.edu. (2018). *Morphological Operations (Image Processing Toolbox)*. [online] Available at: https://edoras.sdsu.edu/doc/matlab/toolbox/images/morph6.html [Accessed 26 Mar. 2018].

[29] Mathworks.com. (2018). *Measure properties of image regions - MATLAB regionprops- MathWorks United Kingdom*. [online] Available at: https://www.mathworks.com/help/images/ref/regionprops.html [Accessed 26 Mar. 2018].

[30] Mathworks.com. (2018). *Morphological operations on binary images - MATLAB bwmorph- MathWorks United Kingdom*. [online] Available at: https://www.mathworks.com/help/images/ref/bwmorph.html [Accessed 26 Mar. 2018].

[31] Mathworks.com. (2018). *Label connected components in 2-D binary image - MATLAB bwlabel- MathWorks United Kingdom*. [online] Available at: https://www.mathworks.com/help/images/ref/bwlabel.html [Accessed 26 Mar. 2018].

[32] Vigneshwar, K., & Kumar, B. (2016). Detection and counting of pothole using image processing techniques. *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 1-4.