2D GEOMETRIC OBJECT SHAPE DETECTION USING DIGITAL IMAGE PROCESSING USING MATLAB

Progress Report

In fulfillment of the requirements for the

NU 302 R&D Project

At NIIT University

Submitted by

Priyamwad Pathak (U101114FCS221)

Koushik Andhavarapu (U101114FCS197)

Sowmya Charlu (U101114FCS142)

Area

NIIT University

Neemrana, Rajasthan

CERTIFICATE

This is to certify that the present research work entitled " 2D Geometric object shape detection using digital image processing using Matlab" being submitted to NIIT University, Neemrana, Rajasthan, in the fulfillment of the requirements for the course at NIIT University, Neemrana,embodies authentic andfaithful record of original research carried out by Priyamwad Pathak, Koushik Andhavarapu, Sowmya Charlu, students of B Tech (CSE) at NIIT University, Neemrana,. She /He has worked under our supervision and that the matter embodied in this project work has not been submitted, in part or full, as a project report for any course of NIIT University, Neemrana or any other university.

Name and Title of the Mentor

Mr. Vikas Upadhyaya.

LIST OF FIGURES

Fig1: Digital images. Source: Self
Fig2: Digital images. Source: Self
Fig3: Digital images. Source: Google images
Fig4: Indexed image. Source: Google images
Fig5: RGB image. Source: Google images
Fig6: Grayscale image. Source: Google images
Fig7: Binary image. Source: Google images
Fig8: Canny edge detection. Source: Google images
Fig9: Various edge detection filters. Source: Google images
Fig10: Computer representation of an image. Source: Google images
Fig11: Digital image as an output. Source: Google images
Fig12: Canny edge detection over an image. Source: Self
Fig13: Proposed methodology. Source: Self
Fig14: Method1 . Source: Self
Fig15: Output of method 1. Source: Self
Fig16: Method1 applied on a straight line. Source: Self
Fig17: Method1 applied on a rectangle. Source: Self
Fig18: Method1 applied on a triangle. Source: Self
Fig19: Method1 applied on a ellipse. Source: Self
Fig20: Method1 applied on a circle. Source: Self
Fig21: Method1 applied on a rectangle. Source: Self
Fig22: Methodology 2. Source: Self
Fig23: Output of method 2. Source: Self
Fig24: Output of global maxima method. Source: Self

Fig25: Corners detected. Source: Self

Fig26: Image containing shapes. Source: Self

Fig27: Otsu's thresholded output of image1. Source: Self

Fig28: Real world image. Source: Self

Fig29: Otsu's thresholded output of image2. Source: Self

Fig30: Result of method 1 on regular shapes. Source: Self

Fig31: Result of method 1 on irregular shapes. Source: Self

Fig32: Real world image1. Source: Self

Fig33: Real world image2. Source: Self

Fig34: Real world image3. Source: Self

Fig35: Output of application of algorithm on real world objects. Source: Self

Fig36: Output of application of algorithm on real world objects. Source: Self

Fig37: Results from research paper. Source: “2D GEOMETRIC SHAPE AND COLOR RECOGNITION USING DIGITAL IMAGE PROCESSING”.

Fig38: Results from research paper. Source: “2D Basic Shape Detection Using Region Properties”.

LIST OF TABLES

Table 1: Table containing the results of the number of regular shapes detected.

Table 2: Table containing the results of the number of irregular shapes detected.

CONTENTS

Title	Page no.
Certificate	II
List of Figures	III-IV
List of Tables	V
Rational	**1-2**
Literature Review	**3-15**
Objectives	**16**
Methodology	**17-25**
Results	**26-33**
Summary	**34**
Future Work	**35**
References	**36-37**

RATIONALE

The project aims at providing an algorithm that involves the use of Digital Image Processing for recognizing 2-D Geometrical Shapes like circle, triangle. rectangle, square, ellipse and straight line and points and similarly for their irregular counterparts with certain amount of deviation from the perfect shape of the object. Our algorithm makes use of Regionprops, Canny edge detection, closing, hole filling and other image processing techniques for the detection. The functions we used are there inside the Image processing toolbox of Matlab. The algorithm was developed and simulated using Matlab. Our algorithm makes of values determined by the regionprops of Matlab and thus compares the range thus decided for each shape against it and thus comes to a conclusion about the shape to be identified. A set of 120 images were used for testing the algorithm and it works fine.

Digital Image processing, as the world knows it, helps to process any digital form of an image through various processes and helps to increase the quality of the picture or try to find out various other important information from the image itself.

The world as we know as of now, communicates through images, pictures taken through phones, cameras etc . and since the digital world is in vogue, it becomes absolutely inevitable for image processing techniques to be developed and invented for various purposes be it in front of the screen or behind the screen like for security purposes etc.

The very idea of Digital Image processing brings to us the thought of processing an image using various techniques for security purposes, for research intended purposes, developing purposes or just to enhance a picture probably. Well, we as a team considered this project for the very objective of identifying 2D geometric shapes which could further be used for online learning, teaching purposes and peer to peer learning as well.

We would be working on developing an algorithm for the very objective and shall be using Matlab for it and under Matlab, we would specifically be

using the image processing toolbox which contains important properties and functions that could come useful.
In terms of educational uses and how it might come useful to the world, here goes:

- Firstly, by identifying 2D geometric shapes and objects, we could help students who are probably too young to recognize shapes and objects, by creating applications based on this algorithm. This shall help in building their cognitive abilities.
- Secondly, the use of this algorithm can also be used widely in places where security is of much importance and the objects of security might contain these shapes and objects. In the recent past there have quite some improvements on developing algorithms based on this very objective and people have been very successful except for a few minor problems and have developed them using various open source techniques as well as Matlab.

We have looked out for various such research papers and realised that most of the algorithms had problems, some of them being not using proper edge detection techniques, or shapes not being detected due to minor differences in the ranges of functions or properties being used. Thus we made sure that we went through a lot of project related research papers published by scholars and tried to develop an algorithm that closely resembled our objective.
The objective initially was to develop an algorithm that detected basic 2D geometric shapes like square, rectangle, triangle, circle, ellipse, point and straight line, but further working on this project we also thought of including the irregular counterparts of these shapes and modified the algorithm so that it would detect the irregular side of these shapes in the sense that if a circle did not completely resemble a circle, the algorithm would still detect the shape to be a circle. Thus, our intention was to create an algorithm using various properties and make our objective possible.

LITERATURE SURVEY

Going through a lot of research papers was a part of the literature review, plus we also referred to the Digital Image Processing textbook written by Gonzalez. Well, What did we learn about this particular field:
Digital image processing first came into picture when an image had to be transmitted across the Atlantic ocean, when they realised that the picture was not proper enough and hat is when they introduced tapes of varying gray levels. This further developed with the advancement of computers and computer techniques. This field further developed when it was used for space research and discovery and its further uses propped up, some of them being:

1. Digital Image Processing being used for computer vision and robotic intelligence.
2. Use in checking the authenticity of currency.
3. Used in video surveillances and traffic surveillance.
4. Used in fault checking in machines and industry.
5. Recognizing certain objects in a particular picture.

What would one mean by an image?

- An image may be defined as a 2-dimensional function, f(x,y), where x and y are spatial plane coordinates. The amplitude of f at any pair of coordinates (x,y), is called the intensity or the gray-level of the image at that point.

What is a digital image?

- When (x,y) and the intensity values of f are all **finite , discrete** quantities, we call the image a digital image.
- A digital image is composed of finite number of elements, each of which has a particular **location and value.**
- Basically numeric representation of two-dimensional image. It consists of intensity values known as pixels, pels.

Examples:

Fig1 and Fig 2

What is Digital Image processing?

- Refers to processing images by means of a digital computer.
- It also includes the use of various algorithms to be performed on images and process them.
- It involves the tasks like reduction of noise, blurring of the image, feature extraction in the image.

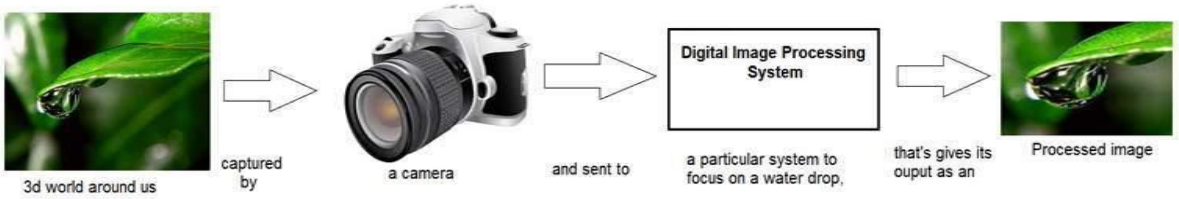


Fig3

Some of the properties that were understood and learnt created the foundation for the project and they were as follows:

Knowledge of the types of images(RGB, Grayscale, Binary and Indexed images)

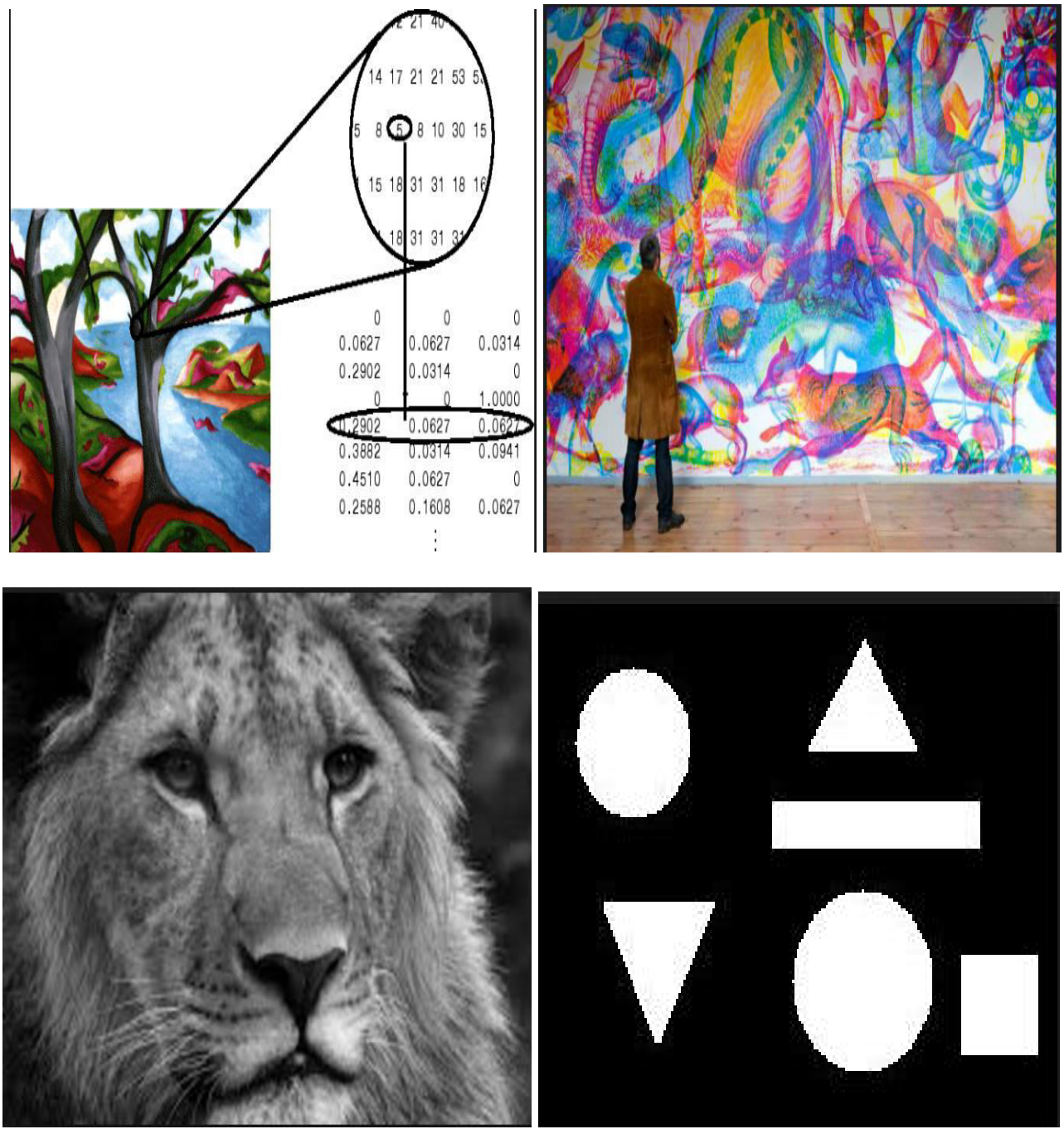


Fig: 4,5,6,7 (in clockwise direction starting from top left: indexed, RGB, Grayscale, Binary).

What is image segmentation?

- A very important part of image processing that subdivides an image into its constituent regions or objects.
    1. Discontinuity
    2. Similarity

Abrupt changes in intensity i.e. discontinuity in points, lines and edges could be detected by various **edge detection techniques**.

The second category is about partitioning an image into regions that are similar as per a set of predefined criteria. E.g.: **Thresholding techniques.**

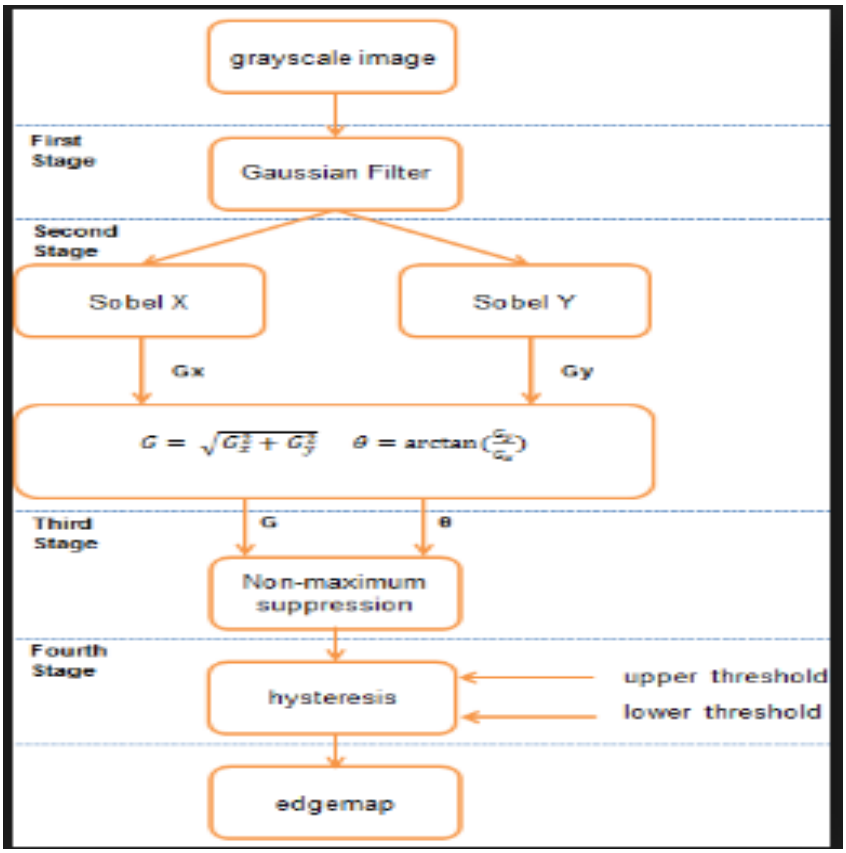
- **Various edge detection techniques**

We compare the intensity value in a pixel with a threshold value that can be obtained. Edge detection serves the base for the image segmentation for separating foreground and background. Some of the edge detection techniques are:

1. Sobel edge detection
2. Prewitt edge detection
3. Canny edge detection

Gradient operator segmentation algorithms for monochrome images are based generally on two properties:

1. Discontinuity
2. Similarity



Boxed element indicates the location of the origin

	H_1	H_2
Roberts [9]	$\begin{bmatrix} \boxed{0} & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} \boxed{1} & 0 \\ 0 & -1 \end{bmatrix}$
Smoothed (Prewitt [6])	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & \boxed{0} & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & \boxed{0} & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Sobel [7]	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & \boxed{0} & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & \boxed{0} & 0 \\ 1 & 2 & 1 \end{bmatrix}$
Isotropic	$\begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & \boxed{0} & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & \boxed{0} & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$

Fig8, Fig9

The optimal edge based segmentation is canny edge detection which uses Gaussian filter in its process. Since there is no noise left after applying and also the threshold step at last makes canny best. We have used canny edge function using edge function in Matlab.

- **Various thresholding techniques**

Thresholding helps in separating the foreground and the background and is an image segmentation analysis that converts a grayscale image into a binary image.

1. Otsu's thresholding
2. Using Gaussian filter

Since computers can only understand numbers, the next question that comes to mind is how image is treated inside the machine so called computer. The simple answer would be the image is represented as intensity values in a two dimensional array. The operations are performed on this image so as to get an desired output. Below is a picture of computer representation of image.

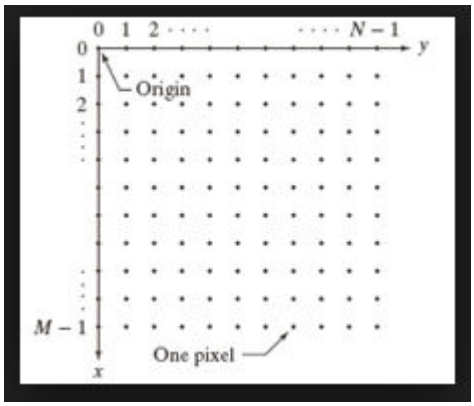


Fig10

Here the M,N indicates the rows and columns for an image and in Matlab the initial index position starts from 1.

Computer vision is the future of everything. The idea is that giving sight to the computers so that they can extract information from and make decisions like a human beings.
There are applications that this project can be used. Some of the main applications are in Robotics, industry for manufacturing products.

The fundamental steps involved in Image processing are :

1. Image Filtering & Enhancement
2. Image Restoration
3. Color Image Processing
4. Wavelets and Multiresolution Processing
5. Compression
6. Morphological Processing
7. Segmentation
8. Representation & Description
9. Object Recognition

Now, let us discuss how these steps are used in our project to achieve our task:

- The first and foremost step would be reading an image to get the desired result. Now various techniques of digital image processing are applied onto the image so as to get the desired information. In order to become suitable for Digital image processing, an image function f(x, y) that is the voltage signal generated when we click an image must be digitized both spatially and in amplitude. Hence we need to convert the continuous data into digital data. Here two steps that make it happen are sampling and quantization. The sampling rate determines the spatial resolution of the image, while the quantization level determines the number of gray levels in the digitized image. The transition between the continuous values of image function and its digital equivalent is called quantization. At this time the computer can store the image in its memory for further processing. Using **imread** function we have our input image in the computer memory. Now certain procedures need to be taken for detecting the shape. Since we have dealt with the regular Google images in the beginning there is no form of noise in the image(the

object is clearly visible), but the main problem arises when we take an image from an acquisition system such as camera, video, etc..This may rise to noise as well as non uniform illumination in the image which leads to the take the other image processing concepts like hole filling, contrast stretching. We first started with the uniform objects and then onto the non uniform objects which had lead us to facing problems. Initially we were having little noise and no illumination because it comes only when you take an image from outside. We had also assumed that the inclination of the object is along the X-Axis which excludes a part in pre-processing step. Otherwise we had to rotate the object and then do the operations. We have taken the object along the above mentioned axis so as to make the algorithm simpler. An algorithm is a step by step procedure to achieve some task. The less assumptions the more is the algorithm speed . Although we had not used all the fundamental steps in DIP mentioned above but we followed the same path.

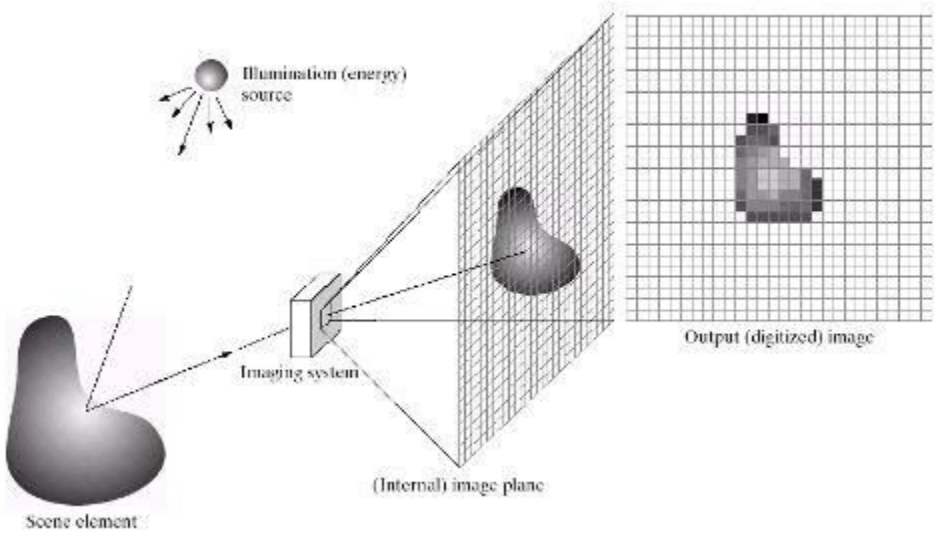


Fig11

- In a signal processing filters like low pass filters, high pass filters are used and when in image the types of filters used are Gaussian filter, median filter, max filter, sobel filter, prewitt filter ,Roberts filter. The efficient filter that has raised over the years is Gaussian filter and we

have used this filter for removing noise and for getting the enhanced image. The way filtering works on an image is that there is a kernel which goes around the image in a zigzag way and then multiply contents of image with kernel and replace the center pixel of resultant matrix. The matlab function that is used **imgaussian** function. We can also specify the sigma value in Gaussian function for the kernel size. The bigger the sigma, better the effect of its on the image.

- Different Images are stored in different formats. Reading an RGB image the computer stores three matrices for red, green and blue. A gray scale image is a single matrix which has its intensity values ranging from 0-255.A binary image is something that is matrix filled with 0 and 1's.we in our algorithm read an RGB image and converted into grayscale image using the **rgb2gray** function. This function returns an image that is having intensity values from 0 to 255.Next step is to convert this image into binary image for identification. The reason we are converting the image into binary image because it has only two classes i.e. foreground and background image. The background of the object is represented in black and the object itself, the foreground is represented in white. This makes further steps easier.

- We have also used morphological processing on the image. The two basic things in it are erosion and dilation. Erosion means shrinking the size of the object whereas dilation is the process of expanding the size of the object. However we had used erosion when we are dealing with non uniform objects because erosion helps in separating two objects when they are connected each other with a amount of few pixels. This is also for different purposes. The hole filling and other techniques are based on same method internally(In algorithm).The function we have used is **imerode** and imdilate. Also there are other functions for opening and closing of an image.

- Representation & Description is also an important step. The labelling of objects is done using bwlabel function inside Matlab. Using this function the computer/IDE randomly allocates a particular number starting from 1.Another function find is used to get the sub row and sub column size of the matrix containing the object. Thus labelling of objects is a plus for us to distinguish between the objects in the foreground.

- The last fundamental step is to detect the object. For this we have used three different types of algorithm. The first one being the bounding box method. A bounding box is the smallest rectangle possible around the object that is present in the image. We also implemented another algorithm which gives the points lie in the corners of a shape, but this is a graph. By the graph we can estimate the no of points and conclude what shape it is. This method gives the points in the corners however we did not get the number of points that were shown in the graph to get the mean of all points. This is the main drawback of the method. Another method was get the points using the extrema function inside Matlab of regionprops. This type is used on uniform objects and also it cannot be used in non uniform objects. The main drawback is that it cannot be non uniform objects.

The secondary data that was collected mostly from research papers which were helpful and served as a guide throughout the process.
Various methods and procedures in the papers described actually guided us as to what properties and methods should be applied and used for proper shape detection.

One of the papers had the following procedure:

- The paper suggests a canny edge detection over the grayscale images.
- They have first converted the RGB images images into a grayscale image.

- Later they have transformed the image into a binary image and inverted it.
- Did a canny edge detection over the images.

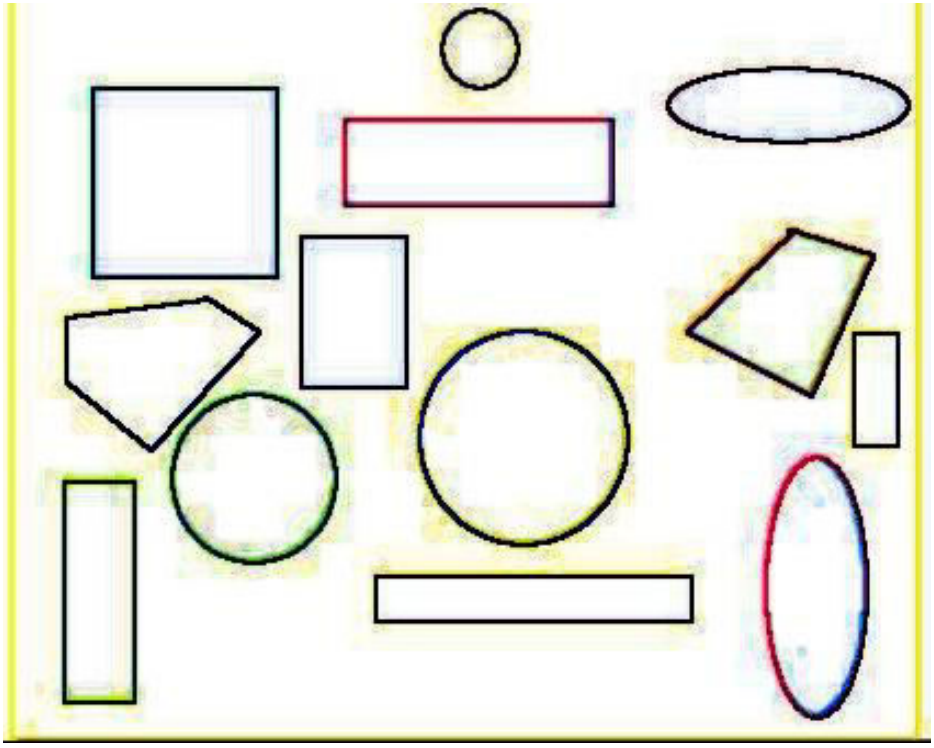


Fig12

A second methodology taken from another paper suggests the following procedure:

- Read/capture image.
- Convert 3 dimensional RGB image to 2 dimensional black and white image.
- Recognize boundaries of objects.
- Calculate area of object.
- Filter out objects below an area threshold.
- Find clockwise inclination of object.
- Rotate object in anti-clockwise direction by angle measured in previous step.
- Find bounding box of the object.

The drawbacks of the paper found out were:

1. The pictures should not contain shadows else the thresholding will also consider the shadows and the value of extent might be miscalculated.
2. The images must be taken in a well lit environment.

The third methodology explained in another paper was as follows:

- Convert an RGB image to HSL (hue, saturation, lightness) image.
- Do Otsu thresholding over the converted image.
- Image fill
- Median filtering
- Do sobel edge detection.
- Check for compactness of the objects in the image.
- Use a shape template
- Get the desired result.

The drawbacks found in the third methodology were as follows:

- If the input image has an uneven intensity, the image shall not be thresholded properly.
- The paper contributors also used images in which the object boundaries touched each other which led to inaccurate calculation in the parameter and area estimation.
- Noises are not completely eliminated and thus considered as objects.

Considering the above methodologies as a guide for the project, a few steps that were taken to design our future work were considered and an outline of the project steps as follows:

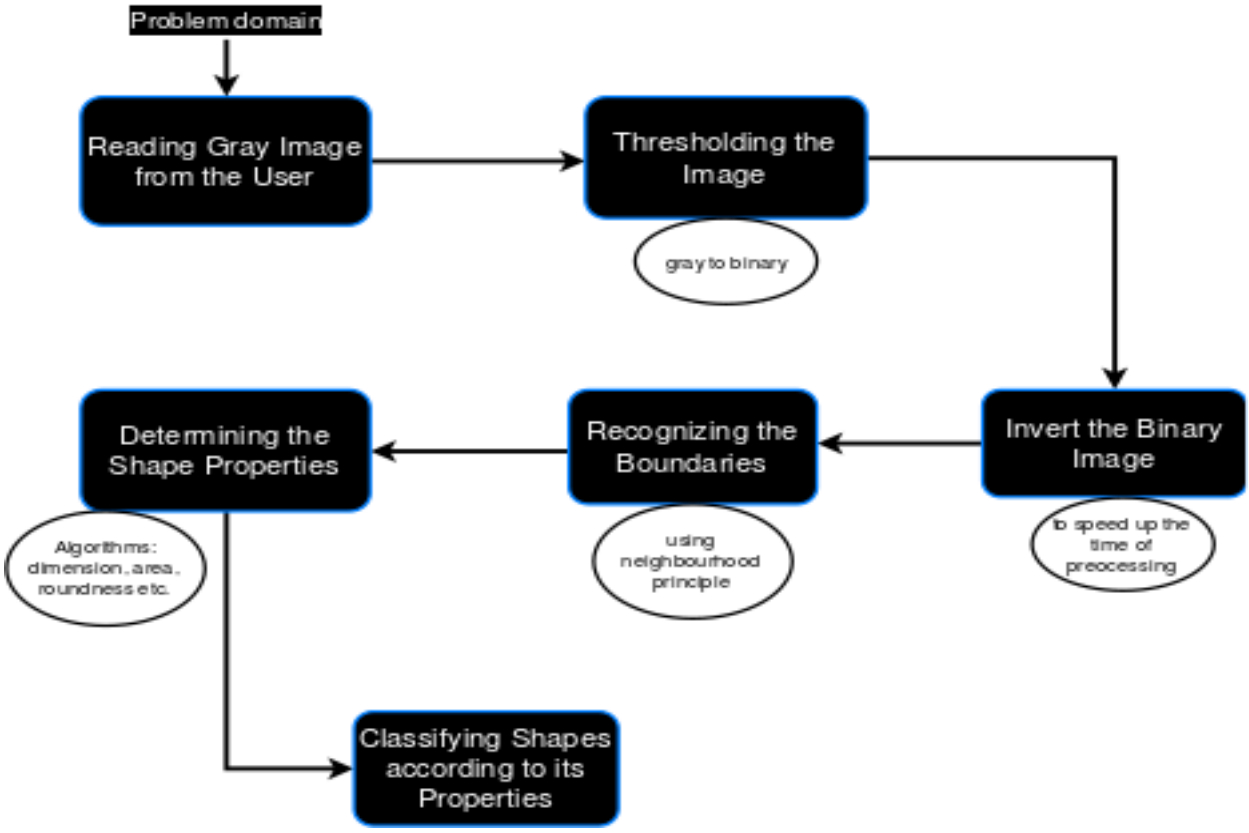


Fig13

OBJECTIVES

The objective was divided into parts as follows:

1) To read and understand all required material necessary for undertaking the project.
2) To detect basic 2D geometric shapes such as

- Point
- Straight line
- Square

- Circle
- Triangle
- Ellipse

3) To detect basic 2D irregular shapes of the following:

- Square
- Circle
- Triangle
- Straight line
- Ellipse

4) Capture an image from the webcam or mobile phone and detect the 2D basic shapes in that image.

METHODOLOGY

For the completion of the very first objective, the methodology pertained to was as follows:

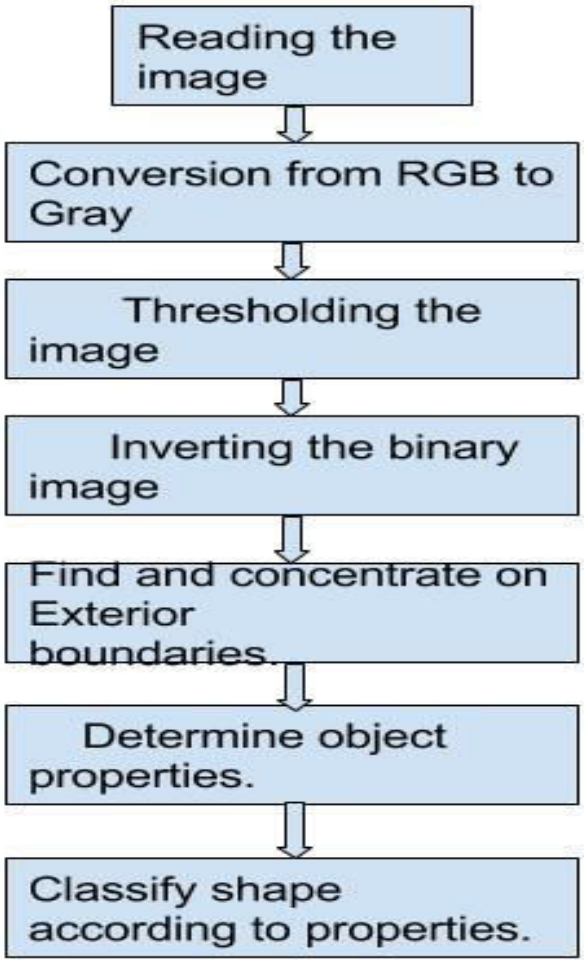


Fig14

- The image is first read and converted to a grayscale image consisting of gray pixels.
- Then the image is thresholded using a threshold value to determine the black and white pixels.
- Exterior boundaries are found and properties like perimeter and area are found.
- The ratio of perimeter^2/area is taken out and with the help of determined ranges, the objects are classified as either a point, straight line, square, circle, triangle or ellipse.

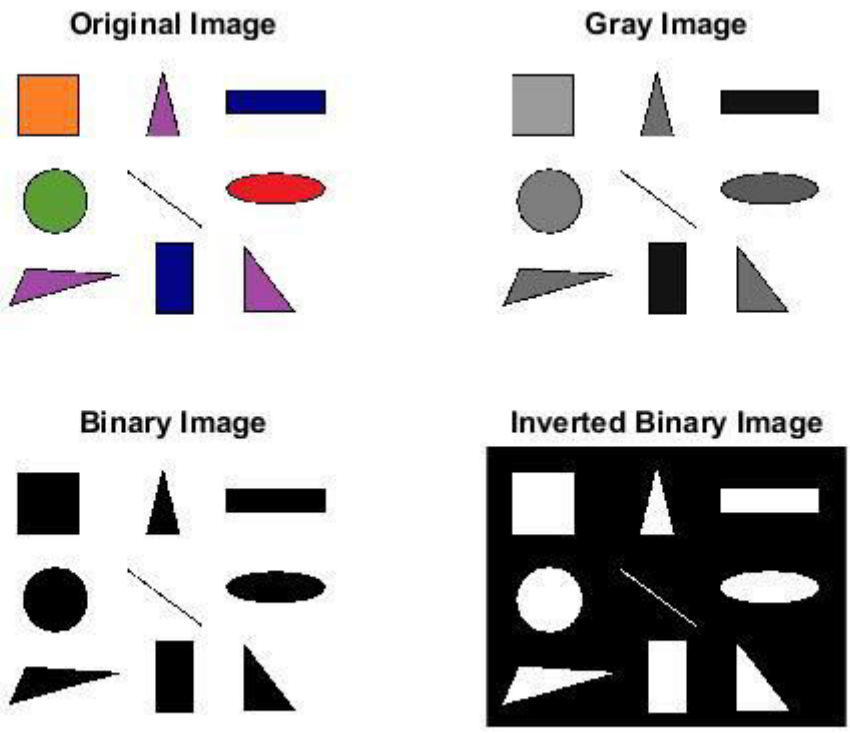


Fig15

The Regionprops of the Matlab image processing toolbox was used and the following were the properties used: i meant insert images for our methodology 1 -

- Centroid
- Area
- Perimeter
- Extent
- Bounding box
- Major Axis Length
- Minor Axis Length

1. **Centroid:** Is a vector, the center of mass of region.
2. **Area**: Is a scalar, the actual number of pixels in a region.
3. **Perimeter:** The distance around a geometrical shape.

4. **Extent:** Scalar; the proportion of the pixels in the bounding box that are also in the region. Computed as the Area divided by area of the bounding box. This property is supported only for 2-D input label matrices.
5. **Bounding box:** The smallest rectangle containing the region.
6. **Major Axis Length:** Scalar; the length (in pixels) of the major axis of the ellipse that has the same second-moments as the region. This property is supported only for 2-D input label matrices.
7. **Minor Axis Length :** Scalar; the length (in pixels) of the minor axis of the ellipse that has the same second-moments as the region. This property is supported only for 2-D input label matrices.

The method was implemented using the combination of Bounding box/ extent property and the property of compactness i.e. the ratio $perimeter^2/area$.

Issues we faced:

1. Initially the code could only detect equilateral triangles, but through research the problem was overcome.
2. The code initially detected squares as rectangles, but through the use of the property of compactness i.e. the ratio of $perimeter^2/area$, this problem too was overcome.

Results:

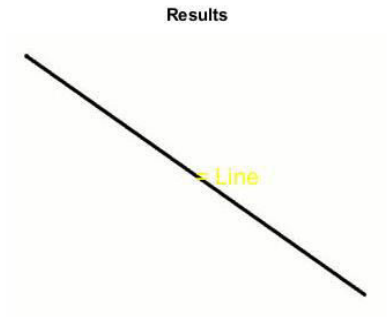


Fig16

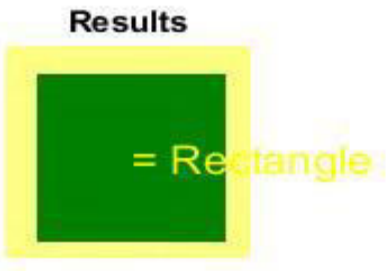


Fig17

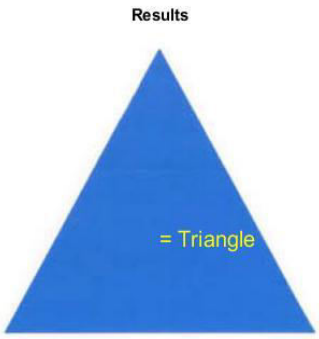


Fig18

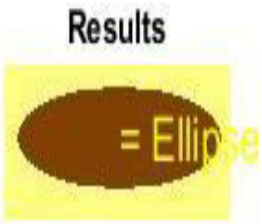


Fig19

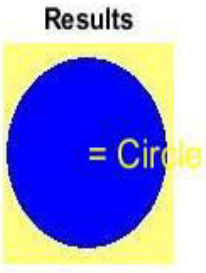


Fig20

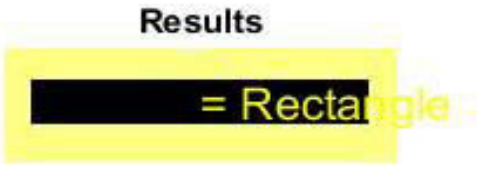


Fig21

Methodology 2:

Further development of the code was done to incorporate real world images captured through mobile phone or webcam and detect the basic shapes in them. The methodology included the following:

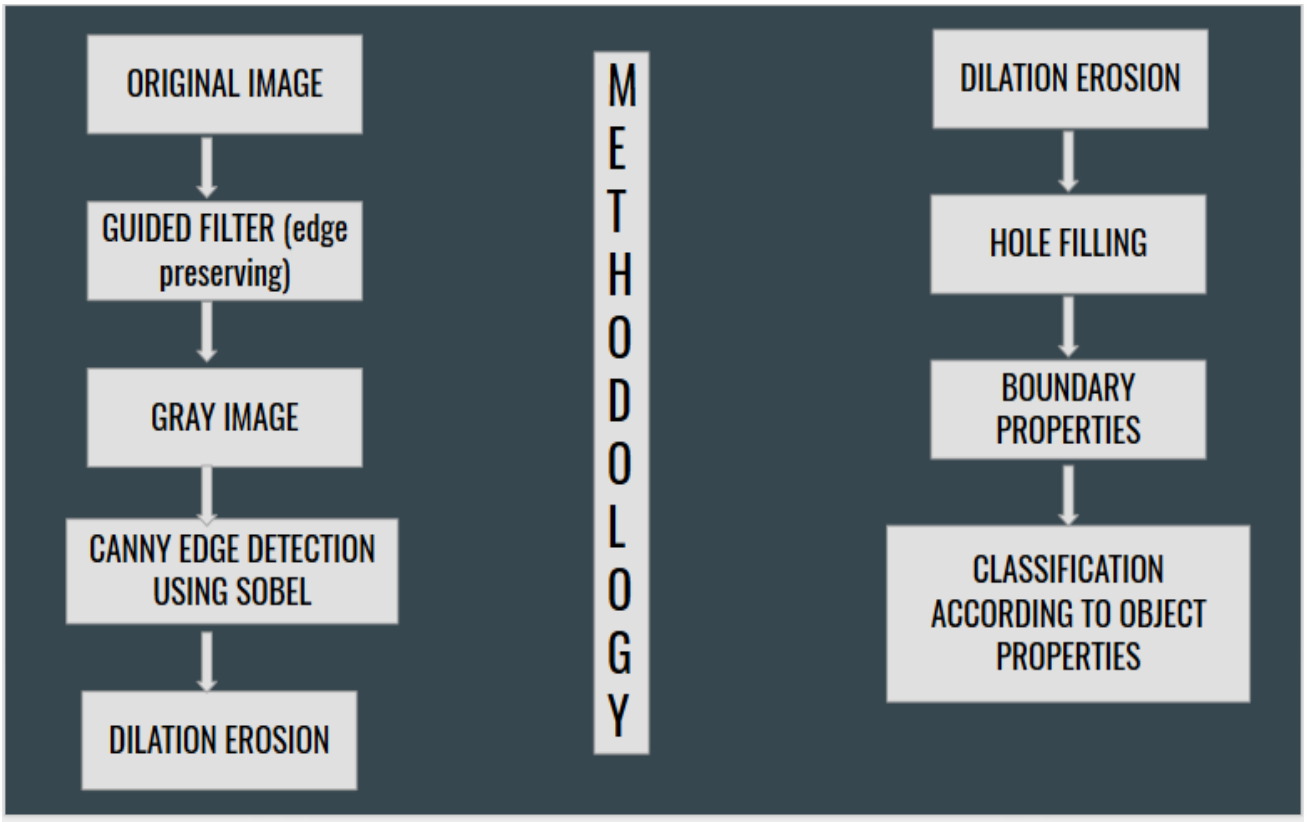


Fig22

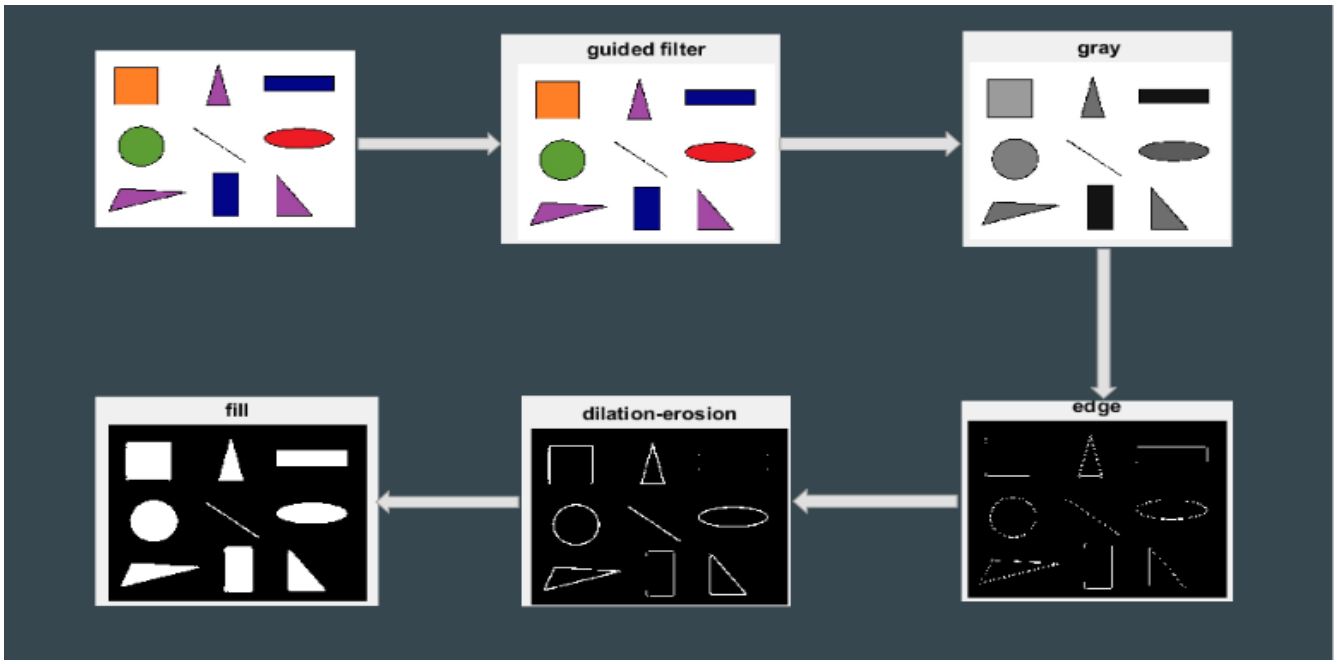


Fig 23

Global maxima method was also used for the detection of basic shapes in an image and identify the number of vertices of that particular shape. The output is something like follows:

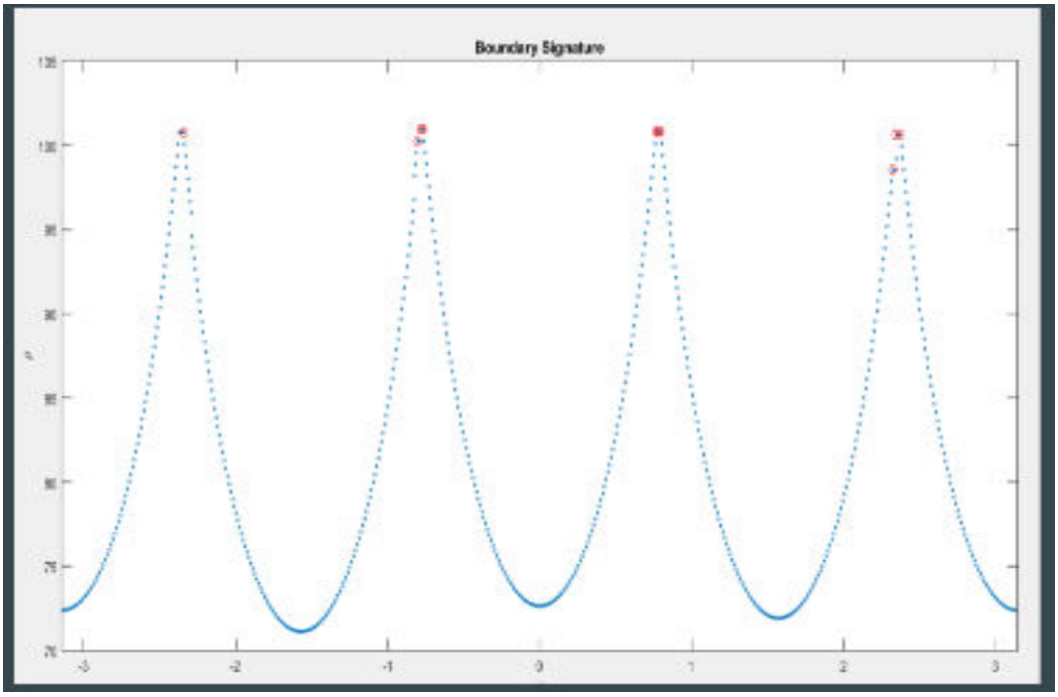


Fig24

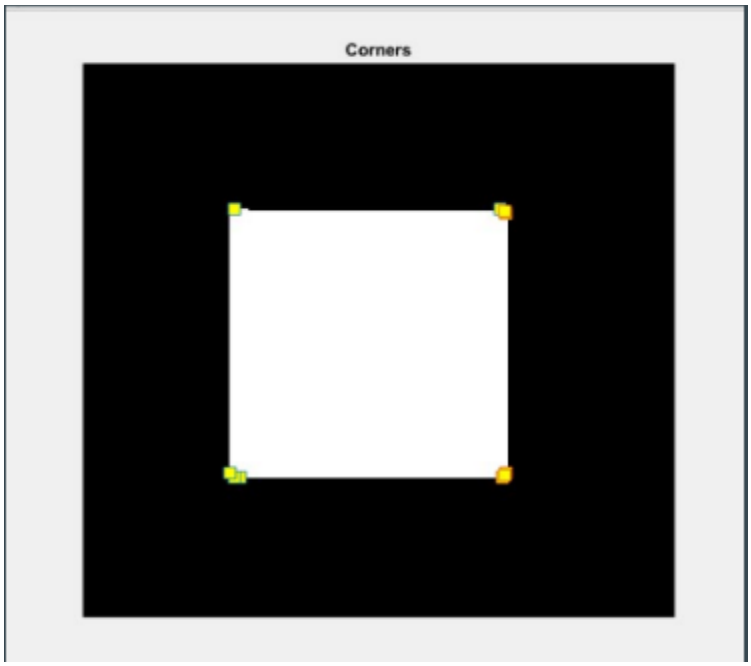


Fig25

The new steps that were included were :

1. **Guided filter:** This performs an edge- preserving smoothing on an image using another second image as a guidance image, which can be the image itself, a different version of the image or a different image altogether. While calculating the output pixel, guided image filtering takes into account the statistics of a region.
2. **Dilation erosion:** These are process of morphological operations. Erosion refers to shrinking of the object whereas dilation refers to increase in size of the object. The way they works is we have a structuring element and this element is used to move around the image and should satisfy some condition like we do with kernel in filters.
3. **Hole filling:** The imfill function performs a flood-fill operation on binary and grayscale images.

Problems that were faced:

1. Our intention of capturing the image through webcam was futile because the resolution of the image was not up to the mark.

2. Later the images captured through phone also had issues like shadows and noise for which we have applied noise removal techniques.
3. There is no generalised thresholding technique for thresholding the images, one technique may work for one object and may not work for the other objects in the same image.

- Otsu's thresholding working right

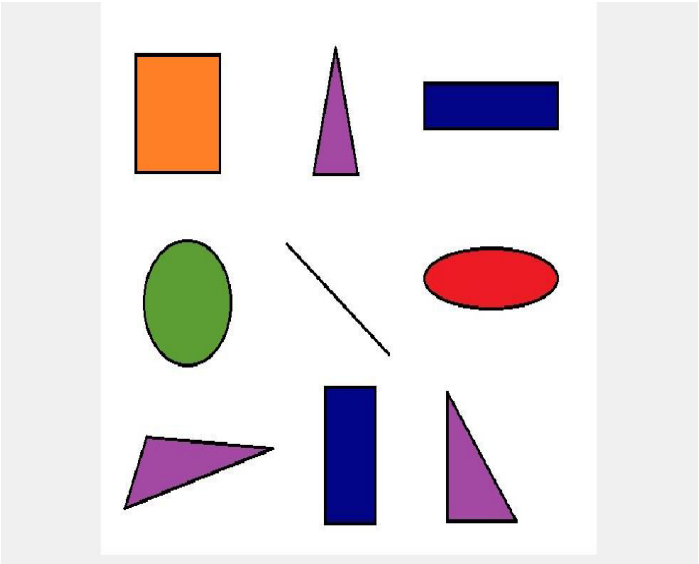


Fig26

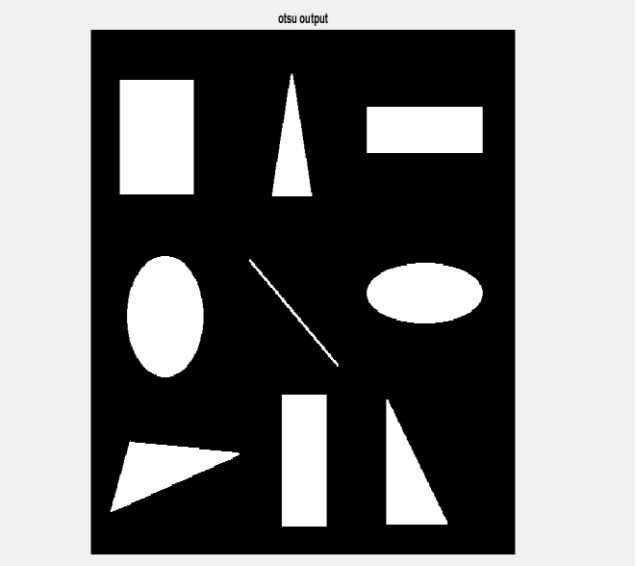


Fig27

- Otsu’s thresholding not working properly

Fig28

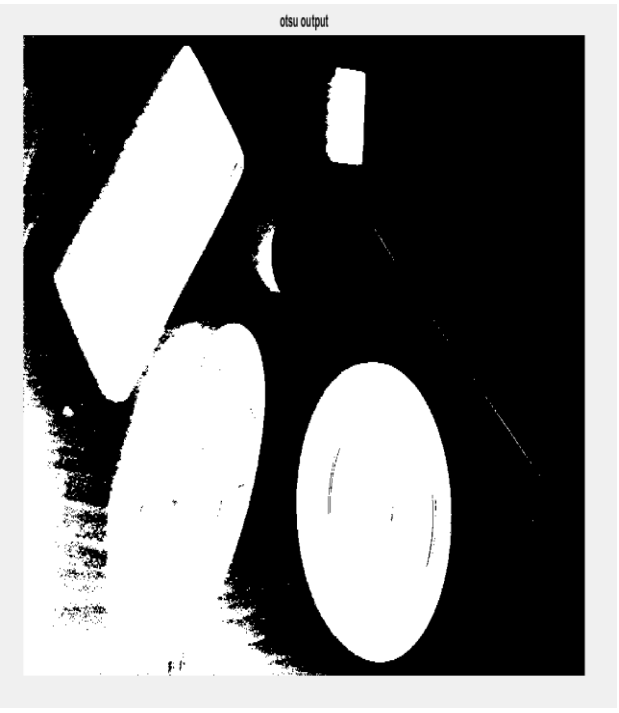


Fig29

RESULTS

Our Algorithm detects both the uniform as well as non uniform objects in an image. However we have to take images separately by keeping the object in an environment and then click a picture. Below is different images explaining the results we had obtained.

- We have detected basic regular and irregular shapes like point, line, square , circle, rectangle , ellipse and triangle properly.
- We tested the algorithm on real world images with proper illumination of the image.
- Our new algorithm considers global maxima on the boundary pixels of the object to count the number of vertices thus detecting the shapes.

Firstly, for a uniform objects in an image:

This is an input image for all shapes that are contained in a single image. Then we have ran our algorithm on this image..Below are results that are obtained in different steps.

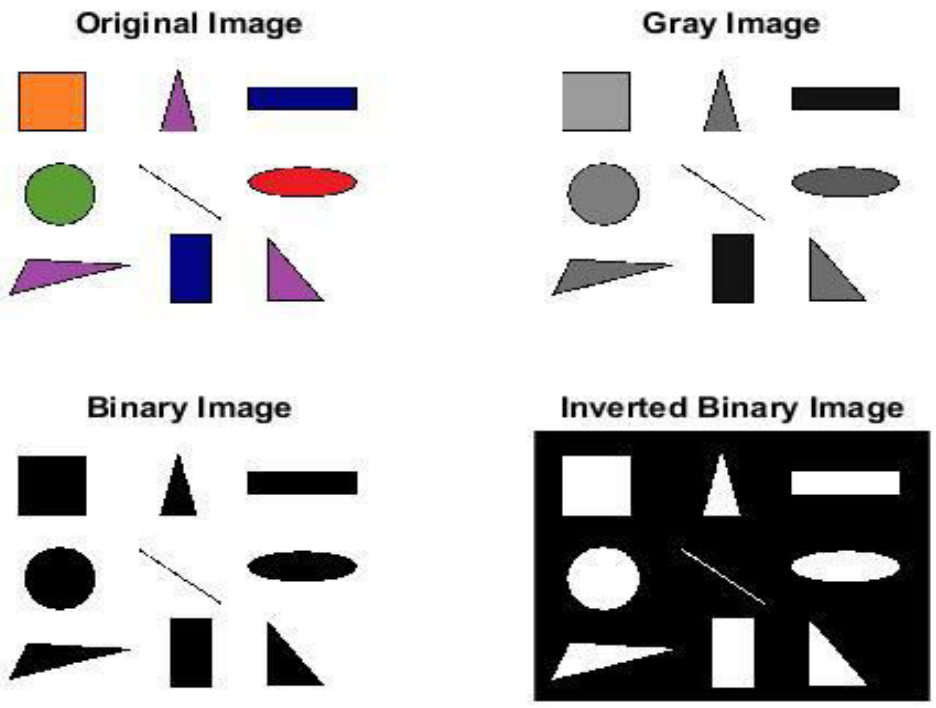


Fig15

The final output is also an image indicating the shape of the object at the centroid position of the object:

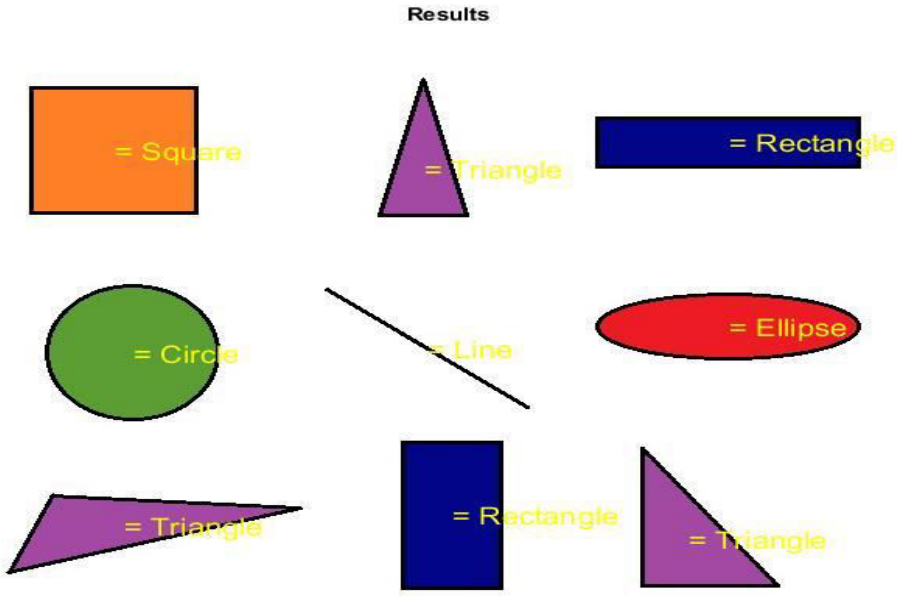


Fig30

For non uniform objects the algorithm results out as:

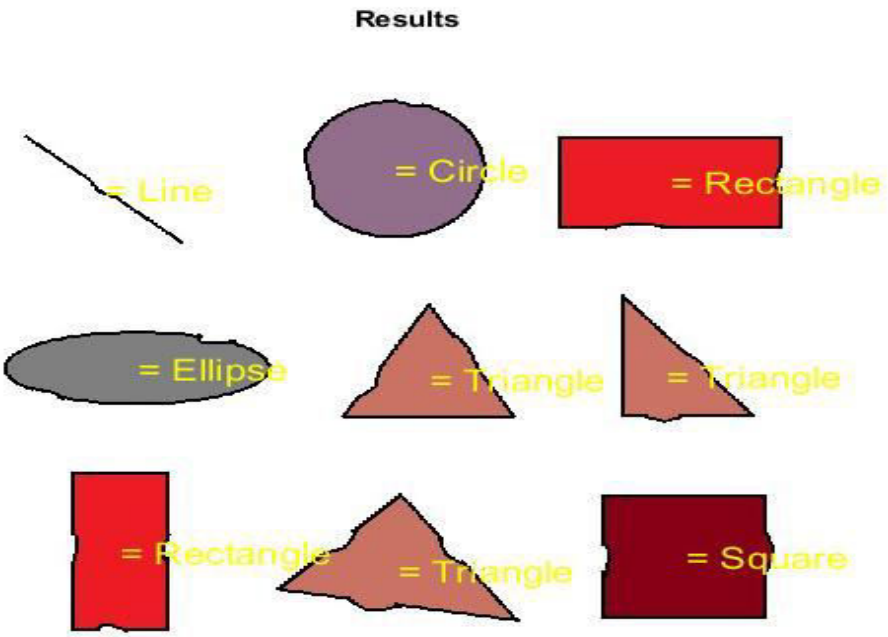


Fig31

Real world images were taken into consideration to detect shapes that are contained in the image, the images are as follows:

Fig32

Fig33

Fig34

The results before identifying the shape is the binary image also called as black and white image. Here the morphological operations are applied and experimentation is done for improving the quality of the object. This makes the algorithm automatic to semi automatic(manually we have to examine

the eroded image).No particular value we can decide for the object shape detection.

Here are the results:

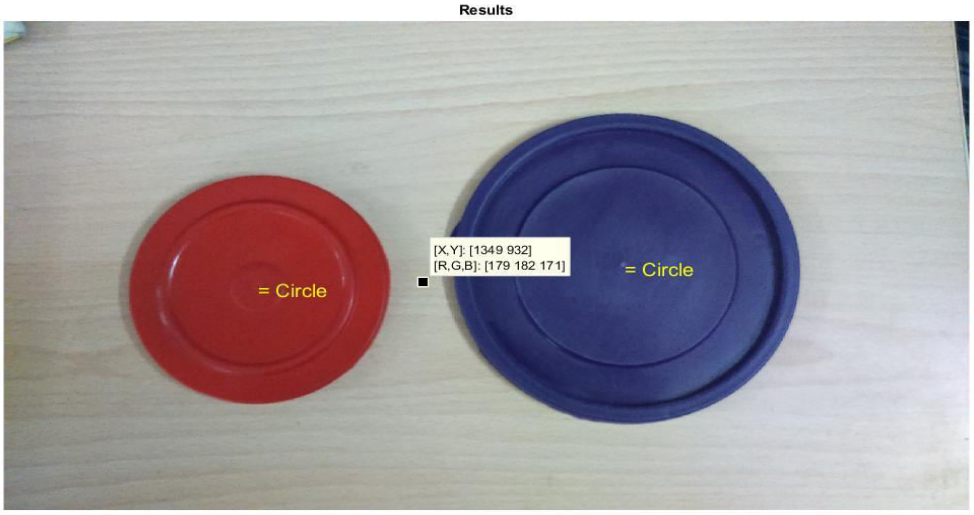


Fig35

Fig36

- The overall results are as follows:

  1) Results for regular shapes:

BASIC REGULAR SHAPES	NO. OF OBJECTS TAKEN	NO OF OBJECTS DETECTED
POINT	**5**	**5**
STRAIGHT LINE	**10**	**10**
SQUARE	**15**	**15**
CIRCLE	**11**	**11**
RECTANGLE	**16**	**16**
TRIANGLE	**20**	**20**
ELLIPSE	**10**	**10**

Table 1

ACCURACY: **100% for mentioned regular shapes.**

2) Results for irregular shapes:

BASIC IRREGULAR SHAPES	NO. OF OBJECTS TAKEN	NO OF OBJECTS DETECTED
SQUARE	10	9
CIRCLE	8	7
RECTANGLE	10	10
TRIANGLE	12	12
ELLIPSE	8	6

Table 2

ACCURACY: **92% for irregular shapes.**

On comparison with different papers:
1) Paper 1

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 6, June 2013

Table 1
Results of test of algorithm on images from database.

Shape of Object	Color of Object	No. correctly recognized/ No. of images tested
Circle	Red	15/15
	Green	15/15
	Blue	15/15
Rectangle	Red	15/15
	Green	15/15
	Blue	15/15
Square	Red	15/15
	Green	14/15
	Blue	15/15
Triangle	Red	14/15
	Green	15/15
	Blue	15/15

Fig 37

- The accuracy of this paper's detection is 98% as compared to ours, but we have implemented the detection of more than just the shapes presented in the paper.

Reference:

"2D GEOMETRIC SHAPE AND COLOR RECOGNITION USING DIGITAL IMAGE PROCESSING".

2) Paper 2

Table 3. Accuracy Table for Basic 2D object detection

Images	False detection of objects (FD)	True detection of objects (TD)	No. Of objects present in the image
Image1	0	13	13
Image2	1	8	9
Image3	1	8	9
Image4	1	3	4
Image5	0	5	5

The accuracy of this algorithm is 90.38% obtained.

Fig 38

- The accuracy of this paper's detection is 90.38% as compared to ours, but we have implemented the detection of more than just the shapes presented in the paper.

Reference:

"2D Basic Shape Detection Using Region Properties".

SUMMARY

- The project implemented describes identifying the shape of an object. This particular algorithm implemented has a variety of uses in terms of building applications for teaching-learning processes, and can further be developed for implementing basic security applications as well.
- The real world does not always include regular and perfect shapes but also irregular shapes and therefore an approach to building an algorithm that also could detect these shapes as well was tried.
- Basic regular 2D shapes were successfully detected with 100% accuracy. The profile time was approximately 3 seconds.
- Basic irregular 2D shapes were successfully detected with 92% accuracy.
- Shapes in real world images were also detected using the above methods and also with the help of global maxima method using boundary pixels for boundary detection.
- Changes were made to the very first implementation of detecting shapes so as to accommodate the irregular as well as real life objects.
- The changes made were mostly by applying guided filter and various other morphological operations.
- Some of the part implemented also made use of extrema points using regionprops. Using extrema, one can estimate the no. of points in an object and then by finding the length(euclidian distance), one can determine which shape it is. This in particular works very well for uniform objects and the number of corners detected could be any.

FUTURE WORK

- Further proposed work is to work on the algorithm and try to accurately detect real world objects with greater precision.
- Also optimizing the algorithm is a part of the upcoming work so that it would take least amount of time to recognize and identify objects.
- Robotic vision: This particular algorithm could also be used for applications with robots probably on a small scale to recognize objects and help children with their learning process.
- Various mobile and computer based applications could be built using this very concept.
- Submitting a research paper on the very topic is something that the project members look eagerly forward to.

REFERENCES

- http://www-rohan.sdsu.edu/doc/matlab/toolbox/images/morph13.html
- https://in.mathworks.com/help/images/ref/graythresh.html
- http://dspace.unimap.edu.my/dspace/bitstream/123456789/7270/1/Shape%20Matching%20and%20Object%20Recognition.pdf
- http://www.sciencedirect.com/science/article/pii/S1047320399904133
- http://homen.vsb.cz/~lud0016/NM/matlab_guide.pdf
- https://in.mathworks.com/matlabcentral/fileexchange/15491-shape-recognition
- www.Wikipedia.org
- Google images
- Shalinee Patel, Pinal Trivedi, Vrundali Gandhi, Ghanshyam I. Prajapati, ”2D Basic Shape Detection Using Region Properties”,International Journal of Engineering Research & Technology (IJERT), Vol. 2 Issue 5, May - 2013.

- Sanket Rege, Rajendra Memane, Mihir Phatak, Parag Agarwal,"2D GEOMETRIC SHAPE AND COLOR RECOGNITION USING DIGITAL IMAGE PROCESSING", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering ,Vol. 2, Issue 6, June 2013.

- Pahul Preet Kaur and Dr.Bikrampal,"2D GEOMETRIC SHAPE RECOGNITION USING CANNY EDGE DETECTION", International conference on computing for sustainable global development,2016.

- Nitasha and Reecha Sharma, "Circle detection using modified canny edge detection algorithm", International journal of research & technology(IJERT),Vol.1,issue 3,may 2012.

- Shambhavi Vijay Chayya, Sachin Khera, Pradeep Kumar,"BASIC GEOMETRIC SHAPE DETECTION AND PRIMARY COLOUR DETECTION USING IMAGE PROCESSING IN MATLAB", International journal of research in engineering and technology",Vol.4,Issue 5,may 2015.

- Shalinee Patel, Pinal Trivedi, and Vrundali Gandhi, "2D BASIC SHAPE DETECTION USING REGION PROPERTIES", International Journal of Engineering Research & Technology, vol. 2, no. 5, pp. 1147-1153, May 2013

- M. Hagedoorn, " PATTERN MATCHING USING SIMILARITY MEASURES", PhD thesis, Universiteit Utrecht, 2000

- Textbook: Digital Image Processing 3rd Edition, Rafael C Gonzalez and Richard E Woods.

---------- x ---------- x ----------