

## 2D Basic Shape Detection Using Region Properties

Shalinee Patel\*

Dept of Information Technology  
SVMIT Engineering College  
Bharuch, Gujarat, India

Pinal Trivedi

Dept of Information Technology  
SVMIT Engineering College  
Bharuch, Gujarat, India

Vrundali Gandhi

Dept of Information Technology  
SVMIT Engineering College  
Bharuch, Gujarat, India

Ghanshyam I. Prajapati

Dept of Information Technology  
SVMIT Engineering College  
Bharuch, Gujarat, India

### Abstract

The Basic 2D Object Detection refers to identify a location that identify and register components of a particular object class at various levels of detail [1]. Image Processing Algorithms are the basis for Image Computer Analysis and Machine Vision [1]. The goal of Basic 2D Object Detection system is to identify the basic geometric shape of objects present in image. It uses some Image Processing algorithms and techniques to detect objects from the image and compare it with the properties of basic geometric shapes to classify that which object is similar to which particular geometric shape like circle, triangle, square and rectangle. The detection of edges of objects from an image is done by using Edge Detection technique [14]. After detecting edges of objects it recognizes the objects, having basic geometric shape. The Edge Detection of object is done by using Canny Edge Detection technique. After that detected objects are labeled as a region. And then, region properties are applied to each region to identify and recognize the shape of that region.

**Keywords:** Canny Edge Detection, Image Segmentation and Shape Recognition.

### 1. Introduction

There are various techniques available to detect object of a particular geometric shape from 2D image. But they are not much reliable techniques that identify features of objects of an image and recognize the object having geometric shape like

circle, square, rectangle and triangle. To identify the shape of detected object, generally the technique was developed is using region properties. The Basic 2D Object Detection is a technique in which, it will identify the shape of object using Edge Detection Technique and Region Properties together to get more reliable and accurate result from other methods of object detection.

The Basic 2D Object Detection is divided in three phases. First is Edge Detection. In this phase, it detects edges of objects of input image using Edge Detection technique. The Edge Detection technique is detects the edges of the objects. So, we can easily separate the objects of given image. Then the second phase is Image Segmentation, in which each object is separated by labeling each region. And finally the third phase is Shape Recognition that identifies the shape of each object or region of an image and recognizes the object having basic geometric shapes like circle, square, rectangle or triangle. This will be achieved by applying region properties to each detected object or region of an input image.

### 2. Literature Survey

As, the Basic 2D Object Detection identifies the shapes of various objects of input image; there is already some techniques or methods are developed for the same goal. But the results of those methods are not much reliable as per requirement. This is because of noisy image, inaccurate detection of object and equal resultant parameter of region properties for a different shape.

So, the Basic 2D object detection is a method in which these problems are eliminated and it becomes more reliable as compared to other methods using edge detection technique with region properties.

## 2.1. Edge Detection

Edges define the boundaries between regions in an image [14]. It detects the edges of the objects. Edge Detection helps in Image Segmentation and Object Recognition [15]. Edge Detection produces an edge map which contains important information about the image. There are many ways to perform Edge Detection. One of them is Gradient based Edge Detection and second one is Laplacian based Edge Detection [4].

There are various popular Edge Detection techniques available, for example Sobel Edge Detection, Prewitt Edge Detection, Robert Edge Detection, Canny Edge Detection etc. [6].

However, Canny's Edge Detection Algorithm performs better than Sobel and Prewitt Operators under noisy condition [2]. Therefore, In Basic 2D Object Detection the Canny Edge Detection technique is applied to detect the object from the input image.

### 2.1.1. Canny Edge Detection Technique

The Canny Edge Detector is widely considered as the standard Edge Detection algorithm in Image Processing [5].

The main steps of canny Edge Detection technique are given below.

- First step is to filter out any noise from the original image before trying to locate and detect any edges.
- Gaussian Filter is used for noise reduction and blurs the image by applying the Gaussian Filter convolution with the original images [15]. The Gaussian Filter can be computed using a simple mask; it is used exclusively in the canny algorithm.
- After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image.

To find gradient of image it will use a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown in Figure 1 [8].

-1	0	+1		+1	+2	+1
-2	0	+2		0	0	0
-1	0	+1		-1	-2	-1
				Gx		Gy

**Figure 1. Masks**

The magnitude, or edge strength, of the gradient is then approximated using the formula:

$$|G| = |G_x| + |G_y| \quad (1)$$

- Then direction of the edge is computed using the gradient in the x and y directions. Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image [6].

The formula for finding the edge direction is just [8]:

$$\Theta = \text{inv} \tan (G_y / G_x) \quad (2)$$

- After the edge directions are known, non-maximum suppression has to be applied. Non-maximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge [5].
- Finally, hysteresis is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold [14].

## 2.2. Image Segmentation

In computer vision, Segmentation is the process of partitioning a digital image into multiple segments. The goal of Segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze [12].

Image segmentation refers to the major step in image processing in which the inputs are images and, outputs are the attributes extracted from those images. Segmentation divides image into its constituent regions or objects [12].

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. The pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture [12].

### 2.3. Region Properties

The closed areas of an Image are considered as region. All regions have some properties like Area, Perimeter, Centroid etc. So, basically the Region Properties are refers to the mathematical features of a particular region of image [7]. Here are some basic properties of regions computed without using the function.

#### 2.3.1. Regionprops

Regions can either describe boundary-based properties of an object or they can describe region-based properties [10]. It measures a set of properties for each labeled region  $L$ .  $L$  can be a label matrix or a multidimensional array [3]. When  $L$  is a label matrix, positive integer elements of  $L$  correspond to different regions [3].

#### 2.3.2. Properties of Regions in 'regionprops'

'regionprops' computes all the shape measurements, listed in Table 1, if called with a grayscale image. 'regionprops' also returns the pixel value measurements are listed in Table 1 [11].

**Table 1. Properties of region**

Area	EulerNumber	Orientation
BoundingBox	Extent	Perimeter
Centroid	Extrema	PixelIdxList
ConvexArea	FilledArea	PixelList
ConvexHull	FilledImage	Solidity
ConvexImage	Image	SubarrayIdx
Eccentricity	MajorAxisLength	
EquivDiameter	MinorAxisLength	

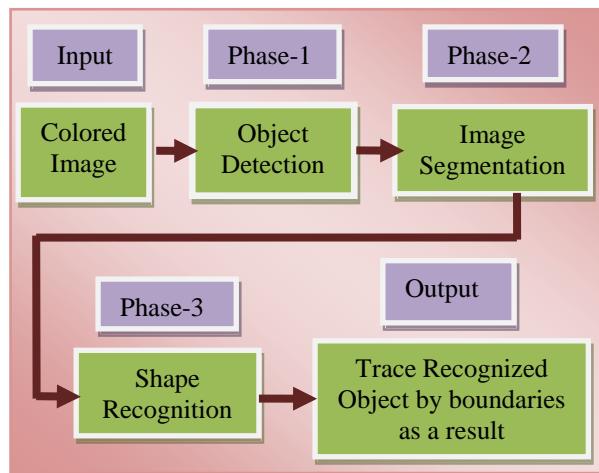
If properties are the string 'all', 'regionprops' computes all the preceding measurements. If properties are not specified or if it is the string 'basic' 'regionprops' computes only the 'Area', 'Centroid', and 'BoundingBox' measurements [11].

### 3. The Proposed Method

Three phases of Basic 2D Object Detection are Edge Detection, Image Segmentation and Shape Recognition.

The input for this system can be any colored image. And output of this system is shape recognized objects with traced boundaries in original image.

Following Figure 2 shows the flow of Basic 2D Object Detection system.



**Figure 2. Phases of Basic 2D Object Detection**

#### 3.1. Phase 1- Object Detection

In this phase, first of all the input image will be converted into grayscale image. If it is a color image it generate a grayscale image as shown in Figure 4(a). It follows the steps described in Figure 3. So, as per threshold value it detects the edges of various objects in image as shown in Figure 4(b).

##### Steps of Phase-1

**Input:** Any colored Image

**Output:** Edge detected Image

##### Step1:

Read image and then convert that colored input image into grayscale image.

##### Step 2:

Filter out noise from the image using simple mask.

##### Step 3:

Find the gradient of the image. And denotes  $G_x$  as x gradient and  $G_y$  as y gradient and by using  $G_x$  and  $G_y$  compute the direction of edge.

##### Step 4:

Perform Non maximum suppression.

##### Step 5:

Apply Hysteresis to eliminating streaking.

**Figure 3. Steps of Phase-1**

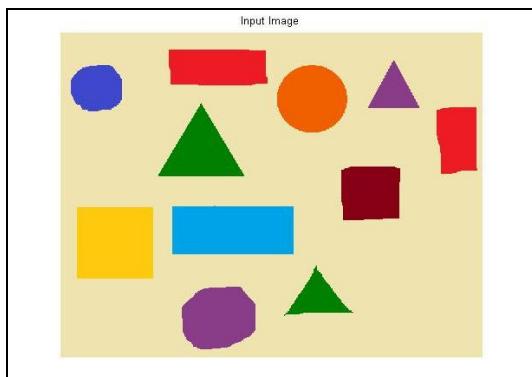


Figure 4. Input Image

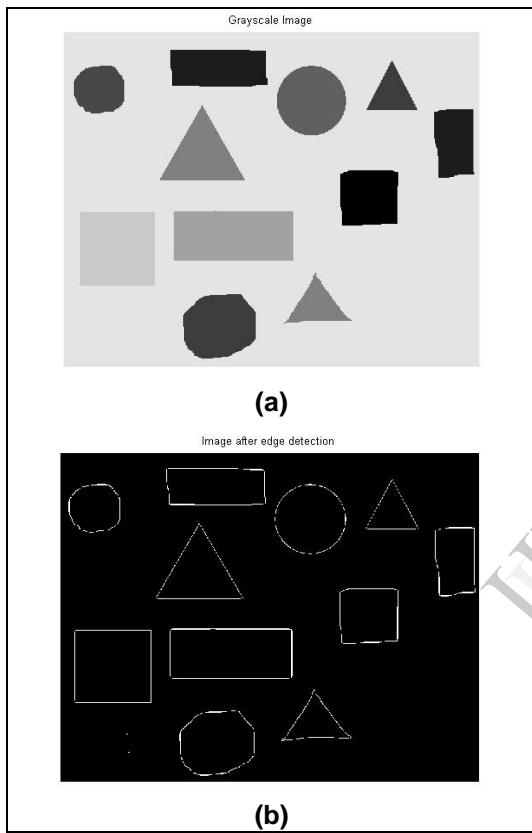


Figure 5. Output of Phase-1 (a) Grayscale Image (b) Image after Edge Detection

### 3.2. Phase 2- Image Segmentation

Segmentation subdivides an image into its constituent parts or objects [6]. In Phase-2, the input will be the output of Phase-1, edge detected image of original image.

The edge detected objects of the image are then filled by the white color as it is a binary image. And create separate regions of the object. Those regions are indicates the objects of the image, labeled by applying labeling method.

By, finding centroid of each region using 'regionprops', print each labeled number on the centroid of each corresponding region. The resultant image of Phase-2 is shown in Fig: 7. Steps of Phase-2 are shown in Fig: 6.

#### Steps of Phase-2

**Input:** Edge Detected Image

**Output:** Segmented Image with labeled region

##### Step 1:

After detecting edges of objects in image the detected objects are considered as a region. They will be converted into a filled separate region.

##### Step 2:

Count Number of filled objects or regions. And label the objects by giving label number to each object.

##### Step 3:

Print label number at the centre of the object by finding centroid of that objects using region property 'centroid'.

Figure 6. Steps of Phase 2

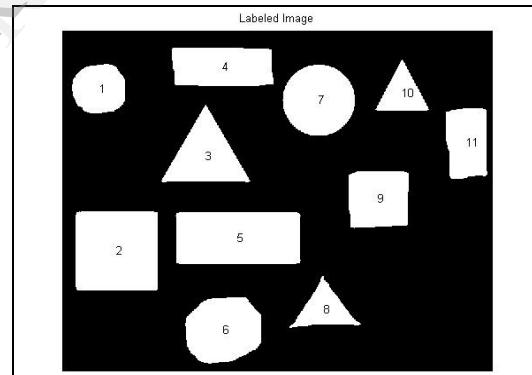


Figure 7. Image with labeled region

### 3.3. Phase 3- Shape Recognition

In the final phase of the Basic 2D Object Detection it will classify and identify objects of different shapes from the image. For this mechanism, it will trace the boundary of objects with different color for different shape.

The classification of objects of particular image is done on the basis of properties of that object's region. The region properties used to identify the shape of the object are 'Area', 'Perimeter', 'Boundingbox', 'Centroid', 'Eccentricity', 'Extent'

and 'FilledArea'. The shape of the object is identified using following formulas of 'metric' and 'circularity'.

$$metric = \frac{4\pi * Area}{perimeter^2} \quad (3)$$

$$circularity = \frac{perimeter^2}{4\pi * Area} \quad (4)$$

Where, Area = area of a region  
Perimeter = Perimeter of a region

### Steps of Phase -3

Input: Image with labeled region

Output: Shape recognized Image

#### Step1:

After finding and labeling the regions of an input image find the properties of regions using 'regionprops' function.

#### Step2:

Find circularities, eccentricity and metric using region properties 'perimeter', 'area' and 'eccentricity'.

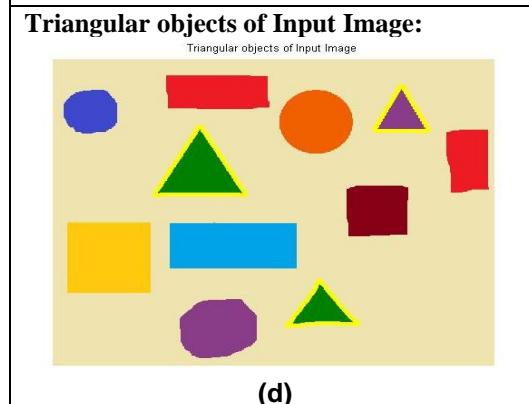
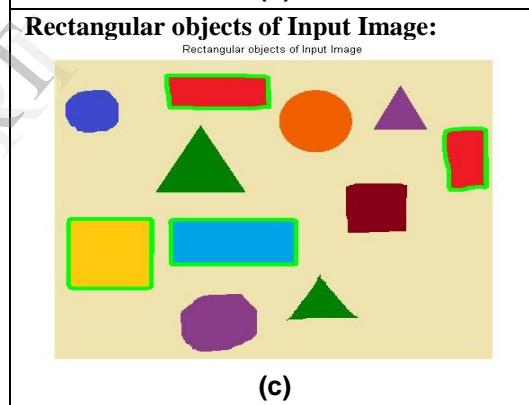
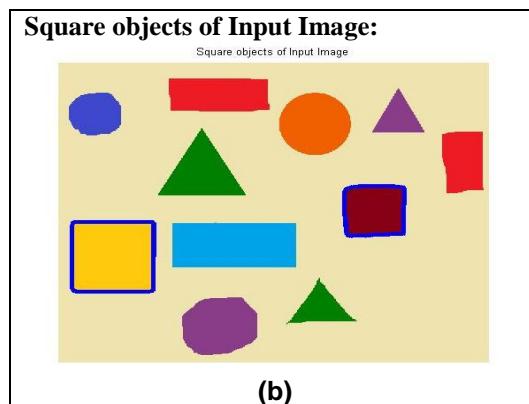
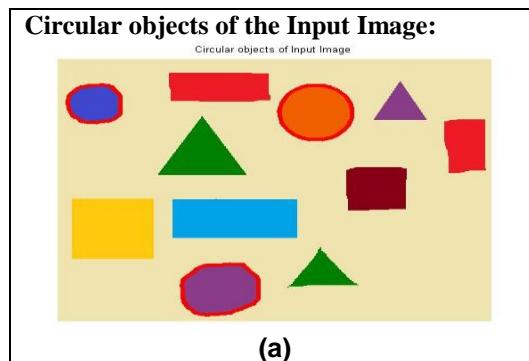
#### Step3:

Identify the shape of object according to the value of circularities, metric and eccentricity.

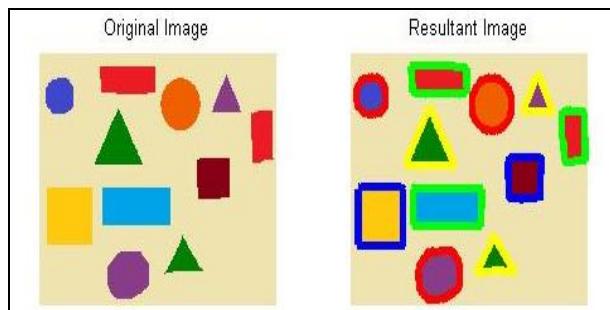
**Figure 8. Steps of Phase-3**

- If the object is similar to 'circle', then it will trace the object by 'red' boundary as shown in Figure 9(a).
- If object is similar to 'square' then it will trace the object by giving 'blue' boundary as shown in Figure 9(b).
- If object is similar to 'rectangle' then it will trace the object by giving 'green' boundary as shown in Figure 9(c).
- If object is similar to 'triangle' then it will trace the object by giving 'yellow' boundary as shown in Figure 9(d).

The final output of the Basic 2D object Detection is shown in Figure 10, in which shape of all objects are identified by different colored boundary together.



**Figure 9. Output of Shape recognition**



**Figure 10. Final output of Basic 2D object Detection**

#### 4. Implementation and Analysis

Implementation of the Basic 2D Object Detection is shown in following table. It will detect the different shape of each object of the input image using region properties.

**Table 2. Implementation of Basic 2D object Detection**

Input Image	Object Detection	Image segmentation	Final Output

#### 4.1. Accuracy of Basic 2D Object Detection

$$\text{Accuracy is calculated as} = \sum_{i=1}^n \frac{\Sigma \text{TD} \times 100}{\Sigma \text{FD} + \Sigma \text{TD}} \quad (5)$$

Where, i = Image 1 to n.

TD = True detection of objects in the image.

FD = False detection of objects in the image.

N = Total number of images [3].

**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.

#### 5. Conclusions and Future Work

In this paper Basic Shape Detection using properties of 'regionprops' is proposed, the shapes of circle, square, rectangle, triangle are detected. This is an efficient and new method for shape detection of different objects. By integrating canny edge detection technique and regionprops properties this method detect the shape of an object more appropriately. With the successful result in Shape Detection of different objects using 'regionprops', this paper focus on same area which need to work on and detect the overlapped objects of input image. This motivates to improve the efficiency of the system. The future work mainly focuses on achieving successful detection of the images in which objects are overlap with each other.

#### References

- [1] Alberto Martin and Sabri Tosunoglu, "Image Processing Techniques for Machine Vision", Florida International University Department of Mechanical Engineering.
- [2] E.Nadernejad, S.Sharifzadeh and H.Hassanpour, "Edge Detection Techniques: Evaluations and Comparisons". Applied Mathematical Sciences, Vol. 2, 2008, pp. 1507 – 1520.

[3] Harpreet Kaur and Manpreet Kaur, "Modified Shape prediction algorithm using over segmentation". International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 4, June 2012 ISSN: 2278-0181.

[4] John J. Oram, James C. McWilliams and Keith D. Stolzenbach, "Gradient-based edge detection and feature classification of sea-surface images of the Southern California Bight", *Remote Sensing of Environment* 112 (2008) 2397–2415, 17 November, 2007.

[5] Lijun Ding and Ardesir Goshtasby, "On the Canny edge detector", *The Journal of Pattern Recognition* 34 (2001) 721}725, January, 2000.

[6] Mr. Salem Saleh Al-amri, Dr. N.V. Kalyankar and Dr. Khamitkar S.D, "Image Segmentation By Using Edge Detection", Salem Saleh Al-amri et. al. / (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 03, 2010, 804-807.

[7] Rafael C. Gonzalez and Rechard E. Woods, "Digital Image Processing", ISBN-978-81-317-2695-2, 2009.

[8] Raman Maini & Dr. Himanshu Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques". *International Journal of Image Processing (IJIP)*, Vol. 3.

[9] Raman Maini, J.S.Sohal, "Performance Evaluation of Prewitt Edge Detector for Noisy Images", *GVIP Journal*, Vol. 6, Issue 3, December, 2006.

[10] Raul Queiroz Feitosa, Gilson A. O. P. Costa, *Computer Vision Part I: Image Segmentation*, October, 2010, pp. 2-118.

[11] Rupali Kate, Dr. Chitode. J. S, "Number Plate Recognition Using Segmentation", *International Journal of Engineering Research & Technology (IJERT)* Vol. 1 Issue 9, November- 2012 ISSN: 2278-0181.

[12] Rupinder Singh, Jarnail Singh, Preetkamal Sharma, Sudhir Sharma, "Edge based region growing". *Int. J. Comp. Tech. Appl.*, Vol. 2, pp. 1122-1126.

[13] S. Jansi and P. Subashini, "Optimized Adaptive Thresholding based Edge Detection Method for MRI Brain Images", *International Journal of Computer Applications (0975 – 8887)* Vol. 51– No.20, August 2012.

[14] Sushil Kumar Singh and Aruna Kathane, "Various Methods for Edge Detection in Digital Image Processing", *IJCST* Vol. 2, Issue 2, June 2011.

[15] T. Kitti, T. Jaruwan and T. Chaiyapon, "An Object Recognition and Identification System Using the Harris Corner Detection Method", *International Journal of Machine Learning and Computing*, Vol. 2, No. 4, August 2012.

[16] Wenshuo Gao, Lei Yang, Xiaoguang Zhang and Huizhong Liu, "An Improved Sobel Edge Detection", 978-1-4244-5540-9/10/\$26.00 ©2010 IEEE.

[17] Yali Amit, "2D Object Detection and Recognition, Models, Algorithms and Networks". The MIT Press Cambridge, Massachusetts London, England, 2002.