Algorithms of Distance, color & Shape Detection for 2-D Images

[1]V.K.PATIL [2]Gajanan Alaspure [3]Trupti Jakkulwar [4]Ganesh Pawar, [5]Aishwarya Bajoriya, [6]Sampada Bulbule

[1]Asst.Prof,Department of E&TC AISSMS's IOIT,Pune, Maharashtra, India
[2,3,4,5,6]Students,Department of E&TC AISSMS's IOIT,Pune, Maharashtra, India

***Abstract*—**This paper represents our work on non contact distance calculation method. It is based on pixel variation of 2D images. This can be done by referencing two points in the image frames. The methodology of distance calculation starts with capturing of particular object image from any distances. Again capture another image of same object with some known distance. If an object goes far from the camera the object has a reduced size in the image. As the object moves closer to the camera the object becomes comparatively bigger. Thus we get the variation of pixel size between two images. Using the pixel variation we can calculate the distance of the object from the camera .This paper focuses on recognition of 2D shapes of objects such as square, circle, triangle, rectangle and ellipse as well as colors of objects. The algorithm was simulated and developed using MATLAB.

***Keywords*—**2D Images, Image processing, MATLAB, Extent, Bounding Box, Object detection, Digital Camera, Shape and color recognition, non contact distance measurement.

*****

I. Introduction

Ultrasonic based and laser based techniques are most commonly used methods for non-contact distance measurement [1]-[2]. Difficulties in using these techniques are that the accuracy of these techniques heavily depends on the surface reflectivity of the object whose distance is to be calculated. These create problems for the object having poor surface reflectivity such as objects made up of woods etc. Hence it causes in loss of accuracy for that object. Recording an image of the particular object during measurement is also an issue for the above mentioned techniques [1]-[2].

Alternately image based techniques have been proposed for distance measurement purpose that used digital camera also called as CCD camera. This technique however requires two cameras at different positions for capturing an image of the object under test. It requires pattern recognition or image analysis of whole image frame for the sake of extracting features from the images for obtaining distance of that object from the digital camera. Therefore it requires large amount of storage capacity and high speed DSP processors. So the disadvantages by using these methods are processing speed, establishment cost and complexity [1]. From the above mentioned techniques, we can conclude that image analysis was not suitable with real time measurements. Because of the problems and difficulties in the above mentioned methods, accurate and reliable measurements were not always guaranteed in real world applications [2].

Corresponding author
[1]varshapatil101@gmail.com

To overcome the problems and difficulties encountered via existing image-based distance measuring methods, this paper presents distance measurement method based on pixel number variation in images of digital cameras by referencing to two arbitrarily designated points in image frame rather than laser projected spots in an image [2]. A general observation is that if an object goes far from the camera the object has a reduced size in the image. As the object moves closer to the camera the object becomes comparatively bigger [2].From this we get the variation of pixel size between two images.

This paper also includes shape detection and color detection. For shape detection bounding box is used. Depending upon the extent which is simply a ratio of area of an object to the area of a bounding box, shape is easily get detected. For color detection, color of an object gets compared with the standard colors.

The remaining part of this paper is discussed as follows. Section II contains relationship between pixel variation and distance of the image. Section III involves the brief view of our proposed method. Calculations are derived in section IV. Section V describes all the applications related to our topic i.e. distance calculation and shape detection etc. Section VI contains experimental results. Lastly, conclusion is in section VII.

II. RELATIONSHIP BETWEEN PIXEL VARIATION COUNT AND DISTANCE OF THE IMAGE

We are assuming that surfaces or objects are perpendicular to the optical axis of digital camera. In the previous researches, it was proved that, on image frame scan line distance and pixel counts are closely related [3]-[4]. Computing actual distance of object decomposed into computing the depth and lateral distance of the object. It has been seen that object height is closely related to the pixel height of the object [2].

The Figure 1 shows that a single digital camera captures the image of the same object twice. Let us assume that it capture the first image at a photographic distance h1. Now repositionised the camera and again capture another image of the same object i.e. from another distance h2. Let hs be the scan line i.e .horizontal scan line. We are assuming that distance L between reference points will remains fixed [2].

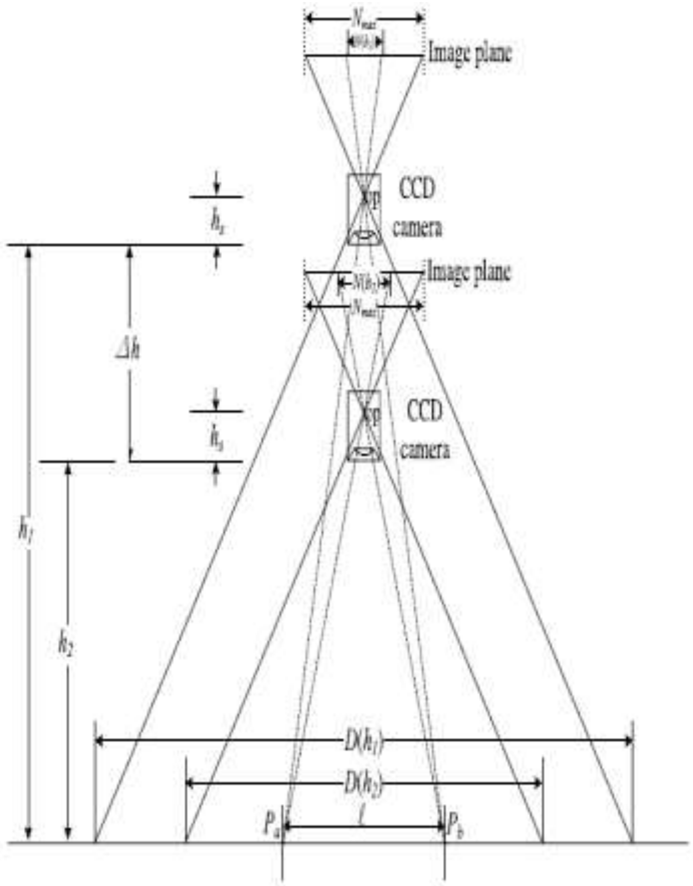

Fig.1. Schematic diagram of distance calculation of object from camera using pixel variation

So the relation between distances and pixel count can be given as:

$$\frac{l}{N(h1)} = \frac{D(h1)}{Nmax} \quad (1)$$

$$\frac{l}{N(h2)} = \frac{D(h2)}{Nmax} \quad (2)$$

$$\frac{D(h1)}{D(h2)} = \frac{N(h2)}{N(h1)} \quad (3)$$

Because there is a displacement, Δh=h1-h2,i.e. from the camera movement along the photographic directions [2]. Hence we can write as:

$$\frac{D(h1)}{D(h2)} = \frac{N(h2)}{N(h1)} = \frac{h1+hs}{h2+hs} \quad (4)$$

But h1=h2+Δh. putting this value in equation 4 as:

$$h2 = \frac{N(h1)}{N(h2)-N(h1)} * \Delta h - hs \quad (5)$$

In a Similar way, if camera moves in forward direction then photographic distance becomes:

$$h1 = \frac{N(h2)}{N(h2)-N(h1)} * \Delta h - hs \quad (6)$$

In the above equation, value of Δh- hs is known to us so that the distance is calculated [2].

III. ANGLE AND DISTANCE MEASUREMENT FOR THE OBJECT

Distance measurement by using the above mentioned technique is only applicable for the objects that are exactly perpendicular to the optical origin of the camera. It is possible to calculate distance for the objects which are perfectly perpendicular by using the above mentioned method. Here we include angle measurement for the objects having oblique plane,. The accuracy using angle measurement will also be greater.

Fig.2 shows the schematic of measurement of object distance on an oblique plane based on pixel variation.

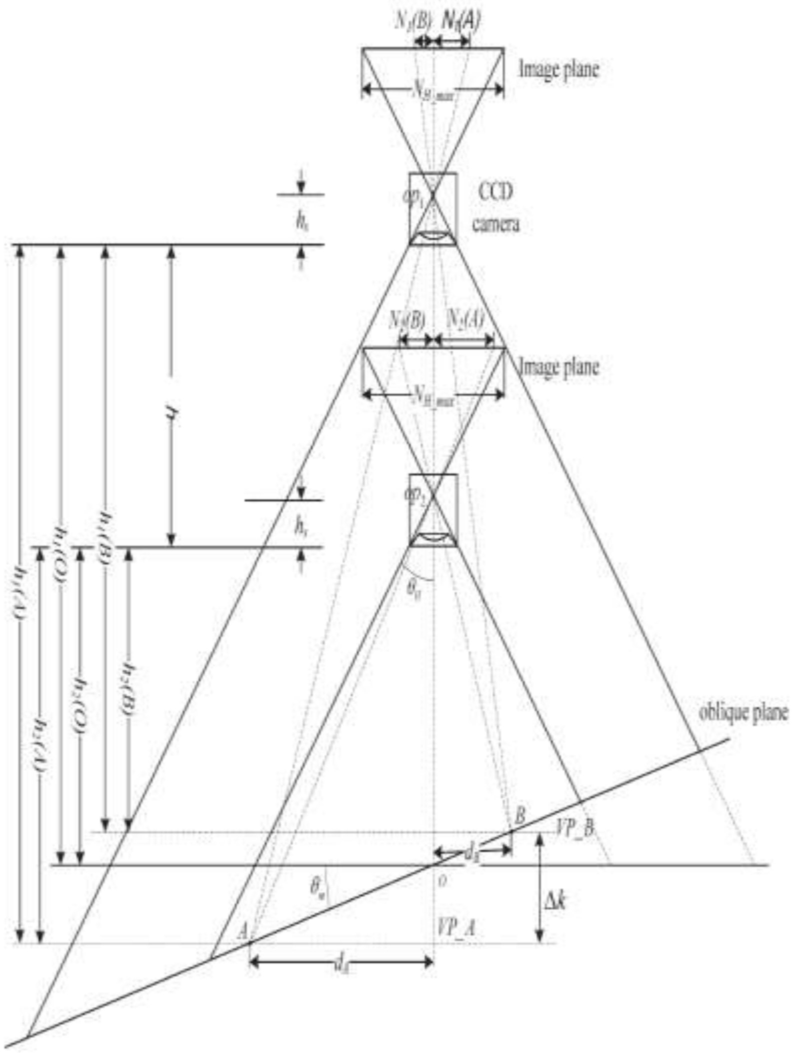

Fig.2. Schematic diagram of distance calculation of object on oblique plane

As shown above there exists an angle i.e. θm between the perpendicular and oblique plane. In the figure, point B is much far as compare to point A. Hence it is not possible to calculate distance by using above mentioned method [2].

To solve the above mentioned method, concept of virtual planes used. Consider the two virtual plane VP_A and VP_B [2]. Therefore the distance between virtual planes and camera can be given by:

$$h1(A) = \frac{N2(A)}{N1(A) - N2(A)} * \Delta h - hs \quad (7)$$

$$h2(A) = \frac{N1(A)}{N1(A) - N2(A)} * \Delta h - hs \quad (8)$$

$$h2(B) = \frac{N2(B)}{N1(B) - N2(B)} * \Delta h - hs \quad (9)$$

$$h2(B) = \frac{N1(B)}{N1(B) - N2(B)} * \Delta h - hs \quad (10)$$

Here h1(B) and h1(A) are respectively the distances from camera 1 to virtual planes VP_B and VP_A [2]. Similarly h2(B) and h2(A) are respectively the distances from camera 2 to virtual planes VP_B and VP_A. From (7) and (10),

$$h1(A) - h1(B) = h2(A) - h2(B) = dA + dB.(tan\theta m) \quad (11)$$

$$\theta m = \tan - 1\left(\frac{\Delta K}{dA + dB}\right) \quad (12)$$

In the above equation Δk is nothing but the distance between virtual points .dA and dB can be expressed as follows:

$$dA = \frac{2N1(A) * N2(A)}{\big(N2(A) - N1(A)\big)NHmax} * \Delta h * tan\theta H \quad (13)$$

$$dB = \frac{2N1(B) * N2(B)}{\big(N2(B) - N1(B)\big)NHmax} * \Delta h * tan\theta H \quad (14)$$

The distance between points A and B can be represented to examine the position of an real world object. This distance can be represented as D(AB):

$$D(AB) = (dA + dB) * sec\theta m \quad (15)$$

Thus using the above equation it is possible to calculate the distance of an oblique plane.

IV. SHAPE DETECTION OF THE OBJECT

The process of shape detection starts by capturing the image from any distance with the help of digital camera. We shall consider that the acquired image is in RGB format which is true color format for an image [5]. The next step is to convert RGB image into the black and white image. To convert the image into black and white it should be first converted into 2D gray scale image. The gray scale image is nothing but matrix that holds luminance value of the image [5]. We obtain the luminance values by combining RGB values using NTSC standard equation given by:

$$Y=0.11B+0.59G+0.3R \quad (7)$$

This image is then converted into black & white image by the process called thresholding [5]. The threshold is set and the luminance of each pixel is compared. The values greater than threshold are replaced with logical one i.e. white and the values lesser than one are replaced with logical zero. The image is now two dimensional arrays with binary elements. Boundaries of an object are recognized by first setting a single pixel on object background as starting point and moving on clockwise or anti-clockwise direction searching for the other object pixel [5].
In this way the objects boundary can be found. After finding the boundary area can be easily calculated by summing the pixels within the boundary extent. Next step is to find the bounding box of an object. The bounding box is an imaginary rectangle that completely encloses the object. Figure 3 shows the concept of bounding box [5].

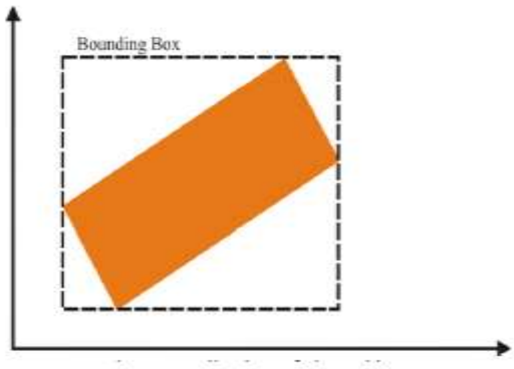

Fig.3: Bounding box

The dimensions of the bounding box are variable. The dimensions i.e. width and height are adjusted according to the shape of the object [5]. If the object has square shape then there exists a bounding box of equal height and width. But if the object is rectangular shaped then exists a bounding box having width greater than height. After forming the bounding box next step is to find an extent for the object which is nothing but ratio of area of an object to the area of the bounding box [5].
Following flowchart shows for shape detection and color detection [5].
The value of this extent is different for different shapes. Its value is 1.000 for square and rectangle and less than 1 for circles.
Hence the conclusion is made from this is that if the value of extent is much less than 1.000 then it is either circle or ellipse. The difference between circle and ellipse is that the eccentricity of circle is zero and for ellipse it is non zero. Hence if there is a non-zero value of eccentricity then it must

be ellipse and if not then it must be a circle. But if the extent is equal to 1.000 then it is either square or rectangle. The decision of whether the given object is square or rectangle is taken on the basis of object dimension i.e. the object with equal sides is square and with different sides is rectangle. For triangles, we know that the area taken by the triangle in the bounding box is much less. Hence if the extent of the object is less than 0.66 then it must be a triangle. So in this way we can differentiate between the above mentioned objects.

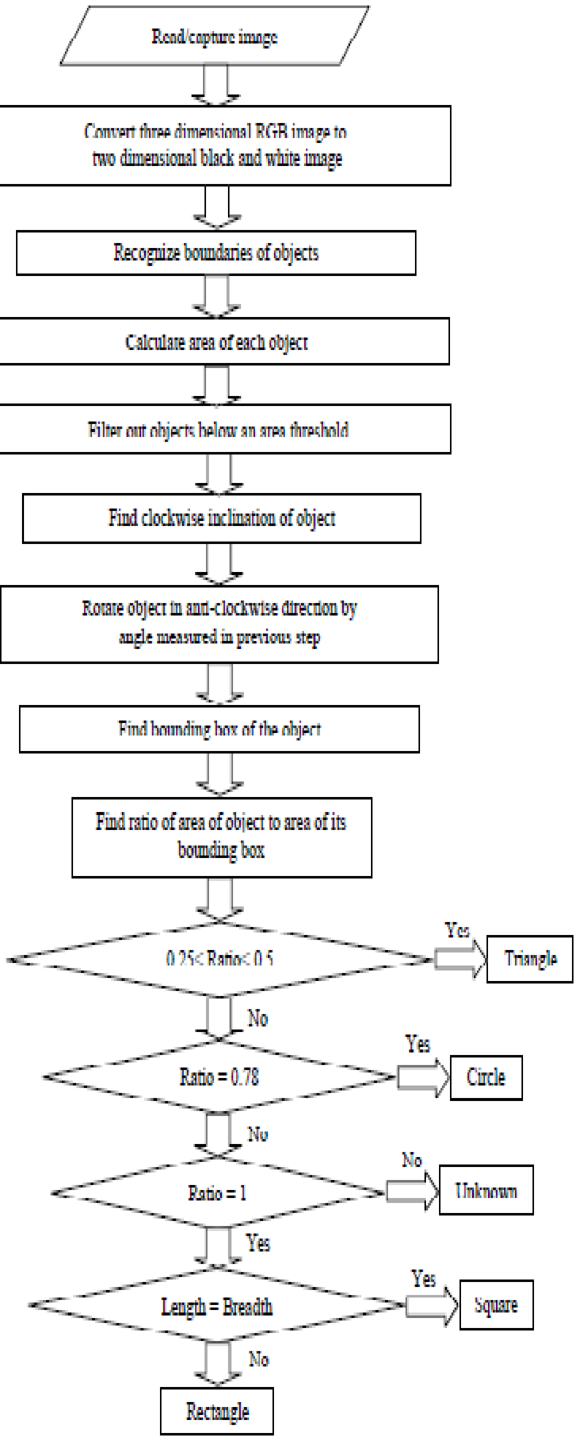

Fig.4: Flowchart for shape detection and color detection

V. COLOR DETECTION OF THE OBJECT

The first step in determining the object color is to initialize all the necessary colors i.e. white, black, blue, red and green in MATLAB. These colors are initialized in terms of their R, G and B values. After initializing the color it will first check for the color of centre pixel of an object. Hence it is necessary to have the object color same as that of centre color. For this there is a function available in the MATLAB. After checking the centre pixel color it will compare that color values with the above initialized color. Resultant values are again checked and the color that has same values will be the color of an object. Hence by using the concept of the additive mixing number of colors can be found.

Observation Table

Object Shape	***Color***	***Camera movement distance***	***Actual Distance***	***Observed Distance***
Triangle	Orange	10	46	48.21
Ellipse	Orange	5	56	61.211
Rectangle	Yellow	15	60	62.109
Square	Blue	8	40	41.339
Circle	Green	10	46	47.465

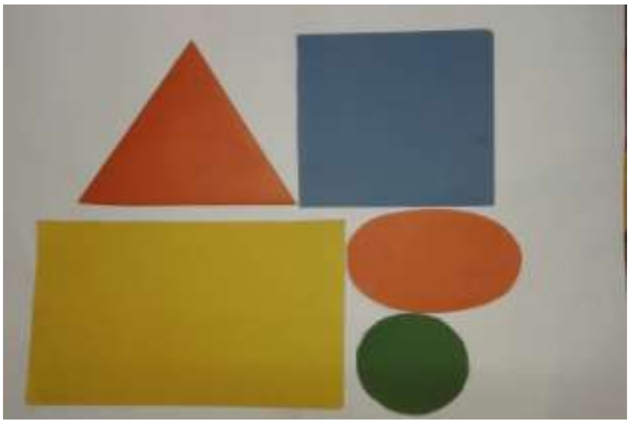

Fig.5: All Shapes used for Experimental Work

Fig.6: Setup

ACKNOWLEDGMENT

Our work is done under the guidance of Prof. V. K. Patil. We would like to thank her for the suggestion to perform shape detection, color detection and distance calculation on 2D images. We also like to thanks AISSMS IOIT, PUNE for providing the resources.

REFERENCES

[1] Chen-Chien James Hsu, Ming-Chin Lu and Yin-Lu Yu,"Angle and distance measurement of objects on an oblique plane based on pixel number variation of CCD images

[2] Chen-Chien James Hsu, Ming-Chin Lu, Yin-Lu yu on "Distance measurement based on pixel variation of CCD Images."

[3] Carullo A, Ferraris F, Graziani S. Ultrasonic distance sensor improvement using a two level neutral network. IEEE Transaction on Instrumentation and measurement.

[4] C.C. Hsu and M.C. Lu, , "Angle and distance measurement of distant objects based on pixel variation of CCD images". May 3–6, 2010, pp. 318–322.

[5] Song KT, Tang WH. Environment perception for a mobile robot using double ultrasonic sensor and a CCD camera.

[6] G. Borgefors, "Distance transformations for arbitrary dimensions" Dig. Image Processing, Graphics, and computer vision, no. 3, val 27, pp. 321–345, September 1984. oblique plane based on pixel number variation of CCD images"