Shape Detection using Geometrical Features

[1]Shalu Gupta, [2]Y. Jayanta Singh
[1]Dept. of Computer Science & Engineering and I.T,
[1]Assam Don Bosco University, Guwahati, Assam, India
[2]National Institute of Electronics and Information Technology (NIELIT),Kolkata, India
[1]shalu2324@gmail.com, [2]y_jayanta@yahoo.com

Abstract:
In this paper, we have presented an approach for object detection system. This approach is used for detect two-dimensional shapes such as lines, rectangle, square, circle, triangle, polygon, star etc. Proposed technique of shape detection is based on the statistical properties of distribution of points on bitmap image and sub-windowimage of a shape. For recognition, we have considered sub-windowbased features and Nearest Neighbours classifier. By applying these features, we achieve maximum recognition accuracy of 96.7% using 4556 samples.

Keywords: Shape detection, Feature extraction,Geometric features.

INTRODUCTION
With the evolution of human machine interface and dynamic digital information processing algorithms and more commanding microprocessors, it is achievable additional perception of the environment and to detect objects with new technologies. The shape detection is a part of object detection. Object detection system dwells activities, namely, image acquisition, image pre-processing, image segmentation, feature retrieving, object classification, object interpretation and object detection as shown in figure 1.

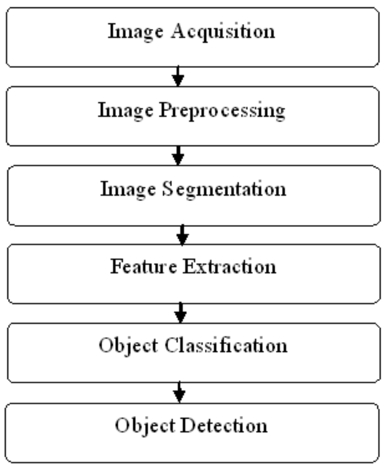


Figure 1: Block diagram of object detection system

A. Image Acquisition
Image acquisition is to acquire an image from the environment using a camera or from already available database and to convert the acquired image converted into digital image format. Pre-processing phase is fueled by digital image.

© 2017 Vidya Publications. **Authors are responsible for any plagiarism issues.**

B. Pre-Processing

Pre-processing is an elementary stage of object detection system. In pre-processing a chain of operations are enacted on digital image i.e. skeletonization, noise removal (filling of holes, clear the boundaries) etc. Skeletonization is performed for contracting the width of an image from piles of pixels to unit pixel or to single pixel. Noise removal is used to remove non-essential bits, which does not play any considerable role in an image and add some pixels, which play an important role in an image. This step increases the chances for success of other steps.

C. Segmentation

In this phase, an input image is segmented into its constituent parts or objects. The image segmentation results, a set of domains that covers together the integral image or a set of curves obtained from the image. In other words, segmentation separates certain features in the image while allowing other part of the image as a background. In case the image consists of a number of features of interest then they can be segmented one after another.

D. Feature Extraction

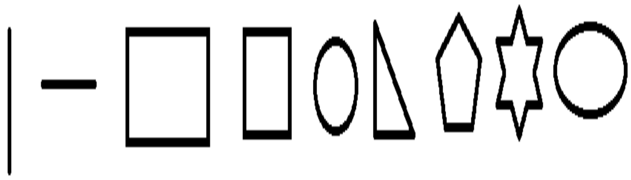
Feature extraction is a prominent step in object detection system. In this phase, we will extract the structural/statistical properties of object for recognition. In this work, we have considered sub-window based features and divide an image into n number of parts each of which is equal size.

E. Object Classification

Classification phase uses the feature which is extracted in the previous for recognition. This step differentiates one class objects from another by using these features. In this study, we have considered nearest neighbor classifier.

F. Object Detection

Detection of object is to identify the object from an image by using digital image processing techniques. The feature extraction step extracts a set of features which can be used for uniquely identifying or grading the shape. There are global and local features which help to detect the objects. Global features include the geometrical shapes, texture, size and coloretc. We propose an approach to detect 2-D shapes such as lines, rectangle, square, circle, triangle, star, polygon, oval, etc. as shown in figure 2.



Vertical Horizontal Square Rectangle Circle Triangle Polygon Star Oval

Figure 2: Model of shapes

RELATED WORK

There is vast volume of literary work on feature extraction techniques and classification techniques; we inspect few approaches nearest to work. Ballard (1981) describes circle detection by using Hough transform. Ullman (1984) uses basic visual operators like color, shape, texture etc. which is integrated in different ways for detection of complicated objects. Manifest from human observation adverts that acquaintance plays an important role in figure/ground assignment by Peterson et al. (1994). Itti et. al. (1998) break down the complicated problem of view understanding by swiftly electing and analyzing in detail. Precise default segmentation of an object from authentic scatter is immensely fussy without the acquaintance of target shape by Palmer(1999). Minut and Mahadevan (2001) have recommended a

© 2017 *Vidya Publications.* Authors are responsible for any plagiarism issues.

visual attention prototype which is based on reinforcement learning of an image. Belongie et. al. (2002)propose the shape context, which conquer for each point the spatial distribution of all other points relative to it on the shape. This semi-local illustration allows establishing of point-to-point communication between different figures of an object even under yielding deformations. Elnagara et. al. (2003) described partition path precisely and then the numerals are separated before restoration is applied. Segmentation uses part detection, but do not use global shape descriptors by Shiet. al. (2003). Lowe (2004) describes SIFT (Scale Invariant Feature Transform) for extracting a large collection of feature vectors which are constant to image translation, scaling and rotation and robust across a substantial range of affine deformation, adding up noise and partly constant to enlightenment changes. In this key locations of an object are defined as maxima and minima of the result of DOG (Difference of Gaussians). Mori et. al. (2005) discussed about connection between the shapes of an object and recognizing these shapes by finding an aligning alteration for correspondences between points on the two shapes. Hanmandluet. al. (2007) and Ranjanet. al. (2009) use grid based features for handwritten Hindi numerals recognition. They have separated the input image into 24 sub-windows and calculate the vector space for each pixel position in the grid from the bottom left corner and normalized. Hebert et. al. (2007) suggest a different way to go away from individual edges, by programming relation between all pairs of edges. Nadernejad et. al. (2008) review different edge detection techniques i.e. the Marr-hildreth Edge Detector, Canny Edge detector, Threshold and Boolean function based edge detection, color and vector angle based edge detection and compare these techniques. Vitto et. al. (2010) presents an object group approach which entirely integrates the paired strengths presented by shape matchers. Toshev et. al. (2012) proposed a ground segmentation technique for drawing out of image regions that are similar to the global properties of a model border construction and chordiogram. Tripathi et. al. (2012) presented a review on different image segmentation techniques i.e. region based and edge based segmentation.

PROPOSED METHODOLOGY

A. Segmentation of an Image

Segmentation is the procedure of isolating an image into multiple segments or regions or a set of pixels available in an image. In this phase, the image is divided into a number of objectsas shown in figure 3.

B. Object Cropping and Resizing

In this phase, the size of segmented object is measured by finding the minimum value of x and y and length and width. In this phase the four elements of an object position are measured i.e. x-min, y-min, width, height that specify the size and location of the crop rectangleas shown in figure 4. After cropping, the image is resized as 128 X 128as shown in figure 5.

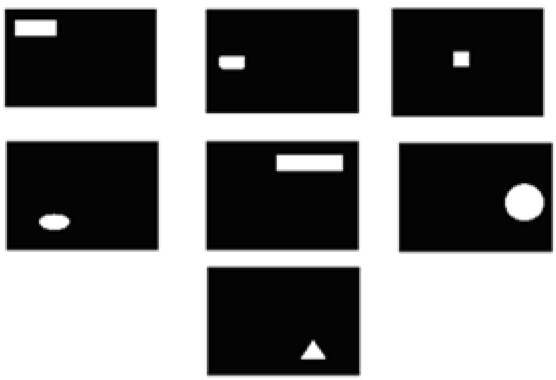


Figure 3: Segmentation of an original image

© 2017 Vidya Publications. **Authors are responsible for any plagiarism issues.**

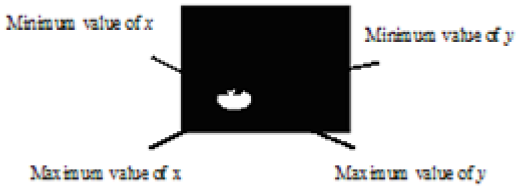


Figure 4: Bitmap of an original image

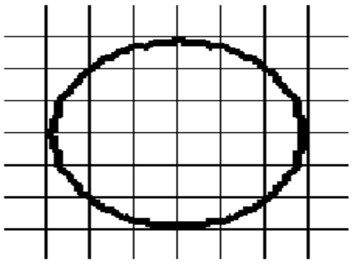


Figure 5: Object cropping and resizing

C. Boundary Structure Model

Boundary structure of an image is to detect the boundary edges of an image or shape. The object or an image is normalized into a window (resize the image) of some size, say, 128×128. It constitutes a perceptually prominent figure/ground union of an image. After normalization, we construct a bitmap image (0 or 1) of normalized image and then perform skeletonization and noise removal. Skeletonization is performed to contracting the width of boundary from piles of pixels to unit pixel or single pixel. Detect the boundary by using erosion and dilation of a shape and remove the extra bits. This procedure of pre-processing is described in figure 6(a) and 6(b).

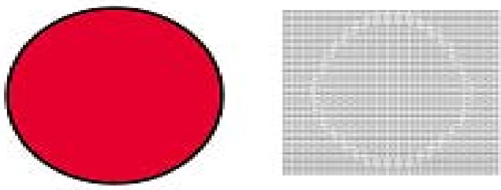


Figure 6: (a) Original Image (b) Bitmap and skeletonization image of an original image.

D. Boundary Structure Segmentation

In this step, the boundary of a shape is divided into n (=64) number of equivalent sub-windows i.e., each one sub-window is of 16×16 pixels. Calculate the number of foreground pixels in each of nsub-windows as shown in figure 7. If the boundary segment has zero foreground pixels then the feature value of corresponding sub-window is considered as zero.

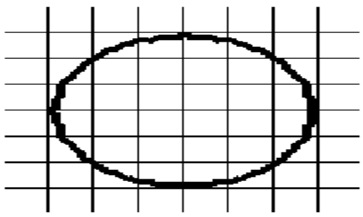


Figure 7: Image is divided into sub-windows.

© 2017 Vidya Publications. Authors are responsible for any plagiarism issues.

E. Diagonal features

In this phase, we have presented diagonal feature extraction method. We have extracted the diagonal features of each sub-window i.e. left and right diagonal. Each sub-window has 31 diagonals. Each diagonal value has been summed up to get a single sub feature of left and right diagonal of each sub-window as shown in figure 8.

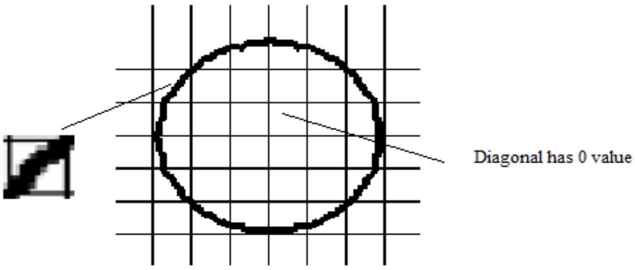


Figure 8: Diagonal feature extraction

Following are the steps that have been used to extract these features which are as follows:

1. Extract the features from the pixels of each one sub-windowby moving next to its diagonals.
2. Each sub-windowhas 31 diagonals; foreground pixels present along each diagonal is summed up in order to get a single sub-feature of each boundary segmented structure.
3. In boundary segmentation whose diagonals do not have any foreground pixel, the feature value of that sub-windowis taken as zero.

By using this diagonal feature, we obtain n features corresponding to each and every boundary segmentation part.

F. Chord Features

Chord is defined as the division of all geometric relationships between pairs of boundary edges, which relates to the interior segmentation. It captures the boundary structure of segmentation as well as the position of interior relative to the boundary of an object. Each chord captures the geometric configuration of two boundary edges and this will help to detect the 2D geometric shape. Each chord (x, y) has its configuration.

In this step, we find the chord of each sub-window by finding the starting position of foreground pixel (x, y) and ending position of foreground pixel (x', y') in each boundary segment and if the boundary segment has zero foreground pixels then the feature value of corresponding sub-window is considered as zero as shown in figure 9. Calculate the directional distance between these points by using this formula,

$$\theta = \tan^{-1}\left(\frac{y'-y}{x'-x}\right) \ldots\ldots\ldots\ldots\ldots(1)$$

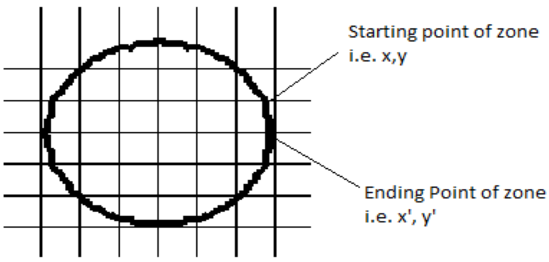


Figure 9: Chord feature extraction

© 2017 *Vidya Publications.* Authors are responsible for any plagiarism issues.

PROPOSED ALGORITHM

- Query image is given by the user.
- Perform preprocessing (filling of holes and removing of extra bits) on a query image.
- Perform segmentation on query image to get the number of objects or sub-images in a query image.
- Perform the resizing and cropping on sub-image.
- Perform skeletonization or thinning and filling of holes on sub-image.
- Calculate the features from the regions of thinning image and store these features as a feature vector. Features are like total number of ON pixels (foreground pixels), chord feature, angle, right and left diagonal etc.
- From the calculated features, detect the shape of an image i.e. circle, rectangle, square, triangle, lines (horizontal or vertical), polygon etc. by using the basic geometrical features of a shape. E.g. a line has an angle of 0 or 90 degree, sides of a square are equal, opposite sides of a rectangle are equal.
- Compare the feature vectors of a query image with the feature vectors of an image database.
- If the feature vectors of a query image match with the feature vectors of an image database then the shape of an object is detected.

CLASSIFICATION

For classification, we have considered nearest neighbours classifier. In nearest neighbor classifier, Euclidean distances from the applicant vector to stored vector are calculated. The Euclidean distance between applicant vector and a stored vector is given by,

$$d = \sqrt{\sum_{k=1}^{N} (x_k - y_k)^2}$$

By using these features, following are the shapes we can determine:

- **Horizontal Straight Line:** If θ is approximately 90^0 and ON/ foreground pixels in each sub-window is greater than the threshold value, in more sub-windows than it is concluded to be a horizontal line. As shown in figure 10, the starting (y) and ending (y') position of y axis approximately remains constant and the difference between starting (x) and ending (x') position of y is some threshold value. If θ value is 0^0 then line is parallel to x axis. This implies that line is horizontal.

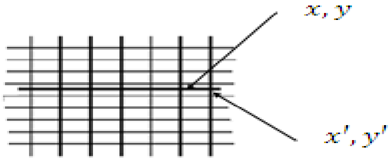


Figure 10: Feature of Horizontal Straight Line

- **Vertical Straight Line:** If θ is approximately 90^0 and ON/ foreground pixels in each sub-window is greater than the threshold value, in more sub-windows than it is concluded to be a vertical line. As shown in figure 11, the starting (x) and ending (x') position of x approximately remains constant and the difference between starting (y) and ending (y') position of y is some threshold value. If θ value is 90^0 then line is parallel to y axis. This implies that line is vertical.

ISSN: 2229-6913 (Print), ISSN: 2320-0332 (Online), UGC Approved Journal (S.No.63019)
Web Presence: http://www.ijoes.vidyapublications.com
© 2017 Vidya Publications. Authors are responsible for any plagiarism issues.

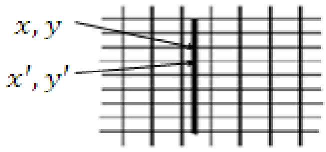


Figure 11: Feature of Vertical Straight Line

- **Circle:** In this, θ values of each sub-window is calculated then the number of positive θ (θ value is greater than 0) of values and number of negative θ (θ value is less than 0) values are calculated as shown in figure 12. Major axis and minor axis of a shape are calculated from the given command in MatLab. If the number of positive and negative θ values are equal with sum threshold value and major and minor axis are equal then it is detected as circle.

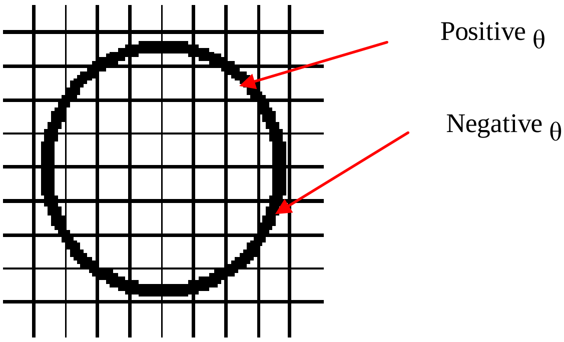


Figure 12: Features of Circle

- **Semi-Circle/Arc:** In this, θ values of each sub-window is calculated then the total number of positive θ (θ value is greater than 0) of values and number of negative θ (θ value is less than 0) values are calculated as shown in figure 13. When the total number of positive θ values are equals to the sum of total horizontal line θvalues and total number of negative θ values then it is concluded to be a semi-circle.

No. of positive θ = No. of negative θ + No. of horizontal line θ

Length of major axis ≠ Length of minor axis

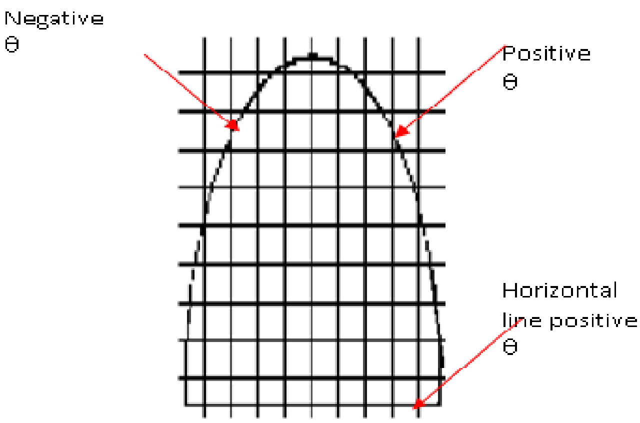


Figure 13: Features of semi-circle

- **Square:** Calculate the horizontal line and vertical lines as described above. Find the total number and location of sub-windows of horizontal and vertical lines. If there are 2 horizontals and 2 vertical lines, number of sub windows of horizontal foreground and number of sub windows of

© *2017 Vidya Publications.* Authors are responsible for any plagiarism issues.

verticalforegrounds are equal, major and minor axis of the shape are equal then it is detected as square as shown 14.

No. of foreground pixel sub-windows of horizontal line = No. of foreground pixel sub-windows of vertical line

Length of major axis = Length of minor axis

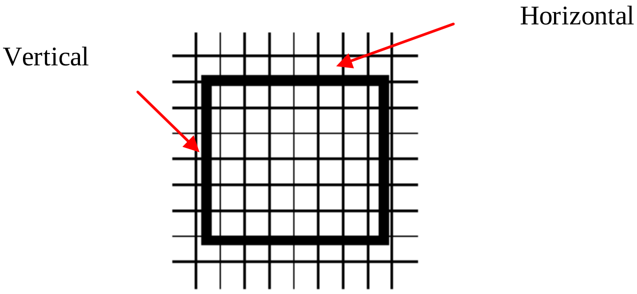


Figure 14: Features of square

- **Rectangle:** Calculate the horizontal line and vertical lines as described above. Find the total number and location of sub-windows of horizontal and vertical lines. When a number of ON/foreground pixel sub-windows in horizontal and vertical line is not equal, then it is concluded to be a rectangle. There are 2 horizontal and vertical lines, but the major and minor axis of the shape is different as shown in figure 15.

Length of major axis ≠ Length of minor axis

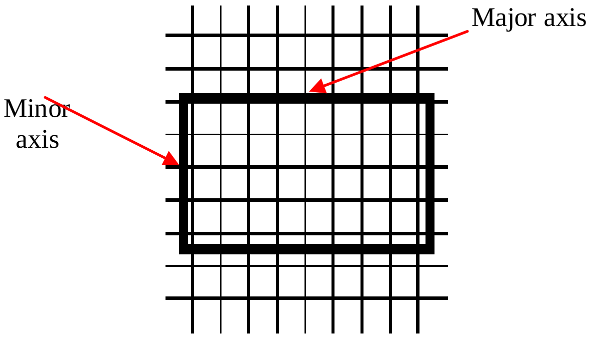


Figure 15: Features of rectangle

EXPERIMENTS AND RESULTS

For experimentation work, we take some samples of shapes from various sources. The experimental results are based on different feature extraction techniques, namely, type of angle of each sub-window, chord features, and diagonal feature, major and minor axis. Here, we considered 4556 samples of different shapes.

We have divided the data set into five different partitioning strategies, namely, a, b, c, d, and e.

a. In the strategy 'a', we have taken 50% data as training dataset and remaining data as a testing dataset.
b. In 'b', we have taken 60% data as training set and 40% data as the testing set.
c. In 'c', we have considered 70% data as training set and remaining 30% data as the testing data set.
d. In 'd', we have taken 80% data as training dataset and 20% data as the testing dataset.
e. In 'e', we have taken 90% data as training dataset and remaining 10% data as the testing dataset.

© 2017 Vidya Publications. Authors are responsible for any plagiarism issues.

We have divided the data set into five different partitioning strategies with accuracy as depicted in Table 1. We have achieved maximum recognition accuracy of 96.7% using nearest neighbours classifier and data set partitioning strategy c as shown in Table 1.

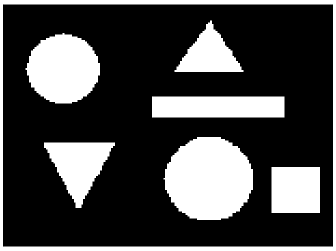


Figure 16: Bitmap Image

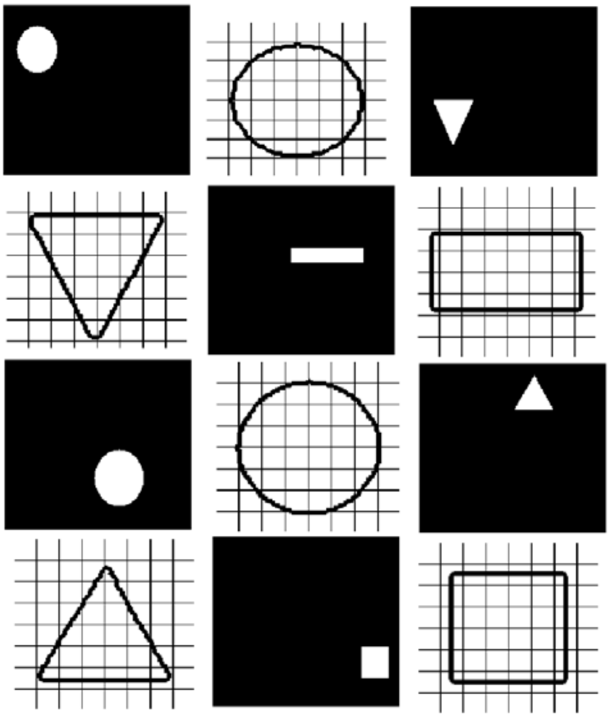


Figure 17: Image segmentation, image cropping and resizing

Table 1: Recognition accuracy achieved using nearest neighbours classifier

Partitioning type	Training data set	Testing data set	Accuracy
a	50%	50%	95.6%
b	60%	40%	96.3%
c	70%	30%	96.7%
d	80%	20%	96.1%
e	90%	10%	96.5%

© 2017 Vidya Publications. Authors are responsible for any plagiarism issues.

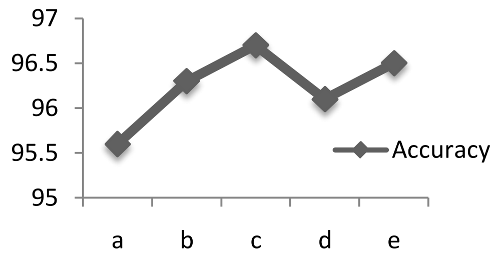


Figure 18: Recognition accuracy using *k*-NN classifier

CONCLUSION

The work presented in this paper proposes some properties to detect 2D geometrical shapes such as line, square, rectangle, circle, triangle, polygon, oval, star etc. The features of the shape that have been considered in this work include boundary structure segmentation, chord feature and diagonal features. We have achieved maximum recognition accuracy of 96.7% using nearest neighbours classifier.

REFERENCES

[1] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes", Pattern Recognition, vol. 13, Issue 2, pp. 111-122, 1981.
[2] S. Ullman, "Visual Routines", cognition, 18, pp. 97-156, 1984.
[3] L. Itti, E. Niebur and C. Koch, "A model of saliency-based visual attention for rapid scene analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, Issue 11, pp. 1254-1259, 1998.
[4] S. Minut and S. Mahadevan, "A reinforcement learning model of selective visual attention", Proceedings of fifth international conference on Autonomous Agents, ACM, pp. 457-464, 2001.
[5] D. G. Lowe, "Distinctive image features from scale invariant key points", International Journal of Computer Vision, vol. 60, pp. 91-110, 2004.
[6] G. Mori, S. Belongie and J. Malik, "Efficient shape matching using shape contexts", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, Issue 11, pp. 1832-1837, 2005.
[7] E. Nadernejad, S. Sharifzadeh and H. Hassanpour, "Edge Detection Techniques: Evaluation and Comparisons", in Applied Mathematics Sciences, vol. 2, Issue 31, pp. 1507 – 1520, 2008.
[8] A. Toshev, B. Taskar and K. Daniilidis, "Object detection via boundary structure segmentation", IEEE Conference on Computer Vision and Pattern Recognition, pp. 950-957, 2010.
[9] S. Tripathi, K. Kumar, B. K. Singh and R. P. Singh, "Image segmentation: A Review", International Journal of Computer Science and Management Research, vol. 1, Issue 4, 2012.
[10] M. Kumar, M. K. Jindal and R. K. Sharma, "Classification of characters and grading writers in offline handwritten Gurumukhi script", International Conference on Image Information Processing, pp. 1-4, 2011.
[11] M. Kumar, R. K. Sharma and M. K. Jindal, "Segmentation of Lines and Words in Handwritten Gurmukhi Script Documents", International Conference on Intelligent Interactive Technologies and Multimedia, pp. 25-28, 2010.
[12] M. Kumar, M. K. Jindal and R. K. Sharma, "k-nearest neighbours based offline handwritten Gurmukhi character recognition", International Conference on Image Information Processing, pp. 1-4, 2011.
[13] Biederman, "Recognition-by-components: A theory of human image understanding", Psychological Review, vol. 94, Issue 2, pp. 115-147, 1987.

© 2017 Vidya Publications. **Authors are responsible for any plagiarism issues.**

[14] A. Elnagara, R. Alhajib, “Segmentation of connected handwritten numeral strings”, Pattern Recognition, vol. 36, Issue 3, pp. 625-634, 2003.
[15] M. Hanmandlu, J. Grover, VK. Madasu and S. Vasikarla, “Inputfuzzy for the recognition of handwritten Hindi numeral”, Proceedings of Information Technology (ITNG’07), pp. 208-213, 2007.
[16] S. V. Rajashekararadhva and S. V. Ranjan, “Zone based feature extraction algorithm for handwritten numeral recognition of Kannada Script”, Proceedings of IACC, pp. 525-528, 2009.
[17] S. Palmer, “Vision science: Photons to phenomenology”, 1999.
[18] M. Peterson and B. Gibson, “Must Figured-Ground Organization Precede Object Recognition? An Assumption in Peril”, Psychological Science, vol. 5, Issue 5, pp. 253-259, 1994.
[19] SX. Yu and J. Shi, “Object–specific figure-ground segregation”, Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 39-45, 2003.
[20] V. Ferrari, F. Jurie and C. Schmid, “From Images to Shape Models for Object Detection”, International Journal of Computer Vision, vol. 87, pp. 284-303, 2010.
[21] M. Leordeanu, M. Hebert, and R. Sukthankar, “Beyond local appearance: Category recognition from pairwise interactions of simple features”, Proceedings of Computer Vision and Pattern Recognition (CVPR), pp. 1-8, 2007.
[22] S. Belongie and J. Malik, “Shape matching and object recognition using shape contexts”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, Issue 4, pp. 509-522, 2002.

© *2017 Vidya Publications.* Authors are responsible for any plagiarism issues.