A Fast Interactive Geometric Shape Recognition Method

Marjuka Ferdousi Lazin, Md. Raihan ul Masood

Department of Computer Science and Engineering, Northern University Bangladesh
Holding no.13, Road no. 17, Banani, Dhaka-1213, Bangladesh

marjuka.ferdousi@gmail.com, bdraihan@yahoo.com

***Abstract*— This paper focuses on a simple, fast and highly accurate procedure for recognizing 2D geometric shapes such as rectangle, trapezium, acute triangle, etc. thereupon providing an interactive platform for the children. Proposed method for recognition is based on fundamental digital image processing and basic metrics of the shapes. On the basis of the pixels position of the shape's image on a predefined bounding-box and geometric logic, broadly 7 features have been addressed for recognizing 10 shapes. Defined novel features and use of decision tree not only simplify the process of recognition but also lower the response time. For experimental evaluation, two different types of dataset consisting a total of 4,725 shapes of 10 different 2D geometric shapes together are used and the overall accuracy of the recognition process for the datasets are 98.79% and 99.39% respectively.**

Keywords—Geometric shape, recognition, down sampling, feature extraction

I. INTRODUCTION

In recent years, with the development in the field of Artificial Intelligent such as CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), GAN (Generalized Adversarial Network), object recognition draws the attention of lots of researchers to enrich the field of Computer Vision. Geometric shape recognition plays a very important role in object recognition by identifying an object in an image or in a video sequence as lots of real-world objects have shapes like basic geometric shapes. For instance honey comb is like Hexagon, the cross section of vegetable okra is like Pentagon, etc. Some road sign using geometric shape such as hexagon is used for stop sign, triangle is used for yield sign, etc. Moreover, geometric shape identification helps children to develop their cognitive mind. Hence geometric shape recognition by a machine can be an interesting and useful domain of research.

The geometric shapes such as circle, triangle, rectangle, ellipse and etc. have some fixed definitions and formed by fixed orientation of straight lines and curves. Considering the geometric definition of each shape, this paper attempts to present a simple but fast and accurate method to classify them. The proposed method concentrates on fast computing with less features. Exploiting the basic geometric definition this work presents some novel features considering the position of each point of a shape's outline in a bounded box.

II. LITERATURE REVIEW

As geometric shape recognition significantly contributes in various fields including image analysis and computer vision; different methods are established along with various commonly used shape recognition algorithms. Considering the flaws of these generally used algorithms such as large amount of calculation or long processing time [1]; more methods are developed such as using edge pixel point eigenvalues [1], artificial neural network [2], [3], combination of fuzzy routines with back propagation algorithm for neural net [3], fuzzy logic [3], [4] and basic geometric properties [4], [5], [6], and also using Image Processing algorithms and techniques [5], [7], [8], [9], [10], [11]. The algorithm using edge pixel point eigenvalues has some benefits such as less time and higher precision but it also has some drawbacks like complication in handling figures with noise, multiple figure image and not working well with all geometric shapes [1]. The method using real-time neural network can identify shapes with color but only for a few properly presented shapes [2].Though method [3] combines neural net and fuzzy logic and has an overall success rate of 97.77%, only three shapes were classified. Also the methods using ANN requires large amount of data and more time for training. The algorithm based on basic geometric features and fuzzy logic has good success rate of 91% with some shapes while having lower recognition rate with others along with some misidentifications [4]. The process mentioned in [6] is similar to the work presented in this paper on the basis of subdivision of shape's image and uses geometric features with a recognition rate of 96.7 %, but too much computation is involved as the method has to perform floating point calculation at each sub-window.

III. PREPROCESSING

While working with input data or images every system has some preprocessing steps. Preprocessing facilitates the work with the target shape in a more manageable and smaller domain. Since main objective of this work is to identify the geometric shapes, preprocessing tasks only focuses on the outline rather than its area, or color. The preprocessing tasks presented in this paper and the interactive interface "Fig. 1" used to accomplish the preprocessing task as well as the whole recognition system are very similar to [12] with some extra addition. Java language is used for the implementation. The preprocessing encompasses the following tasks-

978-1-7281-5842-6/19/$31.00 ©2019 IEEE

A. Sample Input

The input can be taken either by loading images containing a single 2D geometric shape or by direct drawing using mouse, stylus etc.

B. Bounding the Shape

After getting the input shape in the specified area of the interface, the shape is enclosed by a bounding box. The bounding box is drawn using upper-left point and lower-right point of the shape.

C. Down Sampling

To reduce computational cost, the bounded shape is then down sized using down sampling technique. There are several methods for down sampling an image, *f(x, y)* in a spatial domain such as sub-pixel based down sampling [13], using genetic algorithm and discrete wavelet transform [14], etc. As in our case we need not to worry about color or aliasing, we choose a simple but fast down sampling technique that is nearest neighbor algorithm using Java ImageIO library. The dimension (160x140) of the image of 2D geometric shape is scaled down to smaller dimension (21x 21) as shown in the "Fig. 2".

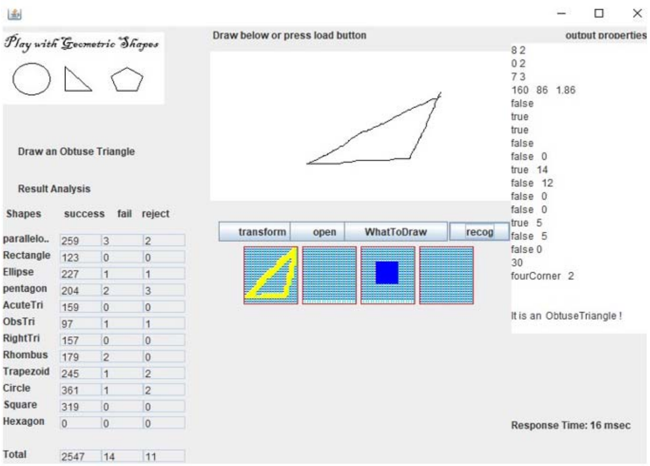


Fig. 1. Interface of the implementation

D. Rotation operation

Since the shape's features of interest (as described in sec IV) are largely affected by the angular orientation from the base line (the lower horizontal line of the bounded box), geometric rotation operation is done to compute features of interest for successful recognition in the preprocessing steps. Hence 2D point rotation is applied to the down sampled shape around the midpoint of the bounded box using the following equation:

$$newX = midX \times \cos\emptyset - midY \times \sin\emptyset \tag{1}$$

$$newY = midY \times \cos\emptyset - midX \times \sin\emptyset \tag{2}$$

Depending upon the orientation either vertical ($\theta_1 \leq \theta_2$) or horizontal ($\theta_1 > \theta_2$) alignment is done as shown in the "Fig. 3.a".

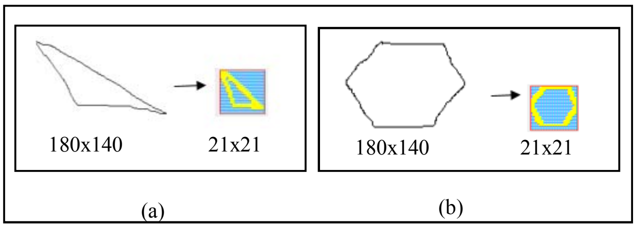


Fig. 2. Output of the down sampled shapes

The shape 'circle' due to 8-symmetry and all types of triangles for their one unique feature which is described in sec IV, are excluded from rotation operation. Other above mentioned eight shapes may need to be rotated to find the desired properties. A simple procedure is followed to find out whether the shape is needed to be rotated or not. Using "Fig. 3.a" and "Fig. 3.b" the procedure is as follows-

1. Scan upward from baseline up to 3 rows to find horizontal line.
2. If horizontal line exists, no rotation operation is performed.
3. Else rotation operation is performed after computing θ_1 and θ_2 from line CB and line CA respectively.

"Fig. 3.c" shows the output of the rotation task. For successful rotation, the 21x21 grid is increased to 30 x 30 as $\sqrt{21^2 + 21^2}$ = *29.70≈30.*

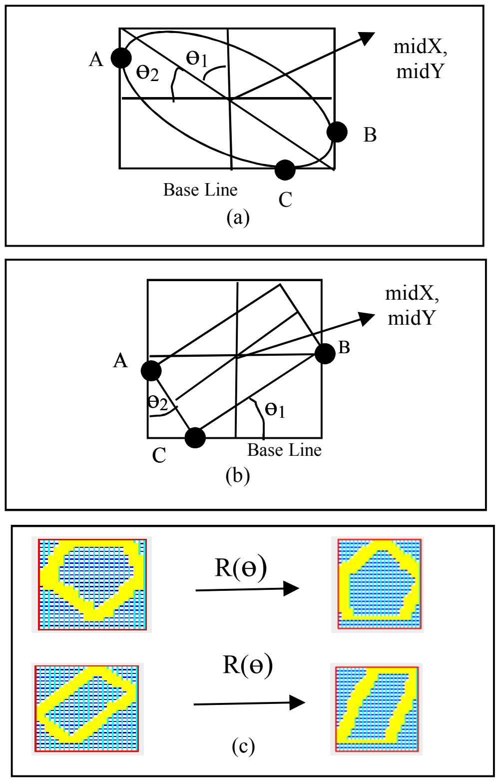


Fig. 3. Rotation

IV. Feature Extraction

For any kind of recognition processes identifying features is obvious. Effective features are always essential towards fast recognition processes. Since the input object is preprocessed as down sampled raster image, the features of that object have been extracted on the basis of bounded down sampled object. For succeeding a fast and efficient recognition of triangle, square, parallelogram, rhombus, trapezium, rectangle, circle, ellipse, hexagon and pentagon, we consider the presence of pixels in some certain area of the 21 by 21 grids “Fig. 4”. Based on those area of interest we redefine the above mentioned geometric shapes by the followings-

1. Left upper corner (LUC)
2. Left bottom corner (LBC)
3. Right upper corner (RUC)
4. Right bottom corner (RBC)
5. Upper line (UL)
6. Bottom line(BL)
7. Left line (LL)
8. Right line (RL)
9. Upper mid (UM)
10. Bottom mid(BM)
11. Left mid(LM)
12. Right mid(RM)
13. Corner Points (CP)
14. Middle Area (MA)
15. Four Corners (FC)
16. Ratio(R)

Brief and explanatory description of the above features are as follows

- LUC/RUC/LBC/RBC:

These features return true value if there exist 1 to 4 pixels in the four corner of the bounded box.

- UL/BL/LL/RL:

If first and last, either one or two rows and columns contain at least 11 continuous presence of pixels including any of the respective LUC/RUC/LBC/RBC with true value, these features return true.

- UM/BM/LM/RM:

These features are present in the test object if 7 plus pixels are present in the exact middle of the four sides of the bounded box and return true value.

- CP:

The feature CP can easily identify the difference between quadrilaterals, pentagon and hexagon. This feature keeps track of corner points of the test shape. It takes value from 1 to 4. It is also considered as overlapped on or close to the four corners of the rectangular box.

- MA:

This feature takes numeric value 0 to 17 (e.g. the highest possible value for a Right triangle is $[\sqrt{21^2+21^2} = 29.70 \approx 30]/2 = 15 + 2)$. This is the total number of pixels in the intersection area of four parallel lines “Fig. 4”. It is worth mentioning that no triangle can be drawn without having significant number of pixels in the intersection area and other geometric shapes should not have any pixels in this region. From geometric rules it is known that the area of a rectangle inscribed in a triangle is half of that triangle’s area and the area of the bounded box is twice as large as the drawn inscribed triangle, hence the dimension of the area of interest is defined as $\frac{L}{2}+1$ and $\frac{W}{2}+1$ where L and W are the length and width of the bounded box respectively.

- FC:

FC takes numeric value 0 to 4 depending upon the Boolean value of LUC/RUC/LBC/RBC. It separates rectangle and square from others. In standard situation

$$FC = \begin{cases} 0, \text{if shape is either circle or ellipse} \\ 4, \text{if shape is either rectangle or square} \end{cases} \quad (3)$$

- Ratio(R):

Length to width ratio of the bounded box. This is the only feature in which the value is computed before down sampling.

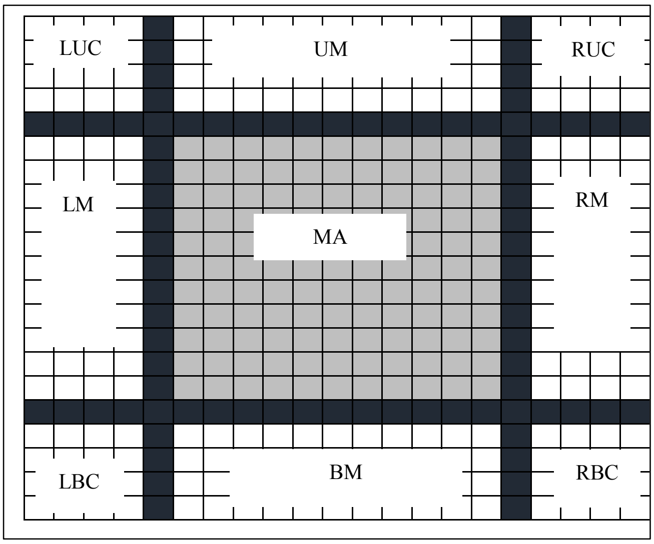


Fig. 4. Areas of interest for the features

V. RECOGNITION

The recognition procedure is done by the logical reasoning based on the features described in sec IV. Depending upon the features a decision tree is used to identify each geometric shape in each step. Though in some cases a single feature is not enough to recognize one particular shape, some features together can set apart a group of shapes from other groups. According to the features (as described in sec IV) the geometric shapes are grouped as follows-

- Gr1- acute, obtuse and right triangle.
- Gr2- ellipse, circle, rectangle and square
- Gr3- parallelogram, rhombus
- Gr4- hexagon
- Gr5- pentagon
- Gr6- trapezoid

The above groups that contain single shape are classified by multiple features, whereas other groups can be identified by a single feature.

According to geometric definition and considering the topology in the bounded down sampled box a chart is prepared with the Boolean values of all the features (sec IV) corresponding to each of the 12 (3 variation of triangle) shapes. The chart presents a concise picture of how the features are related to all the shapes in standard situation "Table I". The values of the chart are tested from 7,600 data samples from 100 individuals. Part of the test data is presented in the "Table. II".

For simplification of the proposed procedure of the recognition system, some Boolean variables (assigning large Boolean expressions e.g. triAll = $[(MU \wedge (\neg MB)) \vee (MB \wedge (\neg MU)) \vee (ML \wedge (\neg MR)) \vee (MR \wedge (\neg ML))]$) are as follows-

- squRect
- cirEll
- triAll
- triRight
- triObs
- triAct
- para
- penta
- hexa
- trapi
- rhom

TABLE I

shapes	Features															
	LUC		*RUC*		*UL*		*RL*		*RM*		*LM*		*CP*	*MA*	*FC*	*R*
	T	F	T	F	T	F	T	F	T	F	T	F				
Circle		√		√		√		√	√		√		0	0	0	1.0
Ellipse		√		√		√		√	√		√		0	0	0	1.3
Pentagon		√		√		√		√		√		√	0	0	0	>1
		√		√		√		√		√		√	0	0	0	>1
		√		√		√		√		√	√		0	0	0	>1
		√		√		√		√	√			√	0	0	0	>1
Square	√		√		√		√		√		√		4	0	4	1.0
Parallelogram		√	√		√			√		√		√	2	0	2	>1.3
	√			√	√			√		√		√	2	0	2	>1.3
Rectangle	√		√		√		√		√		√		4	0	4	1.3
Rhombus		√	√		√			√		√		√	2	0	2	1.0
	√			√	√			√		√		√	2	0	2	1.0
Trapezoid		√		√	√			√		√		√	2	0	2	>1
	√		√		√			√		√		√	2	0	2	>1
	√			√		√		√		√	√		3	0	3	>1
		√	√			√	√		√			√	3	0	3	>1
Hexagon		√		√		√		√	√		√		0	0	0	1.0
		√		√		√		√		√		√	0	0	0	1.0
Right Triangle													3	13	3	>0.0
	√			√		√		√		√	√		3	13	3	>0.0
		√	√			√	√		√			√	3	13	3	>0.0
	√		√		√		√		√			√	3	13	3	>0.0
Obtuse Tri.	√		√		√			√		√	√		3	15	2	>0.0
	√			√		√		√		√		√	3	15	2	>0.0
Acute Tri.		√	√			√		√		√		√	3	9	2	>0.0
		√		√		√		√		√		√	3	9	2	>0.0
	√		√		√			√		√		√	3	9	2	>0.0
		√	√			√	√		√			√	3	9	2	>0.0

The proposed recognition algorithm is as follows-

1. Check the value of 'MA', if it is greater than 10, the test shape must be in GR1 that is a triangle.
   a. If 'triRight' is true, the test shape is a Right Triangle.
   b. If 'triObs' is true, the test shape is an Obtuse Triangle.
   c. If both 'triRight' and triObs is false, the test shape is an Acute Triangle.

2. Check the value of 'FC', if it is equal to 5, the test shape must be in GR2 that is either an Ellipse or a Circle or a Square or a Rectangle.
   a. If 'CP' is equal to 0,
      i. If 'R' is equal to 1, the test shape is a Circle.
      ii. If 'R' is greater than 1, the test shape is an Ellipse.

TABLE II

Shapes	Features											
	LUC		*RUC*		*UL*		*RL*		*RM*		*LM*	
	T	*F*	*T*	*F*	*T*	*F*	*T*	*F*	*T*	*F*	*T*	*F*
Circle	6	271	4	273	7	270	9	268	277	0	277	0
Ellipse	19	981	16	984	31	969	31	969	1000	0	1000	0
Parallelogram	15	985	983	17	983	17	12	988	12	988	14	986
Rectangle	326	2	324	4	327	1	323	5	328	0	326	2
Rhombus	11	859	860	10	862	8	7	863	11	859	7	863
Hexagon	5	795	7	793	791	9	796	4	405	395	391	409
Obtuse Tri.	510	490	512	488	5	995	7	993	21	979	15	985
Acute Tri.	511	489	517	483	240	760	248	752	242	758	259	741

   b. If CP is equal to 4,
      i. If 'R' is equal to 1, the test shape is a Square.
      ii. If 'R' is greater than 1, the test shape is a Rectangle.
3. If 'para' is true, the test shape is a Parallelogram.
4. If 'rhom' is true, the test shape is a Rhombus
5. If 'penta' is true, the test shape is a Pentagon.
6. If 'hexa' is true, the test shape is a Hexagon.
7. If 'trapi' is true, the test shape is a Trapezium.
8. Otherwise test shape is rejected.

VI. TESTING

To test the performance of the recognition scheme, a dataset containing 4,340 hand drawn shapes of 10 different kinds "Table I" each at a time from 100 people (different from sec V) is collected through the interface "Fig. 1". The interface provides a 'result analysis' panel so that the overall testing can be recorded. To make it more interactive the interface is equipped with a button 'whatToDraw'. By clicking this button users can get a random suggestion for which shape to be drawn next. In this way users can get a round of session of drawing, loading and testing the proposed recognition system. Users can also load or draw shape without suggestion and get result from the 'output properties' section.

Performance is also tested using 385 "Table I" images (180 x 140) containing single shape of each 10 types. Sample images are created using MS Paint and the images are augmented using python. Some sample of these kind of test samples are shown in "Fig. 5". The system is tested on a machine with Pentium core i3-7100 3.9 GHz processor and 4GB RAM.

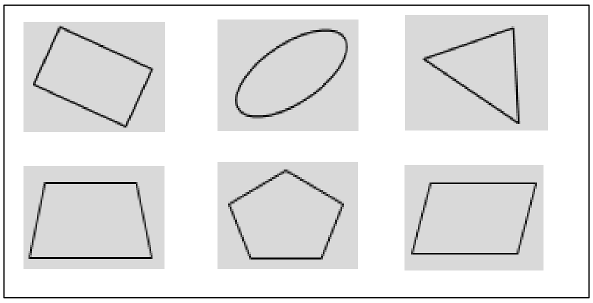


Fig. 5. Image sample containing single shape (showing scaled down version from the original)

TABLE III

Shapes	Data set for testing features value "Table II"	Data set for recognition	
		Hand drawn	*Image*
Circle	277	250	25
Ellipse	1000	480	50
Pentagon	1000	547	50
Hexagon	800	500	50
Triangle	1000	675	50
Square	325	200	20
Rectangle	328	220	15
Parallelogram	1000	458	50
Rhombus	870	532	25
Trapezium	1000	478	50
Total	7600	4340	385

VII. RESULT AND DISCUSSION

The procedure, when tested on two different types of dataset as described in section VI and "Table III" ascertained to be accurate with the overall accuracy of 98.79% and 99.39% respectively. The individual successful shape's recognition rates are shown in the figure "Fig. 6" and "Fig. 7" respectively.

The highest recognition rate (100%) of triangle and rectangle is the evidence that feature MA for triangle and feature CP for rectangle makes them distinct from other shapes. Recognition rate of square and rectangle should be 100%, but due to the value of feature R=1.1, recognition rate has been dropped slightly as it is difficult to draw maintaining the length width ratio = 1.0 without scale. The shape Circle suffers for the same reason. The lowest recognition accuracy is observed for the shape Pentagon due to one very shorter edges compare to other four "Fig. 8.b" and Rhombus because of rotation operation turns it into a Square.

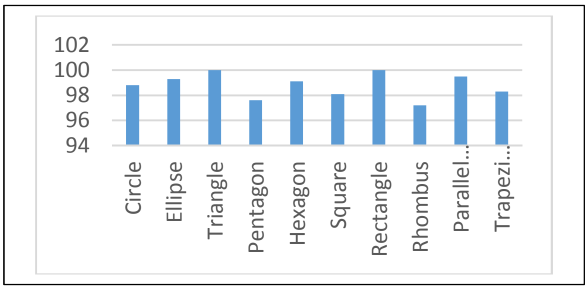


Fig. 6. Success rate of recognition for each shape (Drawn)

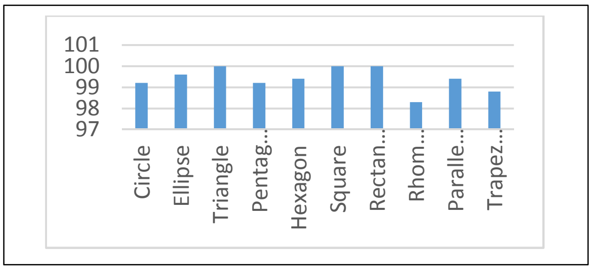


Fig. 7. Success rate of recognition for each shape (from images)

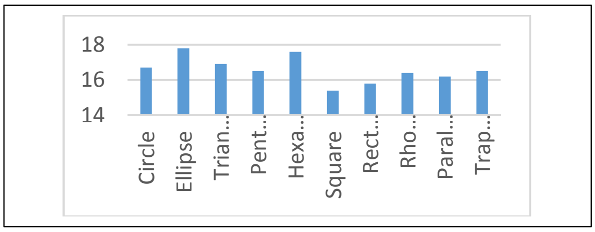


Fig. 8. Recognition response time in msec

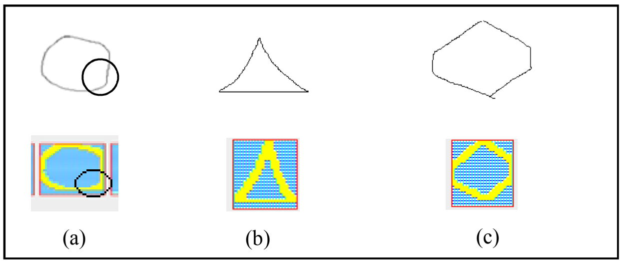
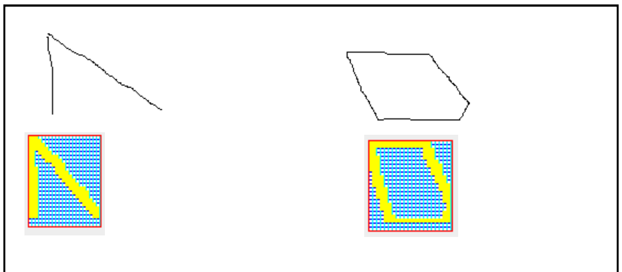


Fig. 9. Strength of some features (a) 'FC' (b) 'MA' (c) 'CP'

The overall accuracy increases for the second dataset "Fig. 7" as a result of the increased recognition rate of Circle and Square. The average response time is recorded as 16.58 msec and the highest and lowest time is observed as 18 msec and 16 msec respectively.



(a) Triangle without base

(b) One of the variation of monohedral pentagon

Fig. 10. Cases of failure

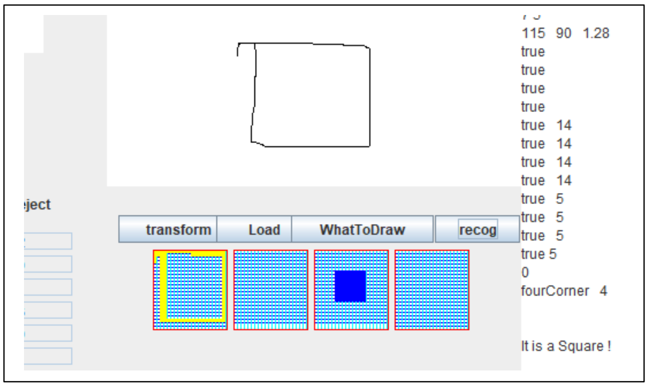


Fig. 11. Successfully taking care of unwanted segment

TABLE IV

Different Methods	Total number of data set	Total number of shapes	Overall Efficiency%	Average Response Time(msec)
[3]	120	4	97.77	Not mentioned
[4]	934	8	91.0	Do
[6]	4556	8	96.7	Do
[7]	180	4	99.0	Do
[9]	70	3	85.0	Do
Proposed method	4340 + 385	10	98.79 and 99.39	16.58

"Fig. 9" and "Fig. 10" represents some strength and weakness of the proposed recognition system. It is also evident that the preprocessing steps successfully handles the noise like unwanted extended section "Fig. 11".The system fails to recognize self-intersection pentagon i.e. pentagram.

Though there is no benchmark of data sample for previous works in this context, "Table IV" represents the comparison between some methods of geometric shape recognition with different data sets.

VIII. CONCLUSION

The proposed method presented in this paper has successfully achieved its goal by recognizing 10 basic geometric shapes through an interactive process using two types of dataset. Observing the pixel positions of the down sampled image inside a bounded-box for each shape and using geometric logic, this work managed to redefine the shapes in terms of some Boolean values which are considered as basic features of the shape geometry. High success rate of the experimental result justifies the redefinition "Table I". The work focused on providing a means of fast and highly accurate geometric shape recognition system using handheld devices.

Taking care of some rejection and recognition failure "Fig. 10" and adding a sound effect in the interface for recognition message, this work can be very handy for children to learn geometric shapes using an electronic drawing pad or smart phone.

REFERENCES

[1] W. Chen, L. Yao, J. Zhou1, and H. Dong, "A Fast Geometry Figure Recognition Algorithm Based on Edge Pixel Point Eigenvalues," Proceedings of the Third International Symposium on Computer Science and Computational Technology (ISCSCT '10) Jiaozuo, P. R. China, 14-15, pp. 297-300, August 2010.

[2] E. Castillejos-Villatoro, G. Nangusé-Vázquez, A. Medina-Santiago, and J. A. Velazquez-Martinez, "Identification of Geometric Shapes with Real-time Neural Networks," Int. J. Advanced Networking and Applications, vol. 8, no. 1, pp. 2973-2978, 2016 .

[3] I. Z. Mihu, A. Gellert, and H. V. Caprita, "Improved Methods of Geometric Shape Recognition using Fuzzy and Neural Techniques," Periodica Politechnica, Transactions on Automatic Control and Computer Science, vol.49 (63), 2004.

[4] J. A. Jorge and M. J. Fonseca, "A Simple Approach to Recognise Geometric Shapes Interactively," in Proc. GREC '99 Selected Papers from the Third International Workshop on Graphics Recognition, Recent Advances, pp. 266-276, September 1999.

[5] S. Patel, P. Trivedi, V. Gandhi, and G. I. Prajapati, "2D Basic Shape Detection Using Region Properties," International Journal of Engineering Research & Technology (IJERT), vol. 2, no. 5, pp. 1147-1153, May 2013.

[6] S. Gupta and Y. J. Singh, "Shape Detection using Geometrical Features," An International Journal of Engineering Sciences, pp. 260-270, Issue December 2017.

[7] S. Rege1, R. Memane, M. Phatak, and P. Agarwal, "2d geometric shape and color recognition using digital image processing," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 2, no. 6, pp. 2479-2487, June 2013.

[8] S. Garg1 and G. S. Sekhon, "Shape Recognition based on Features matching using Morphological Operations," International Journal of Modern Engineering Research (IJMER), vol.2, no. 4, pp. 2290-2292, July-August 2012.

[9] M. F. Zakaria, H. S. Choon, and S. A. Suandi, "Object Shape Recognition in Image for Machine Vision Application," International Journal of Computer Theory and Engineering, vol. 4, no. 1, pp. 76-80, February 2012.

[10] N. El Abbadi and L. Al Saadi, "Automatic Detection and Recognize Different Shapes in an Image," IJCSI International Journal of Computer Science Issues, vol. 10, Issue 6, no 1, pp. 162-166, November 2013.

[11] V. Kumar, S. Pandey, A. Pal, and S. Sharma, "Edge Detection Based Shape Identification," Presented in the National Conference on Emerging Trends in Engineering Science & Technology.

[12] M. R. Masood, "A Fast Recognition Scheme for Off-line Bangla Numerals," 16th Int'l Conf. Computer and Information Technology, 2014.

[13] L. Fang and O. C. Au, "Subpixel-Based Image Down-Sampling With Min-Max Directional Error for Stripe Display," IEEE journal of selected topics in signal processing, vol. 5, no. 2, pp. 240–251, April 2011.

[14] N. Jagadeesan and Dr. R. M. S. Parvathi, "An Efficient Image Downsampling Technique Using Genetic Algorithm And Discrete Wavelet Transform," Journal of Theoretical and Applied Information Technology 31st March 2014. vol. 61, no.3, pp. 506-514, 31st March 2014.