Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 8, August 2016, pg.131 – 137

Slops and Distances for Regular Shape Image Recognition

Dr. Saad Abdual Azize
Associate Professor, Computer Science Dept.
Al_Mamon University College
Baghdad, Iraq
saadabdualazize@yahoo.com

Bilal Sadeq Obaid Obaid
Head of IT and Computer Dept.
Pro-Technique Ltd
Amman, Jordan
www.pro-technique.com, www.unihouse.com.jo
belal_sadeq@yahoo.com

Abstract: The term "geometric shape" refers to any shape that remains virtually unchanged if it is moved around, flipped or reflected off of a surface. When manipulated, a geometric shape does not create a new shape; it remains intact. Such shapes include the circle, square, triangle, rectangle, hexagon and octagon. This research highlights the way common geometric shapes are recognized; this is done by scanning the image, determining eight points, calculating the distance between them, and calculating the slop; finally, the desired shape is determined.
Keywords: Shape recognition, Slop, Euclidean distance, regular shape.

Introduction

Geometrical shapes have been found very commonly in the nature and have wide range of practical applications, they are mainly used in environment detection with robotic and artificial intelligence machines, also they are essential in geometrical images processing, monitoring and controlling systems and so on. For that, discriminate and detect the geometrical shapes is a sensitive and brilliant mission. Those shapes consider the geometrical information which remains when location, scale, orientation and reflection are removed from the description of a geometric object. That is, the result of moving a shape around, enlarging it, rotating it, or reflecting it in a mirror is the same shape as the original, and not a distinct shape.

Objects that have the same shape as each other are said to be similar. If they also have the same scale as each other, they are said to be congruent.

Many two-dimensional geometric shapes can be defined by a set of points or vertices and lines connecting the points in a closed chain, as well as the resulting interior points. Such shapes are called polygons and include triangles, squares, and pentagons. Other shapes may be bounded by curves such as the circle or the ellipse.

The four properties that govern all geometrical shapes are point, line, plane and solid. Geometry is divided into two distinct categories. Plane geometry studies two-dimensional shapes, or items on a flat surface, while solid geometry studies three-dimensional objects.
Two-dimensional shapes include polygons and curved objects. Regular polygons have straight sides which are the same length and angles with the same measurement, such as triangle, circle, square, rectangle, hexagon and octagon.

Rectangle is any quadrilateral with four right angles. It can also be defined as an equiangular quadrilateral, since equiangular means that all of its angles are equal (360°/4 = 90°). It can also be defined as a parallelogram containing a right angle. A rectangle with four sides of equal length is a square. Opposite sides are parallel and of equal length.
Square is a regular quadrilateral, which means that it has four equal sides and four equal angles (90-degree angles, or right angles).[1] It can also be defined as a rectangle in which two adjacent sides have equal length.
Triangle; There are thousands of different constructions that find a special point associated with (and often inside) a triangle, which gives it a unique properties. Often, triangles are constructed by finding three lines associated in a symmetrical way with the three sides (or vertices) and then proving that the three lines meet in a single point. An important tool for proving the existence of this is Ceva's theorem, which gives a criterion for determining when three such lines are concurrent.
Parallelogram; Recall that parallelograms are special a type of quadrilateral whose opposite sides do not intersect. Parallelograms come in the form of rectangles, rhombuses, and squares. When trying to determine the area of a parallelogram, it will be necessary to identify two main components: the **base** and the **height** of the parallelogram. The base of a parallelogram can be on any side of the figure. The height of a parallelogram is the perpendicular distance between any two parallel bases. Let's look at these different parts in the figure below. A parallelogram has opposite sides parallel and equal in length. Opposite angles are also equal.

The rhombus has the following properties:

- All the properties of a parallelogram apply (the ones that matter here are parallel sides, opposite angles are congruent, and consecutive angles are supplementary).
- All sides are congruent by definition.
- The diagonals bisect the angles.
- The diagonals are perpendicular bisectors of each other.

Regular pentagon; A five-sided shape is called a pentagon. A six-sided shape is a hexagon, a seven-sided shape a heptagon, and an eight-sided shape is an octagon.
While there are different names for different types of polygons, the number of sides is more important than the name of the shape. A **regular polygon** has equal length sides with equal angles between each side.

Line Edge detectors using slop

A line is increasing if it goes up from left to right. The slope is positive,. { m>0}
A line is decreasing if it goes down from left to right. The slope is negative. { m<0}
If a line is horizontal then the slope is zero. This is a constant function.
If a line is vertical then the slope is undefined (Figure 1)

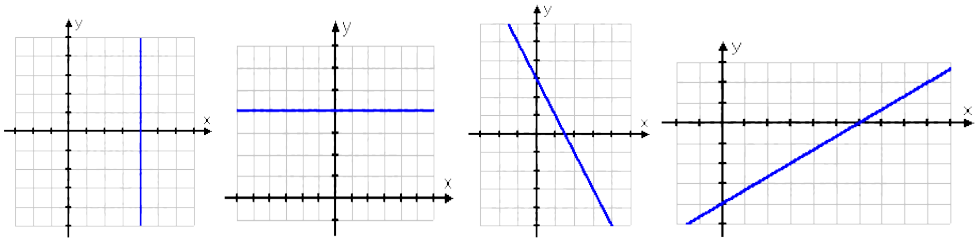


Figure 1: four types of lines and slops

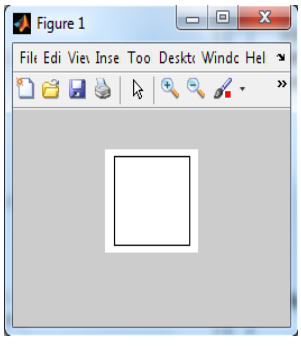
The Proposed Algorithm

The proposed algorithm of recognizing geometrical shapes goes on many steps as follows:

1- Read image.
2- Scan to find 8 points which define the corner of the shape.
3- Sort the 8 point in ascending order.
4- Remove the equal points and find the number of reminding point (N).
5- If N = 4, there's four corner points.
6- Find the Euclidean distance between the four points.
7- If the four distances are equal, then the shape is "square"
   Otherwise, the shape is rectangle.
8- If N = 8, find Euclidean distance between the 8 points.
9- Remove the distances less than 5.
10- If the reminders are three distances then the shape is triangular.
11- If the reminders are four distances then find the slops between points.
   a- If the two slops are positive then the shape is rhombus.
   b- If the two slops are zero then the shape is parallelogram.
12- If the reminders are five distances then find the slops between points.
   a- If one of the slops is zero, two slops are negative and two are positive then the shape is regular pentagon.

Implementation

We apply the algorithm on different shapes as follows:



a: a square image

ans =

8.5000	5.5000
76.5000	5.5000
76.5000	5.5000
76.5000	73.5000
76.5000	73.5000
8.5000	73.5000
8.5000	73.5000
8.5000	5.5000

b: Points found

h =

9	6
77	6
77	6
77	74
77	74
9	74
9	74
9	6

c: Rounded values

r =

68	1
68	3
68	5
68	7

d:Finding distance

point	dis	slop	shape
1,2	68	0	—
2,4	68	-inf	|
3,4	68	0	—
3,1	68	inf	|

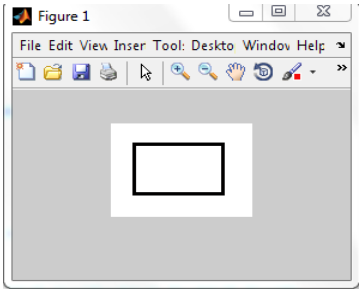
e: Finding distances

Shape either rectangle or rhombus or square
D(1,2) = 68, D(2,4) = 68, D(4,3) = 68, D(3,1) = 68.
p = 1, sp = 0, p = 3, sp = -Inf, p = 5, sp = 0, p = 7, sp = Inf.

Shape is: Square

f: Shape recognition

Figure 2: Applying algorithm on squared shape



ans =

18.5000	17.5000
95.5000	17.5000
95.5000	17.5000
95.5000	64.5000
95.5000	64.5000
18.5000	64.5000
18.5000	64.5000
18.5000	17.5000

h =

19	18
96	18
96	18
96	65
96	65
19	65
19	65
19	18

r =

77	1
47	3
77	5
47	7

point	dis	slop	shape
1,2	47	0	—
2,4	77	-inf	|
3,4	47	0	—
3,1	77	inf	|

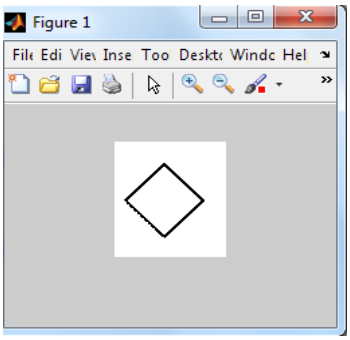
Shape either rectangle or rhombus or square

D(1,2) = 47, D(2,4) = 77, D(4,3) = 47, D(3,1) = 77

p =1, sp = 0, p = 3, sp = -Inf, p = 5, sp = 0, p = 7, sp = Inf

Shape is: Rectangle

Figure 3: Applying algorithm on Rectangular shape



ans =

36.5000	17.5000
38.5000	17.5000
67.5000	46.5000
67.5000	48.5000
38.5000	77.5000
37.5000	77.5000
7.5000	47.5000
7.5000	46.5000

h =

37	18
39	18
68	47
68	49
39	78
38	78
8	48
8	47

r =

41.0122	2.0000
41.0122	4.0000
41.0122	6.0000
41.0122	8.0000

point	Dis	slop	shape
2,3	41.0122	1	
4,5	41.0122	-1	
6,7	41.0122	1	
8,1	41.0122	-1	

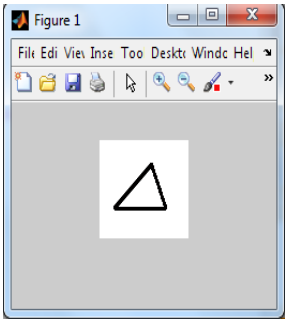
D(1,2) = 2, D(2,3) = 41.0122, D(3,4) = 2, D(4,5) = 41.0122,

D(5,6) = 1, D(6,7) =41.0122, D(7,8) = 1, D(8,1) = 41.0122

p =2, sp = 1, p = 4, sp = -1, p = 6, sp = 1, p = 8, sp = -1

Shape is: rhombus

Figure 4: Applying algorithm on rhombus shape



ans =

47.5000	18.5000
50.5000	18.5000
63.5000	51.5000
63.5000	55.5000
62.5000	56.5000
14.5000	56.5000
13.5000	55.5000
13.5000	52.5000

h =

48	19
51	19
64	52
64	56
63	57
15	57
14	56
14	53

r =

35.4683	2.0000
48.0000	5.0000
48.0833	8.0000

point	Dis	slop	shape
2,3	35	2.5	
5,6	48	0	
8,1	48	-1	

D(1,2) = 3, D(2,3) = 35.4683, D(3,4) = 4, D(4,5) = 1.4142, D(5,6) = 48,

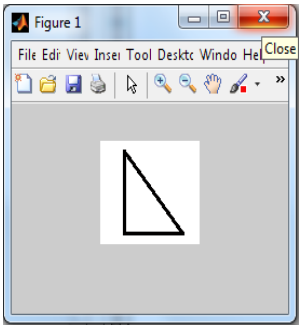
D(6,7) = 1.4142, D(7,8) = 3, D(8,1) = 48.0833, D(2,3) = 35.4683,

D(5,6) = 48, D(8,1) = 48.0833

p = 2, sp = 2.5385, p = 5, sp = 0, p = 8, sp = -1

Shape is : triangle

Figure 5: Applying algorithm on triangular shape



ans =

21.5000	7.5000
24.5000	7.5000
78.5000	73.5000
78.5000	76.5000
78.5000	76.5000
21.5000	76.5000
21.5000	76.5000
21.5000	7.5000

h =

22	8
25	8
79	74
79	77
79	77
22	77
22	77
22	8

r =

85.2760	2.0000
57.0000	5.0000
69.0000	7.0000

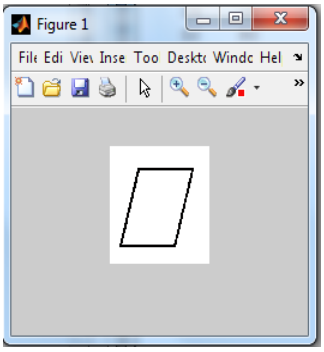
point	Dis	slop	shape
2,3	85.276	-2	
5,6	57	0	
7,8	69	inf	

D(1,2) = 3, D(2,3) = 85.2760, D(3,4) = 3, D(4,5) = 0, D(5,6) = 57, D(6,7) = 0, D(7,8) = 69, D(8,1) = 0, D(2,3) = 85.2760, D(5,6) = 57, D(7,8) = 69

p = 2 sp = 1.2222 p = 5 sp = 0 p= 7 sp = Inf

Shape is : triangular

Figure 6: Applying algorithm on triangular shape



ans=

23.5000	17.5000
70.5000	17.5000
70.5000	17.5000
70.5000	20.5000
55.5000	79.5000
8.5000	79.5000
8.5000	79.5000
8.5000	76.5000

h =

24	18
71	18
71	18
71	21
56	80
9	80
9	80
9	77

r =

47.0000	1.0000
60.8769	4.0000
47.0000	5.0000
60.8769	8.0000

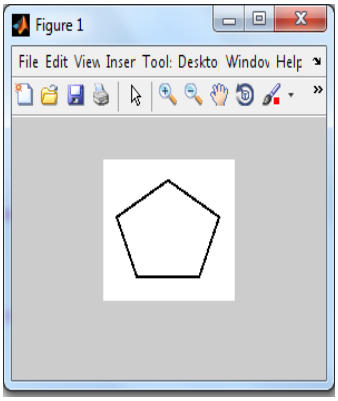
point	Dis	slop	shape
2,3	47	0	
4,5	60.8769	-3.9	
5,6	47	0	
8,1	60.8769	-3.9	

D(1,2) = 47, D(2,3) = 0, D(3,4) = 3, D(4,5) = 60.8769, D(5,6) = 47, D(6,7) = 0, D(7,8) = 3, D(8,1) = 60.8769

p =1, sp = 0, p = 4, sp = -3.9333, p = 5, sp = 0, p = 8, sp = -3.9333

Shape is: parallelogram

Figure 7: Applying algorithm on parallelogram shape



ans =

57.5000	14.5000
60.5000	14.5000
105.5000	40.5000
105.5000	42.5000
87.5000	84.5000
29.5000	84.5000
11.5000	42.5000
11.5000	40.5000

h1 =

12	41
12	43
30	85
58	15
61	15
88	85
106	41
106	43

r =

51.9711	2.0000
45.6946	4.0000
58.0000	5.0000
45.6946	6.0000
52.8394	8.0000

point	Dis	Slop	shape
2,3	51.9	0.5778	
4,5	45.6	-2.333	
5,6	58	0	
6,7	45.6	2.333	
8,1	52.8	-0.5778	

D(1,2) = 3, D(2,3) = 51.9711, D(3,4) = 2, D(4,5) = 45.6946, D(5,6) =58, D(6,7) = 45.6946, D(7,8) = 2, D(8,1) = 52.8394

p = 2, sp = 0.5778, p = 4, sp = -2.3333, p = 5, sp = 0, p = 6, sp = 2.3333, p = 8, sp = -0.5778.

Shape is: regular pentagon

Figure 8: Applying algorithm on regular pentagon shape

Conclusion

Geometrical shapes had been proved to be essential and important in many applications specially with the expansion of computer aided machines which need shape recognition like robotics, intelligent and controlling systems. In this paper we introduced a new technique for recognizing different shapes using line detection, depending on finding distances and slops between some points found on those shapes. The proposed method proved to be practical and easy to implement with great fidelity by applying the algorithm on plenty images.

References

1- Hearn Baker, "computer graphics with open GL", person prentice Hall, 2004.
2- A. Collet and S. Srinivasa, "Efficient multi-view object recognition and full pose estimation", Robotics and Automation, IEEE International Conference, 2010.
3- D. Lowe, "Distinctive image features from scale-invariant key points", International Journal of Computer Vision, vol. 60, no. 2, 2004.
4- Krish Karthik, Snyder Wesley, " A Shape Recognition Algorithm Robust to Occlusion: Analysis and Performance Comparison" , 2007.
5- M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition", Computer Vision, European Conference, 2000.
6- S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features", Computer Vision and Pattern Recognition, IEEE Conference, 2010
7- http://www.skillsyouneed.com/num/curved-shapes.html#ixzz3W9xq4C57
8- http://www.skillsyouneed.com/num/polygons.html#ixzz3W9xDI8qM

