Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I
IMECS 2008, 19-21 March, 2008, Hong Kong

A Novel Approach for Area Computation of Convex Shapes

Neeta Nain, Vijay Laxmi, Bhavitavya Bhadviya *and Nemi Chand Singh †

Abstract—**Shape analysis is widely applied in image registration, segmentation, classification and various other computer vision applications like pattern recognition, medical image diagnostics, integral geometry etc. Area measure plays a very important role in image classification based on size of the image. The area of an object is a convenient measure of the objects' overall size and is a widely used simple scalar descriptor. This paper describes a new scale invariant area computation algorithm for convex shapes with constant time complexity which can also be used for computation of various scalar shape descriptors.**

Keywords: Shape, Area, Signature, Triangulation

1 Introduction

Describing an object shape as numeric shape descriptors is more appropriate than a non numeric, graphic representation of the space-domain technique, because the numeric descriptors can be analyzed easily and efficiently. It is thus very important that the mathematical model contains very precise, concise and robust shape descriptors which are independent of image distortions and geometric manipulations like translation, scaling and rotation. The area of an object is a convenient measure of the objects overall size. Dependent only on the boundary of the object, a measurement of the area disregards gray level variations inside. Area measurements are computed during the extraction of an object from a segmented image. Area measure plays a very important role in image classification based on size of the image. Area measure can further be used for computation of a lot of scalar descriptors like compactness, elongatedness, eccentricity etc.

We propose an area computation algorithm for convex shapes based on method of triangulation in section 2. Section 3 illustrates signature computation using the new method and section 4 summarizes the proposed technique.

*Department of Computer Engineering, Malaviya National Institute of Technology Jaipur, Jaipur, India - 302017. Tel: +91 141 2529140 Email: neetanain@yahoo.com

†Rajasthan Swayat Sashan Mahavidyalaya, Tonk Road, Jaipur, India - 302015.

2 Area of Convex Shapes by Triangulation method

Before we can specify an algorithm for measuring the area of an object, we must establish a definition regarding the boundary of the object. The question that must be resolved is, Are the boundary pixels completely or only partially contained in the object? The simplest uncalibrated area measurement is just a count of the number of pixels inside(and including) the boundary. This though is a very simple algorithm but it has linear complexity.
There is a computationally simple way to compute both the area and signature [1, 2, 3, 4, 5, 6, 7, 8] of a polygon in one traversal of the boundary of the polygon. We propose a new method for area and signature computation of convex object shapes. We iteratively subdivide the boundary into two halves and fit triangles with staring point, midpoint and end point of the boundary as coordinates in each half. Compute the area of the triangle. The first and last point of each half forms the base and the orthogonal distance of the midpoint with the base is taken as the altitude. Exclude each triangle from the boundary and repeat the process until no more triangles can be fitted in the boundary. The terminating condition is three pixels on the boundary making an equilateral triangle, or boundary list as a straight line. The detailed proposed algorithm is enlisted in Algorithm 1:

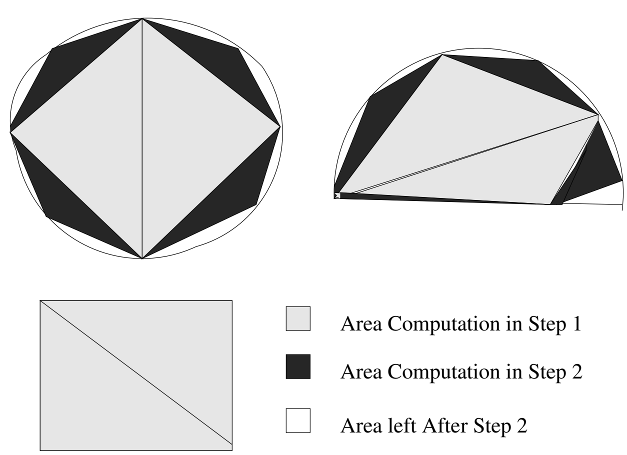

Figure 1: Area Computation by Triangulation Method for Convex Shapes.

For example, Figure 1 shows the result of the first few

ISBN: 978-988-98671-8-8

Algorithm 1 RecursiveArea(S_i)

Require: S_i: List of boundary points of shape S, for $i = 1, 2, \cdots, N$
Ensure: : Area of Convex Shape S
Initialization
$Area = 0$;
$Nsteps = 0$; {No of Steps}
procedure RecursiveArea(S_i)
Divide the list S_i at $i = \frac{N}{2}$ into two lists L_1, L_2
On each list (L_1, L_2)
Construct a Triangle $\Delta T = (T_1, T_2)$
Coordinates for $T_1 = (S_1, S_{\frac{N}{4}}, S_{\frac{N}{2}})$
Coordintes for $T_2 = (S_{\frac{N}{2}}, S_{\frac{3N}{4}}, S_N)$
$Area = Area + Area(T_1) + Area(T_2)$;
$Nsteps = Nsteps + 1$;
if Altitude($T_1 > 1$) **then**
return RecursiveArea(L_1)
end if
if Altitude($T_2 > 1$) **then**
return RecursiveArea(L_2)
end if
end procedure

steps of the algorithm when run on some general convex shapes. The result of area computation by our algorithm on 1200 x 1200 size images is shown in Figure 2. From Fig. 2, we can conclude that for convex shapes like square, the area is computed in constant time, that is, in two steps, for rectangle in nearly 11 steps for circle in 20 - 25 steps. Further we get another metric, after the second step 100% of the area of square is computed, while 85% of area of rectangle is computed and 75% of area of the circle is computed. This could be used as a filter for images to be classified on the basis of their sizes. Complexity of our method for area computation is constant, which is significantly smaller then the traditional pixel count's method which has linear time complexity.

Tests are done on various images. The best case is computation of area of a square which is calculated in two steps irrespective of its size. Area of rectangle is computed in 5 to 10 steps, the worst case is area of a circle, which is computed in 20 to 25 steps. For general 1200 x 1200 figures it varies between 2 to 25.

3 Signatures

When an object border is analyzed, parametrization techniques are used to reduce the dimensionality of images from 2D to 1D. Signatures reduce the boundary representation to a 1D function, which is presumably easier to describe. There are a number of ways to generate signatures. One of the simplest signatures, could be, the plot of the distance from the centroid to the boundary as a function r(θ), where θ is the regular sampling angle as depicted in Figure 3. Similarly complex coordinates, curvature and cumulative angular function can be used as signatures.

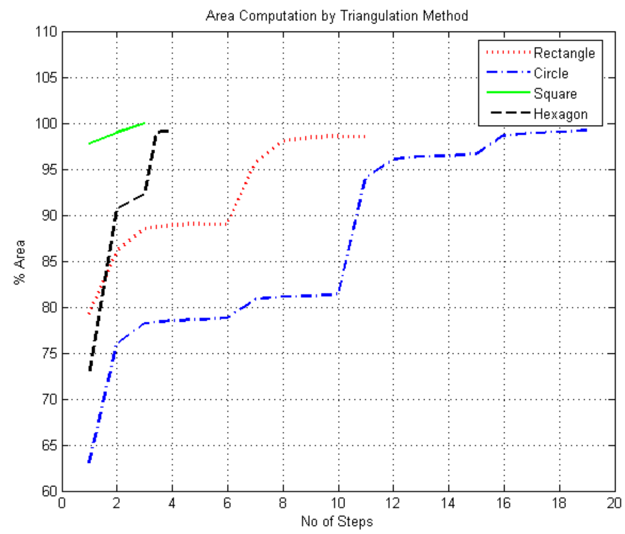

Figure 2: Plot of percentage of area computed v/s the number of steps in which area is computed by method of triangulation

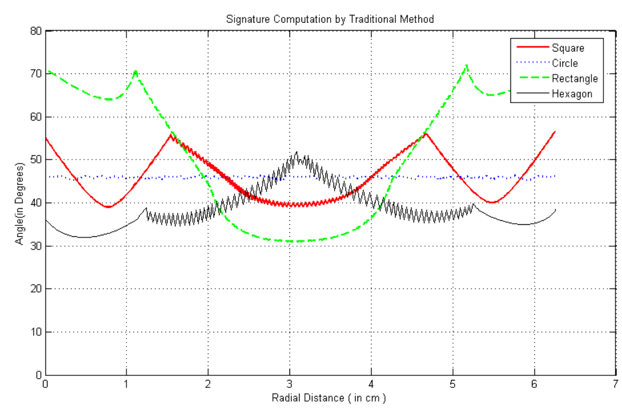

Figure 3: Signature Computed by the Traditional plot of the distance from the Centroid to the Boundary as a function of r(θ)

The number of steps in which area of a convex shape is computed using our triangulation method as explained above can also be used as a unique signature of the convex shape as depicted in Figure 4. The algorithm is run on different convex shapes with size variations from 1 to 10 scale. Figure 4 shows the signature is a constant value *2* for a square. The area of a square is calculated in two steps irrespective of its size, for a rectangle it is calculated in 5 to 15 steps after that it is a constant, the worst case is for a circle which is linear in nature up to 55 steps, after that it is a constant.

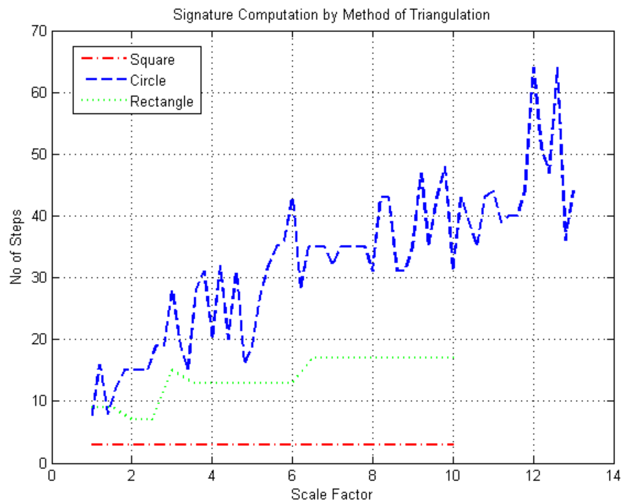

Figure 4: A New Signature by Triangulation Method for Convex Shapes

4 Conclusions

We presented an efficient algorithm for area and signature computation of convex shapes. Experiments are done on various convex shapes. Our area is computed in constant steps for various convex shapes. The best case is computation of area of a square shape in two steps irrespective of its size. The worst case is for circular shapes. For example, area of a 1200 x 1200 circle when scaled 10 times is calculated in 60 steps by our algorithm, which is computed in 1200 steps by the traditional pixel count area method. Our algorithm is almost *20* times faster than the traditional method in the worst case. For the best case it is *600* times faster. This method is effective in capturing both local and global characteristics of a shape invariant to translation, rotation, and scaling, and robust against noise.

References

[1] L. da F. Costa and R.M. Cesar. *Shape Analysis and Classification.* CRC Press, 2001.

[2] H. Freeman. In *Computer Processing of line-Drawing Images*, volume 6, pages 57–97. Computing Surveys, 1974.

[3] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing.* Addison Wesley Longman, 2nd ed., 2000.

[4] V. Hlavac M. Sonka and R. Boyle. *Image Processing, Analysis, and Machine Vision.* Chapman and Hall, 1993.

[5] Neeta Nain Vijay Laxmi Ankur Kumar Jain and Rakesh Agarwal. In *Morphological Edge Detection and Corner Detection Algorithm Using Chain-Encoding*, volume II, pages 520–525. IPCV'06, 2006.

[6] T. Pavlidis. *Algorithms for Graphics and Image Processing.* Computer Science Press, 1982.

[7] I. Pitas and A Venetsanopoulos. In *Morphological Shape Decomposition*, volume 12(1), pages 38–45. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990.

[8] D. Zhang and Guojun Lu. In *Review of Shape Representation and Description Techniques*, volume 37, pages 1–19. Pattern Recognition, The Journal of The pattern Recognition Society, 2004.

ISBN: 978-988-98671-8-8