Recent Research in Science and Technology 2011, 3(7): 73-76
ISSN: 2076-5061
www.scholarjournals.org

www.recent-science.com

RRST-Mathematics

Behavior of Normalized Moments under Distortion and Optimization

Vinay Saxena[1*] and V.V. Kapoor[2]

[1]*Department of Mathematics, Kisan Post Graduate College, Bahraich 271801(U.P.), India*
[2]*Flat No. 431, Plot No. 5, Sector 9 Dwarka, New Delhi 110045, India*

Article Info

Article History

Received	:	*29-05-2011*
Revised	:	*08-07-2011*
Accepted	:	*08-07-2011*

*Corresponding Author

Tel : *+919415178901*

Email:
dr.vinaysaxena@gmail.com

Abstract

Various types of moments have been used to recognize planar shapes. The algorithms are mostly based upon extracting moment features and train the machine to match these features with a database of templates. The shape could be represented by a polygon whose vertices lie on the boundary. The computational complexity of algorithm is a function of number of vertices of polygon. In this paper, we first present one such algorithm for hand drawn shapes. Vertices are picked up randomly as the user draws the shape with the help of mouse on a monitor. We have used a database of four templates for training the machine. The robustness of the algorithm based upon the moment features has been exhibited by matching a test shape that is a distortion of one of template stored in the database. In the second part of paper we have proposed an optimization technique that discards most of the redundant vertices of the polygon representing the shapes, thus reducing significantly the complexity. Integral square error norm is used to calculate optimal vertices and nearest – neighbor (NN) classifier for classifying the shapes. Empirical results have been presented for extracting the moment feature vectors.

Key Words: Hand drawn shapes; Contour sequence moment; Distortion function; Nearest - Neighbor classifier; Polygonal approximation
Zentralblatt Mathematics Subject Classification 2000: 68T10, 68U20, 74P99

Introduction

In this paper, a database of four hand drawn shapes has been used to train the machine to recognize shapes. A two-dimensional (2D) moment introduced by *Hu* [1] and one-dimensional (1D) contour sequence moments by *Gupta* and *Srinath* [2], with one component added to the feature vector has been used.

A computer program using the Borland Visual C++ is developed that picks up randomly the vertices of the polygon representing the shape as the user draws it on the monitor by dragging the mouse. The contour of geometric shape can be described by an ordered set N vertices $p_i = (x_i, y_i)$; i = 1,2,..,N

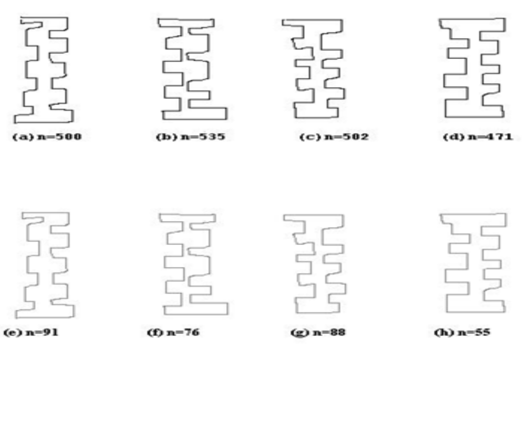

Figure 1. (i) Original shapes a-d (ii) Optimal shapes e-h

where p_{i+1} is a neighbor of p_i (module N). Let Z(i) be the Euclidean distance of the i[th] vertex from the centroid. For each hand drawn shape, the contour sequence moment is defined as the array Z(i): i = 1,2,..,N.The hand drawn shapes a-d are displayed in Figure 1 and the corresponding contour sequences are displayed in Figure 2.

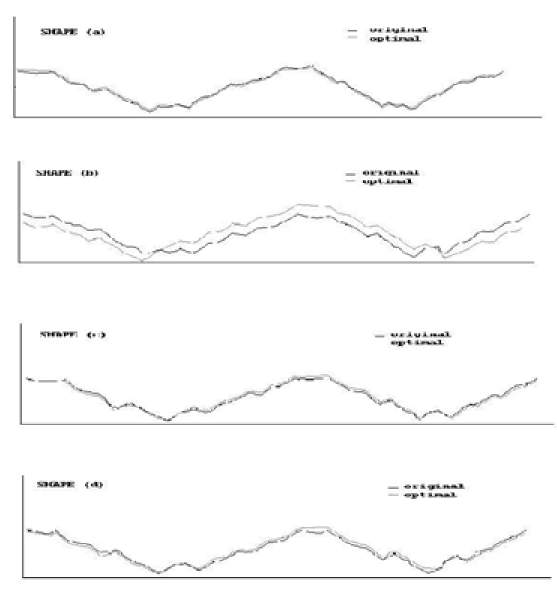

Figure 2. Contour sequence moment and corresponding optimal contour sequence moment

With the help these contour sequences the components (F_1, F_2, F_3, F_4, F_5) of the feature vector F are defined as:

$$F_1 = \frac{[\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^2]^{1/2}}{\frac{1}{N}\sum_{i=1}^{N} Z(i)};$$

$$F_2 = \frac{\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^3}{[\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^2]^{3/2}};$$

$$F_3 = \frac{\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^4}{[\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^2]^{2}};$$

$$F_4 = \frac{\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^5}{[\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^2]^{5/2}};$$

$$F_5 = \frac{\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^6}{[\frac{1}{N}\sum_{i=1}^{N}[Z(i)-m_1]^2]^{3}};$$

$$where\ , m_1 = \frac{1}{N}\sum_{i=1}^{N} Z(i)$$

The feature vectors *F* of the hand drawn shapes are listed in Table I.

Table I: Normalized contour sequence moments of original shapes a-d and their optimal shapes e-h

	F_1	F_2	F_3	F_4	F_5
Original:					
Shape (a)	0.459864	-0.0735808	1.71522	-0.015820	3.47974
Shape (b)	0.460720	-0.0414717	1.67194	-0.114521	3.23941
Shape (c)	0.491434	0.0287178	1.69621	-0.024448	3.42233
Shape (d)	0.466609	0.0556029	1.85951	0.109075	4.06449
Optimal:					
Shape (e)	0.460402	-0.1687660	1.64606	-0.381847	3.20971
Shape (f)	0.550918	0.2842860	2.49774	2.064740	8.85521
Shape (g)	0.486562	0.3314800	2.34005	1.823670	7.39668
Shape (h)	0.431112	-0.0848057	2.03209	-0.479509	5.16413

Classifying Distorted Shapes

Distortion function:

The distortion function takes two arguments: the first a real number p and second an offset (dx,dy) , it randomly selects p percentage of the vertices of the shape and shifts each one of them to any one of the eight neighboring points ± (x±dx,y±dy) randomly.

For example, for p=0.1 and for dx=1, dy=1, 10% of the randomly selected vertices will shift to one of their respective eight nearest-neighbors randomly. The distortion is modeled as a Gaussian random noise. The nearest-neighbor (NN) classifier has been used for matching the distortion with the original. The description of NN classifier is given below.

NN classifier:

The nearest-neighbor classifier labels an unknown shape represented by a 5-dimensional feature vector X =[X_1, X_2, X_3, X_4, X_5] with the label of the nearest neighbor of X among all the training samples. The distance between X and a training sample are measured using Euclidean distance. This is a mapping from 5-dimensional feature vector space onto a 1-dimensional Euclidean space.

Let F^K = [$F^K{}_1$; $F^K{}_2$; $F^K{}_3$; $F^K{}_4$; $F^K{}_5$], K=1,2,3,4 be the 5-dimensional training feature vector of the K-th class. The unknown test sample X is classified to class K^*, where

K^* = arg $\min_K$ d (X,F_K) ;   K=1,2,3,4.

Computer simulation:

Distortion function has been used to introduce monotonically increasing noise. The noise introduced in this manner alters the amplitude duration and over all shape of resulting contour sequence representation. In the training phase, reference feature vectors F^K =[$F^K{}_1$; $F^K{}_2$; $F^K{}_3$; $F^K{}_4$; $F^K{}_5$], K=1,2,3,4 were computed for all four hand drawn shapes. In the testing phase one hundred distorted shapes of each of the shape has been generated using a particular value of dx, dy and p. NN classifier described above has been than used to recognize by matching the distortion with the original shape.

We have found that in almost hundred percent of the cases the machine is able to classify the shapes correctly if the arguments to the Gaussian distortion function are assigned reasonable value. This establishes that the Gaussian distortion has very little effect in case of hand drawn shapes. In other words the normalized moment descriptor is able to withstand small distortions.

In the next section we show that only a small portion of the vertices play an important role and most of the vertices could be eliminated by an optimization technique, thus reducing the time complexity significantly with minimal effect on the accuracy.

Effect of Optimization

The computational complexity of extracting feature vector is obviously proportional to a polynomial function of number of vertices of the polygon representing the shape. In this section we describe an optimization technique and study its effect on the wisdom of machine. The algorithm discards most of the almost collinear vertices whose contribution towards the shape is negligible. The algorithm that is a slight modification of that given by *Ray and Ray* [3] is stated below.

Algorithm for optimal Polygon using Integral Square error:

Comments: The inputs are the data points (x_i, y_i), i = 1,..,N. The outputs are the optimal vertices (x_{j^*}, y_{j^*}) and j*. All arithmetic is in modulo N.

Step 1: Initiate i=1.
Step 2: Set j = i + 1.
Step 3: Compute $F_j = \{(x_j - x_i)^2 + (y_j - y_i)^2\}^{1/2}$
Step 4:Change j to j+1
Step 5:Compute

$$F_j = \{ (x_j - x_i)^2 + (y_j - y_i)^2\}^{1/2} - \frac{\sum_{k=i+1,j-1} \{(y_j - y_i)\, x_k - (x_j - x_i)\, y_k + x_j y_i - x_i y_j\}^2}{\{(x_j - x_i)^2 + (y_j - y_i)^2\}^{1/2}}$$

Step 6: If $F_j \geq F_{j-1}$, then go to step 5
Else write j* = j-1 and (x_{j^*}, y_{j^*})
Set i = j* and go to step 2
Step 7: Repeat this process until j* is repeated.
Step 8: Join (x_{j^*}, y_{j^*}) successively to determine the optimal polygon.
Step 9: End of algorithm

Table II compares the number of vertices in the original shape to that of the optimal shape. We have found that the percentage reduction in number of vertices varies between 80 and 90. Figure2 shows the contour sequence moment and corresponding optimal contour sequence moment of all hand drawn shapes.

The computer simulation, described above, is repeated by using the four optimal shapes. We find that there is practically no effect on the rate of recognition if the noise introduced is small. Indeed, the recognition rate is around 98% even when 25% of vertices are distorted by three pixel each. The machine may falter only if one of the corner vertex is shifted significantly from its original position. That establishes that normalized moment descriptor could be used for hand drawn noisy shapes as well.

Table III exhibits the probability error (the ratio of the number of mismatched to that of the total number of noisy shapes) and the recognition rate after optimization distortion steps. We emphasize that the optimization technique will not compromise on the wisdom of the machine.

Table II: Reduction of vertices using E_2 error norm

	Number of vertices		Comparison rate (n / n_V)	Percentage of data reduction
	Original shapes (n)	Optimal shapes (n_V)		
Shape (a)	500	91	5.49	81.80
Shape (b)	535	76	7.04	85.79
Shape (c)	502	88	5.70	82.47
Shape (d)	471	55	8.56	88.32

Table III: Recognition rate

Distortion			Original shape			Optimal shape		
p	dx	dy	Number of mismatches out of 400	Probability error	Recognition rate	Number of mismatches out of 400	Probability error	Recognition rate
0.25	1	1	0	0.0	100	0	0.0	100
0.25	2	2	0	0.0	100	2	0.005	99.5
0.25	3	3	0	0.0	100	8	0.02	98.0
0.50	1	1	0	0.0	100	1	0.0025	99.75
0.50	2	2	0	0.0	100	5	0.0125	98.75
0.50	3	3	0	0.0	100	9	0.0225	97.75

Conclusion

We conclude that the contour sequence moments are able to recognize hand drawn shapes with almost hundred percent accuracy under small Guassian distortion and the uniqueness theorem given by *Papoulis*[4] for 1D contour sequence moments is stable even under noisy shapes. In the second part, we have shown that the time taken for evaluation of the optimal feature vector was between $(1/10)^{th}$ to $(1/20)^{th}$ of the original but the accuracy remained almost the same.

References

[1] M.Hu, (1962), Visual pattern recognition by moment invariants, *IRE Trans. Inf. Theory* 8, 179-187.

[2] L.Gupta and M.D.Srinath, (1987), Contour sequence moments for the classification of closed planar shapes, *Pattern Recognition* 26, 267-272.

[3] Bimal Kr. Ray and Kumar S. Ray, (1994), A non-parametric sequential method for polygonal approximation of digital curves, *Pattern Recognition Letters* 15, 161-167.

[4] Papoulis, A., (1965),Probability, Random variables and Stochastic Processes, McGraw-Hill, New York.