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

Decomposing a Polygonal Boundary as a List of Line Segments

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

Abstract—**Shape identification is a fundamental property of an object and is an important low level feature to human perception. 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. This paper describes a new shape descriptor using chain- encoding methods, where a shape boundary is represented as a list of vectors of line segments. This is a very compact and efficient method of shape representation which could be exploited for shape compression, classification, segmentation and shape matching techniques.**

Keywords: Shape, Chain Code, Similarity, Standard Deviation.

1 Introduction

Shape is one of the most important low level image features, which remains intact even after the introduction of sampling noise, quantization noise and applications of geometric transformations. Thus shape is a very important feature to human perception. Object recognition is a learned process. The mathematical model that describes an object is very important as our brain stores the mathematical model of the object and when we see any object the brain again computes the mathematical model and compare it with already learned models to determine the object. Describing the 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, scale and rotation.
Two types of shape descriptors are mainly used in image analysis. Boundary based shape descriptors and region based shape descriptors. Boundary based shape descriptors are computationally efficient since they process small number of points as compared to region based image descriptors. This paper describes a transformation invariant boundary based shape descriptor.
Section 2 details the proposed shape representation technique. The test results are discussed in section 3 followed by conclusions in section 4.

2 Shape as a list of straight line segments

Shape analysis [1, 2, 3, 4, 5] requires a compact and reliable shape representation which is robust to sampling noise, translation, rotation and scaling. It has long been known that information about shape is conveyed via the changes in the slope of an object's boundary. The information content is greatest where this change is strongest. Line structure representation of object boundary is preferred for man-made objects. For example, for road map analysis, text analysis, architectural structure analysis etc. Various shape representation techniques are described in [6, 7, 8, 9, 10] .
We present a new algorithm for representing an object's shape as a list of straight line segments. To extract shape we need precise, compact and continuous in a segment boundary. One-pixel thick m-connected boundary is extracted using our own morphological algorithm [3]. One-pixel thick, and m-connectivity avoids redundancy in chain codes. This boundary is *Chain-Encoded* using $8-$way chain encoding [4] method.
The boundary of an object is represented as a polygonal structure, which is initially represented as a chain code. The chain is then post processed to extract straight line segments. The polygon then becomes an assembly of relatively small number of straight-line segments. We have designed a simple method where the occurrence of various vectors of successive similar chains in the boundary chain code are recorded as a vector of line segments. The segment is defined as a part of the boundary with constant slope and run length. The boundary is represented as a Freeman code with 8-way connectivity. Thus in the segment with constant slope, the code (direction) is unchanged, that is the difference between consecutive codes should be zero. We calculate the length of such segments and use run length coding to represent a line segment of some length with a particular slope as a vector. These vectors can be then used for comparing two shapes. The steps are as follows:

*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.

1. Extract one-pixel thick m-connected boundary.

2. Chain Encode Boundary: Chain encode the boundary using 8-way connectivity. The numbers in the chain code represents the direction taken by the next point.

3. Translation Invariance: Make the boundary translation invariant by calculating the first order differences of the chain codes. The first difference in the chain codes represents the turning angles from the previous point. Note that the first difference is computed by treating the chain code as a circular sequence. For constant slope segments this difference would be zero.

4. Run length Coding: The boundary is represented as a vector of straight line segments(vector of constant slope and length of the segment is computed from the run length).

5. Scale Invariance: To expand or to contract a chain by a specified scale factor, one must appropriately scale each chain link and then re quantize. In down sampling, a number of links may merge into one, and in up-sampling one link may cause a string of many links to be generated. The generation of these new links is most easily handled by means of using Bresenham scan conversion technique. Compared to up sampling, down sampling is robust to aliasing effects. Bresenham scan conversion ensures that the error is always $<= \frac{1}{2}$ pixel. When a shape is scaled, the elements of the obtained segment vectors are proportionally increased or decreased. This re-sampling makes the segment scale invariant.

6. Orientation Invariance: Align the polygon with minimum angle with the x - axis. This is done by circularly shifting the above difference code so that the segment starts with the first difference of smallest magnitude. Circular shift makes the shape rotation invariant.

The result is a polygonal approximation (as a vector of straight line segments), compact and coarser than the chain from which it is derived, but one which highlights the significant information conveying features of the boundary. Such a scheme can be utilized for developing a classification system for certain families of line structures, for example, fingerprints, type fonts, and describing chromosomes as well as for providing a basis for a syntactical description of shape. This technique also achieves 20% to 30% compression in shape representation.
It can also be used for shape comparisons. Given two shapes represented as vectors: $v_1 = \{a_1, a_2, a_3, a_4, ..., a_n\}$, $v_2 = \{b_1, b_2, b_3, b_4, ..., b_n\}$, the ratio of the two vectors $\{v_1, v_2\}$ is $\{a_1/b_1, a_2/b_2, a_3/b_3, a_4/b_4, ..., a_n/b_n\}$. For similar shapes the ratio vector would be a constant vector. Root mean square error and standard deviation are used for similarity measure between two shapes. Table 1 depicts the results of the algorithm run on Fig. 1. The results show insignificant difference between similar figure(rotated and translated), and remarkable difference between different figures. Hence the proposed algorithm is geometric transformation invariant.

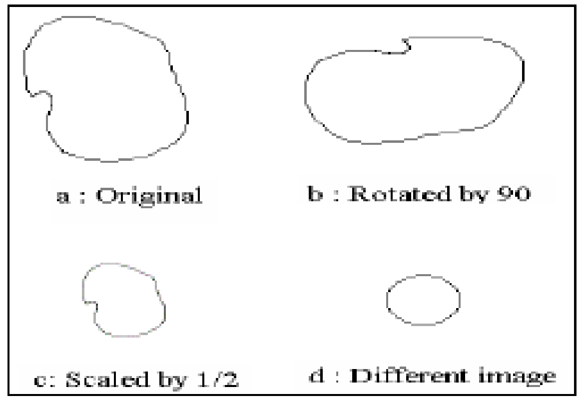

Figure 1: Example Figure with Regular Changes in Slope

Similarity Measure	Original v/s Rotated(90°)	Original v/s Scaled(1/2)	Original v/s Different
Root Mean Square Error	0.0522	0.1430	3.2711
Standard Deviation	0.0261	0.1053	0.5173

Table 1: Similarity Measures for LSLS Algorithm on Figure 1

3 Test results

Tests were performed on three categories of polygonal figures.

1. Straight line polygons. Figures containing only straight lines in the boundary.

2. General polygons. Figures containing both straight lines and curves on the boundary.

3. Curved polygons. Figures containing only curves on the boundary.

Tests for similarity measure on scaled and rotated figures were done for over 100 figures from each category. The images were scaled up to 200% in 20 point increments Sand rotated by full 180° in 20° increments. For poly-

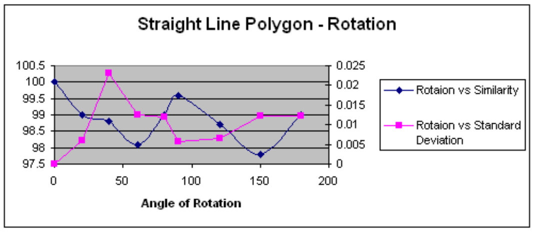

Figure 2: Similarity measure computed on figures having straight line segments at different orientations

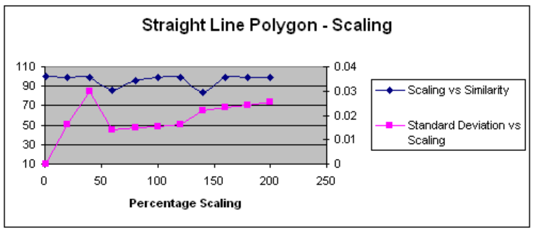

Figure 3: Similarity measure computed on figures having straight line segments at different scales

gons having straight line segments, this representation gives excellent similarity measure $> 98\%$, with standard deviation < 0.023 irrespective of the angle of rotation as shown in Fig. 2. And with respect to scaling it gives similarity measure $> 85\%$, with standard deviation < 0.03 irrespective of scaling factor as shown in Fig. 3 . It also gives very good results on general figures. For general figures the standard deviation varies from $0.0 - 0.12$. The

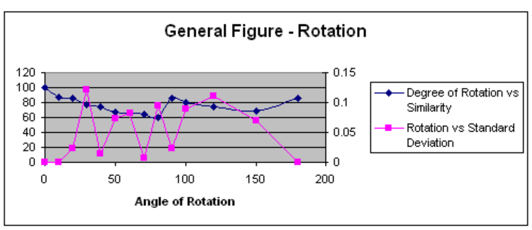

Figure 4: Similarity measure computed on general figures having both straight lines and curved segments at different orientations

similarity measure is very good $> 80\%$ for figures rotated by any multiple of $45°$, it though decreases for intermediate rotations, but that too is not very significant as the similarity is always $> 65\%$ for even the worst cases as shown in Fig. 4. When general figures were scaled upto 200%, the similarity measure was always $> 60\%$ irrespective of the scaling factor as shown in Fig. 5. It also gives very good results on all curved figures as depicted in Fig. 6 where similarity measure is $> 56\%$ irrespective of angle of rotation angle with standard deviation always < 0.23. It though gives a similarity measure $> 50\%$, with standard deviation < 0.49 irrespective of scaling factor as depicted in Fig. 7. The results show insignificant difference between similar figure(rotated and translated), and remarkable difference between different figures. Hence the proposed shape representation is geometric transformation invariant.

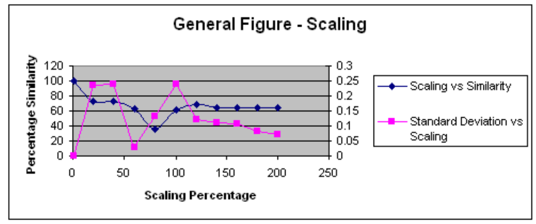

Figure 5: Similarity measure computed on figures having both straight lines and curved segments at different scales

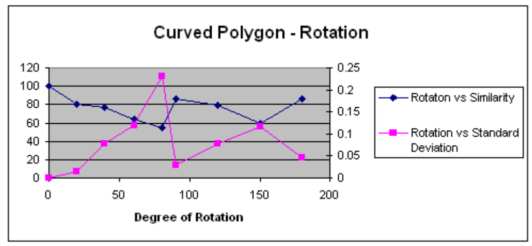

Figure 6: Similarity measure computed on figures having curved segments at different orientations

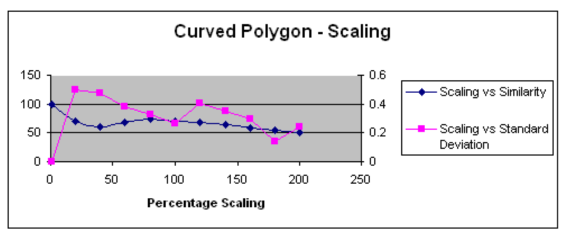

Figure 7: Similarity measure computed on figures having curved segments at different scales

4 Conclusions

We presented an efficient algorithm for representing object shapes as a vector of straight line segments. Excellent results are derived for shape comparison where root mean square error and standard deviation are used as similarity measures. Tests for similarity measure on scaled and rotated figures were done over 100 figures. The images are scaled up to 200% and rotated full 360°. For rotations of the multiple of 90the figures shows 100% similarity. The similarity decreases for intermediate rotations, but that too is not very significant as the similarity is $> 65\%$ for worst cases. For polygons having straight line segments, this representation gives excellent similarity measure $> 85\%$, with standard deviation < 0.03. It also gives very good results on curved figures as depicted in Figure 6 and Figure 7. For general figures the standard deviation varies from $0 - 0.5$. This scheme of representation can also be used as a compression technique for line drawing algorithms. For straight line polygons it could achieve $> 50\%$ compression.

References

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

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

[3] 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.

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

[5] S. Loncaric. In *A Survey of Shape Analysis Techniques*, volume 8, pages 983–1001. Pattern Recognition, 1998.

[6] 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.

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

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

[10] O. E. Badawy and M. Kamel. In *Shape Based Image Retrieval Applied to Trademark Images*, volume 3, pages 375–393. International Journal of Image and Graphics, 2002.