8-32

MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, JAPAN

Represent Image Contents Using Curves and Chain Code

Pornchai Mongkolnam[†], Chakarida Nukoolkit[†] and Thanee Dechsakulthorn
[†]School of Information Technology
[†]King Mongkut's University of Technology Thonburi
Bangkok, Thailand
Email: pornchai@sit.kmutt.ac.th

Abstract

This paper presents a new approach to extracting and representing structural features of images. The approach is based on both a region-based analysis and a contour-based analysis. An image is first segmented into regions, after which an area of each region is computed using a number of bound pixels. A contour of the region is approximated by a B-spline curve to get its control polygon. A convex hull of each region is then computed from the obtained control polygon. Ellipses are used to intuitively represent structural components in the image. Each ellipse is centered at its corresponding convex hull's centroid and is approximated from the convex hull. Color and spatial information of each region is also used. In a multi-scale setting, the Chaikin's algorithm is iteratively applied to the obtained B-spline control polygon in order to get finer control polygons, which in the limit results in the B-spline curve. To compare similarity between the control polygons and their higher-level refined control polygons, we use the chain code representation in concert with the Chaikin's algorithm. Our method could be applied in many applications including image retrieval, image classification, image clustering, image understanding, image manipulation, pattern recognition, and definitely in machine vision.

1. Introduction

Image and video search engines such as those from Google are the compelling and living evidences of how images and videos have become part of our lives and how they could be readily acquired and shared. Various techniques have been proposed to include relevant features such as color, texture and shape into a mere text-based image search in order to improve the desired search results. The keyword-only search has some disadvantages in that keywords are context dependent and do not allow for an unanticipated search. In addition, language barriers and lacks of uniform textual descriptions further weaken the search results.

A sizable work has used B-splines to represent object contours [3, 16]. The contour is first extracted and then approximated by its corresponding B-spline, which in turn is used for curve matching in a data retrieval application. The object matching is an integral part for many applications of shape modeling, machine vision, and image processing. The B-splines and their curvatures are widely used for curve representations instead of a far higher degree Bezier curves because they possess some attractive properties such as smoothness, compactness, local shape controllability, and affine transformation invariance. Much less work, however, has used the B-spline representation in a two-dimensional image analysis. Our work has used the B-splines to approximate extracted image contours, and has applied the Chaikin's algorithm [2] to refine curves at different scales, starting from the control polygons. A coarser representation than the control polygon is an ellipse, which is approximated from a convex hull of the control polygon. An excellent literature review on shapes used in content-based image retrieval (CBIR) can be found in [10, 12, 15]. Knowing image components, structural shapes, and their spatial relationships would yield good search and comparison results, as opposed to simply knowing color and textual information.

The rest of the paper is organized as follows. First, related work on 2D and 3D shape representation is given. Second, Our work on B-splines, curve approximation, elliptical approximation, Chaikin's algorithm, and chain code representation is provided. And last, conclusion and future direction of our work is given.

2. Related Work

Some prior work in a 2D and 3D shape matching and curve representation has greatly influenced this work. Our work has built on previous concepts and extends them to an image component level. Over the past thirty years, work on shape has been an active research area and was mainly driven by object recognition. One recent work [16] proposes a novel 2D shape-matching algorithm based on the B-spline curves and its Curvature Scale Space (CSS) image. Thanks to attractive properties of B-spline, another work [3] chooses the B-splines for curve modeling in matching 2D objects like aircrafts and handwriting over other approaches such as the Fourier descriptors, the polygonal approximation, the medial axis transform, the moments, and the curvature invariant. The algorithm attempts to match and classify planar curves that are modeled as B-splines, independent of any affine transformations.

An image database retrieval method [14] based on shape information and deformable template matching process is proposed by using the following two features to represent shape of an image: a histogram of the edge directions and the invariant moments. Euclidean distance between the edge direction histograms is used as a matching score. A region-based image retrieval method [7] that employs a new image segmentation technique using circular filters based on Bayes' theorem and image texture distribution is also proposed. After segmentation, extracted features of each region including color, texture, normalized area, shape, and location are recorded and compared against other images.

Most previously mentioned work is performed on a single scale shape analysis, which is not able to provide a robust representation. A multi-scale analysis for shapes [5] is used to derive a hierarchical shape representation in that the shape details are progressively screened out whereas the shape characterizing elements are preserved. Besides using a curve matching, a shape matching can also be achieved by matching skeletal or medial axis graphs as done in [11, 13]. The medial axis is used for matching shapes because outline curves alone do not meaningfully represent the interior of the shapes. For a complete overview of image retrieval based on shape we recommend reading in [19].

3. Our Work

We first segment an image using JSEG algorithm [4]. The segmented boundaries are then approximated by B-spline curves. They are chosen because of their desirable local control and lower-degree approximation properties. The B-spline curve [6], *C(u)*, is defined as:

$$C(u) = \sum_{i=0}^{h} N_{i,p}(u) P_i$$

where P_i is a control point, p is a degree, u is parameter, and $N_{i,p}$ is a B-spline basis function and defined as:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

where u_i is known as a knot, a corresponding place where two curve segments join with certain continuity. In the approximation the B-spline curve does not have to pass through all data points except probably the first and last data points. A number of the B-spline control points would reflect a goodness of the approximation. For each data point, an error distance is computed as the square distance between the data point and a corresponding point on the B-spline curve. A sum of all square error distances is used to measure how well the B-spline curve approximates the data points. An objective is to minimize the sum of the error distance in order to get a good approximation of the data points. A problem statement of the B-spline approximation is posed as:

Input: Given a set of n+1 ordered data points, $D_0, \ldots, D_n$.

Output: A B-spline curve of degree p with h+1 control points, $P_0, \ldots, P_h$, which satisfies the following two conditions.

- The curve interpolates the first and last data points, D_0 and D_n and
- The curve approximates the data points in the sense of a least square error distance.

Since the curve contains the first and last data points, we would have $D_0 = P_0$ and $D_n = P_h$. The curve equation is now written as:

$$C(u) = N_{0,p}(u) D_0 + \left(\sum_{i=1}^{h-1} N_{i,p}(u) P_i \right) + N_{h,p}(u) D_n$$

Let parameters be $t_0, \ldots, t_n$. The number of parameters is equal to the number of the data points because we want to find the corresponding point on the curve for each data point. The centripetal parametrization is used and computed as:

$$\frac{\Delta_i}{\Delta_{i+1}} = \left[\frac{\|\Delta x_i\|}{\|\Delta x_{i+1}\|} \right]^{1/2}$$

where $\Delta_i = t_{i+1} - t_i$ and $\Delta x_i = D_{i+1} - D_i$.
The sum of all square error distances is computed as:

$$f(P_1, \ldots, P_{h-1}) = \sum_{k=1}^{n-1} \left| D_k - C(t_k) \right|^2$$

The control points, $P_1, \ldots, P_{h-1}$, are solved such that the objective function $f(.)$ is minimized. Figure 1 illustrates an open-ended curve approximation of the test data.

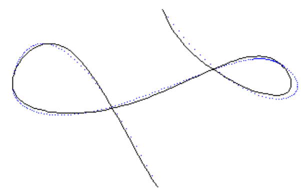


Figure 1. B-spline approximation.

The obtained control polygon of each segmented region is used as a first-level representation. For finer representations, i.e., second-level or third-level, the Chaikin's algorithm would then be applied to the control polygon repeatedly to get to the desired level. The algorithm is defined as: given a control polygon, defined by $\{P_0, \ldots, P_n\}$, a new refined sequence of control points is $\{Q_0, R_0, Q_1, R_1, \ldots, Q_{n-1}, R_{n-1}\}$, where each new pair of points $\underline{Q_i, R_i}$ is computed to be at ¼ and ¾ of a line segment $\overline{P_i P_{i+1}}$. Figure 2 illustrates resulting polylines from a coarse-to-fine scale.

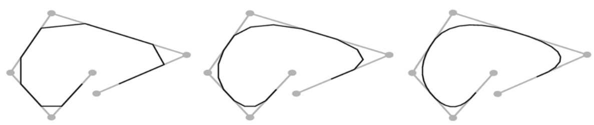


Figure 2. Applying the Chaikin's algorithm to the control polyline three times.

For the coarsest level (level zero), a convex hull of each segmented region is computed from the obtained control polygon (the first-level representation) and then approximated by a corresponding ellipse for a more compact and better representation than the convex hull itself, as illustrated in Figure 5. Ellipse is used in approximation here instead of a circle because of the oblong and unequilateral nature of shapes in general. In addition, a circle is simply a degenerate ellipse. The elliptical approximation method includes the following steps. First, the centroid of each convex hull is computed and used as the center of the ellipse. Second, the lengths of the semi-major axis and semi-minor axis are computed and used to generate the ellipse. Third, sum all the differences between each convex hull's vertex and its corresponding intersecting point on the ellipse when intersect the ellipse by a line from the vertex to the center. Fourth, increase orientation angle of the major axis and minor axis by some degrees (5 degrees in our work) and repeat step two to step four. Last, select the ellipse that results in the smallest sum of differences. Note that we need to tilt the ellipse's orientation up to 90 degrees; beyond the 90 degrees it is simply duplication. Figure 3 illustrates a zero-degree axis orientation of elliptical approximation to a rectangle.

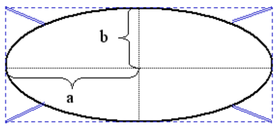
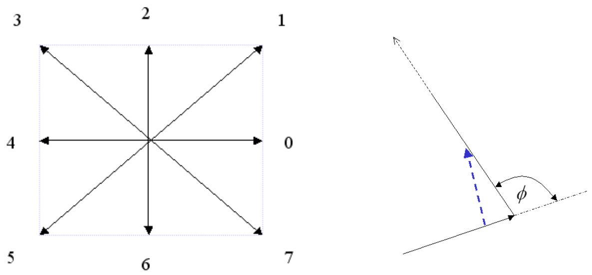


Figure 3. Elliptical approximation to a rectangle indicated by dotted line. The *a* and *b* represent the lengths of semi-major and semi-minor axis. Four dark lines represent the differences between the rectangle's vertices and their corresponding intersecting points on the ellipse.

Multi-scale representation and comparison that involve the applications of both the Chaikin's algorithm and chain code are done from the control polygons to their subsequent subdivision polygons. The 8-direction chain code [9] shown in Figure 4(a) is used to represent shape of the polygon.



(a) (b)

Figure 4. (a) The 8-direction chain code. (b) Corner cutting and its direction after applying the Chaikin's algorithm.

For example, a chain of directional code 2, 4, 6, and 0 could represent an upright square if we start from the lower right corner in a counterclockwise direction. For the tilted one the code would be 1, 3, 5, and 7. Because an angle between any two subsequent codes is 45 degrees, and thus the chain code could be made translation, scale and rotation invariant. In addition to directional code, distance ratio of each polygon's leg is also used to further differentiate similar shapes such as squares and rectangles. The distance ratio is also needed to compute direction at each polygon's corner when applying the Chaikin's algorithm. The direction at the corner is illustrated in Figure 4(b). The resulting direction is from the location ¾ of the start leg to the location ¼ of the next leg, and its angle is always within ϕ.

4. Discussion and Future Work

We have presented a technique for extracting structural shapes of images for possibly uses in various image and video applications. Our work focuses on the multi-scale shape representation, starting from convex hulls to control polygons at different scales and eventually to B-spline curves in the limit. The B-spline approximation is used to get the control polygons, which in turn are used to obtain the corresponding convex hulls. Ellipses are used to intuitively represent extracted image components instead of the convex hulls because of a better and more compact representation. The Chaikin's algorithm is applied to the control polygons repeatedly to get finer control polygons for better structural comparisons at finer levels. The comparison could begin from the elliptical level, to the control polygon, and to the subsequent subdivision polygons. The directional chain code is also used to represent and compare polygons.

We plan to apply the method to image and video retrieval both in the content-based approach and semantic-based approach. Three-dimensional object segmentation and matching for biomedical data could theoretically be extended from our existing 2D work. The different-level control polygons would be extended to multi-level mesh control polyhedrons. The limit B-splines would then be B-spline tensor product or surfaces. The Chaikin's algorithm still works for 3D meshes as it does for the 2D planar curves.

To help us better understand the results and their relationships, better approach to render results meaningfully ought to be done. More sophisticated visualization work in both a two-dimensional and three-dimensional setting is practically needed. To display the result in two-dimensional we could apply the concept of the pictorial summary [18]. The dominant images of the search result or the cluster would be given more spaces and more prominent location than the less significant ones. The Catmull-Clark subdivision surface [1] would be used together with the terrain modeling [8, 17] for a three-dimensional display. The results would be displayed as a relief map of natural terrain where the dominant groups of images are shown at the taller peaks.

Acknowledgments

We wish to acknowledge the support of the National Science and Technology Development Agency (NSTDA) of Thailand under grant F-31-205-20-01 and the KMUTT's Research and Intellectual Property Promotion Center (RIPPC).

References

[1] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Computer-Aided Design (CAD)*, pp. 350-355, 1978.

[2] G. Chaikin, "An algorithm for high speed curve generation," *Computer graphics and image processing*, pp. 346-349, 1974.

[3] F.S. Cohen, Z. Huang, and Z. Yang, "Invariant matching and identification of curves using B-splines curve representation," *IEEE Transactions on Image Processing*, vol. 4, no. 1, pp. 1-10, Jan. 1995.

[4] Y. Deng, B.S. Manjunath, and H. Shin, "Color image segmentation," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR '99, Fort

Collins, CO, vol. 2, pp. 446-451, Jun. 1999.
[5] A. Del Bimbo and P. Pala, “Shape indexing by multi-scale representation,” *Image and Vision Computing*, vol. 17, nos. 3-4, pp. 245-261, 1999.
[6] G.E. Farin, “*Curves and Surfaces for CAGD: A Practical Guide*,” Morgan Kaufmann, 5th edition, Oct. 2001.
[7] B.C. Ko and H. Byun, “FRIP: A region-based image retrieval tool using automatic image segmentation and stepwise Boolean AND matching,” *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 105-113, Feb. 2005.
[8] N.E. Miller, P.C. Wong, M. Brewster, and H. Foote, “Topic Islands™—A wavelet-based text visualization system,” *Visualization’98*, pp. 189-196, Oct. 1998.
[9] H. Nishida, “Shape retrieval from image databases through structural feature indexing,” *Vision Interface ’99*, Trois-Rivieres, Canada, pp. 328-335, May 1999.
[10] A. Pentland, R.W. Picard, and S. Sclaroff, “Photobook: Tools for contents-based manipulation of image databases,” *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233-254, 1996.
[11] T.B. Sebastian and B.B. Kimia, “Curves vs. skeletons in object recognition,” *International Conference on Image Processing*, 2001. Proceedings, vol. 3, pp. 22-25, Oct. 2001.
[12] A.W.M Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, Dec. 2000.
[13] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson, “Skeleton based shape matching and retrieval,” *Shape Modeling International*, pp. 130-139, May 2003.
[14] A. Vailaya, Y. Zhong, and A.K. Jain, “A hierarchical system for efficient image retrieval,” *Proceedings of the 13th International Conference on Pattern Recognition*, vol. 3, pp. 356-360, Aug. 1996.
[15] J.Z. Wang, J. Li, and G. Wiederhold, “SIMPLIcity: semantics sensitive integrated matching for picture libraries,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947-963, Sep. 2001.
[16] Y. Wang and E.K. Teoh, “A novel 2D shape matching algorithm based on B-spline modeling,” *International Conference on Image Processing, ICIP'04. 2004*, vol. 1, pp. 409-412, Oct. 2004.
[17] J.A. Wise, J.J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, “Visualizing the non-visual: spatial analysis and interaction with information from text documents,” *Proceedings of IEEE’95 Information Visualization*, pp. 51-58, Oct. 1995.
[18] M.M. Yeung and B-L. Yeo, “Video visualization for compact representation and fast browsing of pictorial content,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 5, pp. 771-785, Oct. 1997.
[19] D. Zhang, “Image retrieval based on shape,” *Ph. D. Dissertation*, Monash University, Australia, Mar. 2002.

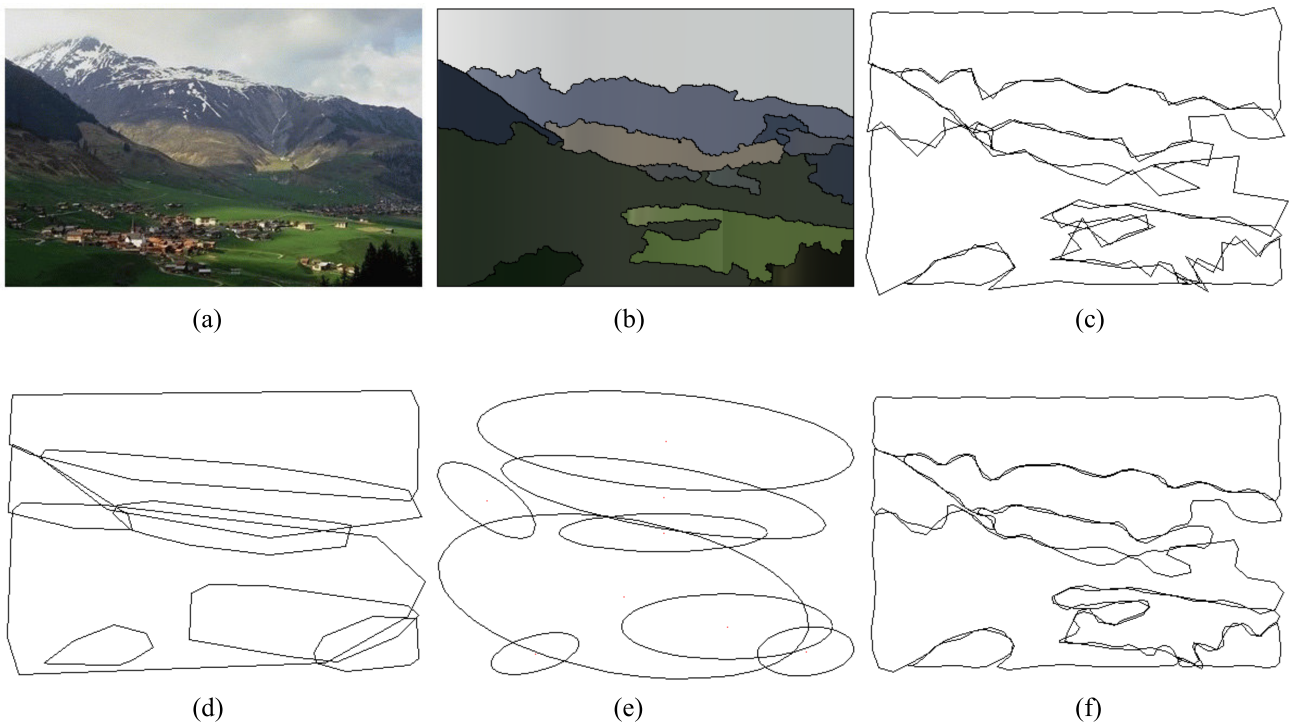


Figure 5. Summary of image segmentation and its various representations. (a) Original image. (b) The image with segmented regions. (c) Obtained control polygons after performing B-spline approximation to the segmented regions. (d) Convex hulls of the corresponding control polygons. (e) Elliptical approximations of the corresponding convex hulls. (f) Resulting polygons after applying the Chaikin’s algorithm to the control polygons one time.