Resiliency and Robustness of Alternative Shape-Based Image Retrieval Techniques *

Maytham Safar  Cyrus Shahabi  and  Chung-hao Tan
Integrated Media Systems Center
Department of Computer Science
University of Southern California
Los Angeles, California 90089–0781
[safar, shahabi, chunghal]@usc.edu

Abstract

Shape of an object is an important feature for image and multimedia similarity retrievals. However, as a consequence of uncertainty, shape representation techniques may sometimes work well only in certain environments, and their performance may depend crucially on the quality of the technique used to represent shapes. In this study we focus on shape-based object retrieval under various uncertainty scenarios and conduct a comparison study on four techniques. We measure the effectiveness of the similarity retrieval of the four different shape representation methods (in terms of recall and precision) under the following situations: 1) in the presence of noise in the database, 2) when the exact corner points are unknown, 3) factoring in the human perception on similarity. Our results show that the similarity retrieval accuracy of our method ($MBC-TPVAS$) is better than that of the other methods under uncertainty and discrepancies.

1. Introduction

Several applications in the areas of CAD/CAM and computer graphics require to store and access large databases. A major data type stored and managed by these applications is representation of two dimensional (2D) objects. Objects contain many features (e.g., color, texture, shape, etc.) that have meaningful semantics. From those, shape is an important feature that conforms with the way human beings interpret and interact with the real world objects. The shape representation of objects can therefore be used for their indexing, retrieval, and as a similarity measure.

Several techniques has been proposed in the literature for shape representation [12, 4]. In [4], shape representation techniques are divided into two categories: boundary-based and region-based. Boundary based methods use only the contour or border of the object shape and completely ignore its interior. On the other hand, the region based techniques take into account internal details (e.g., holes) besides the boundary details. Three boundary-based (termed $MBC$-based) approaches to shape representation and similarity measure were proposed by the authors in [10, 6]. To test the efficacy of one of them ($MBC-TPVAS$), an experiment was designed in [7] to measure its effectiveness compared to three different boundary based shape representation methods in terms of recall and precision. We compared the retrieval performance of $MBC-TPVAS$ with that of the other 2 more established methods (i.e., Fourier descriptors method $FD$ [8], and grid-based method $GB$ [9]) that were used in some commercial systems [4] and as bases for different comparison studies in [9, 4]. In addition, $MBC-TPVAS$ was compared to a new method based on a new indexing technique (i.e., Delaunay triangulation method $DT$ [13, 12]). The results showed the superiority of $MBC-TPVAS$ in accuracy, storage requirement, and computation cost.

In this paper, we focus on the issues of robustness and stability of these alternative shape representation techniques, and study the behavior of such techniques under “uncertainty” issues and situations. Uncertainty could be a consequence of several factors, for example the assumptions that: perfect data is always presented in the database, or that all shape representation methods represent the objects similarly are not valid. As a consequence of uncertainty, shape representation techniques may sometimes
work well only in certain environments, or their performance may depend crucially on the quality of the technique used to represent shapes. Therefore, this study evaluates the effectiveness and robustness of the four shape retrieval methods in the presence of some uncertainty/discrepancies. Specifically, we focus on three sources that introduce uncertainty: 1) the presence of noise in the database, 2) parameters used to identify a shape and its vertices, 3) differences between shape representation techniques and human perception on how they define similarity.

Our results demonstrate the robustness of MBC-TPVAS under various scenarios of uncertainty compared to the other methods. The results show that MBC-TPVAS is not very sensitive to noise and perturbations on the shape vertices. In the presence of noise, MBC-TPVAS performance was comparable to that of M-DT, while it outperformed both GB and FD methods. However, as the noise value increases, MBC-TPVAS outperformed all the other methods and maintained a precision of 70% and higher (at least 30% higher than other methods). The results also show that MBC-TPVAS is not very sensitive to the corner points detection algorithm (VRA). By using different parameter values for VRA algorithm we reduce the number of vertices of a shape while maintaining its general characteristics. The reduction of the number of vertices reduces the precision of the methods. However, MBC-TPVAS observes less degradation in precision than the other methods (by a margin of 15%). Finally, the human perception experiments showed that human perceptions on shape similarity are appreciably different. In addition, all the methods performed comparably and that good precision could be obtained for recall values $\leq 30$. The similar shapes chosen by the 4 methods was always a subset of the shapes chosen by humans.

The remainder of this paper is organized as follows. Sec. 2, provides some background on related work and describes different types of shape retrieval methods considered in this study. Sec. 3 briefly describes the introduction of noise to the database, vertex reduction algorithm (VRA) used for corner points detection, and describes human perceptions on shape similarity. Sec. 4 reports on the performance results obtained from a set of experiments. Finally, Sec. 5 concludes the paper and provides an overview on our future plans.

2. Alternative boundary-based shape description techniques

This study compares four NIP boundary based methods for shape representation and retrieval: Fourier descriptors method (FD) [8], grid-based method (GB) [9], Delaunay triangulation method (DT) [12], and Touch-point-vertex-angle-sequence (TPVAS) [10].

With the TPVAS method, a shape descriptor (signature) of an object depends on a set of features (e.g., touch points TP, and vertex angle sequence VAS) that are extracted from the objects’ minimum bounding circle MBC. Then, a unique object representation is obtained as the discrete Fourier series expansion of the objects’ shape signature. The DT method is a histogram based approach to classify shape features of objects. For each object, high-curvature points, called corner points, are used as the feature points of the object. Then, a Delaunay triangulation of these feature points is constructed. Subsequently, a feature point histogram is obtained by discretization of the angles produced by this triangulation into a set of bins and counting the number of times each discrete angle occurs in the object. The FD method obtains the object representation in the frequency domain as complex coefficients of the Fourier series expansion of the objects’ shape signature. The shape signature of the object is based on shape radii (or centroidal distances), which computes the distance between points uniformly sampled along the object boundary and its centroid. Finally, a discrete Fourier transform of the shape signature is obtained. The last approach for shape representation is GB. With this method, objects are first normalized for rotation and scale. Then, the object is mapped on a grid of fixed cell size. Subsequently, the grid is scanned and a 1 or 0 is assigned to the cells depending on whether the number of pixels in the cell which are inside the object are greater than or less than a predetermined threshold. Finally, a unique binary number is obtained as shape representation by reading the 1’s and 0’s assigned to the cells from left to right and top to bottom. All the methods use the Euclidean distance between two object’s shape signatures as their difference. Except for the grid-based method, where the difference between two objects is the number of cells in the grid which are covered by one shape and not the other, which is the same as the sum of 1’s in the result of the exclusive OR of the two binary numbers.

For our experiments, boundary features (based on shape) are first extracted from the objects during the population of the database. At query time, the user enters a query into the system and features are extracted from the query object. The query object is compared with the database of objects on the basis of the closeness of the extracted features. The efficiency of the shape retrieval methods was measured in terms of recall and precision. This measures the ability and accuracy of the retrieval methods in retrieving the relevant items (similar shapes) in the database.

3. Stability and robustness of shape representation methods

This section describes three scenarios with uncertainty that affects the effectiveness and robustness of the four
shape retrieval methods. Sec. 3.1 describes the effect of the presence of different noise samples in the database. Subsequently, Sec. 3.2 discusses the impact of the parameters used to identify the corner points of a shape. Finally, Sec. 3.3 studies the difference between shape representation techniques and human perception on defining similarity.

3.1. Gaussian noise

Real data is always accompanied with noise (e.g., by using sensors, scanners, etc.). Hence, when evaluating the efficiency of a shape representation technique, perfect data cannot be expected. In addition, a shape representation technique is expected to be stable and not fail to retrieve a similar shape that is distorted (but not enough to make it different). Therefore, it is necessary to address the issues of robustness/stability of the alternative shape retrieval techniques in the presence of noise.

To evaluate the robustness of the alternative shape retrieval methods, a database of corrupted shapes can be used. Shapes can be corrupted by introducing random Gaussian noise to their vertices. This study considers two different methods for introducing noise to the vertices of a shape. With the first method (Noise-M1), noise is introduced by adding random samples following Gaussian distribution to the boundary points according to the following approach: if the coordinates of a vertex (kth vertex) of a shape are \((x(k), y(k))\) then the coordinates of the corresponding vertex on the noisy shape \((x_n(k), y_n(k))\) are given by \(x(k) + gn\) and \(y_n(k) = y(k) + gn\); where \(gn\) is a sample from the Gaussian distribution \(N(0, 1)\). With the second method (Noise-M2), noise is introduced to the shapes following the approach defined in [14] and used in other comparison studies [3, 9]. In this approach, if the coordinates of a vertex (kth vertex) of a shape are \((x(k), y(k))\) then the coordinates of the corresponding vertex on the noisy shape \((x_n(k), y_n(k))\) are given by \(x_n(k) = x(k) + d_k \times r \times c \times \cos(\theta_k)\) and \(y_n(k) = y(k) + d_k \times r \times c \times \sin(\theta_k)\); where \(d_k\) is the distance of boundary point \(k\) to point \(k + 1\), \(\theta_k\) is the angle from the x-axis to the normal direction of the boundary at point \(k\), \(r\) is a sample from the Gaussian distribution \(N(0, 1)\), and \(c\) is a parameter that controls the amount of distortion. Since our shapes are represented by their boundary vertices (i.e., discrete values), the normal direction of the boundary at point \(k\) is approximated by the direction of the vector passing through point \(k\) and the midpoint of the line connecting points \(k - 1\) and \(k + 1\). There are several ways to generate Gaussian random numbers with a specified mean (\(\mu\)) and standard deviation (\(\sigma\)) (termed \(N(\mu, \sigma^2)\)). This study employs the polar form of the Box-Muller transformation method (please refer to [11] for further details). An example of a shape and how it is corrupted by introducing different types and values of noise to its vertices is shown in Fig. 1.

3.2. Corner points detection

The performance of a shape representation technique should not be crucially dependent on the quality of the edge detection algorithms and explicit distinguishable points (e.g., corners or inflection points). Therefore, it is vital to study the impact of using different polygonal approximations and corner points on the retrieval performance of the alternative shape representation techniques. In Sec. 3.2.1, we propose a new algorithm termed vertex-reduction (VRA) to identify better corner points to identify and represent a shape.

3.2.1. Vertex reduction algorithm

Optimally, a straight line could be represented by its two end points. However, typical edge detection algorithms introduce more than two points to represent a straight line. The extra points make the straight line looks like a jagged line (like saw edge). Therefore, we propose a new algorithm termed vertex-reduction (VRA) to identify better corner points to represent a shape, and at the same time reduce the number of its vertices. VRA strives to reduce the number of vertices required to represent the straight lines in a shapes’ boundary. Towards this end, VRA employs two heuristics to decide when to delete or keep a vertex of a shape boundary. VRA algorithm deletes a boundary point of a shape if it meets the conditions described by both of the following two heuristics:

Heuristic-1 The angle (\(\theta\)) between the vectors \((V_{i-1} \text{ and } V_i)\) connecting vertex \(P_i\) and its two adjacent ver-
3.3. Human perception

In our previous study [7], we assumed that human beings only ignore variations in shapes (i.e., affine transformations such as rotation, scaling, or translation) for recognition and retrieval purposes. Hence, we assumed that each shape is relevant only to itself and to its ten variants (e.g., rotation, or scaling variants). However, since humans are better than computers in extracting semantic information from shapes [13, 12], human perceptions on shape similarity are appreciably different. For example, a human being could identify two shapes from a database to be similar even though they are considered irrelevant by the computer; i.e., their shape signatures derived using a shape representation technique are different. Therefore, in this study we incorporate human perception in the evaluation of the effectiveness of the shape matching methods. To this end, we implemented a Web-based (Java) system with a shape database of 100 fish shapes provided to us by [1, 5]. In this system, five query shapes are selected randomly from the database, then the user has to select the ten most similar shapes in the database per query shape. Please refer to [11] for further details on the system and its user interface.

4. Comparison study

In several applications, an important criterion for testing the efficacy of the shape retrieval methods is that for each query object the relevant items (similar shapes) in the database should be retrieved. Therefore, in our previous study [7], experiments were designed to measure the effectiveness of the similarity retrieval of the four different shape representation methods (i.e., F D, G B, M - DT, and M B C - T P V A S) in terms of recall and precision. In this study, however, we addressed the issues of robustness and stability of the alternative shape representation techniques under various scenarios of “uncertainty” such as: 1) the presence of noise in the database, 2) parameters used to identify a shape and its vertices, and 3) differences between shape representation techniques and human perception on defining similarity. We conducted some experiments and the results are presented in Sec. 4.2.

4.1. Experimental setup

We implemented a prototype shape retrieval system in Java, on a Sun Ultra 60 workstation with 512MB of RAM (please visit http://dimlab.usc.edu for an online demonstration). The shape database of the system consists of 351 fish shapes provided to us by [5]. Please refer to [11] for some examples of shapes inside the database.

As advised in [9, 4, 12, 10], we have selected specific values for the parameters used by the different methods.
For GB we employed a grid cell size of 12×12 pixels and the length of the standardized major axis was fixed at 192 pixels. With FD, we used radius-based signature with 64 uniformly sampled boundary points. For both FD and MBC-TPVAS, we kept 8 low frequency coefficients of the Fourier transform. Finally, for M-DT, we used two largest angle histogram with 18 bins of size 10 degrees.

4.2. Experimental results

Given a query object, the system retrieves relevant shapes from the database, in decreasing order of similarity to the query shapes. First, the shape features of the query object using the four different methods are extracted. Second, the Euclidean distances between the shape signature of the query object and all other objects’ shape signatures are computed. Third, the Euclidean distances are ordered in an ascending order for all the methods. Finally, the accuracy of the retrieval methods was calculated as precision-recall. Towards this end, we identify the ranks of the relevant objects to the query object within the ordered list (according to Euclidean distance or to human order from human perception experiment). Note that the retrieval can be done more efficiently using index structures, but since this is not the focus of this study we assumed the sequential scan of the database for all the methods.

4.2.1. Impact of gaussian noise

The performance of our proposed method (MBC-TPVAS) when Gaussian noise is added to the shape vertices was compared with FD, GB, and M-DT methods. Noise was added to all the vertices of the query shapes and the shapes in the database. We conducted experiments by introducing noise according to Noise-M1, and Noise-M2. For Noise-M2 we set c to 0.1 and 0.5. Noise-M2 introduces more noise to the shapes than Noise-M1. In addition, by increasing the value of c the amount of the noise added by Noise-M2 increases. Example of a shape and how it is corrupted by introducing different types and values of noise to its vertices was shown in Fig. 1. Note that as the amount of noise increases, some boundary edges of the shape cross over each other. This cross over leads to different shape signatures of the same object.

The recall and precision results for 25 randomly chosen query shapes on a database of 351 objects are averaged and shown in Fig.4, where the X-axis is the recall and the Y-axis is the precision. Note that N1, N2, and N3 refer to Noise-M1, Noise-M2 with c = 0.1, and Noise-M2 with c = 0.5, respectively. Fig. 4(a) shows the performance of the four methods when no noise is introduced (see [7]). We can conclude that MBC-TPVAS performance is comparable to that of M-DT, while they outperformed both GB and FD methods. Fig. 4(b) shows that our proposed method (MBC-TPVAS) outperformed all the other methods when noise is added. In Figs. 4(a) and (b), the performance of GB method was shown to be lower than that of M-DT method. However, from Figs. 4(c) and (d), it is observed that GB performs better than M-DT when larger noise is added to the shape vertices. The larger noise is due to the use of Noise-M2 instead of Noise-M1. The fall in the performance of M-DT is due to the fact that the Delaunay triangulation algorithm is highly dependent on the corner points, and that the corner points are more sensitive to small changes and noise. However, GB is less dependent on the corner points and hence less sensitive to small changes and noise. In general, Fig. 4 shows that the accuracy of our proposed method (MBC-TPVAS) was the best as the amount of noise increases. This is due to the fact that the angle sequences (AS) and touch points (TP) used by MBC-TPVAS are less sensitive and more robust to small changes and noise once transferred to the frequency domain using Fourier transformation.

Since in [7] we showed that the performance of MBC-TPVAS was comparable to that of M-DT when no noise was introduced to the data, we decided to compare the performance of only those two methods as the amount of noise introduced to the data increases. In Fig. 5 we demonstrate the recall and precision results for the M-DT and MBC-TPVAS methods, respectively; with no noise and when noise is added. It is clear from Fig. 5 that as the amount of introduced noise increases, the performance of the methods decreases. However, the accuracy of our proposed method (MBC-TPVAS) was better than that of M-DT as the amount of noise increased. Fig. 5(b) shows that MBC-TPVAS maintained a precision of 70% and higher while the precision of M-DT (see Fig. 5(a)) dropped to lower than 40%. Hence, this shows that MBC-TPVAS is not very sensitive to noise and perturbations on the shape vertices. This is due to the fact that the angle sequences (AS) and touch points (TP) used by MBC-TPVAS capture the general characteristics of a shape which are less sensitive and more robust to small changes and noise once transferred to the frequency domain using Fourier transformation. The minor fall in the performance of the MBC-TPVAS method is because of the addition of Noise-M2 to the shape vertices that may cause some crossover on the boundary edges of the shapes (see [2]), and hence changes the angle sequences when computing the shape signatures. The results also show that M-DT is very sensitive to noise and perturbations on the shape vertices. This is due to the fact that the Delaunay triangulation algorithm is highly dependent on the corner points, and that the corner points are more sensitive to small changes and noise.
4.2.2. Impact of corner points detection

In a previous study [7], unlike what was expected, the results obtained for our modified DT (M-DT) were better than what was reported in [12]. This is due to the fact that in their approach, they identify corner points as the high-curvature points. Instead, in our experiments, we used VRA algorithm to identify the corner points. Therefore, in this study, we examined the impact of VRA parameters on the accuracy of the corner points detection. Note that the GB method depends on the general characteristics of a shape and is not directly dependent on any explicitly distinguished points such as corners points. Therefore, VRA parameters would have no major impact on its performance. On the other hand, FD, M-DT, and MBC-TPVAS methods depend on the corner points to create their shape signatures. However, since our previous experiments have shown that FD always performed worse than the other methods, we decided to compare the impact of the corner points detection algorithm only on the performance of the M-DT and MBC-TPVAS methods. Towards this end, we conducted two experiments.

First, we varied the value of Threshold-1 from 120 to 170 with an increment of 10, and for each value of Threshold-1 we varied the value of Threshold-2 from 0.1 to 0.6 with an increment of 0.1. Fig. 6(a) depicts the results where the X-axis is the values of Threshold-1, the Y-axis is the values of Threshold-2 and the Z-axis is the average percentage of reduction in the number of vertices over 250 shapes. As shown in Fig. 6(a), the percentage
of reduction in the number of vertices of a shape increases as we increase the value of Threshold-1 and decrease the value of Threshold-2. The best reduction percentage was achieved with values of 170 and 0.1 for Threshold-1 and Threshold-2, respectively.

Second, we evaluated the performance of M-DT and MBC-TPVAS by fixing the best value of Threshold-2 at 0.1 and varying the value of Threshold-1 from 130 to 170 with an increment of 20. Fig. 6(b) depicts the results where the X-axis is the recall values, and the Y-axis is the precision values. From Fig. 6(b) we can see that in general the precision is better for smaller values of Threshold-1. Comparing this result with the results obtained from Fig. 6(a), we can conclude that as the value of Threshold-1 increases (while fixing the value of Threshold-2) the percentage of reduction in the number of vertices of a shape decreases. However, the accuracy (i.e., precision) also reduces. In addition, Fig. 6(b) demonstrates that the reduction in precision of MBC-TPVAS is much lower than that of M-DT. The largest reduction of precision for MBC-TPVAS was 5% (occurred at 100% recall), while the largest reduction for M-DT was 20%. Hence, we can conclude that DT performance highly depends on the technique used to find the representative vertices of the objects. This is due to the fact that the DT uses Delaunay triangulation algorithm that is highly sensitive and dependent on the corner points. Using different parameter values for VRA algorithm leads to different sets of identified corner points, hence, different shape signatures. Fig. 7 shows an example of applying VRA to a fish shape using three different values for Threshold-1 (130, 150, and 170) while fixing the value of Threshold-2 at 0.1. The reduction in the number of vertices of the shape for the three Threshold-1 values were 28.15%, 52.91%, and 69.90%, respectively. Although the reduction percentage was increasing as the value of Threshold-1 increased, the shape maintained its general characteristics except for the case where the value of Threshold-1 was 130 (see Fig. 7(d)). Therefore, when applying VRA algorithm, the choice of values for the parameters is a comprise between the processing time, the required space to store the shapes, and the precision reduction the user can tolerate.

4.2.3. Impact of human perception

To incorporate human perception in the evaluation of the effectiveness of the shape matching methods, we implemented a Web-based (Java) system with a shape database of 100 fish shapes. 25 users participated in the experiment, who selected the 10 most similar shapes to the 5 query shapes out of the 100 shapes in the database. To participate in the experiment, please visit http://dimlab.usc.edu. In order to compute the precision for different values of recall, the relevant shapes for each query should be identified.

Towards this end, the ranking for the 100 shapes with reference to the 5 query shapes were computed, where for each query shape the ranking of a shape in the database was computed as the total number of users who selected the shape as similar to the query shape. Then the top 10 shapes with the largest ranks were considered to be perceptually similar to the respective query shape. The results obtained are presented in Table 1.

For each of the 5 query objects, we obtained the positions for the 10 similar shapes (according to human perception) in the ordered response set (according to Euclidean distance) using all methods (i.e., FD, GB, DT, and MBC-TPVAS). The results obtained are reported in Table 2. Using the results in Table 2, we computed the precision as a function of recall averaged over 5 queries for all methods as shown in Fig. 8. Fig. 8 shows that all the methods observed low precision and did not perform well. Consequently, good precision were obtained only for recall values ≤ 30. The results also show that human perceptions on shape similarity are appreciably different. However, the set of similar shapes chosen by the 4 methods was always a subset of the 10 shapes chosen by humans (see example in Fig. 9). In Fig. 9, we show an example of shapes similar to a query shape using both our method (MBC-TPVAS) and human ranking. We can conclude that the first 2–3 objects returned from the system is almost always the same as the ones picked by humans. We noticed that humans con-
Consider shapes to be similar if their general description is the same and people do not pay attention to some specific details. For example, for the query object shown in Fig. 9, humans considered all oval shapes as matches. On the other hand, the four methods consider shapes to be similar if they have the same general description and in addition they have some common specific details. Hence, for the same query example in Fig. 9, the methods considered oval shapes with small fins as matches.

5. Conclusion and Future Work

In this study we conducted experiments to examine the robustness and stability of the alternative shape representation techniques with regard to discrepancies in representing shapes and under different scenarios of uncertainty. Our results demonstrate the robustness of MBC-TPVAS compared to the other methods. The results show that MBC-TPVAS is not very sensitive to noise and perturbations on the shape vertices, and that it maintained a precision of 70% and higher (at least 30% higher than other methods). This is due to the fact that the angle sequences (AS) and touch points (TP) used by MBC-TPVAS are less sensitive and more robust to small changes and noise once transferred to the frequency domain using Fourier transformation. While the fall in the performance of M-DT is due to the fact that the Delaunay triangulation algorithm is highly dependent on the corner points, and that the corner points are more sensitive to small changes and noise. However, GB is less dependent on the corner points and hence less sensitive to small changes and noise. The results also show that MBC-TPVAS was the least sensitive technique to the corner points detection algorithm. In addition, it had the lowest reduction in precision than the other methods using VRA. On the other hand, DT's performance highly depended on
the parameters used to find the representative vertices of the objects. This is due to the fact that the \( DT \) uses Delaunay triangulation algorithm that is highly sensitive and dependent on the corner points. Consequently, using different corner points leads to significantly different triangulation, hence, different shape signatures. Finally, the human perception experiments showed that human perceptions on shape similarity are appreciably different, and that all the methods performed badly. However, good precision could be obtained for recall values \( \leq 30 \).

We noticed that humans consider shapes to be similar if their general descriptions are the same or if they have similar parts, and people do not pay attention to some specific details. Therefore, we intend to extend this work to identify better features to be used for partial similarity match queries. In addition, we want to extend this work to support three-dimensional objects. Our preliminary investigations show that analogous to \( MBC \) features for 2D objects, we can extract features from 3-D objects by using their minimum bounding spheres.

6. Acknowledgment

The authors would like to thank Xiaoming Sun for his help in implementing our shape based retrieval system. Special thanks also go to F. Mokhtariyan, X. Ding, and S. Abbasi for providing us with the fish shapes data set (available at ftp://ftp.ee.surrey.ac.uk/pub/vision/misc/fish.contours.tar.Z). We would also like to thank Ammar Safar for helping in the implementation of the human perception experiment.

References


