A Shape Matching Approach to Content-Based Image Retrieval

Jong-Seung Park, Hwang-Seok Oh and Duk-Ho Chang

Visual Information Team, ETRI,
161 Kajong-Dong, Yusong-Gu, Taejon, 305-350, South Korea

ABSTRACT

In this paper, to improve the retrieval effectiveness of a content-based image retrieval system, a shape-based object matching method is presented. A new skeleton structure is proposed as a shape representation. The skeleton structure represents an object in a hierarchical manner such that high-level nodes describe parts of coarse trunk of the object and low-level nodes describe fine details. Each low-level node refines the shape of the parent node. Most of the noise disturbances are limited to the bottom levels. The effect of boundary noise is reduced by decreasing weights on the bottom levels. To compute the similarity of two skeleton structures, we consider the best match of spine nodes, nodes in level one of the structure. Both moment invariants and Fourier descriptors are used to compute the similarities of sub-regions. We evaluated the retrieval accuracy and compared the result to that of other shape similarity measures. Experimental results showed that our system gives prominent accuracy in retrieval.

Keywords: Shape matching, Shape-based image retrieval, Skeleton structure, Fourier descriptors, Moment invariants, Object recognition

1. INTRODUCTION

The problem of content-based image retrieval is to obtain a list of images from a huge database which are most similar to the query image. Two most important issues of a content-based image retrieval system is feature extraction and object matching. Feature extraction is for automatic characterization of image content. Based on the similarity measures of image content, relevant images are retrieved.

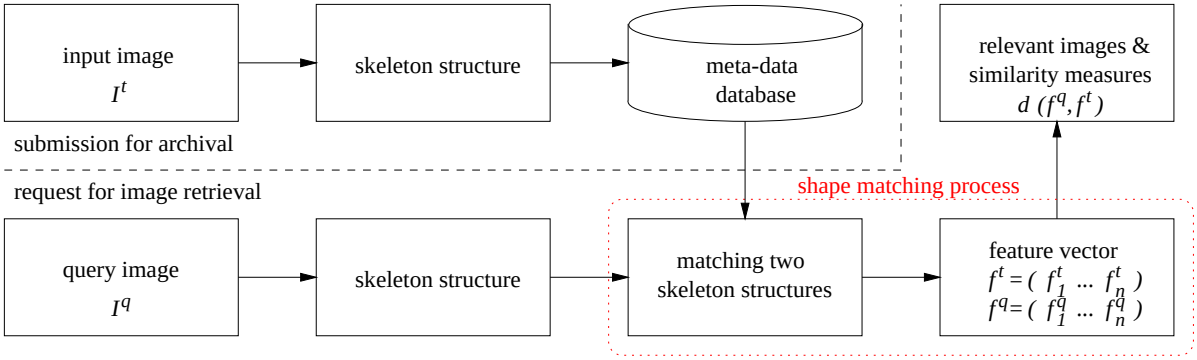


Figure 1. The system flow diagram of our image retrieval system.

The inability of image understanding technology has prohibited the intelligent image retrieval. Most of the previous image retrieval systems were based on the image feature analysis such as colors, textures, and regions. Such features do not represent shape properties of query object quite well, hence irrelevant images are frequently retrieved. Recently, there have been works to handle shape features effectively.[1–3]

In this paper, a shape-based image retrieval method is presented. We assume that images are already segmented. Steps of our algorithm in processing a query image is as following:

Further author information: (Send correspondence to Jong-Seung Park)
Jong-Seung Park: E-mail: park@etri.re.kr WWW: http://vrcenter.etri.re.kr/~parkjs

Step 1: Skeletonize the segmented image.

Step 2: Construct a skeleton structure from the skeleton image.

Step 3: Compute similarities of the skeleton structure of the query image and the skeleton structures that are already saved in a local database.

Image archival process constructs skeleton structures using Step 1–2, and saves them to a local database. Image retrieval process constructs a skeleton structure for the query image and compares the structure to the structures stored in the local database. The overall system flow diagram is shown in Fig. 1.

In the following, in Section 2, our proposed skeleton structure is introduced. Section 3 describes the matching scheme of skeleton structures to compute similarities of the structures. Then experimental results are presented in Section 4. We conclude the paper with discussion and future works in Section 5.

2. SKELETON STRUCTURE

We propose a skeleton structure for the shape representation in the shape-based image retrieval. For each region, a skeleton image is obtained using a distance transform. Using the skeleton image, a skeleton structure is computed. A skeleton structure represents an object in a hierarchical manner such that high-level nodes describe parts of the coarse trunk of the object and low-level nodes describe fine details. Each low-level node refines the shape of the parent node.

Skeleton representation is a natural way of shape description especially for deformable objects such as human, animals, fish and insects. Beside its naturalness, the shape can be reconstructed from the skeleton representation by taking an inverse skeleton transform. The major drawback of skeleton representation is that it is sensitive to noise. For example, approaches of medial axis may cause spurious branches and shape distortions for the jagged boundaries.

In our proposed skeleton structure, the sensitivity to boundary noise is prohibited by employing a hierarchical tree representation. From skeletal primitives, a tree structure is constructed. A set of nodes in a level represents the object in a certain coarseness. The set of nodes in level one, the highest level, corresponds to the description of the coarsest trunk shape to the original shape. The set of all nodes in the structure represents the original shape. Hence most of the noise disturbances are limited to the bottom levels. The effects of noise is reduced by decreasing weights on the bottom levels.

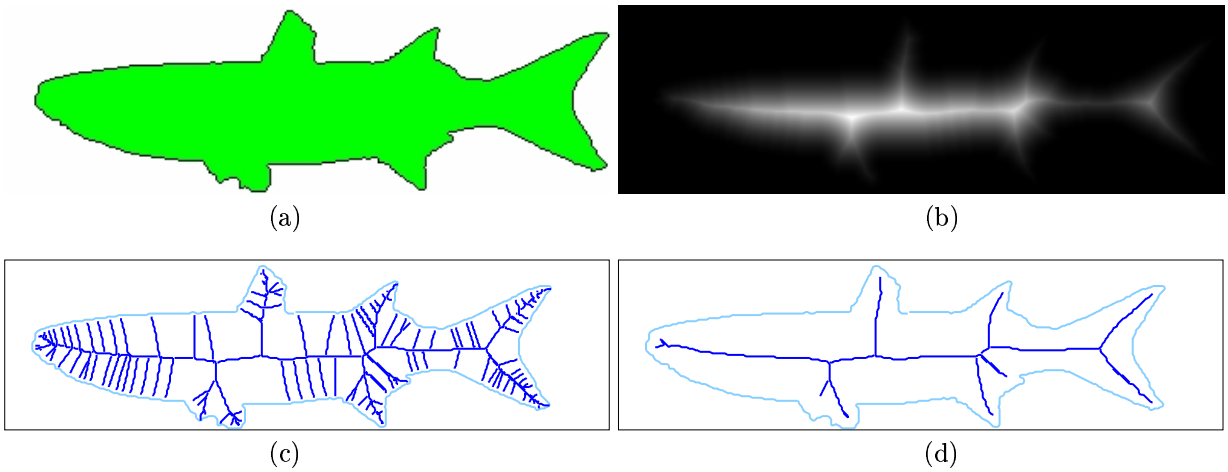


Figure 2. (a) Original object; (b) Distance image; (c) Skeleton image with minimum distance 3 pixels; (d) Skeleton image with minimum distance 10 pixels;

First, we extract skeletal primitives from each region. The skeleton $SK(X)$ of an object shape X is defined as the locus of the centers of the maximal disks that are contained within the shape. To extract the skeleton, numerous methods has been proposed, e.g., methods using a distance transform,[4] methods using morphological operator,[5] methods using Voronoi graph.[6] In this paper, the skeleton is computed using distance transform. To each point

in the region, the distance transform assigns a number that is the distance between the point and its closest border pixel. We use a fast two-pass algorithm[5] to compute the skeleton image.

Figure 2 shows an example of skeleton construction. To avoid too many skeletal branches, we remove the branches having distances less than the given threshold. Figure 2 (c,d) shows the effect of thresholding.

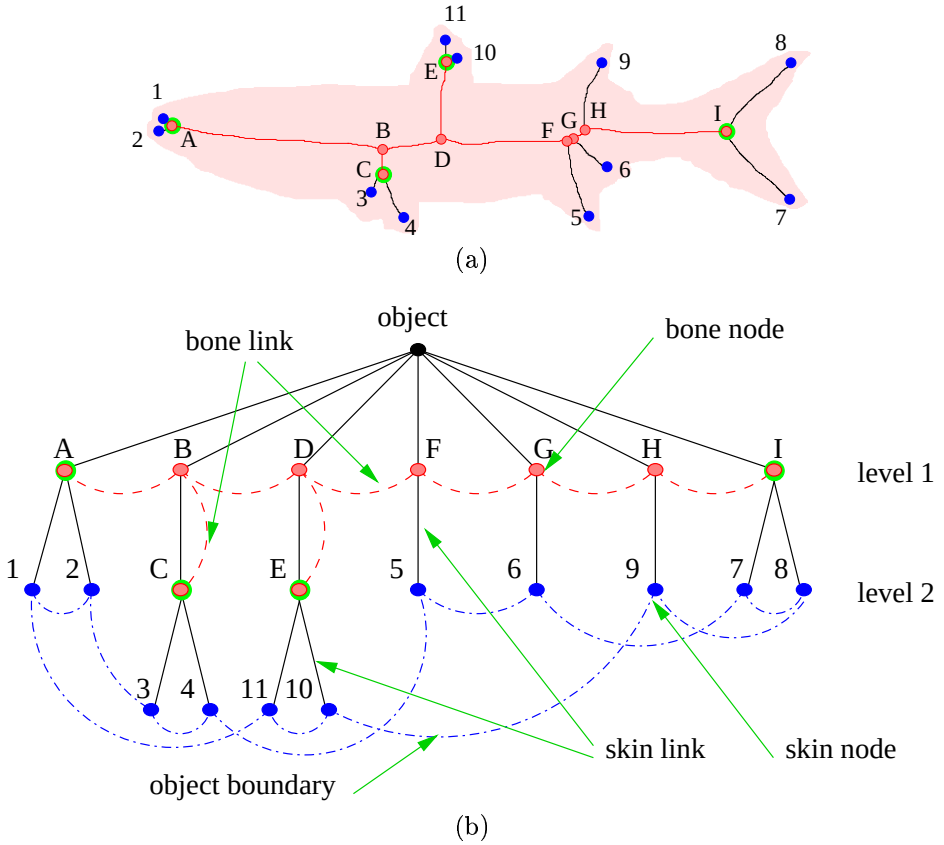


Figure 3. An example of skeleton structure construction. (a) Skeleton nodes and links; (b) Skeleton structure;

Using the extracted skeletal primitives, a tree structure, called *skeleton structure*, is constructed. The end points and junction points of skeletal primitives are considered as nodes and the skeletal segments are considered as links. Nodes are distinguished into two types: bone node and skin node. *Skin node* is a node that has only one link, i.e., the node that is connected to only one skeletal primitive. Generally, nodes near boundary are skin nodes. A node which is not a skin node is called *bone node*. *Skin link* is a link that contains a skin node. A link which is not a skin link is called *bone link*. *Terminal bone node* is a bone node which is connected to only one bone link.

To construct the skeleton structure, two terminal bone nodes are selected from the set of bone nodes. Let $NB = \{NB_1, \ldots, NB_n\}$ be set of bone nodes. If there are more than two terminal bone nodes in NB, two true terminal bone nodes are picked using a simple criterion. Among all possible pairs of terminal bone nodes, the pair (NB_a,NB_b) is selected as true terminal bone nodes if the total area of influence zones of nodes on the path from NB_a to NB_b is maximum. Other terminal bone nodes that are not true bone nodes are considered as virtual terminal bone nodes.

Let $NB' = \{NB_a, \ldots, NB_b\}$ be set of bone nodes on the path between the two true terminal bone nodes N_a and N_b. Then, the set NB' constitutes the spine level (level one) of the tree. All the remaining nodes are attached to the tree recursively. Figure 3 shows an example of tree construction. In Fig. 3, the set of terminal bone nodes is $\{A, C, E, I\}$. Among the terminal bone nodes, A and I are picked as true terminal bone nodes. A bone link connects two adjacent bone nodes and a skin link connects a skin node to the adjacent bone node. The construction algorithm is described in Algorithm 1.

Algorithm 1: skeleton structure construction algorithm.
Input: skeleton image
Output: skeleton structure
compute NS, the set of skin nodes, and LS, the set of skin links from the skeleton image
compute NB, the set of bone nodes, and LB, the set of bone links from the skeleton image
determine two true terminal bone nodes NB_a and NB_b
determine the spine bone nodes NB' and construct a 1-level tree with NB'
$l = 1$;
repeat
  foreach $NB_i \in$ level l
    if there are some nodes in $NS \cup (NB - NB')$ which are adjacent to NB_i
      insert the nodes as childs of NB_i
  $l = l + 1$;
until all nodes in $NS \cup (NB - NB')$ are in the tree
end

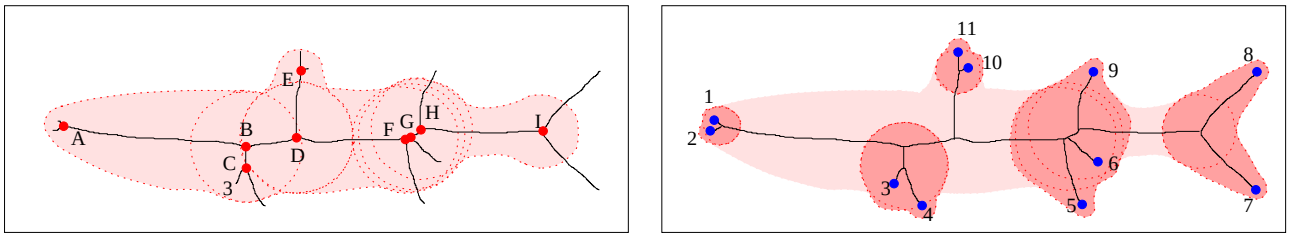


Figure 4. (a) Influence zones of bone links; (b) Influence zones of skin links;

Often, there are too many extra nodes due to the discrete property of region boundaries or noisy jagged boundaries. To achieve fast computation without losing object information, we filter out unnecessary bone nodes and links before the structure construction.

Each link defines an influence zone which is a set of pixels whose nearest skeletal pixel is on the link. Figure 4 shows influence zones of bone nodes and skin nodes. The object region is the sum of all influence zones.

The slight deformations on the boundary may split a node into several nodes and the structure would contains too many nodes. To prevent it, we remove and merge nodes and links according to several rules:

- If the sum of gradient magnitudes of distance image along the skin link is too small, remove the skin link and the skin node connected to the link. In this case, the branch was created due to noise in a boundary, and most of the influence zone of such a link is already contained in other influence zones.
- If two skin links have nearly same influence zones, remove the skin links and skin nodes on them. The redundancy of influence zone is prohibited by removing such nodes and links.
- If two bone nodes are too close, merge two nodes by extending one node to the other. This rule simplifies the skeleton structure of an object.

As a result, a skeleton structure with a few nodes and links is generated from the skeleton image. The structure is enough to represent the shape property.

3. SHAPE MATCHING

To compute the similarity of two skeleton structures, we consider the best match of spine nodes, nodes in level one of the structure. First, all the possible hypotheses of shape matching are generated. Each time when a hypothesis is generated, the similarity measure for the hypothesis is computed. If the hypothesis gives the minimum dissimilarity up to now, it is stored for the further references. Finally, the minimum dissimilarity and the corresponding hypothesis represent the matching of two skeleton structures.

We describe our similarity measure for a given hypothesis. After that, we describe the overall structure matching scheme. Let $NS(O_a) =< l_1, \ldots, l_m >$ be the spine links, the bone links in level one, of object O_a and let $NS(O_b) =< l'_1, \ldots, l'_m >$ be the spine links of object O_b. A hypothesis of matching of two sets defines a partition of $NS(O_a)$ and a partition of $NS(O_b)$ where two partitions have same number of subsets. Let $\{NS(O_a)^1, \ldots, NS(O_a)^k\}$ and $\{NS(O_b)^1, \ldots, NS(O_b)^k\}$ be such partitions. The hypothesis assumes that $NS(O_a)^i$ matches to $NS(O_b)^i$ for each $i = 1, \ldots, k$. The dissimilarity of $NS(O_a)$ and $NS(O_b)$ is defined as

$$\mathrm{dist}(NS(O_a), NS(O_b)) = \sum_{i=1}^{k} \mathrm{d}(NS(O_a)^i, NS(O_b)^i) \ . \tag{1}$$

Each set of links $NS(O_j)^i$, $j = a$ or b, defines a region which is the union of influence zones of links in $NS(O_j)^i$.

The proper similarity measure for the sub-regions should be provided. There have been numerous works on measures for 2-D objects. Among them, moment invariants and Fourier descriptors are considered as two most representative features in 2-D shape matching. Both of the measures are invariant to translation, scale change, and rotation in 2-D space. Note that moment invariants are the region-based measure and Fourier descriptors are the boundary-based measure. In the work of Mehtre et. al,[7] they compared the retrieval efficiency of several methods: reduced chain code, Fourier descriptors, moment invariants, Zenike moments, and Pseudo-Zenike moments. In their experiments, the measure using both Fourier descriptors and moment invariants gave the best average retrieval efficiency. They thought this could be because the human perceptual mechanism uses both these aspects of shape in order to compute similarity. We followed the idea in computing the similarity of two regions. We describe two features, moment invariants and Fourier descriptors, that are used in our method as similarity measures for sub-regions.

3.1. 2-D Feature Selection

Moment invariants are useful in 2-D object recognition. Moments are defined on a continuous image intensity function. For a discrete binary image, a simple approximation is possible using summation operation. Let f be a binary digital image matrix with dimension $M \times N$, and let $S = \{(x, y) | f(x, y) = 1\}$ represent a 2-D shape. The *moment* of order (p, q) of shape S is given by

$$m_{pq}(S) = \sum_{(x,y)\in S} x^p y^q \ .$$

The *central moment* of order (p, q) of shape S is given by

$$\mu_{pq}(S) = \sum_{(x,y)\in S} (x - \bar{x})^p (y - \bar{y})^q$$

where $(\bar{x}, \bar{y})$ is the center of gravity:

$$\bar{x} = \frac{m_{10}(S)}{m_{00}(S)} \ , \ \bar{y} = \frac{m_{01}(S)}{m_{00}(S)} \ .$$

From the central moments, the *normalized central moments*, denoted by η_{pq}, are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \text{ where } \gamma = \frac{p+q}{2} + 1 \text{ for } p + q = 2, 3, \ldots \ .$$

Moment invariants are functions of moments that are invariant under certain transformations. From the second- and third-order normalized central moments, a set of seven invariant moments, which is invariant to translation, scale change and rotation, has been derived[8]:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + 3(\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\
\phi_6 &= (\eta_{20} - \eta_{02}) \left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]
\end{aligned}$$

We compute the feature vector $f_M = (\phi_1, \ldots, \phi_7)$ for each region to obtain the similarity measure. Hu[8] has proved the invariance properties of the seven moments for the case of continuous functions. Additionally, $\phi_1, \ldots, \phi_6$ are reflection invariant, and $|\phi_7|$ is reflection invariant but its sign changes under reflection. Also ϕ_7 is skew invariant. In discrete case, the set of seven moment invariants is still invariant under image translation although the moments are computed discretely. But the invariants are expected not to be strictly invariant under rotation and scale changes due to sampling, digitizing, and quantizing of the continuous image for digital computation.[9]

Moment invariants are not sufficient for distinguishing all shapes, they can be very sensitive to noise, and their values drastically change with occlusion.

Fourier descriptors are 2-D invariant features available from boundary points. Suppose that the boundary of a particular shape has N pixels numbered from 0 to $N-1$ and the contour is described as two parametric equations:

$$x(k) = x_k, \quad y(k) = y_k, \quad k = 0, \ldots, N-1 \ .$$

By considering the equations in the complex plane, the direct parametric representation $z(t)$ is possible:

$$z(t) = x(t) + jy(t) \ .$$

The Fourier descriptors $Z(k)$ of the curve is the discrete Fourier transform coefficients of the complex valued curve $z(t)$:

$$Z(t) = \frac{1}{N} \sum_{k=0}^{N-1} z(k) \exp\left(\frac{-j2\pi kt}{N}\right) \ .$$

A simple normalization of $Z(t)$ makes the Fourier descriptors invariant to the starting point of sampling, rotation, scaling and translation. Each coefficient of a Fourier descriptor has two components, amplitude and phase. By using only the amplitude component, we achieve rotation invariance as well as the invariance to the starting point. By dividing all amplitudes by the amplitude of the first non-zero frequency coefficient, we achieve the scale invariance. Since only the DC coefficient is dependent on the position of shape, it is discarded to achieve the translation invariance. We compute the m-dimensional feature vector f_F from m Fourier descriptors $Z(-m/2), \ldots, Z(-1), Z(2), \ldots, Z(m/2+1)$ by dividing the magnitudes by $|Z(1)|$. In our system, we choose $m = 16 (= 2^4)$ so that the transformation can be conducted efficiently using FFT.

Both moment invariants and Fourier descriptors have invariant properties to rotation, translation, and scale changes. Figure 5 shows the invariant properties. Both of the two features maintained good invariant properties, less than 0.08 in Euclidean distance, during the transformation.

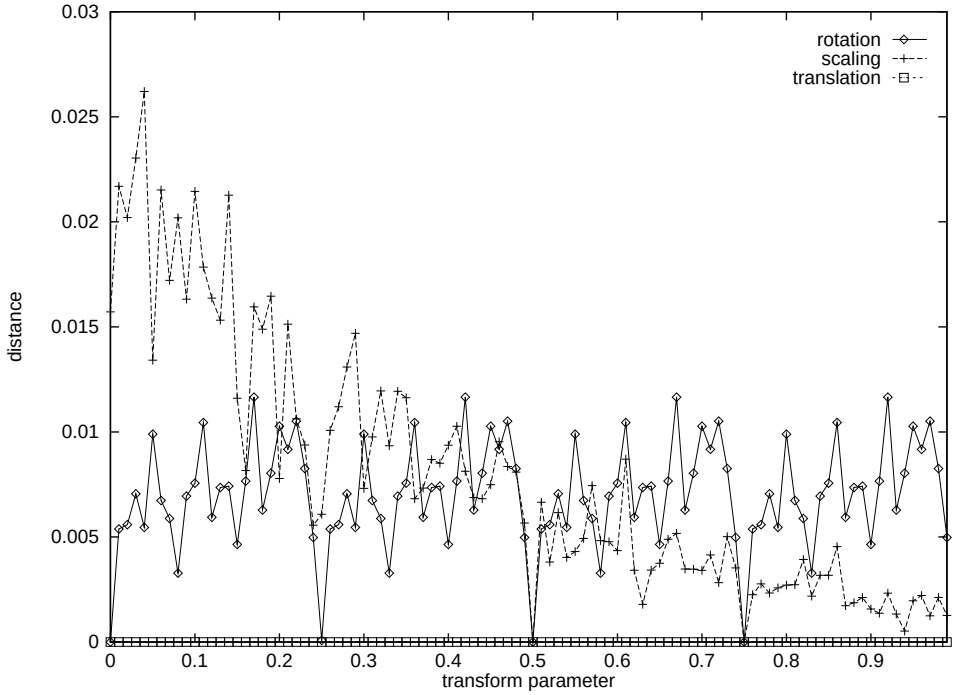
The similarity distance between two feature vectors is computed as the Euclidean distance. The distances of two kinds of feature vectors, moment invariants, $f_M = (\phi_1, \phi_2, \ldots, \phi_7)$, an Fourier descriptors, $f_F = (FD_1, FD_2, \ldots, FD_{16})$, are measured separately. Let f_M and f_F be feature vectors of a region R_a and f'_M and f'_F be feature vectors of another region R_b. Then the distance of moment invariants d_M and the distance of Fourier descriptors d_F are computed as following :

$$d_M(R_a, R_b) = d_E(f_M(R_a), f'_M(R_b)) \text{ and } d_F(R_a, R_b) = d_E(f_F(R_a), f'_F(R_b))$$

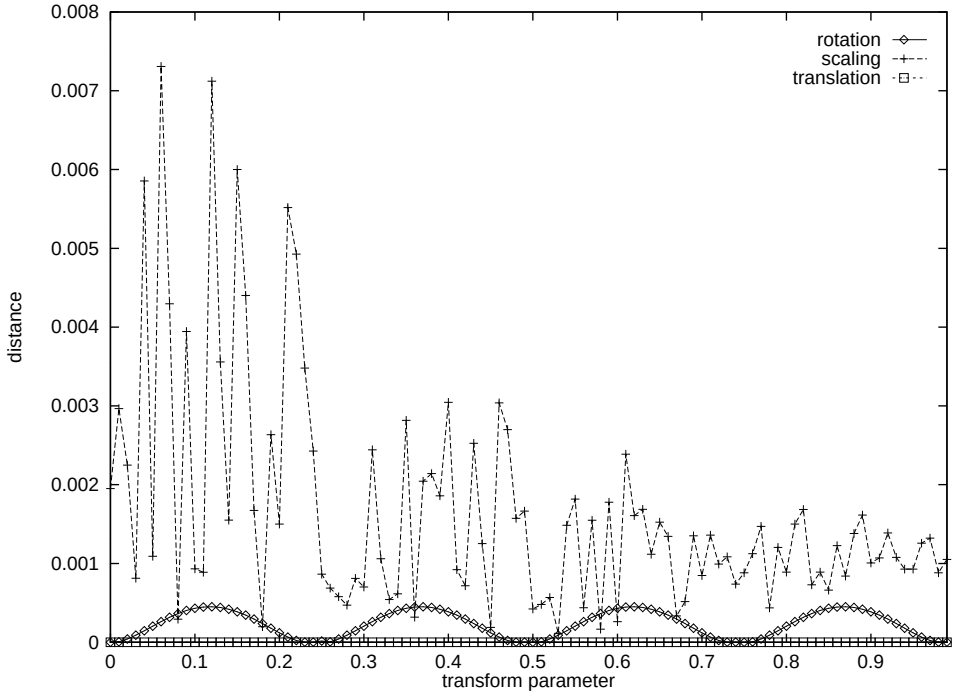
where d_E is the Euclidean distance of two vectors.

Note that some values of feature vectors are quite small or vary a lot. So they need to be normalized for comparison. For the moment invariants, all distances d_M are divided by maximum distance. Similarity for the distances d_F of Fourier descriptors. Let $\tilde{d}_M$ and $\tilde{d}_F$ be the normalized vectors of d_M and d_F, respectively. The distance of two regions is defined as

$$d = \frac{1}{2}\left(\tilde{d}_M + \tilde{d}_F\right) \ . \tag{2}$$



(a)



(b)

Figure 5. (a) Invariant property of Fourier descriptors; (b) Invariant property of moment invariants; Source object is an arrow shape image of size 337 × 145. The transform parameter a normalized scalar (from 0 to 1) for the translation factor, the rotation factor, and the scaling factor. The translation factor ranges from -400 to 400 in pixel units with step size 8, the rotation factor ranges from 0 to 360 in degree with step size 3.6, and the scaling factor ranges from -2 to 2 with step size 0.04 The distance is measured with the Euclidean metric.

To match two skeleton structures, we consider the two partitions, $\{NS(O_a)^1, \ldots, NS(O_a)^k\}$ and $\{NS(O_b)^1, \ldots, NS(O_b)^k\}$, which are obtained from a hypothesis. We compute the dissimilarity of each pair $(NS(O_a)^i, NS(O_b)^i)$ of subsets in the two partitions. The dissimilarity is computed using Eq. 3, instead of Eq. 2.

$$d = \frac{1}{2}\left(w_M \tilde{d}_M + w_F \tilde{d}_F\right) \tag{3}$$

where w_M and w_F are the weights of the vectors $\tilde{d}_M$ and $\tilde{d}_F$, respectively. The weights w_M and w_F for the pair $(NS(O_a)^i, NS(O_b)^i)$ are determined proportional to the region area and the number of boundary pixels of the region, respectively. They are given by the following equations:

$$w_M(NS(O_a)^i, NS(O_b)^i) = \left[\text{area}(NS(O_a)^i)/\text{area}(O_a) + \text{area}(NS(O_b)^i)/\text{area}(O_b)\right]/2.0$$
$$w_F(NS(O_a)^i, NS(O_b)^i) = \left[\text{peri}(NS(O_a)^i)/\text{peri}(O_a) + \text{peri}(NS(O_b)^i)/\text{peri}(O_b)\right]/2.0$$

where function area and peri return the area and perimeter, respectively, of the given region in pixel units. The dissimilarity of two skeleton structures is computed using Eq. 1 where the distance function of two regions is given by Eq. 3.

3.2. Similarity of Skeleton Structures

Moment invariants and Fourier descriptors are useful measures for 2-D shape matching. Unfortunately, in the case of articulated objects such as human or animals, the shape transformation due to the motion of articulations makes the shape matching to original shape fail and the system may regard them as different objects. In our proposed skeleton structure method, a partition of the spine level nodes of the given query object are compared to a partition of the spine level nodes of the target object. Hence this method overcomes the shape deformation of animated object.

To compute the dissimilarity of two objects, A and B, all possible hypotheses are generated. A hypothesis is a combination of the match sequence. For each hypothesis, all weighted sub-region dissimilarities are computed using Eq. 3. The sum of the dissimilarities is assigned to the dissimilarity of the hypothesis. Finally, the best matched hypothesis, the hypothesis having the minimum value of the dissimilarities among all the hypotheses, is chosen. The minimum dissimilarity is regarded as the dissimilarity of object A and object B.

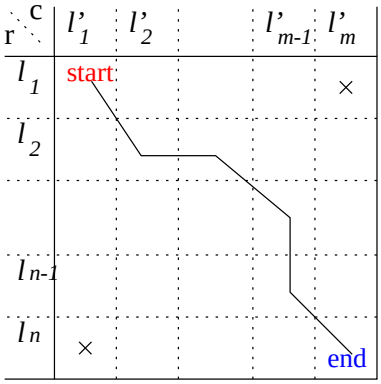


Figure 6. An example of hypothesis. If the path passes the square in the i'th row and in the j'th column, then the the hypothesis assumes that link l_i matches to link l'_j.

To compare two objects O_a and O_b, we generate all possible hypothesis and choose the best one. From the two sets of spine links, $NS(O_a) =< l_1, \ldots, l_m >$ and $NS(O_b) =< l'_1, \ldots, l'_m >$, two partitions, $\{NS(O_a)^1, \ldots, NS(O_a)^k\}$ and $\{NS(O_b)^1, \ldots, NS(O_b)^k\}$, are generated with a positive integer k. The similarity of two partitions is computed using Eq. (1) where the region distance function is given by Eq. (2).

The problem of hypothesis generation is equivalent to finding paths from `start` to `end` as shown in the Fig. 6. The path length, the number of squares that should pass from the starting square to the ending square, is from $\max(n, m)$ to $\max(n, m) + (\min(n, m) - 2)$. If $\min(n, m)$ is one, then there is a unique path.

Algorithm 2: The similarity computation algorithm for two skeleton structures.
Input: two skeleton structures
Output: similarity of two structures and their match
stack = $< (1,1) >$
while stack is not empty
  $path$ = pop(stack);
  (r, c) = last_pt($path$);
  if $r = n$ and $c = m$
    compute *similarity* of two structures with $path$;
    if *similarity* is max until now
      save *similarity* and $path$ for later output;
  else
    (pr, pc) = 2nd_last_pt($path$); /* $(pr, pc) = (-1, -1)$ if path length is one */
    if $r \neq pr$
      push(concat($path$, $< r+1, c >$));
    if $c \neq pc$
      push(concat($path$, $< r, c+1 >$));
    push(concat($path$, $< r+1, c+1 >$));
output minimum dissimilarity and corresponding $path$
end

The overall algorithm to compute the similarity of two skeleton structures is described in Algorithm 2. In the algorithm, the grid shown in Fig. 6 is considered as a matrix with n rows and m columns corresponding to the set of spine links, $\{l_1 \ldots l_n\}$, of an object and the set of spine links, $\{l'_1 \ldots l'_m\}$, of the other object. The starting square is indexed by $(1, 1)$. When a path reached to $(1, c)$ or $(r, 1)$, we ignore the path since there are some parts that are not matched.

4. EXPERIMENTAL RESULTS

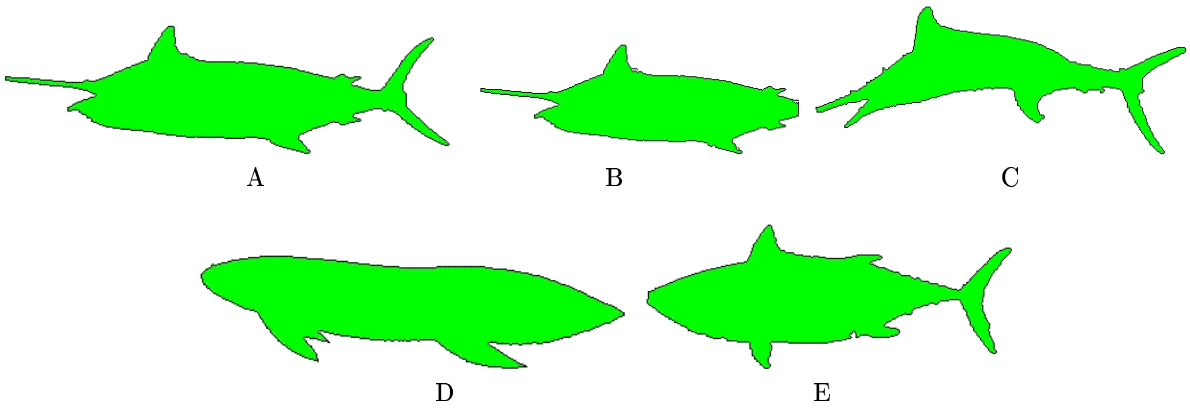


Figure 7. Test set of fish images.

We tested our method on several databases. Figure 7 shows five images from a fish shape database which was originally used in the work of F. Mokhtarian, Abbasi, and Kittler.[2] Figure 8 showed the skeleton structures extracted from the images in Fig. 7.

With the image A, as a query image, we computed the dissimilarities. The retrieval process reported C as the most similar fish, B and C as the two most similar fishs, and B, C, and E as the similar fishs. The image D was not retrieved. The results matched our expectation. In other test sets, we also obtained desired results.

We also implemented other similarity schemes (method using moment invariants only, method using Fourier descriptors only, and method using both moment invariants and Fourier descriptors) and compared the results to

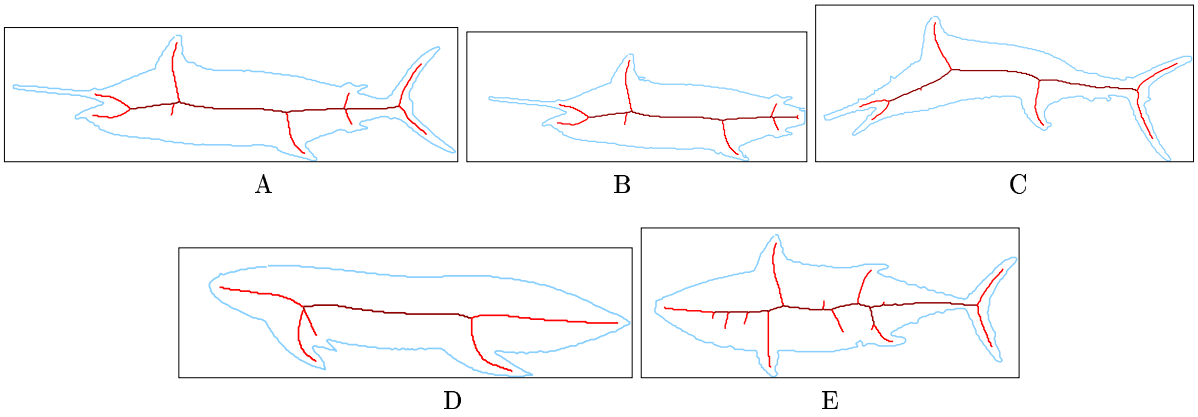


Figure 8. Skeleton structures of the image in Figure 7.

Table 1. Comparison to other similarity schemes.

	A	B	C	D	E
Moment invariants	0.000000	0.884384	1.000000	0.080693	0.729096
Fourier descriptors	0.000000	1.000000	0.554456	0.670575	0.217260
Moment invariants plus Fourier descriptors	0.000000	0.942192	0.777228	0.375634	0.473178
Our method	0.000000	0.357106	0.176234	0.821043	0.382429

that of our scheme. The output result is shown in Table 1. The method using moment invariants only retrieved image D and the method using Fourier descriptors only retrieved image E. Only our method identified that A is similar to B and C.

Beside the novel property of our method, unexpected results may appear when there is a 3-D perspective effect in the shape since the invariance holds only when the deformation is a kind of 2-D affine transformation. Bad results may also appear when the two boundaries are from the same object but a boundary was too much smoothed by a region extraction module.

5. CONCLUSION

In this paper we presented a shape-based image retrieval scheme which represents an object as a skeleton structure. A skeleton structure represents an object in a hierarchical manner and it is not sensitive to noise in object boundary. In our proposed skeleton structure method, the partial shapes of the given query object, each corresponding to spine level nodes, are compared to partial shapes of the target object. The best matched hypothesis among all the possible hypotheses is chosen and it is regarded as the match of two skeleton structures. Hence this method overcomes the shape deformation of animated object.

A drawback of our system is that it depends on the segmentation results. Also there are time consuming tasks in the retrieval process since the two feature vectors are computed in the retrieval process.

As future works of our research, we are developing an image segmentation algorithm which extracts only objects of interest regardless of the complexity of the environment where the object is located in. Future research should also include the shape matching of partially recovered objects or objects having occlusions.

ACKNOWLEDGMENTS

We would like to give thanks to Sadegh Abbasi, Dr. Farzin Mokhtarian, and Dr. Josef Kittler for allowing their fish shape database to be used in our research, Dr. S. Sclaroff for permitting the use of his shape database, and J. H. Park for providing his shape data set.

REFERENCES

1. S. Sclaroff and A. Pentland, "Modal matching for correspondence and recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence* **17**, June 1995.
2. F. Mokhtarian, S. Abbasi, and J. Kittler, "Robust and efficient shape indexing through curvature scale space," in *Proc. British Machine Vision Conf.*, 1996.
3. M. Celenk and Y. Shao, "Rotation, translation, and scaling invariant color image indexing," in *Storage and Retrieval for Image and Video Databases VII, SPIE 3656*, pp. 623–630, 1999.
4. C. Arcelli and G. S. di Baja, "Euclidean skeleton via centre-of-maximal-disc extraction," *Image and Vision Computing* **11**, pp. 163–173, Apr. 1993.
5. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision I*, Addison-Wesley, 1992.
6. D. Attali and A. Montanvert, "Modeling noise for a better simplification of skeletons," in *Proceedings of Int'l Conf. on Image Processing*, vol. 3, pp. 13–16, 1996.
7. B. M. Mehtre, M. S. Kankanhalli, and W. F. Lee, "Shape measures for content based image retrieval: a comparison," *Information Processing and Management* **33**(3), 1997.
8. M. K. Hu, "Pattern recognition by moment invariants," in *Proc. IRE*, vol. 49, 1961.
9. C.-H. Teh and R. T. Chin, "On digital approximation of moment invariants," *Computer Vision, Graphics, and Image Processing* **33**, 1986.