A Comparative Study of Curvature Scale Space and Fourier Descriptors for Shape-based Image Retrieval

Dengsheng Zhang (contact author) and Guojun Lu
Gippsland School of Computing and Information Technology
Monash University
Churchill, Victoria 3842
Australia
dengsheng.zhang, guojun.lu@infotech.monash.edu.au

Abstract

Contour shape descriptors are among the important shape description methods. *Fourier descriptors* (FD) and *curvature scale space descriptors* (CSSD) are widely used as contour shape descriptors for image retrieval in the literature. In MPEG-7, CSSD has been proposed as one of the contour-based shape descriptors. However, no comprehensive comparison has been made between these two shape descriptors. In this paper we study and compare FD and CSSD using standard principles and standard database. The study targets image retrieval application. Our experimental results show that FD outperforms CSSD in terms of robustness, low computation, hierarchical representation, retrieval performance and suitability for efficient indexing.

Keywords: Fourier descriptors, curvature scale space, CBIR, shape.

1. Introduction

Shape is one of the primary low level image features in content-based image retrieval (CBIR). There are generally two types of shape representations: region based and contour based.

Common region based methods use moment descriptors to describe shape [Hu62, TC91, TC88, LP96, Teague80, Niblack et. al93]. These include geometric moments, Legendre moments, Zernike moments and pseudo Zernike moments. Although region-based shape representations can be applied to more general situations, they usually involve more computaion. It has been known that contours are so dominant in visual perception that when drawing object, user always begin by sketching its outline. For image retrieval, contour shape representation can facilitate query by sketching (QBS).

Contour shape representations include global shape descriptors such as eccentricity and circularity [Niblack et al93], shape signatures such as chain code, centroid distance and cumulative angles [FS78, Davies97], spectral descriptors such as FD and wavelet descriptors [ZR72, PF77, KSP95, TB97, YLL98], and curvature scale space descriptors (CSSD) [MAK96, AMK00]. Global shape descriptors are very inaccurate shape descriptors which are not suitable for standalone shape descriptors, they are usually combined with other shape descriptors to discriminate shapes. Shape signatures are local representations

of shape features and are extracted from spatial domain, as a result, they are sensitive to noise. Furthermore, the matching cost using shape signatures is high due to the complex normalization of rotation invariance. Therefore, these representations need further processing using scale space or using spectral transform such as Fourier transform, short-time Fourier transform and wavelet transform. Spectral descriptors prove to be robust, noise insensitive and compact. However, short-time Fourier descriptors and wavelet descriptors suffer the same drawback as the shape signatures in the matching stage [TB97, YLL98, ZL01-2]. Among spectral descriptors, FD has advantages of simple computation and simple normalization (matching). These advantages make FD well suitable for online retrieval.

CSSD has also been widely used in shape representation [MAK96, DT97, DM00]. In the development of MPEG-7, CSSD has been proposed as contour shape descriptors [ISO00], however, CSSD has been only tested on a marine fish shape database [MAK96] which is a poor shape database due to too few varieties of shapes and too many similar shapes in the database. No comprehensive comparison has been made between FD and CSSD. Although Abbasi et al make a comparison between FD and CSSD in [AMK99], however, the comparison has several limitations.

- First, the only performance evaluated is the retrieval precision, other performance factors which are equally essential to image retrieval are ignored in the evaluation.
- Second, the database [SQID] used for the comparison is not suitable for retrieval evaluation because it is not a database of generic shapes. The SQID database consists of only marine fishes. Most shapes in the database are similar for ordinary observer, it is difficult to judge similarity between different varieties. The shapes should be classified using subjective test before they are used for retrieval test. However, they are not been classified. In MPEG-7, SQID database is combined with 200 affine transformed bream fishes to form a database for testing of robustness to affine transform; and only the 200 bream fishes are designated as queries. The 1100 shapes in SQID are not used for queries because they are not classified.
- Third, the evaluation is not an unbiased evaluation. In the test, 17 query sets, each having 8 members, are "carefully selected" as queries. In fact, for each query set, there are far more than 8 perceptually similar shapes in the database (For example, for the shapes in set 4, there are more than 60 similar shapes in the database). The authors only select 8 closely similar shapes for each query set. Even in the selected query sets, shapes from different query sets are quite similar, for example, sets 2 and set 3 are similar, set 13 and set 14 are very similar. Consequently, the reliability of the evaluation result is questionable. Furthermore, why the particular 17 sets are selected as query sets is not justified.
- Fourth, the FD (derived from complex coordinates) it uses is not robust. As has been shown in [ZL01-1], FD derived from centroid distance signature outperforms the FD used in [AKM99] significantly.

Because of the above limitations, the comparison conducted in [AKM99] is not conclusive. Marin [Marin01] compares shape representation using boundary coordinates with shape representation using FD (the same FD as that used in [AKM99]) and shape representation using curvature in scale space. However, due to the use of the fixed scale in the curvature smoothing, the curvature scale space is actually the smoothed curvature representation. Only biological cell shapes are used for the study. Evaluation of performance on individual shape is given in the paper, however, no retrieval effectiveness and details of database are reported. Furthermore, the complex normalization of the shape representations is not suitable for online retrieval.

In this paper we make a comprehensive study and comparison between FD and CSSD. Our approach differs from [AKM99, Marin01] and other common approaches of comparing shape descriptors [MKL97, LS99, Marin01] in several aspects. First, guided principles are used in the evaluation. We use the six principles set by MPEG-7 [KK00] for the overall performance evaluation, they are: good retrieval accuracy, compact features, general application, low computation complexity, robust retrieval performance and hierarchical coarse to fine representation. Second, we use standard database to conduct the retrieval test. The database we use in this paper is MPEG-7 contour shape database. Third, we use large and unbiased query sets to obtain the reliable retrieval results. Fourth, we implement the best versions of the studied descriptors to perform the comparison. Fifth, we use the more common evaluation method, i.e., precision and recall to evaluate the retrieval performance.

The rest of the paper is organized as following. In Section 2, FD and CSSD are described and discussed. In Section 3, comparison of retrieval results and computation efficiency is given in details. We conclude the paper in Section 4.

2. Contour-based Shape Descriptors

In this section, Fourier descriptors and curvature scale space descriptors are described and discussed.

2.1 Fourier Descriptors

In general, FD are obtained by applying Fourier transform on a *shape signature*, the normalized Fourier transformed coefficients are called the Fourier descriptors of the shape. The shape signature is a one dimensional function which is derived from shape boundary coordinates. Different shape signatures have been exploited to obtain FD. Complex coordinates, curvature function, cumulative angular function, centroid distance are the commonly used shape signatures. However, FD derived from different signatures has significant different performance on shape retrieval. As has been shown in [ZL01-1], FD

derived from *centroid distance function* outperforms FD derived from other shape signatures in terms of overall performance.

The first step of computing FD is to obtain the boundary coordinates ($x(t)$, $y(t)$), t = 0, 1, …, N-1, where N is the number of boundary points. In our implementation, the shape boundary points are extracted in a preprocessing stage which consists of binarization, denoising, a *m-connectivity* connection and a 8-*connectivity* contour tracing technique [Pavlidis82]. The *centroid distance function* is expressed by the distance of the boundary points from the centroid (x_c, y_c) of the shape

$$r(t) = ([x(t) - x_c]^2 + [y(t) - y_c]^2)^{1/2}, \qquad t = 0,1, \ldots, N-1$$

where $$x_c = \frac{1}{N}\sum_{t=0}^{N-1} x(t) \qquad y_c = \frac{1}{N}\sum_{t=0}^{N-1} y(t).$$

The three apple shapes (refer to apple 1, 2 and 3 in the paper) and their centroid distance function $r(t)$ are shown in Figure 1. The left apple is the original shape, the other two are the noise affected and defected versions of the original shape. The irregular and defected shapes are likely resulted from segmentation of natural scene where noise, irregularities and occlusions around object boundary are common.

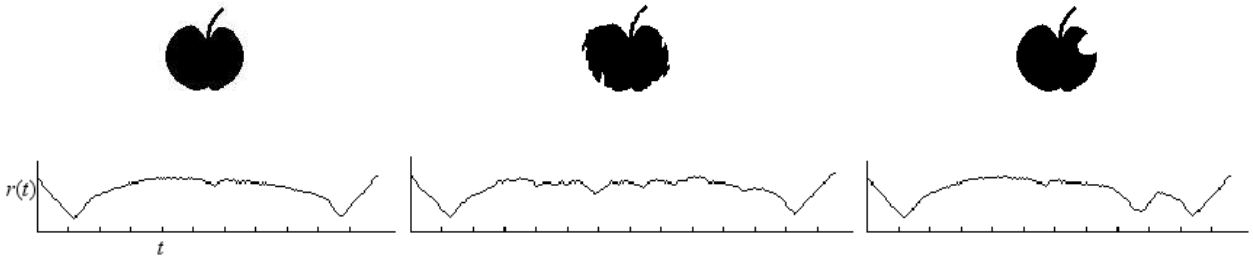


Figure 1. Three apple shapes and their centroid distance signatures.

The discrete Fourier transform of $r(t)$ is then given by

$$a_n = \frac{1}{N}\sum_{t=0}^{N-1} r(t)\exp(\frac{-j2\pi nt}{N}), \qquad n = 0, 1, \ldots, N-1$$

a_n are the Fourier transformed coefficients of $r(t)$. The reconstructed shapes of apple 1, 2 and 3 using the first 3 coefficients are shown in Figure 2 (a) and the reconstructed shapes of apple 1, 2 and 3 using the first 5 coefficients are shown in Figure 2(b). As can be seen, the three reconstructed shapes in Figure 2(a)

are very similar even though their original shapes are affected by serious noise and defections. The same is for the three shapes in Figure 2(b). Since the low frequency coefficients are usually the most significant and are used to derive shape descriptors, the two examples indicate that, shape descriptors using Fourier transformed coefficients are robust to variations of shape boundaries.

(a) (b)

Figure 2. The Fourier reconstructed apple 1, 2 and 3 using (a) the first 3 FD; (b) the first 5 FD.

The acquired Fourier coefficients are translation invariant due to the translation invariance of the shape signature. In order to describe the shape, the acquired Fourier coefficients have to be further normalized so that they are rotation, scaling and start point independent shape descriptors. From Fourier transform theory, the general form of the Fourier coefficients of a contour $r(t)$ transformed through translation, rotation, scaling and change of start point from the original contour $r(t)^{(o)}$ is given by [GRA72]:

$$a_n = \exp(jn\tau)\cdot\exp(j\phi)\cdot s \cdot a_n^{(o)}$$

where a_n and $a_n^{(o)}$ are the Fourier coefficients of the transformed shape and the original shape respectively, τ and ϕ are the angles incurred by the change of start point and the rotation respectively; s is the scale factor. Now considering the following expression

$$b_n = \frac{a_n}{a_0} = \frac{\exp(jn\tau)\cdot\exp(j\phi)\cdot s\cdot a_n^{(o)}}{\exp(j\tau)\cdot\exp(j\phi)\cdot s\cdot a_1^{(o)}}$$
$$= \frac{a_n^{(o)}}{a_0^{(o)}} = b_n^{(o)}\exp[j(n-1)\tau]$$

where b_n and $b_n^{(o)}$ are the normalized Fourier coefficients of the transformed shape and the original shape respectively. The normalized coefficient of the derived shape b_n and that of the original shape $b_n^{(o)}$ only have difference of $exp[j(n\text{-}1)\tau]$. If we ignore the phase information and only use magnitude of the coefficients, then $|b_n|$ and $|b_n^{(o)}|$ are the same. In other words, $|b_n|$ is invariant to translation, rotation, scaling and change of start point. The set of magnitudes of the normalized Fourier coefficients of the shape $\{|b_n|, 0<n\leq N\}$ are used as shape descriptors, denoted as $\{FD_n, 0<n\leq N\}$. Since centroid distance is a

real value function, only half of the FDs are distinct, therefore, only half of the FDs are needed to index the shapes [KSP95]. Finally, a feature vector consisting of half of the normalized FDs is created to index each shape: $\mathbf{f} = \{FD_1, FD_2, \ldots, FD_{N/2}\}$. The similarity between a query shape Q and a target shape T is determined by the Euclidean distance d between their FDs: $d = (\sum_{i=1}^{N/2} | FD_i^Q - FD_i^T |^2)^{\frac{1}{2}}$

The whole process of computing FD from a shape is given in Figure 3.

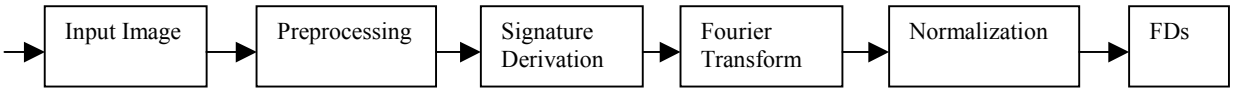


Figure 3. Block diagram of computing FD

In the implementation, 10 very complex shapes are selected from the database to simulate the worst convergence of the Fourier series of their boundary representations, the average spectrum of the 10 shapes show that 60 FDs are sufficient to describe the shape if FDs with normalized magnitude greater than 0.01 are taken as significant features. Based on this initial estimation, we test retrieval performance using different number of FDs, i.e., 5, 10, 15, 30, 60, 90 to find what is the appropriate number of FDs needed for shape description. It is found that the performance of retrieval using 15, 30, 60 and 90 are almost the same. The retrieval performance only degrades slightly when using 10 FDs. The test reveals that when the number of FDs is above 15, the retrieval performance does not improve significantly with increased number of FDs, the retrieval performance does not degrade significantly when the number of FDs is reduced to 10 FDs. It indicates, for efficient retrieval, 10 FDs are sufficient for shape description [ZL01-3].

2.2 CSS descriptors

CSS descriptors [MAK96] are essentially the descriptors of key local shape features. By putting shape boundary into scale space, not only the locations of convex (or concave) segments, but also the degree of convexity (or concavity) of the segments on the shape boundary are detected. Since curvature is a very important local measure on how fast a planar contour is turning, therefore, curvature scale space is exploited. The block diagram of computing CSS descriptors is shown in Figure 4(a).

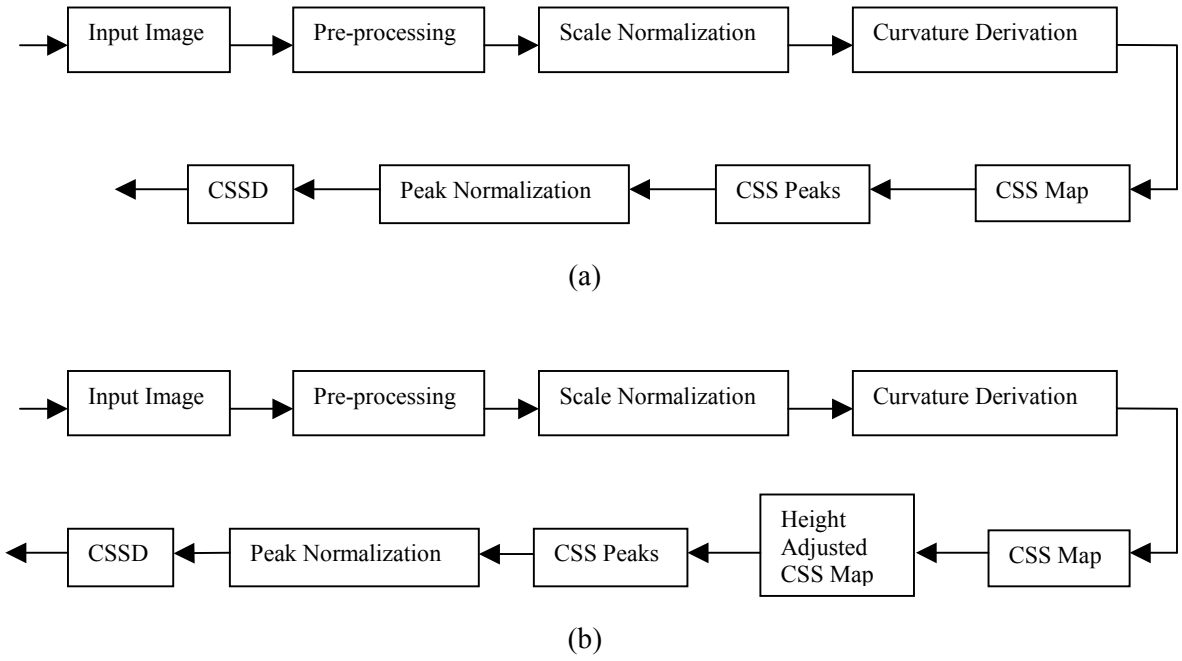


Figure 4. Block diagram of computing (a) CSSD; (b) Enhanced CSSD.

The first step of the process is the same as that in computing FD, the output of it is the boundary coordinates ($x(t)$ $y(t)$), t= 0, 1, 2, …, N-1. The second step is scale normalization which samples all the shape boundary into fixed number of points so that shapes with different number of boundary points can be matched. The normalization is done by equal arclenght sampling technique. The equal arclength sampling best preserves the boundary topological structure. The two main steps in the process are the *CSS contour map* computation and *CSS peaks* extraction. The CSS contour map is a multi-scale organization of the inflection points (or curvature zero-crossing points). To calculating CSS contour map, curvature is first derived from shape boundary points ($x(t)$ $y(t)$), t = 0, 1, 2, …, N-1:

$$k(t) = (\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)) / (\dot{x}^2(t) + \dot{y}^2(t))^{3/2}$$

where $\dot{x}(t)$, $\dot{y}(t)$ and $\ddot{x}(t)$, $\ddot{y}(t)$ are the first and the second derivatives at location t respectively. Curvature *zero-cross points* are then located in the shape boundary. The shape is then evolved into next scale by applying Gaussian smooth:

$$x'(t) = x(t) \otimes g(t,\sigma), \quad y'(t) = y(t) \otimes g(t,\sigma)$$

where $\otimes$ means convolution, and $g(t, \sigma)$ is *Gaussian function*. As σ increases, the evolving shape becomes smoother and smoother. New curvature zero-crossing points are located at the each evolving scale. This process continues until no curvature zero-crossing points are found.

The CSS contour map is composed of all zero-crossing points $zc(t, \sigma)$, where t is the location and σ is the scale at which the zc point is obtained. The peaks, or the maxima of the CSS contour map (only those peaks higher than the threshold are considered) are then extracted out and sorted in descending order. For example, the three peaks of CSS map in Figure 5(a) are (107, 226), (7, 212), (60, 16). The next step is to normalize all the obtained CSS peaks into [0, 1]. The parameter values for the peak normalization are provided in MPEG-7 document [ISO00]. Specifically, the normalization process is as following.

(a) Transform all peak heights according to $p_i' = \alpha \times p_i^{\beta}$, where p_i and p_i' are the original peak height and the transformed peak height of ith peak; α=6, and β=0.46;

(b) Shift all peaks so that the highest peak after transformation is at the origin;

(c) If the highest peak has a height of less than γ=100, remove all peaks;

(d) For any peaks which have a height of less than $p'_{\max} \times \delta$, remove them, where δ=0.05;

(e) Normalize the first peak using a maximum value of κ=1235;

(f) Normalize the other peaks, in descending order of height, using a maximum value of the previous peak height.

Finally, the normalized CSS peaks are used as CSS descriptors to index the shape. The CSS contour map and CSS peaks of the three apple shapes in Figure 1 are shown in Figure 5. The CSS contour map of Figure 5 can be read in this way: at the lower scales, there are more inflection points on the shape boundary, as the scale σ increases, the inflection points become less, finally, at the highest scale, the boundary is smooth and there is no inflection points. That the inflection points present in pairs indicates they represent the end points for each curve segment at certain scale. As σ increases, the shorter segments either disappear or are merged with longer segments. Every CSS contour branch in the map corresponds to a convexity (or concavity) in the actual image boundary. Since each shape boundary has been normalized into the same number of points, the deeper the convexity, the taller the contour; the longer the convexity, the wider the contour. For discussion convenience, the CSS peak map will be used to illustrate CSSD.

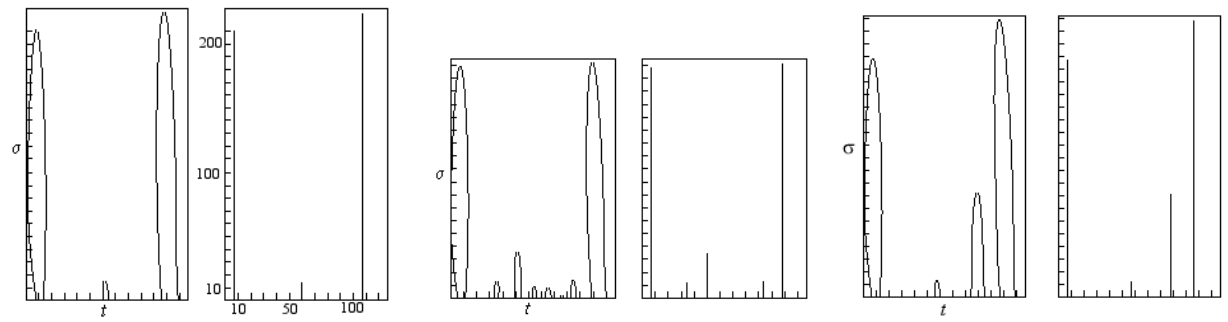


Figure 5. The CSS contour map (left) and CSS peak map (right) of (a) apple 1; (b) apple 2; (c) apple 3.

The normal CSSD computed above can fail to distinguish shallow concavity from deep concavity on the shape boundary. Consequently, dissimilar shapes can be described as similar shapes because of this failure. As an example, the two shapes in Figure 6 (a) and (b) are dissimilar shapes. However, their CSS map in Figure 6(c) and (e) are quite similar. As a result, the CSSDs extracted from the two CSS maps are also very similar, especially the higher peaks (Figure 6(d)(f)). Hence it is impossible distinguish the two shapes by using the acquired CSSDs. The reason of causing this problem is because curve segment with shallow concavity evolves rather slowly to smoothed curve. For example, the shallow concavity (marked with arrow) in shape (a) even evolves to higher scale than the deep concavity (marked with arrow) in shape (b) evolves in the smoothing process. This is observed in Figure 6(c) and (e), the two highest contour branches correspond to the two marked concavities in (a) and (b).

To solve this problem, the *enhanced CSSD* proposed by Abbasi et al [AMK00] is implemented. The enhanced CSSD is extracted from the *height adjusted CSS map*. The height adjusted CSS map is obtained by truncating the contour branches in the original CSS map along the smoothing path. The purpose of the truncation is to obtain the true height of the contour branches in the CSS map. The truncation is based on detecting the maximum curvature of curve segments (corresponding to contour branches in the CSS map) at each scale σ. Once the maximum curvature of a segment at scale σ_0 is below a threshold τ, the part of the contour branch above σ_0 is truncated. The height of that branch is adjusted to σ_0. For example, in Figure 6(g) and (i), after the height adjusting processing, the part of the contour branches above the horizontal bars are truncated and the new height of the contour branches are adjusted to the vertical positions of the horizontal bars. The CSSD obtained from the height adjusted CSS map is called the *enhanced CSSD*. While this adjustment change shallow concavity branches significantly, it does not affects deep concavity branches much. After this processing, shallow concavity can be distinguished from deep concavity. For example, after the height adjustment, the two largest descriptors in the first shape are

now inverted each other with those in the second shape (Figure 6(h)(j)); it is possible to distinguish the two shapes by the acquired enhanced CSSD.

To compare with the extraction of normal CSSD, the extraction of *enhanced CSSD* is illustrated in Figure 4(b). In the following, the enhanced CSSD is referred to as CSSD.

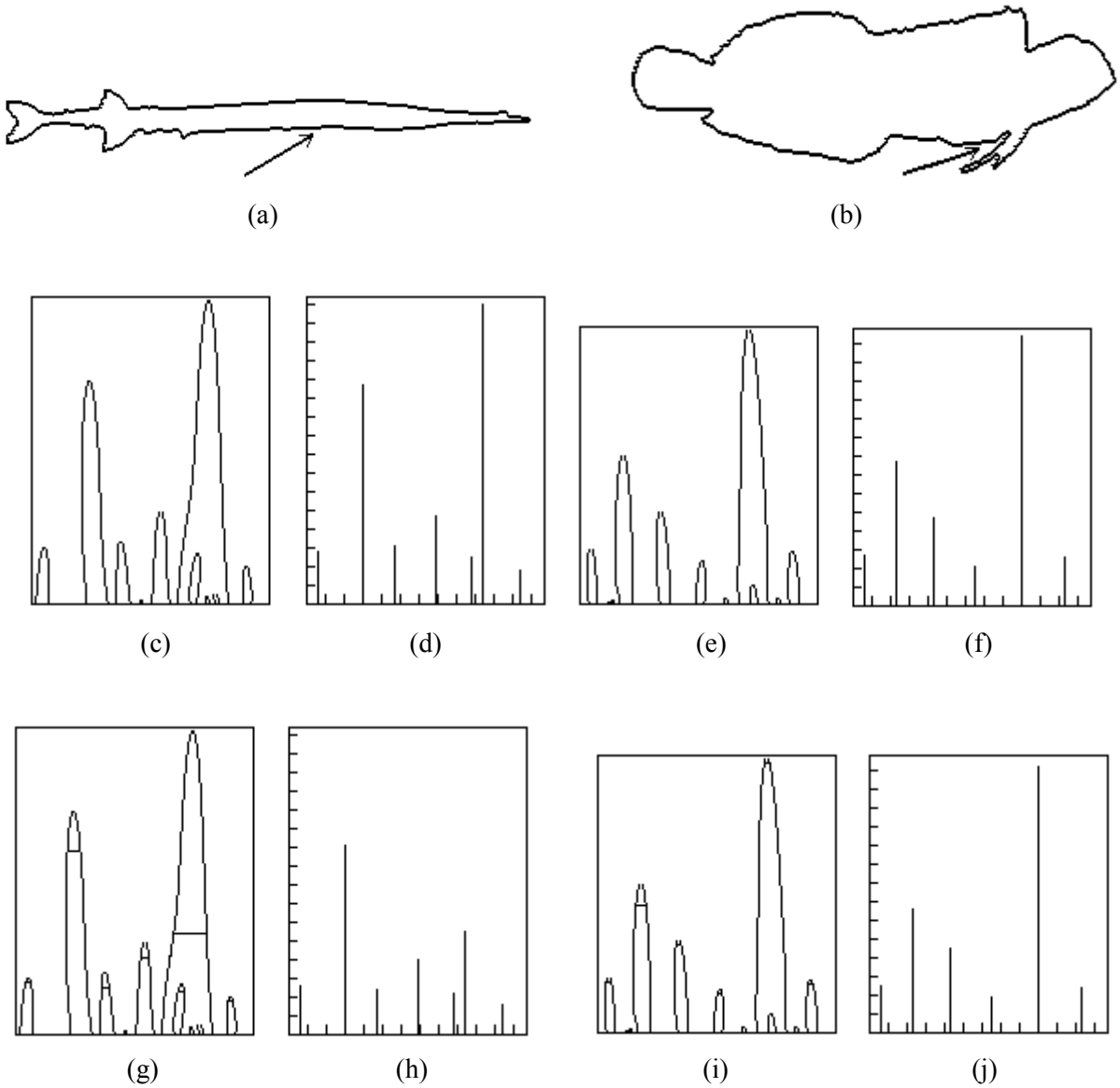


Figure 6. (a) a shape with a shallow concavity (marked with arrow) on the boundary; (b) a shape with a deep concavity (marked with arrow) on the boundary; (c) CSS map of (a); (d) CSS peak map of (a); (e) CSS map of (b); (f) CSS peak map of (b); (g) height adjusted CSS map of (a); (h) enhanced CSS peak map of (a); (i) height adjusted CSS map of (b); (j) enhanced CSS peak map of (b).

CSS descriptors is translation invariant. Scale invariance is achieved by normalizing all the shapes into fixed number of boundary points. In our implementation, this number is η=256 which is recommended in MPEG-7 document [ISO00]. Since rotation of shape causes CSS peaks circular shifting on the *t* axis, the rotation invariance is achieved by circular shifting the highest peak (*primary peak*) to the origin of the CSS map. The similarity between two shapes A and B is then measured by the summation of the peak differences between all the matched peaks and the peak values of all the unmatched peaks [MAK96]. In order to increase robustness, four schemes of circular shifting matching are applied in order to tolerate variations of peak heights of potential matching peaks (more schemes of circular shift matching can be applied to obtain more accurate matching). The four schemes of shift matching are: shifting the primary peak of A (other peaks of A are shifted accordingly) to match the primary peak of B; shifting primary peak of A to match *secondary peak* (second highest CSS peak) of B; shifting secondary peak of A to match the primary peak of B; shifting secondary peak of A to match the secondary peak of B. Since mirror shape has different CSS descriptors from the original shape, the matching has to include the mirrored shape matching. Altogether, 8 schemes of circular shift matching are needed to fulfill the matching between two sets of CSS descriptors. The fact that the peak positions of two similar shapes are usually not matching indicates matching between two sets of CSS descriptors also needs to accept certain tolerance of position variation between two potential matching peaks. In the implementation, this tolerance value is 4% of the whole boundary points which means that if two peaks are within ε=10 point distance, they are matched, otherwise they are not matched.

2.3 Discussions

In the above, the two contour shape descriptors FD and CSSD are described in details. The comparison of the two descriptors is given in this section.

The similarities between FD and CSSD are as following

- Both FD and CSSD are perceptually meaningful. FD capture structural features of the shape boundary; CSSD capture convexities (concavities) on the shape boundary
- Both FD and CSSD are robust to boundary noise and irregularities. With FD, the more significant lower frequencies preserve shape global structures which are robust to irregularities of boundary. Noise influence is eliminated through truncation of high frequencies. With CSSD, higher peaks capture merged convexities (concavities) which are robust to irregularities of boundary. Noise influence is eliminated through thresholding out short peaks.
- Both methods are application independent. No *a prior* knowledge or information about the types of shape boundary is assumed.

- Both representations are compact. The number of FD features needed to describe shape is 10, while the average number of CSSD features needed to describe shape is 6.

The differences between FD and CSSD are as following

- Feature domains. FD is obtained from spectral domain while CSSD is obtained from spatial domain.
- Dimensions. Dimension of FD feature is constant (once the number of coefficients is chosen), while dimension of CSSD feature varies for each shape.
- Feature computation complexity. From Figure 3 and 4, the computation process of CSSD is more complex than that of FD. Computation of CSSD has an extra process of scaling normalization before CSSD extraction, and the extraction of the CSSD feature takes three processes, i.e., CSS map computation, height adjusted CSS map computation, CSS peaks extraction.
- Online matching computation. The online matching of two sets of FDs is simply the Euclidean distance between two feature vectors of 10 dimensions. The online matching of two sets of CSSD involves at least 8 schemes of circular shift matching, and for each scheme of circular shift matching, it involves 6 shifts and the Euclidean distance calculation between two feature vectors of 6 dimensions.
- Type of features captured. FD captures both global features and local features while CSSD does not capture global features.
- Parameters or thresholds influence. For FD, the parameter is the number of FDs, which is predictable [ZL01-3]. For CSSD, the 8 parameters are α, β, γ, δ, κ, τ, η and ε, which have been described in Section 2.2. The parameters are determined empirically.
- Hierarchical representation. FD supports hierarchical coarse to fine representation while CSSD does not. In order to support hierarchical representation, CSSD has to incorporate shape global features such as eccentricity and circularity which are unreliable.
- Suitability for efficient indexing. FD is suitable to be organized into efficient data structure, while CSSD is not, due to its variable dimensions and complex distance calculation.

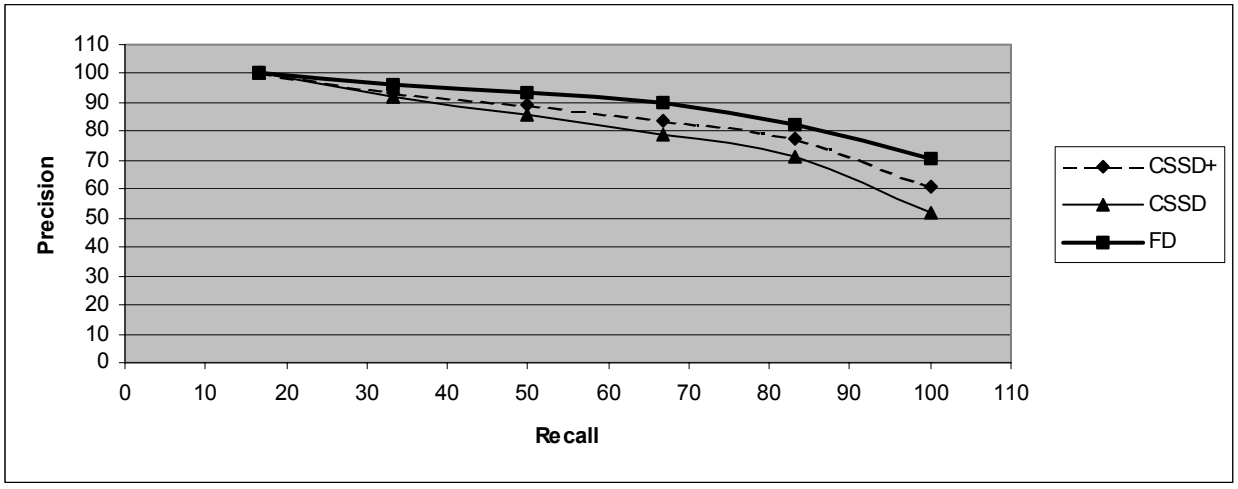
3. Comparison of Retrieval Effectiveness and Computation Efficiency

In this section, the comparison of retrieval performance and computation efficiency of the two shape descriptors is given in details.

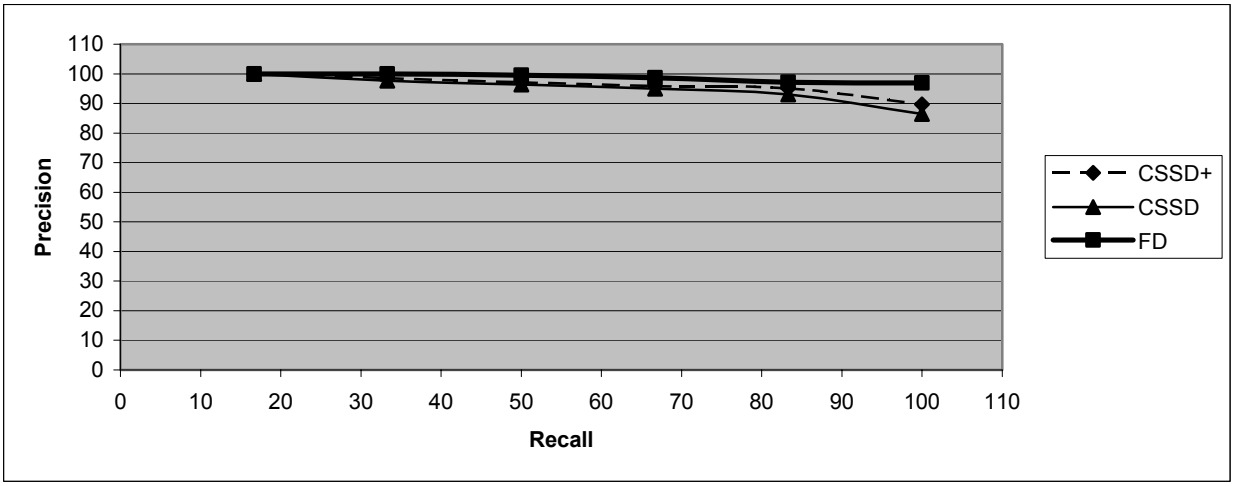
3.1 Comparison of Retrieval Effectiveness

To test the retrieval performance of the FD and CSSD, a Java-based indexing and retrieval framework which runs on Windows platform is implemented. The retrieval test is conducted on MPEG-7 contour shape database [KK00]. MPEG-7 contour shape database consists of shapes acquired from real world objects. It takes into consideration of common shape distortions in nature and the inaccuracy nature of shape boundaries from segmented shapes. The database consists of three parts, Set A, Set B and Set C. Set A has two parts, Set A1 and Set A2, each consisting of 420 shapes of 70 classes. Set A1 is for test of scale invariance, Set A2 is for test of rotation invariance. Set B has 1400 shapes which have been classified to 70 classes. Set B is for similarity-based retrieval and for testing shape descriptors' robustness to various arbitrary shape distortions. Set C consists of 200 affine transformed bream fishes and 1100 marine fishes which are unclassified. The 200 bream fishes are designated as queries. Set C is for testing shape descriptors' robustness to non-rigid object distortions. Since all the member IDs in each class of the sets are known, the retrieval is conducted automatically.

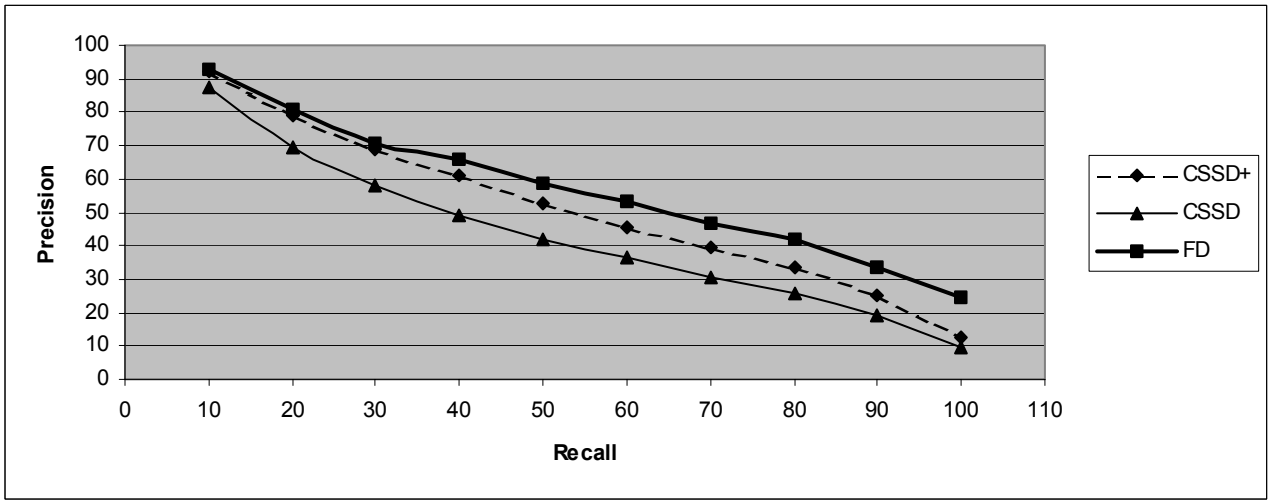
Common performance measure, i.e., precision and recall of the retrieval [Bimbo99], is used as the evaluation of the query result. Precision P is defined as the ratio of the number of retrieved relevant shapes r to the total number of retrieved shapes n, i.e. $P = r/n$. Precision P measures the accuracy of the retrieval and the speed of the recall. Recall R is defined as the ratio of the number of retrieved relevant images r to the total number m of relevant shapes in the whole database, i.e. $R = r/m$. Recall R measures the robustness of the retrieval performance. For Set A and B, all the shapes in the sets are used as queries. For Set C, the 200 bream fishes are used as queries. For each query, the precision of the retrieval at each level of the recall is obtained. The result precision of retrieval using a type of shape descriptors is the average precision of all the query retrievals using the type of shape descriptors. The precision and recall of FD and CSSD are shown in Figure 7(a)-(d). The precision and recall obtained using combined CSSD, or CSSD+, is also shown in the Figures. CSSD+ is explained shortly in this Section.



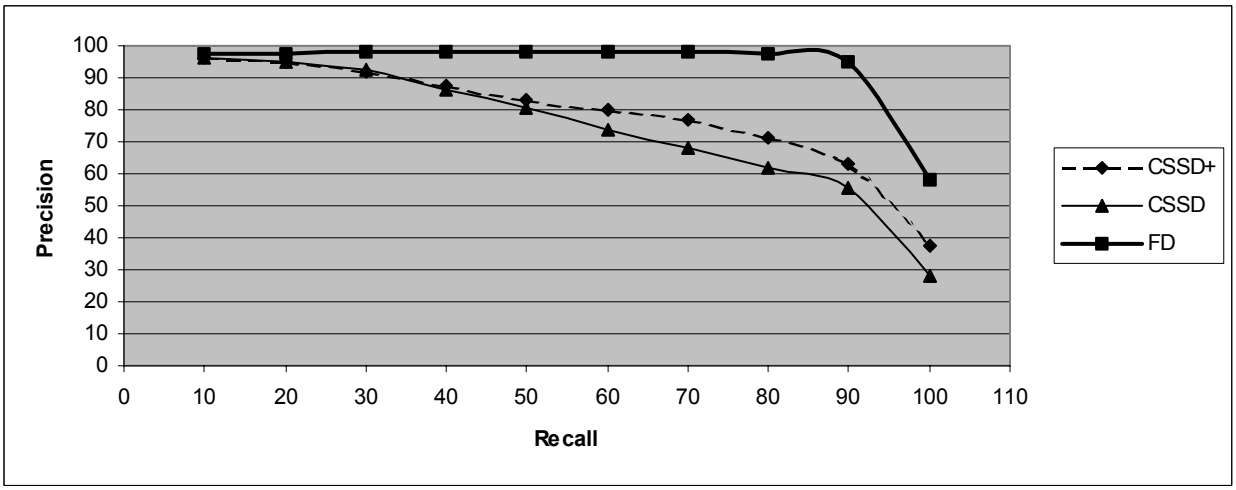
(a)


Precision
Recall
CSSD+
CSSD
FD

(b)


Precision
Recall
CSSD+
CSSD
FD

(c)


Precision
Recall
CSSD+
CSSD
FD

(d)

Figure 7. Average precision and recall of retrieval using FD and CSSD on (a) Set A1; (b) Set A2; (c) Set B; (d) Set C.

It can be seen from the precision-recall charts that FD outperforms CSSD significantly on the performance of scaling, rotation, affine and similarity retrieval, indicating that FD is more robust to general boundary variations than CSSD. In the experiments, it has been found that CSSD robustness to boundary variations is very limited. It is not robust to common boundary variations such as defections and distortions. For example, in the database, there are occluded apple shapes for testing occlusion retrieval. The two occluded apple shapes are both retrieved in the first screen (Figure 8(a)) using FD, the ranks of the two occluded apple are 5 and 13 respectively. The CSSD fails to retrieval any of the occluded apples in the first 36 retrieved shapes (Figure 8(b)), 4 example apple shapes and their CSSD are shown in Figure 9(a)-(d). The CSSD also has very poor performance on the fork shape (Figure 8(d)) while FD has very high performance on this shape (Figure 8(c)). CSSD is easily trapped by shapes with 5 prominent protrusions. 4 example fork shapes and their CSSD are shown in Figure 10(a)-(d).

From Figure 9 and 10, it can be seen that CSSD is able to keep the number of convexity features on the boundary in presence of distortions (Figure 9(a)(d) and Figure 10(a)(b)(d)). However, defections add new peaks to the map (Figure 9(b)(c) and Figure 10(c)), which consequently add net cost to the matching result. The peak heights change drastically in presence of distortions (Figure 9(d) and 10(c)(d)), especially the peak positions have changed so significantly that they cannot be matched properly by circular shift in many cases. For example, the two highest peaks of Figure 9(a) will not be matched to the two highest peaks in Figure 9(c). Similarly, the peaks in Figure 10(a) will be out of match with the peaks in Figure 10(d).

Even though the number of peaks of two CSSDs (of two similar shapes) is the same and there is a match between the peaks in horizontal positions—for example in the cases of Figure 9(a) and (d), Figure 10(a) and (b)—they are very different descriptors after normalization, due to the different order of the height of the peaks. The increase of peaks and mismatch of peaks add heavy costs to the matching result, these effectively result in false retrievals.

In the affine distortion case, when the distortion is significant, the CSSDs generated from the affined shapes become largely different (Figure 11). This explains CSSD's relatively low retrieval performance on Set C (Figure 7(d)). An example of affine retrieval using CSSD is shown in Figure 8(h).

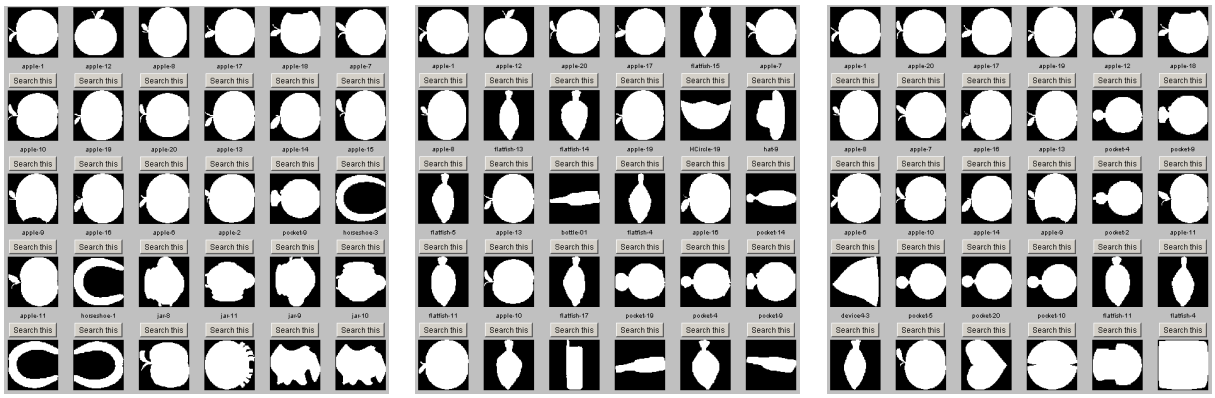
The examples indicate that robustness of CSSD is only in the sense of preserving number of prominent convexities (concavities). Variations of boundary cause drastic change in number of peaks, height of peaks, and especially positions of corresponding peaks. They also indicate that CSSD is only robust to local variations and it is not robust in global sense.

In recognizing this problem, MPEG-7 recommends combing CSSD and global shape descriptors such as eccentricity and circularity [SHB93] to form a more robust shape descriptor. The use of the global descriptors is as following.

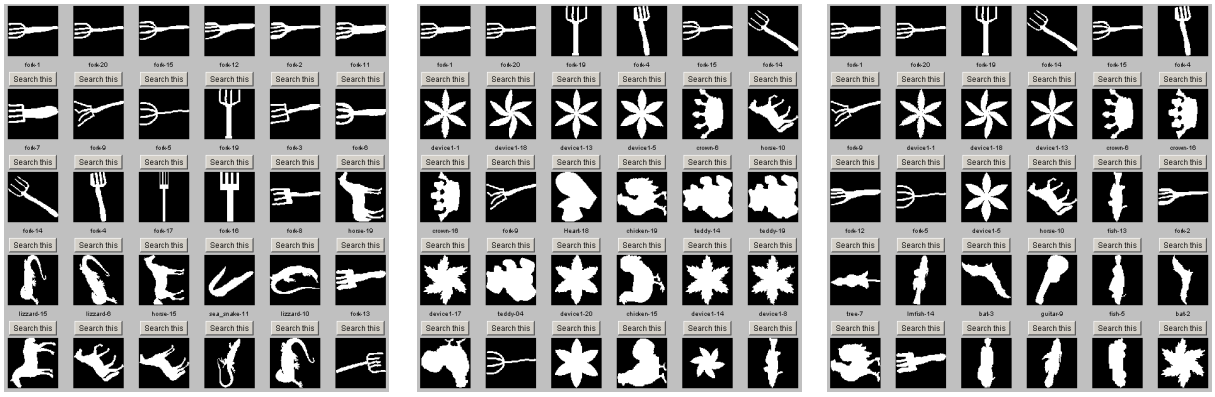
(a) If $D_e = \frac{|E_Q - E_T|}{\max(E_Q, E_T)} \le t_e$ and $D_c = \frac{|C_Q - C_T|}{\max(C_Q, C_T)} \le t_c$, perform matching; where E_Q and E_T are eccentricity of the query and the target shape respectively, C_Q and C_T are the circularity of query and target shape respectively, t_e and t_c are the threshold to filter out irrelevant shapes and t_e=t_c=0.8;

(b) The similarity between query and target shape is determined by $D = \lambda D_e + \mu D_c + D_{css}$, where D_{css} is the distance obtained from the 8 scheme matching described in Section 2.2, and λ=0.8, μ= 0.7.

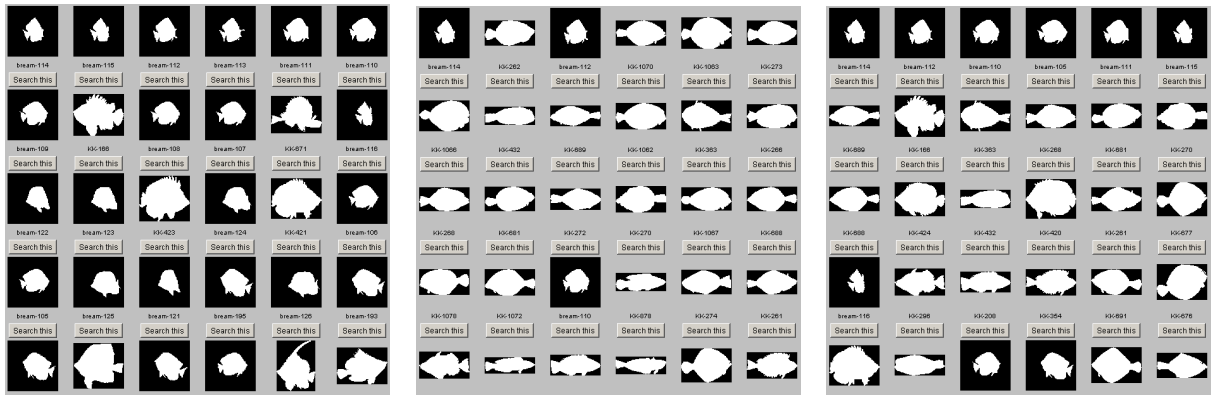
The combined shape descriptor is denoted as CSSD+. The retrieval performance of CSSD+ is also shown in Figure 7 and the corresponding retrievals for the above three queries are shown in Figure 8(c)(f)(i). It is observed that CSSD+ improves CSSD, however, its retrieval performance on all the sets is still lower than FD. In addition, it adds three more parameters: t_e, λ, μ , to the descriptor.



(a) (b) (c)

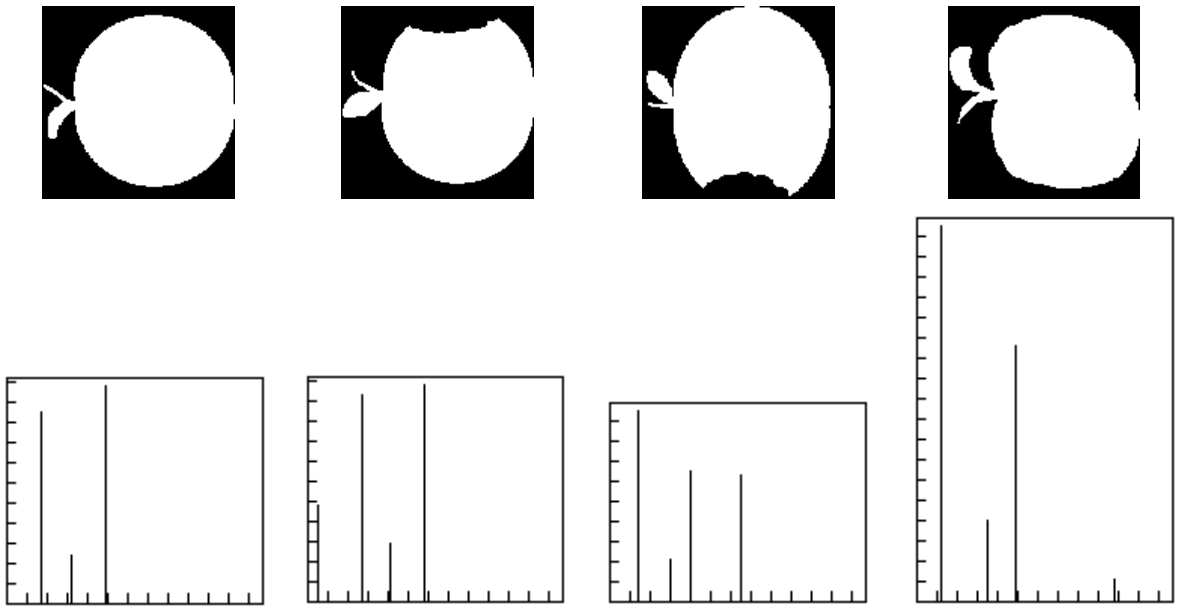


(d) (e) (f)



(g) (h) (i)

Figure 8. Retrieval of apple shapes (a) using FD; (b) using CSSD; (c) using CSSD+. Retrieval fork shapes (d) using FD; (e) using CSSD; (f) using CSSD+. Retrieval bream fish shapes (g) using FD; (h) using CSSD; (i) using CSSD+. In all the screen shots, the top left shape is the query shape and the retrieved shapes are arranged in descending order of similarity to the query. The screen shots are retrieval examples from Set B.



(a) (b) (c) (d)

Figure 9. (a)(b)(c)(d) four apple shapes on the top and their corresponding CSSD at the bottom

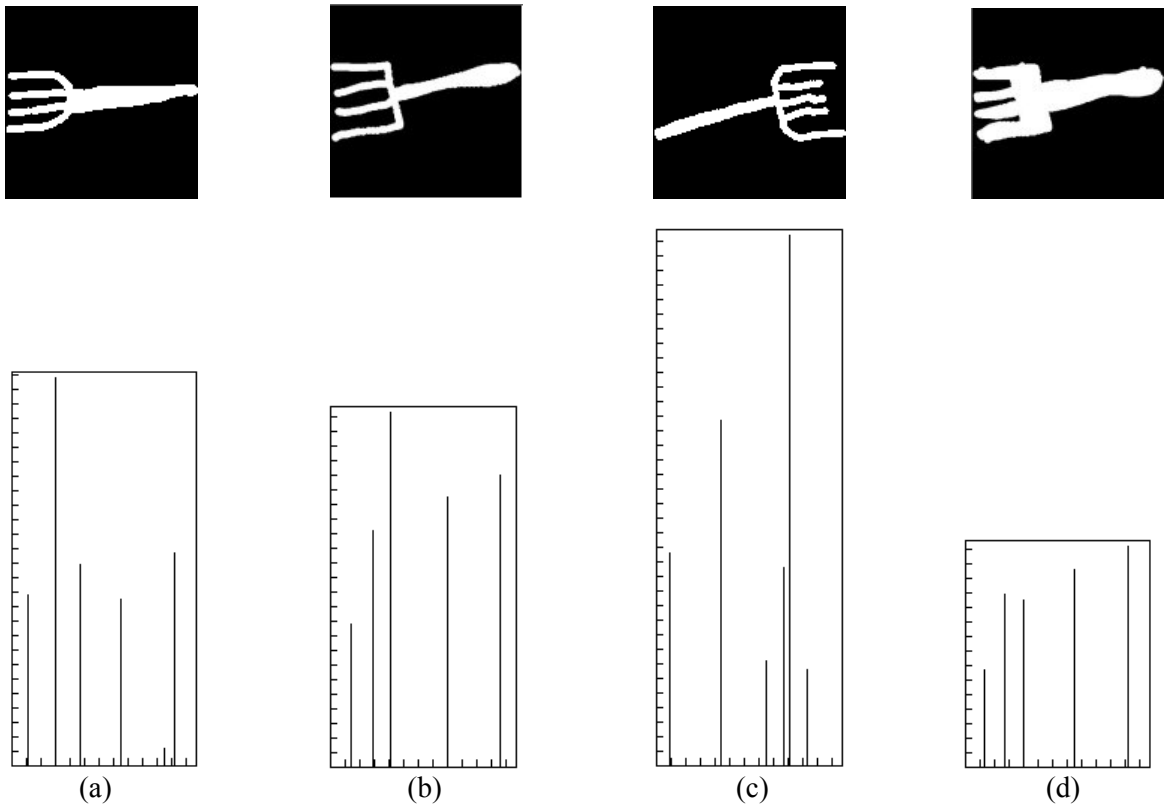


Figure 10. (a)(b)(c)(d) four fork shapes on the top and their corresponding CSSD at the bottom.

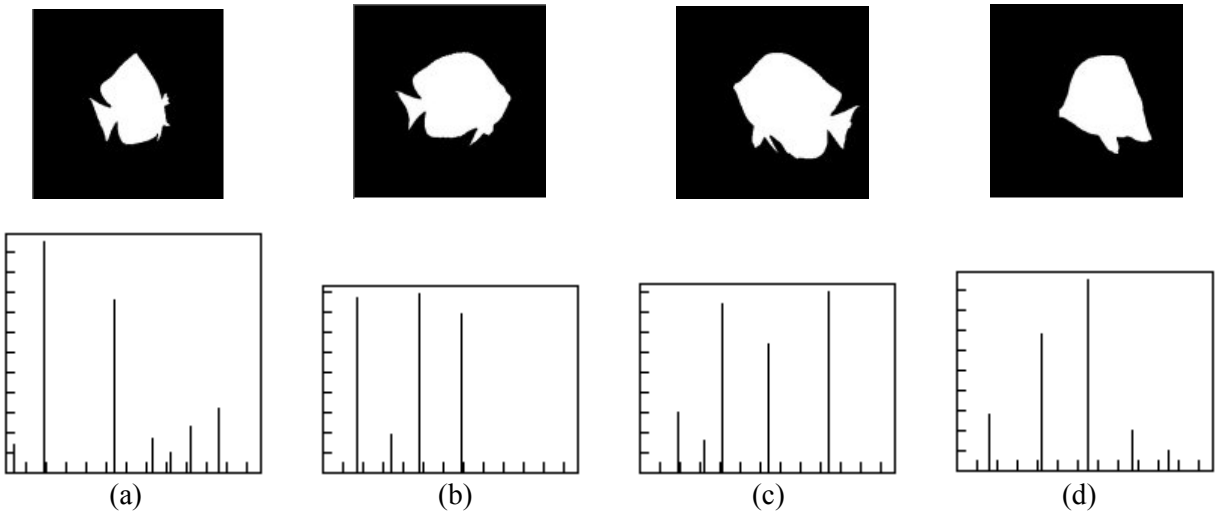


Figure 11. (a)(b)(c)(d) four bream fish shapes on the top and their corresponding CSSD at the bottom.

3.2 Comparison of Computation Efficiency

In order to compare the computation efficiency of the two shape descriptors, the feature extraction and the retrieval are tested on the Windows platform of a Pentium III-866 PC with 256M memory. The time taken for the feature extraction and the retrieval on Set B of MPEG-7 contour shape database is

computed. To eliminate the time sway caused by the system processes running behind, three processes of feature extraction and retrieval have been run respectively. The average time from the three process of running is given in Table 1. It can be seen from Table 1 that FD is much more efficient, especially in terms of average retrieval time.

Table 1. The average elapsed time of feature extraction and retrieval for 1400 shapes

Time / Shape descriptors	Total time of feature extraction of 1400 shapes	Average time of feature extraction of each shape	Total time of retrieval of 1400 queries	Average time of retrieval of each query
FD	81903 ms	59 ms	49711 ms	36 ms
CSSD	120977 ms	86 ms	163621ms	117ms

4. Conclusions

In this paper we have made comprehensive study and comparison between Fourier descriptors and curvature scale space descriptors using standard methodology. Experimental results show that in terms of robustness, low computation, hierarchical representation, retrieval performance and suitability for efficient indexing, FD outperforms CSSD. It has been found from the study that compared with FD, CSSD has several drawbacks.

- CSSD is only robust to local boundary variations, it's not robust in global sense.
- The low dimension advantage is offset by its complex matching.
- The retrieval performance of CSSD on all the sets of MPEG-7 contour shape database is lower than that of FD.
- CSSD does not support hierarchical representation. In order to support hierarchical representation, it has to incorporate other global shape features.
- CSSD is an unstable representation. The representation and retrieval performance depend on empirical factors such as the number of sample points on the boundary, the threshold to eliminate short peaks, the tolerance value for peak position matching, the peak normalization factor and the weight factors for incorporating global descriptors. Altogether, 11 empirical parameters are involved in the matching.
- CSSD is not suitable for efficient indexing due to the expensive matching and variable feature dimensions.

Based on these facts, we recommend that FD be included as one of contour shape descriptors in MPEG-7.

The contributions of the paper are summarized in three aspects. First, two widely used contour shape descriptors are comprehensively studied and evaluated. Second, a simpler contour shape descriptor has been found having significantly better performance than the contour shape descriptors adopted by MPEG-7. Third, a evaluation scheme — featured by guided principles, standard database, large query sets, common evaluation measurement and best versions of features — is presented for comparing different types of shape descriptors. The scheme can be applied to comparing other audio/video descriptors.

In the future, the combination of contour-based and region-based descriptors will be studied to handle very complex shapes and shapes with large boundary indentations and protrusions.

References:

[AMK99] S. Abbasi, F. Mokhtarian, J. Kittler. Curvature Scale Space Image in Shape Similarity Retrieval. Multimedia Systems, 7 (6): 467-476, NOV 1999.

[AMK00] S. Abbasi, F. Mokhtarian, J. Kittler. Enhancing CSS-based Shape Retrieval for Objects with Shallow Concavities. Image And Vision Computing, 18 (3): 199-211 Feb. 2000.

[Bimbo99] A. Del Bimbo. Visual Information Retrieval. pp. 56-57, Morgan Kaufmann Publishers, Inc. San Francisco, USA, 1999.

[Davies97] E. R. Davies. Machine Vision: Theory, Algorithms, Practicalities. Academic Press, 1997.

[DT97] Gregory Dudek, John K. Tsotsos, Shape Representation and Recognition from Multiscale Curvature, Computer Vision and Image Understanding 68(2): 170-189, November 1997.

[DM00] M. Daoudi and S. Matusiak. Visual Image Retrieval by Multiscale Description of User Sketches. Journal of Visual Languages and Computing, 11:287-301, 2000.

[FS78] H. Freeman and A. Saghri. Generalized chain codes for planar curves. In Proceedings of the 4th International Joint Conference on Pattern Recognition, pages 701-703, Kyoto, Japan, November 7-10 1978.

[GRA72] G. Granlund. *Fourier Preprocessing for Hand Print Character Recognition*. IEEE Trans. Computers, 21:195-201, 1972.

[Hu62] Ming-Kuei Hu. Visual pattern Recognition by Moment Invariants. IRE Transactions on Information Theory, IT-8:179-187, 1962.

[ISO00] S. Jeannin Ed. MPEG-7 Visual part of experimentation Model Version 5.0. ISO/IEC JTC1/SC29/WG11/N3321, Nordwijkerhout, March, 2000.

[KK00] H. Kim and J. Kim. Region-based shape descriptor invariant to rotation, scale and translation. Signal Processing: Image Communication 16:87-93, 2000.

[KSP95] Hannu Kauppinen, Tapio Seppanen and Matti Pietikainen. An Experimental Comparison of Autoregressive and Fourier-Based Descriptors in 2D Shape Classification. IEEE Trans. PAMI-17(2):201-207, 1995

[LP96] Simon X. Liao and Miroslaw Pawlak. On Image Analysis by Moments. IEEE Trans. On Pattern Analysis and Machine Intelligence. 18(3):254-266, 1996.

[LS99] G. J. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. Multimedia System 7(2):165-174, 1999.

[MAK96] F. Mokhtarian, S. Abbasi and J. Kittler. Efficient and Robust Retrieval by Shape Content through Curvature Scale Space. Int. Workshop on Image DataBases and Multimedia Search, pp. 35-42, Amsterdam, The Netherlands, 1996.

[Marin99] F. J. Sanchez-Marin. Automatic recognition of biological shapes with and without representations of shape. Artificial Intelligence in Medicine, 18(2):173-186, 2000.

[Marin01] F. J. Sanchez-Marin. Automatic recognition of biological shapes using the Hotelling transform. Computers In Biology And Medicine, 31(2): 85-99 MAR 2001.

[MKL97] B.M.Mehtre, Mohan S. Kankanhalli and Wing Foon Lee. Shape Measures for Content based Image Retrieval: A Comparison. Information Processing & Management, 33(3):319-337, 1997.

[Niblack et al93] W. Niblack et al. The QBIC Project: Querying Images By Content Using Color, Texture and Shape. SPIE Conf. On Storage and Retrieval for Image and Video Databases, vol 1908, San Jose, CA, pp.173-187, 1993.

[Pavlidis82]T. Pavlidis. Algorithms for Graphics and Image Processing. Computer Science Press, 1982, pp.143.

[PF77] Eric Persoon and King-sun Fu. Shape Discrimination Using Fourier Descriptors. IEEE Trans. On Systems, Man and Cybernetics, Vol.SMC-7(3):170-179, 1977.

[SHB93] M. Sonka, V. Hlavac and R. Boyle. Image Processing, Analysis and Machine Vision. Chapman & Hall Computing, 1993, pp.225-227.

[SQUID] http://www.ee.surrey.ac.uk/Research/VSSP/imagedb

[Teague80] Michael Reed Teague. Image Analysis Via the General theory of Moments. Journal of Optical Society of America, 70(8):920-930, 1980.

[TB97] Q. M. Tieng and W. W. Boles Recognition of 2D Object Contours Using the Wavelet Transform Zero-Crossing Representation IEEE Trans. on PAMI 19(8) Aug.1997

[TC88] C.-H. Teh and R. T. Chin. On image analysis by the methods of moments. IEEE Trans. On Pattern Analysis and Machine Intelligence, 10(4):496-513, 1988

[TC91] G. Taubin and D. B. Cooper. Recognition and Positioning of Rigid Objects Using Algebraic Moment Invariants. SPIE Conf. On Geometric Methods in Computer Vision, Vol. 1570, pp.175-186, 1991.

[YLL98] H. S. Yang, S. U. Lee, K. M. Lee. Recognition of 2D Object Contours Using Starting-Point-Independent Wavelet Coefficient Matching. Journal of Visual Communication and Image Representation, 9(2): 171-181, 1998.

[ZL01-1] D. S. Zhang and G. J. Lu. Shape Retrieval Using Fourier Descriptors. Proc. Int. Conference on Multimedia and Distance Education (ICMADE'01), Fargo, ND, USA, June 2001, pp.1-9.

[ZL01-2] D. Zhang and G. Lu. A Comparison of Shape Retrieval Using Fourier Descriptors and Short-time Fourier Descriptors. Accepted to be published on The Second IEEE Pacific-Rim Conference on Multimedia (PCM'01), Beijing, October, 2001.

[ZL01-3] D. S. Zhang and G. Lu. Image Retrieval Based on Shape Using Fourier Descriptors. Submitted to IEEE Transaction on Multimedia, June, 2001.

[ZR72] Charles T. Zahn and Ralph Z. Roskies. Fourier Descriptors for Plane closed Curves. IEEE Trans. On Computer, c-21(3):269-281, 1972.