SHAPE INDEXING BY DYNAMIC PROGRAMMING

Nir Weissberg, Shlomi Sagi, and Doron Shaked

Department of Electrical Engineering
Vision Research and Image Science Laboratory
Technion, Israel Institute of Technology.

HP Laboratories Israel
Technion City
Haifa, 32000

Abstract

With the vast amount of publicly available high-quality images, image databases start to emerge, and image database management methods are becoming essential. An important manner of finding your way around an image database is to look for images similar to a desired reference image. There are many indicators for content similarity of images: color, texture, composition etc. In this paper we focus on shape similarity, and present a new Dynamic Algorithm for Shape Indexing ("DASI") based on Curve-Signatures and Dynamic-Programming.

Introduction

Unlike alphanumeric databases that have simple indexing criteria, indexing of image databases is more complex. The objective of this work is to provide a new method for image indexing by reference examples. Indexing by examples refers to the way in which users query the image database. Typing in "I am looking for an image of such and such characterizations“ requires a very high level of definition on the part of the user, and corresponding generalization on the part of the system. Instead, it is much easier to label an image or even sketch it, by indexing images according to similarity of contours. Former works with the same purpose include Mokhtarian, Abbasi and Kittler [1], [2] who use Curvature Scale Space features, Ma and Manjunath [3] who use Fourier based shape descriptors, and Milios and Petrakis who use Dynamic Programming of convex/concave shape segments [4].

Shape similarity is context dependent, and thus for example, similarity of shapes pertaining to rigid objects is different from those pertaining to non-rigid objects. The proposed shape indexing method is flexible, and enables its intuitive tuning to various similarity types. It was tested on the Surrey database containing 1100 images of fish contours [1].

An additional aspect of shape similarity is that it is subjective. The proposed method may index according to particular shape parts (e.g. the shape of the tail), enabling users to customize the search according to their subjective preferences.

Description of the Algorithm

A curve signature is a parametric description of the curve that is invariant to a group of viewing transformations [5]. The proposed algorithm employs *Curvature*, which is invariant to Euclidean motion (translation and rotation). Different curve signatures might be used in an equivalent manner (e.g. those providing invariance to affine transformations). For invariance to mirror symmetry, each contour is represented by a pair of signatures: The original signature, and its mirror representation.

Dynamic Programming quantifies the difference between a reference signature ***R***, and a test signature ***T***, through the match error of the best warp mapping between the signatures [6]. Indexing is achieved by comparing the reference to each of the database entries and sorting.

Critical Implementation Aspects

The detailed description of the algorithm may be found in [7]. Certain critical implementation details of the proposed Dynamic-Programming routine are defined as follows:

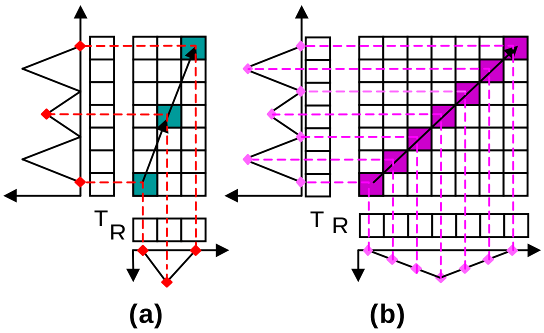

Figure 1: Signature interpolation.

Signature Interpolation: A mapping path with slope bigger than unity (45º) inevitably skips elements in ***T***. In Figure 1(a) this phenomenon results in an undesirable 'perfect' match, due to skipping of a pair of large values of ***T***. To overcome this problem, signatures are interpolated to a fixed length. Figure 1(b) shows the result of the interpolation, in which these values are not skipped, and a match error is noted reflecting the signatures' difference.

Boundary Conditions: To support invariance to the position of the initial signature point, boundary conditions were made uniform. Consequently, instead of one (or a few) warp mappings examined in the speech recognition implementations [6], the proposed algorithm checks a possible route for each point of the test signature.

Warp Deformation Penalties: Warp penalties are designed, on one hand, to preserve the desired global shape of the mapping, and enable local flexibility on the other hand:

Slope Penalty - Proportional to the local deviation from unit slope relative to the starting point of the path. This forces ***R*** and ***T*** to complete a full circle together.

Jump Penalty - Proportional to the square of the local matching gradient. This penalty reduces the amount of ***T*** values that are skipped.

Tuning the Method to Perceptual Similarity

Some fundamental modifications introduced to the basic algorithm as described above, enable an intuitive tuning of the algorithm to reflect perceptual similarity.

<u>Signature</u>: The signature sequence $S_l(i)$ is augmented by two one-sided neighborhood signature sequences describing the signature's difference from it's neighborhood $S_p(i) = S_l(i) - \frac{1}{K+1}\sum_{j=o}^{K} S_l(i-j)$, and $S_f(i) = S_l(i) - \frac{1}{K+1}\sum_{j=0}^{K} S_l(i+j)$. The combined signature is a sequence of 3 vectors $\{S_l, S_p, S_f\}$, employed to enhance the similarity between fish that are related via a relatively constant curvature distortion (e.g. while swimming). The indexing process is thus refined: Shape of similar fish in similar pose (S_l, S_p and S_f similar) are preferred over shapes of similar fish in a different pose (only S_p and S_f similar), which are in turn preferred over all other shapes. Figure 2 is an example of the effect neighborhood signatures has on indexing results.

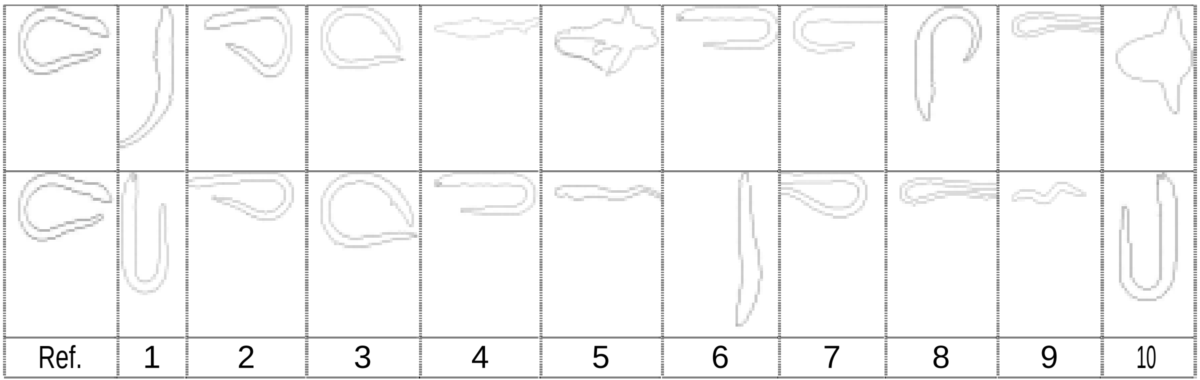

Figure 2: Indexing results without neighborhood signatures on the upper row, and with neighborhood signatures on the lower row.

<u>Shape Smoothing</u>: A curve smoothing pre-process [8] emphasizes the general shape of the image contour by filtering high curvature protrusions of small support, which otherwise tend to dominate the similarity measure in no proportion to their perceptual significance, see Figure 3.

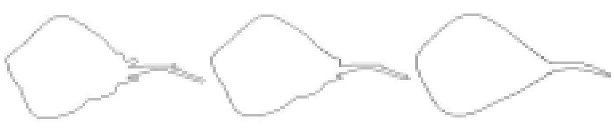

Figure 3: Curve smoothing.

<u>Partial Compare</u>: This option enables the user to specify a boundary segment ***R'***, to be searched for in the database. To implement partial compare, the boundary conditions are

modified. ***R*** is cyclically shifted to the beginning ***R'***, and the mapping is stopped at the end of ***R'***.

Results and Summary

The method presented in this paper, DASI, is a new proposed version of the Dynamic Programming algorithm, specially designed to handle the challenge of image indexing form the shape similarity aspect. It effectively does so. As can be seen in Figure 2, and figure 4, DASI provides good indexing for a large database of fish contours, and handles both the full and partial compare tasks. It is a flexible indexing tool tunable to important perceptual shape-similarity paradigms. Further research is needed in order to speed up the indexing process, see e.g. [4]. Further details, databases, and executables may by found in DASI's homepage [9].

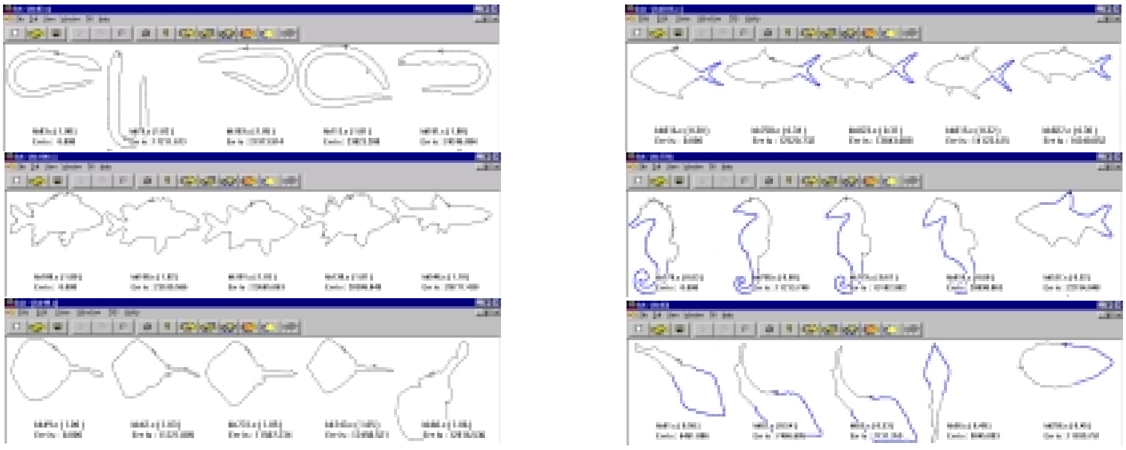

Figure 4: Examples of DASI's indexing results. The reference image is on the left and similar images are sorted from left to right. Partial compare segments are marked blue on the right column.

References

[1] http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html/

[2] F. Mokhtarian, S. Abbasi, and J. Kittler, Proc. BMVC'96, 53 (1996)

[3] W. Y. Ma, and B. S. Manjunath, ICIP (1997)

[4] E. Milios, and E. G. M. Petrakis, IEEE Image Proc. **9**, 141 (2000)

[5] A.M. Bruckstein et al., CVGIP: Image Understanding **58**, 49 (1993)

[6] L.R. Rabiner, A.E. Rosenberg, and S.E. Levinson, IEEE ASSP **26**, 557 (1978)

[7] N. Weissberg, and S. Sagi, Vision Lab Report, EE Dept. Technion - in Hebrew (2000)

[8] A. M. Bruckstein, G. Sapiro, and D. Shaked, Int. J. of PR. and AI **9**, 991 (1995)

[9] http://www.geocities.com/dasi_fish_2000/