

# Statistical techniques to find similar objects in images

*I.K. Fodor*

This article was submitted to the 2003 *Proceedings of the American Statistical Association*, Statistical Computing Section

**October 16, 2003**

***U.S. Department of Energy***

Lawrence  
Livermore  
National  
Laboratory

## **DISCLAIMER**

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This is a preprint of a paper intended for publication in a journal or proceedings. Since changes may be made before publication, this preprint is made available with the understanding that it will not be cited or reproduced without the permission of the author.

# Statistical Techniques to Find Similar Objects in Images

Imola K. Fodor

Center for Applied Scientific Computing

Lawrence Livermore National Laboratory Livermore, California 94551

## Abstract

One problem in similarity-based object retrieval (SBOR) is how to define and estimate the similarity between two objects. In this paper we present a shape similarity measure based on thin-plate splines, and compare its performance with several other measures used in SBOR. We evaluate the methods on both artificial and real images.

**Key words:** image processing, similarity based object retrieval, thin-plate splines.

## 1 Introduction

Similarity-based object retrieval (SBOR) is the process that, given a query object, a (usually large) database, and a measure of similarity, returns a list of objects in the database that are similar to the query.

Finding books similar to a given book (for example, the recommendations at *amazon.com*), and finding images similar to a specified query image, are both specific examples of the general concept. One of the difficulties encountered in such inquiries is the definition of “similar”. Is it the author of the book or the topic of the book? The shape or the texture of the image? Or, is it a combination? The answer to the question can be very different, depending on the notion of the similarity used.

Our work primarily concerns the analysis of large spatio-temporal scientific datasets, and therefore we are interested in SBOR in scientific applications.

One of the unsolved problems in physics is understanding turbulence. In order to understand turbu-

lent flow, it is important to analyze and compare the results of different simulations and experiments. Currently, physicists use the “eyeball” measure to decide which simulation is closest to a particular simulation, or which simulation is closest to the experimentally measured data. Our goal is to enhance the visual comparison by providing more rigorous quantitative measures of similarity using SBOR.

Fig. 1 displays three time steps from a turbulent flow simulation [5]. It is a three-dimensional shock tube simulation: initially, two fluids are separated by a membrane in a stationary tube. Next, a shock is applied to the tube, and the time evolution of the mixing of the two fluids is observed. The mixing is evolved on a  $2048 \times 2048 \times 1920$  grid over 27,000 time steps, and follows the change in variables like pressure, density, velocity, and entropy. The computations were carried out on 960 nodes of the IBM-SP TeraOp system at LLNL. While our ultimate interest is to characterize the similarity among full three-dimensional models, our initial effort has been to develop an SBOR framework for the smaller, two-dimensional problem. Fig. 2 displays a 2D slice of the entropy data, and Fig. 3 a smaller query image. Our immediate goal then can be stated as finding regions in a collection of images that are similar to a query image provided by a user. For example, find regions of Fig. 2 which are similar to the query in Fig. 3.

The rest of this paper is organized as follows. Section 2 describes two simple similarity measures that can be used in SBOR, Section 3 describes the thin-plate spline transform and motivates its use for defining a measure of shape similarity in SBOR, and Section 4 concludes with a summary.

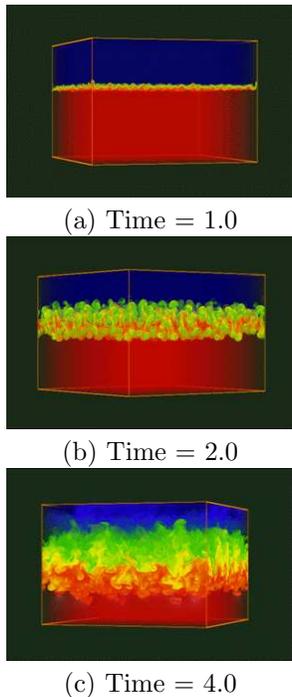


Figure 1: Three time steps in the simulation over the  $2048 \times 2048 \times 1920$  grid.

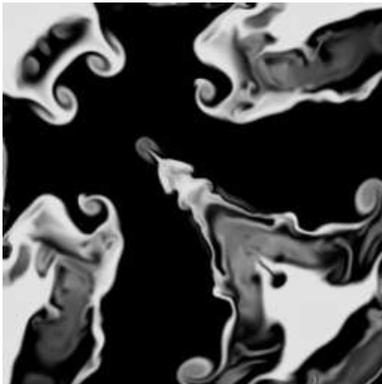


Figure 2: Example 2D data:  $256 \times 256$  pixels.

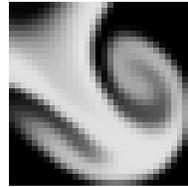


Figure 3: Example query image:  $64 \times 64$  pixels.

## 2 Two simple similarity measures

The normalized pixel-wise difference between two images is the simplest similarity measure that has been suggested in the literature. In this case, the difference between the query and an arbitrary image of the same size is

$$d_p = \frac{1}{IJ} \sqrt{\sum_{i=1}^I \sum_{j=1}^J (p_{ij} - q_{ij})^2}, \quad (1)$$

where  $I$  is the horizontal dimension,  $J$  the vertical dimension of the images,  $p_{ij}$  the intensity at the  $\{i, j\}$ th pixel location in the comparison image, and  $q_{ij}$  the intensity at the  $\{i, j\}$ th pixel in the query image.

Fig. 4 presents the results of a pixel-wise comparison between the original query image in Fig. 3 and  $64 \times 64$  sub-images of the  $256 \times 256$  image displayed in Fig. 2. The red windows highlight the location of the best 100 matches. We obtained the sub-images by sliding windows of  $64 \times 64$  over the original larger image, and then computed the pixel-wise error given in Eq. (1) for each of the sub-images. In our initial implementation, we incremented the sliding windows by a step size of one pixels in both directions, but clearly, such a fine-grained matching is redundant, and larger step sizes are also possible.

Fig. 5 displays two example matches from the 100 best ones. While the image in panel (a) is visually more similar to the query image than the match shown in panel (b) is, the pixel-wise distance of (a) to the query image ( $d_p = 4.96$ ) is larger than the pixel-wise distance of (b) to the query image ( $d_p = 4.09$ ).

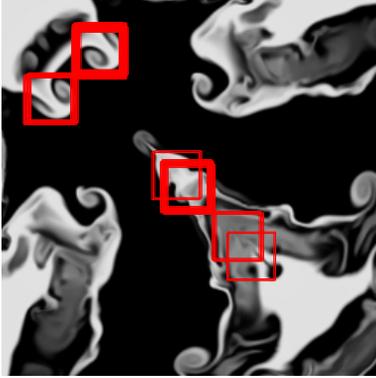


Figure 4: Best 100 matches based on pixel-wise error.

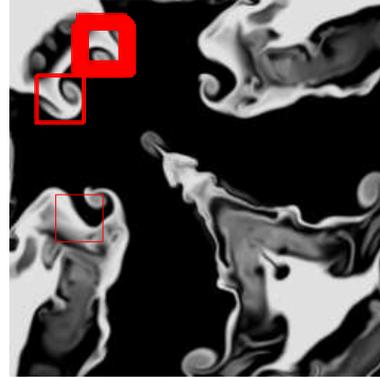


Figure 6: Best 100 matches based on histogram bin error.

Clearly, the pixel-wise error measure in Eq. (1) does not capture the similarity of the shapes.

Fig. 6 displays the 100 best matches to the query image in Fig. 3 using  $d_h$  as a measure of closeness between the images,

$$d_h = \frac{1}{K} \sqrt{\sum_{k=1}^K (n_k - m_k)^2}, \quad (2)$$

where  $K = 10$  denotes the number of bins in the histogram,  $n_k$  is the number of query image pixels falling into the  $k$ th histogram bin, and  $m_k$  the number of comparison image pixels in the  $k$ th bin. The values of the histogram-based error for the two matches in Fig. 5 are  $d_h = 43.73$  and  $d_h = 159.76$ , for the images in panels (a) and (b), respectively. While the histogram-based similarity measure improves on the pixel-based results, it does not capture adequately the shape of the objects.

In the next section, we describe a similarity measure that is based strictly on the shapes of the objects.

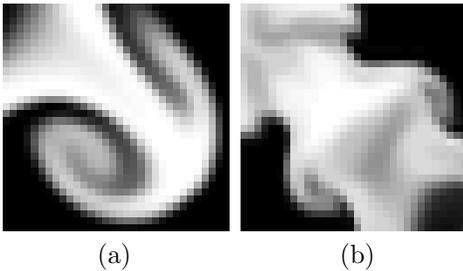


Figure 5: Two matches to Fig. 3. (a)  $d_p = 4.96$ ,  $d_h = 43.73$  (b)  $d_p = 4.09$ ,  $d_h = 159.76$

### 3 A shape similarity measure based on thin-plate splines

As a motivating example, Fig. 7 displays a simple transformation using thin-plate splines (TPS). The

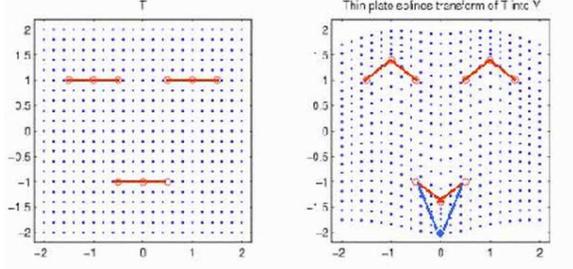


Figure 7: Example TPS transform: from a flat face to a smiley face.

grid points in the first image are mapped into the corresponding grid points on the second image, using the constraint that the nine two-dimensional landmark (also called control, or fiducial) points defining  $T$  in red in the first image get mapped exactly to the nine points that define  $Y_1$  in red in the second image. The  $x$  and  $y$  coordinates of the nine control points of  $T$  and  $Y_1$  are given by  $T = (-1.5, 1.0, -0.5, -0.5, 0.0, 0.5, 0.5, 1.0, 1.5; 1.0, 1.0, 1.0, -1.0, -1.0, -1.0, 1.0, 1.0, 1.0)$  and  $Y_1 = (-1.5, 1.0, -0.5, -0.5, 0.0, 0.5, 0.5, 1.0, 1.5; 1.0, 1.4, 1.0, -1.0, -1.4, -1.0, 1.0, 1.4, 1.0)$ , respectively. The TPS bending energy, defined later in Eq. (18), required for transforming  $T$  into  $Y_1$  is equal to  $e_1 = 1.44$ . If we perturb the fifth control point coordinates of  $Y_1$  from  $(0.0; -1.4)$  to  $(0.0, 2.00)$ , and denote the resulting point  $Y_2$  (shown in blue in the second panel of Fig. 7), then the bending energy increases to  $e_2 = 3.94$ . Intuitively this makes sense, as  $Y_2$  is “further” from  $T$  than  $Y_1$  is.

In the general case of comparing two shapes, following [2], let  $t_j$ ,  $j = 1, \dots, k$  denote the  $k$  two-dimensional landmarks in the first image to be mapped exactly onto the  $y_i$ ,  $i = 1, \dots, k$  two-dimensional landmarks on the second image. We are interested thus in two-dimensional transformation  $\Phi(t_j) = (\Phi_1(t_j), \Phi_2(t_j))^T$  that satisfies the  $2k$  interpolation constraints in Eq. (3):

$$(y_j)_r = \Phi_r(t_j), \quad r = 1, 2, \quad j = 1, \dots, k. \quad (3)$$

A pair of thin-plate splines is defined by

$$\Phi(t) = (\Phi_1(t), \Phi_2(t))^T = c + At_W^T s(t), \quad (4)$$

where  $t$  is  $(2 \times 1)$ ,  $s(t) = (\sigma(t - t_1), \dots, \sigma(t - t_k))^T$ , and

$$\sigma(h) = \begin{cases} \|h\|^2 \log(\|h\|), & \|h\| > 0, \\ 0, & \|h\| = 0, \end{cases} \quad (5)$$

with  $c(2 \times 1)$ ,  $A(2 \times 2)$ , and  $W(k \times 2)$ , as  $2k+6$  parameters. It can be shown [2] that the TPS minimizes, among all interpolating functions from  $T$  to  $Y$ , the total bending energy  $J(\Phi)$ , where

$$J(\Phi) = \sum_{j=1}^2 \int \int_{R^2} \left( \frac{\delta^2 \Phi_j}{\delta x^2} \right)^2 + 2 \left( \frac{\delta^2 \Phi_j}{\delta x \delta y} \right)^2 + \left( \frac{\delta^2 \Phi_j}{\delta y^2} \right)^2. \quad (6)$$

Let  $T$  and  $Y$  denote the  $k \times 2$  dimensional control point matrices

$$T = [t_1 \ t_2 \ \dots \ t_k]^T \quad \text{and} \quad Y = [y_1 \ y_2 \ \dots \ y_k]^T. \quad (7)$$

Natural thin-plate splines, in addition to the  $2k$  interpolation constraints, also satisfy the two additional constraints

$$1_k^T W = 0 \quad \text{and} \quad T^T W = 0. \quad (8)$$

Combining Equations 3 and 8, we obtain

$$\begin{pmatrix} S & 1_k & T \\ 1_k^T & 0 & 0 \\ T^T & 0 & 0 \end{pmatrix} \begin{pmatrix} W \\ c^T \\ A^T \end{pmatrix} = \begin{pmatrix} Y \\ 0 \\ 0 \end{pmatrix}, \quad (9)$$

where  $(S)_{ij} = \sigma(t_i - t_j)$  and  $1_k$  is the  $k$ -vector of ones. If  $\Gamma$  denotes the first matrix of the left hand side of Equation 9,

$$\Gamma = \begin{pmatrix} S & 1_k & T \\ 1_k^T & 0 & 0 \\ T^T & 0 & 0 \end{pmatrix}, \quad (10)$$

then  $\Gamma$  is invertible, provided that  $S$  is. Therefore,

$$\begin{pmatrix} W \\ c^T \\ A^T \end{pmatrix} = \Gamma^{-1} \begin{pmatrix} Y \\ 0 \\ 0 \end{pmatrix}, \quad (11)$$

Partitioning  $\Gamma^{-1}$  as

$$\Gamma^{-1} = \begin{pmatrix} \Gamma^{11} & \Gamma^{12} \\ \Gamma^{21} & \Gamma^{22} \end{pmatrix}, \quad (12)$$

where  $\Gamma^{11}$  is  $k \times k$ , we find the parameters

$$W = \Gamma^{11}Y, \quad (13)$$

and

$$\begin{pmatrix} c^T \\ A^T \end{pmatrix} = (\hat{\beta}_1, \hat{\beta}_2) = \Gamma^{21}Y. \quad (14)$$

If  $Q = (1_k, T)$ , we have

$$\Gamma^{11} = S^{-1} - S^{-1}Q(Q^T S^{-1}Q)^{-1}Q^T S^{-1}, \quad (15)$$

$$\Gamma^{12} = (Q^T S^{-1}Q)^{-1}Q^T S^{-1} = (\Gamma^{21})^T, \quad (16)$$

$$\Gamma^{22} = -(Q^T S^{-1}Q)^{-1}. \quad (17)$$

$\Gamma^{11}$  is called the bending energy matrix. The minimized value of the total bending energy in Eq. (6) is

$$J(\Phi) \equiv e = \text{trace}(Y^T \Gamma^{11} Y). \quad (18)$$

Fig. 8 shows the TPS transform grid and associated bending energy for three artificial examples constructed from the fluid turbulence data. First, we applied the Canny edge detector to the query “mushroom” image in Fig. 3, identified its boundary, and a few control points. The red points in the second column in Fig. 8 shows the resulting control points and define the template image shape. Next, we constructed the three additional mushroom images shown in the first columns of Fig. 8 by slightly disturbing the boundary of the original template. The similarity between the original and the modified image decreases as we proceed from (a) to (c). The second column displays the required TPS grid transformations that map the three synthetically created shapes to the template shape. As expected, the bending energy  $e$  increases as the similarity between the objects increases.

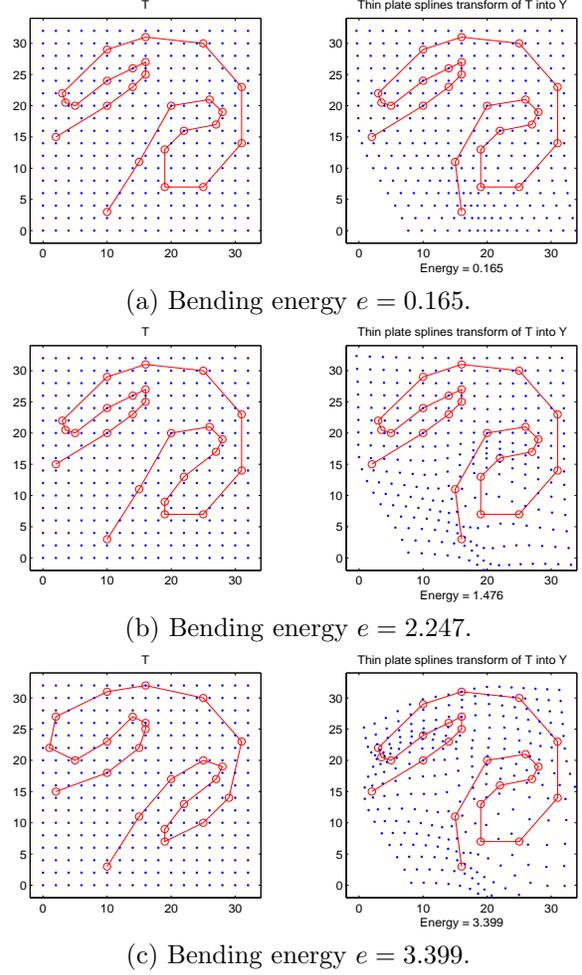


Figure 8: Three TPS transform examples for the fluid turbulence data.

## 4 Summary

In this paper, we briefly investigated the use of the thin-plate splines transform in constructing a similarity measure for shapes. While our results are very preliminary, they are promising nonetheless.

One of the drawbacks of the TPS-based approach is that it assumes that the objects to compare are segmented, and their control points identified and properly ordered. Object segmentation is itself a difficult problem, especially in the context of simulation data where the objects are not always well-defined. Before the TPS-based method can be implemented in an automated manner in large collections of images, we need to have a fast and robust way to identify and align the objects. A promising approach that compares the objects without relying on control points appears in [6]. We plan to investigate more closely the methods suggested in that paper and other alternatives.

Another open issue is how to properly combine the results of different similarity measures, and attach meaningful uncertainty estimates. We plan to implement several additional measures, statistical (e.g. Procrustes analysis [2]) and other (e.g. MPEG-7 descriptors [4]), and investigate means to optimally combine them.

We will incorporate our findings into the the Sapphire SBOR framework [3]. The current implementation, which includes simple features (such as the mean and the standard deviation), histograms, MPEG-7 descriptors, and numerous derived features (such as the Chi-square distance between histograms), has returned good results in moderate size datasets [1].

## Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48, UCRL-JC-154621.

## References

- [1] E. Cantu-Paz, S.-C. Cheung, and C. Kamath. Retrieval of similar objects in simulation data using machine learning techniques. In *SPIE Electronic Imaging*, submitted, 2003.
- [2] I.L. Dryden and K.V. Mardia. *Statistical Shape Analysis*. Wiley, 1998.
- [3] C. Kamath et al. Sapphire: Large-scale data mining and pattern recognition website. <http://www.llnl.gov/casc/sapphire/>.
- [4] B.S. Manjunath, P. Salembier, and T. Sikora, editors. *Introduction to MPEG-7 Multimedia Content Description Interface*. Wiley, 2002.
- [5] A. A. Mirin et al. Very high resolution simulation of compressible turbulence on the ibm-sp system. Supercomputing 99 Conference, November 1999.
- [6] Belongie. S., J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 24(4):509–522, April 2002.

University of California  
Lawrence Livermore National Laboratory  
Technical Information Department  
Livermore, CA 94551

