

# 3-D Diffeomorphic Shape Registration on Hippocampal Data Sets

Hongyu Guo<sup>1</sup>, Anand Rangarajan<sup>2</sup> and Sarang C. Joshi<sup>3</sup>

<sup>1</sup>Dept. of CAMS, Texas A&M University-Corpus Christi

<sup>2</sup>Dept. of CISE, University of Florida

<sup>3</sup>Dept. of Radiation Oncology, University of North Carolina at Chapel Hill

**Abstract.** Matching 3D shapes is important in many medical imaging applications. We show that a joint clustering and diffeomorphism estimation strategy is capable of simultaneously estimating correspondences and a diffeomorphism between unlabeled 3D point-sets. Correspondence is established between the cluster centers and this is coupled with a simultaneous estimation of a 3D diffeomorphism of space. The number of clusters can be estimated by minimizing the Jensen-Shannon divergence on the registered data. We apply our algorithm to both synthetically warped 3D hippocampal shapes as well as real 3D hippocampal shapes from different subjects.

## 1 Introduction

Shape matching is ubiquitous in medical imaging and in particular, there is a real need for a turnkey non-rigid point feature-based shape matching algorithm [1,2]. This is a very difficult task due to the underlying difficulty of obtaining good feature correspondences when the shapes differ by an unknown deformation. In this paper, we attempt to formulate a precise mathematical model for point feature matching.

Previous work on the landmark matching problem [3,4] ignored the unknown correspondence problem and assumed all the landmarks are labeled. And furthermore, there is a considerable amount of point feature data where the cardinalities in the two shapes are not equal and a point-wise correspondence cannot be assumed. On the other hand, previous work on the correspondence problem [5,6] did not solve the diffeomorphism problem. The deformation model used was splines, like thin-plate splines, Gaussian radial basis functions and B-splines. The principal drawback of using a spline for the spatial mapping or the deformation model is the inability of the spline to guarantee that there are no local folds or reflections in the mapping and that a valid inverse exists.

Here, we are not interested in curve and surface matching because unlike a point-set representation of shapes, which is a universal representation, curve or surface representation of shapes usually require prior knowledge about the topology of the shapes. A point-set representation of shapes is especially useful when feature grouping (into curves and the like) cannot be assumed.

As far as we know, there is very little previous work on 3D diffeomorphic point matching. Our previous work in [7] deals with 2-D shapes only. The only known competitors are the distance transform-based approach and the recent work of [8]. Due to the indirect approach of transforming point-sets into distance fields, the distance transform method [9] has not seen wide applicability for point-sets.

## 2 Diffeomorphic Point Matching with the Unknown Correspondence

We use a Gaussian mixture model to describe the data points with a log-likelihood

$$\log p(x|\mathbf{r}, \sigma_T) = \sum_{i=1}^{N_1} \log \sum_{a=1}^N \exp\left(-\frac{1}{2\sigma_T^2} \|x_i - r_a\|^2\right), \quad (1)$$

where  $x$  is a point in  $\mathbf{R}^d$ ,  $\mathbf{r}$  is the collective notation of a set of cluster centers and  $\sigma_T^2$  is the variance of each Gaussian distribution. As pointed out by Hathaway [10], the EM algorithm maximizing (1) can be viewed as an alternative minimization of the following objective

$$E(M, \mathbf{r}) = \frac{1}{2\sigma_T^2} \sum_{i=1}^{N_1} \sum_{a=1}^N M_{ia}^x \|x_i - r_a\|^2 + \sum_{i=1}^{N_1} \sum_{a=1}^N M_{ia}^x \log M_{ia}^x \quad (2)$$

with simplex constraints on  $M$ . We put together the clustering energy of the two point-sets and the diffeomorphic deformation energy induced in space and our objective function is

$$\begin{aligned} & E(M^x, M^y, \mathbf{r}, \mathbf{s}, v, \phi) \\ &= \sum_{i=1}^{N_1} \sum_{k=1}^N M_{ik}^x \|x_i - r_k\|^2 + 2\sigma_T^2 \sum_{i=1}^{N_1} \sum_{k=1}^N M_{ik}^x \log M_{ik}^x \\ &+ \sum_{j=1}^{N_2} \sum_{k=1}^N M_{jk}^y \|y_j - s_k\|^2 + 2\sigma_T^2 \sum_{j=1}^{N_2} \sum_{k=1}^N M_{jk}^y \log M_{jk}^y \quad (3) \\ &+ \sum_{k=1}^N \|s_k - \phi(r_k, 1)\|^2 + 2\sigma_T^2 \lambda \int_0^1 \int_{\Omega} \|Lv(x, t)\|^2 dx dt. \end{aligned}$$

Notice we have multiplied each term in the objective function by a constant  $2\sigma_T^2$ . In the above objective function, the matrix entry  $M_{ik}^x$  is the membership of data point  $x_i$  in cluster  $k$  whose center is at location  $r_k$ . The matrix entry  $M_{jk}^y$  is the membership of data point  $y_j$  in cluster  $k$  whose center is at position  $s_k$ .

The diffeomorphic deformation energy in  $\Omega$  is induced by the landmark displacements from  $r$  to  $s$ , where  $x \in \Omega$  and  $\phi(x, t)$  is a one parameter diffeomorphism:  $\Omega \rightarrow \Omega$ . The diffeomorphism  $\phi(x, t)$  is generated by the flow  $v(x, t)$ .  $\phi(x, t)$  and  $v(x, t)$  together satisfy the transport equation  $\frac{\partial \phi(x, t)}{\partial t} = v(\phi(x, t), t)$  and the initial condition  $\forall x, \phi(x, 0) = x$  holds. This is in the inexact matching form and the displacement term  $\sum_{k=1}^N \|s_k - \phi(r_k, 1)\|^2$  plays an important role

here as the bridge between the two systems. This is also the reason why we prefer the deformation energy in this form because the coupling of the two sets of clusters appear naturally through the inexact matching term and we don't have to introduce external coupling terms as in [7].

### 3 A Diffeomorphic Point Matching Algorithm

Our joint clustering and diffeomorphism estimation algorithm has two components: i) clustering and ii) diffeomorphism estimation.

For the clustering part, we use the deterministic annealing approach proposed by Rose *et al.* [11] in order to avoid poor local minima. For the diffeomorphism estimation, we expand the flow field in term of the kernel  $K$  of the  $L$  operator

$$v(x, t) = \sum_{k=1}^N \alpha_k(t) K(x, \phi_k(t)) \quad (4)$$

where  $\phi_k(t)$  is notational shorthand for  $\phi(r_k, t)$  and we also take into consideration the affine part of the mapping when we use thin-plate kernel with matrix entry  $K_{ij} = -r_{ij}$  and  $r_{ij} = \|x_i - x_j\|$ . After discretizing in time  $t$ , the objective in (3) is expressed as

$$\begin{aligned} & E(M^x, M^y, \mathbf{r}, \mathbf{s}, \alpha(t), \phi(t)) \\ &= \sum_{i=1}^{N_1} \sum_{k=1}^N M_{ik}^x \|x_i - r_k\|^2 + T \sum_{i=1}^{N_1} \sum_{k=1}^N M_{ik}^x \log M_{ik}^x \\ &+ \sum_{j=1}^{N_2} \sum_{k=1}^N M_{jk}^y \|y_j - s_k\|^2 + T \sum_{j=1}^{N_2} \sum_{k=1}^N M_{jk}^y \log M_{jk}^y \\ &+ \sum_{k=1}^N \|s_k - r_k - \sum_{l=1}^S \sum_{t=0}^S [P(t)d_l(t) + \alpha_l(t)K(\phi_k(t), \phi_l(t))]\|^2 \quad (5) \\ &+ \lambda T \sum_{k=1}^N \sum_{l=1}^N \sum_{t=0}^S \langle \alpha_k(t), \alpha_l(t) \rangle K(\phi_k(t), \phi_l(t)) \end{aligned}$$

where  $P(t)$  is the homogeneous form of  $\phi(t)$ . It is easy to find a close form solution for  $d(t)$  and  $\alpha(t)$  after performing a QR decomposition on  $P$  and we use gradient descent to solve for  $\phi_k(t)$  when  $\alpha_k(t)$  is held fixed.

The clustering of the two point-sets is handled by a deterministic annealing EM algorithm which iteratively estimates the cluster memberships  $M^x$  and  $M^y$  and the cluster centers  $r$  and  $s$ . The update of the memberships is the very standard E-step of the EM algorithm [12] and is performed as shown below.

$$M_{ik}^x = \frac{\exp(-\beta \|x_i - r_k\|^2)}{\sum_{l=1}^N \exp(-\beta \|x_i - r_l\|^2)}, \forall ik \quad (6)$$

$$M_{jk}^y = \frac{\exp(-\beta \|y_j - s_k\|^2)}{\sum_{l=1}^N \exp(-\beta \|y_j - s_l\|^2)}, \forall jk. \quad (7)$$

The cluster center update is the M-step of the EM algorithm. This step is not the typical M-step. We use a closed-form solution for the cluster centers which is an approximation. From the clustering standpoint, we assume that the change in the diffeomorphism at each iteration is *sufficiently small so that it can be neglected*. After making this approximation, we get

$$r_k = \frac{\sum_{i=1}^{N_1} M_{ik}^x x_k + s_k - \sum_{l=1}^N \int_0^1 \alpha_l(t) K(\phi_l(t), \phi_k(t)) dt}{1 + \sum_{i=1}^{N_1} M_{ik}^x}, \quad (8)$$

$$s_k = \frac{\sum_{j=1}^{N_2} M_{jk}^y y_j + \phi(r_k, 1)}{1 + \sum_{j=1}^y M_{jk}^y}, \forall k. \quad (9)$$

The overall algorithm is described below.

- **Initialization:** Initial temperature  
 $T = 0.5(\max_i \|x_i - x_c\|^2 + \max_j \|y_j - y_c\|^2)$  where  $x_c$  and  $y_c$  are the centroids of  $X$  and  $Y$  respectively.
- **Begin A:** While  $T > T_{\text{final}}$ 
  - **Step 1:** Clustering
    - Update memberships according to (6), (7).
    - Update cluster centers according to (8), (9).
  - **Step 2:** Diffeomorphism
    - Update  $(\phi, v)$  by minimizing

$$E_{\text{diff}}(\phi, v) = \sum_{k=1}^C \|s_k - \phi(r_k, 1)\|^2 + \lambda T \int_0^1 \int_{\Omega} \|Lv(x, t)\|^2 dx dt$$

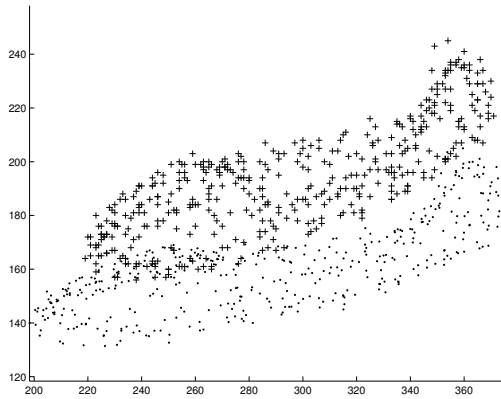
- **Step 3:** Annealing.  $T \leftarrow \gamma T$  where  $\gamma < 1$ .
- **End**

## 4 Experiments and Results

### 4.1 Experiments on synthetic data

We selected one hippocampus point-set and warped it with a known diffeomorphism using a Gaussian Radial Basis Function (GRBF) kernel. We choose  $\sigma = 60$  for the GRBF because with this larger  $\sigma$  we are able to generate more global warping.

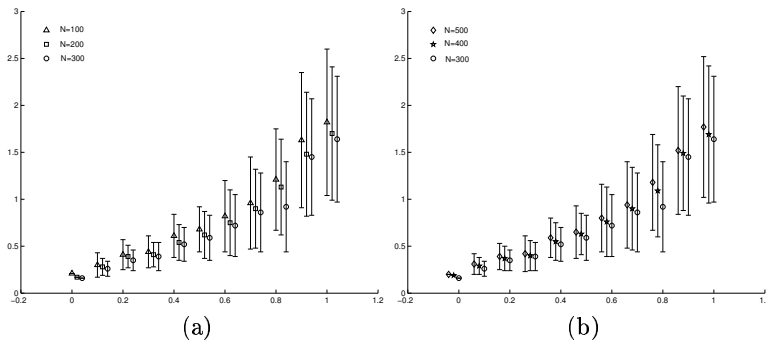
Figure 1 shows the two hippocampal shapes. The set with “+” markers is the original set and the set with point markers is the set after GRBF warping. First, we have no noise added. We used the TPS kernel to recover the diffeomorphism via joint clustering using our algorithm. We experimented with different number of clusters and found the corresponding standard errors. It is easy to see that the standard error goes down as the number of clusters goes up from 100 to 300



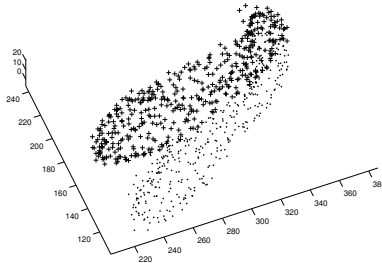
**Fig. 1.** Hippocampal point-sets

and goes up again when the number of clusters increases further. This is because when we have too few clusters, the points are not well represented by the cluster centers. On the other hand, if we have too many clusters, the variance between the two shapes is too big and the deformation increases dramatically. There is an optimal number of clusters and in this case we find it to be 300.

Next we add noise to the warped data and test the robustness of our algorithm to noise. After GRBF warping we add Gaussian noise to the warped data with different variances  $\sigma_N$ . We experimented with ten trials for each noise level from 0.1 to 1.0 and for each cluster level from 100 to 500. We can see the standard error increase with the increasing noise level but it approximately keeps in the range of the noise. This is easy to see when plotted in Figure 2 with error bars. Figure 2(a) has the errors for 100, 200 and 300 clusters and Figure 2(b) has the errors for 300, 400 and 500 clusters. We can see that at the 300 cluster level we obtain the best matching.



**Fig. 2.** Matching results on synthetic data for different number of clusters



**Fig. 3.** Two hippocampus shapes

## 4.2 Experiments on real data

We applied the algorithm on different real hippocampal data sets. We have ten 3D point-sets which were extracted from epilepsy patients with left anterior temporal lobe foci identified with EEG. An interactive segmentation tool was used to segment the hippocampus in the 3D anatomical brain MRI of the ten subjects. The shapes vary from point-set to point-set and the number of points in each point-set varies from 310 to 555. Figure 3 shows two hippocampal shapes. We observe when we have 300 clusters, we have a reasonable  $\sigma = 1.2$  as the average distance between the nearest neighbors is about 2.65.

We did the matching for all the pairs out of ten hippocampus shapes with totally 45 such pairs. We compare three measures for the matching result with different number of clusters in Table 1. The Jensen-Shannon divergence (a special case with equal weights) [13], Hausdorff distance and modified Hausdorff distance [14]. Notice Jensen-Shannon divergence is highly non-linear. When  $p$  and  $q$  are completely independent, namely in our matching case, when the two shapes are completely different,  $D$  has a maximum of  $2 \log 2 = 1.39$ .

From Table 1 we can see that when we have 300 clusters we have the minimum Jensen-Shannon divergence and Hausdorff distance. However, Hausdorff distance is too sensitive to outliers. We also calculated the modified Hausdorff distance as first introduced in [14]. It is easy to see that when the number of clusters increases, the modified Hausdorff distance decreases.

## 5 Discussion

The need for a good 3D point feature matching algorithm arises in various application areas of medical image analysis. To our knowledge, this is one of the first attempts at 3D diffeomorphic point matching. We have demonstrated a 3D joint clustering and diffeomorphism algorithm and applied it to hippocampal point-sets. In the process of careful validation, we investigated the role of the different numbers of clusters in the joint clustering and diffeomorphism optimization process. In the current formulation, we still have a free parameter  $\lambda$  whose role has

to be determined. The immediate future goal is to further address (theoretically and experimentally), the role of free parameters. The same framework can be used for atlas estimation. Finally, once we have a turnkey 3D diffeomorphic feature matching algorithm, we plan to use it for hippocampal shape classification of epilepsy patients [15].

## References

1. Cootes, T., Taylor, C., Cooper, D., Graham, J.: Active Shape Models: Their Training and Application. *Computer Vision and Image Understanding* **61** (1995) 38–59
2. Duta, N., Jain, A.K., Dubuisson-Jolly, M.: Learning 2D shape models. In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. Volume 2. (1999) 8–14
3. Joshi, S., Miller, M.: Landmark matching via large deformation diffeomorphisms. *IEEE Trans. Image Processing* **9** (2000) 1357–1370
4. Camion, V., Younes, L.: Geodesic interpolating splines. In: *Energy Minimization Methods for Computer Vision and Pattern Recognition*. Springer, New York (2001) 513–527
5. Chui, H., Rangarajan, A.: Learning an atlas from unlabeled point-sets. In: *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA)*, IEEE Press (2001) 179–186
6. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding* **89** (2003) 114–141
7. Guo, H., Rangarajan, A., Joshi, S., Younes, L.: A new joint clustering and diffeomorphism estimation algorithm for non-rigid shape matching. *IEEE Workshop on Articulated and Non-rigid motion (ANM)* (2004)
8. Glaunes, J., Trounev, A., Younes, L.: Diffeomorphic matching of distributions: A new approach for unlabelled point-sets and sub-manifolds matching. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 04)*. Volume 2. (2004) 712–718
9. Rosenfeld, A., Plalz, J.L.: Sequential operations in digital picture processing. *Journal of ACM* **13** (1966) 471–494
10. Hathaway, R.: Another interpretation of the EM algorithm for mixture distributions. *Statistics and Probability Letters* **4** (1986) 53–56
11. Rose, K., Gurewitz, E., Fox, G.: Statistical mechanics and phase transitions in clustering. *Physical Review Letters* **65** (1990) 945–948
12. Chui, H., Win, L., Duncan, J., Schultz, R., Rangarajan, A.: A unified non-rigid feature registration method for brain mapping. *Medical Image Analysis* **7** (2003) 112–130
13. Endres, D.M., Schindelin, J.E.: A new metric for probability distributions. *IEEE Transactions on Information Theory* **49** (2003) 1858–1860
14. Dubuisson, M.P., Jain, A.K.: A modified Hausdorff distance for object matching. *ICPR94* (1994) A:566–568
15. Vohra, N., Vemuri, B.C., Rangarajan, A., Gilmore, R.L., Roper, S.N., Leonard, C.M.: Kernel Fisher for shape based classification in epilepsy. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI02)*. (2002) 436–443

Trial\No. Clusters	Jensen-Shannon div.					Hausdorff distance					modified Hausdorff				
	100	200	300	400	500	100	200	300	400	500	100	200	300	400	500
1	0.87	0.31	0.03	0.13	0.21	7.1	7.4	5.7	6.2	7.3	2.8	2.0	1.4	1.2	1.1
2	0.93	0.62	0.47	0.05	0.24	9.3	8.9	7.2	8.3	8.7	3.5	3.1	2.8	2.4	2.3
3	0.76	0.27	0.04	0.16	0.32	7.2	6.1	4.9	5.6	6.4	2.0	1.7	1.4	1.3	1.2
4	0.98	0.52	0.34	0.09	0.45	8.4	7.8	7.2	5.2	6.5	2.7	2.4	2.3	1.7	1.4
5	0.69	0.41	0.14	0.18	0.36	9.6	9.7	8.0	8.4	8.9	3.9	3.6	3.1	2.8	2.7
6	0.57	0.23	0.43	0.78	0.97	9.2	6.3	7.1	7.8	8.6	3.1	2.8	2.5	2.2	2.1
7	0.66	0.21	0.05	0.14	0.30	6.9	5.8	4.4	6.0	7.3	2.4	2.2	2.1	1.7	1.5
8	0.99	0.70	0.25	0.19	0.63	8.9	8.5	7.0	6.4	8.2	3.0	2.6	2.4	2.2	1.9
9	0.85	0.42	0.11	0.68	0.74	9.3	8.0	5.9	7.6	9.1	2.9	2.7	2.3	2.1	1.6
10	0.97	0.62	0.10	0.18	0.55	7.8	7.3	4.7	6.7	8.1	3.2	2.8	2.3	1.8	1.4
11	0.70	0.33	0.06	0.13	0.26	8.7	7.7	5.8	7.4	9.0	2.5	2.1	1.6	1.4	1.2
12	1.02	0.64	0.08	0.44	0.71	9.1	8.3	6.1	7.4	8.6	3.3	3.0	2.5	2.2	2.0
13	0.89	0.54	0.20	0.31	0.65	9.3	8.4	6.5	7.0	8.7	3.6	3.4	3.1	2.4	2.2
14	0.57	0.09	0.15	0.66	0.80	7.4	5.1	5.5	6.8	8.3	3.0	2.7	2.3	2.0	1.8
15	0.88	0.30	0.05	0.29	0.36	8.8	7.1	4.9	6.3	7.8	2.6	2.0	1.5	1.3	1.2
16	0.90	0.75	0.12	0.17	0.44	9.4	9.0	6.1	7.3	8.5	3.2	3.0	2.4	2.1	1.9
17	0.61	0.16	0.28	0.53	0.72	8.6	6.8	7.9	9.0	9.9	3.4	3.5	3.1	2.7	2.4
18	0.91	0.37	0.18	0.40	0.88	9.5	8.2	6.5	7.4	8.0	3.7	3.2	2.9	2.6	2.1
19	1.12	0.80	0.47	0.09	0.28	9.2	7.8	7.2	5.1	6.4	2.8	2.6	2.3	2.2	2.0
20	0.96	0.54	0.33	0.60	0.74	9.6	8.0	7.3	8.7	9.3	3.9	3.5	3.3	3.1	2.7
21	0.65	0.23	0.51	0.78	1.04	8.4	6.1	6.9	7.8	9.5	3.3	3.1	2.8	2.4	2.3
22	0.93	0.46	0.22	0.51	0.68	9.7	8.5	7.0	8.1	9.0	2.9	2.7	2.6	2.3	2.1
23	0.92	0.60	0.28	0.15	0.34	9.6	8.2	7.3	6.5	7.7	2.4	2.1	1.7	1.6	1.5
24	0.80	0.26	0.57	0.69	0.86	7.8	6.6	7.2	8.9	9.6	3.1	2.7	2.5	2.2	1.9
25	1.10	0.62	0.44	0.78	0.97	9.8	7.9	7.6	8.8	9.2	3.8	3.4	3.0	2.8	2.7
26	0.90	0.39	0.05	0.21	0.47	9.0	7.3	5.8	7.0	8.7	2.9	2.4	2.0	1.4	1.2
27	0.58	0.07	0.20	0.56	0.77	7.8	6.0	6.5	7.2	8.3	3.2	2.9	2.5	2.1	1.9
28	0.93	0.51	0.09	0.40	0.63	9.5	8.1	6.1	7.4	8.8	3.0	2.8	2.5	2.1	1.8
29	0.99	0.26	0.18	0.37	0.70	9.7	8.3	6.7	7.0	8.5	3.4	2.7	2.2	1.9	1.7
30	0.60	0.06	0.17	0.54	0.57	7.1	5.6	6.2	7.3	7.8	2.5	2.2	1.8	1.3	1.2
31	0.83	0.19	0.08	0.37	0.76	9.3	6.7	5.8	7.5	8.4	2.7	2.4	2.2	1.6	1.4
32	1.22	0.42	0.57	0.70	0.95	9.9	7.2	7.8	8.5	9.2	3.5	3.0	2.8	2.5	2.3
33	0.80	0.59	0.30	0.86	0.92	8.9	8.0	6.3	7.1	8.7	3.3	3.1	2.6	2.4	2.3
34	0.89	0.76	0.35	0.28	0.67	8.9	8.5	6.7	6.4	7.0	3.0	2.8	2.5	2.2	2.1
35	1.05	0.42	0.37	0.81	1.13	9.6	8.4	7.0	8.1	9.9	3.7	3.4	3.0	2.3	2.0
36	0.92	0.25	0.31	0.60	0.85	8.7	5.9	6.2	7.5	8.4	3.2	2.9	2.5	2.3	2.2
37	0.79	0.35	0.08	0.24	0.40	7.7	6.3	5.4	6.1	7.6	2.4	2.2	1.9	1.6	1.5
38	0.90	0.42	0.16	0.35	0.68	9.5	7.1	5.7	6.6	7.9	2.8	2.4	2.3	2.0	1.8
39	0.86	0.27	0.38	0.50	0.71	9.2	7.3	8.2	8.4	9.0	3.2	3.0	2.7	2.5	2.4
40	0.55	0.04	0.19	0.36	0.67	6.5	5.2	5.8	6.7	7.3	2.0	1.6	1.3	1.1	0.9
41	1.02	0.30	0.47	0.81	0.98	9.4	7.5	7.9	8.6	9.2	3.8	3.2	2.5	2.4	2.3
42	0.43	0.07	0.22	0.56	0.86	7.2	5.8	6.7	7.3	7.9	2.3	1.9	1.6	1.5	1.4
43	0.78	0.56	0.18	0.39	0.61	8.2	7.4	6.0	6.7	7.8	2.9	2.4	2.2	2.0	1.8
44	0.61	0.09	0.25	0.70	0.82	7.1	5.9	6.6	7.5	8.0	2.2	1.9	1.7	1.5	1.4
45	0.44	0.15	0.26	0.53	0.93	7.0	6.5	7.2	7.7	9.1	2.5	2.1	1.9	1.7	1.6

Table 1. Matching metrics of various pairs of shapes