# Fast Computation of Elastic Shape Distance between 2d Objects

Günay Doğan,[1,2] Javier Bernal,[2] Charles R. Hagwood[2]
[1] Theiss Research, [2] National Institute of Standards and Technology (NIST)
{gunay.dogan,javier.bernal,charles.hagwood}@nist.gov

For many problems in biology and medicine, one needs to quantitatively compare shapes of biological objects, e.g., organ boundaries in computational anatomy, cell or colony morphology in cytometry. One might have large databases of such shapes, and may want to cluster, classify or compare such elements. To be able to perform such analyses, one needs the notion of shape distance quantifying dissimilarity of such entities. In this work, we focus on the elastic shape distance of Srivastava et al. [3] for closed planar curves. This provides a flexible and intuitive geodesic distance measure between curve shapes, invariant to translation, scaling, rotation and reparametrization. Computing this distance, however, is computationally expensive. The original algorithm proposed in [3] using dynamic programming (DP) runs in O($N^3$) time, $N$ the number of nodes per curve. In this work, we propose a new fast iterative algorithm to compute the elastic shape distance between shapes of closed planar curves [1, 2]. The asymptotic time complexity of our algorithm is roughly O($N^2$). However, in our experiments, we have observed a subquadratic trend with running times depending on the type of curve data.

Mathematically, the shape distance computation is formulated as a global minimization over triplets of starting points $t_0$ (on the curve), rotations $R$ and reparametrizations $\gamma(t)$. Given planar closed curves $\beta_1$ and $\beta_2$ of unit length, and their shape functions $q_i(t) = \dot{\beta}_i(t)/\|\dot{\beta}_i(t)\|^{1/2}$ the shape distance between them corresponds to the minimum of the following energy:
$E(t_0, \theta, \gamma) \equiv \int_0^1 \|q_1(t) - \sqrt{\dot{\gamma}(t)}R(\theta)q_2(t_0 + \gamma(t))\|^2 dt$.

We propose to minimize this using an alternating approach: We fix $\gamma$ and optimize the energy with respect to $t_0, \theta$ (in $O(N \log N)$ time using FFT-based algorithm from [2]). Then with the optimal $t_0, \theta$ fixed in the energy, we optimize with respect to $\gamma$ (in $O(kN)$ time with nonlinear constrained optimization, $k$ is the number of iterations [1]). We alternate between the two steps with $K$ the number of outer iterations until convergence. The algorithm also involves a $O(\epsilon N^2)$ initialization with our fast *DP*. Thus the overall computational cost of our algorithm is $O(\epsilon N^2 + K(N \log N + kN))$.

We present some of our results demonstrating efficiency gains of our new algorithm compared to the original algorithm in [3] (see the papers [1, 2] for all our results). We test our algorithm on synthetic curves, cell boundary curves, and subsets of Leaf and MPEG7 shape data sets. Our first test is with a synthetic curve to show scalability with respect to $N$. We change the starting point of a limaçon, rotate it, and apply a synthetic reparameterization to it to obtain a second version of it, thus of the same shape and zero distance. The computational cost of the original algorithm becomes very expensive as $N$ increases beyond 256, whereas our new algorithm yields reasonable cost for all $N$ (see Table 1). Finally, we compute the pairwise distance matrix of all the test shape data sets (with $N = 256$). Our algorithm is an order of magnitude faster (see Table 2), even faster for finer samplings, e.g., $N = 512, 1024$.

## References

[1] G. Dogan, J. Bernal, and C. R. Hagwood. A fast algorithm for elastic shape distances between closed planar curves. In *In Proceedings of CVPR*, pages 4222–4230, 2015.

[2] G. Dogan, J. Bernal, and C. R. Hagwood. Fft-based alignment of 2d closed curves with application to elastic shape analysis. In *In Proceedings of the 1st Diff-CV Workshop*, 2015.

[3] A. Srivastava, E. Klassen, S. Joshi, and I. Jermyn. Shape analysis of elastic curves in Euclidean spaces. *IEEE Trans. on PAMI*, 33(7):1415–1428, 2011.

| # of nodes | 64 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|
| Original Algorithm | 1.4 sec | 11 | 92 | 735 | 5917 |
| New Algorithm | 4 | 11 | 19 | 50 | 89 |

Table 1. Timings for computation of the theoretical zero distance between two versions of the same limaçon.

| | Synthetic | Cells | Leaves | MPEG7 |
|---|---|---|---|---|
| Matrix Size | $6^2 = 36$ | $10^2 = 100$ | $75^2K$ | $100^2$ |
| Orig. Algo. | 1 hr | 2.5 hrs | 129 hrs | 240 hrs |
| New Algo. | 12 min | 4.5 min | 12.5 hrs | 38.5 hrs |

Table 2. Timings for distance matrices obtained by computing shape distances for all curve pairs ($N = 256$).