INSTANCE-BASED GENERATIVE BIOLOGICAL SHAPE MODELING

Tao Peng[1], *Wei Wang*[1], *Gustavo K. Rohde*[1] *and Robert F. Murphy*[1,2,3]

[1]Center for Bioimage Informatics and Department of Biomedical Engineering
[2]Departments of Biological Sciences and Machine Learning, Carnegie Mellon University, U. S. A.
[3]External Fellow, Freiburg Institute for Advanced Studies, University of Freiburg, Germany

ABSTRACT

Biological shape modeling is an essential task that is required for systems biology efforts to simulate complex cell behaviors. Statistical learning methods have been used to build generative shape models based on reconstructive shape parameters extracted from microscope image collections. However, such parametric modeling approaches are usually limited to simple shapes and easily-modeled parameter distributions. Moreover, to maximize the reconstruction accuracy, significant effort is required to design models for specific datasets or patterns. We have therefore developed an instance-based approach to model biological shapes within a shape space built upon diffeomorphic measurement. We also designed a recursive interpolation algorithm to probabilistically synthesize new shape instances using the shape space model and the original instances. The method is quite generalizable and therefore can be applied to most nuclear, cell and protein object shapes, in both 2D and 3D.

Index Terms— Generative models, nuclear shape, microscopy, machine learning, shape interpolation, location proteomics

1. INTRODUCTION

Proteins function in different cellular or subcellular compartments to form a living cell system. In systems biology, modeling this complex system from different aspects and at various levels is anticipated to contribute to a final understanding of cell mechanisms [1, 2]. Quantitative, predictive compartment models which capture the spatial properties of subcellular structures is a basic building block for modeling of complex cell behaviors. With high-throughput microscopes capable of generating large numbers of high-resolution images, automated learning methods will be needed to build predictive compartment shape models.

We have previously described approaches for learning generative models of important cell compartments and protein localization [3]. In this work, nuclei were parameterized into B-spline coefficients of a medial axis curve and width from the media axis, and cell boundaries were parameterized into a compressed vector of the ratio of cell boundary position to nuclear boundary position in a polar coordinate system center on the nucleus. Statistical learning was performed on sets of parameters extracted from HeLa cell images to fit proper distributions to them. Algorithms to synthesize new shapes from sampled parameters were described for different protein distributions.

In such parametric modeling methods, the shape space is represented by probabilistic distributions of parameters. Parametric methods usually induce very concise models, but they also have obvious shortages. First, parametric description converts the shape into finite set of parameters, which causes information loss during the simplification process. Second, shape parameter distributions are usually arbitrarily fitted with frequently-used distributions by experience or histogram inspection, which is often not accurate enough in the real shape space. Third, complex shapes (especially 3D shapes) are often not easy to parameterize.

As first described by Yang et al [4], non-rigid registration methods are a valuable alternative to parametric methods for analyzing nuclear shape. Rohde et al [5, 6] proposed a registration-based measurement of distance between deformable shapes using the large deformation diffeomorphic metric mapping (LDDMM) framework [7] combined with Multidimensional scaling (MDS) [8] to reconstruct the nuclear shape space. The work we describe here extends this approach to a generative framework. To overcome the drawbacks of parametric methods, we propose an instance-based approach to model the shape space using kernel density estimation and a method to synthesize new shapes from it. We illustrate the method using 2D nuclear shapes but it is equally applicable to higher dimensional images.

2. METHOD

2.1. Shape space construction

Under the framework of computational anatomy [9], a set of shapes can be related by a group of diffeomorphic transformations, one-to-one smooth differentiable invertible mappings. Let Φ represent a group of diffeomorphisms, over a bounded domain Ω. A deformed image can be expressed as $I(\phi(x)), x \in \Omega$, then a series of images $I_1, I_2, \ldots$ can be generated by a template I and a group of diffeomorphisms $\{I(\phi_i)|\phi_i \in \Phi\}$, thus a shape manifold or ”orbit” can be built. For images I_1 and I_2, we could imagine the transformation that maps image I_1 to I_2 as the endpoint of an ordinary differential equation $\frac{\partial \phi(x,t)}{\partial t} = v(\phi(x,t),t)$, subject to $\phi(x,0) = x$ and $I_1(\phi(x,1)) = I_2(x)$. Then a distance metric can be defined on the group of diffeomorphisms (shape manifold), which can be understood as the incremental effort to transform one image to another:

$$d(I_1, I_2) = \inf_v \int_0^1 \|Lv(\cdot,t)\|dt \tag{1}$$

where $\|\cdot\|$ means one standard L_2 norm on the velocity field $v(x,t)$, and L just represents a linear differential operator, for example $L = (\nabla^2 + \lambda I)$.

This distance can be computed by solving the following optimization problem:

$$\inf_v \int_0^1 \|v(\cdot,t)\|_V^2 dt \tag{2}$$

subject to: $I_1(\phi(x,0)) = I_1(x)$ and $I_1(\phi(x,1)) = I_2(x), x \in \Omega$

In this paper, however, we use the more computationally efficient greedy algorithm [10], which looks for the locally-in-time optima instead of the global optima. In this method, we assume the velocity is constant in each time step Δt and look for the locally optimal velocity field by

$$L^* L v_t + b_t = 0; \tag{3}$$

with $b_t = -(I_0(\phi(x,t) - I_1(x)))\nabla I_0(\phi(x,t))$. After calculating the velocity in each time step, we can update the deformation field based on Eulerian reference frame via $\phi(x, k+1) = \phi(x + \epsilon v(x,k), k)$. We use the symmetric, inverse consistent version described in [11].

After we compute the diffeomorphic distances between all image pairs, a mutual distance matrix can be generated by: $D_{m,n} = d(I_m, I_n)$, with $d(I_m, I_n)$ representing the distance computed by greedy algorithm. Then multidimensional scaling can be applied on D to find a group of low dimensional "Euclidean" coordinates, or shape coordinates, that preserve the pairwise distances [8, 12]. The goal of using MDS is to unfold the shape manifold, built by the group of diffeomorphisms, and represent the manifold in a low dimensional "Euclidean" space. Each coordinate $\mathbf{x}_k$ in the "Euclidean" space with reduced dimension d corresponds to a specific shape of original dataset.

2.2. Shape space distribution learning

Shape coordinates reside in a high dimension with a complex density distribution. We therefore apply the non-parametric method kernel density estimation [13] to approximate the shape space. To simplify the problem, we use a spherical Gaussian kernel since MDS normalize the coordinates in multi-dimensional Cartesian space. The *pdf* is formulated as

$$\hat{p}_h(\mathbf{x}) \quad = \quad \frac{1}{n}\sum_{j=1}^{n} \frac{1}{\left(\sqrt{2\pi}h\right)^d} \exp\left(\sum_{k=1}^{d} \frac{(x_k - x_k^j)^2}{2h^2}\right) \tag{4}$$

Optimal bandwidth is selected to minimize the Kullback-Leibler divergence [13] between approximate density $\hat{p}_h(\mathbf{x})$ and the true density $p(\mathbf{x})$.

$$D_{KL}(p, \hat{p}_h) \quad = \quad \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{\hat{p}_h(\mathbf{x})} d\mathbf{x} \tag{5}$$

which is equivalent to maximizing

$$\int \log[\hat{p}_h(\mathbf{x})] p(\mathbf{x}) d\mathbf{x} \quad = \quad E \log[\hat{p}_h(\mathbf{x})] \tag{6}$$

The expectation of the logarithmic likelihood can be approximated by leave-one-out cross-validation for fixed bandwidth h [14].

$$L(\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n | h) \quad = \quad \sum_{i=1}^{n} \log \hat{p}_{h,i}(\mathbf{x}_i) \tag{7}$$

where $\hat{p}_{h,i}$ is the leave-one-out likelihood estimator

$$\hat{p}_{h,i}(\mathbf{x}) = \frac{1}{n-1}\sum_{j\neq i} \frac{1}{\left(\sqrt{2\pi}h\right)^d} \exp\left(\sum_{k=1}^{d} \frac{(x_k^i - x_k^j)^2}{2h^2}\right) \tag{8}$$

Full optimal bandwidth matrix can be estimated using a method involving Markov chain Monte Carlo [14].

2.3. Shape space triangulation

The purpose of triangulation is to partition the shape space into triangular mesh grids in order to interpolate un-sampled points in the shape space. Given a set of shape points with reduced dimension d, Delaunay triangulation [15] is used to triangulate the space, yielding a set of d dimensional triangles with vertices of $d+1$ points from the original point set. A point is located in the triangulated space by searching the enclosing Delaunay triangle. Both triangulation and triangle search algorithms are implemented in the computational geometry toolbox $qhull$ [16].

2.4. Shape sampling and interpolation

With the original shapes and probabilistic description of shape space we can generate new shapes which also reside in the true shape space and are statistically meaningful, by deforming existing shapes located nearby in the shape space. First, sample a point from the distribution learned in 2.2 in the shape space and locate it in the triangulated space by finding the $d+1$ vertices of the d-dimensional triangle which encloses it. Since diffeomorphisms can be calculated only between pairs of shapes, recursive projection and deformation algorithm is then performed to reach the shape represented by the sampled point, according to the following algorithm.

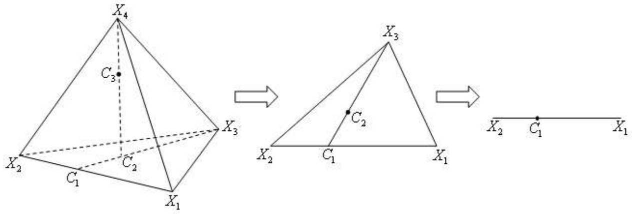


Fig. 1. A demonstration of recursive projection and deformation algorithm in a 3D shape space.

1. Start from dimension *d*, in a k dimensional triangle with vertices $X_1, \ldots, X_{k+1}$, "project" X_{k+1} to the plane defined by $X_1, \ldots, X_k$ via C_k, that is, find the intersection C_{k-1} of radial $\overrightarrow{X_{k+1}C_k}$ and plane $X_1, \ldots, X_k$. Define ratio parameter λ_k: $\overrightarrow{C_k C_{k-1}} = \lambda_k \overrightarrow{X_{k+1}C_k}$ and C_{k-1} that satisfies

   $$|\mathbf{c}_{k-1} - \mathbf{x}_1, \mathbf{x}_2 - \mathbf{x}_1, \ldots, \mathbf{x}_k - \mathbf{x}_1| \quad = \quad 0 \tag{9}$$

   The iterative projection is solved by

   $$\begin{cases} \lambda_k & = \frac{|\mathbf{c}_k - \mathbf{x}_1, \mathbf{x}_2 - \mathbf{x}_1, \ldots, \mathbf{x}_k - \mathbf{x}_1|}{|\mathbf{c}_k - \mathbf{x}_{k+1}, \mathbf{x}_2 - \mathbf{x}_1, \ldots, \mathbf{x}_k - \mathbf{x}_1|} \\ \mathbf{c}_{k-1} & = (1+\lambda_k)\mathbf{c}_k - \lambda_k \mathbf{x}_{k+1} \end{cases} \tag{10}$$

2. Remove the null dimension of set $\{\mathbf{c}_{k-1}, \mathbf{x}_1, \ldots, \mathbf{x}_k\}$ since they are coplanar in the k-dimensional space. This is achieved by a coordinate conversion: projecting all coordinates onto the first $k-1$ principal components of the covariance matrix.

   $$[\mathbf{c}_{k-1}, \mathbf{x}_1, \ldots, \mathbf{x}_k] \leftarrow [\mathbf{P}_1, \ldots, \mathbf{P}_{k-1}]^T [\mathbf{c}_{k-1}, \mathbf{x}_1, \ldots, \mathbf{x}_k]$$

3. On $k-1$ dimensional coordinate set $\{\mathbf{c}_{k-1}, \mathbf{x}_1, \ldots, \mathbf{x}_k\}$ repeat step 1 and 2. Each time a ratio parameter λ_k is calculated by equation (7) until in the end only c_1, x_1, x_2 are left, which are all scalars, and $\lambda_1 = (x_2 - c_1)/(c_1 - x_1)$.

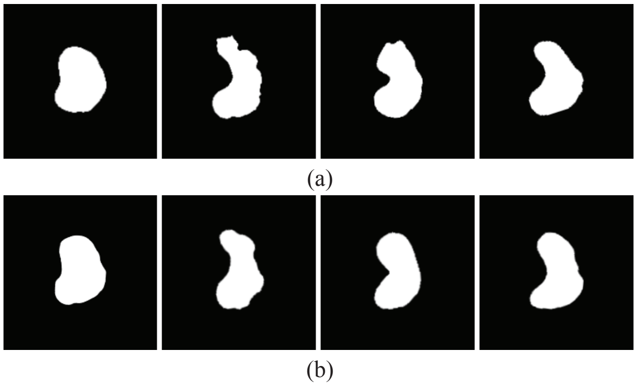
4. Knowing $\lambda_1, \ldots, \lambda_k$, starting from λ_1 and $I(X_1)$, $I(X_2)$, the shapes represented by X_1, X_2, generate an intermediate shape $I(C_k)$ which corresponds to the coordinate $\mathbf{c}_k$ in the shape space by deforming shape $I(X_{k+1})$ to shape $I(C_{k-1})$ and capturing the shape at $1/(1+\lambda_k)$ along the deformation path ($C_0 = X_1$).
5. Repeat step 4 until the destination shape $I(C_d)$ is generated.
6. Output the shape $I(C_d)$ as the synthesized shape image.

3. RESULTS

To build a complete instance-based generative shape model, we used previously acquired two dimensional images of Hela cell nuclei expressing lamin modifications(a total of $n = 160$ nuclei images with relatively complex shapes), and pre-process images by removing variation in overall orientation, position and size, as described in [5].

3.1. Algorithm validation

We assume shapes generated from our recursive interpolation algorithm still reside in the shape space. To validate this assumption, we use leave-one-out re-interpolation. After construction of the shape space, for every shape point which does not reside on the convex hull enclosing all points, we remove it and use the remaining $n-1$ points to construct the triangulated mesh. We locate the excluded point in its enclosing triangular simplex and regenerate the shape it stands for using our recursive interpolation algorithm. Comparison between original shape and interpolated shape shows little difference, see Figure 2.


(a)

(b)

Fig. 2. Comparison between original shape and interpolated shape. (a) Original nuclear shapes. (b) Corresponding shapes interpolated from models built without them.

3.2. Shape space dimension

By registering pairs of binary nuclear images, we calculate the diffeomorphic distance matrix D ($n \times n$) and perform MDS on it. Residue variances (defined to be $1 - R^2(\tilde{D}, D)$, $\tilde{D}(d \times d)$ – pairwise distance matrix in reduced dimension space, which is an approximation of D; R – correlation coefficient between the entries of both matrices) at each reconstruction level are calculated to reflect distance preserving extent after dimension reduction. The intrinsic dimensionality of the data is estimated by looking for the "elbows" at which the residual variance ceases to decrease significantly with added dimension [6, 17]. Figure 3 shows the residue variance as a function of dimension used in distance matrix reconstruction and we conclude that $d = 6$ is a sufficiently descriptive dimension of the space.

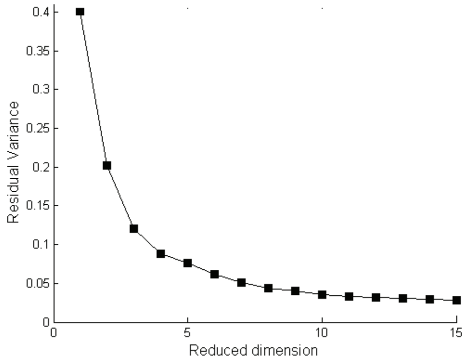


Fig. 3. Residual variance of distance reconstruction as a function of number of dimension in shape space construction using LDDMM-MDS.

3.3. Bandwith selection

In fitting shape space distribution with kernel density estimation, we run cross-validation on bandwidth h varying from 10^{-2} to 1 by steps of 10^{-3}. Figure 4 shows the peak part of the cross-validation score $\hat{E}\log[\hat{p}_h(\mathbf{x})]$ as a function of bandwidth h. Optimal bandwidth is reached at $h = 0.059$ which eventually minimizes the KL-divergence between the true and fitted distributions.

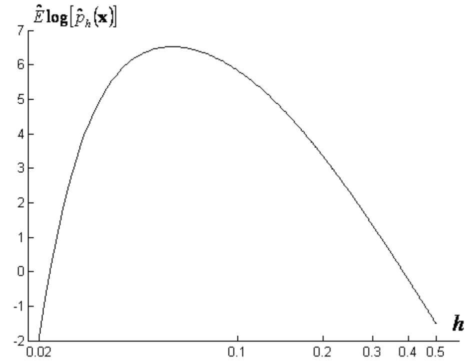


Fig. 4. Approximate expectation of the logarithmic likelihood as a function of bandwidth h.

3.4. Synthesized shape

We sample random vectors according to distribution with optimal bandwidth. Figure 5 shows synthesized nuclear shapes using the recursive interpolation algorithm. Since shapes synthesized from grid interpolation only reside inside the convex hull enclosing all instances, while the "point cloud" like distribution covers infinite space, we re-sample if a point outside the convex hull is generated.

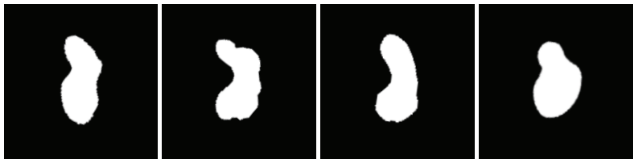


Fig. 5. Examples of synthesized nuclear shapes.

4. CONCLUSION AND DISCUSSION

We propose an instance-based generative modeling method for biological shapes. The model contains three parts: original shape instances, shape coordinate representations in shape space and distribution function of shape space. Under this framework, synthesized shapes generated by deforming real shapes are reasonable instances of the family. More important, LDDMM is not restricted by shape dimension or complexity, which makes the model highly generalizable.

Our method of building generative models for biological shapes can be applied to both 2D and 3D images. Since there is no assumption on the shapes to be modeled, it allows very complicated shapes, such as shapes with many invaginations. Moreover, applications are not limited to static shape modeling. Given time series cell images, we can build non-parametric predictive cell deforming (crawling, spreading etc.) models for live cell simulation. The method can also model composite shapes (piecewise constant images), for example, nested shapes of cell membrane combined with nuclei, revealing the spatial relations of different shape components.

Both non-parametric density estimation and neighboring interpolation requires as many original data as possible. During our validation studies, we noticed that some regenerated shapes did not match the original image well. This is presumably because of an overly sparse distribution of shapes around it, in which case the interpolation is done using shapes faraway in shape space. Determining criteria for the number of images required to partition the space finely enough for a desired level of interpolation accuracy remains a theoretical problem to be investigated. Additionally, all shapes we are able to generate this way must be inside the convex combination (in the shape manifold) of existing points (instances). This limitation is also presumably addressed by collecting large numbers of images, a task greatly simplified by automated microscopes.

5. ACKNOWLEDGEMENT

We thank Karl Rohr for helpful discussions. The research described here was supported in part by NSF grant EF-0331657 and NIH grant GM075205.

6. REFERENCES

[1] T. Ideker, T. Galitski, and L. Hood, “A new approach to decoding life,” *Annu. Rev. Genomics Hum. Genet.*, vol. 2, pp. 343–372, 2001.

[2] H. Kitano, “Computational systems biology,” *Nature*, vol. 420, pp. 206–210, 2002.

[3] T. Zhao and R. F. Murphy, “Automated learning of generative models for subcellular location: building blocks for systems biology,” *Cytometry Part A*, vol. 71A, pp. 978–990, 2007.

[4] S. Yang, D. Khler, K. Teller, T. Cremer, P. Le Baccon, E. Heard, R. Eils, and K. Rohr, “Non-rigid registration of 3d multi-channel microscopy images of cell nuclei,” in *LNCS*, vol. 4190, pp. 907–914. Springer Berlin Heidelberg, 2006.

[5] G. K. Rohde, A. J. S. Ribeiro, K. N. Dahl, and R. F. Murphy, “Deformation-based nuclear morphometry: capturing nuclear shape variation in HeLa cells.,” *Cytometry*, vol. 73A, pp. 341–350, 2008.

[6] G. K. Rohde, W. Wang, T. Peng, and R. F. Murphy, “Deformation-based nonlinear dimension reduction: applications to nuclear morphometry,” in *Proc. IEEE Int. Symp. Biomed. Imaging*, 2008, pp. 500–503.

[7] M. F. Beg, M. I. Miller, A. Trouve, and L. Younes, “Computing large deformation metric mappings via geodesic flows of diffeomorphisms,” *Intern. J. Comp. Vis.*, vol. 61, no. 2, pp. 139–157, 2005.

[8] T. Cox and M. Cox, *Multidimensional Scaling*, Chapman and Hall, 1994.

[9] U. Grenander and M. I. Miller, “Computational anatomy: An emerging discipline,” *Quart. Appl. Math.*, vol. 56, pp. 617–694, 1998.

[10] G. E. Christensen, R. D. Rabbitt, and M. I. Miller, “Deformable templates using large deformation kinematics,” *IEEE Trans. Imag. Proc.*, vol. 5, pp. 1435–1447, 1996.

[11] E. Klassen, A. Srivastava, W. Mio, and S. Joshi, “Analysis of planar shapes using geodesic paths on shape spaces,” *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 26, no. 3, pp. 372–383, 2004.

[12] L. Saul, K. Weinberger, F. Sha, J. Ham, and D. Lee, “Spectral methods for dimensionality reduction,” in *Semisupervised Learning*. MIT Press, 2006.

[13] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*, Springer Texts in Statistics, 2005.

[14] X. Zhang, M. L. King, and R. L. Hyndman, “Bandwidth selection for multivariate kernel density estimation using mcmc,” *Comput. Stat. Dat. An.*, vol. 50, no. 11, pp. 3009–3031, 2006.

[15] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, Springer-Verlag, 2008.

[16] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, “The quickhull algorithm for convex hulls,” *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.

[17] J. B. Tenembaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, pp. 2319–2323, 2000.