Published in final edited form as:
Methods Cell Biol. 2012 ; 110: 179–193. doi:10.1016/B978-0-12-388403-9.00007-2.

CellOrganizer: Image-derived Models of Subcellular Organization and Protein Distribution

Robert F. Murphy
Lane Center for Computational Biology, Carnegie Mellon University, and Freiburg Institute for Advanced Studies, University of Freiburg

Abstract

This chapter describes approaches for learning models of subcellular organization from images. The primary utility of these models is expected to be from incorporation into complex simulations of cell behaviors. Most current cell simulations do not consider spatial organization of proteins at all, or treat each organelle type as a single, idealized compartment. The ability to build generative models for all proteins in a proteome and use them for spatially accurate simulations is expected to improve the accuracy of models of cell behaviors. A second use, of potentially equal importance, is expected to be in testing and comparing software for analyzing cell images. The complexity and sophistication of algorithms used in cell image-based screens and assays (variously referred to as high content screening, high content analysis, or high throughput microscopy) is continuously increasing, and generative models can be used to produce images for testing these algorithms in which the expected answer is known.

Introduction

As traditional reductionist paradigms of biomedical research increasingly give way to systems approaches, the need to build predictive models that synthesize large amounts of information from potentially diverse sources is becoming critical. Most such current models take the form of transcriptional regulatory networks, protein-protein interaction maps, or biochemical reaction simulations. These typically do not consider spatial organization of cells or tissues. Important advances came with systems such as MCell (1), allowed models to be constructed using mesh representations of cells built from electron microscope images, and the Virtual Cell (2) which allowed appropriately processed images to provide surface area and volume for its compartmental models. Ontologies such as the Genome Ontology (GO) can be used to describe protein attributes, including location, primarily at a major organelle level. Such assignments can also be used to create compartmental models (e.g., http://biologicalnetworks.net/tutorials) However, compartmental models suffer from some important limitations, in that they treat all molecules within each compartment as being homogenously distributed, and they do not allow appearance, disappearance, fission or fusion of compartments.

Given the energy expended by cells to maintain their subcellular organization, and the many defects that are associated with alterations in it, models that do not accurately reflect

---

subcellular organization are unlikely to perform satisfactorily at predicting complex cell behaviors or how they respond to changes in conditions. There is therefore a need for computational models that accurately represent the number, size, shape and positions of subcellular structures, the spatial relationships between different structures, and how proteins (and other molecules) are distributed between them (3, 4). In addition, there is a need for a mechanism for representing how all of these vary within a population of cells of a single cell type, within a single cell type under different conditions, among different cell types, and among different organisms. Such models can not only *capture* cell behavior but can also be an important step in *understanding* that behavior, since, for example, a sufficiently detailed model helps distinguish aspects that are conserved and presumably necessary from those that are highly variable and potentially not necessary.

In considering how to build such models, we can distinguish *descriptive* models, which allow one to recognize what state a particular cell is in, from *generative* models, which can also synthesize new examples of cells in particular states. We can also distinguish *theoretical* or *conceptual* models, which posit a particular structure based on a generalized understanding, from *data-driven* models that are learned from data and capture both general behavior and variation in that behavior.

My focus in this chapter will be primarily on methods developed in my group that have been used to learn generative models of cell organization and protein distribution from two-dimensional and three-dimensional fluorescence microscope images (5–11). We have recently grouped these methods as part of the open source CellOrganizer project (http://cellorganizer.org) which includes collaborations with a number of investigators studying particular cell systems.

Components of a model of subcellular organization and protein distribution

While there are a number of ways to break down the tasks necessary for creating such models, we can distinguish at least three major components of a model of the distribution of proteins within cells of a given type under a given condition:

- A model of subcellular organization, including distributions of the number, size, shape and position of each subcellular structure, any of which may be conditional on the model(s) for other structures
- A model representing the probability that a cell of a given type will contain a certain number of molecules of a given protein, the expected fraction of those molecules in each subcellular structure, and a measure of the variation in that fraction from cell to cell
- A model of how each protein is distributed within each structure, which may consist of a self-organizing model that specifies only the affinities between pairs of proteins within each structure

Higher order models can then be built to specify how any of these models change over time and condition: for example, during the cell cycle, in the presence of perturbagens, for cells expressing mutations, or for different cell types.

I will focus below on work on the first two types of components.

Models of subcellular organization

At a conceptual level, the most complete model of subcellular organization is probably the GO (Genome Ontology) Cellular Component ontology (12). A significant effort has been made to capture the vast majority of terms used to describe subcellular structures. The terms in this ontology can be assigned to proteins in order to represent the results of experimental or computational analyses. The advantage of this approach is precisely its disadvantage: general terms such as “mitochondria” can be associated with a protein while leaving many questions about what mitochondria are unanswered. However, to be useful for spatially realistic modeling, ontology terms must be associated with a representation of each organelle’s number, structure and distribution within cells. Currently, such representations are abstract and implicit rather than concrete and they often leave unspecified how the organelle would look in different cell types. For example, the abstract concept of a mitochondrion is well-understood by biologists but most would be hard pressed to accurately describe how mitochondria vary in number, size, shape and distribution from cell type to cell type or organism to organism.

In building generative models, we refer to an individual image, stack or movie to be an *instance* drawn from an underlying model, whether an actual image or a synthetic image. These instances are considered to have been generated by particular *values* for the *parameters* of the model. The model is *generative* if it captures how parameters values can be chosen for new instances.

A critical concept in creating models of subcellular organization is the conditional relationships that exist among different components. This is easily illustrated by considering the task of building generative models of nuclear and cell shape (i.e., the positions of the nuclear and plasma membranes). We could build one generative model from many examples of nuclear shapes, and build another generative model from many examples of cell shapes. If we with to synthesize a new example of a cell containing a nucleus, we can imagine drawing a random example of a nuclear shape from the first model, and drawing a random example of a cell shape from the second. However, there is nothing that would prevent the example nuclear shape from being too wide to fit inside the example cell shape, and nothing to tell us where within the cell shape to put the nuclear shape. We must therefore connect the generation processes, which we do by making the models dependent, or *conditional*, upon each other. In our work, we have chosen to make the cell shape model conditional upon the nuclear shape. As we will see below, this means that during the learning process the relationship between the shapes is captured, and during the generation process, an example nuclear shape is first generated and used to generate an appropriate cell shape.[1] An alternative is to make the models *joint*, in which we learn simultaneously a model for both shapes.

---

[1] Of course, we might also have chosen to make the nuclear shape conditional upon the cell shape. Which order is better will need to be determined by future work.

Another major consideration is whether to make the models *parametric*, in which the values of model parameters explicitly describe various aspects of the sizes and shapes of cell components, or *non-parametric*, in which sizes and shapes are implicitly described by the relationships between examples. This distinction will be made clearer in the next sections where we consider models of cell components and how they can be made conditional upon each other. In each case, we will consider

- the inputs necessary for training the model,
- the means of assessing how adequately the model describes the data,
- what types of outputs that the model can generate.

Nuclear and cell shape models

Nuclear shape - Medial axis models—Nuclear shape is often represented in theoretical models as a sphere or more generally an ellipsoid. Examination of only a few images of some cell types (especially adherent cultured cells) reveals how inaccurate this model can be. A somewhat more accurate model can be learned directly from images (5) using a *medial axis* approach (13). As illustrated in Fig. 1, medial axis construction typically begins by first orienting all nuclear shapes (instances) so that their major axes point in the same direction). Each instance is then represented by the position of a curve bisecting the shape perpendicular to the major axis, and by the width at each position along that curve. These curves can be fit using splines, such that a set of 11 spline coefficients describes each instance. The distribution(s) of these parameters over many instances can then be learned. In this case, two multivariate Gaussian distributions, one for the medial axis position and one for the width, were shown to provide a good representation of nuclear shape in two-dimensional images (5). Sampling from these distributions using a random number generator can be done in order to create synthetic examples from the learned model.

Nuclear shape - Cylindrical spline surface model—For three-dimensional images, the medial axis method can result in an oversimplified shape model. An alternative is to convert the nuclear shape to cylindrical coordinates and then fit a periodic spline surface (11). This is illustrated in Fig. 2. In this case, there is one parameter for the nuclear height and 32 parameters for the coefficients of the spline surface. For a collection of 3D images of HeLa cells, these parameters were also shown to be well-represented by a multivariate Gaussian distribution. As before, parameter values can be randomly sampled from this distribution to generate new nuclear shape instances.

Nuclear shape – Large deformation diffeomorphic metric mapping—These parametric models of nuclear shape have two significant advantages: they can be computed fairly quickly and the parameters (and parameter distributions) can be stored compactly. However, they make assumptions about the characteristics of nuclear shape that need to be captured (e.g., that small bumps can be ignored) and do not handle well many concave or branched shapes. An important alternative therefore is to use non-parametric models such as the large deformation diffeomorphic metric mapping (LDDMM) framework developed by Miller and colleagues (14). In this framework, shape is represented implicitly by measuring differences between pairs of shape instances (see Fig. 3). The distance matrix is then used to create a shape space in which similar shapes are near each other. This approach has been demonstrated to provide an excellent representation of nuclear shape in HeLa cells (6), and the method can be applied to 2D, 3D or 4D images. This power comes at a price: saving the shape model requires storing both the distance matrix (or the shape space) and the example images used to create it. Generating new shape instances can be achieved by interpolating between the original examples (8), but this can be computationally expensive.

An important additional use of non-rigid registration methods is to identify positions *within* nuclei. In an exciting example, the positions of different chromosome regions have been mapped to a common frame of reference using a multiresolution non-rigid registration approach (15). Potentially, position mapping could be combined with modeling of the nuclear shape itself as described above.

Cell shape – Circular and spherical coordinate ratiometric models—Cell shape can also be represented using diffeomorphic methods, using exactly the same approach as used for nuclei. This is appropriate when modeling only the cell shape is desired, but if nuclei are to be included, as discussed above, the nuclear and cell shape models must be conditionally related. This can be achieved using diffeomorphic methods by creating indexed images in which pixels/voxels that are part of the background have one value (e.g., 0), pixels/voxels in the nucleus have a second value (e.g., 1), and pixels/voxels inside the cell but not in the nucleus have a third value. Finding the distance between such indexed images is a bit more computationally demanding.

To create more compact conditional models of cell shape, a simple approach can be used. For 2D images, the coordinates of the cell and nuclear boundary are first mapped to polar coordinates, and then the ratio between the two is calculated for a fixed number of angles (e.g., every degree over 360 degrees) (5). For 3D images, these ratios are calculated for each 2D slice (11). The model is then simplified by keeping only a certain number of principal components (for HeLa cells, 10 components were used for 2D images and 25 for 3D images). The distributions of these components have been shown to follow a multivariate Gaussian, providing a very compact conditional model. To generate instances from the model, a nuclear shape is first generated using one of the methods above, principal component coefficients are chosen using random numbers and converted to the cell/nuclear ratio as a function of angle, and then these ratios are multiplied by the corresponding position on the synthetic nuclear boundary to generate the synthetic cell boundary.

Models of vesicular organelles: shape

Gaussian object models—Many vesicular organelles, such as lysosomes, show a roughly spherical shape in both electron microscope and fluorescent microscope images. Such shapes can be easily modeled if the organelles are well resolved from each other in images. However, vesicular organelles are frequently found quite close to each other, and they can appear to overlap when imaged in two dimensions. Furthermore, sampling noise may make them appear irregularly shaped. One approach to this problem is to assume that the organelles are all spherical (or ellipsoidal) and try to estimate what configuration of organelles gave rise to a particular cell image. This can be done by thresholding the image of an organelle marker to identify connected components that may consist of more than one organelle. As shown in Fig. 4, image processing and parameter estimation can then be used to find the positions and sizes of the individual organelles. A statistical model of the distribution of the number of objects per cell, and the distribution of the Gaussian parameters (covariance matrix) can then be constructed. This method can be used for both 2D and 3D images, although distinguishing different organelles is easier in 3D images.

Outline models—More accurate models can be obtained using methods that seek to estimate the position of the *outline* of vesicular organelles. For example, piece-wise linear closed splines have been used to describe the shape of endosomes (16). Such methods could be combined with eigenshape or diffeomorphic methods to create generative models.

Object type models—Even more detailed (but not necessarily more accurate!) models can be obtained by finding all objects in a large set of cell images and clustering them to identify distinct object types. This approach has been applied to a large collection of HeLa cell images, and the resulting object types were found to enable recognition of different subcellular patterns (17). As discussed below, this approach has been used to estimate the amount of a given probe in different organelles. However, it could also be used as part of a generative model by modeling the number and shape of each object type.

Models of vesicular organelles: position

Regardless of which method is used for estimating object number and shape, a model of the *position* of each object within the cell is also needed. This clearly needs to be conditional upon the cell and nuclear shape model. One simple approach is to represent the position of each observed object in a normalized polar or spherical coordinate system (depending on whether the image is 2D or 3D). To do this, the distance of the center of each object from the nuclear boundary is expressed as a fraction of the sum of the distance from the nuclear boundary and the distance from the cell boundary (this normalized distance can be negative if the object is inside the nucleus). The angle (or angles) of the object’s center to the center of the nucleus are also found. An empirical probability density map is then formed by tabulating these positions for many object from many cells. To use this model to synthesize an image, a number of objects is drawn from the appropriate distribution, a size and shape is drawn for each (depending on which shape model is being used), and distances and angles are chosen randomly according to the density map for each, and converted to actual coordinates for particular cell and nuclear shape instances.

Models of cytoskeletal structures

The methods described above for building nuclear, cell and organelle models all make direct estimates of model parameters from real images. While decomposing a cluster of organelles into individual objects may be difficult, it is usually possible. Some organelles or structure are much more difficult to resolve into individual elements. For example, 2D or 3D images of the distribution of tubulin by either wide-field or confocal microscopy typically show individual microtubules at the cell periphery but a tangle of crossing microtubules near the centrosome. Estimating the number of individual microtubules or their individual paths is nearly impossible. One solution is to use specialized microscope methods, such as speckle microscopy, to resolve individual microtubules. An alternative is to use inverse modeling methods to try to estimate the parameters of a microtubule model, as illustrated in Fig. 5a. A generative model is created and then instances of that model are created for many different sets of parameters. These instances are compared to a real image and the parameters corresponding to the best match are chosen. This approach has been used to study kinetochore-microtubule dynamics (18). We have used a similar approach to build a generative model of microtubules in interphase HeLa cells and 3T3 cells (9, 10). An example of a synthetic microtubule distribution is shown in Fig. 5b.

Putting it all together

Once the various components of a model have been created, it is a simple matter to construct synthetic cell instances. Figs. 6 and 7 show *idealized* images (with no blurring or noise) for instances created from 2D or 3D models, respectively. As discussed below, these idealized images can also be used to estimate how that cell might look if imaged in a particular microscope.

Protein distributions across subcellular structures

The models described above capture how cellular organelles are arranged within a cell, but do not address the critical question of how the tens of thousands of proteins in each cell are distributed among these organelles. Images, especially fluorescence microscope images, can be a major source of information on the subcellular distributions of proteins, and, as mentioned above, may be used directly in cell simulations. The feasibility of using automated pattern recognition approaches to recognize the subcellular patterns of proteins that localize primarily to one organelle has been well demonstrated (for reviews see (19–21)). However, many proteins are found to varying extents in more than one organelle, and therefore a means of determing that distribution is needed.

Boolean vectors: GO terms

Some information about protein subcellular location can be obtained from protein databases, which have at least some GO terms associated with most proteins. However, there are a number of limiations of these annotations, most of which derive from the absence of enough experimentatl data. For example, these databases do not attempt to capture changes in GO terms for different conditions or cell types or distinguish between subcellular locations of different splice isoforms. Nonetheless, when no other information is available, GO terms can be represented as a Boolean vector describing whether a particular protein is or is not found in each organelle.

Dirichlet distributions: Pattern unmixing

What is really need for accurate modeling of a protein is a Dirichlet distribution – a probability distribution (that sums to one) for each molecule of that protein over the different organelles. We can convert the Boolean vector for a particular protein derived from GO terms into a Dirichlet distribution by dividing by the number of organelles it is thought to be found in. This assumes, in the absence of any other information, that it is equally likely to be in each of them. A much better alternative is to try to estimate the amount of a given protein in each organelle or structure. To do this, we define a set of *fundamental patterns* to be a set from which all composite patterns can be constructed. This might correspond to the set of all organelle patterns, but, depending on the extent to which they are distinct, might contain multiple subpatterns for a given organelle. For example, protein distributions in the nucleus have been divided into at least eight nuclear subdomains (22). For a collection of images of a particular protein, we seek to find the Dirichlet distribution over these fundamental patterns. In other words, we estimate how much of the protein would have to be in each pattern in order for the overall image to appear as it does. This task can be viewed as *unmixing* an image formed by mixing fundamental patterns.

We have described two approaches for estimating this: one in which we specify the fundamental patterns in advance and just try to estimate the fractions (referred to as supervised unmixing), and one in which we try to find the fundamental patterns as well as the fractions (referred to as unsupervised unmixing). Using a test set of images created by an automated high content imaging system, we have demonstrated that good estimates of the fractions can be obtained by both the supervised (23) and unsupervised (24) approaches.

Use of models for testing algorithms

A classic problem in testing algorithms for microscope images is that the correct results are frequently not known. A generative model for a desired pattern or structure can be combined with a model of image formation in a particular microscope to generate test images (“phantoms”) for which the correct results from image analysis are known. The process by which an image is formed in a microscope is quite well understood, so accurate models of point-spread functions and sampling noise can be constructed and applied to the idealized images generated by the methods described above. This approach has been applied previously for nuclei (15, 25); the paper by Svoboda et al (25) provides a particularly good image formation model.

The phantom approach can be extended to any combination of the tools in the CellOrganizer project to generate test images with known cell boundaries, object locations and/or subcellular patterns. The accuracy of algorithms can also be determined as a function of the parameters of the generative model, such as cell size or extent of nuclear elongation. Collections of already synthesized synthetic cell images can be found at http://CellOrganizer.org.

Conclusion

In this chapter I have described current approaches for building accurate models of cell organization directly from fluorescent microscope images. These models capture variation in cell organization at the level of the nucleus, cell membrane and individual organelles, and can capture how particular proteins are distributed among cellular components. They represent a significant advance over the use of words (such as GO terms) as the means by which results of experiments on subcellular localization and organization are captured and communicated. Nonetheless, the field is at the beginning, and it is hoped that many investigators will develop and make available tools that improve and extend the approaches described here. Examples of future work that can be anticipated include methods for merging images at different resolutions (especially light and electron microscope images) and methods for describing the interplay between localization and structure for proteins involved in creating subcellular structures.

Acknowledgments

I would like to express my thanks to Michael Boland, Meel Velliste, Ting Zhao, Tao Peng, Luis Coelho, Wei Wang and especially Gustavo Rohde for previous collaborative work described here and for many helpful discussions with them and with Jieyue Li, Taraz Buck, Ivan Cao-Berg, Baek Hwan Cho, Klaus Palme, Hagit Shatkay, Joel Stiles, Leslie Loew, Ion Moraru, Christoph Wülfing, Eric Xing, Gaudenz Danuser, Karl Rohr, and Ivo Sbalzarini. Much of the original work reviewed here was supported by NIH grants GM068845, GM075205, and GM090033, and by NSF grant EF-0331657. Part of the discussions and writing of this article were supported by a Research Award from the Alexander von Humboldt Foundation and an External Senior Research Fellowship from the Freiburg Institute of Advanced Studies.

References

1. Stiles JR, Bartol J, T M, Salpeter EE, Salpeter MM. Monte carlo simulation of neuro-transmitter release using MCell, a general simulator of cellular physiological processes. Proc Comput Neurosci. 1998:279–284.
2. Loew LM, Schaff JC. The Virtual Cell: a software environment for computational cell biology. Trends in Biotechnology. 2001; 19(10):401–406. [PubMed: 11587765]
3. Murphy RF. Communicating subcellular distributions. Cytometry Part A. 2010; 77(7):686–692.
4. Murphy RF. An active role for machine learning in drug development. Nature Chemical Biology. 2011; 7(6):327–330.
5. Zhao T, Murphy RF. Automated learning of generative models for subcellular location: building blocks for systems biology. Cytometry Part A. 2007; 71A(12):978–990.
6. Rohde GK, Ribeiro AJ, Dahl KN, Murphy RF. Deformation-based nuclear morphometry: capturing nuclear shape variation in HeLa cells. Cytometry Part A. 2008; 73A(4):341–350.
7. Rohde GK, Wang W, Peng T, Murphy RF. Deformation-based nonlinear dimension reduction: Applications to nuclear morphometry. Proc 2008 Intl Symp Biomed Imaging. 2008:500–503.
8. Peng T, Wang W, Rohde GK, Murphy RF. Instance-based generative biological shape modeling. Proc 2009 Intl Symp Biomed Imaging. 2009:690–693.
9. Shariff A, Murphy RF, Rohde GK. A generative model of microtubule distributions, and indirect estimation of its parameters from fluorescence microscopy images. Cytometry Part A. 2010; 77A(5):457–466.
10. Shariff A, Murphy RF, Rohde GK. Automated Estimation of Microtubule Model Parameters from 3-D Live Cell Microscopy Images. Proc IEEE Int Symp Biomed Imaging. 2011; 2011:1330–1333. [PubMed: 21804927]
11. Peng T, Murphy RF. Image-derived, Three-dimensional Generative Models of Cellular Organization. Cytometry Part A. 2011; 79A in press.
12. Ashburner M, et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. (Translated from eng). Nat Genet. 2000; 25(1):25–29. (in eng). [PubMed: 10802651]
13. Blum H. Biological shape and visual science. I. (Translated from eng). J Theor Biol. 1973; 38(2): 205–287. (in eng). [PubMed: 4689997]
14. Beg MF, Miller MI, Trouve A, Younes L. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. Int J Comput Vis. 2005; 61:139–157.
15. Yang S, et al. Nonrigid registration of 3-d multichannel microscopy images of cell nuclei. (Translated from eng). IEEE Trans Image Process. 2008; 17(4):493–499. (in eng). [PubMed: 18390358]
16. Helmuth JA, Burckhardt CJ, Greber UF, Sbalzarini IF. Shape reconstruction of subcellular structures from live cell fluorescence microscopy images. (Translated from eng). J Struct Biol. 2009; 167(1):1–10. (in eng). [PubMed: 19358891]

17. Zhao T, Velliste M, Boland MV, Murphy RF. Object type recognition for automated analysis of protein subcellular location. IEEE Trans. on Image Processing. 2005; 14(9):1351–1359.
18. Sprague BL, et al. Mechanisms of microtubule-based kinetochore positioning in the yeast metaphase spindle. (Translated from eng). Biophys J. 2003; 84(6):3529–3546. (in eng). [PubMed: 12770865]
19. Chen X, Velliste M, Murphy RF. Automated Interpretation of Subcellular Patterns in Fluorescence Microscope Images for Location Proteomics. Cytometry Part A. 2006; 69A:631–640.
20. Conrad C, Gerlich DW. Automated microscopy for high-content RNAi screening. (Translated from eng). The Journal of cell biology. 2010; 188(4):453–461. (in eng). [PubMed: 20176920]
21. Shariff A, Kangas J, Coelho LP, Quinn S, Murphy RF. Automated image analysis for high-content screening and analysis. (Translated from eng). Journal of biomolecular screening : the official journal of the Society for Biomolecular Screening. 2010; 15(7):726–734. (in eng).
22. Bauer DC, et al. Sorting the nuclear proteome. (Translated from eng). Bioinformatics. 2011; 27(13):i7–i14. (in eng). [PubMed: 21685104]
23. Peng T, et al. Determining the distribution of probes between different subcellular locations through automated unmixing of subcellular patterns. (Translated from eng). Proc Natl Acad Sci U S A. 2010; 107(7):2944–2949. (in eng). [PubMed: 20133616]
24. Coelho LP, Peng T, Murphy RF. Quantifying the distribution of probes between subcellular locations using unsupervised pattern unmixing. Bioinformatics. 2010; 26 in press.
25. Svoboda D, Kozubek M, Stejskal S. Generation of digital phantoms of cell nuclei and simulation of image formation in 3D image cytometry. (Translated from eng). Cytometry Part A. 2009; 75A(6):494–509. (in eng).

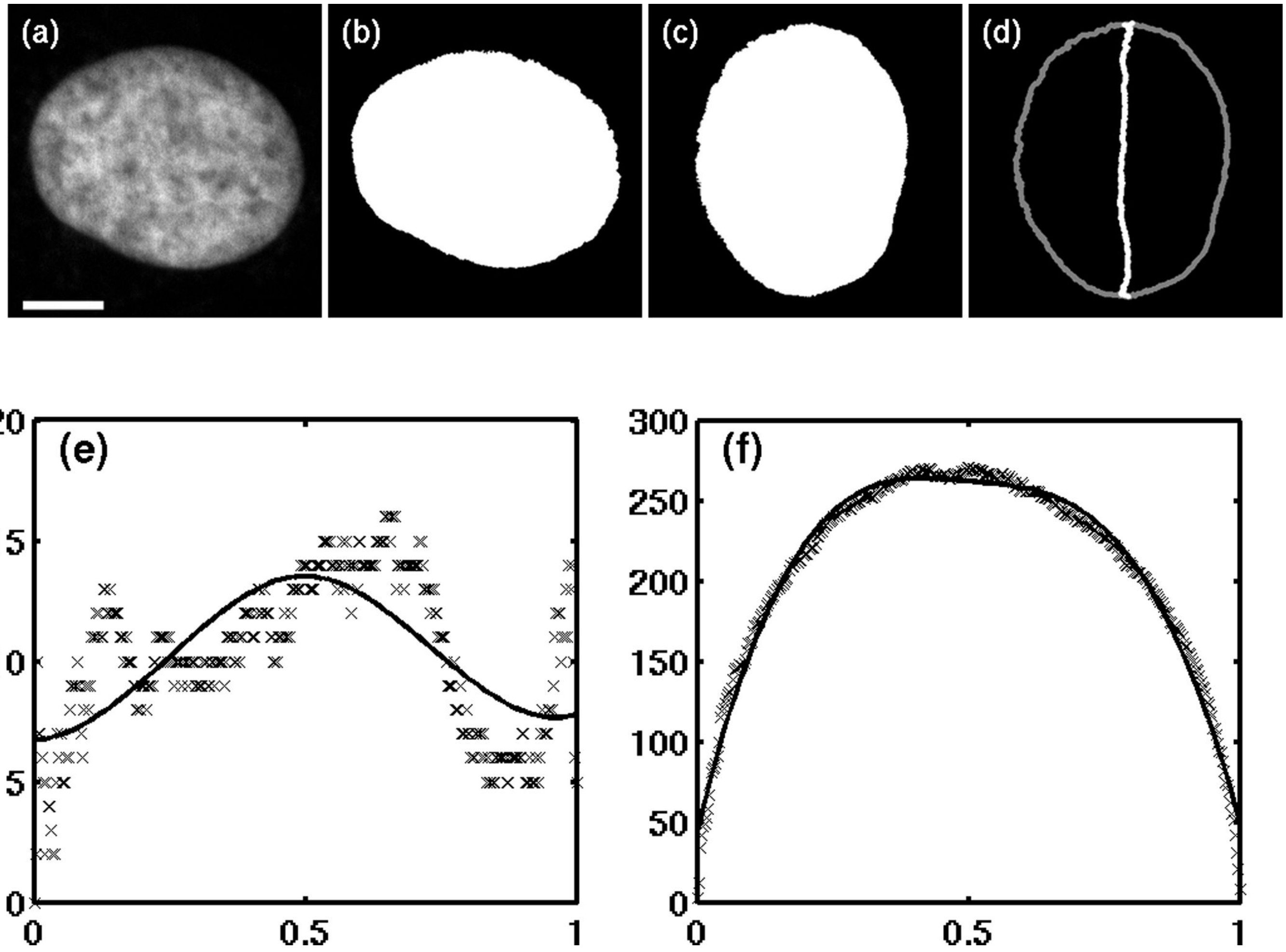

Figure 1.

Illustration of a medial axis method for modeling a 2D nuclear shape instance. The original nuclear image (a) was binarized (b) and rotated so that its major axis is vertical (c). The position of the curve that divides the shape in half horizontally at each vertical position is then found (d). The horizontal positions of the medial axis as a function of the fractional vertical distance are shown by the symbols in (e), along with a B-spline fit (solid curve). The width as a function of fractional distance is shown by the symbols in (f), along with the corresponding fit (solid curve). Scale bar, 5 um. From Zhao & Murphy (5).

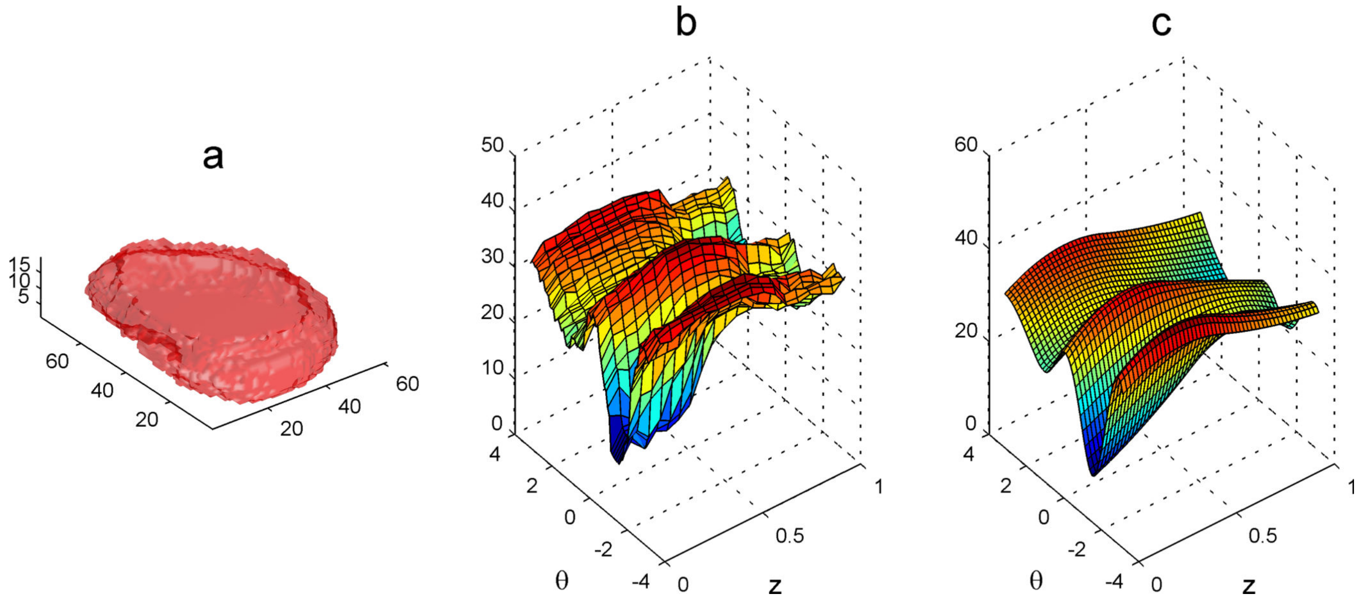

Figure 2.

Illustration of cylindrical spline surface method for modeling a 3D nuclear shape instance. (a) Surface plot of a 3D HeLa cell nucleus. (b) Unfolded surface of the nuclear shape in a cylindrical coordinate system. The surface plot shows the radius *r* as a function of azimuth u and height *z*. (c) B-spline surface fitted to the unfolded nuclear surface. From Peng & Murphy (11).

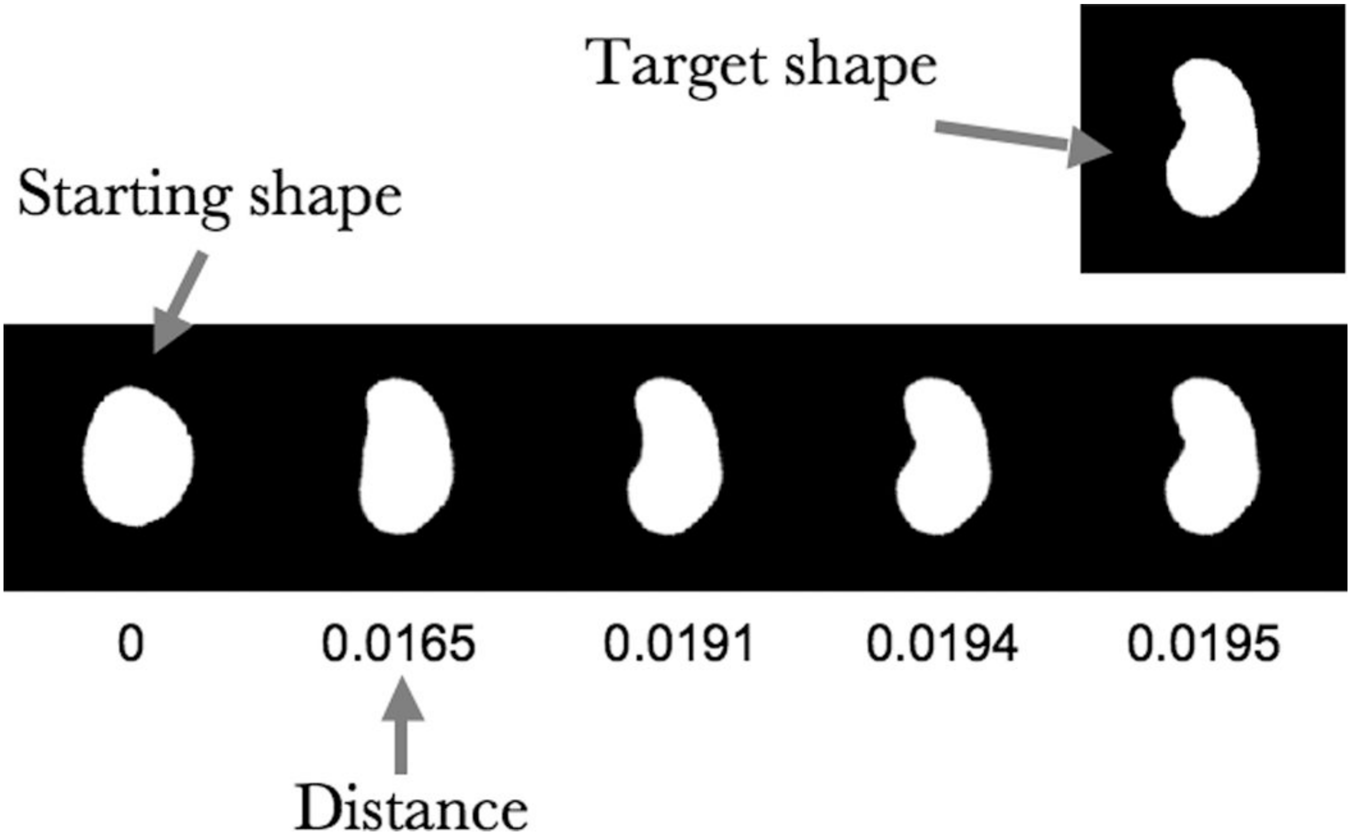

Figure 3.

Determining the distance between two shapes using large deformation metric mapping. The goal is to measure the distance betwee the starting shape and the target shape. This is done by gradually deforming the starting shape to become more similar to the target shape while recording how much perturbation is necessary at each step. From Rohde et al (6).

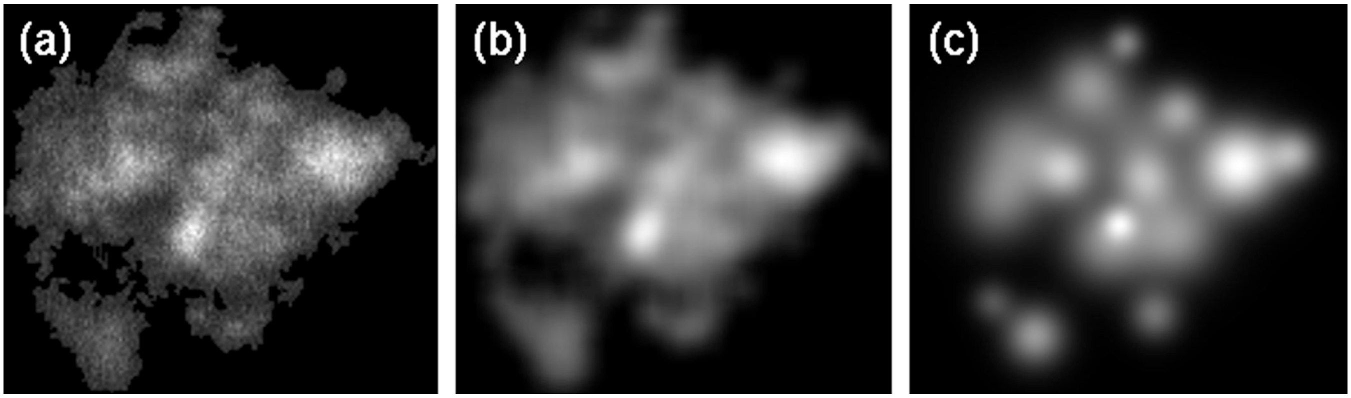

Figure 4.

Illustration of fitting objects using a 2D Gaussian mixture model. A region of a cell containing a single composite object (found by thresholding and connecting above threshold pixels) (a) is smoothed by a Gaussian lowpass filter (b) to faciliate detection of local maxima (“peaks”) in the composite object. Fitting using a spherical covariance matrix (c) yields the estimated positions and sizes of the Gaussian objects assumed to have given rise to the original image. A similar approach is used for 3D images. After Zhao & Murphy (5).

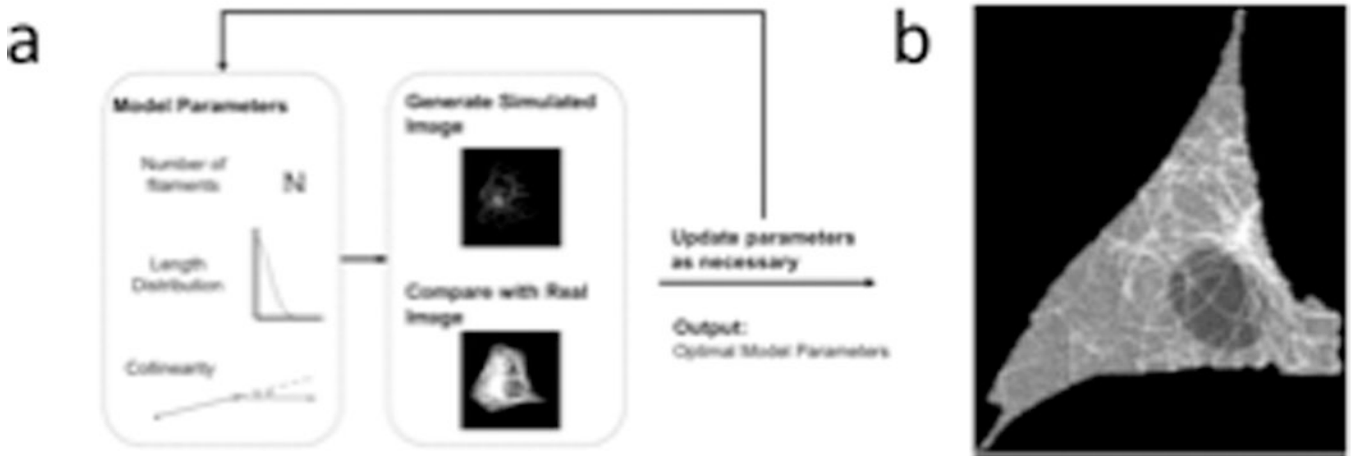

Figure 5.

(a) Overview of inverse modeling approach for estimating parameters of the microtubule generative model. From Sharif et al (9). (b) Example of 2D slice from 3D synthetic image generated by tubulin model.

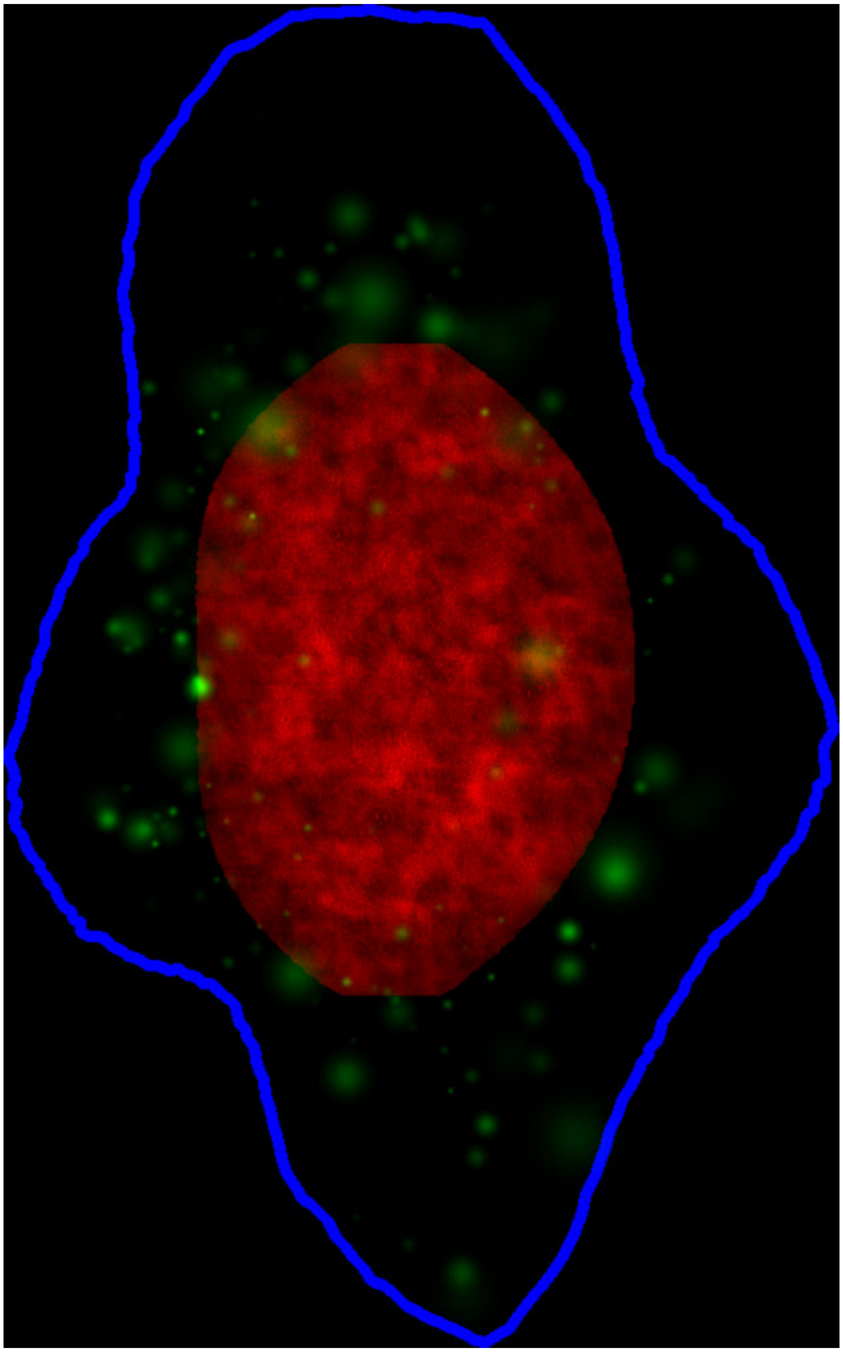

Figure 6.

Example synthetic image generated by a 2D model learned from images of the lysosomal protein LAMP2. The DNA distribution is shown in red, the cell outline in blue, and LAMP2-containing objects in green. From http://murphylab.web.cmu.edu/data/2007_Cytometry_GenModel.html.

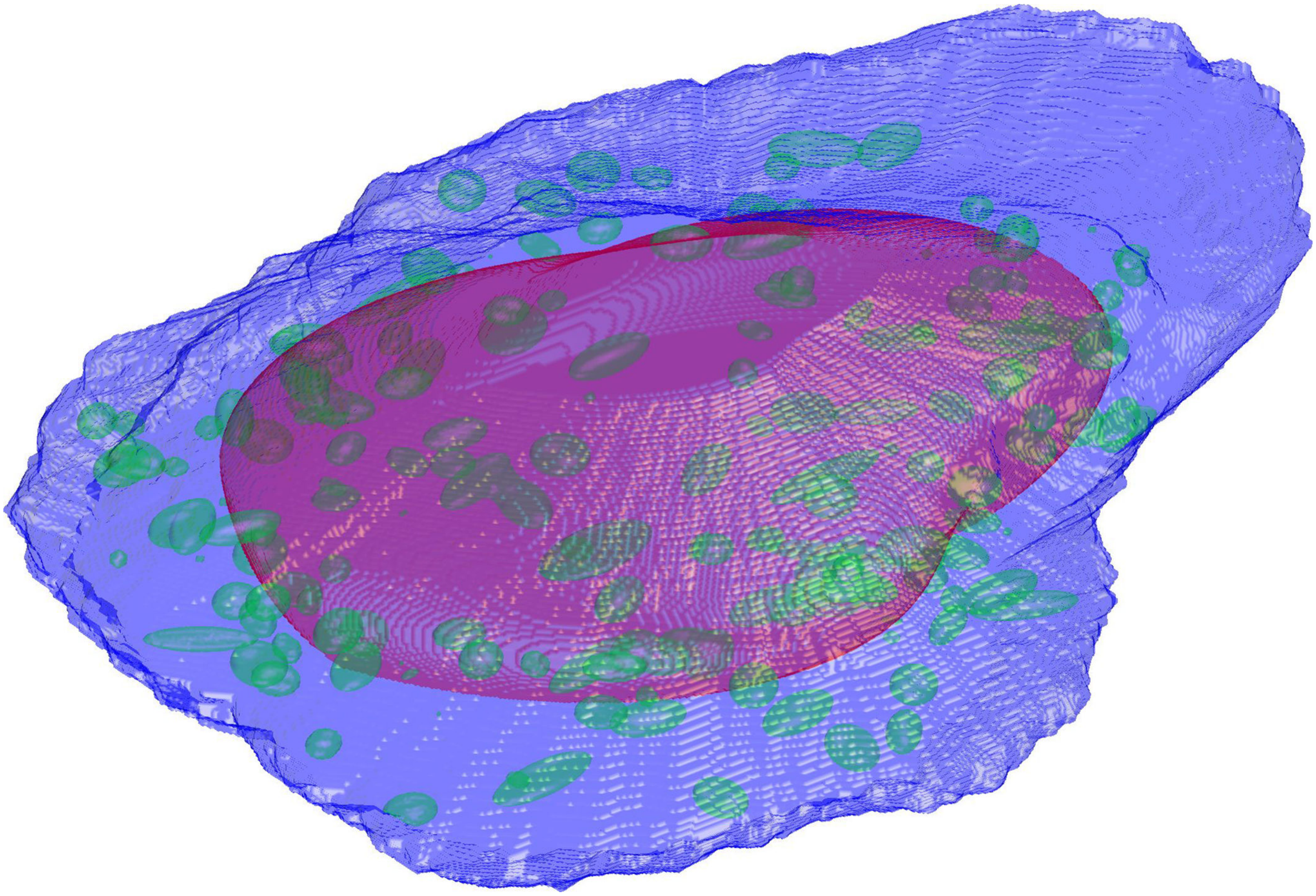

Figure 7.

Example synthetic image generated by a 3D model learned from images of the lysosomal protein LAMP2. The nuclear surface is shown in red, the cell surface in blue, and LAMP2-containing objects in green.