Motion Tracking and Intensity Surface Recovery in Microscopic Nuclear Images

Lopamudra Mukherjee[1], Mingen Lin[1], Jinhui Xu[1], and Ronald Berezney[2]

[1] Department of Computer Science, State University of New York at Buffalo, USA
{lm37, mlin6, jinhui}@cse.buffalo.edu
[2] Department of Biological Sciences, State University of New York at Buffalo, USA
berezney@buffalo.edu

Abstract. Current techniques for microscopic imaging do not provide necessary spatial and temporal resolutions for real time visualization of the nucleus. Images can only be acquired in time lapse mode, leading to significant loss of information between image frames. Such data, if available, can be extremely helpful in the study of nuclear organization and function. In this paper, we present a gamut of geometric-technique-based approaches for solving the problem. Our techniques, working together, can effectively recover complicated motion and deformation as well as the change of intensity surfaces from pairs of images in a microscopic image sequence, and has low time complexity, particularly desirable by many biological applications where large amount of DNA need to be processed. These techniques are also readily applicable to other types of images for reconstructing motion and intensity surfaces of deformable objects.

1 Introduction

A clear understanding of the mammalian cell nucleus and its exquisite architecture which governs gene expression and regulation, remains the final frontier in modern nuclear biology. To this end, recent developments in microscopy, imaging and labelling have led to a significantly improved view of the cell nucleus. This technology has enabled the observation of DNA replication and transcription sites and their organization into higher order units or "zones". Researchers believe that this dynamic interplay and "re-zoning" may form the structural basis for elaborate global coordination of replicational and transcriptional programs within the mammalian nucleus[1]. However, despite significant progress in microscopic imaging technologies, the task of analyzing the nuclear images to derive semantic (and biologically relevant) information is rather difficult. One key impediment is the presence of enormous amount of chromatin in the nucleus and their complicated movement (non-uniform translation, rotation and deformation). This makes it extremely difficult for biologists to do real time visualization of such nuclear activities. This problem is further compounded by the limited spatial and temporal resolution of current microscopic techniques which introduce under-sampling or "gaps" between acquired images. Biological analysis of such data is generally the result of long sessions involving manual calculations and hypothesis (based on experience), making it human error prone

J. Duncan and G. Gerig (Eds.): MICCAI 2005, LNCS 3749, pp. 876–884, 2005.

and very tedious. These problems motivate the design of a fully-automated set of techniques to provide biologists real time visualization of nuclear activities within the scope of existing imaging modalities.

The problem of recovering motion and deformation of moving objects, given only a few snapshots of the objects during their motion, has been a challenging problem in various fields such as computer vision and pattern recognition, and robotics[2, 3]. Most of these approaches provide interesting insights into possible interpolation strategies and work well for the specific objects, images or assumptions they were designed for. Unfortunately, these techniques do not extend well to microscopic images for a number of reasons. They were primarily designed for large sized objects where sufficient information about the mechanics and physics behind the motion is available or can be assumed. For genomic structures, however, where no such data is available, it becomes necessary to exploit the limited information hidden in the images in an effort to recreate an accurate motion of the objects under consideration.

In this paper, we present a suite of geometric techniques that address the problem of "gaps" between microscopic images. We suggest techniques to determine the shape and motion of genomic structures for any number of intermediate (or unavailable) time points in a 2D image sequence. We further extend our idea by not only regenerating the contours but also the continuous changes of intensity surfaces from one image to another. This allows us to generate as many "missing" in-between image frames as desired given as few as two time separated image frames. When a number of nuclear images acquired in time-lapse mode are available, our techniques allow an online recreation of a "movie" sequence, enabling a real-time visualization of nuclear processes with negligible or no manual interaction. These provide a far less tedious biological interpretation of the imaging data.

2 Method

In this section, we overview our method for motion tracking and intensity surface recovery of a set of objects with prior knowledge of only their initial and final positions, shapes and intensities. We illustrate out results on microscopic images of the cell nucleus.

2.1 Simplification and Segmentation

We begin with a sequence of 2D microscopic images of the same set of genomic structures (representing chromatin domains or chromatin foci) taken at different time points. One major problem with microscopic images is that they are blurred in portions by clouds of quasi-foci (formed by overlapping of neighboring foci) which complicates the process of segmentation and intensity recovery . Therefore our first goal is to simplify the images to make segmentation easier. In a recent work [4], we reported our observations of an interesting phenomena that showed that image profiles of chromatin foci, even in the presence of noise,

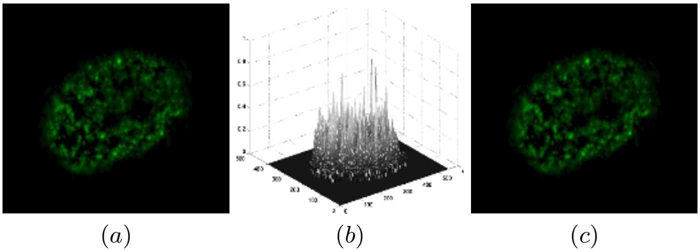


Fig. 1. (a) Original cell nuclear image. (b) The corresponding intensity profile. (c) Approximated cell nuclear image.

resembled certain exponential distributions. We extend that idea to make the task of segmentation easier by preceding it with a simplification process.

The main idea of the simplification is to determine a set of *approximation functions* for approximating the intensity surface of the image within a certain error tolerance ϵ of its original intensity. More specifically, we solve the following problem: Given a set $P = \{p_1, p_2, \cdots, p_n\}$ of pixels with each p_i associated with an intensity value $I_i > 0$, find a small set $F = \{f_1(\cdot), f_2(\cdot), \cdots, f_k(\cdot)\}$ of functions with each $f_i(\cdot)$ defined by a generalized normal distribution surface such that the intensity I_i of each pixel p_i is closely approximated by the maximum value of the k functions at p_i, i.e., $|I_i - \max_{j=1}^{k} f_j(p_i)| \leq \epsilon$ (or $|I_i - \max_{j=1}^{k} f_j(p_i)| \leq \epsilon I_i$) for some small constant $\epsilon > 0$. Each of these functions $f_i(\cdot)$ follows a generalized 2-D normal distribution $f_i(X) = \sigma e^{-(X-\mu)^T Q(X-\mu)}$, where Q is a positive definite matrix (i.e., the intersection of $f_i(\cdot)$ with a plane $z = c$ is an arbitrarily oriented ellipse). The details of the algorithm are omitted due to lack of space. Experimental results have shown that the approximated image generated not only closely resembles the original image (see Fig 1) but makes segmentation and further processing much more tractable since it is just represented by a set of mathematical functions.

The approximated image, free of diffused signals, is then segmented using relative thresholds (w.r.t. the peak value of the normal distribution functions) to shrink the domain of each approximation function and then a merge segmentation method [5] is used to generate the boundary of each foci. This approach shows significantly better results over commonly used segmentation techniques (such as threshold based, edge-detection based, watershed, etc.), especially for images which contain many small-sized objects or whose intensity varies significantly over the scope of the images. However, even after the images are approximated and segmented, the contours may still contain many undesired furrows (undulations or small recesses) caused by other types of unavoidable noise on the periphery. These furrows not only hide the actual boundary of the genomic structures but also increase the processing time dramatically. Taking this into account, we apply our contour simplification algorithm, proposed in [6] to simplify and smooth the boundary within a predecided error tolerance. This process yields the set of contours of both the initial and final images modeled as polygons.

2.2 Motion Tracking

Once the genomic structures have been represented as polygonal contours, the next step is to determine the correspondence between the initial and final polygonal sets. Previous methods to solve this problem have relied on manual estimation or heuristic techniques based on additional feature information [7]. Such approaches do not work directly for our problem, as we need to globally optimize the results for large sets of polygons, as well as consider the shape of individual objects while determining the correspondences. To overcome this difficulty, we propose a technique which takes both the shapes and relative positions of polygons into account and is also tailor-made for the problem at hand.

In determining correspondences between the two polygonal sets, our technique first tries to find a global transformation which moves every polygon in the initial set, P, within a small neighborhood of some polygon in the final set, Q. This step is useful because once such a transformation is determined, local neighborhoods can be examined for proximity. For example, if a transformed polygon p_i of P, overlaps with q_j of Q (or the neighborhood of q_j), it is an indication that q_j may be a possible correspondence for p_i. The notion of neighborhood can be thought of as a square of radius R (or any bounded region) centered roughly in the middle of each $q_j \in Q$. Clearly, when the neighborhood R is large, a global transformation satisfying the criteria above can be easily found, hence a large R is always feasible. To obtain better accuracy, we employ a binary search procedure on R, to determine the smallest *feasible* R, R_{min}.

The above procedure yields a reasonably small value of R_{min}. It also gives for each $p_i \in P$ (or $q_i \in Q$) , one or more probable candidates for correspondence in Q (or P) based on R_{min} and the overlap criteria. We follow up this procedure with another refinement process by modelling the problem as the well-known Maximum Bipartite Graph Matching problem. In our case, the vertex sets are corresponding polygons of P and Q and the edge weights are the distances between individual polygons in these two subsets.

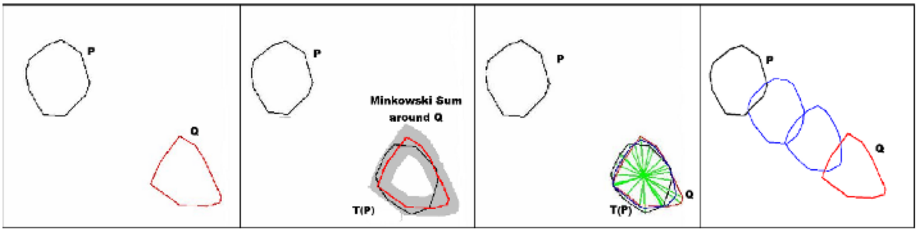
Correpondences established using the above techniques are generally accurate and comparable with the results obtained from other well-established techinques like Nearest Neighbour. Correspondences obtained are mostly one-to one, but in some cases, they may be one-to-many or many-to-one. We attribute this to the chromatin domains disappearing or one domain splitting into a few smaller chromatin domains. These exceptional cases can be reduced to a general case of a one-to-one correspondence of polygons by employing additional routines such as Voronoi diagrams.

Once the global transformations have been determined, we have essentially taken care of capturing the movement of the cell nucleus as a whole. However, each individual chromatin foci have their independent position(rigid) and shape(non-rigid) change which our technique still needs to address.

To account for the rigid transformations, we first scale the two polygons so that they have comparable sizes. We then use a curve matching algorithm to compute the best translation and rotation so that after these transformations, the contours of the two corresponding chromatin domains have the least difference in shape, measured by the Hausdorff distance metric[8]. Given two polygons P and Q, we find the Minkowski sum (union) of a square of size r with Q, which denotes an inner and outer boundary around Q enclosing a region S. The image $T(P)$ of P, under an appropriate transformation T, should always lie in S. In essence, we set a limit on the value of Hausdorff distance, denoted by r and then determine a transformation T such that the Hausdorff distance between $T(P)$ and Q falls within that limit. After rotating Q by an appropriate angle, finding the adequate translation is realized by determining for each edge e of P a set of translations, under which e will always lie within S and then finding the intersection of all such sets in translation space. When no such transformation exists i.e., if there is no intersection, we modify the initial value of r and repeat the above procedure till a transformation is obtained for the minimum possible value of r .

Having found the best match under rigid body transformations, we then proceed to determine the non-rigid shape changes by a routine called shape-deformation. Shape-deformation procedure first computes one or a few central points in the common intersection of Q and $T(P)$ so that all points in $Q \cup T(P)$ can be guarded (i.e., visible to) by at least one of the central points(such points always exits, otherwise we can transform one of the polygons to overlap the other, obtaining a match with a smaller Hausdorff distance, which is not possible). For each central point o, we draw line segments from o to all visible vertices and intersections of the polygons. Any pair of such line segments, say the i^{th} pair, encloses a portion of the boundary, P_i and $T(Q)_i$, of the polygons Q and $T(P)$, respectively. The technique then proceeds to determine the image of $T(P)_i$ under the transformation $S_t(T(P)_i, s, \theta)$, where t is an intermediate time point between the initial and final time point and s and θ are the scaling factor and angle of rotation for the local scaling and rotation which resulted in $T(P)_i$ in being transformed to Q_i. Ultimately, all such local transformations $S_t(T(P)_i)$ are combined to yield the deformed shape of $T(P)$ at time point t.

Combining the rigid body transformations with the shape deformation, our approach obtains a continuous motion and deformation of the whole set of polygons (or contours) in two consecutive images (see Figure 2).



(a) (b) (c) (d)

Fig. 2. (a) Initial and Final positions of a polygon in case of one-to-one correspondence (b) Rigid Transfromations (c) Shape deformation around a central point (d) Shapes and positions of the deforming object at a sequence of intermediate time points

2.3 Intensity Surface Recovery

Once the contours for all desired intermediate images have been generated, we need to recover the intensity surfaces corresponding for each pair of chromatin domains in P and Q. Recall that the intensities of these chromatin domains can be approximated by a set of normal distribution functions. (see Section 2.1). Our objective is to reconstruct for each intermediate contour, the set of normal distribution functions that should represent its intensity surface. In doing so, we derive inspiration from the fact that most 2D functions retain their spatial properties under rigid transformations like translation, rotation or scaling. Also, the shape deformation routine for a particular region can be reduced to a set of unique rigid transformations localized to smaller sub-regions, this holds true for the mathematical functions defining every sub-region. The context of this statement will become clear when we describe the algorithm.

Before discussing the details of the algorithm, we will formally define the notations. Let F_P and F_Q be the set of normal distribution functions for a corresponding pair of chromatin domain P and Q respectively. Consider $T_{P->Q}$ to be the transformation which when applied to P, yields Q. This transformation is symmetrical i.e., $T_{P->Q}^{-1}$ when applied to Q will give P. By the same token, the transformations for generating the kth intermediate image P_k from P can be represented as $T_{k_{P->Q}}$.

Our algorithm works on the principle that the domains of normal distributions are equally affected by transformations such as rotation, translation, scaling and shape-deformations, as are the contours containing them. Hence we use the transformations generated previously, to obtain the domains of normal distributions at the intermediate time points. Also, their intensities are likely to be influenced by the intensities of both the initial and final images. Therefore, we generate the intermediate domains of the normal distributions from either side.

The distribution functions of F_P (and F_Q) are first projected on to the xy plane to yield a set of regions, represented as D_P (and D_Q), as shown in Fig. 3. Though this procedure is “lossy”, the outer periphery of the union of the members of D_P and D_Q makes up P and Q. $T_{k_{P->Q}}$ is then applied to each member of D_P which results in a set of 2D domains. Similarly, $T_{k_{Q->P}}$ is applied to D_Q to obtain another set of domains. These two set of domains can be thought

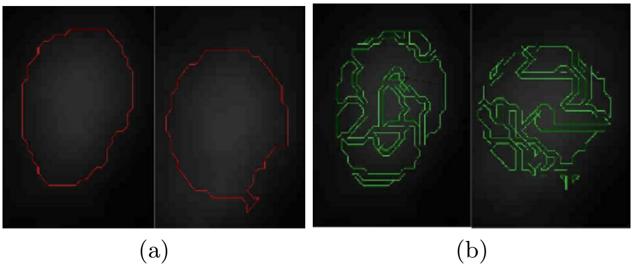


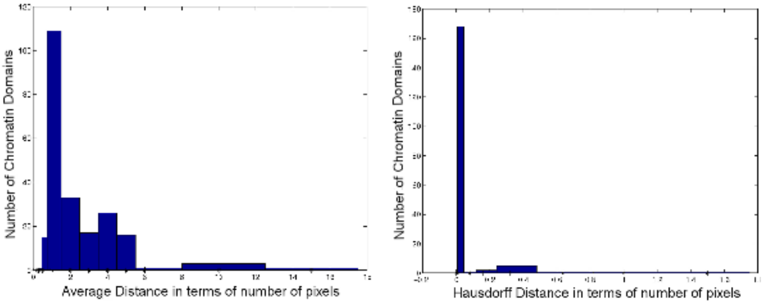
Fig. 3. (a) Two corresponding contours (b) Their normal distribution domains projected on a x-y plane(z axis represent the intensity values)

of as “clones” of D_P and D_Q under transformations in the forward and reverse directions and cover the same region bounded by the contour of $T_{k_{P->Q}}(P)$. At this stage, for each pixel inside the contour $T_{k_{P->Q}}(P)$, we run a simple check for membership in both $T_{k_{P->Q}}(D_P)$ and $T_{k_{Q->P}}(D_Q)$. Note, that owing to the construction, a pixel has to belong to at least one domain in both of these domain sets. Once the membership is established, we refer to their corresponding “clones” in D_P and D_Q and the pixel intensity is then expressed as an affine combination of the intensity values originally associated with the untransformed clones i.e. for a pixel p_{k_l} ,if $I_{D(P)_i}$ and $I_{D(Q)_j}$ be the intensity values associated with the clones $D(P)_i$ and $D(Q)_j$ of its parent domain, then $I_{p_{k_l}} = \alpha * I_{D(P)_i} + (1 - \alpha) * I_{D(Q)_j}$ where α is the ratio of time intervals from the initial to the intermediate time point and the intermediate to the final time point.

Preliminary results indicate that intensity surfaces can be efficiently recovered using this technique and matches the actual intensity surface very well.

3 Results

We implemented our algorithms in C++ using CGAL-2.4 and LEDA Version 4.4 on a 1.6 GHz machine running RedHat Linux Enterprise Version 2.0. We tested our technique using images of chromatin domains taken at an interval of 3 secs. Analysis of the results reveal that for 90% of the chromatin domains, the average movement is in the range of 2-4 pixels(Fig 4a) and deformation of the domains measured by the Hausdorff distance is very small(in the range of .01 pixel unit) for about 98% of the chromatin domains (Fig 4b). These results obtained are consistent with results of independent experiments performed on same types of images and hence pass a quantitative evaluation of the correctness of our technique. Here we can show only small portion of the results due to lack of space. Figures 5 (a) and (g) show the segmented images of chromatin domains in the nucleus of a living cell at time 0 sec and time 3 sec, which are the initial and final time points respectively. The images generated at intermediate



(a) (b)

Fig. 4. Histograms showing (a) Average movement and (b) Hausdorff Distance calculated, vs number of Chromatin Domains for images at an interval of 3 secs

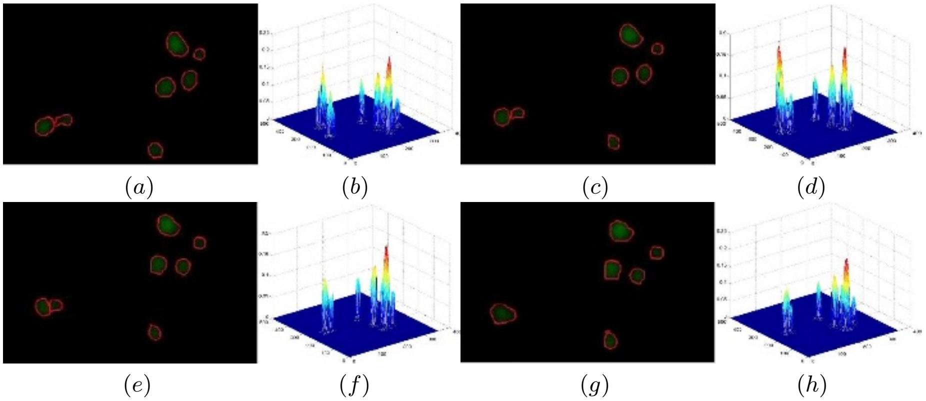


Fig. 5. (a-b) Segmented image at 0 sec with intensity plot (c-d) Image generated at 1 sec with intensity plot (e-f) Image generated at 2 sec with intensity plot (g-h) Segmented image at 3 sec with intensity plot

time points 1 sec and 2 sec are shown in Figures 5(c) and (e). The algorithm can be used to generate continuous motion by generating as many intermediate images as desired. This serves the objective of providing biologists with movies depicting the processes involving genomic structures taking place inside the cell nucleus in an online real time fashion.

4 Discussion

We have presented a novel geometric-technique-based approach for recovering continuous motion, deformation and intensity surface for genomic structures in microscopic nuclear images of living cells. A key feature of our approach is to use normal distribution functions to capture the main structure of each nuclear image, to segment genomic structures from diffused signals, and to recover the intensity surfaces of the segmented genomic structures. Our technique considers both rigid and non-rigid motion and does not require detailed knowledge about the physics and mechanics of the chromatin domains considered. Also, our techniques have low time complexity which is particularly desirable for biological images involving a large amount of data.

References

1. R. Berezney, X.W.: The new paradigm: integrating genomic function and the nuclear architecture. In: J. Cell Biochem Suppl. Volume 31. (1998) 238–342
2. D. Metaxas, D.T.: Shape and nonrigid motion estimation through physics-based synthesis. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. (1993) 580–591

3. L. Zhou, C.K.: Hierarchial structure and nonrigid motion recovery from 2d monocular views. In: CVPR' 00. Volume 2. (2000) 752–759
4. Mingen Lin, Jinhui Xu, S.B.R.B.: Surface approximation based image simplification and applications. In: IEEE Systems, Man and Cybernetics. (2004)
5. Sonka, M., Hlavac, V., Boyle, R.: Image Processing, Analysis and Machine Vision. Brooks/Cole, (Pacific Grove, CA, USA)
6. Lopamudra Mukherjee, Jinhui Xu, A.P.R.B.: Efficient algorithms for motion and deformation recovery with biological applications. In: IEEE Systems, Man and Cybernetics. (2004)
7. Longin Jan Latecki, R.L.: Shape similarity measure based on correspondence of visual parts. IEEE transactions on Pattern Analysis and MAchine Intelligence **22** (2000) 1185–1190
8. H. Alt, B.B., Blomer., J.: Approximate matching of polygonal shapes. In: Ann. Math. Artif. Intell. Volume 13. (1995) 251–266