Automatic Tracking of Individual Fluorescence Particles: Application to the Study of Chromosome Dynamics

Daniel Sage, Frank R. Neumann, Florence Hediger,
Susan M. Gasser and Michael Unser, *Fellow*, *IEEE*

Abstract

We present a new, robust computational procedure for tracking fluorescent markers in time-lapse microscopy. The algorithm is optimized for finding the time-trajectory of single particles in very noisy dynamic (2D or 3D) image sequences. It proceeds in three steps. First, the images are aligned to compensate for the movement of the biological structure under investigation. Second, the particle's signature is enhanced by applying a Mexican hat filter, which we show to be the optimal detector of a Gaussian-like spot in $1/\omega^2$ noise. Finally, the optimal trajectory of the particle is extracted by applying a dynamic programming optimization procedure. We have used this software, which is implemented as a Java plug-in for the public-domain ImageJ software, to track the movement of chromosomal loci within nuclei of budding yeast cells. Besides reducing trajectory analysis time by several 100-fold, we achieve high reproducibility and accuracy of tracking. The application of the method to yeast chromatin dynamics reveals different classes of constraints on mobility of telomeres, reflecting differences in nuclear envelope association. The generic nature of the software allows application to a variety of similar biological imaging tasks that require the extraction and quantitation of a moving particle's trajectory.

Index Terms

Image sequence analysis, particle tracking, dynamic programming, fluorescence microscopy, living cells, chromosome

D. Sage and M. Unser are with the Ecole Polytechnique Fédérale de Lausanne (EPFL), Biomedical Imaging Group, CH-1015 Lausanne, Switzerland.
e-mail: daniel.sage@epfl.ch, michael.unser@epfl.ch

F. Neumann, F. Hediger and S. Gasser are with the Department of Molecular Biology at the University of Geneva, CH-1211 Geneva, Switzerland.
e-mail: frank.neumann@molbio.unige.ch. forence.hediger@molbio.unige.ch, susan.gasser@fmi.ch

The research in the Gasser laboratory is supported by the Swiss National Science Foundation, the NCCR "Frontiers in Genetics", and the University of Geneva.

I. INTRODUCTION

During the past decade, two important technological innovations have helped reshape molecular and cell biological research. One was the development of fluorescent proteins, which allow researchers to selectively label single proteins or DNA loci in vivo [1]. The second is high-resolution fluorescence imaging that was made possible by a new generation of brightfield and confocal microscopes, sensitive CCD cameras and deconvolution algorithms [2]. Thanks to these new tools, biologists are able to observe gene expression and to study molecular dynamics within the living cell at sub-micron resolutions [3]. Static images can be acquired in two (XY) or three (XYZ) dimensions to localize the labeled structures of interest in a living specimen. Dynamic sequences (time-lapse series) can also be used to study the dynamic behaviour of labeled molecules within a living cell. While these methods offer an enormous potential for increasing our understanding of biological events, they also constitute a challenge for quantitative analysis, which requires efficient techniques to evaluate this unprecedented flow of data. Currently, a majority of data analysis and feature extraction is done manually, which is both time-consuming and susceptible to personal bias. Some commercial image analysis tools are available, but their capabilities for automatic feature extraction are limited. The analysis is complicated by the fact that the data are typically very noisy due to the weakness of the fluorescence signal and the need to work at the limit of resolution for light microscopy.

In this work, we address the problem of extracting and quantifying trajectories of moving particles. It is a generic data analysis problem in biophysics. The most common approach involves decomposing the problem into two steps. First, the particles are detected independently in each frame—this constitutes the segmentation phase. Second, the individual trajectories are extracted by linking the detected particles in consecutives images based on the determination of the best match in feature space, a process that we refer to as “frame-to-frame tracking”. A recent study [4] compares the performance of such algorithms. Much work has been devoted to improving frame-to-frame tracking of multiple particles in 3D sequences [5, 6], dealing with problems such as occlusions, split or merge of particles, achieving real-time performance [7], or tracking out-of-focus particles [8-10] using the PSF (Point Spread Function) of the microscope to obtain super-resolution. This frame-to-frame tracking framework is the approach taken by most commercial software packages, such as Volocity 2 from Improvision [11] or Metamorph from Universal Imaging Corporation [12]. The major weakness of the frame-to-frame approach is that it is sensitive to false detections due the ill-posed segmentation problem, the restricted vision of frame-to-frame tracking, and the difficulty of resolving partial occlusions. Because tracking is often done off-line in molecular biology, an algorithm can in principle take into account the whole data set information and not only the recent past and the present as in a real-time tracking system.

In this paper, we explore a robust alternative strategy to particle tracking, in which the detection and tracking are performed simultaneously. Specifically, we show that this problem can be formulated as a global optimization process which we solve using the dynamic programming (DP) method. This method is inspired by an approach developed for the extraction of contours in very noisy ultrasound images [13]. The advantage of such an approach is that it can extract a particle's trajectory reliably in a XYZ space, even when the data is extremely noisy, by taking into account all the frames of the sequence (past+future). Its only limitation is that it can track one particle at a time. It is nonetheless adequate for single molecule imaging, or for multiple particles whose trajectories are sufficiently separated in space. The described software speeds up the analysis process considerably, allowing a reproducible exploitation of the available image data.

The work is motivated by an interesting application in molecular biology, which is the study of the dynamics of chromosomal loci within the eucaryotic nucleus [14]. It is now proven that chromosomes are not positioned randomly within the nucleus, but that they rather occupy specific territories [15]. Nonetheless, specific sites within chromatin are highly dynamic. The combination of defined position and constrained motion may facilitate DNA transactions such as transcription, replication and repair, and defining how DNA and proteins move *in vivo* is crucial for understanding these complex biological processes. To this end, one can label specific chromosomal loci (such as the telomeres at the extremity of the chromosome) by integrating arrays of high affinity protein binding sites and expressing the binding protein fused to GFP (Green fluorescence protein) and visualize them using live, dynamic microscopy. Having a precise trajectory of the DNA locus is important for making statistical analyses, model fitting, and for comparing a array of test conditions [15]. A manual analysis of these image sequences requires that, for each frame, the biologist scores manually the coordinates of the center of the nucleus and the coordinates of the center of the GFP-tagged DNA spot by clicking on the corresponding pixel. These coordinates are reported in a spreadsheet for subsequent statistical analysis. It entails the entry of at least 600 mouse clicks per movie, which is quite tedious and susceptible to operator-related differences. To speed up this process, increase the reproducibility and to better exploit the image data available, we therefore took up the challenge of developing a reliable computational solution for extracting the time-trajectory of these fluorescent markers automatically.

The paper is divided into three parts. In Section 2, we present our computational solution to the generic problem of tracking a moving particle in very noisy image. In Section 3, we evaluate our algorithm and report some validation results. Finally, in Section 4, we consider the application to the problem of telomere localization and present some real experimental results.

II. Automated Tracking of a Single Particle

Our goal is to obtain a complete, reliable description of the time trajectory of a particle from a sequence of noisy images. This section is devoted to the description of our algorithm.

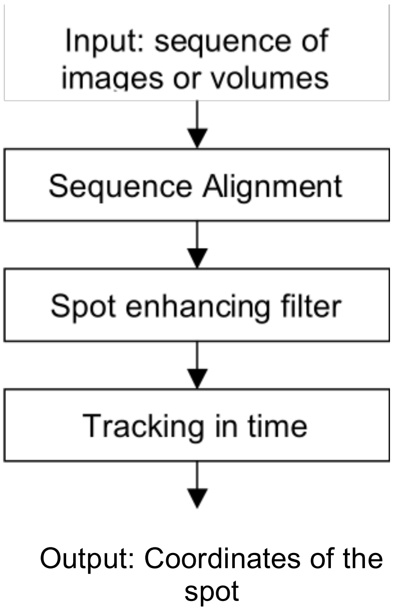

Input: sequence of images or volumes
Sequence Alignment
Spot enhancing filter
Tracking in time
Output: Coordinates of the spot

Fig. 1. Overview of the image analysis procedure.

The processing is performed in three steps, as summarized in the block diagram in Fig. 1. The first component is an alignment module that compensates for the movement of the biological structure under investigation. The second component is a matched filter that is designed to enhance the signatures of the fluorescent markers (spot-like particles) and to reduce the effect of noise. The final component is a tracking algorithm that uses dynamic programming to extract the optimal space-time trajectory (x,y,z,t) of a particle in 3D. The same scheme works for processing 2D or 3D data sequences, but the computational cost increases in proportion to the added degrees of freedom (i.e., the z-component of the trajectory in 3D) and the size of the data set (images versus volumes).

A. Sequence alignment

In practice, one is usually interested in characterizing the movement of a fluorescent spot in relation to some reference structure or background. The background is typically also fluorescently labelled so that it remains visible throughout the process, giving a reference position. The practical difficulty is that the reference does not necessarily stay still during acquisition. We compensate for this effect by applying a realignment algorithm in a pre-processing step. There are two possible approaches at our disposal:

1) When the reference structure is well preserved from one image to the next, we can select one image of the sequence as reference frame and use it to register the other ones. A good registration method is described in the work of Thévenaz [16]. The proposed algorithm is entirely automatic; it is pixel-based and does not require any landmarks. The algorithm is precise, reproducible and reasonably fast, thanks to the use of an efficient multi-resolution optimization strategy, and it is freely available as a plug-in for ImageJ at http://bigwww.epfl.ch/algorithms.html. It can deal with both translational and rotational alignments.

2) When the shape of the reference structure is somewhat variable but is reasonably well described by a curve within some parametric family (e.g. circle, ellipse, or parametric snake), it may be appropriate to consider a detector that is specifically designed for the extraction of such shapes. The detector is then applied to each frame, providing the center position of the reference structure; the images are then realigned translationally with respect to this position. We have chosen this approach for the application in Section 4 because the reference structure (nucleus) is well represented by an ellipse.

B. Spot enhancing filter

The practical difficulty in dynamic fluorescence microscopy is that the data is extremely noisy, due to data acquisition at the detection limit, which is performed to maximize the number of frames while minimizing the destructive effect of laser-induced damage or signal loss due to photo-bleaching. Moreover, one has to take into account background fluorescence and the presence of other structures in the image with levels of intensity very similar to the particles of interest. To improve the robustness of the tracking algorithm, we include a pre-filter that enhances the pattern of interest (particles) while reducing the background and the effect of noise.

In our case, we show that the optimal detector (whitened matched filter) is well approximated by the Laplacian of a Gaussian (LoG).

To this end, we assume that the input signal is given by: $y(\boldsymbol{x}) = \alpha \cdot g(\boldsymbol{x} - \boldsymbol{x}_0) + n(\boldsymbol{x})$ where α is an unknown intensity factor, $\mathbf{x}$ = (x,y,z) is the spatial coordinate, $g(\mathbf{x})$ is a reference model of the particle and $\mathbf{x}_0$ is the particle's position that needs to be determined; $n(\boldsymbol{x})$ is an additive independent noise component that also includes the background structures.

We have found experimentally that the spectral power density of fluorescence microscopy images is isotropic and is well represented by a power law $S_n(\omega) \propto \|\omega\|^{-s}$ where ω is the radial spatial frequency. This corresponds to a multidimensional fractional Brownian motion model [17], which is also typical of a board category of natural images [18]. In our case, we estimate the fractal exponent $s = 1.8$ which is not too different from 2, as justified in Fig. 2. For our implementaion, we have selected the integer exponent value 2 because it leads to a fast Laplacian-based algorithm using separable filters.

Our goal is to now to specify the filter that will produce a maximum at $\mathbf{x} = \mathbf{x}_0$, while minimizing the effect of noise. As is well known from estimation theory [19], the maximum SNR detector is provided by the whitened matched filter. In our case, this solution is obtained as follows:

1) First, one applies the pre-filter $H(\vec{\omega}) = \|\vec{\omega}\|^{s/2}$ that has the effect of transforming the additive $1/\omega^s$ noise component into white noise that is completely uncorrelated.

2) We are then faced with the standard problem of the detection of a known signal in white noise, which can be solved by means of a standard correlation detector. The frequency response of this matched filter is the complex conjugate of the Fourier transform of signal to be detected: $\|\vec{\omega}\|^{s/2} G^*(\vec{\omega})$.

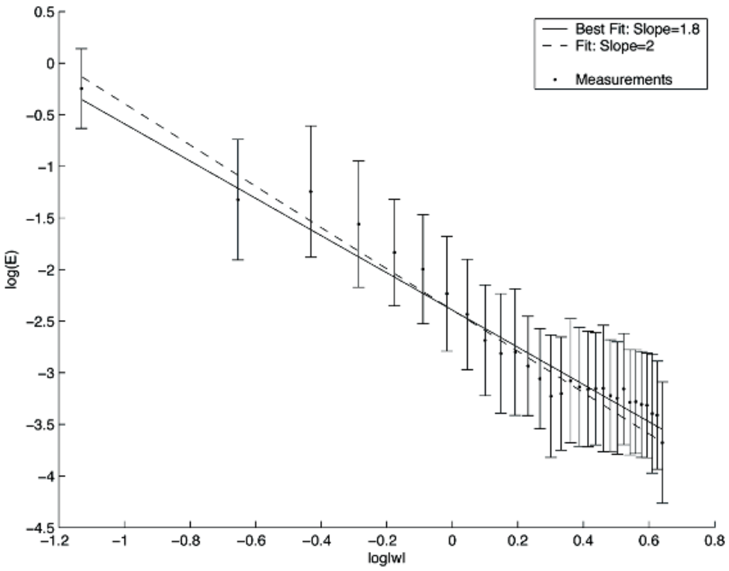


Fig. 2. Estimation of the Fractal exponent *s* of the noise. The radial power spectrum is estimated by averaging the square modulus of the Fourier transform of a noise-only image (including background) over a series of concentric annuli indexed by ω. The results are represented in a log-log plot and fitted with the theoretical power law (straight line). The data points are the average square intensities within each annulus, while the error bars represent the standard deviation over the radial annulus. The optimal fit is $s = 1.8$.

The pre-filter and the optimal detector are combined into a single filter (whitened matched filter), as illustrated in the block diagram in Fig. 3. Going back to the spatial domain, this corresponds to a correlation with a detector whose response is proportional to $\Delta^{s/2} g(\mathbf{x})$ where $\Delta^{s/2}$ is the $s/2$-fractional iterate of the Laplacian operator.

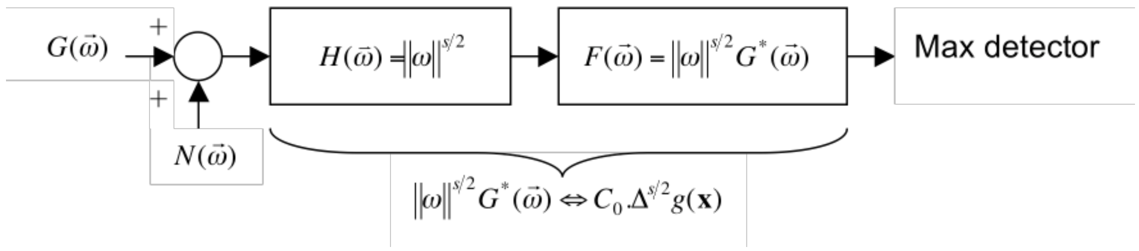

$G(\vec{\omega})$
+
+
$N(\vec{\omega})$
$H(\vec{\omega}) = \|\omega\|^{s/2}$
$F(\vec{\omega}) = \|\omega\|^{s/2} G^*(\vec{\omega})$
Max detector
$\|\omega\|^{s/2} G^*(\vec{\omega}) \Leftrightarrow C_0 . \Delta^{s/2} g(\mathbf{x})$

Fig. 3. Block diagram of the whitened matched filter

To obtain a practical implementation of this operator, we consider the following 3D Gaussian model of the particle:

$$g(\mathbf{x}) = g(x,y,z) = \frac{1}{(2\pi)^{\frac{3}{2}}\left(\sigma_x\sigma_y\sigma_z\right)} e^{-\frac{x^2}{2\sigma_x{}^2}-\frac{y^2}{2\sigma_y{}^2}-\frac{z^2}{2\sigma_z{}^2}}$$

which is parameterized in terms of the standard deviations $(\sigma_x,\sigma_y,\sigma_z)$ for each axis to allow for non-isotropic voxel sizes; typically, in microscopy, the resolution along the Z-axis is much coarser than along the XY-axes.

If we now assume that $s = 2$, which is still compatible with our observations, we obtain an optimal detector that corresponds to the Laplacian of a Gaussian (LoG), also known as the "Mexican Hat". The explicit space-domain formula of this filter in 3D is:

$$LoG = \Delta\, g(x,y,z) = \frac{1}{(2\pi)^{\frac{3}{2}}\left(\sigma_x\sigma_y\sigma_z\right)} \left(\frac{x^2}{\sigma_x{}^4}-\frac{1}{\sigma_x{}^2}+\frac{y^2}{\sigma_y{}^4}-\frac{1}{\sigma_y{}^2}+\frac{z^2}{\sigma_z{}^4}-\frac{1}{\sigma_z{}^2}\right) e^{-\frac{x^2}{2\sigma_x{}^2}-\frac{y^2}{2\sigma_y{}^2}-\frac{z^2}{2\sigma_z{}^2}}$$

The filter parameters (standard deviations) obviously need to be tuned to the size of the fluorescent particles. Pre-filtering the images with this particular filter is optimal for detection purposes; it has the desirable effect of enhancing the spots while at the same time getting rid of some of the background structures (see. Fig. 4).

A final aspect that needs to be dealt with is computational speed. Particularly relevant to the issue is the work of Huertas and Medioni [20] who developed a separable version of the LoG filter for the 2D case which yields a fast algorithm based on successive 1D convolutions along the rows and columns of the image. For 3D data, it is even more advantageous to have a separable implementation to make the computation time acceptable. Here, we extend the formulation of Huertas to the 3D case and express the LoG detector as a sum of 3 separable filters:

$$LoG = \nabla^2 G(x,y,z) = h_{F,\sigma_x}(x).h_{B,\sigma_y}(y).h_{B,\sigma_z}(z) + h_{B,\sigma x}(x).h_{F,\sigma_y}(y).h_{B,\sigma_z}(z) + h_{B,\sigma_x}(x).h_{B,\sigma_y}(y).h_{F,\sigma_z}(z)$$

where $h_{F,\sigma_i}(x) = \frac{1}{(2\pi)^{\frac{3}{2}}\left(\sigma_x\sigma_y\sigma_z\right)} \left(\frac{x^2}{\sigma_i^4}-\frac{1}{\sigma_i^2}\right) e^{-\frac{x^2}{2\sigma_i{}^2}}$ and $h_{B,\sigma_i}(x) = e^{-\frac{x^2}{2\sigma_i{}^2}}$

This new separable implementation of the 3D LoG filter speeds up computation time dramatically. For a (100*100*100) volume data and a LoG detector with $\left(\sigma_x = 3,\ \sigma_y = 3,\ \sigma_z = 3,\right)$, we decrease

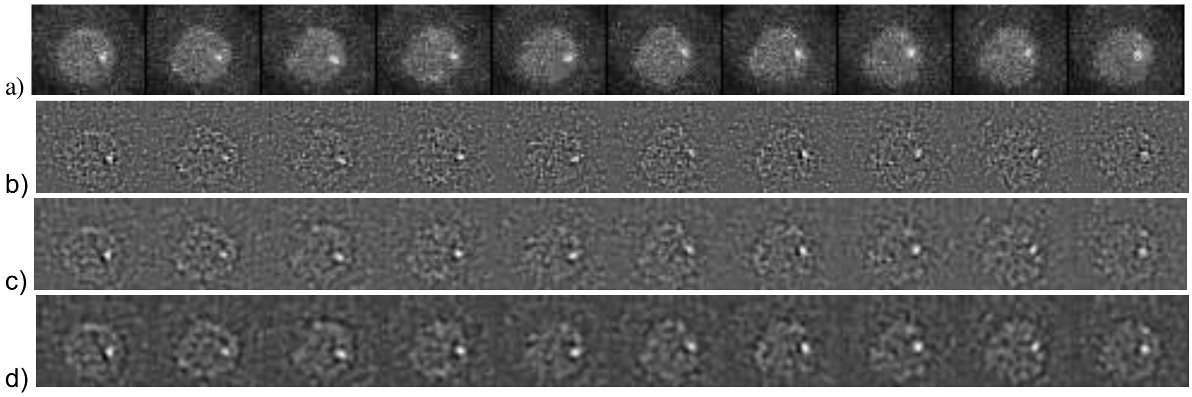


Fig. 4. Typical results of the 2D LoG filter on the 10 first images of a sequence. a) input image sequence (2D+T) of a tagged telomere in a nucleus (47*38 pixels) b) output of the LoG with σ = 0.75 ; c) output of the LoG with σ = 1.00; d) output of the LoG with σ = 1.25.

the time from 145 sec. (for a non-separable implementation in the space domain) to 2.3 sec. for the separable algorithm on an Apple PowerMac DP G5 2.0 GHz.

C. Single particle tracking using Dynamic Programming (DP)

The tracking specifications that are the basis for our formulation are the following: 1) There is one single spot to track over time; the difficulty is that it may be dim or even absent in some parts of the sequence. 2) The movement of the spot is limited to a few pixels from frame to frame. 3) The tracking takes place off line when the whole data has been acquired.

Although the data can be very noisy, *in vivo* movement is generally sufficiently constrained to be formulated as a global optimization problem. In particular, we take advantage of the strong dependency of spot position in one image on the next one (chaining property) and solve the problem using dynamic programming (DP).

We are given a sequence $f(\mathbf{x},t)$ of N preprocessed images. The space $\mathbf{x}=(x,y,z)$ and time (t) indices take integer values with $x=0,\cdots,N_x-1$, $y=0,\cdots,N_y-1, z=0,\cdots,N_z-1$ and $t=0,\cdots,N-1$, respectively. The task is to find an optimal time-trajectory $\{\mathbf{x}_t\}_{t=0,\cdots,N-1}$ that describes the displacement of the particle. A key constraint is that the maximum excursion from one frame to the next is limited: $\|x_t - x_{t+1}\| \le \Delta_x$, $\|y_t - x_{t+1}\| \le \Delta_y$, $\|z_t - z_{t+1}\| \le \Delta_z$. and Δ_x, Δ_y, Δ_z are some user-specified parameters. Since we are considering a discrete grid in space and time, it is possible, at least in principle, to enumerate all possible trajectories and to attempt to select the best possible one based on some objective criterion.

The cost function $\xi(\mathbf{x}_0,\mathbf{x}_1,\cdots\mathbf{x}_{N-1})$ to maximize must incorporate as much problem-specific information as possible. In particular, we would like to favor positions where the intensity is bright or, alternatively, where the response to the spot-enhancing filter is strong and to favor small displacements; this can be achieved by penalizing paths for which the average displacement from one frame to the next is large. These various constraints can be incorporated by defining the following cost function:

$$\xi(\boldsymbol{x}_0,\boldsymbol{x}_1,\cdots\boldsymbol{x}_{N-1}) = \sum_{t=1}^{N-1}(1-\lambda)f(\boldsymbol{x}_t,t)-\lambda\frac{\left\|\boldsymbol{x}_t-\boldsymbol{x}_{t-1}\right\|}{M} \qquad \text{(Eq. 1)}$$

where M is an appropriate normalization factor and where λ $\left(0\leq\lambda\leq1\right)$ is a smoothness-controlling weight that can be adjusted by the user (default value $\lambda = 0.25$).

For each point on the path at time *t*, there is only a cost contribution associated to its position $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$. The implication is that the corresponding discrete optimization problem can be solved most efficiently by dynamic programming.

The thrust of the algorithm is a main loop for $t=0,\cdots,N-1$ whereby all potential (*x*,*y*,*z*,*t*) candidates are examined once only. At each position, the algorithm applies an iterative update formula for computing the maximal cost for reaching the current state. At the end of the process, the optimal solution is retrieved by back tracking. To achieve sub-pixel accuracy, each detected spot position is finally replaced by the center of gravity of a small neighborhood (3*3) in a post-processing step.

When the spot is absent for a period, the trajectory is mainly built using the contribution of the displacement penalization of the cost function given a smooth trajectory.

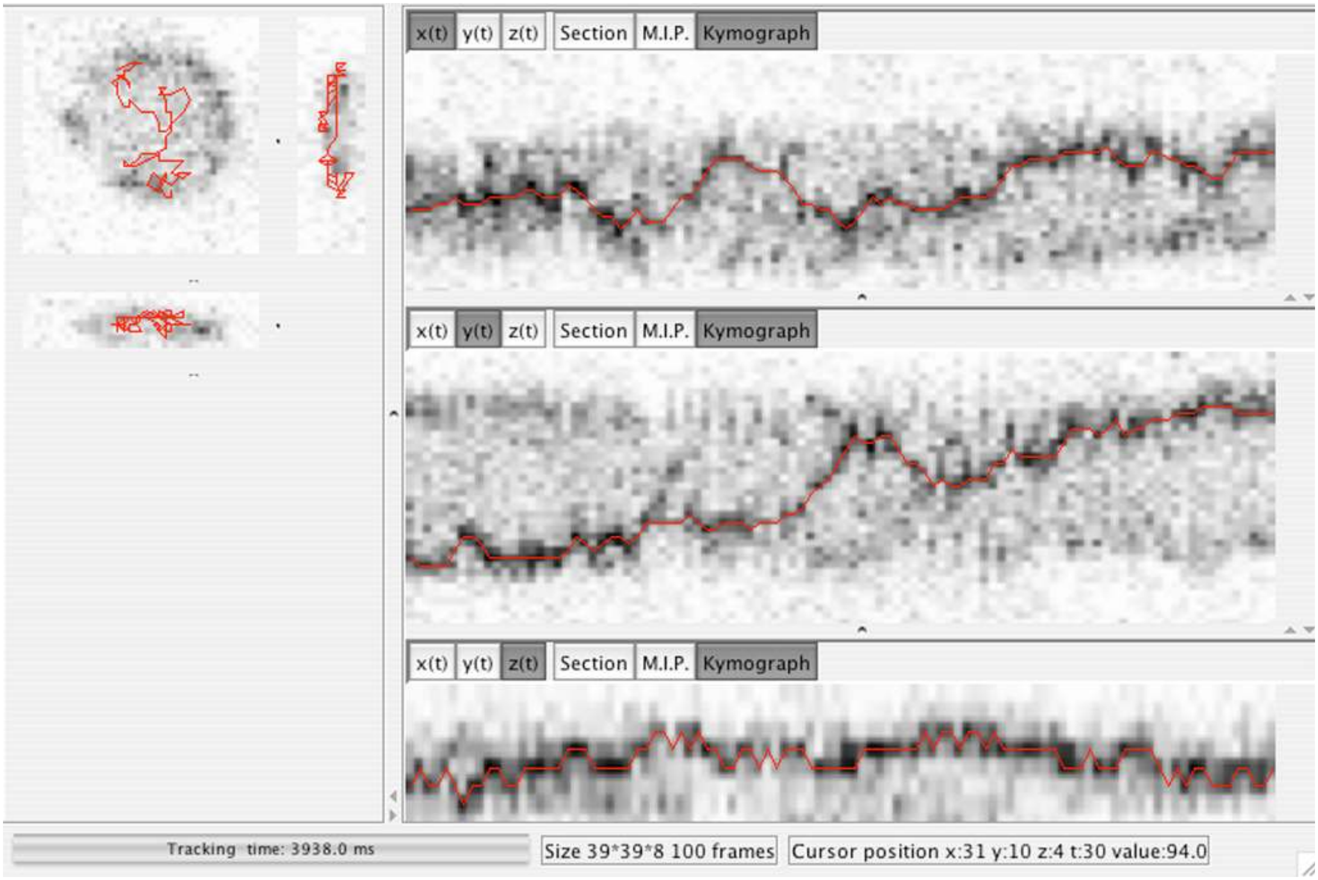
Likewise, it is easy to introduce arbitrary hard constraints on the path by subdividing it into segments. We have the option of constraining the optimization further by specifying additional nodes. In this case, the DP algorithm is applied independently for each segment so as to satisfy the end constraints (nodes). To run faster, the optimal trajectory is recomputed only on the segments adjacent to the new node.

The DP algorithm provides a very attractive solution for the single particle tracking problem. It may eventually be applied to the case of multiple particles by extracting them one by one, keeping in mind that this approach is no more optimal globally. Presently, we do not yet know of a fully satisfactory solution to the multiple particle tracking problem because of combinatory explosion.

III. Experimental Results

A. Implementation and example

Both 2D and 3D versions of the algorithm were implemented as plug-ins for ImageJ [21, 22], a public-domain software package for image processing. The plug-ins, called SpotTracker[1], come with a user-friendly interface with provisions for the interactive display of the results and the editing of the trajectory. This allows for the validation and correction of the automated analysis, which is particularly important in cases of noisy data. An example of the output of a 3D analysis is shown Fig. 5.



Tracking time: 3938.0 ms

Size 39*39*8 100 frames

Cursor position x:31 y:10 z:4 t:30 value:94.0

Fig. 5 Screenshot of the plug-in for ImageJ. On the left side, there are the 3 orthogonal sections XY, XZ, ZY at frame t=30. On the right side there are 3 kymograph sections x(t), y(t) and z(t). The detected trajectory is overlaid in red. The contrast is inverted to facilitate the visualization.

[1] The plug-in for ImageJ, SpotTracker is freely available to the research community at: http://bigwww.epfl.ch/spottracker/

B. Validation of the tracking in presence of noise

For the validation of the tracking, we have processed simulated data to evaluate the tracking performances in presence of noise, as well as real data to compare automatic tracking with manual tracking.

The evaluation was performed on simulated sequences of 240 images of 60*60 pixels organized in 10 segments of 24 images each. The sequences contain a single moving particle approximated by a 2D Gaussian function which moves along a known trajectory. To evaluate the limits of the tracking algorithm, we degraded the images of a sequence by increasing the level of noise from one segment to the next (see. Figs. 6 and 7). The level of Gaussian noise is characterized by the peak signal-to-noise ratio (PSNR) $PSNR_k = 20\ \log\left(\frac{A}{\sigma_k}\right)$ where σ_k is the noise variance for segment *k* and A is the amplitude of the Gaussian-shape particle.

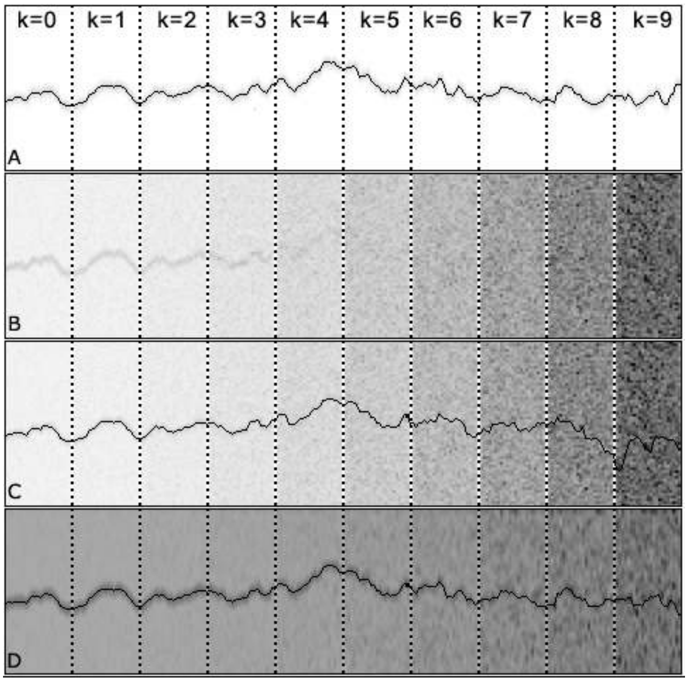


Fig. 6. Result of tracking when the PSNR is degraded by increasing the level of noise in 10 segments of 24 images. The trajectories, which were detected automatically with our program SpotTracker, are overlaid in black on the maximum intensity projection images on y(t). A. Reference sequence. B. Test sequence. C. Test sequence and trajectory. D. Test sequence and trajectory after spot enhancing pre-filtering with the above presented LoG filter. The intensity is inverted to facilitate the visualization.

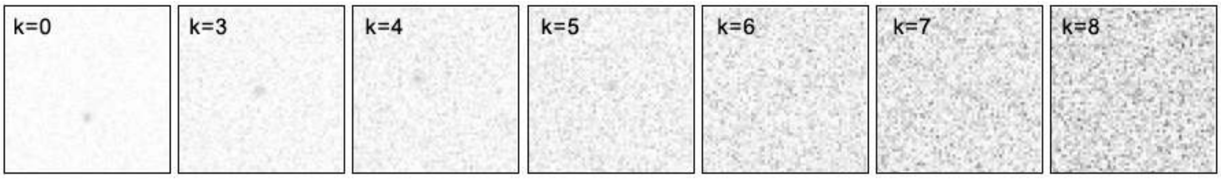


Fig. 7. Images taken in segment k of the test sequence. Even though the spot starts being imperceptible for $k \geq 6$, the algorithm is able to track it nonetheless. The intensity is inverted to facilitate the visualization.

The reference trajectory is compared to the manual one and to the ones detected by the algorithm with and without pre-filtering (spot enhancing filter). The results summarized in the Fig. 8 show that the automatic procedure consistently outperforms the manual tracing. The human observer is unable to trace a particle when the PSNR falls below 9 dB, while the automatic procedure can handle noise levels down to 0 dB (resp. 6 dB) with (resp. without) pre-filtering. Our spot enhancing pre-filter, Laplacian of Gaussian (LoG), clearly improves the performances. Overall, the automatic procedure is capable of reducing the estimation error by a factor of 2, irrespective of the noise level.

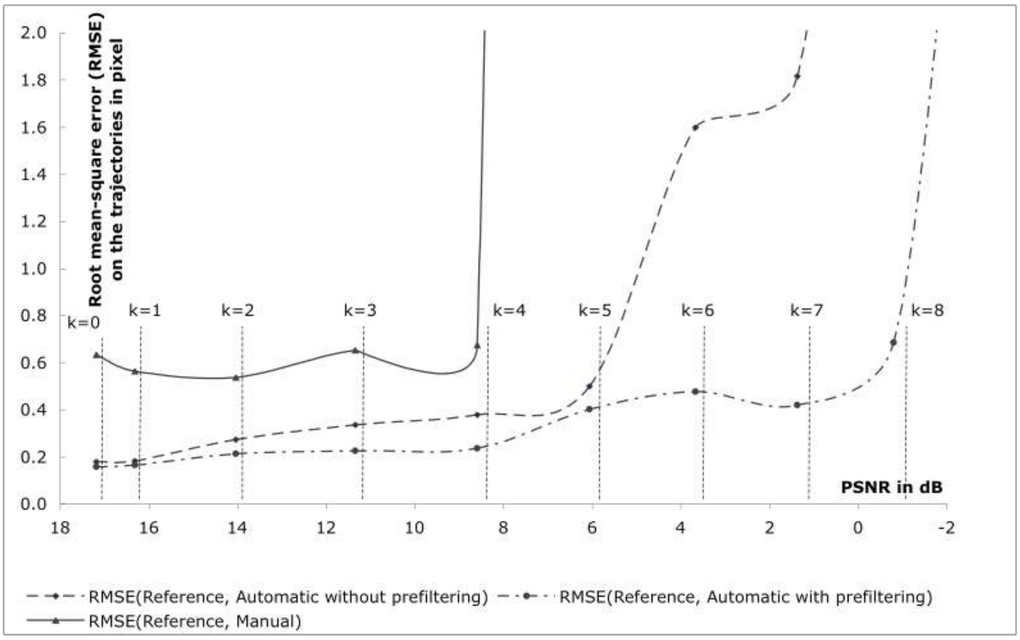


Fig. 8. RMSE on the trajectories when the PSNR is degraded by increasing the level of noise.

C. Comparison of the manual and automatic results on real data

We had four human observers (o=1..4) manually trace a particle on two biological sequences (s=1 and s=2) of 200 2D images. They repeated the experiment 3 times (r=1..3), producing a total of 24 trajectories $\boldsymbol{T}_s\ \{\mathbf{x}_{s,o,r}(t)\}$. For each image t of the sequence s, we defined the interobserver and intraobserver variabilities as:

$$\mathrm{V}_{\text{inter}, s}(t) = \frac{1}{18} \sum_{r=1}^{3} \sum_{i=1}^{4} \sum_{j=i+1}^{4} \left\| \mathbf{x}_{s,i,r}(t) - \mathbf{x}_{s,j,r}(t) \right\|$$

$$\mathrm{V}_{\text{intra, s}}(t) = \frac{1}{12} \sum_{o=1}^{4} \sum_{i=1}^{3} \sum_{j=i+1}^{3} \left\| \mathbf{x}_{s,o,i}(t) - \mathbf{x}_{s,o,j}(t) \right\|$$

The corresponding statistics are reported in Table 1.

		Sequence 1	**Sequence 2**
Number of images in the sequence		200	200
Variability interobserver	Maximum	35.05	43.46
	Mean	3.06	4.81
	Standard deviation	8.11	12.09
Variability intraobserver	Maximum	2.81	6.07
	Mean	0.85	0.95
	Standard deviation	0.43	0.66

Table 1. Variability of the manual tracings of the two test-sequences

Due to the high interobserver variability, we restricted the comparison of the automatic tracings with the mean of the manual trajectories to the set of time frames for which the four observers were in reasonable agreement; i.e., $\mathrm{V}_{\mathrm{inter,s}}(\mathrm{t}) < 2$ (see Table 2) .

	Sequence 1		**Sequence 2**	
Number of images in the agreement region	111		120	
RMSE manual vs. automatic	x	y	x	y
in pixels (Metamorph)	1.88	3.23	1.22	1.00
RMSE manual vs. automatic	x	y	x	y
in pixels (SpotTracker)	0.42	0.60	0.52	0.53

Table 2. Root Mean-Square Error (RMSE) of manual tracings and automatic tracings in the agreement region. The manual tracings are obtained with four human observers. The automatic tracings are obtained with MetaMorph and with our algorithm (SpotTracker).Here, we compare two automatic tracking programs: a standard approach (correlation + frame-to-frame tracking) taken by the commercial software MetaMorph of Universal Imaging Corporation [12] and the one proposed in this paper; i.e., SpotTracker. This benchmark is clearly in favor of our tracking algorithm that yields a RMSE at least inferior by a factor of 2 when compared to the correlation approach. This is not overly surprising because the detection mechanism in MetaMorph is local, as opposed to being global in our approach. These results indicate that the automatic tracings are quite consistent with the manual

tracings when there is agreement among observers. They are obviously also more reproducible than manual tracking.

D. Computational aspects of the tracking

Even though DP is a systematic approach that evaluates all allowable trajectories, it gives the answer quite rapidly for applications with small images. The most time-consuming part of the algorithm is the large number of evaluations (N_{EVAL}) of the cost function for every transition. For n_t volumes of size [n_x, n_y, n_z] and a movement constraint vector (Δ_x, Δ_y, Δ_z) the number of evaluations is:

$$N_{\mathrm{EVAL}} = \left[\left(n_x - \Delta_x\right)R_x + \Delta_x^2\right] \cdot \left[\left(n_y - \Delta_y\right)R_y + \Delta_y^2\right] \cdot \left[\left(n_z - \Delta_z\right)R_z + \Delta_z^2\right].(n_t - 1) \;\; \textit{where} \;\; R_d = 2\Delta_d + 1$$

In Table 3, we report examples of computation timings. From this experiment, we can compute the unitary time to evaluate one transition: this gives 0.025 μs in 2D and 0.037 μs in 3D on an Apple PowerMac DP G5 2.0 GHz.

	nx ny nz [pixel]	nt Nb of frames	$\Delta_x = \Delta_y = \Delta_z$ [pixel]	nn Nb of nodes	time [sec.]
Manual tracing	50 x 50 x 1	1000	-	-	1500.00
	50 x 50 x 1	200	-	-	300.00
Automatic tracing (SpotTracker)	50 x 50 x 1	200	1	1	0.15
	50 x 50 x 1	200	3	1	0.55
	50 x 50 x 1	200	5	1	1.15
	100 x 100 x 1	200	3	1	2.07
	50 x 50 x 1	1000	3	1	2.53
	50 x 50 x 1	200	3	10	0.30
	50 x 50 x 10	100	1	1	2.89
	50 x 50 x 10	100	3	1	22.55
	50 x 50 x 10	100	5	1	69.84
	50 x 50 x 10	200	3	1	47.74
	50 x 50 x 20	100	3	1	52.75

Table 3. Typical computation timings for different image sizes and settings. of the SpotTracker program. *nx*, *ny* and *nz* are the size of the volume (if *nz* =1, it is only 2D sequence). *nt* is the number of images (or volumes) of the sequence. Δ_x, Δ_y, Δ_z are the movement constraint given in pixels. *nn* is the number of nodes imposed by the user. These experiments are done on an Apple PowerMac DP G5 2.0 GHz.

In 2D, the method is fast enough to give an almost immediate feedback. The response time increases significantly in 3D, but it is still manageable, and certainly a few orders of magnitude faster than a manual tracing.

IV. Application: The Study of the Chromosome Dynamics in Budding Yeast

The biological application that motivated the development of the algorithm presented above is the study of the chromosome dynamics. Here, we provide some biological background on the imaging of chromosomal markers, discuss the microscopy set-up and the customized version of the tracking algorithm, and present biological results that were obtained with our software.

A. Telomere localization

Recently, increasing evidence suggests that the nucleus is highly organized into various sub-compartments, probably to regulate nuclear functions [14]. The visualisation of specific DNA sequences in living cells, achieved through the integration of *lac* operator arrays (*lac*$^{\mathrm{op}}$) and expression of a GFP-lac repressor fusion, has provided new tools to examine how the nucleus is organised and how basic events like sister chromatid separation occur [3, 23] . The genetic information of a haploid budding yeast cell is encoded on 16 linear chromosomes. The interphase nucleus is organized into functional sub-compartments, one of which results from the clustering at the nuclear periphery of the genome's 32 telomeres into groups of 4-5 telomeres. This particular organization creates high local concentrations of silencing factors that bind telomeres and promote transcriptional silencing. Despite this positioning, individual yeast telomeres show fairly high rates of mobility within constrained volumes. These characteristics raised many questions about the dynamics of telomeres that could not be answered by classical immuno-microscopy of fixed cells, and has led to the use of live microscopy to analyze the mobility of specific telomeres in a dividing cell, their relationship to the nuclear periphery, to other nuclear landmarks, and to each other. The example discussed below is the tracking of the telomere of the right arm of yeast chromosome 6 [17], keeping in mind that the algorithm is applicable to any marked chromosomal locus (see Fig. 9).

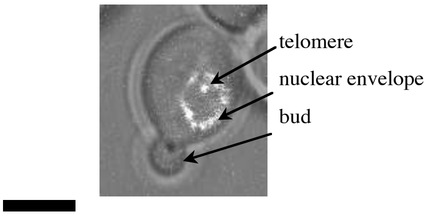


Fig. 9. Image of a yeast cell expressing Nup49-GFP (nuclear envelope) in which Tel VI-R has been tagged. This cell is in a synthesis stage characterized by a small bud. Bar is 2µm.

B. Image acquisition protocol

The strains used in this study contain multimerised *lac*^{op} arrays (usually 256 copies or ~10 kb) integrated at the ends of chromosomes by standard transformation using a linearised construct that integrates by homologous recombination. The array was visualized by expression of a LacR-GFP protein fusion. The nuclear envelope is visualized through expression of a nuclear pore protein fused to GFP, which was also created by integrative transformation with a linear plasmid (Nup49-GFP [15, 24]). Alternatively, cells can carry a fusion of the bacterial *tet repressor* fused to GFP, which, in the absence of tet operator sites creates a diffuse nucleoplasmic signal. This allows determination of the nuclear center, which can be used as reference structure for tracking with the described algorithm. All yeast strains were cultured identically and preferably to an early exponential phase of growth ($<0.5\times10^7$ cells/ml) in synthetic or YPD medium, starting from a fresh overnight culture. Cells were mounted on a depression slide (Milian SA) upon 1.4% agarose containing SD medium with 4% glucose (Fluka). Alternatively, cells were immobilised on a 18mm coverslip.

Confocal microscopy was performed on a Zeiss LSM510 Axiovert 200M, equipped with a Zeiss Plan-Apochromat 100x/ NA=1.4 oil immersion or a Plan-Fluar 100x/ NA=1.45 oil immersion objective. The stage was equipped with a hyperfine motor HRZ 200. Temperature was stabilised using a temperature regulated box surrounding the microscope (The Box, Life Imaging Services). To follow rapid chromatin dynamics in individual cells while maintaining maximal sensitivity and reducing the risk of damage by illumination, a minimal region of interest has to be chosen; scanning speeds, pinhole aperture have to be adjusted; and laser intensities have to be reduced to very low levels. 2D-time-lapse images were acquired, taking one frame every 1.5s. The focus was adjusted manually by an experienced microscopist for best visualization of the spot. As position and mobility of a chromosomal locus can vary with stages of the cell cycle, it is crucial to determine precisely what stage each imaged cell is in. This is done by monitoring bud presence, bud size, as well as the shape and position of the nucleus, as visualised by the Nup49-GFP fusion and a transmission or phase image [25].

C. Image analysis software

A version of the SpotTracker (described in section 2) was customized for the application working on the 2D time-lapse sequence. Here we describe the specificities of the three modules which has essentially the same components as the generic program.

1) Nucleus alignment: The reference structure for our application is the nuclear membrane visualized due to Nup49-GFP, and whose outline on an image is well approximated by a ellipse. We first smooth the image and apply a global threshold to detect the fluorescent structures that are predominantly located in the nuclear membrane. We then fit the thresholded data with an ellipse, using the least squares method of Fitzgibbon which works well for scattered data [26]. The images are then translationally realigned with respect to the center of these ellipses and also cropped to the largest radius. Note that this center detection is almost insensitive to the choice of the threshold.

2) Spot enhancing filter: We use the LoG filter (Mexican Hat) described in Section 2.B. The standard deviation parameter of the filter is tuned to the size of the fluorescent spots (e.g., $\sigma_x = \sigma_y = 1.1$).

3) Tracking using DP: One difficulty that we had to cope with is the labeling of envelope proteins which creates a bright halo at the periphery of the nucleus which can detract the algorithm from tracking the particle which has a similar intensity. To deal with this issue, we modified the cost function (Eq. 1) by adding a third term which penalizes spot positions $\boldsymbol{x}_t$ that are too far from the center of the nucleus $\boldsymbol{x}_c$. Specifically, we used the augmented cost function:

$$\xi(\boldsymbol{x}_0,\boldsymbol{x}_1,\cdots \boldsymbol{x}_{N-1}) = \sum_{t=1}^{N-1}(1-\lambda_1-\lambda_2)f(\boldsymbol{x}_t,t)-\lambda_1\frac{\left\|\boldsymbol{x}_t-\boldsymbol{x}_{t-1}\right\|}{M_1}-\lambda_2\frac{\left\|\boldsymbol{x}_t-\boldsymbol{x}_c\right\|}{M_2}$$

where M_1 and M_2 are suitable normalization factors and where λ_1 and λ_2 with $0 \le \lambda_1 + \lambda_2 \le 1$ are adjustable weights. We note that the outcome of the algorithm is not overly sensitive to the setting of these parameters; typical values are $\Delta = 3$, $\lambda_1 = 25\%$, and $\lambda_2 = 10\%$. Increasing λ_1 makes the trajectory smoother while λ_2 is merely a fallback parameter which is only relevant when the algorithm tends to get hooked on nuclear proteins.

In its standard formulation, the DP approach requires the specification of an initial point ($\mathbf{x}_0$) for the path, which can be located at any t. In our implementation, we have specified this initial point automatically by detecting the strongest response to the matched filter across all images in the sequence; this corresponds to the most likely position of the particle overall.. The DP algorithm is then run twice from this initial point to the beginning and end of the sequence, respectively. This initialization procedure was used in all our experiments and was found to give good and reproducible results. It is obviously also possible to define this initial point manually, but this was not found to be necessary. The program has a graphical user-interface which allows the placement of additional nodes. Thanks to this interactive mode, the biologist expert can control the trajectory and work hand-in-hand with the algorithm when the quality of data is poor.

We have successfully applied the algorithm to a large variety of real data sets. The center of the nucleus was well located in all cases, and the detected trajectories were judged to be satisfactory, even when the images were extremely noisy. When the spot disappears completely for a short period - e.g. when it is out of focus - the algorithm is able to follow it nonetheless using the displacement constraint. On the other hand, it fails when no spot is recognized for extended periods of time; this can happen when the particle is out of focus or inside the nucleus membrane. In this case, the user can easily enter new node points to further constrain the trajectory.

D. Analysis of the mobility of the telomeres

The accuracy and speed of the tracking analysis has permitted a calculation of the absolute dynamics of a given locus by subtracting background nuclear and/or cellular movement from movement of the locus. This is in contrast to previous studies of *in vivo* chromatin dynamics. Movement in the past was either determined as the difference in position of two tagged loci [27-29] or as movement of a single spot in relation to a nuclear reference point, usually chosen as the closest point on the nuclear envelope [15, 24]. For measurements relative to the nuclear envelope, one can take the complement of the distance between a spot and the nuclear envelope, which is its distance from the nuclear center. The changes in this distance reflect the dynamics of a given locus relative to the nuclear periphery (measurement definitions shown in Fig. 10A).

To make a robust comparison of different telomeres and the effects of ablating genes implicated in telomere tethering at the nuclear envelope, the degree of movement must be averaged over many time-lapse series captured under identical conditions from different cells of identical genotype. In practical terms, the mean of distance traveled (or displacement) is calculated from these pooled time-lapse series, and its square is plotted against increasing time intervals $MSD(\Delta t) = E\left\{\left(d(t_0) - d(t_0 + \Delta t)\right)^2\right\}$. When d is calculated as differences in radial position, a particle positioned near the nuclear periphery will always yield low mean squared displacement (MSD). Therefore, this so-called radial MSD (radMSD) is useful primarily to compare the extent of displacement from the nuclear periphery. By using the SpotTracker software to efficiently extract absolute movements of a locus (Fig. 10A), we can detect more subtle differences in the dynamics of telomeres, and from this calculate a true or absolute MSD (absMSD).

We have performed these two analyses on multiple time-lapse sequences of three different telomeres (Tel 6R, 8L and 14L). We have also analyzed Tel 6R movement in a strain lacking one of the proteins that mediates association with the nuclear envelope (Tel 6R *yku70* Δ). From the radial MSD curve in Fig. 10D, we can observe that all three telomeres have a plateau at the same height, except in the strain lacking the telomeric anchoring protein yKu (*yku70Δ)*, for which the deviation is much higher. As the MSD plateau is proportional to the volume of confinement, these curves argue that these three telomeres are constrained to the same extent relative to the nuclear periphery and that elimination of yKu70 releases Tel 6R from this constraint. This nicely correlates with subnuclear position analysis, in which telomere position is determined in stacks of many cells at a single time-point [24]. The calculation of the absMSD, on the other hand, shows that some telomeres, such as Tel 14L, move significantly but remain near the nuclear envelope. Tel 14L mobility is nonetheless lower than that monitored for Tel 6R in the absence of the telomere anchor, yKu70 (Figs. 10B-D). These differences convincingly indicate that anchorage at the nuclear periphery can vary among telomeres, yet can be manipulated by eliminating the telomere-bound protein (yKu) which mediates interaction with the nuclear envelope [24].

This represents but one example of the biological information that can be gathered from this type of analysis. To date, we have processed hundreds of such image sequences in a wide range of mutant cells and have performed a relatively detailed analysis of the results (to be presented elsewhere). A highly precise analysis of the dynamics of telomeres demonstrates that different anchoring mechanisms constrain individual telomeres to different degrees, and that this constraint requires protein-protein interactions at the nuclear envelope. These variations in telomere dynamics may correlate with different propensities for recombination or silencing efficiency.

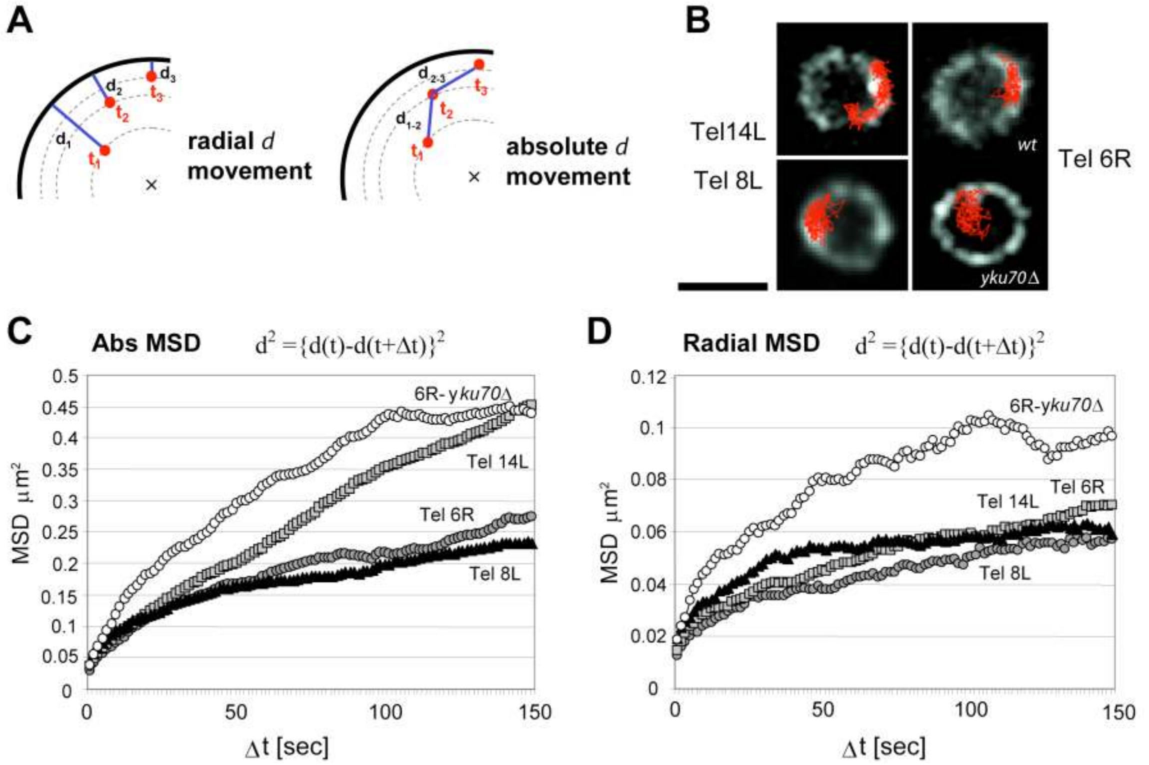


Fig. 10.
A. Schematic representation of distances measured for MSD analysis. For radial analysis, distances are measured for each time point from the center of the spot to the nearest site at the nuclear periphery. For absolute analysis, distance traversed by the chromatin between two time points is measured for each time interval. Bar is 2µm.
B. The position of the indicated telomere focus was marked for each time point for a typical time lapse (200 frames at 1.5s intervals) of G1 phase cells. The compiled trajectory over 5 min. is indicated in red on an idealized section of the nucleus for telomeres 14L (GA1985), 6R (GA1459), 8L (GA1986) and 6R *Δku70* (GA1489).
C. Absolute mean squared displacement (absMSD) analysis was performed by first correcting cellular / nuclear shift (absolute coordinates of the spot are obtained by subtracting the center coordinates from the measured spot coordinates). Distances between positions at different time points are then calculated for each time interval (1.5s-150s., example is shown in A for 1.5s) and the mean square is plotted against time interval (e.g. $\Delta t = 1.5s$: absMSD= mean($d_{1\text{-}2}^2$, $d_{2\text{-}3}^2$,…), $\Delta t = 3s$: absMSD= mean($d_{1\text{-}3}^2$, $d_{2\text{-}4}^2$,…). absMSD are calculated over multiple time lapse series for the indicated telomeres, notably Tel14L, Tel6R, Tel8L and Tel6R in a *Δku70* strain. The height of the MSD plateau is proportional to the radius of constraint and the diffusion coefficient D can be calculated from the initial slope $D = E\{d^2\} / 4\Delta t$. Tel 14L moves within a larger radius of constraint than the other telomeres

and does not reach a plateau within the time scale monitored. It is not as mobile as the Tel 6R that has lost its perinuclear association due to the absence of yKu70 protein [24].

D. Radial mean squared displacement (radMSD) analysis was performed by calculating for each time interval the square average of radial distances $\Delta d^2 = (d(t) - d(t + \Delta t))^2$, e.g. $\Delta t = 1.5$s: radMSD = mean $((d_2-d_1)^2, (d_3-d_2)^2,\ldots)$; $\Delta t = 3$s: radMSD = mean$((d_3-d_1)^2, (d_4-d_2)^2,\ldots)$. For time intervals 1.5s–150s, mean square of all possible Δd is calculated and plotted against time interval for the same tagged telomeres Tel 14L, Tel 6R, Tel 8L and Tel 6R in the *Δku70* background. The radMSD shows that Tel14L remains tightly perinuclear despite the large degree of mobility monitored in Figure 10C.

CONCLUSION

We have presented a new algorithm for tracking the movement of a particle in a sequence of noisy images. The key feature that makes the method robust is that the detection task is formulated as a global optimization problem. The optimal solution is computed most efficiently by dynamic programming. The algorithm has been successfully applied to the analysis of the movement of chromosomal loci within the nucleus of a yeast cell. The results obtained are highly satisfactory, suggesting that the dynamic programming approach has good potential for similar biological imaging problems. While the particle trajectories are usually extracted completely automatically, the software can accept hints or corrections provided by the biology expert. Its response time is sufficiently fast for it to operate in a semi-interactive mode with the priority given to the user input. In its present configuration, the system can track a spot over a sequence of 300-500 images in just a few seconds.

We believe that this software should be quite useful in practice, as it facilitates the extraction of quantitative data for complete and reproducible trajectory with minimal input from the biologist. In our case, we have been able to demonstrate differences in dynamics among telomeres, which reflects differences in the type of interaction with the nuclear envelope. These are only a few of many potential conclusions to be drawn from the application of a robust spot tracking algorithm to the question of chromatin dynamics.

REFERENCES

[1] P. van Roessel and A. H. Brand, "Imaging into the future: visualizing gene expression and protein interactions with fluorescent proteins," *Nature Cell Biology*, vol. 4, pp. E15-E20, 2002.

[2] S. G. Megason and S. E. Fraser, "Digitizing life at the level of the cell: high-performance laser-scanning microscopy and image analysis for in toto imaging of development," *Mechanisms of Development*, vol. 120, pp. 1407-1420, 2003.

[3] A. Belmont, "Visualizing chromosome dynamics with GFP," *Trends in Cell Biology*, vol. 11, pp. 250-257, 2001.

[4] M. K. Cheezum, W. F. Walker, and W. H. Guilford, "Quantitative comparison of algorithms for tracking single fluorescent particles," *Biophysical Journal*, vol. 81, pp. 2378-2388, 2001.

[5] C. J. Veenman, M. J. T. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 23, pp. 54-72, 2001.

[6] A. Genovesio, B. Zhang, and J.-C. Olivo-Marin, "Tracking of multiple fluorescent biological objects in three dimensional video microscopy," presented at Proceedings of the 2003 IEEE International Conference on Image Processing (ICIP'03), Barcelona, Spain, 2003.

[7] G. Rabut and J. Ellenberg, "Automatic real-time three-dimensional cell tracking fluoresence microscopy," *Journal of Microscopy*, vol. 216, pp. 131-137, 2004.

[8] D. Thomann, D. R. Rines, P. K. Sorger, and G. Danuser, "Automatic fluorescent tag detection in 3D with super-resolution: Application to the analysis of chromosome movement," *Journal of Microscopy*, vol. 208, pp. 49-64, 2002.

[9] D. Thomann, D. R. Rines, P. K. Sorger, and G. Danuser, "Automatic fluorescent tag localization II: Improvement in super-resolution by relative tracking," *Journal of Microscopy*, vol. 211, pp. 230-248, 2003.

[10] D. Li, J. Xiong, A. Qu, and T. Xu, "Three-Dimensional Tracking of Single Secretory Granules in Live PC12 Cells," *Biophysical Journal*, vol. 87, pp. 1991-2001, 2004.

[11] Improvision, Volocity, http://www.improvision.com/, Lexington, MA, USA, [Online] Available: http://www.improvision.com/ (visited in 2004).

[12] Universal Imaging Corporation, MetaMorph Imaging System, http://www.image1.com/, [Online] Available: http://www.image1.com/ (visited in 2004).

[13] M. Unser, G. Pelle, P. Brun, and M. Eden, "Automated extraction of serial myocardial borders from M-mode echocardiograms," *IEEE Trans. Med. Imag.*, vol. 8, pp. 96-103, 1989.

[14] S. M. Gasser, "Positions of potential: nuclear organization and gene expression," *Cell*, vol. 104, pp. 639–642, 2001.

[15] P. Heun, T. Laroche, K. Shimada, P. Furrer, and S. M. Gasser, "Chromosome dynamics in the yeast interphase nucleus," *Science*, vol. 294, pp. 2181-2186, 2001.

[16] P. Thévenaz, U. E. Ruttimann, and M. Unser, "A pyramid approach to subpixel registration based on intensity," *IEEE Trans. Image Processing*, vol. 7, pp. 27-41, 1998.

[17] B. Pesquet-Popescu and J. L. Vehel, "Schotastic fractal models for image processing," *IEEE Signal Processing Magazine*, vol. 19, pp. 48-62, 2002.

[18] A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 6, pp. 661-674, 1984.

[19] G. D. Forney, "Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Information Theory*, vol. 18, pp. 363-378, 1972.

[20] A. Huertas and G. Medioni, "Detection of intensity changes with subpixel accuracy using Laplacian-Gaussian masks," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 8, pp. 651-664, 1986.

[21] W. Rasban, ImageJ, http://rsb.info.nih.gov/ij/, NIH, Bethesda, MD, USA, [Online] Available: http://rsb.info.nih.gov/ij/ (visited in 2004).

[22] M. D. Abràmoff, P. J. Paulo Magalhaes, and S. J. Ram, "Image Processing with ImageJ," *Biophotonics International*, vol. 11, pp. 36-42, 2004.

[23] A. F. Straight, A. S. Belmont, C.C.Robinett, and A. W. Murray, "GFP tagging of budding yeast chromosomes reveals that protein-protein interactions can mediate sister chromatid cohesion," *Current Biology*, vol. 6, pp. 1599-1608, 1996.

[24] F. Hediger, F. R. Neumann, G. V. Houwe, K. Dubrana, and S. M. Gasser, "Live imaging of telomeres: yKu and sir proteins define redundant telomere-Anchoring pathways in yeast," *Current Biology*, vol. 12, pp. 2076-2089, 2002.

[25] F. Hediger, A. Taddei, F. R. Neumann, and S. M. Gasser, "Methods for visualizing chromatin dynamics in living yeast," *Methods in Enzymology*, vol. 375, pp. 345-365, 2004.

[26] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 21, pp. 476-480, 1999.

[27] W. Marshall, F,, A. Straight, J. F. Marko, J. Swedlow, A. Dernburg, A. Belmont, A. W. Murray, D. A. Agard, and J. W. Sedat, "Interphase chromosomes undergo constrained diffusional motion in living cells," *Current Biology*, vol. 7, pp. 930-939, 1997.

[28] J. Vazquez, A. S. Belmont, and J. W. Sedat, "Multiple regimes of constrained chromosome motion are regulated during interphase in Drosophila," *Current Biology*, vol. 11, pp. 1227-1239, 2001.

[29] J. R. Chubb, S. Boyle, P. Perry, and W. A. Bickmore, "Chromatin motion is constrained by association with nuclear compartments in human cells," *Current Biology*, vol. 12, pp. 439-445, 2002.