ELSEVIER

Surface Science 359 (1996) 291–305

surface science

Background correction in scanning probe microscope recordings of macromolecules

J.P. Pascual Starink [a], Thomas M. Jovin [b,*]

[a] *AmCell Corporation, 1190 Bordeaux Drive, Sunnyvale, CA 94089, USA*

[b] *Max Planck Institute for Biophysical Chemistry, Department of Molecular Biology, PO Box 2841, D-37018 Göttingen, Germany*

Received 16 June 1995; accepted for publication 25 January 1996

Abstract

In scanning probe microscopy of macromolecules, the specimen to be scanned is deposited on a flat surface such as mica or glass. Unfortunately, there is no guarantee that the orientation of this surface is normal to the scanning directions, or that it is atomically flat. Phenomena such as capillary forces and noise can cause the surface to appear curved in the recording. Compensation for these background components are necessary before meaningful, quantitative measurements can be performed on the images. We describe a procedure to correct for both the shape of the recorded surface and the influence of noise on the background. The surface is modeled with a low-order polynomial, and the noise by a constant line off-set for each scan line. These parameters are initially estimated using all recorded samples, and the result is subtracted from the recording. Then, in an iterative loop, the background is refined by first segmenting the image in back- and foreground pixels, followed by re-estimating the parameters using only the background pixels. For recordings of macromolecules, this procedure performs up to 70% better than the flattening procedure included in the microscope software, in terms of standard deviation of the background pixels as depicted by the final segmentation.

Keywords: Amorphous surfaces; Atomic force microscopy (AFM); Computer simulations; Glass surfaces; Physical adsorption; Scanning force microscopy (SFM); Scanning tunneling microscopy (STM); Surface structure, morphology, roughness, and topography;

1. Introduction

Scanning probe microscopes (SPM) have become an important tool for the investigation of macromolecules in the Å–nm range [1,2]. The molecules are deposited on a flat surface such as glass or mica and scanned by the instrument, thereby revealing topographic details of their structure with high spatial resolution.

Since the inception of SPM, it has been recognized that the recordings suffer from perturbations that limit resolution. These include blurring of details due to the tip geometry, noise corruption, geometric distortions caused by the nonlinear behavior of the piezo actuators, and occasional deterministic distortions such as external vibrations, thermal drift, piezo creep, "double-tip" contact, and perturbations by mobile adsorbates. The major contributions are from the noise and the finite size of the scanning tip. These topics have been addressed extensively. At first, linear noise filters were devised to reduce the noise in STM recording [3–7]. Later, non-linear filters were proposed, particularly for recordings of large

* Corresponding author. Fax: +49 551 2011467; e-mail: tjovin@mpc186.mpibpc.gwdg.de.

PII S0039-6028(96)00367-6

or rough areas [8–10]. The effects of finite tip size can be regarded in terms of a linear operation when scanning very small areas [11]. However, for large or rough areas this operation is also non-linear [12].

In order to reconstruct a recording objectively and accurately, a model for the tip shape, (or a determination thereof) must be available. Attempts to characterize the tip shape have included scanning one tip with another tip [13], scanning known, incompressible objects such as colloidal gold particles [14] or polystyrene latex microspheres [15], characterization by adhesion measurements [16], imaging by scanning and electron microscopy [17,18], and direct imaging using sharp structures [19]. Reconstruction procedures employing the tip shape have been suggested by various authors [12,14,20,21]. Recently, it was demonstrated that the scanning process can be viewed as a gray-value morphological dilation, and that reconstruction can be achieved by its inverse operation, i.e. erosion [10,22–24]. It should be noted that reconstruction is necessarily incomplete; the area under the tip between multiple contact points can never be recovered.

One important source of perturbation in SPM recordings to which little attention has been directed is the background [25,26]. Usually, the software provided with commercial microscopes includes some sort of flattening or background subtraction, and this generally suffices for most applications. However, for structural measurements on recordings of shallow objects like macromolecules, one needs to compensate accurately for the background.

In this paper, we model the background in SPM images by two components. The first is the backplane, i.e. the representation of the surface supporting the specimen. This surface is likely to be tilted with respect to the orthogonal scanning directions. Furthermore, the surface will almost certainly appear curved in the scanned image, even if it is physically flat. The apparent curvature is caused by superposition of physical phenomena such as van der Waals forces, elastic deformation of the supporting cantilever and sample, and capillary forces [27]. It is assumed that a polynomial fit of some appropriate order through the data is sufficient to account for this effect.

The second background component is observed in the slow scan direction as stripes that are higher or lower in intensity than surrounding regions, but without an apparent correlation with the imaged structure. These offsets are caused by a type of noise characteristic for scanning probe microscopes, known as $1/f$ or "flicker noise" [7,28,29]. This noise has a power spectrum given by

$$P(f) = \frac{A}{f^{\beta}}, \tag{1}$$

where f is frequency, A is the amplitude, and β determines the spectral distribution and thus the appearance of the noise [30]. For example, for $\beta = 0$ the noise is white, whereas $\beta = 2$ corresponds to Brownian motion. Stoll and Baratoff [4] claimed that $1/f$ noise in SPM is due to random movements of the tip perpendicular to the scan direction, and to the electronics of the feedback system. Measurements of the noise in the different parts of the STM have revealed that tunnel current noise also contributes to the $1/f$ component [29].

It is clear that the line offsets cannot be removed by applying a global backplane correction and, conversely, that the accuracy of the estimated backplane is diminished by these offsets. In this paper we suggest a procedure for estimating the backplane and the offsets in SPM recordings of shallow objects such as macromolecules deposited on a surface assumed to be flat.

2. Method

2.1. Background estimation

As discussed in Section 1 the background in SPM recordings is modeled as two independent, additive components – the global backplane, and stripes in the slow scan direction which are uniformly shifted in height (gray-value intensity). Subtle interactions between sample and specimen during scanning result in a curved backplane. Furthermore, the surface may not be physically flat, and its orientation may not be normal to the

scan directions. The backplane is modeled with a polynomial of power n in the slow scan direction x, and m in the fast scan direction y

$$f_{ij} = \sum_{l=0}^{m} \sum_{k=0}^{n} a_{kl} x_i^k y_j^l, \tag{2}$$

where a_{kl} are the polynomial coefficients, and f_{ij} is the image value at position (i,j). Note that we use a right-handed coordinate system, where the upper left pixel is at (1,1). The image size is N (rows) × M (columns). To consider the noise, an offset c_i is added to each scan line i. As a consequence, the term $a_{00}x^0y^0$ must be explicitly omitted from the model. As it is incorporated in the offsets, it is redundant. The general expression for the background becomes:

$$f_{ij} = \sum_{k=1}^{n} a_{k0} x_i^k y_j^0 + \sum_{l=1}^{m} \sum_{k=0}^{m} a_{kl} x_i^k y_j^l + c_i. \tag{3}$$

This model is linear in its parameters, and is solved by least-squares error minimization. For that purpose, the equations for f_{ij} are collected in the design matrix A of the problem (see Appendix). The coefficients a_{ij} are stored in vector $\boldsymbol{a} = [a_{10} \ldots a_{n0} a_{01} \ldots a_{nm} c_1 \ldots c_N]^t$, where each c_i is repeated M times. The gray value intensities f_{ij} are put in vector $\boldsymbol{b} = [f_{11} f_{12} \ldots f_{1M} f_{21} \ldots f_{NM}]^t$. In matrix notation, the problem is to determine $\boldsymbol{a}$ such that $|A \cdot \boldsymbol{a} - \boldsymbol{b}|^2$ is minimized. The normal equations of the least-squares problem are written as

$$A^t \cdot A \cdot \boldsymbol{a} = A^t \cdot \boldsymbol{b}. \tag{4}$$

This system is studied in more detail in the Appendix. The model for the backplane has $(n+1) \times (m+1)$ parameters. Extension of this model with the line offsets augments the number of parameters by the number of scan lines N in the recording. As an example, a second-order plane is described by nine parameters, and a typical recording from a NanoScope III SPM contains 512 lines, leading to 520 parameters to be fit. The additional parameters often cause the system of normal equations to be (very close to) singular. That is, different sets of parameter combinations would then fit the data equally well, such that matrix $A^T \cdot A$ is ill-conditioned. A solution for dealing with this problem is to apply singular value decomposition (SVD) to solve the system $A \cdot \boldsymbol{a} = \boldsymbol{b}$. This, however, is hard to execute since the number of rows in the matrix equals the number of samples in the recording, while the number of columns is $(n+1) \times (m+1) - 1 + N$. For a second-order backplane and a 512 × 512 recording, the size of the design matrix is $512^2 \times 520$. Alternatively, one can apply SVD to solve Eq. (4). Any $N \times M$ matrix R, $N \geq M$, can be decomposed into $U \cdot W \cdot V^t$, where U is an $N \times M$ column-orthogonal matrix, W is an $M \times M$ diagonal matrix, and V^t is the transpose of an $M \times M$ orthogonal matrix. Since $A^t \cdot A$ is square, the solution to the normal equations in terms of SVD can be written as

$$\boldsymbol{a} = V \cdot W^{-1} \cdot U^t \cdot \boldsymbol{b}. \tag{5}$$

The inversion of W is achieved by replacing the diagonal elements w_i by their reciprocals. Elements that are very close to zero (i.e. approaching the computer floating-point precision) indicate singularity, and their reciprocals are set exactly to zero so as to obtain the solution vector of smallest length. The number of w_i's close to zero (smaller than the square root of the machine precision) denotes the nullity of the system, thus diagnosing the severity of the singularity.

Even if we choose to solve Eq. (4) by SVD, we have to deal with a system of $(n+1) \times (m+1) - 1 + N$ equations. Solving directly is still very time-consuming and rather inefficient, since the major part of the matrix, namely the lower right $N \times N$ submatrix, is diagonal (see Appendix). This feature can be exploited for the efficient inversion of matrix $A^t \cdot A$ by partitioning into four submatrices:

$$A^t \cdot A = \begin{bmatrix} P & Q \\ R & S \end{bmatrix}, \tag{6}$$

where P and S are square matrices of size $(n+1) \times (m+1) - 1$ and M, respectively. The partitioned representation of the inverse is similar:

$$(A^t \cdot A)^{-1} = \begin{bmatrix} \tilde{P} & \tilde{Q} \\ \tilde{R} & \tilde{S} \end{bmatrix}, \tag{7}$$

where $\tilde{P}$, $\tilde{Q}$, $\tilde{R}$, and $\tilde{S}$ have the same size as P, Q,

R, and S, respectively, and are given by

$$\tilde{P} = (P - QS^{-1}R)^{-1},$$

$$\tilde{Q} = -(P - QS^{-1}R)^{-1}(QS^{-1}),$$

$$\tilde{R} = -(S^{-1}R)(P - QS^{-1}R)^{-1},$$

$$\tilde{S} = S^{-1} + (S^{-1}R)(P - QS^{-1}R)^{-1}(QS^{-1}). \quad (8)$$

These equations are rapidly solved since the largest inversion required involves $\tilde{S}$, which is an $N \times N$ diagonal matrix. Inversion of $(P - QS^{-1}R)^{-1}$ is accomplished by SVD, as described before. Simplified expressions are used in the evaluation of Eq. (8). The initial correction now consists of subtracting the resulting background from the original recording.

2.2. Background refinement

Although the initial correction flattens the surface reasonably well, the estimation procedure is affected by the presence of object pixels. In order to improve the estimation, the procedure should be restricted to the background pixels. The procedure to segment the pixels into fore- and background should be conservative, i.e. the foreground should not only include true object pixels, but also pixels that are in questionable regions, or "dirt" spots. We describe a conservative segmentation procedure suited to our recordings of macromolecules. Obviously, recordings of different objects may require another segmentation strategy.

A median filter is first applied to smooth the data and reduce the noise. For our applications, the size of the filter was chosen to cover approximately 8 × 8 nm. This size depends on the sampling resolution, which varied in our studies from 0.5 (15 × 15 mask) to 2 (5 × 5 mask) nm per pixel. Classification of the pixels is performed on a line-by-line basis using the isodata thresholding method. According to this method, the gray value histogram of the line is divided into two parts by a threshold value, initially the mean value of the line pixel values. This threshold is repeatedly replaced by the average of the mean gray values of the two histogram parts, until the threshold value remains constant. Next, the mean μ_i and the standard deviation σ_i of the background pixels of line i are determined. After processing all lines, the global standard deviation σ is calculated from the individual σ_is. The final segmentation of line i is obtained by classifying those pixels with values smaller than $\mu_i + \sigma$ as background.

Next, the background is re-estimated using only the background pixels. If required, the order of the polynomial can be set higher since the foreground pixels are excluded. One approach is to fit Eq. (3) to the background pixels only. Since in this approach the sums do not run continuously over all the coordinates, Eq. (4) is constructed by updating the matrix and vector elements in a loop over all background pixels. This is described in more detail in the Appendix.

Another strategy to update the background is to perform fitting on a line-by-line basis. This approach may be justified since the scanning process in the microscope is actually one-dimensional. Since the difference in collection time between two adjacent samples on neighboring scan lines is twice the time required to scan a line, the image data between two adjacent scan lines are fairly independent. In this approach, the background pixels of the line are collected, and a simple polynomial of order m is fitted to these data (see Appendix).

Subtracting the refinement from the original recording results in a flatter background than after the initial correction. Consequently, the segmentation of this image will be better, and the refinement procedure can be applied again using the new segmentation. The scheme to correct for the background components is outlined in Fig. 1. The stop criterion is defined as follows: let ξ_i be the standard deviation of the background pixels after the ith iteration, as masked by the segmentation. When the change in ξ between two successive iterations is less than a certain threshold (e.g. 1%), the iteration is stopped. For our recordings of macromolecules, we usually need about three iterations to reach convergence.

3. Discussion

In this section, we discuss our methods and approaches. The validity of the assumption that noise can be modeled by constant line offsets is

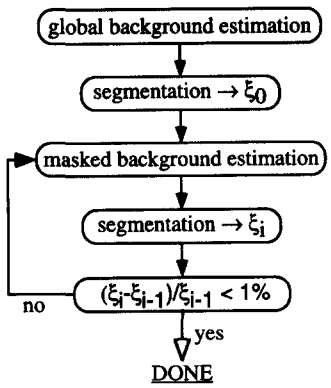

Fig. 1. General steps in the background estimation procedure. The iteration stops when the change in the standard deviation ξ of the image background pixels between the two most recent iterations is less than 1%.

investigated. The stability and time requirements of the methods of solving the normal equations are compared. We define ξ to be the standard deviation of the background pixels, as masked by the (final) segmentation. Background correction by the microscope software will be referred to as flattening.

The experiments were implemented in standard C and linked as a library to the commercial image processing package SCIL-Image version 1.3 (TNO Institute of Applied Physics, Delft, The Netherlands) on a Silicon Graphics INDY workstation.

3.1. Validation of noise model

The nature of $1/f$ noise is such that the vertical displacements of the stripes become more pronounced for larger β, while at the same time the correlation between neighboring noise samples increases. These features lend support to the assumption that these stripes can be modeled by constant line offsets. We expect that compensation of the noise by constant offsets will be increasingly effective for higher β. Several approaches to characterize the noise of the tunneling current in STM have been reported. Tiedje et al. [31] found a perfect $\beta = 1.0$ fit to scans of graphite with tungsten, platinum, and platinum–iridium tips. From measurements in air of graphite with a tungsten tip, Abraham et al. [32] found $\beta = 0.94$. Möller et al. [29] concluded from a search for the cause of $1/f$ noise that the tunneling phenomenon is mainly responsible, and found $\beta = 1.08$. From visual comparison of stationary tip recordings with simulated $1/f$ noise images, Stoll and Marti [28] arrived at $\beta = 1.4 \pm 0.2$. Moreover, Aguilar et al. [7] experimentally found $\beta = 1.17$ and from an average over 30 recordings, the estimated $\beta = 1.0 \pm 0.1$ also demonstrated a dramatic dependency on the acquisition mode and on the STM head used. It is important to stress the relevance of the fast scan velocity with respect to the slow scan direction.

To determine the noise parameters of our NanoScope III microscope configuration under regular conditions, we investigated stationary tip recordings [10]. All average line power spectra of the recordings showed an offset from the baseline, indicating an additional white noise component. Eq. (1) was therefore augmented with constant C to obtain

$$P(f) = \frac{A}{f^{\beta}} + C. \tag{9}$$

Fitting this model to the power spectra led to an error typically ~ 300 times smaller than that achieved with Eq. (1). From over 30 recordings, we found a mean value of $\beta = 1.99 \pm 0.21$. The next experiment shows that at this value, the power in the slow scan direction is ~ 200 times higher than in the fast scan direction.

Generation of $1/f$ noise with a certain β and (spectral) power A was performed according to a method we have reported elsewhere [10]. A one-dimensional, complex signal v containing twice the number of samples in the recording is filled with a random phase, and multiplied with an $f^{-\beta/2}$ amplitude function. This signal is transformed back to the time domain and added to the temporal signal f containing the (test) image

$$f(i,j) \leftarrow f(i,j) + v(2M + i), \tag{10}$$

where M is the number of samples per line, i and j the spatial position, and the factor 2 accounts for the trace and retrace paths of the tip, one of which is not recorded. This approach to noise simulation closely represents the actual scanning process

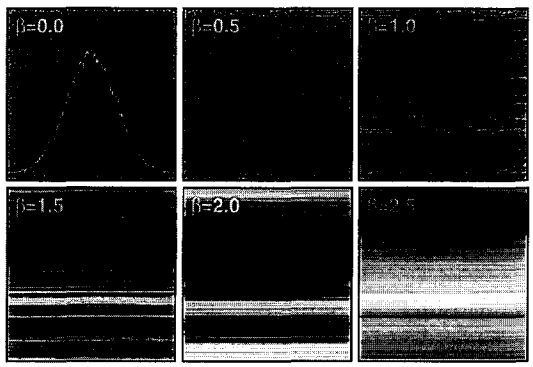

Fig. 2. Generated $1/f$ noise images for β ranging from 0 to 2.5. For $\beta = 0$, the noise is white, as shown by the overlaid gray-value histogram. As β increases, the stripes become more distinct.

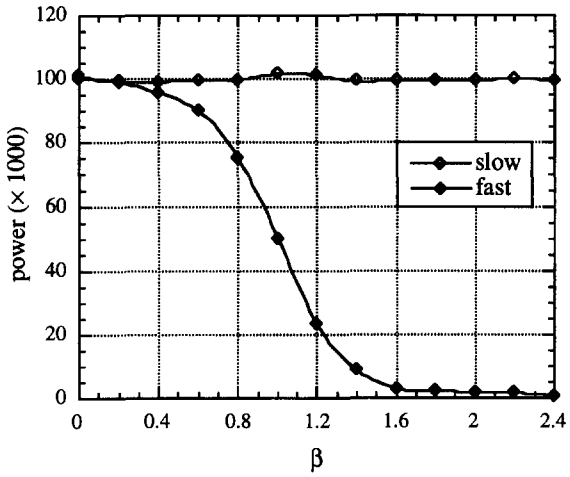

Fig. 3. Average line power in the fast and slow scan directions, measured in generated noise images of size 512 × 512 pixels with β running from 0 to 2.4. Each graph point represents the mean of 25 images, all scaled to a global power of 10^5.

which, apart from the tip–sample interaction, is in fact one-dimensional.

The six generated noise images in Fig. 2 show that the stripes become more distinct as β increases. The power of the signal is concentrated in the lower frequencies, leading to a higher correlation between neighboring noise samples. This suggests that the power of the noise is dominated mainly by the differences between these offsets, and not by correlations between neighboring noise samples in the fast scan direction. To verify this interpretation, the following experiment was performed. For β varying from 0 to 2.4 with a step size of 0.2, 50 noise images were generated with a global power of 10^5. The power was determined in the fast and slow scan directions, and the respective average values are plotted in Fig. 3. As expected, the power in the slow scan direction, perpendicular to the height changes of the stripes, was about 10^5 for all βs, whereas the power in the fast scan direction decreased as β increased.

Thus, we can assume that the noise remains constant for high values of β, but that for low βs the noise exhibits a more random character. To examine the validity of this assumption, a random second-order backplane and a $1/f$ noise image were generated and added. The background was estimated and subtracted from the image. Since the image contained only a background, the resulting signal should have been zero. However it was not, suggesting that the noise in the fast scan direction cannot be resolved by such a low-order backplane, or by the offsets. Therefore, as a measure of performance, we adopted the ratio of the power in the fast scan direction of the noise image to the global power of the result. This amplifies errors in the estimated offsets, while suppressing the offsets in the original noise image. The analysis was performed for randomly generated, second-order backplanes, and for β ranging from 0.5 to 2.5. The noise was scaled such that the signal-to-noise ratio, defined as the variance of the backplane divided by the variance of the noise, was 5, 10, and 15, respectively. The performance at each parameter combination was averaged for five noise images.

The results in Table 1 demonstrate that performance was maintained as β decreased. This could have been caused by the fact that for low βs, the samples were independently arranged around the background line. For large βs, the correlations between neighboring samples were higher, but the occasional high frequencies caused vertical jumps that could not be compensated for by the offsets.

3.2. Solving the normal equations

After adding the line offsets to the model for the backplane, the normal equations are often close to singular. They can usually be solved directly

Table 1
Performance of the background estimation procedure as a function of β for different signal-to-noise ratios (SNR); a second order background with $1/f$ noise was generated and processed; the performance was determined by dividing the global power of the corrected image by the average line power of the noise image; each value represents the average of five experiments

SNR	β				
	0.50	1.00	1.50	2.00	2.50
5	0.99 ± 0.11	1.10 ± 0.32	0.8 ± 0.34	1.2 ± 0.35	0.92 ± 0.48
10	0.93 ± 0.093	0.96 ± 0.16	1.1 ± 0.59	0.77 ± 0.38	1.30 ± 0.42
15	0.94 ± 0.13	0.85 ± 0.14	1.3 ± 0.53	0.97 ± 0.27	0.78 ± 0.36

by Gauss–Jordan elimination or singular value decomposition (SVD). The results, however, may not be entirely what one expects due to the (close to) singular state. This can be seen when the backplane and the offsets are inspected separately. For example, consider an SPM recording of a DNA strand (Fig. 4a). The results of fitting a second-order backplane and the offsets are shown in Figs. 4b and 4c (Gauss–Jordan), 4d and 4e (SVD), and 4f and 4g (partitioning). The results show clear differences for the different background and offset estimates.

Table 2 lists the gray-value ranges, the standard deviations ξ of the original image and the estimated background components. Although they were equal for all approaches ($\xi = 109$, Fig. 4h), only the partitioning result remained within the gray-value range of the original image. Results of the two other methods exceeded this range considerably, indicating instability. Thus, solving the normal equations by partitioning is preferred to the other methods in terms of time requirements and numerical stability.

The time requirements of the three different background estimation approaches were determined for a 512×512 recording for backplane orders 2, 3, and 4 (Table 3). As expected, inversion by partitioning required less time than a solution by Gauss–Jordan elimination or full SVD. The initial correction required less time than the masked refinement, mainly because in the latter the matrix $A^t \cdot A$ needed to be set up iteratively, without recourse to the convenient assumption that all rows and columns contain the same coordi-

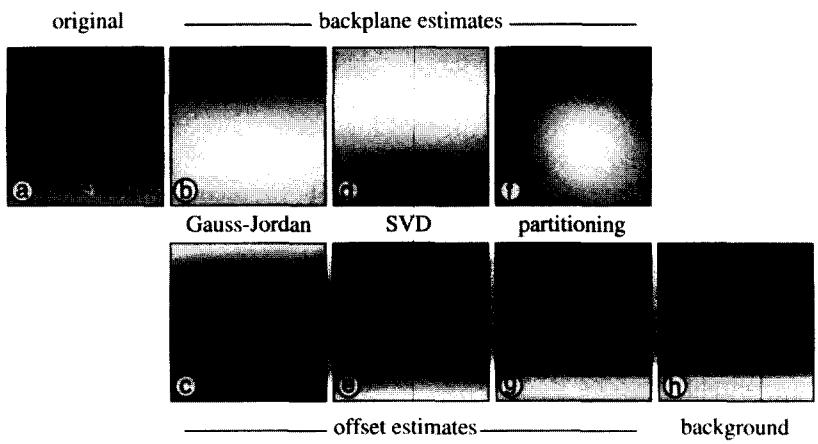

Fig. 4. (a) Estimating the background components of a scanning force microscopy (SFM) recording of plasmid DNA. (b, c) The backplane and offsets estimated by Gauss–Jordan elimination, (d, e) full singular value decomposition, and (f, g) inversion by partitioning all result (h) in the same background. However, only the components resulting from partitioning remain within the gray-value range of the original recording (see Table 2), indicating the better stability and sensitivity of this approach. All images are displayed with fully stretched contrast.

Table 2
Gray value statistics of the backplane and offsets of recording Fig. 4a, as obtained by three different approaches to solving the normal equations; parameter ξ represents the standard deviation of the background pixels

	Original	Gauss–Jordan		SVD		Partitioning	
		Backplane	Offsets	Backplane	Offsets	Backplane	Offsets
Minimum	0	9	−5061	−4481	−520	−19	91
Maximum	3133	5232	1405	1313	6410	77	2399
ξ	630	1448	1710	1590	2038	25	628

Table 3
CPU time (in s) required by the three matrix solution approaches to estimate the initial global background, and the masked, global and line-by-line refinement, for a second, third, and fourth order backplane

	Gauss–Jordan			SVD			Partitioning		
Order	2	3	4	2	3	4	2	3	4
Initial	53	56	59	136	146	156	2.2	3.1	4.2
Masked	64	92	142	148	176	240	16	40	88
Line-by-line	0.7	1.0	1.4						

nate indices. The mask used in all experiments covered the entire recording. Solving the normal equations in the line-by-line update was performed using Gauss–Jordan elimination, since there was no danger of singularity. The time requirements were therefore very small compared to the global refinement.

An additional remark concerning the solution of the normal equations refers to the actual numbers that are substituted for the coordinates x and y. The choice of step size would appear to be arbitrary, and it seems plausible to choose unity. However, due to the high powers involved, the sums in the matrices become very large with respect to the range of the backplane parameters. This causes round-off errors and reduces the sensitivity of the system. The problem can be overcome by scaling the coordinates, e.g. such that they run from 0 to 1. To generate a unity step background concordant with the original recording, the backplane parameters a_{kl} are multiplied by s^{k+l}, where s is the scaling factor.

3.3. Order of backplane

The polynomial power of the backplane should not be too high, since the object pixels which are part of the foreground are also included in the initial estimation. Higher-order polynomials tend to adapt more to the object, thereby disturbing the procedure, especially when the object covers a large part of the recording. For example, a high resolution recording (512 × 512 pixels at 0.12 nm per pixel lateral resolution) of a DNA strand (Fig. 5a) was corrected using a fourth-order backplane. The procedure clearly tried to correct for the strand (Fig. 5b), leaving it in some sort of channel after correction (Fig. 5c). It is therefore suggested that the backplane is estimated by a low-order polynomial and, if required, the result is updated with a higher-order backplane. In general, a first- or second-order backplane in the initial correction, and a second-order backplane in the

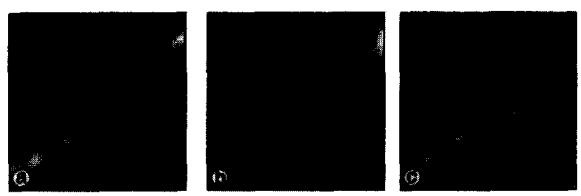

Fig. 5. Fitting a fourth order background to a high resolution SFM recording (0.12 nm per pixel) of DNA (a). The resulting backplane (b) tries to follow the height changes of the strand. In the corrected image (c), the strand appears to lie in a channel.

refinement, seems satisfactory for studies of macromolecules.

3.4. Background refinement

The accuracy of the procedure depends on the ability of the segmentation method to discriminate between background and foreground pixels. The method described here is based on a line-by-line segmentation using the isodata threshold method. When sparsely distributed macromolecules are studied, the background is dominant and the objects are shallow with relatively steep edges. In these studies, isodata thresholding provides the desired result. However, if the objects cover a large part of the image, or if a large gray-value slope between the background and the object is present, the threshold value will be set somewhere in the middle of the gray-value range, resulting in too many background pixels. Other segmentation methods should then be employed to obtain a conservative estimate of the set of background pixels.

Two methods were proposed for updating the background using the segmentation result as a mask: a global approach and a line-by-line approach. The line-by-line update may be justified by the fact that scanning is in fact a one-dimensional process, and that the correlation between scan lines is smaller than between samples in a scan line. Practical problems arise when a line contains no background pixels. This is clear in the line-by-line approach. In the absence of any data, there is nothing to fit. In the global approach, the corresponding element in the diagonal matrix S (Eq. (6)) will be zero, and consequently its reciprocal in S^{-1} does not exist. The SVD will yield a result valid only for those lines containing background pixels.

Another drawback of the line-by-line approach is that all lines are treated as if they are completely independent. This assumption may lead to capricious structures in the slow scan direction, which may be improbable or even physically unrealizable. Consider the high resolution (0.12 nm per pixel) recording of plasmid DNA in Fig. 6a. Due to the high resolution, the edges of the strand are not steep. As a consequence, the segmentation procedure erroneously assigns object pixels to the background (Fig. 6b). The line-by-line update (Figs. 6c and 6d) suffers more from these errors than the global update (Figs. 6e and 6f), for which they are averaged out over the whole image. A solution to this problem may be to update the background of a scan line using a small number of neighboring scan lines. In this case, the local correlation between scan lines is preserved, while the obvious lack of correlation between distant ones is ignored.

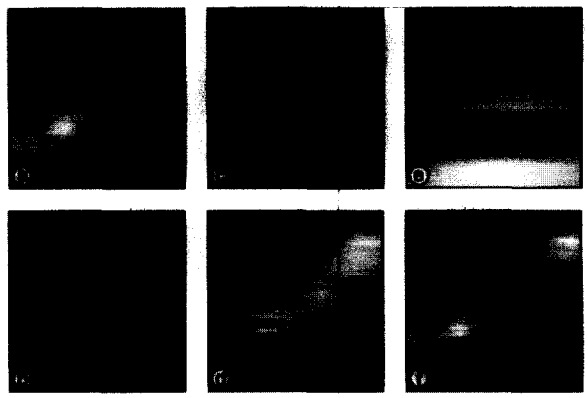

Fig. 6. Refinement of the background of a high resolution SFM recording (0.12 nm per pixel) of DNA (a). The segmentation (b) is used to update the background in the line-by-line approach (c) and the global approach (e). The results show that the line-by-line approach (d) contains improbable capricious features, whereas the global approach (f) does not.

4. Experimental results

The background correction method was applied to SFM recordings of different samples, and was compared with the outcomes of the microscope flattening procedure. The samples were scanned with a NanoScope III SPM (Digital Instruments (DI), Santa Barbara, CA) in contact mode, and were collected as height images of 512×512 samples. The raw recordings were flattened with order two. The background was initially estimated with order two, and iteratively updated using the global approach, also with order two. Iteration was stopped when the change in ξ between two successive steps was less than 1%.

In Fig. 7, each row shows the results of an

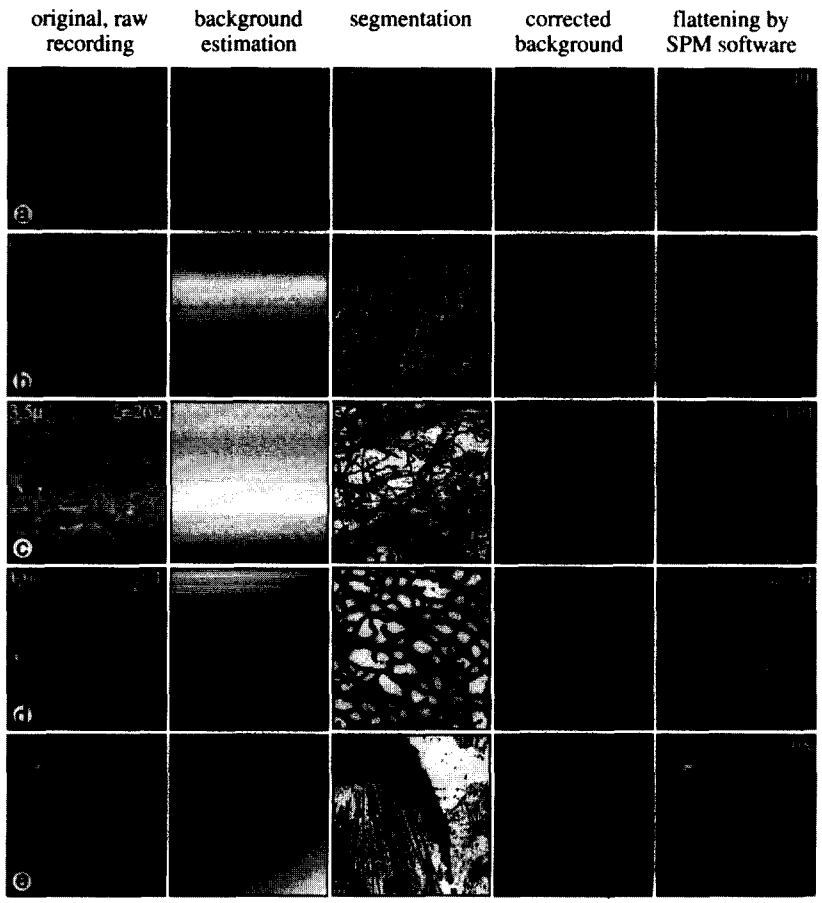

Fig. 7. Processing of SFM recordings of plasmid DNA (a, b), DNA networks (c, d), and rat cells (e). The rows show the various results in the process. Printed on the raw recordings are the scan sizes; printed on all results is the standard deviation ξ of the background pixels masked by the segmentation.

experiment. Fig. 7 (left to right) shows the original raw recording, our background and segmentation estimates, the corrected image and the flattening result. The scan sizes are printed on the raw recordings, and on the raw image and the corrected images the standard deviations ξ of the background pixels are shown.

The rows a and b of Fig. 7 show the processing of recordings of relaxed dimeric circular DNA at two different resolutions. Details of the experimental conditions are given in Ref. [33]. Although flattening did a reasonably good job, it was perceptibly distorted in regions where the strand lies parallel to the fast scan direction. Especially in the low resolution recording (Fig. 7, row a) this led to smears of lowered height in the attempt to compensate for the object. Leaving out the strand from the procedure in our approach eliminated this effect.

Rows c and d of Fig. 7 show recordings of kinetoplast DNA, the mitochondrial DNA of the trypanosomatid *Crithidia luciliae* (see Ref. [34] for experimental details). The DNA consists of several thousands of duplex mini-circles and a small number of maxi-circles concatenated into a network structure [35]. The networks were dialyzed against distilled water, spread directly on glowed-discharge activated mica coated with 1 mM spermine, and air dried. Fig. 7c depicts a network after extensive proteolysis, resulting in extended networks with relaxed DNA interlocked circles and some broken DNA strands. As the recording

shows, the spermine film contains small holes, which potentially interfere with the quantification of the molecules and structures. Since our procedure attempts to flatten the background to zero height, clipping all resulting negative values to zero removed the tiny holes. The recording in row d of Fig. 7 exhibits a backplane that is very curved in the slow scan direction. The microscope software did not yield a satisfactory result; ξ even increased by 8%. Thanks to the correct segmentation, our method was able to reduce ξ by 31%.

Row e of Fig. 7 shows subcellular structures of rat mammary carcinoma cells, such as the nucleus and microvilli extending from the periphery of the cells. Experimental details are reported in Ref. [36]. The nucleus is very high ($\sim 1.2\ \mu$m) compared to the microvilli (~ 100 nm). These large differences in height and capillary forces caused the (flat) glass substrate to appear curved. The microscope flattening procedure was perturbed by the extreme height differences, which resulted in an erroneous flattening. Our procedure correctly discriminated between the glass surface and the objects, determined the right curvature of the imaged surface (backplane), and consequently yielded a nicely flattened background.

Table 4 lists the background standard deviations ξ of the raw recordings, the flattening result, and the corrections after applying our initial correction and after each iteration. Since the flattening procedure is a global approach, it should be comparable with our initial correction. The data show that this was approximately true, but also that our global method performed better in all cases. Furthermore, the performance was drastically improved by the iterative refinement, leading to up to 70% smaller ξs compared to the flattening results.

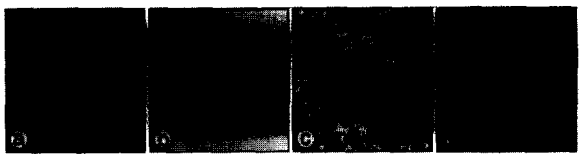

Fig. 8. Background estimation of an already flattened recording. A high resolution SFM recording of a DNA strand (a) shows clear height smearing near scan lines with many object pixels. The second order background (b) and the final segmentation (c) lead to an image without smearing (d).

Finally, applying our method to images already flattened by the microscope software may lead to considerable improvement. Fig. 8a shows an already flattened recording of plasmid DNA. The background estimated with our procedure (Fig. 8b) corrected these mistakes. The resulting image (Fig. 8d) exhibited a considerably flatter surface, as is quantitatively supported by the background standard deviations printed on the images.

Acknowledgements

We thank Dr. Achim Schaper for preparing and scanning the samples of the relaxed plasmid DNA

Table 4
The standard deviations of the background pixels ξ and the percentage change Δ% between two successive estimation steps in the process of estimating the background of the images in Fig. 7

	Image											
	7a		7b		7c		7d		7e		7f	
	ξ	Δ%	ξ	Δ%	ξ	Δ%	ξ	Δ%	ξ	Δ%	ξ	Δ%
Raw	259		305		262		194		138		—	
Global	45.0	82.6	53.2	82.5	121	53.8	182	6.2	113	18.1	107	—
Iteration 1	40.7	9.7	43.4	18.5	92.1	24.1	136	25.6	48.7	57.0	65.1	39.2
Iteration 2	40.6	0.2	42.6	1.7	91.6	0.5	134	1.3	39.7	18.5	63.6	2.4
Iteration 3			42.5	0.4			133	0.2	34.6	12.7	62.9	1.1
Iteration 4									62.8	0.3	33.7	2.6
Iteration 5									33.5	0.6		

and for useful discussions, Lia Pietrasanta for kindly providing the recordings of the rat cell, and Dr. Donna Arndt-Jovin for the preparation and recording of the DNA network structures. This research was sponsored by the European Community Human Capital and Mobility Program (project ERBCHRXCT930177).

Appendix

Practical aspects of implementation

In this Appendix the normal equations are further developed, and we describe our algorithmic approach to solve them. The equations f_{ij} (Eq. (3)) are collected in the design matrix

$$A = \begin{bmatrix} x_1^1 y_1^0 & \dots & x_1^n y_1^0 & \dots & x_1^0 y_1^m & \dots & x_1^n y_1^m & 1 & 0 & \dots & 0 \\ x_1^1 y_2^0 & \dots & x_1^n y_2^0 & \dots & x_1^0 y_2^m & \dots & x_1^n y_2^m & 1 & 0 & \dots & 0 \\ & \vdots & & \vdots & & \vdots & & & & \vdots & \\ x_1^1 y_M^0 & \dots & x_1^n y_M^0 & \dots & x_1^0 y_M^m & \dots & x_1^n y_M^m & 1 & 0 & \dots & 0 \\ x_2^1 y_1^0 & \dots & x_2^n y_1^0 & \dots & x_2^0 y_1^m & \dots & x_2^n y_1^m & 0 & 1 & \dots & 0 \\ & \vdots & & \vdots & & \vdots & & & & \vdots & \\ x_N^1 y_M^0 & \dots & x_N^n y_M^0 & \dots & x_N^0 y_M^m & \dots & x_N^n y_M^m & 0 & 0 & \dots & 1 \end{bmatrix} . \tag{A.1}$$

Let $\tilde{A} = A^t \cdot A$ be the matrix in the normal equations (Eq. (4)). The element $\tilde{A}[k,l]$, and consequently $\tilde{A}[l,k]$, is given by the inner product of the rows k and l of the design matrix A. The matrix $\tilde{A}$ is then expressed in its full form as

$$\tilde{A} = \begin{bmatrix} \sum_i x_i^2 \sum_{j\in B_i} y_j^0 & \dots & \sum_i x_i^{n+1} \sum_{j\in B_i} y_j^0 & \dots & \sum_i x_i^1 \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^{n+1} \sum_{j\in B_i} y_j^m & x_1^1 \sum_{j\in B_i} y_j^0 & \dots & x_N^1 \sum_{j\in B_i} y_j^0 \\ & \vdots & & & & \vdots & & & \vdots & \\ \sum_i x_i^{n+1} \sum_{j\in B_i} y_j^0 & \dots & \sum_i x_i^{2n} \sum_{j\in B_i} y_j^0 & \dots & \sum_i x_i^n \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^{2n} \sum_{j\in B_i} y_j^m & x_1^n \sum_{j\in B_i} y_j^0 & \dots & x_N^n \sum_{j\in B_i} y_j^0 \\ & \vdots & & & & \vdots & & & \vdots & \\ \sum_i x_i^1 \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^n \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^0 \sum_{j\in B_i} y_j^{2m} & \dots & \sum_i x_i^{2n} \sum_{j\in B_i} y_j^{2m} & x_1^0 \sum_{j\in B_i} y_j^m & \dots & x_N^0 \sum_{j\in B_i} y_j^m \\ & \vdots & & & & \vdots & & & \vdots & \\ \sum_i x_i^{n+1} \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^{2n} \sum_{j\in B_i} y_j^m & \dots & \sum_i x_i^n \sum_{j\in B_i} y_j^{2m} & \dots & \sum_i x_i^n \sum_{j\in B_i} y_j^{2m} & x_1^n \sum_{j\in B_i} y_j^m & \dots & x_N^n \sum_{j\in B_i} y_j^m \\ x_1^1 \sum_{j\in B_i} y_j^0 & \dots & x_1^n \sum_{j\in B_i} y_j^0 & \dots & x_1^0 \sum_{j\in B_i} y_j^m & \dots & x_1^n \sum_{j\in B_i} y_j^m & N & \dots & 0 \\ & \vdots & & & & \vdots & & & \ddots & \\ x_N^1 \sum_{j\in B_i} y_j^0 & \dots & x_N^n \sum_{j\in B_i} y_j^0 & \dots & x_N^0 \sum_{j\in B_i} y_j^m & \dots & x_N^n \sum_{j\in B_i} y_j^m & 0 & \dots & N \end{bmatrix} . \tag{A.2}$$

Similarly, the element $\tilde{b}[k]$ of the vector $\tilde{\boldsymbol{b}} = A^t \cdot b$ is calculated as the inner product between column k of

the design matrix A and vector $\tilde{\boldsymbol{b}}$. In its full form, vector $\tilde{\boldsymbol{b}}$ is

$$\tilde{\boldsymbol{b}} = \begin{bmatrix} \sum_i p_i^1 \sum_j p_j^0 f_{ij} \\ \vdots \\ \sum_i p_i^n \sum_j p_j^0 f_{ij} \\ \vdots \\ \sum_i p_i^0 \sum_j p_j^m f_{ij} \\ \vdots \\ \sum_i p_i^n \sum_j p_j^m f_{ij} \\ \sum_j f_{1j} \\ \vdots \\ \sum_j f_{Nj} \end{bmatrix}. \tag{A.3}$$

The sums run over all corresponding coordinates, $i = 1 \ldots N$, $j = 1 \ldots M$. These normal equations may seem complicated to evaluate. We describe an approach to deal with the equations in a simple way.

First, since sampling is usually isotropic in the x and y directions, it is straightforward to replace the coordinates by a common one, p_i. With scaling factor s, this yields $p_i = s \times i$. Next, by dividing up the matrix in the parts P, Q, R and S, as suggested in Section 2.1 (Eq. (6)), the evaluation becomes more structured. The submatrices are treated as separate matrices, so their upper left element is [1,1]. The following definitions are made for any (integer) number i

$$\begin{aligned} i_M &= i_{\mathrm{MOD}}(i+1), \\ i_D &= i_{\mathrm{DIV}}(i+1), \end{aligned} \tag{A.4}$$

where MOD is the integer modulo operator, and DIV is the integer division. With these definitions, the submatrices in Eq. (12) can now be written down in a simplified and structured form in terms of their elements $[k,l]$:

$$\tilde{A} = \begin{pmatrix} P[k,l] = \sum_i p_i^{k_M + l_M} \sum_j p_j^{k_D + l_D} & Q[k,l] = p_l^{k_M} \sum_j p_j^{k_D} \\ & \begin{matrix} N & & \\ & \ddots & \\ & & N \end{matrix} \\ R[k,l] = p_k^{l_M} \sum_j p_j^{l_D} & \end{pmatrix}. \tag{A.5}$$

The range of the indices in the submatrices are given by

$$\begin{aligned} &P[k,l], \quad k = 1 \ldots (n+1)(m+1), \quad l = 1 \ldots (n+1)(m+1), \\ &Q[k,l], \quad k = 1 \ldots (n+1)(m+1), \quad l = 1 \ldots N, \\ &R[k,l], \quad k = 1 \ldots N, \quad l = 1 \ldots (n+1)(m+1), \\ &S[k,l], \quad k = 1 \ldots N, \quad l = 1 \ldots N. \end{aligned} \tag{A.6}$$

Similarly, vector $\tilde{\boldsymbol{b}}$ is given in terms of its element $[k]$ as

$$\hat{\boldsymbol{b}} = \left(\frac{\sum_i p_i^{k_M} \sum_j p_j^{k_D} f_{ij}}{\sum_j f_{kj}} \right), \tag{A.7}$$

where k in the numerator runs from 1 to $(n+1)(m+1)$, and in the denominator from 1 to N.

Since the summations run over the same coordinates for every position, the sums of the different powers of p_i can be calculated beforehand. The matrix $\tilde{A}$ is then very easy to calculate. Let $S_k = \Sigma_i p_i^k$, then $P[k,l] = S_{k_M+l_M} S_{k_D+k_M}$, $Q[k,l] = p_l^{k_M} S_{k_D}$, and $R[k,l] = p_k^{l_M} S_{l_D}$. Unfortunately, the vector cannot be simplified to the same extent as the matrix can because of the inclusion of the image values f_{ij} in the innermost summations.

In the masked update, we observe a similar situation. The matrix $\tilde{A}$ and the vector $\tilde{\boldsymbol{b}}$ in the normal equations are written in the structural form as

$$\begin{pmatrix} P[k,l] = \sum_i p_i^{k_M+l_M} \sum_{j\in B_i} p_j^{k_D+l_D} & Q[k,l] = p_l^{k_M} \sum_{j\in B_i} p_j^{k_D} & \\ & N_1 & \\ R[k,l] = p_k^{l_M} \sum_{j\in B_i} p_j^{l_D} & & \ddots \\ & & N_N \end{pmatrix} \cdot \boldsymbol{a} = \begin{pmatrix} \dfrac{\sum_i x_i^{k_M} \sum_{j\in B_i} y_j^{k_D} f_{ij}}{\sum_{j\in B_k} f_{kj}} \end{pmatrix}, \tag{A.8}$$

where B is the set of the coordinates of the background pixels, and $B_i \subset B$ contains the coordinates of the background pixels on row i. N_i equals the number of background pixels on row i ($N_i = \| B_i \|$). In this equation, it is not possible to use the sums of the powers of the coordinates, because each row contains (background) pixels at different positions. The elements have to be updated for each background pixel separately. Of course, the powers of p_i are calculated beforehand, and the powers at the positions can easily be determined as a function of their position.

In the line-by-line update approach, we fit the (background) samples in scan line i to a simple polynomial of order m. At point (i,j) the value is then given by

$$f_{ij} = \sum_{l=0}^{m} a_l y_{ij}^l. \tag{A.9}$$

These equations are collected in the design matrix, yielding the following linear system of $m+1$ equations to be solved

$$\begin{bmatrix} \sum_{j\in B_i} y_j^0 & \sum_{j\in B_i} y_j^1 & \cdots & \sum_{j\in B_i} y_j^m \\ \sum_{j\in B_i} y_j^1 & \sum_{j\in B_i} y_j^2 & \cdots & \sum_{j\in B_i} y_j^{m+1} \\ \vdots & \vdots & & \vdots \\ \sum_{j\in B_i} y_j^m & \sum_{j\in B_i} y_j^{m+1} & \cdots & \sum_{j\in B_i} y_j^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sum_{j\in B_i} y_j^0 f_i \\ \sum_{j\in B_i} y_j^1 f_i \\ \vdots \\ \sum_{j\in B_i} y_j^m f_i \end{bmatrix}. \tag{A.10}$$

If there are at least $m+1$ background pixels on scan line i, this system is readily solved using normal Gauss–Jordan elimination, and no special precautions need be taken to avoid singularity.

References

[1] B.L. Blackford, M.H. Jericho and P.J. Mulhern, Scanning Microsc. 5 (1991) 907–918.
[2] J. Yang, L.K. Tamm, A.P. Somlyo and Z. Shao, J. Microsc. 171 (1993) 183–198.
[3] E.P. Stoll, SPIE Opt. Eng. Meas. 599 (1985) 442–450.
[4] E.P. Stoll and A. Baratoff, Ultramicroscopy 25 (1988) 149–154.
[5] M. Pancorbo, E. Anguiano, A. Diaspro and M. Aguilar, Patt. Recogn. Lett. 11 (1990) 553–556.
[6] M. Pancorbo, M. Aguilar, E. Anguiano and A. Diaspro, Surf. Sci. 251/252 (1991) 418–423.
[7] M. Aguilar, E. Anguiano, A. Diaspro and M. Pancorbo, J. Microsc. 165 (1992) 311–324.
[8] A.D. Weisman, E.R. Dougherty, H.A. Mizes and R.J.D. Miller, J. Appl. Phys. 71 (1992) 1565–1578.
[9] A. Diaspro, Eur. Microsc. Anal., September (1994) 21–22.
[10] J.P.P. Starink and T.M. Jovin, J. Appl. Phys. (1995).
[11] J. Tersoff and D.R. Hamann, Phys. Rev. B 31 (1985) 805–813.
[12] D. Keller, Surf. Sci. 253 (1991) 353–364.
[13] V.V. Efremov, P.N. Louskinovich and V.I. Nikishin, Ultramicroscopy 42–44 (1992) 1459–1463.
[14] J. Vesenka, R. Miller and W.E. Henderson, Rev. Sci. Instrum. 65(7) (1994) 2249–2251.
[15] C. Odin, J.P. Aime, Z. Elkakour and T. Bouhacina, Surf. Sci. 317 (1994) 321–340.
[16] T. Thundat, X.Y. Zheng, G.Y. Chen, S.L. Sharp, R.J. Warmack and L.J. Schowalter, Appl. Phys. Lett. 63 (1993) 2150–2152.
[17] D.A. Grigg, P.E. Russell, J.E. Griffith, M.J. Vasile and E.A. Fitzgerald, Ultramicroscopy 42–44 (1992) 1616–1620.
[18] G. Reiss, F. Schneider, J. Vancea and H. Hoffman, Appl. Phys. Lett. 57 (1990) 867–869.
[19] F. Atamny and A. Baiker, Surf. Sci. Lett. 323 (1995) 314–318.
[20] D.J. Keller, Surf. Sci. 294 (1993) 409–419.
[21] P. Markiewicz and M.C. Goh, Langmuir 10 (1994) 5–7.
[22] J.S. Villarrubia, Surf. Sci. 321 (1994) 287–300.
[23] N. Bonnet, S. Dongmo, P. Vautrot and M. Troyon, Microsc. Microanal. Microstruct. 5 (1994) 477–487.
[24] D.L. Wilson, K.S. Kump, S.J. Empell and R.E. Marchant, Langmuir 11 (1995) 265–272.
[25] O. Marti and M. Amrein, STM and AFM in Biology (Academic Press, New York, 1993).
[26] G.S. Pingali and R. Jain, in: IMTC/93, (Irvine, CA, 1993), pp. 327–332.
[27] J.L. Hutter and J. Bechhoefer, Surf. Sci. 73 (1993) 4123–4129.
[28] E.P. Stoll and O. Marti, Surf. Sci. 181 (1987) 222–229.
[29] R. Möller, A. Esslinger and B. Koslowski, J. Vac. Sci. Technol. A 8 (1990) 590–593.
[30] M.S. Keshner, Proc. IEEE 70 (1982) 212–218.
[31] T. Tiedje, J. Varon, H. Deckman and H. Stokes, J. Vac. Sci. Technol. A 6 (1988) 372–375.
[32] D.W. Abraham, C.C. Williams and H.K. Wickramasinghe, Appl. Phys. Lett. 53 (1988) 1503–1505.
[33] A. Schaper, J.P.P. Starink and T.M. Jovin, FEBS Lett. 355 (1994) 91–95.
[34] M. Steinert and S. van Assel, Exp. Cell Res. 96 (1975) 406–409.
[35] P.T. Englund, Cell 14 (1978) 157–168.
[36] L.I. Pietrasanta, A. Schaper and T.M. Jovin, J. Cell Sci. 107 (1994) 2427–2437.