ELSEVIER

Ultramicroscopy 92 (2002) 33–46

ultramicroscopy

www.elsevier.com/locate/ultramic

Efficiency of 2D alignment methods

Laurent Joyeux, Pawel A. Penczek*

Department of Biochemistry and Molecular Biology, University of Texas Health Science Center, 6431 Fannin, MSB6.218 Houston, TX 77030, USA

Received 5 April 2001; received in revised form 18 October 2001; accepted 6 November 2001

Abstract

In single particle analysis, the alignment of two-dimensional images is a fundamental step aimed at bringing into register various particle views of biological macromolecules observed with the electron microscope. The computational efficiency of this step is a deciding factor in design of alignment strategies for large sets of noisy data and in development of three-dimensional structure refinement methods. In addition, the accuracy of the alignment method varies depending on the numerical solutions adopted to efficiently perform exhaustive searches for three orientation parameters. The selected alignment methods are analyzed in terms of their computational complexity and the estimates of numbers of arithmetic operations for each method are given. The tests of alignment accuracy are performed using images simulated in accordance with the linear theory of image formation in the electron microscope. It is demonstrated that the efficiency of the alignment methods can be improved if approximate centers of gravity of particle views are known. The accuracy of the methods considered is largely affected, particularly for high noise levels, by the order in which interpolation steps are applied.

1. Introduction

The problem of determining the relative orientations of two-dimensional (2D) images is a fundamental one in single particle analysis of electron microscope (EM) data [1]. The images collected at the microscope are projections of presumably the same biological macromolecule found in various orientations on the support grid. Depending on the specimen, the preparation technique used, and other, rather poorly understood factors, the number of exhibited orientations can vary largely. The observed views may have a nearly uniform distribution, or there can be few strongly preferential orientations. In either case, it is essential to sort through the amassed data set and select homogenous subsets of particle views [2,3]. This is achieved through 2D alignment, which in its core has a step during which two 2D images are brought into register, and through subsequent classification of the data. Most 2D alignment methods used in EM are iterative [4,5]; thus, their efficiency critically depends on the number of arithmetic operations required to align two images

The ultimate goal of single particle analysis is to obtain a faithful representation of the 3D distribution of mass of the biological macromolecule. After initial steps of 2D alignment and classifica-

*Corresponding author. Tel.: +1-713-500-5416; fax: +1-713-500-0652.

E-mail address: pawel.a.penczek@uth.tmc.edu (P.A. Penczek).

0304-3991/02/$ - see front matter
PII: S0304-3991(01)00154-1

tion, a preliminary low-resolution 3D structure is obtained. Various 3D refinement methods aimed at improvement of the resolution of the structure have been proposed [6–11]. In most of them, particle views have to be compared with large numbers of reference images—generated by projecting the 3D structure in a number of quasi-evenly spaced directions—in order to find the orientation parameters necessary to perform the 3D reconstruction. These refinement methods rely heavily on the alignment of two 2D images or, to be more precise, on the comparison of the input data with a set of 2D reference images. Whether this step is implemented exactly as described depends on the particular method used, but most techniques can be conceptually reduced to such an operation. Usually, the number of particle views is larger than the number of reference images by at least an order of magnitude. Therefore, to reduce the time of calculations, every effort has to be made to pre-calculate and store in the computer memory an appropriately transformed set of reference images. Due to a very large number of particle views analyzed in cryo-EM (currently in the range of 10^4, soon expected to reach 10^6 images [12]) the efficiency of 3D refinement methods is the readily recognizable limiting factor in the efforts to improve resolution of the results.

2D alignment methods can be divided into two classes: those that employ exhaustive search in order to find three orientation parameters, and those that use either simplifications (separate searches for translation and rotation parameters) [4], or take advantage of invariant image representations [13–16]. More recently, an interesting approach based on the singular value decomposition of the sinogram image representation was proposed [17]. Although this method is potentially very efficient, the details of the implementation have to be yet fully worked out. Therefore, in what follows we will primarily focus on the exhaustive search methods, as in most cases they yield superior results. In addition, we will make comparisons with an autocorrelation function-based method [18], which due to employment of invariants is computationally very efficient.

2. Methods

The alignment of 2D images in single particle analysis should be considered as a special case of the general problem of template matching [19]. The templates, or as they are referred to in single particle analysis, the reference images, are selected particle views obtained either with the help of a clustering method, or generated as 2D projections of the available 3D reference structure. Thus, we can assume that reference images are at least approximately centered and that there is only one object within the image frame. Individual particle views, as windowed from micrographs, have to be compared with reference images in order to find the most similar one and in this process to find three orientation parameters: two translation parameters and the rotation angle. In what follows, we will assume that the approximate particle size is known and that the image size n was chosen not to exceed the particle size by more than 15–20%. Due to technical reasons, we will also assume that the particle views are always located in the image frame within a circle with a radius $n/2$. The geometrical constraints described are illustrated in Fig. 1.

Finally, due to random orientations of particles on the support grid, it can be expected that half of the particle views will be mirrored versions of the other half. In practice, due to anisotropic shapes of most of the macromolecules, such an even division is unlikely; nevertheless, during both the 2D alignment and 3D refinement of the structure it is necessary to consider particle views related by mirror operation. As will be explained later, some 2D alignment methods yield themselves to this requirement better than others.

All 2D alignment methods considered are aimed at finding such transformation parameters that the least-square discrepancy between two images f and g is minimized:

$$\int |f(u) - g(\mathbf{T}u)|^2 \mathrm{d}u \to \min, \tag{1}$$

where $u = [\,u_x \quad u_y \quad 1\,]^{\mathrm{T}}$ is a vector containing the coordinates, and $\mathbf{T}$ is the transformation matrix

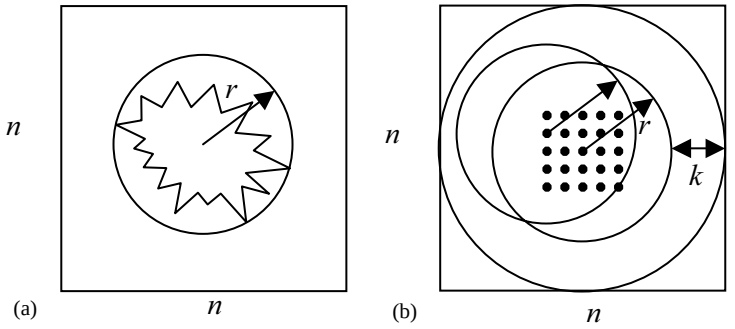

Fig. 1. The geometrical constraints of the 2D alignment problem. (a) The reference particle view is placed within a square image frame $n \times n$ pixels and its size is such that it can be bounded by a circle with a radius no larger than $r = n/2$. (b) The particle view, which size is bounded by the same radius as the reference view, can be located within a circle centered on discrete locations within the image frame, such that the maximum translation is $k = n/2 - r$. The number of possible translations is $(2k+1)^2 = l$.

given by

$$T(\theta, x, y) = \begin{bmatrix} \cos\theta & -\sin\theta & x \\ \sin\theta & \cos\theta & y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

and dependent on three transformation parameters: rotation angle θ and two translations x and y. It has to be noted that minimum of Eq. (1) can be found rapidly using the fast fourier transform (FFT) algorithm if only the x–y translation is sought (2D FFT), or if only the rotation angle is needed (1D FFT).

2.1. Measure of the accuracy of alignment

2D alignment methods yield three transformation parameters: rotation and two translations(θ, x, y). These parameters define how a source image is transformed into a destination image. The transformation T is applied to the source image I_1 to transform all the pixels into destination image I_2, and the new coordinates $v = [\,v_x \quad v_y \quad 1\,]^{\mathrm{T}}$ are computed using a matrix multiplication $v = Tu$.

In order to compare two transformations T_1 and T_2 we have to define a measure of error between them. We will define this measure as a maximum pixel discrepancy calculated over the area of object support that results from application to the image of a combined transformation $T_2^{-1}T_1$:

$$e_{\max}(T_1, T_2) = \max_{u \in D_r} \|\varepsilon_{T_1,T_2}(u)\|, \tag{3}$$

where D_r is a disk with radius r that corresponds to the particle size, and the error vector $\varepsilon_{T_1,T_2}(u)$ is given by

$$\varepsilon_{T_1,T_2}(u) = v - u = T_2^{-1}T_1 u - u = (T_2^{-1}T_1 - I)u. \tag{4}$$

Obviously, if there is no error between two transformations, that is if $T_1 = T_2$, then $\varepsilon_{T_1,T_2}(u) = 0$ and $e(T_1, T_2) = 0$. $\|\varepsilon_{T_1,T_2}(u)\|^2$ is a positive bilinear form; thus, it has only one nontrivial minimum. Since there is no scale transformation between both axes, the form $\|\varepsilon_{T_1,T_2}(u)\|^2$ is also isotropic. $\|\varepsilon_{T_1,T_2}(u)\|^2$ has a form:

$$\|\varepsilon_{T_1,T_2}(u)\|^2 = \alpha\|u - m\|^2 + \beta, \tag{5}$$

where m is the point at which $\|\varepsilon_{T_1,T_2}(u)\|^2$ reaches minimum with value β. In order to calculate the pixel error we have to find point $M \in D_r$, which maximizes $\|\varepsilon_{T_1,T_2}(u)\|^2$. Maximization of $\|\varepsilon_{T_1,T_2}(u)\|^2$ is equivalent to maximization of the distance between M and m. The point M is on C_r, a circle delimiting D_r, which intersects the line defined by the point m and the center of C_r:

$$M = r\frac{-m}{\|m\|}. \tag{6}$$

Thus, the final expression for the pixel error is

$$e_{\max}(T_1,T_2) = \|\varepsilon_{T_1,T_2}(M)\| = \sqrt{\alpha\|M-m\|^2+\beta}. \quad (7)$$

In order to calculate $m, \alpha,$ and β, we have to derive the explicit expression of $\|\varepsilon_{T_1,T_2}(u)\|^2$. The evaluation of $\varepsilon_{T_1,T_2}(u)$ requires calculation of $T_2^{-1}T_1$:

$$T_2^{-1} = T(-\theta_2, -x_2\cos\theta_2 - y_2\sin\theta_2, x_2\sin\theta_2 - y_2\cos\theta_2),$$
$$T_2^{-1}T_1 = T(\Delta\theta, X, Y),$$

where

$$\Delta\theta = \theta_1 - \theta_2,$$
$$X = (x_1-x_2)\cos\theta_2 + (y_1-y_2)\sin\theta_2,$$
$$Y = -(x_1-x_2)\sin\theta_2 + (y_1-y_2)\cos\theta_2. \quad (8)$$

Thus, $\varepsilon_{T_1,T_2}(u)$ is given by

$$\varepsilon_{T_1,T_2}(u) = \begin{bmatrix} \cos\Delta\theta - 1 & -\sin\Delta\theta & X \\ \sin\Delta\theta & \cos\Delta\theta - 1 & Y \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ 1 \end{bmatrix} = \begin{bmatrix} a & -b & X \\ b & a & Y \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ 1 \end{bmatrix} \quad (9)$$

and its square norm by

$$\|\varepsilon_{T_1,T_2}(u)\|^2 = (au_x - bu_y + X)^2 + (bu_x + au_y + Y)^2. \quad (10)$$

The partial derivatives of Eq. (10) are

$$\frac{\partial\|\varepsilon_{T_1,T_2}(u)\|^2}{\partial x} = 2((a^2+b^2)u_x + aX + bY),$$
$$\frac{\partial\|\varepsilon_{T_1,T_2}(u)\|^2}{\partial y} = 2((a^2+b^2)u_y - bX + aY). \quad (11)$$

For m to be the minimum of $\|\varepsilon_{T_1,T_2}(u)\|^2$, both derivatives have to be equal to zero. Solving the corresponding system of equations, we obtain

$$m = \frac{1}{a^2+b^2}\begin{bmatrix} -aX - bY \\ bX - aY \\ 1 \end{bmatrix}. \quad (12)$$

Using Eqs. (6), (8), (10) and (12), the equation for pixel error can be rewritten as

$$e_{\max}(T_1,T_2) = d\left|\sin\frac{\Delta\theta}{2}\right| + \sqrt{\Delta x^2 + \Delta y^2}, \quad (13)$$

where $\Delta\theta = \theta_1 - \theta_2, \Delta x = x_1 - x_2, \Delta y = y_1 - y_2$, and d is the diameter of the particle. The pixel error given by Eq. (13) has two components. The first one is proportional to the angular error between transformations, and the second one is simply the translation error between transformations.

Similar expression can be derived for an average pixel error between two images:

$$e_{ave}(T_1,T_2) = \sqrt{d^2\sin^2\frac{\Delta\theta}{2} + \Delta x^2 + \Delta y^2}. \quad (14)$$

Since both measures are monotonic, they will yield similar results. Nevertheless, the maximum pixel error is more informative as it can be directly related to the highest achievable accuracy of the alignment methods set by the pixel size used. Therefore, in what follows, we will use the maximum pixel error (Eq. (13)) in the analysis of the alignment methods.

3. Results

3.1. Preparation of the test image

In the preparation of the test image, we attempted to emulate the image formation process in the electron microscope, as described by the linear contrast transfer theory (CTF) (weak-phase approximation) [20]. In addition, we designed the test image such that the signal to noise ratio ($\mathrm{SNR} = \sigma^2_{\mathrm{signal}}/\sigma^2_{\mathrm{noise}}$) is well defined and can be easily controlled both in real as well as in Fourier space. In the latter case, the SNR is referred to as spectral SNR (SSNR). The original test image was chosen to be a binary, quasi-rectangular shape approximately 34×18 pixels placed in a window 63×63 pixels (Fig. 2a). First, in order to equalize the power spectrum, the squared Fourier amplitudes of the image were adjusted such that their rotational average became constant as a function of spatial frequency (Fig. 2b). Second, a CTF was

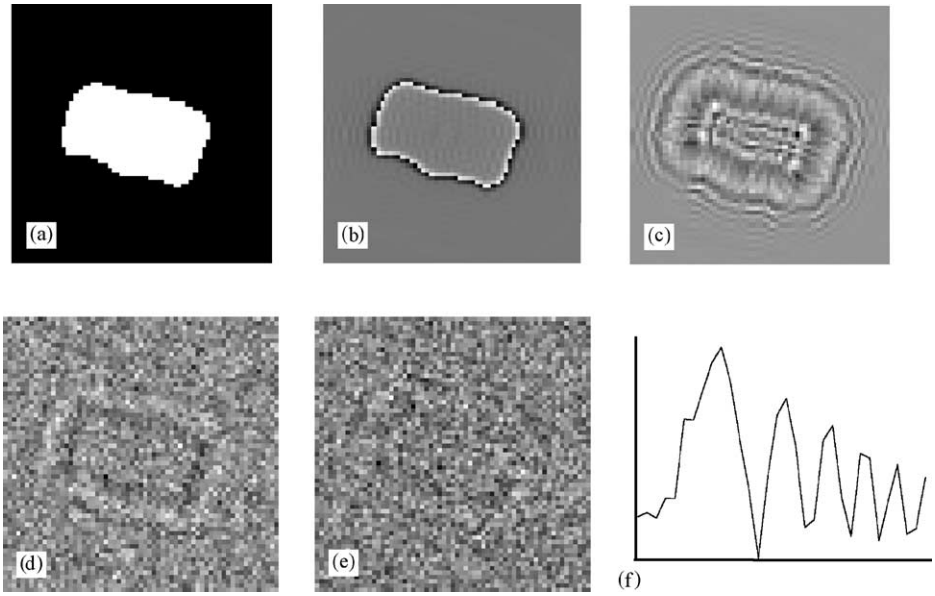

Fig. 2. Preparation of the test image. **(a)** The original binary test image. **(b)** The binary image after an adjustment of its Fourier amplitudes so their rotational average is constant as a function of frequency. **(c)** The final test image obtained by application of a CTF to image (*b*). **(d)** Test image (*c*) corrupted by Gaussian noise resulting in SNR = 0.1. **(e)** Test image (*c*) after application of a Gaussian low-pass filter (half-width = 10 Fourier pixels) and corrupted by Gaussian noise resulting in SNR = 0.15. The SSNR in image (*e*) has an overall shape of the square of the CTF applied to the image attenuated by the Gaussian envelope function. Thus, the SSNR is 0.06 at Fourier pixel 1, increases to a maximum of 0.45 at Fourier pixel 10, has first zero at Fourier pixel 14, and subsequently has a number of maxima with steadily decreasing values. **(f)** The rotationally averaged power spectrum of the test image (*c*).

applied to the image (for the definition of the CTF see Eq. 8 in Zhu et al. [21]). The sign of the CTF was chosen such that the positive contrast of the image was preserved (Fig. 2c). Since the CTF of an electron microscope has a sinusoidal form, the parameters of the CTF used were adjusted so that the value at Fourier pixel zero was 0.1, the first maximum of 0.85 was at Fourier pixel nine, and the first zero-crossing was at Fourier pixel thirteen. Finally, a Gaussian noise, therefore a noise with constant power spectrum, was added to the test image. This was done in two different ways during respective tests. In the first set of tests, only the amplitude of the noise in real space was varied; thus, the resulting images had a CTF-derived shape of the SSNR (Fig. 2d). The level of noise was adjusted to produce test images with SNR in the range between 0.001 and 7.5. In the second set of tests, in accordance with the general tendency of images of macromolecules to have their power spectrum decreasing as a function of spatial frequency and in order to emulate various resolution-limiting factors of the image formation process (usually described by the so-called envelope functions [21]), the test image was low-pass filtered using a Gaussian function with the half-width ranging from 2.5 to 30.0 Fourier pixels. This operation, after addition of a constant level of Gaussian noise, resulted in test images that had SSNR decreasing as a function of spatial frequency. For each of these images, the effective SNR was also calculated.

During the tests, the CTF-modified, but noise-free source image was transformed using a predefined set of transformation parameters (rotation angle 120°, x translation 4 pixels, and y translation -4 pixels). Since the alignment methods tested employ exhaustive searches, the particular values chosen for the transformation parameters are irrelevant. Next, independent

Gaussian noise was added to both images and the transformation parameters were calculated using the chosen 2D alignment method with the source image selected as a reference image. For the alignment method based on resampling to polar coordinates, the search range k was set to 5. The error of the transformation found was calculated using Eq. (13). Since the results proved to be quite unstable, particularly for low SNR, the tests were repeated 1000 times using different realizations of Gaussian noise, and the average pixel error and its standard deviation were calculated and used for the final analysis.

In order to test the dependence of the alignment accuracy on the shape of the test object used, all the tests were repeated using another, quasi-circular test object. Since the results were essentially the same as those described below, for the sake of brevity they are not shown.

All tests were performed using the SPIDER image processing system [22]. The 2D alignment using resampling into polar coordinates is available in this system as the AP NQ command (see the manual of SPIDER commands at http://www.wadsworth.org/spider_doc/spider/docs/spider.html). The remaining methods are implemented as SPIDER scripts and are available upon request from the authors.

3.2. Elementary operations in 2D alignment methods

All 2D alignment methods take advantage of a small number of basic operations. In order to make comparisons, we will define these operations, assign to them the number of arithmetic operations they require, and express each 2D alignment method as a sequence of elementary operations chosen. It has to be understood that efficiency of practical implementations varies, but our concern is with the order of magnitude of the number of operations rather than with intricacies of particular implementation. Thus, we will consider the following five elementary operations:

- shift image in real space by integer number of pixels—negligible, as it does not require interpolation,
- rotate image—n^2,
- project image onto 1D line—n^2,
- convert image to polar coordinates—n^2,
- 1D FFT—$n \log_2 n$,
- 2D FFT—$2n^2 \log_2 n$.

Three of the elementary operations listed (rotation, projection onto 1D line, conversion to polar coordinate) require interpolation of an image in order to determine image values between the original grid points. Both the accuracy of the interpolation and the number of arithmetic operations depend on the interpolation method used. The often-used bilinear interpolation is the simplest and fastest one, but it is the least desirable in terms of the quality of the results. Higher-order interpolations reduce detrimental low-pass filtering effects at the expense of larger number of arithmetic operations necessary to perform them. For the tests described, we used quadratic interpolation implemented in the SPIDER system.

In terms of the elementary operations selected, calculation of a 2D auto-correlation function (ACF) using FFT requires one forward FFT ($2n^2 \log_2 n$), calculation of the modulus of the Fourier transform (n^2), and one inverse FFT ($2n^2 \log_2 n$), bringing the total number of operations to $n^2(4 \log_2 n + 1)$. Taking into account that to avoid aliasing effects the image has to be padded with zeroes to at least double the original size, the number of operations is $4n^2(4 \log_2 n + 5) \cong 16n^2 \log_2 n$.

For each 2D alignment method, two phases of the procedure are considered. During the first one, the reference images are prepared; during the second one, the particle views are compared with the reference images.

3.3. Direct alignment in real space

Two images are compared directly in real space using exhaustive search, which means that Eq. (1) has to be evaluated for all possible orientations of the particle view. During the first phase, $2\pi n/2 = \pi n$ versions of the reference image progressively rotated by angle $360/(\pi n)$ degrees have to be created, which requires $(\pi n)n^2 = \pi n^3$ arithmetic operations. In case many different reference

images are used, this step has to be repeated for all of them. During the second phase, for each possible shift (see Fig. 1) the particle image has to be compared with all the rotated reference images, which requires $(\pi n)n^2 = \pi n^3$ arithmetic operations. With the number of possible shifts $l = (2k+1)^2$, the total number of operations is $l\pi n^3$.

If the mirrored versions of reference images are to be considered, both the memory requirements and the number of operations have to be doubled. The advantages of this method are that except for rotation in real space, no other interpolation or transformation is required and that in the algorithm discrepancy forms other than that given by Eq. (1) can be employed, for example the L_1 norm.

3.4. Direct alignment using 2D FFT

The rotated versions of reference images are required, but the translation parameters are estimated using the standard FFT technique, which is employed to calculate a 2D discrete cross-correlation function (CCF). Therefore, unlike in the direct alignment in real space, the applicability of this method is restricted to the alignment criterion given by Eq. (1). During the first phase, $2\pi n/2 = \pi n$ versions of the reference image progressively rotated by angle $360/(\pi n)$ degrees are calculated (πn^3) and their 2D FFTs are calculated $((\pi n)(2n^2 \log_2 n) = 2\pi n^3 \log_2 n)$ and stored. Thus, for the first phase $n^3\pi(2\log_2 n + 1)$ operations are needed. During the second phase, the FFT of the particle image has to be calculated $(2n^2 \log_2 n)$, multiplied by each rotated version of the reference image $((\pi n)n^2 = 2\pi n^3)$, and inverse FFTs have to be calculated $((\pi n)(2n^2 \log_2 n) = 2\pi n^3 \log_2 n)$, bringing the total number of operations to $2\pi n^3(\log_2 n + 1) + 2n^2 \log_2 n \approx 2\pi n^3 \log_2 n$.

If the mirrored versions of reference images are to be considered, both the memory requirements and the number of operations have to be doubled. The advantage of this method is that there is no restriction on the estimated maximum shift. The disadvantages are caused by the practicalities of the CCF estimation using the FFT technique. First, to prevent real space aliasing effects, the images have to be padded with zeroes to double the image size. This increases the number of estimated operations to $\approx 16\pi n^3 \log_2 n$. Second, due to the padding, the CCF coefficient (k_x, k_y) has $(n - k_x)(n - k_y)$ non-zero contributions; thus, different CCF coefficients have different statistical weights.

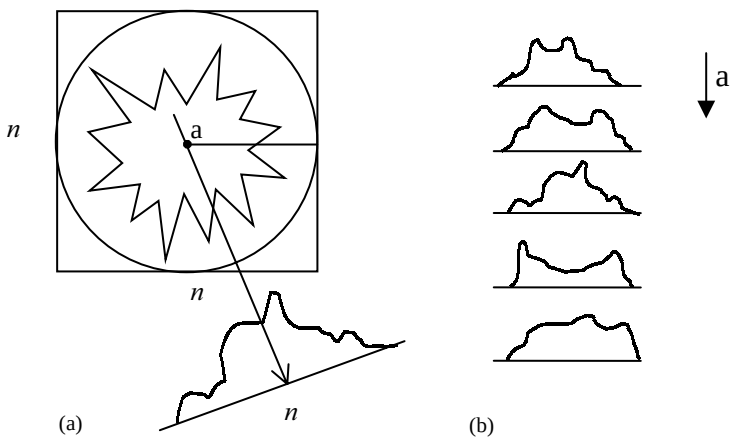

Fig. 3. The principle of 2D Discrete Radon Transform. (a) The 1D projection of a 2D object is calculated by summation of pixels within the circle with radius $r = n/2$ along lines parallel to the direction of projection at the angle α. (b) The set of πn 1D projections forms the 2D DRT. The 1D projections are arranged in a 2D matrix according to the value of the projection angle forming a so-called sinogram.

3.5. Sinograms

Simultaneous search for three orientation parameters can be carried out using a 2D discrete radon transform (DRT) technique, as proposed by Radermacher [7]. First, the 2D DRT is formed by calculation of $\pi n/2$ 1D projections of a 2D image in angular directions evenly spaced in the range $[0,\pi[$ (Fig. 3). This can be carried out either in real space (as assumed in the description below), or using interpolation in Fourier space [23]. The latter approach, even if computationally efficient, is bound to result in non-local interpolation artifacts in real space. In either case, the number of required line projections is reduced because the line projection calculated at the angle $\varphi + 180$ is a mirrored version of the line projection at the angle φ. The resulting set of line projections is often referred to as a "sinogram" [24]. In this representation, the rotation of the image corresponds to permutations and mirroring of the lines in DRT, and the translation to appropriate shifts of the DRT lines [7].

In order to perform the 2D alignment, the 2D DRTs of reference images have to be formed and a 1D FFT of each line projection in the sinogram has to be calculated. This requires $(\pi n)(n^2 + n\log_2 n)$ operations. During the second phase, the 2D DRT of a particle view and 1D FFTs of line projection have to be calculated $((\pi n)(n^2 + n \log_2 n))$. Next, for each of πn rotation angles (corresponding to the appropriate permutation of lines in the sinogram), 1D CCFs between pairs of lines have to be calculated $((\pi n)(\pi n/2)(n\log_2 n) = 0.5\pi^2 n^3 \log_2 n)$ and the cross-correlation coefficients have to be calculated by appropriate summations of 1D CCFs $((\pi n)(\pi n/2)n = 0.5\pi^2 n^3)$ [7], bringing the total number of operations to $0.5\pi^2 n^3(\log_2 n + 1)$. Since the 1D lines in sinograms have to be padded with zeroes to double the length, the number of operation is $\cong 4\pi^2 n^3 \log_2 n$.

Comparisons with mirrored versions of reference images can be implemented efficiently by taking advantage of the fact that the 2D DRT representation of the mirrored image is obtained from the 2D DRT of the original image by changing the assignment of angles in line projections by $\varphi \to 180 - \varphi$.

3.6. Alignment using resampling to polar coordinates

In this method, the image is resampled to polar coordinates with respect to selected origin locations within the image frame (Fig. 3b)—which corresponds to shifts of the image—and the 1D FFT algorithm is used to rapidly calculate the rotation angle for each of the resampled versions of the image. In order to prepare the reference images, it is enough to resample them to polar coordinates (n^2) and calculate 1D FFTs for each of the resulting concentric rings. The length of each ring is no larger than πn; thus, the number of operations is less than $(n/2)\pi n \log_2 \pi n = 0.5\pi n^2 \log_2 \pi n$. During the second phase, the particle view has to be resampled to polar coordinates (n^2), 1D FFTs have to be calculated ($0.5\pi n^2 \log_2 \pi n$), multiplied by 1D FFTs of the reference image (n^2), and finally an inverse 1D FFT has to be calculated ($\pi n \log_2 \pi n$) (for details of the implementation see [4]). This sequence of operations has to be repeated for $l = (k+1)^2$ translations of the particle view, bringing the total number of operations to $0.5l\pi n^2 \log_2 \pi n$.

Comparisons with mirrored versions of reference images can be implemented very efficiently by taking advantage of the fact that in 1D Fourier-polar representation of an image, its mirrored version is obtained by taking the complex conjugate of the 1D Fourier-polar transform.

3.7. Indirect alignment using autocorrelation function

This is the only method considered that does not employ an exhaustive search for orientation parameters. Instead, it is based on the fact that the ACF of an image is invariant with respect to the translation of this image. Thus, the search for orientation parameters can be broken into two steps. First, the rotation angle between ACFs of both images is calculated. Because the ACF is centrosymmetric, there is a 180° ambiguity in the angle obtained, so it has to be resolved by testing both possibilities. Thus, during the second step, one of the images is rotated by the angle found and both the translation parameters and the value of

the correlation coefficient between two images are calculated using the FFT technique. Next, this step is repeated using the angle value increased by 180°, and the corresponding shift and correlation coefficient are found. Finally, two correlation coefficients are compared and orientation parameters corresponding to the larger of the two are chosen.

For the method to perform well, it is necessary to modify the ACFs used. Our goal is to find orientation parameters that minimize the discrepancy between two images in real space, as stated by Eq. (1). This equation can be evaluated in Fourier space:

$$|f-g|^2 \Leftrightarrow |F-G|^2 = |F|^2 - 2FG^* + |G|^2, \qquad (14)$$

where by capital letters we denoted Fourier transforms of respective real space functions. If the search for rotation angle is carried out using an ACFs, the equation corresponding to Eq. (14) is

$$|\mathrm{ACF}(f) - \mathrm{ACF}(g)|^2 \Leftrightarrow ||F|^2 - |G|^2|^2 = |F|^4 - 2|F|^2|G|^2 + |G|^4. \qquad (15)$$

To have the norm of Eq. (15) the same as the norm of Eq. (14), we have to divide FTs of ACFs by moduli of the FTs of respective functions, that is instead of using an ACF defined as

$$\mathrm{ACF}(f) = FT^{-1}(|F|^2) \qquad (16)$$

we have to use the so-called self-correlation function (SCF) [18], defined as

$$\mathrm{SCF}(f) = FT^{-1}(|F|). \qquad (17)$$

In order to prepare reference images, for each its Fourier transform has to be calculated ($2n^2 \log_2 n$) and stored, then its SCF has to be calculated, which requires calculation of moduli of the FT (n^2) and the inverse FFT ($2n^2 \log_2 n$), bringing the total number operations per reference image to $n^2(4 \log_2 n + 1)$. During the second phase, the SCF of the particle view has to be calculated ($n^2(4 \log_2 n + 1)$), the angle between the two SCFs has to be calculated using the resampling to polar coordinates technique described in Section 3.6 ($n^2(\pi \log_2 n + 1) + n \log_2 n$), then the CCF has to be calculated for both the angles found ($n^2 + 2n^2 \log_2 n$) and for the angle increased by 180 degrees ($2n^2 + 4n^2 \log_2 n$). The latter requires additional rotation and an additional FFT of the image. The total number of operations required is $\cong n^2((8+\pi)\log_2 n + 4)$. Since the images have to be padded with zeroes to double the size, the number of operations is $\cong 45n^2 \log_2 n$.

Comparisons with mirrored versions of reference images are achieved during the SCF-based angle calculation using the strategy described in Section 3.6.

3.8. Accuracy of the alignment in case of the uniform noise

The accuracy of three selected 2D alignment methods was compared: "Direct alignment using FFT" (Section 3.4), "Alignment using resampling to polar coordinates" (Section 3.6), and "Indirect alignment using autocorrelation function" (Section 3.7). We omitted two methods. The "Direct alignment in real space" (Section 3.3) is impractical, as indicated by the number of arithmetic operations required (Table 1). Moreover, as far as the dependence on the interpolation errors is concerned, it can be implemented in a way that makes it equivalent to the much more efficient alignment using resampling to polar coordinates (see Discussion). The sinograms-based method is in turn equivalent both in terms of efficiency (see Table 1) and interpolation errors to the direct alignment using FFT.

Test images were obtained as described in Section 3.1. The variance of the CTF-modified simulated image was 0.0189, and after addition of uniform Gaussian noise with the variance ranging from 25 to 0.0025, the test images had SNRs ranging from 0.00075 to 7.56. Since the power spectrum of the simulated image had the shape of squared CTF, the addition of uniform noise resulted in the same, CTF-derived distribution of SSNR in test images.

The plot of pixel error (Eq. (13)) versus SNR for the three alignment methods tested is given in Fig. 4. Since for both extremes of SNR values pixel errors for all methods remained constant, we show only the interesting part of the plot for SNR between 0.01 and 2.0. For each alignment method, the shape of each curve, and thus the general dependence of the accuracy of methods tested on

Table 1
Numbers of arithmetic operations for five selected 2D alignment methods and comparisons of their efficiency

Number of operations per image	3.3	3.4	3.5	3.6	3.7
πln^3	3.3 —	$2(\log_2 n)^{1/2} - 1/2$	$(\pi \log_2 n)^{1/2} - 1/2$	No k	$2(\log_2 n/(\pi n))^{1/2} - 1/2$
$16\pi n^3 \log_2 n$	3.4 $l/(16 \log_2 n)$	—	No k	$2(2qn)^{1/2} - 1/2$	No k
$4\pi^2 n^3 \log_2 n$	3.5 $l/(4\pi \log_2 n)$	~ 1	—	$(2q\pi n)^{1/2} - 1/2$	No k
$0.5\pi ln^2 \log_2 \pi n$	3.6 $2n/\log_2 \pi n$	$(32qn)/l$	$(8q\pi n)/l$	—	$(22q/\pi)^{1/2} - 1/2$
$45n^2 \log_2 n$	3.7 $\pi ln/(45 \log_2 n)$	$\sim n$	$\sim n$	$\pi l/(90q)$	—

n is the image size, k is the maximum translation of the image, $l = (2k + 1)^2$is the number of possible translations, and $q = \log_2 n/\log_2 \pi n$. The alignment methods are: 3.3—direct alignment in real space, 3.4—direct alignment using 2D FFT, 3.5—sinograms, 3.6—alignment using resampling to polar coordinates, 3.7—indirect alignment using autocorrelation function (the numbers refer to the section number in the text containing detailed descriptions of each method). The lower triangle of the table contains ratios of the number of operations of two methods. Thus, cell (3.5, 3.6) contains a ratio of number of operations required by the sinograms methods to the number of operations required by the method based on resampling to polar coordinates. Choosing image size $n = 128$ and maximum translation $k = 6$, we conclude that the sinogram method would require 15 times more arithmetic operations. The upper triangle contains expressions indicating that maximum translation k for which a method that performs explicit translations (3.3 and 3.6) remains faster than a method that yields results for all translations (3.4, 3.5, and 3.7). For example, according to the expression in cell (3.6, 3.5), for image size $n = 128$ the method based on resampling to polar coordinates remains faster than the sinograms method for translations up to $k = 25$.

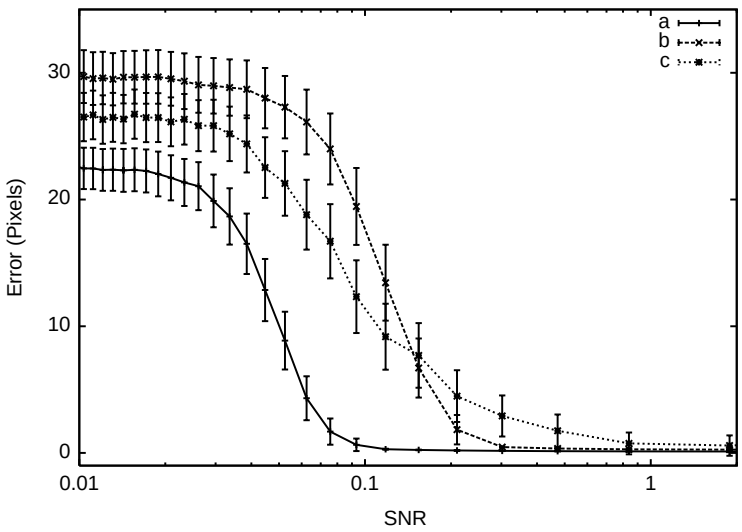

Fig. 4. Pixel error versus SNR for three 2D alignment methods in the case of the uniform noise: (a) alignment using resampling to polar coordinates; (b) indirect alignment using autocorrelation function; (c) direct alignment using 2D FFT. Vertical bars represent the standard deviation (divided by 5 for best visualization).

SNR are similar, although the obtained pixel errors are significantly different, as evaluated by Student's t-test. As expected, the pixel error decreases when SNR increases. For small SNR values (less than 0.04), each curve reaches a maximum that corresponds to the average maximum pixel error of 26, 23, and 30 for the direct alignment using FFT, the alignment using resampling to polar coordinates, and the indirect alignment using autocorrelation function, respectively. In this SNR region, the results of 2D alignment are nearly random, although the distribution of alignment parameters depends on the method used. In general, the distribution of rotation angle found is, for all methods, uniform in the range from 0° to 360°. The distribution of

translation parameters is random and uniform in the range from $-k$ to k for the alignment using resampling to polar coordinates, and Gaussian centered on the origin for the two other methods. The non-uniform distribution of translation parameters in latter cases is caused by the particular implementation of the two methods, which involved application of a circular mask to test images. It is also worth noting that the maximum possible pixel errors for each method are much larger than the average maximum errors obtained. The maximum possible pixel error can be calculated using Eq. (13) and for the range of search parameters used; this error is 63 for the alignment using resampling to polar coordinates, and 100 for two other methods.

Within the range of SNR between 0.2 and 0.04 the pixel error increases sharply. In this relatively narrow region, the reliability of the three alignment methods deteriorates rapidly. This transition occupies the broadest range of SNR for the direct alignment using FFT. Of the main interest is the fact that for such a narrow transition region it is for practical purposes difficult to expect partially correct results of 2D alignment. Instead, it is most likely that, depending on the quality of the data, the results of alignment will be either correct or incorrect for all images analyzed. If we chose the acceptable limit of 2D alignment accuracy to be 0.5 pixel (below this value interpolation artifacts begin to play major role), than the corresponding value of SNR is 0.1 for the alignment using resampling to polar coordinates and 0.3 for the indirect alignment using autocorrelation function. For the direct alignment using FFT, the pixel error is larger than 0.6 for all SNR values used.

3.9. Accuracy of the alignment in case of Gaussian fall-off of the spectral SNR

In this set of tests, we attempted to emulate a more realistic distribution of the SSNR, in which the relative strength of the signal decreases in the high frequency region. This was achieved by an application of a Gaussian low-pass filtration of the simulated image followed by an addition of the uniform Gaussian noise with the variance 0.0225. Depending on the amount of low-pass filtration, the SNR in real space decreases accordingly. For example, for the standard deviation of the Gaussian filter (the filter radius) set to 16 Fourier pixels, the effective SNR in real space is equal to 0.08 and in this case the fall-off of the SSNR is minimal. When the filter radius is set to 8, only the first maximum of the SSNR is retained and its amplitude is reduced by $\sim 33\%$. In this case, the effective SNR is 0.023.

The plot of pixel errors versus SNR for the three alignment methods tested is shown in Fig. 5. The pixel errors for SNR above 0.07 are constant; therefore, we show the results for the SNR range from 5×10^{-4} to 0.1. As in previous tests, pixel error decreases when SNR increases and the overall shape of the error curves is similar. For SNR under 0.03, the error for each method reaches a respective maximum similar to that obtained in previous tests, which means that the alignment results become random. For SNR varying from 0.3 to 0.03, the pixel error decreases rapidly. Again, this transition is different for each method and occupies the broadest range of SNR for the direct alignment using FFT. For SNR larger than 0.3, the pixel errors reach respective minimum values, which are 0.13 for the alignment using resampling to polar coordinates, 0.26 for the indirect alignment using autocorrelation function, and 2.0 for the direct alignment using FFT. For the latter method, pixel error is always larger than 1.0 for all the SNR values tested. The pair-wise comparisons between pixel errors for respective alignment methods indicate that the differences between them are statistically significant, as evaluated by Student's t-test.

The comparison of the results of both tests leads to a conclusion that for a given SNR level, the pixel errors are larger in the case of uniform noise (Section 3.8). This is caused by a different distribution of the SSNR in both tests. In the case of the Gaussian fall-off of the SSNR, the information in Fourier space is accumulated in the low frequency region and this results in lower alignment errors in comparison with the situation in which the same effective SNR in real space originates from SSNR distributed approximately evenly in Fourier space.

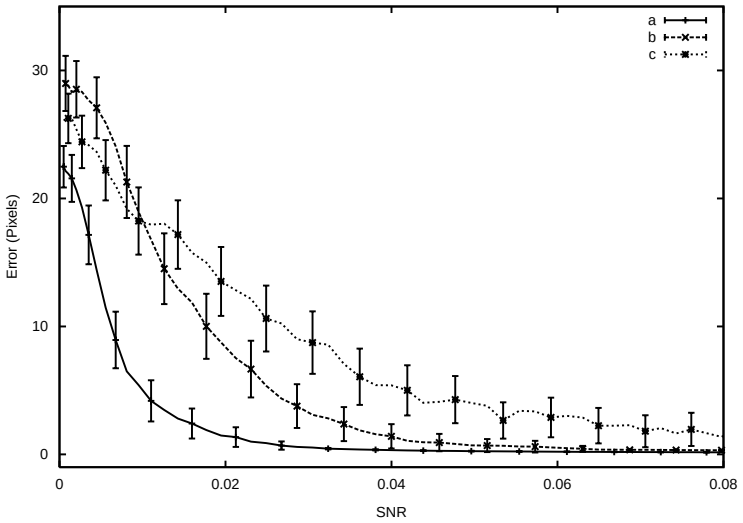

Fig. 5. Pixel error versus SNR for three 2D alignment methods in case of Gaussian fall-off of the spectral SNR: (a) alignment using resampling to polar coordinates; (b) indirect alignment using autocorrelation function; (c) direct alignment using 2D FFT. Vertical bars represent the standard deviation (divided by 5 for best visualization).

4. Discussion

The main factor determining the accuracy of the 2D alignment for the methods tested is the order in which the interpolation steps are applied. Although the translation of the image by an integer number of pixels does not require interpolation, both translation by a non-integer number and any rotation (except for special cases of 90°, 180°, and 270°) do. Any interpolation scheme can be analyzed in terms of frequency response of the associated Fourier-space filter. In image processing applications, preference is given to rapid low-order interpolation schemes. Without going into details, it is enough to assume that the lower the interpolation order, the stronger the low-pass filtering effect of the interpolation on an image. To make matters worse, the characteristics of the associated filter depends on the amount of shift applied. In case of translation, shift by k pixels does not corrupt the image, but shift by $k+x, x=]0,1[$ does, and the low-pass filtering effect is the strongest for $x=0.5$. Similarly, rotation by $\varphi = m\,90, m = \{0,1,2,3\}$ degrees does not corrupt the image, but any intermediate rotation angle will result in a low-pass filtration of the image with the strongest effect for $\varphi = 45 + m\,90, m = \{0,1,2,3\}$ degrees. Thus, application of any method that combines rotation of the reference image (which involves interpolation) with subsequent translation search on a Euclidean grid (which does not involve interpolation) will necessarily favor image orientations with angles close to $\varphi = m\ \ 90, m = \{0,1,2,3\}$ degrees. It has to be noted that the sinograms-based method falls into the described category of alignment methods, as calculation of a line projection is numerically equivalent to a rotation of an image. Therefore, line projections calculated in directions of $\varphi = 45 + m\ \ 90, m = \{0,1,2,3\}$ degrees will have high frequencies suppressed in comparison to line projections calculated at $\varphi = m\,90, m = \{0,1,2,3\}$ degrees.

Admittedly, when the 2D alignment step is performed only once, the error induced by interpolation will be most likely negligible, but in cases when 2D alignment step is repeated numerous times within an iterative procedure (such as 2D reference-free, multi-reference, maximum-likelihood alignments, or 3D structure refinement

procedures), the effects will accumulate and lead to erroneous results. Out of five 2D alignment methods described, only the method based on resampling to polar coordinates (Section 3.6) does not share this problem. The reason is that the order in which orientation parameters are established in this method is reversed. The rotation angle is calculated first, and the translation is found by application of the same rotation-search procedure to images resampled to polar coordinates with respect to origins of the system of coordinates corresponding to translations considered. Thus, irrespectively of any particular orientation parameters, the images that are compared are always interpolated in the same, uniform way.

According to the test results described in Sections 3.8 and 3.9, the accuracy of the alignment ranks the method based on the resampling to polar coordinates as the best, while the direct alignment using FFT ranks as the worst. At the same time, according to the estimations of the number of required arithmetic operations given in Table 1, the direct alignment using FFT is considerably less efficient than the ACF-based method (since this method is equivalent both in terms of interpolation errors and the efficiency to the sinograms-based methods, any conclusions about the former apply to the latter). It is also less efficient than the alignment based on resampling to polar coordinates for broad ranges of possible translations. For example for image size $n=128$, the latter method is more efficient for translations as large as 28 pixels. For the same image size, the alignment based on resampling to polar coordinates is more efficient than the ACF-based method for very restricted searches, namely $k=2$. Since the ACF-based method is remarkably stable for a broad range of SNR values, it should be used whenever the quality of the data is sufficiently high and large translation values are expected. Under any other circumstances, particularly when the images are at least approximately centered (for example by calculation of centers of gravity of the objects) and the translation range can be restricted, the alignment based on resampling to polar coordinates will not only yield better results, but can be almost equally efficient.

5. Conclusions

The techniques of 2D orientation search constitute a foundation of single particle analysis methods. Their efficiency is a major factor in the overall efficiency of 2D and 3D alignment procedures. Therefore, we have expressed five selected 2D alignment methods in terms of five elementary image operations and based on the number of arithmetic operations necessary to perform these elementary operations, we provided detailed analysis of the computational efficiency of the selected methods. We have concluded that the method based on image invariants in the most efficient one, followed by the method based on resampling into polar coordinates, if the translational search in the latter is restricted to small values. We have also tested the accuracy of three 2D alignment methods on a test image prepared according to the basic theory of image formation in the electron microscope. We have demonstrated that in each case, the pixel error remains small for large SNR in the test image and it increases rapidly within a narrow range of decreasing SNR in the data. Therefore, unless the SNR level in the data is known, preference should be given to the most accurate alignment method.

The choice of the 2D alignment method used should be dictated by factors such as the SNR of the data, data collection conditions (particularly the defocus and voltage settings of the electron microscope), and expected range of translation values. In addition, it has to be taken into account that due to iterative nature of many alignment procedures, even minor inaccuracies of 2D orientation search method are likely to be enhanced and may hamper the quality of the final results. The most accurate of the methods tested, the alignment based on the resampling to polar coordinates, is also the most efficient if the maximum translation of particles can be restricted to a predetermined, small value. In this case, nearly exhaustive searches for orientation parameters can be performed effectively within iterative schemes aimed at refinement of three-dimensional structures. Thus, the future work on 2D alignment methods should be focused on development of techniques that

would allow a robust estimation of centers of gravity of particle views.

Acknowledgements

This work was supported, in part, by grant R01 GM60635 to P.A.P.

References

[1] J. Frank, Three-dimensional electron microscopy of macromolecular assemblies, Academic Press, New York, 1996.
[2] J. Frank, Q. Rev. Biophys. 23 (1990) 281.
[3] M. van Heel, B. Gowen, R. Matadeen, E.V. Orlova, R. Finn, T. Pape, D. Cohen, H. Stark, R. Schmidt, M. Schatz, A. Patwardhan, Q. Rev. Biophys. 33 (2000) 307.
[4] P. Penczek, M. Radermacher, J. Frank, Ultramicroscopy 40 (1992) 33.
[5] F.J. Sigworth, J. Struct. Biol. 122 (1998) 328.
[6] P.A. Penczek, R.A. Grassucci, J. Frank, Ultramicroscopy 53 (1994) 251.
[7] M. Radermacher, Ultramicroscopy 53 (1994) 121.
[8] M. Schatz, E.V. Orlova, P. Dube, J. Jager, M. van Heel, J. Struct. Biol. 114 (1995) 28.
[9] T.S. Baker, R.H. Cheng, J. Struct. Biol. 116 (1996) 120.
[10] N. Grigorieff, Curr. Opin. Struct. Biol. 9 (1999) 476.
[11] S.J. Ludtke, P.R. Baldwin, W. Chiu, J. Struct. Biol. 128 (1999) 82.
[12] R.M. Glaeser, J. Struct. Biol. 128 (1999) 3.
[13] M. Schatz, M. van Heel, Ultramicroscopy 32 (1990) 255.
[14] J. Frank, P. Penczek, W. Liu, Scanning Microsc. Suppl. 6 (1992) 11.
[15] M. Schatz, M. Van Heel, Ultramicroscopy 45 (1992) 15.
[16] R. Marabini, J.M. Carazo, Pattern Recognition Lett. 17 (1996) 959.
[17] P.L. Bellon, F. Cantele, S. Lanzavecchia, Ultramicroscopy 87 (2001) 187.
[18] M. van Heel, M. Schatz, E. Orlova, Ultramicroscopy 46 (1992) 307.
[19] A.K. Jain, R.P.W. Duin, J. Mao, IEEE Trans. PAMI 22 (2000) 4.
[20] R.H. Wade, Ultramicroscopy 46 (1992) 145.
[21] J. Zhu, P.A. Penczek, R. Schröder, J. Frank, J. Struct. Biol. 118 (1997) 197.
[22] J. Frank, M. Radermacher, P. Penczek, J. Zhu, Y. Li, M. Ladjadj, A. Leith, J. Struct. Biol. 116 (1996) 190.
[23] S. Lanzavecchia, L. Tosoni, P.L. Bellon, Comp. Appl. Bio. Sci. 12 (1996) 531.
[24] M. van Heel, Ultramicroscopy 21 (1987) 111.