Automatic particle detection in digitized electron micrographs

Thesis

Automatic particle detection in digitized electron micrographs

Judith M. Short

Thesis for the Degree of Master of Philosophy

Laboratory of Molecular Biology,
Medical Research Council,
Hills Road,
Cambridge CB2 2QH

November 10, 2005

DATE OF SUBMISSION: 26 JULY 2005
DATE OF AWARD: 8 NOVEMBER 2005

Abstract

High resolution structural analysis of biological complexes can be carried out by single particle electron microscopy where a large number of particle images are available. Many approaches to automate the process of selection of particle positions from digitized electron micrograph images have been described, but so far none has proved as good as manual selection.

This thesis describes a method which I have developed to locate such biological complexes by matching small boxed areas to a set of reference images using the radius of gyration, complemented by a series of other simple criteria. From the reference images, parameters such as the ratio between the average density of the central area and that in its surrounding band, and the density sum and variance are calculated. They are compared with corresponding values from a moving square window of densities extracted from the micrograph, and the coordinates of successfully matched candidate squares are recorded. Since the same particle is detected in a series of overlapping windows, candidates found to be within close proximity are grouped, and the best-fitting one is selected from each cluster. Along with a small stack of boxed reference images, a few specified parameter values, such as the particle radius and the minimum acceptable distance between particle centres are required to select the windows. Micrograph labels and other areas that do not contain appropriate specimens are automatically ignored in order to minimize false positives, and reduce the computing time.

A computer program SLEUTH written to carry out this method of automatic particle detection includes a graphical user interface to assist the user in setting up the parameter values. The program has been tested successfully

on a variety of different biological structures, from both negatively stained and ice-embedded specimens.

Acknowledgements

I would like to thank my supervisors Dr. Linda Amos and Dr. Tony Crowther for continued support, helpful discussions and advice throughout the work and Dr. Richard Henderson for initiating the project and making it all possible. I would also like to thank all the colleagues who very kindly supplied the micrograph images for testing the software and I am indebted to various members of my family for advice on several aspects of this work and for help with the manuscript.

Contents

1 Introduction **1**

1.1 Reference criteria 3

1.2 Image pre-processing 5

1.2.1 Fourier bandpass filtration 5

1.2.2 Histogram modification 6

1.2.3 Anisotropic diffusion 6

1.2.4 Other filters 7

1.3 Particle detection 8

1.3.1 Template matching 8

1.3.2 Edge detection methods 11

1.3.3 Intensity comparison methods 15

1.3.4 Neural network and learning based methods 17

1.3.5 Texture based and other methods 21

1.4 Summary and comparison of methods 24

1.5 Aims of the present work 28

2 Image preparation **31**

2.1 Density inversion 32

2.2 Image compression 32

2.3 Noise filters 32

2.3.1 Fourier filtration 32
2.3.2 Realspace high frequency filtration 34
2.3.3 Realspace low frequency filtration 37
2.4 Contrast modification functions 39
2.5 Histogram modification 40
2.6 Label masking 42

3 The Selection Procedure **45**
3.1 Isolated object of appropriate size 46
3.1.1 Ratio mean and variance test 47
3.1.2 Adjacency test 48
3.2 Density distribution 49
3.2.1 Density sum 49
3.2.2 Variance 49
3.2.3 Radius of Gyration 50
3.3 Circular and radial density distribution 52
3.3.1 Ring parameter tests 53
3.3.2 Sector parameter test 54
3.4 Clustering 55
3.5 Final particle selection 58

4 Reference Value Preparation **61**
4.1 Pre-processing 61
4.2 Particle alignment 62
4.3 Parameter value determination 63
4.4 Ring parameter reference values 63
4.5 Sector parameter reference determination 64
4.6 Outlier rejection 64

4.7 Determination of acceptability ranges 64

5 Program Structure and Use **67**

5.1 Graphical user interface 69

5.2 Command line interface 77

6 Performance and Discussion **79**

6.1 Results with different particle varieties 79

6.2 Results with defocus pairs 83

6.3 Results with a series of micrographs 88

6.4 Comparison with other methods 91

6.4.1 The "Bake-off" 91

6.4.2 Template matching methods 91

6.4.3 Neural networks 92

6.4.4 Intensity Comparison Methods 92

6.4.5 Edge Detection Methods 92

6.4.6 SLEUTH 93

6.5 Conclusions 94

6.6 Publication 95

6.7 Additional software 96

6.8 Further work 96

Chapter 1

Introduction

Solving high resolution structures of biological particles, such as viruses and protein-DNA complexes, invariably requires many copies of the structure in a complete range of different orientations. Where the complex can be persuaded to form three-dimensional crystals, X-ray crystallographic methods can be used, and the structure may be calculated to atomic resolution. Other structures can be studied by electron microscopical techniques; those which form two-dimensional sheets or flattened tubes may be calculated by electron or two-dimensional diffraction methods or where the sheet forms a helical tube, then helical diffraction methods are available.

When particles will not form any kind of repeating pattern, but remain as independent structures, single particle structural analysis may be used. This method is based on averaging together many images of the structure to minimise noise artifacts and provide information from many different angles of view to build a three-dimensional model.

Particles in a digitized micrograph will be randomly positioned and usually

have a distribution of different orientations. Coordinates for each particle centre must be determined prior to extraction of a box of densities for subsequent alignment and classification procedures, for which several well-established software packages are available (Frank et al., 1996; van Heel et al., 1996; Ludtke, Baldwin and Chiu, 1999). Classes of aligned, summed images represent specific projections of the three-dimensional structure, which may then be calculated by weighted back projection or other methods. The resolution of calculated structures from single particle methods is restricted to about 20Å for stained specimens and currently to about 7Å for ice-embedded specimens (Böttcher, Wynne and Crowther, 1997). In principle, even higher resolution should be possible for ice-embedded specimens (Henderson, 1995; van Heel et al., 2000). Several factors, including contrast transfer function and temperature factor, which reflects contrast loss due to imperfect images, affect resolution; software is under development to correct for these factors with the aim of structure determination to atomic resolution (Grigorieff, 1998; Glaeser, 1999; Rosenthal and Henderson, 2003).

Ultimately, it is the availability of many thousands, perhaps millions of particles which will make possible the calculation of high resolution structures by single particle methods. A high resolution analysis requires many projections, and many particle images for each projection. Imperfections in the specimens and background effects due to various ice artifacts lead to distortions in a calculated model. Furthermore, the signal-to-noise ratio is decreased in low-dose images, which are necessary to minimize radiation damage. However, it may be possible to overcome these problems by averaging together a sufficiently large number of particle images for each projection. The manual selection of such huge numbers of particles is impractical, and several software packages

designed to automate this process have already been described. Many, but not all of them, use some kind of reference criterion for matching purposes; some use a rotated, averaged particle as a template while others simply require the particle dimensions. Pre-processing micrograph images to reduce background noise is often found to be helpful, and several different techniques have been proposed for this purpose. Correlation-based methods dominate the choice of algorithms for automatic particle detection; others include edge detection, neural networks, intensity (density) comparisons, and texture based methods.

1.1 Reference criteria

Detecting particles automatically without reference images by selecting individual isolated areas of high density has been described (Lata, Penczek and Frank, 1995, Adiga et al., 2004, Singh, Marinescu and Baker, 2004). However, biological structures present many different shapes and sizes and it is difficult to see how isolated artifacts such as air bubbles would be excluded in the case of spherically-symmetric particles, and how weak, but true particle images, particularly in the case of low defocus images, would be detected.

The majority of algorithms use some kind of reference. A single rotated, averaged particle image used as a reference template restricts the method to detecting spherical, or near-spherical images (Frank and Wagenknecht, 1984; Thuman-Commike and Chiu, 1995; Plaisier et al., 2004), and some techniques are unable to track any other shape (Boier Martin et al., 1997; Kivioja et al., 2000; Saad, Chiu and Thuman-Commike, 1998). Yu and Bajaj (2004) use simple geometric information, such as the radius in the case of a spherically-symmetric particle and side lengths for rectangular images; this method would be unable to accommodate multiple views of irregularly shaped objects with-

out multiple runs of their software.

Elongated and L-shaped particles are much more difficult to detect as their end and side views differ in size and shape from each other. To overcome this problem, some algorithms use a set of template images to represent as many of the different views as possible. Since the signal-to-noise ratio in single raw particle images has been found to be insufficient to provide workable references, several images per view can first be rotationally aligned to each other and then averaged together (Roseman, 2004). Projections generated from known three-dimensional models have also been used as templates (Rath and Frank, 2004; Wong, et al., 2004); Huang and Penczek (2004) use a clustering and averaging procedure to reduce the number of templates but retain sufficient detail. The necessity of pre-determining a proven structure is a serious drawback and must be taken into account in terms of both user and computing time. Before calculating a three-dimensional model, it is likely that several thousand particles will have to be manually selected. Furthermore, there is a considerable risk that the computed model may be incorrect and hence generate projections unable to match the required raw particles. The user-specified polygon described by Kumar et al. (2004) does not require the model calculation, but like all the multiple template methods, it does require many lengthy rotations. In an attempt to reduce the considerable computional cost of the rotations, Sigworth (2004) derives templates from the two-dimensional eigenimages calculated by the principal component analysis step carried out in the classification stage of a single particle reconstruction. Eigenimages are also used by Ogura and Sato (2004) as recognition filters in the training of their neural network.

Other techniques, including neural network and learning based methods (Mallick, Zhu and Kriegman, 2004; Ogura and Sato, 2004) require training sets of true and false raw images which consist of a few hundred manually picked boxed particle and background areas.

1.2 Image pre-processing

Background noise due to film grain, particle aggregates, differing thicknesses of ice or stain and other artifacts frequently affect automatic particle detection, causing false particle selection, and missed true particles. Several pre-processing methods have been described to minimise these effects.

1.2.1 Fourier bandpass filtration

Fourier bandpass filtration (Ogura and Sato, 2001; Wong et al., 2004; Roseman, 2003) can be used to to reduce the effects of such artifacts as shot noise, which is caused by the small number of imaging electrons, and uneven illumination by eliminating both high and low frequency data. This technique first requires the calculation of a Fourier transform of the micrograph image. High and low frequency data can then be removed and the resulting array back Fourier transformed to provide a de-noised image. This operation has some drawbacks. It can be costly in terms of computing time to calculate Fourier transforms of large images; the time for computing a Fourier transform is proportional to nlog(n), where n is the number of pixels in the image, and a scanned image can be as much as 12000 x 12000 pixels or more. Furthermore, the specific size constraints required by Fourier transformation will almost certainly make it necessary to clip or pad micrograph images to an appropriate size, adding to the user intervention stage.

1.2.2 Histogram modification

Histogram stretching can be used to improve image contrast by redistributing grey-levels (Boier Martin et al., 1997, Nicholson and Malladi, 2004; Wong et al., 2004). Assuming that the data is distributed over a single peak, this is a straightforward operation. However, where an image contains carbon from the edges of carbon holes (which is necessary for the calculation of defocus values for contrast transfer function correction), there would be two or more peaks. The presence of labels and unexposed areas of film also cause undesirable spikes at the edges of the histogram. Adiga et al. (2004) remove such areas by selecting the micrograph area manually.

1.2.3 Anisotropic diffusion

Anisotropic diffusion, in particular the application of the partial differential equation known as Beltrami flow, is a technique which aims to smooth the background while maintaining particle edges (Nicholson and Malladi, 2004; Singh, Marinescu and Baker, 2004; Yu and Bajaj, 2004). The image is first normalized and its contrast is then improved by histogram stretching. The Beltrami flow equation incorporates an edge indicator function which provides minimum diffusion at the edges and extensive diffusion elsewhere. This computationally expensive process is iterated many times. Finally, a further rank-levelling step replaces every pixel by the minimum grey-level in its neighbourhood to correct for uneven illumination (Figure 1.1).

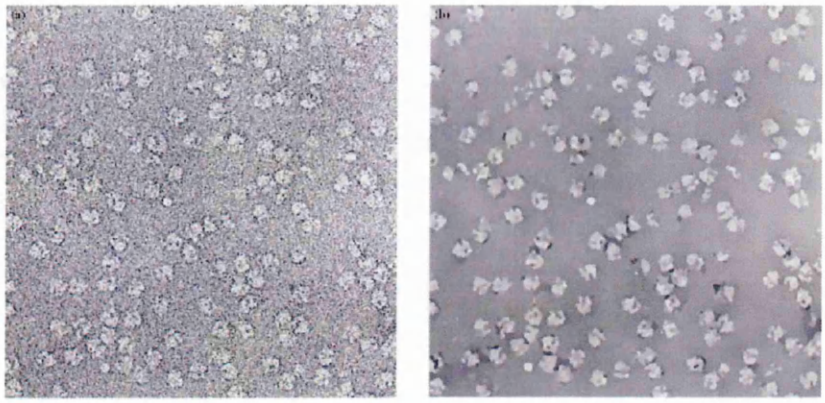

Fig. 1.1 a) original image and b) after de-noising by Beltrami flow.
Reprinted by kind permission of J. Frank and W. Nicholson (Nicholson and Malladi, 2004).

While this operation appears to achieve its objectives quite well, an essential pre-requisite is the manual removal of unwanted areas of film, which is a major disadvantage. Furthermore, the process is certainly much too costly in terms of computing time to be considered as a realistic de-noising technique for my project.

1.2.4 Other filters

The contrast transfer function (CTF) has been used to construct a matched filter (Huang and Penczek, 2004). Each micrograph image is first CTF-corrected, then divided by the noise power spectrum, and finally locally normalized using a Fast Fourier transform technique which determines local mean and variance values within particle-sized windows.

Median filtering is a simple and effective spatial filter for high frequency noise; it maintains edges while removing spike-like components. In this operation, the grey-level of each pixel is replaced by the median of those from neighbour-

ing pixels. Harauz and Fong-Lochovsky (1989) found a 5 x 5 mask appropriate for their particular images. The drawback to this method of filtering is that it can be a lengthy process to compute in the case of a large image and this would be increased still further in the case of a larger mask which might be needed for particles of larger size. Furthermore, it removes only high frequency noise and in practice it is also necessary to reduce low frequency components.

Conversely, the pre-whitening filter described by Sigworth (2004) removes only low frequency components. In this method of filtering, the circularly averaged power spectrum is first computed from blank areas of the micrograph image. It is then fitted to an analytical function and applied to the image in Fourier space. A critical assumption in this algorithm is that all micrograph areas have uniform and identical noise statistics, but in reality large differences can be observed even within individual micrograph images.

In practice it is necessary to reduce both high and low frequency noise to a minimum.

1.3 Particle detection

1.3.1 Template matching

Template matching methods involve scoring a match between a reference image and the micrograph image to detect the presence of a particle. Computation of a cross-correlation between a template image and the micrograph image results in a map with peaks indicating the presence of candidate particles :

$$c(x', y') = \sum_x \sum_y f(x, y)(g(x + x', y + y'))$$

where $f(x, y)$ is the image and $g(x, y)$ is the reference.

The reference is rotationally and translationally aligned relative to the image, the two are then multiplied and values of the product are summed. The result is plotted at position (x', y'). This calculation is most economically carried out in Fourier space where the Fourier transform of the image is multiplied by the complex conjugate of the Fourier transform of the reference; the inverse Fourier transform is then calculated to obtain the cross-correlation function.

A Gaussian profile equal in size to the particle, convoluted with the micrograph image using a standard cross-correlation function, also produces an image with peaks (Lata, Penczek and Frank, 1995; Hall and Patwardhan, 2004). However, the Gaussian distribution depends upon its standard deviation, which is specified by the half-width of the profile, hence this technique is strictly limited to particles of similar size in all directions.

The next stage in template matching is to locate the peaks in the resulting correlation map. Difficulties arise in peak detection due to spatial variation and noise in the images and despite noise suppression techniques a further pruning step is required to remove the many false positives which are invariably detected as peaks.

A simple pre-defined threshold is sometimes used to reject weak peaks and is combined with the calculation of inter-peak distances which are used to indicate the presence of particle aggregates (Frank and Wagenknecht, 1984; Hall and Patwardhan, 2004; Nicholson and Malladi, 2004; Thuman-Commike and Chiu, 1995; Roseman, 2003). However, the determination of an appropriate threshold presents a problem. Hall and Patwardhan (2004) use a set number of standard deviations above the mean; alternatively they use features such

as local mean and variance for matching. Successful matches are pruned by distance clustering, leaving the user to select the clusters. Roseman (2003) passes the decision to the user to select manually the minimum correlation coefficient as the threshold cutoff. Huang and Penczek (2004) calculate a set of cross-correlations between the templates to derive a standard profile. For each peak position, correlations with each template provide a second profile which is matched against the standard profile. Relative entropy is used by Kumar et al. (2004) to reject false positives. Where two discrete functions have probability functions p_k and q_k, then the relative entropy of p with respect to q is defined as :

$$\sum_k p_k log_2 \left(\frac{p_k}{q_k}\right)$$

where p_k and q_k are the probability distributions of the histograms of the box to be tested and the reference box, respectively. The smaller the relative entropy, the more closely matched are the two distributions. Rath and Frank (2004) compare locally normalized cross-correlation functions of adjacent pixel positions, selecting the highest value within an area the size of the template. The statistics s and t are used by Sigworth (2004) to distinguish true from false particles. The correlation peak value s_k of the kth peak of the correlation image is used in conjuction with the error function value of t_k, which is derived from a weighted squared error between particle and reference. Peak shape characteristics provide the thresholds used by Volkmann (2004) in a real-space correlation technique, which filters peaks by distance constraints and an iterative correlation-based outlier screen. Cross-correlation peaks are also evaluated by distance constraints along with approximations to the log likelihood and log likelihood ratios of areas centred on each peak (Wong et al., 2004). Likelihood is defined as the product of individual probabilities of the

data set $\{x_1, x_2, ..x_n\}$:

$$L(x_1, x_2, ..., x_n; a) = \prod P(x_i; a)$$

where the combined probability would be produced from the value a.
For each pixel in the template image, the log probability of observing the corresponding pixel in the image is extracted from tables of logarithms of probability density function values of a range of Gaussian distributions.

The wide variety of approaches to peak filtration suggests that no technique is entirely satisfactory. The two major disadvantages with template matching are sensitivity to noise and processing time. The fast local correlation function described by Roseman (2003) compensates for local variance and improves computation time, but calculation of a correlation map is still a lengthy process, and a manual pruning step is normally required to remove false positives.

1.3.2 Edge detection methods

An advantage of edge detection methods is their insensitivity to shading effects because of their local nature of operation, but they are very sensitive to high frequency noise. Harauz and Fong-Lochovsky (1989) describe a three-phase process of high frequency noise suppression and edge detection, followed by component labelling from which rectangular bounding boxes are derived; the final phase, which they call high-level symbolic processing, is used to select which boxes actually contain suitable particles. Their algorithm is based on a linear-median hybrid edge detector which aims to overcome the effects of high frequency noise when locating the edges. Connected edge regions are extracted by component labelling which gives pixels within a region the same label. Maximum and minimum coordinates and size for each region are used to construct a rectangular surrounding box parallel to the image edges. The

size of the resulting object and its distance from others provide the selection criteria. As described, the method has been tried on only one type of particle (the ribososome) for which a 93% accuracy is claimed. For a large image, median filtering is computationally expensive and despite this operation, the edge detector remains sensitive to high frequency noise.

Less sensitive to noise is the Canny edge detector used by Zhu et al. (2001) to search for filaments in high defocus images whose positions and orientations are then transposed to their closer to focus pairs. The image is first smoothed by Gaussian convolution; the image gradients are then found by a simple two-dimensional first derivative operator to highlight regions with high spatial derivatives. The local derivative is calculated at every pixel position to give a map of density gradients with their directions. Gradients within a region of constant density will be zero, but the converse is true where the density varies, indicating the presence of edges which give rise to ridges in the gradient magnitude image. The algorithm then tracks along the top of the ridges and sets to zero all pixels that are not actually on the ridge top (non-maximal suppression). The gradient array is further reduced by hysteresis thresholding, which is used to track along the remaining pixels. In this case, where the magnitude is below the lower threshold, the pixel is set to zero; if the magnitude is above the higher threshold, it is made an edge. Magnitudes between thresholds are set to zero unless there is a path to a pixel with a gradient above the higher threshold. Discontinuous edges are organised into line segments using the Hough transform : all possible lines are drawn through each edge pixel, an accumulator array stores votes for each intersection of each line with an edge pixel hence peaks in the array indicate the presence of potential lines. Filaments are detected from grouped line segments using

parallelism and information from training data such as inter-line distances.

The Canny edge detector is also adopted by Yu and Bajaj (2004), who remove small connected edge components from the detected edge map where the number of edge pixels in a local region is below a certain threshold. The distance transform of the edge map of the target image is then calculated. The distance transform (Figure 1.2A) calculates a grey-level image similar to the input image except the grey-level density of points inside the foreground regions are modified to show the distance to the nearest boundary from each point (Figure 1.2B).

0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0

→

0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0
0	1	2	2	2	2	1	0
0	1	2	3	3	2	1	0
0	1	2	2	2	2	1	0
0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0

Fig. 1.2A. The distance transform of a simple rectangular shape using the "chessboard" metric. The Euclidean distance can also be used : $D_{Euclid} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

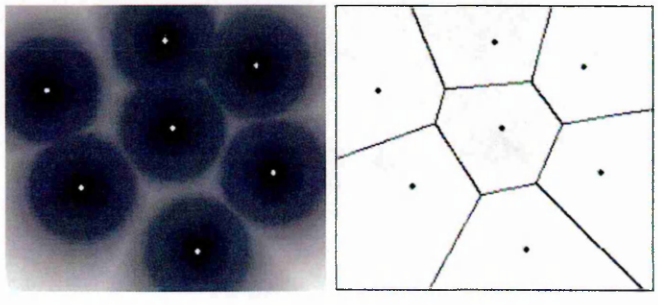

Fig. 1.2B. Example of a distance transform (left), and corresponding Voronoi diagram (right), showing partitioning into convex polygons. *(Reprinted by kind permission of Z. Yu (Yu and Bajaj, 2004).)*

The average distance value along the template contour in the target image provides a measure of goodness-of-fit between target and template at a given location. The template is derived from geometric information of the required particle e.g. radius of circular particles or side lengths in the case of a rectangular image. The Voronoi diagram is computed to estimate initial locations and orientations of rectangular particle views, then centre and orientation refinement is carried out by the distance transform (Figure 1.3).

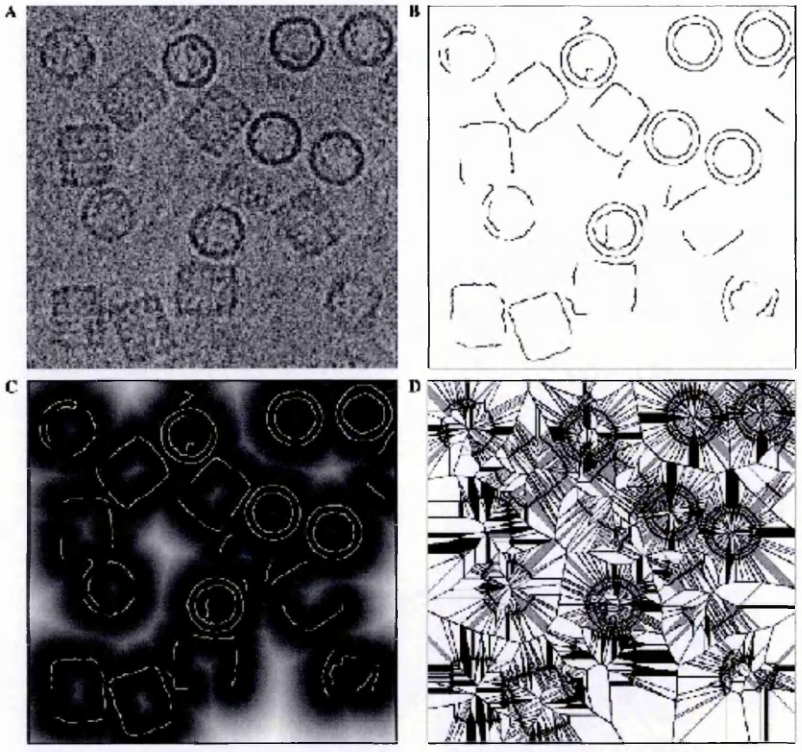

Fig. 1.3. Illustration of edges, distance transform, and Voronoi diagram. A) Original map. B) Edge map obtained by Canny edge detector followed by edge cleaning. C) Distance transform. D) Voronoi diagram.

Reprinted by kind permission of Z. Yu (Yu and Bajaj, 2004).

Although edge detection methods are not sensitive to variations in illumination, they are sensitive to the high levels of density variation both inside and outside particles. Advantages of this approach include their independence of particle shape and orientation, but they do not allow for the exclusion of artifacts.

1.3.3 Intensity comparison methods

The crosspoint technique described by Boier Martin et al. (1997) is a two-step process : marking and clustering. Particle densities are assumed to have lower values than background areas. The marking phase works from top to bottom of the image density array by comparing densities of pairs of pixels at distance $r + 1$ in the horizontal direction, where r is the particle radius. The density difference between the pixels is tested against a threshold, then if the difference exceeds that threshold, the lower density is compared with that of a pixel at $r + 1$ in the vertical direction. If this second difference exceeds the threshold, the lower density element is marked as within the particle (Figure 1.4).

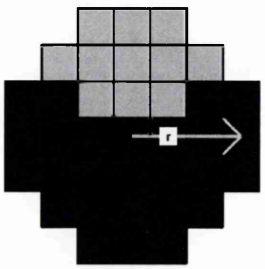

Fig. 1.4. The diagram shows the outcome of the scan procedure in one direction to a spherical particle radius r. The light grey boxes are from the particle and the dark grey boxes have been marked by the algorithm. The pixels have a significant density difference with horizontal and vertical neighbours at distance $r + 1$.
Reprinted by kind permission of T. Baker (Boier Martin et al., 1997).

Improved accuracy is achieved if the transposed image is then scanned again from bottom to top. The second step, clustering, determines the connected components in the marked binary image. Centres of mass of clusters of marked pixels are calculated and the neighbours of each pixel are examined. Clusters of inappropriate size are rejected, then two further filtering steps are carried out. The first compares average densities within each circular area and its surrounding band, and the second applies a morphological "thinning" process to separate individual but very close particles. The method is sensitive to several parameter values including the particle radius r, the number of thinning passes used to disconnect aggregates and the threshold used in the marking phase. A disadvantage to this technique is the necessity of processing a large number of micrographs to optimise the parameter values.

Kivioja et al. (2000) compare averaged density values within a circular area to that in its surrounding ring, by subtraction, as an initial filtering step. Remaining particle positions are then subjected to a comparison of averaged densities from each of eight sectors of a circle drawn around them with their neighbouring surrounding ring sector and also with their adjacent sectors, again by subtraction. A final pruning step applies distance constraints appropriate to the particle radius to remove particle aggregates.

Although these methods are fast they are limited to the detection of spherically-symmetric particles with uniform density in projection.

1.3.4 Neural network and learning based methods

A neural network consists of a set of input nodes, hidden layer nodes and output nodes. The feedforward neural network described by Ogura and Sato (2001) is based on the multilayer perceptron technique (Figure 1.5).

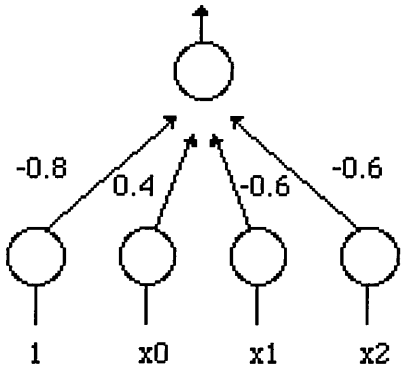

Fig. 1.5. A simple example of a perceptron. The inputs $1, x0, x1, x2$ are weighted by $-0.8, 0.4, -0.6, -0.6$ respectively to the output node. Inputs in the particle detection case would probably be arrays of boxed pixel densities arranged one-dimensionally, where each pixel density is multiplied by each of the weights.

Each input node passes its value to each hidden layer node (artificial neuron) where it is multiplied by a weight associated with that connection. All the values input to each hidden layer node are summed and thresholded by some function, such as the logistic sigmoid :

$$\frac{1}{(1+e^{-x})}$$

where x is the input.

The new values are then passed to the next layer and the process repeated until the output layer is reached (Figure 1.6).

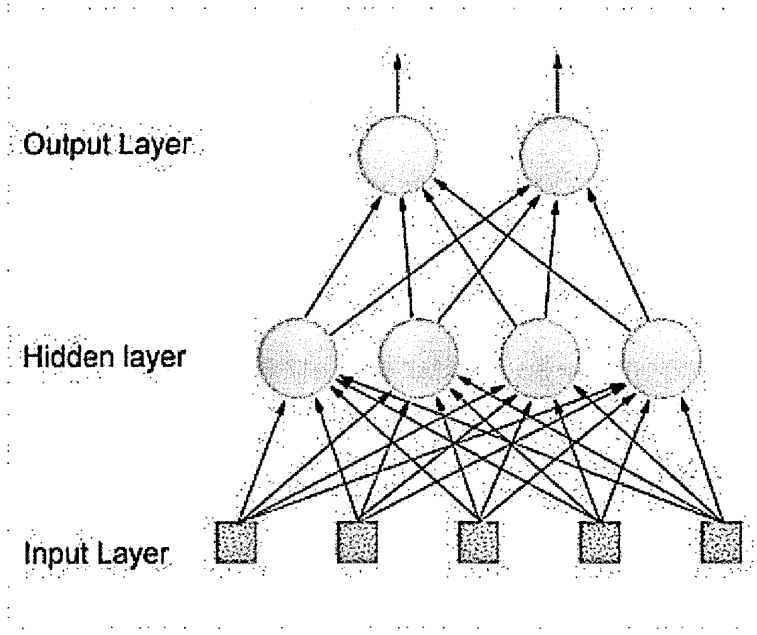

Fig. 1.6. Input nodes are fed into weighted connections to the hidden layer which sends the summed output to the output layer.

The basic principle of the neural network is the iterative modification of the weights for a set of training data to produce the required outcome. The weights may be set to random values at first. Multi-layer networks can use a variety of training techniques, the most popular being back-propagation. In this case the output values are compared with the correct answer to compute the value of some pre-defined error function; the required output value from the positive training set should approach 1 and conversely should tend to 0 from the negative learning set. The error is fed back through the network by one of various techniques and the weights are then adjusted to reduce the error function value. The whole process is repeated to convergence. Ogura and Sato (2001) use 1600 or 1024 input nodes (depending on the particle size), 81 hidden nodes and 1 output node. Input nodes consist of individual pixel densities and the hidden layer receives weighted density values added to the weighted image average, corrected by a bias factor. The number of neurons depends on the

particle size. Training was carried out on 1600 images. These were created from 200 particle images to produce the positive learning data and 200 noise images which constituted the negative data. Each of the particle and the noise images was rotated by 90, 180 and 270 degrees to provide the complete training set; more than 20 cycles were required to complete the training. Images were subjected to considerable pre-processing before being entered into the network and the training operation was extremely time-consuming. However, the authors claim a considerable superiority in accuracy in a comparison with correlation methods. A later improvement on their neural network method uses eigenimages as a recognition filter when setting up weights for the hidden layer (Ogura and Sato, 2004); eigenimages are calculated as part of the principal component analysis step, which is carried out during classification in a single particle analysis. When they also decreased the rotation increment of the training set to 2 degrees, the time taken to train the network was reduced by more than 50% and the pickup accuracy increased from 90% to 98%. Even so, the training time is very heavy and the necessity of a pre-determined model to produce the eigenimages is a major disadvantage.

The learning based method adopted by Mallick, Zhu and Kriegman (2004) is not strictly a neural network, but uses the idea of a training set of true and false particle images to select particles from boxed sub-images in scanned micrographs. Five different types of rectangular feature (Figure 1.7) are generated.

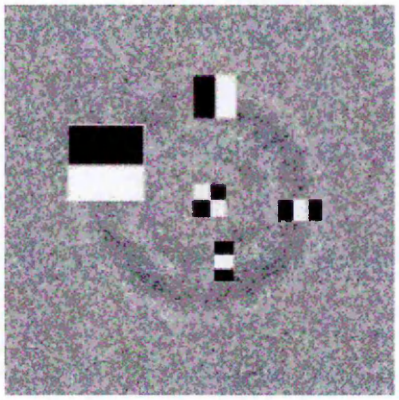

Fig. 1.7. Five rectangular features are used in detection and considered over a range of scales and at all locations.

Reprinted by kind permission of S. Mallick (Mallick et al., 2004).

Features are selected which give the lowest error during the training stage and become known as weak classifiers; a linear combination of features provides a strong classifier. All training images are initially given identical weights which are increased if the images are classified incorrectly in order to weight difficult images more heavily when the next feature is selected. A cascade of classifiers (Figure 1.8) is used as a filter where each classifier is composed of a few featu

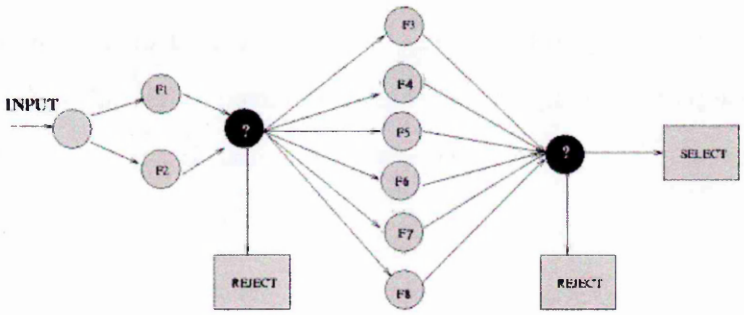

Fig. 1.8. A two stage cascade of classifiers.

Reprinted by kind permission of S. Mallick (Mallick et al., 2004).

The advantage to this approach is that many sub-images can be rejected at an early stage and thus reduce the computation time. The number of features in a classifier increases with selection. This method is fast, but only one type of particle is reported, the Keyhole Limpet Haemocyanin (KLH) used in the "bake-off" at the Multidisciplinary Workshop on Automatic Particle Selection for CryoEM (Zhu et al., 2004) and it does require post-processing. This is necessary to select a single position from a series of overlapping sub-images which represent the same particle. For this they use connected component analysis, taking the mean of each component as the particle position. Furthermore, a very large training set is required; in the case reported 1200 manually selected particle images and 3100 non-particle images were needed.

1.3.5 Texture based and other methods

Use of the variance image to detect the presence of particles with the same average density as the background was proposed by Van Heel (1982); local variances are computed over a small area for each pixel position. Although a high variance value indicates the presence of an object, it does not distinguish true particles from artifacts or aggregates. Lata, Penczek and Frank (1995) convolute with a Gaussian before a peak search is applied. Maxima are determined from areas corresponding to the particle size and are then thresholded. Training requires a) user-selected particles, b) noise areas and c) "junk" from which the standard statistical moments variance, skewness, kurtosis and also an estimate of the particle area are determined for pixel densities x_{ij} :

$$Variance = \sum_{i=1}^{N}\sum_{j=1}^{N}(x_{i,j} - \bar{x})^2,$$

$$Skewness = \sum_{i=1}^{N}\sum_{j=1}^{N}(x_{i,j} - \bar{x})^3,$$

$$Kurtosis = \sum_{i=1}^{N}\sum_{j=1}^{N}(x_{i,j} - \bar{x})^4 - 3,$$

where

$$\bar{x} = 1/N^2 \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}$$

and N^2 is the number of pixels in the box. *Note : The coefficient of kurtosis of the Normal distribution is 3 (Evans, Hastings and Peacock; 1993); the -3 in the formula corrects the value to zero.*

Entropy was also calculated as :

$$Entropy = -\sum_{i=1}^{N}\sum_{j=1}^{N} f_{i,j} log_2 f_{i,j},$$

where

$$f_{i,j} = x_{i,j} / \left(N^2 \bar{x}\right).$$

These values are input as feature vectors to a linear maximum-likelihood discriminant analysis. The function indicates the presence of true or false particles at the peak positions. The success rate was low, at around 60%, and the process also required considerable user intervention.

The binary segmentation algorithm described by Adiga et al. (2004) thresholds an image which has first been de-noised by anisotropic diffusion methods. Their two-step procedure first amplitude-thresholds the de-noised image then carries out connected component labelling. This is followed by thresholding the connected components to produce a bi-level map. Further processing includes morphological opening and closing operations to remove very small isolated artifacts and holes within particles (Figure 1.9).

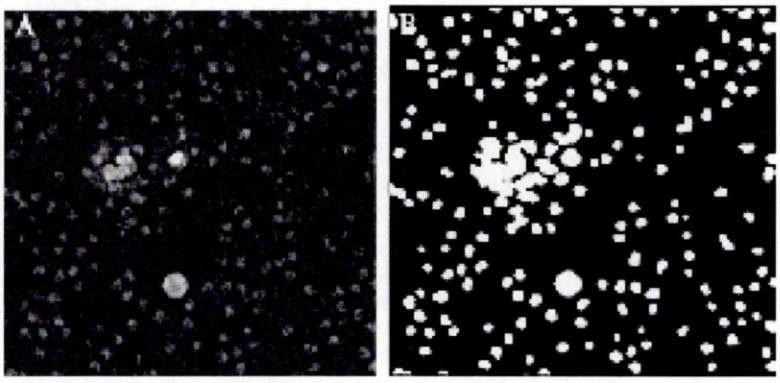

Fig. 1.9. (A) Part of the pre-processed micrograph image. (B) After thresholding and noise removal by morphological filters.

Reprinted by kind permission of R. Glaeser (Adiga et al., 2004).

Individual particles are distinguished from clusters by testing relative size and average density. Remaining clusters are subjected to erosion and dilation operations to separate individual particles which are then filtered according to their relative size. Clusters which still remain are further segmented by a region growing operation over a distance map. To search for missed particles, located positions in the original image are patched with background and the entire procedure is repeated. At least 80% success is claimed for this method which may well be highly specific; only one type of particle (the ribosome) was tested. The algorithm relies heavily on thresholds at several stages, which require tuning independently.

Plaisier et al. (2004) describe a three-step strategy for selecting particle positions : search, sort and select. The search step offers three different methods. The first involves local averaging by computing the pixel density averages inside a disc and in the surrounding band. This is carried out in Fourier space by calculating the convolution of the image with a binary image of a disc

using the convolution theorem, which is followed by a peak searching step. Their second method is template matching by cross-correlation, also followed by a peak search step. The third method calculates local variance; this is based on the assumption that areas of micrographs which contain particles will have a higher local variance than areas of background. The local variance is calculated for an area A at point $\vec{r}$ by :

$$Var_A(\vec{r}) = \frac{1}{N}\sum_{n=1}^{N} I_n^2(\vec{r}) - \frac{1}{N^2}\left(\sum_{n=1}^{N} I_n(\vec{r})\right)^2$$

where N is the number of pixels inside area A and $I_n(\vec{r})$ is the measured density at $\vec{r}_n$.

The sorting phase ranks the selected particle positions by cross-correlating each candidate with the template image and using simple statistical measurements. Final selection is a manual step carried out by the user from the set of sorted images. This method cannot be said to be fully automatic. Furthermore, in the case of the cross-correlation search method, the template is generated by a rotationally averaged image which then restricts the method to spherical or near-spherical particles.

1.4 Summary and comparison of methods

A comprehensive review of currently available methods was reported by Nicholson and Glaeser (2001), which concluded that the problem of automatic particle detection had not been successfully overcome by any of them. Since that time, according to the literature, limited progress appears to have been made. In order to assess the available software in a quantitative way, a Multidisciplinary Workshop on Automatic Particle Selection for Cryo Electron Microscopy was held at the Scripps Institute in 2003 (Zhu et al., 2004). At the workshop, participants were given the opportunity to compare the ac-

curacy of their methods. Twelve groups submitted the results of their own algorithms which were tested on a common dataset. The data consisted of 82 defocus pairs of high magnification micrographs containing the barrel-shaped Keyhole Limpet Haemocyanin particles in ice (Figure 1.10).

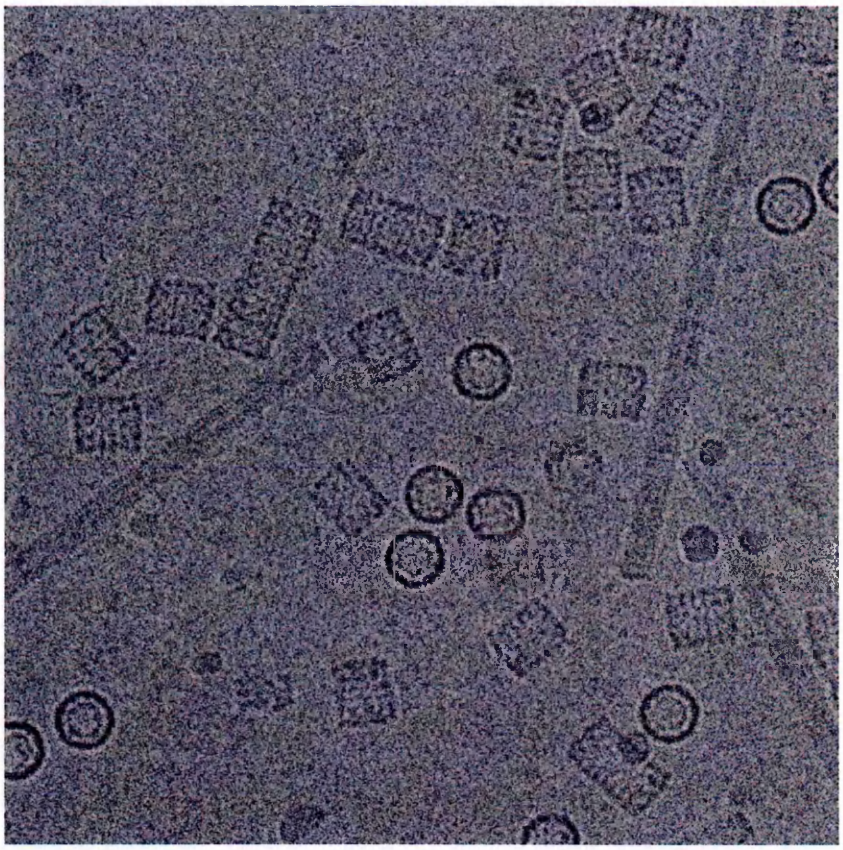

Fig. 1.10. A micrograph image from the dataset containing side, end and tilted views of Keyhole Limpet Haemocyanin particles and Tobacco Mosaic Virus particles.

Rectangular side views only were to be selected from the images. The "confusion matrix" (Figure 1.11) shows the results, which were assessed by comparing each of the participants against each of the others and were measured by the false negative rate (FNR) and false positive rate (FPR). One participant's result was taken as the truth set and each of the others in turn was taken as

the test set. Particles selected in the truth set but not in the test set became false negatives; those in the test set but not the truth set were marked as false positives. Algorithms which achieved both a low FNR and FPR were therefore considered desirable.

	Test											
Truth	Bajaj	Bern	Mouche(M)	Haas(M)	Hall	Ludtke	Mallick	Penczek	Roseman	Sigworth	Volkmann	Zhu
Bajaj(1269)		33.9	24.7	31.0	42.2	51.9	28.0	52.9	17.4	37.4	38.5	24.0
		11.5	8.3	7.0	24.3	21.0	9.8	25.2	14.0	5.1	9.2	11.4
Bern(948)	11.5		16.2	21.5	36.3	43.1	17.7	48.4	10.3	26.4	29.9	17.1
	33.9		23.8	21.0	37.7	30.3	23.1	38.8	30.3	16.7	22.8	28.0
Mouche(1042)	8.3	23.8		11.7	27.4	43.4	14.2	46.8	2.4	23.2	27.4	9.7
	24.7	16.2		2.3	22.0	23.7	11.7	30.7	16.6	4.5	12.2	13.7
Haas(944)	7.0	21.0	2.3		26.2	41.1	12.2	44.0	1.5	18.4	22.9	8.8
	31.0	21.5	11.7		28.2	28.4	18.4	33.9	23.9	8.4	15.7	21.3
Hall(969)	24.3	37.7	22.0	28.2		52.0	30.1	55.9	19.3	35.3	39.3	25.7
	42.2	36.3	27.4	26.2		39.9	33.2	46.6	35.8	25.2	31.7	33.7
Ludtke(775)	21.0	30.3	23.7	28.4	39.9		23.0	48.3	20.3	27.1	32.3	23.5
	51.9	43.4	43.4	41.1	52.0		41.2	50.0	49.4	32.7	39.1	45.4
Mallick(1015)	9.8	23.1	11.7	18.4	33.2	41.2		46.7	7.0	25.8	30.1	14.5
	28.0	17.7	14.2	12.2	30.1	23.0		32.5	22.6	10.3	17.9	20.5
Penczek(799)	25.2	38.8	30.7	33.9	46.6	50.0	32.5		23.7	38.4	39.7	30.2
	52.9	48.4	46.8	44.0	55.9	48.3	46.7		50.0	41.3	44.0	49.1
Roseman(1219)	14.0	30.3	16.6	23.9	35.8	49.4	22.6	50.0		33.1	34.9	17.5
	17.4	10.3	2.4	1.5	19.3	20.3	7.0	23.7		2.7	7.8	7.8
Sigworth(838)	5.1	16.7	4.5	8.4	25.2	32.7	10.3	41.3	2.7		12.3	6.8
	37.4	26.4	23.2	18.4	35.3	27.1	25.8	38.4	33.1		14.6	28.1
Volkmann(861)	9.2	22.8	12.2	15.7	31.7	39.1	17.9	44.0	7.8	14.6		11.5
	38.5	29.9	27.4	22.9	39.3	32.3	30.1	39.7	34.9	12.3		30.0
Zhu(1109)	11.4	28.0	13.7	21.3	33.7	45.4	20.5	49.1	7.8	28.1	30.0	
	24.0	17.1	9.7	8.8	25.7	23.5	14.5	30.2	17.5	6.8	11.5	
Median/Mean												
FNR	11.4/13.1	28.0/27.9	16.2/16.2	21.5/22.0	43.4/44.5	33.7/34.4	20.5/20.8	48.3/47.9	7.8/10.9	27.1/28.0	30.1/30.7	17.1/1
FPR	33.9/34.7	21.5/25.3	23.2/21.7	18.4/18.7	27.1/28.9	30.1/33.6	23.1/23.8	33.9/35.4	30.3/29.8	10.3/15.1	15.7/20.6	28.0/2
Standard Deviation.												
FNR	7.0	7.1	8.6	8.0	6.0	6.7	7.3	4.1	7.9	7.7	8.0	7.8
FPR	11.3	12.8	14.2	14.4	8.6	11.9	13.0	8.2	12.4	12.7	12.4	13.2

Fig. 1.11. The two values in each table cell represent false negative rates (FNR) and false positive rates (FPR) respectively, as percentages. FNR values are positioned in the upper row in the top right diagonal, and in the lower row in the bottom left diagonal. Numbers in parentheses represent the total number of particles selected by the corresponding participant.

It is interesting to note that the two manually picked particle sets (denoted by (M) - Mouche and Haas) differ significantly from each other, which clearly shows that different individuals apply different selection criteria. Several of the algorithms tested selected end views along with the desired side views and so scored higher FPR values demonstrating their inability to distinguish between different views. Although this was a useful experiment, it was somewhat limited in that only a single view of one type of particle was involved. Furthermore, none of the images were negatively stained and labels and unwanted areas of carbon and unexposed film which are present on film were not included; the images were recorded by a CCD device.

Half of the participants used correlation-based template matching methods while the remainder were composed of a variety of feature-based techniques (Table 1.12); neural networks were not represented in the "bake-off".

Bajaj	Feature based. Edge detection (Canny). Voronoi diagram detects rectangles, Distance transform detects circles.
Bern	Template matching. Templates : 3D model projections. Peak filtering by probabilistic model derived from particle images and noise.
Mouche	Manual selection.
Haas	Manual selection.
Hall	Feature based. Convolution with Gaussian using CCF. Distance and peak height constraints, Feature vector matching
Ludtke	Template matching. Templates : aligned images, Peak filtering manually set threshold.
Mallick	Feature based. Training images. Discriminative learning from sub-images.
Penczek	Template matching. Templates : 3D model. Noise power spectrum, CTF, normalization of image, FT x template FT. Filter CC threshold.
Roseman	Template matching. Templates : aligned images. Peak filter correlation coefficient.
Sigworth	Template matching. Templates : 3D model. Noise whitening, Peak filter maximum correlation and weighted sum of power spectrum.
Volkmann	Feature based. Reduced representation template. Real space comparison with image, Distance filter, Outlier screen.
Zhu	Feature based. Edge detection (Canny). Edge connection(Hough transform). Correlation based template removes false positives.

Table. 1.12. This table briefly indicates the algorithms used by the 12 participants.

1.5 Aims of the present work

The human eye and brain locate objects in a noisy background astonishingly quickly and accurately. Background subtraction, integration, smoothing, thresholding, size and shape matching are all essential parts of the recognition process and can be used to select particle image positions in a digitized electron micrograph.

By definition, an ideal automatic particle detection procedure should require

little or no user intervention. However, some initial preparation of reference criteria is inevitable but should be kept to a minimum. This work aims to design and implement software which will include a graphics tool to allow straightforward parameter setting and which will also possess the ability to process an unlimited number of micrographs in a completely automatic way.

Since reference free systems such as edge detection methods must inevitably be unable to distinguish true particles from artifacts of similar size, the algorithm described here uses reference criteria to act as a guide to the detection of real particles. The criteria are derived from a small stack of manually selected boxed particle images which eliminates the time-consuming process of calculating a three-dimensional model to provide template projections.

Enhancement techniques to remove both low and high frequency noise can improve the performance of particle detection considerably. To avoid lengthy computational techniques (such as anisotropic diffusion) for this process, the fast and simple methods of local averaging and high-pass spatial filtering are used, and are effective in removing both high and low frequency noise components. Histogram stretching is a very useful strategy for standardizing image density ranges; through this technique the problem of labels, carbon and unexposed areas of film is taken into account in a totally automatic way. This eradicates the laborious step of selecting areas manually from hundreds of micrographs.

Since this work aims to be able to detect particles of any shape without using lengthy rotations, it is based on matching the radius of gyration. This parameter value is averaged from the set of boxed reference particle images

and compared with the corresponding value from each box of pixel densities in turn in the digitized image. However, the radius of gyration is an insufficient match when used in isolation, and is complemented by other simple properties which are based on a filtering approach which minimises the computation time. Finally, a clustering technique selects the best of several overlapping windows which represent the same particle.

My approach provides a simple and fast method for the automatic selection of a wide variety of specimens from electron microcope images.

Chapter 2

Image preparation

The presence of high and low frequency noise is inevitable in digitized images of biological structures both for negatively stained and for ice-embedded specimens. For the majority of algorithms, such noise presents serious difficulties for automatic particle detection; it is normally necessary to correct for uneven illumination and to reduce the shot noise. In the case of micrograph images, high frequency noise due to the presence of film grain should also be addressed. Furthermore, labels and unexposed areas of film invariably create problems and should also be accounted for in a fully automated system.

Noise cleaning methods based on anisotropic diffusion (Boier Martin et al., 1997; Nicholson and Malladi, 2004; Singh, Marinescu and Baker, 2004; Yu and Bajaj, 2004) appear to work quite effectively, but are currently too expensive in terms of computing time to be considered as a part of my project. However, a variety of other strategies for image enhancement were investigated in order to select the most appropriate method for this work (Pratt, 1991; Gonzalez and Woods, 1992).

2.1 Density inversion

Depending on the algorithm, density inversion is appropriate where the average background density exceeds that of the particles. Ice-embedded specimens normally require inversion while negatively stained images do not.

2.2 Image compression

Image compression by local pixel density averaging within a square box is effective in reducing noise while at the same time minimizing processing time and memory requirements. However, too large a compression may also adversely affect the accuracy of particle selection as important detail may be lost. The compression factor used by this algorithm is calculated as a function of the user-specified particle radius R. It is determined such that the working radius in the compressed image lies in the range 10-15 pixels, which has been found to work well for most of the images tested, although this value may be overridden by the user.

2.3 Noise filters

2.3.1 Fourier filtration

Fourier bandpass filtration is a well known method of removing both low and high frequency components and includes the property that the frequency cutoff thresholds can be controlled precisely. In this technique, a forward Fourier transform is first calculated from the digitized image. Frequencies outside the cutoff thresholds are removed and the modified Fourier array is then back-transformed to produce a de-noised image. It is important that the cutoff thresholds are smoothed in order to prevent unwanted aliasing effects. There are many appropriate functions which can be applied for this purpose;

a computer program called "BANDPASS" was written as part of this project to carry out bandpass filtration. The program offers a choice of three different cutoff distributions : Gaussian, Cosine bell and Cauchy. It includes parameter value tuning to allow various adjustments to the profile shape (Figure 2.1).

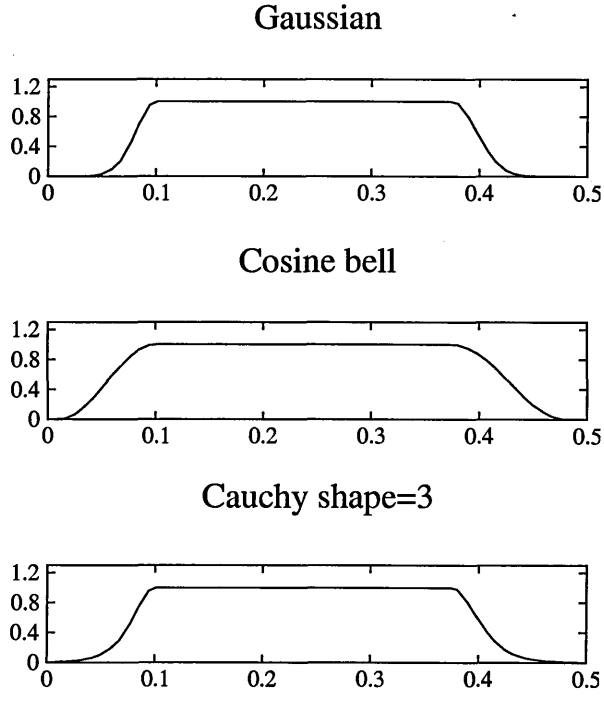

Fig. 2.1 Three distributions used as threshold cutoffs for low and high frequency data. Horizontal units relate to the box size of the realspace image; vertical units indicate the multiplication factor used in Fourier space. The top graph shows the effect of using a Gaussian distribution : $\frac{1}{\pi} e^{-\left(\frac{x-\mu}{\sigma}\right)^2}$. The wider slope of the cosine bell function displayed in the centre graph gives a smoother cutoff : $0.5\left(\cos\left(x\right)+1\right)$, where x is the distance along the horizontal axis. The bottom graph demonstrates the Cauchy function applied to the cutoffs : $\left(1+\left(\frac{x}{a}\right)^2\right)^{-m}$ where x is the distance, a a scale factor and m a shape parameter. The value of m can be varied to modify the shape of the function by controlling the tail length and is set to 1.0 in this case. Increasing the value of m results in a sharper peak.

Results from the three types of cutoff function are shown in Figure 2.2.

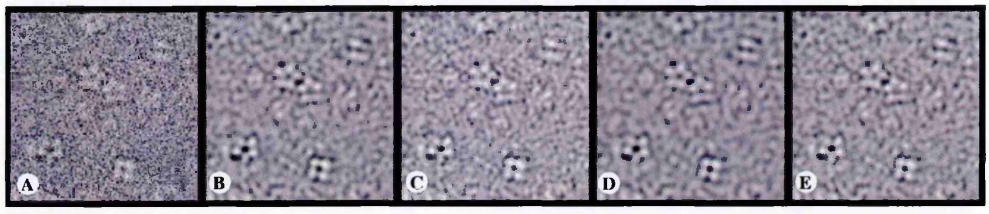

Fig. 2.2 A) Part of a raw image of Ketopantoate hydroxymethyl transferase (KHMT) particles embedded in ice. Bandpass Fourier filtration on this image is shown between cutoff thresholds at 0.05 and 0.15 using B) Gaussian cutoffs C) Cosine Bell D) Cauchy with $m = 1$ and E) Cauchy with shape parameter $m = 5$.

While bandpass filtration provides an effective de-noising technique, it has major disadvantages : specific size constraints and processing time (see Chapter 1). For this project it was abandoned in favour of realspace methods.

2.3.2 Realspace high frequency filtration

Median filtration

Median filters can be used to achieve noise reduction without blurring the image, retaining sharp edges effectively. Each pixel density is replaced by the median density of pixels in its immediate neighbourhood. Cascading the filtering by repeating the method on a treated image improves the noise reduction further, as does increasing the number of pixels from which the median is calculated. However, the technique is computationally expensive; the number of operations grows exponentially with the window size. The median of a five element sequence (a, b, c, d, e) of pixel densities can be expressed as :

$$MED\,(a,b,c,d,e) = max\Big(min\,(a,b,c)\,, min\,(a,b,d)\,, min\,(a,b,e)\,,$$
$$min\,(a,c,d)\,, min\,(a,c,e)\,, min\,(a,d,e)\,,$$

$$min\,(b,c,d)\,,min\,(b,c,e)\,,min\,(b,d,e)\,,min\,(c,d,e)\,\Big)$$

Pseudomedian filtration

The pseudomedian filter (Pratt, 1991) retains some of the properties of the median filter and is simpler and faster to compute :

$$PMED\,(a,b,c,d,e) = \left(\frac{1}{2}\right) max\Big(min\,(a,b,c)\,,min\,(b,c,d)\,,min\,(c,d,e)\,\Big) + \left(\frac{1}{2}\right) min\Big(max\,(a,b,c)\,,max\,(b,c,d)\,,max\,(c,d,e)\,\Big)$$

Maximin/minimax filtration

The maximin and minimax operators (Pratt, 1991) used in the pseudomedian filter can be cascaded to provide a further de-noising technique :

$$MAXIMIN\{S_L\} = max\Big\{\,[min\,(s_1,...,s_M)]\,,[min\,(s_2,...,s_{M+1})]\,,...,\,[min\,(s_{L-M+1},...,s_M)]\,\Big\}$$

$$MINIMAX\{S_L\} = min\Big\{\,[max\,(s_1,...,s_M)]\,,[max\,(s_2,...,s_{M+1})]\,,...,\,[max\,(s_{L-M+1},...,s_M)]\,\Big\}$$

where $\{S_L\}$ is a sequence of pixel densities $s_1, s_2, ..., s_L$ and $M = \frac{(L+1)}{2}$

Outlier replacement

Outlier replacement is a simple noise cleaning technique in which each pixel density is compared to the average of its immediate neighbours.

$$\mathbf{H} = \frac{1}{8}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Where the density difference exceeds some threshold, it is replaced by the neighbourhood average :

$$if \left[d - \frac{1}{8} \sum_{i=1}^{8} d_i \right] > E \quad then \quad d = \frac{1}{8} \sum_{i=1}^{8} d_i$$

where E is the threshold and d is the pixel density to be modified.

Spatial averaging

Spatial averaging is fast, simple and very effective in reducing high frequency noise. The amount of blurring can be controlled by the window size used to calculate the average.

The effects of these methods of real space high frequency noise cleaning on an area of a typical image are demonstrated in Figure 2.3. While the compute-intensive median filter has achieved a considerable reduction of high frequency noise, the faster pseudomedian filter was far less effective. In the case of this particular image, the smoothing effect of the cascaded minimax/maximin operator appears to have enhanced the noise more than the particles and it has become more difficult to distinguish them from the background. The outlier replacement technique goes some way to reduce high frequency noise, but not as efficiently as the very simple and effective spatial averaging, which is the method which was selected as most suitable for this work, although it always benefits from a further step of contrast enhancement which is discussed in a later section in this Chapter.

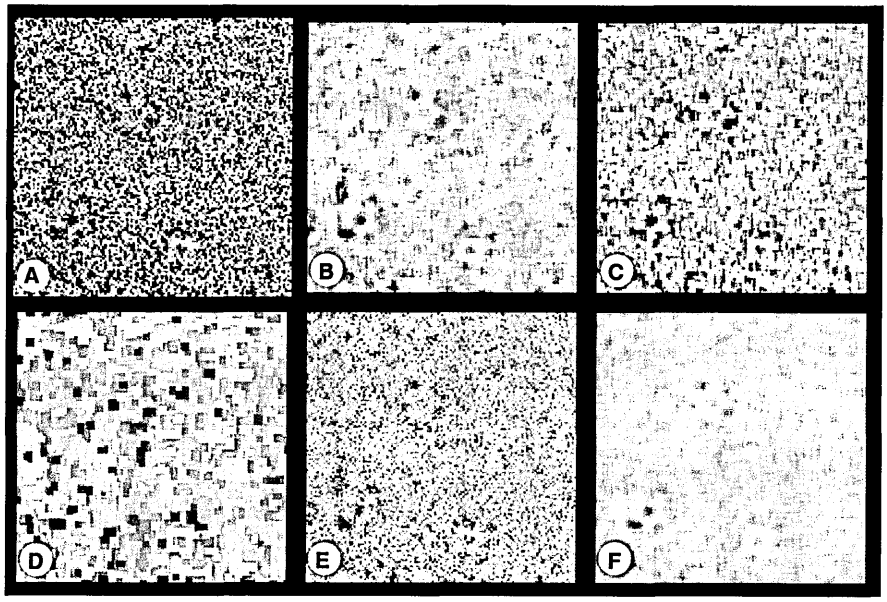

Fig. 2.3 Results of de-noising KHMT particles in ice from a 128 x 128 boxed image. A) Raw image. B) Median filtering using a 5 x 5 box. The pseudomedian filter shown in C) has had a lesser effect even though the same sized window was applied. D) The cascaded minimax/maximin operator. E) Outlier replacement in a 9 x 9 box and F) spatial averaging with a box size of 5 x 5.

2.3.3 Realspace low frequency filtration

Shading effects due to uneven illumination or to varying thicknesses of ice or stain were found to cause major difficulties with automatic particle detection. They can be removed by excluding low frequency components in Fourier space by bandpass filtration. They can also be removed very simply and effectively by high pass spatial filtering without the necessity of Fourier transform calculation (Gonzalez and Woods, 1992). A mask is applied to each N x N box of pixel densities. The mask has positive coefficients near its centre, and negative coefficients elsewhere such that their sum is zero (Figure 2.4) :

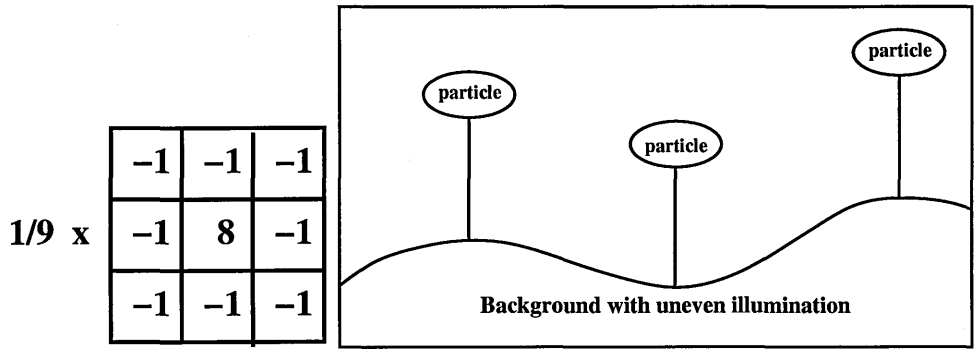

Fig. 2.4 A simple high pass filter mask is shown on the left. This filter works by subtracting local densities from each pixel. As shown in the diagram on the right the height of particle density above the uneven background remains fairly constant so the uneven background can be flattened by the filter.

This filter is used in the work described here and the results are demonstrated in Figure 2.5.

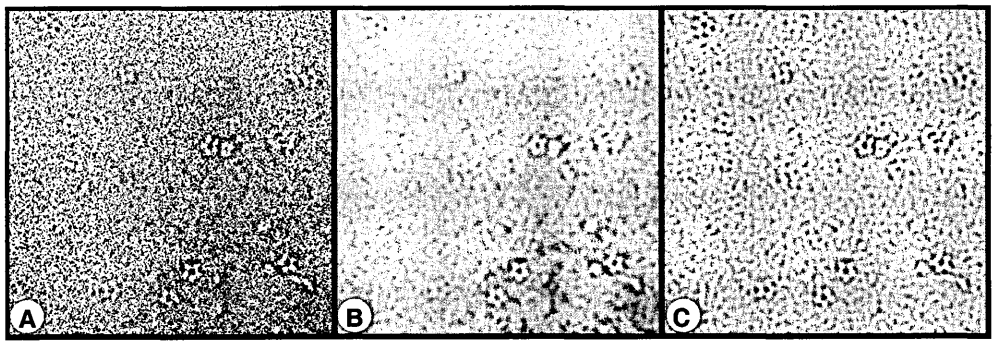

Fig. 2.5 A) Raw image of KHMT particles in ice, B) after low pass spatial averaging to reduce high frequency noise and C) followed by high pass spatial averaging to eliminate uneven illumination.

2.4 Contrast modification functions

In order to enhance the contrast of the image, thereby improving the ability of the software to recognise true particles, particularly when their mean density is barely greater than that of the background, a contrast modification function is applied to the image. The function should stretch the contrast in such a way as to increase the highest pixel densities and decrease the lowest. Six appropriate such functions were investigated as part of this work and the results illustrated in Figure 2.6.

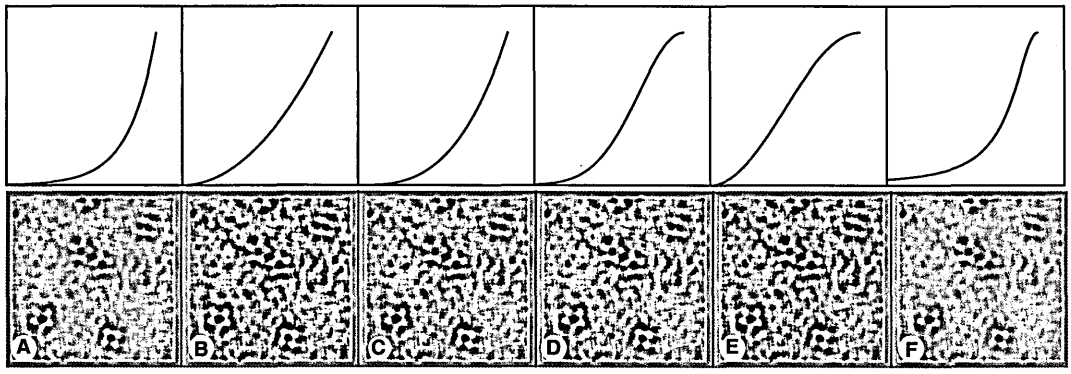

Fig. 2.6 Each graphical representation at the top corresponds to the image immediately below, and demonstrates the results of applying its contrast modification function to the smoothed version of the raw image shown in Figure 2.3 ; in all cases x is the density of each pixel in the image array.

A) exponential : e^x ,

B) square : x^2 ,

C) cube : x^3 ,

D) Gaussian : $\frac{1}{\pi}\, e^{-\left(\frac{x-\mu}{\sigma}\right)^2}$, where μ is the mean and σ the standard deviation,

E) cosine bell : $0.5\,(\cos(x) + 1)$, and

F) the standard form of Cauchy : $\pi\, b \left(1 + \left(\frac{x-a}{b}\right)^2\right)^{-1}$, where a is the median and b is the scale parameter where $b > 0$.

In practice, little difference between the contrast modification functions was observed in terms of the overall particle detection result; the square function (B) was selected as being computationally economical and is included in the image enhancement part of the program.

2.5 Histogram modification

Histogram modification strategies provide other useful methods of image enhancement (Nicholson and Malladi, 2004; Wong et. al., 2004). In one such technique the histogram is used to exclude those pixel densities which fall at the extremities of the range. The remaining densities can then be stretched across the range, thus increasing the contrast. In this work, upper and lower threshold cutoffs are applied for the density modification step. It is assumed that each side of the histogram profile is normally distributed, but with different standard deviations. Each side is therefore thresholded independently of the other. The cutoff range is defined :

$$\mu - m\sigma_l, \mu + m\sigma_r$$

where μ is measured at the histogram maximum (the mode)

σ_l is the halfwidth at halfheight of the left hand side of the peak

σ_r is the halfwidth at halfheight of the right hand side of the peak

and m is a user-specified value typically in the range 0.25 - 5. 0

Images with areas of carbon around ice-filled holes can present a particular problem when the user wishes to exclude particles from such areas. An image of the carbon is necessary for the accurate determination of the contrast

transfer function, the application of which is itself essential to biological structure determination by electron cryomicroscopy to high resolution. However, it remains a problem for particle detection and although it is possible to remove these areas of carbon by hand, this project aims to automate this process. Such images are specially treated by the program to exclude the carbon by using only one side of the peak to set both cutoff thresholds (Figure 2.7). This enhances only the contrast of the area to be searched for particles.

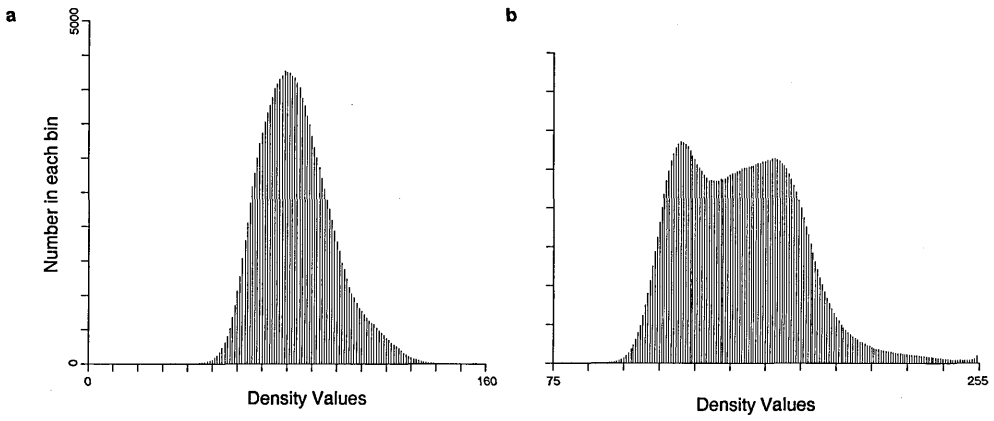

Fig. 2.7 Examples of image histograms from cryo-micrographs with and without areas of thicker carbon. (a) Histogram of an image with no carbon showing a single peak only. (b) Histogram of an image with a large area of carbon showing two overlapping peaks. This cryo-image has not been inverted, therefore the area of ice containing the particles has the higher average value and occupies the right hand peak. The right hand side of this peak is sampled for calculating thresholds for both sides.

2.6 Label masking

A further difficulty in automatic particle detection is presented by the labels written by the microscope along with unexposed areas of the image at the edges. A technique was developed for this work which totally excludes such undesirable regions of the image from particle detection, thus saving manual intervention and processing time by automatically eliminating the possibility of selecting particles from these areas.

A histogram of densities is first calculated; particle-containing densities are included in the large central peak as shown in Figure 2.7 a). Pixel densities found to be at the extremities of the range, either minimum or maximum, are assumed to comprise areas to be excluded from particle selection; unexposed areas and labels will be close to the maximum extremity, and writing on the label is close to, or at, the minimum.

Regions of carbon surrounding ice-filled holes can be detected in the histogram adjacent to the particle-containing density, as shown in Figure 2.7 b). Areas of carbon can be bypassed when particle searching, by referring to a binary mask map. This two-dimensional array, which maps the micrograph image, is calculated from the histogram which has been modified by a user-selected cutoff value to exclude regions to be ignored. The mask is used to bypass any window containing a correspondingly flagged pixel position.

Since it is frequently the case that a few isolated pixels are incorrectly flagged, the entire binary array is further processed by the majority black operator (Pratt, 1991) to rectify the problem. This procedure is useful for removing small spikes (or holes) : pixels in the binary array are set to 1 if four or more

adjacent neighbours are set to 1. Regions of the image indicated by the binary array are ignored by the particle search (Figure 2.8).

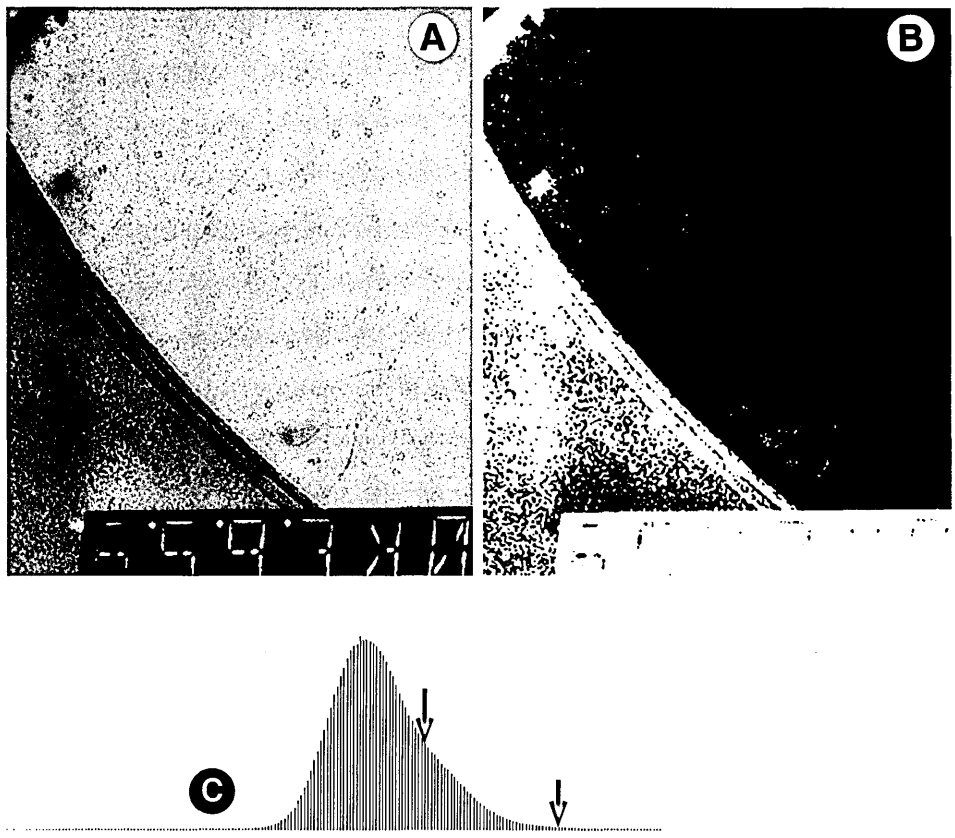

Fig. 2.8 A) Raw image of KHMT particles in ice, which includes part of a label and an area of thick carbon to be excluded from searching on the left. B) The binary mask map computed from the inverted raw image. C) Histogram of the inverted image. The region lying between the red arrows indicates the contribution of the carbon to the histogram. This area is ignored by the program which uses the user-specified density histogram cutoffs in calculating the binary map. Any window containing a corresponding pixel in the mask which is flagged for exclusion will not be searched.

Chapter 3

The Selection Procedure

Moving across the micrograph image pixel by pixel, square windows of densities are extracted each in turn. Every window is considered as a potential particle, being subjected to a series of tests which match parameter values extracted from it to corresponding values pre-determined from a set of user-selected reference images. The aim is to distinguish windows containing true particles from those which do not. Tests are ranked in such a way as to minimise computing time, eliminating failed windows from any further examination as the tests are executed. In practice, the matching procedure results in multiple sets of overlapping windows, where each set represents an individual particle. A clustering algorithm is used to re-arrange these windows according to their proximity to each other. Finally, from each cluster of overlapping windows, a scoring procedure is used to select the window in which the particle is judged to be centred most accurately.

Since a large part of a digitized image is likely to comprise background and other totally unsuitable areas, such as particle aggregates, computing time is saved by firstly eliminating these regions. Labels and unexposed areas of the image are flagged by a pre-processing step previously described in Chapter

2, and hence are ignored. Furthermore, in a single particle analysis it is important that particle images are isolated from neighbouring particles or undesirable artifacts, since such encroachments can affect particle integrity; particles immediately adjacent to any other material are excluded as the next step. Remaining candidate windows should contain only isolated objects of a size which approximates that of the desired particle. These windows are then subjected to further examination for shape and density distribution.

A simple and convenient measure of density distribution, independent of orientation, is provided by the radius of gyration. However, while it gives a measure of radial distribution, it does not describe the angular distribution and so this fundamental property is insufficient to provide an adequate match for particle detection. In this work, the radius of gyration is therefore complemented by other properties which together provide a comprehensive set of criteria for matching true particles to the reference images. By using a set of increasingly sensitive matched filters, which examine different particle attributes, particle positions are detected both accurately and efficiently. The algorithm makes the assumption that the average particle density will be greater than that of the background and a circular mask set in the centre of each window is used to limit the area of interest by ignoring irrelevant corner regions.

3.1 Isolated object of appropriate size

The first two tests aim to exclude windows containing only background or noise artifacts and non-isolated particles immediately adjacent to other objects.

3.1.1 Ratio mean and variance test

Elimination of totally unsuitable windows, such as areas of background or particle aggregates, can be achieved by rejecting those with an inappropriate mean density ratio between a circular central area and that of the annular band immediately surrounding it. Furthermore, contributions from the contrast in a true particle cause its density variance to exceed that of background regions; density variance from the central area is also compared with that in the annular surrounding ring to complement the ratio mean test (Figure 3.1).

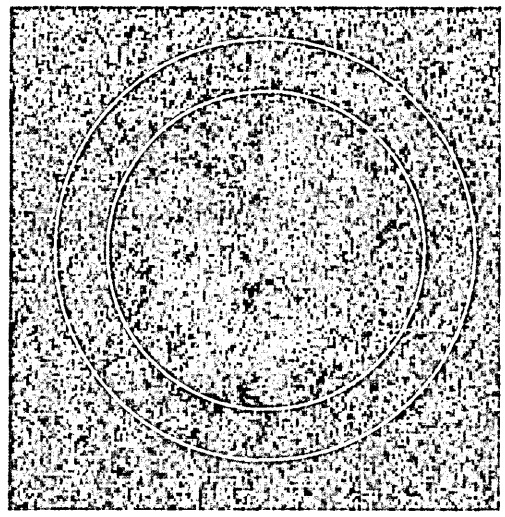

Fig. 3.1 Shows a ribosome particle in ice to illustrate the increased mean density and variance of the particle inside the inner ring compared to the corresponding values in the surrounding annular band.

The width of the annular ring is controlled by the user, who may wish to accept only those particles which are totally isolated and surrounded by a large area of background; alternatively they may choose to detect particles which almost touch. Both density mean and variance inside a circular mask of particle radius R are determined, along with those lying within the sur-

rounding annular band between the radii R and $Rmax$. $Rmax = a * R$, where a is a user-specified variable in the range $1.1 - 1.5$. The two ratios are then calculated :

$$Ratio_{\mu} = \mu_c / \mu_b$$

$$Ratio_{\sigma^2} = \sigma_c^2 / \sigma_b^2$$

where c is the central circular area of radius R

b the area in the annular band bounded by R and $Rmax$

μ_c the central mean, μ_b the band mean

σ_c^2 the central variance, σ_b^2 the band variance.

Typical ratio mean values for ribosomes such as that shown in Figure 3.1 range from 3.2 - 5.5; in particular the ribosome in the figure measured 4.3.

3.1.2 Adjacency test

Windows containing particles with density encroaching into the area surrounding the central area are detected in the following way. A circle radius $Rmax$, centred on the box centre, is divided into eight equal sectors. The mean density of each sector, bounded by radius R, is compared with that in the corresponding area in the adjacent surrounding annular band, bounded by R and $Rmax$. If, for any sector, the outer (annular) sector mean is the higher value, then it indicates that the particle is too close to a neighbour, and the window is therefore rejected (Figure 3.2).

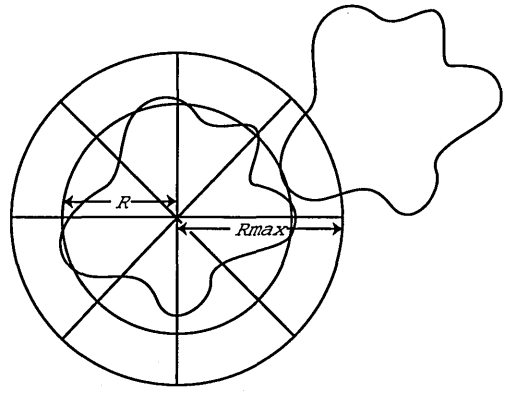

Fig. 3.2 Particles with neighbours encroaching in one or more sectors of the band around the particle image are detected by comparing the mean densities between each sector and its corresponding surrounding annular region.

3.2 Density distribution

Candidate windows not flagged as background or containing particles adjacent to others are examined for appropriate variance, radius of gyration and density sum. These values provide some basic information about the density inside the central circular area.

3.2.1 Density sum

This very simple measure is calculated for pixel densities lying inside a circle, of radius R appropriate to the particle and centred in the box centre, for comparison with the reference value. The set of density values within each window is scaled independently between 0 and 255 before calculating their sum. This is to allow for the detection of weak particles.

3.2.2 Variance

The density variance within a true particle normally differs significantly from that of the background and most noise artifacts. The density variance within

a circular mask radius R and centred on the box centre is calculated and compared with the reference value :

$$\sigma^2 = \frac{\sum_{n=1}^{N} (x_n - \mu)^2}{N}$$

where μ is the mean density within the mask, N is the total number of pixel densities within the mask and x_n is the pixel density at n.

3.2.3 Radius of Gyration

The radius of gyration is the second moment of inertia and provides a measure of the distribution of pixel densities as a function of their distance from the centre of a square box of pixels; its value increases with the distance of the density from the centre. It is affected by high pixel densities in the box corners or anywhere surrounding the particle. This undesirable data may be caused by neighbouring particles or background artifacts. To exclude it, window density values are radially tapered and scaled (Figure 3.3).

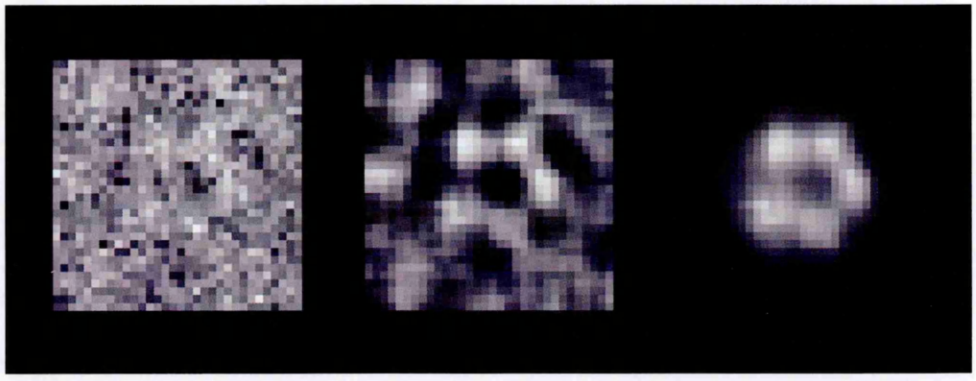

Fig. 3.3 This figure shows an image of a particle of KHMT at various stages of processing. The left hand image shows the raw particle. The central box shows the image after the noise cleaning operations (see Chapter 2). The right hand box shows the effects of the tapering process which removes extraneous material from the box edges, and provides a clearly enhanced particle image.

To prepare the image from which the radius of gyration is measured, pixel densities are first constrained to lie between :

$$\mu_c \pm 3\sigma_c$$

and scaled between 0 and 255. Tapering then takes place from the box centre to *Rmax* by weighting with an exponential function :

$$w = 1 - s\exp^{-t\left(\frac{Rmax-Rdist}{Rmax}\right)^2}$$

where *Rmax* is the particle radius extended to include the outer band

Rdist is the distance from the box centre ranging from 0, *Rmax*

s is an empirically derived scale factor, set to 0.9

t is an empirically derived taper factor, set to 10

and pixel densities beyond *Rmax* are set to 0.

The radius of gyration *Rgyr* is then calculated. The formula for this parameter demonstrates the necessity for the removal of the unwanted data at the extremities of the box, since I comprises the entire box; *Rgyr* is required only for the central region.

$$Rgyr = \sqrt{\frac{\sum_{i=1}^{I} m_i x_i^2}{M}}$$

where M is the total density over all pixels

m_i is the density of pixel i

x_i is the distance of pixel i from the box centre

and I is the total number of pixels

3.3 Circular and radial density distribution

Some undesirable artifacts such as ice contaminants may not be excluded by the previous tests; their size and density can sometimes closely match those of true particles and the radius of gyration cannot always be used to distinguish between totally different particle shapes (Figure 3.4).

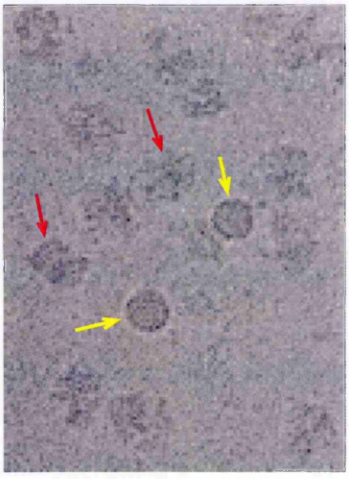

Fig. 3.4a Shows the similarity between ice artifacts (indicated by yellow arrows) and neighbouring true ribosome particles (indicated by red arrows).

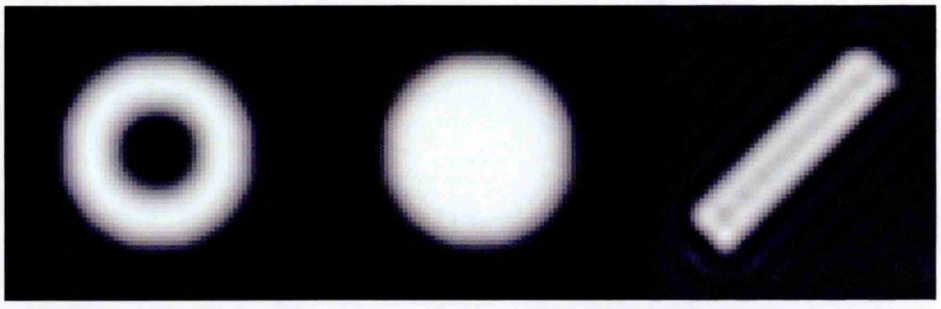

Fig. 3.4b The radius of gyration *Rgyr* for three different images demonstrates its strengths and weaknesses. *Rgyr* measures 16.0 for the left hand ring, 14.5 for the central disc and 14.5 for the rod. It successfully distinguishes the disc from the ring, which have identical maximum radii. However, although their shapes are totally different, it cannot distinguish the rod from the disc, which have identical *Rgyr* values.

In order to select only true particles from candidates which have survived these tests, more sensitive tests examine the density distribution in greater detail, using information from equally-spaced concentric rings and from circular sectors of pixel densities.

3.3.1 Ring parameter tests

Circularly averaged information about the particle shape is obtained from the means and variances of density values extracted from equally spaced concentric rings from tapered particle images. This test is particularly successful in distinguishing particles with strong features such as a central hole or cleft from particles or artifacts which do not (Figure 3.5).

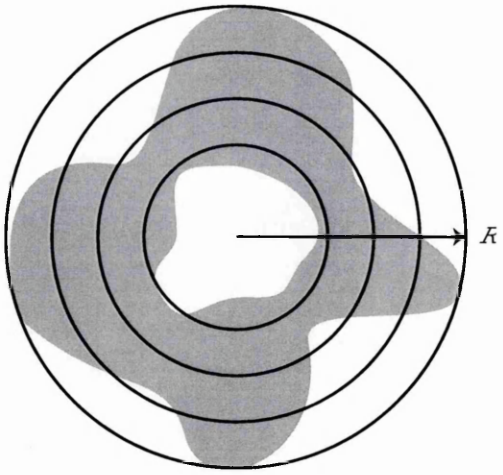

Fig. 3.5 Densities from concentric rings up to particle radius R and spaced apart by a distance of one pixel are sampled from the window images.

From the set of means and the set of variances calculated from pixel densities in the rings, profiles are extracted.

$$\mu_i = \frac{\sum_{j=1}^{J} \mathrm{d}_j^i}{\mathrm{J}} \qquad \sigma_i^2 = \frac{\sum_{j=1}^{J} \left(\mu_i - \mathrm{d}_j^i\right)^2}{\mathrm{J}}$$

where d_j^i is the pixel density, i is the ring number

and J is the number of densities in the ring

These profile values are compared with the corresponding reference profile values using the χ^2 statistic as a measure of goodness-of-fit:

$$\chi_\mu^2 = \sum_{i=1}^{I} \frac{(\mu_{refi} - \mu_i)^2}{\mu_i} \qquad \chi_\sigma^2 = \sum_{i=1}^{I} \frac{\left(\sigma_{refi}^2 - \sigma_i^2\right)^2}{\sigma_i^2}$$

where $refi$ is the reference value for ring i,

and I is the number of concentric rings

3.3.2 Sector parameter test

While the ring mean and variance tests reveal circularly averaged information about the particle shape, the sector test provides additional angular information from the mean pixel density of circular sectors. This test applies a rotation which could consume large amounts of computing time if it were applied to every window extracted from the digitized image. However, as it is the final match and applied solely to windows which have passed all the other tests, it is confined to the preferred candidates. Within a circle of radius R, the image is divided into 16 equal sectors, and the density for each sector is averaged (Fig. 3.6).

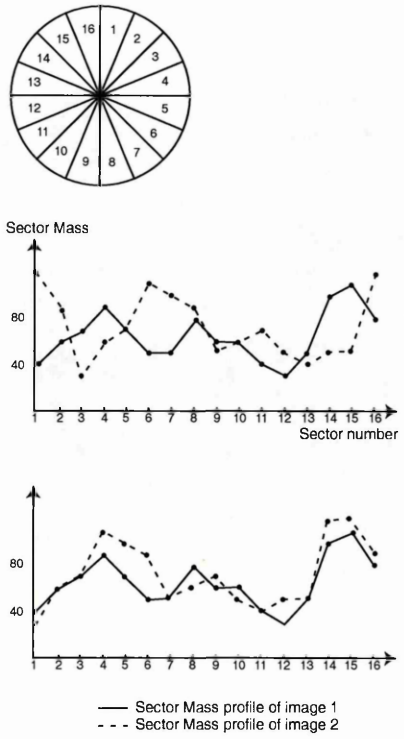

Fig. 3.6 Mean density values from each sector are calculated and aligned to the reference values. The top graph shows two sample profiles, which are shown angularly aligned in the lower graph.

χ^2 values are then determined between the averaged sector means from the candidate window and the aligned reference sector mean profile.

$$\chi_t^2 = \sum_{j=1}^{16} \frac{(\mu_{tj} - \mu_j)^2}{\mu_j}$$

where t is the test image and μ_j is the mean derived from the reference images.

3.4 Clustering

Candidates remaining after the previous tests have been found to require pruning to select the final particle from each set of overlapping windows (Figure 3.7).

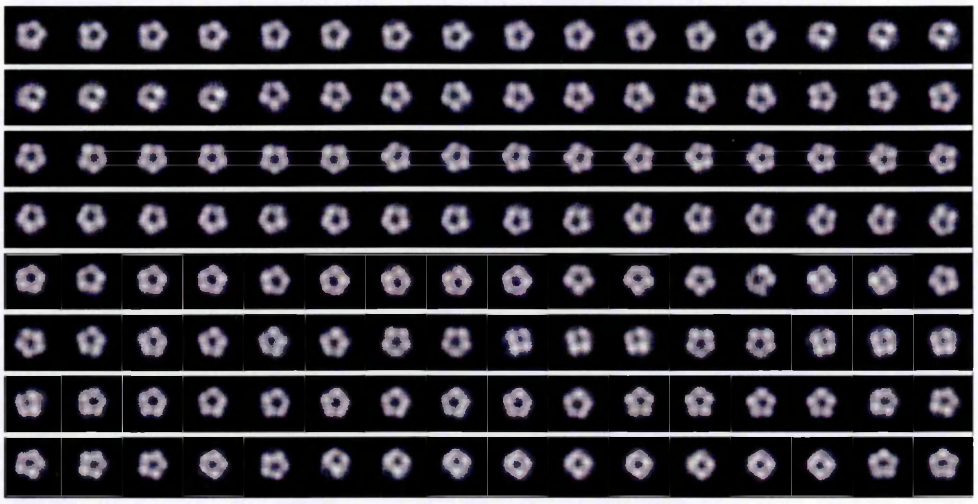

Fig. 3.7 A set of candidate images selected by this algorithm from part of a micrograph image of KHMT particles. The program computes a linear array of boxes. For display purposes the array is shown here arranged in rows such that the first particle in the array is top left and the last is bottom right. There are several sets of overlapping windows for each particle which are clearly not grouped contiguously. Since in the program the window moves across the digitized image from column to column for each row, the windows are ordered first according to their column, then row. When there are several different particles in a single row in the image, then only portions of a candidate cluster are arranged in contiguous order. For example, in row 6, images in positions 9-11 are overlapping windows of the same images as those in positions 14-16 (and position 1 in row 7).

The overlapping windows are sorted into groups by a clustering technique derived from an algorithm devised by Airlie McCoy (private communication). This method produces an array containing candidate particle coordinates grouped by proximity, with pointers to clusters and positions within each specific cluster (Figure 3.8).

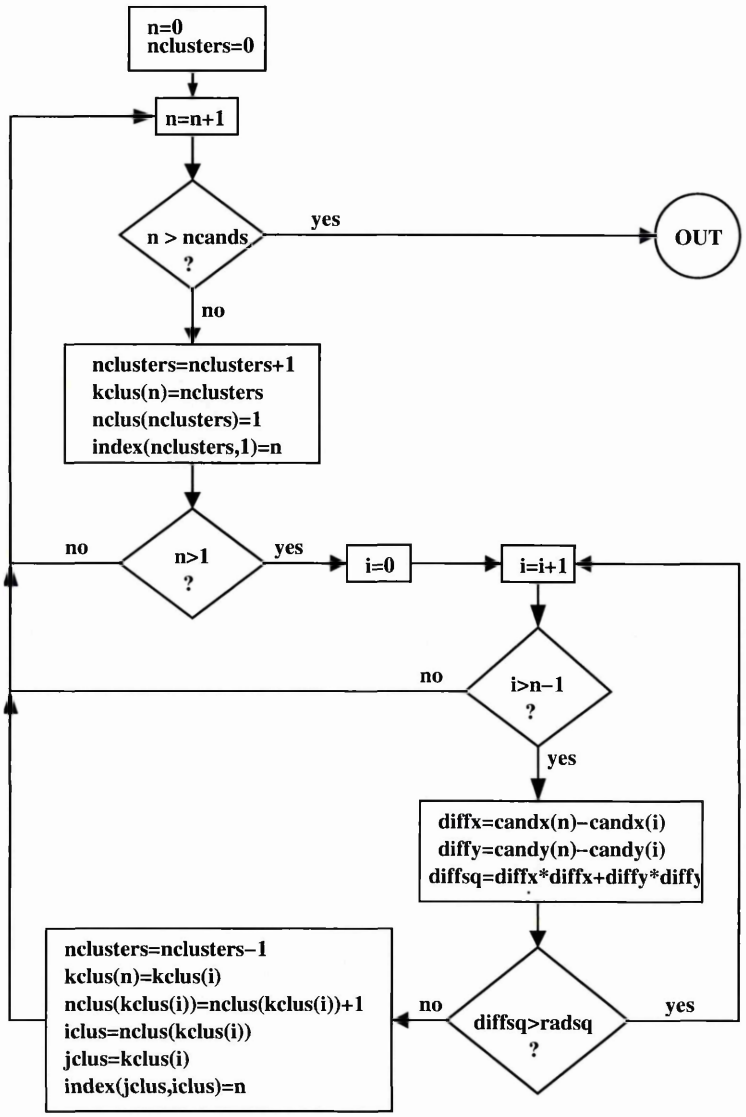

Fig. 3.8 This flowchart demonstrates the indexing of candidate positions into groups. n is the candidate number up to *ncands*, where *ncands* is the total number of candidates; *candx* and *candy* are arrays containing the candidate coordinates. *nclusters* is the total number of clusters. The array *nclus* contains the number of entries per cluster, *kclus* contains the cluster pointer for each entry, and *index* is a two-dimensional array with elements set to the cluster number and the entry number in the cluster and points to the candidate number in the *candx* and *candy* coordinate arrays. *radsq* is R^2 where R is the user-specified particle radius.

3.5 Final particle selection

Clusters are examined for size and any which have a dimension greater than the particle diameter are assumed to be particle aggregates and are therefore rejected. At this stage the window most likely to contain the best centred particle is selected from each cluster. Although matching the centre of gravity to the window centre might have been thought to be the method of choice, this is not used for the following reason. The "middle" of a particle is a more appropriate centre for a rotational alignment in a single particle analysis than the centre of mass. Although these two positions may be identical in the case of a spherically-symmetric particle with homogeneous density distribution, they can also be apart by a considerable distance in particles of other shapes (Figure 3.9).

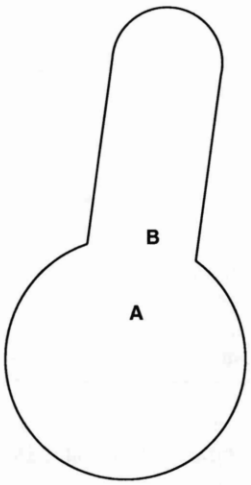

Fig. 3.9 This figure shows a shape where the centre of mass (A) is located some distance away from the particle centre (B).

The ring mean and variance statistics have been found to give very good results when used to select the most suitable window from a cluster. χ^2 values for the ring means and variances are summed for each window; the minimum value in the cluster determines the final choice of window containing the best centred particle.

A final pass examines inter-particle distances, rejecting both of any pair which fail to meet the user-specified distance criterion.

Chapter 4

Reference Value Preparation

The reference values referred to in Chapter 3 are determined from a stack of boxed images which are manually selected using a program such as Ximdisp (Crowther, Henderson and Smith, 1996; Smith, 1999). Images in the stack should be aligned to the box centre and isolated from all other material. They should also be in a box of sufficient size to allow for a surrounding band of density. This part of the program runs only once for each set of micrographs, as all necessary parameter values are stored in a file for future use.

4.1 Pre-processing

Each image in the stack is pre-processed automatically by density inversion and/or compression (where appropriate) followed by de-noising as described in Chapter 2 (Figure 4.1).

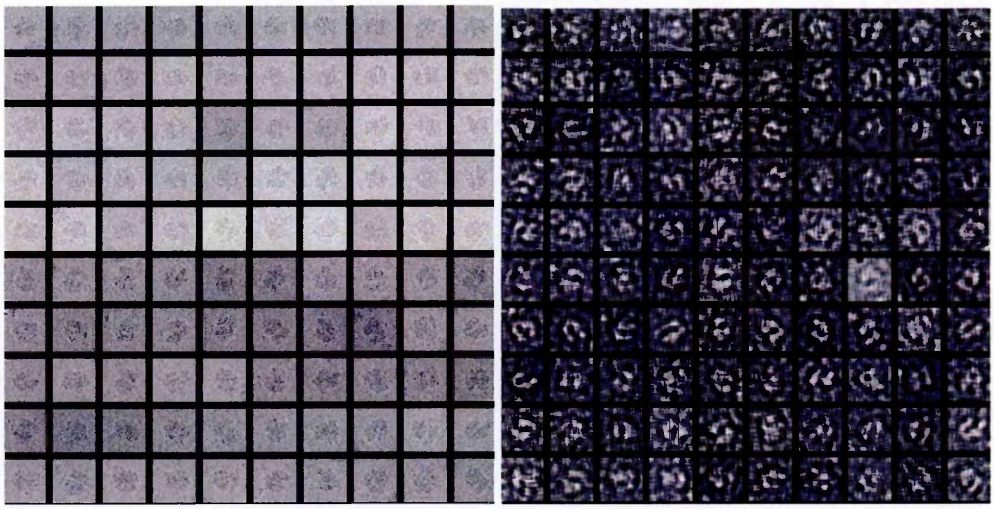

Fig. 4.1 Shows (left) a stack of raw ribosome reference particles and (right) the same stack after inversion, local averaging, contrast enhancement and high-pass spatial filtering.

4.2 Particle alignment

In order to estimate parameter values correctly, it is crucial that each reference particle is centred as accurately as possible; despite careful selection, manually determined centres may be one or two pixels out of alignment. Each reference image is translated to the centre of its box by the following procedure. Pixel densities are first tapered from the box centre to the edge by application of an exponential function (described in Chapter 3). The tapered image is then binary-thresholded; horizontal and vertical minima and maxima are measured from the resulting binary image to give a displacement from the box centre. The reference image is shifted by the measured amount and the procedure is iterated to convergence.

4.3 Parameter value determination

Parameter values extracted from the aligned image are calulated as described in Chapter 3 :

1. Mean density ratio and variance ratio between the central circular region of particle radius R and its surrounding annular band.
2. Variance of central region
3. Density sum
4. Radius of gyration.
5. Ring means and variances
6. Sector means.

4.4 Ring parameter reference values

For each reference image a set of mean densities and variances is determined from equally-spaced concentric rings. The set of means and variances is averaged over N reference particles for each ring.

$$\mu_i = \frac{\sum_{n=1}^{N} \mu_{ni}}{N} \qquad \sigma_i^2 = \frac{\sum_{n=1}^{N} \sigma_{ni}^2}{N}$$

where i is the ring number

χ^2 values for mean and variance are calculated for each reference particle as a measure of goodness-of-fit of the ring parameters, and ranges of values of the χ^2 statistic are stored.

$$\chi_{n\mu}^2 = \sum_{i=1}^{I} \frac{(\mu_{ni} - \mu_i)^2}{\mu_i} \qquad \chi_{n\sigma}^2 = \sum_{i=1}^{I} \frac{(\sigma_{ni}^2 - \sigma_i^2)^2}{\sigma_i^2}$$

where I is the number of concentric rings

4.5 Sector parameter reference determination

For each reference image, mean values from 16 circular sectors are calculated from a circle radius R, the particle radius (see Chapter 3). The sets of sector means for all N reference particles are angularly aligned to each other, using a least-squares fit, and are then averaged together to produce the reference set of means μ_j, where $j = 1, 16$.

4.6 Outlier rejection

Where a user has set a flag to test the reference particle parameter values for consistency, they are compared to each other, outliers are rejected and the procedure iterated to convergence to provide a well-matched set of reference values. A reference is rejected if its value is more than 3σ from the mean μ for any one of the following parameters : ratio mean, density mass or radius of gyration.

4.7 Determination of acceptability ranges

Mean and standard deviation values are calculated for the parameter values determined over the reference images. From these standard deviations (calculated for ratio means, ratio variances, density sum, central variances and radii of gyration), ranges of acceptability for each test are determined. In the case of the ring mean and variance and of the sector mean, minimum and maximum χ^2 values from the reference images are used as goodness-of-fit criteria and stored for use as acceptability ranges.

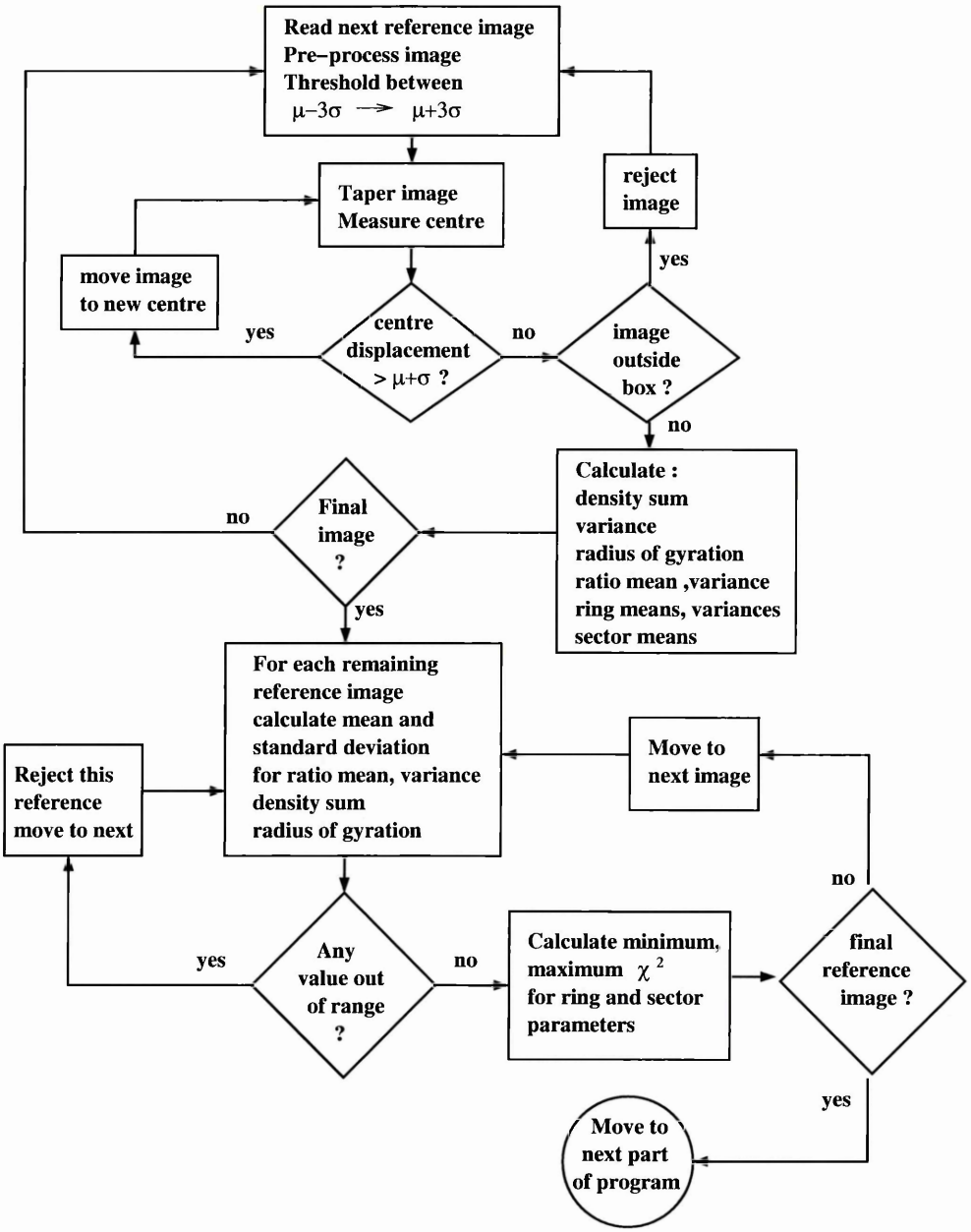

Fig. 4.2 This flowchart illustrates the interative procedure used for selecting the reference parameter values.

Chapter 5

Program Structure and Use

Primarily, the assumption was made that users of the automatic particle detection software SLEUTH would process many micrograph images from the same batch with similar background levels. Functionality of the program is twofold. It can be run using the visualization capability which provides an interactive tool for adjusting parameter values to maximise particle selection accuracy. Once the parameter values are selected, they can be stored in a file. The program can then be run in command line mode from scripts set to process all the micrograph images from the same batch, accessing the previously written file of parameter values.

Pre-requisites include the particle radius in pixels and a stack of around 100 boxed reference particle images. These images can be selected using a visualization program such as Ximdisp (Crowther, Henderson and Smith, 1996; Smith, 1999). The box must be square and of sufficient size to include the annular ring and satisfy the averaging requirements; a suggested box size would be 2 x D, where D is the particle diameter in pixels.

The program itself is further subdivided into two main sections : a) preparation of the references and b) selection of particles from the micrograph image (Figure 5.1).

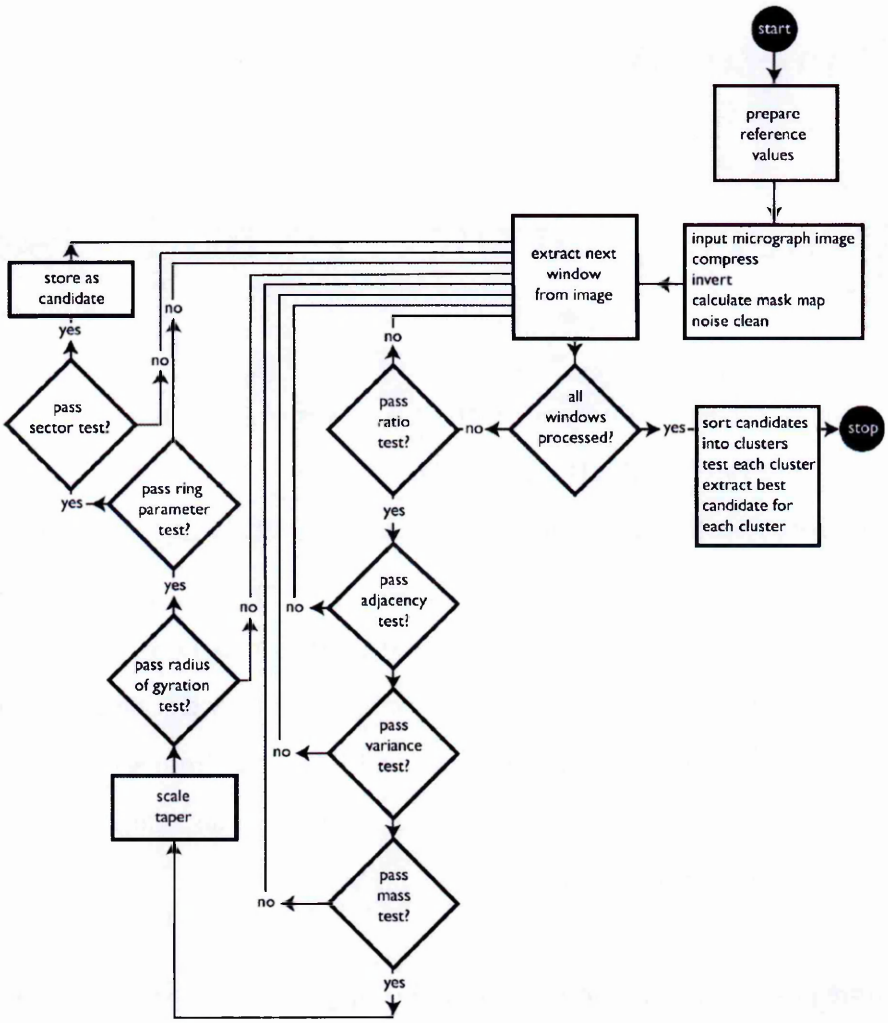

Fig. 5.1 A flowchart describing the preparation of the references is shown in Chapter 4, Figure 4.2. The parameter values determined from the reference images and acceptibility ranges for the tests are stored in a file. This enables the reference preparation section of the program to be bypassed when batch processing micrograph images to select particle positions; all necessary parameter values are read back from the file.

SLEUTH is written in Fortran 77 and has so far been tested under Tru64 UNIX and RedHat LINUX. It requires around 700MB of memory and will process micrograph images up to 67MB.

5.1 Graphical user interface

SLEUTH is interfaced to its display capability through a library of Fortran 77 and C subroutines called Ximagelib, written by the author. The library accesses the X-windows package, and has been in use for several years by other applications software (Smith and Singh, 1996; Smith, 1999). SLEUTH is invoked in interactive mode via a switch (-f) and immediately displays a window with menus and dialog boxes which prompt the user to type in new values or, in appropriate cases, to accept defaults offered by the program. The following values are input for the reference preparation part of the program :

1a) particle radius R in pixels.

2a) radial scale factor to include annular ring (default 1.25).

3a) Minimum inter-particle distance (default 2 x (R + 2.0)).

4a) Number of standard deviations for the ratio test (default 1.0).

5a) Density inversion flag (true if average particle density < average background density).

6a) Compression factor (default calculated such that $10 < R < 15$).

7a) Pixel averaging factor for noise cleaning (default 5 x 5 box).

8a) Flag for testing references (true unless wide variety of reference image shapes).

9a) Filename of input stack of reference images.

10a) Output filename for storing parameter values.

The references are then processed and displayed on the screen (Figure 5.2)

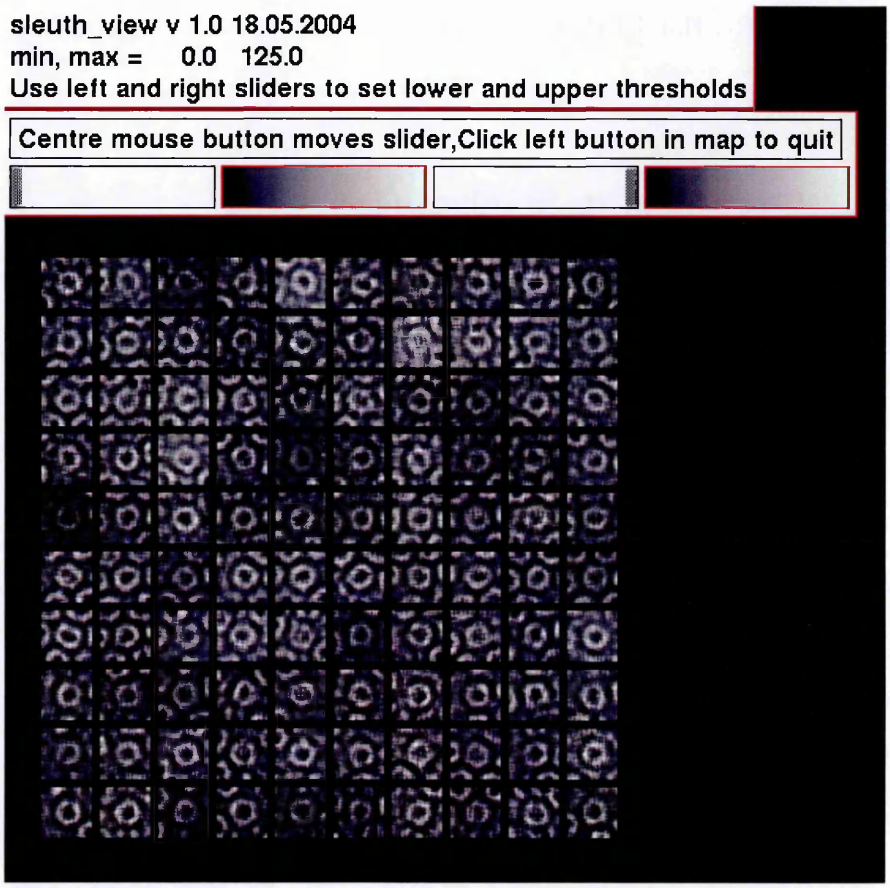

Fig. 5.2 A set of noise cleaned Hepatitis B virus core particle reference images displayed by SLEUTH. Slider bars can be used to modify the image contrast for viewing. The reference particles were selected from a single image and are clearly very close to each other. If the same image is searched for particles, the particle proximity would have implications for choosing the inter-particle distance setting and for the outcome of the adjacency test.

The program then advances to the particle selection stage from a micrograph image. If a part of an image is used, which is recommended in the case of large images in order to save testing time, unwanted areas of carbon and la-

bels (if present) should be included. Information is entered before processing the image :

1b) image filename.

2b) flag to indicate presence of label and/or carbon.

3b) flag for particle adjacency test (true unless particles elongated).

4b) percentage of total density histogram extremities to be ignored (default 5%).

5b) percentage of histogram peak height to be cut for contrast modification (default 0.5%).

6b) number of standard deviations about the mean for parameter matching tests (default 2.5).

7b) output format : MRC, SPIDER, IMAGIC.

8b) output coordinate filename.

At this point the pre-processed map is displayed (Figure 5.3) and it is possible to re-set the compression factor and/or the pixel averaging window size.

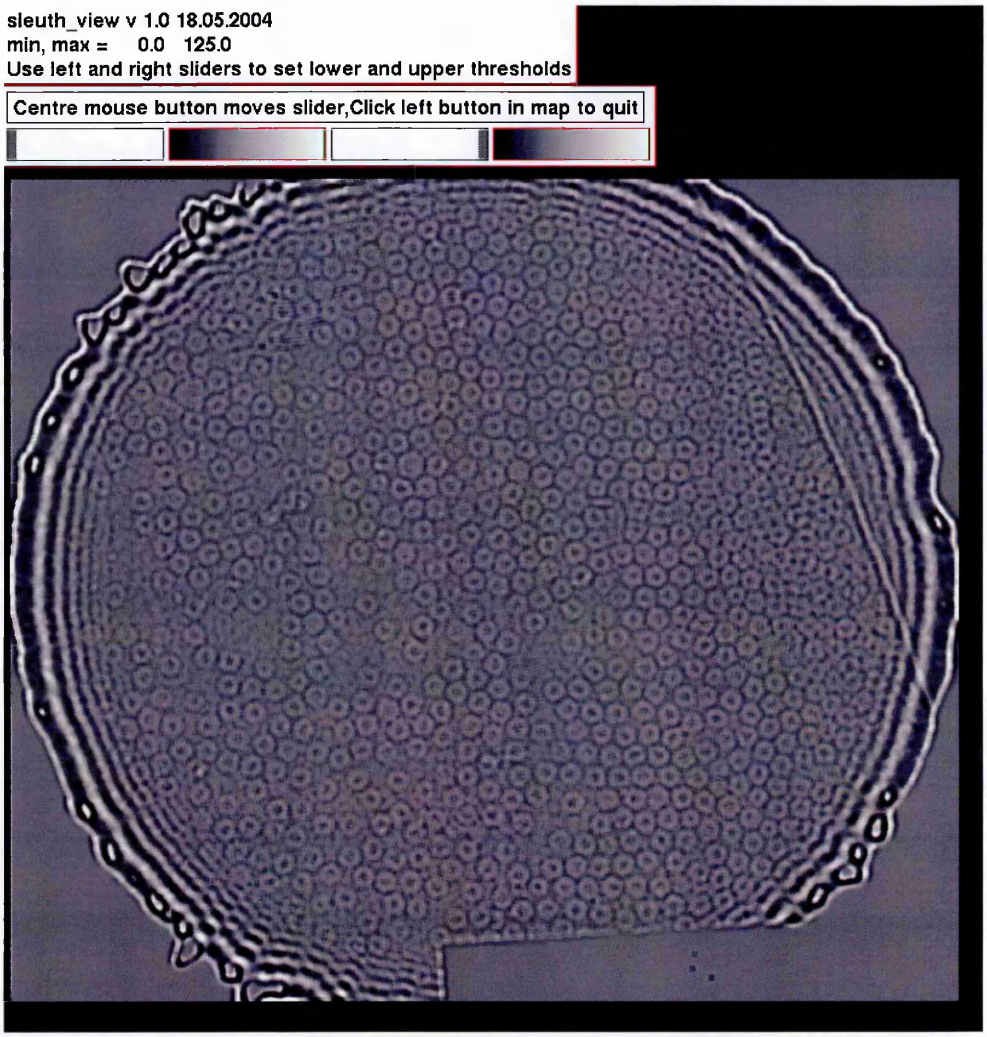

Fig. 5.3 The pre-processed micrograph image of Hepatitis B virus core particles. The micrograph label appears at the bottom of the picture. Slider bars can again be used to modify the contrast for display purposes.

The binary mask is then displayed (Figure 5.4) and at this point the histogram parameter values can be modified to mask out unwanted areas.

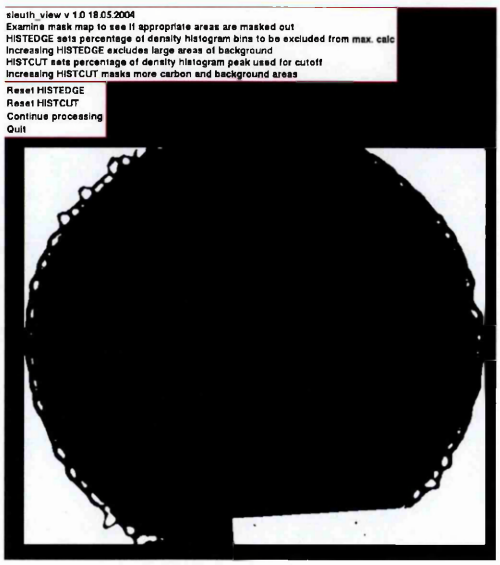

Fig. 5.4a The binary mask for the same image as Figure 5.3. The label and all the unexposed areas of film are in white and will be ignored by the program.

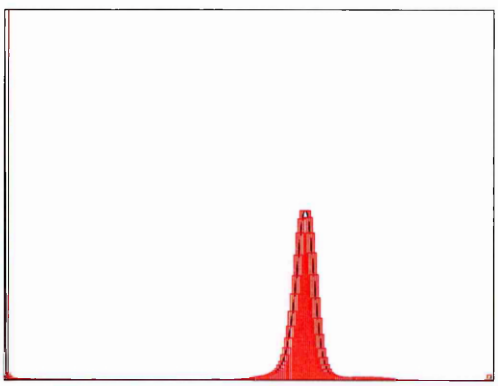

Fig. 5.4b The HISTEDGE parameter sets the percentage of the total density histogram to be excluded at the edges. Its purpose is to eliminate contributions from the label and unexposed areas of film (shown by the vertical red line at the far left and the small peak at the far right of the histogram). HISTCUT is the percentage of the main histogram density peak height to be selected for contrast modification.

After setting all these values, the image is processed and each pixel position not flagged for exclusion by the binary mask map is considered a potential candidate. Comparisons between parameter values from each window and the references now take place. Unsuccessful candidates are rejected as each comparison proceeds. Finally, successful candidate positions selected from each cluster can be displayed overlaid on the original micrograph image (Figure 5.5).

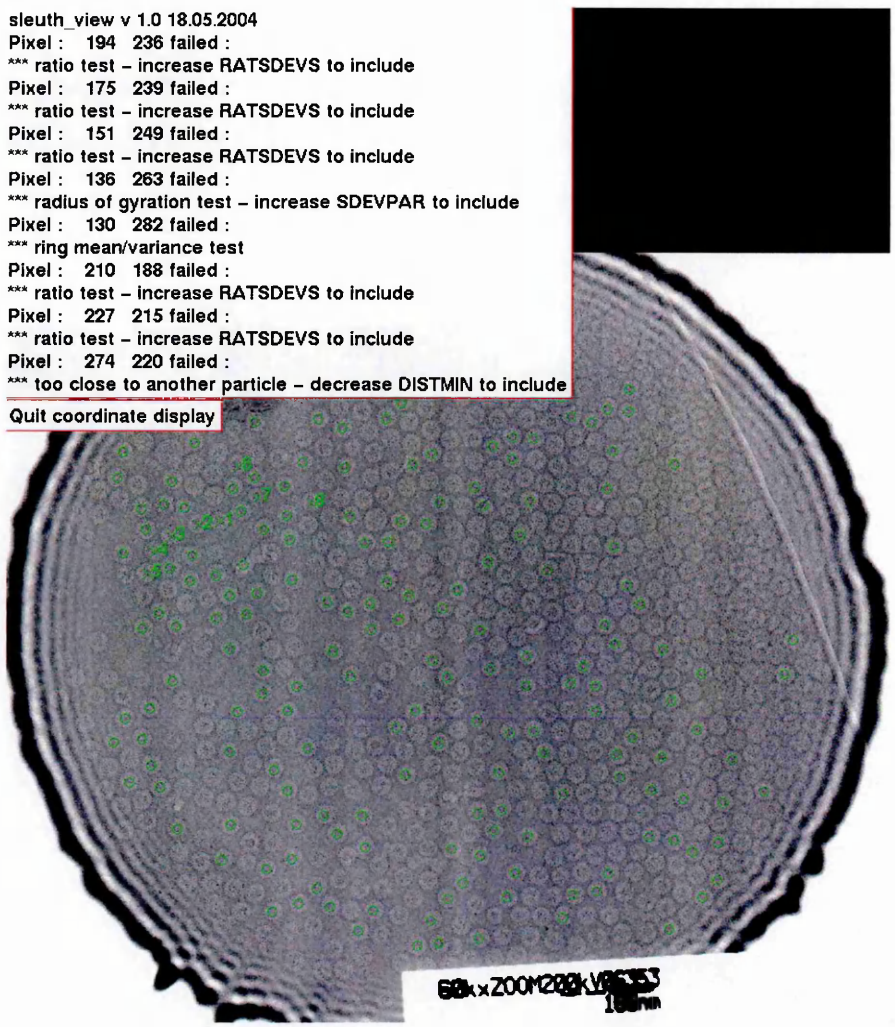

Fig. 5.5 Results of the first pass taking the default values clearly show the necessity for tuning the parameter values, as only a subset of acceptable particles has been selected (green circles). The parameter values can now be modified to provide the most accurate results. At this stage, the user can select a pixel position with the cursor; the outcome for the window centred on that pixel is displayed in the dialog box, indicating how to adjust the parameter values. The dialog box at the top suggests modifications for up to three different parameter values. After modification, the whole procedure can be iterated until a satisfactory result is achieved.

At the same time, a log file is written which provides information about each candidate. This file can be examined while the program is being executed and acts as a further aid to enable the user to adjust the parameter values to obtain the best results (Figure 5.6).

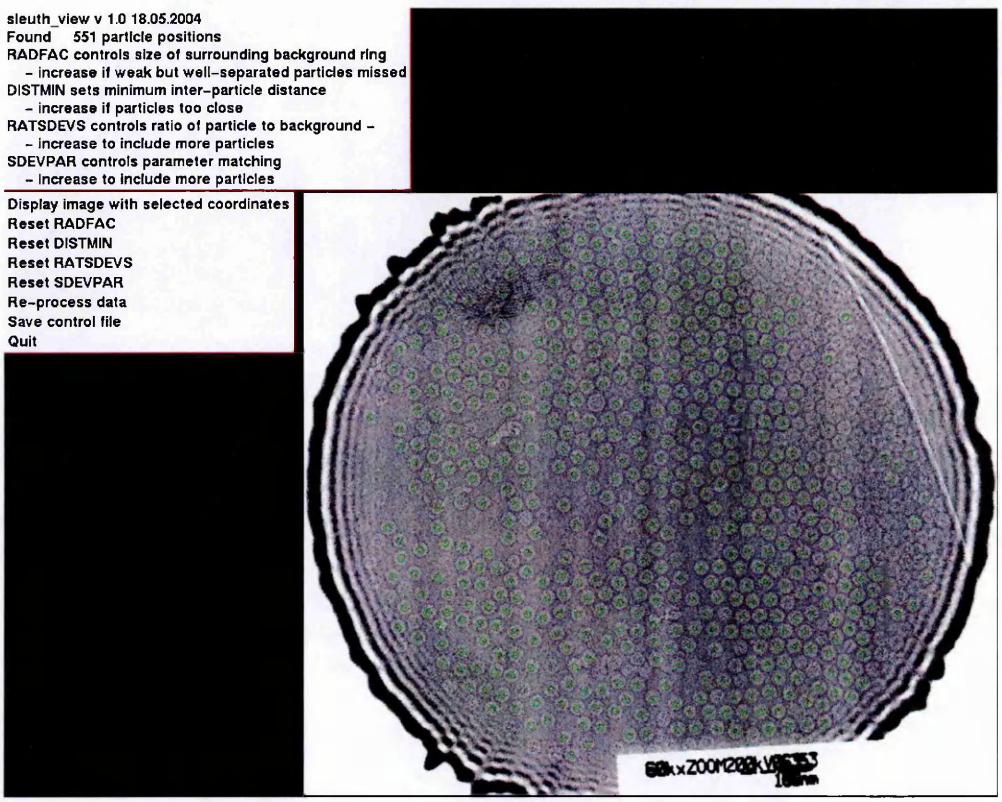

Fig. 5.6 Adjustments to the parameter values have caused the program to select most of the separated virus particles successfully while minimising the number found in the aggregates. The label and the unexposed areas of the film were completely excluded from the search.

Finally, a menu item can be selected for the program to write a control script into a file with all the parameter values set for processing a batch of micrograph images in command line mode.

5.2 Command line interface

SLEUTH runs in command line mode if the -f switch is omitted. It will process references if required, or read parameter values from the file written by a previous run. A typical script which does process references might look like this :

```
#!/bin/csh -x -e
#
time sleuth.exe << eof
1                       !  FLAG SET TO PROCESS REFERENCES
12.0,0.0,0.0,0.0        !  RADIUS, RADFAC, DISTMIN, RATSDEVS
1,0,0,1                 !  INVERT, ICOMPRESS, NPIXLOW, ITESTREFS
reference.stack         !  INPUT REFERENCE FILE NAME
reference.params        !  OUTPUT PARAMETER FILE NAME
film.mrc                !  INPUT IMAGE FILE NAME
1,1                     !  MICTYPE, IADJACENT
0.0,0.0,0.0             !  HISTEDGE, HISTCUT, SDEVPAR
2                       !  IOUT=1 output stack of images, 2 coordinates
film.coords             !  OUTPUT COORDINATE FILE NAME
eof
```

A script to process 3 images from data stored in the parameter file might look like this :

```
#!/bin/csh -x -e
#
time sleuth.exe << eof
2                       !  FLAG SET TO PROCESS REFERENCES
reference.params        !  OUTPUT PARAMETER FILE NAME
film1.mrc               !  INPUT IMAGE FILE NAME
1,1                     !  MICTYPE, IADJACENT
0.0,5.0,3.0             !  HISTEDGE, HISTCUT, SDEVPAR
2                       !  IOUT=2 output coordinate file
film1.coords            !  OUTPUT COORDINATE FILE NAME
eof
#
time sleuth.exe << eof
2                       !  FLAG SET TO PROCESS REFERENCES
reference.params        !  OUTPUT PARAMETER FILE NAME
film2.mrc               !  INPUT IMAGE FILE NAME
1,1                     !  MICTYPE, IADJACENT
0.0,5.0,3.0             !  HISTEDGE, HISTCUT, SDEVPAR
2                       !  IOUT=2 output coordinate file
film2.coords            !  OUTPUT COORDINATE FILE NAME
eof
#
time sleuth.exe << eof
2                       !  FLAG SET TO PROCESS REFERENCES
reference.params        !  OUTPUT PARAMETER FILE NAME
film3.mrc               !  INPUT IMAGE FILE NAME
1,1                     !  MICTYPE, IADJACENT
0.0,5.0,3.0             !  HISTEDGE, HISTCUT, SDEVPAR
2                       !  IOUT=2 output coordinate file
film3.coords            !  OUTPUT COORDINATE FILE NAME
eof
```

Chapter 6

Performance and Discussion

However sophisticated the algorithm, particle detection software is only as good as its results. In practical terms, its performance can be judged by the false positive and negative rates coupled with the requirements for user and computation time. SLEUTH has been subjected to several different types of trial to assess its overall value; tests have been carried out on a variety of particle shapes and sizes, on defocus pairs and on a series of micrographs from the same batch.

6.1 Results with different particle varieties

Several micrographs containing different particle types were tested with SLEUTH, some from ice-embedded and some from negatively stained specimens. Spherically-symmetric, decameric and asymmetric particles were tested, some of which were very small and noisy, with molecular weights ranging from 0.11 to 4 MD. The numbers of particles selected ranged from about 100 to several thousand per micrograph, resulting in an average of 7% false positives and 9% false negatives, although these numbers varied from micrograph to micrograph. These percentage figures were calculated in comparison with manually selected par-

ticle positions. False positives included areas of background, damaged or non-isolated particles which were unlikely to be selected, and false negatives were particles considered acceptable for analysis which were missed by the program. Processing time depended on micrograph size, the compression factor, and the particle population density (Table 6.1).

particle shape	image size	radius	molecular weight	compr. factor	number found	CPU time	%false +ves	%false -ves
asymmetric	5490 x 7080	20	0.11MD	3	3057	270	7	8
decameric	2756 x 3525	10	0.28MD	1	129	725	6	20
decameric	2772 x 3573	22	0.28MD	2	237	320	2	10
spherical	5544 x 7178	26	1.2MD	3	464	406	13	5
asymmetric	1126 x 1406	10	2.5MD	1	717	211	6	5
spherical	785 x 686	9	4MD	1	565	42	7	5

Table 6.1. The results of processing micrograph images of six different specimens on a 500MHz ESV5 Alpha. Image size and particle radius are in pixels, CPU time in seconds. The specimens used were, starting from the top : phosphoinositide 3-kinase gamma (stain), KHMT (ice), KHMT (stain), Hepatitis B virus surface antigen coated ferritin (stain), 70S bacterial ribosomes (ice), Hepatitis B virus core particles (ice).

The relatively heavy processing time recorded for the second image in the table was due to a sparse population of very small particles in a large image, which was precluded from compression by the particle size. Conversely, the considerably larger first image in the table had a crowded population of larger particles which made this image suitable for compression; 30 times as many particles were found in around one third of the processing time compared to the second image in the table.

Extracts from four of these images, overlaid with particle positions selected by SLEUTH, are shown in Figure 6.1, all of which had labels which were flagged successfully by the binary mask.

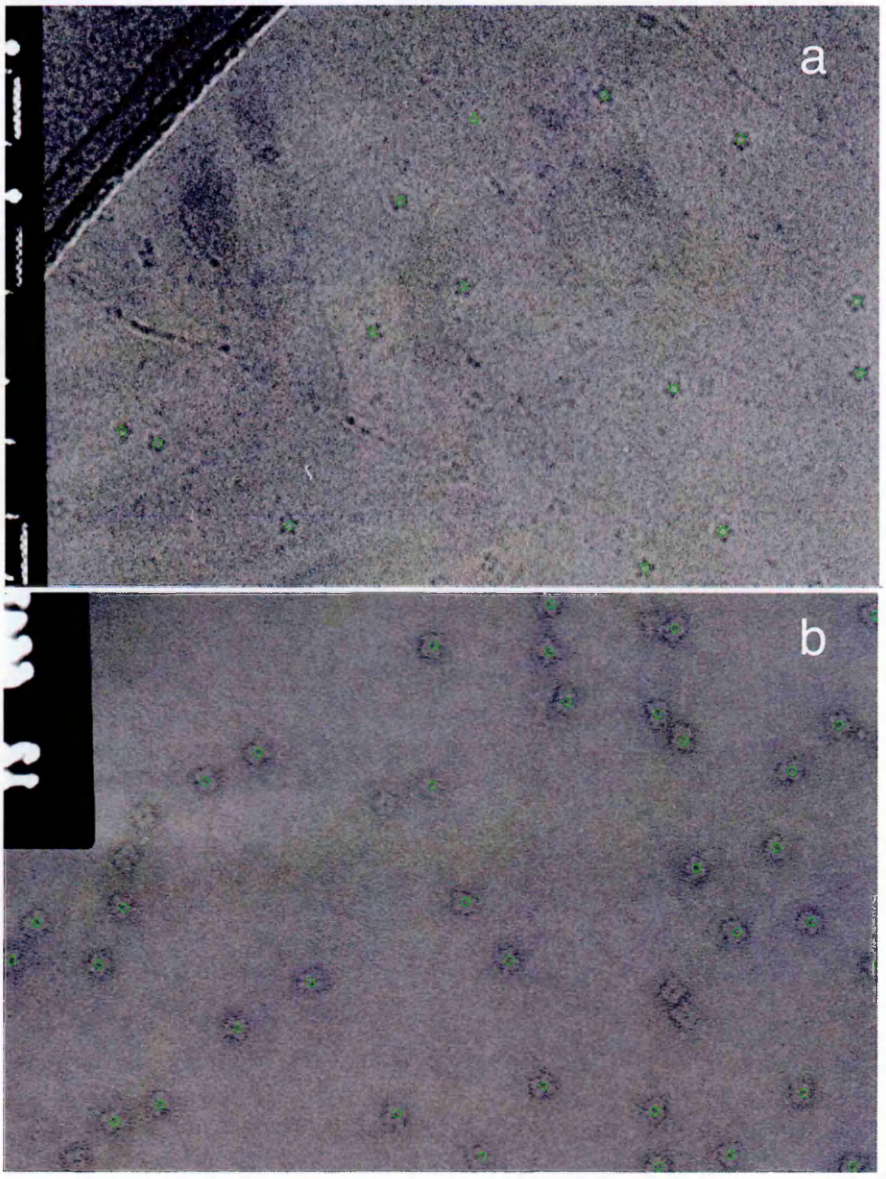

Fig. 6.1a Detected particle positions are indicated by green circles (a) Pentameric, tetrameric and side views of KHMT decamers in ice with a substantial area of thick carbon. (b) KHMT in negative stain.

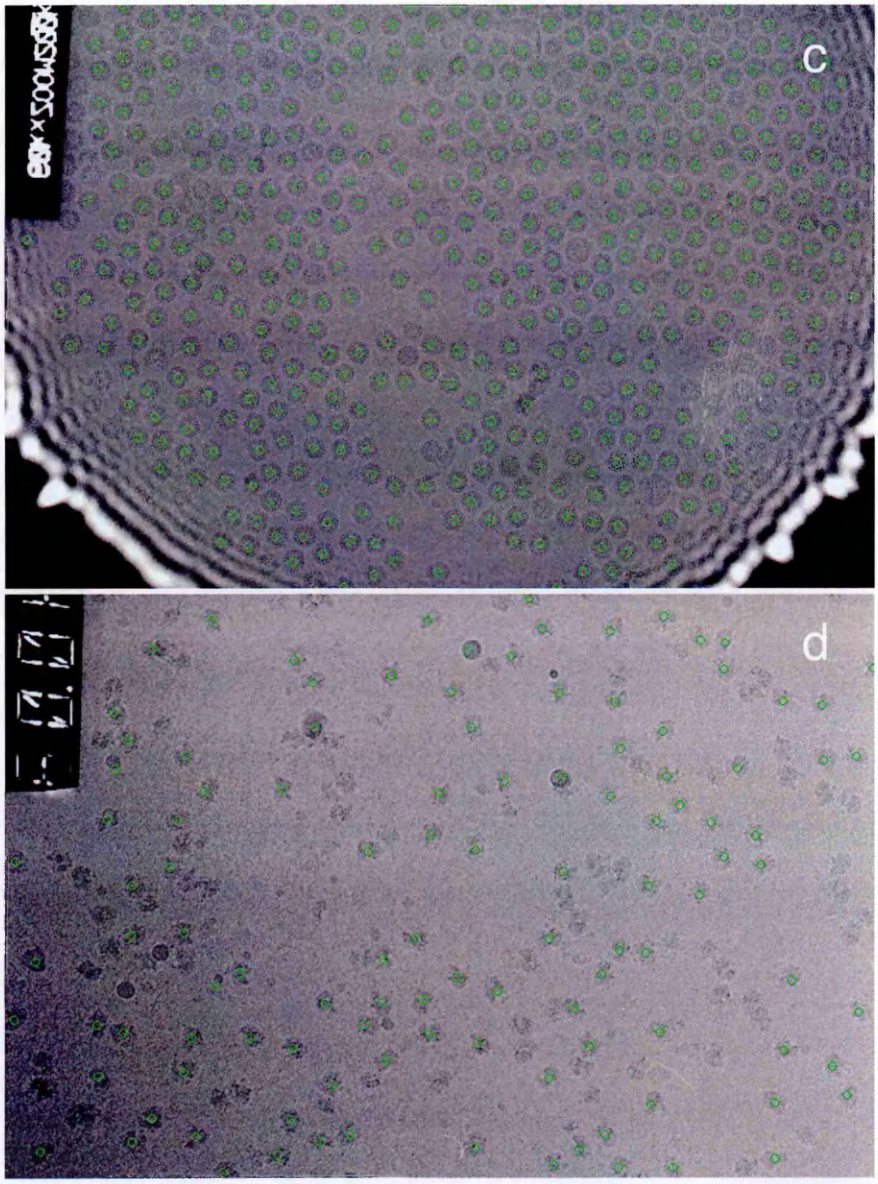

Fig. 6.1b (c) Spherically-symmetric hepatitis B virus core virus particles in ice with areas of unexposed film in the corners which were successfully flagged for exclusion from searching. (d) Asymmetric 70S bacterial ribosomes in ice.

6.2 Results with defocus pairs

High resolution data is obtained from close to focus images, but such images often lack sufficient contrast to make the particles easily visible by eye, so may cause problems for the user when selecting the reference images and setting up the parameter values with the visualization part of the program. They may therefore decide to carry out these procedures with a further from focus, higher contrast image, with a view to using the parameter values to select particles from a series of close to focus images.

To test the performance of SLEUTH when used in such a way, 100 reference particles were selected from three far from focus images, and the software was used in its interactive display mode to set the parameter values from one of these images, which was one of a defocus pair. The corresponding near to focus image of the pair was then subjected to processing using these parameter values but with a range of values for RATSDEVS and SDEVPAR. The two acceptability ranges set by RATSDEVS, which controls the number of standard deviations in the ratio test, and SDEVPAR, which sets the number of standard deviations for the particle mass and radius of gyration comparisons, work in tandem and are crucial to the outcome of the program. Too low a value of RATSDEVS results in the selection of background and other unwanted areas. When RATSDEVS it is correctly set, the very simple but powerful ratio test restricts the search to candidates containing an isolated area of density of the appropiate size; hence is responsible for the speed of the software. The results shown were checked against 83 particles which were carefully selected by eye as being of reasonable shape, size and quality; those which were misshapen, non-isolated or immediately adjacent to the edge of the micrograph image were not used in the test set (Table 6.2).

Parameter values		Near to focus			Far from focus		
RATSDEVS	SDEVPAR	N. near	F.Neg	F.Pos	N. far	F.Neg	F.Pos
0.6	0.6	54	33	4	81	7	5
0.6	0.7	57	30	4	84	5	6
0.6	0.8	59	28	4	86	5	8
0.6	0.9	61	27	5	85	7	9
0.7	0.6	66	23	6	85	5	7
0.7	0.7	67	22	6	87	4	8
0.7	0.8	71	18	6	89	3	9
0.7	0.9	72	17	6	90	4	11
0.8	0.6	72	18	7	85	7	9
0.8	0.7	75	15	7	88	6	11
0.8	0.8	75	15	7	88	6	11
0.8	0.9	77	14	8	91	5	13
0.9	0.6	80	13	10	87	7	11
0.9	0.7	84	10	11	91	6	14
0.9	0.8	85	9	11	90	6	13
0.9	0.9	87	8	12	92	5	14

Table 6.2 shows the total number of positions found by the program with the numbers of manually judged false negatives and positives, for both near and far images of the defocus pair. The images are of Woodchuck Hepatitis B surface antigen coated ferritin particles taken on a Hitachi HF2000 microscope equipped with a field emmission gun and Gatan cold stage operated at 200kV. The micrographs were taken at a magnification of 60,000 and scanned on a Zeiss SCAI scanner with a step size of 7 microns, then compressed by 4 to give a pixel resolution of 4.7Å . Defocus values were calculated at 2.2 and 4.2 microns respectively for the near and far images. A compression factor of 3 was set by the program for the purposes of particle searching. Although the particles are more or less spherically-symmetric, which should make them easy for the program to detect, the distribution of the iron atoms inside the ferritin

cages is completely random, which makes matching these particles a difficult task for any program. Furthermore, the pixel density of the iron sometimes lay at the extremity of the range; this caused a few particles to be missed as they were flagged as though they were part of the micrograph label.

A curious outcome of the algorithm is that relaxing the parameter values sometimes results in the final selection of fewer acceptable particles. This unexpected effect is due to the initial addition of extra candidates to the list, which are then judged by the program as too close to another previously selected candidate, in which case both are rejected.

Occasionally, neighbouring particles with apparently acceptable spacings fail the distance criterion. This happens when the final positions determined by the program do not lie precisely at the particle centre (Figure 6.2).

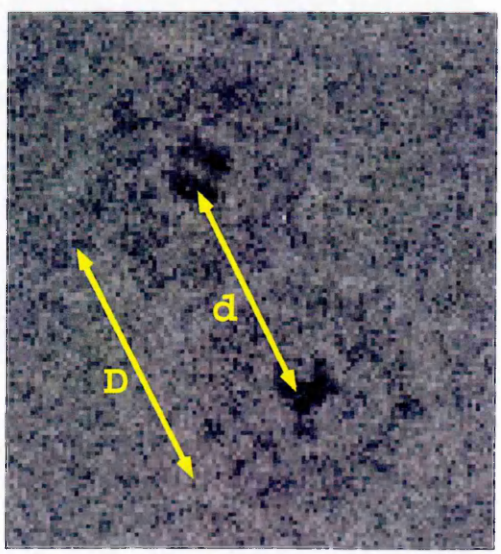

Fig. 6.2 *D* is shown as the minimum distance value DISTMIN and *d* is the distance between two particle centres determined by the program; this pair of particles would fail the minimum distance test.

The results were as might be expected; it was harder for the program to find all the particles in the closer to focus image from reference values computed from a further from focus image. This was mostly due to the reduced contrast, which caused the ratio test to fail more of the particles, some of which were barely discernible from the background. A further consideration is that particles from a second exposure in a defocus pair are frequently damaged by the first exposure; parameter values extracted from second exposure particles may not reflect the particle attributes accurately (Figure 6.3).

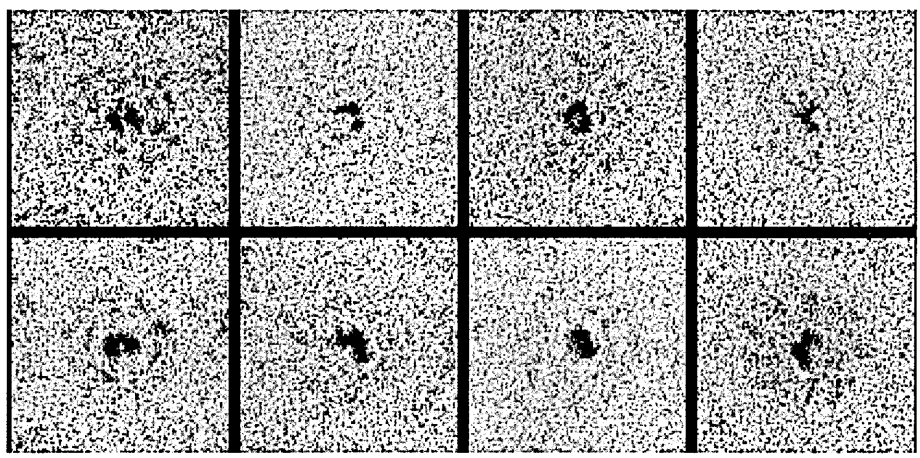

Fig. 6.3 Particle pairs of Hepatitis B surface antigen coated ferritin selected from near to and far from focus images show the damaging effects of the first, near to focus exposures on the second, far from focus images. Near to focus images are at the top.

A test was carried out to determine whether calculating the reference parameter values directly from the same near to focus image as used in Table 6.2 would improve the search outcome. The 83 "good" particle images used for testing the data in the table were boxed from the near to focus image and used as references to calculate the parameter values. The best results were obtained with a value of RATSDEVS set to 0.5 and SDEVPAR at 0.6; the

program detected 79 particles (Figure 6.4) of which 6 were false positives; 10 particles were missed. This represents a false negative rate of 12% and false positive rate of 7%, which compared favourably with the results obtained when the same image was processed using reference values derived from the further from focus image.

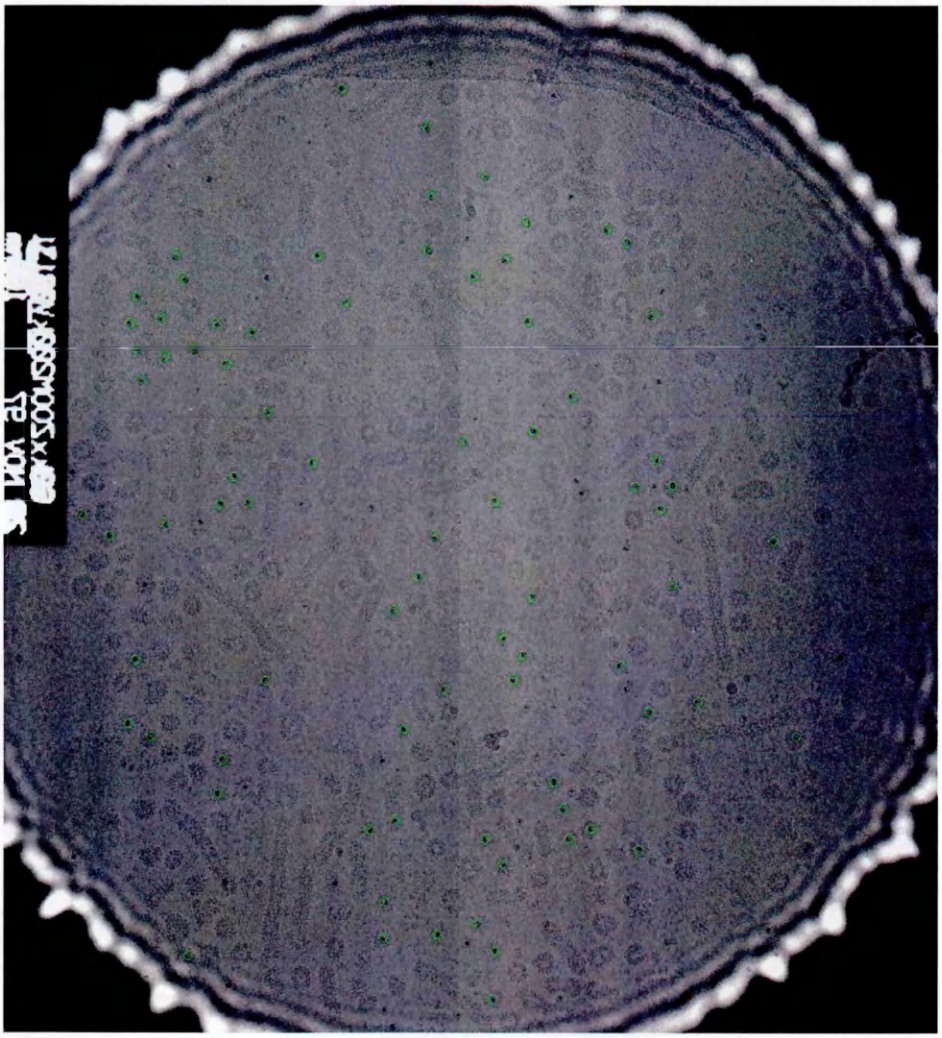

Fig. 6.4 Woodchuck Hepatitis B surface antigen coated ferritin particles detected by SLEUTH indicated by green circles overlaid on the closer to focus image of the defocus pair used to construct Table 6.1. For display purposes, the image has been contrast enhanced. It was processed on a 500MHz ESV5 Alpha in 4 minutes CPU.

It is clear that SLEUTH performs less well on low contrast images, even when the reference values are computed from the same image; the success rate falls further when reference values calculated from further from focus images are applied. Where close to focus images are involved, better results will be obtained by using reference values derived from micrograph images with similar defocus values. If particles cannot be sufficiently well distinguished by eye when selecting references from a low contrast image, a possible strategy might be to transfer positions of reference images selected from a further from focus image to a typical near to focus image for the determination of the reference values.

Possibly the most successful outcome could be achieved by using SLEUTH to select particles from far from focus images and then translate their coordinates to their closer to focus pairs.

6.3 Results with a series of micrographs

With the drive to atomic resolution for single particle methods, hundreds of thousands, and possibly millions of particles may be required; the ultimate aim of this software is to be able to search any number of micrograph images from a single parameter file in a completely automatic way, with an absolute minimum of user intervention and computation time. To test the performance of SLEUTH on a series of micrographs, reference images were selected from a single micrograph, which was also used to set all the parameter values. The resulting parameter file was then used on 7 more micrographs of the same particle type but with varying population densities. Numbers of particles selected and processing time for each micrograph are displayed in table 6.3.

Reference	Number selected	CPU time	Defocus in μ
ribo2a	785	517	1.4
ribo2b	793	522	1.9
ribo2c	353	499	2.6
ribo2d	660	517	2.0
ribo3b	783	502	3.5
ribo3c	531	498	2.7
ribo3d	979	487	2.5
ribo3e	1034	516	2.2

Table 6.3 shows results from processing 8 micrographs of 70S bacterial ribosomes in ice. The numbers of particles selected genuinely reflect the particle density; micrographs were processed on a 500MHz ESV5 Alpha and CPU times are in seconds.

In all, 4884 particles were detected in 59 minutes CPU time. User time taken to select the references and set up the parameter values was around 30 minutes. At the rate of 500 particles per hour, it would take around 10 hours to pick this number of particles manually. However, although they were not counted, there were a few false positives and some false negatives (Figure 6.5), indicating that some manual pruning would be needed.

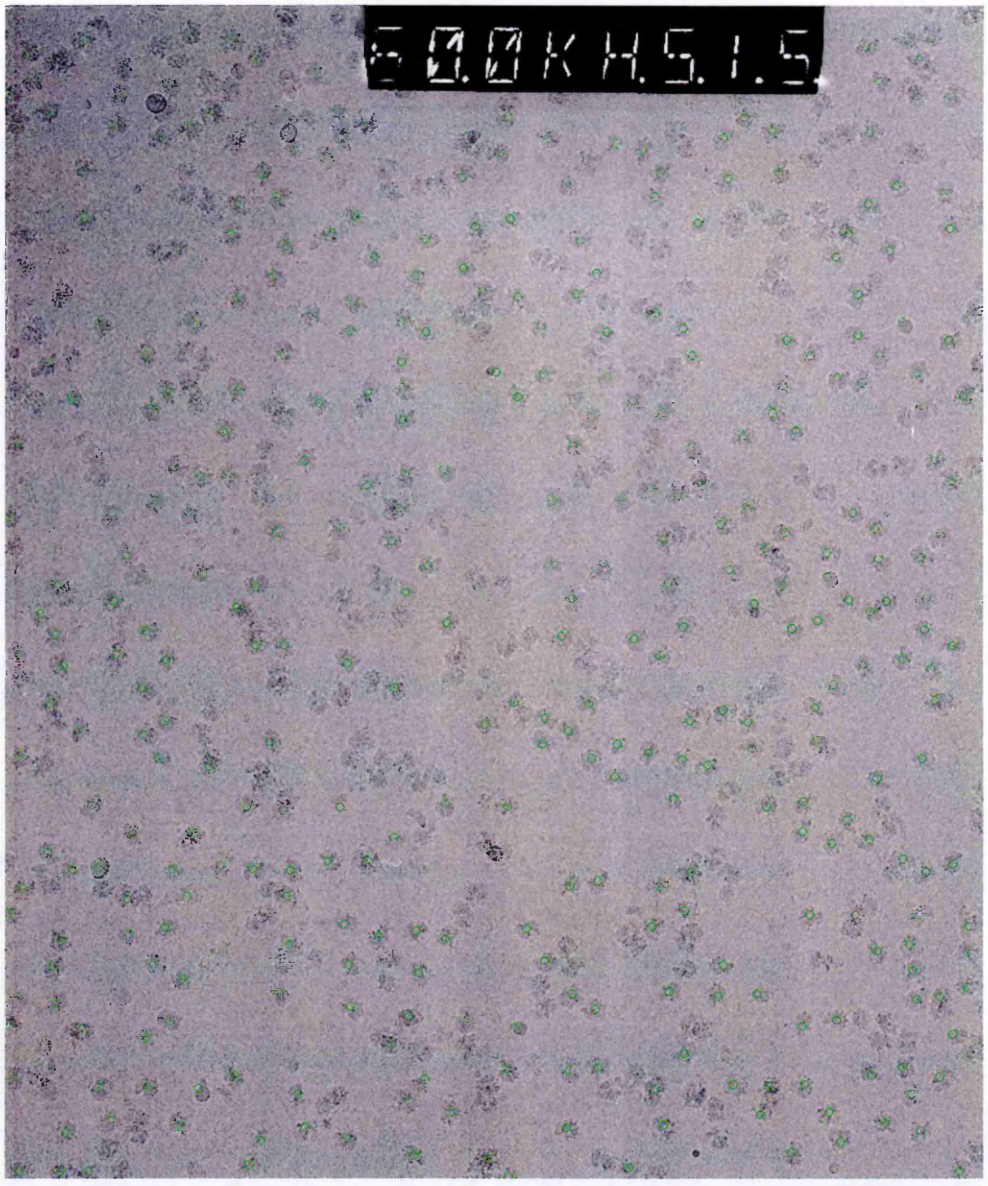

Fig. 6.5 This figure shows a part of the image ribo2b, overlaid with the particle positions (shown by green circles) selected by SLEUTH. The references and their parameters were selected from another image (ribo2d). Many of the particles not selected by the program are members of closely associated pairs and therefore were excluded by proximity.

6.4 Comparison with other methods

6.4.1 The "Bake-off"

The "bake-off" at the Multidisciplinary Workshop on Automatic Particle Selection for CryoEM used their two sets of manually picked particles as the standards against which all the others were tested. The confusion matrix (Figure 1.11) generated as a result of the "bake-off" presented the best false negative rates (FNR) as 2.4% and 1.5% (Roseman) and the best false positive rates (FPR) as 4.5% and 8.4% (Sigworth). However, their corresponding reciprocal rates were 16.1% and 23.9% (FPR) (Roseman), 23.2% and 18.4% (FNR) (Sigworth) respectively. In conclusion, although Roseman's method missed very few particles, a large number of false positives remained, which would have to be manually pruned; the converse was true of Sigworth's algorithm which missed a correspondingly large number of particles. This highlighted the fact that there was no outstanding single method which outperformed all the others. Furthermore, this analysis was restricted to finding only one view of a single type of particle : the rectangular side view of Keyhole Limpet Haemocyanin.

6.4.2 Template matching methods

It is apparent from the literature that the peak detection phase of the standard template matching cross-correlation method is highly compute-intensive, sometimes using hours of computing time to complete this part of the procedure. Preparation of the templates should also be taken into consideration, as this can also be a time-consuming part of the procedure in terms of both user and computing time. Several groups report efforts to improve the computation speed by various strategies (Roseman, 2003; Volkmann, 2004; Wong et al., 2004), apparently with mixed success. Performance rates varied; Rath and

Frank (2004) quote 10% FPR and 15% FNR, Wong et al. report 6% FPR and 17% FNR. A trade-off between FPR and FNR is a universal problem; Volkmann (2004) reported a very low FPR of 2.1% which was counterbalanced by a very high FNR of 41%.

6.4.3 Neural networks

The neural network described by Ogura and Sato (2004) is reported as giving a very high success rate of 98% but at a heavy computational cost while training the images (many hours). No figures were quoted for missed particle rates; A value of 1 hour CPU for processing an image of size 1460 x 4425 was quoted.

6.4.4 Intensity Comparison Methods

The image rendering process of the crosspoint method (Boier Martin et al., 1997) has a heavy CPU overhead. This simple procedure has been in use for many years in their laboratory, but is restricted to spherically-symmetric particles; they quote a 4-12% FPR and a 5-11% FNR.

A faster method is the circular density comparison procedure reported by Kivioja et al. (2000) with a FPR of 2-9%, and FNR of 2-8% taking 28-172 seconds to process files of size 10-170MB. However, this technique is also limited to spherically-symmetric particles.

6.4.5 Edge Detection Methods

The edge detection method described by Zhu et al. (2001), which uses the Canny edge detector to search for filaments from defocus pairs, quote FPR rates of 16-25% but no figures for FNR or CPU time. Yu and Bajaj (2004), who also use the Canny edge detector, quote very fast CPU times but they

do not include processing time for the highly compute-intensive anisotropic filtration step. Furthermore, their method of using the Voronoi diagram and the Distance Transform is currently restricted to the rectangular and circular views respectively, which were used in the "bake-off".

6.4.6 SLEUTH

The rectangular side view Keyhole Limpet Haemocyanin particles, used in the "bake-off" were very close to each other. SLEUTH detected very few of these particles as the majority of them had encroaching, but not always touching, neighbours. Some of these particles failed the adjacency test (Figure 6.6).

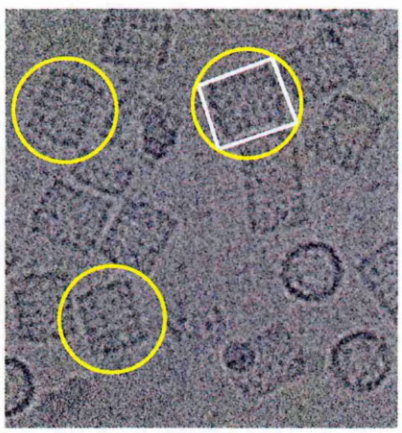

Fig. 6.6 Part of a micrograph containing rectangular side and circular end views of Keyhole Limpet Haemacyanin particles. The particles shown are very close to each other, but not actually touching; the parts of the neighbouring particles shown inside the yellow circles caused SLEUTH to reject such particles.

Other particles failed the minimum distance criterion. This case occurred where the distance between the centres of two particles lying side by side was less than the minimum distance, which was set by the long axis (Figure 6.7).

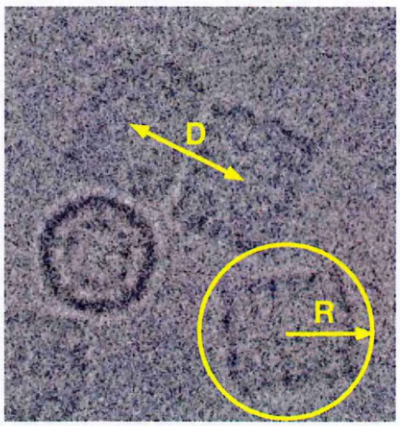

Fig. 6.7 The minimum inter-particle distance (DISTMIN) is determined from the particle diameter $2R$ and as shown here is greater than the distance D between two particles which do not actually touch each other, but would be rejected by SLEUTH.

In principle, particles close to, but not touching their neighbours could be used in a single particle analysis. However, the downstream processing of the currently available single particle packages (Frank et al., 1996; van Heel et al., 1996; Ludtke, Baldwin and Chiu, 1999) provides only a circular masking facility. Such a mask is only appropriate for asymmetric particles where they are completely isolated; particle alignment can be seriously affected by artifacts within the mask.

6.5 Conclusions

As shown in table 6.1, SLEUTH performs very well against all the methods currently described in the literature. When the parameter values are optimised, the detection rate is at least as good as any of them, with the possible exception of the Neural Network algorithm (Ogura and Sato, 2004) for which the FNR is unknown. The very fast processing time achieved by SLEUTH depends on the population density and the size of the particle relative to the

micrograph. There is a time overhead for sparse populations, or small particles which do not allow compression of large images. Despite such considerations, it still outstrips all of the other programs in terms of computation time, with the exception of the circular density comparison method (Kivioja et al., 2000), with which it compares equally well. Unlike most of the other software, SLEUTH has also been tested successfully on a variety of particle shapes and sizes. Reference preparation time is restricted to the selection of 100 typical particle images which takes only a few minutes of user time. Setting up the parameter values using the program in interactive display mode can take a little longer, depending on the size of the micrograph image and the compression factor. Any number of similar micrograph images can be searched from a single parameter file, demonstrating the ability of the software to perform in batch processing mode. Using the figures from Table 6.1, an average of 1.4 seconds CPU per particle was achieved; at that rate SLEUTH could select one million particles in a total of 16.2 days computer processing time. An average of 0.76 seconds CPU time per particle was calculated from Table 6.2 for ribosomes; a million particles would be selected in 8.8 days total CPU.

6.6 Publication

A paper describing this work was published in the special issue of the Journal of Structural Biology resulting from the Multidisciplinary Workshop on Automatic Particle Selection for CryoEMworkshop held in 2003 at the Scripps Institute (Short, 2004).

6.7 Additional software

In addition to SLEUTH, the computer program "BANDPASS" was written in conjunction with this project. It is written in Fortran 77 and is now in routine use for Fourier bandpass filtration of MRC image format images as a part of single particle processing (see Chapter 2).

6.8 Further work

Future plans include the addition of a facility for SLEUTH to be able to select the boxed reference images. Also to be incorporated into the program is the ability to add missed particles manually, using a display of the original micrograph overlaid with particle positions already found. This would be complemented by a pruning capability to allow easy removal of unwanted particles either from the micrograph image or from a gallery of boxed images.

Other plans include the installation of a custom mask for asymmetrically-shaped specimens with a view to solving the problems illustrated in Figures 6.6 and 6.7. In principle, the mask could be calculated automatically from thresholded pre-aligned reference images. Although the rotations which would be needed for this approach would inevitably reduce the computing efficiency when particle searching, a large circular mask could first be applied to exclude areas which consist solely of background. A beneficial extension to this approach could be to float and box the particles within their custom mask, thus also solving the downstream processing problems with circular masks used on asymmetrically-shaped particles. This addition to the software might imply that only one view could be searched; a further requirement would be to include sets of reference particles, each representing a particular view and with

its own mask.

With a view to further automation, hence reducing the number of user-specified parameters, a training set of non-particles could be added to the sets of references. The training set would consist of background, damaged and part particles, aggregates and noise.

Finally, a novel approach to the particle detection problem might be achieved by supplying the reference parameter values to a simple feed-forward, back-propagation neural network, such as that adopted by Ogura and Sato (2004). It is possible that such a method could produce a mechanism with similar, or superior accuracy to theirs, but with much more efficient weight training.

Bibliography

[1] Adiga, P.S.U., Malladi, R., Baxter, W. and Glaeser, R.M. (2004) A binary segmentation approach for boxing ribosome particles in cryo EM micrographs *J. Struct. Biol.* **145**, 142-151.

[2] Boier Martin, I.M., Marinescu, D.C., Lynch, R.E., and Baker, T.S. (1997) Identification of spherical virus particles in digitized images of entire micrographs. *J. Struct. Biol.* **120**, 146-157.

[3] Böttcher, B., Wynne, S.A. and Crowther, R.A. (1997) Determination of the fold of the core protein of hepatitis B virus by electron cryomicroscopy. *Nature* **386**, 88-91.

[4] Crowther, R.A., Henderson, R., and Smith, J.M. (1996) MRC image processing programs. *J. Struct. Biol.* **116**, 9-16.

[5] Evans, M., Hastings, N. and Peacock, B. (1993) Statistical Distributions, *John Wiley and Sons.*

[6] Frank, J., Radermacher, M., Penczek, P., Zhu, J., Li, Y., Ladjadj, M., and Leith, A. (1996) SPIDER and WEB: Processing and visualization of images in 3D electron microscopy and related fields. *J. Struct. Biol.* **116**, 190-199.

[7] Frank, J. and Wagenknecht, T. (1984) Automatic selection of molecular images from electron micrographs. *Ultramicroscopy* **12**, 169-176.

[8] Hall, R.J., Patwardhan, A. (2004) A two step approach for semi-automated particle selection from low contrast cryo-electron micrographs. *J. Struct. Biol.* **145**, 19-28.

[9] Glaeser, R.M. (1999) Electron crystallography : present excitement, a nod to the past, anticipating the future. *J. Struct. Biol.* **128**, 3-14.

[10] Gonzalez, R. and Woods, R. (1992) Digital Image Processing, *Addison-Wesley Publishing Company.*

[11] Grigorieff, N. (1998) Three-dimensional structure of bovine NADH:ubiquinone oxidoreductase (complex I) at 22Å in ice. *J. Mol. Biol.* **277**, 1033-1046.

[12] Harauz, G. and Fong-Lochovsky, A. (1989) Automatic selection of macromolecules from electron micrographs by component labelling and symbolic processing. *Ultramicrscopy* **31**, 333-344.

[13] van Heel, M. (1982) Detection of objects in quantum-noise-limited images. *Ultramicroscopy* **8**, 331-342.

[14] van Heel, M., Harauz, G., Orlova, E.V., Schmidt, R., and Schatz, M. (1996) A new generation of the IMAGIC image processing system. *J. Struct. Biol.* **116**, 17-24.

[15] van Heel, M.,Gowen, B., Matadeen, R.,Orlova, E.V., Finn, R., Pape, T., Cohen, D., Stark, H., Schmidt, R.,Schatz, M. and Patwardhan, A. (2000) Single-particle electron cryo-microscopy : Towards atomic resolution. *Q. Rev. Biophys.* **33**, 307-369.

[16] Henderson, R. (1995) The potential and limitations of neutrons, electrons and X-rays for atomic resolution microscopy of unstained biological molecules. *Q. Rev. Biophys.* **28**, 171-193.

[17] Huang, Z. and Penczek, P. A. (2004) Application of template matching technique to particle detection in electron micrographs. *J. Struct. Biol.* **145**, 29-40.

[18] Kivioja, T., Ravantti, J., Verkhovsky, A., Ukkonen, E. and Bamford, D. (2000) Local average intensity-based method for identifying spherical particles in electron micrographs. *J. Struct. Biol.* **131**, 126-134.

[19] Kumar, V.,Heikkonen, J., Engelhardt, P. and Kaski, K. (2004) Robust filtering and particle picking in micrograph images towards 3D reconstruction of purified proteins with cryo-electron microscopy. *J. Struct. Biol.* **145**, 41-51.

[20] Lata, K.R., Penczek, P., and Frank, J. (1995) Automatic particle picking from electron micrographs. *Ultramicroscopy* **58**, 381-391.

[21] Ludtke, S.J., Baldwin, P.R., and Chiu, W. (1999) EMAN: Semi-automated software for high resolution single particle reconstructions. *J Struct. Biol.* **128**, 82-97.

[22] Mallick, S.P., Zhu, Y. and Kriegman, D. (2004) Detecting particles in cryo-EM micrographs using learned features. *J. Struct. Biol.* **145**, 52-62.

[23] Nicholson, W.V., and Glaeser, R.M. (2001) Review: Automatic particle detection in electron microscopy. *J. Struct. Biol.* **133**, 90-101.

[24] Nicholson, W.V. and Malladi, R. (2004) Correlation-based methods of automatic particle detection in electron microscopy images with smoothing by anisotropic diffusion. *J. Microsc.* **213**, 119-128.

[25] Ogura, T. and Sato, C. (2001) An automatic particle pickup method using a neural network applicable to low-contrast electron micrographs. *J. Struct. Biol.* **136**, 227-238.

[26] Ogura, T. and Sato, C. (2004) Automatic particle pickup method using a neural network has high accuracy by applying an initial weight derived from eigenimages: a new reference free method for single-particle analysis. *J. Struct. Biol.* **145**, 63-75.

[27] Plaisier, J.R., Koning, R.I., Koerten, H.K., van Heel, M. and Abrahams, J.P. (2004) TYSON: Robust searching, sorting, and selecting of single particles in electron micrographs. *J. Struct. Biol.* **145**, 76-83.

[28] Pratt, W. (1991) Digital Image Processing, *John Wiley and Sons.*

[29] Rath, B.K. and Frank, J. (2004) Fast automatic particle picking from cryo-electron micrographs using a locally normalized cross-correlation function: a case study. *J.Struct. Biol.* **145**, 84-90

[30] Roseman, A.M. (2003) Particle finding in electron micrographs using a fast local correlation algorithm. *Ultramicroscopy* **94**, 225-236.

[31] Roseman, A.M. (2004). FindEM-a fast, efficient program for automatic selection of particles from electron micrographs. *J. Struct. Biol.* **145**, 91-99.

[32] Rosenthal, P.B. and Henderson, R. (2003) Optimal determination of particle orientation, absolute hand, and contrast loss in single particle electron cryomicroscopy. *J. Mol. Biol.* **333**, 721-745.

[33] Saad, A., Chiu, W., and Thuman-Commike, P. (1998) Multiresolution approach to to automatic detection of spherical particles from electron cryomicroscopy images. *Proc. 1998 Int. Conf. Image Process.* **3**, 846-850.

[34] Short, J.M., (2004) SLEUTH - a fast computer program for automatically detecting particles in electron microscope images. *J. Struct. Biol.* **145**, 100-110.

[35] Sigworth, F. J. (2004) Classical detection theory and the cryo-EM particle selection problem. *J. Struct. Biol.* **145**, 111-122.

[36] Singh, V., Marinescu, D.C. and Baker, T.S. (2004) Image segmentation for automatic particle identification in electron micrographs based on hidden Markov random field models and expectation maximization. *J. Struct. Biol.* **145**, 123-141.

[37] Smith, J.M. and Singh, M. (1996) System for accurate one-dimensional gel analysis including high-resolution quantitative footprinting. *Biotechniques*, **20**, 1082-1087.

[38] Smith, J.M. (1999) XIMDISP - A visualization tool to aid structure determination from electron microscope images. *J. Struct. Biol.* **125**, 223-228.

[39] Thuman-Commike, P. and Chiu, W. (1995) Automatic detection of spherical particles from spot-scan electron microscopy images. *J. Micros. Soc. Amer.* **1**, 191-201.

[40] Thuman-Commike, P. and Chiu, W. (1996) PTOOL: a software package for the selection of particles from electron cryomicroscopy spot-scan images. *J. Struct. Biol.* **116**, 41-47.

[41] Volkmann, N. (2004) An approach to automated particle picking from electron micrographs based on reduced representation templates. *J. Struct. Biol.* **145**, 152-156.

[42] Wong, H.C., Chen, J., Mouche, F., Rouiller, I. and Bern, M. (2004) Model-based particle picking for cryo-electron microscopy. *J. Struct. Biol.* **145**, 157-167.

[43] Yu, Z. and Bajaj, C. (2004) Detecting circular and rectangular particles based on geometric feature detection in electron micrographs. *J. Struct. Biol.* **145**, 168-180.

[44] Zhu, Y., Carragher, B., Kriegman, D.J., Milligan, R.A. and Potter, C.S. (2001) Automated identification of filaments in cryoelectron microscopy images.*J. Struct. Biol.* **135**, 302-312.

[45] Zhu, Y., Carragher, B., Glaeser, R.M., Fellmann, D., Bajaj, C., Bern, M., Mouche, F., de Haas, F., Hall, R.J., Kriegman, D.J., Ludtke, S.J., Mallick, S.P., Penczek, P.A., Roseman, A.M., Sigworth, F.J., Volkmann, N. and Potter, C.S. (2004) Automatic particle selection: results of a comparative study. *J. Struct. Biol.* **145**, 3-14.