Pattern Recognition Letters 14 (1993) 895–905
North-Holland

November 1993

PATREC 1129

Localization of circular objects

J.P. Pascual Starink

Department of Electrical Engineering, Delft University of Technology, Delft, Netherlands

Ian T. Young

Department of Applied Physics, Delft University of Technology, Delft, Netherlands

Received 17 August 1992

Abstract

Starink, J.P.P. and I.T. Young, Localization of circular objects, Pattern Recognition Letters 14 (1993) 895–905.

Localization of circular objects is easy when all objects are isolated, but becomes more complicated when two or more objects are touching or overlapping. In this paper we discuss a method to accurately localize the centroids and obtain estimates on the sizes. The method consists of a segmentation step followed by a separation step in case of multiple-object regions. Both steps are based on region growing methods.

Keywords. Segmentation, region growing, object localization, image processing, electron microscopy, immunolabelling.

1. Introduction

Extracting object information from images using digital image analysis, requires the object to be separated from the background. This step, commonly referred to as image segmentation, is one of the crucial steps in image analysis, since the accuracy of the object measurements is closely related to the quality of the segmentation result. The goal of image segmentation is to partition the image into mutually exclusive subsets, called object regions. Each region should be uniform and homogeneous with respect to some property, such as gray-value or texture, and significantly differ from each neighbouring region. Due to various reasons such as noise, overprojection, lack of pixel resolution, etc., objects may aggregate and multiple-object regions result from the segmentation procedure. To separate the regions into their objects, sometimes simple morphological operations will suffice, e.g. touching objects may be disconnected with an opening. In this paper we present a method to recover the objects when aggregation is more severe.

The algorithm consists of two steps. The first step is the segmentation step and is based on a combined region growing and edge detection procedure. It is well known that region growing techniques may produce false boundaries, due to the strict definition of region uniformity. Usually, they are quite good in detecting the presence of a region, but show a rather poor accuracy in localizing boundaries. To solve these drawbacks, region growing may be combined with edge detection. For instance, both techniques can be combined into a decision whether a point is a boundary point or a region point [1,5]. Another approach is to

Correspondence to: J.P.P. Starink, Delft University of Technology, Faculty of Electrical Engineering, Information Theory Group – 11.08, Mekelweg 4, 2600 GA Delft, Netherlands.

obtain an initial guess for the contour by region growing, and to optimize this contour with respect to some criteria (e.g. contour smoothness) based on local edge information [8,12].

The second step of the algorithm consists of estimating the number of objects in the regions and localizing their centroids.

2. Method

Throughout the paper, the image is assumed digitized in $N \times N$ pixels, of which the gray-values are represented by $I(x, y)$ for $0 \leqslant x, y \leqslant N-1$. The underlying continuous gray-value function of the image is denoted by $f(x, y)$.

2.1. Segmentation

To be detectable, the regions should be homogeneous with respect to some property. Thus it is possible to manipulate the images such that the regions become more or less bright blobs on a dark background.

Starting from at least one point inside an object region, a region growing procedure is used to extract the region. To solve region growing problems, the edge likelihood of a pixel, derived from an edge detector and an edge strength detector, is integrated in the acceptance criterion of the region growing procedure. The outline of the segmentation scheme is shown in Figure 1.

Peak detection

The problem of locating points – the kernels – is a peak detection problem. However, we are not interested in the true peaks of the underlying gray-value function, which can be difficult to situate, but just in points located inside the regions, somewhere near the peaks. One quick and satisfying method of such a peak detection algorithm is the convergent squares algorithm (CSA). This iterative algorithm locates the peak pixel in the region of highest density. It performs best for images of convex objects, is computationally very efficient and has no empirical parameters [11]. The algorithm is straightforward in detecting the peak. It starts with the image of size $N \times N$, which is subdivided into four overlapping

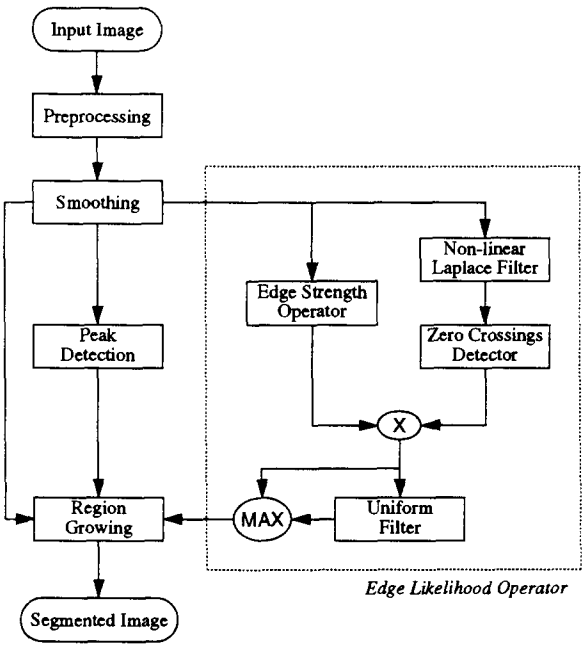


Figure 1. Outline of the segmentation scheme.

subimages of sizes $N-1 \times N-1$. The maximum density subimage is selected for the next iteration. This continues until the comparison is among four pixels, after which the peak is found.

To detect the other peaks, a circular region slightly bigger than the expected object size surrounding the peak last found is set to zero and the CSA is restarted. For this purpose, a rough estimate for the size of the objects is needed. This process is repeated until all peaks have been detected. The peaks are collected in the kernel list, that consists of the coordinates (x_i, y_i), $1 \leqslant i \leqslant m$, of the peak pixels.

Edge detection

Based on the human visual system, Marr and Hildreth [10] have developed an operator consisting of a smoothing followed by a high pass filter, in particular the Laplacian of a Gaussian

$$\nabla^2 * G_\sigma(x, y) \tag{1}$$

where ∇^2 is the classical Laplace operator and G_σ the Gaussian with standard deviation σ_g. The only parameter σ_g in this operator is kept as small as possible for the following reasons:

• High spatial accuracy requires the smallest possible smoothing filter [2].

• The width of the central positive region of the Marr–Hildreth operator is $2\sqrt{2}\sigma_g$. As such, the edge positions of smaller objects are displaced and may even be fused.

• The aim is to detect as many candidate edge pixels as possible. Their relevance towards the segmentation is tested in the region growing procedure.

The actual choice for σ_g depends on the signal-to-noise ratio of the image, although a lower bound for σ_g lies between ca. 0.8 and 1.0 [15].

The digital smoothing filter must be a good approximation of the continuous Gaussian. Since at least 47.5% of the area under the Gaussian lies between 0 and $1.96\sigma_g$, the window with w_g of the discrete Gaussian is set to $[2\sigma_g+1]$. This yields a minimum width of 5 when $\sigma_g \approx 1.0$.

Instead of using the classical Laplace filter, a non-linear filter based on local maximum and minimum filters is used [15]

$$V^2(x, y) = \mathrm{MAX}_n(x, y) + \mathrm{MIN}_n(x, y) - 2I(x, y) \quad (2)$$

where the maximum/minimum is searched in an $n \times n$ square or in a circle with radius n centered at (x, y). The advantage of this filter is that the second-order derivate is taken in the most relevant direction, i.e. perpendicular to the edge direction, while the classical Laplace operator only computes this derivate along both axes of the grid.

Zero-crossings localization

The response of the Marr–Hildreth operator is zero at the inflection of the gray-value surface. In digital images, these zero-crossings usually cannot be detected directly, because they are surrounded by positive and negative operator response. On the other hand, the transition strip may be wider than one pixel in case of ramp edges. In the presented method, the zero-crossings are localized [15] by first assigning all zero-value pixels to the nearest region, either positive or negative. In case of a tie, the pixel is assigned to the nearest negative region. The border pixels of the negative regions are then taken as the edge pixels.

Edge likelihood

To the edge pixels, an edge likelihood measure based on the gradient in the pixels is assigned. Conventional gradient operators like Sobel, Prewitt and Roberts perform rather poorly in noisy images [13]. Lee [9] proposed a filter which is less sensitive to noise. It is also based on local maximum and minimum filters

$$L_n(x, y) = \min[I(x, y) - \mathrm{MIN}_n(x, y), \mathrm{MAX}_n(x, y) + I(x, y)]. \quad (3)$$

To suppress non-ramp edges (texture and noise) to a greater extent, the ramp version of the Lee detector is used, as proposed in [16]

$$L_n^*(x, y) = \min[\mathrm{LOW}_n(x, y) - \mathrm{MIN}_n(x, y), \mathrm{MAX}_n(x, y) - \mathrm{UPP}_n(x, y)] \quad (4)$$

where

$$\mathrm{LOW}_n(x, y) = \mathrm{MAX}_n[\mathrm{MIN}_n(x, y)] \quad \text{and}$$

$$\mathrm{UPP}_n(x, y) = \mathrm{MIN}_n[\mathrm{MAX}_n(x, y)] .$$

The edge likelihood of the edge pixels is obtained by multiplying the result of the zero-crossing detector with the edge strength image. Since the zero-crossing detector has a maximum displacement of one pixel, zero-likelihood pixels are assigned an edge likelihood half the value of the maximum adjacent edge likelihood

$$E_n(x, y) = \max[L_n^*(x, y), \tfrac{1}{2}\mathrm{MAX}_n(L_n^*(x, y))] . \quad (5)$$

Region growing

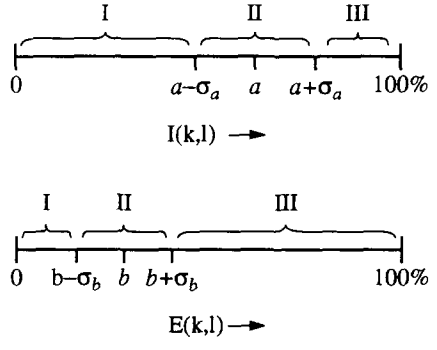
The region growing procedure starts with a region of one pixel, namely the element i of the kernel list for which $I(x_i, y_i)$ is maximal, that is removed from the kernel list. Next, its eight connective neighbours are marked as the initial set of candidate pixels. The candidates are tested against the acceptance criterion. If a candidate (k, l) passes the test, it is moved to the region, otherwise it is removed from the candidate list. The candidate set is extended with the eight connective neighbours of the newly added region pixel that are not yet region pixels or candidate pixels themselves. When no candidates are left, the

region is complete and the next maximum gray-value kernel from the kernel list is processed. Segmentation is complete when the kernel list is empty.

By applying the Marr–Hildreth operator, the gray-value of the candidate has changed to a fraction a of the original gray-value step $I(x_i, y_i)$, and the edge likelihood to a fraction b of this value. The values a and b and their standard deviations σ_a and σ_b mainly depend on the edge type (step, ramp), the widths of the Gaussian (w_g) and the nonlinear Laplace filter (w_l), and the noise.

For the acceptance criterion, the gray-value $I(k, l)$ and edge likelihood $E(k, l)$ are classified according to Figure 2(a). Depending on the resulting classes, Figure 2(b) shows the decision scheme. The decision is one of the following:

1. Reject. The candidate is outside the region.
2. Accept as region pixel. The candidate is added to the region and its neighbours become candidates.
3. Accept as boundary pixel. The pixel is on the region's boundary and is accepted as region pixel.



a. parameter intervals

I(k,l)

E(k,l)	I	II	III
I	reject	region	region
II	boundary	**rule**	region
III	boundary	boundary	boundary

b. decision scheme

Figure 2. Defining three intervals for a candidate's gray-value and edge likelihood (a) used for its classification (b).

However, its neighbours do not become new candidates.

4. In case of statistical doubt, the following rule is applied

$$I(k, l) - \frac{a}{b} E(k, l) \begin{cases} > 0, & \text{region pixel,} \\ \leqslant 0, & \text{boundary pixel.} \end{cases} \quad (6)$$

2.2. Separation

Each region in the segmented image either contains only one object or it contains more objects. If it holds only one object, the centroid is readily estimated by calculating the mean position for all pixels in the region. In the case of a multiple-object region, first the number of objects in the region and estimates for their positions and sizes are determined.

Since the objects are circular, their inscribing circles serve as initial estimates. These circles are identified by the peak values in the distance image [4]. The peak pixel marks the circle centroid, its distance value the circle radius. The number of distance peaks probably exceeds the number of objects really present, due to noise, object irregularities, etc. Therefore, a subset has to be selected. The number of selected peaks then equals the number of objects in the region.

An estimate for the object size is known (used in the CSA), so an estimate for the radius is also known. Thresholding the peak values against this value eliminates the objects that are too small. After selecting a peak, all peaks covered by the corresponding inscribing circle are removed from the kernel list, since all object centers are assumed to be uncovered.

One simple peak selection procedure is to select the peaks in descending order of magnitude. A problem with this procedure is that the maximum peak does not necessarily represent an object center, by which the inscribing circle may cover other, true peaks and disturb the region lay-out. The effect mainly occurs at the 'end objects' of the region: objects having only one neighbour. This problem is avoided by identifying the end objects. Considering the distance image as a mountain range, then the mountain ridge is obtained by calculating the distance skelet. The end points of this skelet are then used to recognize the end objects as peaks that are closer to an end point than to the background. From these, select the peaks in ascending order of distance to the end point. After

detecting the end objects, the remaining peaks are selected in descending order of distance to the background.

There are two reasons why the selected peaks should be treated as initial centroids. First, for blobs the centroid does not coincide with the center of the inscribing circle, and second, not all information contained in the region has been used, since the inscribing circles do not cover the entire region. The region contains *free* pixels, which are used to update the centroid and the object lay-out by assigning them to the objects. The pixels in the region belong to one of these classes:

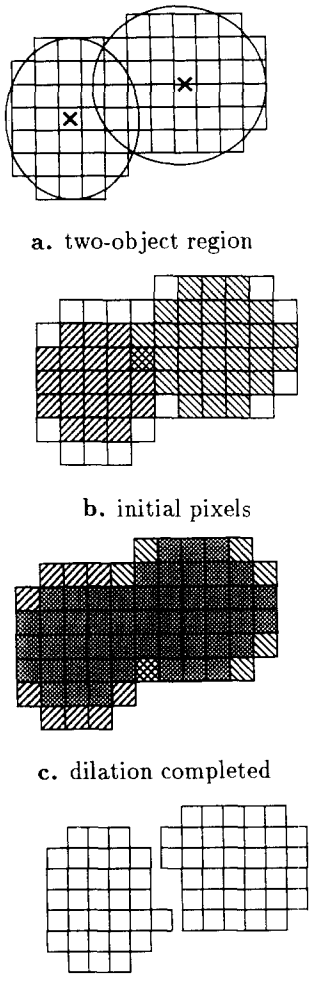
- The *initial* pixels belong to the objects' inscribing circles.
- Pixels belonging exclusively to one object are the *unique* pixels.
- *Shared* pixels are assigned to more than one object.

The initial pixels can belong to more than one object, since the objects are allowed to overlap. The free pixels are obtained by removing all assigned pixels from the region. Each object, starting with the inscribing circle, is dilated. The dilated region is checked against the set of free pixels. When it includes free pixels, the object status remains incomplete, otherwise its status becomes complete and is no longer allowed to dilate. If a pixel is found in one dilation only, it is classified unique, otherwise it is assigned to the corresponding objects as a shared pixel. Then the set of free pixels is determined and the procedure starts again. The iteration stops when all objects are complete. The process is illustrated in Figure 3.

The pixels are used to update the object centroid, but with different weights. The initial and unique pixels do positively belong to the object, so their weight factors are 1, $w_i = w_u = 1$. For the shared pixels however, there is an uncertainty to which object(s) they belong. The probability of a pixel in the dilated region being unique is estimated by the number of unique pixels divided by the total number of pixels. If this probability is assumed the same for all pixels, then w_s equals this probability, given by

$$w_s = \frac{n_i + n_u}{n_i + n_u + n_s}$$

where n_i, n_u and n_s represent the number of initial, unique and shared pixels in the object. With $\boldsymbol{c}_i$, $\boldsymbol{c}_u$ and



a. two-object region

b. initial pixels

c. dilation completed

d. resulting objects

Figure 3. Example of the separation of a two-object region (a). The initial pixels of the inscribing circles are shown in (b). The free pixels are assigned to the objects (c), resulting in two complete objects (d).

$\boldsymbol{c}_s$ being the centroids of these sets, the new centroid of the object is given by

$$\frac{1}{n_i + n_u + w_s n_s} (\boldsymbol{c}_i + \boldsymbol{c}_u + w_s \boldsymbol{c}_s) . \quad (7)$$

This completes the update for the object's centroid. To stabilize the centroids, the update procedure can be iterated until the shift of the centroid is smaller than some number, for example 1/50th of the object radius.

3. Experiments and discussion

This section describes two series of experiments. The first series was performed to determine the parameters of the region growing procedure for different edge types in the presence of noise. In the second series of experiments, the positional error of both the segmentation and separation step are determined. Finally, a practical example will be given on the localization of immunogold labelled proteins.

In general, the signal-to-noise ratio (SNR) is determined from the gray-value step h of the edge and the standard deviation σ_n of the additive noise (zero mean) with

$$\mathrm{SNR} = h^2/\sigma_n^2 . \tag{8}$$

In all experiments, the gray-value of the background was set 100 and the gray-value of the object 200, by which $h = 100$ and $\mathrm{SNR} = (100/\sigma_n)^2$.

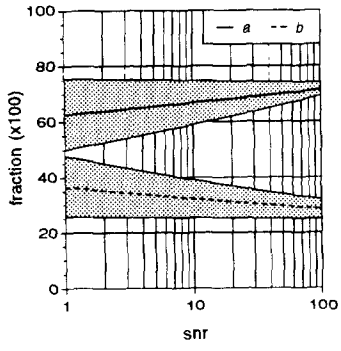
The localization procedure is controlled by four parameters: σ_n, σ_g and the filter widths w_g and w_l. The parameter choice is directed by the SNR of the image. In [15], the optimal parameter choices for σ_g and the width of the Laplacian were determined experimentally. Based on these results, the configurations of the parameters in our experiments were set for two ranges of the SNR (Table 1). Given the SNR, the optimal values for w_l and σ_g and therefore w_g are given.

3.1. Region growing parameters

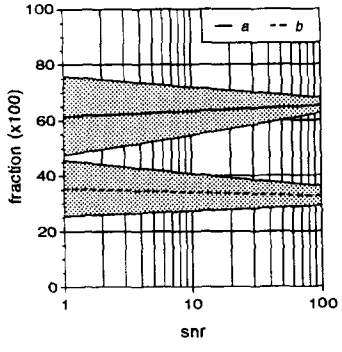
To determine the parameters a and b and their standard deviations σ_a and σ_b, a one-dimensional step edge of height 100 was processed. Besides a step edge, four different types of (linear) ramp edges were generated by blurring the step edge with filters of sizes $w_u = 3, 5, 7, 9$. Normal distributed noise with zero mean was added to the signal. The parameters were determined from 1000 realizations of the signal with additive noise according to Table 1. The edge point was assumed to be located at the zero-crossing closest to the true boundary point. In that element, the gray-value and edge likelihood were collected. The resulting configurations of the region growing parameters are given in Figure 4. The parameters in absence of noise are listed in Table 2. Examining Figure 4, the following remarks can be made:

Table 1
The locally optimal configurations of the smoothing parameter σ_g and the widths of the discrete Gaussian w_g and Laplacian w_l given the signal-to-noise ratio SNR

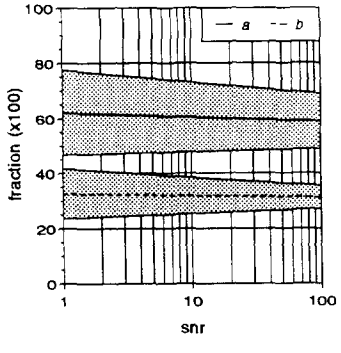
SNR	1	2	3	4	5	6	7	8	9	10	20	30	40	50	60	70	80	90	100
σ_g	3.1	2.3	2.0	1.8	1.7	1.6	1.5	1.4	1.4	1.3	1.2	1.1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
w_l	7	7	7	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
w_g	13	11	9	9	7	7	7	7	7	7	7	7	5	5	5	5	5	5	5
σ_n	100	71	58	50	45	41	38	35	33	32	22	18	16	14	13	12	11	10	10



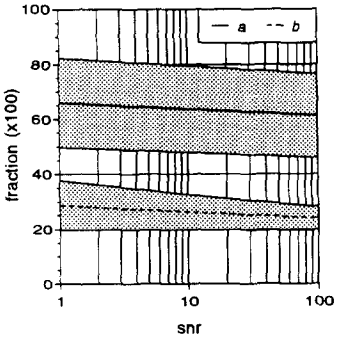
a. width 1 (step edge)



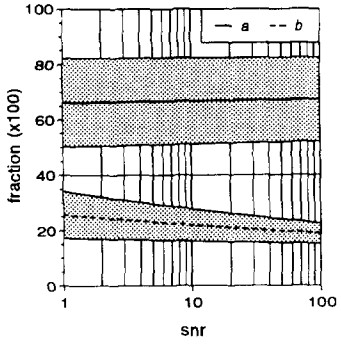
b. width 3



c. width 5



d. width 7



e. width 9

Figure 4. The region growing parameters as determined for a step edge blurred with a uniform filter of different widths (1, 3, 5, 7 and 9) for SNRs in the interval 1, ..., 100.

- As expected, in all cases the standard deviations σ_a and σ_b decrease as the SNR increases.
- Although it is expected that a and b stay constant over the SNR interval, they do vary. Parameter b decreases in all cases as noise is reduced, while there is no such trend for a.

Table 2
The region growing parameters in the noiseless case for a step edge (width 1) and 4 linear ramp edges (width 3, 5, 7 and 9)

	1	3	5	7	9
a	0.70	0.65	0.60	0.57	0.56
b	0.30	0.33	0.32	0.28	0.22

• Less noise corruption makes b to vary less. This trend is not influenced by the edge width, in contrary to σ_a which increases with edge width for high SNR values. For width 9, this parameter even seems to be constant over the entire noise range. This is caused by the fact that for wide edges the gray-value step within the discrete Gaussian window is smaller than for narrow edges. Since the noise is the same for both edges, the wide edges will be corrupted more than the narrow edges (compare the definition of the signal-to-noise ratio, equation (8)).

These remarks indicate that parameter a is more sensitive to noise and to the edge width than parameter b. This sensitivity can be reduced by making parameter σ_g also depend on the edge width. For instance, if the edge width is n, the image may be smoothed with the operator $G_\sigma * u_n$ instead of G_σ.

3.2. Positional error

Both the segmentation and separation step cause a positional error σ_{pos} in the objects' centroids, σ_{seg} and σ_{sep}, respectively. The goal of the experiments described in this section is to quantify these errors.

For the experiments, artificial blobs were used, which were generated by manually drawing circles on a computer screen with a mouse. In total, 50 different blobs were created. All experiments were repeated for three different blob sizes: 25, 50 and 75 pixels (r=2.8, 4.0 and 4.9 pixels), and, if applicable, for the 19 different signal-to-noise ratios as listed in Table 1. For each configuration, the positional error, defined as the square root of the mean squared euclidian distance between the estimated centroid and the true centroid, was obtained from 250 blobs.

First, σ_{seg} was determined by randomly distributing the blobs over the image such that all blobs were isolated. By this, only the segmentation procedure is needed to obtain estimates for the blob centroids, since all regions are single-object regions. The results are plotted in Figure 5.

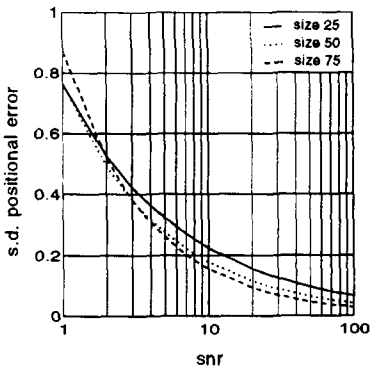


Figure 5. Positional error of objects' centroids in single-object regions as a function of the signal-to-noise ratio for three different object sizes.

As expected, the positional error decreases as the SNR increases for all sizes. Since more noise disturbs the segmentation result to a greater extent, the position error should be larger for smaller objects, which is true, however, only for SNR > 5.

In the next experiment, images containing two object regions were used. First, images were generated without adding noise, in which case there is no need for segmentation, since a simple threshold at gray-value 150 will do a perfect job. The resulting error is due entirely to the separation procedure (Table 3).

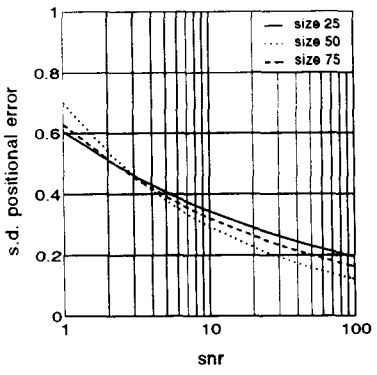
Next, noise was added to the images. The resulting σ_{pos} after the separation then is a combination of the errors due to segmentation and separation. The positional errors are plotted in Figure 6(a). This two-object region experiment was repeated for three-object regions. The results of this experiment are shown in Table 4 and Figure 6(b).

The error σ_{seg} in multi-object regions can be estimated with

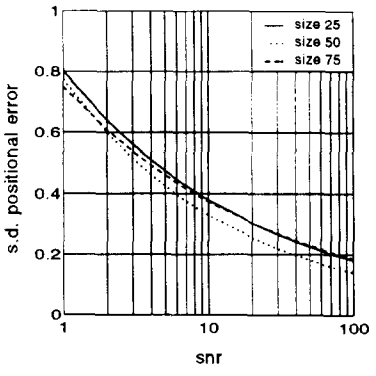
$$\sigma_{seg} = (\sigma_{pos}^2 - \sigma_{sep}^2)^{1/2}$$

Table 3
Positional error of the objects in noiseless two-object regions due to the separation procedure

object size	25	50	75
σ_{pos}	0.15	0.17	0.21



a. two-object regions



b. three-object regions

Figure 6. For two-object (a) and three-object (b) regions, the positional error centroids as a function of the signal-to-noise ratio for three different object sizes.

Table 4
Positional error of the objects in noiseless three-object regions

object size	25	50	75
σ_{pos}	0.15	0.17	0.21

when both errors are statistically independent. Although this is probably not the case, especially for the noisier images, it is the only way to obtain a rough estimate on this parameter.

The following remarks can be made from examining Figures 5 and 6:

- For low SNRs, single-object regions show a larger positional error than the two- and three-object regions, even though the error is only due to the segmentation step. However, as noise reduces, the positional error of the single-object regions is overtaken by the other two and ends at ca. one third of their minimum.
- The positional error in two-object regions is smaller than the error in three-object regions.
- In general, the size and the number of objects per region do not seem to influence the localization very much. Therefore, the error in the localization of four or more objects in a region will probably not deviate too much from that of three-object regions.

3.3. Practical example

In the field of cell biology, immunolabelling has become an important technique to relate proteins or nucleic acids to their locations within the cell. With antibodies raised against these proteins, their locations can be visualized by light microscopy using fluorescent labels, or by electron microscopy using colloidal gold particles.

For this practicle example, aggregation competent cells of the slime mould *Dictyostelium discoideum* were fixed with formaldehyde and glutaraldehyde and prepared for cryosectioning according to Tokuyasu [14] as described previously [6]. The cryosections were labelled with a monoclonal antibody for the crystal protein, the major component of a protein crystal enclosed in the endoplasmatic reticulum [3]. The primary antibody is detected by protein A bound to 10 nm colloidal gold particles (Figure 7(a)). The micrographs were recorded with a Philips EM420 transmission electron microscope on sheet film (Agfa Scientia 23D56) at a primary magnification of 105,000.

The sheet films were placed on a dazzle-light and digitized using a Sony XC-77CE CCD camera coupled to an Imaging Technology VFG frame grabber in a PC-AT compatible. Image processing was performed on a SUN SPARC/2 workstation with the image processing package SCIL-Image. Preprocessing the image towards bright blobs on a dark background consisted of (Figure 7(b)):

-*Noise reduction.* The noise peaks were removed by applying the NOise Peak ELimination filter [7] three times. This filter replaces the center pixel of a 3×3 window by the one but highest (lowest) gray-

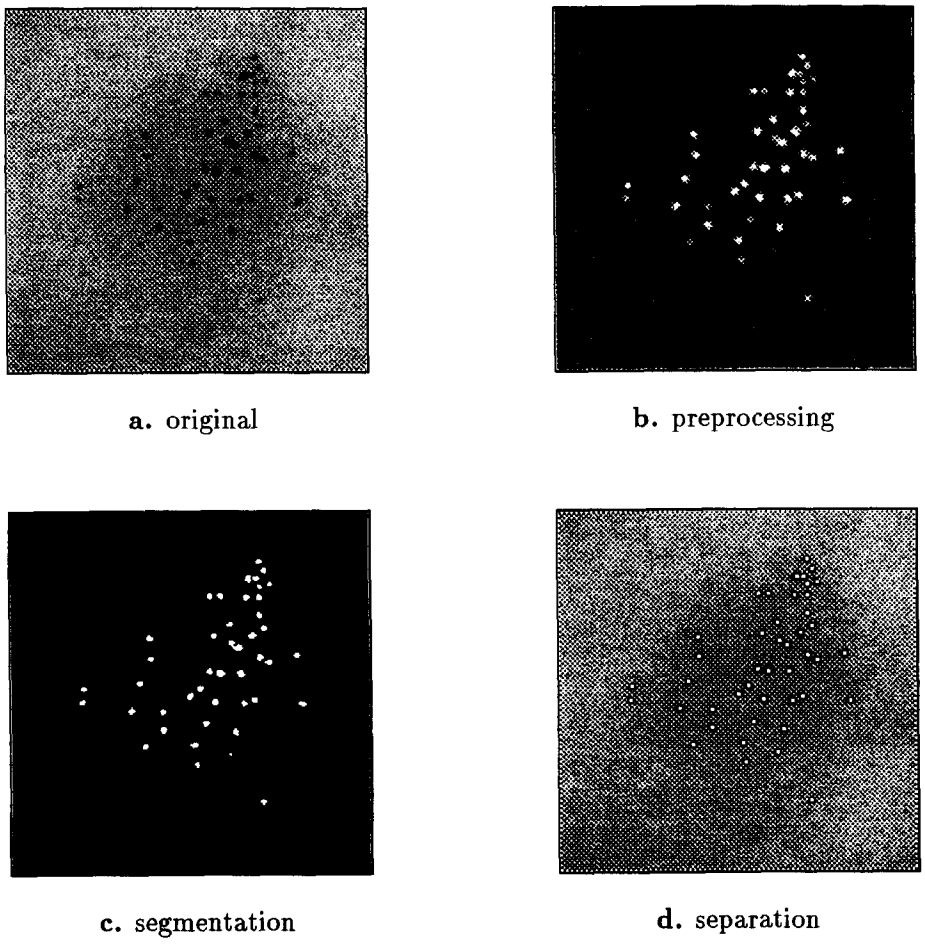


Figure 7. Electron microscope image of the immunogold labelled crystal protein of *Dictyostelium discoideum* at a magnification of 105,100 (a). After preprocessing (noise peak elimination and background subtraction), bright spots on a dark background remain (b). The segmentation procedure was performed with $\sigma_g = 1.1$, implying $w_g = 5$ and $w_l = 7$ (c). The resulting centroids are shown overlaid on the original image in (d).

value if it holds the highest (lowest) gray-value in the window.

–*Background subtraction.* The image was smoothed with $G_{1.4}$. The upper envelope was determined by applying a maximum followed by a minimum filter. Both filters were circular shaped with a radius of 9 pixels. The original image was subtracted from this upper envelope and clipped against 0 and 255.

Finally, the centroids of the gold particles were localized using the method described, segmentation (Figure 7(c)) and separation (Figure 7(d)).

Acknowledgments

We would like to thank B.M. Humbel for kindly providing and preparing the biological materials used for the practical example.

This work was partially supported by "The Netherlands" Project Team for Computer Science Research (SPIN)", project "Three-Dimensional Image Analysis".

References

[1] Bajcsy, M., M. Mintz and E. Liebman (1986). *A common framework for edge detection and region growing.* Univ. Pennsylvania GRASP Lab. Tech. Rep. 61.

[2] Berzins, V. (1984). Accuracy of laplacian edge detectors. *Computer Vision, Graphics, and Image Processing* 27, 195–210.

[3] Bomblies, L., E. Biegelmann, V. Döring, G. Gerisch, H. Krafft-Czepa, A.A. Noegel, M. Schleicher and B.M. Humbel (1990). Membrane enclosed crystals in *Dictyostelium discoideum* cells, consisting of developmentally regulated proteins with sequence similarities to known ertouses. *J. Cell Biol.* 110, 669–679.

[4] Borgefors, G. (1986). Distance transformations in digital images. *Computer Graphics and Image Processing* 34, 344–371.

[5] Gerbrands, J.J., F. Booman and J.H.C. Reiber (1979). Computer analysis of moving radiopaque markers from x-ray cinefilms. *Computer Graphics and Image Processing* 11, 35–48.

[6] Humbel, B.M. and E. Biegelmann. A preparation protocol for postembedding immunoelectron microscopy of *dictyostelium discoideum* cells with monoclonal antibodies. *Scanning Microsc.*, submitted.

[7] Immi, M. (1991). A noise peak elimination filter. *CVGIP: Graphical Models Image Processing* 53, 204–211.

[8] Kass, M., A. Witkin and D. Terzopoulos (1987). Snakes: active contour models. *Proc. 1st Internat. Conf. Computer Vision*, Paris, 259–267.

[9] Lee, J.S.L., R.M. Haralick and L.S. Shapiro (1986). Morphologic edge detection. *Proc. 8th Internat. Conf. Pattern Recognition*, Paris, 369–373.

[10] Marr, D. and E.C. Hildreth (1980). Theory of edge detection. *Proc. Roy. Soc. London Ser. B* 207, 187–217.

[11] O'Gorman, L. and A.C. Sanderson (1984). The convergent squares algorithm: an efficient method for locating peaks in multidimensions. *IEEE Trans. Pattern Anal. Machine Intell.* 6, 280–287.

[12] Pavlidis, T. and Y. Liow (1990). Integrating region growing and edge detection. *IEEE Trans. Pattern Anal. Machine Intell.* 12, 225–233.

[13] Peli, T. and D. Malah (1982). A study on edge detection algorithms. *Computer Graphics and Image Processing* 20, 1–21.

[14] Tokuyasu, K.T. (1973). A technique for ultracryotomy of cell suspensions and tissues. *J. Cell Biol.* 57, 557–565.

[15] Van Vliet, L.J., I.T. Young and A.L.D. Beckers (1989). A nonlinear laplace operator as edge detector in noisy images. *Computer Vision, Graphics, and Image Proccessing* 45, 167–195.

[16] Verbeek, P.W., H.A. Vrooman and L.J. van Vliet (1988). Low level image processing by max-min filters. *Signal Process.* 15, 249–258.