Yecheng Wu[1]
Peter F. Lemkin[2]
Kyle Upton[3]

[1]**Scanalytics/CSPI, 40 Linnell Circle, Billerica, MA**
[2]**Image Processing Section, NCI/FCRDC, Frederick, MD**
[3]**Program Resources Inc., FCRDC, Frederick, MD**

A fast spot segmentation algorithm for two-dimensional gel electrophoresis analysis

An important issue in the automation of two-dimensional gel electrophoresis image analysis is the detection and quantification of protein spots. A spot segmentation algorithm must detect, define the extent of, and measure the integrated density of spots under a wide variety of actual gel image conditions. Besides these functions, the algorithm must be memory efficient to be able to process very large gel images and do this in a reasonable amount of computation time on low-cost computers, such as workstations and personal computers. We have developed a fast spot segmentation algorithm, extending the GELLAB-II segmenter, which extracts spots in a single raster scanning pass through the gel image. The performance analysis of the algorithm will be given in the paper as well as a discussion of the algorithm.

1 Introduction

In the past decade, high resolution two-dimensional polyacrylamide gel electrophoresis (2-D PAGE) has become an important research tool within the biochemical community. The two-dimensional gel electrophoresis protein separation method provides a powerful technique for protein research in molecular biology and medical diagnostics by allowing separation of thousands of proteins and polypeptides according to charge and molecular weight [1–3]. The large number of detectable proteins and the complexity of the images obtained by this technique have made it necessary to develop powerful computer systems to automate the analysis of these images [4–9]. An important issue in the automation of two-dimensional gel electrophoresis image analysis is the detection and quantification of protein spots. This issue has drawn the strong attention of many researchers in the past years [10–15]. All the methods previously developed were primarily based on the use of first or second derivatives for peak or boundary detection. There still exists a need to have a high performance spot segmentation algorithm to handle today's large number of high resolution gels with low-cost personal computers or workstations. A spot segmentation algorithm must detect, define the extent of, and measure the integrated density of spots under a wide variety of actual gel image conditions. Besides these functions, the algorithm must be memory efficient to be able to process very large gel images and do this in a reasonable amount of computation time on low-cost computers, such as workstations and personal computers. We have developed a fast spot segmentation algorithm, extending the GELLAB-II segmenter, which extracts spots in a single raster scanning pass through the gel image.

2 Methods

2.1 Image preprocessing

Image preprocessing functions include median filtering, Gaussian convolution smoothing, background correction and image rotation [16]. The median filter and the convolution filters are used to remove high-frequency image noise that has been introduced by the image acquisition device, gel sample autoradiograph or staining process, and other sources. A fast background correction subtracts inhomogeneous image background from an image so that the spots are more detectable. The filter size of the background correction algorithm is particularly designed to match the sizes of spots so that most of the streaks can be removed from the image as well as the inhomogeneous image background. Since the second derivative or Laplacian is used in the spot detection, it tends to amplify image noise. Therefore it is necessary to smooth the raw image to remove high frequency noise before computing the Laplacian. Convolution-type smoothing filters are implemented for this purpose. This two-stage filtering process is called Laplacian-of-Gaussian filtering [19], and was used in the original GELLAB segmenter [3, 10].

2.2 Spot segmentation

2.2.1 Current GELLAB-II segmentation algorithm

The segmentation algorithm is a sequence of procedures applied to a gel image to detect, define the extent and measure the density of spots. In our previous work, the second derivative generated from the gel image has been used for spot detection [3, 10]. We will briefly review the algorithm here to provide some background information about the segmentation algorithm. First, the second derivative is calculated for the gel image and a central core image is constructed based on the negative region of the second derivative where the directions of the Laplacian must be less than 0 in both x and y directions. A separate magnitude of the Laplacian image is also calculated. The central core image is a binary image where 1 indicates a negative second derivative and 0 otherwise. The regions with a pixel value of 1 will be considered as spot central core area. Since the second derivative tends to amplify image noise in spot detection, the Gaussian smoothing filter must be applied to the original image before calculating the central core image. Spots are then extracted on a spot by spot basis using the central core image. This is done by using two stacks to keep track of all the pixels which are four-neighbor connected within a

Correspondence: Dr. Yecheng Wu, Scanalytics/CSPI, 40 Linnell Circle, Billerica, MA 01821, USA

Abbreviations: dpi, dots per inch

0173-0835/93/1212-1351 $5.00+.25/0

spot extent. It is obvious that the spot extraction process is time-consuming because for each pixel of value 1 within the image the four-neighbor pixels are checked for their connectivity and all candidates are saved in one stack and expanded to another stack. The pixels for each spot are then encoded to differentiate one spot from neighbor spots. Propagation is then performed for each spot using some heuristic rules based on image values, second derivative and the second derivative magnitude values. Spot features are calculated using the propagated extents of spots. The features include spot area, spot integrated density, minimum enclosing rectangle, and shape information. A background density correction is then performed using spot information and a large size averaging filter to generate the background image.

2.2.2 New fast spot segmentation algorithm

The fast spot segmentation algorithm uses the second derivative to generate the central core image similar to our earlier algorithm [3, 10]. The difference is in the spot extraction procedure which currently encodes the entire image in a single raster scanning pass to extract spots on the fly to avoid the time-consuming spot expansion procedure. The algorithm is described as follows: (1) Creation of the central core image and second derivative magnitude image. The central core image is constructed based on the second derivative calculation of the original image. If, at a location, the second derivatives in both column and row directions are negative, the pixel in the central core image is assigned the value of 1. Otherwise it is assigned 0. The second derivative magnitude image is created at the same time by adding the absolute value of the second derivatives in both column and row directions. Since the central core image will later be used to encode the spot regions, we use 16-bit pixels for the image buffer so it can hold up to a maximum of 65535 spots in one image. This maximum number was large enough in all of our applications. More bits can be used for the image buffer if more spots are present in the gels; however, this is not likely in the foreseeable future. (2) Spot extraction: This is the core procedure which encodes the central core image with spot numbers in a single raster scanning pass through the image. After this process, the central core image has been sugmented with spot regions marked by their unique identification numbers. In other words, pixels of a spot have the same code value in the encoded image. The spot extraction algorithm treats each image scanning line in a similar way. Here we give a description of the method to encode one image line or row. For a current row, check each pixel with the following rules.

(2.1) If the value of the pixel is background value 0, it is not a spot central core pixel. Continue to the next pixel.
(2.2) If the value of the pixel is 1 and its top and left neighbors are both 0, encode this pixel with a new spot identification number to it. Then move to the next pixel to the right (Fig. 1a).
(2.3) If the value of the pixel is 1, and its top neighbor has a spot identification number (*i.e.*, > 1) and its left neighbor is a background pixel, encode the pixel with the spot identification number from its top neighbor pixel. Then move to the next pixel to the right (Fig. 1b).
(2.4) If the value of the pixel is 1, and its top neighbor pixel is a background pixel and its left neighbor pixel has a spot identification number, encode the pixel with the spot identification number from its left neighbor pixel. Then move to next pixel to the right (Fig. 1c).
(2.5) If the value of the pixel is 1 and both its top and left neighbors have spot identification numbers, encode the pixel with the spot identification number from its top neighbor pixel and then set the spot numbers of the top and left neighbor pixels to be equivalent. The equivalent spot identification numbers are later mapped to the same identification number (Fig. 1d).
(2.6) Repeat steps (2.1) through (2.5) for all the pixels in the image line.

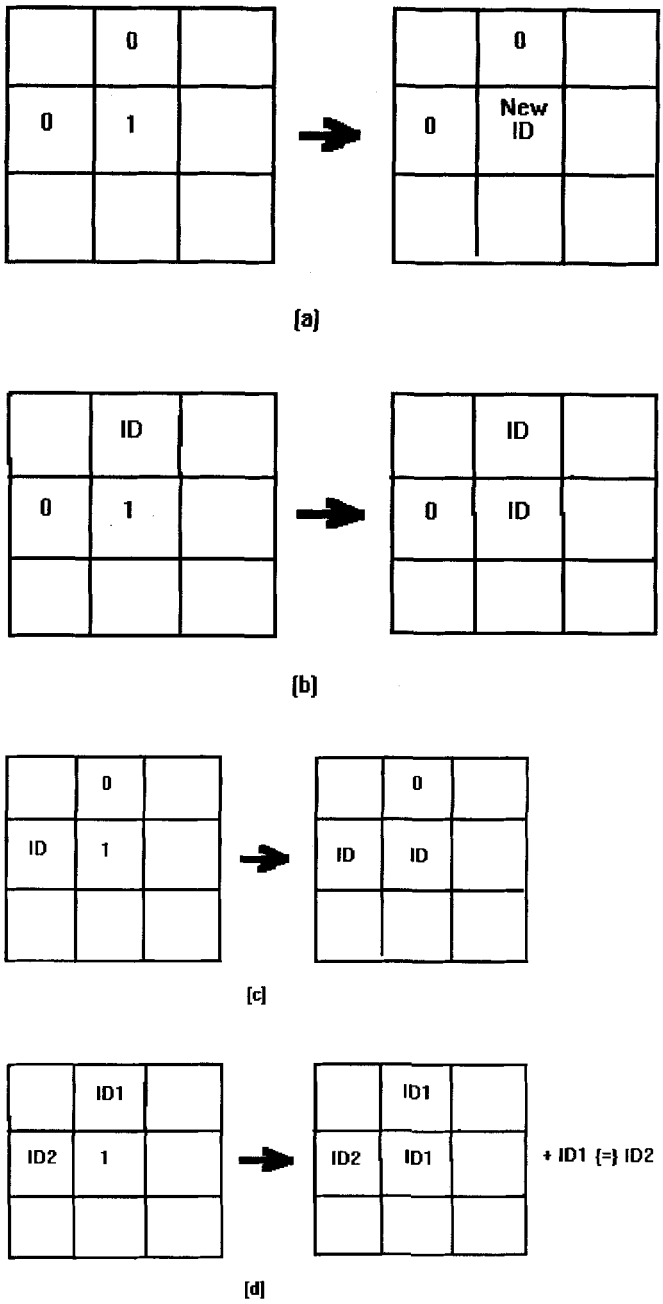

Figure 1. Rules to encode a row of a central core image.

This procedure is performed for all lines in the image to encode the gel's central core image with the spot identification numbers where spot regions of pixels are marked with the same identification number.

(3) Spot propagation is performed on the segmented central core image to find the maximum extent of all spots. Final spot boundaries are generated at this stage and saved for spot quantification, feature calculations and future possible editing. The propagation is done on a spot by spot basis using the minimum enclosing rectangle derived from the encoded central core image. The following describes the steps to do the propagation for a spot.
(3.1) Search the minimum enclosing rectangle of the spot and get all boundary pixels for this spot.
(3.2) For each boundary pixel, check all its eight connected neighbor pixels. If any of its neighbor pixels meet the following criteria, it is marked using the current spot identification number, which means the pixel belongs to the spot. The criteria are: (i) The pixel is a background pixel. (ii) The image intensity value decreases when moving from the spot boundary pixel to the background pixel. (iii) The second derivative magnitude increases when moving from the spot boundary pixel to the background pixel. The newly propagated pixel is then marked as a boundary pixel and checked by the same criteria.
(3.3) Repeat step (3.2) until al the boundary pixels for the spot are checked. Record the boundary points of the spot.
(3.4) Repeat steps (3.1) through (3.3) until all the spots in the gel are processed.

(4) Calculation of spot features. The spot features calculated include spot area, spot integrated optical density, area normalized optical density, spot centroid, spot boundary, minimum and maximum density, minimum enclosing rectangle and some other geometric features. Densitometry calibration coefficients are applied at this time to obtain the actual optical density values from the image data.

3 Results

The fast segmentation algorithm is applied to two two-dimensional gel images as shown in Fig. 2 and 4. The first gel (Fig. 2) was a leukemia autoradiograph gel digitized into a 512 by 512 image using a Vidicon camera [17]. Figure 3 shows a zoomed subregion of the gel with spot segmentation results. The second gel is one of a group of gels generously provided to us by Dr. Jim Myrick at Centers for Disease Control, Atlanta, USA. The silver stained gels are derived from human urine samples and details are described in [18]. The gel shown in Fig. 4 is digitized using Scanalytics/CSPI's desktop

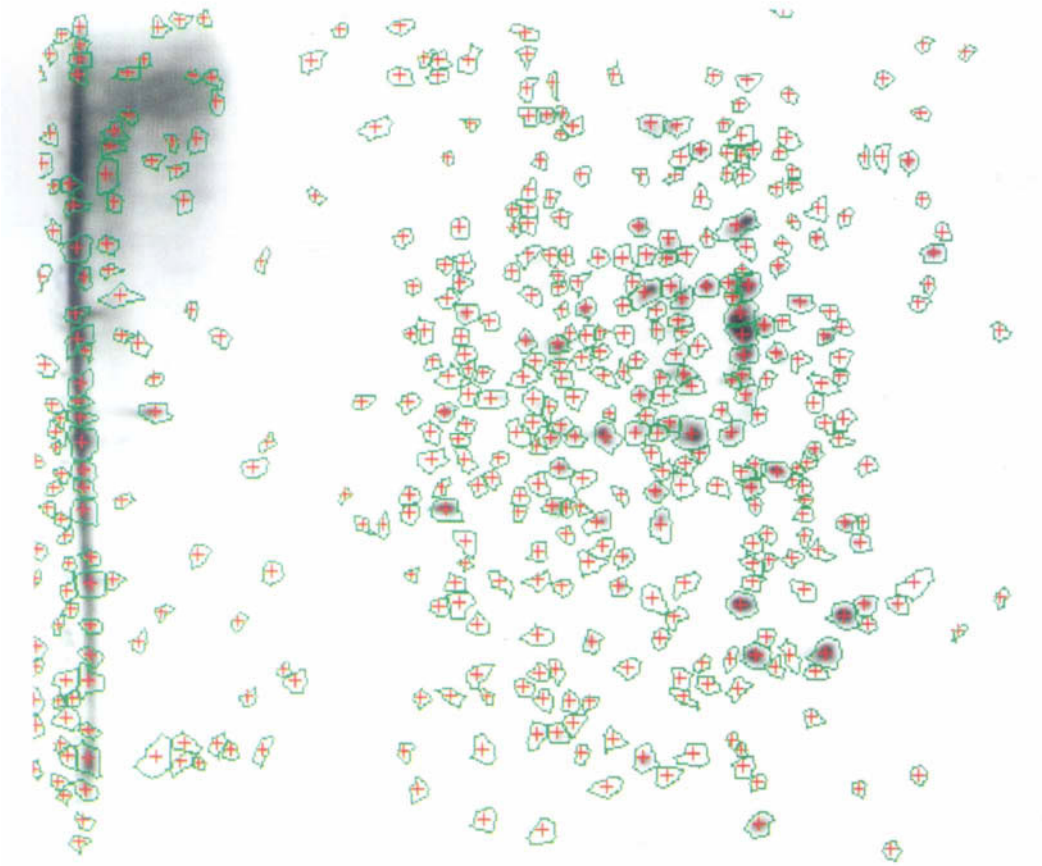

Figure 2. Leukemia autoradiograph gel digitized into a 512 by 512 image using a Vidicon camera, as described in [17].

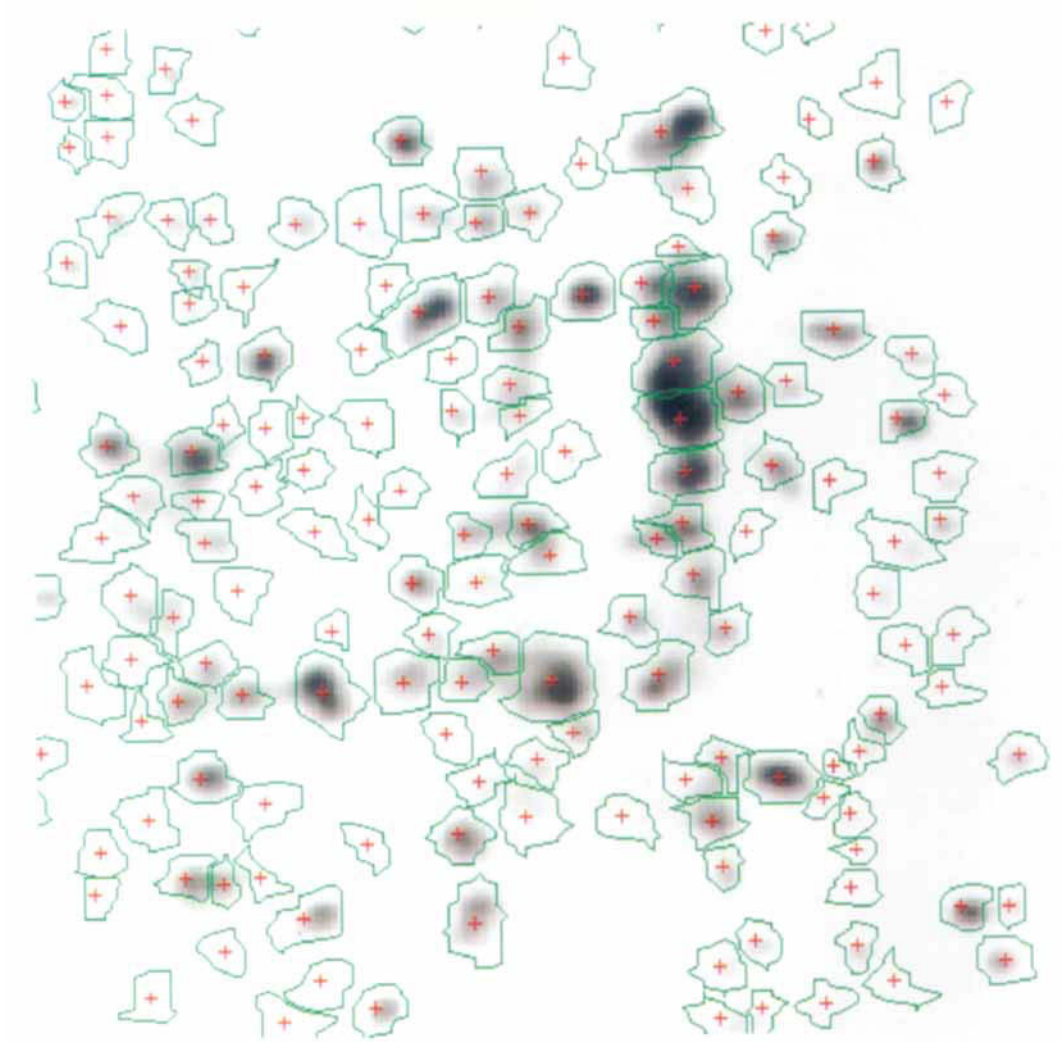

Figure 3. Zoomed subregion of Fig. 2 to show the details of spot segmentation results.

scanner at spatial resolution of 200 dpi (dots per inch) and is the size of 1024 by 1024 pixels. Figure 5 shows a zoomed subregion of Fig. 4.

There is no restriction in terms of image size as the algorithm was implemented to handle variable size and variable resolution (for example, 8-bit, 12-bit and 16-bit) images. The computing time on a 486DX 66MHZ PC for the 512 by 512 image described here is about 9 s. There are more than 600 spots detected and processed for this image, as shown in Fig. 2. One paper has reported the performance of 190 s for its spot detection and quantification program applied on a 512 by 512 size image, running on a Sun Sparcstation that is comparable to the computer we used [13]. By comparison, our algorithm appears to be at least one order of magnitude faster than the one reported. We also applied the algorithm on the second gel, of size 1024 by 1024 pixels. The entire segmentation process takes about 28 s with more than 1000 spots detected and quantitated. With such a performance, the spot detection and quantification can be done interactively on a conventional 486 personal computer or a workstation. Since the algorithm is so fast, the parameters selected for filtering can be interactively adjusted.

In addition to the analysis of computation efficiency, we have also studied the quality of the segmentation algorithm. The results from the new segmentation algorithm and the previous one were compared. The comparison shows that the new algorithm is consistent with the previous method.

4 Discussion

We have described a fast image segmentation algorithm for spot detection and quantification. The algorithm is shape independent and performs spot quantification using the exact spot boundaries detected from a gel image. The procedure utilizes the second derivative of a gel image to encode spot regions in a single raster scanning pass through the entire image. The algorithm is robust and the speed-up is significant in comparison to the previously used algorithm and other work. This increased performance makes it more practical to use low-cost personal computers for the processing and analysis of a large number of high resolution two-dimensional gels.

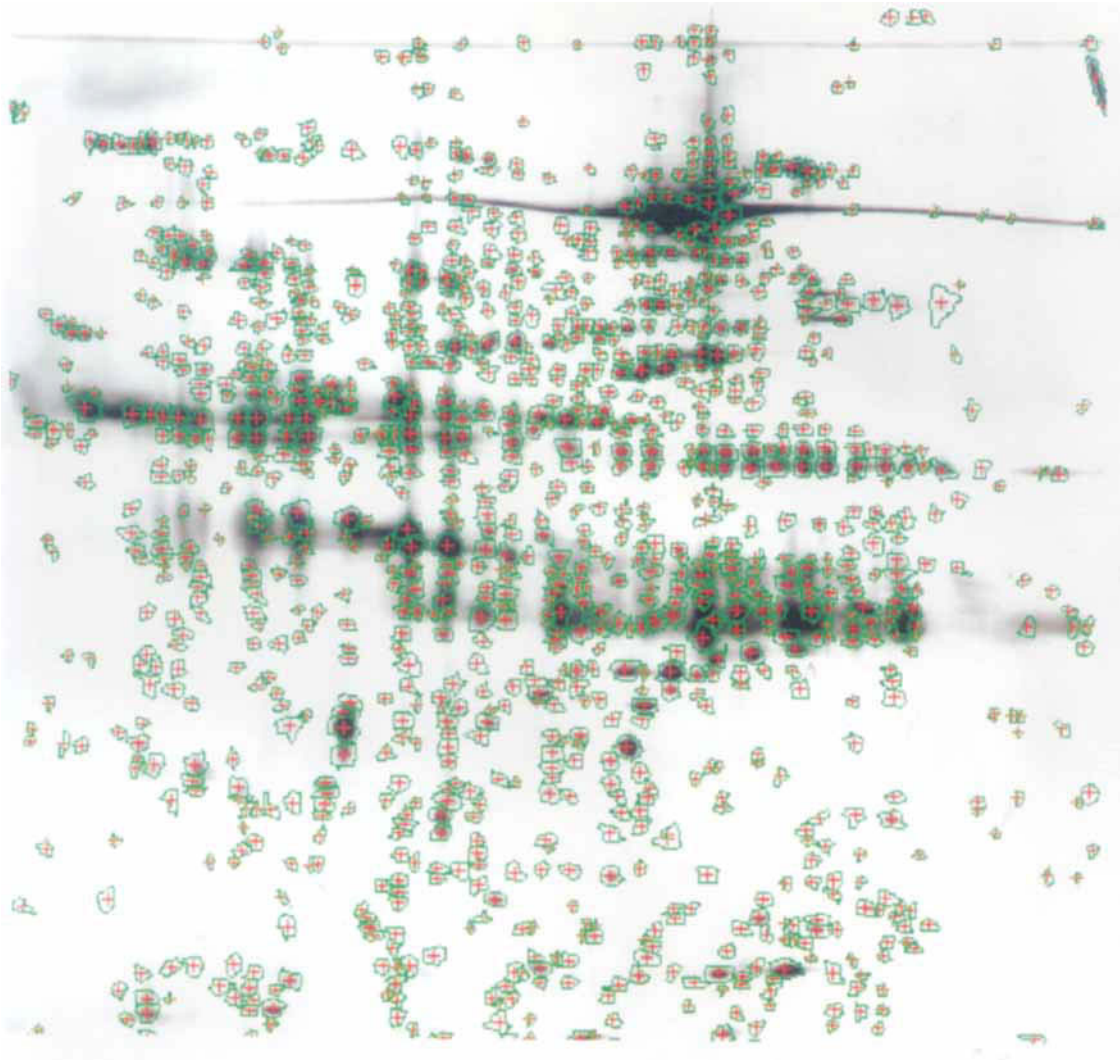

Figure 4. Gel derived from human urine samples. It is digitized using Scanalytics/CSPI's desktop scanner at a spatial resolution of 200 dpi. The segmentation takes about 28 s on this 1024 by 1024 size image and the spots and their boundaries are displayed in the figure.

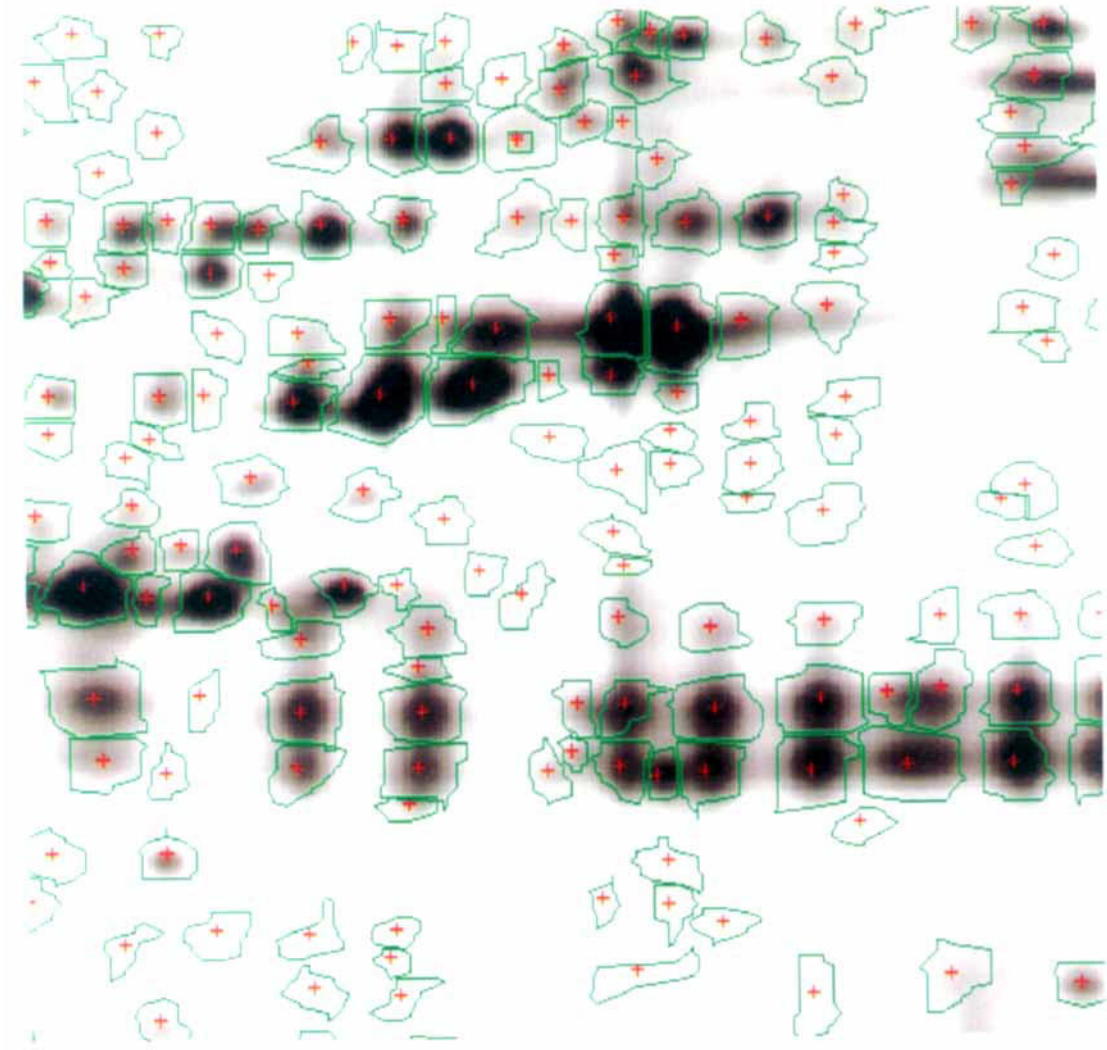

Figure 5. Zoomed subregion of Fig. 4. More details of detected spots and their boundaries are displayed. The segmentation has performed quite well on this gel as seen in the figure. Spots and their boundaries are detected accurately and the streaks in the gel did not affect the segmentation algorithm.

We would like to thank Dr. James Myrick of Centers for Disease Control, Atlanta, for providing some of the gels used in this paper and many useful comments about computer analysis of two-dimensional gel images.

Received June 2, 1993

5 References

[1] O'Farrell, P. H., *J. Biol. Chem.* 1975, *250*, 4007–4021.
[2] Anderson, N. G. and Anderson, N. L., *Clin. Chem.* 1982, *28*, 739–748.
[3] Lipkin, L. E. and Lemkin, P. F. *Clinical Chem.* 1980, *26*, 1403–1413.
[4] Lemkin, P. L. and Lipkin, L. E., in: Geisow, M. and Barrett, A. (Eds.), *Computing in Biological Science*, Elsevier/North Holland Amsterdam 1983, 181–226.
[5] Garrels, J. I., *J. Biol. Chem.* 1979, *254*, 7961–7977.
[6] Lemkin, P. L. and Lester, E. P., *Electrophoresis* 1989, *10*, 122–140.
[7] Vo, K.-P., Miller, M. J., Geiduschek, E. P., Nielsen, C. and Xuong, N. H., *Anal. Biochem.* 1981, *112*, 258–271.
[8] Appel, R. D., Hochstrasser, D. F., Funk, M., Pellegrini, C., Muller, F. A. and Scherrer, J.-R., *Electrophoresis* 1991, *12*, 722–735.
[9] Garrels, J. I., *J. Biol. Chem.* 1989, *264*, 5269–5282.
[10] Lemkin, P. L. and Lipkin, L. E., *Computers Biomed. Res.* 1981, *14*, 272–297.
[11] Skolnick, M. M., Sternberg, S. R. and Neel, J. V., *Clin. Chem.* 1982, *28*, 969–978.
[12] Miller, M. J., Olson, A. D. and Thorgeirsson, S. S., *Electrophoresis* 1984, *5*, 297–303.
[13] Soloman, J. E. and Harrington, M. G., *Computer Appl. Biosc.* 1993, *9*, 133–139.
[14] Pardowitz, I., Ehrhardt, W. and Neuhoff, V., *Electrophoresis* 1990, *11*, 400–406.
[15] Hader, D. P. and Kauer, G., *Electrophoresis* 1990, *11*, 407–415.
[16] Wu, Y. and Mislan, D., *Appl. Theor. Electrophoresis* 1993, *3*, 223–228.
[17] Lester, E. P., Lemkin, P. F., Lowery, J. F. and Lipkin, L. E., *Electrophoresis* 1982, *3*, 364–375.
[18] Myrick, J. E., Caudill, S. P., Robinson, M. K. and Hubert, I. L., *Appl. Theor. Electrophoresis* 1993, *3*, 137–146.
[19] Marr, D., *Vision – A Computational Investigation into the Human Representation and Processing of Visual Information*, W. H. Freeman, New York 1982.