Applied and Theoretical Electrophoresis (1993), **3**, 163–172

Splitting merged spots in two-dimensional polyacrylamide gel electrophoresis gel images*

Peter F. Lemkin[1], James E. Myrick[2] & Kyle M. Upton[3]

[1]*National Cancer Institute, FCRDC, Bld 469, Rm 150, Frederick, MD 21702,* [2]*Centers for Disease Control, 4770 Buford Highway NE (F19), Atlanta, GA 30341-3724 &* [3]*PRI/DynaCorp., FCRDC, Bld 469, Rm 150, Frederick, MD 21702, USA*

We describe a heuristic computer algorithm using boundary analysis for improving spot finding and spot quantitation of large saturated or near-saturated spots in two-dimensional polyacrylamide electrophoresis gels. This spot quantitation is done using spot segmentation, which consists of spot finding and subsequent quantification steps. Occasionally, clusters of large saturated spots may become merged during spot finding. To correct this, the merged spots must be cut apart before quantitation. It is generally obvious from viewing the merged spot's border where they should be cut — at opposing saddlepoints (concavities in the boundary). The algorithm uses an analysis of the missegmented spot's boundary when a saturated spot is detected. If a near-saturated spot is larger than a given size, the spot segmenter program attempts to merge saturated fragments. When merging occurs, the segmenter program analyses the boundary to see if the spot should be split. The new algorithm first finds all robust concavities and then tries to match complementary ones. These paired concavities are then used to guide cutting of the missegmented spot into two or more separate spot regions. Finally, control is returned to the segmenter program to reprocess the data as a set of smaller separated spots.

Keywords: two-dimensional gel electrophoresis; spot quantification; image processing; GELLAB; segmentation; boundary analysis; region splitting, spot detection.

Introduction

Two-dimensional polyacrylamide gel electrophoresis (2D-PAGE) developed by O'Farrell (1975) and others has been used to separate hundreds to several thousands of proteins. Good spot quantification is critical for later data analysis when comparing a set of different gel images from an experiment. This spot quantitation in a gel is done using spot segmentation, which consists first of spot finding and then of quantification steps. Saturation of a protein signal in the gel means that more protein is present than can be measured. This occurs when gels are heavily loaded, the autoradiograph film overexposed, the gel overstained, or the scanner operated in a nonlinear range. When the procedure for segmenting nonsaturated spots during spot finding is optimized, clusters of large saturated spots may occasionally merge. Saturated spots are those in which a spot's density values exceed the dynamic range of the gel preparation and digitization process. This situation occurs because the physical protein detection process, whether stain, film, or scanner digitization, is not stochiometric. Some scanners, such as charge coupled devices (CCDs) or vidicons, are more susceptible to saturation effects than others, such as laser scanners which have linear image-to-grayscale transfer functions and a larger dynamic range. Other solutions are to run gels with difficult sample loading in the case of silver-stained gels, or exposure times in the case of autoradiographs.

After the segmentation process, we compare spots from different gels. We have developed the GELLAB-II system for doing this type of exploratory data analysis (Lipkin & Lemkin, 1980; Lemkin *et al.*, 1982; Lemkin & Lipkin, 1983a, 1983b; Lemkin, 1989; Lemkin & Lester, 1989; Lemkin, 1992). After spot segmentation, we pair spots between experiment gels and a reference gel from the experiment. This process is followed by merging spots that correspond to the same spot in the reference gel into reference spots sets (Rspots) in a single composite gel database. Finally we do an *exploratory data analysis* on the composite gel database to find sets of Rspots that are related to changes in experimental conditions of gel subsets in the database. Proper spot segmentation is crucial for these later steps.

We present a heuristic enhancement to our spot segmentation algorithm which has already been reported (Lipkin & Lemkin, 1980; Lemkin & Lipkin, 1981; Lemkin *et al.*, 1982; Lemkin & Lipkin, 1983a; Lemkin & Rogan, 1991), and is implemented by the GELLAB-II sg2gii program. We split large near-saturated merged spots by using a robust boundary analysis algorithm after the initial Laplacian of Gaussian (LOG) spot detection phase described in these papers. The algorithm is similar to the boundary chain-code analysis algorithm of Solomon & Harrington (1991), but is based on our earlier work in separating touching cells in bone marrow smear optical microscope images described by Lemkin (1979). A similar algorithm was reported by Brenner *et al.* (1977). Harrington's algorithm is based on Freeman boundary chain-code analysis described by Freeman (1974). However, our new algorithm alternatively uses an analysis of run-projection maps (RPM) of the spot boundary (Merrill, 1973; Lemkin, 1978). We define the

Correspondence: P.F. Lemkin

*Portions of this paper were presented at the Annual Meeting of the Electrophoresis Society 21–24 June 1992, Research Triangle Park, North Carolina, USA

Use of tradenames is for identification only and does not constitute endorsement by the Public Health Service or the US Department of Health and Human Services

Received 12 August 1992; revised 22 September 1992

RPM transform below. The run projection map, which we describe in the methods, can be computationally more efficient than the chain-code in the context of our GELLAB-II segmentation algorithm. Thus we investigated this alternative instead of analysing the boundary in terms of the Freeman chain-code. Since the goal of the analysis is to find robust opposing saddlepoints on the boundary where the merged spots should be cut into smaller spots, finding optimum opposed saddlepoints corresponds to finding optimum opposed boundary concavities. Note that we only attempt the spot splitting analysis if a spot region was merged. This presents false splitting of nonmerged spots. Although our algorithm can also find the concavities by using the Freeman chain-code, we present the alternate method of analysis that uses RPMs. Robust pairs of concavities are found subject to constraints in matching opposing angles and positions.

Materials and methods

The new boundary analysis algorithm is illustrated with data from silver-stained, human urine 2D-PAGE gels from a study of cadmium toxicity described by Myrick *et al.* (1992).

Images were scanned with an 8-bit 1K × 1K CCD camera that is part of the Bio Image Visage 2000 system. We used a neutral density step wedge to calibrate each pixel value to optical density (OD). We used the GELLAB-II program ppxcvt to convert the gel image file from Bio Image format to the GELLAB-II format. The CCD camera has an 8-bit digitizer with maximum usable OD resolution of about 1.8–2.0 OD and becomes logarithmic at these high OD values. Other 8-bit CCD cameras and scanners also have these problems. Cooled CCD cameras with 12-bit or better A/D digitizers give a wider and more linear range, but are still logarithmic at the high OD end of their range. Laser scanners with photomultiplier detectors can be linear over their entire range.

Review of original GELLAB-II segmentation algorithm

In describing the original segmenter, we used the properties of the Laplacian function of a symmetric, monotonically increasing (one- or two-dimensional) function (i.e. has general shape similar to a Gaussian). In one dimension, the Laplacian is approximated by a second order difference function

$$\Delta^2 f(x) = f(x-1) - 2f(x) + f(x+1) \qquad (1)$$

These properties include: (i) the central peak of the Laplacian of this function has a negative value; (ii) there are two positive side peaks on either side of the main peak of this function, and most of the area of the function is between these side peaks; (iii) the magnitude of the Laplacian goes toward zero outside of these side peaks.

Nonsaturating spots have a region in the center of the spot where the direction of the Laplacian of a Gaussian smoothed (LOG) filtered gel image is negative in both the X and Y directions. In saturated spots, where $f(x,y)$ is approximately constant, the Laplacian disappears (i.e. becomes zero). Normally, one traverses the gel in a top to bottom, left to right scan searching for such negative LOG regions.

Once a pixel in this region is detected for a new spot which is being segmented, we find all 4-neighbor connected (i.e. north, south, east, west) pixels with this property and put them into a *central-core* region pixel list called the *blob list* (BL). The segmentation process continues by propagating labeled pixels from the central-core region (i.e. adding them to the BL) until we reach the side magnitude peaks of the Laplacian, we run into another spot, or we reach a 'noisy' region. Any one of these criteria stops propagation.

Finally, after some minor edge smoothing and hole and concavity filling, we use this final *propagated central-core* (PCC) of a spot as a mask to define a region whose pixels are integrated in OD space to compute the integrated density, D. This PCC is defined as a function of the Laplacian of the spot — not by any density threshold. This allows us to easily handle non-Gaussian shaped spots and touching spots. Note that D is not corrected for background density at this stage of the segmentation; that process comes next. After all spots are found, we subtract them from the original image to generate an effective background image. In turn this image is smoothed using the zonal notch-filter described by Lemkin *et al.* (1982) to compute a two-dimensional lookup table, $B_{x,y}$, of the gel background image optical density for *all* pixels in the image. For the urine gel database described in this paper, the zonal notch filter used an averaging window of 64 × 64 pixels — considerably larger than the largest spot in the gel (except for albumin). For all spots in the gel, we then estimate the corrected integrated density, D'_j, for a spot, j, with centroid position (x,y) in equation (2) by

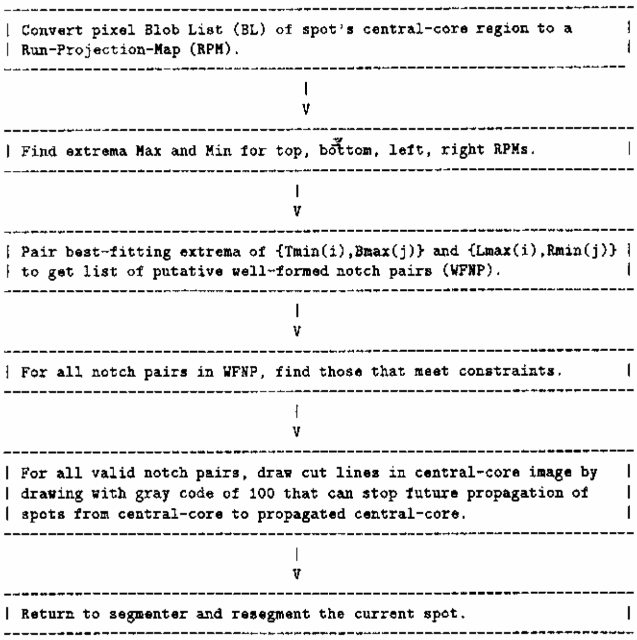
$$D'_j = \mathrm{D}_j - A_j * B_{x_j y_j} \qquad (2)$$

where (x_j,y_j) is the centroid of spot j. Area A of a spot is the number of pixels in the propagated central-core region. At this point, spots are accepted or rejected by testing their features against limits of area, integrated density D', and the OD range of a spot. This original segmenter algorithm is explained in detail by Lemkin & Lipkin (1981, 1983a).

Saturated spot merging algorithm

As mentioned earlier, the Laplacian of a large near-saturated or saturated spot poorly represents the actual shape of its central core. This poor representation results in the fragmentation of these spots. The segmentation may be improved by first merging the fragments together and then splitting out separate spots based on the shape of the initial merged spot.

The older GELLAB-II segmenter has a saturation-merging algorithm similar to that described by Olson & Miller (1988) for merging spots fragmented by saturation. To decrease computation, we apply this algorithm only to candidate spots whose initial central-core size is greater than some reasonably large minimum size. A spot must have an area greater than the lower spot area threshold for it to be an initial candidate for further merge analysis. The second part of the merge candidate test analyses the spots



```
| Convert pixel Blob List (BL) of spot's central-core region to a       |
| Run-Projection-Map (RPM).                                              |

| Find extrema Max and Min for top, bottom, left, right RPMs.           |

| Pair best-fitting extrema of {Tmin(i),Bmax(j)} and {Lmax(i),Rmin(j)} |
| to get list of putative well-formed notch pairs (WFNP).                |

| For all notch pairs in WFNP, find those that meet constraints.        |

| For all valid notch pairs, draw cut lines in central-core image by    |
| drawing with gray code of 100 that can stop future propagation of     |
| spots from central-core to propagated central-core.                    |

| Return to segmenter and resegment the current spot.                    |
```

Figure 1 Flow chart of the spot-splitting boundary analysis algorithm. This algorithm is invoked after applying the spot-merging algorithm (described in the text) if the area of the merged central-core region is > T_{minCC}

densities. The test first computes the maximum and minimum gray value, which is in the initial central-core pixel list. The threshold $T_{\%}$ is a percentage of the darkest gray value in the gel image, and not the darkest possible gray value that is determined by the number of bits/pixel. If the maximum is greater than $T_{\%}$,* the algorithm attempts to merge all adjacent pixels that have a gray value greater than the minimum gray value minus 1 of any 8-neighbor connected pixel to the central-core. It then iteratively expands central-core pixels (adding them to the central-core list) until no more pixels meet the expansion criteria. This spot merging algorithm then yields a new expanded list of central-core pixels with gaps filled between spot fragments.

Although this algorithm tends to merge fragments of saturated spots quite well, it may also occasionally merge adjacent saturated spots. This problem is addressed by the new algorithm presented here.

Splitting algorithm

The basic algorithm flow is given in Figure 1. As input, it uses the blob list (BL) of pixels for the central-core after the saturated spot merging algorithm has been applied. At this point, we have a list of all pixels in what appears to be a reasonable central-core of the spots in question; however, the merged spots are still connected.

Run Projection Map — definition

The first step is to convert the BL pixel list of central-core pixels to RPMs — one for each of the four edges:

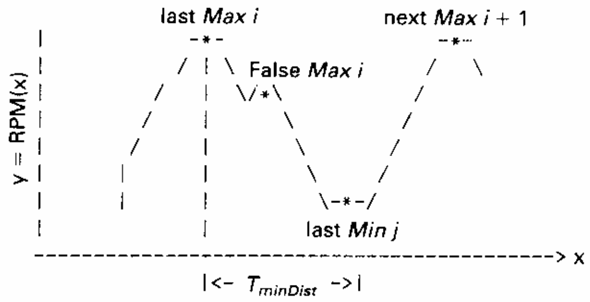


Figure 2 Illustration of minimum-distance peak-finding algorithm used in finding extrema. Extrema must be > $T_{minDist}$ apart (determined from the resolution of spots in the gel). This restriction eliminates finding false peaks as illustrated below when searching for extrema from left to right. After finding the first maximum (or minimum) peak, the algorithm does not look for another of the same type until it moves at least that distance to the right

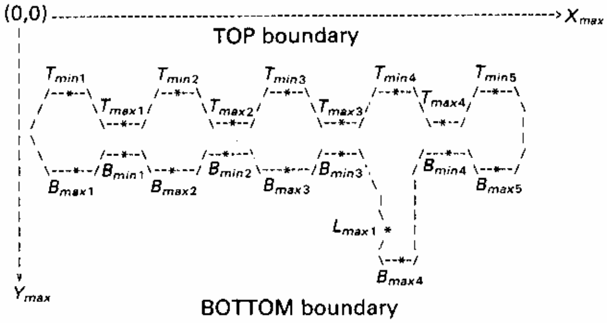


Figure 3 Schematic of the top and bottom, left and right RPM boundaries showing maxima and minima extrema and using a raster, rather than Cartesian, coordinate system. The extrema were found using the minimum-distance peak-finding algorithm that prevents noise on the boundaries from being counted as additional peaks. Note that because the right concavity was weak, there is no right minima corresponding to L_{max1}

top, bottom, right, left. *Each RPM function is a side projection of a spot.* For example, by viewing a spot from the top, we see that the top RPM function $y = f_{top}(x)$ specifies the y as a function of x. Figure 2 illustrates $y = RPM(x)$ for the RPM f_{top}. Similarly, the other RPM functions are: $y = f_{bot}(x)$, $x = f_{left}(y)$ and $x = f_{right}(y)$. We illustrate the computation for the RPM f_{top} as follows — the other three functions are computed similarly. Given the pixel blob list, for each pixel $(x,y) \epsilon BL$, compute $f_{top} = max(f_{top},y)$. Although the RPM is an efficient way to encode the simple objects that are most often seen in two-dimensional gels, it does not work for complex objects like spirals or those which have other types of hidden concavities (Merrill, 1973; Lemkin, 1978).

Next, we find the *maxima* and *minima* extrema for these four RPM functions. To build noise immunity into the extrema finding, we use a minimum-distance peak-finder, illustrated in Figure 2, that tracks the last extrema found but that will not start a new one if the new putative extrema is less than $T_{minDist}$ from the previous extrema of the same type.

The next step is to pair notches. Since the image coordinate system is raster-scan, with (x,y) being (0,0)

*For the urine gel in the example, we used a value for $T_{\%}$ of 99.7%

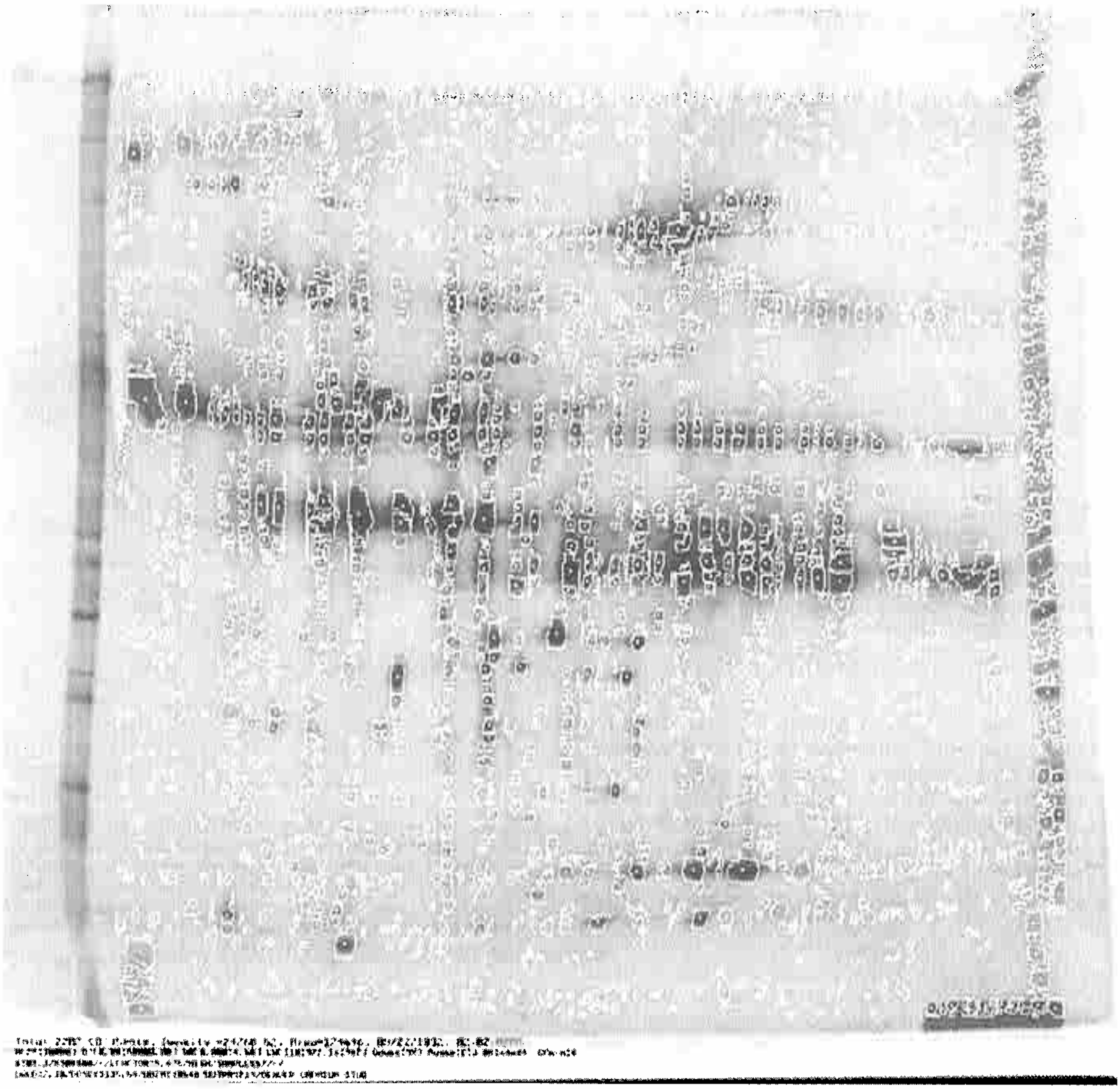


Figure 4 A segmented silver-stained gel of human urinary proteins. No saturated spot merging nor spot splitting was used. The original gel image is smoothed using a 3 × 3 Gaussian filter followed by a 'Busse' super-pixel Laplacian of size 3 × 3 (Lemkin & Rogan, 1991). Noise spots with an initial central-core area <4 were rejected. The zonal notch filter averaging window size is 64 × 64. The five large spots (cf. Figure 5) were incorrectly fragmented into many small spots. The segmenter parameters are tuned for the scanner resolution and characteristics of the set of gels and are robust over the entire set of gels. The sg2gii program parameters were:

BackgroundODfilterSize:	64 × 64 pixels sq.
AreaRangeOfSpot:	25, 10000 pixels sq. of acceptable spots
DensityRange:	0.0005, 50000 OD of acceptable spots
OdRangeInSpot:	0.0001, 4.5 OD of acceptable spots
GaussianSmoothing:	3 × 3 pixels sq.
CCminThreshold:	4 pixels sq. area for considering central core spot
BusseLaplacian:	3 × 3 pixel super-pixel smoothing

at the upper left hand corner and (no. of cols, no. of rows) at the lower right hand corner, we try to find the *best pairs* of top maxima with bottom minima extrema and, correspondingly, the left maxima with right minima extrema. For each (*i,j*) pair (i.e. *i*th maxima and *j*th minima), we compute some features used in finding the best notch pair. We illustrate this for the top/bottom pairing. Left/right pairing is similar. Figure 3 is a schematic of the boundary of the merged spots with the top and bottom maxima and minima indicated.

We now develop a metric in the following equations for evaluating how well the algorithm is able to pair opposed notches. *X* and *Y* refer to the corresponding

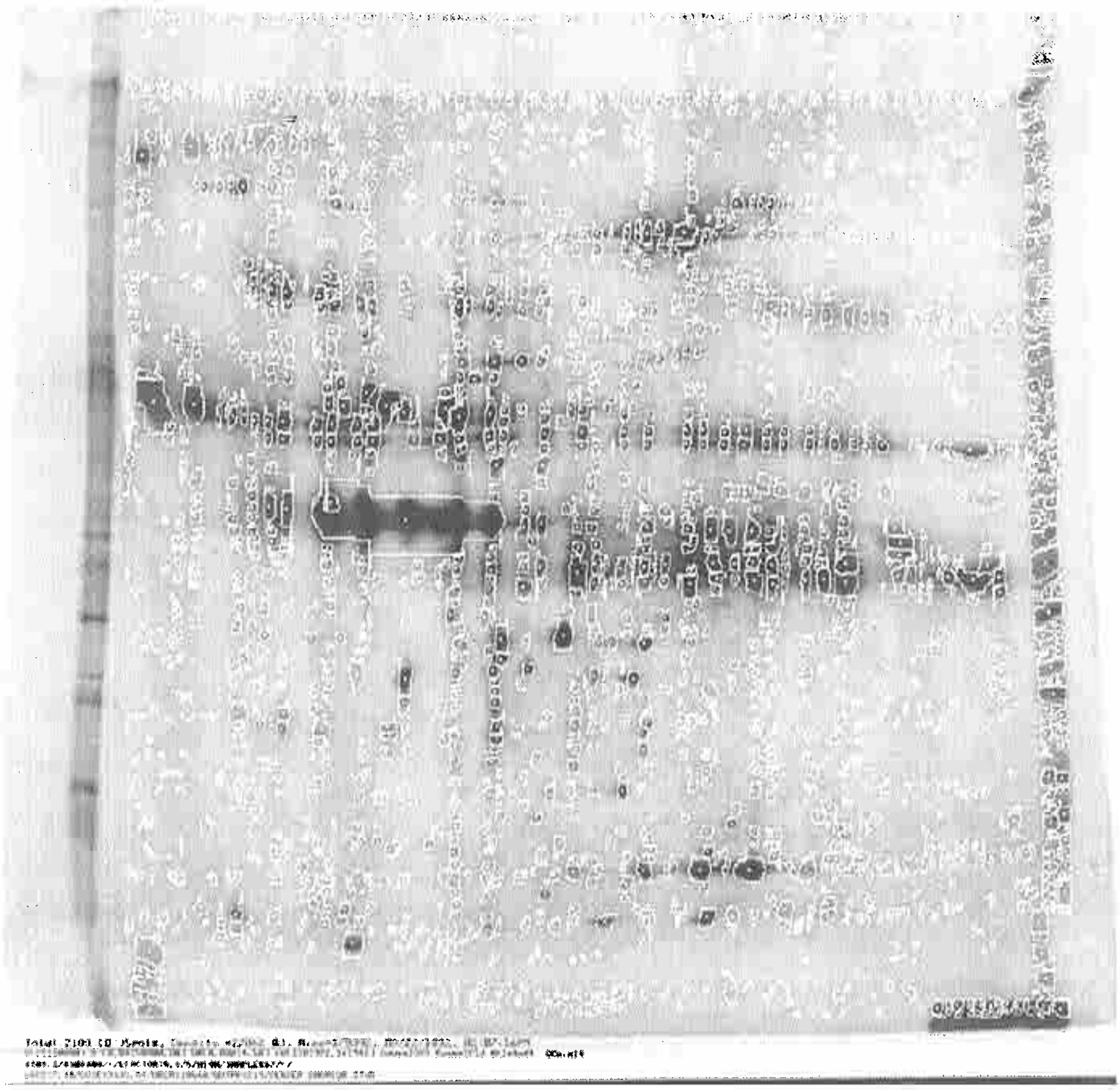


Figure 5 The segmented gel using the saturated spot merging algorithm. Notice that the five large spots were merged into a clearly incorrect 'super spot'. This segmentation was done *without* the extended spot-splitting boundary analysis algorithm described in this paper. The segmenter parameters were the same as for Figure 4 except for: SaturatedSpotMerge:99.7. That is, in the image, pixels >99.7% of other gray values trigger spot merging

coordinates of the extrema instance; e.g. $X(T_{min4})$ and $Y(T_{min4})$. The horizontal *slew*, represented by $dX_{i,j}$ (equation (3)), shows the alignment of the concavities and should be small in matched notches. The vertical distance between the top min_i and bottom max_j is defined by $dNotch_{i,j}$ in equation (4) — again, smaller is better in matched notches. The maximum vertical distance across any spot in the merged spots is defined as *dPerp* in equation (7) using the Y_{max} and Y_{min} that were defined in equations (5) and (6). Alternatively, *dPerp* could be defined in terms of the difference between adjacent maxima $\max_{p,q}(|Tmin_p - Bmax_q|)$ for extrema (p,q) adjacent to extrema (i,j).

Finally, we derive the aspect ratio of the notch and its adjacent spots as $aspectRatio_{i,j}$ in equation (8). For a spot that should be split, this value should be high, while for a noisy spot that should not be split, the value should be around 1.0 — approximately round.

$$dX_{i,j} = |Xmin_i - Xmax_j| \quad (3)$$

$$dNotch_{i,j} = |Ymin_i - Ymax_j| \quad (4)$$

$$Y_{max} = \max_{x \in object} (f_{top}(x), f_{bot}(x)) \quad (5)$$

$$Y_{min} = \min_{x \in object} (f_{top}(x), f_{bot}(x)) \quad (6)$$

$$dPerp = Y_{max} - Y_{min} + 1 \quad (7)$$

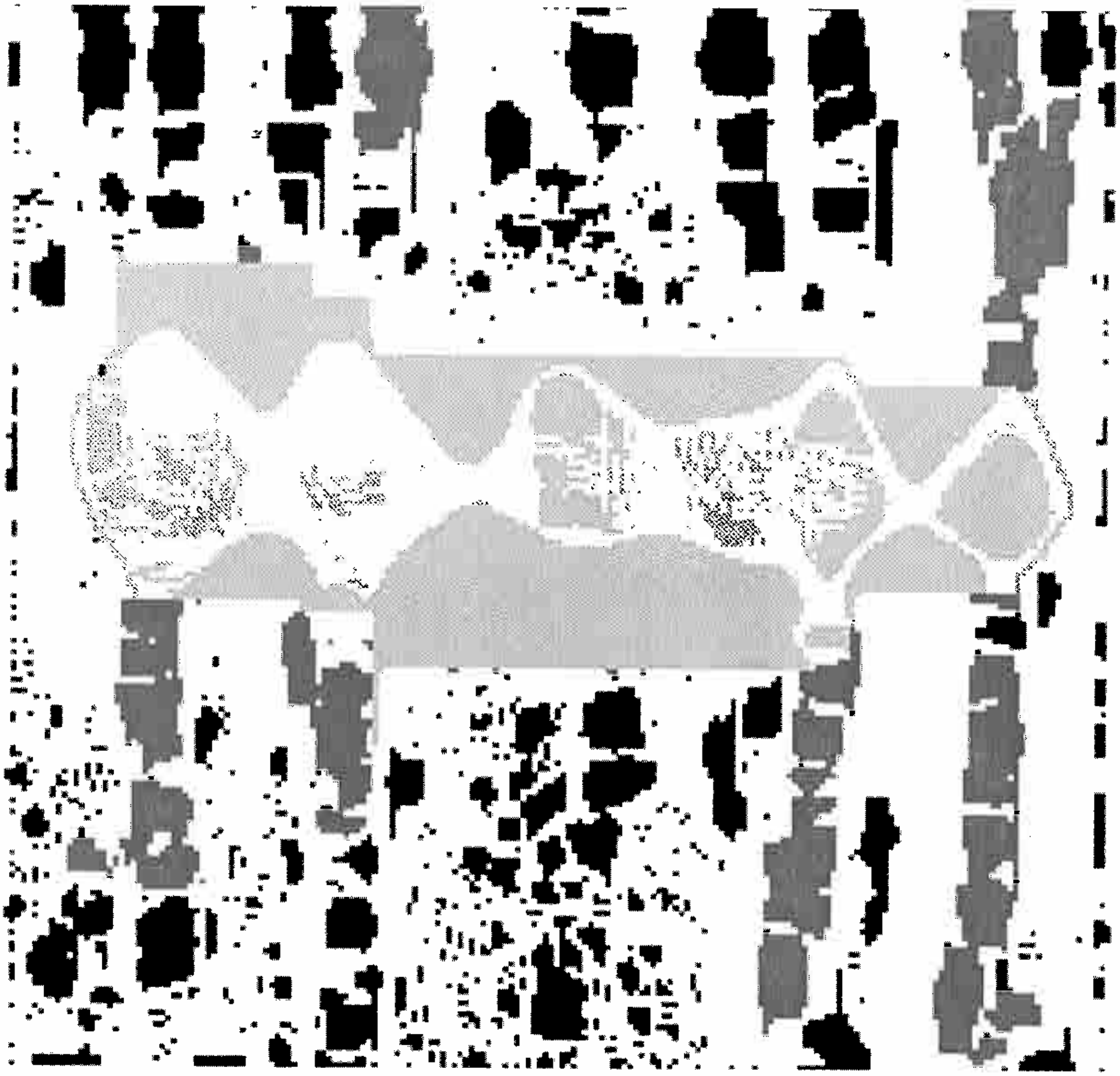


Figure 6 A 4× magnification of the central-core region of the five spots, shown in Figure 5, after applying the saturated spot merging algorithm. We invoked saturation-merging operation because there was a pixel in the central-core that was greater than the saturation threshold (here set to 99.7% of darkest gray value in the image). Notice the potential cut-points between spots at the matching concavities. To make it easier to see the spots of interest, the segmenter's parameters were the same as for Figure 5, except for: AreaRangeOfSpot:[280,10000]

$$aspectRatio_{i,j} = dPerp/dNotch_{i,j} \qquad (8)$$

For parallel RPM extrema (i.e. the *i*th top maxima and the *j*th bottom minima, or the *i*th left maxima and the *j*th right minima), we find the best (*i,j*) pairs such that $dX_{i,j}$ is a minimum and meets the constraints in equation (9). This is called the Well-Formed Notch-Pairs (WFNP). For gels scanned with the Bio Image system at 169 microns pixel^{-1}, threshold $T_{maxPerpDeviation}$ was set to 6 and threshold $T_{minNotchAlignRatio}$ was set to 2.

$$WFNP_{i,j} = (dX_{i,j} \leq T_{maxPerpDeviation}) \wedge (aspectRatio_{i,j} > T_{minNotchAlignRatio}) \qquad (9)$$

Then, for all $WFPN_{i,j}$ notch pairs, we draw cut lines in the central-core image by drawing with a code of 100 in the 8-bit central-core image. Pixels with code 100 can stop propagation of spots from central-core to propagated central-core when this region is resegmented. As described by Lipkin & Lemkin (1980), the spot's central-core pixels are labeled with a sequentially assigned value *n* in the range of [2 : 99] mod 100 while propagated central-core pixels for the same spot are labeled with 100 + *n*. Isolated pixels are labeled with 255 and deleted spots with 254. Any of these nonzero codes can shut off central-core propagation of another spot that is being segmented.

Finally, we return to the segmenter and resegment the current spot and later adjacent spots that we just split off from the current spot.

Results

Figure 4 shows a segmented silver-stained gel of human urinary proteins from the cadmium toxicity study using neither the near-saturated spot merging algorithm nor the spot splitting algorithm. Notice that

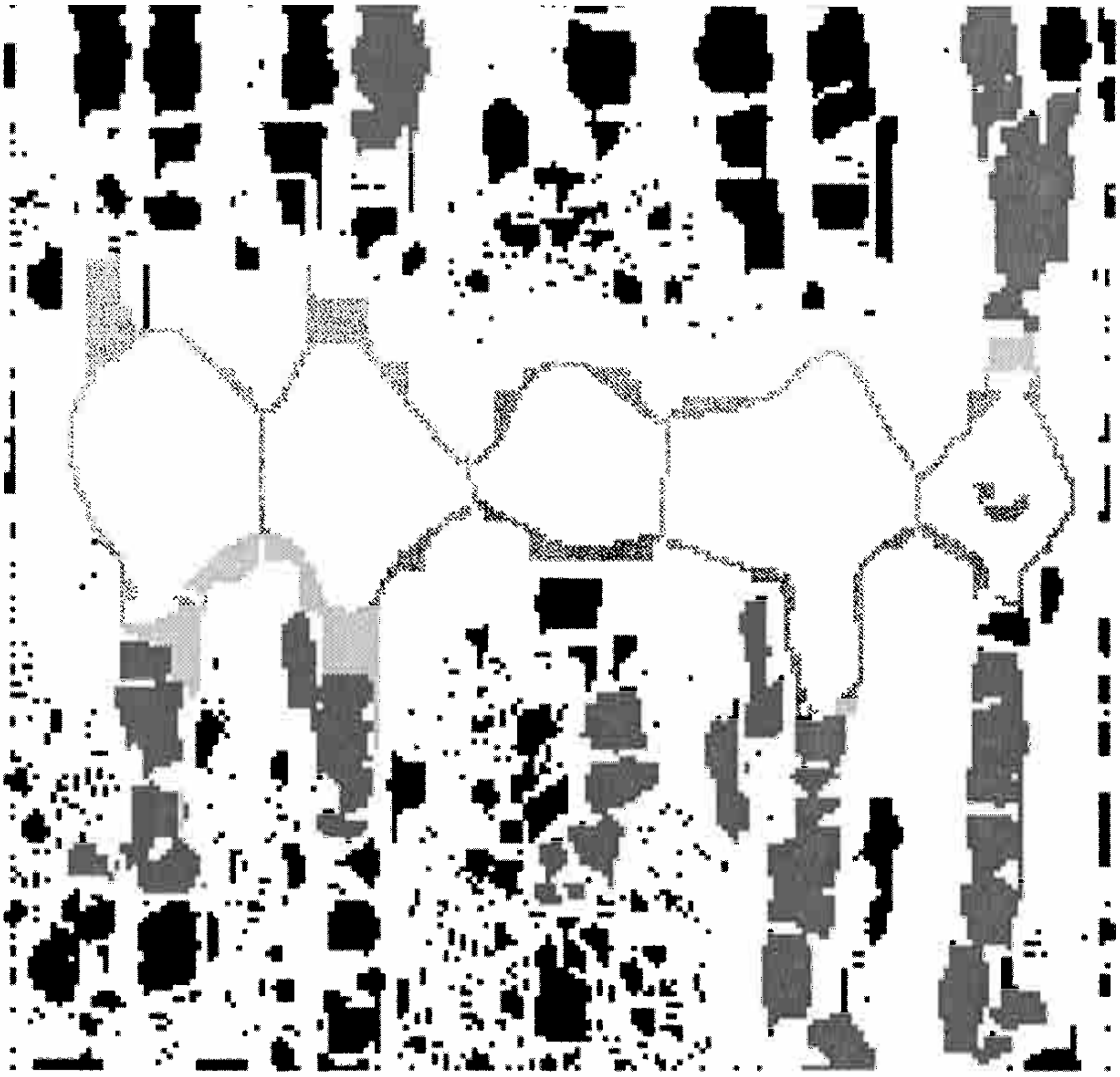


Figure 7 The 4× magnification of the central-core image of the test region with cut-points after merging the saturated spot and performing the spot splitting boundary analysis. The segmenter's parameters were the same as for Figure 6, except for: SplitSpots:RPM,50. The RPM indicates that Run-Projection-Map analysis (rather than chain-code analysis) is to be used. The value of $>T_{minCC}$ is 50

some of the large saturated spots were fragmented (middle left).

Figure 5 shows the segmented silver-stained gel using the saturated spot merging algorithm. Here, the five large spots were merged into a clearly incorrect 'super spot'. Figure 6 shows the central-core region of these spots after applying the saturated spot merging algorithm. This image suggests that opposed concavities are probably a good estimate of where one might cut the spots apart. Finding these cut points is the key idea presented in this paper. After cutting those spots in the central-core image, we resegment then to detect and quantitate them as separate spots.

Figure 3 illustrates the boundary schematic of the merged spots with the top and bottom maxima and minima shown. This schematic may be used as a guide to help read the tables showing numerical results for values in equations (3)–(8). Table 1 lists the number of all extrema found in the RPM analysis for the four sides. The left notch of a weak concavity on the lower part of the third spot was found, but its corresponding right concavity was not located. The latter was too shallow for the current peak-finder settings and was not split correctly. Table 2 lists all of the extrema for the top/bottom RPMs. Table 3 lists the features for all notch pairs derived from the Table 2 data. The WFNPs are shown with a '*' and specify where the central-core image will be cut.

Figure 7 shows the central-core image after applying the spot splitting boundary analysis after the spot cutting. Notice the cut-points that were drawn to mark boundary regions. Figure 8 shows the final, successfully segmented gel image where we had first merged saturated spots and then split the spots using the boundary analysis. Here the spots that were previously missegmented are now correct.

Without the use of the new spot-splitting algorithm, four gels of the 29 gels in the cadmium study were

Table 1 Number of notches found in RPM notch analysis

Side	*Minima*	*Maxima*
Top	5	4 V
Bottom	4 V	5
Left	1	1 H
Right	0 H	1

The number of all extrema found in the RPM analysis for the four sides of the boundary shown in Figure 6. The vertical (horizontal) extrema of the sides that should be compared are marked with a 'V(H)'. Since there were no right minima found, we do not search for horizontal paired notches in this data — only for vertical paired notches

Table 2 List of all the extrema in the RPMs

Peak no. i	T_{min_i}	T_{max_i}	B_{min_i}	B_{max_i}
1	294,465	311,479	312,498	292,511
2	324,467	347,488	349,493	329,511
3	364,471	383,481	381,499	368,501
4	411,470	426,491	426,497	408,523
5	443,474	NA	NA	441,509

The maxima and minima for the top and bottom RPMs are listed. Note that since (x,y) is in a raster coordinate system with (0,0) the upper left hand corner, a maxima is downward and a minima is upward. Since the number of right minima is zero, we do not present it. The *dPerp* for the top/bottom comparison has a value of 174.0

Table 3 Notch pairs derived features

(j,i)	T_{max1}	T_{max2}	T_{max3}	T_{max4}

B_{min1}	dX = 1*
	dN = 19
	ar = 9.1
		dX = 35
		dN = 10
		ar = 17.4
			dX = 71
			dN = 17
			ar = 10.2
				dX = 114
				dN = 19
				ar = 24.6

B_{min2}	dX = 38
	dN = 14
	ar = 12.4
		dX = 2*
		dN = 5
		ar = 34.8
			dX = 34
			dN = 12
			ar = 14.5
				dX = 77
				dN = 19
				ar = 87.0

B_{min3}	dX = 70
	dN = 20
	ar = 8.7
		dX = 34
		dN = 11
		ar = 15.8
			dX = 2*
			dN = 18
			ar = 9.7
				dX = 45
				dN = 19
				ar = 21.8

B_{min4}	dX = 115
	dN = 18
	ar = 9.7
		dX = 79
		dN = 9
		ar = 19.3
			dX = 43
			dN = 16
			ar = 10.9
				dX = 0*
				dN = 19
				ar = 29.0

The derived features used in the constraint tests are listed in this table. Entries marked with * suggest WFNPs ($Tmax_i$, $Bmin_j$) that were the *best* fits. Note for each entry *i,j*, dX is $dX_{i,j}$, dN is $dNotch_{i,j}$, ar is $aspectRatio_{i,j}$

found to have incorrectly merged saturated spots by reviewing all of the segmented gel images. The new segmenter correctly split two of these spots. Merged spots from two gels were still not correctly split. On inspection, it appeared that the merged spots presented a small aspect ratio that tended to mask the cut-points. Since no segmenter will work in all situations, we recommend using manual editing for the few cases that remain.

Discussion

Most of the time, the algorithm is able to split occasionally occurring merged, large, near-saturated or saturated spots.

The algorithm is robust within the current parameters. However, the algorithm might be tuned further by more closely defining the approximate size and steepness of the concavities resulting in even better false-negative false-positive cutting rates. For small, merged, saturated spots with shallow notches, the algorithm may not do as well. Fortunately, most saturation occurs with large spots, so small spot size does not seem to be a problem. Since $aspectRatio_{i,j}$ is independent of spot size, the same set of parameters should be usable for a wide range of spot sizes if the pair slew (equation (3)) is small.

By adjusting the max/min peak finder (i.e. by decreasing *TminDist* from the default by 5 pixels), the algorithm could split a wider range of spot sizes, but possibly at the cost of a higher false-positive rate for notches. We currently have *TminDist* set to reject small spots since most of the problems we have seen occur with large saturated spots. We feel it is better to have a lower false-positive cut rate than a higher false-negative one, and therefore set the threshold conservatively.

Because the RPM is designed around horizontal and vertical runs, it may be possible that spots merged that are not aligned along these axes will merge, using a chain-code based notch finder that eliminates the horizontal/vertical biases of the RPM notch finder. However, in the situations that we have seen, all cases of large saturated spots seem to be well aligned either horizontally or vertically. Such alignment seems to be a property of proteins separated in PAGE gels which could be due to the orthogonal method of how gels are made (i.e. first run pIe gel then orthogonal SDS gel).

A related issue is determining whether a spot should be split. Appel & Hochstrasser (personal communication, June 23 1992 PFL) suggest two methods for handling spots that sometimes appear to be split. First the corresponding spots in *N* gels are found. The spot may be segmented into two spots, or may be hourglass shaped so that it could be split with an algorithm like that described in this paper, or may appear to be one spot. Appel & Hochstrasser suggest that one either treat the separate spots as one if they appear as one in several replicate gels, or alternatively, treat them as separate spots and do one's best to split all that are splittable.

Additional work might be done using synthetic spots to further evaluate the sensitivity and robustness of the splitting algorithm. In particular, these should show cases where cutting is easy, where difficult but where the algorithm still works, where it fails because the cut is difficult and it does not cut, and finally where it cuts where it should not. Knowing this might enable better tuning of the algorithm in the difficult cases.

In the GELLAB composite gel database, these spot fragments or split spots can show up as *ambiguous spots* (AP) if two spots in one gel are matched to one spot in another gel. The cgelp2 composite gel database program (Lipkin & Lemkin, 1980; Lemkin *et al.*, 1982; Lemkin & Lipkin, 1983a; Lemkin, 1989; Lemkin & Lester, 1989; Lemkin, 1992) has a 'MERGE AP SPOTS' operation that merges AP spots into a single corresponding spot to yield the integrated density of merged spots.

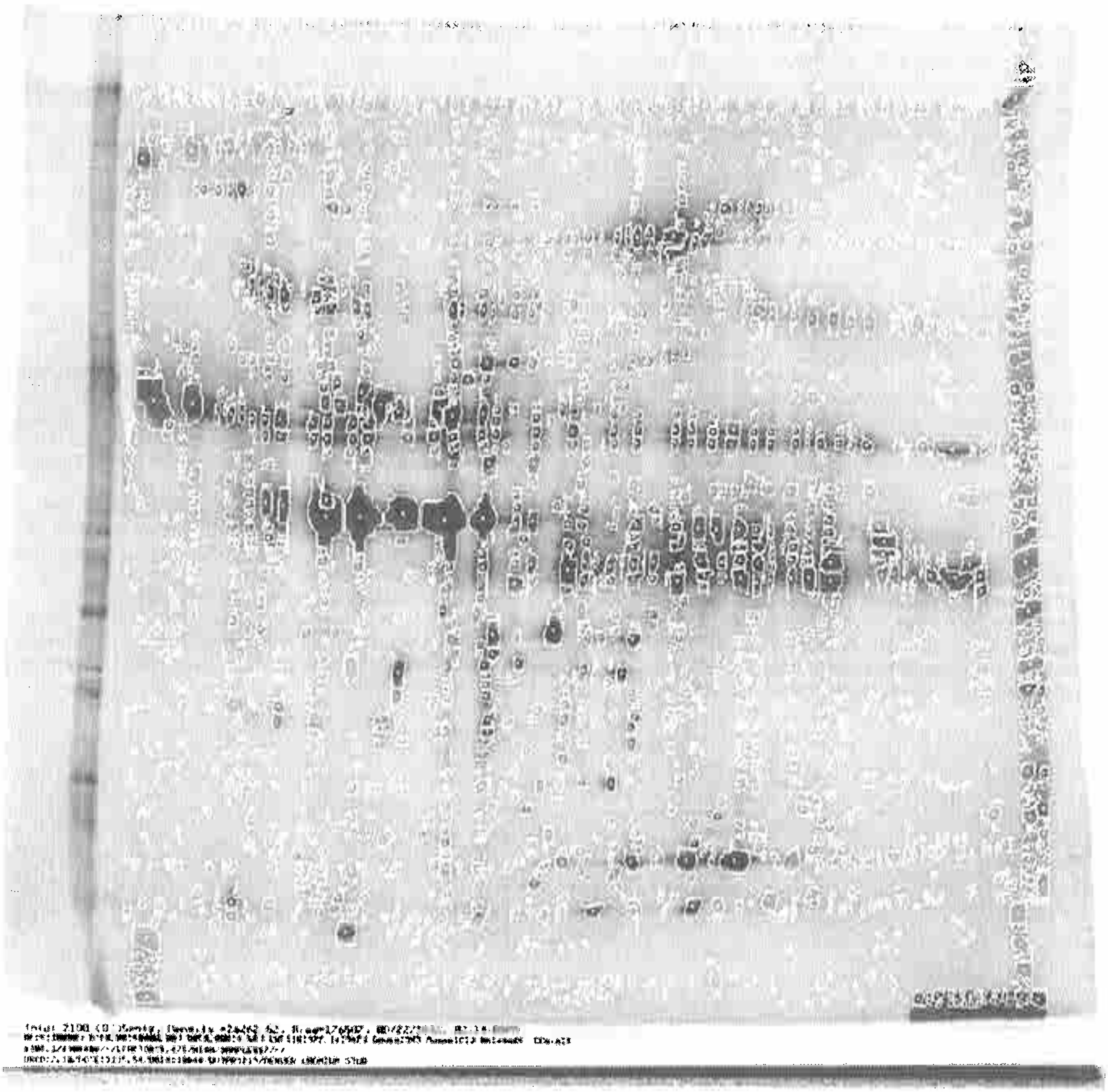


Figure 8 The final segmented gel after boundary analysis using both saturated spot merging and the new spot-splitting algorithm. Notice that the spots were correctly segmented. The segmenter's parameters were the same as for Figure 5, except for: SplitSpots:RPM,50. The value of T_{minCC} is 50

However, this procedure is not a complete solution because not all the fragments should be merged. The AP spots that are candidates for merging include spot fragments, noise spots, and actual new spots. Spot features, such as integrated density, area, distance from other spots, and OD range, have distributions that overlap to a large degree among the different candidate spot types. Therefore, because it is difficult to separate the different spot types, we do not merge AP spots automatically. The merging probably should be done interactively using a spot editor.

This enhancement to the GELLAB-II segmenter helps to resolve difficult spots sometimes found in urine, plasma, and other types of gels because of the occurrence of large saturated spots. Although merging and splitting spots is computationally expensive, the process is invoked only when near-saturation conditions occur in the context of a large spot. Therefore, the additional computation is low and does not significantly increase the duration of the quantification process. Good quantification helps to reduce the generation of false spots from fragmentation; such reduction improves the reliability of database statistical searches and later exploratory analysis.

Acknowledgments

We wish to thank the reviewers and editors for helpful comments which improved both the content and presentation of this paper.

References

Brenner, J., Necheles, T.F., Bonacossa, I.A., Fristensky, R., Weintraub, B.A., Neurath, P.W. (1977). Scene segmentation techniques for the analysis of routine bone marrow smears from acute lymphoblastic leukemia patients. *J. Histochem. Cytochem.*, **25**, 601–613.

Freeman, H. (1974). Computer processing of line drawing images. *Comp. Surv.*, **6**, (1), 57–98.

Lemkin, P.F. (1978). *The Run Length Map: A Representation of Contours and Regions for Efficient Search and Low Level Semantic Encoding*. College Park, MD, U MD Computer Science Center TR-655, 60 pp.

Lemkin, P.F. (1979). An approach to region splitting. *Comp. Graphics Image Proc.*, **10**, 281–288.

Lemkin, P.F. (1989). GELLAB-II, a workstation based 2D electrophoresis gel analysis system. In: Endler, T. & Hanash, S. (eds), *Proceedings of Two-Dimensional Electrophoresis*, Vienna, Austria, Nov 8–11, 1988. VCH Press, Weinheim, Germany, pp. 53–57.

Lemkin, P. F. (1992). Representations of protein patterns from 2D gel electrophoresis databases. In: Pickover, C. (ed.), *The Visual Display Of Biological Information*. World Publishers, Teaneck, NJ (in press).

Lemkin, P.F. & Lipkin, L.E. (1981). GELLAB: a computer system for 2D gel electrophoresis analysis. I. Segmentation and preliminaries. *Comp. Biomed. Res.*, **14**, 272–297.

Lemkin, P.F. & Lester, E.P. (1989). Database and search techniques for 2D gel protein data: a comparison of paradigms for exploratory data analysis and prospects for biological modeling. *Electrophoresis*, **10**, 122–140.

Lemkin, P.F. & Lipkin, L.E. (1983a). Database techniques for 2D electrophoretic gel analysis. In: Geisow, M. & Barrett, A. (eds), *Computing in Biological Science*. Elsevier/North Holland, pp. 181–226.

Lemkin, P.F. & Lipkin, L.E. (1983b). 2D Electrophoresis gel database analysis: aspects of data structures and search strategies in GELLAB. *Electrophoresis*, **4**, 71–81.

Lemkin, P.F. & Rogan, P. (1991). Automatic detection of noisy spots in two-dimensional southern blots. *Appl. Theor. Electrophoresis*, **2**, 141–149.

Lemkin, P.F., Lipkin, L.E. & Lester, E.P. (1982). Some extensions to the GELLAB 2D electrophoresis gel analysis system. *Clin. Chem.*, **28**, 840–849.

Lipkin, L.E. & Lemkin, P.F. (1980). Database techniques for multiple PAGE (2D gel) analysis. *Clin. Chem.*, **26**, 1403–1413.

Merrill, R.D. (1973). Representation of continuous regions for efficient computer search. *CACM*, **16**, 69–82.

Myrick, J.E., Caudill, S.P., Robinson, M.K. & Hubert, I.L. (1993). Two-dimensional electrophoretic detection of possible urinary protein biomarkers of occupational exposure to cadmium. *Appl. Theor. Electrophoresis*, **3**, 137–146.

O'Farrell, P.H. (1975). High resolution two-dimensional electrophoresis of proteins. *J. Biol. Chem.*, **250**, 4007–4021.

Olson, A.D. & Miller, M.J. (1988). Elsie 4: quantitative computer analysis of sets of two-dimensional gel electrophoretograms. *Anal. Biochem.*, **169**, 49–70.

Solomon, J.E. & Harrington, M.G. (1991). A robust high-sensitivity algorithm for automated detection of proteins in two-dimensional electrophoresis gels. *Comp. Applications Biosci.* (submitted).