

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/12923249>

# Identification of relative protein bands in Polyacrylamide Gel Electrophoresis (PAGE) using multiresolution...

Article in *BioTechniques* · July 1999

DOI: 10.1109/TFSA.1998.721415 · Source: PubMed

CITATIONS

4

READS

77

5 authors, including:



**Mehmet Koyutürk**

Case Western Reserve University

78 PUBLICATIONS 1,652 CITATIONS

SEE PROFILE



**Rengul Cetin-Atalay**

Middle East Technical University

101 PUBLICATIONS 773 CITATIONS

SEE PROFILE



**A. Enis Cetin**

Bilkent University

280 PUBLICATIONS 3,930 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Theoretical Network Analysis [View project](#)



Drug Response Prediction as a Link Prediction Problem [View project](#)

# Identification of Relative Protein Bands in Polyacrylamide Gel Electrophoresis (PAGE) Using a Multi-Resolution Snake Algorithm

*BioTechniques* 26:1162-1169 (June 1999)

**Metin Nafi Gürcan, Mehmet Koyutürk, H. Serkan Yıldız, Rengül Çetin-Atalay and A. Enis Çetin**

Bilkent University, Bilkent, Ankara, Turkey

## ABSTRACT

*In polyacrylamide gel electrophoresis (PAGE) image analysis, it is important to determine the percentage of the protein of interest of a protein mixture. This study presents reliable computer software to determine this percentage. The region of interest containing the protein band is detected using the snake algorithm. The iterative snake algorithm is implemented in a multi-resolutional framework. The snake is initialized on a low-resolution image. Then, the final position of the snake at the low resolution is used as the initial position in the higher-resolution image. Finally, the area of the protein is estimated as the area enclosed by the final position of the snake.*

## INTRODUCTION

Polyacrylamide gel electrophoresis (PAGE) is one of the most widely used techniques in molecular biology (1). Denaturing sodium dodecyl sulfate PAGE (SDS-PAGE) is an important method used for the separation of proteins based on the migration of negatively charged proteins, depending on their molecular weight in an electrical field. Its advantage is that proteins can be separated and visualized. The SDS-PAGE technique provides information about the degree of purity of a particular protein in a protein mixture applied on a certain lane of the PAGE during the protein purification process. Each band in the SDS-PAGE image represents a different protein.

In the protein purification process, it is important to determine the efficiency of each purification step in terms of percentage of protein of interest (POI) found in the protein mix-

ture. This paper provides a rapid and reliable way to determine the percentage. Generally, the percentage is determined by eye, and the results are quite subjective. In the currently available computer-assisted gel analysis systems, the analysis is usually based on the one-dimensional (1-D) profile of manually (by computer mouse) extracted lanes. The band of the POI is determined from the peak points in the horizontal profile, and the areas of the rectangular regions are measured. Finally, these measurements are used for purification efficiency calculation. However, rectangular approximations are not accurate representations of the ellipse-like protein regions.

In our scheme (see Availability), the region of interest (ROI) containing the significant protein is detected using the snake algorithm (4). A “snake” is loosely placed around the protein band as an ellipse. Then, the snake shrinks itself until it closely encircles and fits around the protein band. The final position of the snake determines the extent of the protein. The iterative snake algorithm is implemented in a multi-resolutional framework. The snake is initialized on a low-resolution image. Then, the final position of the snake at the low resolution is used as the initial position in the higher-resolution image. The area of the protein is estimated as the area enclosed by the final position of the snake.

## METHODS

Figure 1A shows a typical SDS-PAGE gel image. The vertical lines containing several protein bands in this gel image are called lanes; Figure 1B shows one of them. The SDS-PAGE image is composed of three parts. (i) The background does not carry any useful information. (ii) The second part corresponding to the images of protein bands have darker appearance. (iii) The proteins of interest appear as dark bands. The ratio of total area of the purified protein band region to the total area of other proteins demonstrates the efficiency of each purification step. Figure 2 shows the histograms of the three regions corresponding to the significant protein, other

proteins and the background on a typical lane. The histograms of the regions are obtained by manually segmenting the SDS-PAGE image.

The regions containing POI are nearly ellipse-like regions, which have the smallest gray level on the image. Thus, the boundary of such a region is an ellipse-like closed contour having small total intensity and high gradient.

Snakes are deformable contour models that are increasingly used in image segmentation (4,6). They are planar curves. The shape and location of the curve is determined according to an objective function. Snakes are influenced by the image structure, and the ROI is enclosed by the snake by minimizing an “energy” expression. The total energy of the snake around a closed contour,  $v(s)$ , is defined in Equation 1:

$$E[v(s)] = \oint_v [E_{int}(s) + E_{ext}(s)] ds \quad [\text{Eq. 1}]$$

where  $E_{int}$  and  $E_{ext}$  represent the internal and external energies of the snake, respectively.

The internal energy,  $E_{int}$ , is determined according to the length and bending of the snake and is defined by Equation 2:

$$E_{int} = \mathbf{a}E_l + \mathbf{b}E_b \quad [\text{Eq. 2}]$$

where the first term denotes the length energy,  $E_l$ , the second term denotes the bending energy,  $E_b$ , and the weights  $\mathbf{a}$  and  $\mathbf{b}$  control the relative importance of these energy terms in the total energy. In digital implementation, the length energy is computed simply as the sum of the distances between *snaxels*, which are the control points of the snake curve. The curve is determined by the snaxel points and the pixels on the lines connecting the snaxels. In this study, the number of snaxels is experimentally determined, and a first order (linear) interpolation is used to draw the entire snake curve encircling regions of interest that are protein bands.

If snaxels on the image,  $s_i$  are located at  $(x_i^*, y_i^*)$ , for  $i = 1, \dots, N$ , then the length energy,  $E_l$ , is computed as in Equation 3:

$$E_l = \sum_{i=1}^N \sqrt{(x_i^* - x_{i+1}^*)^2 + (y_i^* - y_{i+1}^*)^2} \quad [\text{Eq. 3}]$$

where  $x_{N+1}^* = x_1^*$  and  $y_{N+1}^* = y_1^*$ , as the snake is a closed contour.

The bending energy,  $E_b$ , controls the smoothness of the snake curve. A suitable measure for the smoothness is the second derivative along the contour. The bending energy can be approximated in digital implementation as in Equation 4:

$$E_b = \sum_{i=1}^N \sqrt{(x_{i-1}^* - 2x_i^* + x_{i+1}^*)^2 + (y_{i-1}^* - 2y_i^* + y_{i+1}^*)^2} \quad [\text{Eq. 4}]$$

where  $x_{N+1}^* = x_1^*$ ,  $y_{N+1}^* = y_1^*$ ,  $x_0^* = x_N^*$  and  $y_0^* = y_N^*$ .

The external energy,  $E_{ext}$ , represents the energy created by the image structure, depending on the location of the snake on the image; it is defined as in Equation 5:

$$E_{ext} = w_n E_n + w_e E_e \quad [\text{Eq. 5}]$$

where the first term,  $E_n$ , defines the line energy, the second term,  $E_e$ , defines the edge energy and the weights,  $w_n$ ,  $w_e$ , determine the contribution of line and edge energies to the total energy, respectively. The line energy is computed simply by adding pixel intensity values,  $I[x,y]$ , on the snake curve, as in Equation 6:

$$E_n = \sum_{j=1}^M I[x_j, y_j] \quad [\text{Eq. 6}]$$

where the index  $j$  runs over all the  $M$  pixels along the snake curve.

The second external energy component,  $E_e$ , is basically the gradient approximation of the underlying image along the snake curve, as in Equation 7:

$$E_e = \sum_{j=1}^M [(I[x_{j+1}, y_j] - I[x_{j-1}, y_j])^2 + (I[x_j, y_{j+1}] - I[x_j, y_{j-1}])^2] \quad [\text{Eq. 7}]$$

where the index  $j$  runs over all the  $M$  pixels along the snake curve. This is not the only form of gradient approximation; however, for our purposes, it is satisfactory.

Our application is based on the fast, greedy snake algorithm described in Reference 8. It is an iterative algorithm, and at each iteration, the snake curve changes in such a way that its total energy is minimized. The algorithm is adapted to our problem so that the resultant snake curve has an ellipse-like shape. This is provided by putting constraints on values of the internal angles of the snake. Initially six snaxels are placed evenly around a circle determined by the peaks of the 1-D profile. Table 1 shows the steps of this algorithm.

## RESULTS

The snake algorithm is implemented in a multi-resolution framework or in the wavelet transform domain (7). The original SDS-PAGE image is first processed by the filter banks previously described (3,5), and low-resolution, quarter-size sub-images ( $I_{ll}$ ,  $I_{lh}$ ,  $I_{hl}$  and  $I_{hh}$ ) are obtained. The snakes are

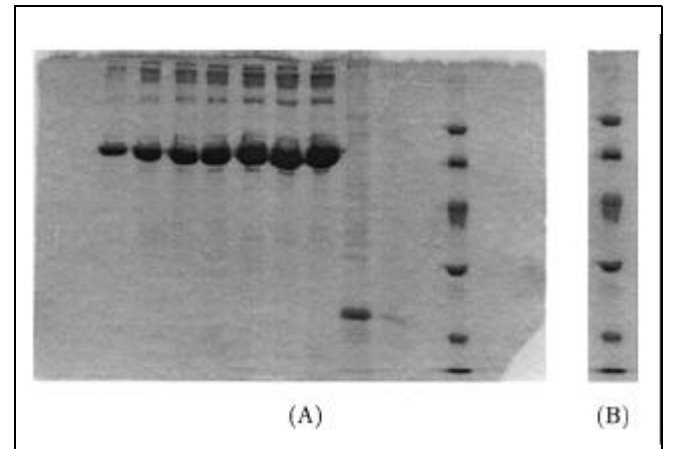


Figure 1. (A) Denaturing PAGE image; (B) a lane of the denaturing PAGE image.

initialized on the low-low sub-image,  $I_{ll}$ . The initial form of the snakes are circles centered at the peak points of the 1-D profile of the lane image. Another initialization is based on the  $I_{lh}$ ,  $I_{hl}$  and  $I_{hh}$  sub-images, which contain the edge information about the original SDS-PAGE image. Therefore, the edges of the protein band region can be detected from these sub-images and used as the initial positions of the snakes.

After the initialization on the low-resolution image, the snake adapts itself to the contour around the protein band in an iterative manner, as shown in Table 1, steps 2–5. The final snake obtained on the low-resolution image is used as the initial estimate on a finer-resolution SDS-PAGE image. This process is repeated until the snake has closely covered the

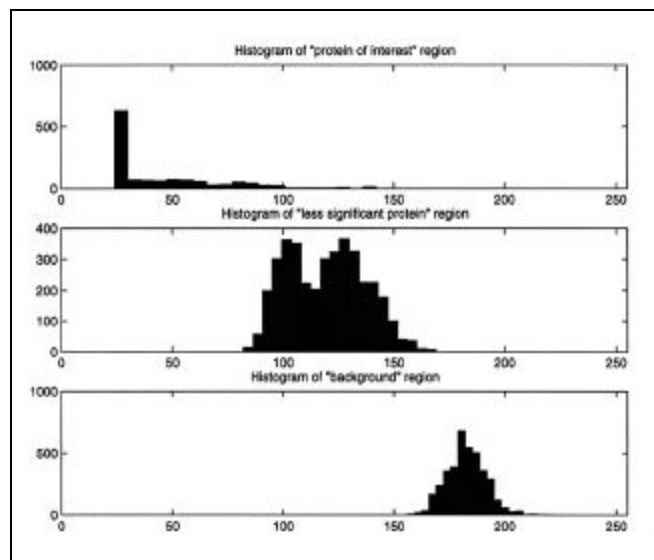


Figure 2. Histograms of three different regions of a PAGE image.

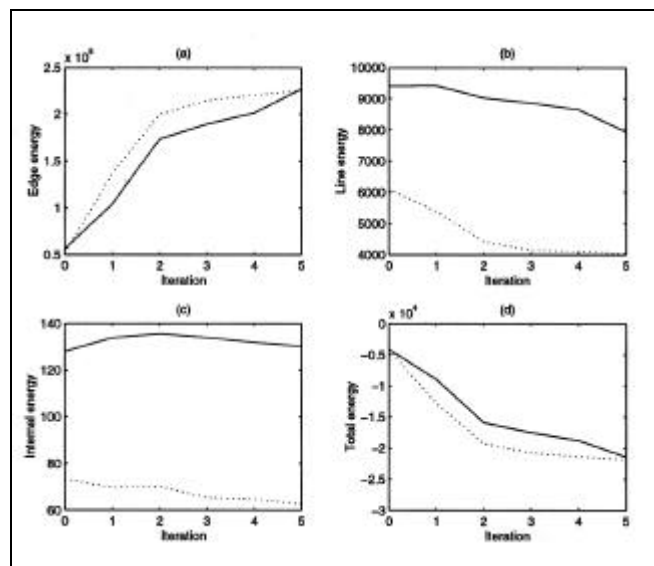


Figure 3. Change of different energy components. Solid lines indicate the energy changes in the high-resolution image, and dashed lines show the energy changes in the lower-resolution image.

Table 1. The Snake Algorithm

1. Get the image  $I[x,y]$ .
2. Determine the initial positions of the snaxels,  $v[i] = (x[i], y[i])$ ,  $i = 1, \dots, N$  (number of snaxels) from the 1-D profile.
3. Calculate the total energy of the snake curve,  $E_{total}$ .
4. The total energy,  $E_{total}$ , is assigned the minimum energy,  $E_{min}$ , i.e.,  $E_{min} = E_{total}$ .
5. For each snaxel  $v[i]$ ,  $i: 1 \rightarrow N$ 
  - 5.1  $E_{min} = E_{total}$ .
  - 5.2 For  $M$  pixels in the neighborhood of each snaxel,  $j: 1 \rightarrow M$ .
    - 5.2.1 Calculate the length energy,  $E_l$ , as in Equation 3.
    - 5.2.2 Calculate the bending energy,  $E_b$ , as in Equation 4.
    - 5.2.3 Calculate the line energy,  $E_n$ , as in Equation 6.
    - 5.2.4 Calculate the edge energy,  $E_e$ , as in Equation 7.
    - 5.2.5  $E[j] = \alpha E_l + \beta E_b + w_n E_n + w_e E_e$ .
    - 5.2.6 If  $E[j] < E_{min}$ 
      - 5.2.6.1 The minimum energy is replaced by the new value  $E_{min} = E[j]$ .
      - 5.2.6.2 The snaxel point is replaced by the image pixel,  $j = j_{min}$ .
      - 5.2.6.3 Move the snaxel  $v_j$  to  $v_{j_{min}}$ .
  - 5.3 Calculate the angle,  $f$  between two lines connecting three snaxels,  $f = \text{angle}(i-1, i, i+1)$ .
  - 5.4 If  $f < 90^\circ$ 
    - 5.4.1 The snaxel  $v_j$  is moved to in between  $v_{i-1}$ ,  $v_{i+1}$  until the angle becomes a wide angle.

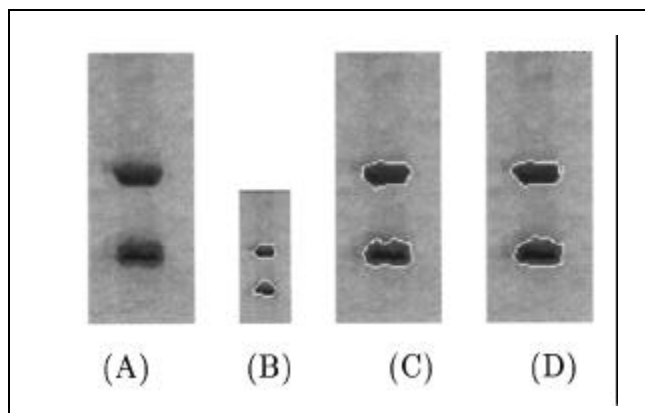


Figure 4. POI is enclosed within the snake: (A) original lane image; (B) snake obtained using only the low-resolution image; (C) snake obtained using only the high-resolution image; (D) snake obtained using the multi-resolution approach.

Table 2. Number of Computations for High- and Low-Resolution

Resolution	Edge	Line	Length	Bending
High	3517	3517	648	324
Low	756	756	132	66

protein band region in the original full-resolution SDS-PAGE image.

The behavior of the different energy components and the total energy of a snake at several iterations at different resolutions are given in Figure 3 for the SDS-PAGE image shown in Figure 1. At the beginning of the convergence process, the slope of the total energy of the snake in the low-resolution image,  $E_{low}$ , is higher than that of the snake in the high-resolution image,  $E_{high}$ , as shown in Figure 3D. In other words, as shown in Equation 8:

$$\left| \frac{E_{low}[k+1]}{E_{low}[k]} \right| > \left| \frac{E_{high}[k+1]}{E_{high}[k]} \right| \quad [\text{Eq. 8}]$$

This means that the snake converges faster on the low-resolution image. This observation verifies the validity of the

multi-resolutional approach. In our simulation studies, we use only one level of sub-band decomposition, but this methodology can be considered in multiple resolution levels as well.

The energy weights defined in Equations 2 and 5 ( $\alpha, \beta, w_n$

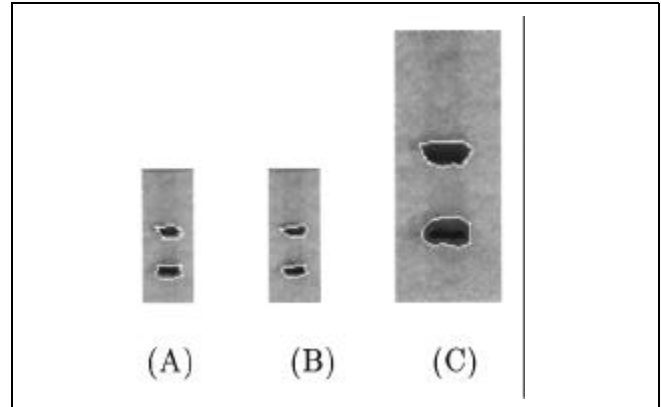


Figure 5. Results of the initialization process: (A) initial position of the snake located on the extracted edge; (B) the snake in Panel A after one iteration on low-resolution image; (C) snake obtained using the multi-resolutional approach with the initial snake in Panel A.

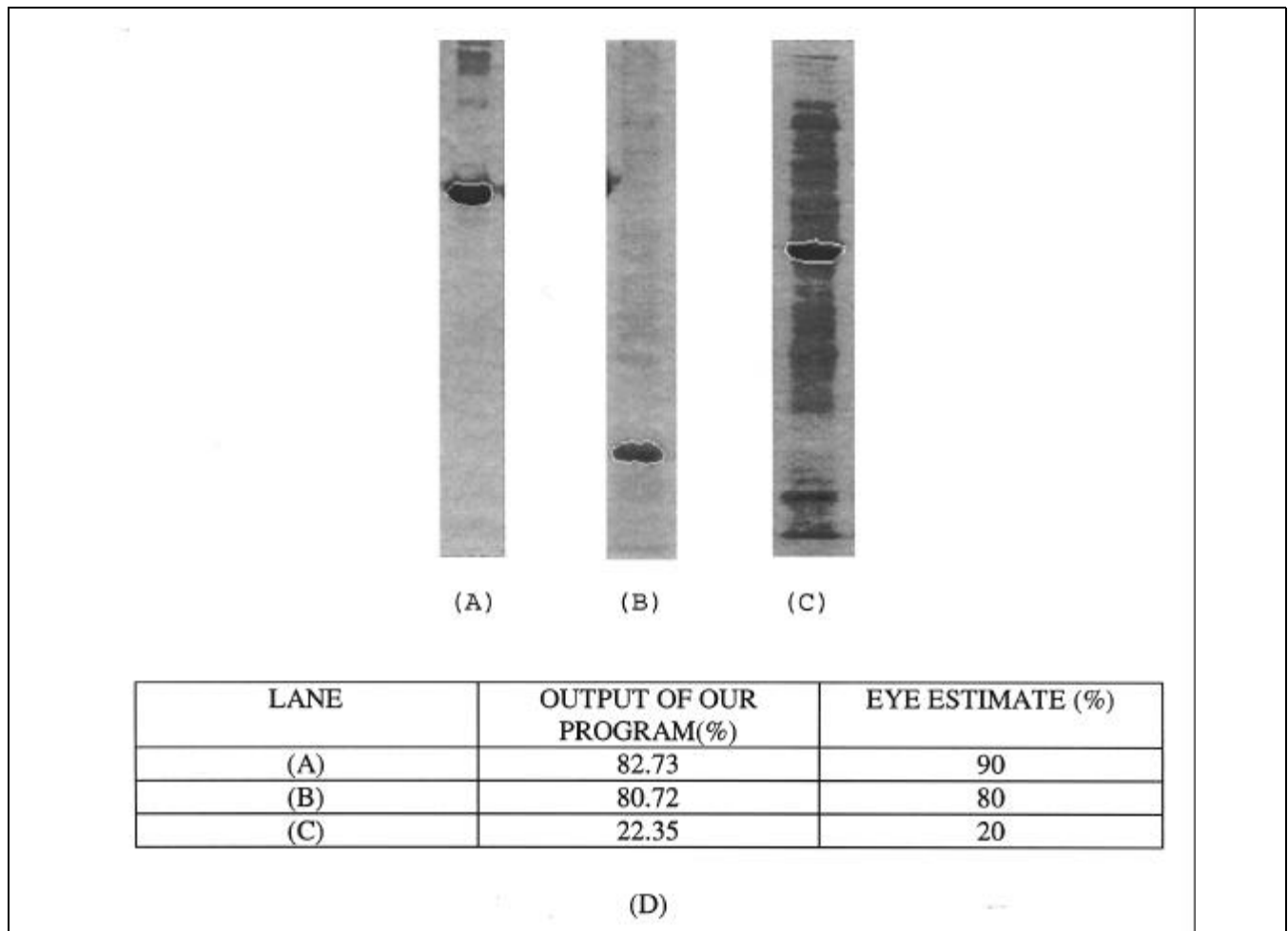


Figure 6. Different ROI for percentage calculations.

and  $w_e$ ) must be adapted to the energy level changes in different resolutions. In the multi-resolutional scheme, at the final iteration of the low-resolution image,  $\alpha$ ,  $\beta$  and  $w_n$  values are halved, and the iterations are continued in the higher-resolution image. At the low resolution, these values are  $\alpha = 4$ ,  $\beta = 4$ ,  $w_n = 0.2$  and  $w_e = -0.1$ .

Figure 4 shows an original lane of the image and the final positions of different snakes obtained using different methodologies. The result in Figure 4C is obtained after six iterations of the snake algorithm on the original image. Figure 4D shows the result of the multi-resolutional approach. The snake obtained at the low resolution after two iterations (Figure 4B) is used as the initial position of the snake in the high-resolution image. Figure 4D shows the results of another three iterations in the high-resolution image. The comparison of Figures 4C and 4D indicates that the final snake converges better to the boundary of the protein area if a multi-resolutional approach is used. Additionally, the number of computations required in the multi-resolutional approach is smaller.

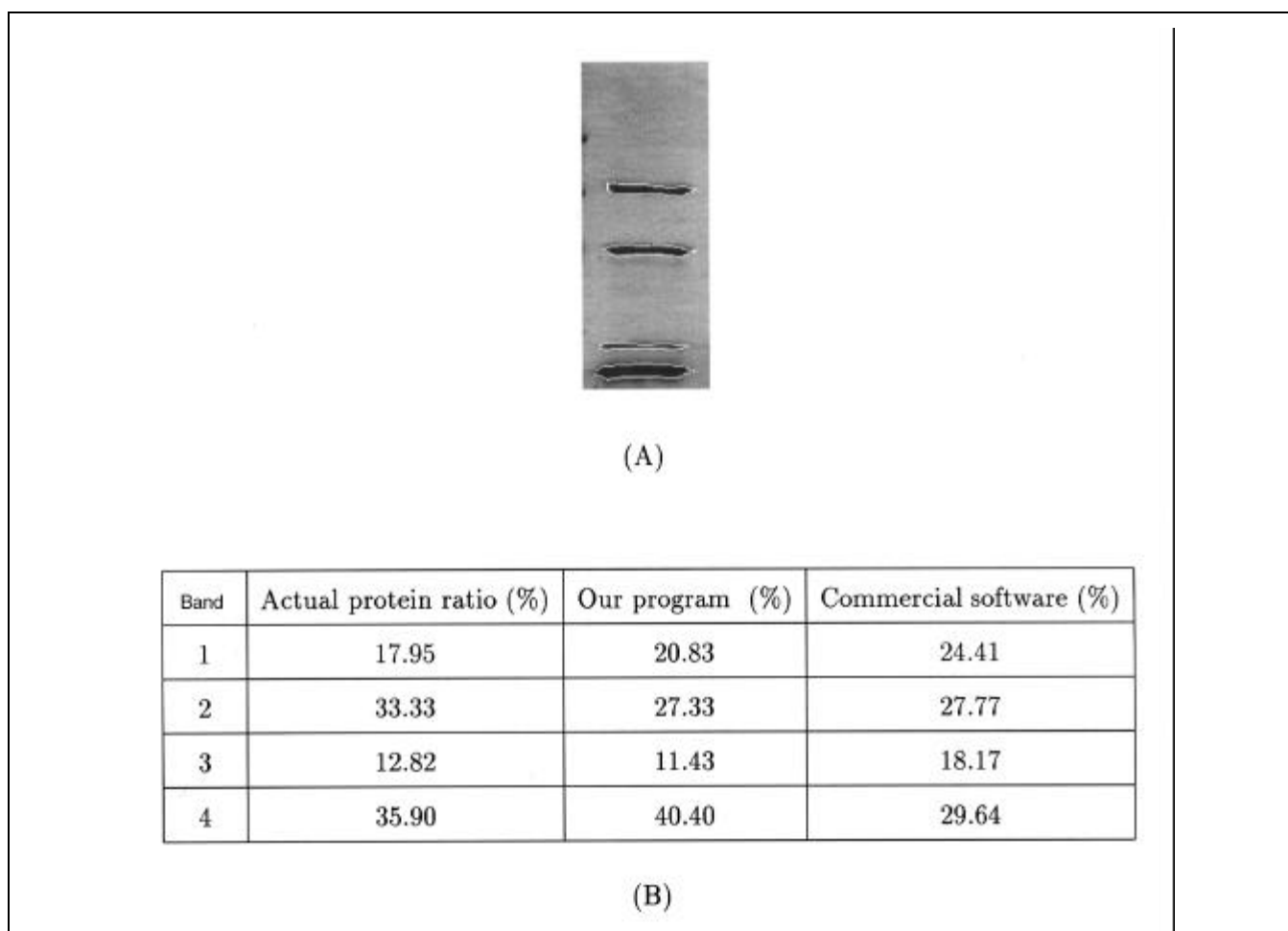
Figure 5 shows the results of the initialization procedure based on the edge information extracted from sub-images  $I_{ll}$ ,  $I_{hl}$  and  $I_{hh}$ . Figure 5A depicts the initial position of the snake

at the sub-image  $I_{ll}$ . The six snakes are located on the extracted edges of the protein region. This snake takes the form shown in Figure 5B after only one iteration. This form is very close to the final position of the snake on the low-resolution image obtained with the other initialization procedure based on the 1-D profile of the lane image. Figure 5C shows the final position of the snake on the high-resolution image after two iterations. There is no significant difference between this result and the result shown in Figure 4D, which is obtained with the other initialization procedure. Edge information-based initialization procedures produce similar results with less number of iterations. However, this initialization procedure requires the extraction of edges from the sub-images.

The ratio of the amount of POI to the amount of insignificant proteins is estimated by Equation 9:

$$100 \times \frac{A_S}{A_T} \quad [\text{Eq. 9}]$$

where  $A_S$  is the area of the region enclosed by the snake, that is, the region of POI;  $A_T$  is the total area of the protein band images. The area,  $A_T$ , is estimated by counting the number of pixels representing all protein band regions, which have gray-



**Figure 7.** (A) SDS-PAGE image of known amounts of protein bands; (B) actual protein amount ratios determined by Bradford protein assay and comparisons of percentage calculations by our software and a commercial software.

level values smaller than a threshold value. The threshold is calculated using the 1-D vertical profile of lanes. The profile is calculated by taking the average of pixel intensity values along the horizontal direction, and the threshold is determined according to the maximum gray level in the profile. The effect of noise in the image is reduced by the averaging operation. Once the peak values are determined according to the initial threshold, the snake algorithm determines the boundaries automatically.

Using this methodology, the percentage of POI is estimated in the examples shown in Figure 6, A, B and C. In all three images, the snake closely encircles the band of POI. The percentages are found to be 82.73%, 80.72% and 22.35% for the lanes in Figure 6, A, B and C, respectively. By eye, the percentages are estimated as 90%, 80% and 20%, respectively. These are tabulated in Figure 6D. In the first example (Figure 6A), our software provides more accurate concentration estimates of POI.

Our method outperforms a commercially available software program that determines the bands by thresholding and rectangular approximation. In an additional experiment, we compared the protein band ratios using our algorithm and the commercial software for known amounts of protein determined by Bradford protein assay (2) for each protein band separately. Figure 7 shows the tabulated results. The actual percentages are 17.95%, 33.33%, 12.82% and 35.90%. Our software estimates them as 20.83%, 27.33%, 11.43% and 40.40%, compared with 24.41%, 27.77%, 18.17% and 29.64% as estimated by the commercial software. The squared values of the differences between the actual percentages and the results of our program and the commercial program are 66.47 and 140.45, respectively. This indicates that our software provides better results than the commercially available software.

## DISCUSSION

In this work, we present a multi-resolution snake algorithm for the identification of relative protein bands in PAGE images. The algorithm is more accurate compared to usual eye exams or to some commercially available programs that utilize rectangular approximations.

The advantage of this multi-resolutional approach is three-fold: (i) the snake algorithm converges better, (ii) it brings robustness to the process; i.e., it is not easily affected by the scanning artifacts in the original SDS-PAGE image and (iii) computational cost is less since smaller images are processed in lower resolutions.

For each iteration of energy calculations, the number of computations depends on the length of the snake and the radius of the search window. With search radius of  $n$  pixels and snake length of  $N$  pixels, the number of computations for a single iteration of energy calculations is  $0(Nn^2)$  (8). In this study, the number of snaxels and the radius of the search window are halved in the lower-resolution images. Therefore, the ratio,  $R$ , of the total computational complexity in the low-resolution image to that in the high-resolution image is as in Equation 10:

$$R = \frac{(n+1)^2}{2 \times (2n+1)^2} \quad [\text{Eq. 10}]$$

where  $(n+1)^2$  and  $(2n+1)^2$  are areas of search windows at the high- and low-resolution images, respectively.

Table 1 shows the number of times each energy component is computed on high- and low-resolution images at the last iteration. The total number of energy-component computations in the low-resolution image is reduced to approximately one-fourth of that in the high-resolution image. This figure is almost equivalent to the value we get if we take  $n = 2$  in Equation 10.

The multi-resolutional snake framework might also find applications in other electrophoretic techniques that require comparative band-image quantifications.

## AVAILABILITY

The computer program described in this paper as well as some sample input and output images are freely available on our Web page: (<http://www.cs.bilkent.edu.tr/~koyuturk/PAGE.html>). You can also obtain the program from the **Software Library** at the BioTechniques Web site ([www.BioTechniques.com](http://www.BioTechniques.com)).

## REFERENCES

1. **Allen, R.C., C.A. Savaris and H.R. Marier.** 1984. Gel Electrophoresis and Isoelectric Focusing of Proteins: Selected Techniques. Walter de Gruyter, Berlin.
2. **Bradford, M.M.** 1976. A rapid and sensitive method for quantitation of microgram quantities of protein utilizing the principle of protein-dye binding. *Anal. Biochem.* 72:248-254.
3. **Çetin, A.E., Ö.N. Gerek and Ş. Ulukuş.** 1993. Block wavelet transforms for image coding. *IEEE Trans. Cir. Sys. Video Tech.* 6:433-435.
4. **Kass, M., A. Witkin and D. Terzopoulos.** 1988. Snakes: active contour models. *Int. J. Computer Vision* 1:321-331.
5. **Kim, C.W., R. Ansari and A.E. Çetin.** 1992. A class of linear-phase regular biorthogonal wavelets. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'92)*, San Francisco.
6. **Leymarie, F. and M.D. Levine.** 1993. Tracking deformable objects in the plane using an active contour model. *IEEE Trans. Pattern Anal. Mach. Intell.* 15:617-633.
7. **Vetterli, M. and J. Kovacevic.** 1995. *Wavelets and Subband Coding*. Prentice Hall, New Jersey.
8. **Williams, D.J. and M. Shah.** 1990. A fast algorithm for active contours. *IEEE Trans. Pattern Anal. Mach. Intell.* 4:592-595.

Received 9 September 1998; accepted 24 February 1999.

### Address correspondence to:

Rengül Çetin-Atalay  
 Department of Molecular Biology and Genetics  
 Bilkent University  
 Bilkent, Ankara, TR-06533, Turkey  
 Internet: [cetin@ee.bilkent.edu.tr](mailto:cetin@ee.bilkent.edu.tr)