Original papers

John J. Tyson[1]
Robert H. Haralick[2]*

Departments of [1]Biology, [2]Computer Science, and [2]Electrical Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA

Computer analysis of two-dimensional gels by a general image processing system

In order to survey changes in accumulation of several hundred proteins during the naturally synchronous nuclear division cycle of *Physarum polycephalum*, we have developed methods for analyzing two-dimensional (2-D) gel electrophoretograms using the general image processing system developed by the Spatial Data Analysis Laboratory at Virginia Tech. In this paper we describe fast and accurate methods for removing non-homogeneous background intensity from a 2-D gel image, for resolving overlapping protein spots, and for estimating the total integrated intensity in a protein spot by Gaussian modeling.

1 Introduction

Pahlic and Tyson [1, 2] have initiated a study of periodic changes in plasmodial proteins during the naturally synchronous nuclear division cycle of the acellular slime mold, *Physarum polycephalum*. Plasmodia are harvested at various stages in the cycle, total cellular proteins are separated by two-dimensional (2-D) gel electrophoresis, spots are revealed by silver staining, digital images of the gels are generated, and these images are processed numerically in order to extract full quantitative detail from the 2-D gels. There are several excellent packages of subroutines specifically designed for digital analysis of 2-D gel electrophoretograms [3–6], and we could have used any one of these packages for our cell cycle study. However, the Spatial Data Analysis Laboratory at this University has developed a general image processing system (GIPSY) implemented on hardware dedicated to image processing tasks, and we decided to apply this general system to the specific needs of gel analysis. This decision was based on economic and temporal constraints (the cost of implementing software from some other source) as well as a desire to expand the capabilities of GIPSY to handle the task of 2-D gel analysis. Furthermore, the flexible, "building block" approach that GIPSY takes to image processing makes it an ideal tool for exploratory research: in search of the best analytical procedure to solve a given problem one can easily try several different approaches by piecing together elementary GIPSY subroutines to implement the various procedures under consideration. In this paper we describe the power and flexibility of GIPSY subroutines to analyze 2-D gels, and we provide a technical description of the calculations on which our cell-cycle study was based.

2 Materials and methods

2.1 Cell culture and gel electrophoresis

Methods of cell culture and 2-D gel electrophoresis employed by Pahlic and Tyson are described elsewhere [1, 2]. Essential-ly, gels were prepared as described by O'Farrell [7] and stained as described by Merril *et al.* [8].

2.2 Photography and scanning

Back-lighted gels were photographed in black-and-white using Kodak Pan X film. Four gels were photographed on each 35 mm frame, along with a photographic step tablet (Kodak No. 2, 21 discrete steps of OD from 0.05 to 3.05). The film was developed as recommended by Kodak, and then scanned with a custom-built laser scanner which took readings of absorbance (-log transmittance) at 50 micron intervals from left to right and top to bottom. This corresponds to a step size of about 400 microns on the original gels, and generates a digital image of approximately 100 000 absorbance readings per gel. At this resolution, small but unambiguous spots were represented by 5–10 absorbance readings and large spots by 50–100 readings. Absorbance readings were recorded on magnetic tape on a scale 0–255. These numbers are referred to as gray-tones and each reading is one "pixel" (*i. e.*, picture element). The film-response curve was measured by plotting the average gray-tone at the center of each step of the step tablet against the optical density of that step. Gray-tone was found to be linearly related to optical density up to OD = 1.6 (above which measurements were not taken). This range is more than sufficient to cover the significant OD values generated by the silver stain.

2.3 Hardware

At Virginia Tech the general image processing system is implemented on a DEC VAX 11/780 computer with 6 megabytes of memory and 2500 megabytes of mass storage. Images are displayed on Aydin high resolution color monitors driven by ADI $512 \times 512 \times 8$ bit graphics display generators. For hardcopy output the color monitors are interfaced to the Matrix 4007 system for 8×10 polacolar instant prints or 35 mm negatives.

3 Image processing

3.1 General approach

Our approach to ditital analysis of 2-D gel electrophoretograms is similar in design to the systems described by Bos-

singer et al. [3], Vo et al. [4], Taylor et al. [5], and Lemkin and Lipkin [6]. The processing of digitized images of 2-D gels can be divided into several stages: (i) removal of random noise from the raw data, (ii) subtraction of local background intensity from the image, (iii) detection and labeling of protein spots, (iv) resolution of overlapping spots, (v) transformation of images to a common coordinate system to achieve optimal registration of corresponding proteins on different gels, (vi) integration of gray-tone surface to measure total amount of protein in each spot, (vii) analysis of protein content in corresponding spots on different gels to find reproducible, statistically significant, cell-cycle related fluctuations in specific proteins. Similarities and differences between our approach and the systems described previously in the literature will be pointed out subsequently, as each step is described in detail.

3.2 GIPSY

GIPSY is an interactive software package designed to be applicable to a wide variety of image-processing tasks and to be implementable on computer hardware from different manufacturers. To achieve these goals, GIPSY is divided into three levels: the kernel, the application subroutines, and the command-processing module. The application subroutines do the actual image processing, the kernel provides an interface between the application subroutines and the machine on which they are implemented, and the command-processing module provides an interface between the application subroutines and the user. Image processing in GIPSY is accomplished by concatenating strings of elementary operations. Each elementary operation is written (in "RATional FORtran") as an application subroutine, accessible through the command-processing module. Each command typically consists of the input file(s), the output file(s), and various flags to specify options. The command-processing module prompts for any necessary information. There are over three hundred commands to do image filtering, classification, geometric spatial transformations, numeric and symbolic recursive and non-recursive neighborhood operations, region growing, and property file generation. This "building block" approach gives the user great flexibility: the elementary operations can be put together in many different ways to accomplish many different image-processing tasks. Some steps in the processing of a 2-D gel image can be accomplished by a single GIPSY command, whereas other steps require the concatenation of elementary operations. In the latter case, the appropriate string of elementary operations are gathered together in a "RUN" file, and the command-processing module automatically applies the individual commands in proper sequence to the input image(s). This feature of GIPSY gives the researcher the flexibility to test a variety of approaches to a given task, but once the best approach is determined, it can be implemented automatically by the command-processing module.

3.3 Signal averaging

To compensate for random noise in the OD readings, a filtering operation was performed on the image. The gray-tone surface in a 5×5 neighborhood of each pixel was modeled by a least-squares best-fitting quadratic surface. In each neighborhood, the gray-tone of the fitted quadratic surface at the central pixel is assigned to the central pixel in the filtered image. As described by Vo et al. [4], this smoothing algorithm

has many advantages over simple averaging and median filtering, as used in most other systems.

3.4 Local background removal

The separation of stained protein from non-specific background staining is accomplished in several steps: (i) local minima of the gray-tone surface are identified, (ii) these minimum gray-tones are propagated over the entire image, thereby assigning to each pixel the value of its spatially closest minimum, (iii) the resulting discontinuities in the image are "smeared" out, and (iv) the average local background is subtracted from the original filtered image. Some comments are in order about each step. The smoothed digital image of a 2-D gel can be thought of as regularly spaced measurements of the heights of a smooth surface above a plane. Like any familiar landscape, such a surface will have peaks (local maxima), pits (local minima), slopes, ridges, valleys and saddle points. These features can be identified by standard image-processing routines: that is, each pixel can be classified as a peak, pit, slope, etc. In general, peaks correspond to the centers of protein spots, slopes to the sides of spots, and pits to the regions farthest removed from any protein. Thus, just as we can use local maxima of the gray-tone surface to identify protein spots, we can use local minima to identify regions of "background staining", i. e., non-specific coloration of the gel by the silver reagents.

Our use of minima of the gray-tone surface as estimators of local background intensity differs from any other approach in the literature. Most groups use more-or-less sophisticated algorithms to locate the "central" regions of all protein spots and then use the complementary set (those pixels not assigned to spots) to define the background. Some average over the background pixels then is defined as the average local background. Since all spot-finding algorithms are conservative in drawing the "boundaries" of a spot, there is a great deal of spill-over from the fringes of protein spots into the complementary set (the "background"). Thus, the average local background defined in this way is likely to overestimate the true background intensity. We contend that local minima of the gray-tone surface are better estimators of local background intensity.

Our intent is to construct an image which, at each pixel, has an estimate of the average local background gray-tone. By identifying local minima we can construct an image containing local minimum gray-tone values at their respective locations and containing no significant information (gray-tone $= 0$) at all those pixels which are not at local minima of the gray-tone surface. This image is sparse: only a few isolated pixels contain significant information (the local minimum gray-tone values); the great majority of pixels are "empty" (gray-tone $= 0$). The next task is to propagate the local minimum values into the "holes" in the image (where gray-tone $= 0$). In the standard image-processing repertoire, holes in an image are filled recursively by propagating gray-tones from the boundary of a hole into the immediate interior. However, for the problem described here most of the image is "hole" and only a few pixels are on the "boundary" of significant regions. To fill the holes recursively consumes a great deal of central processing unit (CPU) time, so we have designed a new "fast-fill" subroutine which propagates minimum values across the entire image in only two passes. The two passes are those required to calculate

the distance (4-distance or 8-distance) of a pixel from the nearest boundary by the method of Rosenfeld and Pfalz [9]. From this algorithm can be generated not only the distance from but also the identity of the nearest boundary pixel, and the gray-tone of a pixel in the "hole" can be identified to the gray-tone of the nearest boundary pixel. In a test run on a typical image of 10^5 pixels, of which only 10^3 pixels contained local minimum values, the fast-fill subroutine was four times faster than the recursive-fill subroutine. The image filled in this way has sharp borders between regions of identical gray-tones. To smear out these discontinuities the image is smoothed by assigning to each pixel the (arithmetical) average gray-tone for a 10×10 window around the pixel of interest. This provides an estimate of the average local background at each pixel, and it is subtracted from the original filtered image.

3.5 Spot segmentation

Once the average local background has been subtracted from the image, regions of major protein staining can be identified by a simple threshold criterion: each pixel for which the gray-tone value exceeds a certain threshold is classified as belonging to a region of major protein staining. (Let M denote the set of all such pixels.) Most other published systems for gel analysis try to identify protein spots before correcting the image for non-uniform background gray-tone levels, and therefore they cannot use a simple threshold criterion but must resort to considerably more complicated procedures. Vo et al. [4] take the same approach as we, subtracting background before detecting spots.

Let U be the set of all pixels in the image, M the set of pixels comprising regions of major protein staining, W the set of pixels comprising regions of minor protein staining, and I the set of pixels with gray-tone values too close to background levels to be given significance. The sets M, W and I are mutually disjoint, and their union is the set $U (U = M \cup W \cup I)$. At this point we have identified the set M by the threshold criterion, and there remains the hard task of separating the set $U \backslash M$ (the complement to M) into W and I. To do this we calculate the average and standard deviation of gray-tone values on the complementary set $U \backslash M$. (Remember that average local background has already been removed.) Those portions of the gray-tone surface on $U \backslash M$ that rise more than one standard deviation above the average local gray-tone are identified as regions of minor but significant protein staining (i. e., set W). Notice that, along with the minor peaks, the shoulders of the major peaks will automatically show up in the set W. All pixels not in M or W are assigned to the "insignificant set", $I = (U \backslash M) \backslash W$. Now let P be the set of all pixels comprising regions of significant protein staining; $P = M \cup W$. On the set P we assign to each pixel the gray-tone value of the original filtered image with background removed. On the set $I = U \backslash P$ we set the gray-tone level at each pixel to 0.

On this image individual spots are now identified by searching for all local maxima in the gray-tone surface. The maxima are numbered in the order they are found. To determine the extent of a spot (i. e., to draw its boundary) the computer calculates what are called "reachability sets". In this instance, the computer searches the image in all directions from each local maximum: any pixel that can be reached by a monotonically decreasing path (an ordered sequence of pixels, proceeding away from a maximum, along which gray-tone values never

increase) is assigned to the reachability set of that maximum, unless the pixel has an associated gray-tone of 0 (in which case we have reached the edge of the significant region of the image) or unless the pixel can also be reached by a monotonically decreasing path from some other maximum (in which case the pixel lies in a valley between two peaks and is arbitrarily assigned to one or the other reachability set). In this way, overlapping spots are automatically resolved. Our procedure for spot segmentation accomplishes much the same objective as the approach described by Vo et al. [4]. No special precautions are taken to handle streaks, which are split into as many spots as there are local maxima in the streak.

In combination, our procedures of signal averaging, local background removal and spot segmentation do an excellent job in resolving a digital gel image into individual protein spots, as judged by a comparison of the computer-generated map of spots with visual examination of the original gel. The only obvious difficulty is that occasionally, especially in heavily stained regions, the programs will assign two or more spots on one gel in an area which appears as a single spot on another gel. Such cases of possibly artificial splitting or coalescence of spots could be minimized by further subroutines, but we have not addressed this problem directly since it rarely occurs for the proteins we have analyzed and can be corrected on an individual basis when pairs of gels are compared.

3.6 Rubber-sheeting the gel images

Because of uncontrolled variations in the separation of proteins along the isoelectric-point and molecular-weight dimensions of 2-D gels of different samples, it is not possible to superimpose the image of one gel onto another and get a reasonable match of corresponding proteins. In order to accomplish this match, one of the gels must first be transformed (i. e., translated, stretched, rotated, etc.) so that it matches as closely as possible some reference gel. This is done digitally by determining the coefficients of an arbitrary second-order polynomial transformation of the row and column coordinate system:

$$r' = a_1 + a_2 r + a_3 c + a_4 r^2 + a_5 rc + a_6 c^2$$
$$c' = b_1 + b_2 r + b_3 c + b_4 r^2 + b_5 rc + b_6 c^2 \qquad (1)$$

where r, c are coordinates in the reference gel, r', c' are corresponding coordinates in the gel to be transformed, and a_1, ..., a_6, b_1, ..., b_6 are undetermined coefficients. To determine these coefficients one specifies the coordinates of at least seven (preferably twenty or more) "landmark" spots on the two gels to be matched, and then chooses the coefficients so as to minimize the sum of the squares of the deviations of the transformed coordinates from their goal positions. For our purposes (see below) it has proven sufficient to use a single second-order polynomial transformation for the entire image. Third-order polynomial transformations do not provide a better overall match of two gels: the landmark spots are brought into closer correspondence of course, but the rest of the image is seriously distorted. The identification of the landmark spots (spread more-or-less uniformly over a gel) is accomplished by positioning a cursor on the image of a gel displayed on a television screen. The computer then calculates and remembers the coordinates of the selected landmark spot on the gel from the position of the cursor on the screen. Once the transformation is specified, one of the gels can be "rubber-sheeted" according-

ly. That is, to assign a gray-tone from gel B to a pixel in gel A's coordinate system, we use the transformation to locate the position on gel B (r', c') that corresponds to the pixel on gel A (r, c). The gray-tone at the pixel closest to r', c' on gel B is then assigned to the pixel at r, c in gel A's coordinate system.

3.7 Quantification of spot properties

Once the protein spots have been resolved into nonoverlapping reachability sets and transformed into the reference coordinate system, certain properties of the spots can be determined, such as position on the gel, maximum gray-tone, size and shape of spot. The most important property is the integral of the gray-tone surface over the extent of the spot because, as long as gray-tone is proportional to protein concentration in the gel, this integral is proportional to total protein content in the spot. We estimate the integral by modeling each spot with a least-squares best-fitting Gaussian surface, and then evaluating the integral of the Gaussian surface exactly. We assume in this section that gray-tones in the image are directly proportional to protein concentrations in the gel. If gray-tone is proportional to optical density but optical density is not linearly dependent on protein concentration in the gel, then gray-tone values must be adjusted at some point in the process to correct for the nonlinear dependence of optical density on protein concentration.

3.8 Gaussian model of an isolated spot

We suppose that the gray-tone surface of an isolated protein spot, $G(r, c)$, can be modeled by a general bivariate Gaussian distribution

$$G(r,c) = G_0 \, g(x), \qquad x = \begin{pmatrix} r-\bar{r} \\ c-\bar{c} \end{pmatrix} \tag{2}$$

where ($\bar{r}$, $\bar{c}$) are the coordinates of the center-of-mass of the spot and G_0 is some constant. The function $g(x)$ is the standard bivariate Gaussian distribution,

$$g(x) = (2\pi)^{-1} \, \Delta^{1/2} \, \exp\left\{ -(x'Ax)/2 \right\} \tag{3}$$

where x' is the transpose of the vector x, $\Delta = \det(A)$ and A^{-1} is the variance-covariance matrix,

$$A^{-1} = \begin{pmatrix} \sigma_{rr}^2 & \sigma_{rc}^2 \\ \sigma_{rc}^2 & \sigma_{cc}^2 \end{pmatrix} = \int\limits_{-\infty}^{\infty}\!\!\int xx' \, g(x) \, \mathrm{d}r \, \mathrm{d}c \tag{4}$$

Since $\int\!\int \geqq g(x) \, \mathrm{d}r\mathrm{d}c = 1$, the total protein content of a spot is proportional to $\int\!\int \geqq G(r,c)\,\mathrm{d}r\mathrm{d}c = G_0$. Thus, the multiplicative factor, G_0, in Eq. (2) is the number we desire. To determine G_0 we insist that $G_0 g(x)$ give the best possible fit, in a least-squares sense, to the observed gray-tone surface. That is, we minimize

$$\sum_{r,c} [G(r,c) - G_0 \, g(x)]^2$$

with respect to variation of G_0. Obviously,

$$G_0 = \sum_{r,c} Gg \Big/ \sum_{r,c} g^2 \tag{5}$$

In this and subsequent formulae, $\sum_{r,c}$ denotes a sum over all pixels within the identified boundary of a spot.

To calculate g at some pixel (r, c), we must know the elements of the matrix A. From the observed gray-tone surface we can calculate the second moments (about the mean) of $G(r,c)$:

$$V = \begin{pmatrix} VRR & VRC \\ VRC & VCC \end{pmatrix} \tag{6a}$$

$$VRR = \sum_{r,c} (r-\bar{r})^2 \, G(r,c) \Big/ \sum_{r,c} G(r,c) \tag{6b}$$

$$VRC = \sum_{r,c} (r-\bar{r})(c-\bar{c}) \, G(r,c) \Big/ \sum_{r,c} G(r,c) \tag{6c}$$

$$VCC = \sum_{r,c} (c-\bar{c})^2 \, G(r,c) \Big/ \sum_{r,c} G(r,c) \tag{6d}$$

The matrix V is not quite the same as A^{-1} because the sum only extends over pixels within a certain boundary determined by the threshold criteria used to identify spots. For an isolated, Gaussian shaped spot, this boundary should be the ellipse

$$E_a = \left\{ x \mid x'Ax = a^2 \right\}$$

where a is some constant. Thus, integrating over the area bounded by E_a, we find that

$$V = \int\!\int xx' \, G(r,c) \, \mathrm{d}r\mathrm{d}c \Big/ \int\!\int G(r,c)\,\mathrm{d}r\mathrm{d}c$$
$$= f(e^{-a^2/2}) \, A^{-1} \tag{7}$$

where

$$f(\rho) = [1 - \rho(1 - \ln\rho)]/(1-\rho) \tag{8}$$

and

$$\rho = e^{-a^2/2} = G_{\mathrm{bdy}}/G_{\mathrm{max}} \tag{9}$$

i. e., ρ is the ratio of gray-tone on the boundary to gray-tone at the peak of the spot.

Now we are ready to calculate G_0. From the average gray-tone on the boundary of a spot and the maximum gray-tone in the interior, calculate ρ. Then, using Eqs. (7) and (8), calculate the matrix A from the matrix V, computed according to Eq. (6). With A in hand, calculate g at each pixel within the boundary of the spot, and then calculate G_0 from Eq. (5). Theoretically, the method just described is applicable only to isolated spots, and in practice it gives quite reasonable results for more-or-less isolated spots (Fig. 1). In this case it has advantages over other Gaussian modeling routines [5, 10] in that (i) we permit correlations between the row and column directions (*i. e.*, we do not assume that the major and minor axes of the surface are parallel to the row and column directions), and (ii) we calculate the elements of the variance-covariance matrix directly from the second central moments of the gray-tone distribution so that there is only one parameter, G_0, that must be optimized by least-squares fitting. For the case of overlapping spots, our procedure should still be reasonably accurate while also being considerably simpler than the recursive optimization of multiparameter surfaces of groups of overlapping proteins [5]. Furthermore, most cases of severe overlap, where our simple method is not valid, occur near major protein spots where the gray-tone level exceeds the linear region of the silver-stain response.

3.9 Comparison of gels

The last step is to identify spots of interest and retrieve their properties. Gel comparison is made by displaying the two images in color on a television screen. If one image controls the intensity of the red beam and the other controls the intensity of
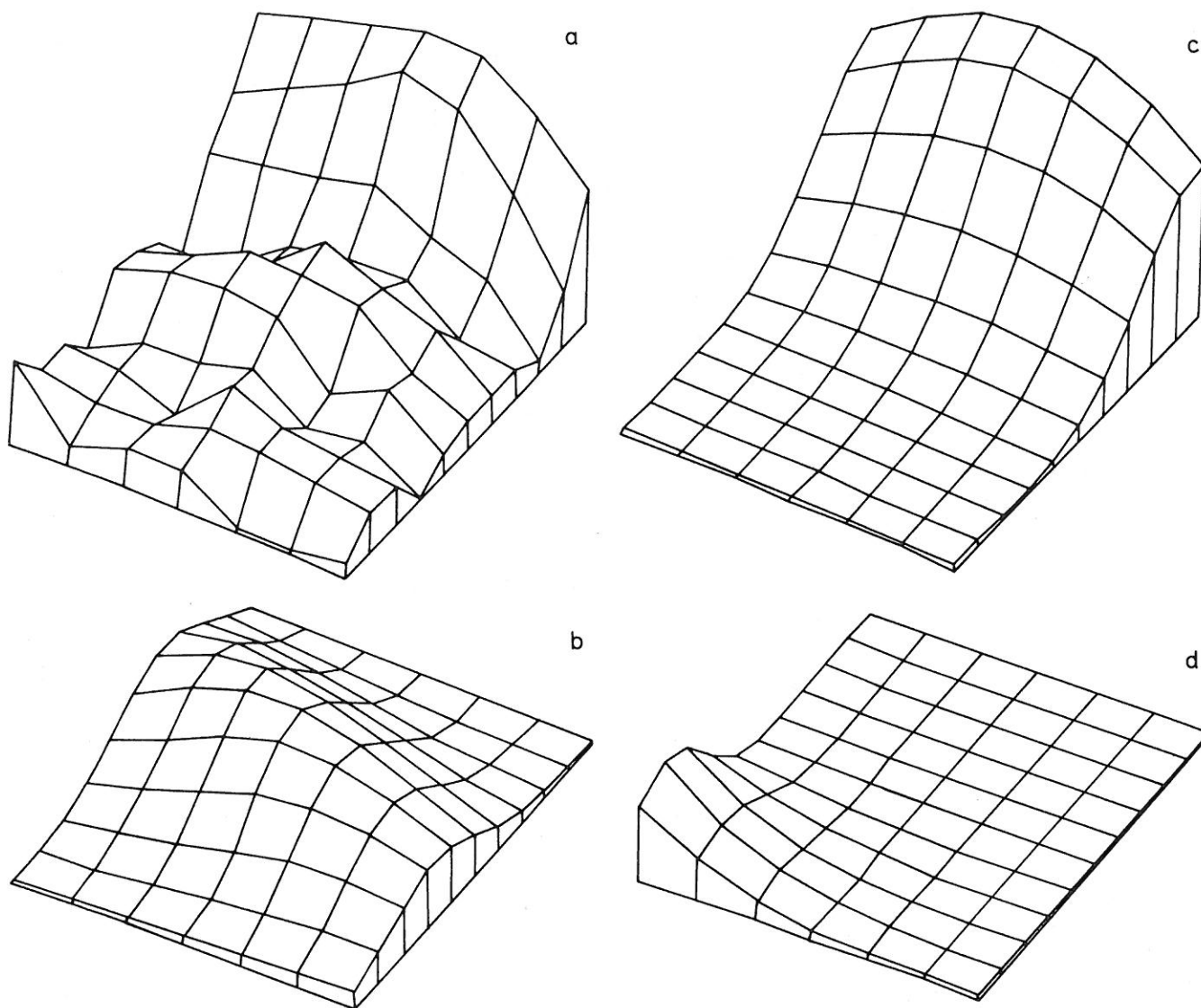
Figure 1. Gaussian modeling of the gray-tone surface. Frame (a) is a small region of a digitized gel image from which background staining has been removed by the methods described in the text. Three protein spots contribute to the gray-tone surface in this region: a small isolated spot in the center, a large spot whose peak is just visible at the top-center, and the edge of a third spot in the lower left corner. In frames (b), (c) and (d) the Gaussian models of these three spots, computed as described in the text, are plotted within the window defined by frame (a). The Gaussian model gives a reasonable description of the silver-stain intensity in these spots.

the green beam, then yellow will appear on the screen wherever spots on the two gels overlap (Fig. 2). Usually the yellow spot has a red "ear" on one side and a green "ear" on the other side because the gels cannot be perfectly aligned. The color image can be quickly and comfortably scanned by eye for major discrepancies such as a green spot without an accompanying red spot, or large differences in size or intensity of corresponding spots. Next, a cursor is positioned on the screen over a spot of interest, and with this information the computer can identify the spot and retrieve its properties. Because we have taken this visual (operator-assisted) approach to identifying corresponding spots on pairs of gels, we have not needed an especially close alignment of gel images. Thus, we could be content with rubber-sheeting gel images by a single second-order polynomial transformation. For automatic (solely digital) matching of spots, the superposition of gel images provided by a single transformation is probably not good enough. In this case, the more complicated "triangulation" method described by Vo *et al.* [4] for superimposing two images would be preferred.

3.10 Advantages of GIPSY

Our approach to digital analysis of 2-D gels has certain advantages over previously puslished systems: (i) GIPSY is designed for implementation on any mid-sized computer while still appearing the same to the user, (ii) GIPSY is designed to handle a wide variety of image processing problems so there may be economies of scale in implementing GIPSY as compared to systems specialized for 2-D gel analysis, and (iii) GIPSY programming is done in a modular fashion, allowing the researcher great flexibility in tailoring computations to the special needs of his or her gel analysis problem. Further information about GIPSY is available from the Spatial Data Analysis Laboratory at Virginia Tech.
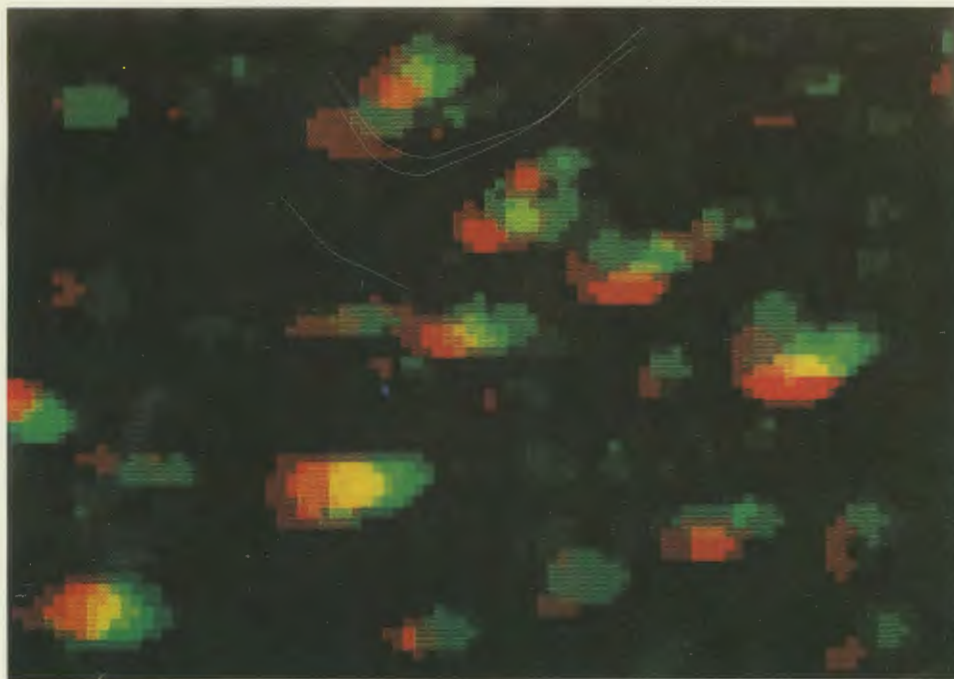
Figure 2. Color comparison of two gel images. The digital image of a small section of a metaphase gel has been displayed on a color TV monitor in green and the digital image of the corresponding section of an S-phase gel has been superimposed in red. (Before display the image of the S-phase gel was transformed so that it would overlap the image of the metaphase gel as completely as practicable with a single second-order polynomial transformation.) Where the two images overlap, various yellow tones result. Since the intensity of the red and green colors reflects the magnitude of the gray-tone on the digital image, the color image gives visual information about quantitative as well as qualitative comparisons. Spot A (see [1]) is in the upper-left corner.

4 Concluding remarks

The image-processing methods described in Section 3 have been used to determine the relative amounts of low-to-moderate abundance proteins on silver-stained 2-D gel electrophoretograms of total acidic plasmodial proteins at various stages in the nuclear division cycle of *Physarum polycephalum*. The results of this study are described in detail elsewhere [2]. Suffice it to say here that we found five polypeptide chains that showed reproducible, significant changes in relative abundance during the cell cycle. One of these spots was noticed on a preliminary visual inspection of the gels [1], another was noticed because of our particular interest in tubulin proteins, and the other three were uncovered by digital image processing of the gels as described herein.

Because of the difficulties inherent in quantifying the nonstoichiometric silver stain, we cannot say that the other several hundred minor proteins on our gels accumulate continuously. There may be other cell cycle dependent proteins that have escaped attention by our methods. However, our observations are consistent with results of similar studies in other organisms [11–16] and with preliminary studies in *Physarum* [17, 18], which have revealed that of the cellular proteins that show up on 2-D gels most are synthesized continuously throughout the cell cycle, but a few (less than 1 %) show significant changes in rate of synthesis. Because all previous computer-assisted surveys of 2-D gels during the cell cycle were done with cellular organisms requiring some sort of selection process to obtain cells at a specific stage of the cycle, it was important to carry out such a survey on an acellular organism with a naturally synchronous cell cycle in order to be sure that synchronization artifacts have not compromised earlier studies.

The results reported by us [1, 2] and others [11–18] demonstrate that, although most proteins are synthesized and accumulate continuously throughout the cell cycle, there are some proteins whose periodic changes in relative abundance

provide further markers for progress through the cell cycle. Furthermore, such changes can be detected at the level of individual polypeptide species by the technique of two-dimensional gel electrophoresis. Since any protein involved in the control of progress through the cell cycle must change in some fashion during the cycle, the techniques described here and elsewhere provide a method of searching for and identifying possible regulatory proteins. The search has just begun.

5 References

[1]. Pahlic, M. and Tyson, J. J., *Biochem. J.* 1983, *212*, 245–247.

[2] Pahlic, M. and Tyson, J. J., *Exp. Cell. Res.* 1985, *161*, 533–540.

[3] Bossinger, J., Miller, M. J., Vo, K. P., Geiduschek, E. P. and Xuong, N. H., *J. Biol. Chem.* 1979, *254*, 7986–7998.

[4] Vo, K. P., Miller, M. J., Geiduschek, E. P., Nielsen, C., Olson, A. and Xuong, N. H., *Anal. Biochem.* 1981, *112*, 258–271.

[5] Taylor, J., Anderson, N. L., Coulter, B. P., Scandora, A. E. and Anderson, N. G., in: Radola, B. J. (Ed.), *Electrophoresis '79*, Walter de Gruyter, Berlin and New York 1980, pp. 329–339.

[6] Lemkin, P. F. and Lipkin, L. E., *Comp. Biomed. Res.* 1981, *14*, 272–297.

[7] O'Farrell, P. H., *J. Biol. Chem.* 1975, *250*, 4007–4021.

[8]. Merril, C. R., Switzer, R. C., and Van Keuren, M. L., *Proc. Natl. Acad. Sci. USA* 1979, *76*, 4335–4339.

[9] Rosenfeld, A. and Pfaltz, J. L., *Pattern Recognition* 1968, *1*, 33.

[10] Lutin, W. A., Kyle, C. F. and Freeman, F. A., in: Catsimpoolas, N. (Ed.), *Electrophoresis '78*, Elsevier-North Holland, Amsterdam 1979, pp. 93–106.

[11] Elliott, S. G. and McLaughlin, C. S., *Proc. Natl. Acad. Sci. USA* 1978, *75*, 4384–4388.

[12] Lutkenhaus, J. F., Moore, B. A., Masters, M. and Donachie, W. D., *J. Bacteriol.* 1979, *138*, 352–360.

[13] Milhausen, M. and Agabian, A., *J. Bacteriol.* 1981, *148*, 163–173.

[14] John, P. C. L., Lambe, C. A., McGookin, R., Orr, B. and Rollins, M. J., *J. Cell Sci.* 1982, *55*, 51–67.

[15] Bravo, R. and Celis, J. E., *J. Cell Biol.* 1980, *84*, 795–802.

[16] Lorincz, A. T., Miller, M. J., Xuong, N. H. and Geiduschek, E. P., *Mol. Cell Biol.* 1982, *2*, 1532–1549.

[17] Laffler, T. G., Chang, M. T. and Dove, W. F., *Proc. Natl. Acad. Sci. USA* 1981, *78*, 5000–5004.

[18] Turnock, G., Chambers, J. and Birch, B., *Eur. J. Biochem.* 1981, *120*, 529–534.