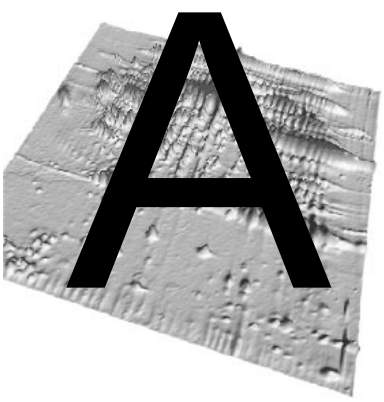
PART VI

APPENDIXES



ALGORITHMS

This chapter presents the outline of algorithms implemented in the Melanie II software. For a detailed description, see [reference 7].

Detecting and quantifying features

The technique used in Melanie II for feature detection and quantification combines spot detection by a non-parametric method (NP), with spot quantification by a parametric method (P). This combined technique achieves detection and quantification in two successive operations as described in the following sections.

Spot detection

Spots are detected using a NP method based on the Laplacian and second derivatives:

Given a 2-DE image $I(x, y)$, a saturation threshold T, a spot S_i, and a point $\vec{p} = (x, y)$. To decide whether point $\vec{p}$ is part of a spot, two decision rules have been defined, depending on the intensity $I(\vec{p})$ at point $\vec{p}$:

1 $I(\vec{p}) \leq T$; this corresponds to a non-saturated value:

$$\vec{p} \in S_i \Leftrightarrow \begin{cases} -\Delta I(\vec{p}) - L < 0 \\ MIN\left(\frac{\partial^2}{\partial x} I(\vec{p}) - R, \frac{\partial^2}{\partial y} I(\vec{p}) - C\right) > 0 \quad \text{when} \quad -\Delta I(\vec{p}) - L \geq 0 \end{cases} \qquad \text{(Eq. A.1)}$$

2 $I(\vec{p}) > T$; this is a saturated value:

$$\vec{p} \in S_i \Leftrightarrow MIN\left(\frac{\partial^2}{\partial x} I(\vec{p}), \frac{\partial^2}{\partial y} I(\vec{p})\right) > 0 \,, \qquad \text{(Eq. A.2)}$$

where L, R, C are three small positive constants. L represents the threshold for the Laplacian of image I, R is the threshold for the second derivative along axis x, and C is the threshold for the second derivative along axis y. 2-DE images may be saturated due to the staining technique and/or image acquisition. Figure 0-1 illustrates a 2-DE spot profile, where the ideal virtual shape is truncated due to the saturation effect.

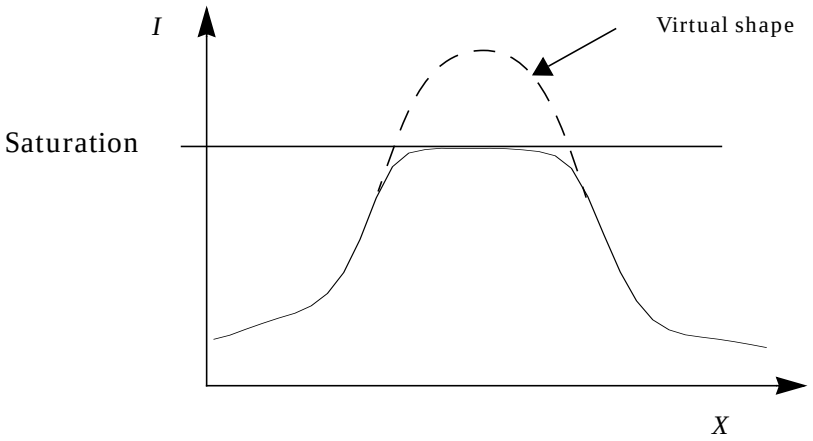


Figure 0-1. Saturation of a 2-DE spot profile due to the staining technique or image acquisition device.

Threshold T is the value above which a pixel is considered to be saturated. The threshold T and Laplacian ΔI in (Eq. A.1) and (Eq. A.2) are defined as follows:

$$\Delta I(\vec{p}) = -\left(\frac{\partial^2}{\partial x} I(\vec{p}) + \frac{\partial^2}{\partial xy} I(\vec{p})\right) \qquad \text{(Eq. A.3)}$$

$$T = max(I) - \frac{saturation}{100} \times (max(I) - min(I)) \qquad \text{(Eq. A.4)}$$

where $saturation$ is a positive value between 0 and 100. $saturation = 100$ means that no pixel is saturated in the whole image.

Spot quantification

The second step in the combined technique may be achieved using one of two alternative methods: direct quantification (described in this section) and quantification by Gaussian modeling (described in *Gaussian fitting on page A-13*). In the first method, the area, volume, percent volume, optical density (OD) and

percent OD of each spot are computed by considering the pixels inside their detected shape. These values are calculated as follows:

1 Spot area (AREA):

$$\text{AREA} = \text{number of pixels} \times \text{pixel area}$$

2 Spot optical density (OD):

$$\text{OD} = \underset{x, y \in spot}{\text{MAX}} (I(x, y))$$

3 Spot percent optical density (%OD)

$$\%\text{OD} = \frac{\text{OD}}{\sum_{s=1}^{n} \text{OD}_s} \times 100 ,$$

where OD_s is the optical density of spot s in a gel containing n spots.

4 Spot volume (VOL):

$$\text{VOL} = \sum_{x, y \in spot} I(x, y)$$

5 Spot percent volume (%VOL):

$$\%\text{VOL} = \frac{\text{VOL}}{\sum_{s=1}^{n} \text{VOL}_s} \times 100 ,$$

where VOL_s is the volume of spot s in a gel containing n spots.

Aligning

Image alignment is performed by polynomial image warping. The warping is performed in four stages. First, control points or landmarks are selected on one image (the reference image). Second, corresponding control points are given in the second image (the data image which is to be aligned to the reference image). Third, the warping transformation is determined by solving equation (Eq. A.6). Given a set of control points, we solve this equation using Least Squares minimization. In the fourth stage, warping is performed on the data image using the following transformation:

$$(x, y) \rightarrow (u(x, y), v(x, y)) , \qquad \text{(Eq. A.5)}$$

where (x, y) are the pixel coordinates in the original data image, (u, v) are the pixel coordinates in the resulting warped data image aligned to the reference image. $u(x, y)$ and $v(x, y)$ are two polynomial functions (one for each axis) determined by the Least Squares method in order to minimize the distance

between the transformed control points in the data image and those in the reference image.

To perform this, the following two systems of linear equations (u and v are linear with respect to a_i and b_i) have to be solved:

$$u = \sum_{i=0}^{n}\sum_{j=0}^{i} a_{ij}x^i y^{j-i} \qquad v = \sum_{i=0}^{n}\sum_{j=0}^{i} b_{ij}x^i y^{j-i} \quad , \qquad \text{(Eq. A.6)}$$

where (x, y) are the landmark positions in the reference image, (u, v) are the positions of the corresponding landmarks in the data image, and a_{ij} and b_{ij} are the polynomial coefficients to be determined. The polynomial order n is specific to the problem. For M control points and polynomial functions of order n, the coefficients a_{ij} and b_{ij} are determined by two systems of M linear equations with T variables, where $T = (n+2)(n+1)/2$. The Least Squares method may be used to solve these equations only if $M \geq T$.

The image warping is computed by mapping the original image using the functions of (Eq. A.6). The inverse of this transformation is frequently required when performing comparisons. In the general form, the inverse of the function is difficult to compute. In the Melanie II system, only the first-order, second-order, third-order polynomials and one variable polynomials have been used. This means that warping along the horizontal axis depends only on the landmarks' horizontal positions $u(x)$, while $v(y)$ depends only on the vertical ones. This simplifies the mapping of images, in which horizontal and vertical lines are mapped into horizontal and vertical lines respectively. Using this transformation, lines are mapped into curbs. With these simplifications, (Eq. A.6) becomes:

- First-order:

$$u = a_0 + a_1 x \qquad v = b_0 + b_1 y \quad , \qquad \text{(Eq. A.7)}$$

- Second-order:

$$u = a_0 + a_1 x + a_2 x^2 \qquad v = b_0 + b_1 y + b_2 y^2 \quad , \qquad \text{(Eq. A.8)}$$

- Third-order:

$$u = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \qquad v = b_0 + b_1 y + b_2 y^2 + b_3 y^3 \quad . \qquad \text{(Eq. A.9)}$$

where (u, v) are the landmark positions in the image that is going to be aligned, (x, y) are the coordinates in the reference image, and a_i, b_i are the polynomial warping coefficients to be computed by minimizing the error between the positions (u, v) and (x, y). Given M landmarks in each gel, this can be expressed as follows:

$$error_x = \sum_{i=0}^{M} (u_i - (a_0 + a_1 x_i + a_2 x_i^2 + a_3 x_i^3))^2 \rightarrow minimum$$

$$error_y = \sum_{i=0}^{M} (v_i - (b_0 + b_1 y_i + b_2 y_i^2 + b_3 y_i^3))^2 \rightarrow minimum \quad , \qquad \text{(Eq. A.10)}$$

where (u_i, v_i) are the landmark positions in the first gel, and (x_i, y_j) are the corresponding landmark locations in the second gel.

The inverse of the above three expressions can easily be computed by algebraic methods. Figure 0-2 illustrates the approximation of the horizontal position of landmarks in two different gels, using: (a) first-order, (b) second-order, (c) third-order polynomials; (d) the inverse third-order polynomial warping which in this case cannot align the two gels in a better way.

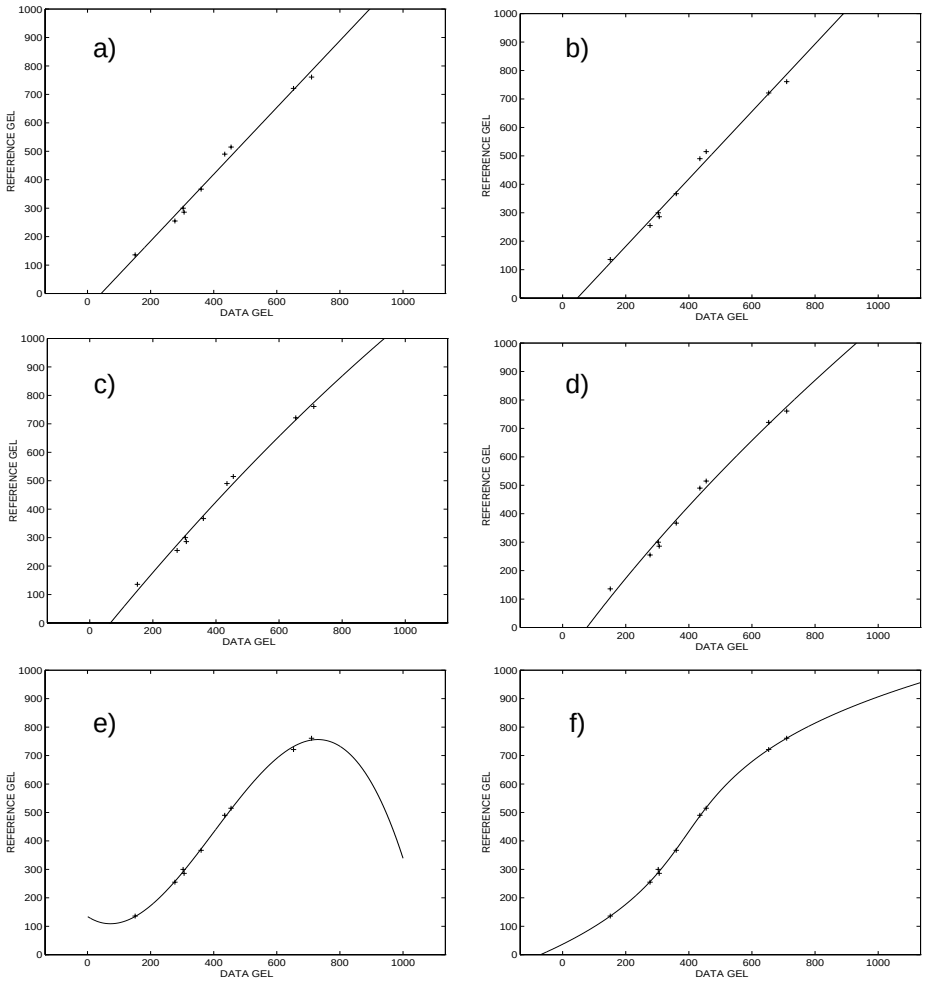


Figure 0-2. One variable polynomial warping functions of: (a) first-order, (b) inverse first-order, (c) second-order, (d) inverse second-order, (e) third-order, (f) inverse third-order polynomial. Inverse polynomial mapping in this example aligns the gels best, and is monotone.

The minimum number of landmarks to compute a_i, b_i depends on the polynomial order. A polynomial order of n requires at least $n+1$ landmarks. For example, a first-order transformation requires at least two landmarks, whereas a third-order polynomial requires at least four landmarks. Figure 0-3 illustrates one variable polynomial transformation of a grid representing a 2-DE image. This was performed using four landmarks and two polynomial functions of one variable.

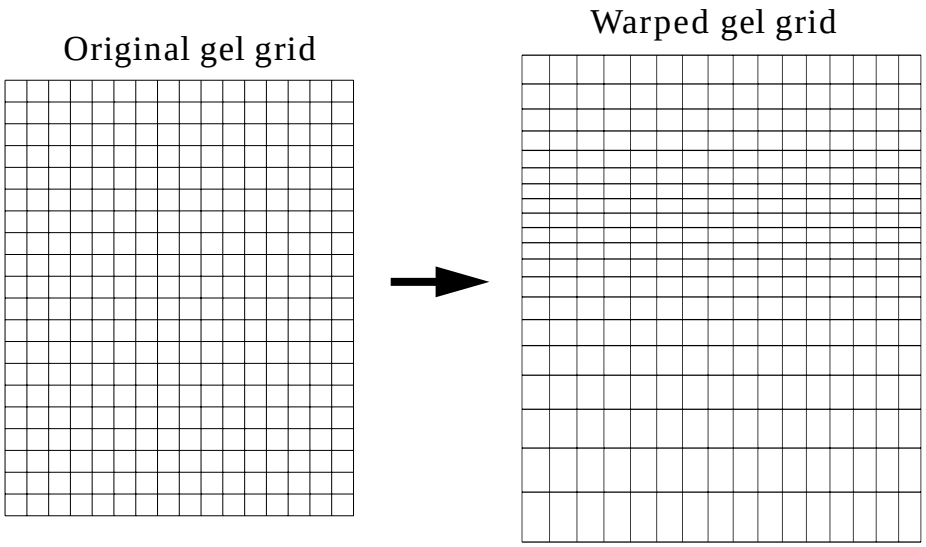


Figure 0-3. : Transformation of a gel grid using two polynomial warping functions of one variable. A second order polynomial warping along X and third order polynomial warping along Y were used, as well as four landmarks.

A set of 2-DE images are aligned in the following steps:

1 A set of landmarks are selected on each image. In Figure 0-4, landmarks are designated by labels $p1$ to $p9$.

2 The polynomial orders for the horizontal and vertical axes are chosen, and one gel is selected as reference gel from the set of images. Then, the rest of the gels is aligned to the reference gel. In Figure 0-4 the two top images illustrate

two gels that are not superimposable, while at the bottom, both gels are shown after alignment with a third and second order polynomial warping transformation along the horizontal and vertical axis, respectively.

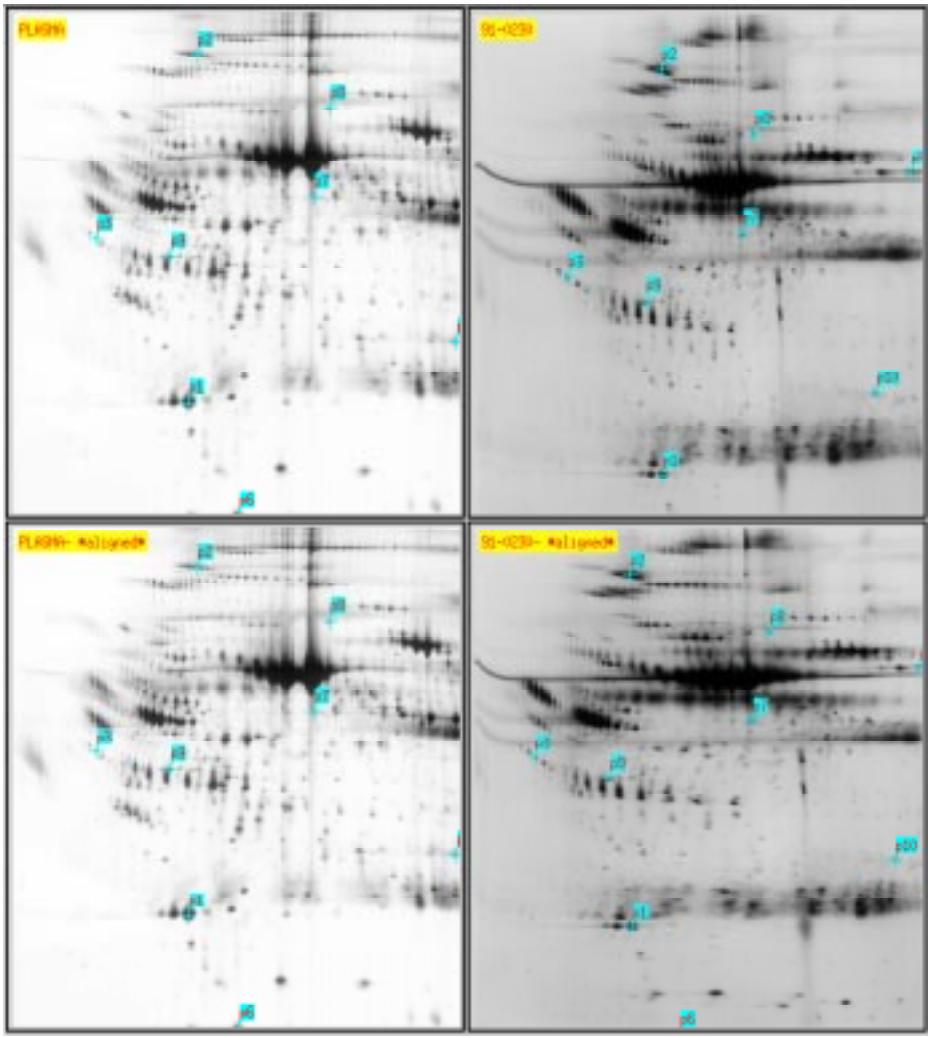


Figure 0-4. : (top left) 2-DE reference image, (top right) image to warp, (bottom) both images after second order polynomial warping along Y, and third order warping along X. Nine landmarks and bilinear interpolation have been used. Visual correspondences are evident in the aligned images.

Matching

Matching algorithm

A powerful matching algorithm has been implemented in Melanie II. It is an improved version of the algorithm described in [reference 5]. Matching between two images is accomplished according to the following basic steps:

1 For each spot in a gel, a list of neighboring spots (cluster) is selected and associated to that spot (Fig. 0-5). The central spot of a cluster is called the primary spot, and its surrounding spots are the secondary spots of the cluster. A given spot is considered to be part of a cluster if its centroid is inside a circle with a fixed radius. This radius is estimated by considering the image dimensions, the number of spots in the image, and a given minimum number of spots for a cluster. Clusters are represented in polar coordinates, and are built before matching, in order to be quickly compared.

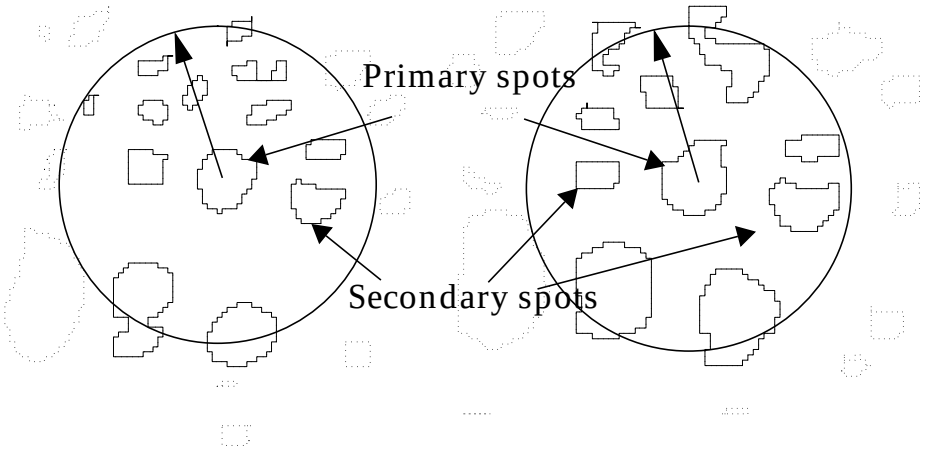


Figure 0-5. Clusters are formed by considering the neighboring spots for each spot in both images. Secondary spots in each cluster are represented in polar coordinates relative to the primary spot.

2 The method starts to match clusters corresponding to the input pairs, if some exist. Otherwise, clusters whose primary spots have the highest intensity value are matched first.

3 Clusters are compared using a similarity measure based on a probabilistic method. This method is similar to calculating the probability K_m that if N darts randomly land on a board with an area A, at least m darts will land on the target whose area is T, with $m \leq N$. This can be calculated by the expression:

$$K_m = \sum_{h=m}^{N} \binom{N}{h} p^h (1-p)^{N-h}, \text{ with } \binom{N}{h} = \frac{N!}{h!(N-h)!}, \qquad \text{(Eq. A.11)}$$

where $p = T/A$, which represents the probability that a dart will land on the target in one single try. In a similar way, the probability for the next random hit within a cluster where $m-1$ spots have already been matched is given by the following equation:

$$p_m = \frac{A_s - A_{m-1}}{A_c - A_{m-1}}, \qquad \text{(Eq. A.12)}$$

where $m \geq 1$, A_s is the sum of the areas of the secondary spots in the cluster (highlighted spots in Figure 0-5), A_c is the total area bounded by the cluster (circular area) and A_{m-1} is the total area of the matched spots in the cluster. Then, the probability that at least m spots were matched in the two clusters by a random process in N trials, where N is the number of spots in one of the clusters, can be estimated by the following expression:

$$K_m \approx \sum_{h=m}^{N} \binom{N}{h} p_G{}^h (1-p_G)^{N-h}, \qquad \text{(Eq. A.13)}$$

where

$$p_G = \left| \prod_{i=1}^{m} p_i \right|^{\frac{1}{m}} \text{ is the geometric mean.} \qquad \text{(Eq. A.14)}$$

(Eq. A.13) is a good approximation of the real probability of randomly finding m or more matches in N tries:

$$K_m = \sum_{h=m}^{N} \binom{N}{h} \prod_{i=1}^{h} p_i \prod_{i=h+1}^{N} (1-p_i), \qquad \text{(Eq. A.15)}$$

where p_i is defined by (Eq. A.12).

4 In given situations, the comparison of primary clusters may not be sufficient and mismatching may occur. In this case, secondary clusters are compared. Thus, more spots are considered, based on the idea that if matches in the primary cluster are correct, then the clusters surrounding them can also be matched.

5 A consistency check is performed in order to detect possible mismatching. Most of them occur among the set of most probable matches. Assuming that the great majority of matches is correct, the parameters of the following rigid transformation are calculated by considering the location of the matched spots and the Least Squares minimization:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} t_y \\ t_y \end{bmatrix} + \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \qquad \text{(Eq. A.16)}$$

For a simple rotation and translation, this equation becomes:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} + \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} . \quad \text{(Eq. A.17)}$$

The test consists on checking whether or not

$$AxD - BxC \approx 1.0 , \quad \text{(Eq. A.18)}$$

which corresponds to a rotation of (Eq. A.17). For each primary cluster, parameters A,B,C and D are computed by considering each set of three matched spots in the cluster. If the evaluation of (Eq. A.18) is 1.0 ± 0.25 and $\theta = \pm 10$ degrees, then the three spots are considered to be *good*.

6 A transformation tries to match the remaining spots. Using the matched *good* spots in two clusters and the Least Squares method, parameters for (Eq. A.16) are computed. The locations of unmatched spots in one gel are mapped using the resulting transformation. If the transformed location of a spot in the first gel lies within the boundaries of a spot in the second gel, given a proximity criterion, then both spots are matched. If unmatched spots cannot be matched, the criteria is relaxed. The unmatched spots are successively examined until no new matches are found. Multiple matches can also be accepted if the users wishes to do so.

Good / bad pairs

While the above described matching algorithm has been optimized for 2-DE gel images, mismatches may sometimes occur. Good matches may be discriminated from bad ones by the use of a quality measure. This measure is computed for each pair according to the following steps:

1 a neighboring region is considered around each pair,

2 corresponding regions are superimposed in order to minimize the vector lengths in the region, using Least Squares,

3 each pair of spots in the selected region is given a score representing the number of times it has been larger than a given threshold, otherwise it is marked *bad,* which reduces the score.

4 a pair is considered to be *good* if its final score is larger than a second threshold, otherwise it is considered to be *bad*. The threshold represents the pair quality.

Good and bad pairs may be selected for further operations. The quality pair evaluation keeps track of possible image alignment.

Creating synthetic gels

Synthetic gels may be created by merging a set of gel images that contain at least three pairwise matched gels. A synthetic gel is created as follows:

1 A reference gel is selected from the set of gels. This gel plays a special role in the creation of the synthetic gel, since the spot positions in the reference gel serve as representative positions for the synthetic gel.

2 Each spot s_R in the reference gel is tested to verify that it is *well matched*, i.e. whether two other spots $s_1 \in G_1$ and $s_2 \in G_2$ exist, where G_1 and G_2 are two other gels from the given set of gels, such that $s \leftrightarrow s_1$ and $s \leftrightarrow s_2$ ($a \leftrightarrow b$ means that spot a is matched with spot b). This operation forms triangles of matched spots that constitute the starting groups (Fig. 0-6).

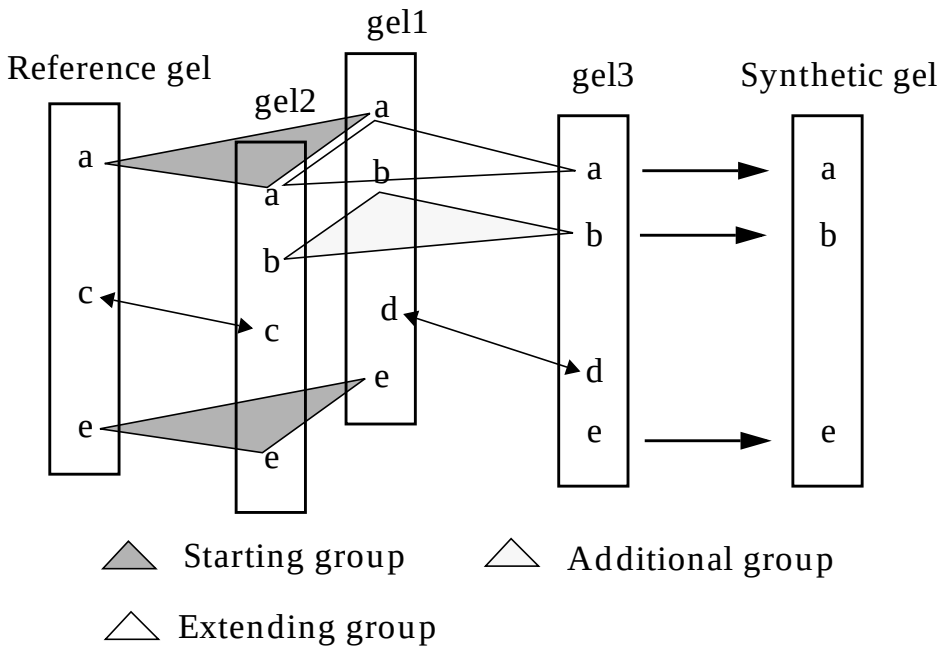


Figure 0-6. Groups of spots are formed by first looking for triangle pairs. Then, after extending them, the synthetic gel is created. Each spot in the synthetic gel represents a formed group.

3 Starting groups are extended by adding supplementary spots, according to a connectivity test. This test verifies whether a spot is matched with at least one other spot in the initial group.

4 After examining all spots in the reference gel, additional groups are formed, starting with spots in the second gel that are not yet part of a group. Then, the same procedure repeats to form additional groups with the remaining gels.

5 A synthetic gel image is created that contains the same number of spots as the number of determined groups. Each spot in the synthetic gel represents a group of *well matched* spots. These representative spots are selected by considering positions, optical densities, and shapes of spots in a given group, according to the following procedure:

- Position: if the group has one spot in the reference gel, the position of this spot is taken. Otherwise, a set of neighbors of the first spot is considered. These neighbors should be included in the group, and have pairs on the reference gel. Then the spot closest to the first included spot is selected, and the translation vector between both

spots is computed. Finally, the position of the spot representing the group on the reference gel is computed by adding this translation to the corresponding closest spot on the reference gel.

- Intensity: this value is equal to the average optical density value among all spots in the group.
- Shape: the spot shape is copied from the spot in the group that has the area closest to the average area in the group.

6 Finally, the dimensions of the synthetic gel are recomputed, since all the spot coordinates have to be positive.

Gaussian fitting

This section describes the elements of 2D-Gaussians. In the next section, the Gaussian fitting algorithm for modeling spots is presented.

Gaussian model

The general form of the two-dimensional Gaussian equation is:

$$g(x, y) = e^{ax^2 + bxy + cy^2 + dx + ey + f} + h, \text{ with } b^2 - 4ac < 0, \qquad \text{(Eq. A.19)}$$

or in its simple form:

$$g(X, Y) = Ae^{\left(\frac{X - X_c}{\sigma_X}\right)^2 + \left(\frac{Y - Y_c}{\sigma_Y}\right)^2} + B, \qquad \text{(Eq. A.20)}$$

where (X_c, Y_c) is the Gaussian's center, A its amplitude, B its asymptotic horizontal plane, (σ_X, σ_Y) the diffusion coefficients along the principal axes.

Figure 0-7 shows a graphical representation of (Eq. A.19). Each spot in a gel may be represented by one Gaussian function at position (x_c, y_c), with ampli-

tude A, background B, the spread (σ_X, σ_Y) along the principal axes, and an orientation angle θ of its principal components.

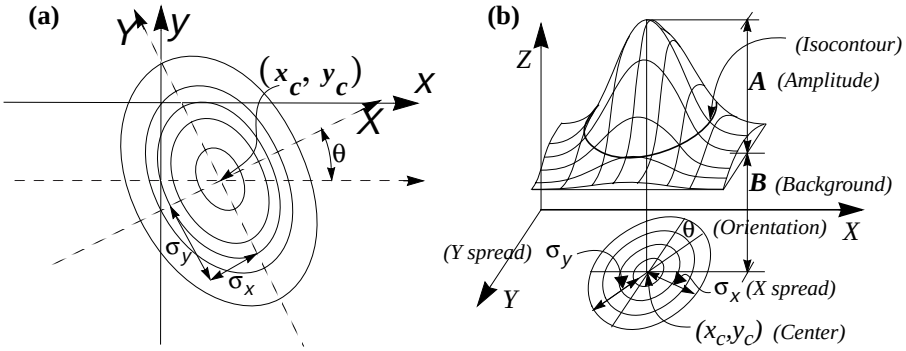


Figure 0-7. (a) 2D-Gaussian contour levels with center (x_c, y_c) and rotation θ (X-Y axes). (b) 2D-Gaussian parameters representing a 2-DE spot gel.

Gaussian fitting

The general fitting problem may be described as follows:

Given a function F, a family of functions Φ, a discrete function I to be fitted, and a domain Ω, defined as:

$$F(\vec{a}) : R^m \rightarrow \Phi$$
$$\phi(\vec{p}) : \Omega \rightarrow R$$
$$I(\vec{p}) : \Omega \rightarrow R \text{ , for } \forall \phi \in \Phi \text{ and } \Omega \subset R^n, \qquad \text{(Eq. A.21)}$$

find $\vec{a}_\infty \in R^m$, such that:

if $\phi_\infty = F(\vec{a}_\infty)$,

then: $\|\phi_\infty - I\| = \text{Min}_{\vec{a} \in R^m} \|F(\vec{a}) - I\| = \|F(\vec{a}_\infty) - I\|$

when using a distance norm,

$$\text{or: } \|\phi_\infty - I\| = \text{Min}_{\vec{a} \in R^m} \sqrt{\sum_{\vec{p} \in \Omega} (\phi_{\vec{a}}(\vec{p}) - I(\vec{p}))^2} \qquad \text{(Eq. A.22)}$$

when using the second order distance norm.

The function ϕ_∞ is the best approximation (best fit) of a discrete function I for a given distance norm.

Considering several vectors $\vec{a} \in R^7$ where $\vec{a} = (a, b, c, d, e, f, h)$ and the expression in (Eq. A.19), it is possible to define a set of Gaussian functions that form a 2D-Gaussian family of functions. The fitting problem using the second order norm and the 2D-Gaussian family of functions is defined as:

Find $\acute{a}_\infty \in R'$, such that:

$$\left\| G(\vec{a}_\infty) - I \right\| = \mathrm{Min}_{\vec{a} \in R^m} \sqrt{\sum_{\vec{p} \in \Omega} \left(G_{\vec{a}}(\vec{p}) - I(\vec{p}) \right)^2} = \sqrt{\sum_{\vec{p} \in \Omega} \left(G_{\vec{a}_\infty}(\vec{p}) - I(\vec{p}) \right)^2}, \quad \text{(Eq. A.23)}$$

where $G(\vec{a}) = e^{a_1 x^2 + a_2 xy + a_3 y^2 + a_4 x + a_5 y + a_6} + a_7$.

Figure 0-8 shows the elements of the fitting process.

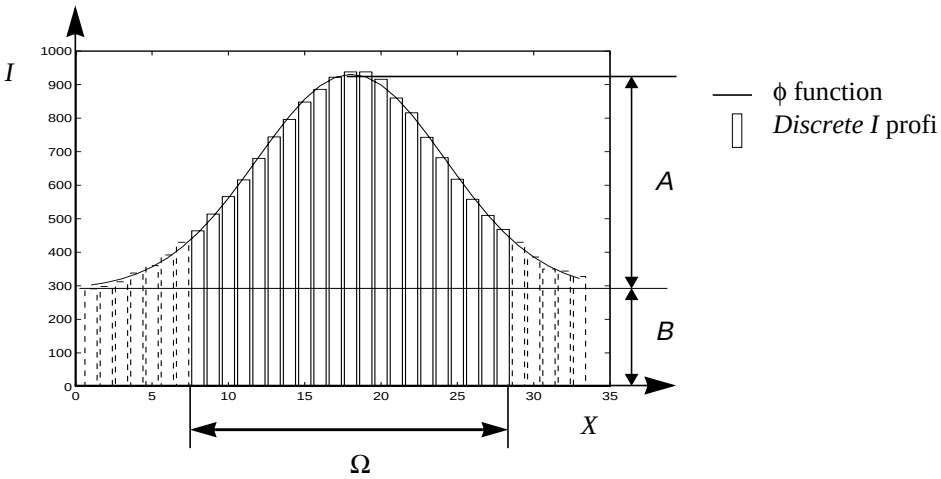


Figure 0-8. Function ϕ , discrete function I , fitting domain Ω , background B , and distance between ϕ and I .

Several minimization methods may be used to find the minimum of (Eq. A.23). Most of them need a first approximation to start the iterative search of the real minimum. This means that the efficiency of these methods strongly depends on the first approximation. An efficient method to calculate a good starting point is to approximate (Eq. A.23):

Considering a_7 to be a constant, and $(I(x, y) - a_7) > 0$ for $\forall (x, y) \in \Omega$, then (Eq. A.23) may be approximated as follows:

$$\mathrm{Min}_{\vec{a} \in R^m} \sqrt{\sum_{(x, y) \in \Omega} (a_1 x^2 + a_2 xy + a_3 y^2 + a_4 x + a_5 y + a_6 - \log(I(x, y) - a_7))^2} .$$

which can be solved by the following system of linear equations:

$$Ax = b \quad \text{, or} \quad \begin{bmatrix} \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_i^2 & x_i y_j & y_j^2 & x_i & y_j & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \times \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = \begin{bmatrix} \cdots \\ \cdots \\ \log(I(x_i, y_j) - a_7) \\ \cdots \\ \cdots \end{bmatrix} \quad \text{. (Eq. A.24)}$$

The dimension of matrix A is $[cardinal(\Omega) \times 6]$. (Eq. A.24) can be solved by the Least Squares method:

$$A^t A x = A^t b, \qquad \text{(Eq. A.25)}$$

where $A^t A$ is a $[6 \times 6]$ symmetrical matrix.

The solution of (Eq. A.25) is illustrated by Figure 0-9 for a 2-DE spot profile. This is an approximation of the exact solution of (Eq. A.23).

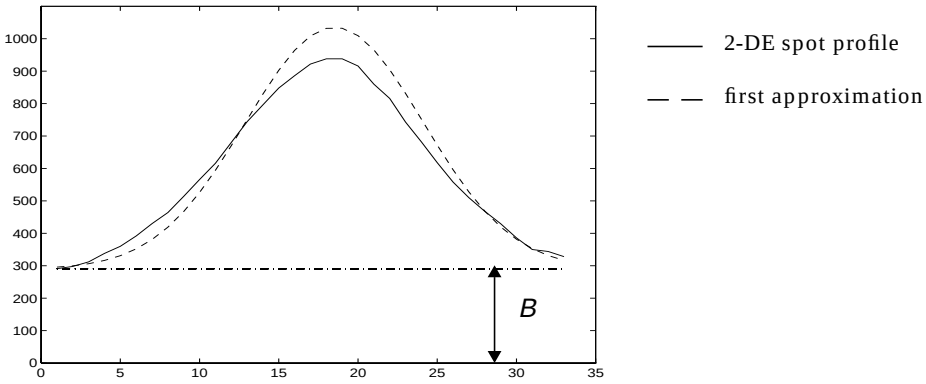


Figure 0-9. First approximation of 2-DE spot profile by fitting the $\log(I(x) - B)$ where $B = min(I) + 1$ with a second degree polynomial.

Starting from this first approximation, (Eq. A.23) can be solved using a non-linear multidimensional minimization algorithm. Non-linear methods may be classified in two groups: methods using the first derivative of the distance function, and methods not using it. Methods in the first group are called the conjugate gradient methods and the quasi-Newton or variable metric methods. The second ones include the downhill simplex method and the Powell method. The Polak-Ribiere's variant of the conjugate gradient method has been chosen as a minimization method that can be described as follows:

Given a function F and an initial point $\vec{p}_0$ (a starting point for the first iteration), move from point $\vec{p}_i$ to point $\vec{p}_{i+1}$, as often as needed, by minimizing F along the line from $\vec{p}_i$ in the direction of the gradient $-\nabla F(\vec{p}_i)$. Stop when $\|F(\vec{p}_{n+1}) - F(\vec{p}_n)\| < \varepsilon$. At this point in time, $\vec{p}_{i+1}$ is the point where F has a minimum value with tolerance ε. The convergence and the result of this algorithm depend on the choice of the initial point $\vec{p}_0$.

The fitting algorithm consists of two steps. First, assuming h is equal to a constant equal to $min(I)$, the initial point $\vec{p}_0$ is computed by using the linear approximation method described above. Second, the real minimum is calculated using the conjugate gradient algorithm, with:

$$F(\vec{p}) = \sum_{(x,y) \in \Omega} \left(e^{ax^2 + bxy + cy^2 + dx + ey + f} + h - I(x,y)\right)^2,$$

where $\vec{p} = (a, b, c, d, e, f, h)$,

and the gradient of $F(\vec{p})$:

$$\nabla F(\vec{p}) = \left(\frac{\partial}{\partial a}F(\vec{p}), \frac{\partial}{\partial b}F(\vec{p}), \ldots, \frac{\partial}{\partial h}F(\vec{p})\right) = 2\begin{bmatrix} \sum_{x,y \in \Omega} x^2 \cdot G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} xy \cdot G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} y^2 \cdot G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} x \cdot G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} y \cdot G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} G(x,y)(G(x,y)+h-I(x,y)) \\ \sum_{x,y \in \Omega} (G(x,y)+h-I(x,y)) \end{bmatrix}, \quad \text{(Eq. A.26)}$$

where:

$$G(x,y) = e^{ax^2+bxy+cy^2+dx+ey+f} + h.$$

Figure 0-10 illustrates the final solution of the conjugate gradient minimization algorithm, using the first linear approximation shown in Figure 0-9. The combination of both methods gives an optimal fit.

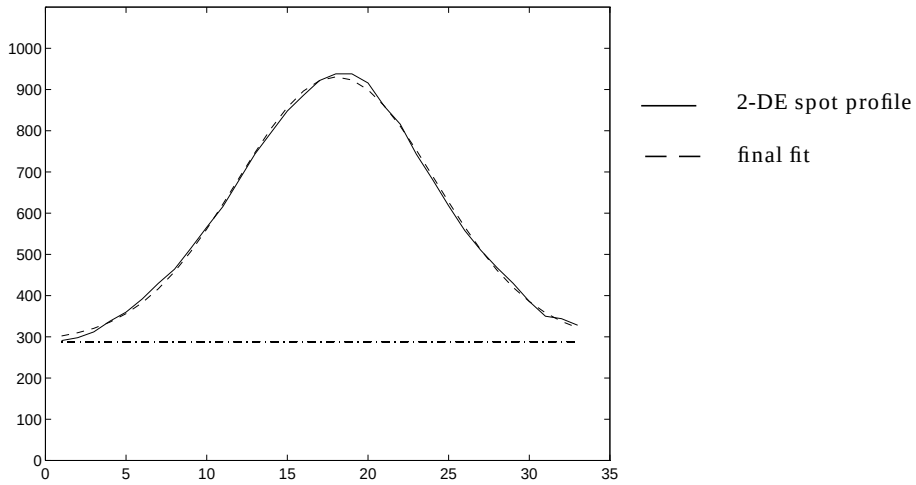


Figure 0-10. Final solution of the conjugate gradient minimization algorithm that best fits a 2-DE spot profile.

Filtering gel images

Smoothing

Melanie II implements four smoothing filters to reduce high frequency noise in images: Gaussian, diffusion, polynomial and adaptive smoothing. The last two methods present the advantage of not affecting image contrast.

Gaussian smoothing

Gaussian smooth of a 2-DE image is performed by convolving it with the following 3x3 mask operator:

$$mask = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad \text{(Eq. A.27)}$$

Diffusion smoothing

Isotropic diffusion is a recursive low-pass filter that performs smoothing in two successive passes, using two different recursive operators. For one-dimensional signals, the filter has the following form:

$$I^{(t+1)}(x) = \frac{1}{2}(I^{(t+1)}(x-1) + I^{(t)}(x+1)) \quad \text{(Eq. A.28)}$$

as a left to right operator, then

$$I^{(t+2)}(x) = \frac{1}{2}(I^{(t+1)}(x-1) + I^{(t+2)}(x+1)) \quad \text{(Eq. A.29)}$$

as a right to left operator, where $I^{(t)}(x)$ represents the intensity value of a one-dimensional signal at position x after t iterations.

Both operators establish an approximation to the solution of the isotropic thermic diffusion equation:

$$\frac{\partial u}{\partial t} = C\frac{\partial^2 u}{\partial x^2}, \quad \text{(Eq. A.30)}$$

or in its discrete form:

$$u_{t+\Delta t}(x) = 2u_t(x) - u_t(x-1) - u_t(x+1), \quad \text{(Eq. A.31)}$$

where $u_t(x)$ represents the temperature in x at time t.

The thermic diffusion equation for a two-dimensional signal is:

$$\frac{\partial u}{\partial t} = C\Delta u \text{, where } \Delta u = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} \quad \text{(Eq. A.32)}$$

and its discrete form is:

$$u_{t+\Delta t}(x,y) = 4u_t(x,y) - u_t(x-1,y) - u_t(x+1,y) - u_t(x,y-1) - u_t(x,y+1) \quad \text{(Eq. A.33)}$$

For a two-dimensional signal, this filter is composed of four operators which have the following forms:

1 Left to right:

$$I^{(t+1)}(x,y) = \frac{1}{2}(I^{(t+1)}(x-1,y) + I^{(t)}(x+1,y)), \quad \text{(Eq. A.34)}$$

2 Right to left:

$$I^{(t+2)}(x,y) = \frac{1}{2}(I^{(t+1)}(x-1,y) + I^{(t+2)}(x+1,y)), \quad \text{(Eq. A.35)}$$

3 Top to bottom:

$$I^{(t+3)}(x,y) = \frac{1}{2}(I^{(t+3)}(x,y-1) + I^{(t+2)}(x,y+1)), \quad \text{(Eq. A.36)}$$

4 Bottom to right:

$$I^{(t+2)}(x,y) = \frac{1}{2}(I^{(t+1)}(x,y-1) + I^{(t+2)}(x,y+1)). \quad \text{(Eq. A.37)}$$

The frequency response of the recursive filter is typical of a low pass filter. High frequencies are better filtered with the isotropic diffusion filter than with the Gaussian or mean filter (Fig. 0-11).

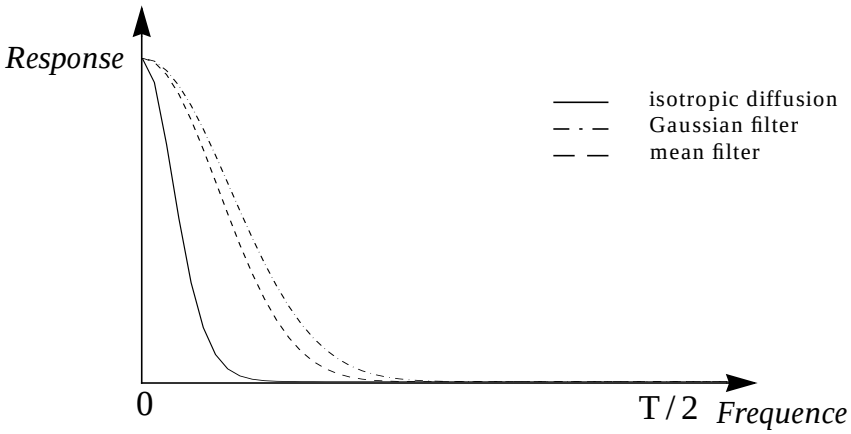


Figure 0-11. Frequency response of the isotropic diffusion filter compared with the Gaussian filter $[1/4,\ 2/4,\ 1/4]$ **and the mean filter** $[1/3,\ 1/3,\ 1/3]$ **. Each filter has been convolved ten times with Dirac.**

Polynomial smoothing

This smoothing technique is based on the approximation of pixel intensity values in a small area A (3x3, 5x5 or 7x7) by a second degree polynomial function. For a polynomial function of degree n, the general equation can be expressed as:

$$P(x, y) = \sum_{i,j}^{n} a_{ij} x^i y^j . \qquad \text{(Eq. A.38)}$$

The principle of the fitting process is to find coefficients a_{ij} of a polynomial function P that best fits the pixels in a small neighborhood A, where the central pixel is (x, y). Choosing $x = 0$ and $y = 0$ for equation (Eq. A.38), only coefficient a_{oo} remains, giving a linear equation that can easily be resolved. This value represents the new value of the central pixel of region A. Coefficient a_{oo} is calculated by minimizing the following expression with the Least Squares method, considering the intensity values $I(x, y)$ of a given image in a small area $A = \{(x, y) / -m \le x, y \le m\}$, centered in $(0, 0)$, as mentioned above:

$$\varepsilon = \sum_{x=-m}^{m} \sum_{y=-m}^{m} (P(x, y) - I(x, y))^2 \rightarrow minimum . \qquad \text{(Eq. A.39)}$$

For a quadratic polynomial function, the minimization of ε results in the following convolution operators, allowing coefficient a_{oo} to be calculated for areas of size 3x3, 5x5 and 7x7 respectively:

$$a_{oo} = \frac{1}{9}\begin{bmatrix} -1 & 2 & -1 \\ 2 & 5 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

$$a_{oo} = \frac{1}{175}\begin{bmatrix} -13 & 2 & 7 & 2 & -13 \\ 2 & 17 & 22 & 17 & 2 \\ 7 & 22 & 27 & 22 & 7 \\ 2 & 17 & 22 & 17 & 2 \\ -13 & 2 & 7 & 2 & -13 \end{bmatrix}$$

$$a_{oo} = \frac{1}{147}\begin{bmatrix} -7 & -2 & 1 & 2 & 1 & -2 & -7 \\ -2 & 3 & 6 & 7 & 6 & 3 & -2 \\ 1 & 6 & 9 & 10 & 9 & 6 & 1 \\ 2 & 7 & 10 & 11 & 10 & 7 & 2 \\ 1 & 6 & 9 & 10 & 9 & 6 & 1 \\ -2 & 3 & 6 & 7 & 6 & 3 & -2 \\ -7 & -2 & 1 & 2 & 1 & -2 & -7 \end{bmatrix}$$

(Eq. A.40)

Polynomial approximation functions of the second degree are appropriate, considering the curved shape of 2-DE images.

Adaptive smoothing

The adaptive smoothing technique is based on the anisotropic diffusion equation. This method smoothes images, but it preserves pertinent discontinuities. The idea is to weight the smooth operator according to the magnitude of the discontinuity d. This means that, for each point in the image, a continuity value w is calculated using a decreasing function $f(d)$, such as $f(0) = 1$ and $f(d) \rightarrow 0$ as d increases, where d represents the amount of discontinuity at each point.

For an image $I(x, y)$, the two-dimensional adaptive filter can be expressed by the following equation:

$$I^{(t+1)}(x, y) = \frac{1}{N^{(t)}} \sum_{i=-1}^{1} \sum_{j=-1}^{1} I(x+i, y+j) w^{(t)}, \quad \text{(Eq. A.41)}$$

where

$$w^{(t)}(x, y) = f(d^{(t)}(x, y)) = e^{-\left(\frac{|d^{(t)}(x, y)|^2}{2K^2}\right)}, \quad \text{(Eq. A.42)}$$

with $d^{(t)}(x, y) = \sqrt{G_x^{\,2} + G_y^2}$, (Eq. A.43)

where G_x and G_y are the gradients along the x and y axes respectively, and

$$N^{(t)} = \sum_{i=-1}^{1} \sum_{j=-1}^{1} w(x+i, y+j) w^{(t)}. \quad \text{(Eq. A.44)}$$

Figure 0-12 illustrates the isotropic and anisotropic diffusion smooth operators on a 1D signal representing the grey-level profile of one row in a 2-DE gel image. (a1) and (b1) show the original profile. (a2) and (a3) show the profile after two, respectively five iterations of the isotropic smooth operator. (b2) and (b3) display the profile after four, respectively ten iterations of the anisotropic smooth operator, with parameter $k = 2$. As one can observe, the second

method preserves significant discontinuities and signal shape better than the first one.

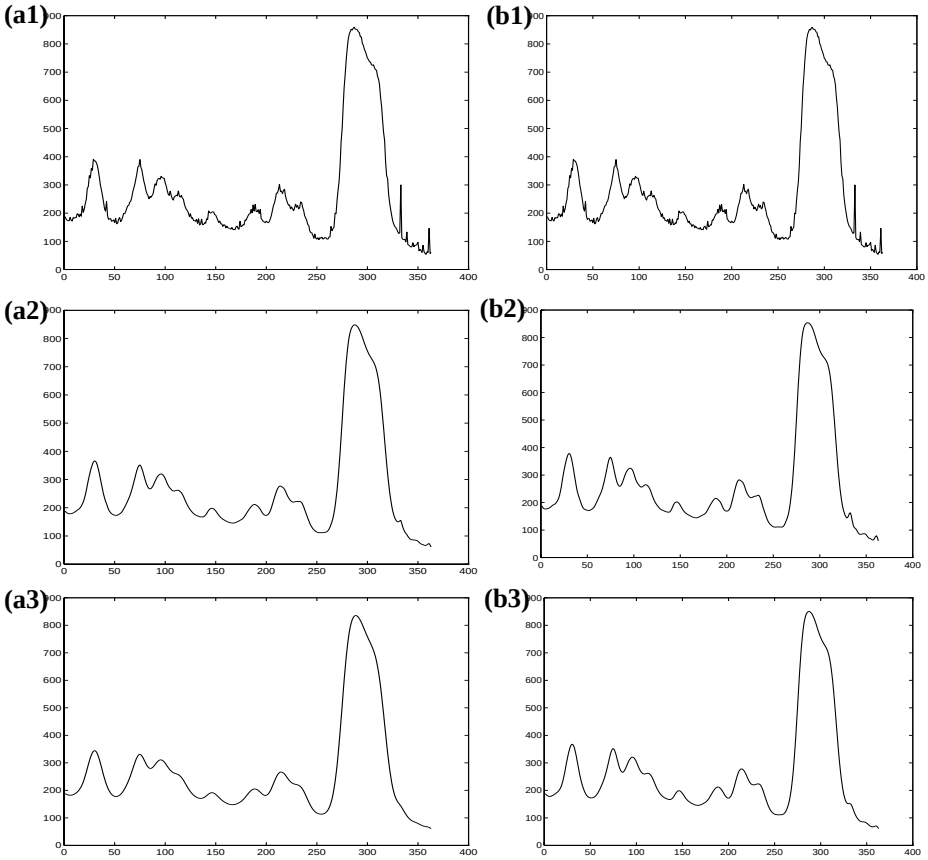


Figure 0-12. (a1) and (b1) represent the original profile of a 2-DE image. (a2), (a3) show the result of the application of isotropic diffusion smooth operator to the original profile after 2, respectively 5 iterations. In (b2),(b3) the anisotropic adaptive smooth operator was applied to the original profile 4, respectively 10 times with k=2. Significant discontinuities and the signal shape are better preserved by the anisotropic smooth method.

Histogram equalization

Histogram equalization is used to improve the visualization aspect of images. This operation attempts to correct the grey level values of a given image by changing its normalized histogram. For a given grey level distribution $p_x(i)$, with $i = 0 \ldots n_x$, a new distribution $p_y(j)$ with $j = 0 \ldots n_y$ is computed by searching a value N for each M such that:

$$\sum_{i}^{M} p_x(i) \leq \sum_{j}^{N} p_y(j) \qquad \text{(Eq. A.45)}$$

which defines a requantification law $y(x) = N(M)$, and

$$P_x(M) = \sum_i^M p_x(i) \qquad \text{(Eq. A.46)}$$

which is the cumulative law of $p_x(i)$.

Using this formalism, the following transformations may be defined:

- Uniform:

$$p_y(y) = \frac{1}{(y_{max} + y_{min})} \text{ with } y_{min} \le y \le y_{max}$$
$$\Rightarrow \quad y(x) = (y_{max} + y_{min}) P_x(x) + y_{min} \qquad \text{(Eq. A.47)}$$

- Exponential:

$$p_y(y) = \alpha \cdot \exp(-\alpha(y - y_{min})) \text{ with } y_{min} \le y$$
$$\Rightarrow \quad y(x) = y_{min} + \frac{1}{\alpha}\log(1 - P_x(x)) \qquad \text{(Eq. A.48)}$$

- Logarithmic:

$$p_y(y) = (y(\log(y_{max}) - \log(y_{min})))^{-1}$$
$$\Rightarrow \quad y(x) = y_{min}(y_{max}/y_{min})^{P_x(x)} \qquad \text{(Eq. A.49)}$$

- Cube-Root:

$$p_y(y) = \frac{1}{3}\frac{\sqrt[3]{y^{-2}}}{\sqrt[3]{y_{max}} - \sqrt[3]{y_{min}}}$$
$$\Rightarrow \quad y(x) = \sqrt[3]{(\sqrt[3]{y_{max}} - \sqrt[3]{y_{min}}) P_x(x) + \sqrt[3]{y_{min}}} \qquad \text{(Eq. A.50)}$$

where y_{max}, y_{min} are the maximum, respectively minimum possible value of the requantification law $y(x)$, and

$$P_x(x) = \sum_i^x p_x(i) \qquad \text{(Eq. A.51)}$$

is the cumulative law.

The Melanie II system equalizes an image by first splitting it into several regions. Then, for each region, the histogram is computed. Next, histograms corresponding to regions are equalized using one of the transformation functions described above. Finally, for each pixel, the new equalized value is computed by bilinear interpolation on the four possible transformations of the value using its neighbor equalized histograms.

Grey level remapping

This filter corrects intensity values by mapping each pixel value of an image I with a dynamic range $[min, max]$ into an image I' with a dynamic range $[MIN, MAX]$ according to a mapping function F, using the following expression:

$$I'(x, y) = F\left(\frac{I(x, y) - min}{max - min}\right)(MAX - MIN) + MIN \qquad \text{(Eq. A.52)}$$

where F is a bijective monotone function into $[0, 1]$ that maps this interval into $[0, 1]$. The function F may be:

- Linear: $F(x) = x$,

$$I'(x, y) = \frac{I(x, y) - min}{max - min}(MAX - MIN) + MIN; \qquad \text{(Eq. A.53)}$$

- Logarithmic: $F(x) = \frac{\log(ax + 1)}{\log(n)}$ in base n,

$$I'(x, y) = \log\left(a \cdot \left(\frac{I(x, y) - min}{max - min}\right) + 1\right)\frac{(MAX - MIN)}{\log(n)} + MIN \qquad \text{(Eq. A.54)}$$

where $a = e^{\log(n)} - 1$;

- Exponential: $F(x) = \frac{\exp(ax) - 1}{\exp(n) - 1}$,

$$I'(x, y) = \left(\exp\left(a \cdot \left(\frac{I(x, y) - min}{max - min}\right)\right) - 1\right)\frac{(MAX - MIN)}{\exp(n) - 1} + MIN \qquad \text{(Eq. A.55)}$$

where $a = n$.

Contrast enhancement

The contrast enhancement filter normalizes pixel values of a given image $I(x, y)$. The interval $[min, max]$ is first mapped to the interval $[0, 1]$, then it is remapped to $[0, 1]$ using a transformation function F. Finally, the new pixel values of the contrasted image are computed using the following expression:

$$I'(x, y) = F\left(\frac{I(x, y) - min}{max - min}\right) \times (\text{Highest grey}), \qquad \text{(Eq. A.56)}$$

where max and min are the highest, respectively lowest intensity values of image $I(x, y)$, *Highest grey* is the highest intensity value supported by the current image format, and F is one of the following transformation functions:

- Quadratic:

  $F(z) = z^2$ with $z \in [0, 1]$ a real value.

- Sigmoid:

$$F(z) = \frac{erf\left(\alpha\left(x - \frac{1}{2}\right)\right)}{2 \cdot erf\left(\frac{\alpha}{2}\right)} + \frac{1}{2}, \quad \text{(Eq. A.57)}$$

where $z \in [0, 1]$, $\alpha = 4$, and

$$erf(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt. \quad \text{(Eq. A.58)}$$

- Wallis:

  The contrast enhancement is performed using a statistical approach. The contrasted image is computed using the following expression:

$$I'(x, y) = \frac{I(x, y)}{S(x, y)}, \quad \text{(Eq. A.59)}$$

  where $S(x, y)$ is the standard deviation given by:

$$S(x, y) = \frac{1}{(2k + 1)^2} \sum_{i=-k}^{k} \sum_{j=-k}^{k} (I(x + i, y + j) - M(x + i, y + j))^2 \quad \text{(Eq. A.60)}$$

  with

$$M(x, y) = \frac{1}{(2k + 1)^2} \sum_{i=-k}^{k} \sum_{j=-k}^{k} I(x, y). \quad \text{(Eq. A.61)}$$

Background filtering

The background of a 2-DE image may be eliminated by using one of two background subtraction algorithms: subtraction of global image minimum and subtraction using polynomial functions.

Subtraction of global image minimum

The minimum pixel value in the gel is subtracted from all pixel values in the image.

Background estimation using polynomial functions

Background estimation of a 2-DE image is achieved by fitting the pixel values located outside the spot shapes with a third order polynomial function of the following form:

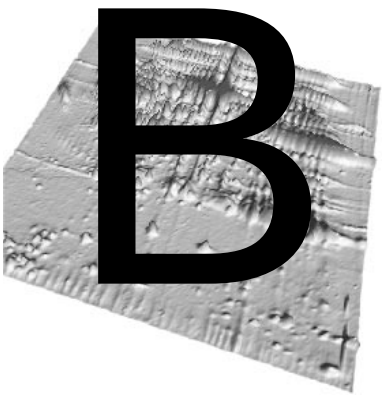
$$B(x, y) = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + a_6 y^2 + a_7 x^3 + a_8 x^2 y + a_9 xy^2 + a_{10} y^3, \quad \text{(Eq. A.62)}$$

where $B(x, y)$ is the estimated background pixel value, x is the horizontal position, and y is the vertical position of the pixel in the image, and $a_1, \ldots, a_{10}$ are the polynomial parameters. These parameters are determined for a given gel in the following steps:

1 spot shapes are detected by the method described in *Spot detection on page A-1*,

2 a grid of 3x3 pixels is chosen in order to select pixels in the image. Only the pixels that belong outside the spot shapes are selected,

3 selected pixels are fitted with (Eq. A.62) using the Least Squares method. This is achieved similarly to the methdo described in *Gaussian fitting on page A-13*.



KNOWN BUGS

PC / Macintosh

PC and **Macintosh**

Melanie II has been designed to work with 256 color displays. If you are using it with a graphics card that is able to display more than 256 colors, you have to set your monitor to 256 colors.

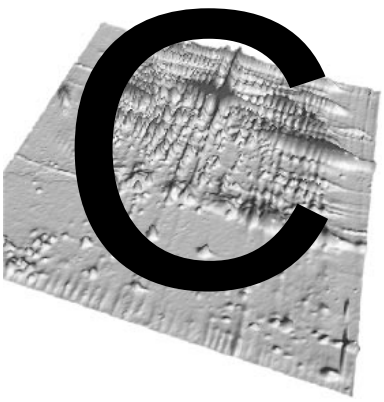
If printed reports are truncated, you may change the paper orientation to landscape in the printer set-up.

Unix

Unix

Netscape Navigator® loads its own color table according to the colors already in use at load time. If you have loaded Netscape Navigator® prior to using Melanie II, and you feel that the colors are badly displayed in Netscape, you should quit the Netscape Navigator® browser and run it again.

If you cannot select a gel, because it was placed outside the visible parts of the screen, then select all gels with *Select* → *Gels* → *Visible Gels* and *Show* → *Show Corner*.



REFERENCES

The Melanie project

1 *The Melanie II 2-DE analysis software WWW page*:
http://www.expasy.ch/melanie/melanie-top.html

2 Appel R.D., Bologna G., Hochstrasser D.F. *Classification tools for diagnostic rule formation from protein maps*. Proc. MIE'93, 11th International Congress of the European Federation for Medical Informatics, Jerusalem, April 18-22, 1993, 40-44.

3 Appel R.D., Hochstrasser D.F., Funk M., Vargas R.J., Pellegrini C., Muller A.F., Scherrer J-R. *The MELANIE project - From a Biopsy to Automatic Protein Map Interpretation by Computer*. Electrophoresis 12, 1991, 722-735.

4 Armitage P., Berry G. *Statistical Methods and Medical Research*. Blackwell Scientific Publications. Oxford, London, 1987.

5 Miller M.J., Olson A.D., Thorgeirsson S.S. *Computer analysis of two-dimensional gels: Automatic matching*. Electrophoresis 5, 297-303, 1984.

6 Pun T., Hochstrasser D.F., Appel R.D., Funk, M., Villars-Augsburger, V., Pellegrini C. *Computerized classification of two-dimensional gel electrophoretograms by correspondence analysis and ascendant hierarchical clustering*. Applied and Theoretical Electrophoresis 1, 3-9, 1988.

7 Vargas R.J. *Two-dimensional gel electrophoresis computer analysis systems: from image acquisition to protein identification*. Ph.D. thesis, Faculty of science, Geneva University, 1996.

8 Wilkins M.C., Hochstrasser D.F., Sanchez J.C., Bairoch A., Appel R.D. *Integrating two-dimensional gel databases using the Melanie II software*. Trends in Biochemical Sciences TiBS December (252), Vol. 21, No. 12, pp. 496-497, 1996.

Two-dimensional electrophoresis (2-DE)

9 *2-D PAGE protocols*:

http://www.expasy.ch/ch2d/technical-info.html

10 Herbert B.R., Sanchez J.-C., Bini L. *Two-Dimensional Electrophoresis: The State of the Art and Future Directions*. In: Proteome research: new frontiers in functional genomics, M.R. Wilkins, K.L. Williams, R.D. Appel, D.F. Hochstrasser (Eds), Springer Verlag, 1997.

2-DE databases

11 *The SWISS-2DPAGE 2-DE database*:

http://www.expasy.ch/ch2d/ch2d-top.html

12 *WORLD-2DPAGE: index to federated 2-DE databases*:

http://www.expasy.ch/ch2d/2d-index.html

13 Appel R.D. *Interfacing and Integrating Databases*. In: Proteome research: new frontiers in functional genomics, M.R. Wilkins, K.L. Williams, R.D. Appel, D.F. Hochstrasser (Eds), Springer Verlag. 1997.

14 Appel R.D., Bairoch A., Sanchez J.C., Vargas J.R., Golaz O., Pasquali C., Hochstrasser D.H. *Federated 2-DE database: a simple means of publishing 2-DE data*. Electrophoresis 17, 1996, 540-546.

15 Appel R.D., Sanchez J.C., Bairoch A., Golaz O., Miu M., Pasquali C., Vargas R.J., Hughes G., Hochstrasser D.F. *SWISS-2DPAGE: a database of two-dimensional gel electrophoresis images*. Electrophoresis 14, 1993, 1232-1238.

16 Appel R.D., Sanchez J-C., Bairoch A., Golaz O., Ravier F., Pasquali C., Hughes G.J., Hochstrasser D. *The SWISS-2DPAGE database of two-dimensional polyacrylamide gel electrophoresis*. Nucleic Acids Research, 1996, Vol. 22, No. 17, 3581-3582.

17 Golaz, O., Hughes, G.J., Frutiger, S., Paquet, N., Bairoch, A., Pasquali, C., Sanchez, J.-C., Tissot, J.-D., Appel, R.D., Walzer, C., Balant, L., Hochstrasser, D.F. *Plasma & Red Blood Cell Protein Maps: The 1993 Update*. Electrophoresis 14, 1993, 1223-1231.

18 Gravel P., Sanchez J.C., Walzer C., Golaz O., Hochstrasser D.F., Balant L.P., Hughes G.J., Garcia-Sevilla J., Guimon J. *Human blood platelet protein map established by two-dimensional polyacrylamide gel electrophoresis*. Electrophoresis 16, 1995, 1152-1159.

19 Hughes, G.J., Frutiger, S., Paquet, N., Pasquali, C., Sanchez, J-C., Tissot, J.D., Bairoch, A., Appel, R.D., Hochstrasser, D.F. *Human Liver Protein Map: The 1993 Update*. Electrophoresis 14, 1993, 1216-1222.

20 Pasquali C., VanBogelen R.A., Wilkins M., Frutiger S., Appel R.D., Vargas R., Sanchez J.C., Hochstrasser D.F. *The ESCHERICHIA COLI SWISS-2DPAGE database*. Electrophoresis 17, 1996, 547-555.

21Peitsch, M.R., Wilkins M.R., Tonella T., Sanchez J.C., Appel R.D., Hochstrasser D.F. *Large-scale protein modelling and integration with the SWISS-PROT and SWISS-2DPAGE databases: The example of Escherichia coli.* Electrophoresis 18, 1997, 498-501.

22Sanchez J.-C., Appel R.D., Golaz O., Pasquali C., Ravier F., Bairoch A., Hochstrasser D.F. *Inside SWISS-2DPAGE database*. Electrophoresis 16, 1995, 1131-1151.

23Sanchez J.C., Golaz O., Frutiger S., Schaller D., Appel R.D., Bairoch A., Hughes G.J., Hochstrasser D.F. *The YEAST SWISS-2DPAGE database*. Electrophoresis 17, 1996, 556-565.

The ExPASy WWW server

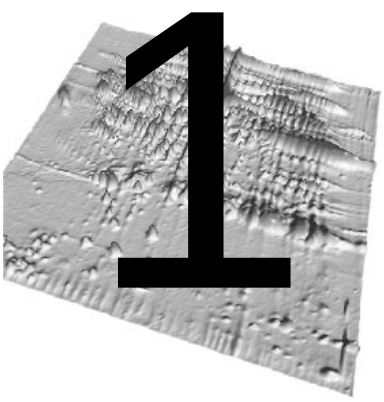
24*The ExPASy WWW server*:

http://www.expasy.ch/

25Appel R.D., Bairoch A., Hochstrasser D.F. *A new generation of information retrieval tools for biologists: the example of the expasy WWW server*. Trends in Biochemical Sciences TiBS June 1994 (222), Vol. 19, No. 6, pp. 258-260.

26Hochstrasser D.F., Appel R.D., Golaz O., Pasquali C., Sanchez J.-C., Bairoch A. *Sharing of Worldwide Spread Knowledge Using Hypermedia Facilities & Fast Communications Protocols (Mosaic & World-Wide Web): The Example of ExPASy*. Meth. Inform. Med., 1995, 34: 75-78.

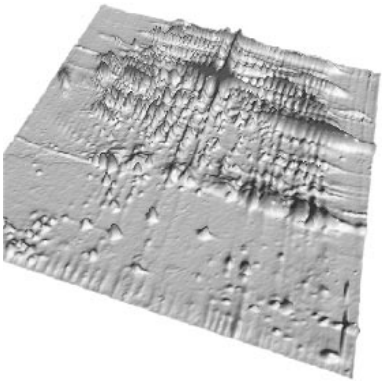
27Wilkins M.R., Gasteiger E., Bairoch A., Sanchez J.C., Williams K.L., Appel R.D., Hochstrasser D.F. Protein identification and Analysis Tools in the ExPASy Server. In 2-D Protein Gel Electrophoresis Protocols, A.J Link (Ed.), Humana Press 1997, in press.



1 TECHNICAL SUPPORT

Bio-Rad provides technical support on the features of the Melanie II software. If you have any problems or any questions on the use of the features, contact your local Bio-Rad office, or in the U.S. call Bio-Rad Technical Service at 1-800-4BIORAD (1-800-424-6723).

Please report any reproducible software problems to Bio-Rad Laboratories.



INDEX

E

F

G

H

I

L

M

N

O

P

Q

R

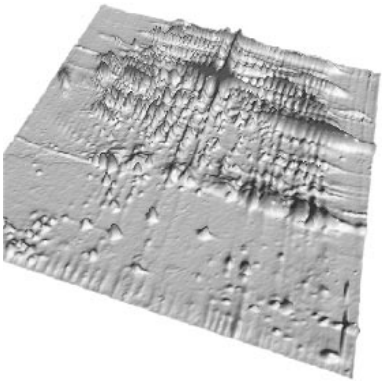
S

T

U

W

Z



CREDITS

The **Melanie** project was begun in 1985 at the Geneva University Hospital and the Computer Science Department of Geneva University, Switzerland by Ron D. Appel, Matthieu Funk, Denis F. Hochstrasser and C. Pellegrini. The following persons have since worked on the Melanie II software:

Software design

- Peter Burke
- Andrew Goodman
- Jan Snydr-Michael
- J. Reynaldo Vargas
- Daniel Walther

Software development

- Horia Ciobanu
- Anca Dima
- Romica Iancu
- Elena Manoila
- Marina Miu
- Gheorghe Popescu

2-D PAGE application

- Olivier Golaz
- Christian Pasquali
- Florence Ravier
- Véronique Rouge
- Jean-Charles Sanchez

Melanie II is a completely redesigned and rewritten software package, based on the Melanie-1 2-DE analysis software that was developed at the Geneva University Hospital and the Computer Science Department of Geneva University as an add-on to the Elsie 2-DE system [see reference 5].

The 3-D view of the Human plasma 2-DE gel image that appears in the Melanie II logo and in the chapter headers has been produced by Luc Bidaut.