

# Mixture Model Analysis of DNA Microarray Images

K. Blekas, N. P. Galatsanos\*, A. Likas and I. E. Lagaris  
Department of Computer Science, University of Ioannina,  
P.O. Box 1186, 45110 Ioannina, Greece  
E-mail: {kblekas,galatsanos,arly,lagaris}@cs.uoi.gr

## Abstract

In this paper we propose a new methodology for analysis of microarray images. First a new gridding algorithm is proposed for determining the individual spots and their borders. Then, a Gaussian Mixture Model (GMM) approach is presented for the analysis of the individual spot images. The main advantages of the proposed methodology are modeling flexibility and adaptability to the data, which are well known strengths of GMM. The maximum likelihood (ML) and maximum a posteriori (MAP) approaches are used to estimate the GMM parameters via the Expectation Maximization (EM) algorithm. The proposed approach has the ability to detect and compensate for artifacts that might occur in microarray images. This is accomplished by a model-based criterion that selects the number of the mixture components. We present numerical experiments with artificial and real data where we compare the proposed approach with previous ones and existing software tools for microarray image analysis and demonstrate its advantages.

**Keywords:** DNA microarray image analysis, microarray gridding, Gaussian mixture models, maximum likelihood, maximum a posteriori, Markov random fields, Expectation-Maximization algorithm, cross-validated likelihood

## 1 Introduction

DNA microarrays [1] are used to measure the expression levels of thousands of genes simultaneously over different time points and different experiments. In microarray experiments, the two mRNA samples to be compared are reverse transcribed into cDNA and then hybridized simultaneously to a glass slide. The end product of a comparative hybridization experiment is a scanned array image, where the measured intensities from the two fluorescent reporters have been colored red (R) and green (G) and overlaid. This array image is structured with intensity spots located on a grid and must be scanned to determine how much each probe is

---

\*To whom correspondence should be addressed.

bound to the spots when stimulated by a laser. Yellow spots have roughly equal amounts of bound cDNA from each sample and so have equal intensity in the R and G channels (red + green = yellow). Gene expression data derived from arrays measure spots quantitatively and can be used further for several analyses [2, 3].

It has been shown [1] that *background correction* is an important task in the analysis of microarray images. This is necessary in order to remove the contribution in intensity which is not due to the hybridization of the cDNA samples to the spotted DNA. The R and G intensities of a perfect microarray image depend only on the dye of interest. However, due to system imperfections and the microarray image generation process, the resulting images, in addition to background fluorescence, contain also other types of undesired signals which are termed in the rest of this paper as *artifacts*. The correction of such artifacts is crucial to making accurate expression measurements, because unlike background fluorescence their spatial location is unknown and can lead to errors propagated to all subsequent stages of the analysis [4].

Processing microarrays images requires two tasks. First, the individual spots and their borders are determined. This process is also known as *gridding*. Second, each spot is analyzed to determine the corresponding gene expression level. A number of software tools have been introduced that are available either commercially or for research only purposes for the analysis of the microarray images [1, 5, 6, 7]. These tools use simple gridding methods, which are based either on a grid with uniform cells, or on manual specifications of the spot borders. For spot analysis some existing tools assume circular spots for example, the ScanAlyze [6] and the GenePix [7]. Others use simplistic local thresholding based techniques, for example the Spotfinder [5].

Histogram-based clustering methods have been also proposed for spot segmentation [8, 9, 10]. However, these methods use the well known  $K$ -means and the  $K$ -medoids algorithms that do not adapt well to irregularly based clusters and do not utilize all the available prior knowledge about the data. Furthermore, all previous proposed methods correct only for

background fluorescence and ignore the presence of artifacts.

The main contributions of this work are two; first, a new automatic gridding scheme and second, the application of Gaussian mixture models (GMM) for analyzing microarray spot images [4]. This allows to bring on bear to this problem all the known advantages and powerful features of the GMM methodology, such as adaptability to the data, modeling flexibility and robustness, that make it attractive for a wide range of applications [11, 12]. The proposed methodology consists of three main steps. First, the new scheme for determining the individual spot borders in a microarray image is presented. This method does not require any human intervention and is very simple and fast. It is hierarchical in nature since it first uses the global and then the local properties of the microarray image, thus it is also very robust.

Second, after determining the spot boundaries, the probability density of each spot pixels is modeled using a GMM with  $K$  components. Two scenarios are possible. First,  $K = 2$  in which case two components are used corresponding to pixels labeled as *background* and *foreground*. Second,  $K = 3$  when in addition to background and foreground we have pixels which are labeled as *artifacts*. The identification of the appropriate value of  $K$  is accomplished using the cross-validated likelihood criterion [13]. This can be considered as *artifact detection and correction* mechanism, since when  $K = 3$  an artifact is identified which is ignored in the subsequent analysis of this spot. Two approaches are proposed for estimating the GMM parameters. The first one is based on the Expectation-Maximization (EM) algorithm [14] for *maximum likelihood* (ML) estimation of the parameters, while the second on a *maximum a posteriori* (MAP) formulation. The latter takes also into account prior knowledge about the spatial assignment of the pixel labels using a Markov Random Field (MRF) model [15].

Finally, based on the clustering results, the means of the background and foreground Gaussian components are used to calculate the normalized log-ratio for the fluorescence intensities ( $\log_2 R/G$ ). This task constitutes the *reduction* step of our approach and characterizes qualitatively each spot by finding its corresponding gene expression value.

The rest of this paper is organized as follows: In section 2 we present the proposed

technique for automatic gridding. Section 3 describes the two GMM approaches for spot image segmentation and the model-based criterion for estimating the number of mixture components. In section 4 we present numerical experiments that test the proposed gridding and clustering methodologies and compare them to existing software packages for microarray image analysis, as well as to recently published methods. For this purpose we used both artificial data, where the "ground truth" is known, together with real data. Finally, we present our conclusions in section 5.

## 2 Automatic Microarray Gridding

The process of determining the spot boundaries is frequently referred to as *gridding*. A variety of microarray gridding methods have been previously suggested in the literature. They determine individual spot boundaries either with user-defined anchor points [6] and semi-automated geometric techniques [10], or with complex methods that are computationally expensive [16]. Since typical microarray images contain hundreds or thousands of spots, a practical gridding method must be fully automatic, fast and simple.

The proposed gridding method uses a scheme that combines global and local segmentation mechanisms for defining the boundaries of each microarray spot. It initially creates *global* boundaries, which are horizontal and vertical straight lines spanning the entire image. To define the global boundaries we add the sums of the R and G intensities along the rows and columns of the microarray image. The resulting signals have multiple peaks each corresponding to the coordinates of a spot center. We use the mid point of two successive peaks of the row and column sums to define the global horizontal and vertical boundaries, respectively. Fig. 1 (a) illustrates this process for a  $5 \times 5$  grid.

In the next step, the global boundaries are refined. The horizontal boundary between spots  $S(i, j)$  and  $S(i + 1, j)$  is refined by locating the minimum of the sum of the rows (within the global boundary) of the R and G intensities of these spots. In the same spirit, the vertical boundary between spots  $S(i, j)$  and  $S(i, j + 1)$  is refined by locating the minimum

of the columns (within the global boundary) sums of the R and G intensities of these spots. This procedure is repeated in a row-by-row or column-by-column fashion, scanning the entire microarray image. Fig. 1 (b) illustrates an example of the global border refinement process.

It must be also noted that in many cases the color channels are not aligned with each other. In such cases one can use image alignment algorithms prior to the gridding task, see for example [17, 18, 19].

### 3 Mixture Models for Spot Analysis

Spot analysis refers to the task of labeling each pixel of a spot as background (B), foreground (F), and artifact (A). This can be viewed as a *clustering* problem which is tackled using GMM. Let  $x^i = [x_R^i, x_G^i]^T$  ( $i = 1, \dots, N$ ) denote the  $i$ th pixel value in a spot area, where the R and G correspond to the red and green intensities, respectively. In other words, the segmentation is applied to the color image and not to each color separately. GMMs [11, 12] represent density functions as a convex combination of  $K$  Gaussian component densities  $\phi(x|\theta^j) = \mathcal{N}(x|\mu_j, \Sigma_j)$ , where  $\mu_j$  is the mean and  $\Sigma_j$  the covariance matrix of the  $j$ th Gaussian, according to the formula

$$f(x^i|\Psi_K) = \sum_{j=1}^K \pi_j \phi(x^i|\theta^j) . \quad (1)$$

The parameters  $0 \leq \pi_j \leq 1$  represent the mixing weights satisfying that  $\sum_{j=1}^K \pi_j = 1$ , while  $\Psi_K$  is the vector of all unknown parameters of the model, i.e.  $\Psi_K = [\pi_1, \dots, \pi_K, \theta_1, \dots, \theta_K]$ , with  $\theta_j = [\mu_j, \Sigma_j]$ .

Having found the parameters of the GMM, the posterior probabilities that the  $i$ th pixel is assigned to the  $j$  component is given by

$$P(j|i) = \frac{\pi_j \phi(x^i|\mu_j, \Sigma_j)}{\sum_{l=1}^K \pi_l \phi(x^i|\mu_l, \Sigma_l)} . \quad (2)$$

Therefore, the  $i$ th pixel is assigned to the label  $l$  with the largest posterior probability ( $P(l|i) > P(j|i) \forall j \neq l$ ).

### 3.1 Maximum Likelihood (ML) Estimation of GMM Parameters

A common approach for estimating the model parameters of the GMM (Eq. 1) is based on maximization of the likelihood (ML)

$$\mathcal{L}(X|\Psi_K) = \sum_{i=1}^N \log f(x^i|\Psi_K) = \sum_{i=1}^N \log \left\{ \sum_{j=1}^K \pi_j \phi(x^i|\theta_j) \right\}. \quad (3)$$

The EM algorithm is a popular method for ML estimation since it is simple to implement and guarantees convergence to a local maximum of the likelihood function [14, 12].

Starting from an initial guess of the model parameters  $\Psi_K$ , at each iteration ( $t$ ) the EM algorithm proceeds in two steps. The  $E$ -step, where the posterior probabilities are computed

$$z_j^{i(t)} = \frac{\pi_j^{(t)} \phi(x^i|\mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{l=1}^K \pi_l^{(t)} \phi(x^i|\mu_l^{(t)}, \Sigma_l^{(t)})}, \quad (4)$$

and the  $M$ -step, where the model parameters are updated

$$\pi_j^{(t+1)} = \frac{1}{N} \sum_{i=1}^N z_j^{i(t)}, \quad (5)$$

$$\mu_j^{(t+1)} = \frac{\sum_{i=1}^N z_j^{i(t)} x^i}{\sum_{i=1}^N z_j^{i(t)}}, \quad \Sigma_j^{(t+1)} = \frac{\sum_{i=1}^N z_j^{i(t)} (x^i - \mu_j^{(t+1)})(x^i - \mu_j^{(t+1)})^T}{\sum_{i=1}^N z_j^{i(t)}}. \quad (6)$$

In image segmentation the spatial adjacency of pixels with the same label is an important prior information that could be also taken into account [20, 21]. Since the ML approach does not provide this capability, an alternative method for maximum a posteriori (MAP) estimation of GMM parameters will be described next. However, before we address this problem, we will elaborate on the problem of selecting the number of the mixture components  $K$ , and see how it fits in the proposed microarray image analysis methodology.

### 3.2 Cross-validated Likelihood for Artifact Identification

The application of the EM algorithm to GMM requires knowledge of the number of the mixture components  $K$  used in the model. Since previous approaches for microarray spot

analysis assume 2 labels, background (B) and foreground (F), it is reasonable to consider GMMs with  $K = 2$ . However, this assumption cannot handle the existence of artifacts which must also be taken into account, see spots in Fig. 7. In this case an additional cluster appears in the data, therefore they are better modeled by a GMM with  $K = 3$ . This effect can be visualized by comparing the scatter plots in the Fig. 6 with those in Fig. 8. Thus, the artifact detection problem corresponds to a model order selection problem between a 2-component or a 3-component GMM.

Cross-validated likelihood [13] provides an efficient model order selection framework for GMMs. Following this scheme, a  $K$ -component model is evaluated by splitting the data in  $u$  disjoint partitions (folds)  $X_s$ ,  $s = 1, \dots, u$  (of approximately equal size). For each fold we estimate the  $\Psi_K^s$  parameters of a GMM with  $K$  components using the dataset  $X - \{X_s\}$ . Then, we calculate the likelihood of this model  $\mathcal{L}(X_s|\Psi_K^s)$  using  $X_s$  as a test set. Next  $\mathcal{L}(X_s|\Psi_K^s)$  is averaged over the  $u$  folds in order to obtain the cross-validated evaluation for the  $K$ -component model

$$CV_K = \frac{1}{u} \sum_{s=1}^u \mathcal{L}(X_s|\Psi_K^s) . \quad (7)$$

The  $CV_K$  value is computed for the two candidate values  $K = \{2, 3\}$  and we select the model order with the largest  $CV_K$ . It must be noted that in our experiments we have selected  $u = 10$  for the number of folds. When  $K = 3$  (existence of artifacts) the criterion used to determine which one of the three is the artifact cluster is the aggregate variance in all dimensions. In other words, the cluster with the largest  $Tr(\Sigma_j)$  is considered as artifact.

### 3.3 Maximum A Posteriori (MAP) Estimation of GMM Parameters

According to this approach [15], the probabilities  $\pi_j^i = P(j|\text{position } i)$  of the pixel located at the  $i$ th position is assigned to the  $j$ th label are considered as additional model parameters that satisfy the constraints:  $0 \leq \pi_j^i \leq 1$  and  $\sum_{j=1}^K \pi_j^i = 1$ . By denoting as  $\Pi = \{\pi^1, \dots, \pi^N\}$  the set of probability vectors and  $\Theta = \{\theta_1, \dots, \theta_K\}$  the set of Gaussian component parameters,

the density function is given by

$$f(x^i|\Pi, \Theta) = \sum_{j=1}^K \pi_j^i \phi(x^i|\theta_j) . \quad (8)$$

Spatial adjacency of pixel labels is taken into account by using a suitable prior density function for the parameter set  $\Pi$ . This is given by the Markov Random Field (MRF) model [20, 15, 21]

$$p(\Pi) = \frac{1}{Z} \exp(-U(\Pi)) , \text{ and } U(\Pi) = \beta \sum_{i=1}^N V_{\mathcal{N}_i}(\Pi) , \quad (9)$$

where  $Z$  is a normalizing constant, and  $\beta$  a regularization parameter. The function  $V_{\mathcal{N}_i}(\Pi)$  is the clique potential function of the pixel label vectors  $\{\pi^m\}$  within the neighborhood  $\mathcal{N}_i$  (horizontally, vertically, and diagonally adjacent pixels) to the  $i$ th pixel and is computed as follows

$$V_{\mathcal{N}_i}(\Pi) = \sum_{m \in \mathcal{N}_i} g(u_{i,m}) , \text{ where } u_{i,m} = |\pi^i - \pi^m|^2 = \sum_{j=1}^K (\pi_j^i - \pi_j^m)^2 . \quad (10)$$

The function  $g(u)$  must be nonnegative and monotonically increasing [20] and we used  $g(u) = (1 + u^{-1})^{-1}$ .

Given the above prior density (Eq. 9), a *posteriori* log-density function can be formed as follows

$$p(\Pi, \Theta|X) = \sum_{i=1}^N \log f(x^i|\Pi, \Theta) + \log p(\Pi) , \quad (11)$$

and maximized for the MAP estimation of the model parameters  $\Pi, \Theta$ . The EM algorithm can also be used for this case [15]. The E-step is given by

$$z_j^{i(t)} = \frac{\pi_j^{i(t)} \phi(x^i|\mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{l=1}^K \pi_l^{i(t)} \phi(x^i|\mu_l^{(t)}, \Sigma_l^{(t)})} , \quad (12)$$

while the M-step requires the maximization of the following log-likelihood [15]

$$Q_{MAP}(\Pi, \Theta|\Pi^{(t)}\Theta^{(t)}) = \sum_{i=1}^N \sum_{j=1}^K z_j^i \{\log(\pi_j^i) + \log(\phi(x^i|\theta^j))\} - \beta \sum_{i=1}^N \sum_{m \in \mathcal{N}_i} g(u_{i,m}) . \quad (13)$$

This gives update equations for the parameters of the component densities,  $\mu_j$  and  $\Sigma_j$  similar to those of Eq. (6) of the ML-approach of the GMM.

However, the maximization of the function  $Q_{MAP}$  with respect to the label parameters  $\{\pi_j^i\}$  does not lead to closed form update equations, since we must take into account the constraints:  $0 \leq \pi_j^i \leq 1$  and  $\sum_{j=1}^K \pi_j^i = 1$ . Due to this difficulty, a Generalized EM scheme was adopted in [15] based on an iterative *Gradient Projection* method. For this approach, the gradient of the MAP function is first projected onto the hyperplane of the constraints, and then a line search is performed along the direction of the projected gradient to find the parameters  $\{\pi_j^i\}$  that maximizes the  $Q_{MAP}$  function.

Here we use an improved M-step in order to maximize  $Q_{MAP}$  with respect to  $\pi_j^i$  by formulating the problem as a *constrained convex quadratic programming* (QP) problem. We found that this is advantageous, since it provides a better and faster update rule for estimating label parameters  $\{\pi_j^i\}$  that meets all the available constraints [22]. A more detailed description of the M-step for this method is given in Appendix A.

## 4 Experimental results

A variety of experiments have been performed to evaluate the proposed methodology for the analysis of DNA microarray images. The test images used were artificially created or obtained from publicly available microarray databases described in [2] and [3].

### 4.1 Gridding experiments

At first, we tested the proposed gridding technique for partitioning grid structures into distinct spot areas. In order to objectively evaluate and compare our method the following experimental study was contacted:

We applied our gridding method, and two other widely used microarray image analysis tools, the Spotfinder [5] and the ScanAlyze [6], to ten (10) spot arrays, (arbitrarily) selected from ten (10) different real microarray images. Thus, in total, nearly 3500 spots were used in

this experiment. Each method was evaluated by visually inspecting the gridding results and assigning each spot to one of three categories: *perfectly*, *marginally* and *incorrectly* gridded. A spot was perfectly, marginally, or incorrectly gridded if the entire, at least 80%, or less than 80% of the spot area was contained in the assigned grid.

The results of this study are shown in Table 1. These results clearly indicate that our method determines the spot areas more accurately than the two other methods. It must be also noted that the Spotfinder and ScanAlyze methods are based on manual gridding. More specifically, the size of the spot array is first defined. Then a rectangle is placed manually on the image. Based on the provided dimensions the rectangle is divided into equal rectangular or circular cells each corresponding to the region of a spot. Thus the outcome of the gridding process for these methods is user dependent, while our method is fully automated. In these experiments, we tried to the best of our ability to optimize the results obtained by the Spotfinder and ScanAlyze tools.

In Fig. 2 we provide the gridding results with one of the ten spot arrays using our approach as well as the two other image analysis tools, the ScanAlyze and Spotfinder. We also provide more detailed gridding results for individual spots in the first column of Figures 5 and 7.

## 4.2 Spot analysis experiments

After identifying the spot regions, we used the proposed GMM-based approach to analyze each spot region. More specifically, the procedure we followed consists of the following four stages:

1. Select the number of components  $K$  of the GMM model using the cross-validated likelihood method. In other words, test for the presence ( $K = 3$ ) or absence ( $K = 2$ ) of artifacts in a spot.
2. Estimate the parameters of the  $K$ -component GMM model using the ML or MAP technique and label each spot pixel with one of the  $K$  labels.

3. If  $K = 3$ , the artifact component (A) of the GMM is identified by using the maximum variance criterion. Then, the remaining two clusters are labeled as F and B using the criterion  $\|\mu^F\| > \|\mu^B\|$ .
4. Calculate the expression value of the corresponding gene according to the normalizing logarithmic ratio:

$$r = \log_2\left(\frac{\mu_R^F - \mu_R^B}{\mu_G^F - \mu_G^B}\right).$$

For comparison purposes we have also implemented two other methods proposed in [8, 9] for spot clustering, namely the  $K$ -means algorithm and the partitioning around medoids (PAM) method. These two methods do not provide model selection capabilities, and thus only two clusters ( $K = 2$ ) were considered,  $B$  and  $F$ .

At this point it should be also noted that filtering, such as low-pass or median, could be used for noise removal in a separate step prior to segmentation [9]. In our methodology, the proposed MAP approach provides a coherent framework for segmentation in which "noise filtering" is implicitly integrated. Furthermore, it uses a GMM to model the data and thus, unlike filtering, it also adapts to their statistics.

#### 4.2.1 With artificial spot images

In order to objectively compare the proposed GMM based methodology with previous ones we conducted Monte-Carlo simulations using artificially created spots for which the "ground truth" is known. The artificial spots were constructed with known mean intensities for the red (R) and green (G) channels both for the background ( $M^B$ ) and the foreground ( $M^F$ ). Then, the images were corrupted with additive white Gaussian noise at ten different levels. For statistical significance, the experiment at each noise level was repeated ten times with different noise realizations. Two criteria were used to evaluate the methods tested: a) the classification (segmentation) error defined as the percentage of mis-classified pixels after clustering, and b) the mean squared error (MSE) of the ratio  $\hat{r}$ , as estimated by each method over the ten

repetitions of each experiment, with respect to the true ratio  $r_{true} = (M_R^F - M_R^B)/(M_G^F - M_G^B)$ , i.e.

$$MSE = \frac{1}{10} \sum_{t=1}^{10} (\hat{r}_t - r_{true})^2 .$$

The MSE from the true ratio was used as a comparison metric since, as mentioned previously, this ratio is the feature used for further analysis of microarray data.

In Fig. 3 (a), (b) we show the resulting classification error and MSE curves as functions of the noise level to illustrate the performance of the four methods. In both curves, the  $x$ -axis corresponds to the signal-to-noise ratio (SNR) calculated in decibel units, while the  $y$ -axis in Fig. 3 (b) is in logarithmic scale. These results, demonstrate that the MAP GMM-based method outperforms all other methods. Furthermore, at all SNR levels, both the ML and the MAP GMM-based approaches provide both better segmentation accuracy and MSE values compared to the other methods, with these differences being quite significant at low SNR levels. In Fig. 4 three examples are displayed corresponding to three different SNR levels showing the segmentation and the ratio value for each one of the compared methods. It must be noted that in the above experiments all clustering methods were identically initialized. Furthermore, MAP parameter  $\beta = 1$  was used for all cases.

#### 4.2.2 With real spot images

We also tested the proposed spot analysis methodology with real data. Figures 5 and 7 illustrate the results obtained for several real spot examples. In each case we present the image segmentation results after labeling the pixels using each of the compared approaches. The spot segmentation map is constructed by setting the intensity value of each pixel equal to the mean value of the cluster that is assigned to. In the case of the proposed MAP approach, three different segmentation maps are presented that correspond to three values (0.01, 0.1, 1.0) for the regularization parameter  $\beta$  of the Gibbs prior (Eq. 9). In total, for each spot we provide six segmentation maps along with the corresponding fluorescent ratios.

More specifically, Fig. 5 represents comparative results from five spot examples where

no artifacts were detected according to the cross-validated likelihood criterion, i.e.  $K = 2$ . In cases where the shape of spots is not regular and their contour is not round (mostly due to retrieval of the microarrayer’s spotting pin), both GMM-based methods generate more regular foreground regions in comparison with the  $K$ -means and PAM clustering approaches. To better comprehend the behaviour of the different clustering methods, we present in Fig. 6 four scatter plots of the R and G pixel intensities for the spot  $S_2$  after labeling using GMM with the MAP (MAP-GMM), the ML (ML-GMM), the  $K$ -means and the PAM methods, respectively.

The main disadvantage of the  $K$ -means and PAM methods is that they are restricted to use as error metric the  $L_2$  distance from the mean or median of the cluster. Thus, they generate clusters which are separable by simple borders as shown in Figures 6, (c) and (d). In contrast, GMM-based methods generate ellipsoidal clusters with complex boundaries as shown in Figures 6, (a) and (b). As a result, the  $K$ -means and PAM methods in this example tend to overestimate the background clusters and provide spots with background "wholes", while the GMM-based methods provide more "uniform" spots.

Fig. 7 illustrates comparative results with another four spot examples that correspond to cases where an artifact was detected, i.e.  $K = 3$ . After labeling, the artifact pixels are excluded from the calculation of the fluorescent ratios. In the absence of an artifact correction methodology, the  $K$ -means and the PAM methods erroneously classify these pixels as foreground since the contribution of the artifact pixels is significant. The differences in the fluorescent ratios  $r$ , among these methods is noticeable. For example, in the case of spots  $S_3$  and  $S_5$  of Fig. 7, the  $K$ -means and PAM methods produce a ratio close to zero ( $r = 0$ ), since they consider as foreground the (yellow) artifact pixels. On the other hand, the proposed MAP-GMM and ML-GMM approaches, detect the presence of the artifact and generate more realistic foreground regions. Thus, the produced fluorescent ratios of about  $r = -0.7$  and  $r = 0.45$  seem to be more realistic for the spots  $S_2$  and  $S_3$ , respectively. We also present in Fig. 8 four plots of the R and G pixel intensity values for these two spot areas after labeling

pixels with the four approaches being compared. Again, the enhanced data fitting capabilities of the GMM-based approaches are obvious.

Another point to make in our experimental study concerns the comparison between the MAP-GMM and ML-GMM estimators. The results in Figures 5, 7 show that both approaches yield similar results in terms of the fluorescent ratios. However, they do not produce the same segmentation maps. For low values of the regularization parameter  $\beta$  ( $\beta \leq 0.01$ ) both methods generate identical segmentation maps. As the value of  $\beta$  grows in MAP-GMM, the contribution of the prior term increases and generates smoother foreground and background regions. Thus, it eliminates isolated foreground pixels located in background regions. While the value of the parameter  $\beta$  must be tuned, in our experiments we observed that a  $\beta$  value in the range  $[0.1, 1.0]$  gives satisfactory results. From this point of view, the MAP-GMM approach can be viewed as a method for noise reduction in the sense that it eliminates the effects of the microarray manufacturing imperfections.

In Fig. 9 we show some comparisons for spot quantification between the proposed method and two existing image analysis tools, more specifically the GenePix [7] and the Spotfinder [5]. Comparisons with the ScanAlyze [6] were not included since GenePix uses the same principle for spot segmentation. From Fig. 9 it is clear that the circle used in GenePix is not representative on many occasions, when the spot is irregularly shaped or when artifact islets are present, of the spot area. In other words, the analysis provided by GenePix is based only on the spatial properties of the spot and does not take into consideration the intensity of the pixels. For example, in spot  $S_5$  shown in Figures 5 and 9 the circle used by GenePix misses completely the crescent shaped spot which the proposed method captures quite accurately. This is also reflected in the large difference of the fluorescent ratios provided by these methods. Also in spot  $S_4$  in Figures 7 and 9 it is clear that the region selected by GenePix segmentation as foreground includes pixels that our algorithm labels as artifact and this is also reflected in the computed fluorescent ratios. Similarly, the thresholding based algorithm used in Spotfinder in certain instances of irregular spots and spots with artifacts

produces faulty segmentations, see for example spots  $S_1$  in Figures 5 and 7, respectively. In these spots also the fluorescent ratios provided by Spotfinder and our method are significantly different.

Finally, the last series of experiments uses an interesting family of microarray images provided by Agilent Technologies that have a specific imperfections: the spots in these images although perfectly circular, contain sometimes artifacts in their perimeter. Agilent provides analysis software that ignores the perimeter of the spot based on what is called as the "Cookie Cutter algorithm" [23]. We tested the proposed methodology with such images<sup>1</sup> and found that it is able to detect the presence of artifacts in these spots using the cross-validation criterion. Furthermore, it classifies as artifact a "don't like" region which is not taken into account during the ratio calculation. For comparison purposes, we also provide the segmentation and the ratio  $r$  results using the  $K$ -means and the PAM algorithms. Since the cross-validation method is specific to the GMM, only two clusters were used in these methods. In Fig. 10 we show five spot examples of this type of images. It is interesting to notice the considerable difference in the  $r$  ratios obtained by the proposed methodology with respect to the other methods for certain spot cases (e.g. case 5).

## 5 Conclusions

In this paper we have proposed a new fully automated approach for the analysis of microarray images. First we describe a new hierarchical gridding procedure based on the vertical and horizontal projections of the color images. This approach is simple, automatic, and provides better results compared with popular existing tools. However, the main novelty of this work is the proposed GMM-based methodology for spot image segmentation. Two methods for estimating the GMM parameters are presented: the ML and a MAP. Both approaches are based on the EM algorithm. A cross-validated likelihood criterion is also used to select the number of components of the GMM. This provides the capability to detect and correct

---

<sup>1</sup>Test images were downloaded from <http://www.silicocyte.com/dis/imagesforevaluation.htm>

artifacts in the spot area. As our experiments demonstrated, the proposed methodology produces better and more accurate results in terms of segmentation maps and fluorescence ratios as compared with existing software tools and other clustering methods proposed in previous works.

## Appendix A: An M-step for estimating the parameters $\pi_j^i$

To maximize  $Q_{MAP}$  (Eq. 13) with respect  $\pi_j^i$  we set its derivative equal to zero and obtain the following quadratic expression

$$4\beta \left[ \sum_{m \in \mathcal{N}_i} \dot{g}(u_{i,m}) \right] (\pi_j^i)^2 - 4\beta \left[ \sum_{m \in \mathcal{N}_i} \dot{g}(u_{i,m}) \pi_j^m \right] (\pi_j^i) - z_j^i = 0, \quad (14)$$

where  $\dot{g}(u)$  indicates the derivative. Let us denote with  $a_j$  the positive root of the above equation. The problem can be formulated as follows:

”Given a vector  $a \in \mathcal{R}^K$  with elements  $a_j \geq 0$  and the hyperplane  $\sum_{j=1}^K y_j = 1$ , find the point  $y$  on the hyperplane with  $y_j \geq 0$  that is closest to  $a$ ”.

This defines the following constrained convex quadratic programming (QP) problem:

$$\begin{aligned} \min_y & \frac{1}{2} \sum_{j=1}^K (y_j - a_j)^2 \\ \text{subject to} & \sum_{j=1}^K y_j = 1 \text{ and } y_j \geq 0, \forall j = 1, \dots, K. \end{aligned} \quad (15)$$

In order to solve this QP problem several approaches can be employed such as *active-set* methods and *penalty-barrier* methods [24]. For this purpose, we have implemented an active-set type of method [22] where we exploit the fact that the Hessian is the identity matrix which in turn leads to closed form expressions for the Lagrange multipliers. The detailed steps for solving this QP problem are given in the next Algorithm 1.

## References

- [1] Y. H. Yang, M. J. Buckley, S. Duboit, and T. P. Speed, “Comparison of Methods for Image Analysis on cDNA Microarray Data,” *Journal of Computational and Graphical*

---

**Algorithm 1 : A sequential convex QP algorithm**

---

Input:  $a \in \mathcal{R}^K$

Output:  $y \in \mathcal{R}^K : \min_y \frac{1}{2} \sum_{j=1}^K (y_j - a_j)^2$  s.t.  $\sum_{j=1}^K y_j = 1$  and  $y_j \geq 0 \forall j$

Set  $D = K$  and  $v_j = 1, \forall j = 1, \dots, K$

1. Calculate  $y_j \forall j = 1, \dots, K$  as :

**if**  $v_j = 1$  **then**

$$y_j = a_j + \frac{1 - \sum_{l=1}^K v_l a_l}{D}$$

**else**  $\{v_j = 0\}$

$y_j = 0$

**end if**

2. Check for termination

**if**  $y_j \geq 0 \forall j = 1, \dots, K$  **then**

STOP

**end if**

3. Update  $v_j \forall j = 1, \dots, K$  and  $D$  as:

**if**  $y_j < 0$  **then**

$v_j = 0$  and  $D = D - 1$

**end if**

4. Go to step 1.

---

*Statistics*, vol. 11, pp. 108–136, 2002.

- [2] A. A. Alizadeh, M. B. Eisen, and et. al, “Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling,” *Nature*, vol. 403, pp. 503–511, 2000.
- [3] J. Mata, R. Lyne, G. Burns, and J. Bahler, “The transcriptional program of meiosis and sporulation in fission yeast,” *Nature Genetics*, vol. 32, pp. 143–147, 2002.
- [4] K. Blekas, N. P. Galatsanos, and I. Georgiou, “An Unsupervised Artifact Correction Approach for the Analysis of DNA Microarray Images ,” in *Proc. IEEE International Conf. on Image Processing (ICIP)*, vol. 2, (Barcelona), pp. 165–168, Sep. 2003.
- [5] P. Hegde, R. Qi, K. Abernathy, and et. al, “A Concise Guide to cDNA Microarray Analysis,” *Biotechniques*, vol. 29, pp. 548–562, 2000.
- [6] M. B. Eisen, “ScanAlyze. <http://rana.lbl.gov/EisenSoftware.htm>,” 1999.
- [7] I. Axon Instruments, “GenePix Pro Documentation. <http://www.axon.com>,” 2002.

- [8] D. Bozinov and J. Rahnenfuhrer, “Unsupervised Technique for Robust Target Separation and Analysis of DNA Microarray Spots through Adaptive Pixel Clustering,” *Bioinformatics*, vol. 18, no. 5, pp. 747–756, 2002.
- [9] R. Nagarajan, “Intensity-Based Segmentation of Microarray Images,” *IEEE Trans. on Medical Imaging*, vol. 22, no. 7, pp. 882–889, 2003.
- [10] A. W.-C. Liew, H. Yang, and M. Yang, “Robust Adaptive Spot Segmentation of DNA Microarray Images,” *Pattern Recognition*, vol. 36, pp. 1251–1254, 2003.
- [11] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford Univ. Press Inc., New York, 1995.
- [12] G. M. McLachlan and D. Peel, *Finite Mixture Models*. New York: John Wiley & Sons, Inc., 2001.
- [13] P. Smyth, “Model Selection for Probabilistic Clustering using Cross-Validated Likelihood,” *Statistics and Computing*, vol. 10, pp. 63–72, 2000.
- [14] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. Roy. Statist. Soc. B*, vol. 39, pp. 1–38, 1977.
- [15] S. Sanjay-Gopal and T. J. Hebert, “Bayesian Pixel Classification Using Spatially Variant Finite Mixtures and the Generalized EM Algorithm,” *IEEE Trans. on Image Processing*, vol. 7, no. 7, pp. 1014–1028, 1998.
- [16] M. Katzer, F. Kummert, and G. Sageter, “A Markov Random Field Model of Microarray Gridding,” in *Proc. ACM Symposium on Applied Computing (SAC)*, (Melbourne, Florida), pp. 72–77, 2003.
- [17] H. S. Baird, “The Skew Angle of Printed Documents,” in *Proc. Conf. of the Society of Photographic Scientists and Engineers*, pp. 14–21, 1987.

- [18] C. Bowman, R. Baumgartner, and S. Booth, "Automated Analysis of Gene-microarray Images," in *Proc. IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pp. 1140–1144, 2002.
- [19] P. Bajcsy, "Gridline: Automatic Grid Alignment in DNA Microarray Scans," *IEEE Trans. on Image Processing*, vol. 13, no. 1, pp. 15–25, 2004.
- [20] P. J. Green, "Bayesian Reconstructions from Emission Tomography Data Using a Modified EM Algorithm," *IEEE Trans. on Medical Imaging*, vol. 9, no. 1, pp. 84–93, 1990.
- [21] Y. Zhang, M. Brady, and S. Smith, "Segmentation of Brain MR Images Through a Hidden Markov Random Field Model and the Expectation-Maximization Algorithm," *IEEE Trans. on Medical Imaging*, vol. 20, no. 1, pp. 45–57, 2001.
- [22] K. Blekas, A. Likas, N. P. Galatsanos, and I. E. Lagaris, "A Spatially-Constrained Mixture Model for Image Segmentation," *IEEE Trans. of Neural Networks (to appear)*, 2005.
- [23] Agilent Technologies, "Agilent Feature Extraction Software. <http://www.chem.agilent.com>," 2003.
- [24] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer-Verlag, New York, 1999.

|               | <i>Proposed</i> | <i>Spotfinder</i> | <i>ScanAlyze</i> |
|---------------|-----------------|-------------------|------------------|
| Perfect (%)   | 89.6            | 72.8              | 48.7             |
| Marginal (%)  | 9.2             | 14.3              | 22.6             |
| Incorrect (%) | 1.2             | 12.9              | 28.7             |

Table 1: Performance of three gridding methods using ten (10) spot arrays.

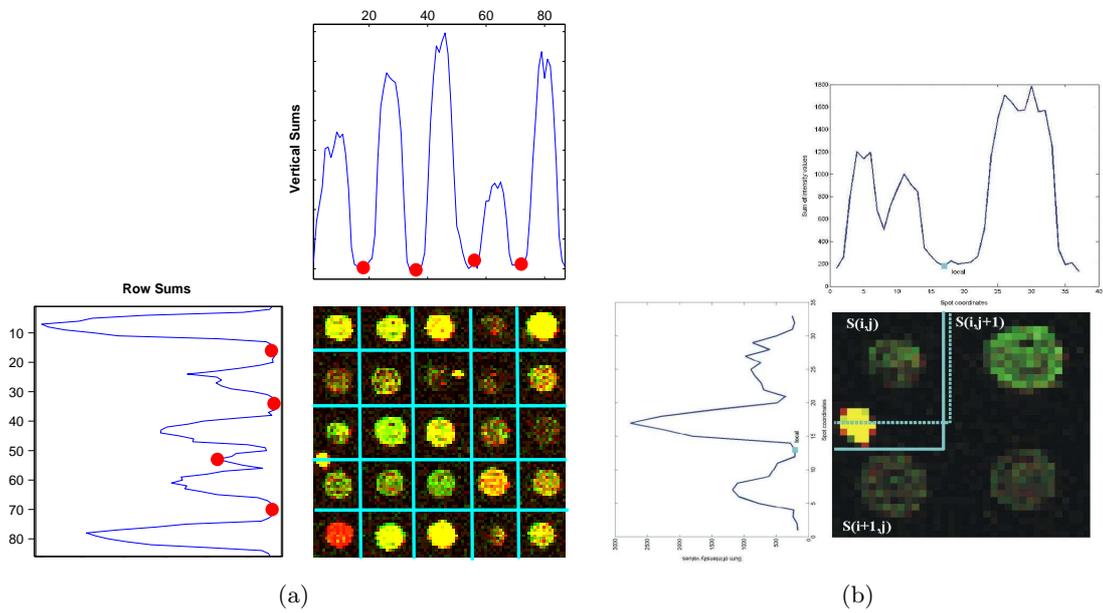
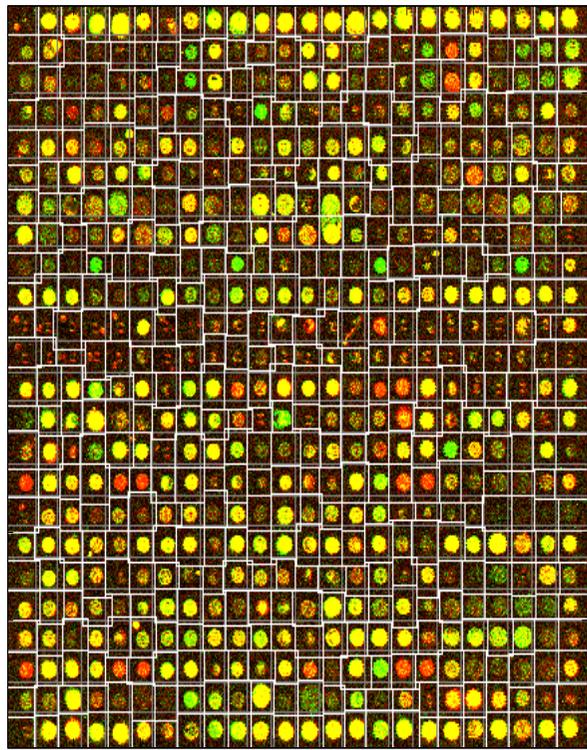
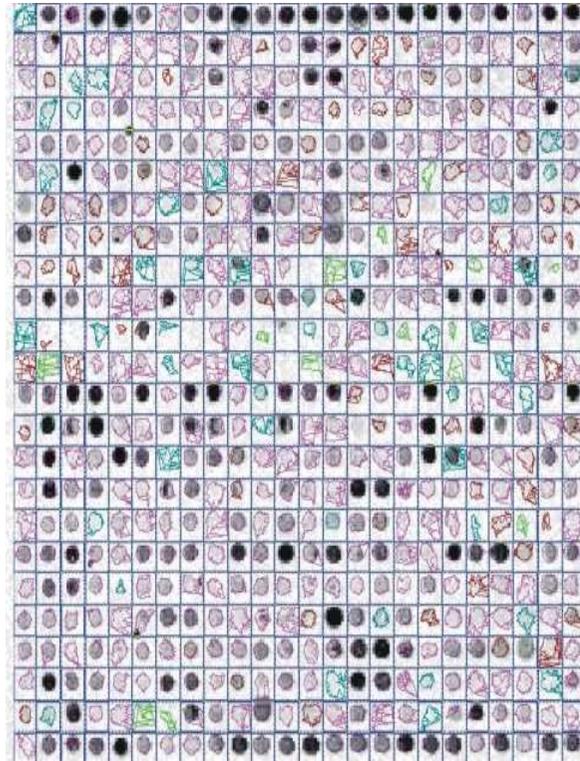


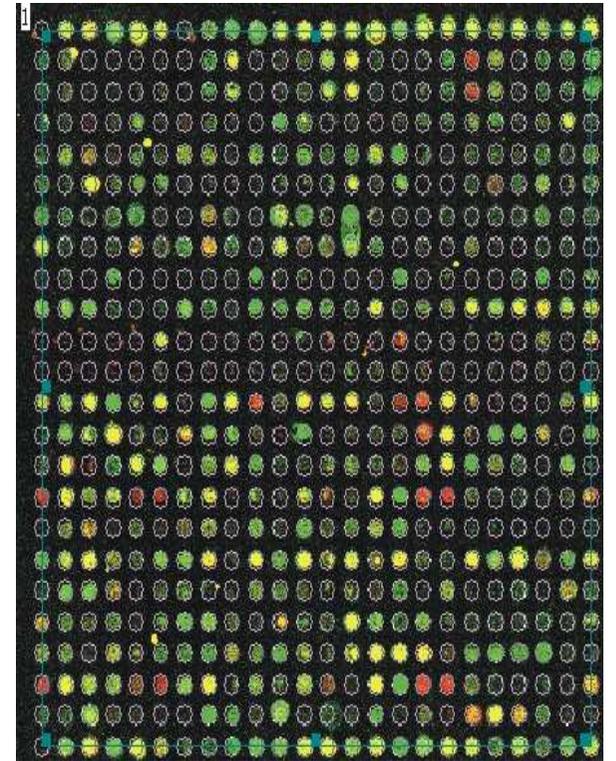
Figure 1: (a). These signals are obtained by summing up the rows and columns of both R and G channels for a  $5 \times 5$  grid structure. Mid points of successive peaks define the horizontal vertical global borders, respectively. (b). The global borders (dotted lines) are refined (solid lines) based on the local sums. The signals on the left and above the microarray image are the local row and column sums, respectively.



(a)



(b)



(c)

Figure 2: Comparative gridding results of our method (a) with two widely used microarray image analysis tools: (b) the Spotfinder and (c) the ScanAlyze.

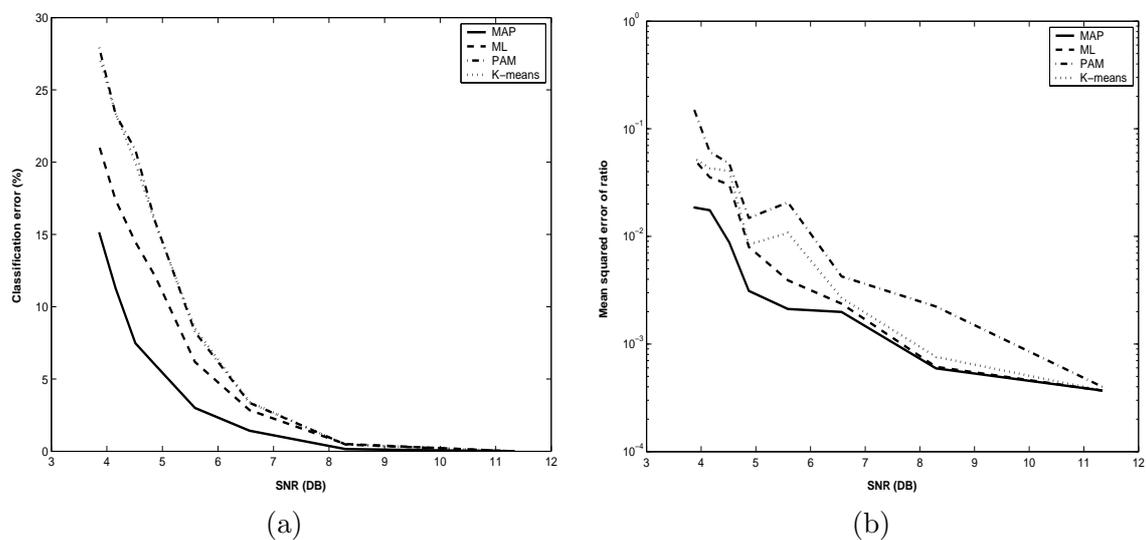


Figure 3: (a) classification error and (b) mean squared error of ratio versus SNR using artificial spot images.

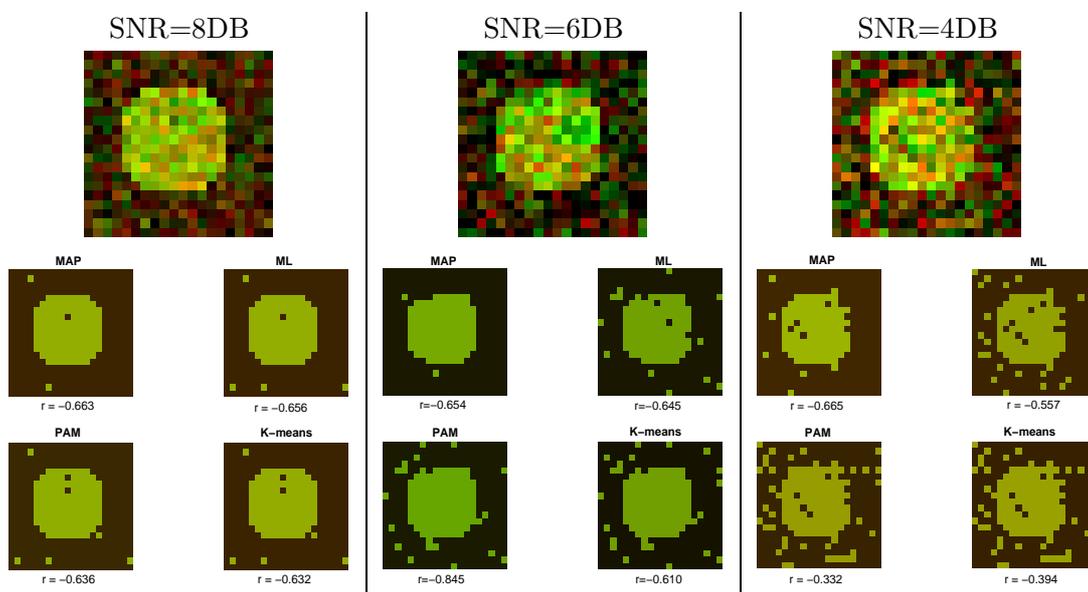


Figure 4: Segmentation maps and fluorescent ratios at different SNRs using three artificial spot images

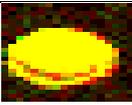
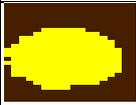
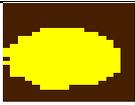
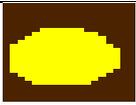
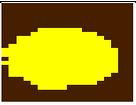
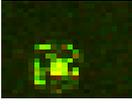
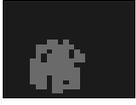
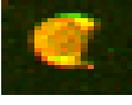
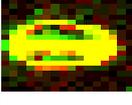
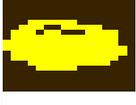
| <i>Original image</i>  | <i>MAP-GMM</i>  |   |   | <i>ML-GMM</i>   | <i>K-means</i>   | <i>PAM</i>  | <i>Existing tools</i>                |
|--|---|---|---|---|--|---|--------------------------------------|
|  | $\beta = 0.01$  | $\beta = 0.1$   | $\beta = 1.0$   |   |  |   |                                      |
| <br>$S_1$   | <br>$r = 0.293$    | <br>$r = 0.294$    | <br>$r = 0.296$    | <br>$r = 0.293$    | <br>$r = 0.212$    | <br>$r = 0.323$    | GenPix: 0.333<br>Spotfinder: -0.213  |
| <br>$S_2$   | <br>$r = -0.922$   | <br>$r = -0.890$   | <br>$r = -0.888$   | <br>$r = -0.892$   | <br>$r = -1.120$   | <br>$r = -1.183$   | GenPix: -0.673<br>Spotfinder: -0.871 |
| <br>$S_3$ | <br>$r = 0.808$  | <br>$r = 0.807$  | <br>$r = 0.801$  | <br>$r = 0.808$  | <br>$r = 0.839$  | <br>$r = 0.889$  | GenPix: 0.875<br>Spotfinder: 0.775   |
| <br>$S_4$ | <br>$r = -0.316$ | <br>$r = -0.318$ | <br>$r = -0.266$ | <br>$r = -0.318$ | <br>$r = -0.289$ | <br>$r = -0.053$ | GenPix: -0.360<br>Spotfinder: -0.136 |
| <br>$S_5$ | <br>$r = 1.507$  | <br>$r = 1.528$  | <br>$r = 1.533$  | <br>$r = 1.527$  | <br>$r = 1.495$  | <br>$r = 1.819$  | GenPix: 2.795<br>Spotfinder: 1.474   |

Figure 5: Comparative results for 5 real microarray spots without artifacts. For each method we give the segmentation map and the estimated fluorescence ratio.

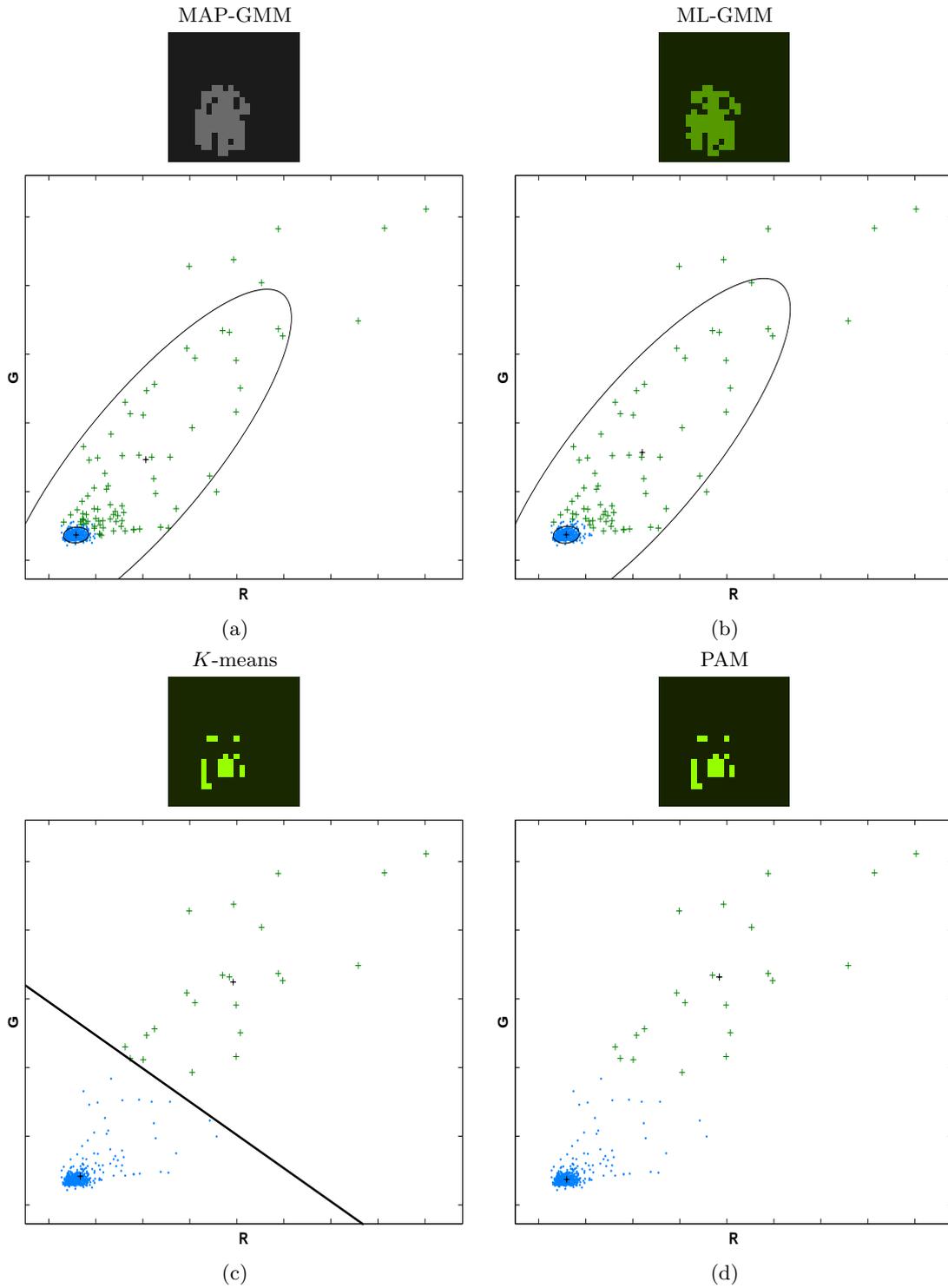


Figure 6: Plot of all pixel values of spot  $S_2$  of Fig. 5 after labeling them with MAP-GMM (a), ML-GMM (b),  $K$ -means (c) and PAM methods (d), respectively. The ellipsoidal clusters resulting from the GMM approaches and the linear boundary between the two clusters in the  $K$ -means case are also shown.

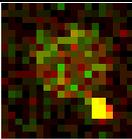
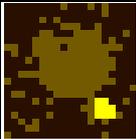
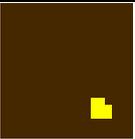
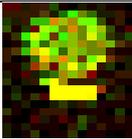
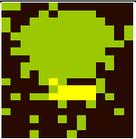
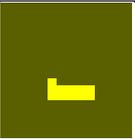
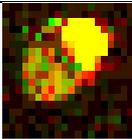
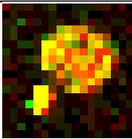
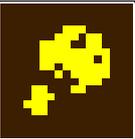
| Original image   | MAP-GMM  |  |  | ML-GMM   | K-means   | PAM   | Existing tools                       |
|--|--|--|--|--|---|---|--------------------------------------|
|  | $\beta = 0.01$   | $\beta = 0.1$  | $\beta = 1.0$  |  |   |   |                                      |
| <br>$S_1$   | <br>$r = 0.173$   | <br>$r = 0.207$   | <br>$r = 0.297$   | <br>$r = 0.175$   | <br>$r = 0.874$   | <br>$r = 0.644$    | GenPix: 0.498<br>Spotfinder: 0.992   |
| <br>$S_2$  | <br>$r = -0.633$ | <br>$r = -0.619$ | <br>$r = -0.686$ | <br>$r = -0.699$ | <br>$r = 0.435$  | <br>$r = 0.286$   | GenPix: -0.732<br>Spotfinder: -0.598 |
| <br>$S_3$ | <br>$r = 0.567$ | <br>$r = 0.464$ | <br>$r = 0.431$ | <br>$r = 0.442$ | <br>$r = 0.025$ | <br>$r = -0.053$ | GenPix: 0.500<br>Spotfinder: 0.423   |
| <br>$S_4$ | <br>$r = 0.648$ | <br>$r = 0.650$ | <br>$r = 0.690$ | <br>$r = 0.647$ | <br>$r = 0.573$ | <br>$r = 0.507$  | GenPix: 0.834<br>Spotfinder: 0.588   |

Figure 7: Comparative results for 4 real microarray spots with artifacts. For each method we give the segmentation map and the estimated fluorescence ratio.

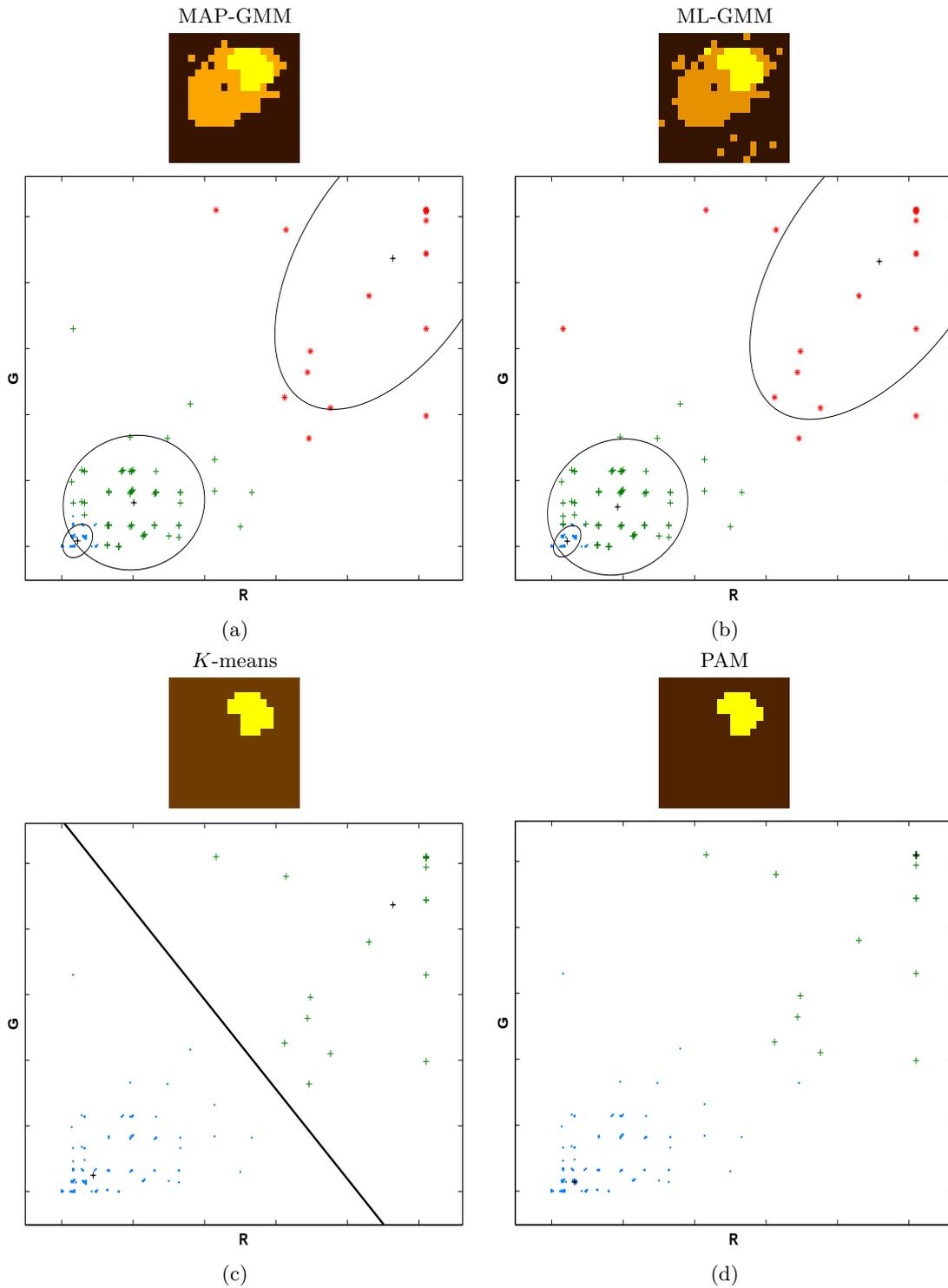


Figure 8: Plot of pixel values in spot  $S_3$  of Fig. 7 after labeling with MAP-GMM (a), ML-GMM (b),  $K$ -means (c) and PAM methods (d), respectively. The ellipsoidal clusters resulting from the GMM approaches and the linear boundary between the two clusters in the  $K$ -means case are also shown.

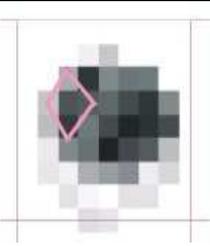
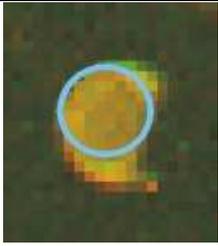
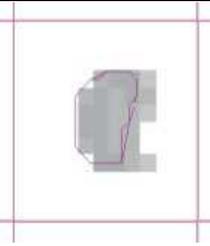
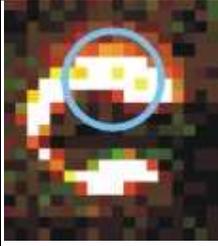
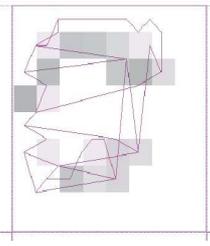
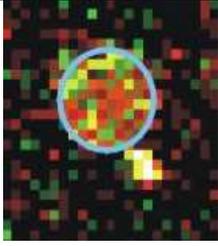
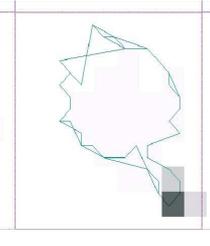
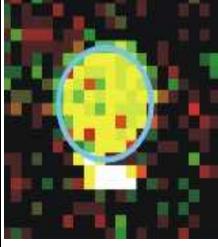
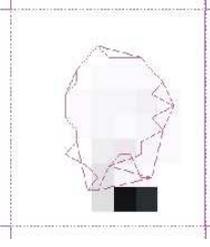
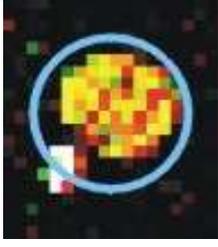
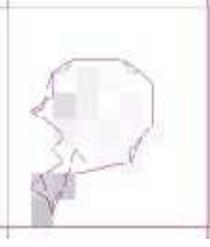
| <i>Original image</i> | <i>GenePix</i>  | <i>Spotfinder</i>   | <i>Original image</i> | <i>GenePix</i>  | <i>Spotfinder</i>  |
|-----------------------|---|---|-----------------------|---|--|
| $S_1$<br>(Fig. 5)     | <br>$r = 0.333$    | <br>$r = -0.213$   | $S_3$<br>(Fig. 5)     | <br>$r = 0.875$   | <br>$r = 0.775$   |
| $S_5$<br>(Fig. 5)     | <br>$r = 2.795$   | <br>$r = 1.474$   | $S_1$<br>(Fig. 7)     | <br>$r = 0.498$  | <br>$r = 0.992$  |
| $S_2$<br>(Fig. 7)     | <br>$r = -0.732$ | <br>$r = -0.598$ | $S_4$<br>(Fig. 7)     | <br>$r = 0.834$ | <br>$r = 0.588$ |

Figure 9: Calculated fluorescent ratios for 6 spot examples using the GenePix and the Spotfinder microarray image tools.

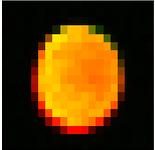
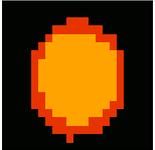
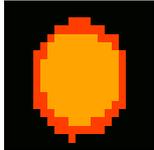
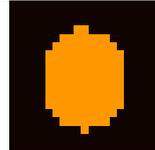
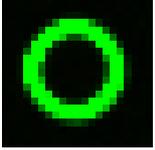
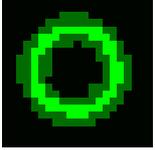
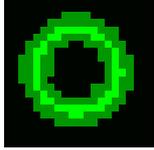
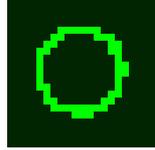
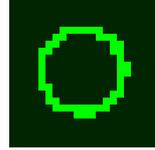
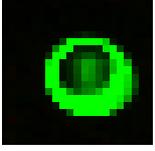
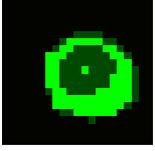
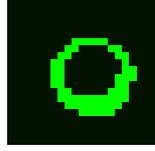
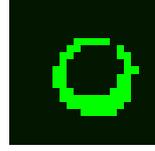
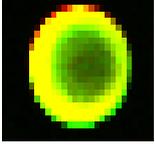
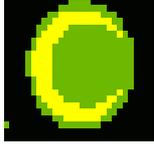
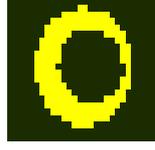
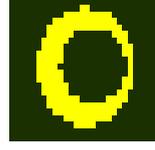
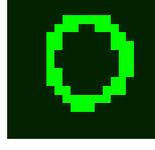
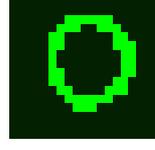
| Original image  | MAP-GMM   | ML-GMM  | <i>K</i> -means   | PAM  |
|---|---|---|---|--|
|    | <br>$r = 2.270$    | <br>$r = 2.211$    | <br>$r = 2.104$     | <br>$r = 2.071$     |
|    | <br>$r = -13.153$  | <br>$r = -16.397$  | <br>$r = -11.779$   | <br>$r = -9.223$    |
|  | <br>$r = -7.034$ | <br>$r = -6.751$ | <br>$r = -8.924$  | <br>$r = -10.064$ |
|  | <br>$r = -0.905$ | <br>$r = -0.744$ | <br>$r = -0.777$  | <br>$r = -0.744$  |
|  | <br>$r = -3.825$ | <br>$r = -3.833$ | <br>$r = -11.974$ | <br>$r = -7.950$  |

Figure 10: Five examples of Agilent Technologies images. The segmentation result together with the calculated ratio value are provided for each clustering method.