BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 12 2005, pages 2875–2882
doi:10.1093/bioinformatics/bti447

Gene expression

Donuts, scratches and blanks: robust model-based segmentation of microarray images

Qunhua Li[1], Chris Fraley[1,*], Roger E. Bumgarner[2], Ka Yee Yeung[2] and Adrian E. Raftery[1]

[1]Department of Statistics, Box 354322 and [2]Department of Microbiology, Box 357242, University of Washington, Seattle, WA 98195, USA

Received on January 18, 2005; revised on March 29, 2005; accepted on April 9, 2005
Advance Access publication April 21, 2005

ABSTRACT

Motivation: Inner holes, artifacts and blank spots are common in microarray images, but current image analysis methods do not pay them enough attention. We propose a new robust model-based method for processing microarray images so as to estimate foreground and background intensities. The method starts with a very simple but effective automatic gridding method, and then proceeds in two steps. The first step applies model-based clustering to the distribution of pixel intensities, using the Bayesian Information Criterion (BIC) to choose the number of groups up to a maximum of three. The second step is spatial, finding the large spatially connected components in each cluster of pixels. The method thus combines the strengths of the histogram-based and spatial approaches. It deals effectively with inner holes in spots and with artifacts. It also provides a formal inferential basis for deciding when the spot is blank, namely when the BIC favors one group over two or three.

Results: We apply our methods for gridding and segmentation to cDNA microarray images from an HIV infection experiment. In these experiments, our method had better stability across replicates than a fixed-circle segmentation method or the seeded region growing method in the SPOT software, without introducing noticeable bias when estimating the intensities of differentially expressed genes.

Availability: spotSegmentation, an R language package implementing both the gridding and segmentation methods is available through the Bioconductor project (http://www.bioconductor.org). The segmentation method requires the contributed R package MCLUST for model-based clustering (http://cran.us.r-project.org).

Contact: fraley@stat.washington.edu

1 INTRODUCTION

Microarray technology is now a widely used tool in a number of large-scale assays including RNA expression (e.g. DeRisi *et al*., 1997; Lipshutz *et al*., 1999) and DNA content analyses (e.g. Pinkel *et al*., 1998; Snijders *et al*., 2003; Pollack *et al*., 1999). While a large number of array platforms exist, a common method for making DNA arrays consists of printing the single-stranded DNA representing the genes of interest on a solid substrate using a robotic spotting device. For gene expression analysis, RNA is extracted from the samples of interest, converted to labeled cDNA and is then hybridized with the arrayed DNA spots. In other applications (e.g. CGH), DNA is labeled and hybridized to the array. In most instantiations of the technology, the hybridization is detected as fluorescent signals from dyes that are either directly incorporated in the labeling steps or are added in a post-staining step through specific binding to some other labeling moiety (e.g. biotin). Fluorescence measurements are then obtained via a scanner or imager. Subsequent image analysis quantifies the fluorescence per spot, which in turn, is related to the relative abundance of the mRNA or DNA in the samples.

A key component of this process is the effectiveness of the image analysis. In this step, the quantification of the amount of fluorescence from the hybridized sample can be affected by a variety of defects that occur during both the manufacturing and processing of the arrays, such as perturbations of spot positions, irregular spot shapes, holes in spots, unequal distribution of the DNA probe within spots, variable background, and artifacts such as dust and precipitates. Ideally these events should be automatically recognized in the image analysis, and the estimated intensities adjusted to take account of them.

Several commercial and research image processing packages have been developed for analyzing microarray data. For segmentation (separating foreground 'signal' or 'feature' from background), the existing methods can be grouped into four categories, namely fixed circle segmentation, adaptive circle segmentation, adaptive shape segmentation and histogram segmentation, as reviewed by Yang *et al*. (2002). Fixed circle segmentation assumes that the spots have a circular shape and fits a circle with a fixed radius to all the spots. It was probably first implemented in ScanAlyze (Eisen, 1999). Spot-on, a customized software written at the University of Washington (Spot-On Image, developed by R.E.Bumgarner and E.Hammersmark), also implements this algorithm. Adaptive circle segmentation improves the method by allowing the radius of the circle to be adjustable. However, a circular spot mask provides a poor fit to irregular spots or donut-shaped spots with inner holes, which are often seen in microarray images.

For segmentation, QuantArray (GSI Luminomics, 1999) applies a threshold to the histogram of pixel values in a target region around a spot, ScanAlyze (Eisen, 1999) uses a circle of fixed radius, GenePix (Axon Instruments Inc., 1999) uses a circle with adaptive radius, UCSF Spot (Jain *et al*., 2002) uses histogram information within a circle and TIGR Spotfinder (TIGR, 2004) is also based on a histogram (http://www.tm4.org/spotfinder.html). Liew *et al*. (2003) use an adaptive circle method, while Bergemann *et al*. (2004) generalize

*To whom correspondence should be addressed.

this by using an adaptive ellipse. They flag spots with inner holes, but the user has to decide what to do with them when estimating intensities and in subsequent data processing. Schadt *et al*. (1999) proposed an adaptive pixel selection algorithm to remove pixels contaminated by noise. Kim *et al*. (2001) used an edge detection method. They were aware of the problem of inner holes and used a threshold of intensity to decide the eligibility of pixels as foreground. Hirata *et al*. (2002) and Angulo and Serra (2003) used mathematical morphology. Their methods can deal with blank spots, but not with spots with inner holes. Glasbey and Ghazal (2003) used a combinatorial way to consider a variety of methods, including fixed circle, proportions of histogram, *k*-means clustering with different preprocessing and different parameters. O'Neill *et al*. (2003) recreate the background slide and subtract it. Their method deals effectively with global artifacts that involve a substantial number of spots but not with inner holes or local artifacts. Steinfath *et al*. (2001) fitted a scaled bivariate Gaussian distribution to pixel values, but using a robust fitting method. Brändle *et al*. (2003) described a robust fitting for the Gaussian spot model using an M-estimator. In this method, each spot is fitted by a single Gaussian whose parameters are estimated using robust fitting techniques that help prevent outliers from distorting the fit. Our method fits the spot with a mixture of up to three Gaussians, in which the components of the mixture identify background, foreground and possible artifact.

Several recent developments belong to the class of adaptive shape segmentation. The seeded region growing approach (Adams and Bischof, 1994) is used to segment microarray images in the SPOT software (Yang *et al*., 2002). The foreground and background are grown from two initial seeds; this method can adapt to various shapes of spots. Histogram methods are intensity-based, and use a target mask that is chosen to be larger than all spots. The pixels are classified as foreground or background using thresholds from the histogram of pixel values within the masked area. Histogram methods do not use any spatial information, and so the resulting spot masks are not necessarily connected. Ahmed *et al*. (2004) provide evidence that, although histogram methods do not take spatial aspects into account, they yield better intensity estimates than other methods.

Many current methods have difficulties with donut-shaped spots, artifacts such as scratches and blank spots, all of which are common in typical microarray images. When the spot is donut-shaped, many current methods identify the outer contour of the spot as the mask; this can lead to downward bias in the estimated intensity. Another common problem is that a foreground is always generated even when no spot is present. This tends to inflate the variance of the estimates. In addition, many image analysis programs provide a number of *ad hoc* methods for identifying the presence of artifacts (such as shape and size thresholds or an unusual number of 'outlier' pixels). In the majority of image analysis programs, spots with artifacts are simply 'flagged' for downstream analyses and may or may not be ignored in subsequent processing.

In this paper, we propose an approach to image segmentation and intensity estimation combining three simple steps: automatic grid finding, model-based clustering of pixel intensities and spatial connected-component extraction. We start by using a very simple automatic gridding method. We then apply model-based clustering to the pixel intensities, which allows us to estimate the number of groups in the target area and hence provides a formal basis for determining whether or not a spot is present, namely when the number of groups estimated is equal to one. Our final 'spot' or foreground estimate consists of the large connected components of the foreground cluster of pixels. An R software package called spotSegmentation implementing the method has been made available as part of the Bioconductor project (http://www.bioconductor.org). In experiments we found the results to be more stable than those from either the fixed-circle method or the region seeded growing method, without introducing substantial biases. We performed our experiments on two-color arrays, but the method is generic and can be extended to other types of array.

By itself, model-based clustering of pixels is a histogram-based method. Thus our method combines the strengths of the histogram method documented by Ahmed *et al*. (2004) with a simple spatial step that improves the spatial coherence of the resulting estimated spots and eliminates small artifacts. It handles donut-shaped spots well because the estimated spots can easily be of this form. It deals effectively with artifacts because they are often small connected components, or else they get classified as a separate group of pixels and are not included in the foreground or background intensity estimates. Perhaps most importantly, it deals explicitly with blanks (locations on the slide where there is no spot); this is something that few other current methods do. In general other methods 'recognize' the lack of a spot via the use of a threshold for overall intensity or signal-to-noise.

The term 'model-based segmentation' has been used to describe methods based on the assumption that the areas of interest follow a parametric form (e.g. Bergemann *et al*., 2004). Fixed-circle and adaptive circle methods are of this type. However, our method does not make this rather restrictive assumption. Instead, our method is called 'model-based' because it is based on model-based clustering of the pixel intensities. In this case, the model simply assumes that pixels belonging to the same feature (be it foreground, background or artifact) will cluster into groups by intensity. Hence, it is very flexible in terms of the shape of the spot that it can accurately recover.

Recently, others have also used clustering methods for image processing of microarray data (Bozinov and Rahnenführer, 2002; Nagarajan, 2003; Glasbey and Ghazal, 2003). Bozinov and Rahnenführer (2002) used *k*-means and Partitioning Around Medoids (PAMs) on the two-dimensional vectors of the intensities, and Rahnenfürer and Bozinov (2004) improve on this by considering only the pixels within the average spot shape, which turns out to be almost exactly a circle. Nagarajan (2003) used the same method, but only on the intensities from the green channel. Glasbey and Ghazal (2003) considered a Gaussian mixture model for the two-dimensional vector of the square root of the intensities. All of these methods consider only two clusters. As Bozinov and Rahnenführer (2002) pointed out and we also observed, fixing the number of clusters to two can mislead the clustering algorithm into taking the large brighter artifacts as foreground and combining the dimmer spot pixels into the background. It also excludes the ability to formally identify blank spots provided in our method by the data-based choice of the number of clusters. Antonio *et al*. (2004) used a clustering method that does not constrain the number of clusters, but in the experiments reported it did not exclude the inner holes of donut-shaped spots.

Automatic gridding is necessary before applying our method, and any automatic gridding method could be used in combination with our segmentation approach. We have developed a very simple technique for gridding, which is included in the spotSegmentation software. More complex automatic gridding methods have been proposed by Jung and Cho (2002) who use nearest-neighbor graph methods, by Galinsky (2003) who use Voronoi diagrams, and by

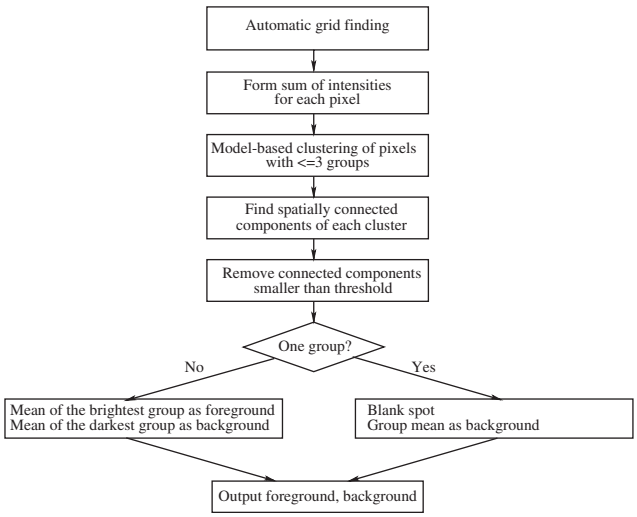

Fig. 1. Outline of model-based segmentation.

Katzer *et al*. (2003) who use a Markov random field approach, as well as by authors of several of the more comprehensive segmentation methods mentioned above. Our automatic gridding method is much simpler, and we have found it to be effective.

In Section 2, we describe our image segmentation method, including automatic gridding, model-based clustering of pixels, spatial connected-component extraction and final estimation of foreground and background intensities. In Section 3, we give the results of applying our method to microarray images from an HIV infection experiment. The results are compared with those from fixed-circle segmentation as implemented in Spot-on and seeded region growing as implemented in SPOT. In Section 4, we describe the R software package, spotSegmentation, implementing the methods.

2 METHODS

Our method consists of several simple steps: automatic gridding, model-based clustering of pixels and spatial connected component extraction. Figure 1 gives an outline of the whole procedure in flowchart form. We now describe each of the steps in turn.

2.1 Automatic gridding

In order to segment the image, we must first identify the location of each spot. This process is called gridding or addressing. A microarray typically consists of several blocks with the same layout. The print-tips on the arrayer are normally arranged in a regular array. Under perfect conditions, the spots in each block locate in an evenly spaced lattice corresponding to the layout of the print-tips. However, the variation during the printing of the array will cause the exact locations of the spots to vary from the prespecified parameter. Even if the irregularities are slight, they can result in significant irregularity in the image and hence have to be corrected.

In order to locate the spots, we do not need to find their centers, but rather the edges of the target mask, i.e. the rectangle containing the spot. As long as the rectangle contains only the pixels from a single spot, it is a valid target mask.

Our algorithm is as follows:

- Sum up the intensities across the pixels in each row and each column.
- Find the local minima of the summed intensities using a sliding window with span approximately equal to the width of a typical spot.

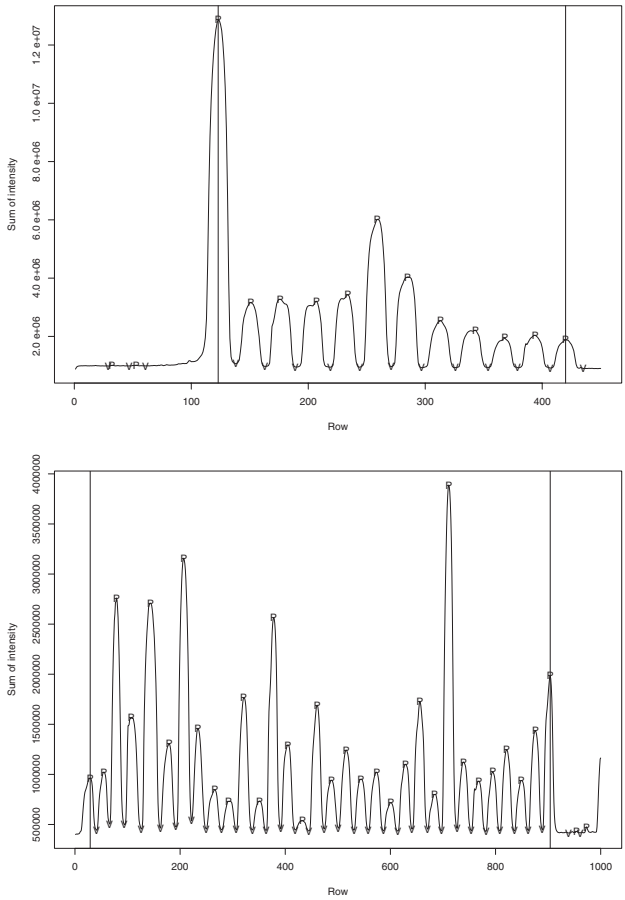

Fig. 2. Row (top) and column sums of intensity and grid on a 12×32 subarray. Because the spots are located loosely on a rectangular grid, the row (column) sums present a peak-valley pattern, with peaks corresponding to the average center of spots on the row (column) and valleys the delimiters of spots. Grids are placed at the valleys of the curves.

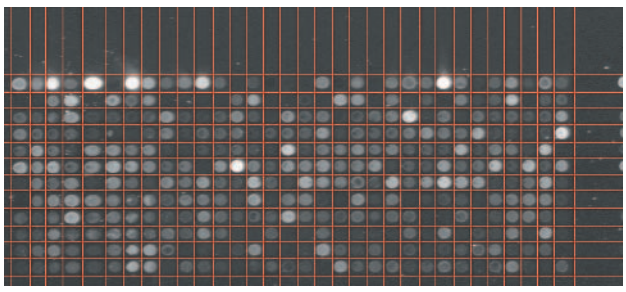

Fig. 3. Array image with the grid boundaries estimated by our method overlaid.

This method is extremely simple and does not require human interaction. The only control parameters to be specified are the number of spots in each row or column (as specified by the array manufacturer), and the size of the sliding window. A crude estimate using the known number of rows and columns suffices for the window size in the arrays we have used.

Figures 2 and 3 show the results of applying the method to a 12×32 subarray from the HIV experiment dataset, which we will describe later. Figure 2 shows the summed intensities; the valleys correspond to the grid

lines. Figure 3 shows the resulting grids; this captures the locations of the spots well.

2.2 Model-based clustering of pixels

The gene expression level is proportional to the pixel intensities of a spot. Thus pixels that are in the spot or foreground should have similar intensities, and pixels that are in the background should also have similar intensities. In addition, pixels that belong to an artifact such as a scratch that is neither spot nor background will tend to have intensities that are different from either. As a result, clustering the pixel intensities makes sense as an approach to segmentation; this is the idea underlying histogram-based methods. Here we apply model-based clustering to the pixel intensities.

In model-based clustering, data x are viewed as coming from a mixture density $f(x) = \sum_{k=1}^{K} p_k f_k(x)$. Here, p_k are the mixing proportions ($0 < p_k < 1$ for all $k = 1, \ldots, K$ and $\sum_k p_k = 1$), and f_k is the probability density function of the observations in group k. In the Gaussian mixture model, each component k is modeled by the multivariate normal distribution with mean μ_k and covariance matrix Σ_k, which has the probability density function

$$\phi(x_i; \mu_k, \Sigma_k) = \frac{\exp\{-\frac{1}{2}(x_i - \mu_k)^{\mathrm{T}} \Sigma_k^{-1}(x_i - \mu_k)\}}{\sqrt{\det(2\pi\,\Sigma_k)}}. \quad (1)$$

The likelihood for data consisting of n observations assuming a Gaussian mixture model with K mixture components is

$$L_{\mathrm{mix}} = \prod_{i=1}^{n} \sum_{k=1}^{K} p_k \phi(x_i; \mu_k, \Sigma_k). \quad (2)$$

For reviews of model-based clustering, see McLachlan and Peel (2000) and Fraley and Raftery (2002).

For a fixed number of clusters K, the model parameters p_k, μ_k and Σ_k can be estimated using the EM algorithm initialized by hierarchical model-based clustering (Dasgupta and Raftery, 1998; Fraley and Raftery, 1998). The number of groups, K, can be estimated by maximizing the Bayesian Information Criterion (BIC). Model-based clustering is implemented in the MCLUST software (Fraley and Raftery, 1999, 2003), which is available at http://www.stat.washington.edu/mclust or http://cran.us.r-project.org

To combine the signals from the two channels, the red and green intensities are summed. Inspection of the resulting histograms, such as that in Figure 4, suggest that it is reasonable to assume that the distribution of the summed intensities is approximately a mixture of Gaussian densities.

We have some prior information about the number of groups of pixels present in the images. Typically, background pixels would be one group and pixels in the spot or foreground would be another. In addition, if an artifact is present, or if the spot is donut-shaped and has an inner hole, the corresponding pixels would form a third group. Thus in most cases we would expect the number of groups, K, to be at most three. We use BIC to choose K, but restrict the possible choices to $K \leq 3$.

We thus have three cases: $K = 1$, $K = 2$ and $K = 3$. The case $K = 1$ corresponds to the situation where there is no spot, i.e. a blank, and our method provides a principled statistical basis for detecting this situation. The case $K = 2$ would arise in the typical situation where there is a spot, with background. And $K = 3$ would be chosen when there is a spot and an artifact or an inner hole.

Note that $K = 1$ may arise from the signal being too low to detect above the noise (e.g. Velculescu *et al*., 1997). It is possible that combining across replicates may make such low signals easier to detect (Townsend, 2004).

2.3 Spatial connected component extraction and intensity estimation

Artifacts often take the form of small disconnected groups, and so a threshold on the size of the connected components in a cluster can identify clusters formed by artifacts in many cases. We apply a four-neighbor connected component labeling procedure (Haralick and Shapiro, 1992) to the clusters to divide them into spatially connected components. We retain only the connected components that meet a given threshold in size and discard the other components. The default threshold we use is 100 pixels, which is about one-sixth of the typical size of a spot on the arrays used in our examples.

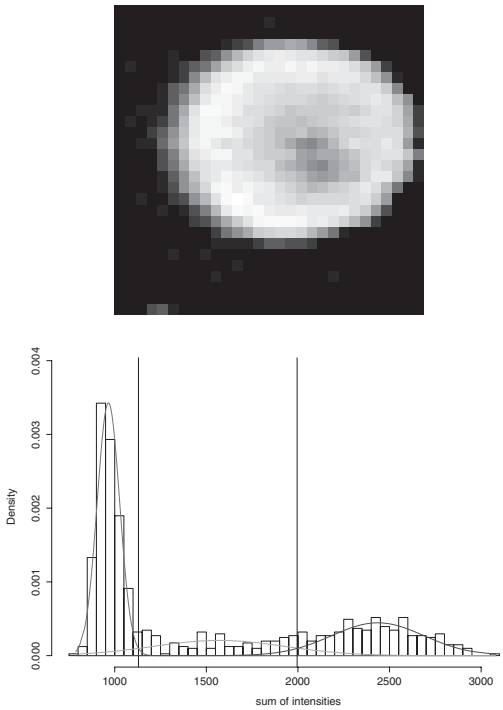

Fig. 4. A rectangle from the grid containing a typical spot (top) and the histogram of spot intensities within it. This histogram was fit by a mixture of three normal densities, which are overlaid.

The brightest and darkest clusters passing the threshold are classified as foreground and background, respectively. If only one cluster passes the threshold, we conclude that there is no spot and that the location is blank. Our estimate of foreground intensity in the Cy3 channel is the mean of the pixels in the foreground cluster. This is similar to the Cy5 channel foreground, where the same pixels are in the cluster for both channels. The background intensities of the two channels are estimated in the same way.

In this way, we leave out the disconnected pixels for intensity estimation. In addition, when three clusters are identified, we also exclude the intermediate cluster of pixels, which often consists of 'suspicious' pixels, such as an inner hole, an artifact or a blurry rim.

The estimated signal is $I^s = I^f - I^b$, where I^f and I^b are the mean intensities of the foreground and background, respectively. The true signal is always non-negative, but occasionally the estimated signal, I^s, is negative. In this case, it is reasonable to assume that the true intensity is small but positive. When this happens, we set I^s to be equal to the 5th percentile of the spot signals on the array.

3 RESULTS

We applied our proposed method to several microarrays which had been produced to identify the genes differentially expressed in HIV-infected cells. The expression levels of 4068 cellular RNA transcripts were assessed in CD4-T-cell lines at 24 h after infection with HIV virus type 1. Here we consider four replicate subarrays, each consisting of $12 \times 32 = 384$ genes, including three HIV-1 genes used as positive controls. All the four replicates shared the same DNA samples. Two of the replicates were from a dye-swap experiment in which the

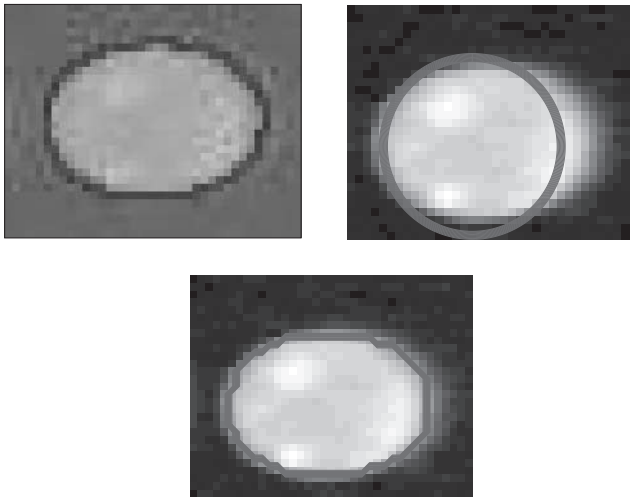

Fig. 5. Segmentation results for a well-formed spot. Top left: SPOT; top right: spot-on; and bottom: model-based segmentation.

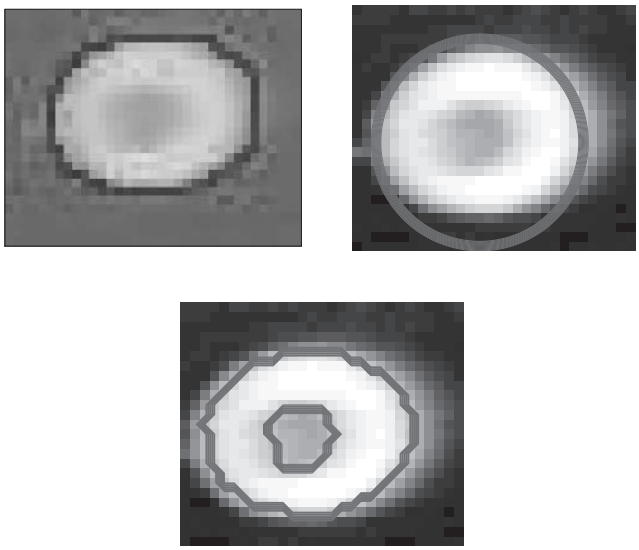

Fig. 6. Segmentation results for a donut-shaped spot. Top left: SPOT; top right: spot-on; and bottom: model-based segmentation.

dyes were switched between the two channels; this can be helpful for canceling out the dye-binding effects. Further details can be found in the original paper (van't Wout *et al*., 2003). The image files can be found at http://expression.microslu.washington.edu/expression/vantwoutjvi2002.html. In these microarrays, a large number of spots have donut shapes with one or more holes in them. We compare our method with two other methods representative of the range of methods available: the well-known software package SPOT, which segments using seeded region growing, and a customized software written at the University of Washington (Spot-On Image, developed by R.E.Bumgarner and E.Hammersmark), which implements fixed circle segmentation and estimates background using four smaller circles in the corners of the rectangle.

Figures 5–10 show the results for different individual spots. Figure 5 shows an ideal case with a single regularly shaped spot

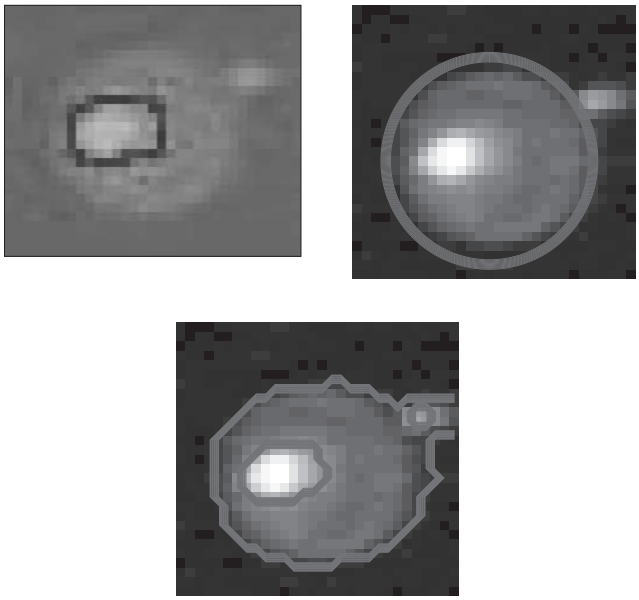

Fig. 7. Segmentation results for another donut-shaped spot. Top left: SPOT; top right: spot-on; and bottom: model-based segmentation.

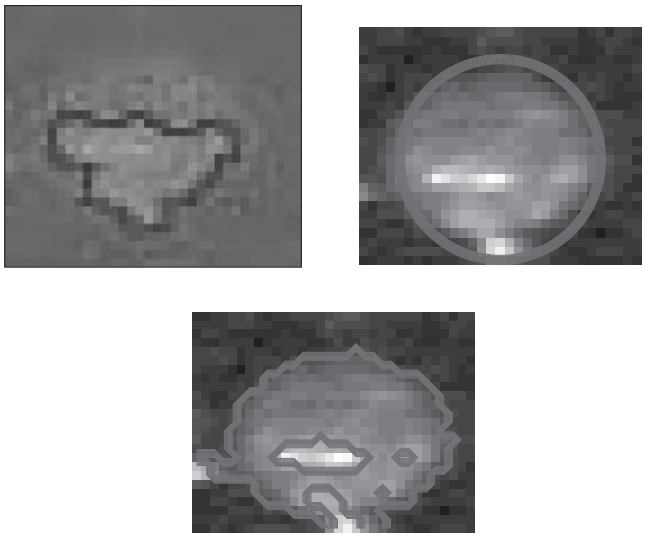

Fig. 8. Segmentation results for an image with spot and artifacts. Top left: SPOT; top right: spot-on; and bottom: model-based segmentation.

and no artifacts. There SPOT and our method both perform well, but the fixed-circle method of Spot-on is inaccurate, missing some of the spot and including some of the background in it.

Figure 6 is an example of a donut-shaped spot. The fixed-circle method again misrepresents the shape of the spot. The seeded region growing method of SPOT takes the spot to be all the pixels inside the outer contour of the donut shape. This includes the inner hole, which is darker than the spot, and so may lead to intensity estimates that are biased downwards. Our method correctly identifies the donut shape. The inner hole is identified as a third cluster, and the pixels in the inner hole are not included in the calculation of either foreground or background intensity.

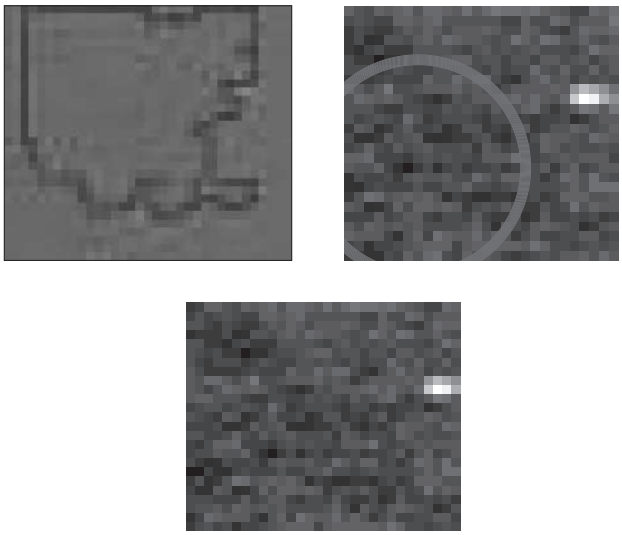

Fig. 9. Segmentation results for an image with no spot and an artifact. Top left: SPOT; top right: spot-on; and bottom: model-based segmentation. Model-based segmentation correctly detects the absence of a spot.

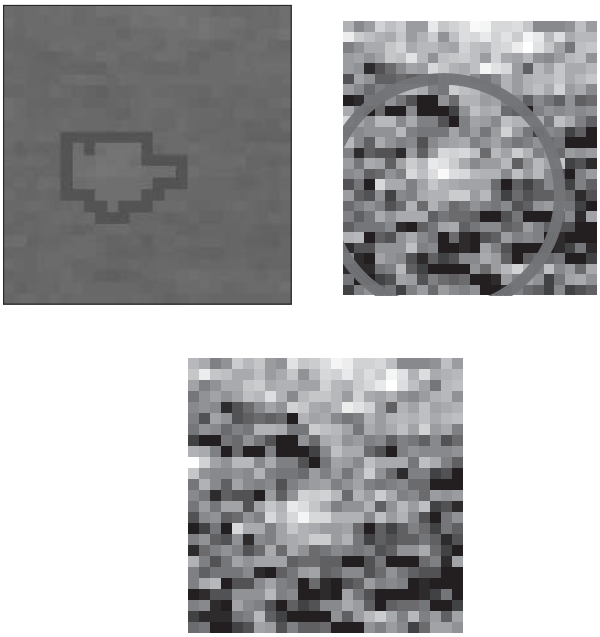

Fig. 10. Segmentation results for an image with no spot (blank). Top left: SPOT; top right: spot-on; and bottom: model-based segmentation.

Figure 7 shows a different kind of donut shape, with a small inner hole that is brighter than the spot. The fixed-circle method again does not perform well. SPOT identifies the small inner hole as the spot. Our method, in contrast, identifies the donut shape of the spot. The inner hole is brighter than the main body of the spot, and it is identified as a third cluster, but it is not identified as the foreground because it is too small to pass the threshold of 100 pixels.

Figure 8 includes several small artifacts, one or two inside the spot, one at the bottom and perhaps one on the left. The fixed-circle segmentation includes most of the artifacts in the spot. SPOT includes the inner artifacts in the spot, and misses part of the spot. Our method finds the shape of the spot correctly and excludes the artifacts.

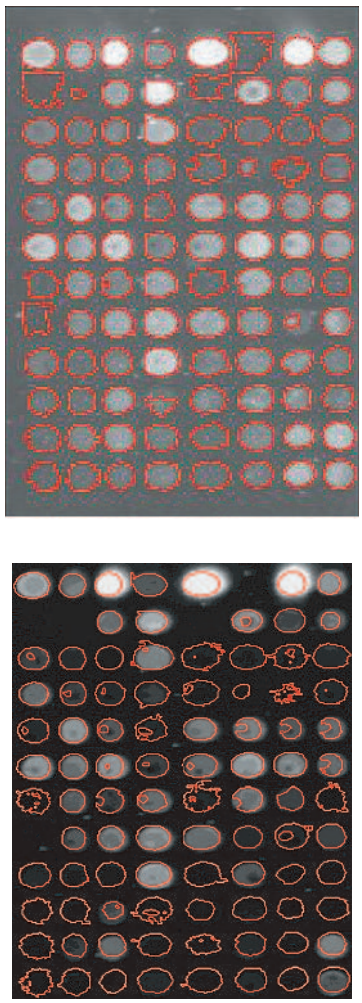

Fig. 11. Segmentation results for a 12 × 8 subset of the array. Upper panel, SPOT; and lower panel, model-based segmentation.

Figure 9 shows a blank spot with a small artifact. The fixed-circle method finds a spot anyway. SPOT finds an oddly shaped area that does not correspond to any real spot. Our method correctly infers that there is no spot in this rectangle. In fact, BIC chose $K = 2$ in this case, with the second cluster being the small artifact, but it was excluded because it was too small, below the 100 pixel threshold.

Figure 10 shows another blank spot. Again, the fixed-circle method and SPOT identify areas that do not correspond to any real spot, while our method concludes that no spot is present. In this case, BIC chose $K = 1$, so the conclusion that there is no spot present was clear-cut.

We now turn to a more global evaluation of the different methods. Figure 11 displays the segmentation results of part of the subarray using our approach (left column), and the results from SPOT (right column). Our criterion is stability of estimated expression levels across replicates. We evaluate the stability of intensity estimation as the variation in the log-ratio estimate, $l = \log_2 I_1/I_2$, across replicates, where I_1 and I_2 are the signal estimates from channels 1

Table 1. Sum of squared differences of estimates of log ratios for 384 genes across four replicates

	SSD	Percentage reduction
Model-based segmentation	657.02	—
Seeded region growing	826.99	20.6
Fixed-circle	1354.10	51.5

The percentage reduction achieved by model-based segmentation relative to the other methods is also shown.

and 2, respectively, as defined in Section 2.3. Stability is measured by the sum of the squared differences (SSDs), defined as

$$\text{SSD} = \sum_{i=1}^{N}\sum_{r=1}^{R}(l_{i,r} - \bar{l}_i)^2, \tag{3}$$

where N is the total number of spots on the array, R is the total number of replicates, $l_{i,r}$ is the log ratio for the i-th spot on the r-th replicate, and $\bar{l}_i$ is the mean of the log ratio across all replicates for the i-th spot. If no foreground is identified, I_1/I_2 is set to 1. We apply median normalization to our estimate as well as the estimates from SPOT and Spot-on before calculating the log ratios.

Table 1 shows the comparison of stability among the three methods. Our method demonstrates better stability than both the fixed-circle method of Spot-on and the seeded region growing method of SPOT. In the 12×32 array, the SSD of our method was 51.5% lower than that of the fixed-circle method and 20.6% lower than that of SPOT.

A good method not only has less variation, but also does not bias the estimated expression levels of highly expressed genes downwards. A method could achieve small variation by reducing the estimated expression levels of all genes, including those that are differentially expressed, but this would not be a very useful method. Because HIV genes are present only in the HIV-infected sample and are highly expressed in the HIV-infected sample, they can be used as positive controls to check whether estimated expression levels of highly expressed genes are biased downwards. There are three HIV genes in the subarray. Table 2 shows the average of the estimated log ratio across the four replicates for these three genes. The estimates from our method are very close to those from seeded region growing. They are a little smaller than those from fixed-circle segmentation, but this is so much more unstable than our method that this does not seem to be of great concern.

As a final assessment, we carried out a subjective evaluation of whether the method was successfully identifying blank spots (i.e. genes that were not expressed on the microarray). Human eyes typically segment images better than machine vision, and so we compared the results from our automatic computer method with those from a subjective evaluation by one of the present authors. The raw images of four replicates of a 12×8 subarray were examined without prior knowledge of the machine segmentation, and the resulting 384 spots were coded into one of the three classes:

- Not expressed: no visible spots in both channels.
- Questionable: no visible spots in one channel and questionable in the other channel, or questionable in both channels.
- Expressed: otherwise.

Table 2. Average log ratios for the three HIV genes across replicates

	HIV1	HIV2	HIV3
Model-based segmentation	−10.46	−11.62	−11.03
Seeded region growing	−10.33	−10.10	−10.50
Fixed-circle	−12.81	−12.75	−12.87

The log ratios (base 2) are median normalized.

Table 3. Cross-classification of subjective against automated assessments of 96 genes in four replicates

Subjective decision	Automated decision Not expressed	Expressed	% Agreement
Not expressed	23	4	85
Questionable	11	15	—
Expressed	5	326	98

Table 3 shows the cross-classification of the subjective decision and the segmentation using our method. The agreement is quite close: 85% for genes not expressed and 98% for expressed genes. In addition, of the genes about which the observer was unsure, our automatic method split them fairly evenly, identifying 42% as not expressed and 58% as expressed.

To assess the effectiveness of the method in identifying artifacts, the same observer subjectively identified the spots that contained artifacts in a single microarray block. Out of 384 spots, 25 were identified as containing artifacts. The method correctly identified 21 of these cases, and missed them in three cases. In the remaining case, there were two artifacts in the spot, one of which was correctly identified and the other missed by the method. When artifacts were missed, the number of pixels involved was small, ∼5–9 pixels as compared with 350–400 pixels in a typical spot. Failure to identify the artifact would not have had a great impact on foreground intensity estimates in the cases. In no case did the method incorrectly identify foreground as artifact.

4 SOFTWARE

The methods described in this paper are implemented in the R language contributed package spotSegmentation. The software consists of two basic functions: spotgrid, which determines rectangles within cDNA microarray slides in which individual spots are located, and spotseg, which determines foreground and background signals within the spots.

The spotgrid function is used to divide a microarray image block into a grid separating the individual spots. It takes as input the intensities from the two channels, along with the known numbers of rows and columns of spots within a block on a slide. The output is the row and column locations defining a grid separating the individual spots. There is an option to display the grid with the image superimposed.

Individual spots can be segmented using the function spotseg. It takes as input the intensities from the two channels, along with the row and column delimiters of the spots within a block on a

slide (e.g. as determined by spotgrid). There is an option to display the various stages of the segmentation process for individual spots, as well as to display the entire block of segmented spots at the end of processing. Mean and median pixel intensities for the foreground and background for each channel and each spot can be recovered through the summary function applied to the output of spotseg. The spotseg function requires the MCLUST package (http://cran.r-project.org/src/contrib/PACKAGES.html) for the clustering phase.

The spotSegmentation package is available through the Bioconductor project (http://www.bioconductor.org).

5 DISCUSSION AND CONCLUSIONS

We have described a two-step method for segmenting microarray images and estimating intensities. The two steps are model-based clustering of pixel intensities and spatial connected component extraction. Both steps are simple to implement. The method provides a principled statistical basis for determining whether or not a gene is expressed at a spot, and thus deals explicitly with blank spots. It also deals effectively with donut-shaped spots with inner holes and with artifacts. In experiments it yielded results that were more stable across replicates than fixed-circle segmentation or the seeded region growing method implemented in the SPOT software, without introducing noticeable biases in the estimated expression levels of differentially expressed genes. We have made available a software package in R, called spotSegmentation, implementing the method.

ACKNOWLEDGEMENTS

The work of Q.L., C.F. and A.E.R. was supported by NIH grant 8 R01 EB002137-02, and that of A.E.R. was also supported by ONR grant N00014-01-10745. The work of R.E.B. was supported by NIH-NIAID grants 5P01AI052106-02, 1R21AI052028-01 and 1U54AI057141-01, NIH-NIEHA 1U19ES011387-02 and NIH-NHLBI grants 5R01HL072370-02 and 1P50HL073996-01. The work of K.Y.Y. was supported by NIH-NCI grant 1K25CA106988-01.

REFERENCES

Adams,R. and Bischof,L. (1994) Seeded region growing. *IEEE Trans. Pattern Anal. Machine Intell.*, **16**, 641–647.

Ahmed,A.A. *et al*. (2004) Microarray segmentation methods significantly influence data precision. *Nucleic Acids Res.*, **32**, e50.

Angulo,J. and Serra,J. (2003) Automatic analysis of DNA microarray images using mathematical morphology. *Bioinformatics*, **19**, 553–562.

Antonio,P.G.D. *et al*. (2004) A dynamical model with adaptive pixel moving for microarray images segmentation. *Real-Time Imaging*, **10**, 189–195.

Axon Instruments Inc. (1999) GenePix 400A User's Guide.

Bergemann,T.L. *et al*. (2004) A statistically driven approach for image segmentation and signal extraction in cDNA microarrays. *J. Comput. Biol.*, **11**, 695–713.

Bozinov,D. and Rahnenführer,J. (2002) Unsupervised technique for robust target separation and analysis of DNA microarray spots through adaptive pixel clustering. *Bioinformatics*, **18**, 747–756.

Brändle,N. *et al*. (2003) Robust DNA microarray image analysis. *Machine Vision Appl.*, **15**, 11–28.

Dasgupta,A. and Raftery,A.E. (1998) Detecting features in spatial point processes with clutter via model-based clustering. *J. Am. Stat. Assoc.*, **93**, 294–302.

DeRisi,J.L. *et al*. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

Eisen,M. (1999) ScanAlyze.

Fraley,C. and Raftery,A.E. (1998) How many clusters? which clustering method?—answers via model-based cluster analysis. *Comput. J.*, **41**, 578–588.

Fraley,C. and Raftery,A.E. (1999) MCLUST: software for model-based cluster analysis. *J. Classif.*, **16**, 297–306.

Fraley,C. and Raftery,A.E. (2002) Model-based clustering, discriminant analysis and density estimation. *J. Am. Stat. Assoc.*, **97**, 611–631.

Fraley,C. and Raftery,A.E. (2003) Enhanced model-based clustering, density estimation and discriminant analysis software: MCLUST. *J. Classif.*, **20**, 263–286.

Galinsky,V.L. (2003) Automatic registration of microarray images. I. Rectangular grid. *Bioinformatics*, **19**, 1824–1831.

Glasbey,C.A. and Ghazal,P. (2003) Combinatorial image analysis of DNA microarray features. *Bioinformatics*, **19**, 194–203.

GSI Luminomics (1999) QuantArray Analysis Software, Operator's Manual.

Haralick,R.M. and Shapiro,L.G. (1992) *Computer and Robot Vision*. Addison-Wesley.

Hirata,R. *et al*. (2002) Segmentation of microarray images by mathematical morphology. *Real-Time Imaging*, **8**, 491–505.

Jain,A.N. *et al*. (2002) Fully automatic quantification of microarray image data. *Genome Res.*, **12**, 325–332.

Jung,H.Y. and Cho,H.G. (2002) An automatic block and spot indexing with k-nearest neighbors graph for microarray image analysis. *Bioinformatics*, **18**(Suppl. 2), S141–S151.

Katzer,M. *et al*. (2003) Methods for automatic microarray image segmentation. *IEEE Trans. Nanobiosci.*, **2**, 202–214.

Kim,J.H. *et al*. (2001) A novel method using edge detection for signal extraction form cDNA microarray image analysis. *Exp. Mol. Med.*, **33**, 83–88.

Liew,A.W.C. *et al*. (2003) Robust adaptive spot segmentation of DNA microarray images. *Pattern Recogn.*, **36**, 1251–1254.

Lipshutz,R.J. *et al*. (1999) High density synthetic oligonucleotide arrays. *Nat. Genet.*, **21**, 20–24.

McLachlan,G. and Peel,D. (2000) *Finite Mixture Models*. John Wiley & Sons.

Nagarajan,R. (2003) Intensity-based segmentation of microarray images. *IEEE Trans. Med. Imaging*, **22**, 882–889.

O'Neill,P. *et al*. (2003) Improved processing of microarray data using image reconstruction techniques. *IEEE Trans. Nanobiosci.*, **2**, 176–183.

Pinkel,D. *et al*. (1998) High resolution analysis of DNA copy number variation using comparative genomic hybridization to microarrays. *Nat. Genet.*, **20**, 207–211.

Pollack,J.R. *et al*. (1999) Genome-wide analysis of DNA copy-number changes using cDNA microarrays. *Nat. Genet.*, **23**, 41–46.

Rahnenfürer,J. and Bozinov,D. (2004) Hybrid clustering for microarray image analysis combining intensity and shape features. *BMC Bioinformatics*, **5**, 47.

Schadt,E.E., Li,C., Ellis,B. and Wong,W.H. (1999) Feature extraction and normalization algorithms for high-density oligonucleotide gene expression array data. *Technical Report 303*, Department of Statistics, UCLA.

Snijders,A.M. *et al*. (2003) Current status and future prospects of array-based comparative genomic hybridization. *Brief. Funct. Genomics Proteomics*, **2**, 37–45.

Steinfath,M. *et al*. (2001) Automated image analysis for array hybridization experiments. *Bioinformatics*, **17**, 634–641.

The Insitute for Genomics Research (TIGR) (2004) *Spotfinder Online Manual*.

Townsend,J.P. (2004) Resolution of large and small differences in gene expression using models for the Bayesian analysis of gene expression levels and spotted DNA microarrays. *BMC Bioinformatics*, **5**, 54.

van't Wout,A.B. *et al*. (2003) Cellular gene expression upon human immunodeficiency virus type 1 infection of $CD4^+$-T-cell lines. *J. Virol.*, **77**, 1392–1402.

Velculescu,V.E. *et al*. (1997) Characterization of the yeast transcriptome. *Cell*, **88**, 243–251.

Yang,Y.H. *et al*. (2002) Comparision of methods for image analysis on cDNA microarray data. *J. Comput. Graphic. Stat.*, **11**, 108–136.