Comparison of methods for image analysis on cDNA microarray data

Yee Hwa Yang[1*], Michael J. Buckley[2*], Sandrine Dudoit[3] and Terence P. Speed[1,4]

Technical report # 584, November 2000

1. Department of Statistics, University of California at Berkeley
2. CSIRO Mathematics and Information Science, Australia.
3. Department of Biochemistry, Stanford University
4. Division of Genetics and Bioinformatics, The Walter and Eliza Hall Institute, Australia.

Address for correspondence:
Yee Hwa Yang
Department of Statistics
University of California, Berkeley
367 Evans Hall
Berkeley, CA 94720-3860
Tel: (510) 642-0613
Fax: (510) 642-7892
E-mail: `yeehwa@stat.berkeley.edu`

* *These authors contributed equally to this work.*

Abstract

Microarrays are part of a new class of biotechnologies which allow the monitoring of expression levels for thousands of genes simultaneously. Image analysis is an important aspect of microarray experiments, one which can have a potentially large impact on subsequent analyses such as clustering or the identification of differentially expressed genes. This paper reviews a number of existing image analysis methods used on cDNA microarray data and compares their effects on the measured log ratios of fluorescence intensities. In particular, this study examines the statistical properties of different segmentation and background adjustment methods. The different image analysis methods are applied to microarray data from a study of lipid metabolism in mice. We show that in some cases background adjustment can substantially reduce the precision— that is, increase the variability — of low-intensity spot values. In contrast, the choice of segmentation procedure has a smaller impact.

In addition, this paper proposes new addressing, segmentation and background correction methods for extracting information from microarray images. The segmentation component uses a seeded region growing algorithm which makes provision for spots of different shapes and sizes. The background estimation approach uses an image analysis technique known as morphological opening. All these methods are implemented in a software package named *Spot*.

Keywords: Image processing; segmentation; background correction; gene expression, automatic gridding; seeded region growing.

1 Introduction

DNA microarrays are part of a new class of biotechnologies which allow the monitoring of expression levels for thousands of genes simultaneously. Applications of microarrays range from the study of gene expression in yeast under different environmental stress conditions to the comparison of gene expression profiles of tumors from cancer patients. In addition to the enormous scientific potential of DNA microarrays to help in understanding gene regulation and interactions, microarrays have very important applications in pharmaceutical and clinical research. By comparing gene expression in normal and abnormal cells, microarrays may be used to identify genes which are involved in particular diseases. These genes may then be targetted by therapeutic drugs.

Image analysis is an important aspect of microarray experiments, one which can have a potentially large impact on downstream analyses such as clustering or the identification of differentially expressed genes. In a microarray experiment, the hybridized arrays are imaged to measure the red and green fluorescence intensities for each spot on the glass slide. These fluorescence intensities correspond to the level of hybridization of the two samples to the DNA sequences spotted on the slide. The fluorescence intensities are stored as 16-bit images which we view as "raw" data. This paper describes and assesses image analysis techniques for extracting measures of transcript abundance from microarray images.

In the last two years, a number of microarray image analysis packages, both commercial software and freeware, have become available. The processing of scanned microarray images can be separated into three tasks.

1. *Addressing* or *gridding* is the process of assigning coordinates to each of the spots. Automating this part of the procedure permits high throughput analysis.

2. *Segmentation* allows the classification of pixels either as foreground—that is, as corresponding to a spot of interest—or as background.

3. The *intensity extraction* step includes calculating, for each spot on the array, red and green foreground fluorescence intensity pairs (R, G), background intensities, and possibly, quality measures.

Estimation of background intensity is generally considered necessary for the purpose of performing *background correction*. The motivation for background correction is that a spot's measured fluorescence intensity includes a contribution which is not specifically due to the hybridization of the mRNA samples to the spotted DNA. Background correction of the spot intensities is usually performed by subtracting background estimates from the red and green foreground values. with the aim of improving accuracy, that is, reducing bias. Spot quality scores may include measures of spot size or shape, or measures of background intensity relative to foreground intensity.

In this paper, we study the effect of various image analysis decisions on the measured log-ratios for the fluorescence intensities, $\log_2 R/G$. In particular, we examine the choice of segmentation procedure and discuss a number of background adjustment methods. We show that in some cases background adjustment can substantially reduce the precision—that is, increase the variability—of low-intensity spot values. We propose new addressing, segmentation, and background correction methods for extracting information from microarray images. We have implemented these methods in a software package named *Spot*. The addressing method in *Spot* uses the fact that microarray images are generally produced in *batches*, and that within a batch important characteristics, particularly the print-tip configuration, are very nearly the same. Exploiting this structure permits more efficient addressing. The segmentation component uses the *seeded region growing* algorithm of Adams and Bischof [2] and allows spots of different size and shape. The background estimation approach uses an image analysis technique known as *morphological opening* (Soille [19]). In addition, *Spot* also implements a local background estimation method that computes the median pixel value in a region near each spot.

The paper is organized as follows. After describing the creation of laser scanned microarray images in Section 2, we review existing microarray image analysis methods in Section 3. Section 4 discusses the procedures we have developed for addressing, segmentation, and background correction of microarray images. The datasets on which the different image analysis methods are compared are described in Section 5. The study design for the comparison is given in Section 6 and the results are presented in Section 7. Finally, Section 8 summarizes our findings and outlines open questions.

2 Creation of scanned microarray images

cDNA microarrays consist of thousands of individual DNA sequences printed in a high density array on a glass microscope slide using a robotic *arrayer*. The relative abundance of these spotted DNA sequences in two DNA or RNA samples may be assessed by monitoring the differential hybridization of the two samples to the sequences on the array. For mRNA samples, the two samples or *targets* are reverse-transcribed into cDNA, labeled using different fluorescent dyes (e.g. the red-fluorescent dye Cy5 and the green-fluorescent dye Cy3), then mixed and hybridized with the arrayed DNA sequences or *probes* (following the definition of probe and target adopted in the January 1999 supplement to Nature Genetics [1]). After this competitive hybridization, the slides are imaged using a *scanner* which makes fluorescence measurements for each dye. The ratio of the fluorescence intensities for each spot is indicative of the relative abundance of the corresponding DNA sequence in the two nucleic acid samples. The diagram in Figure 1 describes the main steps in a cDNA microarray experiment. More details on this particular experiment are given in Section 5.

*** Place Figure 1 about here ***

Fluorescent images can be acquired using a number of devices, including a laser scanning confocal microscope, or *scanner*, and a charge coupled device (CCD) camera. A microarray scanner performs an area scan of a slide and produces for each dye a digital map, or *image*, of the fluorescence intensities for each pixel. Figure 2 describes diagrammatically how the information from fluorescent dye molecules is converted into digital signals. A typical microarray laser scanner operates with the following functions: excitation, emitted light collection, spatial addressing, excitation/emission discrimination, and detection. The scanned region is divided into equally sized pixels and the laser generates excitation light which is focused on a small portion of the glass microscope slide. Fluorescent molecules in this area absorb the excitation photons generated by the laser and emit fluorescence photons. These emitted photons can go in any direction and a fraction of these are gathered by a lens. We are interested in the number of emitted photons and these are typically several orders of magnitude fewer than the excited photons. In order to prevent detecting a large contribution from the excitation light, which would distort the results, a dichroic beam-splitter and band-pass filter are usually put in front of the detector to discriminate between the excitation and emission photons. This discrimination is possible because the excitation light usually has a slightly smaller wavelength than the emission light.

The detector in a scanner converts the emission photons into electric current. One common type of detector is a photomultiplier tube (PMT). A PMT converts each photon into a number of electrons—up to about one million. The amount of amplification can be adjusted by varying the PMT's voltage input. Finally, an analog to digital (A/D) converter is used to convert the electrons into a sequence of digital signals.

The digitization process averages both spatially and temporally and produces, for each pixel, a signal that represents the total fluorescence in the region corresponding to that pixel. When properly processed, this signal should correlate to the area density of dye molecules. More details and a description of the scanner technology are provided in the books *DNA Microarrays : A Practical Approach* [17] and *Microarray Biochip Technology* [18]. Figure 3 displays a diagram for a simple scanner.

*** Place Figures 2 and 3 about here ***

For a typical microarray experiment, the scanner produces two 16-bit tagged image file format (TIFF) files, one for each fluorescent dye. Different dyes absorb and emit light at different ranges of wavelengths. In order to measure the abundance of the two fluorescent dyes for each spot, the scanners are designed to generate excitation light at different wavelengths and detect different emission wavelengths. The commonly used cyanine dyes Cy3 and Cy5 have emission in the 510–550 nm and 630–660 nm ranges, respectively. A sequential scanner will first scan the glass slide with one wavelength and then scan at the other wavelength. Alternatively, a dual laser scanner has two lasers and two detectors and scans the slide at both wavelengths simultaneously.

A pinhole is placed in front of the PMT detector to control the depth of focus of the objective lens so that only light emitted from the glass slide is detected. Light originating from under the glass slide (e.g. caused by a piece of dust) will be out of focus as it reaches the "pinhole" and therefore only a very small fraction of light will be able to pass through. This arrangement reduces the imaging of artifacts on the glass slides, however, it is important to keep the scanning beam flat because failing to do this results in a loss of signal.

There are various types of noise that can affect the final signal produced by the scanner. These can be divided into two categories: source noise and detector noise. Examples of source noise are photon noise, dust on the slides, and treatment of the glass slides. Detector noise includes features of the amplification and digitization process. A perfect image should only reflect measures of the fluorescence intensities for the dye of interest. However, in practice, we have an imperfect system and the images are usually combinations of undesired signals, such as photon noise, electronic noise, laser light reflection, and background fluorescence, as well as the desired fluorescence signals.

Depending on the scanner, a number of settings (e.g. scan rate, laser power, PMT voltage) can be adjusted by the user. In general, a higher laser power excites more photons and generates more source signal and more source noise. A higher PMT voltage amplifies more electrons per photon and generates more detector noise and more signal. It may be desirable to use a higher laser power rather than a higher PMT voltage as this would excite more photons for the signal rather than produce more "signal" per photon. However, high laser power can damage the hybridized samples by photo-bleaching, and depending on the number of scans to be performed on each sample, the laser power will need to be adjusted accordingly.

For some scanners, only the PMT voltage is adjustable, and not the laser power. Setting an extremely high PMT level may saturate pixels, that is, over a certain level of electrons, the A/D converter will register the signal as $2^{16} - 1 = 65535$. In practice, users adjust the level of PMT so that the brightest pixels are just below the level of saturation. This brings up the question of how varying the level of PMT will affect the final results, especially when one might like to use a different PMT level for the two different channels. We performed an experiment in which a small area of a slide was scanned seven times with varying levels of PMT in the Cy3 and Cy5 channels. We found that after proper normalization the log-ratios and ranks for the majority of genes remained the same (Yang *et al.* [20]).

3 Existing image analysis methods

In this section, we review existing image analysis methods, with an emphasis on segmentation and background adjustment. The "raw" data are two 16-bit TIFF images, one for each dye, obtained by scanning a hybridized slide. The goal is to extract for each spotted DNA sequence a measure of transcript abundance in the two labelled mRNA samples, as well as to obtain background estimates and quality measures. This section is not meant to be a survey of all microarray image analysis software packages available. Rather, different packages, proprietary and non-proprietary, are mentioned mainly as examples of implementations of certain methods and algorithms.

3.1 Addressing

The basic structure of a microarray image is determined by the arrayer and is therefore known. The image in Figure 4 exhibits such a structure. That is, it is known that there are, say, four rows and four columns of grids, and that within each grid there are 19 rows and 21 columns of spots. However, to *address* the spots in an image—that is, to match an idealised model of the array with the scanned image data—a number of parameters need to be estimated. These parameters include

- separation between rows and columns of grids,
- individual translation of grids (caused by slight variations in print-tip positions),

- separation between rows and columns of spots within each grid,
- small individual translations of spots, and
- overall position of the array in the image.

The last of these is usually the most highly variable from image to image within a batch. Other parameters that may in some cases need to be estimated as well are

- misregistration of the red and green channels,
- rotation of the array in the image, and
- skew in the array.

*** Place Figure 4 about here ***

It is desirable for the addressing procedure to be as reliable as possible to ensure accuracy of the whole measurement process. Reliability of the addressing stage can be increased by allowing user intervention. However this can potentially make the process very slow. Ideally we seek reliability while attempting to minimize user intervention so as to maximize efficiency. The addressing steps are often referred to as "gridding" in the microarray literature.

Most software systems now provide both manual and automatic gridding procedures. These are very varied and we won't attempt to describe them here. Our proposed addressing method is described in Section 4.3.

3.2 Segmentation

The *segmentation* of an image can generally be defined as the process of partitioning the image into different regions, each having certain properties (Soille [19]). In a microarray experiment, segmentation allows the classification of pixels as *foreground* (i.e. as corresponding to a spot of interest) or *background*, so that fluorescence intensities can be calculated for each spotted DNA sequence as measures of transcript abundance. Any segmentation method produces a *spot mask*, which consists of the set of foreground pixels for a given spot.

Existing segmentation methods for microarray images can be categorized into four groups, according to the geometry of the spots they produce:

1. fixed circle segmentation,
2. adaptive circle segmentation,
3. adaptive shape segmentation, and
4. histogram segmentation.

Table 1 lists different segmentation approaches and examples of software implementations. In general, most software packages implement a number of segmentation methods.

*** Place Table 1 about here ***

3.2.1 Fixed circle segmentation

Fixed circle segmentation fits a circle with a constant diameter to all the spots in the image. This method is easy to implement and works nicely when all the spots are circular and of the same size. It was probably first implemented in the *ScanAlyze* software written by M. B. Eisen [11] and it is usually provided as an option in most software. Figure 5 contains a small portion of an array, with spots ranging from 5 to 10 pixels in diameter. A fixed diameter segmentation is clearly not satisfactory for all the spots.

*** Place Figure 5 about here ***

3.2.2 Adaptive circle segmentation

In this kind of segmentation, the circle's diameter is estimated separately for each spot. The software *GenePix* for the Axon scanner implements such an algorithm [3]. Note that *ScanAlyze* and other software do provide the user with the option to manually adjust the circle diameter spot by spot. This practice can be very time consuming, since each array contains thousands of spots (the experiment described in Section 5 comprised 16 arrays with about 6000 spots per array).

In practice, however, spots are rarely perfectly circular and can exhibit oval or donut shapes [12]. A circular spot mask can thus provide a poor fit as shown in Figure 6 for a non-circular shaped spot. Sources of non-circularity include the printing process (e.g. features of the print-tips) or the post-processing of the slides after printing (e.g. insufficient time of rehydration). Segmentation algorithms that do not place restrictions on the shape of the spots are thus desirable.

*** Place Figure 6 about here ***

3.2.3 Adaptive shape segmentation

Two commonly used methods for for adaptive segmentation in image analysis are the *watershed* (Beucher and Meyer [5], Vincent and Soille [16]) and *seeded region growing* (SRG) (Adams and Bischof [2]). These methods are beginning to be applied in microarray analysis, although not in the most widely-used software packages. The segmentation method implemented in our software *Spot* is SRG (see Section 4.4).

Both watershed and SRG segmentation require the specification of starting points, or *seeds*, and the weak point of segmentation procedures using these methods can be the selection of the number and location of the seed points. In microarray image analysis, however, we are in the rather unusual situation where the number of features (spots) is known exactly *a priori* and the approximate locations of the spot centres are determined at the addressing stage. Microarray images are therefore well-suited to such methods.

3.2.4 Histogram segmentation

This type of method uses a *target mask* which is chosen to be larger than any spot. For each spot, foreground and background intensity estimates are determined in some fashion from the histogram of pixel values for pixels within the masked area. These methods therefore do not use any local spatial information.

The Chen *et al.* [9] segmentation method uses a circular target mask and computes a threshold value based on a Mann-Whitney test. Pixels are classified as foreground if their value is greater than the threshold, and as background otherwise. This method is implemented in the *QuantArray* software for the GSI Lumonics scanner [13].

The "Histogram" method which is also implemented in *QuantArray* uses a square target mask and defines foreground and background as the mean intensities between some predefined percentile values. By default, these are the 5th and 20th percentiles for the background and the 80th and 95th percentiles for the foreground. By computing the foreground intensities from a higher percentile range, this method usually yields a higher estimate of the foreground.

The main advantage of these methods is their simplicity. However a major disadvantage is that quantitation is unstable when a large target mask is set to compensate for spot size variation. Furthermore, the resulting spot masks are not necessarily connected. In Figure 5, a circular target mask with diameter of 9 pixels is chosen to allow the inclusion of all spots. However, such a large diameter also results in the inclusion of pixels from neighboring spots, as shown with spot (2,1). The quantitation will thus reflect spot intensities of neighboring bright spots rather than just spot (2,1) itself (Table 2 and 3).

*** Place Table 2 and 3 about here ***

3.3 Information extraction

Histogram-based techniques measure spot foreground and background values directly, while other methods such as SRG simply segment the image. After detecting the location, size and shape of each spot using one of these other segmentation methods, we thus need to calculate foreground and background intensities, and possibly spot quality measures.

Most microarray analysis packages define the foreground intensity as the mean or median of pixel values within the segmented spot mask. More variation exists in the choice of background calculation method. Popular approaches include taking the median of values in selected regions surrounding the spot mask (*GenePix* [3], *ScanAlyze* [11], *QuantArray* [13]).

The *ScanAlyze* package considers as background all the pixels that are not within the spot mask but are within a square centred at the spot centre. This is represented by the blue square in Figure 7. The median of values among all these pixels is used to estimate the local background intensity. One of the background adjustment methods implemented in *QuantArray* considers the area between two concentric circles, such as the green circles in Figure 7. By not considering the pixels immediately surrounding the spots, the background estimate is less sensitive to the performance of the segmentation procedure. An alternate set of pixels to be considered as background (implemented in *Spot*) is shown as the four pink diamond-shaped areas in Figure 7. These pink regions are referred to as the *valleys* of the array and have the furthest distance from all four surrounding spots. The local background for each spot can be estimated by the median of values from the four surrounding valleys. Depending on the software, the local valley regions are different, but this method of background estimation is somewhat independent of the segmentation results. The background method implemented by *GenePix* effectively calculates the median intensity from local valley regions. Other methods of background calculation implemented in our software will be described in Section 4.5.2.

Using valley pixels which are very distant from all spots ensures to a large degree that the background estimate is not corrupted by pixels belonging to a spot. Such corruption by bright pixels may occur in the other methods, particularly the *ScanAlyze* method, introducing an upward bias into the background estimate. Using remote pixels reduces this bias effectively but entails the use of a smaller number of pixels and therefore increases the variance of the estimate. This is an example of the bias-variance trade-off.

*** Place Figure 7 about here ***

4 New image analysis methods

4.1 Software infrastructure

Our software package, *Spot*, is a prototype system for the analysis of microarray data. *Spot* is built on "*R*" [14], an environment for data analysis which is available as free software under the GNU Public License (GPL). As well as providing a wide range of graphical and statistical tools, *R* supports a well-designed and efficient programming environment. The results of an analysis by *Spot* of microarray image data are returned as a standard *R* object and can immediately be displayed, manipulated and analysed in a number of ways. *Spot* is actually a specialised version of another *R* package called *VOIR*, which is currently being developed by the CSIRO Image Analysis Group, and provides a more general image analysis environment.

4.2 Forming a combined image

The input to the image analysis procedure consists of a pair of unsigned 16-bit images which are stored in TIFF format files. We name these images "$\mathcal{R}$" and '$\mathcal{G}$", for "red" and "green", with $\mathcal{R}$ corresponding to the dye Cy5 and $\mathcal{G}$ to Cy3. Both the addressing and segmentation stages require a single image. This image should not be dominated by either of the two inputs—that is, raw images $\mathcal{R}$ and $\mathcal{G}$ should contribute equally in the combination. Another crucial requirement is that very high image values should be *damped* in the combined image. This is needed to stop very bright pixels dominating in both the addressing and segmentation phases. It is also convenient computationally for the combined image to be an 8-bit image. The automatic addressing and segmentation procedures are performed on the 8-bit combined image. The segmentation method will produce a *spot mask* which is used together with the original 16-bit images for extraction of spot foreground and background intensities.

The following processing is used in *Spot* to produce an 8-bit combined image, $\mathcal{RG}$, and achieve these aims.

- First, a square-root transformation is applied to both the inputs, $\mathcal{R}$ and $\mathcal{G}$, giving $\mathcal{R}'$ and $\mathcal{G}'$. This reduces the domination of very bright pixels in both the addressing and segmentation stages.
- Next, median values, $m_{\mathcal{R}}$ and $m_{\mathcal{G}}$, are computed from these images.
- An initial combination is then computed as

$$\mathcal{G}' + \left(\frac{m_{\mathcal{G}}}{m_{\mathcal{R}}}\right) \cdot \mathcal{R}'.$$

- Finally, values greater than 255 are set to 255.

The first and last of these steps apply damping, while the second and third steps ensure equal balance. The last step ensures that the result, $\mathcal{RG}$, can be stored as an 8-bit image.

We have described here a method for combining the two channels, $\mathcal{R}$ and $\mathcal{G}$, into a single image for the purposes of automatic analysis. This is a different process from the standard method of combining $\mathcal{R}$ and $\mathcal{G}$ by *overlaying* for the purposes of *visualisation*.

4.3 Automatic addressing

The addressing procedure relies on the concept of a *batch* of image data. For the purposes of image analysis, a batch is a collection of microarray images whose overall geometric structure is the same. These will typically correspond to slides printed by the same arrayer and the same print-head at around the same time, and scanned in a similar manner.

The geometry of microarray images can vary in a number of ways:

- Basic structure: The most fundamental geometric structure is the arrangement of grids (e.g. 4-by-4) and the arrangement of spots within grids (e.g. 19-by-21).

  Images within a batch must be identical in terms of this basic structure.

- Print-tip configuration: A second major aspect of geometric structure is print-tip configuration. The print-tips on the arrayer's print-head do not in general have a perfectly regular layout. That is, while print-tips are nominally in a regular array, for example, 4-by-4, slight bends or other effects mean that in practice small irregularities in their layout are usually present. Even if the irregularities in the print-tip configuration are very slight, they can result in significant irregularity in the grids of the microarray slide and hence in the image.

  We assume that slides in the same batch have print-tip configurations which are very nearly the same.

- Overall translation: Various factors, image cropping in particular, can lead to an overall shift in all spot positions from image to image. We *do* expect such variation within batches; in fact a key component of the addressing process is an estimate of the overall shift between the current image and the template.

- Rotation and skewing: At present we do not expect other distortions such as rotation or skewing. Significant amounts of such distortion will therefore lead to incorrect results.

To begin the analysis of a batch of images, the user chooses one of the images to serve as a *template* for the whole batch. The user then specifies—by point-and-click with a mouse in an image display window—some features on this template image. Specifically, the user identifies the top-left spot in each grid as well as the bottom-right spot in the bottom-right grid. This process captures firstly print-tip configuration information, and secondly the average separation between rows and columns of spots within a grid.

The remainder of the addressing procedure is automatic. Although we will not explain all the details here, there are two parts to the process. Firstly, the software estimates an overall shift of the grid in the current image relative to the template image. That is, the grid location in the new image is initially estimated by translation of the template grid. Secondly, small adjustments are estimated for each of the rows and columns of spots within each grid. This allows for small variations in structure between grids, as well as inaccuracies in the template specification. Note that there are two essentially equivalent representations of these estimated grids. The first, defined as *fitted foreground grids*, consists of horizontal and vertical lines passing through the (estimated) centres of the spots. The second, defined as *fitted background grids*, consists of horizontal and vertical lines passing through the (estimated) centres of the *gaps between the spots.*

4.4 Segmentation: Seeded region growing

Segmentation of foreground (spots) and background is carried out in *Spot* using the *seeded region growing* (SRG) algorithm of Adams and Bischof [2] mentioned in Section 3.2.3. Briefly, this method works as follows. A number of *seeds* are provided as input to the algorithm. These are groups of pixels which serve as starting points for a region growing process. Often seeds consist of only a single pixel, but they can be of any size and do not need to form a connected set. We describe below how we construct the seeds in this application of SRG.

After specification of seeds, the algorithm proceeds by growing all the foreground and background regions simultaneously until all pixels in the image have been allocated to one of the regions. At each stage, all pixels which are as yet unallocated, but which have at least one neighbor which has already been allocated, are considered for allocation. Out of all these region-neighboring pixels, the algorithm selects the one whose pixel value is nearest (in terms of absolute grey-level difference) to the average of the pixel values in the neighboring region. The process repeats until all pixels have been allocated. Pixel queues are used to optimize the efficiency of the procedure.

For microarray segmentation using SRG, the foreground and background seeds are chosen using the grids calculated in the addressing stage. An obvious way to choose a seed for each spot is to choose a single pixel from the intersections of the horizontal and vertical grid lines of the fitted foreground grid. However, it is possible, particularly when the spot is small, that this intersection pixel may not be inside the spot because of local irregularities or small errors in the grid estimation. To overcome this problem, a point is chosen by finding the maximum of the combined intensity surface over a small region centred at the intersection pixel. The foreground seed is then set to be an n-by-n square of pixels centered on this point. The integer n is specified by the user.

Background seeds need to be computed also. A very simple approach would be to use the intersection points from the fitted background grids as background seeds, or indeed to use all of the grids together as one large background seed covering most of the image. Such a procedure has the advantage of separating the foreground seeds from each other and therefore ensuring that the segmented spots cannot merge or bleed into one another. There are, however, two reasons why the use of such large background seeds is undesirable. The first is that background intensity is often locally varying and poor performance is expected for SRG if regions are not homogeneous in intensity. A second reason is that we require *local* estimation of background intensity and this can be obtained by having smaller, more local background regions. For these reasons we construct background seeds as *crosses* as illustrated in Figure 8. It can be seen that this design for background seeds achieves the aim of separating spots from each other while at the same time producing relatively small, local background regions. SRG is applied using these seeds to the "combined" image of Section 4.2.

4.5 Information extraction

4.5.1 Spot intensity

Each pixel value in a scanned image represents the level of hybridization at a specific location on the slide. The total amount of hybridization for a particular spotted DNA sequence is proportional to the *total fluorescence* at the spot. The natural measure of spot intensity is therefore the *sum* of pixel intensities within the spot mask. Since later calculations are based on the ratio of fluorescence intensities, we compute the *average* pixel value over the spot mask. This gives identical results, as the ratio of averages is equal to the ratio of sums.

Other statistics which can be computed within each spot mask and for each of the two channels are *median* pixel value and measures of *variability* in pixel value.

4.5.2 Background intensity

The segmentation procedure described in Section 4.4 produces local background regions as well as segmented spots. Because of the structure of the foreground and background seeds, there are four such background regions surrounding each spot. These can be used in a variety of ways to compute local background estimates. One such procedure, the *valley* method (Section 3.3), computes the median pixel value in each background region, and then for each spot computes the average of the four corresponding background medians as the local background estimate for that spot.

Our preferred approach to background adjustment relies on a non-linear filter called *morphological opening*; see Soille [19] for a detailed description. Morphological opening is applied to the original images $\mathcal{R}$ and $\mathcal{G}$ using a square structuring element with side length at least twice as large as the spot separation distance. This operation removes all the spots and generates an image which is an estimate of the background for the entire slide. For individual spots, background is estimated by sampling this background image at the nominal centre of the spot. We simply chose to sample this image rather than take an average over a "background region", because very similar results are expected from both methods. A very large window was used to create the morphological background image, hence it is expected to have very slow spatial variation.

Morphological opening results in smaller background estimates than other simpler methods. More importantly though, morphological background estimation is expected to be *less variable* than the other methods, because spot background estimates

- are based on pixel values in a large local window, and yet
- are not corrupted (i.e. biased upwards) by brighter pixels belonging to or on the edge of the spots.

4.5.3 Quality measures

In addition to the actual spot foreground and background intensities, it is desirable to also collect statistics describing the quality of these measurements. Variability measures mentioned in Section 4.5.1 are one kind of quality measure. Other quality measures provided in *Spot* include spot size (area in pixels), a circularity measure and relative signal to background intensity [6]. We have yet to make use of these measures in our analyses. Table 4 shows the output from *Spot* for the image shown in Figure 5.

The above image processing steps generate two main quantities for each spot on the array: R and G, which are measures of transcript abundance for the red and green labeled mRNA samples, respectively. The two values are usually combined into a single log-intensity ratio, $\log_2 R/G$, which measures relative transcript abundance in the two samples. A positive (negative) log-ratio indicates over-expression (under-expression) in the red labeled mRNA sample compared to the green. Before proceeding to any inference or clustering, normalization is needed in order to identify and remove systematic sources of variation (e.g. different labeling efficiencies and scanning properties of the dyes, print-tip or spatial effects) and allow between-slide comparisons (Yang *et al.* [20]).

*** Place Table 4 about here ***

5 Experimental data

A) Apo AI experiment

The apo AI experiment was carried out as part of a study of lipid metabolism and artherosclerosis susceptibility in mice. Apolipoprotein AI (apo AI) is a gene known to play a pivotal role in HDL metabolism. Mice with the apo AI gene knocked-out have very low HDL cholesterol levels and the goal of the apo AI experiment was to identify genes with altered expression in the livers of these knock-out mice compared to inbred control mice.

The treatment group consisted of eight mice with the apo AI gene knocked-out and the control group consisted of eight "normal" C57Bl/6 mice. For each of these 16 mice, target cDNA was obtained from mRNA by reverse transcription and labelled using a red-fluorescent dye, Cy5. The reference sample used in all hybridizations was prepared by pooling cDNA from the eight control mice and was labelled with a green-fluorescent dye, Cy3. In this experiment, target cDNA was hybridized to microarrays containing 5,548 cDNA probes, including 200 related to lipid metabolism. Note that we call the spotted cDNA sequences "genes", whether they are actual genes, ESTs (expressed sequence tags), or DNA sequences from other sources.

Each of the 16 hybridizations produced a pair of 16-bit images, which were processed using a number of segmentation and background correction methods (Table 5). The main quantities of interest produced by the image analysis methods are the (R, G) fluorescence intensity pairs for each gene on each array. In order to identify and remove systematic sources of variation in the measured expression levels and allow between-slide comparisons, the data were normalized using a within-slide spatial and intensity dependent normalization method (Dudoit *et al.* [10], Yang *et al.* [20]).

After image processing, background correction and normalization, the gene expression data can be summarized by a matrix X of log-intensity ratios $\log_2 R/G$, with k rows corresponding to the genes being studied and $n = n_1 + n_2$ columns corresponding to the n_1 control hybridizations (C57Bl/6) and n_2 treatment hybridizations (apo AI knock-out). In the experiment considered here $n_1 = n_2 = 8$ and $k = 5,548$.

Differentially expressed genes were identified by computing t-statistics. For gene j, the *t-statistic* comparing gene expression in the control and treatment groups is

$$t_j = \frac{\bar{x}_{2j} - \bar{x}_{1j}}{\sqrt{\frac{s_{1j}^2}{n_1} + \frac{s_{2j}^2}{n_2}}},$$

where $\bar{x}_{1j}$ and $\bar{x}_{2j}$ denote the average background corrected and normalized expression level of gene j in the n_1 control and n_2 treatment hybridizations, respectively. Similarly, s_{1j}^2 and s_{2j}^2 denote the variances of gene j's expression level in the control and treatment hybridizations, respectively. Large absolute t-statistics suggest that the corresponding genes have different expression levels in the control and treatment groups. The analysis of this dataset is described in detail in Dudoit *et al.* [10].

B) Follow-up experiment

For the apo AI experiments, the 20 clones with the largest absolute t-statistics were selected and spotted on a miniarray. Some clones actually comprised more than one clone and these were purified and re-checked. Each of the approximately 50 distinct clones from the top 20 clones were spotted eight times on the miniarray down a single column in the same print-tip group. Another 72 genes were spotted in the same pattern for normalization purposes.

Pooled mRNA from four apo AI knock-out mice (treatment) and four wild-type C57Bl/6 mice (controls) was hybridized to the miniarray. Anticipating that most genes on the miniarray were differentially expressed, a dye swap experiment was done to allow normalization between the two channels. In the first hybridization, the treatment (apo AI ko) mRNA is labeled red and the control mRNA is labeled green. In the second hybridization, the original labeling is reversed, with treatment labeled green and control labeled red. Here again, pairs of 16-bit images were processed using the segmentation and background correction methods described in Table 5. However, normalization was performed jointly for both slides as described in Yang *et al.* [20]. For each selected clone, we thus have 16 measurements of relative transcript abundance in the treatment and control mice, eight in each of two slides.

6 Study design

The data from the two experiments described above (Callow *el. al.* [7]) are used to compare the merits of various microarray image analysis methods. The methods we consider are implemented in our own image analysis software *Spot*, the publicly available software *ScanAlyze* [11], and two commercial packages, *GenePix* from Axon Instrument Inc. [3] and *QuantArray* from GSI Lumonics [13]. In this study, we classify broadly the various background methods implemented in software packages into four categories. These are

1. *Local background :* Background intensities are estimated by focusing on small regions surrounding the spot mask. Usually, the background estimate is the median of pixel values within these specific regions. Most software packages we have encountered implement such an approach.

2. *Morphological opening:* This approach is described in Section 4.5.2.

3. *Constant background :* This is a global method which subtracts a constant background for all spots. The approaches previously described assume that the non-specific binding to a spot can be estimated by the surrounding area. However, some findings (Lou [15]) suggest that the binding of fluorescent dyes to "negative control spots" (e.g. spots corresponding to plant genes that will not hybridize with the mRNA samples of interest) is lower than the binding to the glass slide. If this is the case, it may be more meaningful to estimate background based on a set of negative control spots. When there are no negative control spots, one could approximate the average background by, for example, the 3rd percentile of all the spot foreground values.

4. *No adjustment:* Finally, we also consider the possibility of no background adjustment at all.

Unless specified otherwise, spot foreground intensities are calculated by taking the mean intensity of the pixels within the spot mask. The different methods are summarised in Table 5.

*** Place Table 5 about here ***

Comparison of foreground intensities and local background intensities across methods

We first compare the different image analysis methods in terms of their estimates of local background and foreground spot intensities (i.e, spot intensities before background correction). The foreground intensities reflect properties of the segmentation method. We are interested in seeing whether some methods produce systematically lower foreground and background intensity estimates than others and also whether the estimates from different methods are correlated. To this end, we produce scatter-plots of the local background and foreground intensity estimates for pairs of methods implemented in our software *Spot* and other methods listed in Table 5.

Correlation of background corrected spot intensities and local background intensities

Ideally, there should be no correlation between background corrected spot intensities and local background intensities. Otherwise, the signal could be dependent on factors other than the hybridization of the target to the probe. For example, when there is high background near the edges of the cover-slip due to target dehydration during the hybridization, local background and foreground spot intensities are likely to be correlated [12]. Plots of background *vs.* uncorrected spot intensities are therefore not very informative. This issue is especially important for low intensity spots, since background subtraction has a larger impact on the R/G ratio for such spots than for high intensity spots. Thus, for each background method, we examine the correlation between background corrected spot intensities and background intensities for the spots in the lower half of the spot foreground intensity distribution.

Within slide variability

For the follow-up experiment (B), each slide contains eight replicated spots for each of the 122 clones. We can thus compare image analysis methods in terms of the within slide standard deviations (SD) of the log-intensity ratios of the replicated spots. A small within slide variance is a desirable property. However, one can argue that in the extreme case when the log-ratios from each spot are set to the same constant value, there will be no variability, but the log-ratios will be very inaccurate. This is a classic example of bias and variance trade-off. Unfortunately, since we do not have "true" expression levels for each of the clones, we are not able to assess bias and address the question of accuracy fully in this experiment (more detailed discussion in Section 8).

***t*-statistics**

For experiment (A), we can get some idea of the bias and variance properties of the different image analysis methods by considering the t-statistics, their denominators and the corresponding adjusted p-values. In some sense, the denominator of the t-statistic, which is based on the SDs of the log-intensity ratios for the eight treatment and eight control hybridizations, is measuring between slide variability (which in this case is confounded with between mouse variability). An image analysis method which has a small t-denominator is thus desirable. We use boxplots to compare the distribution of the t-denominators (standard error (SE) measuring between slide variability) across the various methods.

The t-statistics allow us to consider bias and variability together. Proxy measures of bias in this case are the numerators of the t-statistics. In the apo AI experiment, the apo AI gene is knocked-out in the eight treatment mice, so one expects the t-statistics to take on very large negative values for this gene. Therefore, checking that an image analysis method that reduces t-denominator doesn't result in a smaller t-statistic in the three apo AI genes spotted on the array provides us with some assurance that such a method doesn't reduce variance by reducing accuracy. For such a comparison, we plot the t-statistics for different image analysis methods and focus on extreme negative t-values by truncating the plot at -4.

In addition, a good image analysis method should enable a clear distinction between differentially expressed genes and noise; this is reflected in the t-statistics as well as the adjusted p-values. A t-statistic with a large numerator relative to its denominator for the differentially expressed genes demonstrates its ability to distinguish the genes with differential expression from the rest of the genes used in the experiment. Similarly, a large jump in the adjusted p-values between the least extreme of the differentially expressed genes and the most extreme of the remaining genes also reflects such a capability. These adjusted p-values are calculated based on the Westfall and Young step-down adjusted p-value algorithm described in Dudoit *et al.*[10].

7 Results

Comparison of foreground intensities and local background intensities across methods

Scatter-plots of foreground and background estimates between any two image analysis methods are an informative way of comparing the effects of the two methods. Figure 9 shows scatter-plots of base-2 logarithms of foreground intensities (red stars) calculated from `S.morph` versus the corresponding quantities from `SA`, `QA.fix`, `GP` and `QA.hist`, respectively. Background estimates are plotted as blue crosses.

Consider foreground values first. In all four plots, foreground intensity values are positively correlated with higher intensity values having a tighter correlation. Figures 9 (a) and (b) show that the SRG segmentation (`S.morph`) gives higher foreground estimates than both *ScanAlyze*'s and *QuantArray*'s fixed circle methods (`SA` and `QA.fix`), while Figure 9 (c) shows that `GP` and `S.morph` produce similar foreground estimates. A comparison between `S.morph` and `QA.hist` (Figure 9 (d)) shows that the latter tends to have slightly higher foreground estimates. This is to be expected, since the histogram method `QA.hist` calculates foreground intensities as the mean of pixels values between two high percentiles (80th and 95th percentiles), whereas the SRG method computes the mean of all pixel values within the spot mask.

The background values across methods tend to have very low correlations. This is a concern, and suggests that little useful information is being provided by any of the various background estimates, all of which, in varying degrees, are noisy. *ScanAlyze*'s local median (`SA`) has smallest variability in this data, followed by morphological opening (`S.morph`) and valley median (`GP`). The *QuantArray* estimates (`QA.fix` and `QA.hist`) are extremely variable. The very high values sometimes produced by the concentric-circles method, `QA.fix`, are probably the result of neighbouring spots being included in the region between the circles and contaminating the pixel values in that region with some high values.

In addition, the background estimates are considerably lower for the morphological opening method `S.morph` than for all other methods. This is because the opening filter is like a rank filter with rank value near zero, in that it often returns a value which is close to the minimum pixel value in the neighbourhood.

*** Place Figure 9 about here ***

Correlation of background corrected spot intensities and local background intensities

Figure 10 (a) displays a plot of background corrected spot intensities (foreground minus background) versus corresponding background intensities for the `S.morph` method applied to an individual slide from the apo AI experiment. The data are from the method `S.morph`. Only values from the lower half of the foreground intensity distribution are displayed. The plot shows a random cloud of points, meaning that, as desired, background and corrected foreground intensities are only very weakly correlated. Table 6 shows correlations for similar data for two mice and each of the various image analysis methods. Except for `QA.fix`, values are near zero, indicating desired behaviour for all the other methods. Figure 10 (b) displays the data for mouse #8 and method `QA.fix`. Substantial negative correlation can be seen. It was seen above that the method `QA.fix` often returns very high background estimates. Clearly for such high background values the foreground-minus-background difference will be very small, resulting in negative correlation between background and foreground-minus-background.

*** Place Figure 10 and Table 6 about here ***

Within slide variability

Before looking at results, let us consider the effect of the magnitudes of the background estimates on variability. Suppose for the sake of simplicity that we are using constant background estimates, r and g when computing spot intensity ratios,

$$\frac{R_i - r}{G_i - g}.$$

These ratios are simply the slopes of the lines joining data points (R_i, G_i) and the fixed background point (r, g). If the background point is moved downward and to the left, that is, reducing both r and g, then these slopes will tend to become less variable. As a limiting case, consider both r and g being equal to the same very large negative number. In this case, all the slopes will be very close to 1, and their SD—the SD of the ratios—will be very small. For non-constant (i.e. locally varying) background estimates, the same basic argument applies: smaller background estimates tend to give smaller variability of ratios of background corrected spot intensities for replicated spots. This demonstrates the inadequacy of the within-group variability of ratios or log-ratios as a performance measure; one also needs to consider bias in the estimation of background. To address the issue of bias and variability simultaneously, t-statistics are considered below.

Bearing these observations in mind, let us consider the variability of data shown in Table 5. Dataset (B) comes from a dye swap experiment in which each slide contains 122 different genes spotted eight times. For each segmentation and background method described in Table 5, we calculated the within slide SDs of the log-ratios $\log_2 R/G$ for the eight replicates for each of the 122 genes and reported their median in Table 7.

For both slides, median SDs for no background adjustment are around 0.06, except for `QA.adp.nbg`. The higher variance for `QA.adp.nbg` suggests poor performance of Chen's method for foreground estimation. Note that no adjustment is equivalent to constant adjustment with background values $r = 0$ and $g = 0$. The constant background method `S.const` gives an SD approximately twice as large as "No Adjustment": 0.14. We believe this is mostly due to this constant estimate having values of both r and g which are larger than the zero values implicit in "No adjustment".

The locally varying background estimates have SD's ranging from 0.08 to 0.27. The smallest SDs are for the morphological opening method, `S.morph`, and the *QuantArray* method, `QA.hist`, while the highest is for `QA.adp`. Some of this effect is due to differences in the magnitude of the background estimates; for example, the low SD for `S.morph` is due in part to the fact that this background estimate tends to be a low value (see Figure 9). However some of this variation in SDs between methods is due to the stability of the estimates; for example, `QA.adp` estimates background using only eight pixels. Such an estimate must be noisy and therefore a high SD for this method is not surprising. With this data, however, it is impossible to explain with confidence why the different methods perform as they do. Some variation between methods is due to the relative *magnitudes* of background estimates and some to their relative *stabilities*.

*** Place Table 7 about here ***

t-statistics

A similar trend can be seen when looking at between slide variability for dataset (A). Figure 11 shows boxplots of t-denominators (i.e., estimated SE for the average difference in expression levels between the treatment and control groups) for the various image analysis methods of Table 5. As before, the different segmentation methods with no background adjustment have the smallest SEs, followed by `S.morph` and `QA.hist`.

Figure 12 shows a plot of t-values for different image processing methods, truncated at -4 to allow us to focus on extreme negative t-values. Methods `S.morph` and `S.const` perform best in terms of their ability to detect differential expression in the three apo A1 genes which have been knocked-out. For methods `S.morph` and `S.const` the t-values for each of the three apo A1 genes are less than -11 whereas some other methods have t-values as high as -5 for the apo A1 genes.

As well as the three apo A1 genes, another 5 genes were found to be differentially expressed by most image processing methods. All 8 genes were confirmed by RT-PCR ([7]). Some methods performed better than others in their ability to distinguish these eight genes from all the other genes in the experiment. For example, method `SA` produced t-values of -7.85 or less for all of these eight genes. Similar good performance was achieved by all other methods with background correction, except `QA.adp`. Of the methods without background correction, only `S.nbg` performed well in this regard.

The gap between t-values for these eight genes and for the remaining genes are largest for methods `S.nbg`, `S.valley`, `SA`, `S.morph` and `S.const`. This shows an ability to clearly distinguish between differentially expressed genes and noise. Table 8 shows the p-values for the extreme-valued t-scores, adjusted for multiple comparison. It can be seen from this table that for all methods except `QA.adp` and `QA.adp.nbg` there is a jump in adjusted p-value between the least extreme of the 8 genes and the most extreme of the remaining genes. Among these methods, the largest jump is observed with `S.morph` followed by `S.nbg` and `SA`.

*** Place Figures 11and 12 about here ***

8 Discussion

In this paper, we have discussed image analysis methods for extracting information from microarray scanned images. We have compared a number of existing background correction and segmentation methods on scanned images from two sets of experiments: the apo A1 experiment (dataset (A)) with replicated treatment and control slides, and the follow-up dye-swap experiment (dataset (B)) with replicated spots on a slide. The comparison indicates that the choice of background adjustment method can have a large impact on the background-corrected log-ratios which are the primary outputs of the image analysis system. In contrast, the various segmentation approaches (fixed or adaptive circles or shapes) have a smaller impact. The comparison further suggests that seeded region growing segmentation with morphological background correction provide good estimates of foreground and background intensities. These proposed approaches are implemented in the software package *Spot*, which also provides an automatic gridding procedure.

The motivation behind background adjustment is the belief that a spot's measured intensity includes a contribution not specifically due to the hybridization of the target to the probe, but to something else, for example, non-specific hybridization and other chemicals on the glass. We would like to measure this contribution and subtract it in order to obtain a more accurate quantitation of hybridization. The glass slides are treated chemically so that the spotted cDNA fragments will bind to them. After the cDNA spots are printed, the slides are treated again so that target DNA does not bind to them. Nevertheless, some binding of the target to the slide may occur. Furthermore, there may be some fluorescence away from the spots due to the slide's surface treatment and the glass. It seems likely that the fluorescence from regions of the slide not occupied by DNA is different from that from regions occupied by DNA. It follows that measuring the intensity in some region near a spot and subtracting it may not be the best way to correct for this extra contribution. It would be interesting to compare morphological and local background estimates to estimates based on negative controls (i.e. spotted DNA sequences which should have no hybridization signal).

Our comparison of different methods for estimating such undesired contribution suggests that morphological opening provides a better estimate of background than other methods. The log-ratios $\log_2 R/G$ computed after morphological background correction tended to exhibit low within and between slide variability. In addition, based on our examination of the t-statistics for dataset (A), this method didn't seem to compromise accuracy. Most of the image processing methods were able to identify the three apo AI clones (having strong negative ratios), although the morphological method of *Spot* tended to have more extreme t-

values for these clones than any of the other packages, *GenePix*, *ScanAlyze* and *QuantArray*.

As well as the three apo A1 clones, the expression of five other genes were consistently found to be suppressed in the treatment group. Some of these genes were under-expressed due to the lack of apo A1 and others (apo CIII) due to their close proximity to the apo A1 locus (Callow *et al.* [7]). In terms of separating all eight genes related to apo A1 from the remaining unaffected genes, *Spot* performed very well, closely followed by *ScanAlyze*, and then *GenePix*. Performance of the *QuantArray* package was generally poor, as measured by these experiments.

Morphological opening tends to provide lower background estimates than other methods. This is because the morphological opening filter produces output values which are close to the minimum pixel value in the neighborhood of each pixel, whereas all other methods base their background estimates on average or median pixel values in local neighborhoods. Sets of background-corrected ratios

$$\frac{R_i - r_i}{G_i - g_i}$$

have smaller variability if the background points (r_i, g_i) are further away from the foreground points, (R_i, G_i). A consequence is that the variability of these ratios, and hence of the log-ratios also, tends to be smaller for the morphological background estimates, just because these background estimates are smaller, and so further away from the foreground points. Because of this phenomenon, the variability of replicate log-ratios is not in itself a useful measure of performance, as smaller variability can be achieved simply by using lower, or darker background estimates.

Morphological opening seems to provide stable estimates of background. In contrast, estimates based on means or medians in local neighborhood regions (e.g. local valleys) tend to be more variable. In order to study morphological opening in more detail we varied the size of the structuring element when processing the images from one of the follow-up experiments (`C3K5`). Opening with a smaller element tends to produce more variable background estimates, as it performs filtering using a smaller window. Thus, when the structuring element is too small (say, similar to the size of an average spot), the within slide SD's for replicated spots are similar to those for local background methods. The within slide SD's tend to decrease when the size of the structuring element increases. This is to be expected, as a larger structuring element will yield more stable background estimates. The SD's remain similar for a wide range of structuring elements sizes. Finally, in the extreme case when the size of the element is taken to be the size of the image, the SD's increase again. In this case the method is equivalent to using a constant background adjustment.

One of the main findings of our study is that the choice of background correction method has a larger impact on the log-intensity ratios than the segmentation method. Thus, finding the best segmentation method was not the primary focus of the paper. While fitting circles to spots is not sufficiently adaptive to accurately segment all spots, SRG is potentially *too* adaptive. We believe that a compromise method is needed to produce very good segmentation of microarray images. One possibility is to *regularise* the SRG algorithm. This is an interesting area for possible future research.

For a more conclusive study of the statistical properties of different image processing methods, one would need a more rigorous assessment of bias, such as one based on an external measure of truth. One might verify estimated expression levels via northern blot or RT-PCR. However, it is difficult to compare the quantitations from RT-PCR to those from microarrays (Bartosiewicz *et al.* [4], Chen *et al.* [8]). Alternatively, to fully address the bias issue, one might perform a series of dilution experiments, which would bypass the need for knowing the true fold changes. For an experiment with a handful of differentially expressed genes, one could look at the log-ratio of such genes across various dilutions.

The comparison of different background correction methods indicates that estimates based on means or medians over local neighborhoods tend to be quite noisy and can potentially double the SD of the log-ratios. At the other extreme, no background adjustment seems to reduce the ability to identify differentially expressed genes, as shown with the decrease in the magnitude of the t-statistics for experiment (A). Therefore, we recommend performing an intermediate background adjustment, which provides less variable estimates than local background methods and more accurate estimates than raw intensities (no background correction at all). Morphological opening seems to provide a good balance in terms of the bias/variance trade-off.

Acknowledgments

We would like to acknowledge Richard Beare, Mark Berman, Kevin Cheong, Ryan Lagerstrom and Hugues Talbot from the CSIRO Image Analysis Group for providing us with their image analysis software as well as guiding us through the many difficulties in implementing the algorithms as a library in *R*. We would also like to thank Matthew J. Callow from the Lawrence Berkeley National Laboratory for providing the data for this comparison study. Members of the Ngai Lab at UC Berkeley and Chuang Fong Kong from the Peter MacCallum Cancer Institute in Melbourne have been very helpful in quantifying M. Callow's data using the *GenePix* and *QuantArray* software, respectively. Finally, we would like to thank William F. Kolbe of the Lawrence Berkeley National Laboratory for helpful discussions on the operation of microarray scanners.

This work was supported in part by the NIH through grants 5R01MH61665-02(YHY) and 8R1GM59506A(TPS), and by an MSRI and a PMMB postdoctoral fellowship (SD).

References

[1] *The Chipping Forecast*, volume 21, January 1999. Supplement to Nature Genetics.

[2] R. Adams and L. Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16:641–647, 1994.

[3] Axon Instruments, Inc. *GenePix 4000A User's Guide*, 1999.

[4] M. Bartosiewicz, M. Trounstine, D. Barker, R. Johnston, and A. Buckpitt. Development of a toxicological gene array and quantitative assessment of this technology. *Archives of Biochemistry and Biophysics*, 376:66–73, 2000.

[5] S. Beucher and F. Meyer. The morphological approach to segmentation: the watershed transformation. In *Mathematical morphology in image processing, volume 34 of Optical Engineering*, chapter 12, pages 433–481. Marcel Dekker, New York, 1993.

[6] M. J. Buckley. *The Spot user's guide*. CSIRO Mathematical and Information Sciences, August 2000. `http://www.cmis.csiro.au/IAP/Spot/spotmanual.htm`.

[7] M. J. Callow, S. Dudoit, E. L. Gong, T. P. Speed, and E. M. Rubin. Microarray expression profiling identifies genes with altered expression in hdl deficient mice. *Genome Research*, 2000. Submitted.

[8] J.W. Chen, R. Wu, P.C. Yang, J.Y. Huang, Y.P. Sher, M.H. Han, W.C. Kao, P.J. Lee, T.F. Chiu, F. Chang, Y.W. Chu, C.W. Wu, and K. Peck. Profiling expression patterns and isolating differentially expressed genes by cDNA microarray system with colorimetry detection. *Genomics*, 51:313–324, 1998.

[9] Y. Chen, E. R. Dougherty, and M. L. Bittner. Ratio-based decisions and the quantitative analysis of cdna microarray images. *Journal of Biomedical Optics*, 2:364–374, 1997.

[10] S. Dudoit, Y. H. Yang, M. J. Callow, and T. P. Speed. Statistical methods for identifying genes with differential expression in replicated cDNA microarray experiments. (Submitted), 2000.

[11] M. B. Eisen. `ScanAlyze`, 1999. `http://rana.Stanford.EDU/software/` for software and documentation.

[12] M.B. Eisen and P. O. Brown. DNA arrays or analysis of gene expression. *Methods in Enzymology*, 303, 1999.

[13] GSI Lumonics. *QuantArray Analysis Software, Operator's Manual*, 1999.

[14] R. Ihaka and R. Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5:299–314, 1996.

[15] X. J. Lou. Human primary cell gene expression monitoring using cDNA microarrays. Abstract for Microarray Algorithms and Statistical Analysis: Methods and Standards, Lake Tahoe Center, November 1999.

[16] L.Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. Pat. Anal. Machine Intell.*, 13:583–598, June 1991.

[17] M. Schena, editor. *DNA Microarrays : A Practical Approach*. Oxford University Press, 1999.

[18] M. Schena, editor. *Microarray Biochip Technology*. Eaton, 2000.

[19] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer, 1999.

[20] Y. H. Yang, S. Dudoit, P. Luu, and T. P. Speed. Normalization for cDNA microarray data. (manuscript in preparation), 2000.

Table 1: Segmentation methods and examples of algorithms and software implementations.

Methods	Software/algorithms
Fixed Circle	*ScanAlyze*, *GenePix*, *QuantArray*.
Adaptive Circle	*GenePix*.
Adaptive Shape	*Spot*, region growing and watershed.
Histogram	*QuantArray* and adaptive thresholding.

Table 2: Quantitation output for the mini 3-by-3 spots image from the Cy3 (green) dye shown in Figure 5. The table lists for each spot, the spot foreground intensities estimated by various segmentation methods described in Table 5.

Spot		Segmentation					
R	C	`S.nbg`	`GP.nbg`	`SA.nbg`	`QA.fix.nbg`	`QA.hist.nbg`	`QA.adp.nbg`
1	1	13177	11183	7475	11834	14850	10520
1	2	8000	8867	5396	8355	9034	7559
1	3	3138	2407	2379	3305	3610	3359
2	1	8434	4455	3301	13111	28902	28270
2	2	4119	3891	1722	2471	3696	3596
2	3	15473	18848	8387	13301	20030	17408
3	1	2399	1793	1591	2177	2499	2181
3	2	1246	1166	814	1035	1225	1347
3	3	35959	40980	29819	44318	51014	37473

Table 3: Quantitation output for the mini 3-by-3 spots image from the Cy3 (green) dye shown in Figure 5. The table lists for each spot, the background intensities estimated by various segmentation and background methods described in Table 5.

Spot		Background						
R	C	`S.valley`	`S.morph`	`GP`	`SA`	`QA.fix`	`QA.hist`	`QA.adp`
1	1	463	262	522	1044	470	463	1494
1	2	463	262	534	1075	429	428	528
1	3	481	262	529	580	488	422	1086
2	1	481	262	547	1036	475	400	8078
2	2	463	262	536	1073	457	383	611
2	3	562	262	513	585	487	409	561
3	1	481	262	548	1084	448	457	502
3	2	483	262	546	1082	429	408	541
3	3	483	262	535	539	548	455	651

Table 4: An example of parts of the quantitation output produced by *Spot* for the mini 3-by-3 spots image shown in Figure 5. For the Cy3 (green) channel, "Gmean" refers to the average of the foreground pixel values, "GIQR" refers to the interquartile range or IQR (a robust measure of spread) of the logged foreground pixel values, and "Gvalley" and "morphG" refer to the background intensity estimates from the local background valley method `S.valley` and the morphological opening method `S.valley`. "Rmean", "RIQR", "Rvalley", and "morphR" refer to similar foreground and background measures for the Cy5 (red) channel. The log-ratios for each spot are calculated as $log_2 \frac{Gmean-morphG}{Rmean-morphR}$ and are stored in the column "lratio". "Area" measures the number of foreground pixels for each spot. "Circularity" is defined as $\frac{4 \cdot \pi Area}{Perimeter^2}$ and provides a measure of the circularity of each segmented spot mask. For a perfectly circular spot, the circularity measure is 1. In practice, we will observe circularity scores which are greater than 1. This happens mainly for very small spots and is due the nature of the perimeter algorithm currently used. A more precise perimeter algorithm will be implemented in the future. The signal to noise ratios for the Cy3 and Cy5 channels are stored in "Gs2n" and "Rs2n", respectively. These are the ratio between background corrected spot intensity and background.

Spot_ID	Gmean	GIQR	Rmean	RIQR	Gvalley	Rvalley
1	13177	0.62	10327	0.47	463	1282
2	8000	0.14	5070	0.13	463	1185
3	3138	0.27	3211	0.19	481	1213
4	8433	0.25	8635	0.40	481	1265
5	4118	0.35	3776	0.22	463	1265
6	15473	0.90	5603	0.65	456	1231
7	2399	0.21	2995	0.16	481	1253
8	1245	0.27	2107	0.11	483	1265
9	35959	0.81	31807	0.73	483	1247

morphG	morphR	lratio	area	circularity	Gs2n	Rs2n
260	1208	-.50	31	.88	5.62	2.92
261	1174	-.99	32	.83	4.88	1.73
261	1146	-.48	56	.64	3.46	0.84
260	1208	-.13	11	1.38	4.96	2.62
262	1163	-.57	16	1.03	3.88	1.14
262	1144	-1.77	40	.74	5.86	1.95
250	1185	-.26	30	.94	3.03	0.57
262	1157	-.12	20	1.12	1.91	-0.42
262	1164	-.22	61	.85	7.09	4.68

Table 5: Description of image analysis methods used in the comparison study.

Name	Description
`S.nbg`	Software: *Spot*. Segmentation: Seeded region growing. Background : None.
`GP.nbg`	Software: *GenePix*. Segmentation: Proprietary algorithm that results in adaptively sized circles. Background : None.
`SA.nbg`	Software: *ScanAlyze*. Segmentation: Fixed circles, 10 pixels in diameter. Background : None.
`QA.fix.nbg`	Software: *QuantArray*. Segmentation: Spot intensity is the mean of pixel values between the 45th and 85th percentile within a fixed circle of 9 pixels in diameter. Background : None.
`QA.hist.nbg`	Software: *QuantArray*. Segmentation: Spot intensity is the mean of pixel values between the 80th and 95th percentile of a 11-by-11 pixels square. Background : None.
`QA.adp.nbg`	Software: *QuantArray*. Segmentation: Chen's method, with a circular target mask of 10 pixels in diameter and a 0.001 p-value cut-off. Background : None.
`S.valley`	Software: *Spot*. Segmentation: Seeded region growing. Background : Median from "valley of spot".
`GP`	Software: *GenePix*. Segmentation: Proprietary algorithm that results in adaptively sized circles. Background : Median from "valley of spot".
`SA`	Software: *ScanAlyze*. Segmentation: Fixed circles, 10 pixels in diameter. Background : Median value in local square region.
`QA.fix`	Software: *QuantArray*. Segmentation: Spot intensity is the mean of pixel values between the 45th and 85th percentile within a fixed circle of 9 pixels in diameter. Background : The mean of pixel values between the 5th and 55th percentile of the background mask. The background mask is the region between two circles with diameter of 11 and 13 pixels, and concentric with the spot mask.
`QA.hist`	Software: *QuantArray*. Segmentation: Spot intensity is the mean of pixel values between the 80th and 95th percentile of a 11-by-11 pixels square.

continued on next page

continued from previous page	
Name	Description
	Background : The mean of pixel values between the 5th and 20th percentile of a 11-by-11 pixels square.
`QA.adp`	Software: *QuantArray*. Segmentation: Chen's method with a circular target mask of 10 pixels in diameter and a 0.001 p-values cut-off. Background: The mean of the median 8 background pixels in the background mask (as shown by the pixels included between the two green concentric circles in Figure 7).
`S.morph`	Software: *Spot*. Segmentation: Seeded region growing. Background : Based on morphological opening. The structuring element is a square region with sides of length 2.5 times the approximate spot to spot separation.
`S.const`	Software: *Spot*. Segmentation: Seeded region growing. Background : Constant subtraction; the constant value is the 3rd percentile of all the foreground spot intensities.

Table 6: Correlation between background corrected signal intensities and background intensities for spots with foreground intensities below the median (approximately 3000 spots). The data are from the scan images for the Cy3 (green) dye for knock-out (KO) mice 5 and 8 in experiment (A).

Methods	K0 #5	KO #8
S.valley	-0.23	-0.06
S.morph	-0.02	0.18
SA	-0.17	0.05
GP	-0.18	-0.02
QA.fix	-0.42	-0.40
QA.hist	0.09	0.18
QA.adp	-0.14	-0.04

Table 7: Within slide variability for the 8 replicated spots in experiment (B). In hybridization *C3K5*, the control mRNA is labeled with Cy3 and the knock-out mRNA is labeled with Cy5. In hybridization *C5K3*, the labeling is reversed. The numbers are medians of the 122 standard deviations of $\log_2 R/G$ for the 8 replicated spots, multiplied by 100 and rounded to the nearest integer.

Background methods	Segmentation methods	*C3K5*	*C5K3*
Local background	S.valley	17	21
	GP	11	11
	SA	12	14
	QA.fix	18	23
	QA.hist	9	8
	QA.adp	27	26
Morphological opening	S.morph	9	9
Constant	S.const	14	14
No adjustment	S.nbg	6	6
	GP.nbg	7	6
	SA.nbg	6	6
	QA.fix.nbg	7	6
	QA.hist.nbg	7	6
	QA.adp.nbg	14	14

Table 8: Names and adjusted p-values of the 9 genes with largest absolute t-statistics for each of the methods described in Table 5. The first column gives the method name, and columns 2 to 10 give the names and adjusted p-values of the top 9 genes for each of the methods. For example, column 2, with the header "1", gives the adjusted p-values for the gene with the most extreme t statistic. The adjusted p-value calculation is based on an algorithm of Westfall and Young described in Dudoit *et al.* [10]. The symbols "A1", "A3", "SD", "E" and "O" denote apo A1, apo CIII, sterol desaturase, a novel EST and other genes, respectively. The first four were confirmed by RT-PCR (Callow *et al.* [7]).

		1		2		3		4		5		6		7		8		9
`S.nbg`	A1	0	A3	0	A1	0	E	0	SD	0	A1	0	A3	.01	SD	.01	O	.57
`SA.nbg`	SD	0	A1	0	E	0	A1	0	A3	0	A1	.02	SD	.03	O	.15	A3	.16
`GP.nbg`	A1	0	A1	0	A3	0	SD	0	E	.01	SD	.01	A1	.01	A3	.05	O	.22
`QA.fix.nbg`	A1	0	SD	0	E	0	A1	0	A3	0	SD	.02	A1	.03	A3	.08	O	.11
`QA.hist.nbg`	SD	0	E	0	A1	0	A1	0	A3	0	A3	.02	SD	.03	A1	.03	O	.11
`QA.adp.nbg`	A1	0	A3	0	SD	0	A1	0	SD	.04	E	.05	O	.36	A1	.53	O	.55
`S.valley`	A1	0	SD	0	A1	0	A3	0	A3	0	E	0	A1	.01	SD	.01	O	.26
`SA`	SD	0	A1	0	A1	0	A3	0	A1	0	E	0	A3	0	SD	.01	O	.47
`GP`	SD	0	A1	0	A1	0	A3	0	A3	0	SD	0	E	.01	A1	.02	O	.19
`QA.fix`	A1	0	A1	0	A3	0	SD	0	SD	.01	A3	.01	A1	.03	O	.27	E	.41
`QA.hist`	SD	0	A3	0	A1	0	A1	0	E	0	A3	.01	SD	.01	A1	.02	O	.14
`QA.adp`	A1	0	A3	0	SD	0	A1	0	SD	.03	A1	.50	O	.50	E	.55	O	.67
`S.morph`	A1	0	SD	0	A1	0	A3	0	A1	0	E	0	A3	0	SD	.01	O	.60
`S.const`	A1	0	A1	0	SD	0	A3	0	A1	0	A3	0	E	.01	SD	.01	O	.28

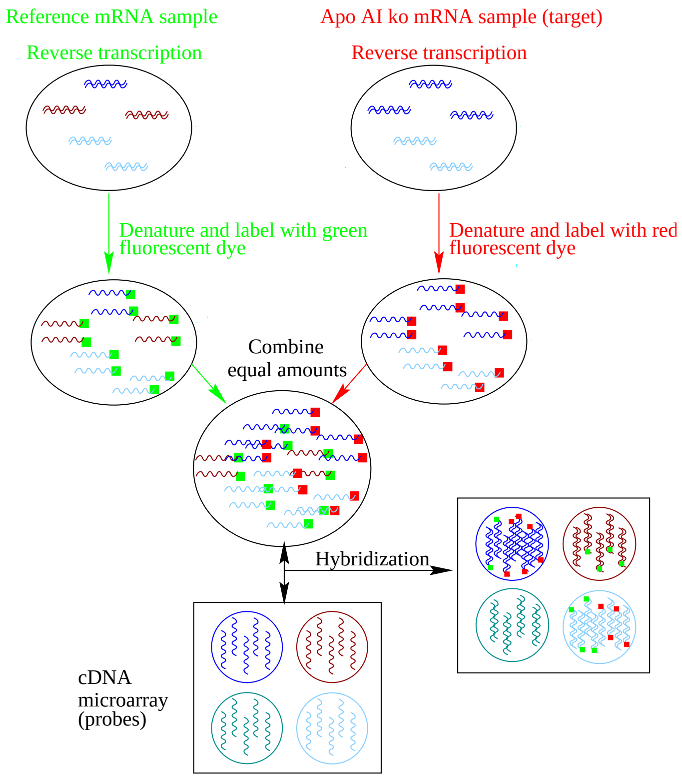


Figure 1: cDNA microarray experiment for apo AI knock-out mice. For each apo AI knock-out mouse, target cDNA is obtained from liver mRNA by reverse transcription and labeled using a red-fluorescent dye (Cy5). The reference sample (green-fluorescent dye Cy3) used in all hybridizations is prepared by pooling cDNA from the 8 C57Bl/6 control mice. The two target samples are mixed and hybridized to a microarray containing 5,548 cDNA probes. Following this competitive hybridization, the slides are imaged using a scanner and fluorescence measurements are made separately for each dye at each spot on the array.

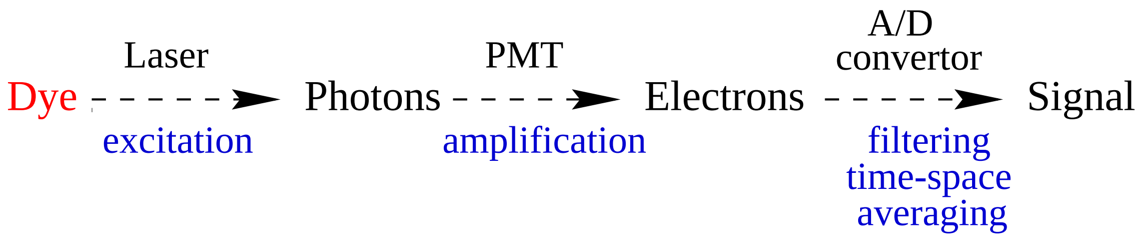


Figure 2: Inside a scanner: diagram summarizing the different processes involved in the imaging of a hybridized slide. The fluorescence dyes absorb energy from the excitation light given out by the laser and emit photons. A PMT detector then converts and amplifies the photons to electrons. An A/D converter finally converts the signal into a digital signal.

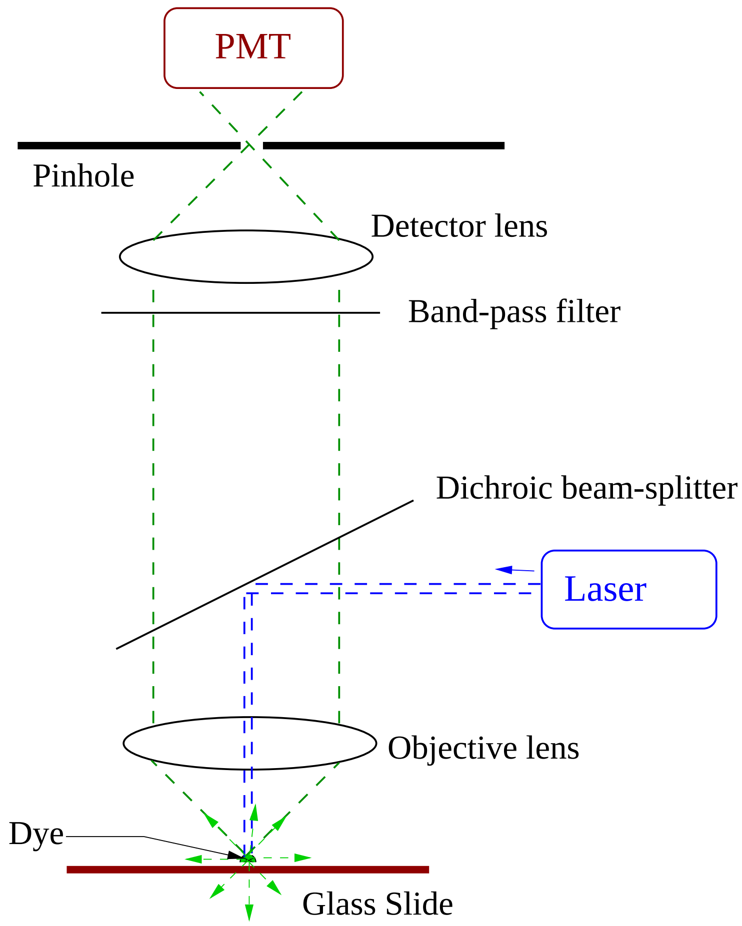


Figure 3: Diagram of a standard confocal scanning arrangement for a microarray scanner.

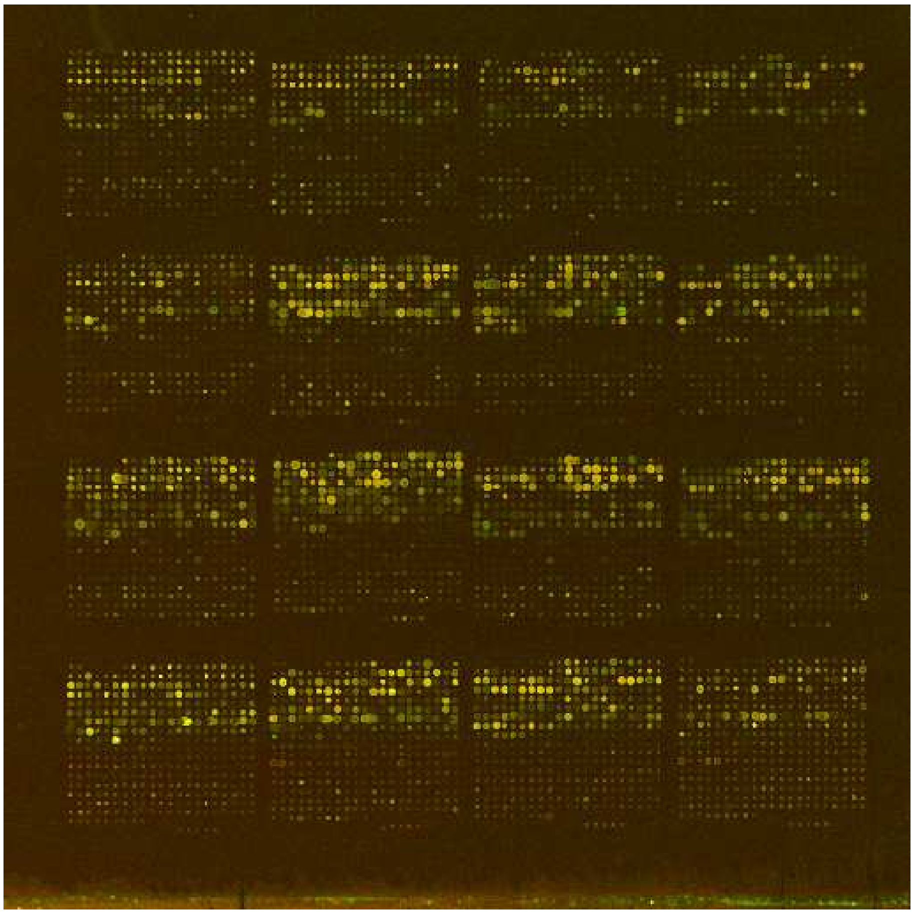


Figure 4: apo AI. RGB image for visualizing the results from the microarray experiment for knock-out mouse #8. For display purposes, the two 16-bit TIFF images (scan output from the Cy3 and Cy5 channels) were compressed into 8-bit images using a square root transformation. This transformation is required in order to display the fluorescence intensities for both wavelengths using a 24-bit composite RGB overlay image. In this RGB image, blue values (B) are set to zero, red values (R) are used for the Cy5 intensities, and green values (G) are used for the Cy3 intensities. Bright green spots represent genes under-expressed in the knock-out mouse, bright red spots represent genes over-expressed in the knock-out mouse, and yellow spots represent genes with similar expression in the knock-out mouse and the reference sample. The coordinates of the three apo AI clones are (2,2,8,7), (4,1,8,6), and (3,3,8,5), where, for example, (2,2,8,7) is the spot in row 8 and column 7 of the spot matrix which is in row 2 and column 2 of the grid matrix.

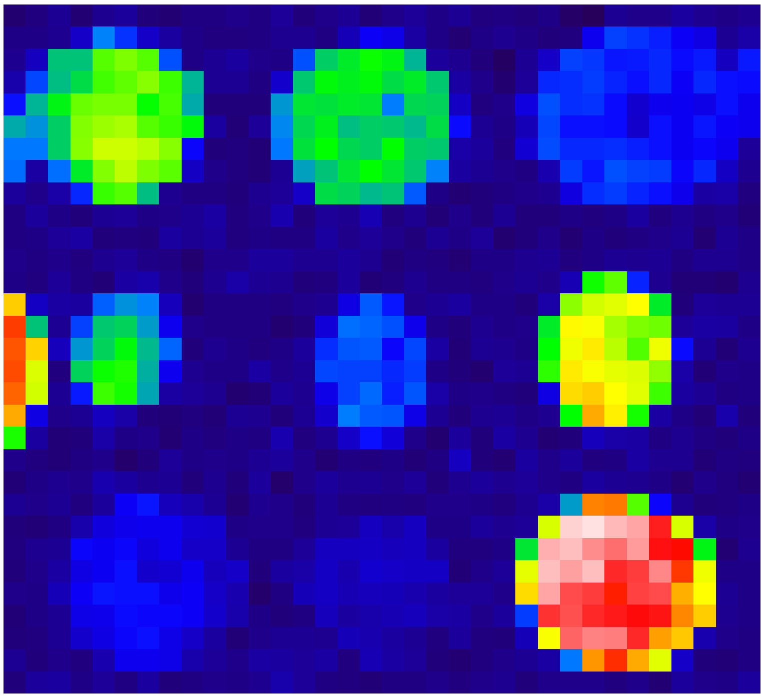


Figure 5: Small portion of the scanned image from the green (Cy3) channel for knock-out mouse #4 in experiment (A). This image displays nine spots using a “rainbow” colour map, where the blue end of the spectrum represents low pixel values and the red end of the spectrum represents high pixel values. Note the different sizes and shapes of the spots.

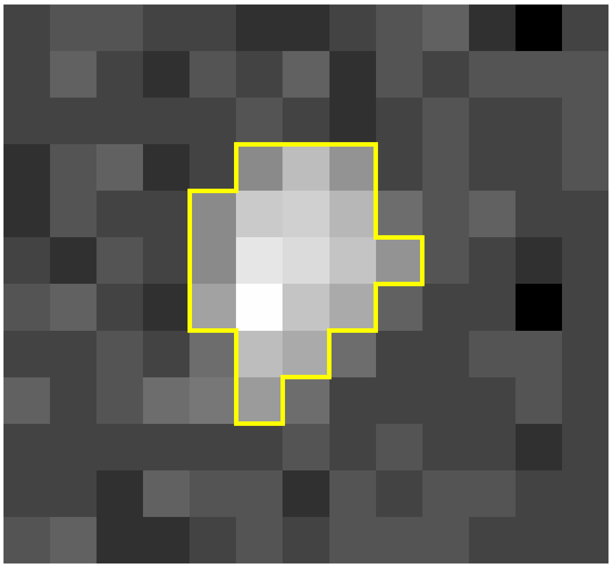


Figure 6: An example of a non-circular shaped spot. The yellow line shows the result of the SRG segmentation. The pixels inside the yellow line are classified as foreground and the other pixels are classified as background.

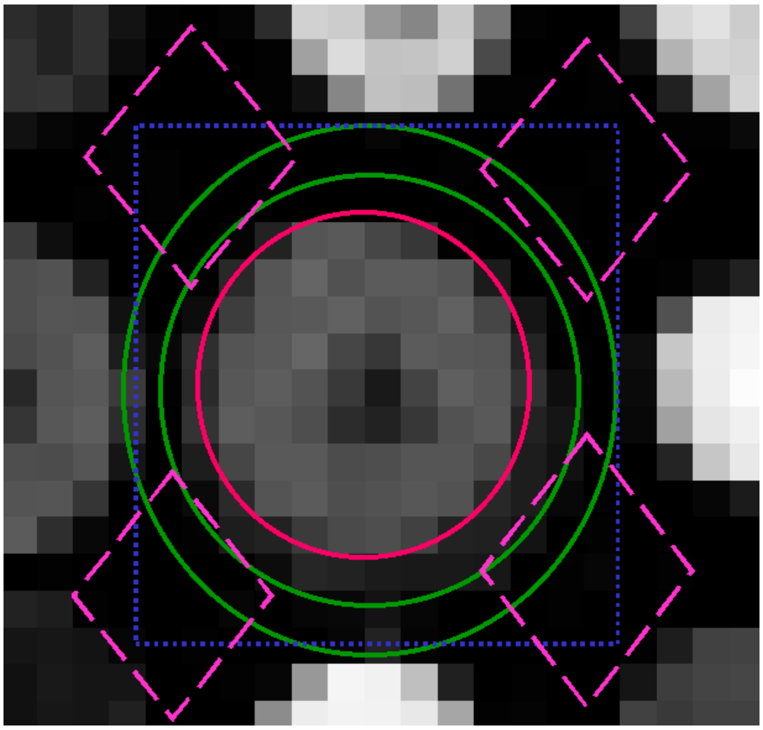


Figure 7: Image illustrating different background adjustment methods. The region inside the red circle represents the spot mask and the other regions bounded by coloured lines represent regions used for local background calculation by different methods. Green: used in *QuantArray*; blue: used in *ScanAlyze*; and pink: used in *Spot*. This image is from KO mouse #8 in experiment (A).

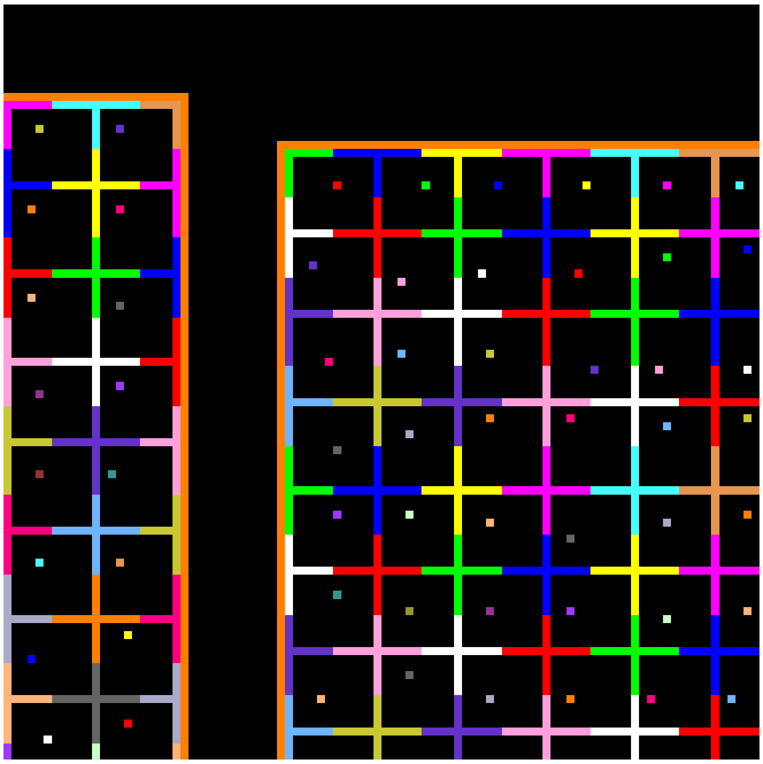


Figure 8: Image illustrating the selection of foreground and background seeds. Foreground seeds are chosen by finding the maximum of the combined intensity surface over a small region centred within the square (single point within the square). The background seeds are constructed as *crosses* based on the fitted background grid.

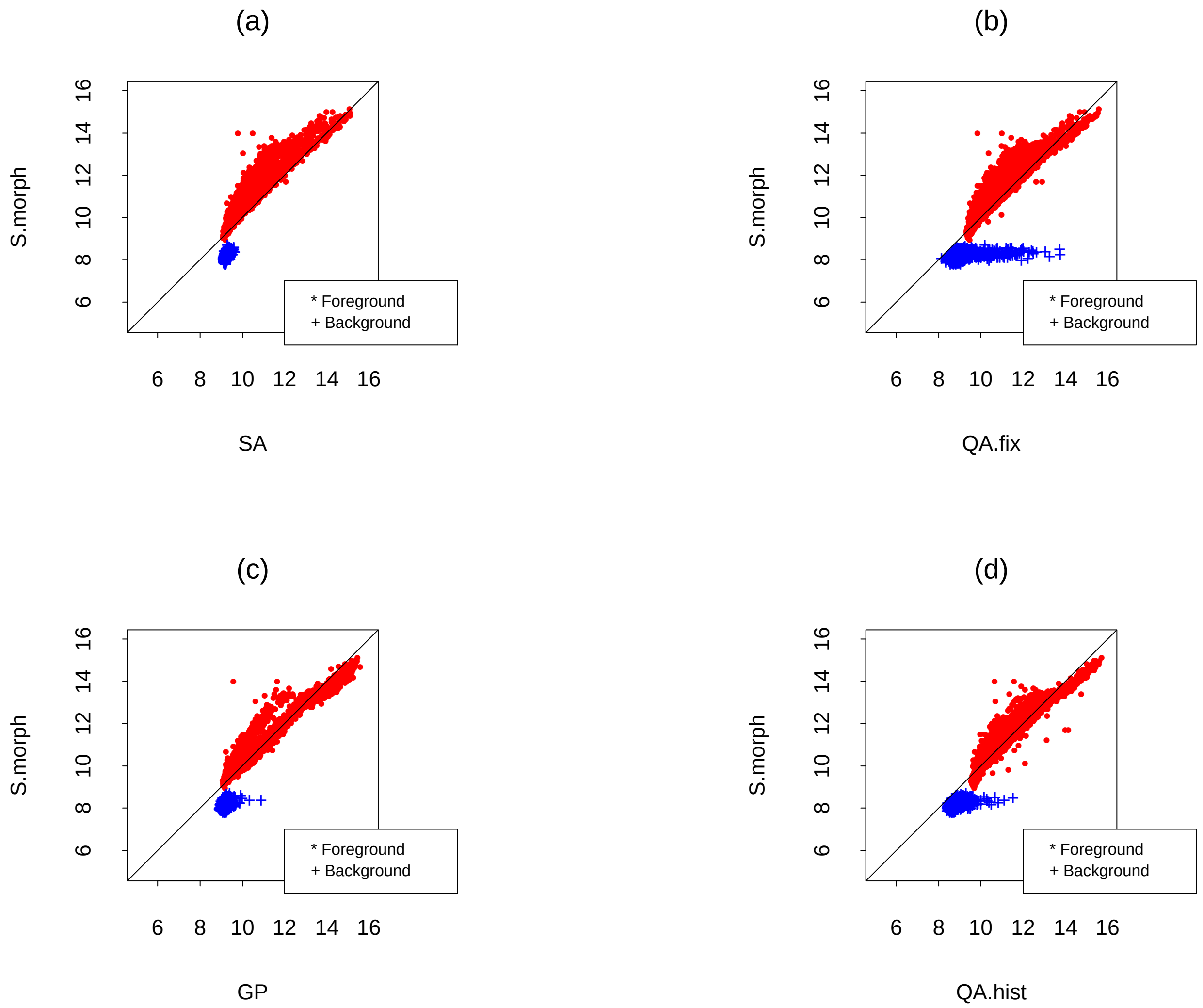
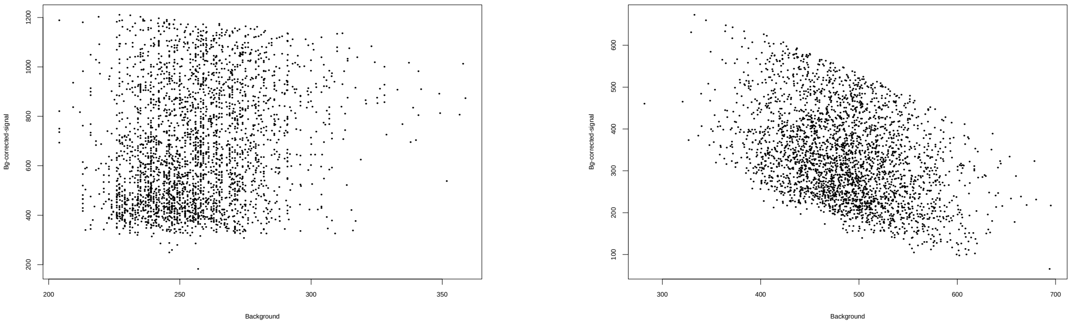


Figure 9: Scatter-plots of foreground (red stars) and background intensities (blue crosses) for (a) `S.morph` vs. `SA`; (b) `S.morph` vs. `QA.fix`; (c) `S.morph` vs. `GP`; and (d) `S.morph` vs. `QA.hist`. The intensities were log transformed (base 2) and are from the scanned Cy3 microarray image for knock-out mouse #8 in experiment (A).



(a) (b)

Figure 10: Plot of background corrected intensities versus background intensities for (a) the morphological background adjustment method of *Spot* (`S.morph`); (b) `QA.fix`. The data are from the scanned Cy3 microarray image for knock-out mouse #8 in experiment (A). Only values from the lower half of the foreground intensity distribution are displayed.

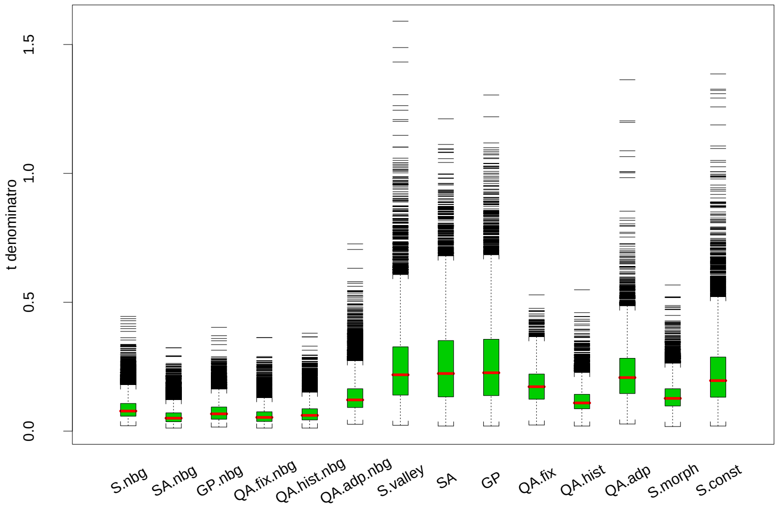


Figure 11: Comparison of the t-denominators (estimating between slide variability) for different image analysis methods in the apo AI experiment (A).

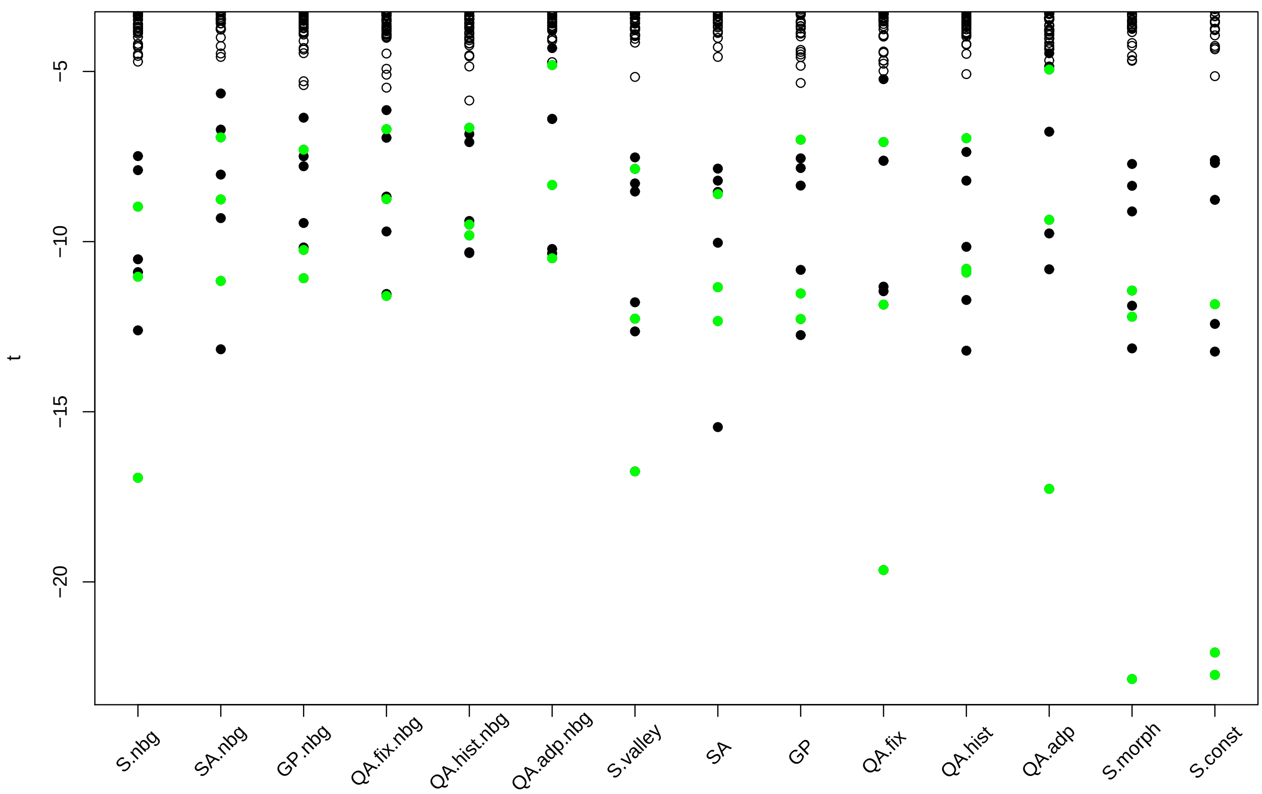


Figure 12: Plot of t-values for different image processing methods. Green solid circles represent the three apo A1 genes. Solid black circles represent the other five genes who were confirmed to change using RT-PCR. Empty circles represent the remaining 6376 genes where no effect is expected. Only t-values less than -4 have been shown.