DNA MICROARRAY IMAGE PROCESSING

Peter Bajcsy[1], Lei Liu[2] and Mark Band[2]
[1]National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign (UIUC)
[2]The W. M. Keck Center for Comparative and Functional Genomics, University of Illinois at Urbana-Champaign (UIUC)

1 INTRODUCTION

Microarray data processing spans a large number of research themes starting from (1) microarray image analysis, (2) data cleaning and pre-processing, semantic integration of heterogeneous, distributed bio-medical databases, (3) exploration of existing data mining tools for bio-data analysis, and (4) development of advanced, effective, and scalable data mining methods in bio-data analysis [9]. The objective of any microarray data analysis is to draw biologically meaningful conclusions [12], [38]. In order to support this objective, we will focus on microarray image processing issues in this chapter. We provide an overview of microarray technologies, overall microarray data processing workflow and management, microarray layout and file format, image processing requirements and existing spot variations, and image processing steps. The image processing steps outlined in this chapter include grid alignment, foreground separation, spot quality assessment, data quantification and normalization.

2 MICROARRAY TECHNOLOGIES

Understanding cellular processes and the relationships between cells of differing function and metabolic pathways is essential for the understanding of the life sciences. With the increased availability of genome sequence due to technological and computing advances, recent years have shown a radical change in the way biology is carried out, shifting towards a systems approach as opposed to a focus on individual genes [43]. The accumulation of sequence data for large compliments of genes has set the stage for high throughput technologies for gene expression, gene polymorphisms and DNA copy number variation. Until the end of the last century the ability to measure gene expression or DNA polymorphisms were restricted to individual genes through the traditional separation and hybridization methods of Southern or Northern blots or quantitative or semi-quantitative PCR using radioactive labeling or chemiluminescence [68]. New high throughput methods that have emerged include differential display [52], serial analysis of gene expression (SAGE) [74], massive parallel signature sequencing (MPSS) [18] and DNA microarrays [24]. Over the past 10 years microarray technologies have been integrated into research involving the relationship of genotype and gene expression to disease [38], development [5], environmental stress [49], behavior [76] and evolution [58]. The availability of commercially produced microarrays, production equipment and reagents, and the widespread introduction of academic and industry core facilities has resulted in an exponential increase in publications based on these technologies. For example, a keyword search for "microarray" on the NCBI Pubmed site for the years 1995-2004 brings up only the seminal paper by Schena et al. in 1995 [64] with an increase to 21, 292, 1514 and 3082 for the years 1998, 2000, 2002 and 2004 respectively (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed). Typical applications of

microarrays are constantly evolving and today include gene expression, genotyping with single nucleotide polymorphism (SNP) detection [21], protein binding assays [35], chromatin immunoprecipitation (CHIP) [41], comparative genomic hybridization (CGH) [56], and microRNA detection [6].

DNA microarrays are typically composed of thousands of DNA sequences, called probes, fixed to a glass or silicon substrate. The DNA sequences can be long (500-1500bp) cDNA sequences or shorter (25-70 mer) oligonucleotide sequences. Oligonucleotide sequences can be presynthesized and deposited with a pin or piezoelectric spray or synthesized in situ by photolithographic or ink-jet technologies.

Relative quantitative detection of gene expression or gene copy number can be carried out between two samples on one array or by single samples comparing multiple arrays. In the first, samples from two sources are labeled with different fluorescent molecules (Cy3 and Cy5, or Alexa 555 and Alexa 647) and hybridized together on the same array. The labels Cy3 or Alexa 555 correspond to a green fluorescent wavelength, and Cy5 and Alexa 647 to red wavelength (Cy dyes are made by Amersham, now GE lifescience, the Alexa dyes are molecular probes made by now Invitrogen). The array is then scanned by activation with lasers at the appropriate wavelength to excite each dye. The relative fluorescence between each dye on each spot is then recorded and a composite image may be produced. The relative intensities of each channel represent the relative abundance of the RNA or DNA product in each of the two samples. The alternative approach is to label each sample with the same dye and hybridize to separate arrays. The absolute fluorescent values of each spot may then be scaled and compared with the same spot between both arrays.

The discovery of novel technologies has led to an increase in the number commercial companies offering off the shelf or custom designed arrays. Almost all are based on in situ oligo synthesis or deposition. Examples of these technologies include chips produced by Affymetrix (http//:www.affymetrix.com) using fixed masks with photolithography, NimbleGen (http//:www.nimblegen.com) using photolithography with a digital micromirror device (DMD), Agilent (http://www.chem.agilent.com) using inkjet and phosphoramidite chemistry, and Combimatrix (http://www.combimatrix.com) using semiconductors for in situ synthesis. Although commercial arrays are in general more expensive on a per unit basis than those produced in core or individual labs they generally offer much more stringent quality control and uniformity. The choice of using a commercial source or producing ones own arrays lies in a number of factors including the number of arrays planned in an experiment, the organism chosen as a model and the amount of labor, and cost, that can be allocated to a project.

The basic methods for extracting data from a microarray image involve identification and measurement of fluorescent intensity for each individual sequence element on the array. Depending on the particular platform, data acquisition software will need to identify the array format, including the array layout, spot size and shape, spot intensities, distances between spots, resolution, and background fluorescence. Many different factors can influence the quality of an image and the complexity of image analysis [81]. A few commercial applications such as the Affymetrix GeneChip adhere to strict protocols and conditions which have been standardized and optimized. However, many other technologies may utilize different components and protocols for array production, sample labeling, hybridization and image acquisition which introduce many sources of variation.

Printing parameters such as pin size and shape, printing speed, temperature and humidity, printing buffers and deposition surface will all affect the size and morphology of the individual spots. The type of glass and coating, blocking agents, hybridization and wash buffers will all affect background fluorescence. All of these and many other factors must be optimized to a particular technology and even to a particular experiment. Image analysis programs must be easily adapted to these varying parameters.

3 MICROARRAY DATA PROCESSING WORKFLOW AND MANAGEMENT

3.1 Microarray Data Processing Workflow

Given a particular microarray technology, microarray images are generated by scanners using confocal laser microscopes. Each microarray image is a representation of the scanned microarray slide with several blocks of 2D arrays. The task is "How can one draw biologically meaningful conclusions based on microarray image data and information extracted about gene expression levels?"

Since the invention of microarray technology in 1995 [64], researchers developed several microarray image processing methods, statistical models and data mining techniques that are specific to DNA microarray analysis [59]. These analyses are usually part of a microarray data processing workflow that includes, grid alignment, spot segmentation, quality assurance, data quantification and normalization, identification of differentially expressed genes and their significance testing, and data mining. An example of microarray data processing workflow is illustrated in Figure 1. The subset of

image processing steps is enclosed with a dashed line in Figure 1. The goal of microarray image analysis steps is to extract intensity descriptors from each spot that represent gene expression levels and input features for further analysis. Biological conclusions are then drawn based on the results from data mining and statistical analysis of all extracted features.

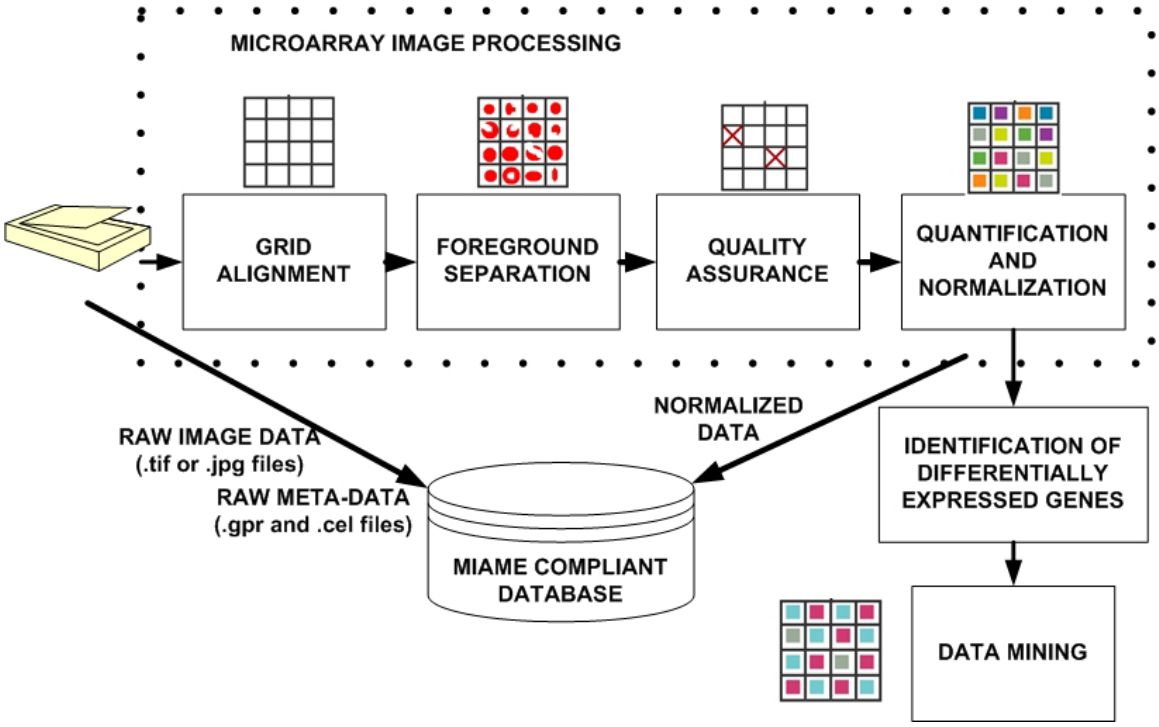

Figure 1: Microarray data processing workflow. The diagram stresses the requirement to archive both raw and processed data.

In this data processing workflow, one should be aware of the nature of microarray measurements. The raw and processed microarray measurements are not expressed in any objective unit but in relative intensity changes using a reference that is rarely standardized between experiments. Furthermore, different microarray platforms and experimental designs generate microarray data with various layouts. In addition, image

processing parameters, normalization techniques and other statistical analyses may vary for each batch of data. Thus, it is critical to adopt standards that allow objective comparisons of (a) microarray data and (b) processing results in order to support validity of biological conclusions.

A typical microarray experiment should be accompanied by (a) microarray slide layout and the results of image analysis (raw intensity, normalized intensity, normalized ratio), (b) the technology used (e.g., one color, Affymetrix GeneChips, or two color cDNA or Oligo microarray), (c) experiment design (common control, loop design, or complex loop design), and (d) normalization methods for those cases when raw image data are not available. Recording data processing workflow and managing information about data processing would help reducing the cost of unnecessary duplicate experiments as suggested by the microarray standardization efforts [31].

3.2 Data Management

Figure 1 also includes a database labeled as MIAME (Minimal Information About Microarray Experiments) compliant. The standardized database is important from a data management perspective since there is a need for public repositories of microarray data [15]. The functions of the public repositories would be in providing access to supporting data for research and publications based on microarray experiments. Such repositories are under development by the National Center for Biotechnology Information (developed the Gene Expression Omnibus), the DNA Database of Japan, and the European Bioinformatics Institute (developed ArrayExpress). However, it is less clear exactly what information should be stored in such databases. A consortium has already defined the

needs of a database standard to preserve context-rich information of microarray data. Starting from 1999, the Microarray Gene Expression Data Society (MGED) has been working to solve this problem, and the group published the MIAME standard [15]. The current focus is on (a) establishing standards for microarray data annotation and information exchange, (b) facilitating the creation of microarray databases and relevant software tools implementing these standards, and (c) promoting sharing of high quality, well-annotated data within the life sciences community. A long-term goal is to extend the current microarray standardization efforts to other domains of functional genomics and proteomics using high throughput technologies.

The MIAME standard encompasses six areas: (1) Experimental design: the set of hybridization experiments as a whole. (2) Array design: each array used and each element (spot) on the array. (3) Samples: samples used, extract preparation and labeling. (4) Hybridizations: procedures and parameters. (5) Measurements: images, quantification, and specifications. (6) Normalization controls: types, values, and specifications. Each of these microarray areas contains information that can be provided using controlled vocabularies, as well as fields that use free-text format.

There exist MIAME-compliant databases and commercial software packages. For example, a number of existing microarray databases in the public-domain claims MIAME-compatibility, such as BASE [62] (http://base.thep.lu.se), GeneX [54] (http://www.ncgr.org/genex/index.html), and MaxdSQL (http://www.bioinf.man.ac.uk/microarray/maxd). A couple of commercial packages, such as GeneTraffic (http://www.iobion.com), and Partisan ArrayLIMS (http://www.clondiag.com/products/sw/partisan) should be also MIAME-compliant. AS a

result of the standardization efforts by the MGED working groups, microarray data standardization specifications become more accessible, and provide the ground for building integrated microarray databases.

4 MICROARRAY IMAGE LAYOUT AND FILE FORMAT

4.1 Microarray Image Layout

The layout of any microarray image is dependent on (a) the type of equipment used to synthesize the array and (b) considerations for image analysis. In almost all layouts, spots are arranged within a two-dimensional (2D) grid with spot locations defined by row and column or by absolute (X, Y) coordinates. Many commercial technologies may have a fixed layout with image analysis mechanisms optimized to the particular layout, such as the Affymetrix GeneChip system. Affymetrix GeneChips are designed with composite sequences representing a transcript of a gene, generally 11 to 20 short oligo sequences designed from different regions of the same transcript. Each individual oligo from the transcript is synthesized at different locations across the GeneChip in order to compensate for local variation of signal intensity. Signal or foreground intensities from each probe within a transcript are then combined for data analysis. Changes to these layouts can involve large initial investments in mask design and synthesis.

Most spotted microarrays using print pins, inkjet or piezoelectric mechanisms have the flexibility to create multiple layouts. In these cases an array of pins or jets are used. Each printing unit, or pin, will create an individual block of spots. The number of pins

and arrangement in the print head can be changed and will determine the block arrangement on the array. The distance between spots as well as row and column numbers within each block can be controlled through the printing software. A grid analysis file is created containing data related to the number of blocks, rows and columns and distance between blocks; rows, columns and distances between features within blocks; approximate spot diameter, together with the annotation of the genes or product represented by each feature. Most image analysis programs also require coordinates of landmark spots for initial grid alignment. When planning the microarray layout, distinct features which will provide constitutively high fluorescent signals, such as housekeeping genes, may be included in the corners of each grid in order to further enhance automated or manual visual alignment of grids.

4.2 Microarray Image File Formats

Typically, laser scanning of a cDNA or oligo microarray slide generates two 16-bit TIFF files [71]. These two files contain information about fluorescence from red and green dyes. The specification for the TIFF file format version 6.0 is publicly available and the format is suitable for saving 1-bit (binary), 4-bit, 8-bit (byte) and 16-bit (short) data. The choice of 16-bits per pixel is based on the dynamic range of fluorescence measurements and sensitivity of laser scanners. The fluorescence values after amplification and analog to digital conversion should be within the interval [0, 2^16-1 = 65,535], otherwise the high values would be truncated to the maximum (also called pixel saturation). The TIFF file format specification version 6.0 also includes image compression options (lossy Lempel-Ziv and Welch compression, lossless modified Huffman run-length coding). It is not recommended to use any lossy compression in

order to prevent spot information loss, and to avoid increased uncertainty of extracted spot statistics. Similarly, while microarray images are sometimes stored in other very common file formats, for instance, in the JPG file format using the compression algorithm based on discrete cosine transform (DCT) [61], one should be aware that any lossy compression will deteriorate microarray image processing accuracy. It is recommended to use microarray image file formats without lossy image compression.

5 MICROARRAY IMAGE PROCESSING REQUIREMENTS

In order to choose an appropriate image processing approach, one has to understand variations of input microarray images in terms of (1) the image content including foreground and background morphology (e.g., grid layout, spot location, shape and size), and intensity information (e.g., spot descriptors derived from foreground and background intensities), (2) the computer characteristics of input digital images (e.g., number of channels, number of bytes per pixel, file format). Figure 2 shows two examples of microarray images and their very different appearance. These variations have to be compensated by microarray image processing algorithms so that the processing performance meets expected accuracy and speed requirements.

What are our expected accuracy and speed requirements on microarray image processing? To answer this question, we consider an ideal microarray image first. Next, we describe our current understanding of the sources of image variations. Finally, we set the image processing requirements that one should strive to meet.

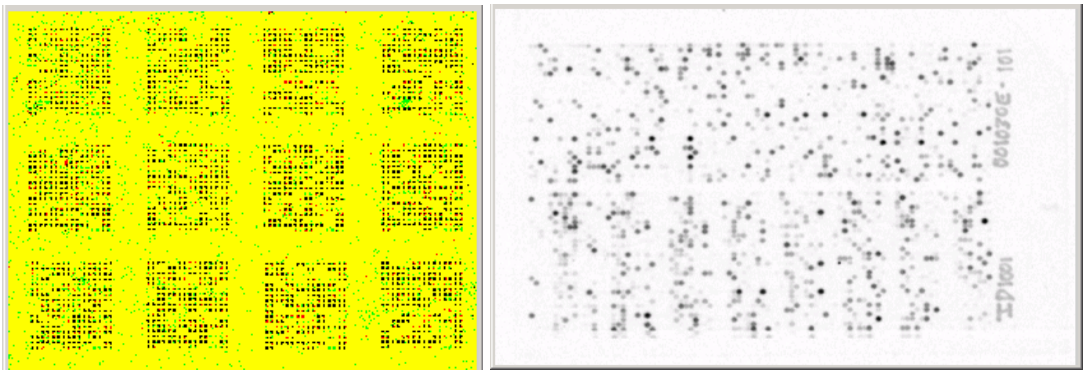

Figure 2: Examples of microarray images with double-fluorescent (left) and radioactive (right) labeled samples that differ in terms of the content (spot geometry, spot size and intensity meaning) and computer characteristics (number of channels and number of bytes per pixel).

5.1 Ideal Microarray Image

First, let us define an "ideal" cDNA microarray image in terms of its image content. The image content would be characterized by deterministic grid geometry, known background intensity with zero uncertainty, pre-defined spot shape (morphology), and constant spot intensity that (a) is different from the background, (b) is directly proportional to the biological phenomenon (up- or –down-regulation), and (c) has zero uncertainty for all spots. Figure 3 shows an example of such an ideal microarray image. While finding such an ideal cDNA image is probably a pure utopia, it is a good starting point for understanding image variations and possibly simulating them [11].

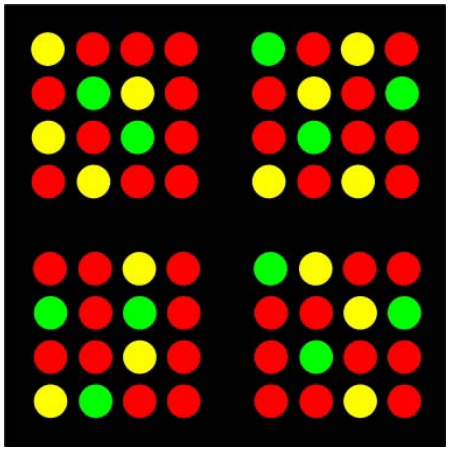

Figure 3: Illustration of an “ideal” microarray image.

Another aspect of an “ideal” cDNA microarray image can be expressed in terms of statistical confidence. If one could not possibly acquire an ideal microarray image, then a high statistical confidence in microarray measurements would be obtained with a very large number of pixels per spot (theoretically it would reach infinity). However, the cost of experiments, the limitations of laser scanners in terms of image resolution, storage of extremely high resolution images and other specimen preparation issues are the real world constraints that have to be taken into account.

The above considerations about an “ideal” microarray image can be used for simulations [11]. Simulations of cDNA microarray images can generate data sets for testing multiple microarray processing algorithms since it is difficult to obtain (a) physical ground truth as an image valuation standard because of the image preparation complexity, and (b) large number of replicates of biological samples as a statistically significant standard because of the cost. In addition, simulations can provide scientific insights about various impacts of microarray preparation fluctuations on the accuracy of final biological conclusions. However, while simulations improve our understanding, they have to be verified by processing real microarray images. Another challenge with

simulations is related to setting input simulation parameters since they might depend on individual laboratory procedures and on each microarray acquisition apparatus.

5.2 Sources of Microarray Image Variations

Next, let us investigate sources of image variations. The cDNA technology is a complex electrical-optical-chemical process that spans cDNA slide fabrication, mRNA preparation, fluorescence dye labeling, gene hybridization, robotic spotting, green and red fluorophores excitation by lasers, imaging using optics, slide scanning, analog to digital conversion using either charge-coupled devices (CCD) or photomultiplier tubes (PMT), and finally image storage and archiving. It is hard to estimate the number of random factors in this complex electrical-optical-chemical process and hence we will list only a few factors. We should perhaps mention that some of the variations are temporally varying, some are ergodic (no sample helps meaningfully predict values that are very far away in time from that sample), and some appear as systematic errors more than as random errors. We overview a few sources of image variations observed in foreground, background and intensity information.

Variations of microarray image channels: Based on the cDNA labeling type used during microarray slide preparation (hybridization), one can obtain, for instance, single-, double- or multi-fluorescent images. Most microarray data contain double-fluorescent images from scanners that operate at two wavelengths, e.g., 532nm (red) and 632nm (green) wavelengths forming two channels shown in Figure 2 left. In general, microarray image data can consist of any number of channels. It is possible to foresee the use of more than two or three channels in future, for example, by using hyperspectral imaging [10].

Another variation of microarray image channels is the storage file format, data compression and data accuracy (number of bytes per pixel). For example, a storage file format with lossy data compression introduces undesirable spatial blur of spots and the microarray image analysis becomes less accurate. Similarly, the number of bytes per pixel has to accommodate the dynamic range of an analog signal produced by the red or green fluorophores excitation due to laser illumination. Dynamic range corresponds to the maximum minus minimum measured amplitude, and any value outside of the range [min, max] will be mapped to either min or max. For a fixed number of bytes and increasing dynamic range, the uncertainty of each intensity measurement increases. In other words, the bins for all analog values converted to the same digital number are becoming wider.

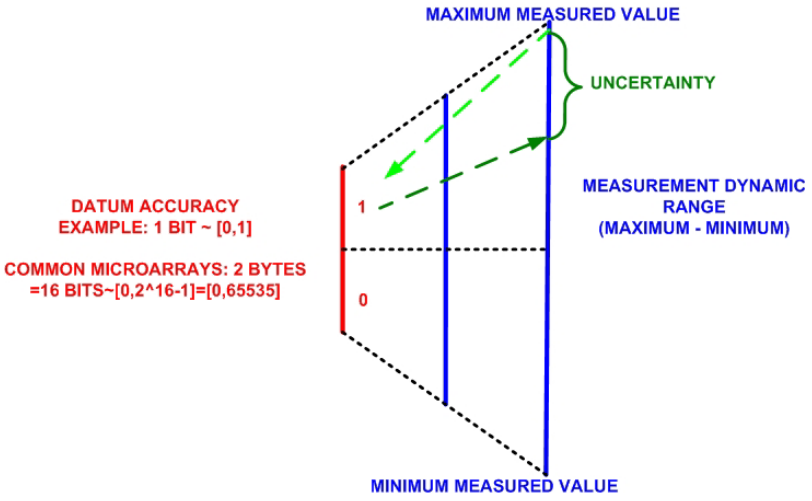

Figure 4: Illustration of data accuracy, uncertainty and dynamic range dependencies.

In general, microarray image processing algorithms should be able to handle any number of input channels, file format and data accuracy. It should be understood that image analysis results will contain some uncertainty due to file storage and datum accuracy constraints.

Variations of grid geometry: A microarray slide preparation should be considered as one source of variation in grid geometry [20], [39], and [76]. For example, it is important to know that if a spotting machine with several dipping pins prints multiple 2D arrays of spots, then the dipping pins might bend over time and cause irregularity in a 2D arrangement of the printed spots [20]. Similarly, any rotational offset of a slide or dipping pins will cause a rotated 2D grid in a microarray image with respect to the image edge. Figure 5 shows an example of a rotated sub-grid with irregularly spaced rows and columns.

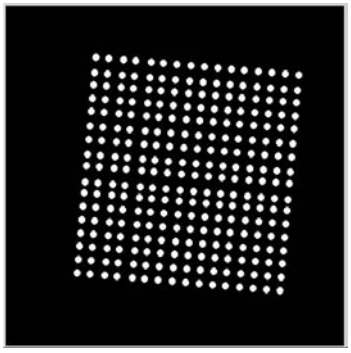

Figure 5: Irregularly spaced and rotated grid geometry of microarray spots.

Other sources of variations in spot locations are the slide material, such as nylon filters, glass slides, and probe types, such as radioactively labeled probes and fluorescently labeled probes [69]. These variations can be caused (a) by mechanical strain (nylon filters), or (b) by low discrimination power for small surface areas (glass slides), strong background signal (fluorescently labeled probes) or strong signal interference of neighboring spots (radioactively labeled spots). The variations due to mechanical strain introduce warping into the grid geometry. It is important to understand the strain extreme cases in order to limit the search space of grid geometry.

Due to a small discrimination power, many spots might not be detected [20]. Figure 2 illustrates that many spots might be missing from a 2D array because spot signals are undistinguishable from the background. The absence of spots in a grid poses a challenge for automated grid alignment in addition to other spot location variations. Clearly, missing spots decrease the likelihood of successfully identifying grid configurations by any data driven approaches because of a smaller amount of grid evidence. For example, a fully automated grid alignment method would fail to detect correctly a grid if one row of spots from the grid along its border would be completely missing (no evidence about the row existence as illustrated in Figure 6).

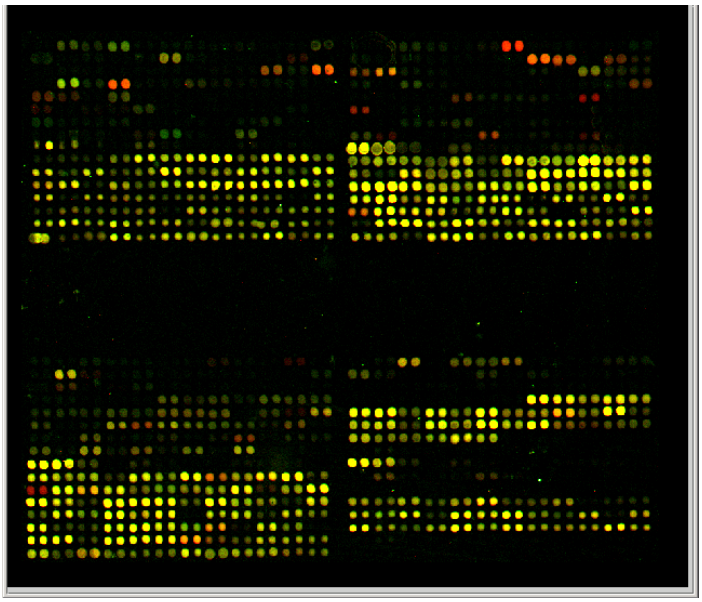

Figure 6: Four sub-grids on one microarray slide. The lower right sub-grid has one less row than other sub-grids.

Variations of background: Background variations occur due to (a) microarray slide preparation (hybridization and spotting errors), (b) inappropriate acquisition procedures (presence of dust or dirt), and (c) image acquisition instruments (non-linearity

of imaging components). While the (a) and (b) types of background variations should be detected by microarray quality assurance (see example in Figure 7), the variation due to image acquisition instruments cannot be removed by a user. Thus, many image processing algorithms compensate for background variations by modeling its probability distribution function (PDF). The most frequent model is the Gaussian PDF (also denoted as Normal PDF) [11]. Other statistical models to consider would be a uniform PDF or a functional PDF depending on the observed properties of acquired images. For instance, a functional PDF could simulate a positive or negative slant surface function (background intensity shading) that would be combined with spike noise, where spike noise intensities follow an exponential distribution. Figure 8 shows background examples that could be modeled by Normal or Student's t PDF models. It is also necessary to mention that while all channels of microarray images might follow the same PDF, each channel would likely have different parameters for the chosen PDF model.

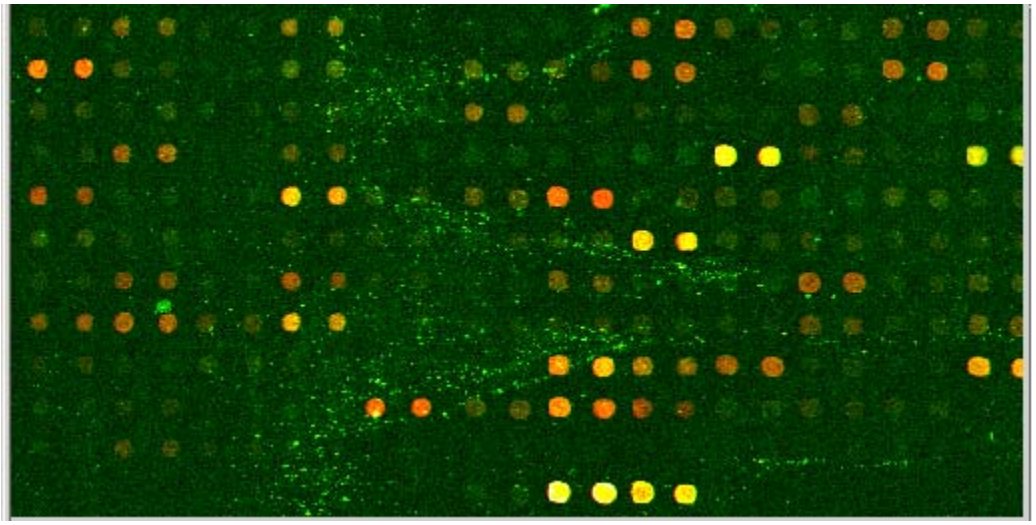

Figure 7: Background variation due to slide washing that should be detected by quality assurance.

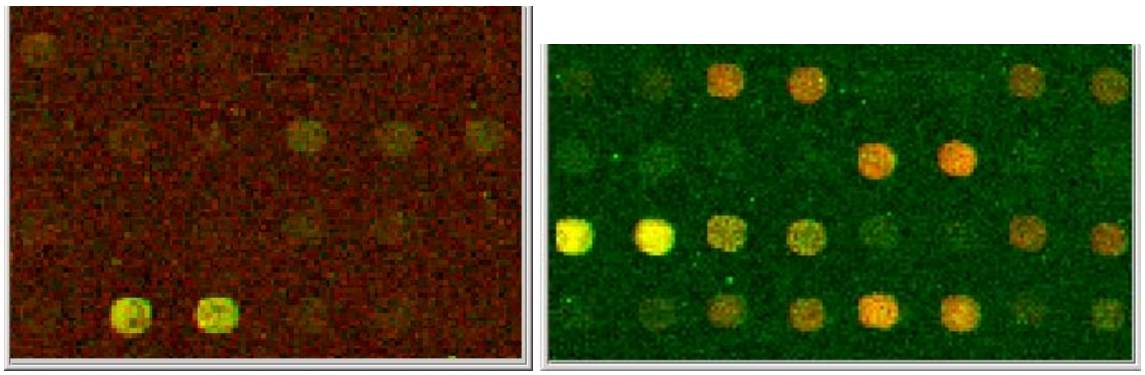

Figure 8: Examples of background noise that could be modeled with PDF models of noise. (Normal PDF – left and Student's t PDF – right).

Variations of spot morphology: Another issue to mention is the shape of microarray grid elements (or grid shape primitives). Although the majority of current cDNA microarray imagery is produced with circular spots as shape primitives, one can find the use of other primitive shapes, e.g., lines or rectangles (see the CLONDIAG chip [23]). It is very likely that other primitive shapes than a round spot shape will be used in microarray technology in the future. Figure 9 shows examples of rectangular and triangular shapes.

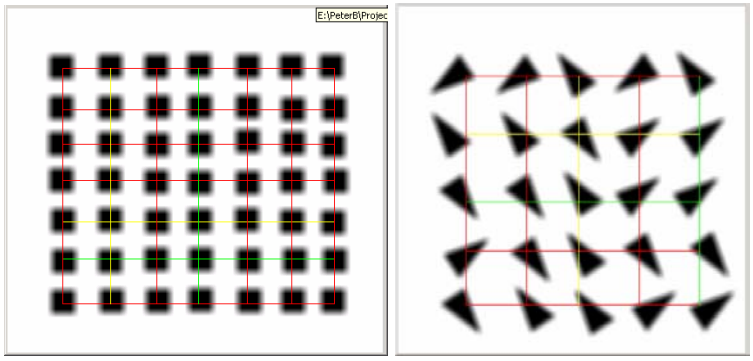

Figure 9: Examples of spot morphologies other than circular.

For the currently most common circular spots, there exists a large number of shape deviations (equals to the total number of foreground and background pixel

combinations inside of a grid cell). Figure 9 shows a few classes of morphological deviations as found in microarray images. There are many more spot deviations that have to be analyzed during spot quality assessment in order to determine a validity of measured spot information and our confidence in deriving any conclusions based on the spot measurement. The spot deviation analysis helps identifying success and failure of a particular spot experiment.

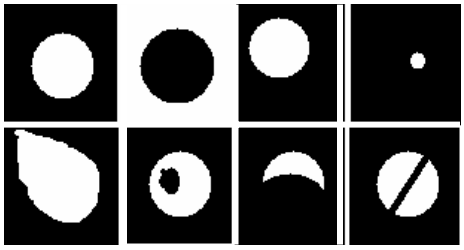

Figure 10: Spatial and morphological variations of spots (from left to right, top row first): (a) a regular spot, (b) an inverse spot or a ghost shape, (c) a spatially deviating spot inside of a grid cell, (d) a spot radius deviation, (e) a tapering spot or a comet shape, (f) a spot with a hole or a doughnut shape, (g) a partially missing spot and (h) a scratched spot.

Variations of foreground and background intensities: Foreground and background intensity variations are also present in microarray image analysis due to slide materials and several labeling techniques. For example, while the fluorescent labeling type leads to microarray images with dark background and bright spots (signal), other labeling types with or without radio-isotopic labels lead to images with bright background and dark spots (see Figure 2 right). A slide material introduces another intensity variation, for example, coated glass slides or nylon membrane or silicon chips. One should understand that it is the background and foreground intensity difference that is relevant to the biological meaning. However, the range of the intensity difference (max – min) and the amplitude of background and foreground variations affect the

discrimination of these two classes, as well as our confidence in accurate separation of background and foreground.

Although we described variations of background and the dark-bright schemes for background and foreground, we did not address the issue of foreground spot intensity variations. The reason for this is that microarray images often represent experiments of a discovery type. When discovering biological properties, one cannot predict measurement outcomes such as spot intensity profiles. Thus, one should only adjust parameters of measurement instruments to fully cover the dynamic range of spot intensities so that intensity values are not saturated and possibly discernable from others. As of now, intensities of each spot are modeled according to our previously described ideal microarray image but future research might reveal additional information in the intensity profiles of individual spots.

5.3 Summary of Microarray Image Processing Requirements

After reviewing variations of microarray images, one would like to design automated microarray image processing algorithms that are robust to all variations. The robustness would include (1) any number of channels, (2) any storage and computer representation, (3) variable grid and spot locations, (4) unknown background noise, (5) variable background and foreground dark-bright schemes, (6) deviations from spot shapes and (7) deviations from expected spot intensity profiles. Furthermore, the processing algorithms should recognize those cases when missing spots disable automation (or accurate automated image processing) because of the lack of grid evidence.

For anyone who performs scientific experiments with microarray technology, it is important to guarantee microarray image processing repeatability. Assuming that an algorithm is executed with the same data, we expect to obtain the same results every time we perform an image processing step. In order to achieve this goal, algorithms should be "parameter free" so that the same algorithm can be applied repeatedly without any bias with respect to a user's parameter selection. Thus, for instance, any manual positioning of a grid template is not only tedious and time-consuming but also undesirable since the grid alignment step cannot then be repeated easily. A concrete example of the repeatability issues is presented in [50], where authors compared results obtained by two different users from the same slide (optic primordial dissected from E11.5 wild-type and aphakia mouse embryos) while using the ScanAlyze software package [34]. Each user provided the same input about grid layout first, and then placed multiple grids independently and refined the spot size and position. The outcome of the comparison led up to two-fold variations in the ratios arising from the grid placement differences.

Finally, the amount of microarray image data is growing exponentially and so one is concerned about preparing sufficient storage and computational resources to meet the requirements of end users. For example, finding a grid of spots can be achieved much faster from a sub-sampled microarray image (e.g., processing one out of 5x5 pixels), but the grid alignment accuracy would be less than if the original microarray image had been processed. There are clearly tradeoffs between computational resources (memory and speed/time) and alignment accuracy given a large number of microarray images [8]. While this issue might be resolved without any accuracy loss by using either supercomputers or distributed parallel computing with grid-based technology [37], it

might still be beneficial to design image processing algorithms that could incorporate such resource limitations.

6 GRID ALIGNMENT METHODS

A grid alignment (also known as addressing or spot finding [14] or gridding [76]) is one of the processing steps in microarray image analysis that registers a set of unevenly spaced, parallel and perpendicular lines (a template) with the image content representing a two-dimensional (2D) array of spots [8]. The registration objective of the grid alignment step is to find all template descriptors, such as, line coordinates and their orientations, so that pairs of perpendicular lines intersect at the locations of a 2D array of spots in a microarray scan. Furthermore, this step has to identify any number of distinct grids of spots in one image.

There are two views on microarray grid alignment. First, grid alignment methods could be viewed in terms of automation as manual, semi-automated and fully automated [29, Chapter 3], [46, Chapter 6]. Second, grid alignment techniques could be viewed in terms of their underlying image analysis approaches as template-based and data-driven [8].

6.1 Automation Level of Grid Alignment Methods

Manual grid alignment methods: Given the fact that one expects a spot geometry to be very similar to a grid (or a set of sub-grids), a manual alignment method is based on a grid template of spots. A user specifies dimensions of a grid template and a radius of each spot to form a template. Computer user interfaces like a computer mouse

are available for adjusting the pre-defined grid template to match the microarray spot layout.

To compensate for many microarray image variations described in the previous section, one could possibly obtain "perfect" grid alignment assuming that human-computer interface (HCI) software tools are built for adjusting shape and location of each spot individually. It is apparent that this approach for grid alignment is not only very time consuming and tedious, but also almost impossible to repeat or use for high-throughput microarray image analysis.

Semi-automated grid alignment methods: In general, there are some parts of grid alignment that can be reliably executed by computers, but other parts that are dependent on user's input. One example would be a manual grid initialization (selection of corner spots, specification of grid dimensions), followed by automated search for grid lines and grid spots [76]. The automated component can be executed by using either a grid template that is matched to the image content with image correlation techniques, or a data-driven technique that assumes intensity homogeneous background and heterogeneous foreground. The benefits of semi-automated grid alignment methods include reductions of human labor and time, and an increase of processing repeatability. Nevertheless, these methods might not suffice to meet the requirements of high-throughput microarray image processing.

Fully-automated grid alignment methods: These methods should reliably identify all spots without any human intervention based on one-time human setup. The one-time setup is for incorporating any prior knowledge about an image microarray layout into the grid alignment algorithms in order to reduce their parameter search space.

Many times, the challenge of designing fully-automated grid methods is to identify all parameters that represent prior knowledge and quantify constraints for those parameters. Typically, these methods are data driven and have to optimize internally multiple algorithmic parameters in their parameter search space to compensate for all previously described microarray image variations.

While it is everyone's ultimate goal to design fully automated grid alignment methods, one has to understand that these methods depend entirely on data content. For example, if there is a missing line of spots (spot color is indistinguishable from background) then an algorithm would not be able to find any supporting evidence for a grid line. One approach to this problem is the assignment of algorithmic confidence scores to each found grid. Grids with low confidence can be set aside for further human inspection whereas the grids with high algorithmic confidence can be processed without any human intervention. Another approach is to build into a microarray image some fiduciary spots that could guide image processing and provide a self-correction mechanism.

6.2 Image Analysis Approaches to Grid Alignment

6.2.1 Template-Based Approaches

The template-based approach is the most prevalent in the previous literature and existing software packages, e.g., GenePix Pro by Axon Instruments [4], ScanAlyze [34], or GridOnArray by Scanalytics [65]. Most of the currently available software packages enable manual template matching [4] (GenePix), [34] (ScanAlyze), [20] (Dapple), by adjusting spot size, spot spacing and grid location. Some software products already incorporate an automatic refinement search for a grid location given size and spacing of

spots [4] (GenePix), [57] (QuantArray). The refinement is executed by maximizing correlation of (1) an image template formed based on user's inputs and (2) the processed microarray image over a set of possible template placements (e.g., translated and rotated from the user defined initial position). It is possible to employ deformable templates and Bayesian grid matching [42] to achieve certain data driven flexibility into grid alignment.

The template-based approach is viewed as appropriate if the measured grid geometry does not deviate too much from the expected grid model as defined by a template [65]. If measured spot grids are unpredictably irregular then this approach leads to (a) inaccurate results or (b) unacceptable costs for creating grid templates that would be custom-tuned to each batch of observed grid geometries. An example of alignment inaccuracies is shown in Figure 11.

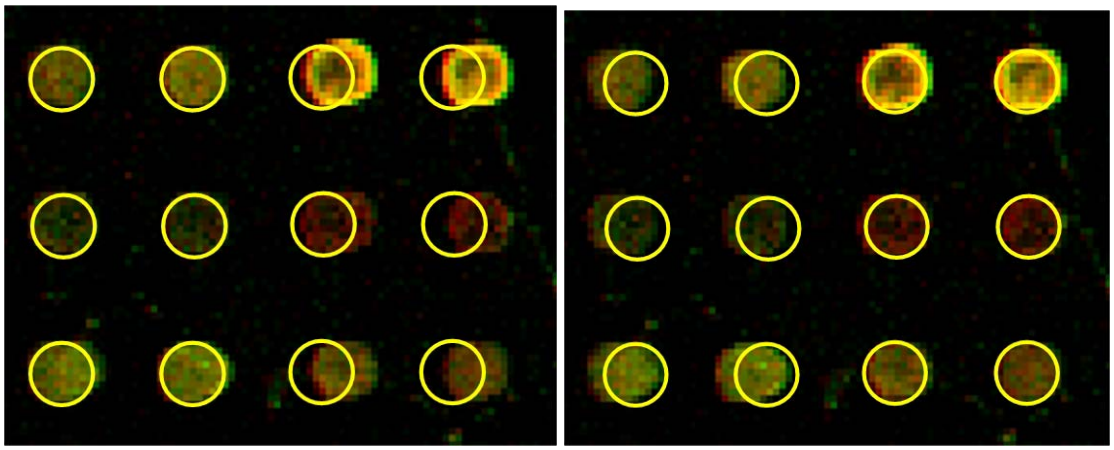

Figure 11: Template-based alignment results obtained by visually aligning the left two columns (left) or the right two columns (right) of microarray spots.

6.2.2 Data-Driven Approaches

There are several components of data-driven algorithms and each component solves one part of the grid alignment puzzle. We overview basic components of such data-driven algorithms for grid alignment.

Finding grid lines: The first "core" component that finds grid lines is (a) based on statistical analysis of 1D image projections [7], [25], [45], [69], or (b) used as part of image segmentation algorithms [48], [53]. The algorithmic approach based on 1D image projections consists of the following steps [8], [69]. First, a summation of all intensities over a set of adjacent lines (rows or columns) is computed and denoted as a projection vector. Second, local extremes (maxima for bright foreground or minima for dark foreground) are detected within the projection vectors. These local extremes represent an approximation of spot centers. The tacit assumption is that the sought lines intersect a large number of high contrast and low contrast areas in contrary to the background that is assumed to be intensity homogeneous with some superimposed additive noise. Third, a set of lines is determined from the local extremes by incorporating input parameters (e.g., number of lines) and by finding consistency in spacing of local extremes. Fourth, all intersections of perpendicular lines are calculated to estimate spot locations. The input microarray intensities can be pre-processed to remove dark-bright schema dependency (e.g., by edge detection [8]), or to enhance contrast of spots (e.g., by matched filtering or spot amplification [14]). Figure 12 illustrates 1D projections derived from a pre-processed image by Sobel edge detection algorithm [61].

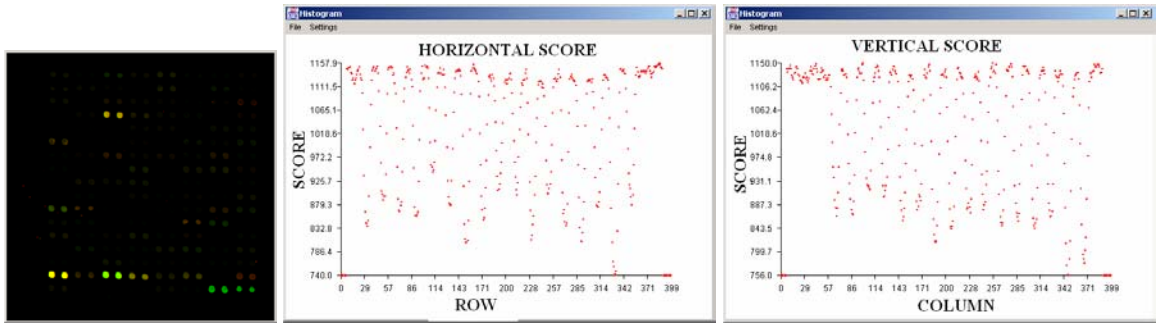

Figure 12: A microarray image (left) and its 1D projection scores (modified summations) derived from the original image after pre-processing by Sobel edge detection.

The other algorithmic approach to finding grid lines that is based on image segmentation [53] uses adaptive thresholding and morphological processing to detect guide spots. The guide spots are defined as the locations of good quality spots (circular in shape, of appropriate size and intensity consistently higher than the background), for instance, the spots in Figure 13. With the help of guide spots and given the information about microarray layout, the final grid can be estimated automatically. The drawback of this approach is the assumption about the existence of guide spots and the absence of spurious "spots" due to contamination. Other segmentation–based approach reported in [48] uses region growing segmentation to obtain partial grids that are then evaluated by grid hypothesis testing.

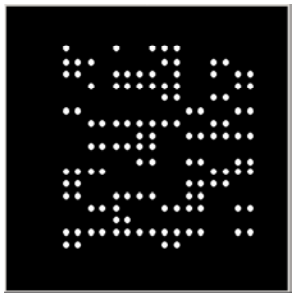

Figure 13: An example of guide spots as used in [53].

Processing multiple channels: The second component of data-driven methods tackles usually the problem of fusing multiple image channels (also called bands). The fusion problem could include cross-channel registration issues since each channel is acquired at a different time, and a spatial offset might occur between the acquisitions. Furthermore, the fusion problem has to bring together either input channels for grid alignment or the results of grid alignment obtained for each channel separately. The former fusion problem can be approached by standard registration techniques. The latter fusion problem could be solved by performing a logic OR operation [8] as illustrated in Figure 14, or by linear combination weighted by the median values [76]. The fusion of all channels with logic Boolean OR operator will propagate foreground and background intensity variations into the grid alignment algorithm and increase its robustness assuming that there is little spurious variation in the background. The option of fusing channels beforehand reduces multi-channel computation and avoids the problem of merging multiple grids detected per each channel.

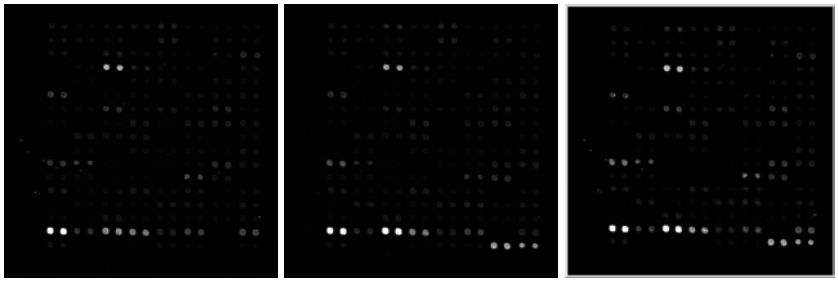

Figure 14: Microarray images of red (left) and green (middle) channels that are fused by Boolean OR function before processing (right).

Estimating grid rotation: The third component of data-driven methods addresses the problem of grid rotation. One approach to this problem is an exhaustive search of all expected rotational angles [8]. This approach is motivated by the fact that the range of

grid rotations is quite small, and therefore the search space is small. An initial angular estimate can be made by analyzing four edges of a 2D array [69]. The disadvantage of this approach is that small angle image rotations introduce pixel distortions because rotated pixels with new non-integer locations are rounded to the closest integer location (row and column). Another approach to the grid rotation problem is the use of discrete Radon transformation [14]. In this case, the grid rotation angle is estimated by (a) performing projections in multiple directions (Radon transformation) and (b) selecting the maximum median projection value. While Radon transformation is computationally expensive, a significant speed-up can be achieved by successive refinement of angular increments and limiting the range of angular rotations.

Finding multiple grids: The fourth component of data-driven methods tackles the problem of multiple grids or multiple distinct 2D sub-arrays of spots. These distinct grids are also arranged in a 2D array format, thus the number of expected distinct grids can be defined by the number of grids along horizontal (row) and vertical (column) axes. These numbers can be specified as input parameters since they are considered to be our prior knowledge about microarray slides. Given the input parameters, an algorithm has to partition an original image into sub-areas containing individual grids. Due to the nature of most frequently occurring microarray images, one approach is to divide the original images into rectangular sub-areas based on the input parameters and process each sub-area separately.

If the input parameters are not available then the problem can be approached by treating the entire image as one grid, searching for all irregular lines in the entire image, and then analyzing the spacing of all found mutually perpendicular grid lines [8]. Every

large discontinuity in the line spacing will indicate the end of one and beginning of another sub-grid (2D arrays of spots). An example result is shown in Figure 15.

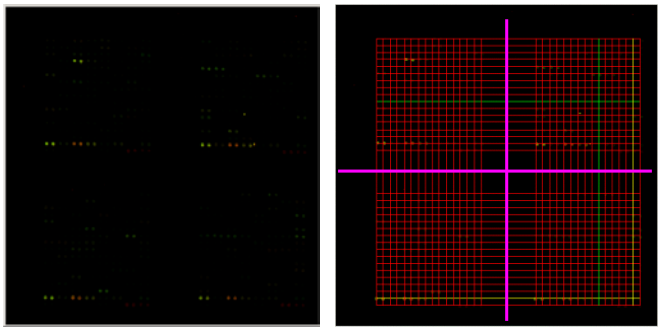

Figure 15: An example result of processing the original image (left) with the proposed algorithm and analyzing discontinuities in line spacing (right) to partition the original image into sub-images containing one sub-array per sub-image.

Speed and accuracy tradeoffs: Another optional component of data-driven methods could incorporate the speed and accuracy tradeoffs by image down-sampling option. It is well known that the speed of most image-processing algorithm is linearly proportional to the number of pixels since every pixel has to be accessed at least once and processed in some way. If two microarray images of the same pixel size and with the same content would contain NxM spots of radii R1 (image 1) and R2 (image 2), such that R1<R2, then the alignment of image 2 with spots of radius R2 could be performed faster by R1/R2 sub-sampling without any loss of accuracy with respect to the alignment performed on image 1. From this follows that the tradeoff between speed (or computational requirements) and grid alignment accuracy is also a function of spot size. In practice, down-sampling (or local averaging) is preferred instead of sub-sampling in order to preserve local spot information that could be completely eliminated by sub-sampling.

Repeatability and parameter optimization: In order to introduce fully automated methods and hence microarray image processing repeatability, it is necessary to address the issue of algorithmic parameter optimization. The first part of this task is to discriminate one-time setup parameters, e.g., number of grids or number of lines, from the data dependent parameters, e.g., size of spatial filters or noise thresholds. Next, it is beneficial to limit the ranges of the parameters that should be optimized by specifying their lower and upper bounds, e.g., grid angular rotation. This step reduces any unnecessary computation cost during optimization. Finally, an optimization strategy has to be devised so that a global optimum rather than a local parameter optimum is found for a given "optimality" metric.

While the benefit of parameter optimization is a fully automated grid alignment tool, the drawback of optimization is the need for more computation and hence slower execution speed. From a system performance view point, it is desirable to create optional user-driven inputs for algorithmic parameters in order to incorporate any prior knowledge about microarray image layout. Users that do not specify any microarray layout information will use more computational resources than users that partly describe input data. Nonetheless, the availability of optional algorithmic inputs and embedded parameter optimization techniques let end users decide between the two application extremes, such as real-time performance with limited computational resources and off-line processing with supercomputing resources.

Incorporating prior knowledge about grids: The most common prior knowledge about microarray layout includes number of grids (along rows and along columns), number of lines per grid, and perhaps spot radius. Other inputs about corner

spot locations, line spacing, grid rotation or background characteristics should be easily incorporated into grid alignment algorithms. It is also possible that an irregularly spaced grid as detected by a data-driven method should be overruled by a strict regularity requirement on the final grid. For example, due to our prior knowledge about printing, the requirement to generate a grid with equally spaced rows could be incorporated into the final grid by (a) computing a histogram of distances between adjacent already detected rows, and (b) selecting the most frequent distance as the most likely correct row spacing **[8]**. One can then choose the row with the highest algorithmic confidence (score) as the initial location and place the final grid according to the regularity constraint.

The data-driven approaches are capable of finding irregular grids but are prone to misalignment due to spurious or missing spots and are also dependent on many parameters. One can achieve significant cost savings with data-driven approaches when the majority of microarray slides meet certain quality standards and a fully automated algorithm flags images that are beyond its reliable processing capability.

7 FOREGROUND SEPARATION

The outcome of grid alignment is an approximation of spot locations. A spot location is usually defined as a rectangular image area enclosing one spot (also denoted as a grid cell). The next task is to identify pixels that belong to foreground (signal) of expected spot shape and to background. We refer to this task as foreground separation and it involves image segmentation and clustering.

The term image segmentation is associated with the problem of partitioning an image into spatially contiguous regions with similar properties (e.g., color or texture), while the term image clustering refers to the problem of partitioning an image into sets of pixels with similar properties (e.g., intensity or color or texture) but not necessarily connected. The objective of segmentation inside of a grid cell is to find one segment that contains the foreground information. If a spot could be formed by a set of non-contiguous regions/pixels, then image clustering can be applied. While microarray image segmentation and clustering problems result in grouping pixels based on intensity similarities, it is quite frequent to use a spatial template-based separation, where the template follows a spot shape model. We should also mention foreground separation methods that assign foreground and background labels to pixels based on both intensities and locations.

We describe next the foreground separation methods using (1) spatial templates, (2) intensity based clustering, (3) intensity based segmentation, and (4) spatial and intensity information. We also address the issue of foreground separation from multi-channel microarray images.

7.1 Foreground Separation Using Spatial Templates

This type of signal separation assumes that a spot is centered inside of a grid cell and it closely matches the expected spot morphology. The spatial template consists typically of two co-centric circles, where the pixels inside of the smaller circle are labeled as foreground (signal) and the pixels outside of the larger circle are labeled as background (see Figure 16). All pixels in between of the two co-centric circles are viewed as transition pixels and are not used. Clearly, this type of foreground separation will fail for

spots with varying radii or spatial offsets from the grid cell center, and will include all pixels with artifacts (e.g., dust particles, scratches, or spot contaminants). The consequence of poor signal separation will lead to artificially increased background level and distorted signal to background ratio. A quantitative comparison of the results obtained from circular spots and segmented spots can be found in [45].

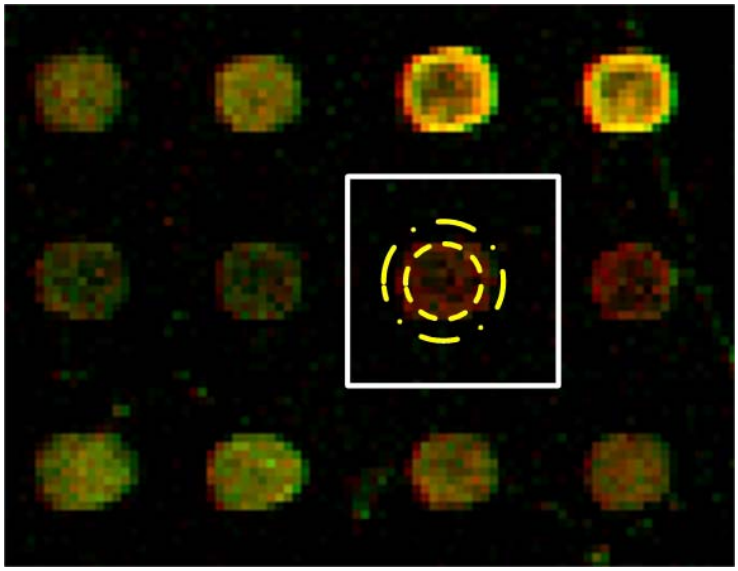

Figure 16: Illustration of a grid cell and the separation using spatial co-centric circular templates.

7.2 Foreground Separation Using Intensity Based Clustering

This type of signal separation boils down to a two class image clustering problem (or image thresholding) [69]. Image thresholding is executed by choosing a threshold intensity value and assigning the signal label to all pixels that are above the threshold value (or below depending on a microarray image dark-bright scheme). The threshold value can be chosen by computing the expected percentage of spot pixels inside of a grid cell based on the knowledge about image resolution and spot radius. The thresholding approach can be viewed as clustering by determining a cluster separation boundary.

Other clustering approaches use cluster intensity representatives, for instance, K-means or K-medoids [40], and the similarity between any intensity and the particular representative in order to assign pixel label (cluster membership). These methods can also be applied to the foreground separation problem [13].

Let us consider an example with thresholding. If a spot physical radius is about one micrometer and the microarray image resolution is 10 pixels per micrometer, then a spot area is equal to 314 pixels (π*radius^2). For a spot spacing (center to center) equal to twice the spot diameter (4 micrometers or 40 pixels), we can estimate the percentage of spot pixels as 314*100/(40x40)= 19.63% (spot area divided by grid cell area). Thus, an intensity threshold value would be equal to 19.63%*(max intensity – min intensity). This approach performs well when all pixels inside of a spot are different from the background. It fails for spots with varying radii, low contrast and high noise.

Figure 17 shows examples of accurate and inaccurate foreground separation. In this example, we used an advanced K-means clustering algorithm [7] that iteratively re-assigns foreground and background pixel labels untill the cluster's centroid intensities do not change significantly.

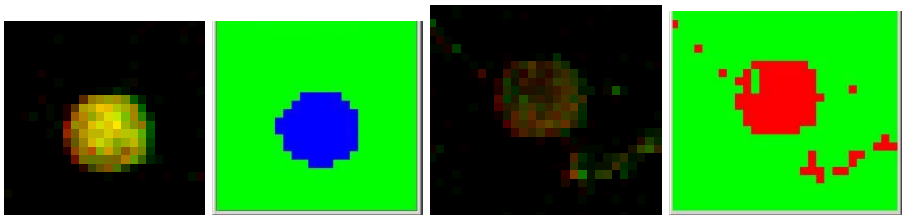

Figure 17: Examples of accurate (left – original image, and second from left - label image) and inaccurate (second from right – original image, and right – label image) foreground separation using intensity based clustering. The results were obtained using the Isodata (advanced K-means) algorithm [7].

7.3 Foreground Separation Using Intensity Based Segmentation

There are many segmentation methods available in the image processing literature [61, Chapter 6], and we will describe only those that have been frequently used with microarray images, such as seeded region growing and watershed segmentation.

Seeded region growing segmentation starts with a set of input pixel locations (seeds) [76], [25]. The segmentation method groups simultaneously pixels of similar intensities with the seeds to form a set of contiguous pixels (regions). The grouping is executed incrementally for a decreasing similarity threshold. The segmentation is completed when all pixels have been assigned to one of the regions grown from the initial seeds. In the case of microarray images, the foreground seed could be chosen either as the center location of a grid cell or as the maximum intensity pixel inside a grid cell. Similarly, the background seed could be selected either as the middle point between two spots or as the minimum intensity pixel inside a grid cell.

Morphological segmentation by watershed transformation is based on image operators derived from mathematical morphology [3]. There are two basic operators, dilation and erosion, and two composite operators, opening and closing. These operators are frequently used for filtering light or dark image structures according to a pre-defined size and shape. In the case of microarray images, morphological operators can filter out structures that deviate too much from the expected shape and size of a spot. Segmentation by watershed transformation can be viewed as the analysis of a grid cell intensity relief consisting of (a) no peak (missing spot), (b) one peak (clear spot) and (c) multiple peaks (vague spot). The case of multiple peaks is treated by searching for peak separation boundaries with the morphological operators that mimic watersheds (flooding image

areas below peaks). The outcome of the segmentation step is the region that corresponds to the most likely spots according to the morphological analysis of grid cell image intensities.

The main difference between foreground separation using clustering and using segmentation is illustrated in Figure 18. If a spot segment (region) is correctly identified then the segmentation approach will exclude dark pixels from the foreground assuming that they are surrounded by a connected set of pixels. In contrary, the clustering approach will include to the foreground cluster pixels that belong to the background or the intensity transitioning area. These pros and cons can be seen in the middle and right images in Figure 18.

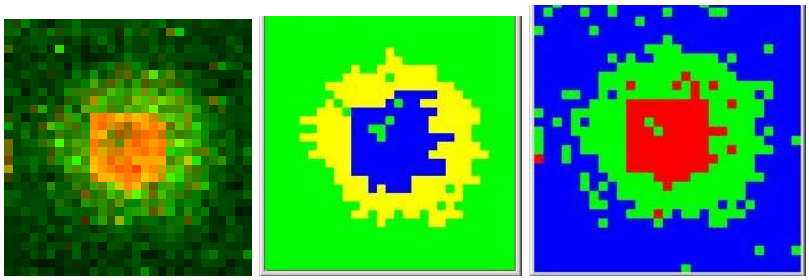

Figure 18: An example of pros and cons of foreground separation using intensity based clustering and segmentation. Left – original image, middle – segmentation result and right – clustering result. The results were obtained using the Isodata (advanced K-means) [72] and region growing algorithms [7].

Another issue to consider while choosing the most appropriate foreground separation technique is the priority order for selecting correct foreground pixels. There are certain grid cells where multiple interpretations are plausible as illustrated in Figure 19. If two segments of approximately the same size are detected inside of a grid cell (see Figure 19) then should we select (a) the brighter segment or (b) the segment with less

irregular shape or (c) the segment closer to the grid center? If a scratched spot consisting of two half disks is considered as a valid spot then should we include into foreground all segments of the same intensity that are close to or connected to the main segment positioned over the grid center? These decisions require ordering priorities in terms of expected region intensity, location and spot morphology.

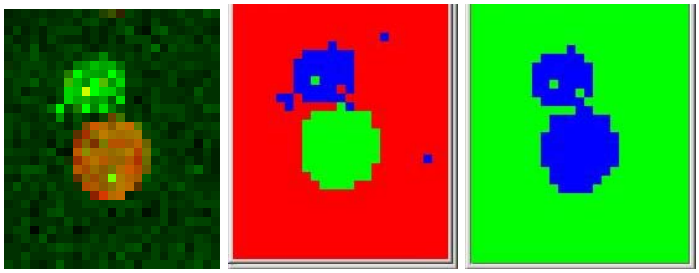

Figure 19: Multiple interpretations of the original grid cell image on the left side. The interpretation can vary based on prior region intensity and/or location and/or morphology information.

7.4 Foreground Separation Using Spatial And Intensity Information (Hybrid Methods)

Several foreground separation methods try to integrate the prior knowledge about spot morphology (spatial template), spot location and expected intensity distribution. These methods could be viewed as a sequence of steps consisting of segmentation or clustering image partitions, spatial template image partitions, statistical testing, and foreground/background trimming.

Spatially constrained segmentation and clustering: For instance, foreground separation using segmentation leads to a connected region that is fitted to a spatial template [53]. If the best-fitted circle deviates too much from the template then the spot is labeled as invalid. Another example would be foreground separation using clustering

with additional minimization constrain on cluster dispersion [13]. The particular choice of clustering could be the partitioning method based on K=2 medoids (PAM) with Manhattan distance as the similarity metric. This method in [13] was reported to be robust to the presence of noise in microarray images.

Mann-Whitney statistical testing: This foreground separation algorithm is executed by randomly selecting N pixels from the background and N pixels with the lowest intensities from the foreground over an expected spatial template of a spot [22]. Next, the two sets of pixels are compared according to the Mann-Whitney test [67, Test 12] with critical values of 0.05 or 0.01. The Mann-Whitney non-parametric test is a technique designed for evaluating a hypothesis whether or not two independent samples represent two populations with different median values. Iteratively, the darkest foreground pixels are replaced with those pixels that have not yet been chosen, and evaluated until the Mann-Whitney test satisfies the statistical significance criteria. The foreground separation is then achieved by selecting all pixels with higher intensities than the background pixels that passed the statistical significance test. It is apparent that this method relies on good selections of background pixels but incorporates our prior knowledge about spot template and expected intensity distributions. Unfortunately, this method cannot detect the presence of artifacts that bias the foreground separation results.

Spatial and intensity trimming: This approach is based on analyzing intensity distributions of foreground and background pixels as defined by a spatial template and then discarding those pixels that are classified as distribution outliers [29, Chapter 3]. Spatial trimming is achieved by initial foreground and background assignments over a spot template while intensity trimming is accomplished by removing pixels with intensity

outliers with respect to foreground and background intensity distributions. The goal of spatial and intensity trimming is to remove (a) contamination pixels (e.g., dust or dirt) in foreground and background regions, and (b) artifact pixels (e.g. doughnut spot shape) in foreground region. Figure 20 illustrates a couple of examples where contamination pixels would skew the resulting gene expressions if they would not be trimmed off.

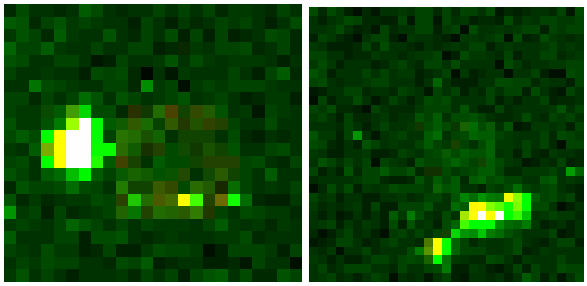

Figure 20: A couple of grid cell examples where contamination pixels have to be trimmed.

The trimming approach is similar to Mann-Whitney statistical testing but the statistical testing of the trimming method is applied to foreground and background pixels (intensity distribution analysis) instead of only to background pixels in the case of Mann-Whitney statistical testing. The spatial trimming can be improved by using two co-centric circles that define foreground, background and transient pixels. The transient pixels are eliminated from the analysis since they are not reliable. During intensity trimming, the choice of intensity threshold values that divide distribution outliers from other intensities depends on a user and the values are related to a statistical confidence. Empirically, a good performance is obtained when the threshold values eliminate approximately 5-10% of each, foreground and background, cumulative distributions [29, Chapter 3]. However, this approach should not be used when a spot size is very small (3-4 pixels in diameter) since the underlying statistical assumption of this analysis is the use of a sufficiently large

number of samples (pixels). For example, for a spot of the radius equal to two pixels, there would be only π*2^2=12.57 foreground pixels, and the number of foreground outliers would be 5%*π*2^2= 0.63 pixel.

7.5 Foreground Separation From Multi-Channel Microarray Images

During the foreground separation step, one has to address the issue of multi-channel processing. For example, the red and green input image channels from a cDNA slide can be treated separately or together. Let us consider the foreground separation using intensity thresholding. The foreground separation threshold values can be computed by considering (1) Euclidean distances to each pixel represented as a two-dimensional intensity vector (hypersphere separation), (2) intensities for red and green channel pixels separately (volume separation), (3) correlated intensities for red and green channel pixels (hyperplane separation), or (4) intensities of pixels after fusing red and green channels with some non-linear operators (e.g., after fusing with the Boolean OR operator). Depending on the choice of thresholding approach, the foreground separation boundary for a two-channel microarray image will lead to circular, rectangular, linear or non-linear curves as illustrated in Figure 7 and Figure 8

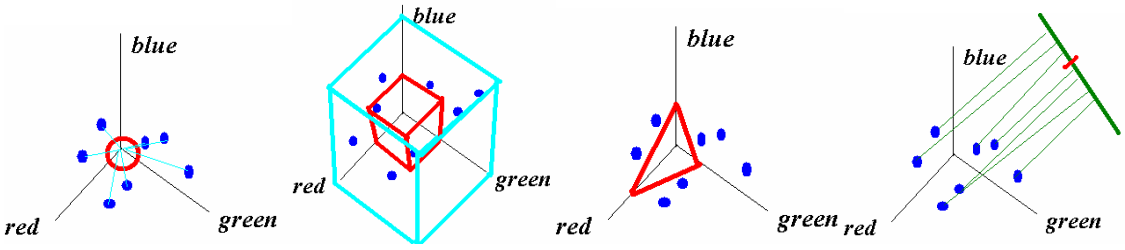

Figure 21: Visualization of four types of separation boundaries for foreground versus background using intensity based thresholding. From left to right: hypersphere, volume, hyperplane and point in a projected space as boundary types.

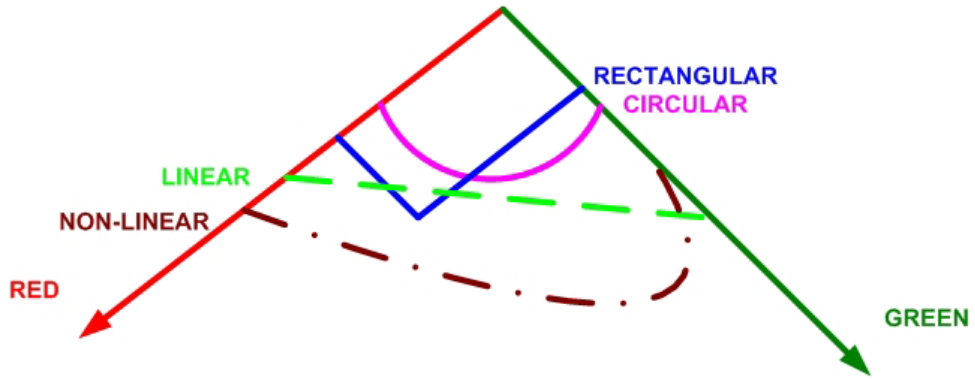

Figure 22: Possible foreground separation boundaries for two-channel input data.

Each of the aforementioned separation boundaries leads to a different set of spot and background labels. One should be aware of different statistical assumptions about a joint PDF of multiple channels associated with each separation boundary. A few examples of the results obtained using multiple boundary types are shown in Figure 23. As expected, the total count of foreground pixels varies based on the multi-channel separation method; sphere-15913, volume-509, plane-15877, nonlinear AND – 13735 and nonlinear OR – 16045 (400x400 image size, two bytes per pixel).

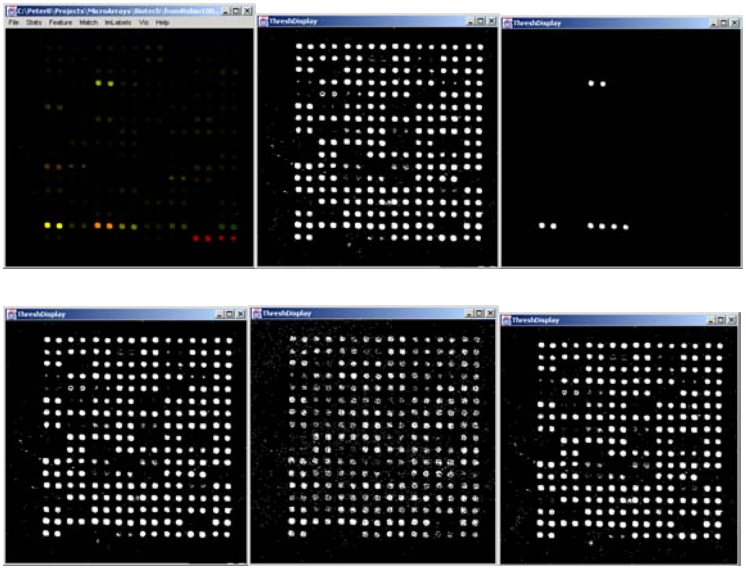

Figure 23: Examples of the results for spot versus background separation obtained from the two-channel input image shown in the top row (left) with multiple boundary types; hypersphere (top row, middle), volume (top row, right), hyperplane (bottom row, left), nonlinear after AND operation (bottom row, middle) and nonlinear after OR operation (bottom row, right).

8 SPOT QUALITY ASSESSMENT

The main goals of image-based spot quality assessment (or grid screening) are (1) to identify grid cells that contain valid spots, and (2) to eliminate invalid spots from further analysis. In order to detect invalid or defective spots, one has to define (a) spot validity criteria (metrics), for example, as deviations from the "ideal" microarray image (see Section 5.1), and (b) the deviation threshold values separating valid and invalid spot categories. In general, criteria for evaluating spot validity can be divided into two classes. The first class of spot validity criteria is for assessing foreground and background intensities. It includes assessing (a) absolute background and foreground levels, (b) background variation, (c) foreground saturation and (d) foreground-to-background intensity ratio (or signal-to-noise ratio). The second class is for evaluating morphological properties of foreground, such as spot shape and size irregularities, or spot location (position offset).

In addition to spot quality assessment, one would like to understand the relationships between the detected defects of invalid spots and the sources of those detected defects in the

microarray experiments. This type of analysis is usually referred to as spot quality control. While spot quality assessment is necessary for generating reliable data and for automation of high-throughput microarray systems, it is really our ultimate goal to analyze spot defects and prevent their occurrence in future. For example, according to [30, Table 1], one could relate sources of microarray experiment fluctuations with certain defects, e.g., a non-specific background factor would be related to the occurrence of spatially bleeding spots, or an amplification (PCR protocol) would be related to the occurrence of saturated spots.

In this section, we will focus only on image-based spot quality assessment (or valid spot detection) since spot quality control is still an active area of research. We will provide a brief description of a few commonly used quality assessment criteria that represent a subset of numerous quality assessment definitions and quality criterion variations found in the literature [75], [46], [76], [28], and [29]. The criteria presented here can be combined with other spot quality control techniques, for example, those that are based on spiked genes or housekeeping genes, and those that are based on spot replicas.

8.1 Criteria for Assessing Background and Foreground Intensities

Background intensity variations: There are two types of background variation criteria. First, local and global background variability metrics are designed for assessing local and global background noise. The metrics are indirectly proportional to the background variation, for instance, defined as a multiplicative of the background estimates of standard deviation [75]. While the local metrics can detect the presence of contaminants in a grid cell, the global metrics provide indications about variations across an entire microarray slide.

Second, the metrics that relate local and global background variations can detect excessively high local background within a slide. These metrics are designed based on the observation that some grid cells might have higher average background noise than the overall slide. For example, according to the designed formulas below [4], the quality metric q would approach one for valid spots and zero for invalid spots.

$$q_{BKG}^{LOC\&GLOB1} = \frac{\mu_{BKG}^{GLOBAL}}{\mu_{BKG}^{LOCAL} + \mu_{BKG}^{GLOBAL}}; \quad q_{BKG}^{LOC\&GLOB2} = \frac{m_{BKG}^{GLOBAL}}{m_{BKG}^{LOCAL} + m_{BKG}^{GLOBAL}}$$

(1)

where q is the quality metric, m is a median, μ is a mean. The notation of FRG refers to foreground and BKG to background.

Foreground and background intensity uniformity: It is assumed in this case that foreground and background should have uniform intensity distribution. In other words, a large variation of foreground intensities indicates less trustworthy spot. Similarly, a large variation of background intensities might signal noise in slide preparation. Thus, for detecting foreground defects, one could use the statistical metric provided in Equation (2) [29, Chapter 3]. The metric approaches one for valid spots (zero variance), and compensates for the fact that spots with higher intensity magnitudes might have larger variations (division by the foreground sample mean).

$$q_{FRG}^{STAT} = 1 - \frac{\sigma_{FRG}}{\mu_{FRG}}$$

(2)

where q is the statistical quality metric, μ is the mean and σ is the standard deviation of foreground pixels (FRG).

Another pair of metrics for foreground and background relates absolute intensity values according to the formula below [4].

$$q_{FRG}^{ABS} = 1 - (I_{FRG,\max} - I_{FRG,\min}) \Big/ Range; \; q_{BKG}^{ABS} = 1 - (I_{BKG,\max} - I_{BKG,\min}) \Big/ Range$$

(3)

where q is the quality metric using absolute intensity values, I is the maximum or minimum intensity of foreground (FRG) or background (BKG), and Range is an intensity range.

Due to many fluctuations during microarray slide preparation, one could also lessen the requirement on probability distribution uniformity. One might hypothesize that regardless of the expected probability distribution function (PDF) of foreground pixels, e.g., uniform, Gaussian, Weibull, Beta, Exponential, or Gamma, the PDF model should be consistent for all spots on a microarray slide. This requirement would be referred to as distribution model consistency [7]. It is possible to introduce this type of a quality metric by estimating PDF model types for all spots and scrutinizing spots that follow a PDF model different from the PDF model of the majority of spots. The type of a PDF model can be estimated based on a parametric probability distribution plane [26, pp. 29] by using higher order central moments (skew and kurtosis) of spot intensities. However, these types of quality screening might require better understanding of microarray image intensities at macroscopic and microscopic levels.

Foreground intensity saturation: It has been understood that intensity saturation occurs when pixel intensities exceed the detection range of a scanning devise (e.g., a photomultiplier tube or an electron detector) and the recorded intensity is truncated. As a

result of saturation, estimations of gene expressions are biased [28]. Although, it is not clear how to discriminate saturated pixels of highly expressed genes from saturated pixels due to contaminants, one could apply the saturation metrics first to both types of saturated pixels, then apply spot shape metrics and iteratively refine the results.

In order to detect saturation, continuous or categorical metrics have been proposed. A continuous metric computes the ratio of a number of saturated and foreground pixels as defined below [4].

$$q_{SATURATION}^{CONT} = 1 - count_{saturated} \Big/ count_{all}$$

(4)

A categorical metric assigns a value denoting valid or invalid spot based on a thresholded count of saturated pixels. The formula is provided in Equation (5) below. For example, according to [75], if a spot contains less than T=10% of saturated pixels then it is a valid spot under the assumption that a sample mean or median values are extracted. The median value is less affected by saturation since, theoretically, as long as the count of saturated pixels is less than 50%, the median value will not change.

$$q_{SATURATION}^{CATEG} = \begin{matrix} 1; & if & count_{saturated} < T\% \\ 0; & if & count_{saturated} \geq T\% \end{matrix}$$

(5)

Signal-to-noise ratio: The most commonly explored spot property is a signal-to-noise ratio (SNR). The SNR criterion eliminates spots with very weak signal (1<SNR< thresh), no signal (SNR~1), or ghost spots (SNR<1). It is based on intensity information and defined either with sample mean and median values according to the formula below.

$$q_{SNR}^{MEAN} = \mu_{FRG}/(\mu_{FRG} + \mu_{BKG}); \quad q_{SNR}^{MEDIAN} = m_{FRG}/(m_{FRG} + m_{BKG})$$

(6)

8.2 Criteria for Assessing Morphological Properties of Foreground

Spot shape: There are multiple metrics for assessing spot shape and we provide a few examples. The underlying assumptions in spot shape metrics are that a valid spot should have (a) all pixels inside of a circular region labeled as foreground (consistency of spot area), (b) the perimeter of pixels labeled as foreground equal to the expected circumference of a spot (consistency of spot perimeter), and (c) the cross sections through the centroid of all pixels labeled as foreground equal to the expected diameter of a spot (consistency of spot diameter).

First, the area-based spot shape quality metrics can be computed according to the following formulas (see [29, Chapter 3], [75]):

$$q_{SHAPE}^{AREA1} = \frac{|A - A_0|}{A_0}; \quad q_{SHAPE}^{AREA2} = \exp(-\frac{|A - A_0|}{A_0})$$

(7)

where A is the area of the pixels labeled as foreground, and A_0 is the expected spot area. This metric can be modified to reflect the percentage of ignored pixels [46, Chapter 6] according to the formula below.

$$q_{SHAPE}^{AREA3} = \frac{|A - A_0|}{A} * 100\%$$

(8)

Second, the perimeter-based spot shape quality metrics would be computed according to the previous formulas, where A, and A_0 would be replaced with the perimeter of the area labeled as foreground and the circumference of a spot. However, for small spots the perimeter estimate is very inaccurate due to the nature of digital images. Thus, this metric is modified to a ratio of the estimated A and the expected circumference C of a spot according to the formula below (see [4]).

$$q_{SHAPE}^{PERIM} = 4\pi A \big/ C^2$$

(9)

Another perimeter-based spot quality metric can be defined if a foreground region is constrained by a grid cell boundary [46, Chapter 3]. In this case, the metric is defined as a ratio of open perimeter and total foreground region perimeter, where the open perimeter is the coinciding length of the foreground region with the grid cell boundary. This metric might detect spills or any spot-to-spot bleeding.

Third, the diameter-based spot shape quality metrics assess spot deviation from the expected circular shape either by estimating a diameter from an area [7] or by measuring the cross section lengths through the spot centroid in multiple angular directions. If the estimated diameter or the cross section length deviates from the expected value by more than a user specified percentage then the spot is invalid. The quality metrics are defined below.

$$q_{SHAPE}^{X-SECTION1} = \frac{|L - L_0|}{L_0}; \quad q_{SHAPE}^{X-SECTION2} = \exp(-\frac{|L - L_0|}{L_0})$$

(10)

where L is the cross section length of the pixels labeled as foreground, and L_0 is the expected length.

In the above metrics, we have included only the spatial spot information. It is possible to incorporate spatial and intensity information into a quality metric by asserting a model of an expected spot intensity profile. For instance, one could assume that the spatial distribution of spot intensities would follow a Uniform model or a Gaussian model [14], [69]. Thus, a quality metric would be computed by (a) fitting the model to the spot foreground intensities [14] or estimating Gaussian model parameters [69], and (b) evaluating the deviation from the model. Unfortunately, the underlying assumption about spatial distributions of spot intensities has not been proven to be justifiable since the pixel level intensities are not yet understood well.

Spot location (spot displacement or position offset): The spot location metric is defined as the Euclidean distance between a centroid of all pixels labeled as foreground and the expected spot center. The tacit assumption in this case is that the grid alignment algorithm is very accurate and hence one can consider the center of each grid cell to be the expected spot center (or the ground truth value for quality assessment). In general, the metric reflects our beliefs that a detected spot closer to the expected position is more trustworthy than a spot far away.

8.3 Applying Spot Quality Criteria

After defining multiple quality assessment metrics, one would like to combine a set of metrics and flag invalid spots. In order to combine multiple metrics, each metric has to be normalized (or weighted) depending on the range of its values. For instance, all metrics could be normalized to span the range of values between 0 and 1. Next, a composite

quality score can be formed by applying operators to a selected set of metrics. The most frequent operator is multiplication for continuous metrics, and Boolean AND operator for categorical metrics, as shown in Equation (11). The logic behind choosing these operators is the fact that one would like to impose all quality criteria simultaneously during spot quality assessment. However, a special treatment is usually given to incorporating saturation metrics [28], [75].

$$q_{COMPOSITE}^{CONT} = \prod_{i=1}^{m} q_i; \quad q_{COMPOSITE}^{CATEG} = \bigcap_{i=1}^{m} q_i$$

(11)

Another spot quality application issue arises when spot quality assessments are performed on multiple image channels. In general, each channel can be evaluated separately, and the final decision about validity of each spot can be reached by a voting mechanism (i.e., if the majority channel specific evaluations leads to an invalid label then the spot is flagged as invalid). It is also possible to create composite spot quality scores by combining quality metrics for all channels and all criteria.

The challenges in applying spot quality metrics is in choosing (a) the most appropriate screening criteria, (b) meaningful threshold values, (c) operators for combining several screening criteria and (d) a mechanism for evaluating multiple image channels. There is still a need to define standard sets of image-based spot quality criteria and introduce them into commercial software packages. Some of the commercial software packages, for example, GenePix and QuantArray, have already incorporated the most common quality assurance metrics as they are summarized in Table 1.

Table 1: Spot screening criteria used in GenePix and QuantArray software packages. μ is a sample mean, m is a median, σ is a standard deviation, A is an area of a convex hull, k is a multiplier, I is an image intensity, Range is an intensity range and count is a pixel count.

Inspection Criterion	Description
SNR for each channel	$\mu_{FRG}/(\mu_{FRG}+\mu_{BKG});\ m_{FRG}/(m_{FRG}+m_{BKG})$
Foreground and background variability	$k_{FRG}*\mu_{FRG}/\sigma_{FRG};\ k_{BKG}*\mu_{BKG}/\sigma_{BKG}$
Excessively high background	$\mu_{BKG}^{GLOBAL}/(\mu_{BKG}^{GLOBAL}+\mu_{BKG});\ m_{BKG}^{GLOBAL}/(m_{BKG}^{GLOBAL}+m_{BKG})$
Saturation	$1-count_{saturated}/count_{all}$
Proportion of foreground above μ+k*σ of background	$count_{>k*\sigma}/count_{all}$; $if\ \mu_{FRG}>\mu_{BKG}+k\sigma_{BKG}\ then\ count_{>k*\sigma}+1$
Spot Shape	$4\pi A/Perim^2$
Foreground and background uniformity	$1-(I_{FRG,\max}-I_{FRG,\min})/Range;\ 1-(I_{BKG,\max}-I_{BKG,\min})/Range$

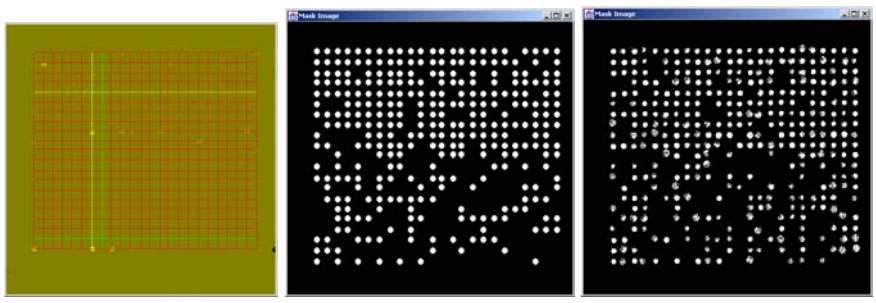

Figure 24: Examples of spot quality screening. The input two-channel microarray image with an overlaid unevenly spaced grid (left). The results of screening with the SNR criterion (middle), and with the spot location and diameter criteria after Mann-Whitney foreground separation using hyperplane (Euclidean distance) thresholding (right). The grid alignment, foreground separation and screening results were obtained using I2K software package [7].

9 DATA QUANTIFICATION AND NORMALIZATION

Given a set of valid spots and two sets of image pixels labeled as foreground and background in each spot, there is a need to extract descriptors of each valid spot for further gene regulation evaluation. Data quantification (or spot feature extraction) refers to extracting descriptive values of foreground and background pixels for each spot. Ideally, extracted descriptors (also called features or attributes) should be directly proportional to the mRNA quantity in the solution that was deposited in a spot, and should represent the deposited gene expression level. However, fluorescent intensity measurements in each channel might be scaled or distorted differently according to some linear or non-linear functions during data preparation steps. Thus, normalization of extracted spot descriptors is desirable.

9.1 Quantification or Extraction of Spot Descriptors

In general, we could divide spot descriptors into two categories, such as (1) absolute and relative descriptors, and (2) statistical and deterministic descriptors. However, before presenting particular candidates for spot descriptors, it is important to understand the microarray experimental design in terms of gene expression outcomes. As

mentioned in Section 3.1, raw microarray intensities cannot be interpreted as absolute measurements due to random and systematic variability in microarray image data preparation. Thus, in cDNA gene expression experiments, one is interested in the statistical difference in gene expression levels between the probe and target (also referred to as test and reference, and denoting the mRNA mixture hybridized to the array and the library on the array). Based on these considerations, we will focus on relative statistical descriptors.

Spot Descriptors: Relative descriptors of cDNA spots are computed as ratios, logarithmic ratios or regression ratios of values derived from red and green channels [28]. The values can be raw intensities or some absolute descriptors of raw intensities. Statistical descriptors characterize sets of pixel intensities that are viewed as realizations of a random process following a certain probability distribution. The most common statistical descriptors of the two sets of foreground and background image pixels are their sample means, medians and modes. These descriptors are defined in every statistical textbook [67]. Other statistical descriptors have been proposed, for example, the volume of foreground intensity as defined in Equation (12) (see [46, Chapter 6]).

$$FRG\ Volume = (\mu_{FRG} - \mu_{BKG}) * A_{FRG} \quad \textbf{(12)}$$

where μ is the sample mean, and A is the foreground area. Examples of the forms of microarray spot descriptors using ratio or logarithmic ratio are provided below.

$$des_{RATIO}^{X} = \frac{X_{FRG}^{\text{CHANNEL 0}}}{X_{FRG}^{\text{CHANNEL 1}}} \quad \textbf{(13)}$$

$$des_{LOG\ RATIO}^{X\ WRT\ BKG} = \log_2 \left(\frac{X_{FRG}^{\text{CHANNEL 0}} - X_{BKG}^{\text{CHANNEL 0}}}{X_{FRG}^{\text{CHANNEL 1}} - X_{BKG}^{\text{CHANNEL 1}}} \right)$$

(14)

where X is the symbol for sample mean or median or mode, the subscripts FRG and BKG refer to foreground and background, and the superscript CHANNEL refers to red or green microarray laser scans. While Equation (13) represents a direct ratio of absolute values, Equation (14) is a logarithmic ratio of relative differences (X WRT BKG stands for X with respect to background). The motivation for using relative differences is to reduce the effect of non-specific fluorescence (e.g., auto-fluorescence of glass slides). In Equation (14), one has to take special care of the cases when foreground intensities are smaller than background intensities in one of the channels (so called ghost spots with reverse spot contrast polarity). Additional statistical parameters, such as standard deviation, skew or kurtosis, can be extracted to measure an intensity distribution shape (spread, skew and symmetry) with higher order central moments. Statistically speaking, these statistical parameters indicate the confidence intervals of extracted descriptors. For instance, high standard deviation means large variation of computed sample means across multiple spots, and hence our confidence in obtaining the exact descriptor over and over is low (high uncertainty of absolute values for repeated experiments).

The regression ratios are quite often used as part of red and green channel normalization [59]. In this case, the goal is to extract descriptors that adjust for (a) the different efficiency of red and green fluorescent labels when being scanned (red dyes have higher efficiency than green dyes), and (b) the different quantities of initial mRNA from the two samples. The underlying assumption for computing regression ratios is that pixel intensities in red and green channels are linearly dependent. A regression ratio is the

estimate of this linear relationship and is based on correlation analysis. The regression ratio is computed by fitting a zero-intersecting straight line to a scatter plot formed from red and green intensities of foreground and background pixels. If the regression ratio k is used for adjusting the fitted line Y=k*X to Y'=X' then this type of analysis is also denoted as linear calibration of red and green channels. The calibration mechanism is illustrated in Figure 25.

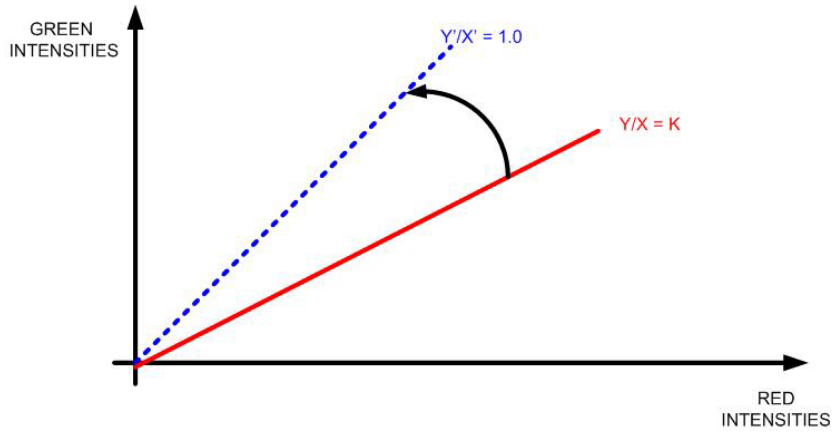

Figure 25: Illustration of red and green channel calibration.

Visualization of Spot Descriptors: This problem has been neglected in the past and might not have been an issue when the number of spots was low. The most typical way of visualizing spot descriptors is to open a spread sheet, and visually inspect spots and their descriptors represented in a tabular form. Nevertheless, as the number of spots is increasing, the tabular form with thousands of rows does not provide very efficient visual inspection mechanism.

Given the fact that microarray spot layouts are on a regular 2D grid, it seems very natural to present extracted spot descriptors in the same grid-based form as the original grid-based layout of spots. This visualization approach preserves the relative spatial locations of spots. Furthermore, every descriptor value from a set of possible descriptors

can be visually inspected by sweeping through a stack of "spot descriptor" images (or feature images). An example of this type of visualization is shown in Figure 26.

Figure 26: An example of extraction of spot descriptors and their visualization using I2K [7]. Original grid (left), derived feature images of sample mean (second left), standard deviation (second right) and skew (right).

The long term goal of visualization is to combine gene expression descriptors with additional information about genes. For example, the ultimate goal could be to show microarray gene expression information with 3D structure of nucleic acid sequences at multiple scales, such as macroscopic (populations), microscopic (chromosomal locations), sub-microscopic (pharmacological pathways), sub-microscopic/atomic (DNA sequences), and atomic (protein structure) [1].

Selection of Spot Descriptors: Naturally, one would like to select the most appropriate spot descriptors for further microarray data analysis. It is apparent that absolute deterministic descriptors, like sum of intensities, would be spot size dependent, and contamination and saturation sensitive. Similarly, absolute descriptors would be inappropriate for cDNA microarray spots since the fluorescent intensity measurements depend on the reference label. Given the aforementioned relative statistical spot descriptors, the pros and cons of each descriptor can be described as follows.

The use of sample mean descriptors reduces spot intensity variation but it is sensitive to intensity outliers. In contrary, the use of median descriptors is more resistant to outliers but it is also more computationally expensive. The median of a foreground or background set of size N is determined by rank ordering all intensities according to their values and selecting the middle value if N is odd or the average of middle two values if N is even [26]. Clearly, the median computation is more computationally complex than averaging all pixel intensities. The mode descriptor (or the "most-likely" intensity) is measured as the most often occurring intensity in a set of foreground or background intensities. It is resistant to outliers and inexpensive to compute, but it might be difficult to estimate reliably when the frequency of intensity occurrences (intensity histogram) contains multiple peaks (multi-modal intensity distribution). When the intensity distribution is uni-modal and symmetric then mean, median and mode estimates are equal.

In terms of the selection of operators for computing relative descriptors, one could view the problem from statistical modeling and correlation analysis view points. From a statistical modeling view point, it is preferable to use operators (transformations) that lead to a random variable following a Gaussian distribution because of the ease of mathematical manipulations using the Gaussian distribution model. Previous studies of microarray spot intensity distributions have shown that pixel intensity values separated by one or two other pixels can be considered as independent draws from a Log-normal or Gamma distributions [77]. Thus, using a Gaussian distribution model for logarithmically transformed raw intensities would be appropriate. Nonetheless, the distribution of raw

intensities should be verified for particular microarray data since data sets and microarray technologies vary a lot.

From a correlation analysis view point, we assume that pixel intensities in red and green channels are linearly dependent and a regression ratio is the estimate of this linear relationship. Unfortunately, the relationship between channels might not be linear. For example, it has been shown that the deviation from the expected linear dependency increases with the increasing intensity values and it has a large spread for low intensity values [60]. Thus, regression ratio is an appropriate descriptor for microarray images with high intensity contrast between foreground and background.

Improving Robustness of Spot Descriptors: One should also mention other techniques that improve robustness of statistical descriptors. For example, it is quite frequent to introduce trimming of intensity outliers based on histograms followed by the computation of statistical descriptors. The trimming percentages for high and low intensity values are set by a user and vary according to data. Another improvement can be achieved by combining trimming of intensity outliers and spatial outliers defined by a spot spatial template. The spatial trimming eliminates contamination pixels and "bleeding" parts of a spot but introduces all problems associated with template-based foreground separation.

9.2 Normalization

Data quantization and normalization steps are closely related and frequently interchanged. The motivation for normalizing microarray images and/or extracted descriptors comes from the fact that one would like to compare results obtained from multiple slides, scanners, or laboratories, and with multiple microarray techniques. The

difficulty of performing meaningful comparisons arises from different slide preparations (e.g., amounts of mRNA), scanner settings, microarray protocols or labeling specifics. The purpose of normalization is to adjust for these variations, primarily for label efficiency and hybridization efficiency, so that we can discover true biological variations as defined by the microarray experimental studies. In general, the approaches to normalization can be divided to (1) methods using statistical descriptors, (2) techniques using control spots, and (3) correlation (regression) analyses.

Normalization using statistical descriptors: Statistical descriptors include sample mean, median, mode or percentile of intensity distribution. This particular normalization can be performed by either division or subtraction of statistical descriptors. For example, one could apply Z-transformation to this problem that consists of subtracting sample mean from all intensities and dividing their values by the standard deviation (see Equation (15)). The Z-transformation would normalize intensities but would not compensate for labeling non-linearity.

$$I_{Z-TRANSFORM}^{NORM\ STAT}(row,col) = \frac{I(row,col) - \mu}{\sigma} \quad \textbf{(15)}$$

where μ is the mean and σ is the standard deviation of an entire image.

Another example for cDNA microarray data normalization would be background correction. Equation (14) shows a specific way of normalizing each spot based its local background statistics. The same type of normalization could be performed for each sub-grid or any group of spots.

Normalization using control spots: This technique requires inserting spots of known intensities or genes of known expression level into a microarray slide. By detecting control spots, one can normalize all other spots with respect to the reference intensities defined by the control spots. In this case, it is recommended to scatter the control spots across an entire slide so that local variations can be normalized accurately.

Normalization using regression analyses: As it was mentioned before, regression ratios are quite often used as part of red and green channel normalization **[59**]. The normalization methods for two-channel arrays can be characterized as (1) within-slide normalization (location or scale), (2) paired-slide normalization (dye-swap), and (3) multiple slide normalization [80]. The within-slide normalization can be divided into (a) location global normalization (log(red/green) – normalization factor), (b) location intensity dependent normalization (log(red/green) – normalization factor as a function of spot intensity), (c) location within-print-tip-group normalization (log(red/green) – grid dependent normalization factor as a function of spot intensity) and (d) scale normalization (modeling spread of various print-tip groups). The most commonly used normalization technique is the location global normalization [80], assuming zero-offset linear dependency between red and green channels (red=k*green). The normalization factor c of the model (log(red/green) – c) is computed as c=log(k) so that the normalized log-ratios have zero mean or median.

Many researchers have performed studies about normalization strategies and their significance, and showed the importance of normalization. For instance, based on the normalization experiments with multiple Arabidopsis thaliana clones reported in [70], the average pin-wise strategy was recommended. The pin-wise strategy was defined as a

slide-wise normalization of the diluted and constant signals, followed by averaging of the dilution and control signals over several slides, and then computing regression ratios. Another strategy is to pre-filter microarray images or descriptors before normalization in order to eliminate "meaningless" values, for instance, negative intensity values [46, Chapter 7]. The image filters should spatially smooth intensities over a small neighborhood of pixels, e.g., by using convolution, rank or adaptive filters [61]. It is also common practice to pre-filter descriptors by using the median of background plus three times Median Absolute Deviation (MAD) of the control genes as a threshold value.

One should also be aware that the regression analysis can be applied under different red and green channel dependency models. While the majority of analyses assume linear dependency, it is also possible to assume non-linear models, such as piece-wise linear, polynomial or curve dependency models [29, Chapter 12]. Approaches to model non-linear red and green channel dependencies are based on (1) introducing higher order models (e.g., locally weighted polynomial regression (LOWESS or LOES) [29, Chapter 12], exponential model [63]), (2) dividing an intensity range into segments where linearity can be assumed (piece-wise linear model), or (3) combining the two previous strategies.

When it comes to using non-linear normalization models, one has to be aware of the trade-offs related to modeling generality, continuity and accuracy. For example, if the exponential model has been observed in experimental data [63], then it is preferred since it contains fewer parameters than a multi-parameter polynomial regression model. However, the polynomial regression model might be more accurate on average for a large collection of microarray data sets. Similarly, piece-wise models will achieve better

accuracy than models using only one-model for the entire range of values. However, a piece-wise linear model will introduce rate discontinuity artifacts for intensity differences (e.g., two equal pairs of intensity differences might become radically different when normalized by different piece-wise linear models).

10 Processing Affymetrix Microarray Data

So far we have focused primarily on cDNA microarray technology since the Affymetrix technology based on oligonucleotide arrays is proprietary [2]. Many of the concepts and approaches described in the previous sections are applicable to Affymetrix images. Nonetheless, the Affymetrix technology is different in the following three aspects. First, cDNA arrays are appropriate for detecting long DNA sequences while oligonucleotide arrays are designed for detecting only a short DNA sequence. In order to detect long sequences with Affymetrix technology, one has to detect multiple short sequences first and then combine the values to compare the results with cDNA results. Second, oligonucleotide arrays contain only foreground and therefore the extracted descriptors represent absolute gene expression level. Third, the Affymetrix technology has been much more expensive than the technology with coated glass slides.

From an image processing view point, the Affymetrix chips are easier to process since there is no background and the spot shape is rectangular. Figure 27 shows an example image of an Affymetrix chip that is processed by proprietary software to extract all statistical intensity information. Nevertheless, the images might also contain detrimental defects as shown in Figure 28.

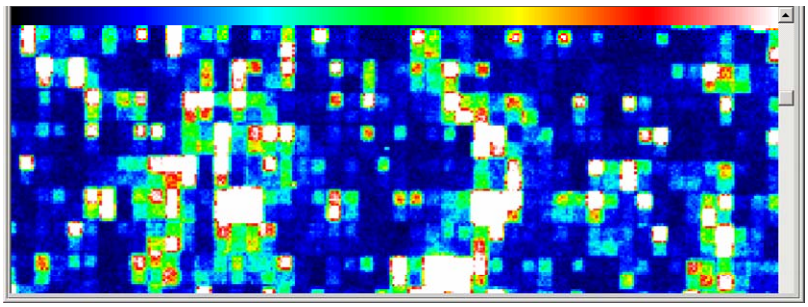

Figure 27: An example of Affymetrix chip with rectangular spots.

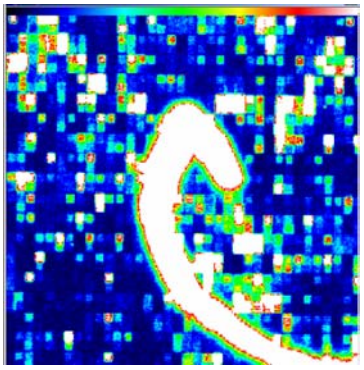

Figure 28: An example of Affymetrix chip contamination.

Due to the lack of information about Affymetrix image processing, we will focus only on the problem of comparing cDNA and Affymmetrix descriptors. For Affymetrix data, several software methods are available to generate ratios and perform data normalization. For example, one method is based on the average difference (AD) between Perfect Match (PM) and Mis-Match (MM) probe pairs since the Affymetrix arrays represent a gene using 20 match and mismatch probes with 25 nucleotides per probe. Another method is based on the model-based expression indexes developed by Li and Wong [51] and their normalization using invariant sets. In this case, the goal is equalize distributions of probe intensities for all arrays in a set of arrays using level quantile normalization [17].

The increasing number of platforms for microarray analysis raises the question of which platform is most repeatable and which most accurately represents the true biological phenomena being tested. The amount of variation or agreement in results across platforms is a major issue today. Many studies comparing identical samples across platforms have presented contradictory results and repeatability or precision has not necessarily proven to be the major factor in gaining accurate biological predictions [78]. Both normalized cDNA and Affymetrix expression ratios may have skewed variances that are dependant on signal intensity. The common log transformations used for data normalization may often increase this differential variation. Variance stabilizing transformations have been presented for both Affymetrix [32] and for two color arrays [33]. Controlling the variance across platforms may allow for more accurate cross platform comparisons. Clearly a better understanding of the methods used for image acquisition and analysis is one of the critical factors in reducing variation across platforms and will contribute to informed decisions on technology preferences.

11 SUMMARY

Microarray image processing is a basic component of learning about gene expression. We have overviewed several processing steps and researchers will have to address a few additional challenging issues in extracting reliable information about microarray experiments. One of the future challenges of image processing will be the optimization of data extraction and the fine play between over saturation of an image and signals below detection level. A series of questions arises in this context. How can we

increase the dynamic range? Can we use partially saturated spots? In other words, can we extract data only from pixels that have not reached saturation, in order to get useful data when median signal may be too high? Shall we reject low quality spot data or attempt to extract whatever useful data can be saved? Can individual spots that are saturated be flagged and rescanned at lower PMT values in an automated fashion until relevant ratios are obtained? Can we construct composite images from different scanning intensities to maximize the number of spots that (a) fall into detectable ranges with good ratios and (b) are not biased by pixels that are too high or too low in intensity? Can a low powered pre-scan be done first, as is done today, and then instead of a global scan at set levels, adjust the PMT on a local basis to adjust for spots that are saturated?

Other challenges are related to microarray image storage and archival, standardization, automation and fully automated high-throughput processing requirements. There is also a lack of understanding of microarray images at pixel level and uncertainty propagation. The integration of gene expression information with other biological measurements and prior knowledge is also an open area of research. The above questions and challenges have to be answered by additional research and development.

ACKNOWLEDGEMENTS

The authors would like to thank the W. M. Keck Center for Comparative and Functional Genomics at the University of Illinois in Urbana-Champaign for providing several microarray images for this work. Dr. Peter Bajcsy acknowledges the support from the National Center for Supercomputing Applications (NCSA), University of Illinois at

Urbana-Champaign (UIUC), and the National Institute of Health (NIH) under Grant No. R01 EY10457.

REFERENCES

[1] Adams R. M, B. Stancampiano, M. McKenna and D. Small, "Case Study: A Virtual Environment for Genomic Data Visualization," IEEE Transactions on Visualization, 2002, Oct. 27- Nov. 1, 2002, Boston, MA, USA (published as CD).

[2] Affymetrix Inc., "Gene Chip Arrays," Product Description at http://www.affymetrix.com/index.affx

[3] Angulo J. and J. Serra, "Automatic Analysis of DNA Microarray Images Using Mathematical Morphology," Bioinformatics, Vol 19. NO. 5, 2003, pp. 553-562.

[4] Axon Instruments Inc., "GenePix Pro," Product Description at http://www.axon.com/GN_Genomics.html

[5] Arbeitman MN, Furlong EE, Imam F, Johnson E, Null BH, Baker BS, Krasnow MA, Scott MP, Davis RW, White KP. 2002. Gene expression during the life cycle of Drosophila melanogaster. Science. 297:2270-5.

[6] Babak T, Zhang W, Morris Q, Blencowe BJ, Hughes TR. 2004. Probing microRNAs with microarrays: tissue specificity and functional inference. RNA.10:1813-9.

[7] Bajcsy P., "Image To Knowledge (I2K)," Software Documentation at http://alg.ncsa.uiuc.edu/tools/docs/i2k/manual/index.html.

[8] Bajcsy P., "Gridline: Automatic Grid Alignment in DNA Microarray Scans," IEEE Transactions on Image Processing, VOL 13, NO 1, pp.15-25, January 2004.

[9] Bajcsy P., J. Han, L. Liu and J. Young, "Survey of Bio-Data Analysis from Data Mining Perspective," Chapter 2 of Jason T. L. Wang, Mohammed J. Zaki, Hannu T. T. Toivonen, and Dennis Shasha (eds.), *Data Mining in Bioinformatics*, Springer Verlag, 2004, pp.9-39.

[10] Bajcsy P and R. Kooper, "Prediction Accuracy of Color Imagery from Hyperspectral Imagery," SPIE 2005, Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XI, 28 March - 1 April 2005, Orlando (*Kissimmee*), Florida USA.

[11] Balagurunathan Y., E. R. Dougherty, Y. Chen, M. L. Bittner, and J. M. Trent, "Simulation of cDNA Microarrays via a Parameterized Random Signal Model," Journal of Biomedical Optics, 7(3), 2002.

[12] Baldi P. and S. Brunak, "Bioinformatics, The Machine Learning Approach," Second Edition, The MIT Press, Cambridge, Massachusetts, 2001.

[13] Bozinov D. and J. Rahnenfuhrer, " Unsupervised Technique for Robust Target Separation and Analysis of DNA Microarray Spots Through Adaptive Pixel Clustering," Bioinformatics, Vol. 18, NO. 5, 2002, pp. 747-756.

[14] Brandle N., H. Bischof and H. Lapp, "Robust DNA Microarray Image Analysis," machine Vision and Applications (June 2003) 15: 11-28.

[15] Brazma A., P. Hungamp, J. Quackenbush, G. Sherlock, P. Spellman, C. Stoecker, J. Aach, W. Ansorge, C. A. Ball, H.C. Causton, T. Gaasterland, P. Glenisson, F. C. P. Holstege, I. F. Kim, V. Markowitz, J. C. Matese, H. Parkinson, A. Robinson, U. Sarkans, S. Schulze-Kremer, J. Stewart, R. Taylor, J. Vilo & M. Vingron "Minimum

Information About a Microarray Experiment (MIAME) – toward standards for microarray data, Nat. Genet. 29, 365-371, December 2001.

[16] Brazma A., A. Robinson, and M. Ashburner. One-stop shop for microarray data. Nature, 403:699-700, 2000.

[17] Bolstad, B.M., Irizarry, R.A., Astrand, M., and Speed, T.P. 2003 A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics 19:185-193. Editorial. Free and public expression. Nature 420:851, 2001.

[18] Brenner S., M. Johnson and J. Bridgham *et al.*, Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays, *Nat. Biotechnol.* 18 (2000) (6), pp. 630–634

[19] Brown C. S., P. C. Goodwin and P. K. Sorger (2001), "Image metrics in the statistical analysis of DNA microarray data ", Proceedings of the National Academy of Sciences, 98(16):8944-8949.

[20] Buhler J., T. Ideker, D. Haynor, "Dapple: Improved Techniques for Finding Spots on DNA Microarrays," UV CSE Technical Report UWTR 2000-08-05.

[21] Butcher LM, Meaburn E, Liu L, Fernandes C, Hill L, Al-Chalabi A, Plomin R, Schalkwyk L, Craig IW. 2004. Genotyping pooled DNA on microarrays: a systematic genome screen of thousands of SNPs in large samples to detect QTLs for complex traits. Behav Genet. 34:549-55

[22] Chen Y., E. R. Dougherty, and M. L. Bittner, "Ratio-Based Decisions and the Quantitative Analysis of cDNA Microarray Images," Journal Of Biomedical Optics 2(4), 364–374, 1997.

[23] CLONDIAG Chip Technologies," FluorIS: Array Standardization Tool," Product Description at http://www.clondiag.com/products/dispo/fluoris/index.php

[24] Chee M., R. Yang and E. Hubbell *et al.*, Accessing genetic information with high-density DNA arrays, *Science* 274 (1996) (5287), pp. 610–614.

[25] CSIRO Mathematical and Informational Sciences, "Spot Image Analysis Software," Product Documentation at http://experimental.act.cmis.csiro.au/Spot/index.php

[26] Cullen A. C. and H. C. Frey, "Probabilistic Techniques in Exposure Assessment," Plenum Press, NY, 1999.

[27] Davies E.R., "Machine Vision: Theory, Algorithms, Practicalities," Academic Press, 1997.

[28] Dodd L. E., E. L. Korn, L. M. McShane, G.V.R. Chandramouli and E. Y. Chuang, "Correcting Log Ratios for Signal Saturation in cDNA Microarrays," Bioinformatics, VOL. 20, NO. 16, 2004, pp. 2685-2693.

[29] Draghici S., Data Analysis Tools for DNA Microarrays. Chapman & Hall CRC Mathematical Biology and Medicine Series, 2003.

[30] Draghici S., A. Kuklin, B. Hoff and S. Shams. "Experimental Design, Analysis of variance and Slide Quality Assessment in Gene Expression Arrays," Current Opinion in Drug Discovery and Development, Vol. 4. NO. 3, 2001, pp. 332-337.

[31] Dudley AM, Aach J, Steffen MA, Church GM 2002. Measuring absolute expression with microarrays with a calibrated reference sample and an extended signal intensity range. Proc Natl Acad Sci U S A. 2002 May 28;99(11):7554-9.

[32] Durbin, B.P., Hardin, J.S., Hawkins, D.M., and Rocke, D.M. (2002) "Avariance-stabilizing transformation for gene-expression microarray data,"Bioinformatics, 18, S105—S110.

[33] Durbin B.P., and Rocke D.M. 2004 Variance-stabilizing transformations for two-color microarrays. Bioinformatics 20:660-667.

[34] Eisen M., "ScanAlyze," Product Description at. http://rana.lbl.gov/EisenSoftware.htm

[35] Espina V, Woodhouse EC, Wulfkuhle J, Asmussen HD, Petricoin EF 3rd, Liotta LA. 2004. Protein microarray detection strategies: focus on direct detection technologies. J Immunol Methods.290:121-33.

[36] Friend S. H. and R. B. Stoughton, "The Magic of Microarray," Scientific American, February 2002, pp. 44-49.

[37] Foster I. and C. Kesselman. "Computational Grids," *Chapter 2 of "The Grid: Blueprint for a New Computing Infrastructure"*, Morgan-Kaufman, 1999.

[38] Golub T.R., D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M Loh, J.R. Downing, M.A. Caligiuri, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," Science 286: (5439)1999, pp. 531-537.

[39] Goryachev A. B., P. F. MacGregor and A. M. Edwards, "Unfolding Microarray Data," Journal of Computational Biology, Volume 8, Number 4, 2001, pp. 443-461.

[40] Han J. and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2001.

[41] Hanlon SE, Lieb JD. 2004. Progress and challenges in profiling the dynamics of chromatin and transcription factor binding with DNA microarrays. Curr Opin Genet Dev. 14:697-705.

[42] Hartelius K. and J. M. Cartstensen, "Bayesian Grid matching," IEEE Trans. on PAMI, Vo. 25, NO. 2. February 2003, pp.162-173.

[43] Ideker T, Galitski T, Hood L. 2001 A new approach to decoding life: systems biology. Annu Rev Genomics Hum Genet. 2:343-72.

[44] Imaging Research Inc, "Array Vision," Product Description at http://www.imagingresearch.com/products/Genomics_Software.asp.

[45] Jain A. N., T. A. Tokuyasu, A. M. Snijders, R. Segraves, D. G. Albertson and D. Pinkel, "Fully Automated Quantification of Microarray Image Data," Genome Research, Vol. 12, Issue 2, February 2002, pp. 325-332.

[46] Kamberova G. and S. Shah (editors), DNA Array Image Analysis - Nuts and Bolts. Data Analysis Tools for DNA Microarrays. DNA Press LLC, MA, 2002.

[47] Karo M., C. Dwan, J. Freeman, J. Weissman, M. Livny, E. Retzel, "Applying Grid technologies to bioinformatics," *Proceedings of the 10th IEEE International Symposium on High Performance Distributed Computing*, 2001 pp. 441 –442.

[48] Katzer M., F. Kummert and G. Sagerer, "Robust Automatic Microarray Image Analysis," In Proceedings of the International Conference on Bioinformatics: North-South Networking, Bangkok, 2002.

[49] Kawasaki S, Borchert C, Deyholos M, Wang H, Brazille S, Kawai K, Galbraith D, Bohnert HJ. 2001. Gene expression profiles during the initial phase of salt stress in rice. Plant Cell. 13:889-905.

[50] Lawrence N. D., M. Milo, M. Niranjan, P. Rashbass, and S. Soullier, "Reducing the variability in cDNA microarray image processing by Bayesian inference," Bioinformatics, Vol 20., NO. 4, 2004, 518-526.

[51] Li C and Wong WH: Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *Proc. Natl. Acad. Sci.* 2001, 98: 31-36.

[52] Liang P. and A.B. Pardee, Differential display of eukaryotic messenger RNA by means of the polymerase chain reaction, *Science* 257 (1992) (5072), pp. 967–971.

[53] Liew A W-C., H. Yan, and M. Yang, "Robust Adapttive Spot Segmentation of DNA Microarray Images," Pattern Recognition 36 (2003), 1251-1254.

[54] Mangalam H. et al. GeneX: An Open Source gene expression database and integrated tool set, IBM Systems Journal, vol40. no2. 552-569, 2001.

[55] Moore S. K., "Understanding The Human Genome," IEEE Spectrum, November 2000, pp. 33-42.

[56] Oostlander AE, Meijer GA, Ylstra B. 2004. Microarray-based comparative genomic hybridization and its applications in human genetics. Clin Genet. 66:488-95.

[57] Packard BioChip Technologies, LLC, “Quant Array Analysis Software,” Product Description at http://www.packardbioscience.com/products/521.asp.

[58] Preuss TM, Caceres M, Oldham MC, Geschwind DH. 2004. Human brain evolution: insights from microarrays. Nat Rev Genet. 5:850-60.

[59] Quackenbush J: Computational analysis of microarray data. *Nat. Rev. Genet.* 2001, 2(6):418-427.

[60] Rocke D. and B. Durbin, "A model for measurement error for gene expression arrays", Journal of Computational Biology, 8(6):557-569.

[61] Russ J. The Image Processing Handbook. Third Edition. CRC Press with IEEE Press. Published by CRC Press LLC. 1999.

[62] Saal L. H., C. Troein, J. Vallon-Christersson, S. Gruvberger, Å. Borg and C. Peterson. BioArray Software Environment: A Platform for Comprehensive Management and Analysis of Microarray Data. Genome Biology, 3(8): software0003.1-0003.6, 2002

[63] Samartzidou H., L. Turner, and T. Houts, “Lucidea Microarray ScoreCard: An integrated tool for validation of microarray gene expression experiments,” Innovation Forum, Microarrays, Life Science News 8, 2001 Amersham Pharmacia Biotech.

[64] Schena M, Shalon D, Davis RW, and Brown PO: Quantitative monitoring of gene expression patterns with complementary DNA microarray. *Science* 1995, 270: 467-470.

[65] Scanalytics Inc., “MicroArray Suite,” Product Description at http://www.scanalytics.com/product/hts/microarray.html

[66] Seo J. and B. Shneiderman, "Interactively Exploring Hierarchical Clustering Results," Computer, July 2002, pp.80-86.

[67] Sheskin D. J. Handbook of Parametric and Nonparametric Statistical Procedures. Second Edition, Chapman and Hall CRC, 2000.

[68] Southern, E. M. 1975. Detection of specific sequences among DNA fragments separated by gel electrophoresis. Journal of Molecular Biology 98: 503-517.

[69] Steinfath M., W. Wruck, H. Seidel, H. Lehrach, U. Radelof, and J. O'Brien, "Automated image analysis for array hybridization experiments," Bioinformatics 2001 17: 634-641.

[70] Schuchhardt J., D. Beule, A. Malik, E. Wolski, H. Eickhoff, H. Lehrach and H. Herzel, "Normalization strategies for cDNA microarrays," Nucleic Acids Research, Oxford University Press, 2000, Vol. 28, No. 10 E47-e47.

[71] TIFF File format specification, Revision 6.0, June 3, 1992, Aldus Corporation.

[72] Tou J. T., and R. C. Gonzales, "Pattern Recognition Principles," Addison-Wesley Publishing Company, 1974.

[73] Tseng G. C., Min-Kyu Oh, L. Rohlin, J. C. Liao, and W. H. Wong Issues in cDNA microarray analysis: quality filtering, channel normalization, models of variations and assessment of gene effects Nucleic Acids Res. 2001 29: 2549-2557

[74] Velculescu V.E., L. Zhang, B. Vogelstein and K.W. Kinzler, Serial analysis of gene expression, *Science* 270 (1995) (5235), pp. 484–487.

[75] Wang X., S. Ghosh, and Sun-Wei Guo (2001), "Quantitative quality control in microarray image processing and data acquisition", Nucleic Acids Research, 29(15):e75.

[76] Whitfield CW, Cziko AM, Robinson GE. 2003. Gene expression profiles in the brain predict behavior in individual honey bees. Science. 302:296-9.

[77] Wit E. and J. McClure, "Statistical Adjustment of Signal Censoring in Gene Expression Experiments," Bioinformatics Vol. 19 no. 9 2003, Pages 1055-1060.

[78] Woo Y., Affourtit J., Daigle S., Viale A., Johnson K., Naggert J., and Churchill G. 2004. A comparison ofcDNA, oligonucleotide, and Affymetrix GeneChip gene expression platforms. J. of Biomolecular Techniques. 15:276-284.

[79] Yang Y.H., M. J. Buckley, S. Dudoit and T.P.Speed, "Comparison of Methods for Image Analysis on cDNA Microarray Data," Technical Report #584, Department of Statistics, University of California at Berkeley, November 2000.

[80] Yang Y.H., Dudoit S., Luu P., and Speed T.P.: Normalization for cDNA microarray Data. *Microarrays: Optical Technologies and Informatics.* SPIE BIOS 2001, San Jose, CA.

[81] Yue H, Eastman PS, Wang BB, Minor J, Doctolero MH, Nuttall RL, Stack R, Becker JW, Montgomery JR, Vainer M, Johnston R. 2001. An evaluation of the performance of cDNA microarrays for detecting changes in global mRNA expression. Nucleic Acids Research, 29, No. 8 e41-1