Lossless Compression of DNA Microarray Images

Yong Zhang, Rahul Parthe, and Don Adjeroh
Lane Department of Computer Science and Electrical Engineering,
West Virginia University, Morgantown WV 26506
{yzhang, rahulp, don}@csee.wvu.edu

Abstract

Microarray experiments are characterized by a massive amount of data, usually in the form of an image. Based on the nature of microarray images, we consider the microarray in terms of its structure and statistics. Based on the microarray image model, we propose a context-based method for lossless compression of microarray images using prediction by partial approximate matching (PPAM). In synchronization experiments, the raw data consists of two channel microarray images. The correlation between these two channel microarray images is explored in order to improve the compression performance. Our results show that, the proposed approach produces a better compression result when compared with results from the best-known microarray compression algorithm.

1. Introduction

It is now common to use microarrays for genome-wide monitoring of gene function and gene expression under different conditions. Evidently, the result of such large-scale experiments will be a massive amount of data. For instance, Spellman et al [27] reports of over 400,000 points of measurements in their experiments with *S. Cerevisiae*. Thus, there is an immediate challenge to find efficient and effective methods for managing the unprecedented volume of microarray image data [2].

The problem of storage for the microarray data is compounded by the need to store both the semi-processed data (example the expression ratios) used for analysis, and the original expression levels (from the Cy5 (Red) channel for reference conditions and Cy3 (Green) channel for experimental conditions). These original expression levels are required to be stored in the event of a possible need to re-analyze or revalidate the microarray data in future. Similar to the data explosion problem in DNA sequences, the growth of microarray data over time has been observed to be exponential[1].

The rest of the paper is organized as follows. In Section 2, we describe the special nature of microarray images, and propose a component-based model for microarray images. Section 3 describes our proposed schemes for compressing microarray images. Section 4 presents experimental results on compression using real microarray images. Section 5 concludes the paper.

2. Nature of microarray images

2.1. Formation of microarray images

To understand the nature of microarray images, it may be useful to have some basic idea of the processes involved in the formation of such images. The detailed descriptions can be found in [1, 8, 15]. The basic steps are as follows: (i) Isolate single stranded mRNAs from the cell or tissue and use these to generate sets of cDNAs. (ii) Using florescent labeling, attach tags to each mRNAs, to differentiate the mRNA molecules from the control (reference) cell (Green dye) and those from the experimental cell (Red dye). (iii) Mix the labeled samples and incubate in a hybridization solution with the cDNA samples already immobilized on the microarray spot. (iv) Using a reader or scanner, detect the mRNA abundance in each spot. (v) Based on the florescent tags, create digital images indicating the respective expression levels for each of the differently tagged molecules. The result is a set of two intensity images, one for the expression level of the reference (control) tissue/cell (the Cy3 or the Green channel), and the other for the sample (experimental) tissue/cell (the Cy5 or the Red channel).

Depending on the spacing between the spots and the overall size of the microarray, this procedure allows for a potentially high density of spots on the array (hence

[1] For example, see http://www.ncbi.nlm.nih.gov/Class/NAWBIS/Modules/Expression/exp45.html .

larger images), making it possible to measure expression profiles for tens of thousands of genes simultaneously. To capture the large range of possible expression levels, the intensities are usually represented as a 16-bit integer. With pixel spacing of about 2 microns per pixel, at 16 bits per pixel, image sizes of up to 50MB are common [8]. For genome-wide expression analysis, with say 20,000 genes under 5000 experimental conditions, we are looking at about 191MB per image per channel.

2.2. Microarray image model

Our image model is simple. We consider the microarray image in terms of its structure and statistics. For the structural model, we use a simple two-component model. The first component is the foreground, obtained using a spot segmentation procedure [8, 12]. We call the remaining areas the background. After identifying all the spots and their respective dimensions, we form the foreground image by rearranging the spots into an image, but without the previously adjourning background. Similar to the foreground, we rearrange the remaining parts of the microarray into the background image. With the foreground image, we may or may not be able to visually distinguish between the spots. However, in general, pixels in the foreground image will have relatively high intensity values.

To capture the different statistical nature of the two components, we describe the overall microarray image using a mixture model. Given $\Sigma = \{x_1, x_2, ..., x_{|\Sigma|}\}$, the set of symbols representing the distinct pixel values that appeared in the microarray image, we use a mixture model to describe the probability of occurrence of symbol x in the microarray image:

$$P(x) = w_0 P_0(x) + \sum_{i=1}^{m} w_i P_i(x) \,, \sum_{i=1}^{m} w_i = 1 \qquad (1)$$

where $P_0(x)$ is the model of the background, $P_i(x)$ $i = 1,2,...,m$ are the m mixing components, and w_i's are the weights. The mixing components could depend on the spots, and/or some other considerations. It was also observed that the distribution of the background image could be further modeled by some mixture model. For the mixing components, we assume that they are independent but identically distributed, with different parameters. Based on empirical observations, we model the foreground and the background using the gamma distribution:

$$P_i(x) = \frac{1}{\Gamma(\alpha_i) \cdot \beta_i} \left(\frac{x}{\beta_i} \right)^{\alpha_i - 1} \exp\left(-\frac{x}{\beta_i} \right) \qquad (2)$$

where β_i and α_i are scale and shape parameters respectively, with $\beta_i > 0$, $\alpha_i > 0$ and $\Gamma(x)$ is the gamma function of x, given by:

$$\Gamma(x) = \int_0^\infty u^{x-1} \exp(-u) du \qquad (3)$$

Thus, the different spots could exhibit a different distribution, based on their scale and shape parameters. This flexibility in the model is important, especially in capturing the longer tails exhibited by some spot distributions. By simply changing the parameters, we can obtain different distributions, such as the exponential distribution. We notice that the foreground and background components have evidently different distributions. In empirical fitting of the probability, we use a further scale factor on the $P_i(x)$ above to obtain $P_i'(x) = \theta_i . P_i(x)$ as the final probability, with $\theta_i > 0$. The parameters of the distribution can be estimated using standard methods [9]. However, the initial estimates will usually not give the best results. We suggest an iterative approach, whereby the initial estimates can be refined by iteratively fitting the distribution on the data, until one with the best fit is obtained. The criteria for best fit could be based on quantitative measures, such as the mean square error, or the relative entropy.

3. Microarray image compression

3.1. Context-based lossless image compression

Different methods have been proposed for compressing microarray images. For instance, SLOCO [7], provided a simple extension of LOCO [13], the basic algorithm used in JPEG-LS by including summary information about the microarray and the spots. In [5], a spiral scanning method was proposed based on the circular nature of the spots. Clearly, this approach will only work well for first-order contexts, where pixels are predicted by their immediate neighbor in the scanning path. In [6], wavelet-based lossy and lossless compression schemes were proposed for microarrays, using one level of decomposition. The lossless compression resulted in an expansion (16.22bpp) rather than compression of the microarray data. MicroZip [10] used arithmetic coding and the Burrows-Wheeler Transform (BWT) for lossless compression of microarray data, after dividing the pixel values into their least significant bits (LSB) and most significant bits (MSB). Their method produced comparatively superior results in lossless compression.

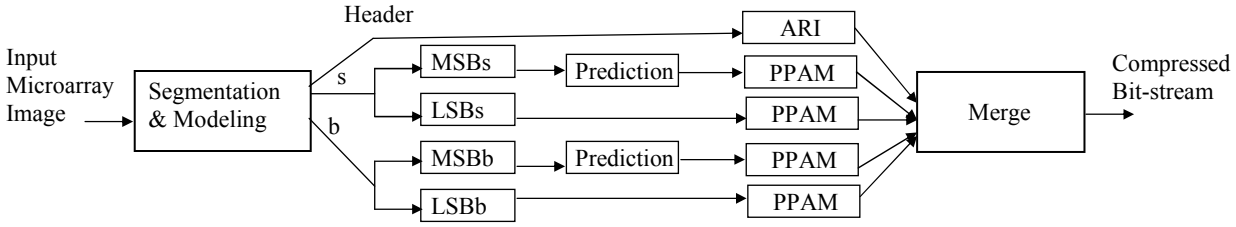

Figure 1: Microarray image compression scheme. ARI stands for arithmetic coding.

In this work, we focus on lossless compression of microarray images. We start by considering approaches that have been successful in compressing natural images. Most successful lossless image compression algorithms are context-based and they exploit the two-dimensional spatial redundancy in natural images [11, 14]. The basic motivation for context-based approaches is the promise of improved compression. Let an image be represented by a sequence $S = \{s_i, i = 1,...,|S|\}$, with symbols taken from a fixed alphabet, $\Sigma = \{\sigma_i, i = 1,...,|\Sigma|\}$. Σ is typically the set of distinct pixel gray levels in the image, or the set of distinct prediction errors, after applying some prediction scheme. Let the corresponding symbol probabilities be $p(\sigma_i), i = 1,...|\Sigma|$. Then the minimum number of bits required to encode the image without context modeling is the entropy of the source $H(S)$:

$$H(S) = -\sum_{\sigma_i \in \Sigma} p(\sigma_i) \log_2 p(\sigma_i) \qquad (4)$$

If contexts are considered, the conditional probability distribution for the set of symbols S_j' with context C_j will be $p(s_i \mid C_j), i = 1,...|S_j'|$, and the minimum bits used to encode the source should be:

$$H(S \mid C) = -\sum_{j=1}^{M} \left(p(C_j) \sum_{s_i \in S_j'} p(s_i \mid C_j) \log_2 (p(s_i \mid C_j)) \right) = -\sum_j \sum_{s_i} p(s_i, C_j) \log_2 (p(s_i \mid C_j)) \qquad (5)$$

where M is the total number of contexts. Since conditioning reduces entropy, the number of bits needed to encode the image is also reduced by using context modeling: $H(S \mid C) \le H(S)$.

Although the lossless image coding algorithms are based on some form of contexts, they always tend to use *fixed-order* contexts, which may limit their ability in estimating the conditional probability distributions. Our approach to microarray image compression is based on the recently proposed PPAM – *prediction by partial approximate matching* [16]. PPAM is an image compression technique, which extends the PPM text compression algorithm [3, 4] by considering the special characteristics of natural images. PPAM introduces two important concepts in image compression: the use of *variable-order* contexts, and the use of *approximate* contexts [16].

3.2. Microarray image compression by PPAM

Given the data sizes involved, compression is mandatory in efficient storage of microarrays. The requirement for efficient storage is worsened by the need to store the ratio image, sometimes along with the Cy5 and Cy3 channels. Compression of microarray images is, however, a tough challenge. Various characteristics of microarrays make them difficult to compress, most notably the noise and spots (random edges) in the microarray image. In terms of computation, the high resolution typically required for microarrays is also an issue for the potential number of contexts, and the attendant computations required (Some of our test images are 5496×1956, at 16-bit resolution). Because of the high bit depth, microarrays are often split into two eight-bit images, one part for the 8 most significant bits (MSB), and the other for the 8 least significant bits (LSB).

Fig. 1 shows a general block diagram for the proposed approach. Given the component-based microarray image model, the first step is to partition microarrays into its different components. We record important information about the microarray image in general, and each spot in particular as summary information. The summary information includes the image dimensions, the center location for each recognized spot, the mean and median intensity value within each spot, the spot dimensions (in terms of the radius of the bounding circle), etc. The summary information is compressed without loss and stored independently. This means that the summary information can be extracted and used independently for further analysis, without needing a full decompression. The foreground part of the image is also compressed without loss. The background part can be compressed with or without loss, since a certain loss of the background information may not affect the analysis of the microarray sequences.

For each component, we apply the PPAM algorithm independently. With each component, we further partition the pixel representation into its MSB and LSB (similar to previous methods). Then to compress the data, we first pass the MSB through an error prediction

scheme, and then feed the prediction residuals to the PPAM context model and encoder. For the LSB, given that prediction often increases the entropy, attesting to its random nature, we do not perform prediction. Rather, we feed the LSB part directly to PPAM context model and encoder.

3.3. Correlation between two-channel images

As previously mentioned, in synchronization experiments, microarray data are obtained from synchronized cells and suitable controls. Fluorescently labeled cDNA is synthesized using Cy3 ("green") for all controls and Cy5 ("red") for all experimental samples. Thus the raw data files consist of two channel images. If we can identify some correlation between the two channel images, we can then reduce the uncertainty in the images (thus improving the overall compression) by exploiting the correlation. Fig 2 shows the image histograms of the MSB part of the two channel images from the same microarray. In Fig 3, the scatter plot shows the correlation between the histograms of the two channels. From Fig. 2 and Fig. 3, it is obvious that there is a strong correlation between the green and the red channel images. In this project, we explore the potential correlation-based compression improvement by simply considering the compression of the difference between the two channel images, instead of coding "green" and "red" channels separately. Define the difference image *chd* as: *chd=ch1-ch2*, where *ch1* and *ch2* represent the corresponding channel images respectively. The *chd* image is more suitable than *ch1* or *ch2* for compression.

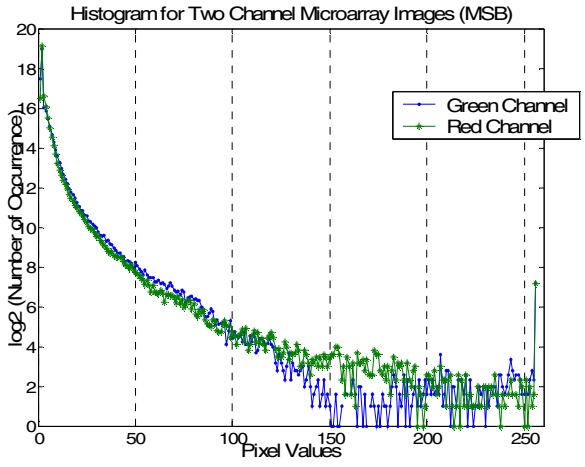

Figure 2: Channel histograms (MSB) for a sample microarray image

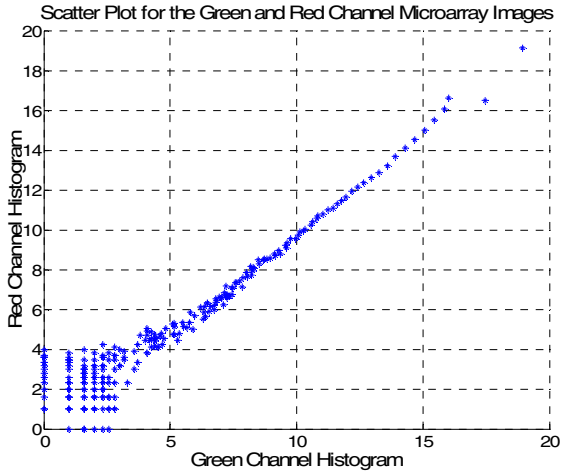

Figure 3: Scatter plot for MSB part for sample microarray image

4. Results

4.1. Setup and data sets

The experiments were performed in a MATLAB 6.5 environment, running on a PC with Pentium IV, 2.8GHz with 512MB. We used the three test images (labeled **Array1**, **Array2**, and **Array3**) that were used in evaluating MicroZip [10][2]. We also downloaded images from the Stanford Yeast Cell-Cycle Regulation Project data set[3]. We used three images from the Pheromone data set. The images are of size 1024×1024 each, and are labeled **y744n40, y744n100, and y744n101**, respectively.

4.2. Compression results

Table 1. PPAM performance

Image	Results	Overall
Array1LSB	8.010	11.380
Array1MSB	3.370	
Array2LSB	7.870	9.260
Array2MSB	1.391	
Array3LSB	7.460	8.120
Array3MSB	0.663	

Table 1 presents the compression performance using the proposed method on the three microarray images used in MicroZip. We note in particular that the proposed methods can compress the LSB, paving the way for lossless compression at rates less than 8bpp, for an original 16-bit microarray image. Table 2 shows the comparative coding performance of PPAM with state-of-the art compression algorithms. The cost of the header

[2] http://www.cs.ucr.edu/~yuluo/MicroZip/
[3] http://cellcycle-www.stanford.edu/

(summary) information is not included in the tables. On average, the extra cost required to store the summary information in a lossless manner was about 0.04bpp. Table 3 is shown that the correlation-based compression method has reduced the overall compression result by 0.569 bits per pixel for the test microarray images.

Table 2. **Comparative results for different methods on test microarray images**

Image	PPAM	PPMD	BWT	MicroZip	JPEG-LS
Array1	11.380	11.550	11.739	11.490	11.834
Array2	9.260	9.440	9.776	9.570	9.788
Array3	8.120	8.140	8.542	8.470	8.256

Table 3. **Compression results on two channel microarray images**

Image	Results	Overal	Image	Results	Overall
Ch1LSB	4.582	6.900	**Chd**	5.488	6.331
Ch1MSB	2.318		**Chd(Sign)**	0.843	
Ch2LSB	4.581	6.301	**Ch2LSB**	4.581	6.301
Ch2MSB	1.720		**Ch2MSB**	1.720	

5. Conclusion

Given the amount of data typically generated by microarray based experiments, there is need to find methods to store the generated data efficiently. In this work, starting with the nature of microarray images, we have proposed a simple model that captures both the structure and the general statistics in microarray images. Based on the model, we further proposed a method for compression of microarray images. The compression scheme is unusual in its use of approximate (rather than exact) contexts in modeling the symbol probabilities. The results showed that the proposed methods produced superior results in terms of compression performance. In particular, in compression of test 16-bit microarray images, the proposed method produced smaller overall bits per pixel when compared with MicroZip, the best-known microarray compression algorithm.

6. References

[1] P. Baldi and G. W. Hatfield, "DNA microarrays and gene expression," Cambridge University Press, 2002.

[2] D. E. Bassett, M. B. Eisen and M. S. Boguski, "Gene expression informatics- its all in your mine," *Nature Genetics Supplement*, vol. 21, pp. 51-55, January 1999.

[3] J.G. Cleary and W.J. Teahan, "Unbounded length contexts for PPM", *The Computer Journal*, 40(2/3), 67-75, 1997.

[4] J.G. Cleary and I.H. Witten, "Data compression using adaptive coding and partial string matching", *IEEE Transactions on Communication*, 32(4), 396-402, 1984.

[5] N. Faramarzpour, S. Shirani, J. Bondy, "Lossless DNA Microarray Image Compression," *IEEE Proceedings of the Data Compression Conference* (DCC '04') , 2004 IEEE.

[6] J. Hua, Z. Liu, Z. Xiong, Q, Wu and K. Castleman, "Microarray BASICA: Background adjustment, segmentation, image compression and analysis of microarray images", *Proceedings of the International Conference on Image Processing*, vol. 1, 2003.

[7] R. Jornsten, W. Wang, B. Yu, and K. Ramchandran. "Microarray image compression: SLOCO and the effects of information loss," *Signal Processing Journal* (*Special issue on genomic signal processing*), 2002.

[8] M. Katzer, F. Kummert, G. Sagerer, "Methods for automatic microarray image segmentation," *IEEE Transactions on Nanobioscience*, vol. 2, No. 4, pp. 202-214, December 2003.

[9] D. Kundu and M. Z. Raqab, "Generalized Rayleigh distribution: Different methods of estimation," http://home.iitk.ac.in/~kundu/pap.html.

[10] S. Lonardi, Y, Luo, "Gridding and Compression of Microarray Images," *Proceedings, 2004 IEEE Computational Systems Bioinformatics Conference* (CSB 2004), 2004.

[11] G. MacBeath and S.L. Schreiber, "Printing proteins as microarrays for high-throughput function determination", *Science*, 289(5485): p. 1760-1763, September 8, 2000.

[12] X. H. Wang, R. S. H. Istepanian, Y. H. Song, "Microarray Image Enhancement by Denoising Using Stationary Wavelet Transform," *IEEE Transactions on Nanobioscience*, vol. 2, No. 4, pp. 184-189, December 2003.

[13] M. J. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standandization into JPEG-LS", *IEEE Transactions on Image Processing*, 9(8), 2000.

[14] X Wu and N. Memon, "Context-based, adaptive, lossless image coding", *IEEE Transactions on Communications* 45(4), 437-444, 1999.

[15] "Microarrays: Chipping away at the mysteries of science and medicine," http://www.ncbi.nlm.nih.gov/About/primer/microarrays.html.

[16] Y. Zhang and D. Adjeroh, "Prediction by partial approximate matching for lossless image compression," in *Proc. DCC 2005*, Snowbird, UT, Mar. 2005, pp.494.