On the Bitplane Compression of Microarray Images.

Rebecka Jörnsten, Yehuda Vardi, Cun-Hui Zhang.

Abstract. The microarray image technology is a new and powerful tool for studying the expression of thousands of genes simultaneously. Methods for image processing and statistical analysis are still under development, and results on microarray data from different sources are therefore rarely comparable. The urgent need for data formats and standards is recognized by researchers in the field. To facilitate the development of such standards, methods for efficient data sharing and transmission are necessary, that is compression. Microarray images come in pairs: two high precision 16 bits per pixel intensity scans ("red" and "green"). The genetic information is extracted from the two scans via segmentation, background correction and normalization of red-to-green image intensities. We present a compression scheme for microarray images that is based on an extension of the JPEG2000 lossless standard, used in conjunction with a robust L1 vector quantizer. The L1 vector quantizer is trained on microarray image data from a replicate experiment. Thus, the image pairs are encoded jointly. This ensures that the genetic information extraction is only marginally affected by the compression at compression ratios 8:1.

1. Introduction.

The cDNA microarray image technology is a powerful tool for monitoring the expressions of thousands of genes simultaneously. An experiment is comprised of hundreds of images, each image easily over 30MB. Since image processing and statistical analysis tools are still under development, the images are always kept. Current focus on the development of standards makes efficient data transmission an important problem. Though the cost of disk space for storage is decreasing, efficient transmission requires compression.

We have developed a compression scheme for microarray images, using a bitplane coding strategy. The *Most Significant Bitplane* (MSB) is constructed using the Segmented LOCO algorithm ([2], [1]). This bitplane contains most of the predictable structure of the microarray images. The MSB reconstructions are sufficient for image segmentation and visual inspection of microarray images. We refer to the reconstruction error of the MSB as the *Least Significant Bitplane* (LSB). This bitplane contains fine structure and detail. We reconstruct the LSB at locally

Key words and phrases. L1 Vector Quantization, cDNA Microarrays, Compression.

varying degree of distortion, or loss. The degree of loss can be chosen by the user, or be based on parameters such as local image intensities. We compare L1 and L2 norm scalar quantization (SQ) schemes and vector quantization (VQ) schemes for lossy reconstruction of the LSB. We find that the best results are achieved with a L1 norm vector quantizer design. We train the L1 vector quantizer on replicate image data, using the modified Weiszfeld algorithm of Vardi and Zhang ([6]) to construct the quantizer reconstruction levels (corresponding to multi-variate medians). We use a nearest neighbor allocation and simulating annealing scheme to find the optimal quantizer bin partitions. The bitrate of the compressed microarray images is determined by the number of quantization bins used in different regions of the images. We find that a bitrate of $\sim$4 bpp (cmp 32 bpp uncompressed) is sufficient for most tasks, such as image segmentation, and gene expression level extraction with a variety of existing methods.

The cDNA microarray image technology measures gene "activity" through relative mRNA abundance. DNA probes, corresponding to genes, are spotted onto a microscopic glass slide by a robotic *arrayer*. A reference sample of mRNA is labeled with a green fluorescent dye (Cy3), and a sample of interest with a red dye (Cy5). The two mRNA samples are mixed and allowed to hybridize onto the array. A laser scan of the array produces two intensity images. The intensity ratio for each probe, or spot, is proportional to the relative abundance of mRNA in the two samples. The raw microarray image data thus consist of two high precision (16 bpp) scans. The images are structured, with intensity spots roughly located on a grid. An example of an image scan (gray level intensity) is shown in Figure 1. As can be seen from the figure, the spots are submerged in a noisy, non-stationary background. The spots are ideally circular and of equal size, but due to experimental variation this is often not the case in practice. The background exhibits an intensity drift across the array, and can be corrupted by high-intensity speckle noise. Spots are often smeared, or "bleed" into each other.

The image processing steps applied to microarray images are segmentation, background correction, and normalization. To conserve space, we give a brief description of some methods here, and refer to Yang et al for details ([4]). Segmentation identifies the target regions where hybridization occurred. Examples of schemes used are fixed, and adaptive circle segmentation, and adaptive shape segmentation. We need to correct for non-specific hybridization (hybridization with surface chemicals, non-probe material) in order to make experiments and spots comparable. This is done via background correction, where we estimate the *local* background intensities using control spots, or by sampling the background pixels. Alternatively, a robust background estimate can be obtained using filtering operations (min/max convolutions) ([4]). Ratios of background corrected mean spot intensities R/G are computed, and a log transform applied to compress the scale, and even out the skewed distribution. The quantity obtained is the vector of log intensity ratios $M = \log_2 \frac{R}{G}$. Other quantities of interest are measures of quality, e.g. spot variances and shapes, and mean log intensities $A = \log_2 \sqrt{RG}$. Normalization

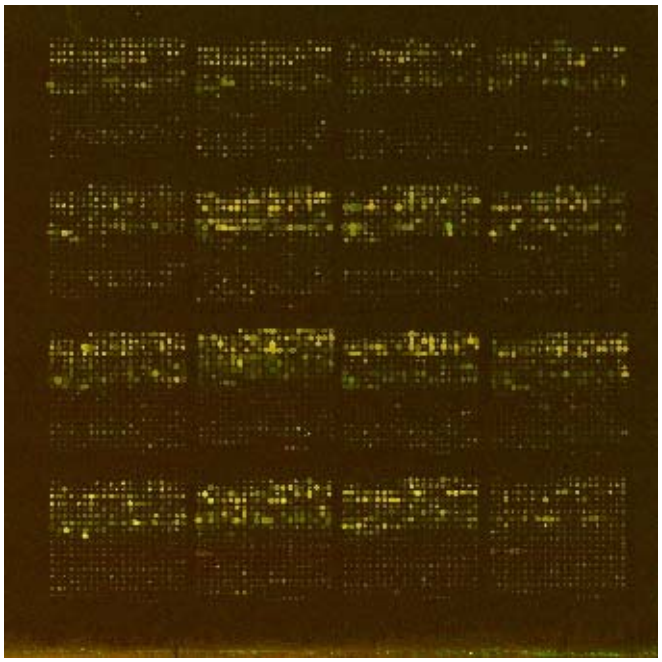


FIGURE 1. Microarray Image, 4×4 print-tips with 19×21 spots.

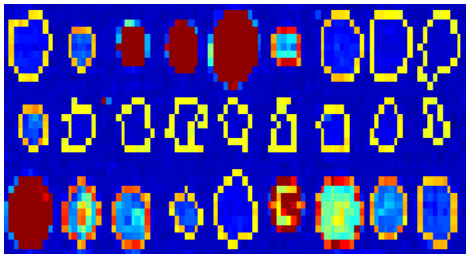


FIGURE 2. Adaptive shape segmentation.

of M removes systematic variation such as dye bias, and spatial effects. We apply the normalization scheme of Dudoit et al [5], which is non-linear, and print-tip specific. For each print-tip on the array, we estimate a locally linear fit of M on A. The residual vector from the fit $\tilde{M}$ is used in subsequent analysis.

It is widely recognized that the quality of microarray data is often poor. Since various image processing tools are applied to microarray images, results from different labs are rarely comparable. This motivates storage of image data, and is indicative of the need for data sharing to develop standards. To facilitate data sharing, we present a compression scheme tailored to the microarray application. Lossless compression of microarray images is easier for researchers to accept. However, since the images are noisy, there is no need to store/share them at full

precision. Indeed, if modest compression has a large impact on the extracted gene expressions, this should cast doubt on the extraction method. Our focus here is thus on preserving information in *lossily compressed* microarray images, such that the downstream tasks are unaffected by the compression. Here, we will discuss results obtained on 8 replicate image pairs from the APO AI experiment, courtesy of Matthew Callows, Genome Sciences, Lawrence Berkeley National Lab. The experiment is described in detail in [5].

In section 2 and 3 we describe our compression scheme. In section 4 we discuss results obtained on the 8 replicate arrays. We conclude in section 5 with ideas for future work.

2. Segmented LOCO - Compression of the Most Significant Bitplane.

We here consider using lossy reconstructions of the images for genetic information extraction and subsequent analysis. "Loss" is not clearly defined for microarray images since multi-step processing is used. Therefore, we cannot state a simple loss function over which to optimize the compression design. Though segmentation does not require lossless reconstructions, background correction and normalization are more difficult problems. In low intensity spot regions, small changes can have a large impact, especially on background correction. We therefore need to keep high precision in low intensity spot regions, but can use coarse image reconstructions near high intensity spots. Our aim is to ensure that the effect of compression is smaller than the variability between replicated experiments. We define this as *acceptable loss* for microarray image compression.

The performance of state-of-the-art image compression schemes on microarrays is poor. We list some of the reasons why. Microarray images are very noisy. The regions of interest (ROI) are many, small (thousands, 8-16 pixel diameter), and close. Application of an image transform leads to smearing between spots, and wavelet-based algorithms are dominated by the edges around high intensity spots at low bitrates . Medical imaging lossless ROI coding is not possible since the background pixels of microarray images contain information. For these reasons, we take a spatial prediction approach. To avoid "smearing", we encode the spot and background image regions separately. We first transmit an *overhead* defining the ROI and background, i.e. a segmentation map. We then apply a segmented version of the JPEG2000 near-lossless standard, LOCO, to create a *Most Significant Bitplane* (MSB). This variant of LOCO is called *Segmented LOCO* ([1]).

We denote the red and green image scans by X^R and X^G respectively, and apply the segmented LOCO algorithm with an "error bound" δ. This produces an image approximation $X^R_{MS} = LOCO(X^R)_\delta$, which we refer to as the Most Significant Bitplane (MSB) of X^R, and similarly for X^G. The MSBs represent $2^{16-\log_2(2\delta+1)}$ bits per pixel precision reconstructions of X^R and X^G, using a pixel domain predictive method (details below). We also form the *Least Significant Bitplanes* (LSB) of X^R and X^G, as $X^R_{LS} = X^R - X^R_{MS}$, and similarly for X^G_{LS}. The

range of the LSBs is by construction $[-\delta, \delta + 1]$. The MSBs contain most of the easily predicted structure of the images, and these reconstructions are sufficient for many tasks (image segmentation, extraction of high-intensity spots, visual inspection). The LSBs contain more detailed information. To ensure that the genetic information extraction for low intensity spots is not affected by the compression, we may need to add a lossless or lossy reconstruction of the LSBs to the MSBs in low intensity spot regions of the images. We call the detailed image reconstructions $\hat{X}^j = X^j_{MS} + \hat{X}^j_{LS},\ j = R, G$, where $\hat{X}^j_{LS}$ is a lossy or lossless reconstruction of X^j_{LS}. In section 4 we discuss how to choose the level of precision in the reconstructions of the LSBs. In the remainder of this section we describe the segmented LOCO algorithm that produces the MSBs.

The segmented LOCO scheme builds on LOCO ([2]), the JPEG2000 near-lossless standard. LOCO uses a robust causal spatial predictor, same as a simple edge detector. To improve on this fixed predictor, a context based *adaptive predictor* is also applied. The contexts are defined by the vector of quantized local gradients. Each triplet of gradients (horizontal (2), vertical) forms a context class. Based on the past performance of the context predictors, an adaptive estimate of prediction bias is obtained, and added to the fixed prediction. The prediction errors are quantized with a uniform quantizer (UQ) with bin-widths $2\delta + 1$, and bin-center reconstructions. This puts bound δ on the maximum pixel-wise error. The quantized errors follow an approximate geometric distribution, and can be efficiently encoded using Golomb codes. A separate Golomb code is used for each context class, defined by Golomb parameters k. Efficient encoding of the smooth regions of an image is achieved by means of a runlength code. Run interruption pixels are encoded using similar techniques as for regular pixels. Prediction errors for each context class are stored and used to update k (details in [2], [1]).

The overhead of the Segmented LOCO algorithm contains the segmentation map, spot means and standard deviations, and local background means and standard deviations. The spot and background means are encoded using adaptive Lempel-Ziv, and the standard deviations encoded conditional on the means. The segmentation map is encoded using a chain code. The average cost of the overhead for the 16 images from the replicate experiment is 0.376 bpp. If no re-processing of the images is needed, the overhead contains *all* relevant information for downstream analysis. In addition, it contains spot quality measurements such as spot shapes, and variances.

To prevent smearing we encode the spots and background separately. The spots are too small for adaptive prediction, so we apply fixed prediction and a fixed Golomb code for each spot. To avoid additional overhead, an approximate estimate of the optimal Golomb parameter is used, $\hat{k} = \max(0, \lceil \log_2(\lfloor \frac{\bar{A}/1.3+\delta}{2\delta+1} \rfloor) \rceil)$, where $\bar{A}$ is the MAD estimate of spot standard deviation, and 1.3 is an approximation factor estimated from the data. We encode the spots in a row scan manner, with missing context pixels (background) filled in with the spot means from the overhead. The background is encoded in a row scan fashion in image *sub-blocks*, which allows

for subset reconstruction. We fill in the missing context pixels (spots) with values equal to the local background means (from the overhead). The background context Golomb parameters are estimated as $\hat{k} = max(0, \lceil \log_2(Au/N) \rceil)$, where $Au = \lfloor A/(2\delta+1) \rfloor$. A is the accumulated absolute prediction error within the current pixel context class, and N the context counter. If the context of the pixel indicates a smooth region we apply a *segmented* runlength coding strategy. If a spot is encountered during a run, we skip ahead to the next background pixel. We compute the vector of local gradients at the new position, and if the maximum gradient difference is smaller than δ we continue the run. Runs interrupted by the gradient criterion are encoded as "expected interruptions", since these criteria are available at the decoder from causal information and from the overhead. Other interruptions are encoded in the same manner as in standard LOCO.

After prediction and quantization, the quantization errors (LSBs, $X^j_{LS},\ j = R, G$) are close to uniformly distributed in the range $[-\delta, \delta+1]$. Thus, we cannot reduce the first order entropy ($\sim \log_2(2\delta+1)$) much via predictive coding. By encoding the *sign error planes* of the LSBs we can achieve the entropy bitrate. The first sign error planes are defined as $sign[X^j_L S],\ j = R, G$. These $\{-1, 1\}$ images can be losslessly encoded with bitrate 1 bits per pixel (bpp) each. The new reduced error image ($X^j_{LS} - sign[X^j_{LS}]$) has range $[-\lfloor \delta/2 \rfloor, \lceil \delta/2 \rceil]$. We then encode the sign error plane of the reduced error image in the same fashion. Encoding the first i sign error planes results in a new error image with range $[-\lfloor \delta/2^i \rfloor, \lceil \delta/2^i \rceil]$. There are thus $\log_2(2\delta+1)$ sign error planes to encode for a lossless reconstruction. Despite this apparently inefficient code, sign error plane encoding achieves the LSB lossless bitrate $\log_2(2\delta+1)$, and gives better total (MSB+LSB) lossless compression results than the JPEG2000 lossless standard. For the 16 bits per pixel (bpp) microarray images we choose $\delta = 127$, which results in 8 bpp LSBs.

3. L1-VQ - Compression of the Least Significant Bitplane.

Though the sign error plane encoding results in a better lossless compression ratio than the JPEG2000 lossless standard, this does not guarantee that the *lossy* reconstructions can be used for genetic information extraction. A lossy reconstruction can be obtained by encoding only the first i_0 sign error planes. If $i_0 = 0$ we use reconstructions $\hat{X}^j = X^j_{MS}, j = R, G$. If $i_0 = \log_2(2\delta+1)$ the reconstructions are lossless. If we use an intermediate i_0 the image reconstructions are given by $\hat{X}^j = X^j_{MS} + \hat{X}^j_{LS}(i_0),\ j = R, G$, where $X^j_{LS} - \hat{X}^j_{LS}(i_0)$ has an error range $[-\lfloor \delta/2^{i_0} \rfloor, \lceil \delta/2^{i_0} \rceil]$. Sign error encoding corresponds to uniform scalar quantization (SQ) of the LSBs X^j_{LS}. Since marginally the LSBs of the red and green image scans are close to uniformly distributed, a uniform quantizer is optimal (in MSE sense). An empirical estimate of the optimal quantizer for each scans is indeed very close to a uniform quantizer, both under the L1 and L2 norm criterion. For a SQ with K bins, each pixel is thus mapped to the nearest of the K bin-centroids (or medians). The vector of quantization bin indices, denoting which bins the pixels have been allocated to, is encoded and transmitted to the decoder. In the uniform

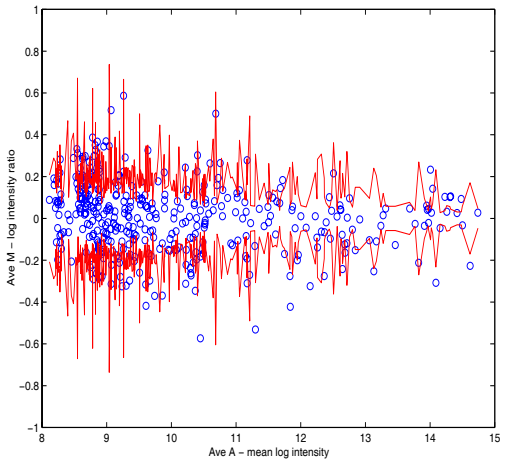


FIGURE 3. Approximate confidence interval for M_0 (pointwise).

distribution case the bitrate for a K level SQ is thus given by $log_2(K)$. If K is even, the K level SQ corresponds to the encoding of the first $K/2$ sign error planes. We can achieve locally varying image distortion by using different values for K in different regions of the image. This is called *bitallocation*.

Though marginally the LSBs are close to uniformly distributed, for which a uniform quantizer is optimal, a joint encoding scheme may be better. The reason for this is illustrated in Figure 3, where the mean normalized log intensity ratios M_0 are plotted against the mean (mean) log intensities A_0, for 399 gene spots from a print-tip on the array depicted in Figure 1. Figure 3 shows the pointwise approximate confidence interval for M_0, where the standard errors are estimated from the 8 replicate arrays and centered at 0. It is apparent that low intensity spots (small A_0) carry a lot more uncertainty that high intensity spots. The log intensity ratio is unstable when both the red and green intensities are small. The scalar quantization and separate encoding of the red and green image scans for lossy reconstruction can thus have a large impact on the small intensity spots. In the worst case we get maximum errors $\delta/2^{i_0}$ of opposite signs for the two scans, at each pixel location. A vector quantization scheme has many benefits. Firstly, we can avoid the hazardous situation just described where the "cumulative" error of the two image scans is large. Secondly, a vector quantization in the spatial domain (within each scan) can reduce the bitrate further and allows for more flexibility in bitallocation. A vector quantizer (VQ) maps a vector of pixels to a vector-valued bin-centroid or multi-variate median. Together, spatial image blocks of size $d \times d$ for the two scan form a three-dimensional block $2 \times (d \times d)$. We can train a VQ on the LSBs from the 8 replicate arrays ($\sim$ 55000 blocks) to minimize the empirical $L1$ or $L2$ loss of assigning blocks to their nearest quantization bin centroids (or multi-variate medians). Each three-dimensional pixel block is thus allocated to a bin, and the bin index encoded and transmitted to the decoder. If the number of

VQ quantization bins is K, the bitrate is $\log_2(K)$. We also need to transmit the VQ code book, that is the bin centroids (or medians). However, for the very large microarray images the cost of transmitting the $K \times (2 \times (d \times d))$ size code book is negligible compared to the cost of transmitting the MSBs and quantized LSBs. The VQ scheme is geared toward minimizing the L2 or L1 norm *joint* loss for the two image scans, and we can thus avoid the "cumulative" error effect in low spot intensity regions.

We now give a brief description of the bitallocation setup we employ, that is the number of quantization bins K we use in different image regions. Spots that are affected by compression are associated with a small A. We thus pick levels of A and allot more bits (large K) to the small A regions of the images, and fewer bits (small K) to the high A regions. The A for each spot region is available from the overhead of the segmented LOCO algorithm. We use bitallocation levels $A = 9, 10$ and 11 (see Fig. 3). When we encode the red and green LSBs separately, the $A < 9$ regions are encoded at rate 1.58 bpp, whereas the $9 \leq A < 11$ regions are encoded at rate 1 bpp. Above $A = 11$ we use only the MSB reconstructions. This means that where $A < 9$ a scalar quantizer (SQ) with K =3 quantization bins is constructed separately for the red and green scans. In regions where $9 \leq A < 11$ we use $K = 2$ and for all other image regions $K = 0$. The vector quantizer design is more flexible. Here we choose to quantize $2 \times (2 \times 2)$ image blocks. That is, the green and red scans are jointly quantized in spatial blocks of size 2×2 pixels. In the $A < 10$ regions we use $K = 512$ quantizer bins for the VQ setup, corresponding to a bitrate of 1.18 bpp, Where $10 \leq A < 11$ we use $K = 256$ quantizer bins, corresponding to bitrate 1 bpp. The average bitrate for the 8 replicate arrays is 4.21 bpp using the VQ setup, and 4.27 bpp in the SQ case. The VQ bitrate is the actual bitrate, not a first order entropy approximation. This includes the 0.376 bpp overhead, and is the bitrate for *both* image scans (cmp (2×16) 32 bpp uncompressed). The SQ bitrate is based on a first order entropy approximation for the image regions where $A < 9$. We compare both L1-norm and L2-norm scalar and vector quantizers for the compression of the microarray images. Results are presented in the next section.

4. Results.

In this paper we show results obtained using adaptive shape segmentation and robust filtering background correction (section 1), though similar results were obtained with other methods but omitted here to conserve space. We first compare lossless compression ratios using Segmented LOCO with sign error plane encoding, standard LOCO, and the progressive SPIHT algorithm. As a baseline for lossy + error plane coding we also compare with a wavelet zero-tree coding + residual entropy coding scheme (WT+ZT+EC). This scheme has been found to be very efficient for the encoding of natural and medical images. On microarrays however, the results using SPIHT and (WT+ZT+EC) are dismal, with lossless compression ratios 1.65:1 and 1.69:1 respectively (Table 1). The LOCO compression ratio

is 1.85:1. We get better though similar results with segmented LOCO, 1.87:1, but our bitrate also includes the overhead cost (0.376 bpp). We cannot hope to achieve much better lossless compression ratios with any method. The 8 bpp LSBs are almost random, and unpredictable. This puts a ceiling of 2:1 on the lossless compression ratio.

Method	Compression ratio
SPIHT	1.65:1
WT+ZT+EC	1.72:1
LOCO	1.85:1
Segmented LOCO	1.87:1

TABLE 1. Lossless compression ratios.

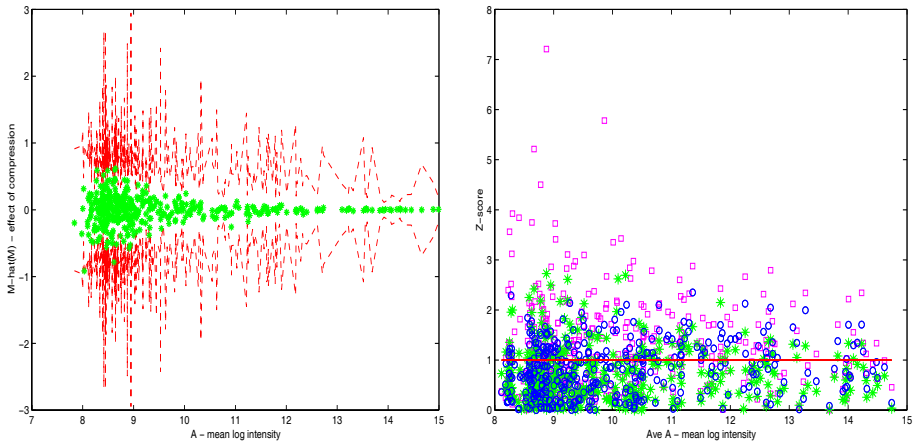
We also compare the extracted information from *lossy* microarray images, to the extracted information from the lossless image data. Both the scalar quantization and vector quantization scheme at bitrates ~ 4 bpp are able to preserve large absolute $\tilde{M}$ values accurately. As can be seen in Figure 4 (a), the difference between using lossy reconstructions and lossless ones is negligible for large $\tilde{M}$, and for large A. For small $\tilde{M}$ and small A, the effect of compression is more apparent. However, as can be seen in Figure 4 (a) the effect is smaller than the noise level of the data. The dashed lines in Figure 4 (a) correspond to 2 times the standard deviation of the 8 replicates, that is the array-to-array variability, and the lossy-lossless differences are well within these SD bands. We can draw this same conclusion by computing the "z-score" for each spot. We compute the mean spot log intensity ratio M_0 over all replicates, and the standard deviation σ_M. We compute the z-score $z = |\hat{M} - M_0|/\sigma_M$, where $\hat{M}$ is the log intensity ratio of a spot using the lossy or lossless image reconstructions for a single array. The z-score for array 2 using the lossless reconstructions are shown in Figure 4 (b) (circles), together with the z-score using the VQ (L1 norm) lossy reconstructions of array 2 at bitrate 4.21 bpp (stars), and the SQ (L1 norm) lossy reconstructions at bitrate 4.27 bpp (squares). The VQ z-scores rarely exceed 2, about as often as the z-scores of the lossless reconstructions. In fact, for $A > 12$ the VQ z-scores are *smaller* than the lossless z-score, indicating that the VQ compression acts as a type of shrinkage for the large $\tilde{M}$ and A spots. We can thus think of VQ compression as "denoising" of the microarray images, prior to genetic information extraction. The SQ z-scores are much higher, especially for small A. This indicates that a joint encoding scheme of the green and red image scans better deals with the problem of compression and instability in the low A regions.

To summarize the effect of compression we also compute the average L1-risk for each spot, over the 8 replicate arrays. By the average L1-risk we refer to the quantity $\frac{1}{8}\sum_{r=1}^{8}\sum_i |\hat{M}_{r,i} - M_{0,i}|$, where i denotes the spot number and r the replicate array number. In Table 2 we see that the VQ schemes (both L1 and L2) show comparable risk to the risk of using lossless reconstructions, with the L1-VQ

scheme performing slightly better than the L2-VQ. The SQ schemes have a much higher risk.

Quantizer	**L1-risk**	Bias
Lossless	85.3	0.0048
L1-VQ	87.9	-0.0011
L2-VQ	89.0	-0.0052
L1-SQ	95.8	0.0133
L2-SQ	96.2	0.0981

TABLE 2. Average L1-risk and bias over the 8 treatment arrays.



(a) (b)

FIGURE 4. (a) The effect of compression is below the noise level (VQ reconstructions - lossless reconstruction, with 2*SD bands from the 8 replicate arrays). (b) z-scores: lossless (circles), VQ (stars), SQ (squares). VQ and lossless z-scores are comparable. The SQ z-scores are much higher.

5. Conclusion.

We present a lossless and lossy compression scheme for microarray images. The lossless bitrate is lower than the JPEG2000 lossless standard, and lower than some of the best reported methods in the literature. At bitrates $\sim$4 bpp, we find that the tasks of image segmentation, and genetic information extraction are only marginally affected by the lossy compression. The effect is smaller than the array-to-array variability, that is below the level of the noise in the data. The information in low intensity spots is better preserved by the joint encoding using an L1 norm

vector quantizer (L1-VQ) of the two image scans, compared to using separate encoding schemes (SQ), or L2 vector quantizers.

This paper presents some initial results for L1-VQ and bitplane microarray image compression. We intend to expand the study to a range of bitrates and bitallocation schemes, and on microarrays from other sources. It is possible that better results can be obtained via context based VQ schemes, such as TSVQ (tree structured VQ). The microarray image technology is becoming a standard tool in many labs and vast quantities of image data is produced daily. Therefore, the development of efficient and reliable compression tools is an important task.

References

[1] Jornsten, R. *Data compression and its statistical implications, with an application to the analysis of microarray images*, PhD Thesis, Department of Statistics, UC Berkeley, 2001.

[2] Nerhav, M., Seroussi, G., Weinberger, M. *Modeling and low-complexity adaptive coding for image prediction residuals*, Int'l Conference on Image Processing (1996), Lausanne.

[3] Said, A. and Perlman, W. *Reversible image compression via multiresolution representation and predictive coding*, Proc SPIE, **2094** (1993), 664-674.

[4] Yang, Y., Buckley M.J, Dudoit, S. Speed, T. *Comparisons of methods for image analysis on cDNA microarray data.* UC Berkeley, Techn. report. 2000.

[5] Yang, Y. H., Dudoit S., Luu P. and Speed, T. *Normalization for cDNA Microarray Data*, SPIE BiOS 2001, San Jose, California, January 2001.

[6] Y. Vardi, and C-H. Zhang. *The multi-variate L1-median and associated data depth.* Proceedings of the National Academy of Sciences, **97** (2000), 1423-1426.

Department of Statistics Rutgers University, PISCATAWAY, NJ, 08854, USA
E-mail address: `rebecka,cunhui,vardi@stat.rutgers.edu`