Progressive Lossy-to-Lossless Compression of DNA Microarray Images

Miguel Hernández-Cabronero*, Ian Blanes, *Member, IEEE*, Armando J. Pinho, *Member, IEEE*, Michael W. Marcellin, *Fellow, IEEE* and Joan Serra-Sagristà, *Senior Member, IEEE*

***Abstract*—The analysis techniques applied to DNA microarray images are under active development. As new techniques become available, it will be useful to apply them to existing microarray images to obtain more accurate results. The compression of these images can be a useful tool to alleviate the costs associated to their storage and transmission. The recently proposed Relative Quantizer (RQ) coder provides the most competitive lossy compression ratios while introducing only acceptable changes in the images. However, images compressed with the RQ coder can only be reconstructed with a limited quality, determined before compression. In this work, a progressive lossy-to-lossless scheme is presented to solve this problem. Firstly, the regular structure of the RQ intervals is exploited to define a lossy-to-lossless coding algorithm called the Progressive RQ (PRQ) coder. Secondly, an enhanced version that prioritizes a region of interest, called the PRQ-ROI coder, is described. Experiments indicate that the PRQ coder offers progressivity with lossless and lossy coding performance almost identical to the best techniques in the literature, none of which is progressive. In turn, the PRQ-ROI exhibits very similar lossless coding results with better rate-distortion performance than both the RQ and PRQ coders.**

***Index Terms*—DNA microarray images, Image compression, Quantization**

I. Introduction

DNA microarrays are a state-of-the-art tool in biology and biomedicine, employed to monitor in parallel the function and regulation of thousands of genes of many different organisms [1]. Recent publications based on DNA microarray data can be found in the literature [2]–[11]. When a DNA microarray experiment is performed, two biological samples are put on a microarray chip, which is then scanned to produce two grayscale images. These images are analyzed to extract the genetic data of interest The different parts of the analysis process are under active development [12]–[28]. This work does not focus on the analysis of DNA microarray images and, hence, a discussion of previous works is out of scope. As new analysis techniques are developed, it will be desirable to apply them to obtain more accurate genetic data from previously performed experiments. However, repeating all parts of the experiment is usually not an option because the required biological samples may not be available some time after performing the original experiment, or because the re-analysis may need to be performed in another laboratory. Therefore, storing the DNA microarray images is paramount to enable future, more accurate genetic data extraction. In order to facilitate the management and transmission of these images, image compression emerges as a valuable tool.

The lossless compression of DNA microarray images has proven to be a very challenging task. Compression ratios significantly better than 2:1 are not generally obtained even by algorithms specifically designed for such images [29]. On the other hand, lossy coders can yield arbitrary compression ratios at the cost of modifying the images. Subsequent analysis techniques may be distorted by these modifications, although sufficiently small distortions can be considered acceptable [30]–[32]. Several generic image compression approaches (or adaptations thereof) have been applied to DNA microarray images [30], [31], [33]–[36]. Since these coding techniques are not specifically designed with the analysis of microarray images in mind, results for these methods may not be optimal. The recently proposed Relative Quantizer (RQ) coder [29], expressly designed for this type of images, introduces only acceptable changes in the images and exhibits higher rate-distortion performance than all previously existing lossy coding algorithms. However, it does not allow a progressive lossy-to-lossless reconstruction of the compressed images. In this work, a progressive lossy-to-lossless compression scheme based on the RQ coder is proposed.

The rest of this paper is structured as follows. Section II describes the most relevant features of the technique presented in [29]. A lossy-to-lossless coding approach is proposed in Section III and its compression performance is analyzed in Section IV. Finally, Section V draws some conclusions.

II. The Relative Quantizer

The lossy compression method presented in [29] is based on a non-uniform scalar quantizer called the Relative Quantizer (RQ). This quantizer is applied independently to unsigned pixels of images of bitdepth $B \geq 1$. The quantization intervals of the RQ are fully determined by an integer parameter $k \in \{1, \ldots, B\}$ that controls the precision of the quantization process. The first 2^k intervals have size 1 and, hence, pixel intensities in $\{0, \ldots, 2^k - 1\}$ are preserved losslessly. The next 2^{k-1} intervals have each size 2^1 and the following 2^{k-1}

This work has been partially funded by FEDER, the Spanish Government (MINECO) and the Catalan Government under projects TIN2015-71126-R, TIN2012-38102-C03-03, FPU AP2010-0172 and 2014SGR-691.

*M. Hernández-Cabronero, I. Blanes and J. Serra-Sagristà are with the Universitat Autònoma de Barcelona, Bellaterra 08193, Spain (e-mail: mhernandez@deic.uab.cat).

A. J. Pinho is with the Signal Processing Lab, DETI/IEETA, University of Aveiro, 3810-193, Portugal.

M. W. Marcellin is with the University of Arizona, Tucson, AZ 85721-0104, USA.

intervals have size 2^2. Each successive group of 2^{k-1} intervals has size $2^3, 2^4$, etc until the last group, which contains intervals of size 2^{B-k}. A diagram of the quantization intervals of the RQ and their associated indices is shown in Fig. 1 for $B = 4$ and all possible values of k. This definition of the RQ was developed to preserve information that is important to microarray image analysis. In this regard, RQ was shown to yield superior results to modern bit plane coding architectures which employ uniform quantization. Specifically, it yields very competitive compression performance while introducing only demonstrably acceptable distortion in subsequent analysis results [29]. Thus, the lossy-to-lossless scheme proposed in Section III is based on the RQ.

In [29], the quantization indices produced by the RQ are coded with the lossless compressor described in [37]. The bitplanes are compressed in raster order using an arithmetic coder (AC), beginning with the most significant bitplane. The probabilities used to drive the AC are computed based on a model that employs information from previously encoded bits. The position of the bits employed to extract that information is referred to as the *context*. A key property of this algorithm is the fact that the actual context employed in each bitplane is calculated at compression time. To calculate the best context, a greedy algorithm evaluates each candidate context by compressing a rectangular region of the center of the image using that context. After that, the best candidate is selected. Thence, this algorithm is able to very accurately adapt to the individual properties of each bitplane. As a result of this flexibility, this compressor exhibits the best performance for this type of images and the RQ-quantized versions thereof. Therefore, a version of this algorithm is employed in the lossy-to-lossless coder proposed in Section III.

The main drawback of the RQ coder is the fact that compressed images can only be reconstructed up to a certain precision determined by the chosen value of k. If the user wants to reconstruct the image losslessly or at different qualities, several compressed versions of the image need be kept. This approach multiplies the storage requirements and, thus, is not practical. In what follows, a microarray-specific progressive lossy-to-lossless scheme based on the RQ and the compression algorithm from [37] is presented and its performance is analyzed.

III. Progressive Lossy-to-Lossless Coding

A. *Progressive Representation of DNA Microarray Images*

In this section, the relationship between the quantization intervals of the RQ for different values of k is exploited to define a progressive representation of DNA microarray image pixels. As described in Section II, the quantization intervals of the RQ for a given image bitdepth B are determined by an integer parameter $k \in \{1, \ldots, B\}$. The parameter $k = 1$ corresponds to the most aggressive (least precise) quantization and $k = B$ corresponds to not performing any quantization. A key observation is that the RQ with parameter $k_0 > 1$ is actually a *refined* version of the RQ with parameter $k_0 - 1$. More specifically, all quantizer intervals that include more than one pixel intensity for $k_0 - 1$ are divided in half for k_0. Let us consider the case $B = 4$, shown in Fig. 1. The interval with index 3 for $k = 1$ is divided into the intervals with indices 4 and 5 for $k = 2$. Likewise, the interval with index 6 for $k = 2$ is divided into the intervals with indices 8 and 9 for $k = 3$. On the other hand, intervals that contain only one pixel intensity cannot be further refined and their index is identical to the contained pixel intensity. For example, for $k = 3$, only intervals with index $i > 7$ can be refined. For $k = B$, all intervals contain only one pixel intensity and their index is identical to that intensity. Analogous relationships between the quantization intervals apply for any $B > 1$ including $B = 16$, the bitdepth of DNA microarray images.

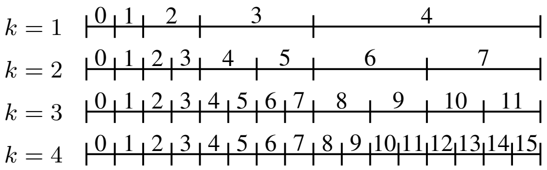

Fig. 1: Quantization intervals of the RQ for $B = 4$ and all possible values of k. The index of each interval is also indicated.

The previous observations can be formalized to enable a lossless, progressive representation of these type of images. Given a pixel intensity p, let $\mathrm{RQ}_k(p)$ be the quantization interval corresponding to p for the RQ with parameter k and let $|\mathrm{RQ}_k(p)|$ be the number of intensities assigned to that interval. Clearly, $\mathrm{RQ}_B(p) \subset \mathrm{RQ}_{B-1}(p) \subset \cdots \subset \mathrm{RQ}_1(p)$. If the index of the $\mathrm{RQ}_1(p)$ interval is encoded, the decoder knows that $p \in \mathrm{RQ}_1(p)$. If $|\mathrm{RQ}_1(p)| > 1$, the exact value of p is not known, and additional information can be encoded to allow a more precise reconstruction of that pixel. Since $|\mathrm{RQ}_1(p)| > 1$, the interval $\mathrm{RQ}_1(p)$ is divided into two intervals of size $|\mathrm{RQ}_1(p)|/2$ for $k = 2$, as in the example above. Therefore, only one *refinement bit* is needed to signal which of these two intervals corresponds to $\mathrm{RQ}_2(p)$. Hereinafter, a refinement bit equal to 0 (resp. 1) is used when $\mathrm{RQ}_{k+1}(p)$ equals the lower (resp. upper) half of $\mathrm{RQ}_k(p)$. By encoding this bit, the range of possible reconstruction values is halved and, hence, the precision is doubled. Likewise, if $\mathrm{RQ}_2(p)$ comprises more than one value, another refinement bit can be encoded so that the decoder can determine which candidate interval corresponds to $\mathrm{RQ}_3(p)$. By successively applying this refinement process, it is possible to sequentially determine $\mathrm{RQ}_1(p), \ldots, \mathrm{RQ}_B(p)$, *i.e.*, the quantization indices corresponding to p for all values of k. Recall that, by definition, $\mathrm{RQ}_B(p)$ allows a lossless reconstruction of the original pixel intensity p.

Based on this, we define here a lossless, progressive RQ-based (PRQ) representation of the pixel p as

$$\mathrm{PRQ}(p) = \mathrm{RQ}_1(p), \Delta_{1\to 2}(p), \ldots, \Delta_{B-1\to B}(p), \qquad (1)$$

where $\Delta_{k\to k+1}(p)$ is the refinement bit needed to obtain $\mathrm{RQ}_{k+1}(p)$ from $\mathrm{RQ}_k(p)$. For instance let $B = 4$ and $p = 11$. As can be seen in Fig. 1, $\mathrm{RQ}_1(11) = 4$, $\mathrm{RQ}_2(11) = 6$, $\mathrm{RQ}_3(11) = 9$ and $\mathrm{RQ}_4(11) = 11$. Therefore, the refinement bits are $\Delta_{1\to 2}(11) = 0$ (lower half), $\Delta_{2\to 3}(11) = 1$ (upper half) and $\Delta_{3\to 4}(11) = 1$ (upper half). Thus, $\mathrm{PRQ}(11) =$

$4, 0, 1, 1$. Note that $|\mathrm{RQ}_n(p)| = 1$ implies that the interval need not be refined and, hence, $\Delta_{m\to m+1}(p)$ need not be signaled for any $m \geq n$. For instance, for $B = 4$ and $p = 3$, $\mathrm{PRQ}(3) = 2, 1$.

The PRQ representation of any pixel p can also be expressed in binary form. As shown in [29], the total number of quantization intervals of the RQ with parameter k is given by $(B - k + 2)2^{k-1}$. Hence, for DNA microarray images ($B = 16$), 17 quantization intervals are employed for $k = 1$. Thus, the index of $\mathrm{RQ}_1(p)$ can be expressed using $\lceil \log_2 17 \rceil = 5$ bits. Once the first element of the PRQ is signaled, each refinement bit $\Delta_{k\to k+1}(p)$ provides enough information to recover the quantization interval index for the RQ with the next value of k. Therefore, at most $B - 1 = 15$ such refinement bits need be coded to enable the recovery of the original pixel value p. By sequentially appending the refinement bits to the index of $\mathrm{RQ}_1(p)$, any pixel can be expressed in a progressive lossy-to-lossless way by signaling at most 20 bits.

B. *Progressive Compression*

In what follows, a progressive lossy-to-lossless coder for DNA microarray images based on the PRQ representation is introduced. This coder is hereinafter referred to as the *PRQ coder*.

When compressing an image, its 20-bpp PRQ representation is first computed. The resulting data are coded with a version of the algorithm introduced in [37]. This version includes two modifications to adapt the original algorithm to the particularities of the PRQ representation and improve its coding efficiency. As described in Section II, the compressor in [37] proceeds by sequentially coding each of the image bitplanes, beginning with the most significant bit. Hence, the elements of the PRQ representation are coded in the order described in Equation (1). As explained in III-A, some of the 15 refinement bits are not needed for a given pixel p when $|\mathrm{RQ}_n(p)| = 1$ for $n < 16$. Therefore, the first modification consists in not coding any unneeded refinement bit of the PRQ representation. As described in II, the original algorithm selects the optimal context by comparing several candidates, each of which is evaluated by compressing a rectangular $N \times M$ region in the center of the image. If that region is not representative of the whole image, a sub-optimal candidate context would be selected. Thus, the second modification to [37] consists in evaluating the candidate contexts by compressing NM pixels uniformly sampled across the image. Since sampled pixels are not confined in a relatively small region of the image, an overall more precise context can be selected, which can improve the compression performance at a similar computational cost.

When the image is decompressed, the 20-bpp PRQ representation of the image is first obtained by applying a version of the decoder presented in [37]. This version includes modifications analogous to those described above to make it compatible with the output of the encoder. For each pixel p, the index of the $\mathrm{RQ}_1(p)$ interval is then obtained from the 5 most significant bits of its PRQ representation. Finally, the refinement information contained in subsequent bitplanes is successively applied until $\mathrm{RQ}_{16}(p)$ is recovered. By definition, the index of this interval is identical to the original pixel intensity p and, thus, the image can be losslessly recovered. As discussed later in Section IV, the 20 bpp PRQ representation can be losslessly coded with approximately the same performance as the original 16 bpp pixels.

The correct decoding of truncated data is required to enable a progressive lossy-to-lossless coding pipeline. Since the algorithm proposed in [37] is designed for purely lossless coding, it needs be adapted to accept truncated versions of the encoded data produced at the encoder. If 5 or more complete bitplanes are decoded before the end of file (EOF) is reached, $\mathrm{RQ}_1(p)$ and possibly some refinement bits are available for each pixel p. Hence, $\mathrm{RQ}_n(p)$ can be computed for some n with $1 \leq n \leq 16$, depending on the number of refinement bits available. As in the original RQ, the recovered value of p is calculated using the interval midpoint of $\mathrm{RQ}_n(p)$, rounded up to the next integer. If less than 5 complete bitplanes are available, then the index of $\mathrm{RQ}_1(p)$ –corresponding to $k = 1$– would need to be estimated for some pixels. Even though it is possible to do so, this scenario should be generally avoided due to the relatively high distortion introduced in subsequent analysis processes for $k = 1$ [29].

An additional enhancement is described now to improve the coding performance of the progressive lossy-to-lossless PRQ coder described above. The original algorithm introduced in [37] assigns equal priority to all pixels of the image. Thus, all bits of a bitplane are coded before proceeding to the next bitplane. However, almost all information relevant to the analysis of microarray images is contained in pixels belonging to the so-called *spots* [1]. For a given image, all such spots can be enclosed within a single rectangular *region of interest* (ROI). Pixels that do not lie within the ROI are relatively unimportant for subsequent analysis processes. If all bitplanes inside the ROI are coded prior to any bit planes of pixels from outside the ROI, relevant information is placed closer to the beginning of the compressed file, significantly improving the rate-distortion performance of the PRQ coder, while still allowing for lossless decoding of the entire image, if desired. The version of the PRQ coder that includes the ROI prioritization enhancement is hereinafter referred to as the *PRQ-ROI coder*.

IV. Compression Performance

The compression performance of the proposed PRQ and PRQ-ROI coders is addressed in this section. First, the bitrate required to obtain a lossless compression is surveyed. After that, the rate-distortion results yielded by these lossy-to-lossless algorithms are discussed.

To test the lossless compression efficiency, 228 real DNA microarray images where compressed with the proposed coders. This corpus contains most images used for the benchmarking of microarray image compressors in the literature. The average compression results in bpp –calculated as the total number of compressed bits required for a lossless recovery of the images divided by the total number of pixels in all images– is provided in Table I. Results for the best-performing lossless compressor for DNA microarray images

TABLE I: Average lossless compression bitrate in bpp and execution time expressed in relation to [37].

	Neves and Pinho [37]	PRQ coder	PRQ-ROI coder
Bitrate	7.909	7.871	7.892
Time	100%	97.38%	99.76%

–first published in [37]– are also provided for comparison. The average time for compressing and decompressing 4 times each of the 228 images, expressed as a percentage of the execution time of [37], is also provided in the table.

It can be observed that both the PRQ and the PRQ-ROI coders achieve a slightly better lossless coding efficiency than the best state-of-the-art lossless compressor [37], even though the latter does not offer lossy-to-lossless capabilities. This can be explained by the modifications described in Section III-B (skipping of unneeded refinement bits and improved candidate context evaluation). It can also be observed that the PRQ-ROI coder yields a lossless compression performance almost identical to that of the PRQ coder. This suggests that the lossless coding overhead due to the ROI prioritization capabilities is negligible. As can be seen in the table, the PRQ and PRQ-ROI coders are, respectively, 2.62% and 0.24% faster than the non-progressive algorithm from [37]. These differences are due to the fact that the PRQ-based coders skip the coding of all unneeded refinement bits, which compensates for the larger amount of bitplanes that need be processed.

Since the proposed PRQ coder is a progressive lossy-to-lossless algorithm, it is paramount to analyze its rate-distortion performance. It is worth noting that DNA microarrays are not typically viewed by humans. Rather, they are analyzed by specialized software such as GenePix [38]. Thus, it is important to assess the amount of distortion introduced in the results of the analysis process. Traditional image distortion metrics such as MSE are not suitable for this purpose because they do not characterize the analysis result distortion [29]. That is, small MSE values do not necessarily correspond to small errors in the analysis. For this reason, the microarray-specific measures introduced in [29] –the $\mathrm{ARE}_{\mathrm{CRM}}$ and the FWDOC measures– are hereinafter employed to assess the distortion introduced in the images. Unlike traditional metrics, these microarray-specific measures directly compare the data yielded by real analysis software when applied to the original or the modified images. Hence, the $\mathrm{ARE}_{\mathrm{CRM}}$ and the FWDOC measures provide an accurate measure of the distortion introduced in subsequent analysis processes. In this work, 44 of the 228 aforementioned DNA microarray images are considered. The rest of the images could not be used because of the lack of publicly available analysis software compatible with these images.

First, the 44 images were compressed with the PRQ-Uniform and the PRQ-ROI coders and the resulting codestreams were truncated at 7 different lengths. The first truncation point was selected so that only the first element of the PRQ representation of each pixel is available. The next truncation point was chosen so that the first refinement bit is also available for all pixels. Each of the successive truncation points was selected so that exactly one more refinement bit is available for all pixels, as compared to the previous truncation point. For each truncation point and coder, a reconstructed version of the 44 images was then obtained. Finally, the two aforementioned distortion measures were calculated for each reconstructed version. The average distortion results for the PRQ and the PRQ-ROI coders as a function of the average bitrate after truncation is provided in Fig. 2 for the $\mathrm{ARE}_{\mathrm{CRM}}$ and the FWDOC measures, respectively. The rate-distortion results for the non-progressive RQ coder for $k \in \{1, \ldots, 7\}$ are also provided for comparison.

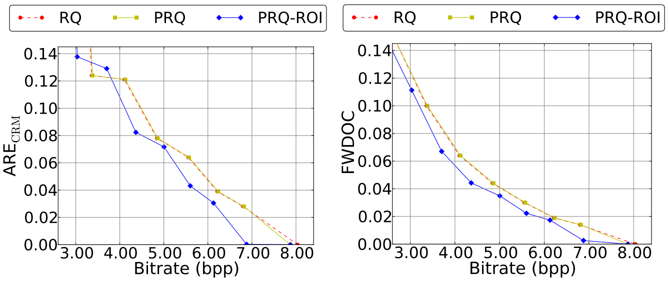

Fig. 2: Rate distortion results for the $\mathrm{ARE}_{\mathrm{CRM}}$ measure (left) and the FWDOC measure (right).

It can be observed that the PRQ and the RQ coders yield almost identical results for all tested bitrates. This suggests that the PRQ representation introduces only a negligible overhead even when only some of the refinement bits are coded. It can also be seen that the PRQ-ROI coder generally yields significantly better rate-distortion results than both the PRQ and RQ coders. This can be explained by the fact that the information important for subsequent analysis is coded before the relatively unimportant information of the background. In light of these data, it appears that the ROI-prioritization is an effective way of enhancing their rate-distortion performance.

V. Conclusions

Better analysis techniques for DNA microarray images are being actively investigated. Hence, it is convenient to store the images to enable future re-analysis of the data. The compression of this type of images is a useful tool to reduce the storage and management costs and to accelerate the sharing of these images. Lossy coding algorithms can yield high compression ratios introducing only acceptable distortion in subsequent analysis processes. A lossy compression method called Relative Quantization (RQ) was recently proposed. In spite of its competitive compression performance, an image coded with the RQ can only be reconstructed with a certain quality level determined before compression and it is not possible to recover the original image. This work introduces an original solution to this problem. First, a lossless representation of DNA microarray images is defined. Then the PRQ coder, a progressive lossy-to-lossless coder based on this representation, is proposed. Finally, an enhanced version of this coder that includes region-of-interest prioritization –the PRQ-ROI coder– is described. The proposed PRQ-ROI exhibits significantly better rate-distortion results than the non-progressive RQ coder without introducing any overhead in the lossless compression bitrate nor in the average execution time.

References

[1] S. Moore, "Making chips to probe genes," *IEEE Spectr.*, vol. 38, no. 3, pp. 54–60, Mar. 2001.

[2] J. Rysa, "Gene expression profiling of human calcific aortic valve disease," *Genomics Data*, vol. 7, pp. 107 – 108, 2016.

[3] R. I. Astuti, D. Watanabe, and H. Takagi, "Nitric oxide signaling and its role in oxidative stress response in schizosaccharomyces pombe," *Elsevier Nitric Oxide*, vol. 52, pp. 29 – 40, 2016.

[4] T. Kiba, "Ventromedial hypothalamic lesions downregulate multiple immune signaling pathways in rat pancreatic islets," *Neuroscience Letters*, vol. 610, pp. 177 – 181, 2016.

[5] C. Dzik, S. T. Reis, N. Viana, G. Brito, I. Paloppi, W. Nahas, M. Srougi, and K. R. M. Leite, "Analysis of the Gene Expression of Renal Cell Carcinomas after Neoadjuvant Treatment with Sunitinib: An Exploratory DNA Microarray Study," *Journal of Cancer Science & Therapy*, vol. 7, no. 12, 2015.

[6] J. S. Yook, M. Okamoto, R. Rakwal, J. Shibato, M. C. Lee, T. Matsui, H. K. Chang, J. Y. Cho, and H. Soya, "Astaxanthin supplementation enhances adult hippocampal neurogenesis and spatial memory in mice," *Molecular Nutrition & Food Research*, 2015.

[7] K. Ichinose, T. Ushigusa, A. Nishino, Y. Nakashima, T. Suzuki, Y. Horai, T. Koga, S.-y. Kawashiri, N. Iwamoto, M. Tamai, K. Arima, H. Nakamura, Y. Obata, K. Yamamoto, T. Origuchi, T. Nishino, A. Kawakami, and G. C. Tsokos, "Lupus nephritis IgG induces the expression of calcium/calmodulin-dependent kinase type IV in podocytes and alters their function," *Arthritis & Rheumatology*, 2015.

[8] X. Ma, S. Zheng, Y. Shu, Y. Wang, and X. Chen, "Association of the glu504lys polymorphism in the aldehyde dehydrogenase 2gene with endothelium-dependent dilation disorder in chinese han patients with essential hypertension," *Internal Medicine Journal*, 2015.

[9] K. Ishikawa, "Periostin in the pathogenesis of proliferative vitreoretinopathy," *Nippon Ganka Gakkai Zasshi*, vol. 119, no. 11, pp. 772–780, November 2015.

[10] Y. Li, T. Xiong, H. Wu, and Y. Yang, "Visual DNA microarray coupled with multiplex-PCR for the rapid detection of twelve genetically modified maize," *BioChip Journal*, pp. 1–6, 2015.

[11] Y. Hu, H. Yan, M. Mammel, and H. Chen, "Sequence-independent amplification coupled with dna microarray analysis for detection and genotyping of noroviruses," *AMB Express*, vol. 5, no. 1, 2015.

[12] K. Blekas, N. Galatsanos, A. Likas, and I. Lagaris, "Mixture Model Analysis of DNA Microarray Images," *IEEE Trans. Med. Imag.*, vol. 24, no. 7, pp. 901–909, Jul. 2005.

[13] J. Ho and W.-L. Hwang, "Automatic Microarray Spot Segmentation Using a Snake-Fisher Model," *IEEE Trans. Med. Imag.*, vol. 27, no. 6, pp. 847–857, Jun. 2008.

[14] E. Zacharia and D. Maroulis, "An Original Genetic Approach to the Fully Automatic Gridding of Microarray Images," *IEEE Trans. Med. Imag.*, vol. 27, no. 6, pp. 805–812, Jun. 2008.

[15] L. Rueda and I. Rezaeian, "A fully automatic gridding method for cDNA microarray images," *BMC Bioinformatics*, vol. 12, no. 1, p. 113, 2011.

[16] G.-F. Shao, F. Yang, Q. Zhang, Q.-F. Zhou, and L.-K. Luo, "Using the maximum between-class variance for automatic gridding of cDNA microarray images," *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 10, no. 1, pp. 181–192, Jan. 2013.

[17] L. Rueda, Ed., *Microarray Image and Data Analysis: Theory and Practice*. CRC Press, 2014.

[18] L. Srinivasan, Y. Rakvongthai, and S. Oraintara, "Microarray image denoising using complex Gaussian scale mixtures of complex wavelets," *Journal of Biomedical and Health Informatics*, vol. 18, no. 4, pp. 1423–1430, 2014.

[19] M. Muhammad Fuad, "Hierarchical clustering of dna microarray data using a hybrid of bacterial foraging and differential evolution," in *Theory and Practice of Natural Computing*, ser. Lecture Notes in Computer Science, A.-H. Dediu, L. Magdalena, and C. Martín-Vide, Eds. Springer International Publishing, 2015, vol. 9477, pp. 46–57.

[20] A. Fukushima, T. Paul, R. Shingaki, T. Koiso, S. Umeno, and K. Ueno, "A proposal for improvement of genotyping performance for ethnically homogeneous population using dna microarray," in *IEEE 37th Engineering in Medicine and Biology Society Conference (EMBC)*, Aug 2015, pp. 6816–6819.

[21] N. Chlis, E. Bei, K. Moirogiorgou, and M. Zervakis, "Extracting reliable gene expression signatures through stable bootstrap validation," in *IEEE 37th Engineering in Medicine and Biology Society Conference (EMBC)*, Aug 2015, pp. 4458–4461.

[22] B. Belean, M. Borda, J. Ackermann, I. Koch, and O. Balacescu, "Unsupervised image segmentation for microarray spots with irregular contours and inner holes," *BMC Bioinformatics*, vol. 16, no. 1, pp. 1–12, 2015.

[23] E. Aminian, S. Yazdani, and H. Saberkari, "Automated cdna microarray segmentation using independent component analysis algorithm," *Applied Medical Informatics*, vol. 37, no. 3, pp. 21–30, 2015.

[24] J. Harikiran, P. V. Lakshmi, and R. K. Kumar, "Multiple Feature Fuzzy c-means Clustering Algorithm for Segmentation of Microarray Images," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 5, no. 5, 2015.

[25] H.-N. Nguyen, C. Kervrann, C. Cauchois, and V. Paveau, "Automatic core segmentation and registration for fast tissue microarray de-arraying," in *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*, April 2015, pp. 1439–1442.

[26] Y. Feng, K. Song, and J. Liu, "A microarray image gridding method based on projection transformation and power spectral analysis," in *2015 International Symposium on Computers & Informatics*. Atlantis Press, 2015.

[27] K. M. Karun, V. Binu, K. M. Karun, K. Prasad, N. Nair, K. Manjunatha, and K. Prasad, "Review on image segmentation methods in cdna microarray experiments and a novel algorithm for segmentation," *International Journal of Emerging Science and Engineering (IJESE)*, vol. 3, no. 5, pp. 23–24, 2015.

[28] G. Shao, S. Wu, and T. Li, "cDNA microarray image segmentation with an improved moving k-means clustering method," in *IEEE International Conference on Semantic Computing (ICSC)*, Feb. 2015, pp. 306–311.

[29] M. Hernández-Cabronero, I. Blanes, A. J. Pinho, M. W. Marcellin, and J. Serra-Sagrista, "Analysis-Driven Lossy Compression of DNA Microarray Images," *IEEE Trans. Med. Imag.*, vol. 35, no. 2, pp. 654–664, Feb. 2016.

[30] R. Jornsten, W. Wang, B. Yu, and K. Ramchandran, "Microarray image compression: SLOCO and the effect of information loss," *Signal Processing*, vol. 83, no. 4, pp. 859–869, Apr. 2003.

[31] J. Hua, Z. Liu, Z. Xiong, Q. Wu, and K. Castleman, "Microarray BASICA: Background adjustment, segmentation, image compression and analysis of microarray images," *EURASIP Journal on Applied Signal Processing*, vol. 2004, no. 1, pp. 92–107, Jan. 2004.

[32] Q. Xu, J. Hua, Z. Xiong, M. L. Bittner, and E. R. Dougherty, "The effect of microarray image compression on expression-based classification," *Signal Image and Video Processing*, vol. 3, no. 1, pp. 53–61, Feb. 2009.

[33] N. Faramarzpour and S. Shirani, "Lossless and lossy compression of DNA microarray images," in *Proceedings of the IEEE International Data Compression Conference, DCC*, 2004, pp. 538–538.

[34] S. Lonardi and Y. Luo, "Gridding and compression of microarray images," in *Proceedings of the IEEE Computational Systems Bioinformatics Conference*, 2004, pp. 122–130.

[35] T. J. Peters, R. Smolikova-Wachowiak, and M. P. Wachowiak, "Microarray image compression using a variation of singular value decomposition," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 1-16, 2007, pp. 1176–1179.

[36] M. R. N. Avanaki, A. Aber, and R. Ebrahimpour, "Compression of cDNA microarray images based on pure-fractal and wavelet-fractal techniques," *ICGST International Journal on Graphics, Vision and Image Processing, GVIP*, vol. 11, pp. 43–52, March 2011.

[37] A. J. R. Neves and A. J. Pinho, "Lossless Compression of Microarray Images Using Image-Dependent Finite-Context Models," *IEEE Trans. Med. Imag.*, vol. 28, no. 2, pp. 194–201, Feb. 2009.

[38] Molecular Devices, "GenePix Pro [Online]. Available http://moleculardevices.com/."