Lossy-to-lossless compression of microarray images using expectation pixel values

Luís M. O. Matos
luismatos@ua.pt
António J. R. Neves
an@ua.pt
Armando J. Pinho
ap@ua.pt

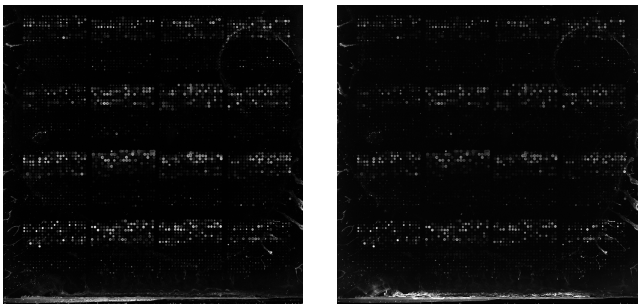
Signal Processing Laboratory
IEETA/DETI
University of Aveiro,
3810-193 Aveiro, Portugal

Abstract

This manuscript presents a lossless bitplane-based method for compression of microarray images based on expectation pixel values. The method compress each image on a bitplane basis, from the most to the least significant bitplanes, based on arithmetic coding driven by a 3D finite-context models and Expectation-based Bitplane Coding (EBC). It produces an embedded bitstream that allows progressive decoding of the original image. The EBC is a technique where we can select future pixels to create the context template. EBC alone is not effective to compress microarray images, therefore, we decided to use an hybrid approach using EBC , intra-plane and inter-plane to create a more suitable context template. EBC is more efficient for compressing the least significant bitplanes where the expected values are close to the pixels true values. The results obtained with the proposed method are slightly better in some cases however, generally, they are very similar.

1 Introduction

The raw data resulting from a microarray experiment consist of a pair of 16 bits per pixels of grayscale images (see Figure 1). These images, depend on the size of the array and the resolution of the scanner, and may require tens of megabytes to be stored or transmitted without any compression loss. Due to this fact, and the need for long-term storage and efficient transmission, the development of lossless compression methods with progressive decoding capabilities is an important challenge. In the literature, we can find several proposals for lossless compression of microarray images [2, 3, 5, 6, 7]. The method proposed by Neves *et al*. [6], which uses a 3D finite-context model and arithmetic coding, is the one of the most promising approaches. The main idea of the proposed method is to use EBC instead of intra-plane coding, in each bitplane where EBC provides best compression results compared to intra-plane coding.



(a) Green channel (b) Red channel

Figure 1: Example of a pair of images (1041 × 1044 pixels) that results from a microarray experiment.

2 Expectation-based Bitplane Coding (EBC)

Kikuchi *et al*. [4] proposed recently the idea of EBC. This approach can be useful in some bitplanes, where the neighbors bits are not so meaningful as they seem. Sometimes, the neighboring bits are only refining bits that fine-tune the value of that particular pixel. In 2010, Chen *et al*. [1] proposed a lossless compression algorithm that uses EBC. Our proposal uses a similar approach that can be applied to microarray images. Suppose $p = \{p_{15}, p_{14}, \ldots, p_2, p_1, p_0\}$, where p_i is the value of the pixel p_i at bitplane i. When we are encoding the n$^{\text{th}}$ bitplane, if the pixel p is already encoded at the current bitplane, its expectation value is formulated as

$$E(x) = \sum_{i=n}^{15} 2^i + (2^{n-1} - 1). \quad (1)$$

On the other hand, if the pixel p has not been encoded at current bitplane, its expectation value is defined as:

$$E(x) = \sum_{i=n}^{15} 2^i + (2^n - 1). \quad (2)$$

During the compression process, we use this expectation values instead of the real bit values only when the EBC is more efficient to compress the pixels of the current bitplane. In case of using the EBC, if the expected value of the context is lower than the expected value of the current pixel, the context bit used is 0, otherwise 1 is used. The variable size template used is described in Figure 2 and as we can see, there are two future pixels (pixels 4 and 3) that are used in the context template. The pixel denoted by 0 represents the current pixel being compressed. Using this approach, we can select future pixels for the context template, which is more efficient for the least significant bitplanes, where the neighboring bits of each bitplane are merely refining bits and they only fine-tune the final value of that particular pixel.

		9		
	5	2	6	8
7	1	0	3	
		4		

Figure 2: Context template[4].

3 Algorithm

The compression algorithm is based on simple bitplane separation, where each binary layer (from the least to the most significant one) is then compressed using 3D finite-context models and arithmetic coding. In Figure 3, we can see an example of a 3D context configuration used to compress a microarray image. As we can see, there is inter-plane and intra-plane pixels that can be used in the context template. However, in our approach, we decided to keep the inter-plane configuration that is used in [6] and try to use intra-plane and EBC competing with each other. In each bitplane, we use inter-plane + EBC or inter-plane + intra-plane, depending which one saves more bits. The selected approach must be sent as side information to the decoder in order to be possible to decode the original image correctly. This only requires 16 extra bits, which is a small cost to pay compared to the image compression data.

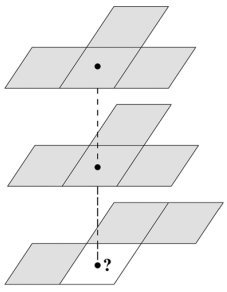


Figure 3: 3D context configuration [6].

4 Results and conclusions

The microarray images used in the experimental results were collected from three publicly available sources. 1) 32 images that we refer to as the APO_AI set; 2) 14 images forming the ISREC set; 3) three images previously used in MicroZip. The image sizes ranges from 1000×1000 to 5496×1956 pixels and all of them have 16 bits per pixel. The average results presented take into account the different sizes of the images, i.e., they correspond to the total number of bits divided by the total number of image pixels. In Table 1, we have the compression results for the algorithm proposed in [6] and our approach. As we can see, there are small improvements in some images but, generally, the results are very similar. Furthermore, we noticed that, in the last 8 least significant bitplanes, our approach uses more times EBC instead of intra-coding, which leads to the conclusion that EBC is more effective to compress the least significant bitplanes than intra-coding (see Fig. 4). It is possible to conclude that in the least significant bitplanes, the algorithm actually generates better results. As future work, it will be interesting to explore other alternatives to EBC or using a different equation to obtain the expected value.

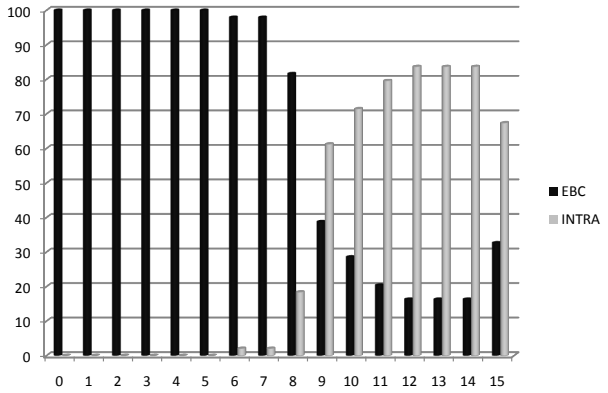


Figure 4: Percentage of utilization of each context (INTRA vs EBC) for each plane, from the least significant bitplane (0) to the most significant one (15).

Set	Image	Micro3DEncBest	Proposed
APO_AI	1230c1R	10.591	10.589
	1230c1G	10.847	10.847
	1230c2R	10.734	10.734
	1230c2G	10.971	10.971
	1230c3R	9.868	9.870
	1230c3G	10.244	10.244
	1230c4R	10.271	10.272
	1230c4G	10.430	10.430
	1230c5R	9.682	9.682
	1230c5G	9.970	9.970
	1230c6R	10.413	10.413
	1230c6G	10.265	10.265
	1230c7R	9.702	9.702
	1230c7G	10.164	10.164
	1230c8R	10.485	10.485
	1230c8G	10.389	10.389
	1230ko1R	10.348	10.348
	1230ko1G	10.076	10.079
	1230ko2R	10.093	10.093
	1230ko2G	10.015	10.015
	1230ko3R	10.138	10.138
	1230ko3G	10.367	10.367
	1230ko4R	10.130	10.130
	1230ko4G	10.032	10.032
	1230ko5R	10.093	10.093
	1230ko5G	10.390	10.390
	1230ko6R	9.978	9.978
	1230ko6G	10.194	10.194
	1230ko7R	9.984	9.984
	1230ko7G	10.255	10.255
	1230ko8R	10.191	10.191
	1230ko8G	10.232	10.232
	Average	**10.236**	**10.236**
ISREC	Def661Cy3	10.412	10.412
	Def661Cy5	8.874	8.874
	Def662Cy3	9.156	9.155
	Def662Cy5	10.581	10.592
	Def663Cy3	10.125	10.125
	Def663Cy5	9.538	9.547
	Def664Cy3	9.991	9.992
	Def664Cy5	11.192	11.192
	Def665Cy3	11.429	11.429
	Def665Cy5	12.536	12.536
	Def666Cy3	9.346	9.319
	Def666Cy5	11.062	11.069
	Def667Cy3	9.218	9.225
	Def667Cy5	9.328	9.328
	Average	**10.199**	**10.200**
Microzip	array1	11.126	11.142
	array2	8.470	8.470
	array3	7.712	7.712
	Average	**8.014**	**8.016**
AVERAGE		**11.9327**	**11.9339**

Table 1: Compression results in bits per pixel for the proposed method and the method presented in [6]. The 16 extra bits required as side information are included.

References

[1] Minjie Chen, P. Franti, and Mantao Xu. Lossless bit-plane compression of images with context tree modeling. In *International Conference on Green Circuits and Systems (ICGCS), 2010*, pages 605–610, June 2010. doi: 10.1109/ICGCS.2010.5542992.

[2] N. Faramarzpour and S. Shirani. Lossless and lossy compression of DNA microarray images. In *Data Compression Conference (DCC 2004), 23-25 March*, page 538, Snowbird, UT, USA, March 2004.

[3] N. Faramarzpour, S. Shirani, and J. Bondy. Lossless DNA microarray image compression. In *Proc. of the 37th Asilomar Conf. on Signals, Systems, and Computers, 2003*, volume 2, pages 1501–1504, November 2003.

[4] H. Kikuchi, K. Funahashi, and S. Muramatsu. Simple bit-plane coding for lossless image compression and extended functionalities. In *Picture Coding Symposium, 2009. PCS 2009*, pages 1–4, May 2009. doi: 10.1109/PCS.2009.5167351.

[5] S. Lonardi and Y. Luo. Gridding and compression of microarray images. In *Proc. of the IEEE Computational Systems Bioinformatics Conference, CSB-2004*, Stanford, CA, August 2004.

[6] A.J.R. Neves and A.J. Pinho. Lossless Compression of Microarray Images Using Image-Dependent Finite-Context Models. *IEEE Transactions on Medical Imaging*, 28(2):194 –201, Feb 2009. ISSN 0278-0062. doi: 10.1109/TMI.2008.929095.

[7] Y. Zhang, R. Parthe, and D. Adjeroh. Lossless compression of DNA microarray images. In *Proc. of the IEEE Computational Systems Bioinformatics Conference, CSB-2005*, Stanford, CA, August 2005.