Cellular Automata Technology in Scalable color image coding

[1]Lilly Raffy Cheerotha, [2] Ameefa P K,
[1] [2]Assistant Professors, Department Of Computer Science And Engineering, IES College of Engineering, Chittilapilly, Trissur

Abstract-*A scalable color image coding algorithm is a multi resolution representation of the data. It can be often obtained using a linear filter bank. Reversible cellular automata have been proposed recently as simpler, nonlinear filter banks that produce a similar representation. The original image is decomposed into four sub bands, such that one of them retains most of the features of the original image at a reduced scale. The project discusses the utilization of reversible cellular automata and arithmetic coding for scalable compression of color images. In the binary case, the proposed algorithm that uses simple local rules compares well with the JBIG compression standard, in particular for images where the foreground is made of a simple connected region. For complex images, more efficient local rules based upon the lifting principle have been designed. They provide compression performances very close to or even better than JBIG, depending upon the image characteristics. In the gray scale case, and in particular for smooth images such as depth maps, the proposed algorithm outperforms both the JBIG standards under most coding conditions. In color images after sampling equally optimal transform per component could be computed. Cellular automata transform is a new scheme to enhance resolution in terms of compression ratio.*

Index Terms – *Arithmetic coding, cellular automata (CA), scalable image coding, RCA*

I. INTRODUCTION

Focusing on the compression of bi-level images, a first scalable solution for binary images, i.e., images made of black or white pixels, was proposed within the JBIG coding standard [1]. In this framework, several versions of the same input image, at different spatial resolutions, are formed and encoded. Unfortunately, this paradigm requires coding a whole image for each resolution layer introducing a significant information redundancy in the coded bit stream. Therefore, more efficient schemes for scalable coding of binary images have been proposed in literature [2]. Recent works on this subject (see [3] and [4]) have been focusing on obtaining a high compression ratio too, but at the expense of scalability. As for multilevel images, these coding strategies proved to be ineffective and, therefore, most of the successive algorithms adopted a wavelet-based decomposition of the original signal, followed by an accurate reordering and modelization of the data to be coded. As a result of this research work, image coding experts finalized the JPEG2000. More recently, a novel binary transform, based upon cellular automata (CA) theory, has permitted the design of effective scalable coders for binary images that inherit many properties of the wavelet-based image coders
[6]. This paper presents a scalable lossless image coding algorithm based upon reversible cellular automata (RCA). In practice, appropriate reversible rules are used to transform the input image into four subimages with a lower resolution. Each of these is then converted into a bit stream using a context-based adaptive arithmetic coder whose contexts are computed from the values of (already-coded) neighboring pixels, in the same (intraimage) or in the others (interimage) subimages. The RCA approach is applied to binary images, grayscale images and color images.

The rest of the paper explains about related work in section II,proposed work in section III and in Section IV implementation details are explained.

II. RELATED WORK

ITU-T (1993) Recommendation T.82. suggested[4] the Joint Bi-level Image experts Group (JBIG) . The JBIG experts group[4] was formed in 1988 to establish a Standard for the progressive encoding of bi-level images. A progressive encoding System transmits a compressed image by first sending the compressed data for a reduced resolution version of the image and then enhancing it as needed by transmitting additional compressed data, which builds on that already transmitted. This Recommendation I International Standard defines a coding method having progressive, progressive-compatibles sequential, and Single-Progressions sequential modes and suggests method to obtain any needed low-resolution renditions. It has been found possible to effectively use[4] the defined coding and resolution-reduction algorithms for the lossless coding of grayscale and colour image a s well as bi-level images. This Specification defines a method for lossless compression encoding of a bi-level image (that is, an image that, like a black-and-white image, has only two colors). The defined method tan also be used for coding grayscale and color images. being adaptive o image characteristics, it is robust over image type. On scanned matches printed characters, observed compression ratios have been from 1,l to 1,5 times as great as those achieved by the MMR encoding algorithm (which is less complex) described in Recommendations T.4 (G3) and T.6 (G4). On Computer generated images of printed characters, observed compression ratios have been as much as 5 times as great. On images with greyscale rendered by half toning or dithering, observed compression ratios have been from 2 to 30 times as great. The method is bit-preserving high meanest hat it, like Recommendations .4 and T.6, is distortion less and that the final decoded image is identical to the original. The method also has “progressive” capability. When decoding a progressively coded image, a low-resolution rendition of the original image is made available first with subsequent doublings of resolution as more data is decoded. Arithmetic coding is superior in most respects to the better-known Huffmanmethod. It represents information] at least as compactly-sometimes considerably more so. Its performance is optimal without the need for blocking of input data. It accommodates adaptive models easily and is computationally efficient. Yet many authors and practitioners seem unaware of the technique. Indeed there is a widespread belief that Huffman coding cannot be improved upon. We aimto rectify this situation by presenting an accessible implementation of arithmetic coding and by detailing its performance characteristics. We start by briefly reviewing basic concepts of data compression and introducing the model-based approach that underlies most modern techniques. We then outline the idea of arithmetic coding using a simple example, before presenting programs for both encoding and decoding. In these programs the model occupies a separate module so that different models can easily be used. Next we discuss the construction of fixed and adaptive models and detail the compression efficiency and execution time of the programs, including the effect of different arithmetic word lengths on compression efficiency. Finally, we outline a few applications where arithmetic coding is appropriate.

III. PROPOSED WORK

Cellular Automata are mathematical idealizations of physical systems in which space and time are discrete, and physical quantities take on a finite set of discrete values.A cellular automata consists of a regular uniform lattice with discrete variables at each site(“cell”).The state of a cellular automata completely specified by the values of the variables at each site. A CA evolves in the discrete time steps,with the value of the variable at one site be affected by the values of the variables in its neighbourhood on the previous step.

The term cellular automata is plural. To simplify our lives, we’ll also refer to cellular automata as “CA.” A cellular automaton is a model of a system of “cell” objects with the following characteristics. The cells live on a grid. Each cell has a state. The number of state possibilities is typically finite. The simplest example has the two possibilities of 1 and 0 (otherwise referred to as “on” and “off” or “alive” and “dead”). Each cell has a neighbourhood. This can be defined in any number of ways, but it is typically a list of adjacent cells.

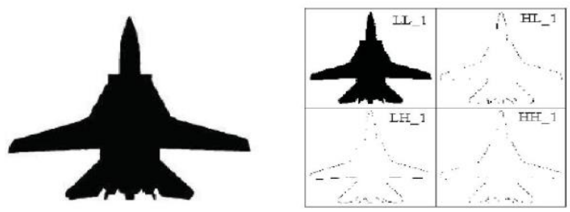
In proposed system the existing work can be extended by introducing encoding to colour images. A scalable colour image coding algorithm is a multi resolution representation of the data. It can be often obtained using a linear filter bank. Reversible cellular automata have been proposed recently as simpler, nonlinear filter banks that produce a similar representation. In colour images after sampling equally optimal transform per component could be computed. Cellular automata transform is a new scheme to enhance resolution in terms of compression ratio.

IV. IMPLEMENTATION WORK

A.SUB BAND CODING WITH CELLULAR AUTOMATA

Cellular Automata (CA) are dynamical systems and models of massively parallel computation that share many properties of the physical world . A cellular automaton consists of an infinite lattice of identical cells arranged regularly, with a natural notion of neighbourhood. Each cell is provided with a state from a finite number of possible states. The CA evolves to a new global state, or configuration, by updating the states of the cells synchronously, in discrete time steps, according to a local update rule, which takes into account the current state of each cell and its neighbours. More precisely, let us consider a d-dimensional CA, with a finite state set Q. The cells are positioned at the integer lattice points of the d-dimensional Euclidean space, indexed by Zd. The configuration of the system at any given time is a function c : Zd Q that provides the states of all cells. Let Cd Q denote the set of all d-dimensional configurations over the state set Q. The neighbourhood vector N = (¯v1, ¯v2, . . . , ¯vn) of the CA specifies the relative locations of the neighbours of the cells: each cell ¯x Zd has neighbours, in positions ¯x + ¯vi for i = 1, 2, . . . , n. The local rule f : Qn Q determines the global dynamics F : Cd Q Cd as stated in (1),Q as follows: for every c E Cd Q and ¯x E Zd we have

F(c)(¯x) = f[c(¯x + ¯v1), c(¯x + ¯v2), . . . , c(¯x + ¯vn)] (1)



(a) original image (b) one level

B. REVERSIBLE RM-BAND CA TRANSFORMATION

Let us consider binary images as the initial configuration of a 2-dimensional 4-bandCA with 16 states, where cells are 2 × 2 blocks with state q = (qll, qhl, qlh, qhh) Q, and qll, qhl, qlh, qhh E {0, 1} are the pixels (or components) inside the cell corresponding to sub bands LL, HL, LH and HH respectively. An elementary rule changes components values of only one subband according to some permutations defined over its domain. The permutation actually used is determined by looking at the other m – 1 subbands of neighbouring cells. This operation is invertible by the inverse permutation since the processed components are not affecting each other and the other subbands remain unchanged. The rule used for pixels in the LL subband

(c) three levels (d) elementary rule(LL)

C. MULTILEVEL DIGITAL IMAGE COMPRESSION

When dealing with multi level images, we must be able to cope with a substantially more complex correlation structure in the image data.

Non-binary RCA filter banks could be used where the initial configuration directly stores the values of the image samples. This case allows for a “true” non-linear filtering, which can be hard to design but gives optimum performance.

Multi-level images can be seen as a collection of their bit-planes from B0 (less significant) to BP 1 (most significant), that can be thought as P bi-level images. Binary

RCA filter banks that store each Bi could be used whose local rules depend also on the actual values of some samples in the "upper" planes Bi+1 to BP 1. In this case a "sub-optimal" non-linear filtering could be achieved with relaxed design difficulties.

Independent binary RCA filter banks that store each Bi could be used and designed, with local rules transforming Bi into Bti both dependently or independently from the plane index i. The residual inter-plane correlation is then exploited in the CA-transformed domain.

Some form of inter-plane prediction could be used for prediction of bit-plane Bi (i < P 1) from planes Bi+1 to BP 1, such that only the prediction error Ei must be actually coded (providing that the upper planes are encoded first). Then, independent *binary* RCA filter banks that store each Ei could be used (with local rules transforming Ei into Et i both dependently or independently from i).

V. CONCLUSION

In proposed system the existing work is extended by introducing encoding to colour images. A scalable colour image coding algorithm is a multi resolution representation of the data. It can be often obtained using a linear filter bank. Reversible cellular automata have been proposed recently as simpler, nonlinear filter banks that produce a similar representation. In colour images after sampling equally optimal transform per component could be computed. Cellular automata transform is a new scheme to enhance resolution in terms of compression ratio.These are number of benefits in colour image coding like less storage space and for transmission of image**.**

VI. REFERENCE

[1] Cappellari.L, Cruz-Reyes.C, Calvagno.G and Kari.J "Lossy to lossless spatially scalable depth map coding with cellular automata," in *Proc. IEEE Data Compress. Conf.*, Snowbird, UT, Mar. 16–18, 2009, pp. 332–341.

[2] Cappellari.L, Cruz-Reyes.C, Calvagno.G and Kari.J "Lossy to lossless spatially scalable depth map coding with cellular automata," in *Proc. IEEE Data Compress. Conf.*, Snowbird, UT, Mar. 16–18, 2009, pp. 332–341.

[3] Cappellari.L, Calvagno.G, "Lifting-based design of reversible cellular

[4] automata for scalable coding of binary images," in *Proc. IEEE Int. Conf. Image Process.*, Cairo, Egypt, Nov. 7–11, 2009, pp. 1901–1904.

[5] *JBIG Bi-Level Image Compression Standard*, ISO/IEC 11544, ITU-T and ISO/IECJTC1/SC29/WG1, 1993, ITU-T Recommendation T.82.

[6] Kari.J, "Theory of cellular automata: A survey," *Theor. Comput. Sci.*, vol. 334, no. 1–3, pp. 3–33, 2005. Kari.J "Reversible cellular automata," in *Proc. DLT*, 2005, pp. 57–68.

[7] Milani.S, Cruz-Reyes.C, Kari.J. and Calvagno.G, "A binary image scalable coder based on reversible cellular automata transform and arithmetic coding," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 16–18, 2009, p. 460.