World Academy of Science, Engineering and Technology
International Journal of Aerospace and Mechanical Engineering
Vol:1, No:7, 2007

Cellular Automata Based Robust Watermarking Architecture towards the VLSI Realization

V. H. Mankar, T. S. Das, and S. K. Sarkar

***Abstract*—**In this paper, we have proposed a novel blind watermarking architecture towards its hardware implementation in VLSI. In order to facilitate this hardware realization, cellular automata (CA) concept is introduced. The CA has been already accepted as an attractive structure for VLSI implementation because of its modularity, parallelism, high performance and reliability. The hardware realizable multiresolution spread spectrum watermarking techniques are very few in numbers in spite of their best ever resiliency against signal impairments. This is because of the computational cost and complexity associated with their different filter banks and lifting techniques. The concept of cellular automata theory in order to form a new transform domain technique i.e. Cellular Automata Transform (CAT) have been incorporated. Since CA provides spreading sequences having very low cross-correlation properties, the CA based pseudorandom sequence generator is considered in the present work. Considering the watermarking technique as a digital communication process, an error control coding (ECC) must be incorporated in the data hiding schemes. Besides the hardware implementation of entire CA based data hiding technique, the individual blocks of the algorithm using CA provide the best result than that of some other methods irrespective of the hardware and software technique. The Cellular Automata Transform, CA based PN sequence generator, and CA ECC are the requisite blocks that are developed not only to meet the reliable hardware requirements but also for the basic spread spectrum watermarking features. The proposed algorithm shows statistical invisibility and resiliency against various common signal-processing operations. This algorithmic design utilizes the existing allocated bandwidth in the data transmission channel in a more efficient manner.

***Keywords*—**Cellular Automata, Watermarking, Error Control Coding, PN sequence, VLSI.

I. Introduction

OVER the past few years, there has been tremendous proliferation in the World Wide Web together with availability of relatively inexpensive digital recording and storage devices. This phenomenon has given rise to the way of easier replication and distribution of multimedia digital content without any loss in quality. Therefore, digital watermarking is intended by its developers as the solution to the need to provide value added fortification on top of data encryption and scrambling for content protection. Digital watermarking algorithms can be thought as digital communication scheme where an auxiliary message is embedded in digital multimedia signals and are available wherever the latter signals move. The decoded message latter on serves the purpose of security in communication, copyright protection, copy control, data authentication, broadcast monitoring, digital signatures, fingerprinting etc. Robustness is an essential criterion in digital multimedia watermarking schemes along with visual transparency, high data embedding rate, low computation cost and complexity of the algorithms needed for data embedding and recovery purpose. All these requirements are related in conflicting manner and the particular algorithmic development emphasizes to a greater extent on one or more such requirements depending on the type of application [1-5].

Various information-embedding algorithms have been proposed in this still emerging field. There are basically two major methods that are used to embed watermarks into multimedia signals. One is directly realized in spatial domain, which has a high capacity but poor robustness and can be easily broken. The other is achieved in the transform domain, such as DFT, DCT, DWT and so on with comparatively higher resiliency against volumetric distortions. Several modulation schemes are being used in either domain such as LSB, SS, HVS etc. with their different degree of performance [1-5]. Recently spread spectrum (SS) modulation based systems have received considerable attention because of its efficient trade-offs among the data hiding requirements compared to the rest. On the other hand, DWT as unitary transform becomes a useful tool for the same due to its better signal decomposition capability [3]. This watermarking process can be a part of a scanner, a digital camera, or any other multimedia device so that the digitized images are watermarked right at the origin. The hardware implemented watermarking schemes has advantages over the software implementation in terms of low power, high performance, and reliability. [6]. The CA has been accepted as an attractive hardware structure for VLSI implementation because of the following characteristics.

- Simple and identical processors in the network (*modularity*).
- Time synchronous processing (*parallelism*).
- Fixed distance or neighbourhood communication.

A comparative view of the hardware implemented watermarking techniques is provided in the current literature

Authors are with Department Of Electronics & Telecommunication Engineering, Jadavpur University, Kolkata-700032, India (e-mails: vijaymankar@yahoo.com, tirthasankardas@yahoo.com, sksarkar@etce.jdvu.ac.in).

[6]. The hardware implemented multiresolution watermarking techniques are very few in numbers in spite of their best ever resiliency against signal impairments. This is because of the computational cost and complexity associated with their different filter banks and lifting techniques. Hence, the concept of cellular automata theory is incorporated in forming a new transform technique having multi-spectrum planes similar to the space-frequency tiling of the DWT i.e. Cellular Automata Transform (CAT) [17-18]. CA provides spreading sequences having very low zero lag cross-correlation properties. Therefore, CA based SS watermarking scheme is more practically feasible in hardware. Considering the watermarking technique as a digital communication process, an error control coding (ECC) must be incorporated in the data hiding schemes. The CA can also help a lot in this regard. Besides the hardware implementation of entire CA based data hiding technique, the individual blocks of the algorithm using CA provide the best result than that of some other methods irrespective of the hardware and software techniques.

On the basis of above analysis and discussion, we have proposed a CA based spread spectrum (SS) watermarking algorithm towards its realization in hardware. The present work initiates the implementation framework for the watermarking using CA that will facilitate the hardware realization. The paper is organized as follows. The section II provides an overview of cellular automata. The CA architecture and SS watermarking principle is given in section III. The proposed watermarking algorithm is explained in section IV. The different components required to realize the proposed work using CA are given in subsequent sections such as Cellular Automata Transform (CAT) in section V, the CA based PN sequence generator in section VI and CA based error correcting codes in section VII. The results and discussion are given in section VIII and it is concluded in section IX.

II. Cellular Automata (CA)

J. Von Neumann showed that a CA can be universal. However, due to the complexity associated with Von Neumann's rules, they were never implemented on a computer. The researchers have tried to develop simpler and more practical architectures of CA to use it in various diversified application areas. [8]. Stephen Wolfram has studied a family of simple one-dimensional CA (now famous Wolfram rules) and proposed that even these simplest rules are capable of emulating complex behavior. The reasons behind the popularity of CA are due to their simplicity, and enormous potential in modeling complex systems [8].

A cellular automata (CA) is an array of sites (cells), which evolves, in discrete steps. At discrete time steps, all cells simultaneously update their states depending on their current state and those of their immediate neighbours. The neighbourhood can vary depending upon the dimensionality of the CA. The CA register may possess null boundary conditions (i.e., the first and last cells assumes neighbour cell to have a zero value) or be cyclically connected (i.e., the CA forms a ring connecting the first and last cells). For a binary CA, each cell determines its next value on the basis of the eight possible combinations of the present values of itself, and its left and right neighbours (i.e., 000, 001, 010, etc.). The next-state values corresponding to each possible input form a number, which is referred to as the "rule number" under the classification scheme of Wolfram [8-14].

The CA, characterized by a rule known as rule 90, specifies an evolution from neighbourhood configuration to the next state as:

Neighbourhood:	111	110	101	100	011	010	001	000
Next state:	0	1	0	1	1	0	1	0

Decimal 90

Hence, the corresponding combinational logic of rule 90 is

$$x_i(t+1) = x_{i+1}(t) \bullet \bar{x}_{i-1}(t) + \bar{x}_{i+1}(t) \bullet x_{i-1}(t)$$
$$= x_{i-1}(t) \oplus x_{i+1}(t),$$

where $\oplus$ and $\bullet$ are the operations XOR and AND, respectively, Thus for rule 90, the next state of i^{th} cell depends on the present states of its left and right neighbours. Similarly, the combinational logic for rule 150 is given by $x_i(t+1) = x_{i-1}(t) \oplus x_i(t) \oplus x_{i+1}(t)$ that is, the next state of i^{th} cell depends on the present states of its left and right neighbours and on its own present state. Often, the evolution of a CA is shown using a state time diagram. The time axis runs vertically, thereby, showing successive values in the CA. There are in general at least two distinct methods of initialising a CA. One method is to begin with a simple state such as a nonzero value at a single central site; the other method is to begin with each site randomly initialised to 0 or 1 with $p(0) = p(1) = 0.5$. While the description of one-dimensional CA is very simple, the different CA rules produce a very wide range of global behaviour.

Wolfram has formulated four basic classes of behaviour for one-dimensional CA. Class 1 automata evolves to homogeneous final global states, class 2 evolves to periodic structures, class 3 exhibits chaotic behaviour, and class 4 yield complicated localized and propagating structures. Wolfram considers class 3 CA to be an abstract model of randomness in nature, and therefore, suitable for pseudorandom number generation [8]. This is because the cumulative effect of many iterations in class 3 CA is equivalent to performing very complicated transformations on the initial starting value. This evolution often becomes so complicated that its outcome can be found only by observation or simulation, i.e., there is no known closed form solution.

An additive CA is characterized by EXOR and/or EXNOR dependence. If in a CA the neighbourhood dependence is EXOR, then it is called a non-complemented CA and the corresponding rule is referred to as a non-complemented rule. For neighbourhood dependence of EXNOR, the CA is called a complemented CA. The corresponding rule involving the EXNOR function is called a complemented rule. In a complemented CA, single or multiple cells may employ a complemented rule with EXNOR function. If in a CA, the same rule applies to all cells, then the CA is called a uniform

CA; otherwise the CA is called a hybrid CA [9].

III. Architecture and SS Watermarking Principle

In general Fourier, Fourier-Mellin, DCT, Walsh/Hadamard, Spline etc. image transforms provide only one spectrum plane for embedding watermark data, so the hidden information can be easily removed. To increase the flexibility in data concealment, cellular automata transform (CAT) is being used in the proposed technique [17-18]. Therefore, it can provide many transform pattern verifying different CA bases thereby recovering the weak points having only one transform planes in DFT, DCT etc. transform domain methods. Moreover, it is possible to get many channels with various CA bases and rule numbers. Hence it is the complexity of CA that provides difficulty to attacker to find out the position of hidden information. In this way CAT helps to go for the best-suited spectrum planes for selective data embedding with greater data hiding capacity [18].

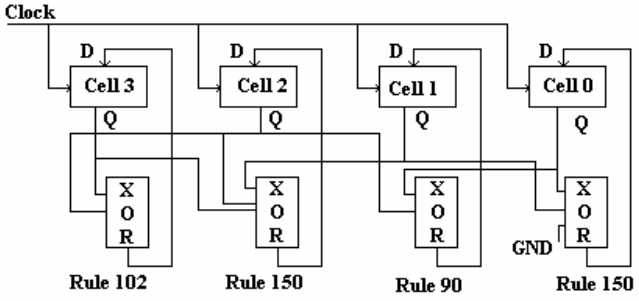

Fig. 1. A hybrid CA

Higher detection reliability is achieved if PN sequences possess very low zero lag cross-correlation among each other as well as with image block when image prediction is not used to evaluate the cross-correlation. The conventional maximum length sequence generated by LFSR does not provide a good detection. According to Mayer *et al* [5], sequence generation from Hadamard basis or Gram-Schmidt orthogonalization provides significant detection reliability but at the cost of poor spectrum spreading. The sequence generated from LFSR followed by Hadamard/Gram-Schmidt modulation gives better results with a comparable significant spectrum spreading. But genetic algorithm based 2D CA helps to obtain the ever-best PN sequences with higher quality of detection reliability [5].

Since the image restoration does not result in a perfect copy of the original cover image and the embedded signal is low power, the estimate of the embedded signal is poor. This results in a demodulated message signal that may have a substantial number of bit errors, indicated by a high-embedded signal BER. Therefore, to allow for the sub optimal performance of the signal estimation process, we have incorporated the use of error-control codes to correct the large number of bit errors. Any ECC that is capable of correcting the high signal estimation BER can be used within SS. As the proposed algorithm is intended for hardware realization, it suggests that all the steps of the process must be implemented by the additive CA theory only. This motivated us to incorporate the use of CA based ECC. Chaudhury *et al*. has given a CA based block code. Although the generated code words by this ECC are not optimal, it is employed in our algorithm because of its regular, modular and cascadable structure of the CA that can be easily implemented in VLSI technology [9-10].

On the basis of the above analysis on watermarking principle, we have proposed an entirely CA based SS watermarking architecture which can easily be implemented in VLSI.

IV. Proposed Watermarking Algorithm

A. Embedding

Let B denotes the binary valued watermark bit string as a sequence of N bit long information.

$$B = \{b_1, b_2, \dots, b_n\}, \qquad b_i \in \{1, 0\}$$

To project the host signal or cover image I into watermarking space ξ the image transformation χ is applied to the image *i.e.*

$\chi : (I_L) \rightarrow [C_L]$ where C is the projected image and L, L' are length of vector I and C, respectively.

For binary signaling, optimal modulation functions are antipodal signal pairs. For embedding of N bit watermark, a set PN_k of N two dimensional orthogonal sequence PN_i, $i = \{1, 2, \dots, N\}$ is used where k defines the secret key used as initializing seed to generate the set. These sequences/ code patterns can be considered as uniformly distributed random sets of independent random variables having a zero mean, unit variance bi-level distribution. Hence, in order not to introduce inter symbol interference (ISI) [7]

$$PN_i = \{(x,y),\ \forall PN_i\ (x,y) \neq 0\}$$
$$PN_i,\ PN_j = \phi\ \forall i \neq j$$

The watermark W is defined as the superposition of all modulated and weighted PN sequence or code patterns PN_i:

$$[W_{L'}] = \sum_{i=1}^{N} (b_i')\ \alpha [PNi_{L'}], \tag{1}$$

where α is the weighting factor or modulation index and b' represents the bit value mapped from {0,1} to {-1,1}. The watermarked or stego image is now given by adding watermark W to image representation in embedding space ξ and applying inverse transformation:

$$[I_W]_L = \chi^{-1}\ ([C_{L'}] + [W_{L'}] \tag{2}$$

B. Detection

The introduced watermarking scheme can be seen as a modulation system in which image acts as additive noise. Nevertheless, it is a common method in digital watermarking to use a linear correlator as detection statistics.

Let the watermarked/ stego image is projected into watermarking space by applying the image transform:

$$\chi : [\hat{I}_W]_L \rightarrow [\hat{C}_W] \tag{3}$$

Now the detection statistics or decision variable t_i is

obtained by evaluating the zero lag cross-covariance function between the signal features of projected stego image and each PN sequence/ code pattern PN_i

$$t_i = \left\langle PN_i - m_1(PN_i), \hat{C}_W - m_1(\hat{C}_W) \right\rangle (0) \tag{4}$$

where $m_1(X)$ represents the average of the sequence X. If X_k represents the elements of X with $k = 1, 2, \ldots, L'$

$m_1(X)$ can be mathematically expressed as follows:

$$m_1(s) = \frac{1}{L'} \sum_{k=1}^{L'} s_k \tag{5}$$

The symbol (0) in equation (4) indicates the zero lag cross-correlation and for two sequences S and R, the zero lag cross-correlation is given by

$$\langle S, R \rangle (0) = \frac{1}{L'} \sum_{k=1}^{L'} s_k r_k$$

where s_k and r_k are the elements of sequence S and R respectively with k=1,2,...,L'. The bit b_i' is detected as -1 if $t_i > 0$ and as 1 otherwise. Therefore, the computation of t_i becomes

$$\begin{aligned} &= \left\langle PN_j - m_1(PN_j), [C + W - m_1(C)] \right\rangle \\ &= \left\langle PN_j - m_1(PN_j), \left[C + \alpha . \sum_{i=1}^{N} b_i' PN_i \right] - m_1(C) \right\rangle \\ &= \left\langle PN_j - m_1(PN_j), C \right\rangle + \alpha . \sum_{i=1}^{N} b_i' \left\langle PN_i . PN_j \right\rangle - \left\langle PN_j, m_1(C) \right\rangle \\ &= \left\langle PN_j, \hat{C}_W \right\rangle \end{aligned} \tag{6}$$

The above analysis indicates that the code patterns used for spread spectrum watermarking should posses some specific properties. Watermark detection is improved if the following conditions are satisfied.

i. PN_i, $i = 1, 2, \ldots L'$ should be distinct sequences with zero average.
ii. The spatial correlation = $\langle PN_i, PN_j \rangle$ $i \neq j$ should be minimized. Ideally, sequences PN_i and PN_j should be orthogonal.
iii. Each PN_i for $i = 1, 2, \ldots, L'$ should be uncorrelated with image coefficient block C when image prediction (for estimating image distortion) is not used before evaluating the cross–correlation [7].

Since, code patterns are zero mean and non-overlapping orthogonal sequences, so above properties (i) and (ii) are satisfied.

V. Cellular Automata Transform (CAT)

We are given a physical process described by a set of discrete values f_i. This function is defined in a physical cellular space of lattice grid i. The function can be mathematically expressed in the following manner [18]:

$$f_i = \sum_j c_j B_{ij} \qquad \forall i \tag{7}$$

where B_{ij} are the basis functions and c_j are the associated transform coefficients defined in cellular automata frequency space j. The basis functions are related to the evolving field (i.e. the states) of the cellular automata (CA). Note that each point on the physical grid i has an associated basis function (spanning the entire physical space i and CA space, j). Equation (7) represents a mapping of process f (in the physical domain) into c (in the cellular automata domain) using the building blocks B as transfer functions. In many applications, we seek to obtain transform coefficients c with properties not necessarily possessed by the original function f. Alternatively, the transform process should reveal things about f not readily observed in the physical domain. This transformation is performed because the set c_j is more amenable to further processing to realize the specified objective.

One-dimensional GF (2) cellular spaces offer the simplest environment for generating CA transform with a smaller number of bases. It is possible to generate two-dimensional GF (2^p) CA bases from combinations of one-dimensional base elements [18]. The structure of an n-cell GF (2^p) is given in appendix A.2.

A. One Dimensional Bases

In a one-dimensional space consisting of N cells, the transform base is

$$B_j \equiv B_{ij} \text{ for } i, j = 0, 1, \ldots, (N - 1) \tag{8}$$

For the data sequence f_i ($i = 0, 1, \ldots, (N - 1)$), we have

$$f_i = \sum_{j=0}^{N-1} c_j B_{ij} \tag{9}$$

where $\{c_j\}$ are the transform coefficients.

There are a host of ways in which B_{ij} can be expressed as a function of $a \equiv a_{it}$, (i, $t = 0, 1, \ldots, (n - 1)$), a_{it} being the state of the i^{th} cell at the t^{th} instant of a n cell CA. The basis function therefore directly depends on the evolving field of the underlying cellular automata. The following basis functions can be used as transform bases:

$$B_{ij} = 2\, a_{ij} - 1 \tag{10}$$

$$B_{ij} = 2\, a_{ij}\, a_{ji} - 1 \tag{11}$$

$$B_{ij} = 2\, \rho_{ij}\, \rho_{ji} - 1$$

$$\rho_{ij} = 2\, a_{ij}\, a_{ji} - 1 + \rho_{ij-1} \tag{12}$$

$$\rho_{i0} = 2\, a_{ji} - 1$$

B. Two Dimensional Bases

In a 2D square space consisting of N x N cells, the transform base $B_j \equiv B_{ikjl}$, $i, k = 0, 1, 2, \ldots\ldots, (N - 1)$. For the data sequence f_{ik} ($i, k = 0, 1, \ldots, (N - 1)$), we have

$$f_{ik} = \sum_{j=0}^{N-1} \sum_{l=0}^{N-1} c_{jl} B_{ikjl} \qquad i, j = 0, 1, \ldots, (N-1) \tag{13}$$

in which c_{jl} are the transform coefficients.

The essence of cellular automata transform (CAT) is that we can always find out CA rules (and its associated neighbourhood, initial/boundary configuration, lattice arrangement, etc), which will result in basis functions and

transform coefficients with properties we desire for a given problem. The chief strength of CAT is the huge number and varied nature of the basis functions available to us.

VI. CA BASED PN SEQUENCE GENERATOR

The most popular hardware pseudorandom (PN) sequence generator is the linear feedback shift register (LFSR). The sequence has a cycle length of 2^{n-1} using an n-bit shift register, providing the polynomial describing the register is primitive over GF (2). There are, in general, three methods for generating PN sequences using LFSR viz. serial-in parallel-out LFSR, parallel LFSR waiting n clocks**,** the parallel LFSR with no wait cycles [11]. Moreover, a standard computer software based PN sequence generators are also available. On the other hand, the CA based PN sequence generator can also be realized using different CA rules. It is shown that these CA-based PN sequence generators may provide an alternative to conventional LFSR-based generators. In addition to improved randomness properties these new PN sequence generators also have implementation advantages that they can be designed to require only adjacent neighbour communication and they are cascadable, i.e., the physical length of the generator can be increased or decreased by simply adding or subtracting cells. This means that CA-based PN sequence generator is very appropriate for incorporation in a CAD tool. However, recent work on cascadable LFSR implementations also provides such possibilities for the LFSR [11].

The PN sequence generator associated with CA is more advantageous because of its reduced cross-correlation properties compared to that of LFSR. The CA approach merely requires changing the number of cells and a new starting value in order to implement variable length PN sequence. It is possible to generate CA based PN sequence having a statistical weighting to one region of the pattern space. The disadvantage of some CA based PN sequence generators using single rule (such as rule 90 or rule 150) is reduced cycle length. This drawback has been overcome by considering hybrid CA PN sequence generators (such as combination of rule 90 and rule 150).

The literature survey reveals that 1D CA based PN generators have been extensively studied in the past. This study convincingly shows that 1D CA is superior with respect to other widely used methods such as LFSR. Therefore, generators are essentially handcrafted by studying the structure of the bit pattern generated over time. Chowdhury *et al.* has given methodology for PN sequence by using 2D CA. They have concluded that 2D CA is better than that of 1D CA having same size (equal number of grid cells) in terms of the quality of the resulting PN sequences.

The cellular processing using evolutionary genetic algorithm to automatically generate 2D CA based PN sequence generators have been employed [15]. Marco Tomassini *et al.* have shown that applying extensive battery of statistical randomness tests such as Marsaglia's Diehard suite to their evolved CA; they rapidly produce high quality random number sequences. Moreover, they are able to handcraft even better PN sequence generators based on observations of the evolved CA, which in addition to high quality PN sequence generators also satisfy given hardware constraints. Non-uniform or inhomogeneous CA function is similar to uniform one; the only difference is that CA rules are not identical for all cells. The non-uniform CA exhibits the properties such as simplicity, parallelism and locality [15]. Hence, for the proposed work, non-uniform CA based 2D PN sequence generators specified by [15] briefly summarized as below are used.

The high quality CA based PN sequence generators have restricted the representation of the rule tables to allow for any 64 additive rules-those involving only XOR and XNOR logic. Since there are 64 possible (additive) rules, we need 6 bits to describe a rule.

Let $s_{i,j}(t)$ be the state of the cell at row i and column j, at time t. Its state at the next time step, $s_{i,j}(t+1)$ is then computed as follows:

$$s_{i,j}(t+1) = X \oplus (C \bullet s_{i,j}(t)) \oplus (N \bullet s_{i-1,j}(t)) \oplus (W \bullet s_{i,j-1}(t)) \oplus (S \bullet s_{i+1,j}(t)) \oplus (E \bullet s_{i,j+1}(t)) \quad (14)$$

where X, C (center), N (north), S (south), W (west), and E (east) are binary variables. C, N, S, W, and E denote whether the respective neighbouring cell state is taken into account (a value of 1) or not (a value of 0). The binary variable X demarcates linear (X = 0) from nonlinear (X = 1) additive rules. The genome of a cell is then given by the 6-bit string *XCNWSE*. For example, rule 15 (001111) represents the following function:

$$s_{i,j}(t+1) = s_{i-1,j}(t) \oplus s_{ij-1}(t) \oplus s_{i+1,j}(t) \oplus s_{i,j+1}(t)$$

In the present work, we have used CA PN sequence generators using rules 15, 31, 47 and 63. The procedure for generating CA PN sequences are given below:

i. The CA is run for four time steps, each cell thus producing a sequence of four bits (in time), which are treated collectively as a hexadecimal digit.
ii. These hexadecimal digits are then juxtaposed-cell after cell and line after line in a left-right, top-down manner to create a sequence of x x y random numbers, where x, y are the grid's dimensions.

The process is then iterated. For example, an 8 x 8 grid will produce 64 hexadecimal random digits every four-time steps [15].

VII. CA-BASED ERROR CORRECTING CODE (CAECC)

A. Basic Principle

The generation of a system of code words using the regular structure of CA is introduced here. The code words with any specified distance can be generated by suitable choice of the CA**.** The resulting block code is called as CA-based error correcting code (CAECC) [10].

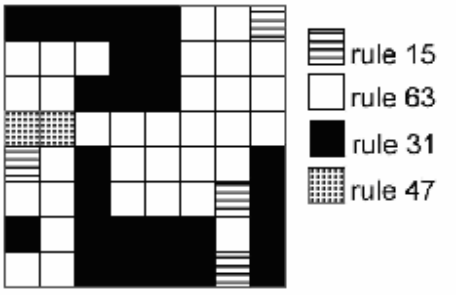

Fig. 2 Rule Map for 2-D PNsequence

Let I represent a k-bit information word $i_0, i_1 \ldots i_{k-1}$ to which $(n - k)$ check bits $c_0, c_1 \ldots c_{n-k-1}$ are added to form a (n, k, t) code. The CA is initially loaded with the incoming k information bits. It is then allowed to run for p cycles. The value of the integer p depends on the number of errors to be detected and corrected. Depending on the values of n and k, a function f_c is formed on the k-bit state of the CA to generate $(n - k)$ check bits. The function f_c compresses either single power T^p $[I]$ or multiple powers (T^{p1} $[I]$, T^{p2} $[I]$, . .), where T is the characteristic matrix of the CA**,** and p, $p1$, $p2$ are integers. Thus, codewords are given by [10]

$C = I, f_c\,(T^p\,[I])$: when one single power is taken or
$C = I.\ \mathrm{f_c}(T^{p1}\,[\,I\,]T^{p2}[\,I], \ldots)$; when multiple powers are taken
$= \{i_0, i_1 \ldots i_{k-1}, c_0, c_1 \ldots c_{n-k-1}\}$.

B. Generation of t-Distance Code

The problem of generating a t-distance (n, k, t) code is now considered for any arbitrary value of t. To arrive at the CAECC, first a k-bit CA is selected using the recursive algorithm. Now, the CA is required to run for p ($\geq t - 2$) cycles to generate CAECC. Instead of using a single p^{th} power of T, several powers of T are selected and concatenated. Thus, if I represents the information word, we attach T^{p1} (I), T^{p2} (I). . . as the check bits, for some integers $p1$, $p2$ etc., less than p. Subsequently, reduction in the number of check bits can be achieved using the same compression scheme. The sequential steps are as follows:

i. Load the k-bit CA with the k information bits. (Theorem 1 & 3)
ii. Run the CA for p cycles, where $p = t - 2$. (Theorem 2).
iii. Take one single or multiple power(s) of T and check whether the generator matrix satisfies the conditions noted in Theorem 1.
iv. Reduce the number of check bits (*compression*) to get a t distance code while satisfying the conditions noted in Theorem 1. (Theorem 4)

This strategy can be applied to any arbitrary values of k and t. By utilizing the above results, CA structures suitable for CAECC have been exhaustively identified for different values of k [10]. The referred theorems for CAECC are given in appendix A.1.

C. Decoding of CA Based Error Correcting Code

This section considers the syndrome decoding of CAECC in order to correct the erroneous bit positions. The scheme employs an inverse transformation built around the group properties of CA. The following steps are used to formulate the decoding schemes.

i. Compute an $(n - k)$-bit *syndrome S,* which is generated as

$$S = [H'][I'] \oplus [C] \quad or$$

$$[S] = [H]\begin{bmatrix} I' \\ C' \end{bmatrix}$$

where H is an $(n - k) \times n$) *parity matrix* formed by concatenating H' with an identity matrix I_{n-k} - k , i.e., $H = [[H'][I_{n-k}]]$. In this equation, an $(n - k) \times n$ matrix is multiplied with an $n \times 1$ matrix to get a $(n - k) \times 1$ matrix. Since $n > k$. there cannot be a unique reverse mapping

$$[S] = \begin{bmatrix} I_e \\ C_e \end{bmatrix}$$

Hence nonzero value of syndrome indicates the presence of error in the received word.

Solve a system of equations to determine the particular error vector that satisfies the syndrome equations [10].

D. Decoding Scheme (Direct Computation)

Since there are $n - k$ equations with n unknowns in the decoding of CAECC, there cannot exist a unique solution. The number of such n-bit vectors is evidently, $\leq \sum_{x=1}^{t'} \binom{x}{n}$ where t' is the number of errors to be corrected. So the search complexity is $O(n^{t'})$. It is essential to compute the error vector directly from the syndrome in order to reduce this complexity. The steps in the method are outlined below:

i. Augment k rows to the matrix $[H]_{(n-k) \times n}$ to make it an $n \times n$ nonsingular square matrix T_{avg}.

$$[T_{avg}][E] = \left[\frac{[H]}{[\text{AddedRows}]}\right]_{n \times n}\begin{bmatrix} I_e \\ C_e \end{bmatrix} = \begin{bmatrix} S \\ S_{avg} \end{bmatrix}$$

ii. For all permissible error vectors with t' or fewer 1's. Tabulate the relationship between S and S_{avg}.

$$[T_{avg}][E] = \left[\frac{[H]}{[AddedRows]}\right]_{n \times n}\begin{bmatrix} I_e \\ C_e \end{bmatrix} = \begin{bmatrix} S \\ S_{avg} \end{bmatrix}$$

iii. Synthesize an extended neighbourhood CA with $[T_{avg}]^{-1}$ as its characteristic matrix. (Lemma 1)

Generate the error vector directly by following the flow of control as per [10] for the given received information and check bits.

VIII. Results and Discussions

The above proposed CA based SS watermarking technique is tested over a large number of benchmark images viz. fishing boat, lena, bandon, peppers, cameraman etc. It is quite clear that imperceptibility, robustness efficiency and payload capacity all are suitably optimised along with the computational cost and complexity for this CA based watermarking algorithm. The experimental results on some of the images against various signal processing attacks such as linear and non-linear filtering, lossy compressions, image sharpening, histogram equalization, AWGN, geometric image impairments (Rotation, Scaling and Translation) etc are given in Tables I, II and III. The recoveries of embedded information against the effect of image degradation operations for fishing boat image are shown in the Fig. 3. Here the imperceptibility quality is measured in terms of Peak Signal to Noise Ratio (PSNR) and structural similarity index measurement (SSIM) methods whereas normalised cross-correlation (NCC) is used for resiliency property. The results imply the better subjective and objective recognition along with the good quality of watermark image for a given data-embedding payload. The inherent advantages of the proposed scheme may be summarised as below:

The greatest advantage of our projected scheme is the flexibility of adjusting the private key size (PN sequences) as well as encoded message length without any overhead. This is possible due to the regular, modular and cascadable structure of CA that can be easily implemented with modern VLSI technology.

Table I shows the results where in each of the image, watermark has been inserted according to the anticipated scheme. It gives better imperceptibility with higher data embedding capacity because of the multispectrum plane decomposition of CAT.

The robustness efficiency of the CA based data hiding technique has resulted in the superior quality watermarked image. Moreover, the CA based watermarking is tuned to counterfeit different volumetric distortions as a built-in function thereby effectively increasing the insertion/extraction speed of watermarking. Our comments are well substantiated with the depicted numerical data in the Tables II and III.

The CA based transforms (CAT) having multi frequency domains or multi channels appear with different quality of watermarked versions related to the flexible choice of rule numbers. Even this method is simple in respect to computational cost and complexity with superior reconstruction image quality at higher compression ratio. This is also reflected in the result tables.

2D CA can produce PN sequences at much higher rate without recourse to the time spacing parameter thereby facilitating the hardware implementation easier.

The simulation results obtained not only confirms the validation of the watermarking technique with benchmark quality but also leads the way of present prototype towards hardware domain since the individual blocks of the entire architecture is designed with the CA concept.

IX. Conclusion

In this paper we have applied the cellular automata concept to the design of SS watermarking architecture, which will initiate the way towards its hardware realization using VLSI technology. The security and resiliency of this CA based data-hiding scheme along with its execution speed are emphatically better. The stochastic cellular programming with its different CA rules helps a lot to handcraft different blocks of watermarking scheme thereby tailoring them to meet the hardware realization constraints in a more convenient way. This is the effective step where we have implemented the entire algorithm concatenating all CA based individual blocks using software means. This will open the avenue for the hardware realization of the proposed work in near future as CA bridges the gap between the software and hardware representation in a more efficient way.

TABLE I
Numerical Results of Robustness for different CA rules

Rule Number (CA bases)	PSNR (dB)	SSIM
15	40.0	0.9878
27	39.5	0.9869
43	41.2	0.9916
84	40.0	0.9875
112	38.4	0.9798
153	35.5	0.9583
171	41.2	0.9916
224	40.1	0.9889
245	42.9	0.9937

TABLE II
Numerical Results for different Attacks

Attack	Normalized Cross-Correlation (NCC)
LPF (5 times)	0.72
Median (5 times)	0.74
Gaussian (5 times)	0.77
Histogram Equalization	0.83
Edge Enhancement	0.91
Cropping (0:127, 0:127)	0.87
Invert	0.92
Range (150-100)	0.93
Scaling	0.64
Add Noise (10%:10%)	0.91
Translation	0.87
Rotation	0.47
Collusion (5 images)	0.55

TABLE III
Numerical Results against compression

Quality Factor (QF)	NCC JPEG	NCC JPEG2000
25	0.41	0.26
35	0.57	0.41
45	0.64	0.62
55	0.73	0.70
65	0.89	0.87
75	0.94	0.91

APPENDIX

A. Theorems and Corollary [10]

Theorem 1: A *k*-cell CA having characteristic matrix *T* generates a *t*-distance code if for some integer *p*, *Tp* satisfies the following conditions. For all integer *i*, $0 < i < t$, the bitwise sum of any *i* columns of *Tp* contains at least (*t* - *i*) 1's.

Corollary 1: A CA will generate a (2*k, k,* 4) code in *p* cycles if T^p satisfies the following three conditions:

a) every column of T^p contains at least three l's,

b) any two columns of *T''* differ in at least two positions,

c) the bitwise vector sum of any three columns of T^p is a nonzero vector.

Theorem 2: If a k-cell CA generates a (2*k*, *k*, *t*) code in *p* cycles, then $p \geq t$-2.

Theorem 3*:* For any given k-bit information *I*, (I, $T_k^2[I]$) will give rise to a (2 *k, k*, 4) code.

In general, adding *k* check bits to *k* information bits to obtain a distance-4 code is too expensive. A method of reducing the number of check bits generated without affecting the SEC-DED property of CAECC is given below.

Theorem 4: In a (2*k*, *k*, 4) code, we can replace check bits *Ci* and *Cj* by $Ci \oplus Cj$ provided Tk^2 satisfies the following two conditions:

1) the rows *i* and *j* do not have 1's in the same position, and

2) the reduced matrix obtained from T_k^2 by replacing rows *i* and *j* by their linear vector sum satisfies the conditions of Corollary 1.

B. General Structure of an n-cell GF (2^p) CA

The Fig. 1 depicts the general structure of an *n*-cell GF (2^p) CA. Each cell of such a CA having *p* number of memory elements can store an element {0, 1, 2, ..., $2^p - 1$} in GF(2^p) [16].

An *n* cell GF (2^p) CA can be characterized by the $n \times n$ characteristic matrix *T*, where

$$T_{ij} = \begin{cases} w_{ij}, \text{ if the next state of the } i^{th} \text{ cell depends on the present} \\ \quad \text{state of the } j^{th} \text{ cell by a weighted } w_{ij} \in GF(2^p) \\ 0, \quad \text{otherwise} \end{cases}$$

F = an n symbol inversion vector with each of its element in GF (2^p).

The state of a GF (2^p) CA at time *t* is an *n* symbol string, where a symbol $\in$GF(2*p*) is the content of a CA cell. If s_t represents the state of the automata at the t^{th} instant of time, then the next state, at the $(t + 1)^{\text{th}}$ time, is given by

$s_{(t+1)} = T * s_t + F$, and

$s_{(t+n)} = T^n * s_t + (I + T + T2 + \cdots + T^{n-1}) * F$.

The '*' and '+' operators are the operators of the Galois Field GF (2^p). If the *F* vector of GF (2^p) CA is an all zero vector, the CA is termed as linear CA; else it is an Additive CA [16].

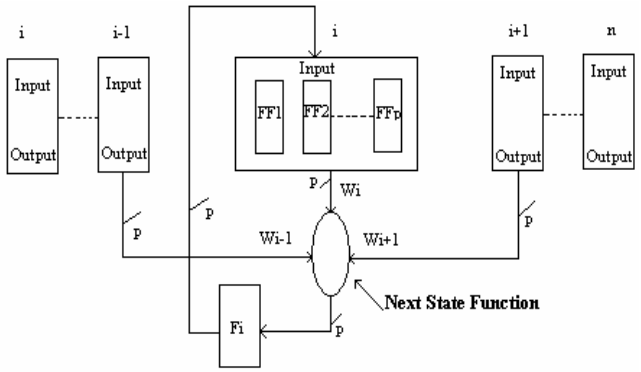

Fig. 1 General structure of a GF (2^p) CA
(For *p*=1, it's a conventional GF (2) CA)

REFERENCES

[1] I. J. Cox, J. Killian, F. T. Leighton, T. Shamoon, "Secured Spread Spectrum Watermarking for Multimedia", *IEEE Trans on ImageProcessing*, vol. 6, pp 601-610, June 2000.

[2] C. T. Hsu, J. L. Wu, "Hidden signatures in images", *Int. Conf. On Image Processing*, vol. 3, Switzerland, Sept. 1996.

[3] P. Meerwald, A. Uhl, "A survey of wavelet-domain watermarking algorithms", *Proc. of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents* III, CA, USA4314, Jan. 2001.

[4] H. S. Malvar and D. A. F Florencio, "Improved Spread Spectrum: A new Modulation Technique for Robust Watermarking," *IEEE Trans on signal processing*, pp 898-905, April 2003.

[5] J. Mayer, A. V. Silverio, J. C. M. Bermudez, "On the Design of Pattern Sequences for Spread Spectrum Image Watermarking," *InternationalTelecommunications Symposium,* Brazil.

[6] Saraju P. Mohanty, N. Ranganathan and K. Balakrishnan, "Design of a Low Power ImageWatermarking Encoder using Dual Voltage and Frequency," *VLSID2005.*

[7] T. S. Das, A. K. Sau, and S. K. Sarkar, "Spread Spectrum Image Watermarking for Secured Multimedia Data Communication", on *International Journal of Signal Processing,* Vol.3, No.3, ISSN 1304-4478, pp.148-157, (May, 2006).

[8] Niloy Ganguly, Biplab K Sikdar, Andreas Deutsch, Geoffrey Canright, and P Pal Chaudhuri, "A Survey on Cellular Automata".

[9] S. Nandi, B. K. Kar, and P. Pal Chaudhuri, "Theory and Applications of Cellular Automata in Cryptography," *IEEE Transactions on Computers*, Vol. 43, No. 12, December 1994.

[10] Dipanwita Roy Chowdhury, Saugata Basu, Indranil Sen Gupta, and Parimal Pal Chaudhuri, "Design of CAECC-Cellular Automata Based Error Correcting Code," *IEEE Transactions On Computers,* Vol. 43. No. 6, June 1994.

[11] Peter D. Hortensius, Robert D. Mcleod, Werner Pries, D. Michael Miller, And Howard C. Card, "Cellular Automata-Based Pseudorandom Number Generators for Built-In Self-Test," *IEEE Transactions On Computer-Aided Design,* Vol. 8. No. 8, August 1989.

[12] T. S. Das, V. H. Mankar, S. K. Sarkar, "Cellular Automata Based Robust Spread Spectrum Image Watermarking," *Indian Conference on Intelligent Systems ICIS' 07*, January 19-20, 2007.

[13] T. S. Das, V. H. Mankar, S. K. Sarkar, "Spread Spectrum based Robust Image Watermark Authentication," *International Conference on Advanced Computing & Communications ICACC2007*, February 9-10, 2007.

[14] S. K. Sarkar, *et al.*, "A One Dimensional Cellular Automata Based Security Scheme Providing Both Authentication and Confidentiality", *Journal of IEE*, Vol. 87, pp. 1-7, May 2006.

[15] Marco Tomassini, Moshe Sipper, and Mathieu Perrenoud, "On the Generation of High-Quality Random Numbers by Two-Dimensional Cellular Automata," *IEEE Transactions on Computers,* Vol. 49, No. 10, October 2000.

[16] P. Dasgupta, S. Chattopadyay, I. Sengupta, "An ASIC for Cellular Automata based Message Authentication," In Proc. Intl. Conf. on VLSI Design, India, pages 538-541, January 1999.

[17] Reiko Shiba, Seok Kang and Yoshinao Aoki, "An Image Watermarking Technique using Cellular Automata Transform," *IEEE conference proceedings*, 2004.
[18] Kolin Paul, D. Roy Choudhury and P. Pal Chaudhuri, "Cellular Automata based Transform Coding for Image Compression".
[19] M. Mukherjee, N. Ganguly, and P. Pal Chaudhuri. Cellular Automata based Authentication. In *Proc. of Fifth International Conference on Cellular Automata for Research and Industry, ACRI 2002, Switzerland,* pages 259-269, October 2002.

Vijay Harishchandra Mankar received the B. E. and M. Tech. degrees in Electronics Engineering from Nagpur University, MS, India in 1992 and 1995, respectively.
Presently working as a Lecturer in Government Residential Women's Polytechnic, Yavatmal (MS), India. He is currently deputed to Jadavpur University to carry out his Ph. D. under QIP. His field of interest includes digital signal processing, digital image processing, data hiding and watermarking.

Tirtha Sankar Das received his B. Tech. in Electronics and Telecommunication Engineering from Vidysagar University in year 2002 and M. E. from Bengal Engineering & Science University, Shibpore, WB, India in 2004.
At present he is a Lecturer in Electronics and Communication at Gurunanak Institute of Technology, Panihati, Kolkata, India. He is currently doing his Ph.D from Jadavpur University. His field of interest spans digital image processing, signal processing, communication and VLSI.

Dr. Subir Kumar Sarkar received his B. Tech. From University of Calcutta, India in 1981, M. Tech. from University of Calcutta, India in 1983 and Ph. D (Tech) from Institute of Radio Physics and Electronics, University of Calcutta in 1999.
From 1992 to 1999 he was lecturer in Bengal Engineering and Science University, Shibpore, Howrah. Currently he is Professor in Jadavpur University, Kolkata, India. His present field of interest is application of soft computing tools in simulations of device models and also in the field of High Frequency and Low Power Consuming Devices and their parameter optimization and wireless mobile communication. His research interests also include single electron devices and next generation digital electronics and image processing.

Fig. 3 Robustness efficiency for subjective recognition