Blind Audio Watermarking Technique Based on Two Dimensional Cellular Automata

Hazem Hiary[1], Abdel Latif Abu Dalhoum[1], Alia Madain[1], Alfonso Ortega[2] and Manuel Alfonseca[2]

[1]*The University of Jordan, Amman 11942, Jordan* [2]*Universidad Autónoma de Madrid, Madrid, Spain*

Abstract
In this paper we propose a new method of digital audio watermarking based on two dimensional cellular automata; the method increases the dimension of the audio and uses cellular automata in generating the key of watermark embedding. The watermarking method is blind, and does not require the original host audio or any of its features to extract the watermark; the watermark can be easily extracted using the right key. The experimental results show that the watermarks are imperceptible; and show a high similarity between the original and the watermarked audio. Cosine similarity and peak signal-to-noise ratio were used to measure the similarity between the original audio and the watermarked audio.
Keywords*: Digital Audio Watermarking, 2D Cellular Automata, Game of Life, Multimedia Security and Copyright Protection*

1. Introduction

Digital watermarking is a rapidly growing research field with a large number of papers published every year. While most researchers in the field of multimedia watermarking focused on image watermarking, in the last decade many audio watermarking schemes have been proposed [1].

Digital audio watermarking is the process of embedding information into a digital audio signal. The information embedded into the signal may be used for various applications such as authentication, copyright protection, and proof of ownership [2]. The most important audio watermarking techniques used are [3]: least significant bit (LSB), phase coding, echo hiding, and spread spectrum watermarking (SSW).

Cellular automata (CA) are simple models of computation which can exhibit complex behavior [4]. CAs were used to scramble digital multimedia files [5, 6] and in digital image watermarking [7, 8].

Despite the similarity between digital watermarking of image and audio files, there are major differences [9]. The generation of imperceptible audio watermarks is a challenging task. The human auditory system (HAS) is more sensitive than the human visual system (HVS), and hence the watermarking of audio signals is much more difficult than watermarking images [10].

Other than the difference in having less space to embed the watermark, audio watermarking reduces the image watermark from two-dimensional space to one-dimensional space before embedding, which reduces security [11]. In this paper, the proposed watermarking algorithm increases the dimension of the audio before embedding the watermark. The experimental results test the imperceptibility of the watermark using two formulas, namely, the peak signal to noise ratio and the cosine similarity.

This paper is organized as follows: Section 2 discusses related works; Section 3 provides background information about cellular automata and their capabilities; Section 4 gives the evaluation formulas used in the imperceptibility experiments; Section 5 provides the proposed embedding and extraction algorithms. In Section 6 results of our experiments are presented and discussed, and finally in section 7 we conclude the work done and give direction for future work.

2. Related Works

Audio watermarking schemes are diverse. Many researchers depend on methods used in artificial intelligence, such as watermarking based on kernel clustering [12], support vector machines [13], and Artificial Neural Networks [14]. The proposed scheme depend on cellular automaton which is a discrete model studied in computability theory.

In terms of audio watermarking these methods can be classified into two basic groups [15], one that embeds the watermark in the time domain [16], and others that embed the watermark in the transform domain [17]. Our proposed watermarking algorithm is considered a time domain algorithm and it is related to the scrambler we introduced earlier in [5].

Scramblers are mainly used as a pre-process and/or a post-process of a security related algorithm, such as watermarking algorithms. The watermarking scheme proposed here and the scrambling scheme proposed in [5] depends on the CA ability to generate new indices. In the scrambling case the indices are used to shuffle the host audio file and in the case of this paper the indices are used to embed the watermark.

Many audio scrambling schemes were proposed in literature, such as scrambling based on Arnold transform [18], Imitation [19] and cosine number transform [20]. An interesting work that is already applied in watermarking is the work done in [21] and [22], where the authors propose changing the dimensions to address the problems in one-dimensional linear mapping. In this paper, the dimension changes in order to map one-dimensional audio files to two-dimensional cellular automata (CA) indices.

We choose the CA scheme for many reasons. First, CA is not a specific approach to scrambling but a general model that have been proven to be capable of solving any computable task which can be of a special benefit in systems that rely on this model. Second, CAs design is open and flexible; there are no restrictions or mathematical formulas that restrict the construction of CAs [23]. Third, CA is a discrete computational model that is capable of different behavior dynamics resulting from different rules including stable, periodic, chaotic and complex ones. Finally, CA is known for being a parallel model, which increases the performance of applications relying on it.

In the domain of image watermarking, CAs were used as a pre-process of the watermarking, for example, in [7] CA were used to scramble the image before the watermark is embedded into the host image. CA were also used to generate a key to specify which pixel of the watermark is embedded in the host image first [8], where the watermark is the two most significant bits (MSB) of a gray image embedded in the least significant bits (LSB) of the host image.

In this paper we use two dimensional CA to watermark audio in the time domain. The audio dimension is increased before the watermark is embedded, and CA is used to generate a key of watermark embedding. The watermarking method is blind, and does not require the original host audio or any of its features to extract the watermark, and the watermark extraction process is straightforward if the right key is available.

3. Cellular Automata

In this section we discuss the type of CA used in our experiments. CAs are one of the oldest models of natural computing [24]. CA classes have been applied in many applications covering different domains, for example, CAs were used in Protein folding [23], modeling bacterial hyper structures [25], and in data mining [26].

The proposed algorithm uses two dimensional grid with a periodic boundary, where the extreme cells at opposite sides of the grid are adjacent to one another [27] therefore the grid used is considered a torus, and each cell in that grid has two possible states: *zero* or *one* which corresponds to *dead* or *alive* states in the game of life terminology. The transitional function uses the cell state and the state of its neighbors at time *t* in order to decide the cell's next state at time *t+1*.

There are many variations of the neighborhood type, for example, the most common type used in two-dimensional CA studies is called the von Neumann neighborhood which refers to the cells directly adjoining the four faces of a cell (the left, right, upper and lower cells), another common type is the Moore neighborhood which refers to all surrounding cells including the direct and diagonal ones (nearest eight neighbors) [28]. In our experiments we will use Moore neighborhood because we are using the game of life rules, which would not have been so successful if it were defined assuming the von Neumann neighborhood [29].

We can find classes of CA equivalent to Turing machines. According to Wolfram [30], it seems that the patterns which arise from different types of cellular automata can almost always be assigned to one of just four basic classes. In *class 1*, patterns evolve into a stable, homogeneous state; *class 2* patterns evolve to a periodic state; in *class 3* a chaotic behavior appears; and in *class 4*, configurations contain structures interacting in complex and interesting ways.

In this paper we consider the same transition rules used in the Game of Life (GL). GL belongs to Wolfram's *class 4*, GL rules are defined as follows [31]:

A. Survivals. Which means that if a cell is alive at time *t*, it will remain alive at time *t+1* if and only if it has either two or three live neighbors at time *t*. If the CA is simulated by counters and empty cells, then the survival happens when the counter with two or three neighboring counters survives for the next generation.

B. Deaths. Which means that if a cell is alive at time *t*, it will die at time *t+1* if four or more neighbors are alive at time *t*; also it will die if it has one or no alive neighbors at time *t*. If the CA is simulated by counters and empty cells, then each counter with four or more neighbors is removed from overpopulation and every counter with one neighbor or none dies from isolation.

C. Births. Which means that if a cell is dead at time *t*, it becomes alive at time *t+1* if exactly three of its neighbors are alive at time *t*. If the CA is simulated by counters and empty cells, then each empty cell adjacent to exactly three neighbors is a birth cell. A counter is placed on it at the next move.

4. Evaluation Formulas

The concept of similarity is an important evaluation criterion in many fields. Cosine similarity [32] is often used in information retrieval and to compare documents in text mining, but other uses exist, for example it is used in chemistry to calculate the similarity between molecules [33].

The similarity between two audio signals is calculated as shown in Equation 1 [11], where *N* denotes the length of the audio and *P* and *P'* denote the two different audio sequences. The resulting similarity (*SIM*) ranges from *0* to *1*, where *1* means that the two signals are exactly the same.

$$SIM = \frac{\sum_{i=0}^{N-1} |P\prime(i) \times P(i)|}{\sqrt{\sum_{i=0}^{N-1} P\prime(i)^2 \times \sum_{i=0}^{N-1} P(i)^2}} \tag{1}$$

The peak signal-to-noise ratio (*PSNR*) is a widely recognized measure used in quantifying the quality of multimedia; in digital audio watermarking *PSNR* is a common difference metric used to measure the fidelity of the watermarked audio signal [34]. As any difference metric, *PSNR* measures the difference between the original (*P*) and watermarked (*P'*) audio signals. Because signals can have a wide dynamic range, *PSNR* is usually expressed in terms of a logarithmic decibel scale.

Generally, a higher *PSNR* value indicates that the signal is of higher quality and that the watermarking algorithm makes the watermark more imperceptible. It can be calculated as follows [35]

$$PSNR = 10.\log_{10} \frac{N \times \max\limits_{1 \le i \le N} |P(i)|^2}{\sum_{i=1}^{N} (P(i) - P\prime(i))^2} \qquad (2)$$

where N is the length of the audio.

5. Proposed Algorithm

The watermarking process is divided into two processes, the first one is the embedding process where the watermark is inserted into the host audio signal in different positions, and the second one is the extraction process where the watermark is retrieved from the right positions based on the key.

5.1. Embedding Algorithm

The embedding algorithm takes the host audio file (H) and the watermark image (W) as input and produces the watermarked audio (R) and the key necessary for the extraction process. First the host audio dimension is increased to *2D*, then a cellular automaton (CA) is used to generate the list of indices (L) that specifies where to embed the watermark image (W). After embedding the watermark image bits in the positions specified by L, the host audio dimension is decreased to *1D* again.

By default, the embedding algorithm embeds all the watermark image bits, but if the algorithm is used to embed a smaller part of the watermark image bits then the proportion (P) must be specified.

Let N be the length of the host audio file and P the proportion the watermark takes from the bits of the host audio; the embedding algorithm is as follows:

(1) Insert the data samples of the host audio (H) in a square matrix A of size $M \times M$ where: $(M-1)_2 < N < M_2$. All the positions after the last occupied index are considered out of range.

(2) Create an initial random configuration K_0, of size $M \times M$ for cellular automata generations.

(3) Initialize a status matrix B of size $M \times M$ with all initial values set to *zero*.

(4) Starting with the initial configuration K_0, run cellular automata to get K_1. For each cell in K_1, if $K_1(i,j) = 1$, and the position *(i,j)* is within the range, add *(i,j)* to the list of indices L, and set *B(i,j)* to *1*.

(5) Set the number of generations (q) to *2*.

(6) While *length(L) < length(W)*, run cellular automata on the previous key K_{q-1} to get K_q, and increment q by *1*. For each cell in K_q, if $K_q(i,j) = 1$ and $B(i,j) = 0$, and the position *(i,j)* is within the range, add *(i,j)* to the list of indices L, and set *B(i,j)* to *1*.

(7) Insert the watermark bits into the matrix A in the positions specified by L (the number of bits inserted depends on the proportion P).

(8) Decrease the dimension of the matrix A to become one dimensional.

(9) Write the watermarked audio file (R) with the same parameters of the host audio file H (same format, sample rate and number of bits per sample).

The embedding process of the image watermark in the audio signal is shown in Figure 1, where the game of life rules generate a list of indices, and the watermark image bits are embedded in the original audio based on these indices to produce the watermarked audio.
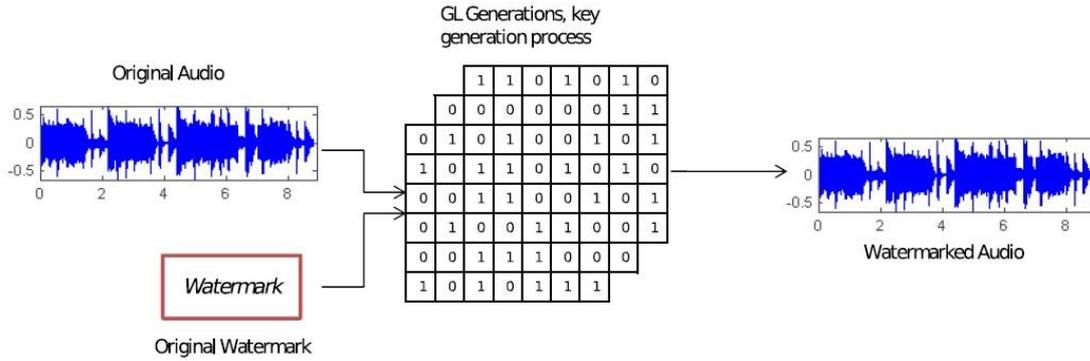
Figure 1. Embedding Process

5.2. Extraction Algorithm

The watermarked audio signal and the key are required for the extraction algorithm. The algorithm does not use the original audio signal or any of its features to extract the watermark. If not all the watermark image bits are embedded in the host audio and a proportion is specified in the embedding algorithm, then this proportion will also be required for the extraction algorithm.

The key is the initial random configuration K_0, from which the list of indices L can be regenerated, but in some applications there is no need to repeat the process of generating the indices, so the list of indices L can be used to extract the watermark directly.

The process of extracting the watermark is the inverse of the embedding process, where the insertion process is replaced with a retrieval process.

The key is given to indicate the positions the watermark is embedded in, without the key the watermark will look like a random one, which increases the watermark robustness.

6. Results

In an effort to measure the effect of the proposed scheme, two audio waves were watermarked using two different watermarks of different sizes, and the peak signal to noise ratio and cosine similarity between the original audio and the watermarked audio were calculated.

We chose two audio files; the first wave represents music content (Figure 3(a)) while the other represents speech content (Figure 4(a)). The details of both waves are given below in Table 1.

In our experiments only one channel is used, although the algorithm can be extended to audio files with multi channels. The extension can be made in two ways; the first is to embed the watermark into more than one channel, and the second is to repeat the same watermark in all channels.

Table 1. Description of the Tested Audio Files

File name	music	speech
Number of samples	422752	130000
Number of channels	2	1
Duration (seconds)	8.8073	2.9478
Bit rate (kbps)	1536	705
Number of bits/sample	16	16
Sample ratio	48000	44100
File type	WAV	WAV

Two gray image watermarks were used in the experiments, as shown in Figure 2; the size of the first watermark is *192×46* pixels, while the size of the second watermark (peppers image) is *128×128* pixels. Other types of watermarks are possible, such as text and binary image watermarks.


Figure 2. Watermarks used in Experiments

Figure 3 shows the original and the watermarked music wave file. The *PSNR* of the music file after embedding the first watermark is *62.07*, and *61.50* after embedding the second watermark (peppers image). The *SIM* in both cases is *0.99*, which indicates that the original and watermarked waves are very similar, and it is difficult to distinguish the difference between them.

The *PSNR* of the speech file is *60.38* when the first watermark is embedded and *59.75* when the second watermark is embedded and in both cases the *SIM* is *0.99*. Figure 4 shows the waveform of the original and the watermarked speech audio file. As can be seen, the waveform is very similar in structure and form to the original.
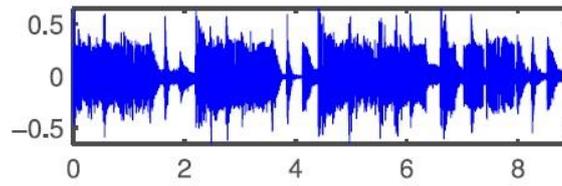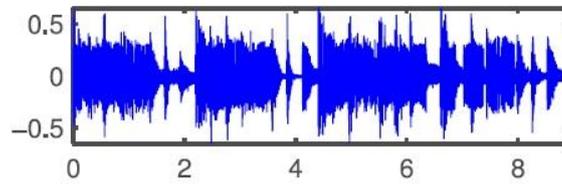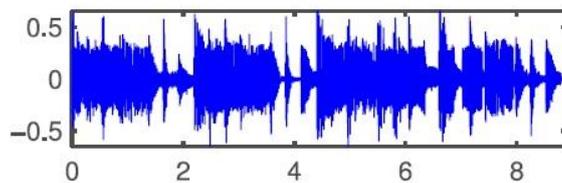
Figure 3. (a) Waveform of the original music file, (b) after embedding first watermark, (c) after embedding peppers watermark
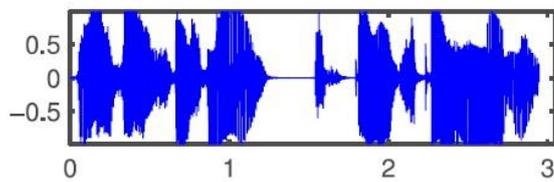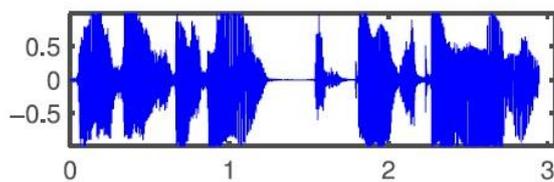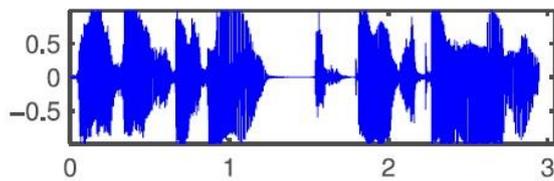
Figure 4. (a) Waveform of the Original Speech File, (b) after Embedding First Watermark, (c) after Embedding Peppers Watermark

Listening to the original audio and the watermarked audio, it is hard to tell the difference between them. If the audio is not attacked, then extracting the watermark returns the same embedded watermark, where the *PSNR* between the embedded and the extracted watermarks is infinity.

7. Conclusions and Future Work

In this paper, a digital audio watermarking scheme based on two-dimensional cellular automata was proposed. A Cellular automaton is used to generate the list of indices where the image watermark is embedded. The positions where the image watermark is embedded form the key used to extract the watermark.

The extraction process is simple when the key is available and does not require the use of the original audio. Nevertheless, if the key is not available, the watermark will look random. In the experiments the watermarks were chosen of different sizes and the host audio files used in the experiments cover music and speech files. In addition, the peak signal to noise ratio and cosine similarity between the original audio and the watermarked audio were calculated. The experimental results show that the watermark which is a gray image is imperceptible when embedded in the host audio file using the proposed scheme.

Future plans include extending the proposed scheme to include multichannel audio and studying other types of Cellular Automata and their effect on the watermarking results.

References

[1] M. Fallahpour and D. Megías, "High Capacity Method for Real-Time Audio Data Hiding Using the FFT Transform", Communications in Computer and Information Science, Springer, vol. 36, **(2009)**, pp. 91-97.

[2] V. Bhat K, I. Sengupta and A. Das, "Audio Watermarking Based on Quantization in Wavelet Domain", Springer, Lecture Notes in Computer Science, vol. 5352, **(2008)**, pp. 235-242.

[3] M. Davarynejad, S. Sedghi, M. Bahrepour, C. Ahn, M. Akbarzadeht and C. Coello, "Detecting Hidden Information from Watermarked Signal Using Granulation Based Fitness Approximation", Springer, Applications of Soft Computing, vol. 58, **(2009)**, pp. 463-472.

[4] P. Sarkar, "A Brief History of Cellular Automata", ACM Computing Surveys (CSUR), vol. 32, no. 1, **(2000)**, pp. 80-107.

[5] A. Madain, A. Abu Dalhoum, H. Hiary, A. Ortega and M. Alfonseca, "Audio scrambling technique based on cellular automata", Multimedia Tools and Applications, vol. 71, no. 3, **(2014)**, pp. 1803-1822.

[6] Abu Dalhoum, A. Madain and H. Hiary, "Digital image scrambling based on elementary cellular automata", Multimedia Tools and Applications, **(2015)**, pp. 1-16.

[7] R. Ye and H. Li, "A Novel Image Scrambling and Watermarking Scheme Based on Cellular Automata", Proceedings of International Symposium on Electronic Commerce and Security, Guangzhou City, China, **(2008)** August 3-5, pp. 938-941.

[8] O. Adwan O, A. Ayyal Awwad, A. Sleit and A. Abu Dalhoum, "A Novel Watermarking Scheme Based on Two Dimensional Cellular Automata", Proceedings of International Conference on Computers and Computing (ICCC'11), Lanzarote, Canary Islands, Spain, **(2011)** May 27-29, pp. 88-94.

[9] P. Bassia, I. Pitas and N. Nikolaidis, "Robust audio watermarking in the time domain", IEEE Transactions on Multimedia, vol. 3, no. 2, **(2001)**, pp. 232-241.

[10] Lei and K. Lo, "A Robust Audio Content Protection Scheme", Springer, Lecture Notes in Computer Science, vol. 5879, **(2009)**, pp. 491-498.

[11] H. Li, Z. Qin and L. Shao, "Audio Watermarking Pre-process Algorithm", Proceedings of IEEE International Conference on e-Business Engineering (ICEBE'09), Macau, China, **(2009)** 21-23 October, pp. 165-170.

[12] H. Peng, J. Wang and Z. Zhang, "Audio watermarking scheme robust against desynchronization attacks based on kernel clustering", Multimedia Tools and Applications, vol. 62, no. 3, **(2013)**, pp. 681-699.

[13] X. Wang, P. Niu and W. Qi, "A new adaptive digital audio watermarking based on support vector machine", Journal of Network and Computer Applications archive, vol. 31, no. 4, **(2008)**, pp. 735-749.

[14] L. Zheng, R. Chen, X. Cheng and L. Li, "Digital Audio Watermarking Based on Artificial Neural Networks", International Journal of Education and Management Engineering, vol. 2, no. 1, **(2012)**, pp. 23-28.

[15] R. Zezula and J. Misurec, "Audio Digital Watermarking Algorithm Based on SVD in MCLT Domain", Proceedings of International Conference on Systems (ICONS'08), Cancun, Mexico, **(2008)** 13-18 April, pp. 140-143.

[16] R. Martinez-Noriega, M. Nakano and K. Yamaguchi, "Self-Synchronous Time-Domain Audio Watermarking Based on Coded-Watermarks", Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), Darmstadt, Germany, **(2010)** 15-17 October, pp. 135-138.

[17] H. Yang, D. Bao, X. Wang and P. Niu, "A robust content based audio watermarking using UDWT and invariant histogram", Multimedia Tools and Applications, vol. 57, no. 3, **(2013)**, pp. 453-476.

[18] N. Augustine, S. N. George and P. P. Deepthi, "Compressive Sensing Based Audio Scrambling Using Arnold Transform", Communications in Computer and Information Science, Springer, vol. 420, **(2014)**, pp. 172-183.

[19] M. Ballesteros L and J. M. Moreno A, "Speech scrambling based on imitation of a target speech signal with non-confidential content", Circuits, Systems, and Signal Processing, vol. 33, no. 11, **(2014)**, pp. 3475-3498.

[20] J. B. Lima and E. F. da Silva Neto, "Audio encryption based on the cosine number transform", Multimedia Tools and Applications, vol. 75, no. 14, **(2016)**, pp. 8403-8418.

[21] H. Li and Z. Qin, "Audio scrambling algorithm based on variable dimension space", Proceedings of International Conference on Industrial and Information Systems (IIS'09), Haikou, China, **(2009)** 24-25 April, pp. 316-319.

[22] H. Li, Z. Qin, L. Shao, S. Zhang and B. Wang, "Variable Dimension Space Audio Scrambling Algorithm against MP3 Compression", Springer, Lecture Notes in Computer Science, vol. 5574, **(2009)**, pp. 866-876.

[23] A. Madain, A. Abu Dalhoum and A. Sleit, "Computational modeling of proteins based on cellular automata", International Journal of Advanced Computer Science and Applications (IJACSA), vol. 7, no. 7, **(2016)**, pp. 491-498.

[24] J. Kari, "Theory of cellular automata: a survey", Theoretical Computer Science, vol. 334, no. 1-3, **(2005)**, pp. 3-33.

[25] L. Sceller, C. Ripoll, M. Demarty, A. Cabin-Flamand, T. Nyström, M. Saier and V. Norris, "Modelling Bacterial Hyperstructures with Cellular Automata", Springer, Unifying Themes in Complex Systems, **(2006)**, pp. 147-156.

[26] T. Fawcett, "Data mining with cellular automata", ACM SIGKDD Explorations Newsletter, vol. 10, no. 1, **(2008)**, pp. 32-39.

[27] S. Shin and K. Yoo, "Analysis of 2-State, 3-Neighborhood Cellular Automata Rules for Cryptographic Pseudorandom Number Generation", Proceedings of International Conference on Computational Science and Engineering (CSE'09), Vancouver, BC, Canada, vol. 1, **(2009)** 29-31 August, pp. 399-404.

[28] L. Kier, P. Seybold and C. Cheng, "Modeling Chemical Systems Using Cellular Automata", Springer, Netherlands, **(2005)**.

[29] H. Nishio, "How Does the Neighborhood Affect the Global Behavior of Cellular Automata", Springer, Lecture Notes in Computer Science, vol. 4173, **(2006)**, pp. 122-130.

[30] S. Wolfram, "A New Kind of Science", Wolfram Media, USA, **(2002)**.

[31] M. Gardner, "Mathematical Games - The fantastic combinations of John Conway's new solitaire game "life"", Scientific American, vol. 223, **(1970)**, pp. 120-123.

[32] J. Zhang and R. Korfhage, "A distance and angle similarity measure method", Journal of the American Society for Information Science, vol. 50, no. 9, **(1999)**, pp. 772-778.

[33] A. Leach and V. Gillet, "An Introduction to Chemoinformatics", Springer, Netherlands, **(2007)**, pp. 99-117.

[34] Acevedo, "Audio watermarking quality evaluation", e-Business and Telecommunication Networks, Springer, **(2006)**, pp. 272-283.

[35] H. Peng and J. Wang, "Optimal audio watermarking scheme using genetic optimization", Annals of Telecommunications, vol. 66, no. 5-6, **(2011)**, pp. 307-318.