Reversible Watermarking Using Small-World Cellular Neural Network

Kazuya TSURUTA and Yoshifumi NISHIO

Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan
Phone:+81-88-656-7470
FAX:+81-88-656-7471
Email: tsuruta@ee.tokushima-u.ac.jp

1. Introduction

Studies of network map are very important, because they help us to understand the basic features and requirements of various systems. So far many connection topologies of network assumed to be either completely regular or completely random have been studied in the past. Cellular Neural Network (CNN) model invented by Chua and Yang in 1988 [1] is a typical of those completely local connectivities, which is presented as a preferred implementation of locally and regularly coupled neural networks. The CNN has been successfully used for various high-speed parallel signals processing applications such as image processing, pattern recognition as well as modeling of various phenomena in nonlinear systems [1]-[3]. However, in many cases in real life, many network topologies such as biological, technological and social networks are known to be not completely random nor completely local but somewhere in between. This was modeled in an interesting work by Watts and Strogatz in 1998 [5] as the small-world model. The model is a network consisting of many local links and fewer long range 'short cuts'. Therefore, it has a high clustering coefficient like regular lattices and a short characteristic path length of typical random networks. Interesting examples are shown by collaboration of movie stars, connectivity of internet web pages or neural nets, etc.

Recently, the authors have proposed Small-World Cellular Neural Network (SWCNN) [6, 7], which is constructed by introducing some random couplings between cells of the original Chua-Yang CNN. In [6, 7] we have reported some basic results using the concept of SWCNN. In this paper, we propose an image wartermarking technique that the connection topology of SWCNN plays the role of "key" and investigate its features.

2. Network Topologies of Watermark SWCNN

In this section, we describe the connection topology of SWCNN for watermark composed of a two-dimensional M by N array structure.

The watermark SWCNN is obtained by introducing random couplings between cells in the original Chua-Yang CNN. We introduce a probability p_c, which means what percentage of CNN cells occur random coupling to another cell. Namely, we choose a cell $c(p, q)$ that connects to $c(i, j)$ at random over the network. We repeat this process for all cells with the probability p_c.

We assume that, besides its local couplings, each cell in the array has up to one random coupling to another cell, moreover, the coupling direction is unilateral. Thereby, for $M \times N$ array, it has $M \times N$ couplings with maximum p_c. Figure 1 shows a sketch map of the SWCNN consisting of 4×4 cells. Obviously, when $p_c = 0$, the SWCNN is completely the same with the original CNN, and the maps correponding to $0 < p_c < 1$ and $p_c = 1$, shown in the middle part and the right hand side of Fig. 1, respectively.
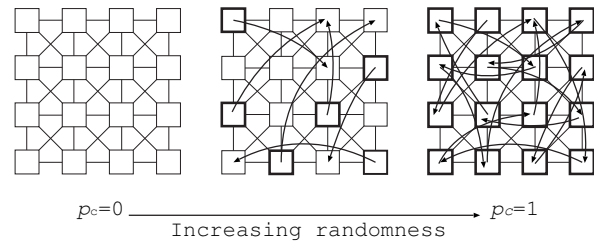

$p_c=0$ Increasing randomness $p_c=1$

Figure 1: A two-dimensional watermark SWCNN architecture with different probability p_c.

The state equation of each cell $c(i, j)$ of the SWCNN is formulated by Eq. (1), and the output equation is defined by Eq. (2).

$$\dot{x}_{ij}(t) = -x_{ij}(t) + I + \sum_{c(k,l) \in N_r(i,j)} A(i, j; k, l) y_{kl}(t)$$
$$+ \sum_{c(k,l) \in N_r(i,j)} B(i, j; k, l) u_{kl}(t) + w_c M(i, j; p, q) y_{pq}(t)$$

$$(1)$$

$$y_{ij}(t) = \frac{1}{2}(|x_{ij}(t) + 1| - |x_{ij}(t) - 1|) \qquad (2)$$
$$(i = 1, 2, ..., M, \quad j = 1, 2, ..., N.)$$

where $N_r(i, j)$ denotes the neighbor cells of radius r of a cell $c(i, j)$; A, B, and I are real constants called as feedback template, control template and bias current, respectively; x_{ij}, y_{ij} and u_{ij} denote the state, input and output of the cell, respectively; $M(i, j; p, q)$ describes the small-world map that is previously created by a program with indicating the probability p_c, if there is a coupling between one cell $c(i, j)$ and another cell $c(p, q)$, then the $M(i, j; p, q)$ is equal to 1, otherwise zero; and w_c stands for the coupling weight between the randomly coupled cells.

In order to investigate the features of the network, we calculated the characteristic path length $L(p_c)$ and the clustering coefficient $C(p_c)$ as varying p_c. The characteristic path length $L(p_c)$ is defined as the number of edges in the shortest path between two vertices, averaged over all pairs of vertices [5]. The clustering coefficient $C(p_c)$ is defined as follows; Suppose that a vertex v has k_v neighbours; then at most $k_v(k_v - 1)/2$ edges can exist between them. Let C_v denote that fraction of these allowable edges that actually exist. Define $C(p_c)$ as the average of C_v over all v [5].
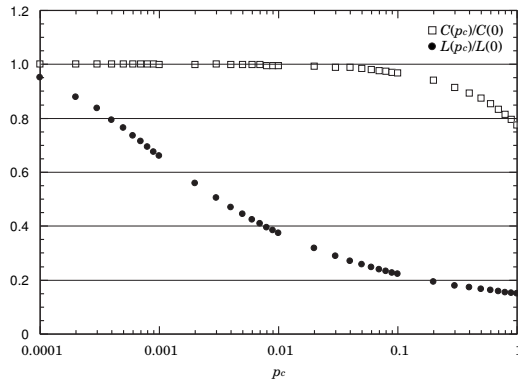

Figure 2: Characteristic path length and clustering coefficient of watermark SWCNN.

The results are shown in Fig. 2. Because we make a restriction such that one cell has at most one random coupling, the clustering coefficient does not approach zero even if p_c becomes 1.0. However, the characteristic path length becomes shorter and the network possesses the feature of the small-world networks.

3. Image Watermarking Using SWCNN

In this section, we describe the watermark technique using SWCNN. The watermark is a technique to embed some information to the target host image. The watermark embedded to the host image is used for to detect a modification,

authenticate a image and protect a copyright. Here, watermarks should be invisible for observer. The watermark is classified into robust and fragile technique. A robust watermark is intended to leave the mark by resisting attacks such as modification and destroy the mark for copyright protection, meanwhile a fragile watermark is able to detect alteration by breaking the mark actively.

3.1. Watermark Generator

In CNN literature, pseudo-random pattern generator for cryptograph and the watermarking technique based on cellular automaton are described [8, 9]. In our method, the watermark image is scrambled by randomly coupled cells, and its connection pattern plays the role of key. Moreover, $p_c = 1.0$ is desirable to distribute the scrambled image uniformly. For example, we adopted the template as follows:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \begin{array}{l} B = 0, \\ I = 0, \\ w_c = -1. \end{array} \qquad (3)$$

Simulated results are shown in Fig. 3. We observe from this simulation, the SWCNN can scramble the input image and the output pattern is not correlated with the initial state image. Needless to say, the output image changes by adopting other keys.
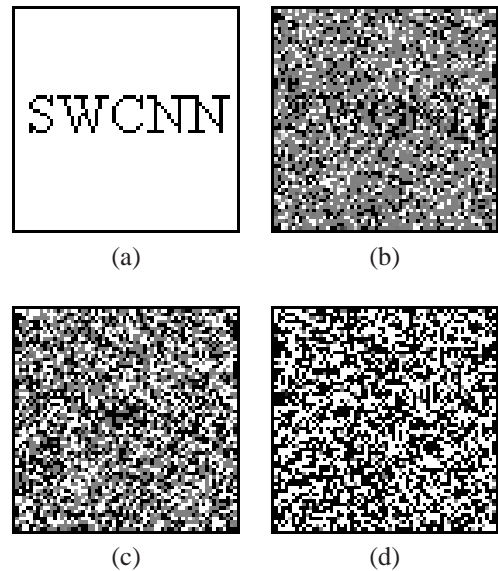

Figure 3: (a) 64×64 initial state image with $p_c = 1.0$. (b)-(d) Output images at $t = 5\tau$, 10τ and 50τ.

As already mentioned, the watermark is generated by the small-world map. Hence, it is able to generate watermark and to embed it to the host (input) image simultaneously by

SWCNN. The watermark SWCNN template form becomes as:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ w_c = c, \quad I = 0. \quad (4)$$

Thus, the cell's state equation in the watermark SWCNN can be rewritten in the following form:

$$\dot{x}_{ij} = -x_{ij} + f(x_{ij}) + w_{ij} \quad (5)$$

where,

$$f(x_{ij}) = 0.5a(|x_{ij} + 1| - |x_{ij} - 1|),$$
$$w_{ij} = 0.5c(|x_{pq} + 1| - |x_{pq} - 1|) + bu_{ij}.$$

Since we run the system backwards to recover the watermark image from marked image, it is necessary to limit the range of state value $|x_{ij}| \leq 1$. From Eq. (5), for the self-feedback a, the dynamic routes are shown in Fig. 4.
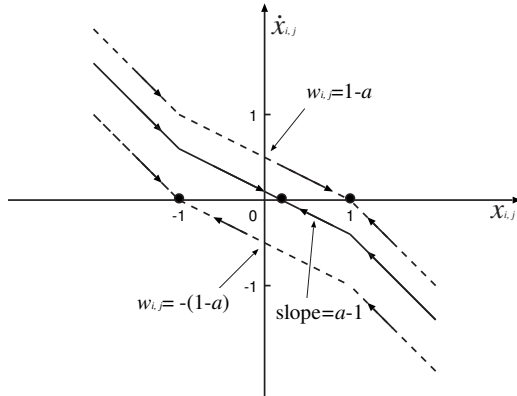

Figure 4: The dynamic routes of watermark SWCNN with various w_{ij}, where $0 < a < 1$ and black dots in $\dot{x}_{ij}$-intercept denote the stable point.

From the above, we assume that the parameters are satisfy the following conditions;

$$|a| + |c| + |b| \leq 1. \quad (6)$$

3.2. Simulation

In the watermark SWCNN, the number of the key increases drastically as the image size increases. This feature is suitable to embed a secret information (image). In this simulation, the secret Lena image is embedded to the host image with parameter set as: $a = 0.005$, $b = 0.99$ and $c = -0.005$. Simulation results are shown in Fig. 5. The watermarked image Fig. 5(d) contains visually imperceptible scrambled Lena image.

The robustness can be improved by increasing the parameter a and $|c|$, or decreasing the parameter b. The simulation results with different parameter settings are shown in Fig. 6. Figure 6(a) is a host image and Fig. 6(b)-(f) show the results after watermarking with $a = -c = (1 - b)/2 = 0.02, 0.04, 0.06, 0.08$ and 0.1, respectively.

We run the system backwards to recover the embedded image from the watermarked image, and this is a private watermark that requires host and marked image. Ordinarily, the output image is quantized to 8*bit* grayscale image in CNN, but because of quantization error, it is difficult to recover the image completely. Therefore, we use 16*bit* quantized image for this purpose. The simulation results are shown in Fig. 7. The parameter set are the same as watermarking operation. From Fig. 7, we observed that the watermarking SWCNN is reversible.

Figure 5: (a) Input (host) image. (b) Initial state (watermark) image. (c) Transient image at $t = 1\tau$. (d) Watermarked output image with embedded Lena image.

4. Conclusions

In this article, a watermarking technique using SWCNN has been proposed and its features are investigated. We designed template of SWCNN for the watermaked image reversibly. Moreover, the proposed method is able to generate watermark and embed to the host image simultaneously. Although the techniques which use pseudo-random pattern are existing, the proposed method has shown a new potential of SWCNN. Our important future researchs are investigations of robustness of the watermarking SWCNN and analyzing it in more detail.
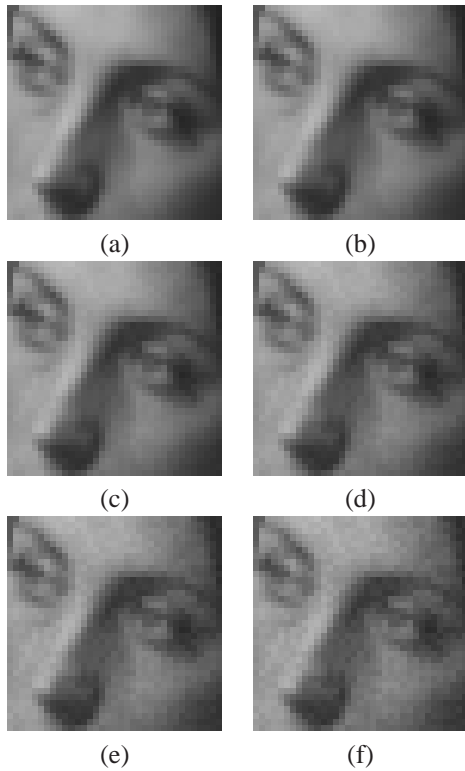
Figure 6: Zoomed input and marked image. (a) Input(host) image. (b)-(f) Watermarked image with $a = -c = (1-b)/2 = 0.02, 0.04, 0.06, 0.08$ and 0.1, respectively.

Figure 7: Extracting embedded information. (a) Input image. (b) Marked image. (c) Transient image at $t = 6\tau$. (d) Recovered watermark output images.

References

[1] L.O. Chua and L. Yang, "Cellular neural networks: theory and applications," IEEE Trans. Circuits & Syst., vol.35, no.10, pp.1257-1290, Oct. 1988.

[2] L.O. Chua and T. Roska, "The CNN paradigm," IEEE Trans. Circuits & Syst., vol.40, no.3, pp.147-156, Mar. 1993.

[3] Z. Yang, Y. Nishio and A. Ushida, "Image processing of two-layer CNNs – applications and their stability –," IEICE Trans. Fundamentals, vol.E85-A, no.9, pp.2052-2060, Sep. 2002.

[4] Z. Yang, Y. Nishio and A. Ushida, "Characteristic of mutually coupled two-layer CNN and its stability," Journal of Circuits, Systems, and Computes, vol.12, no.4, pp.473-490, Aug. 2003.

[5] D.J. Watts and S.H. Strogatz, "Collective dynamics of small-world networks," Nature 393, pp.440-442, 1998.

[6] K. Tsuruta, Z. Yang, Y. Nishio and A. Ushida, "Small-world cellular neural networks for image processing applications," Proceedings of ECCTD'03, vol.1, pp.225-228, Sep. 2003.

[7] K. Tsuruta, Z. Yang, Y. Nishio and A. Ushida, "Diffusion analysis of diffusion-preserving small-world CNN," Proceedings of CNNA'04, pp.352-357, Jul. 2004.

[8] K.R. Crounse, T. Yang and L.O. Chua "Pseudo-random sequence generation using the CNN universal machine with application to cryptography," Proceedings of CNNA'96, pp.433-438, Jun. 1996.

[9] M.E. Yalçin, J. Vandewalle, P. Arena, A. Basile and L. Fortuna "Watermarking on CNN-UM for image and video authentication," International Journal of Circuit Theory and Applications, vol.32, pp.591-607, Apr. 2004.