Digital Watermarking Based on Stochastic Resonance Signal Processor

Shuifa Sun[1,2,3], Sam Kwong[2], Bangjun Lei[3], and Sheng Zheng[1,3]

[1] College of Electrical Engineering and Information Technology, China Three Gorges University, Yichang 443002, China
{watersun, zsh}@ctgu.edu.cn
[2]Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong SAR, P.R. China
CSSAMK@cityu.edu.hk
[3]Institute of Intelligent Vision and Image Information, China Three Gorges University, Yichang 443002, China
Bangjun.Lei@ieee.org

Abstract. A signal processor based on an bi-stable aperiodic stochastic resonance (ASR) is introduced firstly. The processor can detect the base-band binary pulse amplitude modulation (PAM) signal. A digital image watermarking algorithm in the discrete cosine transform (DCT) domain is implemented based on the processor. In this algorithm, the watermark and the DCT alternating current (ac) coefficients of the image are viewed as the input signal and the channel noise of the processor input, respectively. In conventional watermarking systems, it's difficult to explain why the detection bit error ratio (BER) of a watermarking system suffering from some kinds of attacks is lower than that of the system suffering from no attack. In the present watermarking algorithm, this phenomenon is systematically analyzed. It is shown that the DCT ac coefficients of the image as well as the noise imported by the attacks will cooperate within the bi-stable ASR system to improve the performance of the watermark detection.

Keywords: digital watermarking; stochastic resonance; signal processing.

1 Introduction

The concept of digital watermarking has been put forward for more than 10 years [1-4, 15]. Digital watermarking plays a very important role in preventing multimedia works from being pirated. Its idea is to embed some critical information by replacing parts of original media data with certain so-called watermark. The watermark can then be detected purpose by the receiver. Two approaches to digital watermarking were proposed for grayscale images in [1]. In the first approach, a watermark in the form of a sequence-derived pseudo noise (PN) codes is embedded in the least significant bit (LSB) plane of the image data. A frequency-domain watermarking was introduced by Cox and Kilian [2]. The watermark contains a sequence of numbers with a normal distribution with zero mean and a variance of one. The watermark is inserted into the

H.H.S. Ip et al. (Eds.): PCM 2007, LNCS 4810, pp. 367–376, 2007.

image in the frequency-domain to produce the watermarked image. To verify the presence of the watermark, the similarity between the recovered watermark and the original watermark is measured. A blind watermarking detection algorithm based on the correlation detection was further proposed by Barni [3]. Barni computed the global DCT transformation for a given image, and then selected some middle and low frequency DCT coefficients to embed the marker. Quantization-index modulation (QIM) methods, introduced by Chen and Wornell [4], possess attractive practical and theoretical properties for watermarking. The method consists on coding the message by modifying the original data itself, where the elements of the message act as an index that select the quantizer used to represent them. Imperceptivity and robustness are two basic requirements to digital watermarking. Imperceptivity means that the difference between the embedded media and the original media should be imperceptible. For instance, the difference should not be easily perceived by humans' eyes for an image watermarking. Robustness means that the watermarking system should be able to survive some attacks. From the signal processing perspective, however, digital watermarking involves detecting weak signals in the presence of strong background noises.

Stochastic resonance (SR) is an effective approach for signal processing [5-11]. From this perspective, SR effect is commonly understood as first an increase and then a decrease in the signal-to-noise ratio (SNR) at the output with varying noise level at the input. Other quantitative measures, such as bit error ratio (BER), can also be employed. In this study, a signal processor based on ASR [7-10] is investigated. A digital image watermarking algorithm in the discrete cosine transform (DCT) domain is then implemented based on this ASR signal processor.

The paper is organized as follows. A signal processor based on the nonlinear bi-stable dynamic system is investigated in Section 2. A digital image watermarking algorithm in DCT domain is implemented based on the ASR signal processor in Section 3. Experimental results are presented in Section 4 and conclusions are drawn in Section 5.

2 Bi-stable ASR signal processor

From the signal processing perspective, the mathematical model of a nonlinear bi-stable dynamic system can be written as

$$dx/dt = -dV(x)/dx + Input(t), \tag{1}$$

where $V(x)$ is the quartic potential function and can be written as $V(x) = -ax^2/2 + \mu x^4/4$. The parameters a and μ are positive and given in terms of the potential parameters. The quartic potential function $V(x)$ represents a bi-stable nonlinear potential with two wells and a barrier. The input can be written as $Input(t) = h(t) + \xi(t)$, where $h(t)$ is the input signal and $\xi(t)$ is the noise. If the signal $h(t)$ is an aperiodic signal and the SR effect occurs, it is called an ASR system [4].

This bi-stable system was used by Hu *et al.* [7], Godivier & Chapeau-Blondeau [8] and Duan & Xu [9] to detect the base-band pulse amplitude modulated (PAM) aperiodic binary signal $h(t)$ in the presence of the channel noise $\xi(t)$. The signal waveforms can be represented as $h_1(t) = -A$ and $h_2(t) = A$ for $(n-1)T_s \leqslant t < nT_s$, $n = 1,2,\ldots$

If the amplitude of the aperiodic signal A is not larger than the critical value A_{CR} $=\sqrt{4a^3/27\mu}$ [11], then the input signals is called sub-threshold signal and supra-threshold signal otherwise. Here, a parameter Q_{SR} called SR-Degree is defined as follows:

$$Q_{SR} = A/A_{CR}. \tag{2}$$

When the SR-Degree Q_{SR} <1, the aforementioned nonlinear bi-stable dynamic system corresponds to the sub-threshold system and supra-threshold system otherwise. This study is limited in the sub-threshold system, i.e. $A < A_{CR}$. The time interval T_s is termed as bit duration and the code rate $r=1/T_s$. The BER of the system is $P_e= P(1)P(0|1)+ P(0)P(1|0)$. Readers should refer to [7-10] for more details about this signal processor.

3 Watermarking Based on ASR Signal Processor

In the remainder of this paper, a digital image watermarking algorithm based on the aforementioned ASR signal processor is implemented. The watermark sequence and the DCT alternating current (ac) coefficients of the image are viewed as the weak signal and the noise of the ASR signal processor, respectively, making up the input of the ASR signal processor.

3.1 Watermark Embedding

Referring to Eq. (1), the DCT ac coefficients can be viewed as the sampled noise $\zeta^E(l)$ ($0 \leqslant l < L$ and L is the total number of DCT ac coefficients used) and the watermark as the signal $h^E_i(j) \in \{1,-1\}$ ($i \in \{1,2\}$, $h^E_1=1$, $h^E_2=-1$, $0 \leqslant j < J$, and J is the length of the binary watermark sequence). The steps for the watermark embedding are as follows:

STEP 1. Forward DCT

For a image A^S of size $N_1 \times N_2$, applying global DCT to it, we get a DCT coefficient matrix B^S of the same size, where N_1 and N_2 are the width and the height of the image, respectively.

STEP 2. Selecting DCT ac coefficients

Selecting the DCT ac coefficients from matrix B^S line by line and sorting them according to their absolute values, we get a vector C^S. Those coefficients whose absolute values lie between M and $M+L-1$ elements in the vector C^S is picked up and form a vector C^E of length L. In this process, $1 \leqslant M$ and $(M+L) < N_1 \times N_2 - 1$. This means that the DCT direct current (DC) coefficient is not used, which is usually the biggest one in all the DCT coefficients and at the position 0 in the vector C^S. It is because that a change in this coefficient during the watermarking process will modify the mean level of the image and thus makes it perceptible. In fact, the same logic also holds for the first few DCT ac coefficients. Such as for the simulations given in Sec.4, M is 1000. At the same time, the position of the selected coefficients C^E in matrix B^S is recorded for later use.

STEP 3. Randomization of the selected DCT ac coefficients
The DCT ac coefficients of image are rearranged after the DCT transformation. The magnitudes of the low frequency coefficients are usually larger than those of the high frequency coefficients. In order to make the DCT ac coefficients more like the white noise, the selected DCT coefficients should be disordered. The method used is described as follows:

A. Given a random seed k, we generate a sequence of random numbers, which will be recorded and served as position index information in following step.
B. For an empty vector ξ^E with the same size of C^E, we select the DCT ac coefficients from C^E one by one, and put them at certain position of the vector ξ^E according to the position information obtained in the previous Step A.

It is clear that the position index sequence changes according to the random seed k. So, the parameter k can work as the key of the watermarking system. The parameter M and L can also work as the key of the watermarking system if necessary. But it is more appropriate to transmit it as the side information of the system.

STEP 4. Spreading the watermark sequence
Spreading the watermark sequence from one of length J to one of length L with an up-sample function and making $h_i^E(j_q \Delta t) = h_i^E(j)$ ($0 \leqslant q < Q$, where $Q \times J = L$ and $Q\ \Delta t$ equals to the duration of a binary watermark code), we then obtain a spread watermark sequence. To simplify, we remove Δt of $h_i^E(j_q \Delta t)$ in the numerical simulation, which means that $h_i^E(j_q)$ is the qth sample of the jth watermark code.

STEP 5. Embedding watermark
The watermarked sequence $\xi^M(l)$ is obtained according to the following Eq. (3)

$$\xi^M(l) = f \times h_i^E(j_q) + \xi^E(l), \tag{3}$$

where f is the scaling parameter. Using the positions recorded in STEP 3 and STEP 2 accordingly, we can replace the selected coefficients in the matrix B^S with the watermarked sequence $\xi^M(l)$ one by one and get the watermarked matrix B^M.
STEP 6. Inverse DCT
Applying the inverse DCT on the matrix B^M, we obtain the watermarked image A^M.

3.2 Watermark Detection

The first three steps for the watermark detection are the same as those in the above embedding procedure. The whole procedure for watermark detection is as follows:

STEP 1. Forward DCT
Given a $N_1 \times N_2$ image matrix A^U, applying DCT, we get a DCT coefficient matrix B^U.

STEP 2. Selecting DCT ac coefficients
Using the position information recorded in STEP 2 of subsection 3.1, we can select DCT ac coefficients at those positions from the matrix B^U and get a vector C^U.

STEP 3. Randomization of the selected vector
Using the random seed k to randomize the selected vector C^U with the same method given in STEP 3 of subsection 3.1, we obtain a vector $\xi^U(l)$ to test.

STEP 4. Detection
Substituting the sequence $\xi^U(l)$ into Eq.(1) as the sampled input $Input(t)$ of the bi-stable system, we get the following Eq. (4) [14]

$$x(l+1) = \Delta t(ax(l) - bx^3(l) + \xi^U(l)) + x(l). \tag{4}$$

Without loss of generality, we may assume that $x(0)=0$ in the numerical simulation. We can then get the detected sequence $h_i^D(j_q) = x(l+1)$ and recover the watermark h_i^D by using the sample at the end time of each bit duration [8] or use the statistical method proposed in [10] to improve the detection performance of the system. Comparing h_i^D with h_i^E, the total bit error number and BER of the watermarking system are obtained.

For both watermarking algorithms in [2] and [3], the DCT ac coefficients of the image are actually viewed as the additive white Gaussian noise because the correlation detector, which is the optimal detector for the communication system in the presence of this kind of noise, is used. In the present watermarking system, not only the selected DCT ac coefficients sequence but also the noise imported by the attack is viewed as the additive white Gaussian noise. If the image A^U is the watermarked image A^M, which means that the watermarking system has not been attacked, the aforementioned ASR signal processor works well. If indeed the watermarking communication system suffers from some kinds of attack, $\xi^U(l)$ is still a white Gaussian noise because it is a combination of two additive white Gaussian noises. Therefore, the aforementioned ASR signal processor will work well in both cases.

4 Numerical Simulations and Results

In this section, the numerical simulations on the above implementation are introduced. The original image is the standard gray-scale image "Lena" with a size of 512×512, as shown in Fig. 1(a). The watermark is a PN sequence h_i^E generated by a linear feedback shift register with 8 registers. The length of the watermark code J is 255. The first 10 codes are shown in Fig. 1(b). The parameters M and L for the simulations were 1000 and 255×500, respectively. This means that the parameter Q=500. The scaling parameter f was 5. The watermarked image is shown in Fig. 1(c) and the peak signal to noise ratio ($PSNR$) was 37.28 dB. Comparing it with the original image in Fig. 1(a), we hardly can see any difference. From Fig.2 one can clear see that the parameter-induced stochastic resonance phenomenon happened in the watermarking system. In the following attack tests, the optimal value a_{SR} (=39) will be used to detect the watermark.

4.1 Adding Salt and Pepper Noise Attack Test

Some pulse noises are added into the image when the image is transmitted or processed. This kind of noises are often called salt and pepper noises and make the

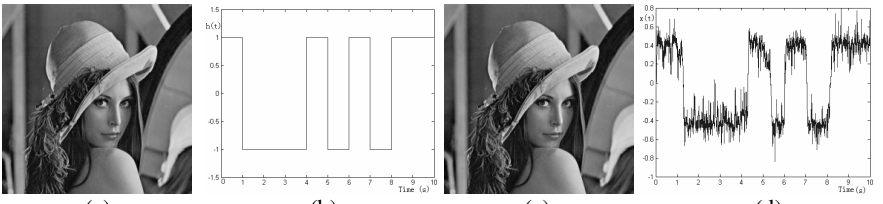


Fig. 1. (a) Original image 'Lena'. (b) Original watermark code. (c) Watermarked image. (d) Detected sequence from (c) when the bi-stable system parameter a=39. Other simulation parameters are the same as those in Fig. 2.

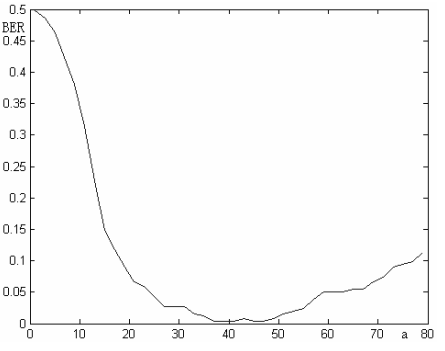
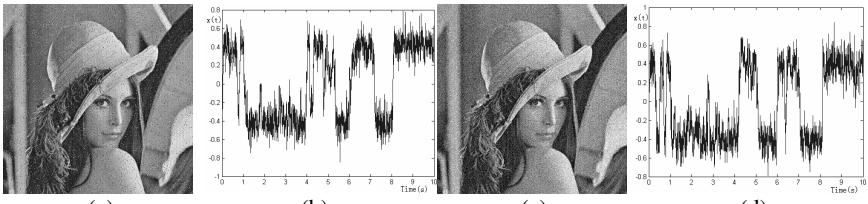


Fig. 2. Relationship between the *BER* and the parameter a of the bi-stable system. The SR-Degree Q_{SR} is 0.86. The parameter a changes from 1 to 80 with a step of 2. The sample interval Δt in Eq. (4) is 0.002. The recovered waveform for the first 10 codes is shown in Fig. 1 (d) where the bi-stable system parameter a =39.

image looking quite dirtier and older. Fig. 3(a) is the image when the watermarked image was corrupted by the salt and pepper noise with density of 2.5%, the peak signal to noise ratio (PSNR) was 21.37 dB. Comparing it with the watermarked image in Fig. 1(c), the difference is obvious. As shown in Fig. 3(b), however, the trace of the watermark is clear and there are only 8 error codes among all 255 transmitted codes.

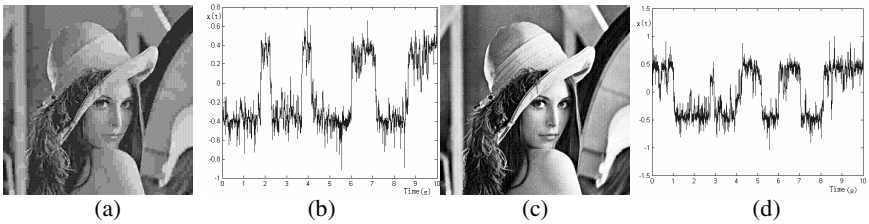
4.2 Adding Gaussian Noise Attack Test

Another typical noise in the image transmission or processing is Gaussian noise, which makes an image blurry. Fig. 3 (c) is the image when the watermarked image is corrupted by a Gaussian noise whose mean and covariance are 0 and 0.01, respectively. Comparing it with the watermarked image in Fig.1 (c), we can see the difference between them clearly and *PSNR*=20.06 dB. But the trace of the watermark is still obvious as shown in Fig.3 (d) and there were only 9 error codes among all 255 transmitted codes.



(a) (b) (c) (d)

Fig. 3. Adding salt and pepper noise attack case and adding Gaussian noise attack case. (a) Attacked image after adding spiced salt noise. (b) Detected waveform from (a) for the first 10 codes. (c) Image after adding Gaussian noise. (d) Detected waveform from (c) for the first 10 codes. Other simulation parameters are same with those of Fig. 2.



(a) (b) (c) (d)

Fig. 4. JPEG compression attack case and histogram equalization attack case. (a) Image after JPEG compression. (b) Detected waveform from (a) for the first 10 codes. (c) Image after histogram equalization. (d) Detected waveform from (c) for the first 10 codes. Other simulation parameters are the same as those in Fig. 2.

4.3 JPEG Compression Attack Test

JPEG compression is a widely used image processing method. Fig. 4(a) is the image when the watermarked image was compressed using JPEG with a quality factor 5. Comparing it with the watermarked image in Fig. 1 (c), the difference caused by block effect is obvious and *PSNR*=26.95 dB. But from the detected result shown in Fig. 4(b), there was still a trace of the watermark and the *BER* is 39.6%. Our explanation is given as follows: the JPEG compression forced many high frequency DCT ac coefficients to zeros and the watermark embedding in these coefficients were then removed. It is why that many digital watermarking algorithms do not use high frequency DCT ac coefficients as the carrier of watermark. Embedding watermark only into the low frequency DCT ac coefficients will overcome this defect of the watermarking system.

4.4 Histogram Equalization Attack Test

In the image processing, histogram equalization is usually used to increase the contrast of an image. Fig. 4(c) is the resulting image when the histogram of the watermarked image was equalized. The difference between them can be easily told by comparing it with the watermarked image in Fig. 1(c) (*PSNR*=19.18dB). But from the detected result shown in Fig. 4(d), there were trances of the watermark code and there were only 6 error codes among all 255 transmitted codes.

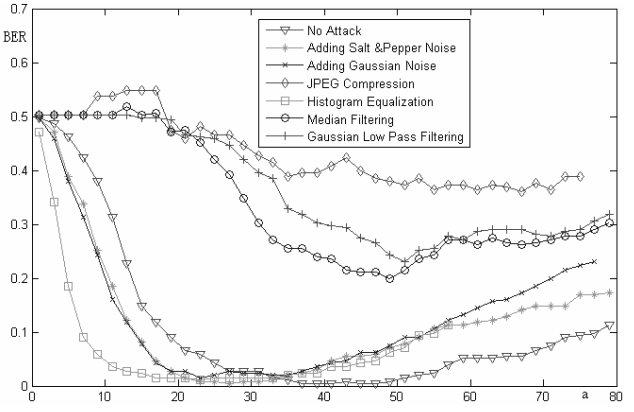


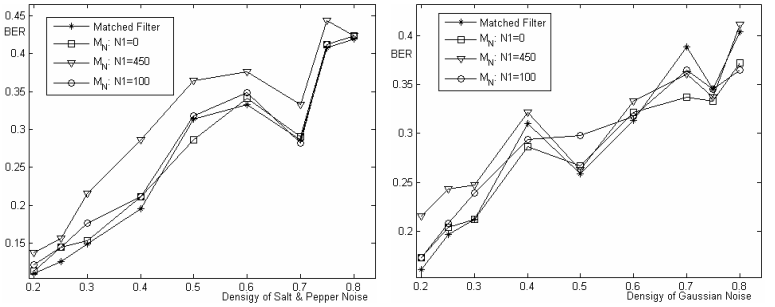
Fig. 5. Relationship between *BER* and the parameter *a* of the bi-stable system in the watermarking communication systems for the image Lena. Other simulation parameters are same with those of Fig. 2.

4.5 SR Effect in the Presence of Attack

Figure 5 shows the relationship between the parameter *a* and the *BER* for all of the preceding attacks cases, as well as the no attack case. one can clearly see that the parameter-induced stochastic resonance effect in some of the watermarking communication systems being attacked. So, not only the DCT ac coefficients of the image themselves but also the combination of them and the noise imported by the attacks will cooperate with the bi-stable system to improve the watermark detection performance. It is also clear that the number of erroneous bits when the system suffers from the pulse noise attack or the Gaussian noise attack is less than that when the system does not suffer any attack, such as a=17 in Fig.5. For both watermarking algorithms proposed by Cox and Kilian [2], and Barni *et al.* [3], the DCT ac coefficients of the image are also viewed as the noise of the watermarking communication system. Whereas, for watermarking algorithms based on the ASR signal processor, the detection BER when the system suffers from some kinds of attacks is lower than that when the system does not suffer from any attacks. This is unbelievable for conventional watermarking systems but is reasonable for the ASR signal processor based on the nonlinear system, which the conventional linear signal processor does not have [8].

4.6 Comparison with the Matched Filter

We next wish to compare the ASR signal processor with the matched filter. The matched filter is the optimal linear filter in the presence of additive Gaussian noise. The matched filter correlates with the received signal (the signal-noise mixture) $u(t)$ with a replica of a binary pulse of the watermark signal $h(t)$. In our case, the impulse response of the matched filter $r(t)$ is A if $t \in [0, T_s]$ and 0 otherwise. The signal at the output of the matched filter y (t) is $y(t) = \int_{-\infty}^{t} r(t-t')u(t')dt'$. At every time



(a) (b)

Fig. 6. Comparison with the matched filter. The other parameters for the experiments are same with those of Fig. 2. M_N stand for the new method given in [10] and N1 is also defined in [10].

multiple of input pulse duration T_s, the output of the matched filter $y(t)$ was read and a decision was made that $u(t)= A+ \xi(t)$ if $y(t)>0$ and $u(t)= -A + \xi(t)$ otherwise (assuming $P_0=P_1$). Every decision was perfectly synchronized with the end of each binary pulse of the information-carrying signal $h(t)$. In Fig. 6, we compare the performance of the ASR signal processor to that of the matched filter, with these relevant conditions of detection. It can be seen from Fig.6 that the performance of the matched filter is not always better than that of the ASR signal processor, especially when the density of the noise is high. The results also show that the method proposed in [10] is an effective method when the noise density is high.

5 Conclusion

In this paper, a signal processor based on ASR was investigated. A digital image watermarking algorithm based on this ASR signal processor was then implemented. The experimental results showed that, under certain circumstances, extra amount of noises can in fact improve rather than deteriorate the performance of some communication systems. It was also found that the performance of a matched filter is not always better than that of the ASR signal processor. In the algorithm, both the selected DCT ac coefficients and the noise imported by the attacks are viewed as the additive white Gaussian noise. However, as shown by the results in [15], the characteristics of cover data (the selected DCT ac coefficients here) and the distortion vectors (the noise imported by the attack) are different for given images and attacks. So, further extension to this paper could be to study the ASR signal processor in the presence of other kinds of channel noises [16-17] and apply it to watermarking.

Acknowledgments. This work was supported partly by a grand from the National Natural Science Foundation of the People's Republic of China, No. 60572048. Shuifa Sun acknowledges the City University of Hongkong Strategic Grant 7001955.

References

1. Schyndel, R.G., Tirkel, A.Z., Osborne, C.F.: A Digital Watermark. In: Proc IEEE Int Conf on Image Processing, pp. 86–90. IEE Computer Soc. Los Alamitos (1994)
2. Cox, I., Kilian, J.: Secure spread spectrum watermarking for multimedia. IEEE Trans. on Image Processing 6, 1673–1687 (1997)
3. Barni, M., Bartolini, F., Capellini, V., Piva, A.: A DCT-domain system for robust image watermarking. Signal Processing 66, 357–372 (1998)
4. Chen, B., Wornell, G.: Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding. IEEE Trans. on Information Theory 47, 1423–1443 (1998)
5. Benzi, R., Sutera, S., Vulpiani, A.: The Mechanism of Stochastic Resonance. J. Phys. A 14, 453–457 (1981)
6. Collins, J., Chow, C., Imhoff, T.: Aperiod stochastic resonance in excitable systems. Phys. Rev. E. 52(4), R3321–R3324 (1995)
7. Hu, G., Gong, D., Wen, X., et al.: Stochastic resonance in a nonlinear system driven by an aperiod force. Phys. Rev. A 46, 3250–3254 (1992)
8. Godivier, X., Chapeau-Blondeau, F.: Stochastic resonance in the information capacity of a nonlinear dynamic system. Int. J. Bifurcation and Chaos 8, 581–590 (1998)
9. Duan, F., Xu, B.: Parameter-induced stochastic resonance and baseband binary PAM signals transmission over an AWGN channel. Int. J. Bifurcation and Chaos 13, 411 (2003)
10. Sun, S., Kwong, S.: Stochastic resonance signal processor: principle, capacity analysis and method. Int. J. Bifurcation and Chaos 17, 631–639 (2007)
11. Moss, F., Pierson, D., O'Gorman, D.: Stochastic resonance: Tutorial and update. Int. J. Bifurcation and Chaos 4, 1383–1398 (1994)
12. Xu, B., Duan, F., Chapeau-Blondeau, F.: Comparison of aperiodic stochastic resonance in a bistable system realized by adding noise and by tuning system parameters. Physical Review E 69, 061110, 1–8, (2004)
13. Gammaitoni, L., Hanggi, P., Jung, P., Marchesoni, F.: Stochastic resonance. Rev. Mod. Phys. 70, 223–287 (1998)
14. Mitaim, S., Kosko, B.: Adaptive stochastic resonance. Proc. IEEE. 86, 2152–2183 (1998)
15. Cox, I., Miller, M.L., McKellips, A.L.: Watermarking as communications with side information. Proc. IEEE 87, 1127–1141 (1999)
16. Jia, Y., Zheng, X., Hu, X., Li, J.: Effects of colored noise on stochastic resonance in a bistable system subject to multiplicative and additive noise. Phys. Rev. E 63 031107(2001)
17. Xu, B., Li, J., Duan, F.: Effects of colored noise on multi-frequency signal processing via stochastic resonance with tuning system parameters. Chaos, Solitons & Fractals 16, 93–106 (2003)