Secure Key Generation for Static Visual Watermarking by Machine Learning in Intelligent Systems and Services

Kensuke Naoe, Keio University, Japan

Hideyasu Sasaki, Ritsumeikan University, Japan

Yoshiyasu Takefuji, Keio University, Japan

ABSTRACT

The Service-Oriented Architecture (SOA) demands supportive technologies and new requirements for mobile collaboration across multiple platforms. One of its representative solutions is intelligent information security of enterprise resources for collaboration systems and services. Digital watermarking became a key technology for protecting copyrights. In this article, the authors propose a method of key generation scheme for static visual digital watermarking by using machine learning technology, neural network as its exemplary approach for machine learning method. The proposed method is to provide intelligent mobile collaboration with secure data transactions using machine learning approaches, herein neural network approach as an exemplary technology. First, the proposed method of key generation is to extract certain type of bit patterns in the forms of visual features out of visual objects or data as training data set for machine learning of digital watermark. Second, the proposed method of watermark extraction is processed by presenting visual features of the target visual image into extraction key or herein is a classifier generated in advance by the training approach of machine learning technology. Third, the training approach is to generate the extraction key, which is conditioned to generate watermark signal patterns, only if proper visual features are presented to the classifier. In the proposed method, this classifier which is generated by the machine learning process is used as watermark extraction key. The proposed method is to contribute to secure visual information hiding without losing any detailed data of visual objects or any additional resources of hiding visual objects as molds to embed hidden visual objects. In the experiments, they have shown that our proposed method is robust to high pass filtering and JPEG compression. The proposed method is limited in its applications on the positions of the feature sub-blocks, especially on geometric attacks like shrinking or rotation of the image.

Keywords: *Copyright Protection, Digital Watermarking, Key Generation, Machine Learning, Neural Network*

DOI: 10.4018/jssoe.2010092103

INTRODUCTION

In this article, we propose a method of key generation scheme (Figure 1) for static visual digital watermarking (Figure 2) by using machine learning technology, neural network as its exemplary approach for machine learning method.

The proposed method is to provide intelligent mobile collaboration with secure data transactions using machine learning approaches, herein neural network approach as an exemplary technology. First, the proposed method of key generation is to extract certain type of bit patterns in the forms of visual features out of visual objects or data as training data set for machine learning of digital watermark. Second, the proposed method of watermark extraction is processed by presenting visual features of the target visual image into extraction key or herein is a classifier generated in advance by the training approach of machine learning technology. Third, the training approach is to generate the extraction key which is conditioned to generate watermark signal patterns only if proper visual features are presented to the classifier. In our proposed method, this classifier which is generated by the machine learning process is used as watermark extraction key.

The proposed method is to contribute to secure visual digital watermarking without losing any detailed data of visual objects or any additional resources of hiding visual objects as molds to embed hidden visual objects. The proposed method has used neural network for its training approach not limited but open in its applications to other machine learning approaches including fuzzy, Bayesian network and others. In this article, the target content is a static visual data which are constructed with

Figure 1. Key generation scheme in embedding procedure

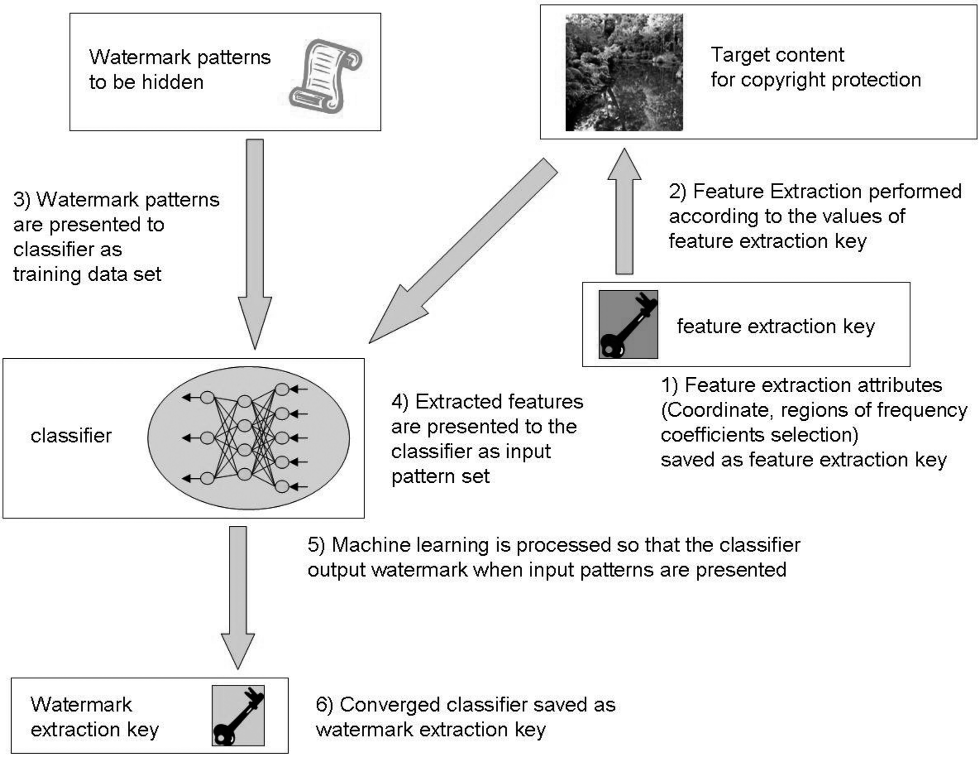
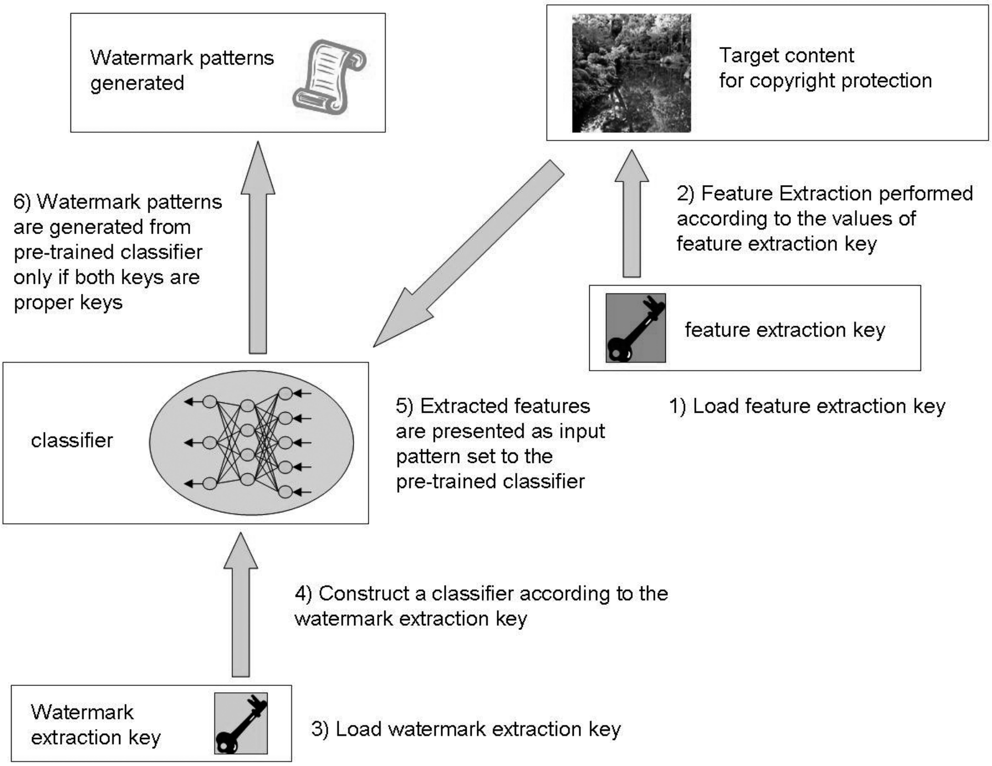


Figure 2. Watermark extraction scheme in extraction procedure


Watermark patterns generated
Target content for copyright protection
6) Watermark patterns are generated from pre-trained classifier only if both keys are proper keys
2) Feature Extraction performed according to the values of feature extraction key
feature extraction key
classifier
5) Extracted features are presented as input pattern set to the pre-trained classifier
1) Load feature extraction key
4) Construct a classifier according to the watermark extraction key
Watermark extraction key
3) Load watermark extraction key

discrete data set and we have demonstrated the feasibility of solving this problem by using neural network model. We would enhance our method by using those other approaches, such as fuzzy for dynamic visual data like video stream data and Bayesian network for continuous data structures. This article is different from the previous work by Ando et al. (Ando, R. & Takefuji, Y., 2003) in terms of embedding size where this article does not embed any information to the target content and also implies that the machine learning algorithm is not limited only to the neural network model as proposed in our previous work (Naoe, K. & Takefuji, Y., 2008).

Issues and Definitions

The Service-Oriented Architecture (SOA) demands supportive technologies and new requirements for mobile collaboration across multiple platforms. One of its representative solutions is intelligent information security of enterprise resources for collaboration systems and services (Chiu, & Leung, 2005; Kafeza, Chiu, Cheung, & Kafeza, 2004).

Digital watermarking became a key technology for protecting copyrights. Digital watermarking protects unauthorized change of the contents and assures legal user for its copyright. Meanwhile, steganography conceals a hidden messages to a content but the existence of a message is kept secret (Artz, 2001). For the purpose of digital watermarking, content should not be encrypted or scrambled. Digital watermark is often embedded imperceptibly to human receptors to avoid contaminating the content and not to distract the content from the original expression. For imperceptible images, the human visual system (HVS) model is often used (Delaigle, Vleeschouwer, & Macq, 1998). Perceptible watermark are sometimes used, but it limits the use of the images. Therefore, main concern in this research area has focused on imperceptible watermark.

In general, the robustness and imperceptibility of digital watermarking take trade-off relationships. Embedding information must be placed in perceptually significant signal for it to be robust against removal attacks, but it is known that modifying these regions will lead to perceptual degradation of signal (Cox, Kilian, Leighton, & Shamoon, 1996). Therefore, an ideal digital watermarking algorithm should have minimal amount of embedding information. There is also a difficulty to fulfill the non-repudiation requirement by using a robust watermarking scheme alone. To address this problem, there must be a solid distribution protocol and secure infrastructure framework to put in practice. A contribution from previous work by Cheung, Chiu, & Ho, (2008) which proposed a distribution protocol and secure system framework addressed this problem and its watermarking algorithm can be replaced with another algorithms as long as the watermark can be inserted in the encrypted domain where digital contents are encrypted by public keys. Our proposed method generates pair of extraction keys where one of them can act as public key and also does not alter the target content. Therefore, our proposed method has an ability to collaborate with this previous work.

Backgrounds and Related Works

Here, we overview the backgrounds and related works in the research areas of digital watermarking and machine learning, neural network as an exemplary model. We discuss digital watermark, steganography, cryptography and machine learning.

The emergence of the Internet with rapid progress of information technologies, digital contents are commonly seen in our daily life distributed through the network. Due to the characteristics of digital contents are easy to make an exact copy and to alter the content itself, illegal distribution and copying of digital contents has become main concerns for authors, publishers and legitimate owners of the contents (Sasaki, 2007). There are several ways to protect digital content. One can protect the content by cryptographic-based schemes (Rothe, 2002), but this avoids free distribution and circulation of the content through the network because the decryption key must be shared, which most of the time not desirable for authors of the contents. To address this problem, the researches have developed "Public Key Infrastructure" (Adams, & Lloyd, 1999; Maurer, 1996) and "Secure Key Sharing" (Law, Menezes, Qu, Solinas, & Vanstone, 2003; Eschenauer, & Gligor, 2002). Moreover, digital watermarking technologies are noticed to substitute or to complement the conventional cryptographic schemes (Cox, Doerr, & Furon, 2006). Furthermore, in the era of mobile devices and technologies, conventional cryptographic schemes are highly power consuming and time consuming, which are not suitable for these types of devices (Prasithsangaree, & Krishnamurthy, 2003). Hence, necessary information must be obtainable with less calculation cost and power consumption.

Watermarking technique is one technique of an information hiding techniques. Information hiding provides a reliable communication by embedding secret code into a content for the various purposes: intellectual property protection, content authentication, fingerprinting, covert communications, content tracking, end-to-end privacy, secure distribution, etc. (Hung, Chiu, Fung, Cheung, Wong, Choi, Kafeza, Kwok, Pun, & Cheng, 2007; Wolf, Steinebach, & Diener, 2007; Kwok, Cheung, Wong, Tsang, Lui, & Tam, 2003). The researches in information hiding has a history (Kahn, 1996) and namely the researches in digital watermarking and steganography have been active (Katzenbeisser, & Fabien, 2000). Both are very similar but their applications are different (Cox, I., Miller, Bloom, Fridrich, & Kalker, 2007).

There are many digital watermarking and steganography algorithms, though it is difficult to use one algorithm together with another, because each other obstruct the embedded information and causing one to destroy another. Because our proposed method does not damage the target content, it has an ability to collaborate with another algorithm to strengthen the security of information hiding method. This characteristic is useful where one already manage digital rights using one watermarking algorithm or controls the file integrity using hash functions. If one wishes to strengthen the robustness of watermark using another algorithm, one must examine and assure that applying the algorithm will not affect embedded watermarking signals in advance. Furthermore, applying another watermarking algorithm will alter the fingerprint of the content managed by hash functions and forces administrator to recalculate a new hash values after applying new watermarking algorithm, which most of the time, result in higher calculation cost and time. Because our proposed method does not affect the target content at all, one can apply new watermark seamlessly without altering the fingerprint using the proposed method.

Machine learning and statistical analysis are very effective approach to approximate unknown class separating functions and to find potentially useful patterns in the data set (Bishop, 2006). Basically, machine learning algorithm uses training set to adjust the parameter of the model adaptively. When target vector or teacher patterns are presented for an training set, training or learning process is performed to condition the model to output a proper patterns which is same or close to as the target vector. Some of the noticeable models in machine learning are Bayesian network (Bernardo, & Smith, 2001; Jensen, 1996), fuzzy logic (Klir, & Yuan, 1995; Klir, & Yuan, 1996), support vector machine (Cortes, & Vapnik, 1995; Burges, 1998), and neural network (Kohonen, 1988; Bishop, 1995). The proposed method uses machine learning approach to generate watermark extraction keys and herein uses neural network model in this article.

The basic principle of neural network is that neuron, or most atomic unit of neural network, only has a simple function of input and output signal but is capable of complex function when these neurons are organically connected forming a network. Mathematical models of these networks are called neural networks, and processing these artificial neural networks on computers is called neural computing. Neural computing is a classic research topic and neural networks are known to be capable of solving various kinds of problems by changing the characteristics of neuron, synaptic linking and structure of the network.

The proposed method uses a multi-layered perceptron model for neural network model. Multi-layered perceptron basically has a synaptic link structure in neurons between the layers but no synaptic link between the neurons within the layer itself (Rosenblatt, 1958). Two layer perceptron can approximate data linearly but cannot classify data nonlinearly. Multilayered perceptron with more than three layers are known to have an approximation ability of a nonlinear function if properly trained, but then there was no ideal learning method for this kind of training. This model became popular when a training method called back propagation learning was introduced (Rumelhart, & McClelland, 1986). Other neural network models are RBF network model (Broomhead, & Lowe, 1988), pulsed neural network model (Johnson, 1994), Elman network model (Elman, 1990) and many others.

Structure

This article is organized as follows. Section 1 gives a background and some related works of digital watermarking as one application of information hiding technique, and general overview of machine learning technique especially focusing on neural network model. Section 2 explains our methodology for generating extraction key sets for extracting watermark patterns. Section 3 gives an experimental conditions and results of our experiments to examine the robustness of our watermarking method. Section 4 and 5 will give some discussions, conclusions and some future work to be addressed.

METHODOLOGY

In this section, we explain how our proposed method generates extraction keys from the target content and how to retrieve watermark information from the target content using the extraction keys generated in the watermark embedding procedure. With our method, the use of machine learning approach is the key methodology. Here, adjustment of neural network weights to output desired hidden watermark pattern by supervised learning of the neural network is performed. This conditioned neural network works as a classifier or watermark extraction key to recognize a hidden watermark pattern from the content. Therefore, extractor uses this neural network weights as extraction keys for extracting the hidden watermark patterns. Extractor must have proper visual features used for generation of extraction key and proper network weights of neural network which is the extraction key in our proposed method. Considering the difficulties for secret key transportation, this method should be applied in situations where the hider and the extractor is a same person or to use certification authorities to assure the integrity of the key as Cheung, Chiu, and Ho, (2008) proposed.

Our Proposed Method

Here, we explain the procedures for generation of extraction keys and extraction of watermark patterns. Generation of extraction key is included in the embedding process and extraction of watermark patterns is included in the extraction process. First, we simply demonstrate the procedures which must be taken for embedding process and extraction process.

Embedding process consists of following procedures:

1. Feature extraction by frequency transformation of target image
2. Selection of the feature regions according to the feature extraction attributes and saved as feature extraction key
3. Prepare watermark patterns to be embedded
4. Generation of extraction key to output watermark patterns by back propagation learning of neural network using feature extraction key
5. Save the generated neural network classifier as watermark extraction key

First, frequency transformation of the image is performed. There are several methods to transform an image to frequency domain, such as Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). In our method, DCT is used to be robust against some image compression. Compression using DCT is known to be very effective, and is employed in MPEG format and JPEG format. DCT divides an image into N*N pixel small blocks, here we call sub-blocks. General DCT generates 8*8 pixel size sub-blocks. We must select certain amount of sub-blocks from frequency domain of target image and DCT coefficients are chosen diagonally from those sub-blocks. The same amount of unique sub-blocks must be chosen from the target image as number of classification patterns, which in this method is the watermark pattern. Sufficient number of neural networks must be prepared, which will be the number of binary digits to satisfy the watermark patterns. In case for choosing 32 watermark patterns, five networks are enough to represent 32 different classification values because five binary digits are sufficient to distinguish for 32 patterns. Learning of all networks is repeated until the output value of neural network satisfies a certain learning threshold value. After all network weights are generated, the coordinates of sub-blocks and the values of network weights are saved. Extractor will use this information to extract hidden codes in the extraction process.

Extraction process consists of following procedures:

1. Obtain feature extraction key and watermark extraction key

2. Feature extraction of target image using the feature extraction key
3. Construct a neural network model using watermark extraction key
4. Observe the output patterns from neural network using both keys

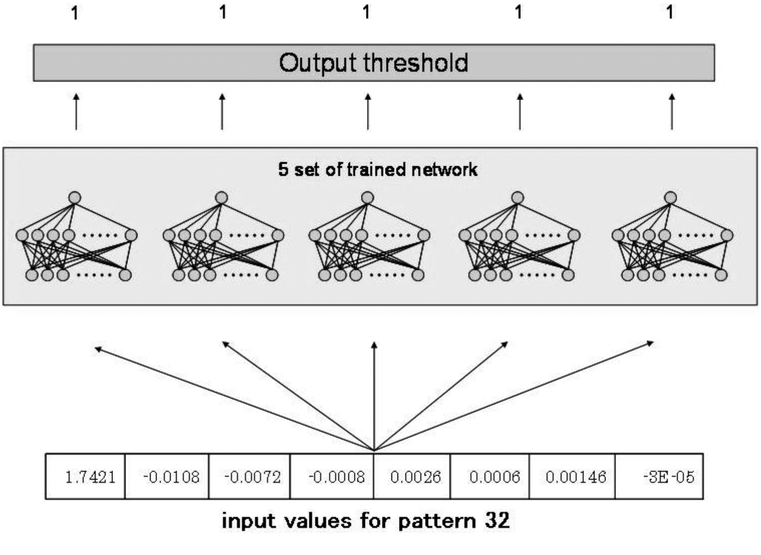
Extractor must receive both proper feature extraction key and watermark extraction key to obtain proper watermark pattern. Only by having the proper feature extraction key will lead the user to the proper input patterns to the neural network. By knowing proper watermark extraction key, extractor can induce the structure of the neural network and only proper neural network is able to output the proper hidden watermark patterns. After constructing the neural network, extractor examines the output value from the network with the input values induced from the feature extraction key. This procedure is shown in Figure 3. Each network output either 1 or 0 with the aid of threshold for output unit.

Furthermore, here we explain each procedure more concretely. For key generation procedure, frequency transformation of the target content is processed for visual feature extraction of target content. Frequency transformation can be anything like DCT, DFT or DWT, here we use DCT because of its simplicity and its structure of feature vectors of a transformed image in frequency domain. This frequency transformation is done after converting the target content to YCbCr color domain. Basically, the transformation from RGB color signal to YCbCr signal is used to separate a luma signal Y and two chroma components Cb and Cr and mainly used for JPEG compression and color video signals. In our proposed method, instead of using RGB color space directly, YCbCr color space is used to make use of human visual system characteristic. The conversion from RGB to YCbCr is calculated using the following equation:

$$Y = 0.299R + 0.587G + 0.114B$$
$$Cb = -0.169R - 0.322G + 0.500B$$
$$Cr = 0.500 - 0.419G - 0.081B$$

Then, training of neural network is processed. For the training, one must decide the structure of neural network. The amount of units for input layer is decided by the number of pixels selected from target content data. In

Figure 3. System structure for extraction



our proposed method, the feature values are diagonal coefficient values from frequency transformed selected feature sub-blocks. For better approximation, one bias neuron is added for input layer.

The neural network is trained to output a value of 1 or 0 as an output signal. In our proposed method, one network represents one binary digit for corresponding watermarking patterns. The adequate amount of neurons in the hidden layer, for back propagation learning in general, is not known. So the number of neurons in hidden layer will be taken at will. In our proposed method, ten hidden units are used. For better approximation, one bias neuron is introduced for hidden layer as well. Once network weights are generated to certain values, the proposed method use these values and the coordinates of selected feature sub-blocks as feature extraction key and watermark extraction key. These keys must be shared among the hider and the extractor in order to extract proper hidden watermark patterns from the contents.

Now, we demonstrate an overview of multilayered perceptron model. In multilayered perceptron model, signals given to the input layer will propagate forwardly according to synaptic weight of neurons through the layers and finally reaches to the output layer as shown in Figure 4.

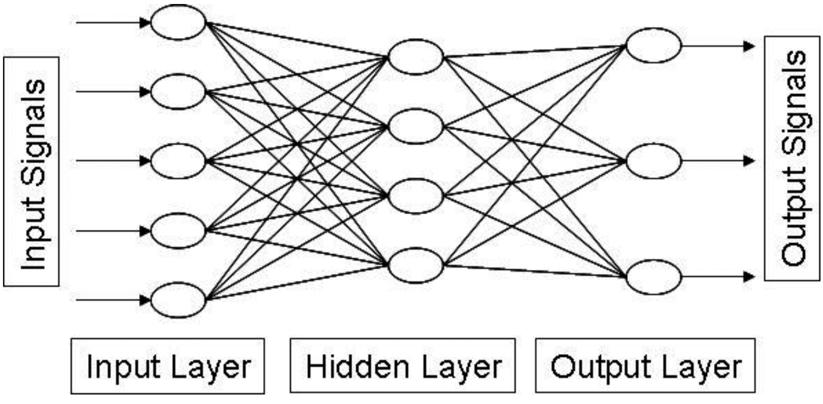
Signal that is being put to the neuron is converted to a certain value using a function and outputs this value as output signal. Normally sigmoid function is used for this model and this function is expressed as follows:

$$f(x) = \frac{1}{1 + e^{-x}}$$

Each synaptic link has a network weight. The network weight from unit *i* to unit *j* is expressed as W_{ij} and the output value for unit *i* is expressed as O_i. The output value for the unit is determined by the network weight and the input signal from the former layer. Consequently, to change the output value to a desired value for a certain input value patterns, the adjustment of these network weights must be conducted and this process is called learning. In our proposed method, we use back propagation learning as the learning method.

Back propagation learning is the process of adjusting the network weights to output a value close to the values of the teacher signal values which are presented to the neural network. Back propagation learning is a supervised learning (Rumelhart, & McClelland, 1986). This method tries to lower the difference between the presented teacher signal and the output signal dispatched for certain input value patterns by adjusting the network weight. The difference between the teacher signal values and the actual output signals are called as error and often ex-

Figure 4. Multi-layered perceptron model



pressed as δ. The error will propagate backward to the lower layer and network weights are adjusted using these values. When teacher signal t_k is given to the unit *k* of output layer, the error δ_k will be calculated by following function:

$$\delta(x) = (t_k - O_k) \cdot f'(O_k)$$

To calculate the error value δ_j for hidden unit, error value δ_k of the output unit is used. The function to calculate the error value δ_j for hidden unit *j* is as follows:

$$\delta_j = (\sum_k \delta_k w_{jk}) \cdot f'(O_j)$$

After calculating the error values for all units in all layers, then network can change its network weight. The network weight is changed by using following function:

$$\Delta\delta_k w_{jk} = \eta \delta_j O_i$$

η in this function is called learning rate. Learning rate is a constant which normally has a value between 0 and 1 and generally represents the speed of learning process. The lower the learning rate is the more gradual the learning process will be, and the bigger the learning rate is the more acute the learning process will be. Sometimes, this parameter must be tuned for stable learning.

For extraction process, same neural network structure is constructed using the generated watermark extraction key. Only with the proper feature extraction key and watermark extraction key will be able to output the corresponding watermark patterns. These embedding and extraction procedure are shown in diagram in Figure 5 and 6.

We define necessary parameters for embedding and extracting. For embedding, there are two parameters to decide on. First is the number of watermarking patterns to embed. More the number of watermarking patterns, the more data can be embedded, but introducing large number of watermarking patterns will result in high calculation cost. Second parameter is the coordinate of the sub-blocks. Coordinates will determine the input patterns to the neural network for the embedding and extraction process. For extracting, there are two keys to be shared, that is the watermark extraction key and feature extraction key, between the embedding and extracting users. Former is the neural network weights created in the embedding process. Latter is the coordinates of feature sub-blocks. Only with the presence of the proper keys are able to generate proper watermark patterns as shown in Figure 5 and 6.

EXPERIMENTS

In this experiment, we will examine if we can retrieve a watermark patterns from both original image and graphically changed image. For the latter, we choose high pass filter and JPEG compressed image as the alteration method.

We used TIFF format Lenna image, which is 512*512 pixels in size, as target content data. Original and high pass filtered Lenna image is shown in Figure 7. We embedded 32 different watermark patterns as hidden signals. That is, hidden signals are [00000] for pattern 1, [00001] for pattern 2, ... [11111] for pattern 32. Five neural networks are used for classification of 32 patterns. Each network output value represents the binary digits of watermark patterns. In this experiment, network 1 represents the largest binary bit and network 5 represents the smallest binary bit. The number of hidden layer units is set to 11 including one bias unit. Learning process is repeated until the output values generate to a learning threshold of 0.1. Also, the output threshold in this experiment is set to 0.5. This means that if output value is larger than 0.5, output signal is set to 1, and if it is less than 0.5, output signal is set to 0. Again, this neural network structure is shown in Figure 3.

With the threshold value of 0.5, the proposed method was able to extract proper signals

Figure 5. Embedding procedure

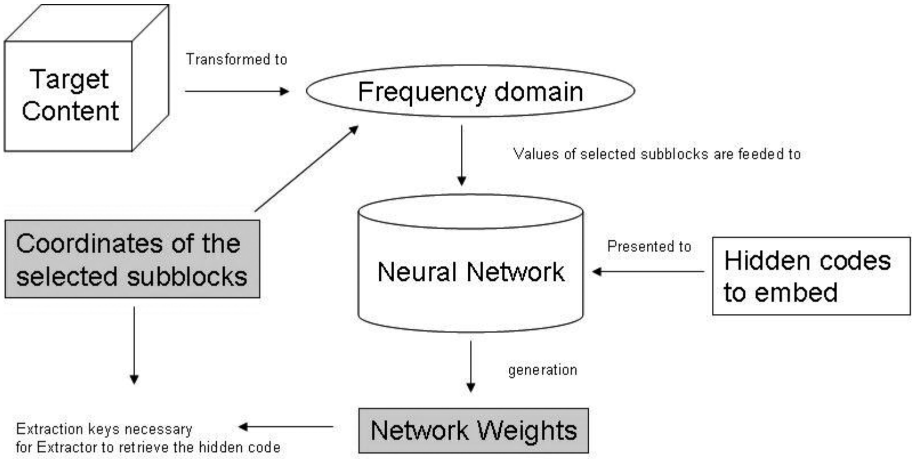
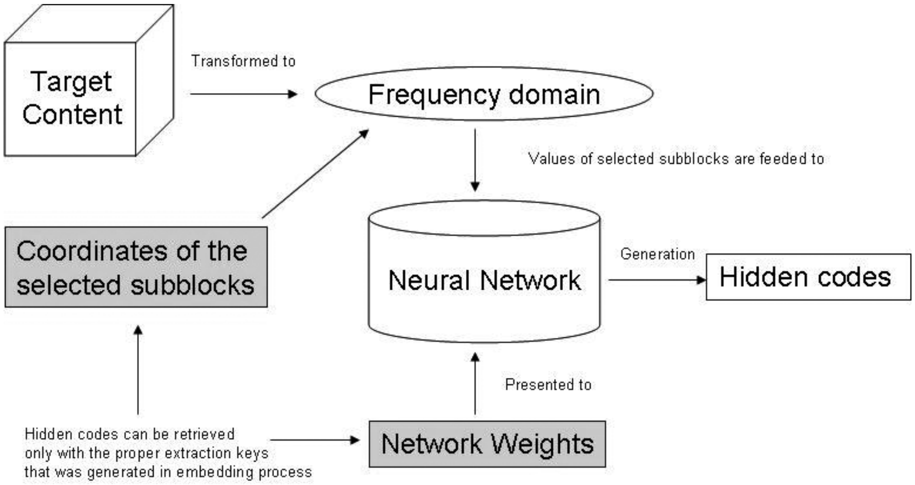

Target Content
Transformed to
Frequency domain
Values of selected subblocks are feeded to
Coordinates of the selected subblocks
Neural Network
Presented to
Hidden codes to embed
generation
Extraction keys necessary for Extractor to retrieve the hidden code
Network Weights

Figure 6. Extracting procedure


Target Content
Transformed to
Frequency domain
Values of selected subblocks are feeded to
Coordinates of the selected subblocks
Neural Network
Generation
Hidden codes
Presented to
Hidden codes can be retrieved only with the proper extraction keys that was generated in embedding process
Network Weights

for proper patterns. For example, output signals in binary bit pattern for pattern 1 is [00000], output signals for pattern 11 is [01010] and etc. The results of this experiment, output signals for original image, JPEG compressed image and high pass filtered image are shown in Figure 8, 9, 10, 11 and 12 for each neural network.

The output signals retrieved from high pass filtered image are shown to be slightly different compared to the output signals for the original image and also the output signals retrieved from JPEG compressed image is damaged more than of filtered image, but with the same output threshold of 0.5, we were able to

Figure 7. Original Lenna image and filtered image

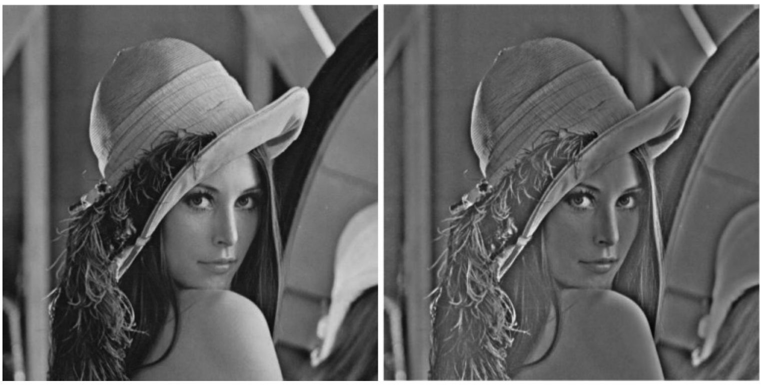
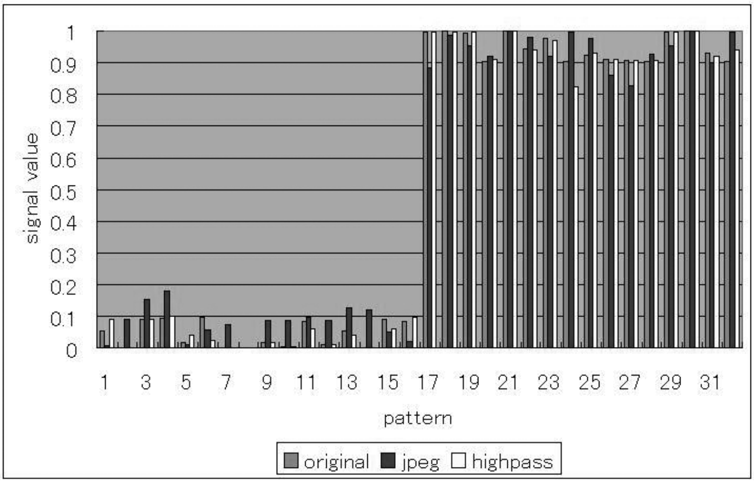


Figure 8. Signal values for network 1



retrieve same hidden watermark patterns for all 32 set from high pass filtered image and JPEG compressed image. These results showed the robustness to high pass filtering and JPEG compression alteration.

DISCUSSIONS

We would like to discuss the contributions and limitations on our propose method in its applications to intelligent information security of enterprise resources for collaboration systems and services.

Our proposed method has not embedded any additional information to the target images of the experiments. Also, in the experiment we have shown the possibility of a robust digital watermarking scheme without embedding any data into the target content. With the perspective of information security, this method has the possibility for the applications for both digital watermark and steganography.

Our proposed method is logically limited in its use for digital watermarking in the generation time of feature extraction key and watermark extraction key: the former feature extraction is a

Figure 9. Signal values for network 2

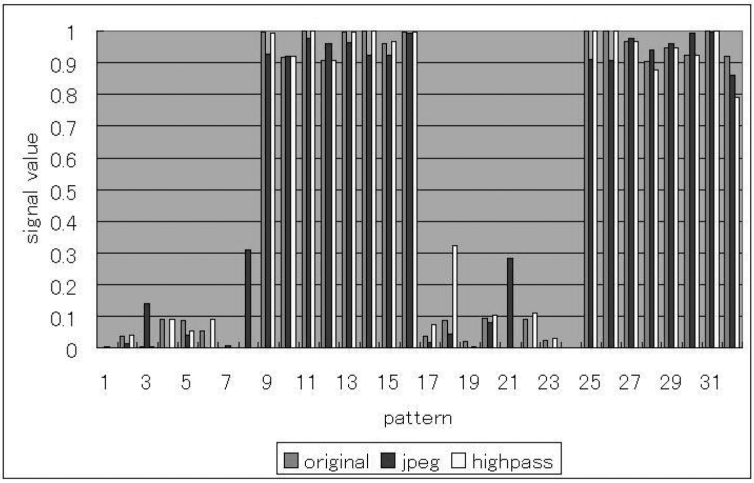
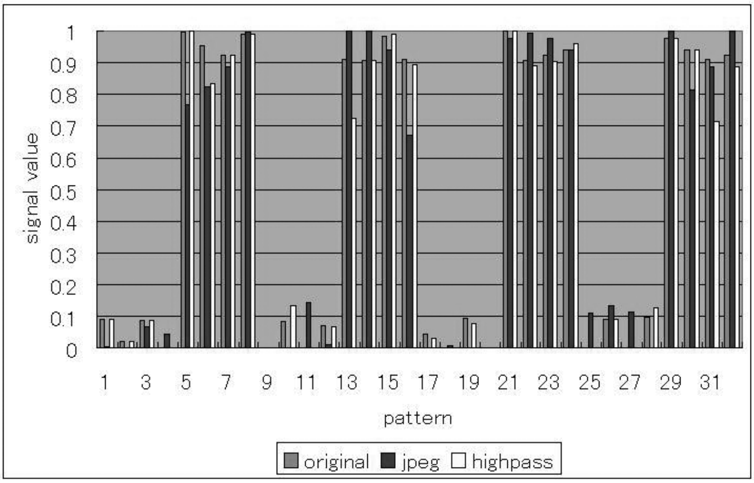


Figure 10. Signal values for network 3



preprocess before the neural network learning; And, the latter processing time for watermark extraction is less than a second in practice of experiment. The generation of watermark extraction key is a learning process of neural network which is equivalent to a non-linear approximation of patterns. The elapsed time for the process of watermark extraction key is measured about a few minutes, although the generation of watermark patterns using watermark extraction key took only time of less than a second. The generation speed of key is not influential, and only the speed for generation of watermark pattern is important.

Meanwhile, our proposed watermark extraction method relies on the position of the feature sub-blocks. It is weak to geometric attacks like shrinking, expanding, and rotation of the image. This problem will be considered as future works of our proposed method.

Figure 11. Signal values for network 4

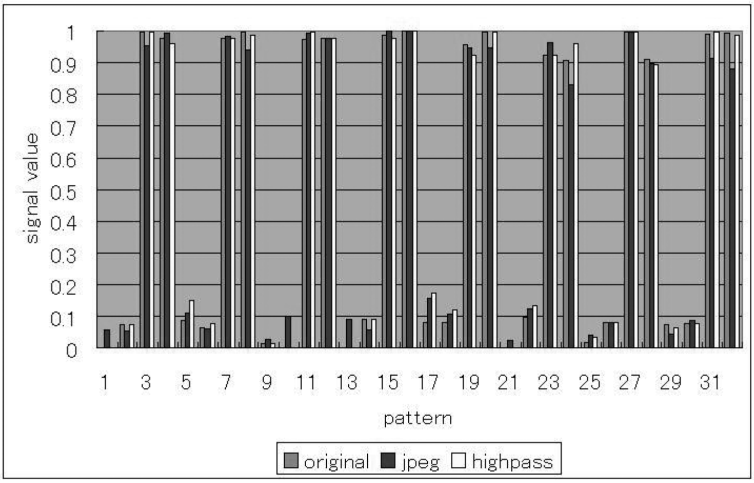
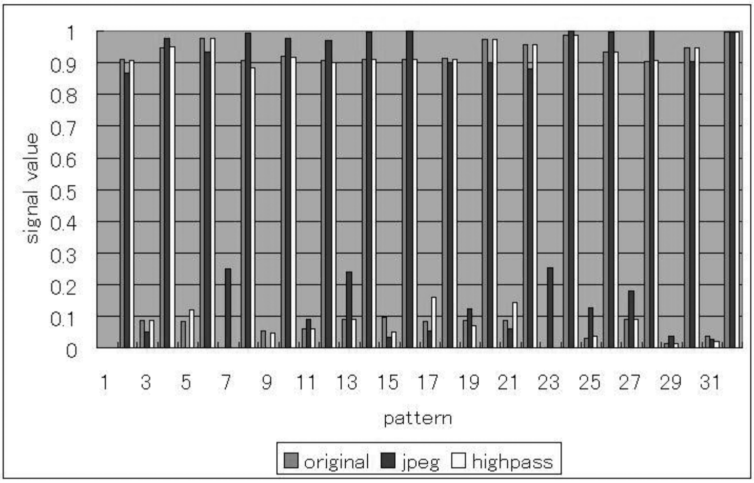


Figure 12. Signal values for network 5



CONCLUSION AND FUTURE WORK

The SOA demands supportive technologies and new requirements of which representative solution is intelligent information security of enterprise resources for collaboration systems and services. Digital watermarking became a key technology for protecting copyrights. In this article, we have proposed a key generation method for static visual watermarking by machine learning, neural network as an exemplary method. Key generation does not involve any embedding of data into target content which it means that it is a damage-less watermarking method. This characteristic is effective when user must not damage the content but must conceal a secret code into target content.

The proposed method uses multi-layered neural network model for classifying the input visual feature patterns to corresponding hidden watermark patterns. For input visual feature values, we used DCT coefficients on YCbCr domain. The proposed method does not limit to DCT as frequency transformation method only, one can use DFT and DWT otherwise. Also, machine learning method can be replaced with others such as Bayesian network and fuzzy.

In the experiment, we have shown that our proposed method is robust to high pass filtering and JPEG compression. Also, because our propose method does not alter the target content, it is applicable to steganography. Meanwhile, because the proposed method relies on the position of the feature sub-blocks, it is weak to geometric attacks like shrinking or rotation of the image. This must take into consideration for future work.

ACKNOWLEDGMENT

Kensuke Naoe is supported financially in part by the Mori Fund of Keio University, on this study. Hideyasu Sasaki is supported financially in part by the Grant-in-Aid for Scientific Research ("KAKENHI") of the Japanese Government: No. 21,700,281 (FY 2009-2011), on this study.

REFERENCES

Adams, C., & Lloyd, S. (1999). *Understanding public-key infrastructure: concepts, standards, and deployment considerations.* Macmillan Technical Publishing.

Ando, R., & Takefuji, Y. (2003, January). Location-driven watermark extraction using supervised learning on frequency domain. *WSEAS Transactions On Computers*, *2*(1), 163–169.

Artz, D. (2001, May/June). Digital steganography: hiding data within data. *IEEE Internet Computing*, *5*(3), 75–80. doi:10.1109/4236.935180

Bernardo, J., & Smith, A. (2001). Bayesian theory. *Measurement Science & Technology*, *12*, 221–222.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press, USA.

Bishop, C. M. (2006). *Pattern recognition and machine learning (information science and statistics)* (New edition ed.). Springer-Verlag.

Broomhead, D. S., & Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, *2*, 321–355.

Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, *2*(2), 121–167. doi:10.1023/A:1009715923555

Cheung, S.-C., Chiu, D. K. W., & Ho, C. (2008). The use of digital watermarking for intelligence multimedia document distribution. *Journal of Theoretical and Applied Electronic Commerce Research*, *3*(3), 103–118. doi:10.4067/S0718-18762008000200008

Chiu, D., & Leung, H. (2005). Towards ubiquitous tourist service coordination and integration: a multi-agent and semantic web approach. In *Proceedings of the 7th international conference on Electronic commerce* (pp. 574–581).

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Cox, I., Doerr, G., & Furon, T. (2006). Watermarking is not cryptography. *Lecture Notes in Computer Science*, *4283*, 1–15. doi:10.1007/11922841_1

Cox, I., Kilian, J., Leighton, T., & Shamoon, T. (1996). Secure spread spectrum watermarking for images, audio and video. *Image Processing, 1996. Proceedings., International Conference on, 3*.

Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2007). *Digital watermarking and steganography* (2nd ed.). Morgan Kaufmann.

Delaigle, J. F., Vleeschouwer, C. D., & Macq, B. (1998). Watermarking algorithm based on a human visual model. *Signal Processing*, *66*(3), 319–335. doi:10.1016/S0165-1684(98)00013-9

Elman, J. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Eschenauer, L., & Gligor, V. (2002). A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on computer and communications security* (pp. 41–47).

Hung, P. C., Chiu, D. K., Fung, W. W., Cheung, W. K., Wong, R., & Choi, S. P. (2007). End-to-end privacy control in service outsourcing of human intensive processes: A multi-layered web service integration approach. *Information Systems Frontiers*, *9*(1), 85–101. doi:10.1007/s10796-006-9019-y

Jensen, F. (1996). *Introduction to Bayesian networks*. Springer-Verlag New York, Inc. Secaucus, NJ, USA.

Johnson, J. (1994). Pulse-coupled neural nets: translation, rotation, scale, distortion, and intensity signal invariance for images. *Applied Optics*, *33*, 6239–6253. doi:10.1364/AO.33.006239

Kafeza, E., Chiu, D., Cheung, S., & Kafeza, M. (2004). Alerts in mobile healthcare applications: requirements and pilot study. *IEEE Transactions on Information Technology in Biomedicine*, *8*(2), 173–181. doi:10.1109/TITB.2004.828888

Kahn, D. (1996). The history of steganography. In *Proceedings of the First International Workshop on Information Hiding* (pp. 1–5). London, UK: Springer-Verlag.

Katzenbeisser, S., & Fabien, A. P. (Eds.). (2000). *Information hiding techniques for steganography and digital watermarking*. Artech House Publishers.

Klir, G., & Yuan, B. (1995). *Fuzzy sets and fuzzy logic: theory and applications*. Prentice Hall Upper Saddle River, NJ.

Klir, G. J., & Yuan, B. (Eds.). (1996). Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh. World Scientific Publishing.

Kohonen, T. (1988). An introduction to neural computing. *Neural Networks*, *1*(1), 3–16. doi:10.1016/0893-6080(88)90020-2

Kwok, S., Cheung, S., Wong, K., Tsang, K., Lui, S., & Tam, K. (2003). Integration of digital rights management into the Internet Open Trading Protocol. *Decision Support Systems*, *34*(4), 413–425. doi:10.1016/S0167-9236(02)00067-2

Law, L., Menezes, A., Qu, M., Solinas, J., & Vanstone, S. (2003). An efficient protocol for authenticated key agreement. *Designs, Codes and Cryptography*, *28*(2), 119–134. doi:10.1023/A:1022595222606

Maurer, U. (1996). Modelling a public-key infrastructure. *Lecture Notes in Computer Science*, 325–350.

Naoe, K., & Takefuji, Y. (2008, September). Damageless Information Hiding using Neural Network on YCbCr Domain. *International Journal of Computer Sciences and Network Security*, *8*(9), 81–86.

Prasithsangaree, P., & Krishnamurthy, P. (2003). Analysis of energy consumption of RC4 and AES algorithms in wireless LANs. In *Proceeding of IEEE Global Telecommunications Conference* (pp. 1445-1449).

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. doi:10.1037/h0042519

Rothe, J. (2002). Some facets of complexity theory and cryptography: A five-lecture tutorial. *ACM Computing Surveys*, *34*(4), 504–549. doi:10.1145/592642.592646

Rumelhart, D., & McClelland, J. (1986). *Parallel distributed processing: explorations in the microstructure of ognition, vol. 1: foundations*. MIT Press Cambridge, MA, USA.

Sasaki, H. (Ed.). (2007). *Intellectual property protection for multimedia information technology*. IGI Global.

Wolf, P., Steinebach, M., & Diener, K. (2007). Complementing DRM with digital watermarking: mark, search, retrieve. *Online Information Review*, *31*(1), 10–21. doi:10.1108/14684520710731001

Kensuke Naoe, a PhD candidate, graduated from the Faculty of Environment and Information Studies of Keio University in 2002. He received the master's degree from the Graduate School of Media and Governance at Keio University in 2004. His major was artificial neural network and information security. His interested research areas are digital information hiding, machine learning, network intrusion detection and malware detection.

Hideyasu Sasaki, PhD & Esq. is the editor-in-chief of the International Journal of Organizational and Collective Intelligence *(IJOCI). Professor Sasaki is a graduate of the University of Tokyo,*

B.A. and LL.B., in 1992, 1994, received an LL.M., from the University of Chicago Law School in 1999, an MS and a PhD in cybernetic knowledge engineering with honors from Keio University in 2001, 2003, respectively. He is an associate professor at Department of Information Science and Engineering, Ritsumeikan University, Kyoto, Japan. He was an assistant professor at Keio University from 2003 to 2005. His research interests include decision science and intelligence computing, especially mathematical modeling of decision making under time constraint. Dr. Sasaki has also experienced lawyering and litigations as attorney-at-law in N.Y., U.S.A. He is an associate editor at the International Journal of Systems and Service-Oriented Engineering *(IJSSOE) and a reviewer for the* Journal of Information Sciences, Elsevier*, in 2008, 2009 and* ACM Transactions on Knowledge Discovery from Data *(KDD) in 2008. He is active in program committees of the ACM International Conference on Management of Emergent Digital EcoSystems (MEDES), ICIW/SLAECE, SOMK, SoCPaR, ICADL, LAoIS, etc.*

Yoshiyasu Takefuji is a tenured professor at the Faculty of Environment and Information Studies of Keio University, since April 1992 and was on tenured faculty of electrical engineering at Case Western Reserve University, since 1988. Before joining Case, he taught at the University of South Florida and the University of South Carolina. He received his BS (1978), MS (1980), and PhD (1983) from electrical engineering from Keio University. His research interests focus on neural computing, security, internet gadgets, and nonlinear behaviors. He received the National Science Foundation/Research Initiation Award in 1989, the distinct service award from IEEE Trans. on Neural Networks in 1992, and has been an NSF advisory panelist. He has received the best paper award from AIRTC in 1998 and a special research award from the US air force office of scientific research in 2003. He is currently an associate editor of International Journal of Multimedia Tools and Applications*, editor of* International Journal on Computational Intelligence and Applications*, and editor of* International Journal of Knowledge-based intelligent engineering systems*. He was an editor of the* Journal of Neural Network Computing*, an associate editor of* IEEE Trans. on Neural Networks, Neural/parallel/scientific computations*, and* Neurocomputing*. He has published more than 120 journal papers and more than 100 conference papers.*