

Full Length Research Paper

Digital image watermarking using Learning Vector Quantization and hyperanalytic wavelet packet transform

Hai Tao^{1*}, Jasni Mohamad Zain¹, Mohammad Masroor Ahmed¹ and Ahmed N. Abdalla²

¹Faculty of Computer Systems and Software Engerring, University Malaysia Pahang, Malaysia.

²Faculty of Electrical and Electronic Engineering, University Malaysia Pahang, Malaysia.

Accepted 20 September, 2011

Digital watermarking techniques have been largely applied into copyright protection and authentication of multimedia data. This paper proposes a novel digital watermarking algorithm for digital images by combining the hyperanalytic wavelet packet transform (HWPT) decomposition and Learning Vector Quantization (LVQ) neural network. For inserting watermark, the non-overlapping blocked original images are decomposed according to two-dimensional hyperanalytic quadrantal wavelet packet and the preprocessed watermark image is embedded into the selected coefficients windows with different angles. Subsequently, in order to accomplish watermark strength maximum and to decrease the visual distortion, a competitive learning procedure is applied to train with the set of LVQ training patterns and the trained LVQ is attributed automatically to classify a set of testing patterns while encoding corrected errors. Thus, it maintains the visual quality and meanwhile reduces the error rate by providing maximum possible required information. The simulation results demonstrate the proposed watermarking procedure has remarkable performances in the imperceptibility and robustness to general signal processing operations and some geometric attacks.

Key words: Watermarking, hyperanalytic wavelet packet transform, learning vector quantization.

INTRODUCTION

Today's information driven economy is featured by the tremendous growth of telecommunication networks techniques and the explosion of day-to-day data processing of the immense amount of multimedia data. And protecting proprietary rights becomes much more problematic when digital content can easily be disseminated through communication channels such as the Internet. Consequently, there is an urgent demand for techniques to ensure tamper-resistance and protect the copyright of digital contents as which could easily be disseminated through communication channels such as the Internet. Amongst them, digital watermarking techniques have been applied by embedding some watermark that contains information of a copyright sign or indicates the ownership of the content, is embedded into

the host multimedia product, because that embedded watermark remains imperceptible and indiscernible and it is hard to remove by unauthorized persons even under certain manipulations such as addition of noise, compression, tampering and scaling operations. And then only the authorized recipient of the digital content can extract or detect the watermark from the watermarked products (Cox and Miller, 2002).

Throughout recent years, many approaches have been used to operate watermarking in wavelet domain, primarily because of its proven efficiency and its straight forward implementation for image watermarking (Zhang et al., 2004; Wang and Lin 2004).

Additionally, Zhang et al. (2004) presented an adaptive block-based blind watermarking algorithm. In this algorithm, the coefficients of the detail subbands are modified by using the statistical characteristic of the detailed subbands to embed and blind detect the watermark adaptively.

*Corresponding author. E-mail: buliu0315@gmail.com.

The quantizing super trees are proposed for a wavelet-based watermarking technique in (Wang and Lin, 2004). Each watermark bit is embedded in various frequency bands and the information of the watermark bit is spread throughout large spatial regions. Allowing for better representation of image features, complex wavelet transform (CWT) (Thompson et al., 2007; Bulow and Sommer 2001; Mabtoul et al., 2006) is introduced into digital watermarking methods. For handling the cases of grey scale, color image and video watermarking, (Thompson et al., 2007) applied spread transform embedding into the complex wavelet domain. The combination of the watermark insertion and the human visual system satisfies the imperceptibility requirement of watermarking systems in this method. In addition, spread transform watermarking has emerged as a method of taking the advantages of spread spectrum and quantization based data hiding algorithms, offering improved levels of capacity and robustness. Since the Hyperanalytic Wavelet Transform (HWT) by Sofia and Georgios (2006) is a special case of the Gabor filters with complex coefficients it has the directional advantages of the Gabor filters but requiring less computation the dual-tree complex wavelet transform (DT-CWT). Furthermore, it is better than DWT as it is approximately shift invariance and has good directional selectivities in two dimensions. In Nafornita et al. (2007), it is proposed a watermarking scheme using pixel-wise masking in the HWT domain.

Artificial intelligence approaches for watermark insertion and extraction remain effective watermarking techniques applied to the frequency domain and spatial domain (Fu et al., 2004; Hou and Yen, 2010; Wen et al., 2009). In Fu et al. (2004), a novel watermarking scheme based on the support vector machine has been proposed. For improving the scheme's robustness, the SVM can be fused with traditional watermarking systems and memorize the relationship between the watermarked bit and the host. The detector's robustness can be reached by the good generalization ability of the SVM. In Hou and Yen (2010), the special structure and property of image in multiwavelet domain are applied to design the watermarking algorithm, and a mean value modulation technique is employed to modulate a set of multiwavelet coefficients in approximation sub-bands. The mean value modulation technique can efficiently reduce effects of image distortion when suffering from different attacks. In order to robustly extract watermark, SVMs is used to learn mean value relationship between watermark and coefficients in multiwavelet sub-bands. However, it is important to mention that the previously cited watermarking schemes based on SVM cause overfit because of too much training patterns. In Wen et al. (2009), it presents a blind digital watermarking algorithm based on probabilistic neural network. The host image is decomposed into wavelet domain, then watermark bits embedded in the selected coefficients blocks.

In watermark extraction, the original watermark is retrieved by neural network. At the phase of extracting watermark data, the trained probabilistic neural network recovers the watermark from the watermarked images. However, only the standard deviation is selected for deciding the threshold, which may cause poor robustness to compression and less satisfactory transparency.

In this paper, a robust blind watermarking scheme based on LVQ neural network in HWPT domain is introduced with low computational complexity, good visual quality and reasonable resistance toward various image processing operations. In order to eliminate the correlation of watermark image pixels and to enhance system robustness and security, a watermark image as copyright sign is preprocessed with an affine scrambling. And then the watermark insertion is implemented in eight hyperanalytic wavelet subbands coefficients $\pm\arctan(1/3)$, $\pm\arctan(1/2)$, $\pm\arctan(2)$ and $\pm\arctan(3)$ angles selectively. Moreover, LVQ neural network is applied to memorizing the relation between the watermark and the matching watermarked image successfully. Thus the watermark can be recovered exactly from the distorted image without the original images. The watermark is extracted blindly from the distorted image directly in HWPT domain. Through several experiments, the derived results demonstrate that the proposed scheme is not only robust against operations such as lowpass filtering, noise adding, and JPEG compression etc, but also robust against scaling and cropping attacks and advantageous performances compared to other schemes.

CONSTRUCTION OF HYPERANALYTIC WAVELET PACKET TRANSFORM

Hilbert transform

In Mathematics and signal processing, the Hilbert transform is a linear operator which takes a function $u(t)$ and produces a function $H(u)(t)$, in order to resolve a particular example of the Riemann–Hilbert problem for holomorphic functions with the same domain. It is a fundamental method in Fourier analysis, and supplies a concrete means for determining the conjugate of a given function or Fourier series. Furthermore, the Hilbert transform is the most important in signal processing where it is employed to derive a signal $u(t)$ in the analytic representation.

In many signal processing and communication applications, a fundamental problem appearing is that of extracting the amplitude $a(t)$, and instantaneous phase ($\rho(t)$ of a real, modulated signal $u(t) = a(t)\cos(\rho(t))$. The Hilbert transform can be considered as the convolution of $u(t)$ with the function $h_H(t) = 1/(\pi t)$. Because retrieval of $h_H(t)$ is ill-posed which is not

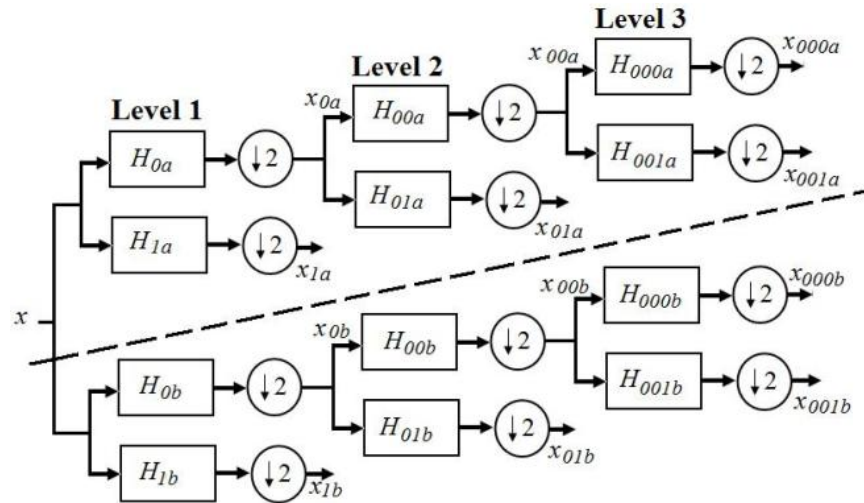


Figure 1. The dual-tree complex wavelet transforms filtering.

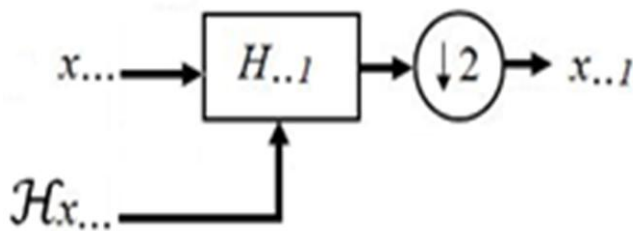


Figure 2. HWT implementation architecture.

integrable the integrals which is not convergent for the defining convolution. Therefore, the Hilbert transform of a

signal $u(t)$ is defined by $H(u)(t) = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{u(\tau)}{t-\tau} d\tau$. If

the underlying amplitude function $a(t)$ is assumed to be relatively narrowband compared with $u(t)$, then the analytic signal can be represented by:

$$u_a(t) = u(t) + jH(u)(t) = a(t)\cos(\rho(t)) + ja(t)\sin(\rho(t)) = a(t)e^{j\rho(t)} \quad (1)$$

Where $|a(t)| = |u_a(t)|$.

Hyperanalytic wavelet transform

For In the DWT domain, a small shift in the input signal can produce a very different series of wavelet coefficients at the output one, due to the exploited decimation

operation in the transform and poor directional selectivity. For overcoming disadvantages, the DTCWT provides a powerful tool allowing for better representation of image features in signal and image analysis, which is better than DWT as it is approximately shifting invariant and has good directional selectivities in two dimensions. This means that with a given complex wavelet coefficient, the squared magnitude provides an accurate measure of spectral energy at a particular location in the space, scale, and orientation. It also means that CWT-based algorithms will automatically be almost shifting invariant, thus reducing many of the artifacts of the critically sampled DWT. Here, it illustrates basic configuration of the dual-tree filtering approach used to obtain in Figure 1. The DTCWT coefficients may be interpreted as arising from the DWT associated with a quasi-analytic wavelet, however the design of these quadrature wavelet pairs is quite complex and it can be done only through approximations. In the new implementation of the HWT, a Hilbert transform of the data is applied. The real wavelet transform is then applied to the analytical signal associated to the input data, and complex coefficients are obtained. The imaginary part of an analytic signal is represented by the Hilbert Transform (HT) of its real part. The filtering unit of HWT is designed as Figure 2.

The wavelet functions ($f(x,y) = \phi(x)\psi(y)$) are constructed such that the HWT coefficients are hyperanalytic signals in their spatial indexing, and refer to such mother wavelets as hyperanalyticizing. The Hilbert transform of complex wavelet ($\psi(x) = \psi h(x) + j\psi g(x)$) is applied representation of HWT implementing. Then, the real wavelet transform is applied to the analytical 2-D signal associated to the input data for obtaining complex coefficients.

The mother wavelet function that constructs monogenic coefficients is a 2-D monogenic signal, and any

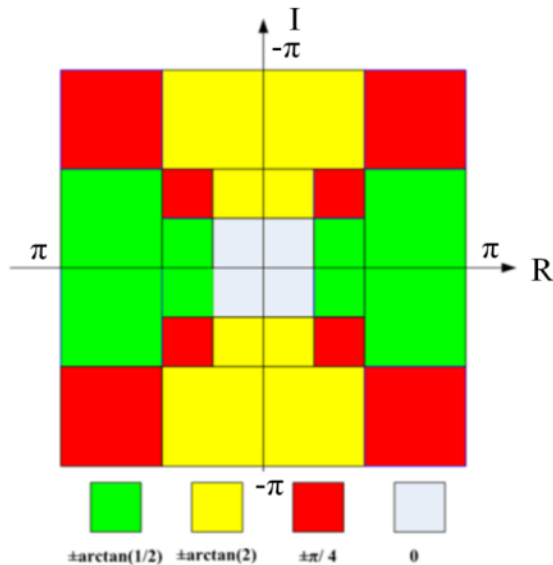


Figure 3. Frequency-domain partition resulted from a two-level 2-D HWT decomposition, where R and I are the real axis and the imaginary axis of the complex frequency domain, respectively.

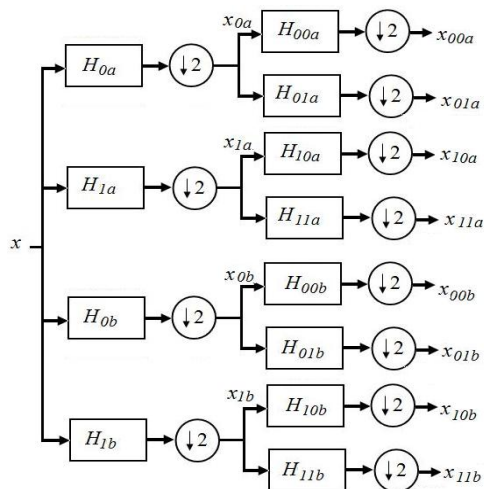


Figure 4. The diagram of HWPT.

quaternionic analysis filter produces hypercomplex wavelet coefficients as a hypercomplex wavelet. The separable 2D-HWT implementation produces the oriented wavelets of four trees, and each one implementing a 2D-DWT. The first tree is employed to the input image. 1D discrete Hilbert transforms are utilized to compute the second and the third trees with the rows (Hx) or columns (Hy) of the input image. The fourth tree is implemented to the outcome obtained after the computation of the two 1D discrete Hilbert transforms of the input image. The hypercomplex signal of a real-valued image $f(x,y)$ is

defined as:

$$f^+(x, y) = f_\theta(x, y) + iH_{\theta x} \{f(x, y)\} + jH_{\theta y} \{f(x, y)\} + kH_{\theta y} \cdot H_{\theta x} \{f(x, y)\}$$

$$= f_\theta(x, y) + if_{H;\theta}^{(1)}(x, y) + jf_{H;\theta}^{(2)}(x, y) + kf_{H;\theta}^{(3)}(x, y) \quad (2)$$

Where $i^2 = j^2 = k^2 = -1$, $k = ij = -ji$ and θ is an arbitrary direction.

The enhancement of the directional selectivities of the 2D-HWT is implemented by linear combinations of subband coefficients belonging to each quadrant 2-D DWTs. This process is repeated several times until the desired final scale is reached.

Comparing to the 2D-DWT, which has only three preferential orientations: $\pi / 2$ (vertical details) and $\pi / 4$ (diagonal details). There are two parts in diagonal details and one represents higher-frequency detailed information which covers the high-frequency component. However, for a two-dimensional hyperanalytic quadrantal wavelet, multi-resolution decomposition of HWT of an input image has the more directional selectivities, presented in Figure 3. According to the directions of θ , the 2D-HWT has six advantageous orientations: $\pm\arctan(1/2)$, $\pm\pi/4$ and $\pm\arctan(2)$.

Hyperanalytic wavelet packet transform

In this section, it introduces a quad-tree wavelet transform called Hyperanalytic Wavelet Packet Transform (HWPT), by noting that the approximate shifting invariance is achieved by a real biorthogonal transform with the double-sampling rate at each scale. HWPT is obtained by computing parallel quadruple wavelet trees, which are subsampled differently. In HWPT, it analyzes an image simultaneously at different resolution levels and orientations and allocates the further iterative decomposition of the high-pass subbands as well as the low-pass ones, which provides more directional selectivities and better frequency localizations. For example, in 2-level HWPT decomposition, there are 32 frequency localization and 26 different orientations such as $\pm\arctan(1/4)$, $\pm\arctan(1/3)$, $\pm\arctan(1/2)$, $\pm\arctan(2/3)$, $\pm\arctan(3/4)$, $\pm\arctan(1)$, $\pm\arctan(5/4)$, $\pm\arctan(4/3)$, $\pm\arctan(3/2)$, $\pm\arctan(5/3)$, $\pm\arctan(2)$, $\pm\arctan(3)$ and $\pm\arctan(4)$. Taking advantages of shifting invariances and directionality selectivities, the decomposition structure and subband selection are determined by considering the better performances of the watermark insertion and further the reduction of computational complexity. The block diagram of decomposition of HWPT is illustrated in Figure 4, where presents real and imaginary parts coefficients from a and b respectively.

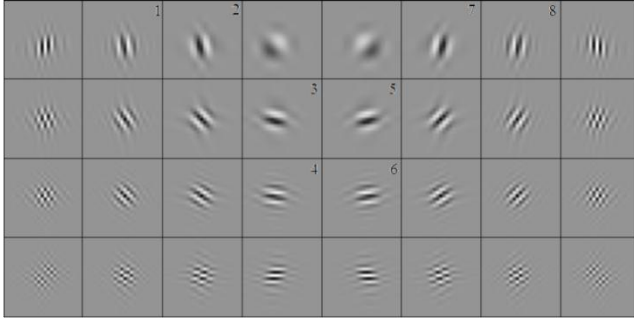


Figure 5. Impulse responses of the HWPT at level 2.

And the corresponding the impulse responses of two-level decomposition is shown in Figure 5.

Discrete HWPT can be described as follows, where an image W ($N \times M$ pixels) is decomposed into each approximation image and the difference components images represented in following notation:

$$\begin{aligned}
 W_{4k(m,n)}^{(1)q,p+1} &= \sum_k \sum_l \varphi_i(l) \varphi_j(k) W_{n;k+2m;l+2n}^{q,p} \\
 W_{4k+1(m,n)}^{(2)q,p+1} &= \sum_k \sum_l \varphi_i(l) \psi_j(k) W_{n;k+2m;l+2n}^{q,p} \\
 W_{4k+2(m,n)}^{(3)q,p+1} &= \sum_k \sum_l \varphi_j(k) \psi_i(l) W_{n;k+2m;l+2n}^{q,p} \\
 W_{4k+3(m,n)}^{(4)q,p+1} &= \sum_k \sum_l \psi_i(l) \psi_j(k) W_{n;k+2m;l+2n}^{q,p} \quad (3)
 \end{aligned}$$

Where $\varphi_i(l)$ and $\psi_j(l)$ are 1-D complex filters applied along the columns and rows respectively. Note that in the φ and ψ , i and j are the quaternionic imaginary numbers that satisfy $ji = k$. $W_{0,(m,n)}^0$ is the pixel value of coordinates (m,n) of image W . At each step, the p -level subbands of coefficients W_k^p is decomposed into four quarter-size images of $(p+1)$ -level which coefficients are :

$$W_{4k(m,n)}^{(1)q,p+1}, W_{4k+1(m,n)}^{(2)q,p+1}, W_{4k+2(m,n)}^{(3)q,p+1} \text{ and } W_{4k+3(m,n)}^{(4)q,p+1}.$$

In the process of inverse HWPT, the original image can be reconstructed from these HWPT coefficients. At different resolution levels and orientations, the approximation images and their different detailed sub-images can be used to reintegrate the reference images of higher resolution. It should be noted that HWPT has a variety of predominant applications because there are some following advantages. Firstly, it provides efficient baseband simulation techniques with the complex envelope of the analytic bandpass signal using reduced sampling rate and compacted data transmission.

Secondly, it offers better directional selectivities of three phases for the analysis using the phase concept with the same amount of computational resources and computation required time. Thirdly, it is the simplicity and the flexible structure with any orthogonal or biorthogonal mother wavelet. Furthermore, HWPT framework provides more detailed description of high frequency parts of signal and it is universal in adapting the transform to an image without training or assuming any statistical property because the same frequency bandwidths can provide good resolution regardless of quadrantal wavelet frequencies.

LEARNING VECTOR QUANTIZATION

Learning vector quantization (LVQ) by Aras et al. (1999) is a subtype of artificial neural networks developed by Kohonen and colleagues. LVQ is a nearest an associative automatically nearest-neighbor classifier that arbitrarily classifies patterns into classes with an error correction encoding procedure related to competitive learning, which contains input layer, competitive layer and output layer.

The input layer contains one node for each input feature and the output layer contains one node for each class. The competitive layer will automatically learn to classify input vectors in a supervised manner. However, the classes that the competitive layer finds are dependent only on the distance between input vectors. The basic idea is to overcast the input space of samples with codebook vectors, each one representing a region marked with one class. A codebook vector can be considered as a prototype of one class member, localized in the center of a class or decision region in the input space. A class can be exemplified by an arbitrary number of codebook vectors however one codebook vector represents one class only.

During the training process of the LVQ, the Euclidean distance from a training vector x , to each node's codebook vector w_i , in the competitive layer (Kohonen layer) is calculated according to the equation:

$$d_i = w_i - x_2 = \left[\sum_{i=1}^N (w_{ij} - x_j)^2 \right]^{\frac{1}{2}} \quad (4)$$

The nearest node is declared to be the winner, and its code-book vector is iterated according to whether the winning node is in the class of the training vector:

If the winner is the correct class, then $w_{i+1} = w_i + \alpha(x - w_i)$.

If the winner is the error class, then $w_{j+1} = w_j + \alpha(x - w_j)$.

Where w_{i+1} is the updated code-book vector after

iteration, w_i the input vector, α and β are learning parameters. The orientation of the code-book adaptation while using Equation 4 depends upon whether the class of training pattern and the class allocated to the reference vector are same. If they are same, the reference vector is moved near to the reference vector training template, otherwise, it is moved away. This movement of the reference vector is controlled by the learning parameters. In this paper, the advanced version of Kohonen's LVQ algorithm, LVQ3, has been adopted. A brief description of the algorithm is given below:

In LVQ3, two code-book vectors w_i and w_j , which are the two nearest neighbors to x , are found by using the Euclidean distance criterion simultaneously, where x and w_i belong to the same class, one of them is correct, say, and x and w_j belong to different classes. Moreover, x must fall into a zone of values called the window, which is defined in terms of relative distances d_i and d_j from w_i and w_j to x , respectively. Then x is defined to fall in a window of relative width ω if

$$\min_{i,j} \left(\frac{d_i}{d_j}, \frac{d_j}{d_i} \right) > \frac{1-\omega}{1+\omega} \quad (5)$$

Furthermore, ensuring that the code-book vectors continue to approximate the respective class distributions, the input sample x is located inside the window between the two closest code-book vectors w_i and w_j , then:

$$\begin{aligned} w_i(t+1) &= w_i(t) - \alpha(t)(x - w_i(t)) \\ w_j(t+1) &= w_j(t) + \alpha(t)(x - w_j(t)) \end{aligned} \quad (6)$$

Additionally, even when x , w_i and w_j , belong to the same class, the code-book vectors are adjusted to enhance the improvement as follows for $k = i, j$:

$$w_k(t+1) = w_k(t) + \varepsilon(t)\alpha(t)(x - w_k(t)) \quad (7)$$

In (6) and (7), t is the discredited time index, and $\varepsilon(t)$ and $\alpha(t)$ are called the learning rate and relative learning rate, respectively.

During testing, LVQ classifies an input vector by assigning it to the same class as the output unit has its code-book vector closest to the input vector. LVQ shows good performance for complicated classification problems because of its fast learning nature, reliability, and

convenience of use. It performs particularly well with small training sets. This property is significantly important for image processing applications, where training data is very limited; it reduces the computational complexity incurred in the testing of traditional non-parametric methods.

THE PROPOSED SCHEME

The basic ideas of embedding and extracting watermark are as following: Firstly, the original host image is partitioned into non-overlapping small blocks, and then each block is decomposed by hyperanalytic wavelet packet transform. Secondly, considering the better performances of accomplishing maximum the watermark strength while decrease the visual distortion, the watermark insertion is implemented in eight hyperanalytic wavelet subbands coefficients $\pm\arctan(1/3)$, $\pm\arctan(1/2)$, $\pm\arctan(2)$ and $\pm\arctan(3)$ angles selectively in every block. For further computational complexity reduction, the selective blocks are embedded watermark according to calculation of the expectations and the standard deviations in these windows. The embedded watermark consists of two parts, a binary signature which is pretreated in order to eliminate the correlation of watermark image pixels and enhance the system security, and reference position sequence is to preserve the same trained LVQ neural networks in the insertion and extraction procedure. LVQ neural network is applied to memorize standard deviation value modulation between the embedded watermark and the coefficients of matching watermarked image with a set of training patterns. Finally, the classifier of trained LVQ is employed to classify a set of testing patterns for the watermark extraction. Following the results produced by the classifier, the watermark is embedded and extracted.

The watermark insertion

Step 1: The host image ($M \times N$) is partitioned into non-overlapping small blocks with each size of 48×48 , where two-level HWPT is preformed. After the decomposing coefficients, 32 frequency localizations are obtained and eight hyperanalytic wavelet subbands coefficients are selected such as, $\pm\arctan(1/3)$, $\pm\arctan(1/2)$, $\pm\arctan(2)$ and $\pm\arctan(3)$ angles selectively in every block for maintaining transparency and robustness of the presented watermarking scheme. Splitting each subband into 4 windows and calculating the standard deviation and expectation of each small window (3×3) for decomposition structures selection can well consider to the performances of the watermarking algorithm and computational complexities according to:

$$D_{standard} = \left(\sum_{n=-1}^1 \sum_{m=-1}^1 (I(i+m, j+n) - D_{average})^2 / 8 \right)^{1/2} \quad (8)$$

$$s.t \quad D_{average} = \sum_{n=-1}^1 \sum_{m=-1}^1 I(i+m, j+n) / 9 \quad (9)$$

Where $I(i, j)$ is the central coefficients of selected window and $I(i+m, j+n)$ are the coefficient of each small window, variable m, n represent the coefficients coordinates.

The watermark distribution is adaptive to the standard deviation of each window in the spatial domain. Moreover, the small spatial windows with larger deviation would be modified significantly. On the other hand, the small structural windows of smaller deviation wouldn't be modified. However, the watermark distribution scheme would result in a worse robustness to compression and a less satisfying transparency in HWPT domain because the small structural windows of smaller deviation could be significant to allow the watermark information embedding in HWPT domain. Thus, according to the potential drawback, the sorting ascending strategy is employed to choose and control the threshold T for modifying adaptively coefficients in both the expectation and the deviation of each 3×3 window within the selected hyperanalytic wavelet subbands coefficients. Calculate the value of $D_{threshold}$ in each 3×3 window for obtaining threshold T according to the equation

$$D_{threshold} = \tau_1 \times D_{average} + \tau_2 \times D_{standard} \quad (10)$$

Threshold T is selected by the amount of t windows, which is assigned by a secret key K_1 .

Step 2: The watermark W segment in two subsequences S and L , and digital signature S with $m_s \times n_s$ bits and reference position information L with l bits. Both of S and L are embedded to selective small coefficients windows with better performance of robustness and transparency.

The signature information $S (m_s \times n_s)$ need to be pretreated in order to eliminate the correlation of watermark image pixels and enhance system robustness and security. For the advantages of reducing computed complexities and obtaining easily inverse transform comparing with Arnold transform, the watermark image is pretreated through an affine scrambling. The affine scrambling is showed as equation,

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}, \text{ where } \begin{vmatrix} a & b \\ c & d \end{vmatrix} \neq 0 \quad (11)$$

For enhancing the statistical imperceptible through embedding watermark, series of $\{-1,1\}$ values substitute for $\{0,1\}$ which is the value of watermark image by scrambling, respectively. The new watermark is

generated ($w'_i = w_i \cdot p_i$), according to a sequence of the binary pseudo-random p_i modulating the watermark, where $p_i \in \{-1,1\}$ and $0 \leq i < m_s \times n_s$. The reference position sequence L is not modified in the embedding process, which is in the character of preservation of the same learned LVQ in the insertion and extraction of watermark information. Finally, the watermark $W = \{w(t), 1 \leq t \leq M, M = m_s \times n_s + 1\}$ is a binary sequence, which is selected randomly position sequence $P = \{p_M(i, j)\}$ according to secret key K_2 assigned by the image owner.

Step 3: In the procedure of training the LVQ neural networks, each pixel can be forecasted by its neighbors because it has high relevance to its neighbors. The relationship is employed to be learnt by the training process of LVQ neural networks so that it maintains unchanged or changes a little when the watermarked image robust to some image operations. The training data in every 3×3 window is extracted from the each reference position in a subset of $P, Q = \{I(i_t, j_t)\}$. The training dataset is obtained by the neighbors of the central pixel which are (12)

Tan-sigmoid function f_{tan} is adopted by denoting a transmission function in the competitive layer for training LVQ and it is given by $f_{tan} \left(\sum_{i=1}^N w_{ij} - x_j \right)$.

Step 4: In each embedding position, the forecasting dataset is obtained in a 3×3 window and it is given by (13).

By the training with a nearest associative automatically nearest-neighbor classifier, the forecasting value is obtained in each window. From a training vector X to each node's code-book vector w_i in the competitive layer, the Euclidean distance d_t^{LVQ} is obtained. Log-sigmoid function f_{log} is employed by denoting a transmission function in the output layer with

$$f_{log} \left(\sum_{i=1}^N w_{ij} - d_t^{LVQ} \right) = p_{t_2}.$$

Compared the forecasted pixel values and the original ones, the strategy of watermark information insertion are as follow (14)

Where $\alpha_{t_2} = \sigma T_{t_2}$ denotes the strength of watermark and σ is an embedding constant for adjusting the compromise between visual quality and robustness of the watermarked image, and T_{t_2} is $D_{threshold}$ in block t_2 , and

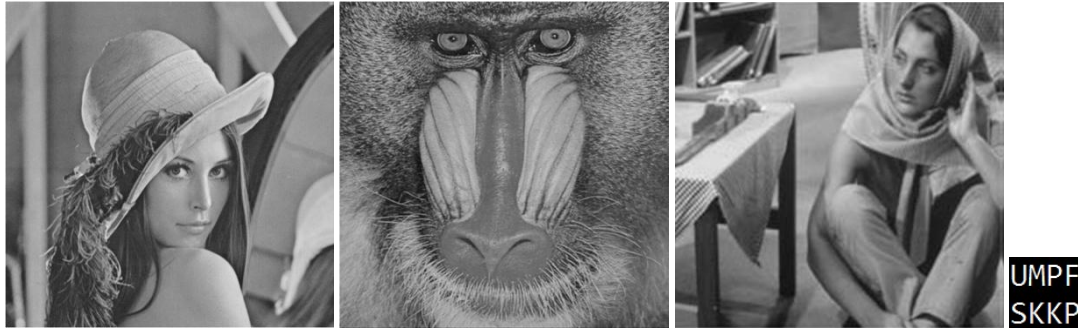


Figure 6. the original images of (a) Lena; (b) Baboon; (c) Barbara and (d) watermark.

$I(i_{t_2}, j_{t_2})$ is the central coefficient of selected window.

Step 5: Signature bits are embedded into the selective windows and 2-level inverse HWPT of the sub-images is performed. Then, the watermarked image can be obtained.

The watermark extraction

In the proposed algorithm, watermark can be extracted without the original image and LVQ is employed to recognize the extracted watermark because it classifies an input vector by assigning it to the same class as the output unit has its code-book vector closest to the input vector. The procedure is summarized as follows:

$$X = \left\{ \begin{matrix} x_{i_1} = \{I(i_{t_1}-1, j_{t_1}-1), I(i_{t_1}-1, j_{t_1}), I(i_{t_1}-1, j_{t_1}+1), I(i_{t_1}, j_{t_1}-1), I(i_{t_1}, j_{t_1}+1) \\ x_{i_2} = \{I(i_{t_1}+1, j_{t_1}-1), I(i_{t_1}+1, j_{t_1}), I(i_{t_1}+1, j_{t_1}+1)\}, t_1=1, 2, \dots, l \end{matrix} \right\} \quad (12)$$

$$X_c = \left\{ \begin{matrix} x_{i_1} = \{I(i_{t_2}-1, j_{t_2}-1), I(i_{t_2}-1, j_{t_2}), I(i_{t_2}-1, j_{t_2}+1), I(i_{t_2}, j_{t_2}-1), I(i_{t_2}, j_{t_2}+1) \\ x_{i_2} = \{I(i_{t_2}+1, j_{t_2}-1), I(i_{t_2}+1, j_{t_2}), I(i_{t_2}+1, j_{t_2}+1)\}, t_2=l+1, l+2, \dots, l+m \end{matrix} \right\} \quad (13)$$

$$\delta_{t_2} = \begin{cases} \max(I(i_{t_2}, j_{t_2}), p_{t_2} + \alpha_{t_2}) & \text{if } s_{t_2} = 1 \\ \min(I(i_{t_2}, j_{t_2}), p_{t_2} - \alpha_{t_2}) & \text{if } s_{t_2} = 0 \end{cases} \quad (14)$$

Step 1: Two-level hyperanalytic wavelet packet transform is performed on the watermarked image and eight hyperanalytic wavelet subbands coefficients are selected and each selected subband is split into 4 windows according to the secret key K_1 . The selected randomly position sequence $P = \{p_M(i, j)\}$ is obtained according to secret key K_2 .

Step 2: The LVQ model can be constructed by a set of training patterns training. The training dataset component is extracted from the attacked watermarked image. In the training process, the training data in every 3×3 window is extracted from the each reference position in a subset of P , $Q = \{I(i_{t_1}, j_{t_1})\}$. The training dataset X' is obtained by the neighbors of the central pixel which are (15)

Step 3: The actual output dataset are forecasted to extract signature information by using the well-trained LVQ in each embedding position by (16) By the training with a nearest associative automatically nearest-neighbor classifier, the forecasting value is obtained in each window. From a training vector X to each node's code-book vector w_i in the competitive layer, the Euclidean distance d'_i is obtained. The watermark bits are extracted by comparing the relation between the actual pixel values and forecasted ones by the trained LVQ neural networks. The relations are described as follows (17)

Step 4: A complete watermark sequence w' is obtained and inverse affine transform perform on the sequence, then binary watermark is extracted from the revised selective image window.

EXPERIMENTAL RESULTS

In this section, some numerical experiments and simulations are presented to evaluate the performance of the proposed watermarking scheme. In all experiments, the 8-bit grayscale images (512×512) including “Lena” “Baboon” and “Barbara” are tested (shown in Figure 6. (a)–(c)), and a 32×32 binary image is regarded as the owner's signature which is used to verify image copyrights as shown in Figure 6(d) and in order to

eliminate the correlation of watermark image pixels and enhance system robustness and security through affine scrambling. In order to implement the proposed watermarking scheme, the experimental parameters should be determined. In LVQ network, the w value is between 0.2 and 0.3. As LVQ is a fine-tuning method for the active neuron code-book vectors, one usually starts with a fairly small value, like $\alpha(0) = 0.01$ and lets it decrease linearly to zero during the total number of iterations of the training set. The train pattern number is by 100 for the training LVQ well because the performance of training the LVQ isn't well with a small number of train patterns, but too much training patterns may cause overfit. For controlling $D_{\text{threshold}}$, it is set $\tau_1 = 0.1$ and $\tau_2 = 0.5$. The embedding strength constant σ is determined, which controls the compromise between the robustness and the transparency of watermark and it is set 0.8. The experimental results are compared with other schemes in (Nafornita C, et al., (2007); (Wen X, et al., (2009)).

Imperceptibility

$$x'_c = \left\{ \begin{matrix} x'_i = \{I'(i_h - 1, j_h - 1), I'(i_h - 1, j_h), I'(i_h - 1, j_h + 1), I'(i_h, j_h - 1), I'(i_h, j_h + 1)\} \\ I'(i_h + 1, j_h - 1), I'(i_h + 1, j_h), I'(i_h + 1, j_h + 1) \}, t_1 = 1, 2, \dots, l \end{matrix} \right\} \quad (15)$$

$$x'_c = \left\{ \begin{matrix} x'_i = \{I'(i_2 - 1, j_2 - 1), I'(i_2 - 1, j_2), I'(i_2 - 1, j_2 + 1), I'(i_2, j_2 - 1), I'(i_2, j_2 + 1)\} \\ I'(i_2 + 1, j_2 - 1), I'(i_2 + 1, j_2), I'(i_2 + 1, j_2 + 1) \}, t_2 = l + 1, l + 2, \dots, l + m \end{matrix} \right\} \quad (16)$$

$$w'_k = \begin{cases} 1, & I'(i_{t_2}, j_{t_2}) > P_{t_2} \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

$$UQI = \frac{4\sigma_{xy} \bar{x} \bar{y}}{(\sigma_x^2 + \sigma_y^2) [\bar{x}^2 + \bar{y}^2]} \quad (18)$$

where

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i, \sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2, \sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \bar{y})^2$$

$$\text{and } \sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})$$

The imperceptibility determines to which extent the embedding process has altered the perceptual image quality. Image quality is usually measured using Universal Image Quality Index (UQI) (Wang Z and Bovik AC (2002)) value between the original image $x = \{x_i | i = 1, 2, \dots, 480 \times 480\}$ and watermarked image $y = \{y_i | i = 1, 2, \dots, 480 \times 480\}$ (18)

The UQI of a watermarked image is determined by two main factors. On the one hand, given a fixed number of

bits to be embedded, the UQI is determined by the embedded strength. The larger embedded strength leads to a more robust watermark, but results in a lower UQI, and vice versa. On the other hand, given fixed the watermark strength, the number of bits to be embedded decides the UQI of the watermarked image. The more bits embedded the lower value of UQI, and vice versa.

In Simulation results, Table 1 shows that the results of watermark embedding on standard images Lena, Baboon and Barbara, and the results of UQI is limited over 0.9, which guarantees a good watermark transparency. From Table 1, in the novel technique, UQI is more than 0.95 and better than schemes (Nafornita C, et al., (2007); Wen X, et al., (2009)) in the watermark invisibility.

Robustness

After extracting the watermark, the Bit Error Rate (BER) is employed to quantify the correlation between the original watermark and the extracted one is used. It is defined as follows:



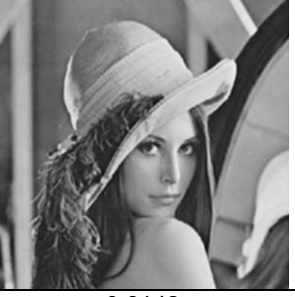
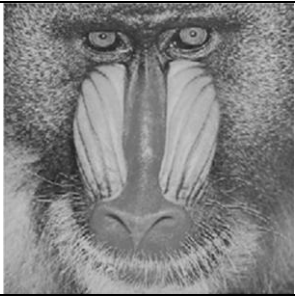

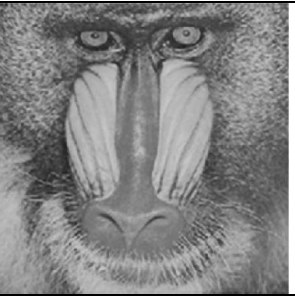



$$BER = \frac{\sum_1^{m_s} \sum_1^{n_s} w(i, j) \oplus w'(i, j)}{m_s \times n_s} \quad (19)$$

Where m_s and n_s are the height and width of the watermark, respectively. $w(i, j)$ and $w'(i, j)$ are the watermark bits located at coordinates (i, j) of the original watermark and the extracted watermark. Here $w(i, j)$ is set to 1, if it is watermark bit 1; otherwise, it is set to 0. $w'(i, j)$ is set in the same way. So the value of $w(i, j) \oplus w'(i, j)$ is either 0 or 1. Here both nongeometric and geometric attacks are considered. The software of Photoshop CS4 is used to simulate these attacks. Table 5 shows watermarks are extracted for the distorted host image "Lena". It can be seen from Table 5 that the extracted watermark images are all recognizable, which also demonstrates that the watermarks are successfully detected.

Robustness to JPEG compression

With the wide-spread use of image compression standard JPEG, lossy compression is a highly common form of image processing. It can be seen that the proposed scheme, schemes (Nafornita C, et al., (2007); Wen X, et al., (2009)) have the best robustness to compression when JPEG quality factor is more than 90, but the robustness to JPEG lossy compression is remarkable in our algorithm when JPEG quality factor is less than 40.

Table 1. The performance of attack free.

Tested images	Proposed scheme	Scheme (Nafornta et al.)	Scheme (Wen et al.)
Lena			
PSNR	0.9831	0.9317	0.9142
Baboon			
PSNR	0.9719	0.9328	0.9271
Barbara			
PSNR	0.9839	0.9314	0.9023

From Table 2, it is shown that our algorithm displays excellent watermark robustness to JPEG compression.

Robustness to additive noise and filterings

Lowpass filters are a family of filters that are commonly applied in image processing, including averaging filters and Gaussian filters etc., and therefore, lowpass filters are of interest to watermark designers, and additive noise is another attack a watermarked image often undergoes. Comparing to schemes by Nafornta et al. (2007); Wen et al. (2009), the proposed scheme is more robustness to lowpass filters and additive noise. Table 3 shows that the proposed algorithm has more outstanding performance on the attacks of lowpass filtering and additive noise than schemes (Nafornta et al., 2007; Wen et al., 2009), but others' schemes are sensitive to filterings and noisy attacks.

Robustness to images scaling and cropping

The proposed scheme also is particularly interested in the watermark performance under the attacks of image scaling and cropping. Four different scaling and cropping attacks were tried. Table 4 shows the simulation results about the proposed algorithm outperforms schemes (Nafornta et al., 2007; Wen et al., 2009).

DISCUSSION

This paper proposed a blind image watermarking algorithm based on the learning vector quantization neural networks and the hyperanalytic wavelet packet transform.

HWPT offers better directional selectivity and more detailed description of high frequency for the analysis using the phase concept with the same amount of computational resources but the reduced computation

Table 2. The performance of JPEG compression.

Attacks		Scheme		
JPEG	Proposed	Nafornta et al.	Wen X et al.	
Lena				
90	0.0927	0.1017	0.1021	
80	0.0964	0.1245	0.1167	
70	0.1139	0.1751	0.1827	
60	0.1614	0.2015	0.2564	
50	0.2125	0.2675	0.3364	
40	0.2546	0.4037	0.3513	
Baboon				
90	0.0921	0.1025	0.0981	
80	0.0956	0.1185	0.1042	
70	0.1163	0.1621	0.1782	
60	0.1783	0.1974	0.2453	
50	0.2316	0.2249	0.3055	
40	0.2978	0.3817	0.3583	
Barbara				
90	0.0925	0.0986	0.0965	
80	0.0967	0.1157	0.1085	
70	0.1149	0.1731	0.1787	
60	0.1704	0.2058	0.2275	
50	0.2257	0.2752	0.2528	
40	0.2825	0.4133	0.3259	

Table 3. The performance of lowpass filters and additive noise

Attacks	scheme		
	Proposed	Nafornta et al.	Wen et al.
Lena			
Median Filter3*3	0.1635	0.2053	0.1862
5*5	0.2249	0.2942	0.1996
Gaussian filter 3*3	0.1239	0.1783	0.1571
5*5	0.2007	0.2478	0.2076
Gaussian Noise (0,0.01)	0.1059	0.1271	0.1092
(0,0.02)	0.1738	0.2025	0.1954
Salt and Peppers Noise (0,0.01)	0.0987	0.1183	0.1081
(0,0.02)	0.1542	0.2004	0.1819
Baboon			
Median Filter3*3	0.1503	0.1978	0.1784
5*5	0.1923	0.2732	0.2264
Gaussian filter 3*3	0.1341	0.1943	0.1639
5*5	0.1929	0.2307	0.2245
Gaussian Noise (0,0.01)	0.1011	0.1317	0.1275
(0,0.02)	0.1687	0.1951	0.1853
Salt and Peppers Noise (0,0.01)	0.1029	0.1228	0.1164

Table 3. Contnd.

(0,0.02)	0.1369	0.1973	0.1645
Barbara			
Median Filter3*3	0.1679	0.1957	0.1817
5*5	0.2011	0.2617	0.2322
Gaussian filter 3*3	0.1467	0.1873	0.1538
5*5	0.1981	0.2591	0.2115
Gaussian Noise (0,0.01)	0.1098	0.1254	0.1172
(0,0.02)	0.1685	0.2018	0.1811
Salt and Peppers Noise (0,0.01)	0.0927	0.1183	0.1013
(0,0.02)	0.1559	0.2173	0.1769

Table 4. The performance of image sclaing and cropping.

Attacks	Lena			Baboon			Barbara		
	scheme								
	Proposed	(Naornita et al.)	(Wen et al.)	Proposed	(Naornita et al.)	(Wen et al.)	Proposed	(Naornita et al.)	(Wen et al.)
Scaling									
0.5	0.2024	0.2557	0.2257	0.1825	0.2118	0.2071	0.1923	0.2274	0.2082
7	0.1253	0.1433	0.1345	0.1173	0.1572	0.1411	0.1171	0.1547	0.1352
1.2	0.1541	0.1789	0.1522	0.1651	0.1981	0.1767	0.1573	0.1803	0.1644
1.5	0.2186	0.2542	0.2351	0.1856	0.2236	0.1983	0.2076	0.2372	0.2158
Cropping									
0.1	0.1232	0.1437	0.1373	0.1041	0.1283	0.1093	0.1058	0.1247	0.1192
0.2	0.1586	0.1887	0.1625	0.1486	0.1689	0.1556	0.1518	0.1779	0.1653
0.3	0.1771	0.1984	0.1886	0.1611	0.1916	0.1721	0.1648	0.1939	0.1741
0.4	0.2192	0.2373	0.2251	0.1957	0.2275	0.2171	0.2085	0.2342	0.2173

Table 5. Results of extracted watermarks in proposed scheme.

JPEG 90	JPEG 80	JPEG 70	JPEG 60	JPEG 50
JPEG 40	Median Filter 3*3	Median Filter 5*5	Gaussian filter 3*3	Gaussian filter 5*5
Gaussian Noise (0,0.01)	Gaussian Noise (0,0.02)	Salt and Peppers Noise (0,0.01)	Salt and Peppers Noise (0,0.02)	Scaling 0.8
Scaling 0.9	Scaling 1.1	Scaling 1.2	Cropping 0.1	Cropping 0.2
Cropping 0.3	Cropping 0.4			

time required due to the flexible structure with any orthogonal or biorthogonal mother wavelets. To maintain transparency and robustness of the presented watermarking scheme, windows for watermark insertion are selected and a bit watermark is embedded to the central pixel in each window. Since the central image pixels have high correlation with its neighbors, these relations can be predicted by the LVQ neural networks training. Moreover, the distorted watermark information bits are effectively extracted without limitation of a specific set of attacked operations. Experiments demonstrate that the watermarked image is numerically indistinguishable from the original one, simultaneously showing that the proposed scheme is robust against general signal processing operations such as JPEG compression, median filters, Gaussian filters, additive noise and some geometric attacks and the performance of the proposed scheme is better than others' schemes. The proposed algorithm is suitable for applications like copyright protection, where the embedded data is the information relevant to the owner of the digital images media.

ACKNOWLEDGMENT

We would like to thank the helpful suggestions and comments from editors and anonymous reviewers. This work was supported in part University Malaysia Pahang Postgraduate Research Grant Scheme (PRGS100324).

REFERENCES

- Aras N, Oommen BJ, Altinel IK (1999). The Kohonen network incorporating explicit statistics and its application to the travelling salesman problem. *Neural Netw.*, 12: 1273-1284
- Bulow T, Sommer G (2001). Hypercomplex signals-a novel extension of the analytic signal to the multidimensional case. *IEEE Trans. Signal Process.*, 49: 28-44.
- Cox IJ, Miller ML. (2002). The first 50 years of electronic watermarking. *EURASIP J. Appl. Signal Process.*, 2: 126-132
- Fu Y G, Shen R, Lu H (2004). Watermarking scheme based on support vector machine for color images. *Ints. Electron. Eng. Elect. Lett.*, 40(16): 986-987.
- Hou CT, Yen SL (2010). Robust lossless image watermarking based on a-trimmed mean algorithm and support vector machine. *J. Syst. Software.*, 83: 1015-1028
- Mabtoul S, Ibn-Elhaj E, Aboutajdine D (2006). A Blind Image Watermarking Algorithm Based on Dual Tree Complex Wavelet Transform. *Proceedings of Int. Symp. Commun. Control Signal Process.*, 3: 13-15
- Nafornita C, Firoiu I, Boucher JM, Isar A (2007). A new watermarking method based on the use of the hyperanalytic wavelet transform. *Proc. SPIE.*, 7: 401-404
- Sofia CO, Georgios M (2006). The hyperanalytic wavelet transform, 456 Dept. Math., Imperial College, London, U.K., Imperial College Statistics 457 Section Tech. Rep., TR-06-02.
- Thompson AI, Bouridane A, Kurugollu F (2007). Spread Transform Watermarking for Digital Multimedia Using the Complex Wavelet Domain. *Proceedings of the 2007 ECSIS Symposium on Bio-inspired, Learn. Intell. Syst. Secur.*, pp. 123-132.
- Wang SH, Lin YP (2004). Wavelet Tree Quantization for Copyright Protection Watermarking. *IEEE Trans. Image Process.*, 13(2): 154-165.
- Wang Z, Bovik AC (2002). A Universal Image Quality Index. *IEEE Signal Process. Lett.*, 9: 81-84.
- Wen X, Zhang H, Xu X, Quan J (2009). A new watermarking approach based on probabilistic neural network in wavelet domain. *Soft Comput.*, 13: 355-360
- Zhang G, Wang S, Wen Q (2004). An adaptive block-based blind watermarking algorithm. *Proc. IEEE ICSP.*, pp. 2294-2297.