# WATER MARKING SCHEME WITH HIGH CAPACITY CDMA

**Dr K RameshBabu** [1], **Vani.kasireddy** [2]

[1] Professor, ECE Dept, hitam, jntuh, Hyderabad, AP, India

[2] vani.kasireddy ,PG student,ECE dept,hitam,jntuh,Hyderabad,AP,India

**Abstract:**

In this paper, we propose a high capacity CDMA based watermarking scheme based on orthogonal pseudorandom sequence subspace projection. We introduced a novel idea to eliminate the interference due to the correlation between the host image and the code sequences in the watermark extraction phase, and therefore, it improve the robustness and message capacity of the watermarking scheme. We give the implementation steps of the proposed scheme and test its performance under different attack conditions by a series of experiments. Experimental results show higher robustness than the canonical scheme under different attack conditions.

**Keywords:** CDMA, watermarking, high capacity, oval approach sub space projection, and wavelet transform.

## 1. Introduction

### A. Digital watermarking life-cycle phases

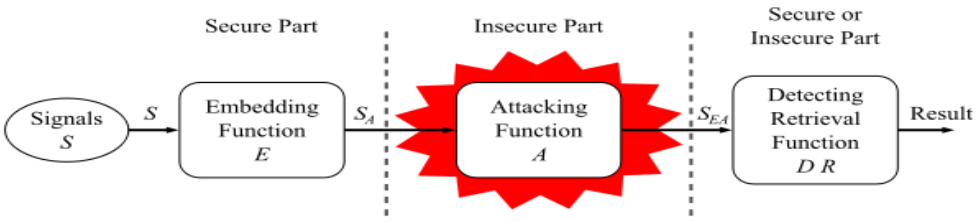


Fig 1.1 Digital watermarking life-cycle phases

Then the watermarked digital signal is transmitted or stored, usually transmitted to another person. If this person makes a modification, this is called an attack. While the modification may not be malicious, the term attack arises from copyright protection application, where pirates attempt to remove the digital watermark through modification. There are many possible modifications, for example, lossy compression of the data (in which resolution is diminished), cropping an image or video or intentionally adding noise. Detection (often called extraction) is an algorithm which is applied to the attacked signal to attempt to extract the watermark from it. If the signal was unmodified during transmission, then the watermark still is present and it may be extracted. In robust digital watermarking applications, the extraction algorithm should be able to produce the watermark correctly, even if the modifications were strong. In fragile digital watermarking, the extraction algorithm should fail if any change is made to the signal.

**B Digital Watermark:** Also referred to as simply watermark, a pattern of bits inserted into a digital image, audio, video or text file that identifies the file's copyright information (author, rights, etc.). The name comes from the faintly visible watermarks imprinted on stationary that identify the manufacturer of the stationery. The purpose of digital watermarks is to provide copyright protection for intellectual property that's in digital format.

**C. General Framework for Digital Watermarking:** Digital watermarking is similar to watermarking physical objects except that the watermarking technique is used for digital content instead of physical objects. In digital watermarking a low-energy signal is imperceptibly embedded in another signal. The low energy signal is called watermark and it depicts some metadata, like security or rights information about the main signal. The main signal in which the watermark is embedded is referred to as cover signal since it covers the watermark. The cover signal is generally a still image, audio clip, video sequence or a text document in digital format.

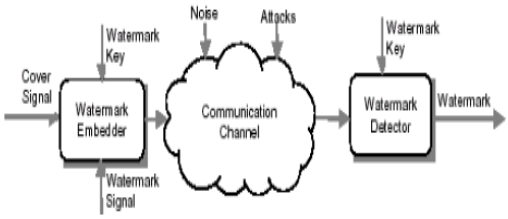


Fig 1.3: watermark embedded and a watermark detector

**D. Digital Watermarking System:** The digital watermarking system essentially consists of a watermark embedded and a watermark detector (see Figure). The watermark embedded inserts a watermark onto the cover signal and the watermark detector detects the presence of watermark signal. Note that an entity called watermark key is used during the process of embedding and detecting watermarks. The watermark key has a one-to-one correspondence with Watermark signal (i.e., a unique watermark key exists for every watermark signal). The watermark key is private and known to only authorized parties and it ensures that only authorized parties can detect the watermark. Further, note that the communication channel can be noisy and hostile (i.e., prone to security attacks) and hence the digital watermarking techniques should be resilient to both noise and security attacks.A watermark is an identifying feature, like a company logo, which can be used to provide protection of some "cover" data. A watermark may be either visible i.e. perceptible or invisible i.e. Imperceptible both of which offer specific advantages when it comes to protecting data. Watermarks may be used to prove ownership of data, and also as an attempt to enforce copyright restrictions.

**Types of watermarking:**

**i)** Visible watermarking.
**ii)** Invisible watermarking
**iii)** Watermarking applications:

## 2. Cdma Watermarking:

### 2.1. Development of CDMA water marking model

There are several prior references to CDMA watermarking of digital video. Hartung and Girod's work is notable in recognizing CDMA as a viable choice for video watermarking. Their approach parses the video into a linear stream of individual pixel elements. A watermark represented by a binary pattern is then expanded by an *m*-sequence and added pixel-by-pixel to the uncompressed video. Watermark recovery is done by matched filtering. In this paper we build upon the work reported in by developing a more complete model for CDMA-based video watermarking in a multi-user/multiple media environment. In particular, instead of linearizng the video as a 1-D pixel stream, we model the video as a *bit plane* stream; the 2D counterpart of bit stream used in (1). By closely following the conventional CDMA model, it is possible to address a variety of watermark removal/destruction attempts that go beyond random noise attacks. For example, any spread spectrum watermarking that relies on *m*-sequences is extremely sensitive to timing errors. Frame drops, intentional or otherwise, destroy the delicate pattern of an *m*-sequence and can seriously challenge watermark identification.We model digital video as a function in time and space represented by $I(x, y, t)$. $I(x, y, t)$ can then be sequenced along the time axis as bit planes:Where $i(.)$ is the $n$th bit plane of the $j$th frame positioned at $t = jTf + nTb$ . $Tf$ and $Tb$ are frame length and bit plane spacing respectively and are related by $Tf = bTb$ where $b$ is the number of bit planes per frame.

$$I(x, y, t) = \sum_{j} \sum_{n=0}^{b} i(x, y, t - (jT_f + nT_b)) \qquad \text{Eq.3}$$

Two questions arise at this point, 1): how is a watermark defined? and 2): where in the bit plane stream is it inserted. We define the watermark by a bit plane, $w(x, y)$, spatial dimensions of which match that of the video frames. The content of the watermark plane can be selected to suite varied requirements. It can contain a graphical seal, textual information about the source or any other data deemed appropriate for watermarking. In the context of CDMA, $w(x, y)$ can be thought of as the message. This message is then spread using a 2D *m*-sequence or *m*-frames $f(x, y, t)$. To generate *m*-frames, a one dimensional m sequence is rearranged in a 2D pattern. Depending on the period of the *m*-sequence and the size of each video frame, the 1D to 2D conversion may span up to $k$ frames and will repeat afterwards. Spreading of the "message", i.e. the watermark $w(x, y)$ is now defined by a periodic frame sequence given

by $w_{ss} = w(x, y) \sum_{j=0}^{k-1} \phi_j(x, y, t_j)$ ..... Eq. 4

Where f j (x, y, t j ) is f j positioned at yet to be determined locations t = t j . wss must now be aligned with and inserted into video bit plane stream in (3). The embedding algorithm works as follows. In every frame the bit plane at t = t j is tagged then removed and replaced by f j (x, y, t j ). The question now is which bit planes are tagged and in what order? It is safe to assume that in most cases the LSB plane bit distribution is random and can be safely replaced by the watermark. However, LSB plane is vulnerable to noise and other disturbances but bit planes can be used to embed a watermark with small to negligible effect on quality. In one example, watermark placement in one of 4 lower bit planes did not significantly impact video quality.In order to embed wss in the video, we define a separate multilevel sequence v and use v( j) as pointer to the jth bit plane position. There are many ways to create v. One simple method is to start with a binary m-sequence u and add 3 cyclically shifted versions. Let u = (uo,u1,L,up-1) be an m-sequence of period p. Define D as an operator that cyclically shifts the elements of u to the left D(u) = (u1,L,up-1,uo ). We define v by v = u + D (u) + D2 (u) where Dk is the kth cyclic shift of u. The new sequence now has two key properties, 1): it is still periodic with period p and 2): it is a 4 valued sequence taking on amplitudes in the range {0, 1… 3}. The significance of 4 values is that the watermark will be limited to 4 lower bit plane positions. This number can clearly change. We can now align wss in (4) with the timeline defined in (3)

$$w_{ss} = w(x,y)\sum_{j=0}^{k-1}\phi(x,y,v(j)T_b) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad \text{Eq. 5}$$

wss is now a spread spectrum version of the watermark at pseudorandom locations determined by v( j). The second task is accomplished by using v (j) Tb as pointers to the candidate bit planes where the watermark must be inserted. In order to take the last step, the designated bit planes must be removed and replaced by the corresponding elements of wss . The formalism to achieve this goal is through the use of a gate function defined by

$$gate(t - v(j)Tb) = \begin{cases} 0 \\ 1 \end{cases} \quad \text{for t=v(j)Tb} \qquad 0 \le t \le T_c \qquad \ldots\ldots\ldots \qquad \text{Eq .6}$$

Multiplying video bit plane stream in (3) by the gate function above removes the bit plane at *v*( *j*). The spread watermark bit plane stream in (5) is positioned such that the individual planes correspond exactly to the planes just nulled by the gate function. Putting it all together, CDMA watermarked video can be written as

$$I_{wm}(x,y,t) = \sum_j \left\{ \sum_{n-0}^{b-1} i(x,y,jT_f + nT_b)gate(t - jT_f - v(n)T_b) + w(x,y)\phi(x,y,jT_f + v(n)T_b) \right\}$$

$$\phi_j + k = \phi_j \qquad \text{Eq.7}$$

## 3. Wavelet Transforms:

It provides the time-frequency representation. often times a particular spectral component occurring at any instant can be of particular interest. In these cases it may be very beneficial to know the time intervals these particular spectral components occur. For example, in EEGs, the latency of an event-related potential is of particular interest. Wavelet transform is capable of providing the time and frequency information simultaneously, hence giving a time-frequency representation of the signal.

**3.1.The Continuous Wavelet Transform:** The continuous wavelet transform was developed as alternative approaches to the short time Fourier transform to overcome the resolution problem. The wavelet analysis is done in a similar way to the STFT analysis, in the sense that the signal is multiplied with a function, {it the wavelet}, similar to the window function in the STFT, and the transform is computed separately for different segments of the time-domain signal. However, there are two main differences between the STFT and the CWT: 1. The Fourier transforms of the windowed signals are not taken, and therefore single peak will be seen corresponding to a sinusoid, i.e., negative frequencies are not computed. 2. The width of the window is changed as the transform is computed for every single spectral component, which is probably the most significant characteristic of the wavelet transform. The continuous wavelet transform is defined as follows

$$CWT_x^{\psi}(\tau,s) = \Psi_x^{\psi}(\tau,s) = \frac{1}{\sqrt{|s|}}\int x(t)\psi^*\left(\frac{t-\tau}{s}\right)dt \qquad \ldots\ldots\ldots\ldots\ldots \qquad \text{Eq 3.1}$$

As seen above variables, **tau** and **s,** the **translation** and **scale** parameters, respectively. **psi(t)** is the transforming function, and it is called **the mother wavelet** .

**The scale**

The parameter **scale** in the wavelet analysis is similar to the scale used in maps. As in the case of maps, high scales correspond to a non-detailed global view (of the signal), and low scales correspond to a detailed view. Similarly, in terms of frequency, low frequencies (high scales) correspond to a global information of a signal (that usually spans the entire signal), whereas high frequencies (low scales) correspond to a detailed information of a hidden pattern in the signal (that usually lasts a relatively short time). Cosine signals corresponding to various scales are given as examples in the following figure.

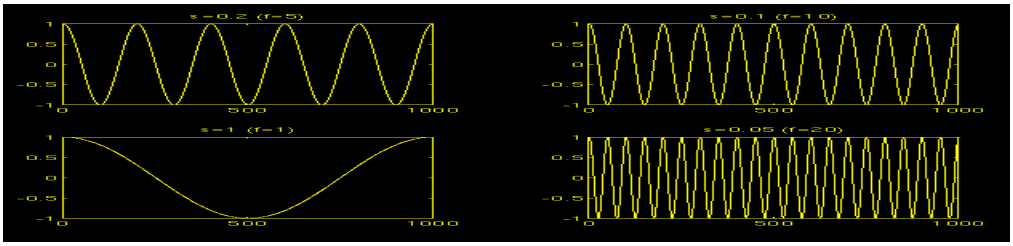


Fig 3.1: Cosine signals corresponding to various scales

Fortunately in practical applications, low scales (high frequencies) do not last for the entire duration of the signal, unlike those shown in the figure, but they usually appear from time to time as short bursts, or spikes. High scales (low frequencies) usually last for the entire duration of the signal. Scaling, as a mathematical operation, either dilates or compresses a signal. Larger scales correspond to dilated (or stretched out) signals and small scales correspond to compressed signals. All of the signals given in the figure are derived from the same cosine signal, i.e., they are dilated or compressed versions of the same function. In the above figure, **s=0.05** is the smallest scale, and **s=1** is the largest scale. In terms of mathematical functions, if **f(t)** is a given function **f(st)** corresponds to a contracted (compressed) version of **f(t)** if **s > 1** and to an expanded (dilated) version of **f(t)** if **s < 1** . However, in the definition of the wavelet transform, the scaling term is used in the denominator, and therefore, the opposite of the above statements holds, i.e., scales **s > 1** dilates the signals whereas scales **s < 1** , compresses the signal. This interpretation of scale will be used throughout this text.

**3.2 Computations of Cwt**

Interpretation of the above equation will be explained in this section. Let **x(t)** is the signal to be analyzed. The mother wavelet is chosen to serve as a prototype for all windows in the process. All the windows that are used are the dilated (or compressed) and shifted versions of the mother wavelet. There are a number of functions that are used for this purpose. The Morlet wavelet and the Mexican hat function are two candidates, and theyare used for the wavelet analysis.

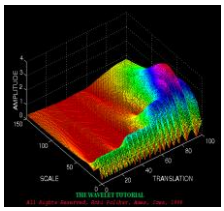


Fig 3.2: (CWT) of this signal

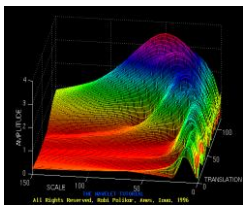


Fig 3.3 CWT signal with high frequencies

Note that in Figure 3.2 that smaller scales correspond to higher frequencies, i.e., frequency decreases as scale increases, therefore, that portion of the graph with scales around zero, actually correspond to highest frequencies in the analysis, and that with high scales correspond to lowest frequencies. Remember that the signal had 30 Hz (highest frequency) components first, and this appears at the lowest scale at translations of 0 to 30. Then comes the 20 Hz component, second highest frequency, and so on. The 5 Hz component appears at the end of the translation axis (as expected), and at higher scales (lower frequencies) again as expected. Now, recall these resolution properties: Unlike the STFT which has a constant resolution at all times and frequencies, the WT has a good time and poor frequency resolution at high frequencies, and good frequency and poor time resolution at low frequencies. Figure 3.3 shows the same WT in Figure 3.2 from another angle to better illustrate the resolution properties: In Figure 3.3, lower scales (higher frequencies) have better scale resolution (narrower in scale, which means that it is less ambiguous what the exact value of the scale) which correspond to poorer

**frequency resolution :**Similarly, higher scales have scale frequency resolution (wider support in scale, which means it is more ambitious what the exact value of the scale is) , which correspond to better frequency resolution of lower frequencies.The axes in Figure 3.2and 3.3 are normalized and should be evaluated accordingly. Roughly speaking the 100 points in the translation axis correspond to 1000 ms, and the 150 points on the scale axis correspond to a frequency band of 40 Hz (the numbers on the translation and scale axis **do not correspond to seconds and Hz, respectively** , they are just the number of samples in the computation).

## 4. Discrete Wavelet Transforms

### 4.1.Need of Discrete Wavelet Transform

Although the DWT enables the computation of the continuous wavelet transform by computers, it is not a true discrete transform. As a matter of fact, the wavelet series is simply a sampled version of the CWT, and the information it provides is highly redundant as far as the reconstruction of the signal is concerned. This redundancy, on the other hand, requires a significant amount of computation time and resources. The discrete wavelet transform (DWT), on the other hand, provides sufficient information both for analysis and synthesis of the original signal, with a significant reduction in the computation time.The DWT is considerably easier to implement when compared to the CWT.

**4.2. Discrete wavelet transforms (DWT):** The foundations of the DWT go back to 1976 when Croiser, Esteban, and Galand devised a technique to decompose discrete time signals. Crochiere, Weber, and Flanagan did a similar work on coding of speech signals in the same year. They named their analysis scheme as **sub band coding**. In 1983, Burt defined a technique very similar to sub band coding and named it **pyramidal coding** which is also known as multi resolution analysis. Later in 1989, Vetterli and Le Gall made some improvements to the sub band coding scheme, removing the existing redundancy in the pyramidal coding scheme. Sub band coding is explained below. A detailed coverage of the discrete wavelet transform and theory of multi resolution analysis can be found in a number of articles and books that are available on this topic, and it is beyond the scope of this tutorial.

**4.2.1The Sub band Coding and the Multi resolution Analysis:** The frequencies that are most prominent in the original signal will appear as high amplitudes in that region of the DWT signal that includes those particular frequencies. The difference of this transform from the Fourier transform is that the time localization of these frequencies will not be lost

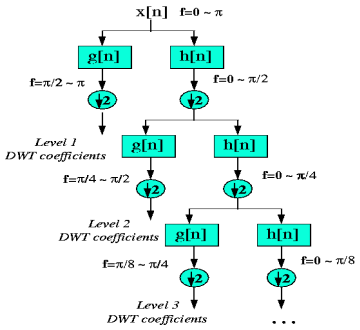


Fig 4.1. Bandwidth of the signal at every level

However, the time localization will have a resolution that depends on which level they appear. If the main information of the signal lies in the high frequencies, as happens most often, the time localization of these frequencies will be more precise, since they are characterized by more number of samples. If the main information lies only at very low frequencies, the time localization will not be very precise, since few samples are used to express signal at these frequencies. This procedure in effect offers a good time resolution at high frequencies, and good frequency resolution at low frequencies. Most practical signals encountered are of this type. The frequency bands that are not very prominent in the original signal will have very low amplitudes, and that part of the DWT signal can be discarded without any major loss of information, allowing data reduction. Figure 4.2 illustrates an example of how DWT signals look like and how data reduction is provided. Figure 4.2a shows a typical 512-sample signal that is normalized to unit amplitude. The horizontal axis is the number of samples, whereas the vertical axis is the normalized amplitude. Figure 4.2b shows the 8 level DWT of the signal in Figure 4.2a. The last 256 samples in this signal correspond to the highest frequency band in the signal, the previous 128

samples correspond to the second highest frequency band and so on. It should be noted that only the first 64 samples, which correspond to lower frequencies of the analysis, carry relevant information and the rest of this signal has virtually no information. Therefore, all but the first 64 samples can be discarded without any loss of information. This is how DWT provides a very effective data reduction scheme.

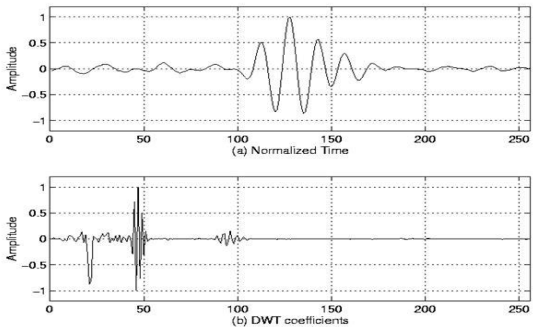


Fig 4.2 a,b Example of a DWT

One important property of the discrete wavelet transform is the relationship between the impulse responses of the high pass and low pass filters. The high pass and low pass filters are not independent of each other, and they are related by

$$g[L-1-n] = (-1)n.h[n]$$

Where g[n] is the high pass, h[n] is the low pass filter, and L is the filter length (in number of points). Note that the two filters are odd index alternated reversed versions of each other. Low pass to high pass conversion is provided by the $(-1)^n$ term. Filters satisfying this condition are commonly used in signal processing, and they are known as the Quadrature Mirror Filters (QMF). The two filtering and sub sampling operations can be expressed by

$$y_{high}[k] = \sum_n x[n].g[-n+2k]$$
$$y_{low}[k] = \sum_n x[n].h[-n+2k]$$

The reconstruction in this case is very easy since half band filters form orthonormal bases. The above procedure is followed in reverse order for the reconstruction. The signals at every level are up sampled by two, passed through the synthesis filters g'[n], and h'[n] (high pass and low pass, respectively), and then added. The interesting point here is that the analysis and synthesis filters are identical to each other, except for a time reversal. Therefore, the reconstruction formula becomes (for each layer)

$$x[n] = \sum_{k=-\infty}^{\infty} (y_{high}[k].g[-n+2k]) + (y_{low}[k].h[-n+2k])$$

However, if the filters are not ideal half band, then perfect reconstruction cannot be achieved. Although it is not possible to realize ideal filters, under certain conditions it is possible to find filters that provide perfect reconstruction. The most famous ones are the ones developed by Ingrid Daubechies, and they are known as Daubechies' wavelets. Note that due to successive sub sampling by 2, the signal length must be a power of 2, or at least a multiple of power of 2, in order this scheme to be efficient. The length of the signal determines the number of levels that the signal can be decomposed to. For example, if the signal length is 1024, ten levels of decomposition are possible. Interpreting the DWT coefficients can sometimes be rather difficult because the way DWT coefficients are presented is rather peculiar. To make a real long story real short, DWT coefficients of each level are concatenated, starting with the last level. An example is in order to make this concept clear:Suppose we have a 256-sample long signal sampled at 10 MHZ and we wish to obtain its DWT coefficients. Since the signal is sampled at 10 MHz, the highest frequency component that exists in the signal is 5 MHz. At the first level, the signal is passed through the low pass filter h[n], and the high pass filter g[n], the outputs of which are sub sampled by two. The high pass filter output is the first level DWT coefficients. There are 128 of them, and they represent the signal in the [2.5 5] MHz range. These 128 samples are the last 128 samples plotted. The low pass filter output, which also has 128 samples, but spanning the frequency band of [0 2.5] MHz, are further decomposed by passing them through the same h[n]

And g[n]. The output of the second high pass filter is the level 2 DWT coefficients and these 64 samples precede the 128 level 1 coefficients in the plot. The output of the second low pass filter is further decomposed, once again by passing it through the filters h[n] and g[n]. The output of the third high pass filter is the level 3 DWT coefficients. These 32 samples precede the level 2 DWT coefficients in the plot. The procedure continues until only 1 DWT coefficient can be computed at level 9. This one coefficient is the first to be plotted in the DWT plot. This is followed by 2 level 8 coefficients, 4 level 7 coefficients, 8 level 6 coefficients, 16 level 5 coefficients, 32 level 4 coefficients, 64 level 3 coefficients, 128 level 2 coefficients and finally 256 level 1 coefficients. Note that less and less number of samples is used at lower frequencies, therefore, the time resolution decreases as frequency decreases, but since the frequency interval also decreases at low frequencies, the frequency resolution increases. Obviously, the first few coefficients would not carry a whole lot of information, simply due to greatly reduced time resolution. To illustrate this richly bizarre DWT representation let us take a look at a real world signal. Our original signal is a 256-sample long ultrasonic signal, which was sampled at 25 MHz. This signal was originally generated by using a 2.25 MHz transducer; therefore the main spectral component of the signal is at 2.25 MHz. The last 128 samples correspond to [6.25 12.5] MHz range. As seen from the plot, no information is available here; hence these samples can be discarded without any loss of information. The preceding 64 samples represent the signal in the [3.12 6.25] MHz range, which also does not carry any significant information. The little glitches probably correspond to the high frequency noise in the signal. The preceding 32 samples represent the signal in the [1.5 3.1] MHz range.As you can see, the majority of the signal's energy is focused in these 32 samples, as we expected to see. The previous 16 samples correspond to [0.75 1.5] MHz and the peaks that are seen at this level probably represent the lower frequency envelope of the signal. The previous samples probably do not carry any other significant information. It is safe to say that we can get by with the 3rd and 4th level coefficients, that are we can represent this 256 sample long signal with 16+32=48 samples, a significant data reduction which would make your computer quite happy.

One area that has benefited the most from this particular property of the wavelet transforms is image processing. As you may well know, images, particularly high-resolution images, claim a lot of disk space. As a matter of fact, if this tutorial is taking a long time to download, that is mostly because of the images. DWT can be used to reduce the image size without losing much of the resolution. Here is how: For a given image, you can compute the DWT of, say each row, and discard all values in the DWT that are less then a certain threshold. We then save only those DWT coefficients that are above the threshold for each row, and when we need to reconstruct the original image, we simply pad each row with as many zeros as the number of discarded coefficients, and use the inverse DWT to reconstruct each row of the original image. We can also analyze the image at different frequency bands, and reconstruct the original image by using only the coefficients that are of a particular band. I will try to put sample images hopefully soon, to illustrate this point.Another issue that is receiving more and more attention is carrying out the decomposition (sub band coding) not only on the low pass side but on both sides. In other words, zooming into both low and high frequency bands of the signal separately. This can be visualized as having both sides of the tree structure of Figure 4.1. What result is what is known as the wavelet packages we will not discuss wavelet packages in this here, since it is beyond the scope of this tutorial. Anyone who is interested in wavelet packages or more information on DWT can find this information in any of the numerous texts available in the market.And this concludes our mini series of wavelet tutorial. If I could be of any assistance to anyone struggling to understand the wavelets, I would consider the time and the effort that went into this tutorial well spent. I would like to remind that this tutorial is neither a complete nor a through coverage of the wavelet transforms. It is merely an overview of the concept of wavelets and it was intended to serve as a first reference for those who find the available texts on wavelets rather complicated. There might be many structural and/or technical mistakes, and I would appreciate if you could point those out to me. Your feedback is of utmost importance for the success of this tutorial.

## 5 Water Marking Schemes

### 5.1. the Channel Model of Canonical CDMA based Watermarking Schemes

Since discrete wavelet transform (DWT) is believed to more accurately models aspects of the Human Visual System (HVS) as compared to the FFT or DCT, watermark information are embedded in the wavelet domain for many CDMA based watermarking schemes. The host image is first transformed by orthogonal or bi orthogonal wavelets to obtain several sub band images (each sub band image consists of wavelet coefficients). Then some of them are selected for watermark embedding. Suppose sub band image I is chosen for watermark embedding and the message is represented in binary form $b = (b_1, b_2, \ldots\ldots b_L)$ where $b_i \in \{0,1\}$ we first transform b into a binary polar sequence m of $\{-1,1\}$ by the following formula

$$m_i = 1 - 2b_i, \qquad \text{i=1, 2, .....L.} \qquad (1)$$

According to the CDMA principles, the message m is encoded by L uncorrelated pseudo sequences $\{s_1, s_2, \ldots\ldots\ldots s_L\}$ generated by a secrete key, such as m sequences, gold sequences, etc.. Since it is possible to make them orthogonal with each other, we simply assume that they are orthogonal unit vectors, i.e,

$$\langle s_i, s_j \rangle = \delta = \begin{cases} 0, i \neq j, \\ 1, i = j. \end{cases} \quad \text{i, j=1,2,…….,L.} \qquad (2)$$

Where, <•, •> denotes inner product operation. The pseudorandom noise pattern W is obtained as follows

$$W = \sum_{i=1}^{L} m_i s_i, \qquad (3)$$

This submerges the watermark message. Then the pseudorandom noise pattern W is embedded into the sub band image I as follows

$$I_w = I + \lambda W, \qquad (4)$$

Where $\lambda$ is a positive number, called the water mark strength parameter. Then an inverse wavelet transform is performed to obtain the water marked image.

In the water marked extracting phase, the water marked image is transformed by the same wavelet transform that is used in the watermark embedding phase to obtain the sub band image $\hat{I}_w$ that contains the watermark message, i.e.,

$$\hat{I}_w = I + \lambda W + n, \qquad (5)$$

Where n is the distortion due to attacks or simply quantization errors if no other attack is performed. Then the orthogonal pseudo sequences $\{s_1, s_2, \ldots\ldots\ldots s_L\}$ are generated using the key, and the inner product between each $s_i$ and $\hat{I}_w$ is computed:

$$\langle s_i, \hat{I}_w \rangle = \langle s_i, I \rangle + \lambda m_i + \langle s_i, n \rangle \qquad (6)$$

The canonical CDMA based methods decide the sign of m i by computing the inner product on the left most of (6), i.e.,

$$\hat{m}_i = \begin{cases} 1, if \langle s_i, \hat{I}_w \rangle > 0, \\ -1, otherwise. \end{cases} \qquad (7)$$

Where $\hat{m}_i$ denotes the estimated value of $\hat{m}_i$ This equivalent to neglecting of correlation between $s_i$ and the host image I, and the host image I , and the correlation between $s_i$ and the attack distortion n . When the message size is small, we can take a large watermark strength parameter λ, so we have no problem to neglect those small values. But when the message size is large, problem occurs. For the convenience of analysis, we ignore the third term in (6) at present. Then we have

$$\langle s_i, \hat{I}_w \rangle \approx \langle s_i, I \rangle + \lambda m_i . \qquad (8)$$

As the message size increases, the watermark strength parameter λ becomes smaller and smaller in order to keep the imperceptibility. So the influence of the host image's contents becomes more and more prominent as the message size increases. Experimental results also confirm this fact. So we must find a way to eliminate or reduce the interference of the host image so that we can improve the robustness of the CDMA watermarking scheme considerably.

**5.2. High Capacity CDMA Watermarking Scheme**: In the previous subsection we have analyzed, the influence of the host image's content to the robustness of the canonical CDMA watermarking schemes. In order to eliminate this influence, we project the host image onto the linear subspace S generated by the orthogonal pseudorandom sequences, i.e.

$$P_s(I) = \sum_{i=1}^{L} \langle s_i, I \rangle s_i. \qquad (9)$$

If we keep the projection coefficients $\{c_i = \langle s_i, I \rangle : i = 1,2,\ldots\ldots,L\}$ as a secret key, then we can subtract $P_s(I)$ from the watermarked sub band image I, Before watermark extraction, therefore, we can decide the sign of $\hat{m}_i$ by computing

$$\langle s_i, \hat{I}_w - P_s(I) \rangle \approx \langle s_i, I + \lambda W - P_s(I) \rangle \quad = \lambda \langle s_i, W \rangle = \lambda m_i, \qquad (10)$$

Which is not affected by the host image's contents, and therefore, provides a more robust way for CDMA based watermarking.

**5.2.1.Watermark Embedding Process:**The watermark embedding process of the proposed high capacity CDMA scheme is the same as the canonical one except for a preprocessing step of calculating the projection coefficients $\{c_i = \langle si, I \rangle : i = 1,2.....L\}$, which should be kept as a key for watermark extraction. Fig. 1 gives the flow chart of the watermark embedding process.

Here we give the watermark embedding steps:

**Step 1:** decompose the host image into sub band images using orthogonal or bi orthogonal discrete wavelet transform (DWT), and chose one or several sub band images I for watermark embedding;

**Step2:** generate the orthogonal pseudorandom sequences $\{s_1, s_2, ..... s_L\}$ using the secret key (key1);

**Step3:** project the sub band images I onto the linear subspace S generated by the orthogonal pseudo sequences, and keep the projection coefficients $\{c_i = \langle si, I \rangle : i = 1,2.....L\}$ as the second secret key (key2) which will be used in the watermark extraction phase;

**Step4:** encode the watermark information using formula (1) and (3) to get the pseudorandom noise pattern W;

**Step5:** embed the pseudorandom noise pattern W into the sub band image I using formula (4);

**Step6:** perform inverse discrete wavelet transform (IDWT) to obtain the watermarked image.

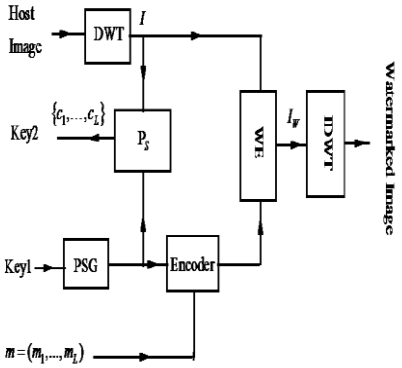


Fig 5.1 the watermark embedding process of the proposed scheme.

Key1 is the key used to generate the orthogonal pseudo sequences; PSG is the pseudo sequence generator; P*S* is the orthogonal projection operator; Key2 is generated by the projection operator, which consists of the projection coefficients, will be used in the watermark extraction phase; DWT denotes the discrete wavelet transform; WE denotes watermark embedding; IDWT denotes inverse wavelet transform.

**Watermark Extraction Process:** Now we give the watermark extraction steps:

**Step1:** decompose the received image into sub band images using the same wavelet transform as the one used in the watermark embedding phase, and choose the corresponding sub band images $\hat{I}_w$ for watermark extraction;

**Step2:** generate the orthogonal pseudorandom sequences $\{s_1, s_2, ..... s_L\}$ using the secrete key (key1);

**Step3:** eliminate the projection component from $\hat{I}_w$ by

$$\tilde{I}_w = \hat{I}_w - P_s(I) = \hat{I}_w - \sum_{j=1}^{L} c_j s_j \qquad (11)$$

Where Ci are the projection coefficients kept in the second secret key (key2);

**Step4:** extract the embedded message $m = (m_1, m_2, .....m_L)$ by correlation detection

$$\hat{m}_i = \begin{cases} 1, if \langle s_i, \tilde{I}_w \rangle > 0, \\ -1, otherwise \end{cases} \qquad (12)$$

Step5: transform the extracted message $m = (m_1, m_2, .....m_L)$ into the original watermark $b = (b_1, b_2, .....b_L)$ by

$$b_i = (1 - m_i) \div 2, i = 1,2, .....L \qquad (13)$$

## 6. Performance Test

We have performed a series of experiments to test the robustness of the proposed scheme. Seven 512x512 grayscale images (a. airplane, b. baboon, c. Barbara, d. boats, e. gold hill, f.Lena, g. pepper.) are chosen as test images. The watermarks are binary sequences of different size. The pseudorandom sequences we used are generated by pseudorandom number generators and we orthogonalize them by Cholesky decomposition method. Of course other choices of pseudo sequences such as m sequences, gold sequences may be more suitable for watermarking; we will test them in the future.

### A. Capacity VS Bit Error Rate (BER)

The first test we have performed is to test the relationship between message capacity and the bit error rate of the extracted watermark for both the canonical and newly proposed schemes. The bit error rate (BER) is calculated by the following formula:

$$BER = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left|W(i,j) - EXW(i,j)\right| \qquad (14)$$

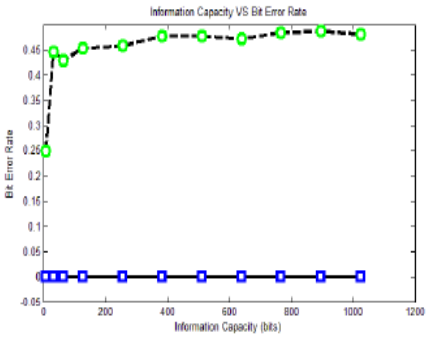


Fig 5.2 the relationship between message capacity and the bit error rate

Where W denotes the original watermark, ExW denotes the extracted watermark. In this test, we embed the watermarks into the lower resolution approximation image (LL) of the 2- level biorthogonal discrete wavelet decomposition of the test image using both canonical and the newly proposed CDMA based schemes, no attack is performed on the watermarked image except for quantization errors. Then extract watermarks from the watermarked image using corresponding watermark extraction schemes and compare the extracted watermark with the original one. The watermark size (number of information bits) vary from 16 to 1015, we have chosen 11 discrete values for our test. For each watermark size value, we perform the watermark embedding and extracting process on all 7 test images, and calculate the average BER. In the whole test we carefully adjust the watermark strength parameter λ so that the peak signal to noise ratio (PSNR) of the watermarked image take approximately the same value for different watermark sizes and different test images. Fig. 2 gives the experimental results. The horizontal axis indicates the information capacity, i.e., the number of bits embedded in the test image. The vertical axis indicates the average BER. From fig. 2 we see that as the information capacity increases the BER of the canonical CDMA based scheme increases and approaches to 0.5. But for the proposed scheme, the bit error rate keeps to be zero until the message capacity takes the value of 1024 bits. Of course, if the message capacity keeps on increasing, the bit error rate cannot always be zero, it will increase and approach to 0.5 in the long run. On the hand, for the canonical scheme, if the message size is large, the bit error rate is high even no attack is performed on the watermarked image. This phenomenon has not taken place in the tests for the proposed scheme yet. The reason is that the interference of the correlations between the test image and the pseudorandom sequences used for encoding the watermark message is cancelled in the proposed scheme. Fig. 2 also shows that the proposed scheme has higher information capacity than the canonical CDMA based watermarking scheme when no attack other than quantization errors is performed.

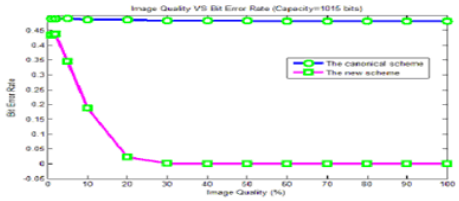


Fig 5.3 Image quality VS BER for JPEG attacks of different attack intensity

**B. Robustness to Noising Attacks:** The second test is to test the robustness to noising attacks of both schemes. In this test, we first generate binary watermarks of capacity 128, 256, 512 and 1015 bits, then embed them into the 7 test images using both watermark embedding schemes to generate 14 watermarked images, and then add Gaussian noise of different intensity to the watermarked images to generate the noising attacked images, then extract watermarks from those attacked images using corresponding watermark extraction scheme. The intensity of noising attack is measured by noise Rate RI , i.e.,

$$RI = \frac{\sigma}{R}, \qquad (15)$$

Where σ is the standard deviation of the noise,R is the range the pixel values of the image I , i.e., $R = \max_{x,y} I(x,y) - \min_{x,y} I(x,y).$ (16)

We have added Gaussian noise with RI vary from 0.05 to 0.5 and calculated the average BER of the extracted watermark for each RI value and each value of watermark capacity. Fig. 3 gives the BER-RI plot with watermark capacity=1015, 512, 256,128. We see that BER of the new scheme is much smaller than the one of the canonical scheme.

**C. Robustness to JPEG Attacks** :The third test is to test the robustness to JPEG attacks of both schemes. In this test, we compress the watermarked images using JPEG compressor (JPEG imager v2.1) with quality factors vary from100% to 1% before watermark extraction. Fig. 4 shows the BER of both schemes under JPEG compression attacks with different quality factors. The horizontal axis indicates the quality factor that measures the extent of lossy JPEG compression, the smaller the quality factor, the higher the compression extent. From fig. 4 we see that the proposed scheme is highly robust to JPEG compression.

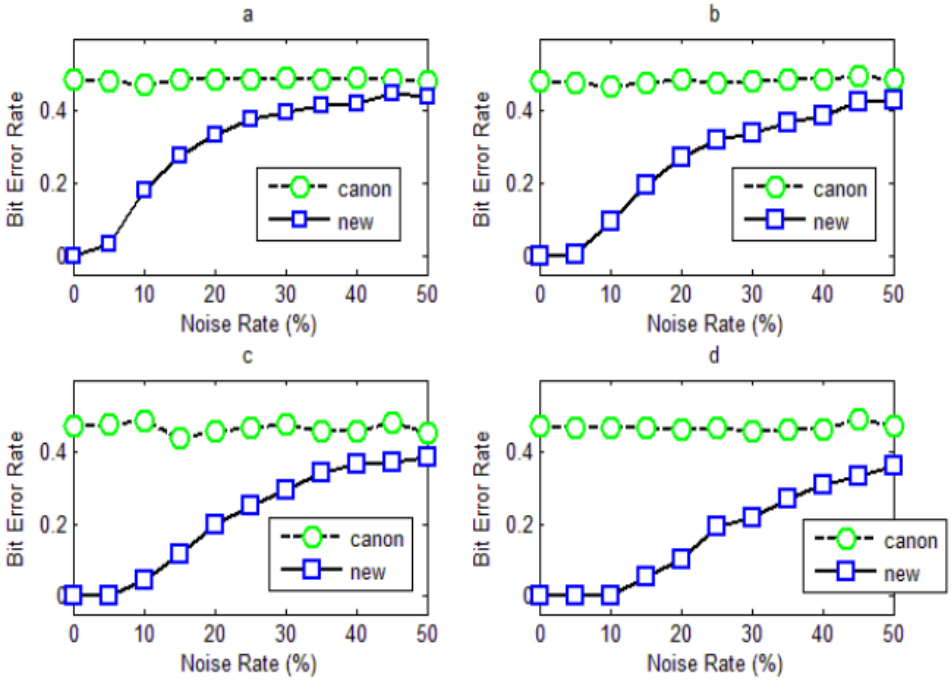


Fig 5.4 BER-RI plot with different values of watermark capacity.
a. watermark capacity=1015; b. watermark capacity=512; c. watermark capacity=256; d. watermark capacity=128. 'canon' in the legend indicates the canonical scheme; 'new' indicates the new scheme.

**D. Robustness to other Attacks**

We test the robustness to median filtering and jitter attacks of both schemes. In the median filtering test, we filter the watermarked image using a 5x5 median filtering template before watermark extraction. In the jitter attack test, before watermark extraction, we first randomly drop a row and a column of the watermarked image, then randomly duplicate a row and a column to keep the image size unchanged. This attack can destroy the synchronization of the watermark, which often leads to the failure of watermark extraction for many existing watermarking schemes. The experimental data are list in table I. We see that the proposed scheme is robust to both attacks but the canonical scheme is not.

**7. Code:**

```
Clc;Clear all; Close all;
start_time=cputime;
i=uigetfile('.jpg','select the host image');
s0=imfinfo(i);
i=imread(i);
i=imresize(i,[256 256]);
imbytes=size(i,1)*size(i,2)*s0.BitDepth;
imshow(i),title('original image');
bpp=numel(i)/imbytes
if size(i,3)>1
i=rgb2gray(i);
end
g=im2double(i);
% canonical CDMA based watermarking%%%%%%
[LL LH HL HH]=dwt2(g,'haar',1);
I=LL;
lambda=0.1;     % set the gain factor for embeding
disp('Embedded Bits')
b=[1 1 1 1 1]
% b=randint(1,128);
L=length(b);
for i0=1:L
m(i0)=1-2.*b(i0);            %% eq-1
end
for i1=1:L
for j1=1:L
if i1==j1
s(i1,j1)=1;
else                 %% eq-2(key-1)
s(i1,j1)=0;
end   end      end
for i2=1:L
W(i2)=sum(m(i2).*s(i2));    %% eq-3
end
W=imresize(W,size(I));
%W=round(2*(rand(128,128)-0.5));
iw=I+lambda.*W;                %% eq-4
IW=idwt2(iw,LH,HL,HH,'haar');
imwrite(IW,'watermarked.jpg')
figure,imshow(IW,[]);title('watermarked image')
n=randn(size(I));
IW1=I+lambda.*W+n;             %% eq-5
iss=s(1:L).*I(1:L)+lambda.*m+s(1:L).*n(1:L);  %%eq-6
%iss=ceil(iss);
for i3=1:length(iss)
if iss(i3) > 0
m1(i3)=1;          %%eq-7
else
m1(i3)=-1;
end
end
%-----------Proposed methodology----------%%%
for i4=1:L
P(i4)=sum((s(i4).*I(i4)).*s(i4));  %% eq-9
end
```

```
for i5=1:L
   c(i5)=s(i5).*I(i5);  %%------key-2------------------------
end
I1=imread('watermarked.jpg');
I1=im2double(I1);
A=input('Select Attack \n (1) Gussian Noise \n (2) Salt & Pepper Nose \n (3) JPEG Compression          : ');
switch (A)
   Case 1
      WI=imnoise (I1,'gaussian', 0.01);
      PSNR_Attack=psnr (I1, WI)
      BER=biter (I1, WI)
   Case 2
      WI=imnoise (I1,'salt & pepper', 0.02);
      PSNR_Attack=psnr (I1, WI)
      BER=biter (I1, WI)
   Case 3
      T = dctmtx (8);
      B = blkproc (I1,[8 8],'P1*x*P2',T,T');
      Mask= [1   1   1   1   0   0   0   0
            1   1   1   0   0   0   0   0
            1   1   0   0   0   0   0   0
            1   0   0   0   0   0   0   0
            0   0   0   0   0   0   0   0
            0   0   0   0   0   0   0   0
            0   0   0   0   0   0   0   0
            0   0   0   0   0   0   0   0];
      B2 = blkproc (B, [8 8],'P1.*x', mask);
      WI = blkproc (B2, [8 8],'P1*x*P2', T', T);
    PSNR_Attack=psnr (I1, WI)
    BER=biter (I1, WI)
end
figure, imshow(WI); title('Attacked Watermarked Image');
L1=medfilt2 (WI, [3 3]);
figure, imshow(L1); title('De-noised Image');
Dim=imsubtract (WI, L1);
figure, imshow(Dim); title('Difference Image');
[LL1 LH1 HL1 HH1]=dwt2 (WI,'haar', 1);
IW2=LL1;
IW2=IW2 (1: L)-P;          %% eq-11
for i6=1:L
   isss (i6)=s(i6).*IW2(i6);
end
for i7=1:length(isss)
    if isss(i7) > 0
       m2 (i7) =1;         %%eq-7
    else
       m2(i7)=-1;
    end ;end
disp('Received Bits')
 b1=(1-m2)./2   %% received bits
 % display processing time
elapsed_time=cputime-start_time
sigma=0.05;
R=max(max(I))-min(min(I));
RI=sigma/R;
Q=[0:10:100];
```

```
for jj=1:length(Q)
   mnn=jpogcomp(i,Q(jj));
   brr(jj)=biter( double(i),mnn);
end
br=[1:4];          br1=[5:11];        brr=[br/0.45 br1/0.45];
figure,plot(Q,1./(brr),'k*-')
xlabel('image quality');
ylabel('--->BER');title('image quality Vs BER ');
cap=[0 10 20 30 40 50 100 150 200 250 300 350 400 450 500 550 600];
ber1= [0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066 0.066];
figure, plot(cap,ber1,'--rs');
title ('Information capacity VS Bit Error Rate');
xlabel ('Information capacity (no.of.bits)');
ylabel ('Bit Error Rate');
```

## 8.Comparisions:

| Watermarking scheme | Median Filtering | | Jitter Attack | |
|---|---|---|---|---|
| | PSNR(db) | BER | PSNR(db) | BER |
| The canonical Scheme | 43.9731 | 0.4732 | 44.0650 | 0.4836 |
| The Proposed Scheme | 44.0235 | 0.0856 | 44.0167 | 0.0664 |

## 9. Conclusions

In this paper, we propose a high-capacity CDMA based watermarking scheme based on orthogonal pseudorandom sequence subspace projection. The proposed scheme eliminates the interference of the host image in the watermark extraction phase by subtracting the projection components (on the linear subspace generated by the pseudorandom sequences) from the host image. So it is more robust than the canonical CDMA based scheme. We analyzed and test the performance of the proposed scheme under different attack conditions and compared with the canonical CDMA based scheme. We find that the proposed scheme shoes higher robustness than the canonical scheme under different attack conditions. The expense of high robustness is that an additional key that consists of projection coefficients is needed for the water mark extraction. But this additional memory cost is worthwhile in many situations since it improves both robustness and security of the watermarking system. In the near future we will analyze and test the proposed scheme intensively and use it to design watermarking systems resistant to geometrical attacks and print-and-scan attacks.

## References


[1] Santi P. Maity, Malay K. Kundu, "A Blind CDMA Image Watermarking Scheme In Wavelet Domain," International Conference on Image Processing, vol. 4, Oct. 2004, pp. 2633-2636.

[2] J. K. Joseph, Ò. Ruanaidh, P. Thierry, "Rotation, Scale and Translation Invariant Spread Spectrum Digital Image Watermarking", Signal Processing. Vol. 66(3), 1998, pp. 303-317

[3] C. G. M. Silvestre, W. J. Dowling, "Embedding Data in Digital Images Using CDMA Techniques". In: Proc. of IEEE Int. Conf. on Image Processing, Vancouver, Canada 1(2000)589-592

[4] T. Kohda, Y. Ookubo, K. Shinokura, "Digital Watermarking Through CDMA Channels Using Spread Spectrum Techniques". In: IEEE 6th Int. Sym. on Spread Spectrum Techniques and Applications, Parsippany, NJ, USA, Vol. 2, 2000, pp. 671 –674

[5] B. Vassaux , P. Bas, J. M. Chassery, "A New CDMA Technique for Digital Image Watermarking Enhancing Capacity of Insertion and Robustness". In: Proc. of IEEE Int. Conf. on Image Processing, Thessalonica, Greece, Vol. 3, 2001, pp. 983 -986

[6] G. M. Bijan, "Exploring CDMA for Watermarking of Digital Video". Villanova University, Villanova, PA, USA, 1985, http://www.ece.villanova.edu/~mobasser/mypage/3657-10.pdf.

[7] G. M. Bijan, "Direct Sequence Watermarking of Digital Video Using m- frames". In: Proc. Of IEEE Int. Conf. on Image Processing, Chicago, Illinois, USA, Vol. 2(1998) 399 -403.

[8] L. Xie and G. R. Arce, "Joint wavelet compression and authentication watermarking," in Proc. IEEE ICIP, Chicago, USA, vol. 2, pp. 427–431, Oct., 1998.

[9] H. Inoue, A. Miyazaki, A. Yamamoto, and T. Katsura, "A digital watermark based on the wavelet transform and its robustness on image compression," in Proc. IEEE ICIP, Chicago, USA, vol. 2, pp. 391–395, Oct., 1998.
[10] Jong Ryul Kim and Young Shik Moon, A robust wavelet-based digital watermark using level-adaptive thresholding, in: Proc. ICIP, Kobe, Japan, pp.202-212, Oct. 1999.
[11] G. Langelaar, I. Setyawan, R.L. Lagendijk, Watermarking Digital Image and Video Data , in IEEE Signal Processing Magazine, Vol 17, pp. 20-43, September 2000.
[12] M. Hsieh, D. Tseng, and Y. Huang, "Hiding digital watermarks using multiresolution wavelet transform," IEEE Trans. on Indust. Elect. vol. 48, pp. 875–882, Oct. 2001.
[13] Y. Seo, M. Kim, H. Park, H. Jung, H. Chung, Y. Huh, and J. Lee, "A secure watermarking for jpeg-2000," in Proc. IEEE ICIP, 2001, Thessaloniki, Greece, vol. 2, pp. 530-533, Oct. 2001.
[14] P. Su, H. M. Wang, and C. J. Kuo, "An integrated approach to image watermarking and jpeg-2000 compression," J. VLSI Signal Processing, vol. 27, pp. 35–53, Jan. 2001.
[15] P. Meerwald, A.Uhl, "A Survey of Wavelet-Domain Watermarking Algorithms", in: Proc. of SPIE, Security and Watermarking of Multimedia Contents III, San Jose, USA, Jose, CA, USA, vol.4314, pp. 505-516, Jan. 2001.
[16] Santi P. Maity, Malay K. Kundu, "A Blind CDMA Image Watermarking Scheme In Wavelet Domain," in: International Conference on Image processing, vol. 4, Oct. 2004, pp. 2633-2636.
[17] O. Alkin and H. Caglar, "Design of efficient m-band coders with linear phase and perfect reconstruction properties," IEEE Trans. Signal Processing, vol. 43, pp. 1579-1590, 1995.
[18] P. Steffen, P. N. Heller, R. A. Gopinath, and C. S. Burrus, "Theory of regular M-band wavelet bases," IEEE Trans. Signal Process., vol. 41, no. 12, pp. 3497–3510, Dec. 1993.
[19] Y. Fang, N. Bi, D. Huang, and J. Huang, "The m-band wavelets in image watermarking," in Proc. IEEE ICIP, 2005, vol. 1, pp. 245–248.

**Authors Brief Profile:**

Dr K Rameshbabu[1]

Dr K.RameshBabu professor, ECE Hyderabad Institute of Technology And Management, Hyderabad, holding B.E (ece),M.E, PhD having 16+years of experience in electronics and communication Engineering area .he is member in ISTE,IEEE & Java Certified Programmer(2,0)PGDST holder. he has lot of experience in academics and industrial related real time projects. He is paper setter for many autonomous universities and visiting professor for many areas like image processing, electron devices etc. he can be reached at: dr.kprb.ece@gmail.com

vani.kasireddy[2]

Vani.kasireddy is Good student, pursuing M.Tech vlsisd from ECE dept, Hitam.she is active member, in technical education along the student related motivation activitities.lot of interest in research areas related to, image processing digital Electronics, trouble shooting hardware kits etc. she can reach at: vanisri234@gmail.com