CHAPTER 4

Hardware Implementation of Image and Video Watermarking for Ownership Verification

Amit Joshi, Monica Bapna, Aniruddh Malpani, Ashwini Kumar Goyal
and Manisha Meena

The growth of multimedia technology has lead to easier and faster ways of communication. At the same time, it has broadened up the gap between security and

Amit Joshi
Malaviya National Institute of Technology
Jaipur,30201,India
e-mail: amjoshi.ece@mnit.ac.in

Monica Bapna
Physics Department, IIT Bombay
Bombay, India
e-mail: monica.bapna13@gmail.com

Aniruddh Malpani
Robert Bosch Engineering and Business Solutions Limited

Ashwini Kumar Goyal
Oracle Financial Software Services PVT. LTD.

Manisha Meena
T & IE, Pipeline Division, Indian Oil Corporation Ltd
Rajkot, India

Editors: C.W. Ahn, M. Ali, M. Pant, *Embodying Intelligence in Multimedia Data Hiding*
DOI: 10.15579/gcsr.vol5.ch4, GCSR Vol. 5, pp. 59-82, 2016

transmission of various multimedia objects. Video and image are the most popular shared objects over the open network through various portable devices. Digital watermarking is the data hiding technique where the owner's content is inserted into the original content (i.e. image and video) for ownership verification. The process of embedding the watermark in real time watermark has more significance because there is no delay between the capturing of image/video and the insertion of the watermark. The hardware implementation is an ideal choice of real time watermarking since it outperforms the software implementation in terms of area, power and speed. In this chapter, two different algorithms are proposed, one is based on a variable block size concept and the other uses AC prediction technique. The performance of the algorithms is calculated and compared with similar previous work. The algorithms are prototyped on the FPGA to validate the real time performance of hardware.

4.1 Introduction

With the enhancement of technology, various multimedia objects such as video, image, audio and speech are exchanged rapidly once they have been captured through various handheld devices [2]. Image and video are the most popular multimedia objects which are shared easily over the network. The major concern is the integrity after the data has been transmitted over the channel. With the transition from analog to digital technology, the process of editing and duplication of the original content has become easier [3]. The digital watermarking is a data hiding technique which provides legitimacy of an original content. There is a wide range of applications of digital watermarking which includes copy control, transaction tracing with fingerprinting, broadcast monitoring, tamper proofing, ownership identification etc. [20]. The watermarking algorithm should have some fundamental features for some specific application which are mainly imperceptibility, robustness, security, payload capacity and blind/non blind retrieval of the watermark [22, 40].

Over the years, many data hiding techniques have evolved for security [14]. In *steganography* technique, the message is hidden in the image/video or other media file in an unnoticeable manner [12]. *Digital watermarking* is a similar technique, but differs with the intent of security along with authenticity. In the digital watermarking method, the owner's identity is embedded inside any multimedia object and the idea is to have ownership of the originator. *Cryptography* is also an ancient technique which covers the message's content, but it does not hide the existence of the message [42]. On the other hand, steganography and watermarking are used to hide the message itself. Once the cover message is decrypted, then the cryptography is no more secure while steganography and watermarking require the knowledge of the extraction process. In watermarking algorithm, the cover media is altered with the help of a secret message for proof of ownership. There may be chances where the parties who have access to the cover media may try to remove it. The watermark algorithm should contain an additional notion, there has to be a provision of resilience against attacks attempted by the intruder to eliminate the watermark. The watermark should always stay together with cover after being inserted. As a contrast to cryptography, watermarks are used to protect the content, even after the process of decoding [21].

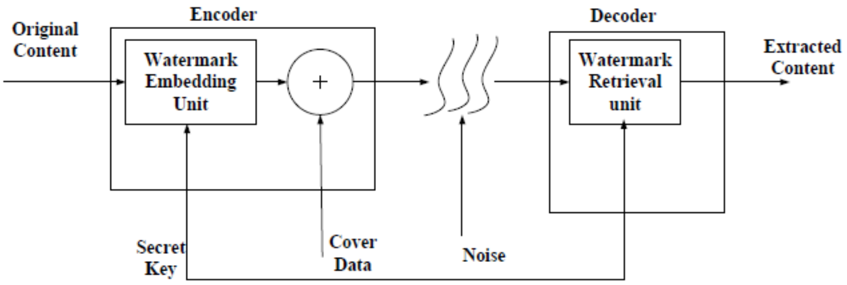


Figure 4.1: Block diagram of watermarking process.

The general block diagram of watermarking system is shown in Fig.(4.1). There are two essential units in the system, the first is watermark embedding unit and the second is a watermark retrieval unit. The watermark embedding unit (at encoder) helps to embed the watermark in the original object, whereas the watermark retrieval unit (at decoder) attempts to extract the watermark from the watermarked content. The object has to pass through a noisy channel between watermark embedding and retrieval unit [5]. The channel may experience severe attacks like distortion, random noise, compression, quantization, attenuation, any kind of tampering, frame dropping/swapping, frame adding etc. The robustness of the algorithm has to be verified against all such kinds of attacks. Normalized Correlation (NC) is measured to verify the robustness of the particular algorithm. The embedding process should not introduce the perceptual distortion in the original content. PSNR and MSE are calculated to check the visible effect on the watermarked content.

4.1.1 Key Issues in Watermarking

The process of embedding the watermark can be implemented on hardware or software platform. The hardware watermarking is a better approach because it helps to embed the watermark in real-time, which is considered to be more secure than software watermarking [40]. The software watermarking is an offline process where the original video/image is captured at the first stage and subsequently, the watermark is embedded in the second stage. The approach introduces the certain delay between capturing the object and inserting the watermark. This process is more vulnerable towards the attacks because there may be a case where the content is being attacked before the watermark is embedded. Hardware watermarking is more efficient than software watermarking because it has excellent performance in terms of speed, power and area [22]. There were very few attempts towards the real time watermark embedding on the hardware platform. In this chapter, our focus is to develop the image/video watermarking algorithm which can be efficiently portrayed in portable device for real time watermark embedding.

The digital watermarking has been applied to various signal processing applications. The algorithm should focus on human visual system and should able to use properties of signal transforms (such as Fourier and Discrete Cosine Transform (DCT)). The algorithm should sustain various forms of attacks (e.g. noise, compression, rotation, cropping). The algorithm includes some key features such as robustness, perceptual distortion and real time performance depending on the particular application. There is always a tradeoff among imperceptibility, payload and computational demand of an algorithm. Video is comprised of a sequence of frames (which may also be considered as still images) where some issues have to be focused before applying any watermarking algorithm. Firstly, the transmission of video requires huge bandwidth, therefore the watermarking algorithm has designed miserly for real time implementation. Second, the video compression algorithms are already computationally intensive and the inclusion of the watermarking algorithm further increases the computational demand. The process of inserting the watermark should be adapted for compression standard to be useful in commercial applications.

4.1.2 Video Coding and Watermarking

Video coding is an essential requirement for various video appliances such as mobile TV, internet video streaming, video conferencing, digital TV, Blue-Ray Discs and DVD etc. Video compression is the fundamental requirement for video coding in different video encoders/decoders and storage media. The encoder converts the original video in one of the compression format while the decoder helps to change back the compressed video into original format. Today, all video codecs (encoder/decoder) have been designed to have compatibility with all the latest compression standards. Presently, two main standard bodies, the International Standards Organization (ISO) and the International Telecommunications Union (ITU)) are mainly developing the various video standards [37]. JPEG is the most popular image coding standard developed by ISO for the storage of still images. MPEG is the commonly used video coding standard for DVD and digital television system [4]. Later on, ITU developed H.261 standard, which is useful for video telephony through ISDN. Afterwards, it has been upgraded to H.263 standard that is used for video communication for a wide range of high data rates application.

The H.264/AVC standard or MPEG-4 Part 10 is the latest video compression standard developed through the joint collaboration of two bodies, ITU-T Video Coding Experts Group (VCEG) and ISO/IEC JTC1 Moving Picture Experts Group (MPEG) [1]. It is the extension of MPEG-2 and MPEG-4 and provides better video compression efficiency [6]. H.264/AVC standard includes a wide range of applications such as video broadcasting, video on demand and online video streaming. H.264 encoder consisting of integer transformation, prediction modules and other relevant blocks have an efficient encoding for the generation of compressed bit stream. The decoder requires the reverse processes and inverse integer transform to reconstruct again the video sequence. By comparison to previous coding standards, H.264 coding defines the video stream as a syntax of the bit stream. H.264/AVC also supports various low bit rate applications such as mobile phones to high quality video requirement in HDTV and beyond.

4.2 Literature Survey

From the past few years, some substantial works have been done for data hiding which mainly focused on the hardware implementation for various applications. Cox et al. [15] explained the data hiding techniques such as watermarking, cryptography and steganography along with their applications. The algorithms for all data hiding techniques were discussed briefly. The importance of data hiding techniques and implementation strategies for different applications were also discussed briefly in the paper [28].

4.2.1 Software Watermarking

Song Han et al. [34] presented a brief survey on image watermarking. The survey includes the classification of various techniques according to their application domains. The advantages and disadvantages of different image watermarking techniques were summarized. Chowdhury et al. [13] proposed an efficient matrix based approach using spatial domain. The watermark is embedded with EX-OR operation of the pixels which are selected from rows and columns. The original image is processed with bit plane slicing module and LSB plane is selected for watermark embedding. The algorithm is blind and has no visible artifacts at the time of watermark embedding. The method has an excellent real time performance because it inserts the watermark in spatial domain. Potdar et al. [33] suggested another spatial domain method using a a gray level modification for secure communication. The algorithm embeds the invisible watermark by modification of pixel values of a gray level image. It has higher payload capacity and involves less computational complexity.

Chang et al. [9] developed an image watermarking scheme using the quantization concept for ownership application. The watermark is considered as binary logo and is embedded in high frequency and middle frequency band at the first decomposition level of DWT. The algorithm posses greater robustness against various attacks such as cropping, gamma correction, JPEG compression, JPEG2000, salt and pepper, resizing and rotation. Fu et al. [17] explained the novel DWT based digital watermarking method for image authentication. The random watermark is generated and then inserted in low frequency components of DWT. The proposed method is blind and has security against all the attacks including quantization. Wang and Pearmain [43] proposed blind image watermarking based on AC prediction. The concept of AC estimation was suggested with help of surrounding DC values. The watermark is inserted depending on the difference between estimated and original AC coefficients. Hung [18] described a robust watermarking system based on Integer DCT using adaptive AC prediction. The algorithm uses 4×4 Integer DCT transformation to reduce the blocking artifacts caused by 8×8 conventional DCT. The robustness of the algorithm is improved by introducing an error checking capability. Qi et al. [35] explained an adaptive image watermarking algorithm based on human visual system. A total mask is constructed with the help of edge masking, luminance masking and texture masking. The watermark is inserted in middle frequency values of DCT where highly texture regions are formed as an unnoticeable manner. Chaturvedi et al. [11] presented an algorithm based on the average value concept used to insert the watermark in the middle-band coefficients of DCT domain. The watermark is embedded at two different locations at AC(u1,v1) and AC(u2,v2) for every 8×8 block. These two coefficients are chosen for watermarking as they produce the identical quantization values in JPEG quantization table.

Jayamalar and Radha [19] provided different video watermarking technique in order to verify the robustness for ownership identification. The techniques were used to embed the robust watermark and also helped to maintain the fidelity of the original signal. Liu and Zhao [30] proposed two different schemes with key frame extraction from a video stream. The first method is useful for video segmentation using histogram matching method. The second method is based on the concept of extracting the various features from I-frame, P-frame and B-frame of the GOP. The performance of the algorithm is measured with fidelity and compression ratio. Cruz-Ramos et al. [16] developed a blind video watermarking algorithm where 2-D binary patterns are embedded such as company trademarks, and logo in wavelet co-efficient of video frames to have copyright protection. The scheme is robust against common video attacks and has low computational complexity. The algorithm was tested against different form of attacks as MPEG-2 Compression, frame dropping/swapping, collusion attacks and noise contamination. Raghavendra and Chetan [36] presented a robust and blind video watermarking scheme using scrambled watermark. The frame of the video is transformed using wavelet transform at the second level of decomposition. Then watermark is embedded in some selected DWT coefficient. The related work of hardware implementation is explained in the next section.

4.2.2 Hardware Watermarking

Wong [44] presented a watermarking method used to embed the watermark in Least Significant Bit (LSB) of each pixel of the image. The original image is separated in different blocks where LSB of each block is modified. Then, the watermark of each block is calculated with bit-wise EXOR operation using a hash function. The algorithm is extended by Brunton and Zhao [7] who developed the real time video watermarking system using Graphics Processing Units (GPUs). They had provided the solution for video authentication and tamper localization. Mathai et al. [31] designed the video watermarking scheme known as Just Another Watermarking Scheme (JAWS) which uses shifting of reference pattern. The generated watermark is scaled down by a constant factor and then subsequently inserted in each frame of the video. The frames are stored using dual port memory during the process of watermarking. Mohanty and Kougianos [32] illustrated the visible watermarking method for ownership verification applications. The method is useful for MPEG-4 compression that inserts the logo in a video stream. The hardware performance of the algorithm is also tested using Altera Cyclone II FPGA. Roy et al. [38] developed the high performance architecture for video authentication. MJPEG based video compression algorithm uses the concept of pipeline and parallel processing for the improvement of overall system performance. The algorithm posses an excellent robustness to withstand several potential attacks, including cropping and segment removing of the video sequence.

Joshi et al. [23] presented a video watermarking based on H.264 coding standard. The algorithm is based on Integer DCT where fully parallel and pipeline architectures were designed for better speed and area performance respectively. The algorithm uses the scene change detection concept for robustness against temporal attacks. Later on, the same algorithm is implemented on FPGA for real time ownership verification application [24]. The parts of the same watermark are inserted in various frames of a particular scene of the video to improve the performance. The algorithm is simulated by using MATLAB and then it is prototyped on the FPGA to verify the performance on hardware.

4.3 Proposed Image Watermarking based on Variable Block Size

The proposed algorithm uses the variable block size concept and is useful for ownership verification. The watermark is embedded in original image by replacing either minimum or maximum value of the block. The size of the block is defined by k and is adaptively selected during the watermark embedding [27]. Embedding can be done by two ways with overlapping blocks and non overlapping blocks. In non overlapping blocks, each block is treated as individual block, where the size of block is k which contains total n elements (1 to n elements). The next block contains $(n+1)$ to $2n$ elements. In overlapping blocks, each block contains 1 to n elements and the next block has 2 to $(n+1)$ elements. The size of the block is fixed while the $(n-1)$ elements are overlapped in each k size of the block. The block size is selected adaptively for different k and different values of k are $4, 6, 20, 50, 80$ and 100 considered for performance evaluation. There is always a tradeoff between payload and perceptual distortion. With increase of block size the payload is also enhanced, but it affects more visual distortion of watermarked image. Proposed watermarking is based on wavelet transform and is implemented using lifting based wavelet which is the simplest wavelet transformation method. The original image/frame is decomposed at the second level of the low frequency band (LL). If the watermark is embedded in the high frequency band then any signal processing attack is most likely to destroy the high frequencies or middle frequency components. This may lead to loss of the watermark. Therefore, it is advisable to insert the watermark in low frequency band to have more security against such type of attacks. The original watermark is considered as binary logo. The schemes include mainly three steps. In the first step, the original image/frame is preprocessed by the decomposition method. The image/frame is transformed in two level wavelet decomposition to embed the watermark. The second step is watermark embedding, where some DWT coefficients are altered as follows:

$$\begin{cases} C\left(i\right) = \max\left\{C\left(i\right), C\left(i+1\right), C\left(i+2\right), C\left(i+3\right), ..., C\left(i+k\right)\right\}, & if\, W\left(j\right) = 1 \\ C\left(i\right) = \min\left\{C\left(i\right), C\left(i+1\right), C\left(i+2\right), C\left(i+3\right), ..., C\left(i+k\right)\right\}, & otherwise \end{cases} \tag{4.1}$$

where $C\left(i\right)$ equals to the i-th wavelet coefficient of the LL band of the original frame/ image and $W\left(j\right)$ is the j-th pixel of the binary watermark which carries the sequence of ones and zeroes, while k is considered as variable block size which can be changed adaptively.

Once the process of watermark marking is completed, then inverse wavelet transform is applied to have the watermarked images/frames in third step. The extraction process of the algorithm is blind. The watermark image/frame is initially transformed with two levels of wavelet decomposition. Subsequently, the watermark is extracted as per the following condition:

$$\begin{cases} W(j) = 1, & if\, W_C(i) > median \\ & \{W_C(i), W_C(i+1), W_C(i+2), W_C(i+3), ..., W_C(i+k)\} \\ W(j) = 0, & otherwise \end{cases} \tag{4.2}$$

where W_C represents the watermarked coefficients.

4.3.1 Hardware Implementation

The block diagram of the proposed algorithm is developed as shown in Fig.(4.2). The original image is read through MATLAB, then generate the *.coe* file using *fprintf* in MATLAB. This file contains the pixel values of the image.

Now, the single port memory IP core is generated using Xilinx ISE and *.coe* file is loaded to the IP core. In the next step, the lifting based wavelet architecture converts spatial domain pixel values in the frequency domain coefficients. The low frequency coefficients are used for watermark embedding. The watermark is also processed in parallel for pre-processing before embedding. Same way, the pixel values of original watermark are loaded in single port memory IP core with help of *.coe* file. Now, as per the embedding process, the coefficients of the low frequency band are sorted in watermark embedding architecture. Eventually, one bit is inserted in the original image as per the embedding rule discussed in the above section. After the completion of watermarking, the watermarked coefficients are transformed by inverse lifting wavelet. These coefficients are again stored in internal memory. Now, the watermarked image is constructed using MATLAB with stored coefficients. The lifting architecture and watermark embedding architecture are discussed separately below.

4.3.1.1 Lifting Wavelet Architecture

The lifting based wavelet scheme is implemented with LeGall 5/3 method. LeGall 5/3 is an efficient approach because it is simple and lossless transformation. The odd and even samples values are computed as Eq.(4.3) and Eq.(4.4) below:

$$y(2n+1) = x(2n+1) - \left[\frac{x(2n) + x(2n+2)}{2}\right] \tag{4.3}$$

$$y(2n) = x(2n) + \left[\frac{y(2n-1) + y(2n+1) + 2}{4}\right] \tag{4.4}$$

The lifting process is mainly divided into two separate phases known as the *predict phase* and the *update phase*. Equation 4.3 represents the predict phase and Eq.(4.4) defines the update phase of the lifting transform. As denominator carries the value

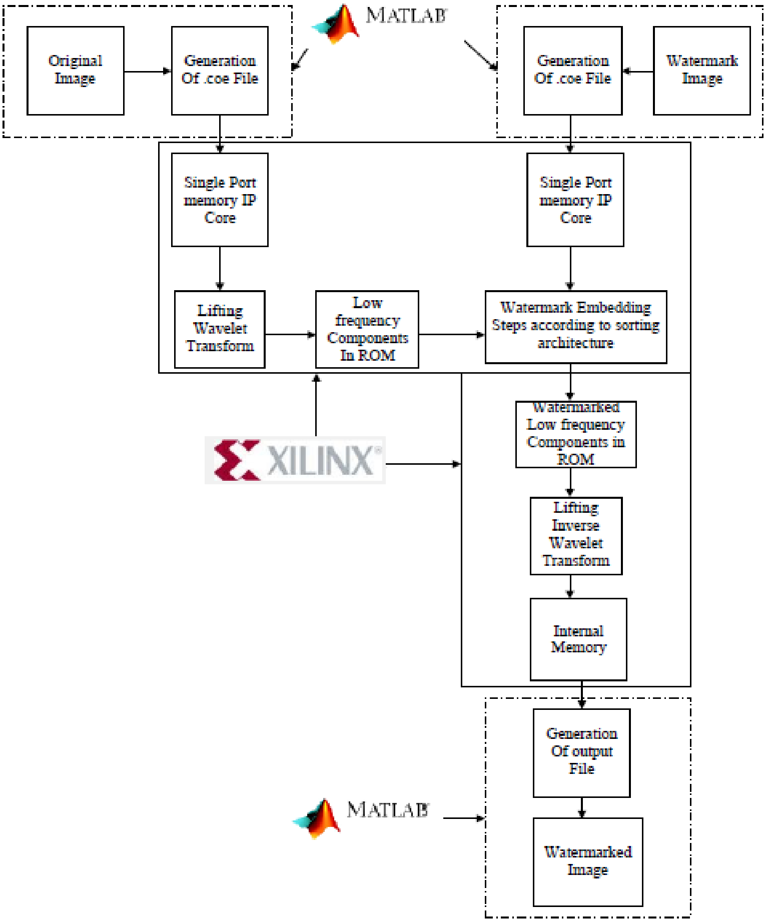

MATLAB
Original Image
Generation Of .coe File
Generation Of .coe File
Watermark Image
Single Port memory IP Core
Single Port memory IP Core
Lifting Wavelet Transform
Low frequency Components In ROM
Watermark Embedding Steps according to sorting architecture
XILINX
Watermarked Low frequency Components in ROM
Lifting Inverse Wavelet Transform
Internal Memory
Generation Of output File
MATLAB
Watermarked Image

Figure 4.2: Block diagram of the proposed algorithm.

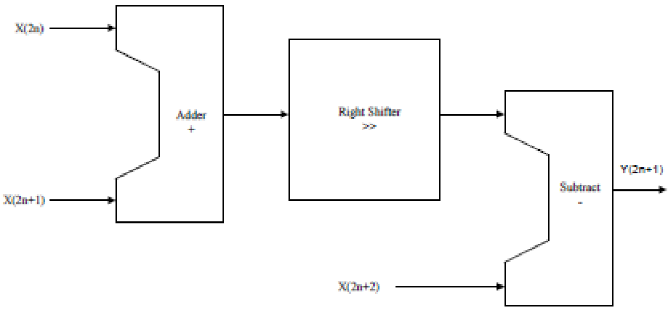


Figure 4.3: Predict phase of LeGall 5/3 lifting wavelet.

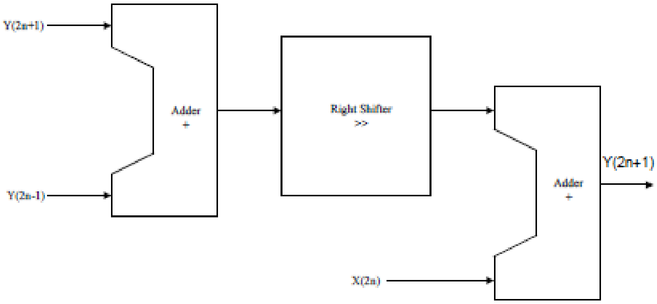


Figure 4.4: Update phase of LeGall 5/3 lifting wavelet.

in the form of 2^i, where i is an integer number, then divisor operation can easily be replaced by the right shift operation. Thus, the predict phase is implemented with the adder, shifter and subtraction modules only. The detailed architecture of predicting phase is shown in Fig.(4.3) below. The update phase is similarly implemented as per the architecture in Fig.(4.4).

4.3.1.2 Watermark Embedding Architecture

It includes the sorting of the values from the k size blocks (Fig.(4.5)). According to block size, the buffer is designed which can store k number of values. In order to sort the values, the comparator compares the values and stores them as per either ascending or descending order.

4.3.2 Results and Analysis

The performance of the proposed algorithm is verified on MATLAB platform. The original RGB image is considered as *"Lena"* of 512×512 pixels size and the watermark is a binary logo of 40×40 pixels size, which are shown in Fig.(4.6a) and Fig.(4.6b), respectively.

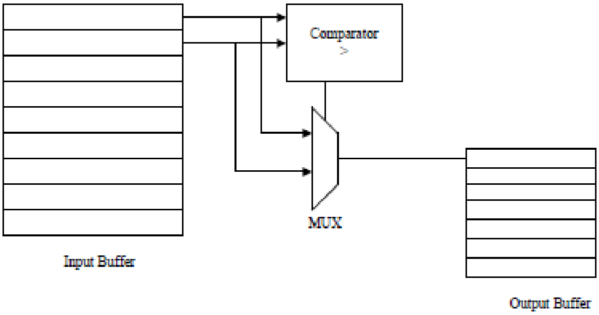


Figure 4.5: Watermark embedding with sorting architecture.

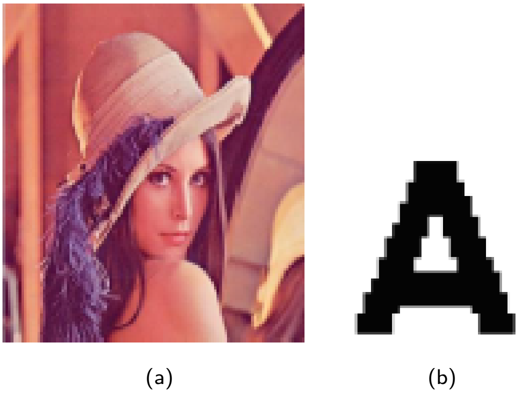


Figure 4.6: Original test data set: (a) image and (b) watermark.

There are various quality measurement parameters such as MSE, PSNR and NC have been calculated [25]. MSE is calculated as follows:

$$MSE = \frac{1}{X \times Y} \sum_{m=0}^{X-1} \sum_{n=0}^{Y-1} \left(I_0\left(m,n\right) - I_W\left(m,n\right)\right)^2 \tag{4.5}$$

where $I_0\left(m,n\right)$ is the original image/frame, $I_W\left(m,n\right)$ is the watermarked frame and $X \times Y$ is the size of the image/frame.

PSNR is inverse of MSE and computed as follows:

$$PSNR = 20 \log_{10} \frac{255}{\sqrt{MSE}} \tag{4.6}$$

NC is defined as

$$NC = \frac{\sum_{p=0}^{X-1} \sum_{q=0}^{Y-1} I\left(p,q\right) W\left(p,q\right)}{\sqrt{\sum_{p=0}^{X-1} \left(I\left(p,q\right)\right)^2 \sum_{q=0}^{Y-1} \left(W\left(p,q\right)\right)^2}} \tag{4.7}$$

The quality of the extracted watermark is extremely sensitive to the size of the block and also depends on whether the watermark is embedded on overlapping block or non overlapping block. However, it has been observed that with the increments in block length, the visual distortion becomes more prominent in the watermark image. The overlapping approach of block length has been considered and different block length k has been selected and NC values are noted in Table 4.1. Here NC is calculated between original and extracted watermark without any attack. In the following Table 4.1, the resultant watermark and watermarked image are shown. PSNR and MSE are computed between original and watermarked frame/image, while NC is calculated between original and watermarked image. PSNR measures the invisibility performance and NC checks the robustness feature of the algorithm.

Similarly, Table 4.2 and Table 4.3 show the extracted watermark and watermarked image. PSNR and MSE are again calculated with same manner and NC is measured to verify the robustness between original and watermarked image.

For the non overlapping technique, the watermark is not much resilient against rotation and median filtering attacks. With the technique of overlapping blocks, the watermark is robust against all the possible attacks. From the obtained values, it has been verified that the correlation parameter is above 75%.

4.3.3 Hardware Performance Measurement

The architectures of lifting based wavelet scheme and watermark embedding based sorting scheme are implemented on SPARTAN 3E FPGA using Xilinx ISE 14.1 tool and the results are shown in Table 4.4.

4.3.4 Comparison of Related Work

The algorithm is tested against all common signal processing attacks and results are reported in Table 4.5 and Table 4.6.

Table 4.1: Results for overlapping block for different block size.

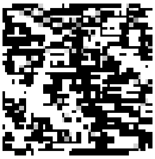
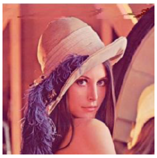
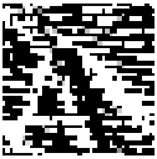
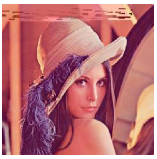
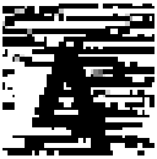
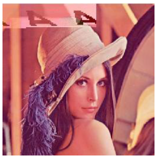
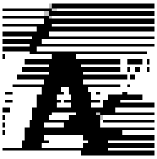
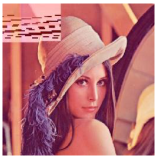
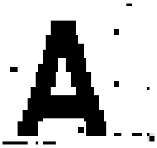
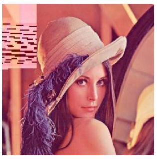
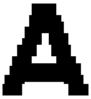
Selected block size (k)	**Performance Measurement**	**Watermarked Image**	**Retrieved Watermark**
4	MSE =8.5460		
	PSNR = 38.8132		
	NC = 0.464151		
6	MSE = 19.98215		
	PSNR = 35.1243		
	NC = 0.500377		
20	MSE = 148.4865		
	PSNR = 26.41393		
	NC = 0.527547		
50	MSE = 312.5945		
	PSNR = 23.18099		
	NC = 0.581133		
80	MSE = 452.0989		
	PSNR = 21.57847		
	NC = 0.98033		
100	MSE = 578.3155		
	PSNR = 20.5092		
	NC = 1.0000		

Table 4.2: Results for non overlapping block technique (block size $k = 100$).

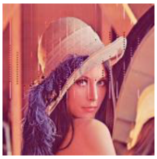
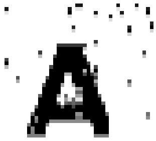
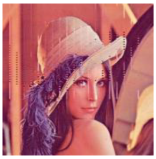
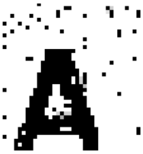
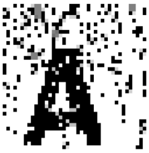
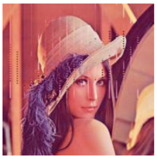
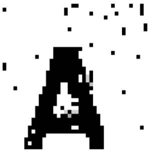
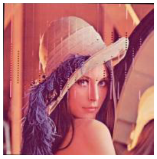
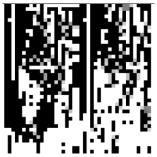
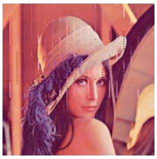
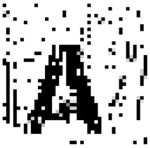
Attack	Performance Measurement	Watermarked Image	Retrieved Watermark
Normal Retrieval	MSE =0		
	PSNR = Infinite		
	NC = 0.9773		
Salt and Pepper Noise	MSE = 9.8754		
	PSNR = 38.1854		
	NC = 0.9627		
Gaussian Noise	MSE = 0.2186		
	PSNR = 54.7350		
	NC = 0.8423		
Compression	MSE = 0		
	PSNR = Infinite		
	NC = 0.9759		
Rotation	MSE = 490.2413		
	PSNR = 21.2267		
	NC = 0.5057		
Median Filtering	MSE = 0.7120		
	PSNR = 49.6066		
	NC = 0.8890		

Table 4.3: Results for overlapping block technique (block size k = 100).

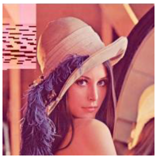
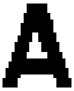
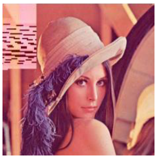
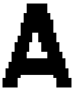
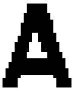
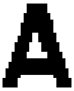
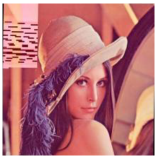
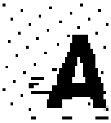
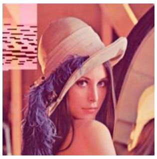
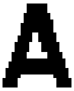
Attacks	**Performance Measurement**	**Watermarked Image**	**Retrieved Watermark**
Normal Retrieval	MSE =578.3155		
	PSNR = 20.5092		
	NC = 1.0000		
Salt and Pepper Noise (0.002)	MSE = 63.5585		
	PSNR = 30.0991		
	NC = 1.0000		
Gaussian Noise (0 mean and 0.001 variances)	MSE = 68.3791		
	PSNR = 29.7816		
	NC = 1.0000		
Compression (Quality Factor 75%)	MSE = 48.5970		
	PSNR = 31.2647		
	NC = 1.0000		
Rotation (1 degree clock wise)	MSE = 524.9266		
	PSNR = 20.9298		
	NC = 0.7917		
Median Filtering (3×3)	MSE = 40.2004		
	PSNR = 32.0885		
	NC = 1.0000		

Table 4.4: Synthesis results of hardware implementation.

Resources	Lifting based Wavelet scheme	Watermark embedding based on sorting scheme
Slices	530/4656 (11%)	256/4656 (5%)
Slice FFs	398/4656 (8%)	198/4656 (4%)
LUTs	978/9312 (10%)	474/9312 (5%)
I/O	60/232 (25%)	38/232 (16%)

Table 4.5: PSNR between original and watermarked images.

Attacks	Sujatha & Sathik [41]	Sathik & Sujatha [39]	Proposed method (block size =4)
Without any attacks	54.1047	59.1668	77.3390
Gaussian Noise (0 mean & 0.001 variance)	30.0770	30.0997	49.6370
Salt and Pepper noise (0.02 density)	31.8352	32.1381	36.0450
Median Filter(3×3)	29.5819	29.5727	49.6010
10% compression	43.0162	43.1448	74.1470

Table 4.6: NC values between original and extracted watermark.

Attacks	Chang's et al. [10]	Lin's et al. [29]	Proposed method
Without any attacks	0.97	1.00	1.00
Gaussian Noise (0 mean & 0.001 variance)	0.73	0.75	0.76
Salt and Pepper noise (0.02 density)	0.63	0.67	0.89
10% compression	0.70	0.85	0.91

To evaluate the performance of the algorithm, *"cameraman"* is used as original image and binary logo is used as a watermark. Table 4.5 shows that the proposed algorithm has admirable PSNR values in comparison to similar work. Sujatha and Sathik [41] described a new robust watermarking scheme which used to embed the watermark in the first level of decomposition of wavelet transform. The watermark was inserted in high frequency bands in order to achieve robustness against normal image processing attacks. Sathik and Sujatha [39] presented a novel blind and robust watermarking method. The watermark was embedded in low frequency components with the help of low pass scaling and band pass wavelet function. The proposed algorithm is used to insert the watermark in the second level of a low frequency sub band of wavelets. The increasing level of decomposition helps to boost the robustness of the algorithm against several attacks such as low pass filter and compression. The proposed algorithm is based on replacing the coefficient values by either maximum or minimum of k size block. In wavelet, the values of the particular sub band are correlated. Therefore, small change in the value does not introduce the visible distortion in the watermarked image. Proposed algorithm has excellent PSNR values to justify the invisibility criteria. As shown in Table 4.6, Chang's et al. method [10] decomposed the host image in different sub bands. Two different watermarks are embedded into the first level decomposition of the low frequency band and middle frequency band for authentication and ownership respectively. Lin's et al. method [29] showed the integer wavelet based watermarking scheme which is used to embed the watermark in selected fixed blocks. The watermark is inserted in the middle frequency HL and LH band at the first level of DWT of the host image. The watermark bits are embedded in even column blocks of HL sub band and odd column blocks of the LH sub band. From the obtained results in Table 4.6, it is evident that the proposed scheme outperforms all other schemes in terms of NC. For simulation, *"baboon"* image is considered as original image and binary logo is used as a watermark. The extracted watermark is more precise than other schemes because it posses greater robustness against all attacks. The architecture of the proposed algorithm is developed with the parallel and pipelining approach. The real-time implementation of the proposed algorithm is achieved and has excellent hardware performance. The algorithm is also extended to video where each frame is considered as an individual image and the proposed algorithm is applied in the same way to video watermarking.

4.3.5 Video Watermarking Algorithm

Video is the prominent object used in our everyday life. It is considered as a sequence of $35-40$ frames per second and each frame is treated as a still image. The proposed image watermarking for variable block size can be extended for video watermarking. However, the frame rate cannot be changed significantly. It has been observed that the video watermarking scheme has a frame rate around $30-32$ frames per second.

4.4 Proposed DCT Based Watermarking Using Self-Reference Approach

In the paper [26], the DCT based watermarking method was proposed to embed the watermark by calculating the difference between original and predicted AC coefficient. The limitation of the approach is that the difference is obtained through trial and error mechanism in order to avoid visible distortion in the blocks.

In the paper [18], the embedding technique used is:

$$if\, w\left(k\right)=0,\, F_k\left(u,v\right)=\begin{cases}\Delta\times Q_s\left(\frac{F_k(u,v)}{\Delta}\right), & if\, u=v=0\\ F_k\left(u,v\right), & otherwise\end{cases} \tag{4.8}$$

$$if\, w\left(k\right)=1,\, F_k\left(u,v\right)=\begin{cases}\Delta\times Q_o\left(\frac{F_k(u,v)}{\Delta}\right), & if\, u=v=0\\ F_k\left(u,v\right), & otherwise\end{cases} \tag{4.9}$$

where $Q_o$indicates to turn the value of x to the most approximate odd number, Q_s indicates to turn the value of x to the most approximate even number and Δ is a parameter of quantization.

This technique is simple, and can be used for all the blocks of the image. However, this multiplication and division by Δ increases the processing unnecessarily. It was thought that checking just one digit of the DC coefficient (tens digit, ones digit or the first digit after the decimal point or any desired place value) and making it even or odd according to the watermark bit would reduce the computation. From experiments, it was observed that modifying the digit just to the right of the decimal place could not retrieve the watermark. Modifying the tens place of the DC coefficient gave visible distortion, but good retrieval on attacks also. The units digit was thus a trade-off between visibility and robustness.

Hung et al. [18], use middle frequency for embedding watermark bit, the coefficient is set to a positive value for watermark bit of 1, else it is set to a negative value for watermark bit of 0. This motivated the compilation of another simpler algorithm which would incorporate the best features of each of the above attempts. Embedding was now done in the DC coefficient and in AC(0,4) and AC(4,0). The capacity of this algorithm is excellent since embedding can now be done in all 8×8 blocks.

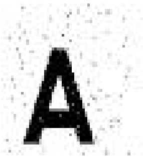
The DCT based proposed algorithm can be extended for video. The watermark is embedded in some randomly selected frames by pseudo random generator. The same random number is used at the extraction side to retrieve the watermark. The results of the proposed algorithm for the case of the images of Fig.(4.7) are depicted in Table 4.7. The proposed DCT based algorithm can be implemented on hardware and is discussed in the subsequent section.

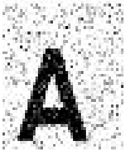
4.4.1 Hardware Implementation

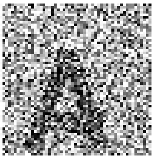
The proposed DCT based algorithm has two main hardware blocks: the one is an Integer DCT module and the other is a watermark embedding unit. The architecture for both blocks are shown in Fig.(4.8) and Fig.(4.9), respectively. The 2-D DCT is implemented with separable property, where the column wise 1-D DCT is applied followed

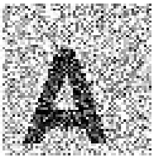
Table 4.7: Results for the proposed DCT based watermarking algorithm.

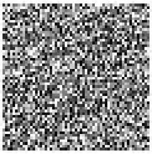
Attacks	Performance Measurement	Watermarked Image	Retrieved Watermark

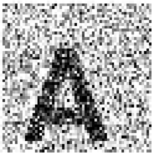
Normal Retrieval	MSE = 1.5818
	PSNR = 46.13911
	NC = 0.991280
			

Salt and Pepper Noise (0.002)	MSE = 35.1519
	PSNR = 32.6713
	NC = 0.9366
			

Gaussian Noise (0 mean and 0.01 variance)	MSE = 314.9934
	PSNR = 23.1477
	NC = 0.665432
			

JPEG Compression (Quality Factor 75%)	MSE = 1.956421
	PSNR = 45.2161
	NC = 0.771859
			

Rotation (1 degree clockwise rotation)	MSE = 452.6975
	PSNR = 21.5727
	NC = 0.507645
			

Median Filtering (3×3)	MSE = 8.9191
	PSNR = 38.6275
	NC = 0.758720
			

(a) (b)

Figure 4.7: Original test data set: (a) image and (b) watermark.

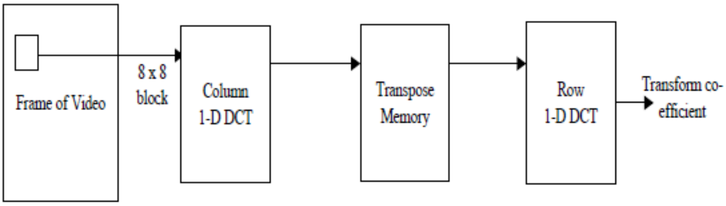


Figure 4.8: Integer 2-D DCT implementation.

by row wise 1-D DCT [24]. The watermark embedding scheme uses multiplication and division operations to generate the watermarked coefficient.

4.4.2 Hardware Performance

The execution of the algorithm is carried out on hardware along with Integer 2-D DCT and watermark embedding scheme which have been implemented using Xilinx ISE 14.1 on SPARTAN 3E FPGA and results are reported in Table 4.8.

Table 4.8: Synthesis results of hardware implementation.

Resources	2-D DCT Module	Watermark embedding scheme
Slices	512/4656 (11%)	326/4656 (7%)
Slice FFs	45/4656 (1%)	416/4656 (9%)
LUTs	834/9312 (9%)	645/9312 (7%)
I/O	58/232 (25%)	32/232 (13%)

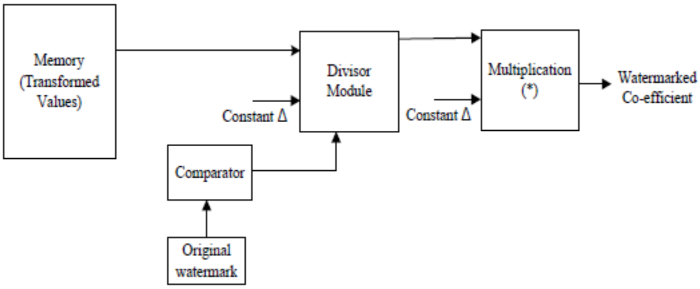


Figure 4.9: Watermark embedding scheme.

4.5 Conclusion

The chapter presents two different watermarking algorithms which are useful to verify the authenticity in the case of image/video ownership verification applications. The main focus is on the development of real time watermark embedding system. The performance of both algorithms is validated through MATLAB and then implemented on the hardware platform. The first method is based on DWT and uses two different approaches, overlapping block technique and non overlapping block technique. Non overlapping block technique provided the excellent results for invisibility criteria, but is less robust against some substantial attacks. While in case of the overlapping technique, the exceptional robustness is achieved against some serious distortion based attacks. Both approaches have been implemented on the FPGA to demonstrate the real time performance. Second algorithm is based on DCT watermarking that has a significant robustness against the attacks and also satisfies the invisibility criteria. However, the algorithm requires high computational complexity due to the involvement floating point structure. The area utilizations of both the proposed algorithms have been calculated in order to verify the hardware performance.

References

[1] I. Amer, W. Badawy, and G. Jullien. A high-performance hardware implementation of the H.264 simplified 8x8 transformation and quantization [video coding]. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 1137–1140, 2005.

[2] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra. A SIFT-based forensic method for copy¨cmove attack detection and transformation recovery. *IEEE Transactions on Information Forensics and Security*, 6(3):1099–1110, 2011.

[3] M. Barni and F. Bartolini, editors. *Watermarking Systems Engineering: Enabling Digital Assets Security and other Applications*. CRC Press, 2004.

[4] S. Biswas, S.R. Das, and E.M. Petriu. An adaptive compressed MPEG-2 video watermarking scheme. *IEEE Transactions on Instrumentation and Measurement*, 54(5):1853–1861, 2005.

[5] J. Blake and S. Latifi. Digital watermarking security. *Defense Science Journal*, 61:408–414, 2011.

[6] D. Bouslimi, G. Coatrieux, M. Cozic, and C. Roux. A joint encryption/watermarking system for verifying the reliability of medical images. *IEEE Transactions on Information Technology in Biomedicine*, 16(5):891–899, 2012.

[7] A. Brunton and J. Zhao. Real-time video watermarking on programmable graphics hardware. In *Canadian Conference on Electrical and Computer Engineering*, pages 1312–1315, 2005.

[8] R. Chandramouli, M. Kharrazi, and N. Memon. Image steganography and steganalysis: concepts and practice. In *2nd International Workshop on Digital Watermarking (IWDW)*, pages 35–49, 2004.

[9] C.C. Chang, C.C. Lin, and Y.S. Hu. An SVD oriented watermark embedding scheme with high qualities for the restored images. *International Journal of Innovative Computing, Information and Control*, 3(3):609–620, 2007.

[10] C.C. Chang, W.L. Tai, and C.C. Lin. A multipurpose wavelet-based image watermarking. In *International Conference on Innovative Computing, Information and Control (ICICIC)*, pages 70–73, 2006.

[11] R. Chaturvedi, A. Sharma, N. Hemrajani, and D. Goyal. Analysis of robust watermarking technique using mid band DCT domain for different image formats. *International Journal of Scientific and Research Publications*, 2(3):1–4, 2012.

[12] M. Chen, R. Zhang, X. Niu, and Y. Yang. Analysis of current steganography tools: classifications & features. In *International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pages 384–387, 2006.

[13] N. Chowdhury and P. Manna. An efficient method of steganography using matrix approach. *International Journal of Intelligent Systems and Applications*, 4(1):32–38, 2012.

[14] I. Cox, M.L. Miller, and J.A. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2002.

[15] I.J. Cox, J. Kilian, F.T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing*, 6(12):1673–1687, 1997.

[16] C. Cruz-Ramos, R. Reyes-Reyes, M. Nakano-Miyatake, and H. Perez-Meana. A blind video watermarking scheme robust to frame attacks combined with MPEG2 compression. *Journal of Applied Research and Technology*, 8(3):323–339, 2010.

[17] Y.G. Fu and H.R. Wang. A novel discrete wavelet transform based digital watermarking scheme. In *2nd International Conference on Anti-counterfeiting, Security and Identification (ASID)*, pages 55–58, 2008.

[18] K.M. Hung. A novel robust watermarking technique using IntDCT based AC prediction. *WSEAS Transactions on Computers*, 7(1):16–24, 2008.

[19] T. Jayamalar and V. Radha. Survey on digital video watermarking techniques and attacks on watermarks. *International Journal of Engineering Science and Technology*, 2(12):6963–6967, 2010.

[20] A. Joshi, V. Mishra, and R.M. Patrikar. *Watermarking*, chapter Real Time Implementation of Digital Watermarking Algorithm for Image and Video Application, pages 65–90. INTECH, 2012.

[21] A.M. Joshi and A. Darji. Efficient dual domain watermarking scheme for se-

cure images. In *International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom)*, pages 909–914, 2009.

[22] A.M. Joshi, A. Darji, and V. Mishra. Design and implementation of real-time image watermarking. In *IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, pages 1–5, 2011.

[23] A.M Joshi, V. Mishra, and R.M. Patrikar. Design of real-time video watermarking based on integer DCT for H.264 encoder. *International Journal of Electronics*, 102(1):141–155, 2015.

[24] A.M Joshi, V. Mishra, and R.M. Patrikar. FPGA prototyping of video watermarking for ownership verification based on H.264/AVC. *Multimedia Tools and Applications*, 2015.

[25] A.M. Joshi, R.M. Patrikar, and V. Mishra. Design of low complexity video watermarking algorithm based on integer DCT. In *International Conference on Signal Processing and Communications (SPCOM)*, pages 1–5, 2012.

[26] N. Kaur, M. Kansal, and G. Singh. DCT and thresholding based digital video watermarking. *International Journal of Applied Information Systems*, 2(6):9–12, 2012.

[27] S. Kimpan, A. Lasakul, and S. Chitwong. Variable block size based adaptive watermarking in spatial domain. In *IEEE International Symposium on Communications and Information Technology (ISCIT)*, volume 1, pages 374–377, 2004.

[28] E. Kougianos, S.P. Mohanty, and R.N. Mahapatra. Hardware assisted watermarking for multimedia. *Computers & Electrical Engineering*, 35(2):339–358, 2009.

[29] H.J. Lin, C.W. Lu, and C.M. Chiang. DWT-based watermarking technique associated with embedding rule. In *10th WSEAS International Conference on Signal Processing, Computational Geometry and Artificial Vision (ISCGAV)*, pages 23–28, 2010.

[30] G. Liu and J. Zhao. Key frame extraction from MPEG video stream. In *Third International Symposium on Information Processing (ISIP)*, pages 423–427, 2010.

[31] N.J. Mathai, D. Kundur, and A. Sheikholeslami. Hardware implementation perspectives of digital video watermarking algorithms. *IEEE Transactions on Signal Processing*, 51(4):925–938, 2003.

[32] S.P. Mohanty and E. Kougianos. Real-time perceptual watermarking architectures for video broadcasting. *Journal of Systems and Software*, 84(5):724–738, 2011.

[33] V.M. Potdar and E. Chang. Grey level modification steganography for secret communication. In *2nd IEEE International Conference on Industrial Informatics (INDIN)*, pages 223–228, 2004.

[34] V.M. Potdar, S. Han, and E. Chang. A survey of digital image watermarking techniques. In *3rd IEEE International Conference on Industrial Informatics (INDIN)*, pages 709–716, 2005.

[35] H. Qi, D. Zheng, and J. Zhao. Human visual system based adaptive digital image watermarking. *Signal Processing*, 88(1):174–188, 2008.

[36] K. Raghavendra and K.R. Chetan. A blind and robust watermarking scheme with scrambled watermark for video authentication. In *IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA)*, pages 1–6, 2009.

[37] I.E. Richardson. *H. 264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia*. John Wiley & Sons, 2004.

[38] S.D. Roy, X. Li, Y. Shoshan, A. Fish, and O. Yadid-Pecht. Hardware implementation of a digital watermarking system for video authentication. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2):289–301, 2013.

[39] M.M. Sathik and S.S. Sujatha. A novel DWT based invisible watermarking technique for digital images. *International Arab Journal of e-Technology*, 2(3):167–173, 2012.

[40] Y. Shoshan, A. Fish, X. Li, G. Jullien, and O. Yadid-Pecht. VLSI watermark implementations and applications. *International Journal Information Technologies and Knowledge*, 2:379–386, 2008.

[41] S.S. Sujatha and M.M. Sathik. A novel pixel based blind watermarking algorithm by applying Fibonacci transform. In *1st Amrita ACM-W Celebration on Women in Computing in India (A2CWiC)*, page Article No. 39, 2010.

[42] S.M. Thampi. Information hiding techniques: a tutorial review. In *ISTE-STTP on Network Security & Cryptography (LBSCE)*, 2004.

[43] Y. Wang and A. Pearmain. Blind image data hiding based on self reference. *Pattern Recognition Letters*, 25(15):1681–1689, 2004.

[44] P.W. Wong. A public key watermark for image verification and authentication. In *International Conference on Image Processing (ICIP)*, volume 1, pages 455–459, 1998.