Hardware Assisted Watermarking for Multimedia

Elias Kougianos, Saraju P. Mohanty
VLSI Design and CAD Laboratory (VDCL)
University of North Texas, Denton, TX 76203.

Rabi N. Mahapatra
Department of Computer Science
Texas A&M University, College Station, TX 77843.

Abstract

Digital media offer several distinct advantages over analog media, such as high quality, ease of editing, and ease of processing operations such as compression and high fidelity copying. Digital data is commonly available through digital TV broadcast, CD, DVD, and computing devices such as personal computers. The ease by which a digital media object can be duplicated and distributed has led to the need for effective digital rights management tools. Digital watermarking is one such tool. Watermarking is the process of embedding extra data called a watermark into a multimedia object, like image, audio, or video, such that the watermark can later be detected or extracted in order to make an assertion regarding the object. During the last decade, numerous software based watermarking schemes have appeared in the literature and watermarking research has attained a certain degree of maturity. But hardware based watermarking systems have evolved more recently only and they are still at their infancy. The goal of hardware assisted watermarking is to achieve low power usage, real-time performance, reliability, and ease of integration with existing consumer electronic devices. In this paper, we survey the hardware assisted solutions proposed in the literature for watermarking of multimedia objects. The survey is preceded by an introduction to the background issues involved in digital watermarking.

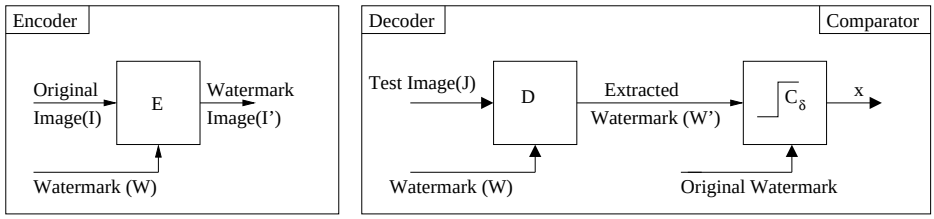
Key words: Image Watermarking, Video Watermarking, Digital Rights Management (DRM), VLSI Architectures, Watermarking Chip

1. Introduction

Electronic watermarking was invented in 1954 by Emil Hembrooke of the Muzac Corporation (1). A vast research community involving experts from computer science, cryptography, signal processing, and communications has come together in the last decade to develop watermarks suitable for various applications. Digital watermarking is intended by its developers as the solution to the requirement of providing value-added protection on top of data encryption and scrambling for content protection. Like any other technology under development, digital watermarking raises a number of essential questions (1; 2; 3). In this paper we will attempt to address some of these questions followed by a detailed discussion on hardware based solutions for image and video watermarking available in the current literature.

Digital watermarking is a method that inserts some information into a multimedia object and generates a watermarked multimedia object (4; 5). The object may be an image, audio, video or text. Watermarking has many different applications (6; 7; 8; 1; 9; 10), such as ownership evidence, fingerprinting, authentication and integrity verification,

Email addresses: eliask@unt.edu, smohanty@unt.edu (Saraju P. Mohanty), rabi@cs.tamu.edu (Rabi N. Mahapatra).



(a) The Encoder (b) The Decoder

Fig. 1. General Framework for Watermarking Showing the Encoder and Decoder Structure (12; 7; 19; 22)

content labeling and protection, and usage control. The success of any watermarking scheme is determined by its performance against intentional and unintentional attacks (4; 11; 12; 13; 14). The requirements for fulfilling desired characteristics and the requirements for performance against attacks are mutually conflicting.

Recently, hardware implementations of watermarking algorithms are being presented in the literature. Hardware implementation is essential for low power, real-time performance, high reliability, low cost applications, and also for easy integration with existing consumer electronic devices (15; 16; 17). For example, watermarking chips can be integrated with any existing digital still image camera. The hardware modules can also be integrated with a JPEG-codec (18).

In turn, the JPEG codec can be part of a scanner, a digital camera, or any other multimedia device so that the digitized images are watermarked right at the capture time.

The rest of the paper is organized as follows: Section 2 discusses the general framework of watermarking, summarizes the various types of watermarks, and applications of watermarking. Section 3 discusses different attacks on watermarks and watermarking systems along with benchmarks that can be used to test their effectiveness. We elaborate on hardware based watermarking circuits and systems available in the current literature for images and video in Sections 4 and 5, respectively. Section 6 elaborates on the challenges faced by the watermarking research community and the difficulties involved in making use of them in practice. The paper is concluded in Section 7.

2. General Framework, Types, and Applications of Watermarking

In general, watermarking is the process that embeds data called a watermark into a multimedia object such that it can be detected or extracted later to make an assertion about the object. The object may be an image or audio or video. A simple example of a digital watermark would be a visible "seal" placed over an image to identify the copyright. However the watermark might contain additional information including the identity of the purchaser of a particular copy of the material. In this section, we introduce the general framework of watermarking, discuss different types of watermarking and their applications.

2.1. *General Framework*

As demonstrated in Fig. 1, any watermarking scheme consists of several distinct parts (12; 7; 19; 20; 21): the watermark, the encoder, the decoder and comparator. The watermark is usually an image in the case of a visible watermarking scheme. In the case of invisible watermarking, it can be a binary image, random or pseudorandom number. Each owner has a unique watermark or an owner can also place different watermarks in different objects. The insertion algorithm incorporates the watermark into the object. The verification algorithm authenticates the object, determining both the owner and the integrity of the object.

2.1.1. *The Encoder*

Let us denote an image by I, a watermark by $W = \{w_1, w_2, \ldots\}$ and the watermarked image by $\hat{I}$. E is an encoder function which takes an image I and a signature W in order to generate a new image which is called a watermarked image $\hat{I}$:

$$E\left(I, W\right) = \hat{I}. \tag{1}$$

It may be noted that the watermark W may be dependent on several factors such as the original image size, original image features, a user key and identity. In all such cases, the encoding process described by Eqn. 1 still holds.

2.1.2. *The Decoder*

It is necessary to decode a watermark in order for it to be useful for resolving ownership or authentication. The decoding process may use very distinct approaches depending upon the watermark insertion process and the nature of the watermarking algorithm. In some watermarking schemes, a watermark can be extracted in its approximately original form, a procedure known as watermark extraction. In other cases, we can only detect whether a specific given watermarking signal is present in an object, a procedure called watermark detection. While watermark extraction can prove ownership, watermark detection can only verify ownership. A decoder function D takes an image J (which can be a watermarked or non-watermarked image, and possibly corrupted) whose ownership is to be determined and recovers a watermark W' from the image:

$$D\left(J, I\right) = W'. \tag{2}$$

In this process an additional image I may also be included which is often the original and un-watermarked version of J.

The extracted watermark W' is then compared with the owner sequence by a comparator function C_δ and a binary output decision is generated. It is 1 if there is match and 0 otherwise, which can be represented as follows:

$$C_\delta\left(W', W\right) = \begin{cases} 1, \ \mathrm{c} \leq \delta, \\ 0, \ \text{otherwise}. \end{cases} \tag{3}$$

Where $c = C_\delta\left(W', W\right)$ is the correlation between the two signatures and δ is some threshold. Thus, the watermarking scheme can be treated as a triplet (E, D, C_δ).

2.2. *Types of Digital Watermarks*

Digital watermarking research has matured over the last decade and a large number of watermarking techniques are available in the current literature. All these watermarking techniques can be divided into different categories in various ways.

2.2.1. *Spatial Vs Frequency Domain Watermarking*

Watermarks can be embedded in either the spatial or the frequency domain. The various transformations that have been used as alternatives to the spatial domain are Discrete Cosine Transform (DCT), Fourier Transform (FT), Wavelet Transform (WT), and Fractal etc. The spatial based methods are less computation intensive and are easy to implement using both software and hardware. On the other hand, frequency based method have several advantages over spatial domain based methods (23; 24; 25). Lossy compression is an operation that usually eliminates perceptually non-salient components of an image or sound. If one wishes to preserve watermarking following such an operation, the watermark must be placed in the perceptually significant region of the data. Most processing of this sort takes place in the frequency domain. Hence, the watermark must be placed in the significant frequency region (low frequency component) of the image or sound spectrum. Cropping may be a serious threat to any spatially based watermark but it is less likely to affect frequency based schemes.

2.2.2. *Based on Multimedia Objects*

Watermarking techniques can be divided into four categories according to the type of multimedia object to be watermarked as follows (26; 14; 23): image watermarking, video watermarking, audio watermarking, and text watermarking. It is difficult to devise a watermarking algorithm that can work effectively for all these objects. The secure spread spectrum algorithm proposed in (23) claimed to work effectively for image, video and audio. In (7), it is pointed out that watermarking of halftoned images has to be dealt with differently. Text or document watermarking is treated quite differently from image, audio or video watermarking (27; 28).

2.2.3. *Based on Human Perception*

Based on human perception, digital watermarks can be divided into visible and invisible types (26; 14; 7; 12). A visible watermark is a secondary translucent image overlaid into a primary image. The watermark appears visible to a viewer only on careful inspection. On the other hand, the invisible watermark is completely imperceptible. The invisible-robust watermark is embedded in such a way that alterations made to the pixel value are perceptually not noticeable and can be recovered only with the appropriate decoding mechanism. The invisible-fragile watermark is embedded in such a way that any manipulation or modification of the image would alter or destroy the watermark. There are multiple watermarking algorithms presented in the literature such as (29; 30; 31; 32) that serve different purposes. For example, the dual watermark in (31) is a combination of a visible and an invisible-fragile watermark. An invisible robust private watermarking scheme requires the original or reference image for watermark detection whereas public watermarks do not (12; 22). The class of invisible robust watermarking schemes that can be attacked by creating a "counterfeit original" is called invertible. Otherwise, the watermarking scheme is non-invertible. A watermarking scheme (E, D, C_δ) is called quasi-invertible if, for any watermarked image $\hat{I}$, there exists a function E^{-1} such that (1) $E^{-1}(\hat{I}) = (I', W')$, and (2) $C_\delta(D(\hat{I}), W') = 1$, where E^{-1} is a computationally feasible function, W' belongs to the set of allowable watermarks, and the images I and I' are perceptually similar. Otherwise, the watermarking scheme is nonquasi-invertible. The non-invertible or nonquasi-invertible approaches are invulnerable to sophisticated attacks and are more useful in resolving ownership (12).

2.2.4. *From Applications Point of View*

From an application point of view, digital watermark could be source or destination based (33). The source-based watermark is desirable for ownership identification or authentication where a unique watermark identifying the owner is introduced into all the copies of a particular image being distributed. The watermark could also be destination-based where each distributed copy gets a unique watermark identifying the particular buyer. The destination-based watermark could be used to trace the buyer in the case of illegal reselling.

2.2.5. *Based on Embedding Techniques*

Based on the embedding techniques used, the watermarking schemes can be categorized as additive, multiplicative, or quantization (4). In additive schemes, there are usually very weak dependencies between host data and watermark (25), whereas in multiplicative schemes samples of the original data are multiplied by an independent factor (25). In quantization based watermarking schemes, strong local dependencies exist between the created watermark and the host data, but the dependencies appear to be statistically independent (34).

2.2.6. *Hardware based Watermarking Systems*

A hardware based watermarking system can be designed on a field programmable gate array (FPGA) processor board, digital signal processor (DSP) board, or custom IC (35; 36). Moreover, the graphics processing unit (GPU) has been recently explored for hardware assisted real-time watermarking (37; 38). In (35), the authors have pointed out that the FPGA platform is more relevant for DVD. There are very few algorithm implementations available using FPGA boards. Some cell based ASIC designs are available for both visible and invisible watermarking of images and videos in real time. The choice between FPGA or cell based IC is a trade-off between cost, power consumption, and performance (17; 39; 40; 16; 41). From an integration point of view custom based ASIC designs may be more useful.

2.2.7. *Desired Characteristics of Hardware Based Watermarking Systems*

The desired features for hardware watermarking schemes are given below (16; 40; 39; 41): The silicon area needed for the chip implementation should be as small as possible, memory requirements should be low, power consumption of the chip should be minimal, and the chip should be highly integrable with other devices with low operational latency and should be suitable for real-time operation. The chip should also use blind watermarking schemes to avoid resorting to the original image for detection or extraction, thus reducing the system memory requirements.

3. Attacks on Watermarks and Watermarking Systems and Benchmarks

A watermarked object is likely to be subjected to certain manipulative processes before it reaches the receiver. Common signal processing functions such as analog-to-digital conversion, digital-to-analog conversion, sampling, quantization, requantization, dithering, recompression, linear and nonlinear filtering, low-pass and high-pass filtering, addition of Gaussian and non Gaussian noise are common manipulations. An attack is any processing that impairs or misleads the watermark detector (4). The performance of a watermarking algorithm against these attacks reflects its quality (42; 13; 14; 5; 43; 44; 45; 46). Similarly, it is anticipated that an embedded system realization of watermarking can face several physical attacks similar to the ones suggested for cryptography in the literature (47).

The requirements for fulfilling desired characteristics and the requirements for performance against attacks are mutually conflicting. There are many watermarking algorithms proposed in the current literature and a metric has been developed for their comparison so that the user can make a decision to use one of the algorithms that best suits his need. The benchmarks essentially combine many attacks in a unified framework and allow the user to test the watermarking system for its applicability. There are several benchmarks available, such as Stirmark, (11; 48), Unzign (49), Checkmark (50; 51), and Certimark (4). In this section, we provide a broad discussion on the various attacks and available benchmarks.

3.1. *Attacks*

We discuss various possible intentional and unintentional attacks that a watermarked object is likely to be subjected to. We classify the attacks into four different types, such as removal and interference attacks, geometric attacks, cryptographic attacks, and protocol attacks (52; 49; 51; 50; 4). Besides these four types, there is another class of attacks called estimation based attacks (4). In estimation based attacks, estimates of either the watermark data or the original object can be obtained using stochastic methods. The estimation based attacks can be classified as removal, protocol, or desynchronization depending on the way the estimate is used.

3.1.1. *Removal and Interference Attacks*

Removal attacks intend to remove the watermark data from the watermarked object. Such attacks exploit the fact that the watermark is usually an additive noise signal present in the host signal. Moreover, the interference attacks are those which add additional noise to the watermarked object. Lossy compression, quantization, collusion, denoising, remodulation, averaging, and noise storm are some examples of this category of attacks. The collusion attack occurs when a number of authorized recipients of the multimedia object come together (collude) to generate an unwatermarked object by averaging all the different watermarked objects.

3.1.2. *Geometric Attacks*

Geometric attacks are specific to images and videos. The geometric attacks do not actually remove the watermark, but manipulate the watermarked object in such a way that the detector cannot find the watermark data. This type of attack includes affine transformations such as rotation, translation, and scaling. Warping, line/column removal and cropping are also included in this family of attacks. Another example of geometric attack is the mosaic attack (52; 11). In the mosaic attack, the watermarked image is divided into several parts and rearranged using proper HTML code, thus constructing a watermarked image in which the watermark detector will fail to provide desired results. Local pixel jittering is an efficient spatial domain geometric attack.

3.1.3. *Cryptographic Attacks*

The above two type of attacks, removal and geometric, do not breach the security of the watermarking algorithm. On the other hand, cryptographic attacks deal with the cracking of the security. For example, finding the secret watermarking key using exhaustive brute force method is a cryptographic attack. Another example of this type of attack is the oracle attack (22). In the oracle attack, a non-watermarked object is created when a public watermark detector device is available. These attacks are similar to the attacks used in cryptography.

3.1.4. *Protocol Attacks*

The protocol attacks exploit the loopholes in the watermarking concept. One example of such attack is the IBM attack (20; 12). The IBM attack is also known as the deadlock attack, inversion attack, or fake-original attack. This attack embeds one or several additional watermarks such that it is unclear which was the watermark of the original owner. Watermarking of an already watermarked image is called rewatermarking. In some inversion attacks, a fake original object is created that produces the same results through the detector as that of real original object.

3.1.5. *Physical Attacks*

This kind of attacks can take place when the watermarking system and corresponding digital rights management system is realized in the embedded system framework using custom-IC or system-on-a-chip (SoC) technology. The scenario is analogous to the encryption system as presented in (47; 53; 54), as watermark generation often involves a secret key and linear feedback shift register (LFSR). These attacks take advantage of the switching activities that take place during the execution of the static CMOS circuit in which different capacitances are switched which can be determined through power and timing analysis from which the watermarking or encryption keys can be determined, thus breaking the security.

3.2. *Benchmarks*

In this subsection, we discuss benchmarks developed combining various attacks in unified framework. These benchmarks will serve as standardization and testing tools for watermarking systems so that watermarking becomes useful in practice.

3.2.1. *Stirmark*

Stirmark is one of the earliest watermarking benchmarks (22; 11; 48). The authors point out that the proposed watermarking schemes could survive basic manipulation, such as rotation, shearing, resampling, resizing, and lossy compression, but they fail when a combination of such manipulations are used simultaneously. This was the motivation for the development of the Stirmark suite. The Stirmark includes several attacks, such as compression, geometric transformations, processing for signal enhancement, and noise addition. The geometric transformations included are rotation, cropping, scaling, generalized geometric transformation, random geometric transformation, and geometric transformation with medium compression. A combination of nonuniform scaling, rotation, and shearing is called a general geometric transformation. In random geometric transformations, the watermarked image is slightly stretched, sheared, shifted and/or rotated by an unnoticeable random amount (11).

3.2.2. *Unzign*

The Stirmark benchmark discussed above uses both global and local geometric distortions. Contrastingly, the Unzign benchmark introduces local pixel jittering and is very efficient in attacking spatial domain watermarking techniques (4).

3.2.3. *Checkmark*

The Stirmark benchmark, while strongly using geometric distortions, does not contain estimation based attacks. A second generation benchmark called Checkmark (50; 51) includes estimation based attacks. In other words, it considers the prior information about the watermark. The different type of attacks include denoising, wavelet compression, copy attack, active desynchronization, denoising followed by random geometric distortion (of Stirmark), geometrical attacks, and denoising followed by perceptual remodulation. The benchmark includes new removal attacks, such as, maximum likelihood estimation attacks, maximum a posteriori (MAP) based attack, denoising assuming low pass watermark, and denoising followed by perceptual remodulation. Projective transforms, collage attacks, and non-uniform line removal are the new geometric attacks included in the benchmark.

Table 1
Image Watermarking Hardware Proposed in the Current Literature

Research Work	Design Type	Watermarking Type	Multimedia Object	Working Domain	Chip Statistics
Garimella et. al. (55)	Custom IC	Invisible Fragile	Image	Spatial	0.13μ, $3453 \times 3453mm^2$ $1.2V$, $37.6\mu W$
Garimella et. al. (56)	Custom IC	Invisible Fragile	Image	Spatial	0.13μ, $545\mu m \times 525\mu m$ $1.14V, 166.6MHz, 9.19mW$
Mohanty et. al. (57; 58)	Custom IC	Invisible Robust-Fragile	Image	Spatial	0.35μ, $3.3V$, $545MHz$, $2.0547mW$
Mohanty et. al. (59; 58)	FPGA board	Invisible Robust	Image	Spatial	Xilinx, Virtex XCV50-BG256-6, $50.398MHz$
Mohanty et. al. (17; 40)	Custom IC	Visible	Image	Spatial	0.35μ, $3.34 \times 2.89mm^2$ $3.3V$, $292MHz$, $6.93mW$
Nelson et. al. (60)	Custom IC	Invisible Robust	Image	Spatial	0.18μ, $1.8V/3.3V$ $1122\mu m \times 1302\mu m$
Lukac and Plataniotis (61)	Custom IC	Visible	Image	Spatial	NA
Tsai and Lu (18)	Custom IC	Invisible Robust	Image	DCT	0.35μ, $3.064 \times 3.064mm^2$, $3.3V$, $50MHz$, $62.78mW$
Mohanty et. al. (39; 41)	Custom IC	Invisible-Robust Visible	Image	DCT	0.25μ, $16.2mm^2$, $1.5, 2.5V$ $70, 280MHz$, $0.3mW$
Mohanty et. al. (38)	GPU	Invisible-Robust	Image	DCT	NA
Hsiao et. al. (62; 63)	Custom IC	Invisible-Robust	Image	Wavelet	NA
Seo and Kim (64)	FPGA board	Invisible Robust	Image	Wavelet	$82MHz$, 4037 LABs, 85 ESBs Altera APEX20KC
Fan et. al. (65)	Custom IC	Visible	Image	Wavelet	NA

3.2.4. *Certimark*

The European project *certification for watermarking techniques (Certimark)* (4) aims to issue international certification to watermarking algorithms. Certimark has several objectives, such as to design, develop, and publish a complete benchmark suite, to make the benchmark available to both customers and suppliers, to set up the certification process for watermarking schemes, and to conduct research on the open problems in watermarking.

4. Image Watermarking Hardware Systems

In this section, we discuss hardware based image watermarking circuits and systems. We have arranged them in the order of spatial, DCT, and wavelet domain watermarking. A comparative view of the proposed image watermarking chips is provided in Table 1.

4.1. *Spatial Domain Watermarking Architecture and Chip*

4.1.1. *An Invisible-Fragile Watermarking Chip*

A scheme for invisible-fragile watermarking in the spatial domain, where the differential error is encrypted and interleaved along the first sample and its VLSI architecture has been proposed by Garimella et al. (55). The watermark can be extracted by accumulating the consecutive LSBs of pixels, followed by decryption. The extracted watermark is then compared with the original watermark for image authentication. The ASIC is implemented using 0.13μ CMOS technology. The area of the chip is $3453\times3453\mu m^2$ and consumes $37.6\mu W$ power when operated at $1.2V$. The critical path delay of the circuit is $5.89ns$. The advantage of this proposed chip is that it consumes a very small amount of power. However it works for gray-scale images only. The structural details of the datapath architecture are missing.

The watermark is a text document converted into ASCII code. This ASCII is encrypted using differential pulse code modulation techniques (DPCM) to enhance security. The size of the watermark used is 2048 bits. The encrypted ASCII code replaces the least significant bits (LSB) of a gray scale image. For watermark extraction, the consecutive LSBs of the pixels are accumulated and decrypted, which forms an extracted watermark. The extracted watermark is then compared with the original ASCII code, thus authenticating the image.

4.1.2. *An Invisible-Fragile Watermarking Chip for Color Images*

The corresponding chip for color image watermarking is presented in (56). They designed RGB - YUV and YUV - RGB converters, watermark encryption, embedding, extracting, detection, and authentication modules. The watermark is embedded in the least significant bit (LSB) pixels of an 8-bit gray scale image. They first transformed the color image from RGB to YUV color space, then inserted the encrypted watermark and retransformed from YUV to RGB color space using pipeline logic. In the proposed watermarking algorithm, an RGB image is transformed into YUV where the luminance component Y is the watermarking channel. Two LSBs of Y are used for watermarking where the watermark is an encrypted bit stream of ASCII coded text file. Each encrypted watermark bit is swapped with the first LSB of Y and the second LSB is the complemented bit of the first LSB. This encryption is achieved with a Differential Pulse Code Modulation (DPCM) technique. Using this technique, each watermark byte is subtracted from the previous byte and the difference value is used in watermarking. The watermarked image is detected by accumulating consecutive LSBs of the channel and decrypting and authenticating the image.

4.1.3. *An Invisible-Robust-Fragile Watermarking Chip*

A chip that can insert both invisible robust and invisible fragile watermarking is described by Mohanty et al. (57; 58). In invisible-robust watermarking, a ternary watermark is embedded in the original image using an encoding function that involves addition of a scaled grey value of neighboring pixels. A binary watermark generated from pseudorandom numbers is XORed with an original image bit-plane in the invisible-fragile watermarking scheme. The chip is implemented using 0.35μ CMOS technology and consumes $2.0547mW$ when operated at $3.3V$ and $545MHz$ frequency.

The invisible-robust algorithm proposed in (66) and the invisible-fragile algorithm proposed in (31) are implemented in this chip. The VLSI implementations of watermarking encoding algorithms are given, thus making it a part of the JPEG encoder. In invisible-robust watermarking, the watermark W is a ternary image of three pixel values, 0, 1 or 2, which are generated using a digital key K. The watermarked image I_W is obtained by altering the pixels of the original image I as follows:

$$I_W(i,j) = \begin{cases} I(i,j) & \text{if } W(i,j) = 0 \\ E_1\big(I(i,j), I_N(i,j)\big) & \text{if } W(i,j) = 1 \\ E_2\big(I(i,j), I_N(i,j)\big) & \text{if } W(i,j) = 2. \end{cases} \tag{4}$$

For watermark strength factors α_1 and α_2, these encoding functions are defined as follows:

$$\begin{aligned} E_1(I, I_N) &= (1-\alpha_1)I_N(i,j) + \alpha_1 I(i,j), \\ E_2(I, I_N) &= (1-\alpha_1)I_N(i,j) - \alpha_2 I(i,j), \end{aligned} \tag{5}$$

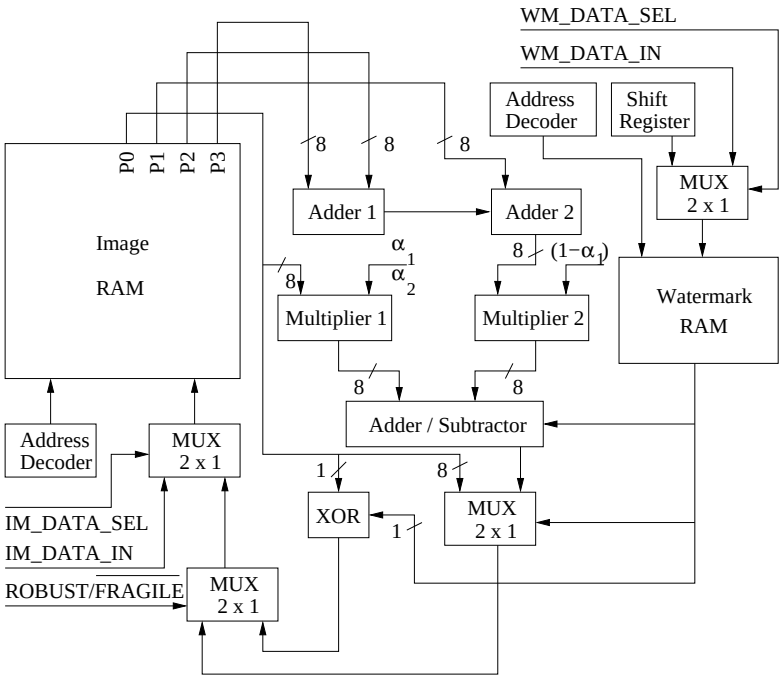


Fig. 2. Architecture for Invisible-Robust/Fragile Watermarking Chip (57)

where α_1 and α_2 satisfy $1 > \alpha_1 > 0$ and $-1 < \alpha_2 < 0$. The neighborhood image pixel gray value I_N is calculated as the average gray value of neighboring pixels of the original image as shown below:

$$I_N(i,j) = \frac{\frac{I(i+1,j)+I(i+1,j+1)}{2} + I(i,j+1)}{2}. \tag{6}$$

In the invisible-fragile insertion process, a pseudo-random binary-sequence {0,1} in inserted in the k^{th} bit plane of the original image using an XOR function:

$$\begin{aligned} I_W[0 \rightarrow k-1](i,j) &= I[0 \rightarrow k-1](i,j) \\ I_W[k](i,j) &= I[k](i,j) \text{XOR } W(i,j) \\ I_W[k+1 \rightarrow 7](i,j) &= I[k+1 \rightarrow 7](i,j). \end{aligned} \tag{7}$$

Finding the candidate bit plane for watermark insertion is an iterative process, but the $2^{nd}(k=2)$ bit plane is chosen as the candidate for watermark insertion (for LSB $k=0$). The watermarked image I_W is obtained by merging all the bit planes.

Fig. 2 shows the proposed datapath architecture that can perform both invisible robust and invisible fragile watermark embedding. There are two RAMs, image RAM and watermark RAM used for temporary storage of image and watermark data during computation. The watermarked image is stored back in the image RAM. The watermark sequence is generated using shift registers, however external watermark data can also be used. There are address decoders to decode the address of RAMs. The main functional units used are multipliers, adders, adder/subtractor, and XOR. The multiplexers are used whenever a selection between multiple signals is needed at a particular unit input. A controller that handles the datapath is designed as a FSM of five different states.

4.1.4. *An Invisible-Robust Watermarking Module*

In (59; 58), the FPGA based implementation of an invisible-robust insertion algorithm is given. The algorithm proposed by Tefas and Pitas (66) is chosen for implementation. The implementation of the proposed architecture is

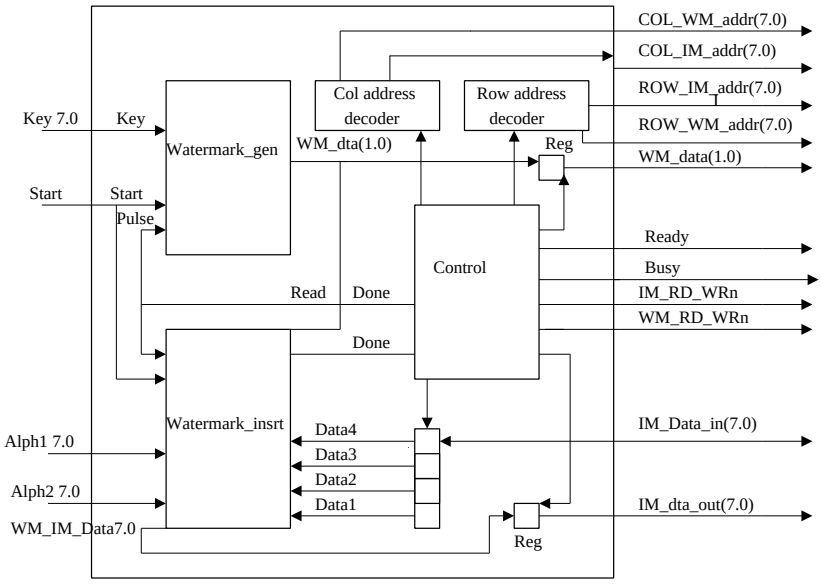


Fig. 3. Invisible-Robust Spatial Domain Watermarking Module (59)

carried out in a Xilinx FGPA board. The synthesis results using VIRTEX technology with XCV50-BG256-6 target device indicate that the chip can be run at $50.398MHz$ frequency.

Fig. 3 shows the functional diagram of the encoder chip. The encoder consists of submodules, such as watermark generator, watermark insertion, address decoders, registers, and a controller. It is assumed that there is external storage (RAM) to load/store the images (watermark image, host image). The watermark generator module consists of a linear feedback shift register (LFSR). Since the LFSR is the most important module for most of the present day watermarking schemes, special attention is give for its design. Besides other issues, a significant design criterion is to avoid the LFSR from randomly powering up and becoming permanently stuck with the prohibited states. The watermark insertion module consists of adders, adder/subtractor, multipliers, and shifters that implement the watermarking insertion described in (66). The controller is designed as an eight state FSM.

4.1.5. *A Visible Watermarking Chip*

A watermarking VLSI architecture that can insert two visible watermarks in images is proposed by Mohanty et al. (17; 40) along with the introduction of the secure digital camera (SDC) for online copyright protection. Either of the two watermarks can be inserted depending on the requirements of the user. The chip is implemented using 0.35μ CMOS technology, occupies an area of $3.34 \times 2.89mm^2$ and consumes $6.9286mW$ when operated at $3.3V$ and $292.27MHz$ frequency.

Two spatial domain visible watermarking algorithms have been implemented in this proposed chip. The first algorithm implemented is the visible watermarking proposed by Braudaway, et. al. (67) for protection of publicly available images. The second algorithm is the spatial domain visible algorithm proposed by Mohanty et al. (31). Let I be the original image, W be the watermark image and I_W be the watermarked image. The watermarked image for the first algorithm is obtained as follows:

$$I_W(m,n) = \begin{cases} I(m,n) \; + \; \left(\dfrac{\alpha_I}{903.3}\right) W(m,n)I(m,n) & \text{for } I(m,n) \leq 2 \\ I(m,n) \; + \; \left(\dfrac{\alpha_I C_1}{6.0976}\right) W(m,n)I(m,n) & \text{for } 2 < I(m,n) \leq 64 \\ I(m,n) \; + \; \left(\dfrac{\alpha_I C_2}{6.0976}\right) W(m,n)I(m,n) & \text{for } 64 < I(m,n) \leq 128 \\ I(m,n) \; + \; \left(\dfrac{\alpha_I C_3}{6.0976}\right) W(m,n)I(m,n) & \text{for } 128 < I(m,n) \leq 192 \\ I(m,n) \; + \; \left(\dfrac{\alpha_I C_4}{6.0976}\right) W(m,n)I(m,n) & \text{for } 192 < I(m,n) < 256. \end{cases} \tag{8}$$

Where α_I is a global scaling factor and $C_1, C_2, C_3,$ and C_4 are linear regression coefficients used to linearize the cubic root. In the second algorithm the watermarked image blocks are obtained by modifying the k^{th} original image block i_k as shown below:

$$i_{W\,k} \; = \; \alpha_k \, i_k \; + \; \beta_k \, w_k \qquad k = 1, 2 \ldots, \tag{9}$$

where $i_{W\,k}$ is the k^{th} watermarked image block, α_k and β_k are scaling and embedding factors respectively, depending on local and global host image statistics. The scaling and embedding factors for the k^{th} image block are obtained using the following equations:

$$\begin{aligned} \alpha_k \; &= \; \alpha_{min} \; + \; (\alpha_{max} - \alpha_{min}) \; \frac{1}{\hat{\sigma_{k}}_I} exp\left(-(\hat{\mu_{k}}_I - \hat{\mu}_I)^2\right), \\ \beta_k \; &= \; \beta_{min} \; + \; (\beta_{max} - \beta_{min}) \; \hat{\sigma_{k}}_I \left(1 - exp\left(-(\hat{\mu_{k}}_I - \hat{\mu}_I)^2\right)\right). \end{aligned} \tag{10}$$

Where $\hat{\mu_{k}}_I$, $\hat{\mu}_I$, and $\hat{\sigma_{k}}_I$ are respectively, the normalized mean gray value of k^{th} image block, the normalized standard deviation of the gray values of the k^{th} image block, and normalized mean gray value of the image.

The proposed architecture is shown in Fig. 4, which can insert one of the visible watermarks in the input host image. It consists of several units, such as α_k and β_k calculation units, edge detection unit, register files, comparator, multiplier and adder. The α_k and β_k calculation unit computes the scaling and embedding factors for the watermark insertion. The edge detection unit detects the edges of the host image using first order derivatives. The constants used during the watermarking process are stored in a register file. The adder and multipliers do the pixel processing. Both the α_k and β_k calculation units and the edge detection unit are constructed using adder, adder/subtractor, multiplier, divider, comparator, accumulator, and exponential units. The select signal selects between first and second algorithm. A five state finite-state-machine (FSM) controller drives the datapath.

4.1.6. *An Invisible-Robust Watermarking Image Sensor*

A CMOS technology based active pixel sensor (APS) is developed by Nelson et. al. (60) that has built-in functionality for watermarking. The proposed circuit contains additional components for sensor-specific watermark generation. The layout was designed using TSMC $0.18\mu m$ CMOS process that operates at $1.8V/3.3V$.

4.1.7. *A Visible Watermarking Image Sensor*

In (61), Lukac and Plataniotis have presented a single-sensor camera that has integrated capability for visible watermarking. The visible watermarks are added during the color filter array (CFA) data acquisition process and then images are made available for storage or distribution.

4.2. *DCT Domain Watermarking Architecture and Chip*

4.2.1. *An Invisible-Robust Watermarking Chip*

An image watermarking chip capable of inserting invisible-robust watermarks in the DCT domain has been designed by Tsai and Lu (18). The watermark system embeds a pseudo-random sequence of real numbers in a selected set of DCT coefficients. The watermark is extracted without using the original image, which is a major advantage

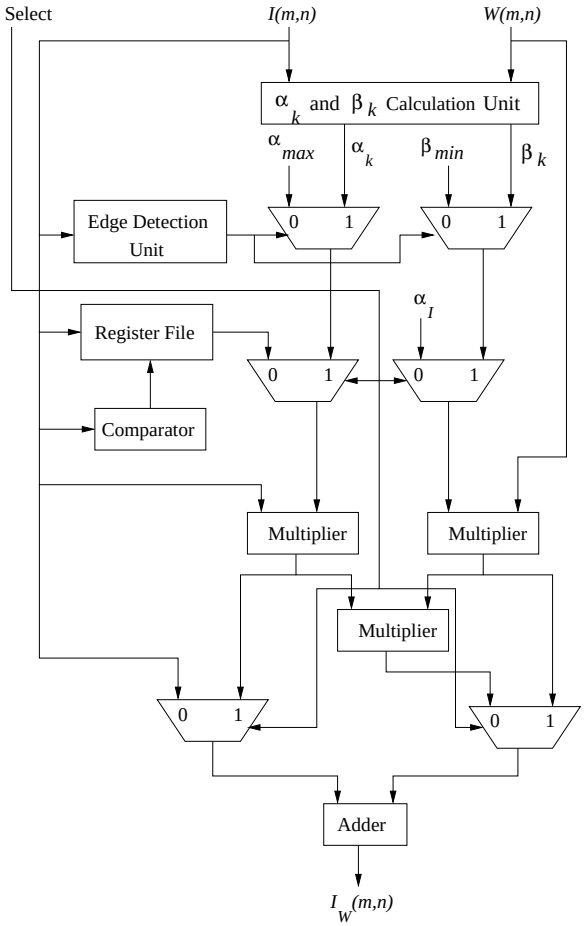


Fig. 4. Architecture for Spatial Domain Visible Watermarking (17; 40)

of this hardware based watermarking scheme. The watermark chip is implemented using TSMC $0.35\mu m$ CMOS technology and occupies a die size of $3.064 \times 3.064mm^2$ for 46374 gates. The chip consumes $62.78mW$ power when operated at $50MHz$ with $3.3V$ supply voltage. Since the original watermark need not be stored for extraction purposes, the proposed method is memory efficient, but the power consumption of the chip is fairly high.

The watermark sequence is constructed from a Gaussian random number generator $N(0, 1)$. The length of the watermark sequence is the number of original DCT coefficients to be watermarked. The watermark is inserted to three significant image DCT coefficients of each nonoverlapping 8×8 block. The strength of the watermark is based upon the average brightness and texture complexity. The image blocks are classified as, (1) dark and weak texture (strongest watermark), (2) bright and strong texture, and (3) rest (weakest watermark). If w_i represent the watermark sequence (where, $0 \leq i \leq n$), $F_k(u, v)$ denotes the DCT coefficients of the k^{th} image block, and α_k represents a scaling factor, then the watermarked DCT coefficients are computed as follows :

$$F_k^i(u,v) = \begin{cases} F_k(u,v) + \alpha_k w_i, \ 3k < i < 3(k+1), \ (u,v) \in \{(0,1),(1,0),(1,1)\}, \\ F_k(u,v) \qquad\qquad\qquad\qquad\qquad \text{otherwise.} \end{cases} \tag{11}$$

The scale factors α_k are 2, 6, and 9 for class 1, class 2, and class 3 image blocks, respectively. The watermark extraction is based on the correlation of the original and extracted watermarks. The watermark is extracted from possibly corrupted watermarked image DCT blocks.

The watermarking chip is designed as an integrated module with a JPEG encoder, which consists of other modules such as hash, DCT/IDCT, quantization, zigzag scan, VLC encoder, etc. Fig. 5 shows the architecture for watermark-

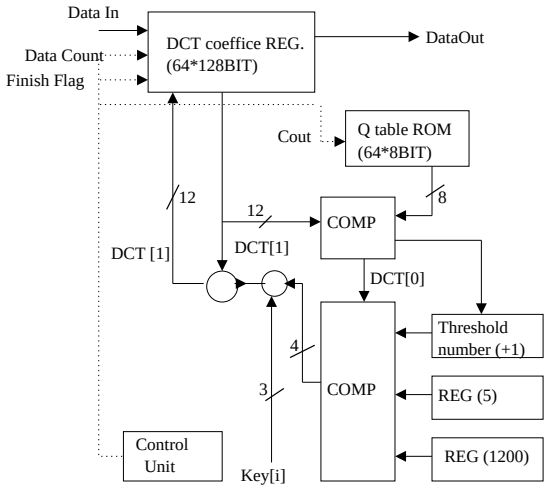
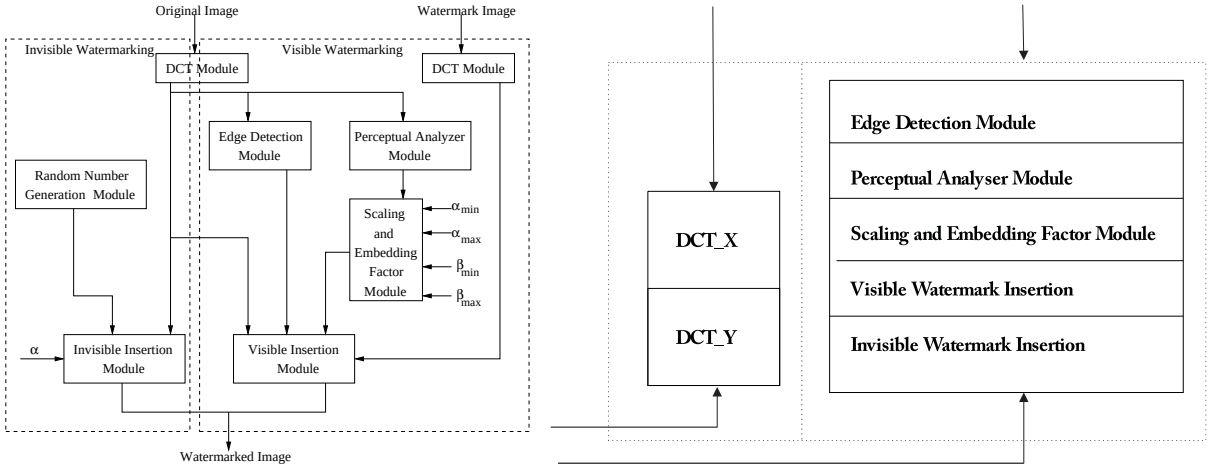


Fig. 5. Architecture for Watermarking Embedding Module (18)



(a) Datapath Architecture (b) Dual Voltage and Frequency Operation

Fig. 6. Low Power DCT Domain Invisible and Visible Watermarking Chip (39; 41)

ing functionality. A register is used to store the DCT coefficients, ROM is used to store the watermark sequence, three other registers are used to store three different threshold numbers and a comparator is available for comparison during the watermark extraction process. A controller implemented as a finite state machine (FSM) performs the watermarking tasks using the datapath.

4.2.2. *A Low Power Invisible-Robust and Visible Watermarking Chip*

This is the first ever low power watermarking chip available in the literature. A solution that has both invisible and visible DCT domain watermark insertion functionalities is developed in (39; 41). The significant feature of this chip is the low power design using advanced VLSI techniques, such as multiple supply voltages, dynamic frequency clocking, and clock gating. The performance improvement is done with a three stage pipeline and parallel architecture. A prototype chip is designed and verified using TSMC 0.25μ CMOS technology. It runs at a dual frequency of $280MHz$ and $70MHz$ and at a dual voltage of $2.5V$ and $1.5V$ and contains $1.4M$ transistors. The average power consumption of the chip is estimated to be $0.3mW$, which is 81% less than single supply voltage and single frequency operation.

The spread spectrum invisible watermarking algorithm proposed by Cox et al. (23; 24; 25) and the DCT domain visible watermarking algorithm proposed by Mohanty et al. (68; 69; 19) are implemented. In the invisible insertion

algorithm, the watermark $X = x_1, x_2, ..., x_{1000}$ is computed where each x_i is chosen according to $N(0,1)$, where $N(0,1)$ denotes a normal distribution with mean 0 and variance 1. The watermark x_i is inserted in the DCT domain of the image by setting the frequency components in the original image C_I to obtain watermarked image C_{I_W} using the following:

$$C_{I_W}(m,n) = C_I(m,n) * (1 + \alpha x_i)\,. \tag{12}$$

Where α is the watermarking strength. The visible insertion algorithm is as follows: The original image I (one to be watermarked) and the watermark image W are divided into blocks. The DCT coefficient C_I for all the blocks of the original image are calculated. The scaling and embedding factors for each block are computed as below:

$$\begin{aligned} \alpha_k &= {\sigma^*}_{AC\,I_k}\ \ exp\left(-({\mu^*}_{DC\,I_k} - {\mu^*}_{DC\,I})^2\right) \\ \beta_k &= \frac{1}{{\sigma^*}_{AC\,I_k}}\ \ \left(1 - exp\left(-({\mu^*}_{DC\,I_k} - {\mu^*}_{DC\,I})^2\right)\right). \end{aligned} \tag{13}$$

The α_k and β_k are scaled to the range $(\alpha_{min}, \alpha_{max})$ and $(\beta_{min}, \beta_{max})$, respectively. The edge blocks are determined, and the α_k and β_k for edge blocks are taken to be α_{max} and β_{min}, respectively. The DCT coefficients C_W for all the blocks of the watermark image are calculated. The visible watermark is inserted in the host images block-by-block and the watermarked image block is obtained as follows:

$$c_{I_W\,k} = \alpha_k\, c_{I\,k} + \beta_k\, c_{W\,k}. \tag{14}$$

The two algorithms are modified to make them hardware amenable before developing their VLSI architectures and custom IC.

Fig. 6(a) shows the proposed architecture and Fig. 6(b) shows how different modules are operated at different supply voltages and frequencies to minimize power consumption. The modules used for invisible watermark insertion are DCT, random number generator, and invisible insertion. After the DCT coefficients of the host image are calculated using the DCT module, the insertion module adds random numbers to them. The α parameter is also given as input to the insertion module. The five modules involved in visible watermarking are DCT module, edge detection module, perceptual analyzer module, scaling and embedding factor module, and visible watermark insertion module. The edge detection module determines the edge blocks in the original image. The scaling factor α_k and the embedding factor β_k are computed by the scaling and embedding factor module. The last module is the visible watermark insertion module. It serves the purpose of inserting the watermark into the original image.

4.2.3. *GPU Based Invisible-Robust Watermarking*

In (38), Mohanty et. al. have demonstrated the use of the graphics process unit (GPU) for invisible-robust image watermarking. This research advocates that the native pipeline available in a GPU can be utilized for real-time watermarking. It is anticipated that the GPU can be utilized in resource constrained environments for high-performance real-time computing.

4.3. *Wavelet Domain Watermarking Architecture and Chip*

4.3.1. *An Invisible-Robust Watermarking VLSI Architecture*

Hsiao, Tai, and Chang (62; 63), introduced a progressive watermarking module in the framework of an image encoder. They designed an encoder-decoder for high speed image/video compression application using an embedded zero wavelet (EZW) tree algorithm. A progressive digital watermarking scheme is included in the EZW encoder-decoder for copyright protection.

The watermarking algorithm that has been implemented here is a progressive invisible-robust algorithm which can detect the watermark information from partially received data. The watermarking scheme embeds pseudorandom numbers along with wavelet coefficient and the detector process involves computation of correlation between the received and original watermark sequences. The pseudorandom number generated watermark sequence is inserted in the bit position next to the first non-zero bit of each significant wavelet coefficient.

4.3.2. *An Invisible-Robust Watermarking Module*

A blind watermarking algorithm is proposed and its FPGA based real-time hardware implementation using Altera is presented by Seo and Kim (64). In this scheme one or more image bit-planes of the 2-dimensional discrete wavelet transform (2DDWT) DC coefficients are replaced with the watermark. The designed watermarking hardware module operates at $82MHz$, and uses 4037 LABs (24% of logic array blocks) and 85 ESBs (3% of embedded system blocks) of an Altera APEX20KC EP20K400CF672C7 device. The watermarking is integrated with a JPEG 2000 codec, which has an operating frequency of $66MHz$.

The watermark is constructed from binary numbers. The DC coefficient of the DCT is chosen as the candidate for the watermark insertion as it is not quantized during the compression process, thus making the watermark highly robust against compression attacks. At the same time it is ensured that the bit plane is selected carefully so that the watermark is imperceptible to the human eye as the DC sub-band contains the most important information. In the watermarking insertion process, the watermark is decomposed into a 4 level 2DDWT. The binary watermark is inserted in the second bit-plane, the third bit-plane from the LSB-plane. In another variation of insertion, multiple bit-planes are chosen based on the outcome of an LSFR (linear feedback shift register). In this case, the bit-plane selection is a two step process as follows: Suppose that bit-plane p is to be chosen from k bit-planes. The initial value of LFSR is a secret key. Then, k parallel outputs of LFSR $(I_{k-1}, I_{k-2}, ..., I_0)$ are chosen followed by computation of p as, $p = \left[I_{k-1}.2^{k-1} + I_{k-2}.2^{k-2} + ... + I_0.2^0\right] \mod k$. The watermark is extracted following the reverse steps of the insertion process. The watermark is claimed to be robust to JPEG compression attacks, blurring attacks, and sharpening attacks.

4.3.3. *A Visible Watermarking Chip*

The authors in (65) propose an adaptive discrete wavelet transform (DWT) based visible watermarking design. Host image and watermark are transformed into three - level multi - resolution structures. The host image signal is divided into two sequences with same pattern length. Processing time is reduced using two - path parallel processing architecture. The signal is sent to different processing elements by the demultiplexers. The watermark image is embedded by modifying the coefficients of the image.

5. Video Watermarking Hardware Systems

In this section, we discuss watermarking circuits and systems designed for video. A comparative view of the proposed image watermarking chips is provided in Table 2. We have arranged them as per their working domain.

5.1. *Spatial Domain Watermarking Architectures and Chips*

5.1.1. *An Invisible-Robust Watermarking Module*

A real-time watermarking embedder-detector for broadcast monitoring in the framework of a VLIW DSP processor is presented in (70; 71; 76; 77). The authors present the European Esprit project VIVA that aims at intellectual property protection of professional TV broadcast surveillance systems. In the insertion procedure, pseudo-random numbers are added to the incoming video stream based on the luminance value of each frame and the watermark detection is based on the calculation of correlation values.

The watermarking embedding and detection is carried out in the spatial and Fourier domains, respectively. The watermark embedding consists of addition of pseudorandom numbers to the video stream. A user key is considered to be the initial sequence which is then shifted and summed to generate n basic pseudorandom patterns, out of which one pattern is chosen, thus creating a key dependent pseudorandom number. The size of the pseudorandom number is $N \times M$, which is the same as the video frame size. The depth of insertion determines the robustness and imperceptibility of the watermark. While large values of watermark make it more robust, small values make it less visible. The algorithm is refined with a visual model to increase robustness as well as invisibility. The watermark detector retrieves the key if the watermark is present and does not need the original video stream. For detection, a group of G frames are folded and summed to a buffer of size $N \times M$, which is then correlated with each of the n basic pseudorandom numbers used during the embedding process. The calculation of correlation is equivalent to computation of two-dimensional cyclic convolution which can be efficiently performed in the Fourier domain as a product followed by inverse Fourier

Table 2
Video Watermarking Hardware Proposed in Current Literature

Research Work	Design Type	Watermarking Type	Multimedia Object	Working Domain	Chip Statistics
Strycker et. al. (70; 71)	DSP processor board	Invisible Robust	Video	Spatial Fourier	$100MHz$
Maes et. al. (35)	FPGA board Custom IC	Invisible Robust	Video	Spatial Fourier	$17\ kG$ Logic $14\ kG$ Logic
Tsai and Wu (72)	Custom IC	Invisible-Robust	Video	Spatial	NA
Brunton and Zhao (37)	GPU	Invisible-Fragile	Video	Spatial	NA
Mathai et. al. (73; 16)	Custom IC	Invisible Robust	Video	Wavelet	0.18μ, $3.53mm^2$, $1.8V$, $75MHz$, $160mW$(approx.)
Vural et. al. (74)	Architecture	Invisible-Robust	Video	Wavelet	NA
Petitjean et. al. (75)	FPGA board DSP processor	Invisible Robust	Image Video	Fractal	$50MHz$ takes $6\mu sec$ $250MHz$ takes $118\mu sec$

transformation. If the correlation so obtained, known as decision variable, exceeds the predefined threshold the video is declared as watermarked.

The Philips Trimedia TM-100 VLIW DSP processor running at $100MHz$ was chosen as the experimental platform. The number of basic patterns $n = 2$, the size assumed to be $N \times M = 128 \times 128$, the number of frames, $G = 12$, and the payload is 8 bits. With careful selection of algorithms to facilitate real time operation the insertion and detection schemes are implemented. The generates SIMD (single instruction multiple data) operations from high-level description to improve performance.

5.1.2. *An Invisible-Robust Watermarking Chip for DVD*

Maes et al. (35) present the Millennium watermarking system proposed for copy right protection of DVD video. Some specific issues, such as watermark detector location and copy generation control have been discussed. The watermark insertion consists of addition of a scaled version of the watermark to the original signal in the spatial domain, while the detection is done calculating the correlations as multiplications in the Fourier domain. The schemes are the same as that used in (70; 71; 76; 77), the only difference being the use of Gaussian random numbers as the watermark instead of pseudorandom numbers. Another important aspect is that in the case of DVD watermarking, the amount of memory usage has to be kept minimal.

A real-time watermark detector is implemented on three different platforms, such as on high-end Silicon Graphics workstation, on a Trimedia processor board and on a FPGA board. The authors suggested that the FPGA platform is more relevant for DVD watermarking. A comparative view of FPGA implementation and customized IC implementation is given. The customized IC implementation is less costly in terms of silicon area and also more secure due to use of the random number generator.

5.1.3. *An Invisible-Robust Watermarking Chip*

Tsai and Wu (72) implemented a watermarking system suitable for raw video sequences using the spread spectrum technique. The proposed architecture is implemented in standard video encoders such as MPEG-1/2/4, H.26x, Motion JPEG, to achieve real-time intellectual property protection.

The embedding algorithms consist of three steps. First the watermark image (video frame) is read and binary transformation is performed using a chaotic function. Then, a pseudo-noise sequence of length 63 and 31 is produced for both spatial and chaotic transformed domains. Finally the insertion is performed using human visual characteristics

that dictate the weighting factor value α for watermark insertion to be 16 for edge blocks and 4 otherwise. The watermark detection involves computation of correlation using the original pseudorandom noise used during the watermark insertion.

The authors have built a watermarking system in Verilog. This could be combined with an MPEG encoder to compress and protect the video sequence at the same time. The function IP is divided into three modules namely datapath and inserting component, pseudo noise generator and watermarking information generator. The pseudo-noise generator includes 63 and 31 bits length PN code. The chaotic and error correction function is included into the third module. The embedded watermarking system can be configured into spatial embedded watermark mode. This design assigns a logo into each frame in the MPEG encoder efficiently to achieve real time IP protection on a digital video capture device.

5.1.4. *An Invisible-Fragile Watermarking using GPU*

Brunton and Zhao (37) present an invisible-fragile watermarking system using a GPU that provides real-time operational capability. The authors have pointed out that the current GPUs do not allow bit-wise operation. All the texture data are converted to floating-point 4-vectors in the fragment shaders. A simple vertex shader has been used for every step of insertion and extraction process.

5.2. *Wavelet Domain Watermarking Architecture and Chip*

5.2.1. *An Invisible-Robust Watermarking Chip*

Mathai et al. (16; 73) present a hardware implementation of the video watermarking algorithm proposed in (70; 71; 76; 77). Floating point datapath architectures for both watermarking embedder and detector are proposed. The chip is implemented using 0.18μ CMOS technology. It is estimated that at $1.8V$ the embedder consumes $60mW$ and the detector consumes $100mW$ power while operating at $75MHz$. The CMOS processor core occupies $3.53mm^2$ and watermarks at a rate of $3M$pixels$/sec$.

5.2.2. *An Invisible-Robust Chip*

In (74), Vural et. al. have presented a video watermarking scheme called "traceable watermarking" targeted for digital cinema. The large scale integration (LSI) watermarking encoder is targeted for JPEG2000.

5.3. *An Invisible-Robust Fractal Domain Watermarking Module*

In (75) Petitjean et. al. have proposed a real-time video watermarking scheme that embeds the watermark using fractal approximation. The extraction process is just the reverse of the insertion and has the same complexity. Both watermark encoder and decoder have been ported to DSP and VLIW processors. The implementation on general purpose PC was time efficient.

To increase the robustness of the watermark, the insertion process exploits the affine transformation invariance properties of fractal coding. The watermark insertion process consists of three steps: formatting and encryption, cover generation, and embedding. In the formatting and encryption step, the message bits to be hidden are over-sampled and duplicated to obtain a watermark of the size of the host image (a video frame). The watermark is encrypted with pseudorandom noise generated using an XOR. The cover signal is generated as the difference of the original image and its fractal approximation. The cover is modulated with the watermark which involves forcing some of the cover pixels to zero depending on the sign and the watermark bit. The watermarked image is obtained by adding the modulated cover with fractal approximation. The watermarked extraction is the opposite process to that of the insertion process. The fractal approximation of the watermarked image under test is computed; the cover is then decoded using demodulation.

6. Challenges and Limitations Watermarks and Watermarking Schemes

Watermarking research has many technical challenges. The robustness and imperceptibility trade-off makes the research quite interesting. To attain imperceptibility, the watermark should be added to the high frequency components of the original signal. On the other hand, for robustness the watermark can be added to the low frequency components only. Thus, the watermarking scheme can be successful if the low frequency components of the original signal are used as the host for watermark insertion. In this section, we discuss the various technical issues related to watermarking, such as properties of the human visual system and spread-spectrum communication, which are commonly exploited for making watermarking schemes successful (78; 79; 80; 81; 82; 83; 84; 85; 86; 31; 68; 22). We also discuss the limitations of watermarking systems from the commercial applications point of view and examine standardization initiatives undertaken for watermarking which are necessary in order for it to be useful in a court of law.

6.1. *Properties of Visual Signals*

Since image and videos are visual signals, it is necessary to understand the behavior of visual signals in order to find ways to hide additional information in them. Visual signals are generally recognized as amplitude plots, intensity versus space displays of image information and intensity versus space and time displays of video scenes. These waveforms reveal a lot of information about the properties of the signals. Some of the properties of visual signals are listed below.

6.1.1. *Nonstationarity*

Nonstationarity property is common to all signals. Image and video signals contain a wealth of segments of flat or slowly changing intensity, as well as edges and textured regions. While the edges need to be preserved to maintain perceptual quality, the textured regions need to be judiciously used to store additional information.

6.1.2. *Periodicity*

There exist line to line and frame to frame periodicity in image and video signals. They are not exactly periodic but there exists redundancy between frames and lines. These redundancies are exploited in any compression scheme, and need to be considered during the watermarking process.

6.1.3. *Power Spectral Density*

The visual signals tend to have low-pass frequency spectra in a global or longtime average sense. On the other hand, in short-term frequency analysis in a local sense, say in a particular region of the image, parts of the visual signals either have all-pass or high-pass frequency spectra. This property can be taken into account while developing adaptive watermarking algorithms.

6.1.4. *Properties of Color Signals*

In a typical signal, the UV or IQ (chrominance) components have lower energy and lower bandwidth compared to the luminance component Y. That is why in data compression algorithms, these components are subsampled and use a coarser quantization. This feature needs to be considered while developing watermarking algorithms for color image or video.

6.2. *Properties of the Human Visual System (HVS)*

The success of any watermarking scheme lies in making the best use of the human visual system (HVS). In this section, we discuss the various properties of the human visual system which are exploited in designing watermarking algorithms (86; 87; 78; 85; 14).

6.2.1. *Brightness sensitivity*

The human eye is sensitive in perceiving a low intensity signal in the presence of backgrounds of different intensity. As the surrounding region intensity is increased, the relative intensity in dark areas is reduced and the sensitivity in the light areas is increased. When the mean value of the noise square is the same as that of the background, the noise square tends to be most visible against a mid-grey background. This characteristic is known as Weber's law. This means that the eye has high sensitivity at low intensity levels and greatly reduced sensitivity at high intensity levels.

6.2.2. *Frequency sensitivity*

From psychovisual studies it has been found that the human visual system has a general bandpass characteristic with peak sensitivity between 3 and 4 cycles per degree and reduced sensitivity at higher and lower spatial frequencies. The strength of the watermark levels in a particular spatial frequency has to therefore be inversely proportional to the relative sensitivities of the corresponding spatial frequencies.

6.2.3. *Texture sensitivity*

The visibility of distortion depends on the background texture. The distortion visibility is low when the background has a strong texture. In a highly textured image block, energy tends to be more evenly distributed among the different DCT coefficients. In a flat-featured portion of the image the energy is concentrated in the low frequency components of the spectrum. This indicates that in strong texture regions more watermark signal can be added.

6.2.4. *Just Noticeable Distortion (JND) and Minimally Noticeable Distortion (MND)*

The properties of visual signals can be studied by performing spatio-temporal local analysis of the input signal. These properties then can be used to derive a perceptual distortion threshold. This threshold can be a function of time, space or frequency. If the distortion caused by the watermarking algorithm is at or below the threshold, then the degradation in the signal quality is imperceptible. The threshold expresses a critical distortion profile which is also known as just noticeable distortion (JND). Similarly, minimal noticeable distortion (MND), defines a suprathreshold generalization of itself, which is not transparent, but still perceptually optimum.

6.3. *How much Watermark Signal to add and Where?*

Watermarking is a reliable communication process in which the watermark signal is reliably transmitted in the host multimedia signal (88; 89; 90; 91; 92). In (82), reliable communication is proven to be theoretically possible with a condition on information rate not to exceed the channel capacity (80; 93). Information theory can be used to derive the rules to decide about the strength of watermark requirement and location of the watermark.

The strategy for communicating the watermark is as follows. As a watermark should be imperceptible, the signal to noise ratio (SNR) is severely limited. Reliable communication can only be assured by increasing the bandwidth to compensate for the poor SNR. Hence, during the watermarking process the maximum number of suitable transform domain coefficients should be exploited for hiding information in the image. Watermarking may be considered as being an application of spread spectrum communications. The Shannon limit may be approached by applying error control codes and robust error correction techniques may be used, if necessary. This provides the answer how much watermark can be reliably added to the host signal. The answer to the second part of the question on "where to embed?", is that the watermark should be placed in those areas where the local noise variance is smaller than the threshold and not at all in those areas where the local noise variance exceeds the threshold.

6.4. *Spread Spectrum Communications*

It is clear that the watermark should not be placed in the perceptually insignificant regions of the image or its spectrum since many common signals and geometric processes attack these components. For example, a watermark placed in the high frequency spectrum of an image can be easily eliminated with little degradation to the image by any process that directly or indirectly performs low-pass filtering. The problem then becomes how to insert a watermark into the most perceptually significant regions of a spectrum without such alterations becoming noticeable. Clearly, any

spectral coefficient may be altered, provided such modification is small. However, minute changes are very susceptible to noise.

To solve this problem, the frequency domain of the image or sound is viewed as a communication channel, and correspondingly, the watermark is viewed as a signal that is transmitted through it. Attacks and unintentional signal distortions are thus treated as noise to which the immersed signal must be immune. Thus, watermarking can be considered as an application of spread spectrum communications (5). In spread spectrum communication, one transmits a narrow band signal over a much larger bandwidth such that the signal energy present in any single frequency is imperceptible. Similarly, the watermark is spread over very many frequency bins so that the energy in any one bin is very small and undetectable. Nevertheless, because the watermark verification process knows the location and context of the watermark, it is possible to concentrate these many weak signals with a high SNR. The relevant mathematical framework for spread spectrum watermarking techniques has been developed and used in (23; 25; 94; 95).

6.5. *Challenges for Designing Hardware based Schemes*

It is evident that the watermarking chip is absolutely necessary for low power, real-time performance, high reliability, low cost applications, and also for easy to integrate with existing consumer electronic devices. Software-only schemes are not adequate for these purposes. The design of a watermarking chip involves trade-offs of performance, power consumption, silicon area, memory requirement, and integrability. For real-time watermarking, the chip design must be done using a pipelined or a parallel architecture, but such architectures consume large amounts of both leakage and dynamic power. Thus, the watermarking algorithm should be designed in such a way that it can suitably be implemented in hardware with as minimal a pipelining and parallel architecture as necessary. The on-chip memory requirement becomes an issue when the algorithms need original multimedia data during the detection process. With the increase in memory, the cost of the chip increases, and also there is an increase in power consumption. Implementing floating point operations is a strong technical challenge from a hardware design prospective. Similarly, the division and multiplication operations in the algorithms usually consume more power compared to the addition or subtraction operations. Hence, a watermarking algorithm involving more multiplication/division operations compared to addition/subtraction operations gives less room to the hardware designer for low power and high performance design. It is well known that most of the watermarking algorithms use random number or pseudo-random numbers. Designing of low power random number generators and reliable storage of random number keys in hardware is a design challenge. The linear feedback register (LFSR) used to generate pseudo-random numbers is a sequential circuit with combinational feedback logic. LFSRs need to be designed such that typical problems, such as getting stuck in the prohibited state, can be eliminated. The watermarking chip should be designed and placed in consumer electronic appliances such that it does not affect the normal operation of the host device in any way; in other words, the watermarking chip should not be an overhead for the host device.

6.6. *Limitations of Watermarks*

In the case of visible watermarking, the declaration of copyright is obvious from the visible mark (96; 68). In this scheme, the watermark needs to cover a large important portion of the host multimedia object while preserving the quality of the host object. If the cost of removing the watermark from the object is as expensive as the creation of the new object, it will be sufficient to make it useful in practice. Of course, it is valid only as long as the visible watermarks are registered with some law enforcement authority as being associated with an owner. However, the use of visible watermarks is limited.

On the other hand, for invisible watermarking techniques the issues are quite complicated. All the research work undertaken by the watermarking community has assumed robustness as the most useful property and highly robust watermarking schemes have been developed (12). However, for an invisible watermark, the robustness property alone is not sufficient to guarantee content protection. Any watermark can suffer from counterfeit original attack and can result in proving multiple ownerships. This shows that invertible and quasi-invertible invisible watermarking techniques cannot resolve rightful ownership. Moreover, any one can successfully prove the ownership of any object in the absence of standardization of watermarking schemes (97; 12).

Digital watermarking can not prevent illicit activity – it can only provide evidence of such an act after it has taken place (10). On the other hand, cryptographic techniques can protect data. Thus, digital watermarking can also be used as a back up protection along with cryptographic technology (98).

6.7. *Standardization*

In order that digital watermarking finds commercial applications in the legal framework, its standardization is necessary. Invisible watermarking has the greatest need for standardization (46; 97), which requires the formation of a standards body (98; 1; 46; 35; 3).

The Digital Millennium Copyright Act (DMCA) can be used in the case of deliberate removal or attacks of watermark (98) thus providing the minimal legal support for watermarking schemes. Several cross-industry organizations, such as the Copy Protection Technical Working Group (CPTWG), the Digital Audio-Visual Council (DAVIC), the International Federation of Phonographic Industry (IFPI), and the Secure Digital Music Initiative (SDMI) are putting maximum effort to come up with digital watermarking standards that will allow its use and acceptance.

7. Conclusions

In this paper, we have considered various aspects of digital watermarking. We have provided a broad perspective of watermarking including its basic theory, its types, technical requirements, and various other issues. We have surveyed watermarking chips proposed in the current literature. The need of low power, low cost, high performance real time operation, and high security with watermarking done at the data acquisition stage drives the VLSI implementation of the watermarking schemes. The VLSI implementation is still at a relatively early stage of development and a lot more research is needed before a complete VLSI architecture with all the desired characteristics can be obtained. The collaborative work of the VLSI, signal processing, and watermarking communities is required. In particular, with sufficient feedback from the VLSI community, the watermarking algorithms should be designed so that their hardware implementation becomes feasible. Sufficient attention should be provided for the design of ultra low power watermarking chips. Above all, defining the scope and framework for digital watermarking systems addressing legal, political, business and deployment issues acceptable to different parties, such as content creators, music recording companies, movie production houses, book publishers, the open-source software development community, the consumer electronics industry, software companies, government organizations, legal fraternity and end-users, is a huge task and needs to be completed sooner or later.

References

[1] I. J. Cox, M. L. Miller, Electronic Watermarking : The First 50 Years, EURASIP Journal of Applied Signal Processing 2002 (2) (2002) 126–132.

[2] M.M.Yeung, Digital Watermarking, Communications of the ACM 41 (7) (1998) 31–33.

[3] J. Zhao, E. Koch, J. O'Ruanaidh, M. Yeung, Digital watermarking : What will it do for me? and what it won't!, in: Proceedings of SIGGRAPH Conference, 1999, pp. 153–155.

[4] S. Voloshynovskiy, S. Pereira, T. Pun, J. Eggers, J. Su, Attacks on Digital Watermarks: Classification, Estimation-based Attacks and Benchmarks, IEEE Communications Magazine 39 (9) (2001) 118–126.

[5] A. Sequeira, D. Kundur, Communications and information theory in watermarking : A survey, in: Proceedings of SPIE Multimedia Systems and Application IV, Vol. 4518, 2001, pp. 216–227.

[6] F. Mintzer, G. Braudaway, M. Yeung, Effective and Ineffective Digital Watermarks, in: IEEE International Conference on Image Processing (ICIP-97), Vol. 3, 1997, pp. 9–12.

[7] N. Memon, P. W. Wong, Protecting Digital Media Content, Communications of the ACM 41 (7) (1998) 35–43.

[8] R. Barnett, Digital Watermarking : Application, Techniques, and Challengs, IEE Electronics and Communication Engineering Journal (1999) 173–183.

[9] W. Bender, W. Butera, D. Gruhl, R. Hwang, F. J. Paiz, S. Pogreb, Applications for Data Hiding, IBM Systems Journal 39 (3 and 4) (2000) 547–568.

[10] J. Zhao, E. Koch, C. Luo, In business today and tommorrow, Communications of the ACM 41 (7) (1998) 67–72.
[11] F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn, Attacks on Copyright Marking Systems, in: Proceedings of the 2nd International Workshop on Information Hiding, 1998, pp. 218–238.
[12] S. Craver, N. Memon, B. L. Yeo, M. M. Yeung, Resolving Rightful Ownerships with Invisible Watermarking Techniques: Limitations, Attacks and Implications, IEEE Journal on Selected Areas in Communications 16 (4) (1998) 573–586.
[13] F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn, Information Hiding - A Survey, Proceedings of the IEEE 87 (7) (1999) 1062–1078.
[14] M. Swanson, M. Kobayashi, A. Tewfik, Multimedia Data Embedding and Watermarking Technologies, Proceedings of the IEEE 86 (6) (1998) 1064–1087.
[15] F. Bao, Multimedia Content Protection by Cryptography and Watermarking in Tamper-resistant hardware, in: Proceedings of the ACM Workshops on Multimedia, 2000, pp. 139–142.
[16] N. J. Mathai, D. Kundur, A. Sheikholeslami, Hardware Implementation Perspectives of Digital Video Watermarking Algortithms, IEEE Transanctions on Signal Processing 51 (4) (2003) 925–938.
[17] S. P. Mohanty, N. Rangnathan, R. K. Namballa, VLSI Implementation of Visible Watermarking for a Secure Digital Still Camera Design, in: Proceedings of the 17th International Conference on VLSI Design, 2004, pp. 1063–1068.
[18] T. H. Tsai, C. Y. Lu, A Systems Level Design for Embedded Watermark Technique using DSC Systems, in: Proceedings of the IEEE International Workshop on Intelligent Signal Processing and Communication Systems, 2001.
[19] S. P. Mohanty, Digital Watamerking of Images, Master's thesis, Department of Electrical Engineering, Indian Institute of Science, Bangalore, India (1999).
[20] S. Craver, N. Memon, B. L. Yeo, M. Yeung, Can Invisible Watermarks Resolve Rightful Ownerships?, Tech. rep., IBM Research Report, RC 20509 (1996).
[21] F. Hartung, M. Kitter, Multimedia Watermarking Techniques, Proceedings of the IEEE 87 (7) (1999) 1079–1107.
[22] M. Kutter, F. A. P. Petitcolas, A fair benchmark for image watermarking systems, in: Proceedings of SPIE Security and Watermarking of Multimedia Contents, Vol. 3657, 1999, pp. 226–239.
[23] I. J. Cox, J. Kilian, T. Shamoon, T. Leighton, Secure Spread Spectrum Watermarking of Images, Audio and Video, in: Proc IEEE International Conf on Image Processing, Vol. 3, 1996, pp. 243–246.
[24] I. J. Cox, J. Kilian, T. Shamoon, T. Leighton, A Secure Robust Watermarking for Multimedia, in: Proc. of First International Workshop on Information Hiding, Vol. 1174, 1996, pp. 185–206.
[25] I. J. Cox, J. Kilian, T. Leighton, T. Shamoon, Secure Spread Spectrum Watermarking for Multimedia, IEEE Transactions on Image Processing 6 (12) (1997) 1673–1687.
[26] H. Berghel, Watermarking Cyberspace, Communications of the ACM 40 (11) (1997) 19–24.
[27] J. T. Brassil, S. Low, N. Maxemchuk, L. O'Gorman, Electronic Marking and Identification Techniques to Discourage Document Copying, IEEE Journal on on Selected Areas in Communications 13 (8) (1995) 1495–1504.
[28] J. T. Brassil, S. Low, N. F. Maxemchuk, Copyright Protection for the Electronic Distribution of Text Documents, Proceedings of the IEEE 87 (7) (1999) 1181–1196.
[29] H. Guo, N. D. Georganas, A Novel Approach to Digital Image Watermarking Based on a Generalized Secret Sharing Scheme, ACM-Springer Verlag Multimedia Systems Journal 9 (3) (2003) 228–238.
[30] N. P. Sheppard, R. S. Naini, P. Ogunbona, On Multiple Watermarking, in: Proceedings of the ACM Multimedia workshops on multimedia and security: new challenges, 2001, pp. 3–6.
[31] S. P. Mohanty, K. R. Ramakrishnan, M. S. Kankanhalli, A Dual Watermarking Technique for Images, in: Proceedings of the 7th ACM International Multimedia Conference (Vol. 2), 1999, pp. 49–51.
[32] X. S. Hua, J. F. Feng, Q. Y. Shi, Public Multiple Watermarking Resistant to Cropping, in: Proceedings of the 6th international conference on pattern recognition and information processing, 2001, pp. 263–268.
[33] G. Voyatzis, I. Pitas, The Use of Watermarks in the Protection of Digital Multimedia Products, Proceedings of the IEEE 87 (7) (1999) 1197–1207.
[34] B. Chen, G. W. Wornell, Dither Modulation : A New Approach to Digital Watermarking and Information Embedding, in: Proceedings of the SPIE Conference on Security and Watermarking of Multimedia Contents (Vol. SPIE-3657), 1999, pp. 52–57.
[35] M. Maes, T. Kalker, J. P. M. G. Linnartz, J. Talstra, G. F. G. Depovere, J. Haitsma, Digital Watamarking for

DVD Video Copyright Protection, IEEE Signal Processing Magazine 17 (5) (2000) 47–57.
[36] O. M. Alattar, A. M. Alattar, A fast hierarchical watermark detector for real-time software or low-cost hardware implementation, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), 2005, pp. 973–976.
[37] A. Brunton, J. Zhao, Real-time Video Watermarking on Programmable Graphics Hardware, in: Proceedings of Canadian Conference on Electrical and Computer Engineering, 2005, pp. 1312–1315.
[38] S. P. Mohanty, N. Pati, E. Kougianos, A Watermarking Co-Processor for New Generation Graphics Processing Units, in: Proceedings of the 25th IEEE International Conference on Consumer Electronics (ICCE), 2007, pp. 303–304.
[39] S. P. Mohanty, N. Ranganathan, K. Balakrishnan, Design of a Low Power Image Watermarking Encoder using Dual Voltage and Frequency, in: Proceedings of 18th IEEE International Conference on VLSI Design, 2005, pp. 153–158.
[40] S. P. Mohanty, N. Ranganathan, R. K. Namballa, A VLSI Architecture for Visible Watermarking in a Secure Still Digital Camera (S^2DC) Design, IEEE Transactions on Very Large Scale Integration Systems 13 (8) (2005) 1002–1012.
[41] S. P. Mohanty, N. Ranganathan, K. Balakrishnan, A Dual Voltage-Frequency VLSI Chip for Image Watermarking in DCT Domain, IEEE Transactions on Circuits and Systems II (TCAS-II), 53 (5) (2006) 394–398.
[42] I. J. Cox, J. P. M. G. Linnartz, Some General Methods for Tampering with Watermarks, IEEE Journal on Selected Areas in Communications 16 (4) (1998) 587–593.
[43] J. M. G. Linnartz, M. van Dijk, Analysis of Sensitivity Attack Against Electronic Watermarks in Images, in: Proceedings of the 2nd International Workshop on Information Hiding, Vol. 1525, 1998, pp. 15–17.
[44] J. M. G. Linnartz, T. Kalker, G. Depovere, Modelling the False Alarm and Missed Detection Rate for Electronic Watermarks, in: Proceedings of the 2nd International Workshop on Information Hiding, Vol. 1525, 1998, pp. 329–343.
[45] M. Ramkumar, A. N. Akansu, Image Watermarks and Counterfeit Attacks : Some Problems and Solutions, in: Proceedings of the Content Security and Data Hiding in Digital Media, 1999.
[46] F. Mintzer, G. W. Braudaway, A. E. Bell, Opportunities for Watermarking Standards, Communications of the ACM 41 (7) (1998) 57–64.
[47] S. Ravi, A. Raghunathan, P. Kocher, S. Hattangady, Security in embedded systems: Design challenges, ACM Transaction on Embedded Computing Systems 3 (3) (2004) 461–491.
[48] F. A. P. Petitcolas, Watermarking Schemes Evaluation, IEEE Signal Processing 17 (5) (2000) 58–64.
[49] F. Hartung, J. K. Su, B. Girod, Spread Spectrum Watermarking: Malicious Attacks and Counterattacks, in: Proceedings of SPIE Security and Watermarking of Multimedia Contents, Vol. 3657, 1999, pp. 147–158.
[50] F. A. P. Petitcolas, Watermarking Schemes Evaluation, IEEE Signal Processing 17 (5) (2000) 58–64.
[51] S. Pereira, S. Voloshynovskiy, M. Madueo, S. Marchand-Maillet, T. Pun, Second Generation Benchmarking and Application Oriented Evaluation, in: Proceedings of 3rd International Workshop on Information Hiding, 2001.
[52] A. Nikolaidis, S. Tsekeridou, A. Tefas, V. Solachidis, A Survey on Watermarking Application Scenarios and Related Attacks, in: Proceedings of the IEEE International Conference on Image Processing, Vol. 1, 2001, pp. 991–994.
[53] W. Geiselmann, R. Steinwandt, Power attacks on a side-channel resistant elliptic curve implementation, Information Processing Letter 91 (1) (2004) 29–32.
[54] V. Carlier, H. Chabanne, E. Dottax, H. Pelletier, Generalizing square attack using side-channels of an aes implementation on an fpga, in: Proceedings of the IEEE International Conference on Field Programmable Logic and Applications (FPL), 2005, pp. 433–437.
[55] A. Garimella, M. V. V. Satyanarayan, R. S. Kumar, P. S. Murugesh, U. C. Niranjan, VLSI Impementation of Online Digital Watermarking Techniques with Difference Encoding for the 8-bit Gray Scale Images, in: Proceedings of the International Conference on VLSI Design, 2003, pp. 283–288.
[56] A. Garimella, M. V. V. Satyanarayana, P. S. Murugesh, U. C. Niranjan, ASIC for Digital Color Image Watermarking, in: Proceedings of 11th IEEE Digital Signal Processing Workshop, 2004, pp. 292–295.
[57] S. P. Mohanty, N. Ranganathan, R. K. Namballa, VLSI Implementation of Invisible Digital Watermarking Algorithms Towards the Developement of a Secure JPEG Encoder, in: Proceedings of the IEEE Workshop on Signal Processing Systems, 2003, pp. 183–188.

[58] S. P. Mohanty, E. Kougianos, N. Ranganathan, VLSI architecture and chip for combined invisible robust and fragile watermarking, IET Computers & Digital Techniques (CDT) 1 (5) (2007) 600–611.
[59] S. P. Mohanty, R. K. C., S. Nayak, FPGA Based Implementation of an Invisible-Robust Image Watermarking Encoder, in: Lecture Notes in Computer Science, Vol. 3356, 2004, pp. 344–353.
[60] G. R. Nelson, G. A. Jullien, O. Y. Pecht, Cmos image sensor with watermarking capabilities, in: Proceedings of the IEEE International Conference on Circuits and Systems (ISCAS), 2005, pp. 5326–5329.
[61] R. Lukac, K. N. Plataniotis, Secure single-sensor digital camera, IEE Electronics Letters 42 (11) (2006) 627–629.
[62] S. F. Hsiao, Y. C. Tai, K. H. Chang, VLSI Design of an Efficient Embedded Zerotree Wavelet Coder with Function of Digital Watermarking, in: Proceedings of the IEEE International Conference on Consumer Electronics, 2000, pp. 186–187.
[63] S. F. Hsiao, Y. C. Tai, K. H. Chang, VLSI Design of an Efficient Embedded Zerotree Wavelet Coder with Function of Digital Watermarking, IEEE Transactions on Consumer Electronics 46 (3) (2000) 628–636.
[64] Y. H. Seo, D. W. Kim, Real-Time Blind Watermarking Algorithm and its Hardware Implementation for Motion JPEG2000 Image Codec, in: Proceedings of the 1st Workshop on Embedded Systems for Real-Time Multimedia, 2003, pp. 88–93.
[65] Y. C. Fan, L. D. Van, C. M. Huang, H. W. Tsao, Hardware-Efficient Architecture Design of Wavelet-based Adaptive Visible Watermarking, in: Proceedings of 9th IEEE International Symposium on Consumer Electronics, 2005, pp. 399–403.
[66] A. Tefas, I. Pitas, Robust Spatial Image Watermarking Using Progressive Detection, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 3), 2001, pp. 1973–1976.
[67] G. W. Braudaway, K. A. Magerlein, F. Mintzer, Protecting Publicly Available Images with a Visible Image Watermark, in: Proceedings of the SPIE Conference on Optical Security and Counterfiet Deterrence Technique (Vol. SPIE-2659), 1996, pp. 126–132.
[68] S. P. Mohanty, K. R. Ramakrishnan, M. S. Kankanhalli, A DCT Domain Visible Watermarking Technique for Images, in: Proceedings of the IEEE International Conference on Multimedia and Expo, 2000, pp. 1029–1032.
[69] S. P. Mohanty, K. R. Ramakrishnan, M. S. Kankanhalli, An Adaptive DCT Domain Visible Watermarking Technique for Protection of Publicly Available Images, in: Proceedings of the International Conference on Multimedia Processing and Systems, 2000, pp. 195–198.
[70] L. D. Strycker, P. Termont, J. Vandewege, J. Haitsma, A. Kalker, M. Maes, G. Depovere, Implementation of a Real-Time Digital Watermarking Process for Broadcast Monitoring on Trimedia VLIW Processor, IEE Proceedings on Vision, Image and Signal Processing 147 (4) (2000) 371–376.
[71] L. D. Strycker, P. Termont, J. Vandewege, J. Haitsma, A. Kalker, M. Maes, G. Depovere, An Implementation of a Real-time Digital Watermarking Process for Broadcast Monitoring on a Trimedia VLIW Processor, in: Proceedings of the IEE International Conference on Image Precessing and Its Applications (Vol. 2), 1999, pp. 775–779.
[72] T. H. Tsai, C. Y. Wu, An Implementation of Configurable Digital Watermarking Systems in MPEG Video Encoder, in: Proceedings of the IEEE International Conference on Consumer Electronics, 2003, pp. 216–217.
[73] N. J. Mathai, A. Sheikholeslami, D. Kundur, VLSI Implementation of a Real-Time Video Watermark Embedder and Detector, in: Proceedings of the IEEE International Symposisum on Circuits and Systems (Vol. 2), 2003, pp. 772–775.
[74] S. Vural, H. Tomii, H. Yamauchi, Video Watermarking For Digital Cinema Contents, in: Proceedings of the 13th European Signal Processing Conference, 2005, pp. 303–304.
[75] G. Petitjean, J. L. Dugelay, S. Gabriele, C. Rey, J. Nicolai, Towards Real-time Video Watermarking for Systems-On-Chip, in: Proceedings of the IEEE International Conference on Multimedia and Expo (Vol. 1), 2002, pp. 597–600.
[76] P. Termont, L. D. Strycker, J. Vandewege, J. Haitsma, T. Kalker, M. Maes, G. Depovere, A. Langell, C. Alm, P. Norman, Performance Measurements of a Real-time Digital Watermarking System for Broadcast Monitoring, in: Proceedings of the IEEE International Conference on Multimedia Computing and Systems (Vol. 2), 1999, pp. 220–224.
[77] G. Depovere, T. Kalker, J. Haitsma, M. Maes, L. D. Strycker, P. Termont, J. Vandewege, A. Langell, C. Alm, P. Norman, G. O'Reilly, B. Howes, H. Vaanholt, R. Hintzen, P. Donnelly, A. Hudson, The VIVA Project : Digital Watermarking for Broadcast Monitoring, in: Proceedings of the IEEE International Conference on Image

Precessing (Vol. 2), 1999, pp. 202–205.
[78] K. N. Ngan, Adaptive Cosine Transform Coding of Images in Perceptual Domain, IEEE Transactions on Acoustics, Speech and Signal Processing 37 (11) (1989) 1743–1750.
[79] J. L. Mannos, D. J. Sakrison, The Effects of a Visual Fidelity Criterion on the Encoding of Images, IEEE Transactions on Information Theory 20 (4) (1974) 525–536.
[80] R. G. Gallager, Information Theory and Reliable Communications, Wiley, 1968.
[81] A. J. Viterbi, CDMA Principles of Spread Spectrum Communications, Addison-Wesley Inc., 1996.
[82] C. E. Shannon, A Mathematical Theory of Communication, Bell Systems Technical Journal (1948) 379–423.
[83] A. K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, NJ, 1989.
[84] B. Tao, B. Dickinson, Adaptive watermarking in dct domain, in: Proceedings of the IEEE International Conference on Accoustics, Speech and Signal Processing, Vol. 4, 1997, pp. 2985–2988.
[85] N. Jayant, J. Johnston, R. Safranek, Signal Compression Based on Models of Human Perception, Proceedings of the IEEE 81 (10) (1993) 1385–1422.
[86] J. F. Delaigle, C. Devleeschouwer, B. Macq, I. Langendijk, Human Visual Systems Features Enabling Watermarking, in: Proceedings of the IEEE International Conference Multimedia and Expo, 2002, pp. 489–492.
[87] M. S. Kankanhalli, Rajmohan, K. R. Ramakrishnan, Content based watermarking for images, in: Proceedings of the 6th ACM International Multimedia Conference, 1998, pp. 61–70.
[88] S. D. Servette, C. Podilchuk, K. Ramchandran, Capacity Issues in Digital Watermarking, in: IEEE International Conference on Image Processing, ICIP-98, Vol. 1, 1998, pp. 445–449.
[89] M. Ramkumar, A. N. Akansu, Capacity Estimates for Data Hiding in Compressed Images, IEEE Transactions on Image Processing 10 (8) (2001) 1252–1263.
[90] M. Ramkumar, Data Hiding in Multimedia - Theory and Applications, Ph.D. thesis, Department of Electrical and Computer Engineering, New Jersy i Institute of Technology, Newwark, NJ, USA (2000).
[91] R. Sugihara, Practical Capacity of Digital Watermark as Constrained by Reliability, in: Proceedings of the International Conference on Information Technology: Coding and Computing, 2001, pp. 85–59.
[92] M. Ramkumar, A. N. Akansu, Information Theoritic Bounds for Data Hiding in Compressed Images, in: Electronic Proceedings of the IEEE Signal Processing Society Workshop on Multimedia Signal Processing, 1998, pp. 267–272.
[93] J. G. Proakis, Digital Communications, McGraw-Hill, 1995.
[94] F. Hartung, B. Girod, Watermarking of Uncompressed and Compressed Video, Signal Processing 66 (3) (1998) 283–301.
[95] F. Hartung, B. Girod, Digital Watermarking of Raw and Compressed Video, in: Proceedings of the European EOS/SPIE Symposium on Advanced Imaging and Network Technologies, 1996, pp. 205–213.
[96] F. C. Mintzer, L. E. Boyle, A. N. Cazes, B. S. Christian, S. C. Cox, F. P. Giordano, H. M. Gladney, J. C. Lee, M. L. Kelmanson, A. C. Lirani, K. A. Magerlein, A. M. B. Pavani, F. Schiattarella, Towards online Worldwide Access to Vatican Library Materials, IBM Journal of Research and Development 40 (2) (1996) 139–162.
[97] S. Craver, B. L. Yeo, M. Yeung, Technical Trials and Legal Tribulations, Communications of the ACM 41 (7) (1998) 45–54.
[98] A. M. Eskicioglu, E. J. Delp, An Overview of Multimedia Content Protection in Consumer Electronics Devices, Elsevier Signal Processing : Image Communication 16 (2001) 681–699.

Elias Kougianos is currently an Assistant Professor at the Dept. of Electrical Engineering Technology, at the University of North Texas (UNT), Denton, TX. From 1988 through 1997 he was with Texas Instruments, Inc., in Houston and Dallas, TX. Initially he concentrated on process integration of flash memories and later as a researcher in the areas of Technology CAD and VLSI CAD development. In 1997 he joined Avant! Corp. (now Synopsys) in Phoenix, AZ as a Senior Applications engineer and in 2001 he joined Cadence Design Systems, Inc., in Dallas, TX as a Senior Architect in Analog/Mixed-Signal Custom IC design. He has been at UNT since 2004. His research interests are in the area of Analog/Mixed-Signal/RF IC design and simulation and in the development of VLSI architectures for multimedia applications. He is author or co-author of over 30 peer-reviewed journal and conference publications. He is a senior member of IEEE.

Saraju P. Mohanty is currently an Assistant Professor at the Dept. of Computer Science and Engineering, University of North Texas (UNT). His research is in "Design and CAD for Low-Power High-Performance Nanoscale VLSI". This can be classified to the following inter-related categories: (i) Power-Performance Modeling and Optimization for Nanoscale VLSI Circuits, (ii) Design and CAD for Nanoscale Digital and Analog/Mixed-Signal Circuits, and (iii) VLSI Architecture for Multimedia Processing. He develops power, leakage, and performance models, incorporates them in CAD flow through optimization methodology, and demonstrates them through computational intensive applications. He is an author of 60 peer-reviewed journal and conference publications. He serves on the program committee of several international conferences. He is a member of ACM, IEEE, ACM-SIGDA, IEEE-Circuits and Systems Society, IEEE-Computer Society, and IEEE-Consumer Electronics Society.

Rabi N. Mahapatra is currently an Associate Professor at the Computer Science Department and directing the Embedded System Codesign Research Group at Texas A&M University. His current research interests are distributed embedded systems, low-power scheduling, processor safety evaluation, and system-on-chip. He has published more than 90 refereed journal and conference papers and serves on the editorial board for one journal and on the program committee of a numerous of international conferences and workshops. His research has been funded by NSF, DoT and industries like IBM, Boeing, Locke Martin, BAE, and Smith Aerospace. He is a distinguished visitor of the IEEE Computer Society, Ford Fellow and a Senior Member of the IEEE.