A Class of Authentication Digital Watermarks for Secure Multimedia Communication

Liehua Xie and Gonzalo R. Arce, *Fellow, IEEE*

***Abstract*—A new approach to digital signatures for imaging, which adapts well to multimedia communications in lossy channels is introduced. Rather than attaching the signature's bit-string as a file-header, it is invisibly etched into the image using a new watermarking algorithm. The watermark is "nonfragile," tolerating small distortions but not malicious tampering aimed at modifying the image's content. In particular, the rank-order relationship in local areas throughout the lowest level of the DWT is exploited to encode the watermark. An edge-based message digest is used. The signature is in the form of binary data and the wavelet decomposition coefficients are modified according to this binary sequence. The signature is also embedded and tested within the SPIHT compression algorithm. The information capacity is studied and the experimental results confirm a logarithm relation between the bit rate and the quantization level, which is similar to the Shannon's capacity theorem. Experiments are performed to examine the signature's transparencey and robustness.**

***Index Terms*—Bit capacity, content based image authentication, nonfragile digital signature, wavelet compression domain watermark.**

I. Introduction

DIGITAL signatures (DS) are electronic protocols used for the authentication of electronic documents whereby a receiver of a message can verify the identity of the sender and the integrity of the message [1]. With the advent of multimedia communications over the Internet, it is natural, and critical in many applications, to provide security mechanisms in the transmission of imagery data. To this end, the need of digital image signatures emerges for applications where the security, integrity, and authenticity of images are important. Military and forensic imaging are two such areas where the security features of digital signatures are desirable.

Digital signatures do not simply encrypt the entire message with a secret key—although direct cryptography does provides security, it can be a prohibitively computationally expensive approach particularly with large multimedia data sets [2]. Instead, digital signatures encrypt a message digest which is in essence a "fingerprint" of the electronic file. Message digests, such as Rivest's MD5 [3], generate a compact digest of the message using cryptographic one-way hash functions of the message [2]. The message digest represents the message such that even if one bit of the original message is changed, a different message digest would be obtained from the modified message. In addition, it is computationally infeasible for an attacker to device a substitute message that would produce an identical message digest. The crypted message digest is attached to the document and subsequently jointly transmitted or stored. The receiver recovers the original message digest from the digital signature by decrypting it with the sender's public key. He then computes a new message digest from the received message, and if it matches with the one recovered from the digital signature, the receiver is confident that the message was not altered, and that it came from the sender who owns the public key used to check the signature. The generation of a traditional digital signature at the transmitter, and the authentication process carried out at the receiver are illustrated in Figs. 1 and 2.

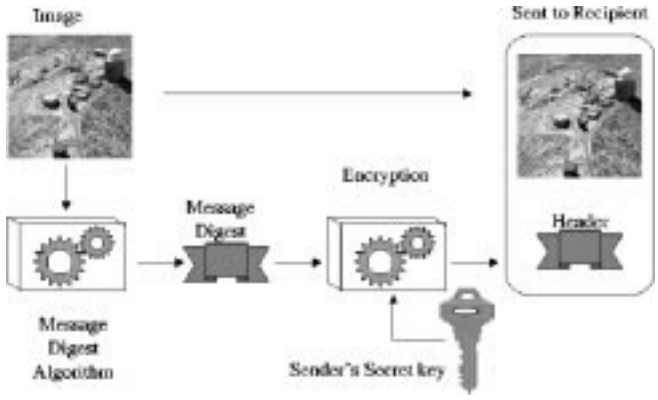

Fig. 1. Traditional DS sender: signing with private key.

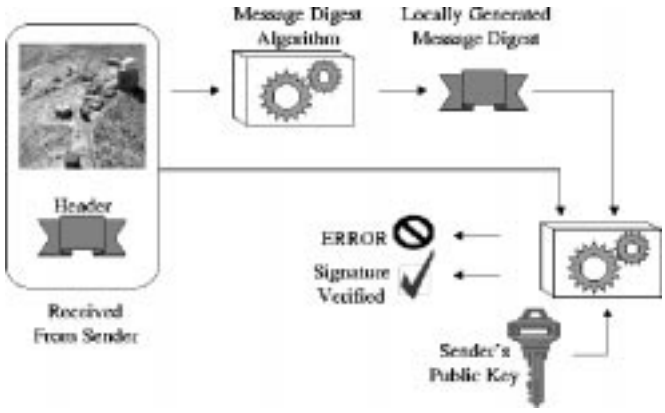

Fig. 2. Traditional DS authentication receiver: verification with public key.

Manuscript received October 25, 1999; revised July 28, 2001. This work was supported in part by the National Science Foundation under Grant CDA-9703088, by Dupont, and through collaborative participation in the Advanced Telecommunications and Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Naohisa Ohta.

The authors are with the Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716 USA (e-mail: xie@udel.edu; arce@udel.edu).

Publisher Item Identifier S 1057-7149(01)09362-9.

Although traditional digital signatures can be used with image data where the raw or compressed image files are treated as data files, it will be apparent shortly that traditional digital signatures are neither efficient nor adequate in many lossy communication environments. Consider the case of layered image coding, as an example, where "low-priority" image bits can be dropped during transmission due to network congestion, introducing small or negligible distortion in the image reconstruction at the receiver. In the likely event of "low-priority" bit loss during transmission, a conventional digital signature, would fail the authentication protocol in this example since the received image data and the signed data are not identical. This drawback is even more serious in broadcast and internet multicast applications where lossy channels are the norm.

In this paper, we develop a new approach to create digital signatures for imaging that adapts well to applications in multimedia communications. Rather than attaching the signature's bit-string as a header to the image file, we invisibly etch the digital signature into the image data using watermarking methods. In general, the watermarked signature is still public key encrypted for added security if needed. Embedded in the image data, the encrypted signature cannot be removed by file conversions or simple image manipulations. Furthermore, the digital signature is not "fragile" in that it tolerates small or negligible distortions caused by compression or other standard manipulations performed in multimedia communications. It does not, however, tolerate other malicious tampering that modifies the content of the image by the addition or removal of objects.

II. Etching Signatures in the Wavelet Domain

A. Desirable Etching Characteristics

We start with the underlying assumption that compression is inevitably used during transmission. Assuming the contrary is not practical in most applications. Moreover, if compression is assumed, no compression is included as a special case in the assumed framework. We also assume that the communication is lossy where bit errors can occur due to noise or as a consequence of congestion in multiplexed networks. Thus, it is possible that the transmitted and received image data are not identical.

These assumptions indicate that the etching of digital signatures must be robust to lossy compression and that the authentication mechanisms must not be "fragile," tolerating minor distortions on the data. The former requirement suggests that the watermarking and compression algorithms used for secure transmission should be coupled in some way in order to attain higher efficiency. The latter requirement suggests that the signature must record "strong" rather than "weak" image features. The use of conventional message digests, such as MD5, are not appropriate in this case as they are "hard" one-way hashing functions disallowing authentication even if a single bit of the data is modified. Note we consider image compression as an acceptable image transformation in the communication channel. Other manipulations such as geometrical attacks by image rotation or shifting, and image enhancement by histogram equalization and sharping, etc., are taken as image tampering because such manipulations are not normal in image transmission. In this paper, we mainly study the impact of image compression on the watermarks.

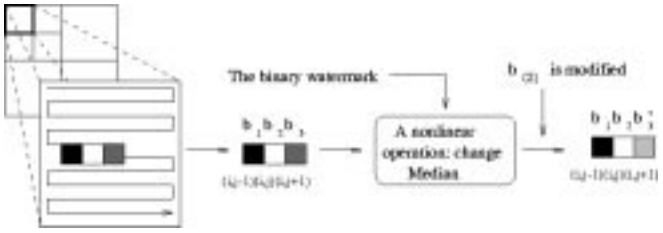

Fig. 3. Watermark engraving structure. The median of a nonoverlapping running window $b_{(2)}$ is modified according to the watermark bit x. Figure depicts the case when $b_3 = b_{(2)}$.

The watermarking of digital signatures must also be *blind* where the authentication process at the receiver end can be carried out without any knowledge of the original image being transmitted [4]. In order to implement a blind decoder, the watermarking process etching the digital signature must introduce *memory* by relating several image coefficients with the etching of a single watermark bit rather than encoding the watermark bit into only one coefficient. This concept is analogous to sequence modulation in communication channels [5]. Finally, the watermarking algorithm for etching digital signatures should be invisible to the naked eye.

B. Watermarking a Wavelet Transformed Image

As suggested above, higher efficiency can be attained if the watermarking and compression algorithms are jointly considered. In this paper, we focus on compression algorithms which are based on wavelet decompositions. The extension of the algorithms proposed here to other compression approaches, such as JPEG, is straight-forward.

First, a "soft" message digest of the image is generated which only captures the "strong" image features [6]. More on the generation of "soft" message digest will be said later on. It suffices to say at this point that traditional cryptographic message digests are not adequate. Thus given the bit-sequence of a "soft" message digest, the goal is to etch it into the wavelet decomposed image representation.

At first, for ease of explanation, we present the watermarking algorithm in the wavelet domain without compression. Once the fundamental concepts of the watermarking are presented, the watermarking algorithm is then integrated within a *zero-tree* type wavelet compression algorithm [7], namely the SPIHT codec [8]. To assure robustness, the bit sequence of the message digest will be etched into the low-frequency band of the wavelet image representation. The etching algorithm is illustrated in Fig. 3 and is described next.

We slide a nonoverlapping running window through the entire low frequency band of the wavelet decomposed image. Assume a 3×1 window is used although other window shapes can be used as well. Elements within the window are denoted as b_1, b_2, b_3, which are the coefficients' value at locations with coordinates $(i-1, j)$, (i, j), $(i+1, j)$. Given the coefficients b_1, b_2, b_3, we denote the corresponding rank-ordered coefficients as $b_{(1)} \leq b_{(2)} \leq b_{(3)}$. We then perform a nonlinear transformation algorithm, changing the median of these coefficients while

keeping the remaining coefficients the same. A detailed description of this operation will be presented at the end of this section. Denote the modified median by $b'_{(2)}$, which is obtained by the transformation,

$$b'_{(2)} = f\left(\alpha, b_{(1)}, b_{(3)}, x\right) \tag{1}$$

where x is the watermark bit sample to be etched in the location of the window and α is a user defined tuning parameter.

At the receiver, watermark extraction is an inverted etching process. The decoder needs only to know the value of α, the watermark length, and the necessary key if the watermarks are encrypted. A rectangular window of the same size as the one used in watermark engraving is applied to the received image. A sequence with elements: $B_{(1)}$, $B_{(2)}$ and $B_{(3)}$ is obtained as the window is shifted. We can get two possible values of $B'_{(2)}$

$$B'_{(2)} = \begin{cases} f(\alpha, B_{(1)}, B_{(3)}, 0) & \text{if } x = 0 \\ f(\alpha, B_{(1)}, B_{(3)}, 1) & \text{if } x = 1 \end{cases} \tag{2}$$

where

x possible value of watermark sample;
$B_{(1)} = b_{(1)}$;
$B_{(3)} = b_{(3)}$.

We compare the distance between $B_{(2)}$ and the two possible values of $B'_{(2)}$ in (2). x is then selected as the one which makes $B'_{(2)}$ closest to $B_{(2)}$. Thus, the watermarked bit associated with the window at each location is extracted as

$$x = \arg \min_{x \in (0,\,1)} |B_{(2)} - f(\alpha, b_{(1)}, b_{(3)}, x)|. \tag{3}$$

Shifting the decoding window throughout the entire watermarked image, we obtain the entire embedded watermark sequence.

C. Rank-Order Based Transformation

Here, we introduce the nonlinear transformation in (1) which is employed in the signature engraving scheme. A rank-order manipulation is motivated due to the concern of edge preservation [9], [10]. Edge properties determine the local coefficients' rank order, therefore, the rank-order must be preserved as well. Basically, the transformation changes the median of a local area to a value set by its neighbors. Given the coefficients b_1, b_2, b_3 and the corresponding order statistics $b_{(1)}$, $b_{(2)}$, $b_{(3)}$, the scaled midpoint is first defined as the spacing parameter

$$S_\alpha = \alpha \frac{|b_{(1)}| + |b_{(3)}|}{2} \tag{4}$$

where α is a tuning parameter with its default value of 0.05. S_α is adaptive to the overall local coefficients' magnitude so that the strength of the watermark is varied according to the local characteristics of the coefficients in the observation window. Next, the range of the coefficients $(b_{(1)}, b_{(3)})$ is partitioned into M intervals, each interval of length S_α. Note that if $S_\alpha > b_{(1)} - b_{(3)}$, we consider the local area to be too smooth to contain a watermark and the set of coefficients are skipped. Thus, a watermarked local area must satisfy

$$S_\alpha \le b_{(1)} - b_{(3)}. \tag{5}$$

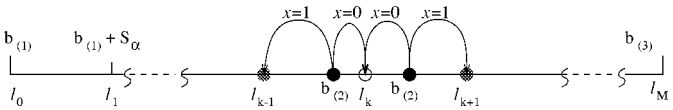

Fig. 4. Two possible median samples, $b_{(2)}$, being adjusted to $b'_{(2)}$ according to the watermark. k is assumed odd.

The boundary of the partitions are denoted as $\ell_0, \ell_1, \ldots, \ell_M$, with $\ell_0 = b_{(1)}$, $\ell_1 = b_{(1)} + S_\alpha, \ldots, \ell_M = b_{(3)}$, with M being the smallest integer for which $MS_\alpha > b_{(3)} - b_{(1)}$. Define the region in the interval $[\ell_{k-1}, \ell_k)$ as R_k, then the median $b_{(2)}$ is transformed into $b'_{(2)}$ as

$$b'_{(2)} = \begin{cases} \ell_k & \text{case } A \\ \ell_{k-1} & \text{case } B \end{cases}$$

where

$$\text{case } A \triangleq \{k \text{ is odd and } x = 0, \text{ or } k \text{ is even and } x = 1\}$$

$$\text{case } B \triangleq \{k \text{ is even and } x = 0, \text{ or } k \text{ is odd and } x = 1\}$$

where x is the bit of the watermark being inserted in the location of the window. For example, if the bit of the watermark to be etched at a particular window location is 0, and if $b_{(2)}$ lies in R_1, then $b'_{(2)} = b_{(1)}$. If the watermark bit is 1, and $b_{(2)} \in R_1$, then $b'_{(2)} = b_{(1)} + S_\alpha$. In Fig. 4, for instance, k is assumed odd. By adjusting α, we are able to tune the strength of the watermarks. Since $0 \le |b'_{(2)} - b_{(2)}| \le S_\alpha$ and S_α is proportional with α, when α increases, the range of the change on $b_{(2)}$ is larger and too much change on $b_{(2)}$ will generate artifacts on the image. On the contrary, when α decreases, image quality will be more likely preserved. However, the watermark is more vulnerable to noises because it is weaker. Therefore, there is a tradeoff for α between the watermark's robustness and transparency.

The proposed scheme is able to engrave the signature data into a large number of images while preserving their image quality. Figs. 5 and 6, for instance, depict an original 288×500 image and its corresponding watermarked image with $\alpha = 0.05$. The test image is chosen because it contains distinct objects that can be effectively used to illustrate image tampering. The peak signal to noise ratio (PSNR) is 38. Fig. 7 depicts a watermarked image where the tuning parameter α is made too large and the watermarked image quality is degraded. Fig. 8 visually shows the artifacts introduced as α is increased. The lower left section of the image represents the error between the original in Fig. 5 and the watermarked image with $\alpha = 0.05$ in Fig. 7. The top right section is the error between the original and the watermarked image with $\alpha = 0.5$ in Fig. 7. The blocking artifacts on the latter case are clearly seen.

III. Watermarking Within a Wavelet Compression Framework

The need to implement the watermark algorithm within a compression algorithm arises when compression is used during transmission. The signature engraving must be embedded within the particular compression scheme adopted by the protocol. Here, we adopt the SPIHT compression algorithm [8].

Fig. 5. Original image.

Fig. 6. Watermarked image with $\alpha = 0.05$ and two-level DWT decomposition.

Fig. 7. Watermarked image with $\alpha = 0.5$ and two-level decomposition.

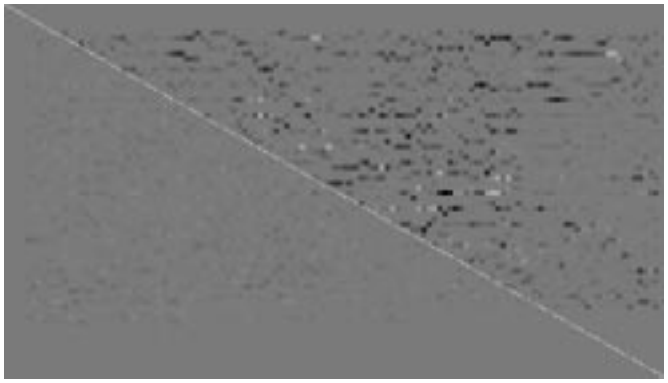

Fig. 8. Upper right part: the difference between the original image and the image in Fig. 7. Lower left part: the difference between the original image and the image in Fig. 6.

A. SPIHT Algorithm

In essence, the SPIHT algorithm aims to keep large coefficients (quantized) and throw away small coefficients in the DWT transform domain. It uses a tree-structured indexing in order to exploit the pyramid decomposition. In particular, the concept of parent–child dependencies is used. A coefficient at a coarse scale is referred to as a parent and all coefficients at the next finer scales at the same spatial location and of similar orientation are referred to as the children. Furthermore, the coefficients at all finer scales on the same spatial location and of similar orientation are called the descendents of that coefficient. The brothers of a coefficient are those coefficients having the same parent. We denote the coefficient at coordinate (i, j) as $c_{i,j}$ and define

$$D_{i,j} \triangleq \text{The descendents of } c_{i,j},$$
$$B_{i,j} \triangleq \text{The brothers of } c_{i,j}.$$

Fig. 9 depicts the hierarchical relationship of the coefficients obtained by a two-level DWT decomposition of image I. The hierarchical tree structure of the coefficients is shown in Fig. 10. In Fig. 9, A_0^0 is a coefficient in the high frequency component (HH0) at the highest level of a two-level decomposition. A_0^1, A_1^1, A_2^1, A_3^1 are the children of A_0^0 and they are brothers to each other. A dark square, which covers a 2×2 area, is used in Fig. 9 to show the brother relationship of those coefficients under the window. A_0^2, A_1^2, …, A_{15}^2 together with A_0^1, …, A_3^1 are the descendents of A_0. For the convenience of implementation, we regard A_0^0 and its three adjacent coefficients A_1^0, A_2^0 and A_3^0 as virtual brothers at level 0 and they are enclosed in the dark dotted line square.

Next, we examine those features in the SPIHT algorithm relevant to our application. The algorithm implements a progressive transmission method by using the binary representation of the magnitude-ordered coefficients. The coefficients are ordered by magnitude and the most significant bits are transmitted first. Furthermore, subset partitioning using the spatial tree structure is employed to expedite the search for the significant coefficients. In Fig. 10, node A_0^0 and all of its descendents are an example of a partitioned subset. A coefficient $c_{i,j}$ is considered significant if it satisfies

$$|c_{i,j}| \geq 2^n$$

where n is the bit level. $n = n_0, n_0 - 1, \ldots, n_m$, where n_0 is the highest bit level, $n_0 = \lfloor \log_2 \max_{(i,j) \in I}(|c_{i,j}|) \rfloor$ and n_m is the last bit level at the end of the coding when the requested compression ratio is met. For notational simplicity, we let $n_m = m$. The significance test decides whether the coefficient is to be coded or discarded at the current bit level.

In the coding algorithm, the brother set $B_{i,j}$ is treated as an indivisible group, which is made up by 2×2 adjacent pixels. Either all members in $B_{i,j}$ are to be tested at the nth bit level, or the group is discarded. Therefore, at the end of transmission, each member in the group will be represented with the same accuracy. This fact is important in our algorithm since it allows us to conveniently track such a group while at the same time

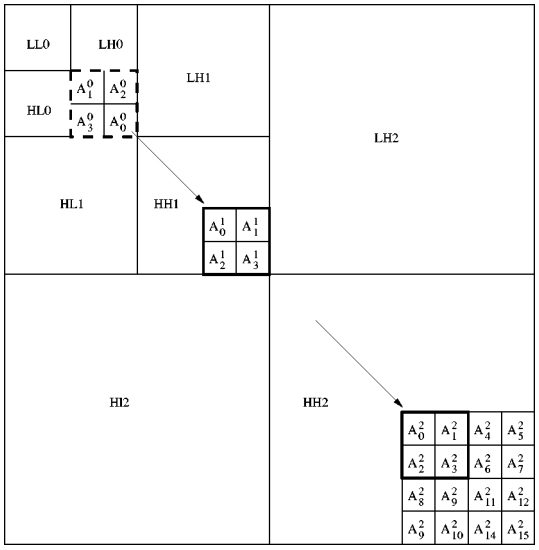

Fig. 9. Two-level DWT decomposition.

the group is made up of coefficients that represent the information of a neighboring area. In our algorithm within the compression framework, rather than sliding a nonoverlapping window throughout the decomposed image, we slide the non overlapping 2×2 window at locations where the $B_{i,j}$ group yields a significant coefficient.

B. Signature Engraving

At the encoder, the SPIHT algorithm is executed first. The output of the SPIHT algorithm generates a hierarchical list of the significant coefficients. To insert the watermark, the 2×2 nonoverlapping window is scanned through all the brother sets where at least three coefficients survive quantization, i.e., those coefficients are significant at the last bit level, the mth bit level. Another test is next performed on each set $B_{i,j}$ to decide whether or not a watermark bit can be inserted. Since the window is 2×2 at each location and contains four coefficients, c_1, c_2, c_3, c_4, we select three of these coefficients such that these correspond to the ones with the largest absolute value. These three coefficients are then sorted by their actual value yielding the three order-statistic $b_{(1)}, b_{(2)}, b_{(3)}$ in ascending order. To apply the watermarking, we define

$$W(B_{i,j}) = \begin{cases} 1, & \text{if } b_{(3)} - b_{(1)} \geq 2^m \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $W(B_{i,j})$ indicates if watermark engraving in the local window $B_{i,j}$ is admissible. m is the last bit level to determine the significance of the coefficient. For those $B_{i,j}$ that satisfy $W(B_{i,j}) = 1$, we proceed with the watermarking process. We calculate the scaling factor S_α.[1] At the quantization stage, S_α is quantized into the multiples of the quantization constant $q = 2^m$, i.e., $S_\alpha \leftarrow \lfloor S_\alpha/q \rfloor \times q$ where the function $\lfloor \cdot \rfloor$ is the lowest integer truncating "floor" function. Since $S_\alpha = 0$ is meaningless, the minimum of S_α is $q = 2^m$, i.e., $S_\alpha \geq 2^m$.

[1]Recall $S_\alpha = \alpha((|b_{(1)}| + |b_{(3)}|)/2)$.

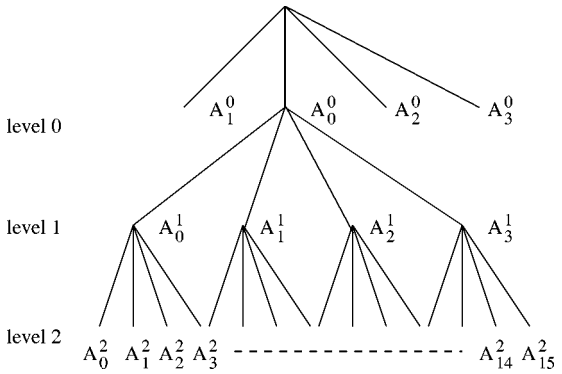

Fig. 10. Hierarchical tree structure used by SPIHT coding.

Thus, the lower bound of S_α is determined. When (5) is combined, we get

$$2^m \leq S_\alpha \leq b_{(1)} - b_{(3)}. \tag{7}$$

Thus, if a local area is considered significant for watermarking, the image data in the area must contain a margin larger than the compression quantization threshold. Note that the watermarking locations are no longer constrained to be in the top level of the DWT pyramid. The locations are distributed in all bands and are determined by the local characteristic of the underlying image.

C. Multibit Engraving

In general, the engraved signature is represented by binary data such that one bit at a time is engraved where admissible. The watermark capacity, however, can be increased if we engrave as many bits as we can in each window location. This is possible if the range $b_{(3)} - b_{(2)}$ in (6) is significantly larger than the threshold 2^m.

Recall that information is hidden in a local area where the coefficient range overcomes quantization constant $q = 2^m$, which is the minimum quantization threshold used when coding ends. Thus, when $b_{(3)} - b_{(1)} \geq q$, we perform the watermarking method by partitioning the range $(b_{(3)}, b_{(1)})$ into intervals of length S_α and by modifying $b_{(2)}$ according to the watermark sample. However, when S_α is several times larger than q, we can refine the engraving process and split S_α into smaller segments. Formerly, we inserted two watermark values only: 0 and 1. Now if the distance is twice as large as that of the quantization constant, a 3-ary symbol (0, 1, or 2) can be inserted. This approach is called "multiple bit engraving." Fig. 11 depicts the concept behind multibit engraving where the size of partition used in binary engraving is $3q$. Two bits of information can be hidden in this case. As we can see, multiple bit engraving is similar in principle to the binary approach.

D. Edge Information Message Digest

As stated previously, a message digest in the traditional sense does not fit the needs of a "soft" message digest for the image. For example, the MD5 algorithm [3] hardly tolerates any distortions in that it produces a 128-bit message digest using a strict one-way hashing algorithm. Hash algorithms for images have been studied in [11]–[14]. In our approach, the edges of the

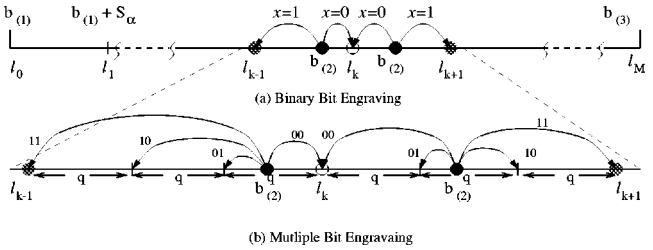

Fig. 11. Comparison of binary and multibit engraving methods: (a) binary engraving method and (b) multibit engraving method.

image are used as the content's fingerprints in order to make the message digest robust to minor artifacts introduced during communication. Usually, the edge detector is applied at the rescaled image with a reduced size, or the lowest frequency component of wavelet decomposition, for the following reasons. First, the number of watermark bits we can embed in an image is not large enough to carry the edge map of the image with the original size. Secondly, the edge map of a lower resolution image will be less sensitive to minor artifacts introduced by compression and simple image manipulations.

The size of the edge map is image dependent, however, the decoder must know the length of the watermark sequence to extract the signature. One solution is to set up "rules" agreed by both sides: the sender and the receiver. For example, both sides can adopt a Sobel edge detection on the LL component of a three-level wavelet decomposition. As long as the encoder and decoder use the same edge detector, the edge-based message digest proves reliable. If a change of edge information is detected, the authentication algorithm decides that the image has been tampered. In fact, it is desirable that the message digest is of fixed length for the convenience of authentication. A two level message digest can be generated, where the MD5 algorithm is applied to the edge-based message digest. For added security, the message digest can be encrypted either by a secret key shared by both sides or using a public key encryption [2].

We note that edge maps are not strictly cryptographic hash functions as it is computationally feasible to find another image which provides the same edge map; however, it satisfies our needs to detect gross image changes. A comparison between the edge information carried by the original image (Fig. 5) and the corrupted image (Fig. 12) is shown in Fig. 13. The difference of their edge maps is shown in Fig. 13(c). Fig. 13(d) overlaps the edge difference in dark color and the received image. Furthermore, due to the multiresolution decomposition performed by the wavelet transform, if tampering has occurred, the edge-based message digest is able to roughly point out the place where the tampering took effect by checking the watermark bits embedded at the lowest level of DWT.

E. Algorithm Comparison

Here we compare our watermarking algorithm with another wavelet-based technique developed by Kundur and Hatzinakos [15]. The two algorithms were developed independently but both etch watermark in the transform domain by dividing the distance between the maximum and the minimum into equal length intervals, associating the boundaries between the intervals with a zero-valued or a one-valued bit alternatively,

Fig. 12. Tampered image with one building replaced.

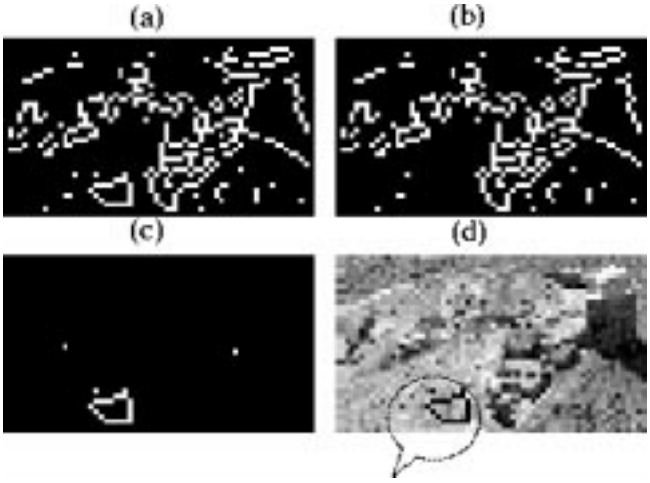

Fig. 13. (a) Edge of the original image; (b) the edge of the corrupted image; (c) the difference between (a) and (b); and (d) (c) as seen on the corrupted image. The questionable area is highlighted in the figure.

and adjusting the median to the nearest boundary associated with the same binary bit as the watermark.

The difference between the two etching processes lies in how the distance is divided and the choice of transform coefficients. In Kundur and Hatzinakos [15], the distance between the maximum and the minimum is divided into a fixed, user defined number of intervals, $2Q - 1$. Meanwhile, the watermarks are embedded in different resolutions of the *detail* images. In our algorithm, the number of intervals is adaptive to the magnitude of the maximum and the minimum and is tunable by α. We believe that the adaptive approach will lead to less artifacts and watermark capacity. For watermark etching in the uncoded transform domain, we choose to use the *approximate* image, the lowest frequency component (LL0) for its robustness. In fact, we tested the use of fixed number of intervals in our algorithm and found objectionable visual artifacts unless Q is sufficiently large. In Kundur and Hatzinakos [15], the image "Barb" was watermarked with $Q = 4$ and no visible artifacts was reported. However, artifacts are found if $Q = 4$ is used to etch the watermarks in the LL0 band. Therefore, we believe the image quality is better preserved if the number of intervals is adaptive responding to the values of the local coefficients.

We believe that the watermarking and compression algorithms used for secure transmission should be coupled in order

to attain higher efficiency. In Kundur and Hatzinakos [15], the watermarks are carried by the details of image (HL, LH, HH), which are fragile to wavelet compression. Through Sections II and III, we contributed a watermarking method embedded within the SPIHT compression scheme. Further contribution to the information capacity of data hiding is presented in the coming section.

IV. Information Bit Rate

The bit rate of a watermark (measured in bits per pixel) indicates the amount of information that can be embedded within an image, which is particularly important for content based signatures since the length of the binary data representing the signature is often large. An overview of the capacity issue in data hiding in images is given in [16].

A. Information Capacity for Multibit Engraving Scheme

In this section, we show that the information capacity for multibit engraving scheme is bounded by the rate given by Shannon's channel capacity theorem, although a different interpretation of noise is used. According to Shannon's results [17], given the average transmitter power S, and assuming that the noise is white Gaussian noise of power N having a bandwidth W, by sufficiently complicated encoding systems it is possible to transmit binary digits at a rate

$$C \leq W \log_2 \left(1 + \frac{S}{N}\right) \tag{8}$$

where C is the channel bit rate.

The inequality sets up the upper bound on the possible rate for an error-free communication channel subject to Gaussian noise. The principle Shannon used to derive the upper limit is that two symbols modulated at the sender in a communication channel must have a noise margin in order to get the correct decision at the receiver. With the assumption of white Gaussian noise, Shannon proved that "the probability of a given perturbation depends on the *distance* from the original signal and not on the direction." Two signals are distinguishable if their noise region is nonoverlapping since overlapping result confusion at the receiver. Therefore, the capacity problem can be formulated by finding the maximum rate of signals which satisfy 1) the average power is S and it is band-limited to W and 2) the distance between two signal in their geometrical representation is larger than the noise power N.

In our signal and distortion model, the compressed image is approximated as a continuous, band-limited channel. We regard the compressed image I as the signal and the quantization truncation as the source of noise. Let us assume the image I, has power S and the quantization constant is q where $q = 2^m$. How many bits of information can we embed?

As discussed in Section III-B [see (6) and (7)], a local area to be watermarked is selected if it meets the following constraints: a) $d_{1,3} \geq q$; b) $q \leq S_\alpha \leq d_{1,3}$, where the range of $(b_{(3)}, b_{(1)})$, i.e., $b_{(3)} - b_{(1)}$, is denoted by $d_{1,3}$, and $q = 2^m$. During multibit signature engraving, we change the median in a window to a fixed value set by its neighbors and the watermark bit. More bits can be embedded in a selected local area with larger S_α, however, b) shows that S_α is upper bounded by $d_{1,3}$. In order to derive the capacity upper-bound, maximum use of the available margin is assumed, which implies that b) may be ignored at this moment and we examine a) only.

To arrive at the information capacity, we exploit the similarity between the proposed signature scheme and signal detection in a communication channel. a) discloses the lower bound of $d_{1,3}$. A watermark bit can only be inserted in a local window where the distance between the maximum and the minimum coefficients is large enough to overcome the quantization. The median in a window is changed by watermarking, and the window is placed in a significant local area where the image data in the local window contains a margin larger than the quantization constant q. In other words, we are exploiting the local variance of an image. Because of coefficient truncating, the quantization constant q becomes the unit for measurement. The local variance is evaluated using multiples of q. Therefore, quantization becomes a critical factor for information embedding. It is clear that whether or not a watermark signal can be engraved is determined by the range of $(b_{(3)}, b_{(1)})$, i.e., the distance of the two signals. The information bit rate we can achieve by our engraving scheme is equal to the maximum rate of signals with noise power $N = q^2$. The following evolves directly from Shannon's theorem.

Theorem 4.1: The information bit rate we can engrave in an image is bounded by

$$B \leq W \log_2 \frac{S+N}{N} \tag{9}$$

where

- B information bit rate;
- S image power;
- N $= q^2$;
- q quantization constant;
- W image's bandwidth.

An intuitive understanding of this result is important. When S is very small such that it goes to zero, so does B. This result corresponds to images with constant areas having small variance where little information can be etched. On the contrary, when S is larger, the capacity is larger. So images with larger variance can hide more information.

When b), the constraint on S_α is taken into account, a factor α is used to adjust the strength of the watermarks. The effect of α on the capacity of the watermarking algorithm is described in the Appendixes and is summarized in Section IV-C.

B. Information Capacity for Binary Engraving Scheme

Next, we consider the binary engraving case. In the previous section, it was shown that the bit rate is related to the quantization constant. Their relationship is studied as follows. In the coding algorithm, the higher bits of a coefficient are sent first and the coding ends whenever the desired compression ratio is met. It may happen that some of the coefficients have been quantized to 2^m, however, the coding may end before the remaining coefficients are quantized by 2^{m+1}. Hence, the coefficients sent may be truncated at two different quantization thresholds, q and

$2q$, because of the incomplete scan of the last bit level. Thus, different quantization thresholds may exist and the fraction of coefficients that are truncated by each threshold vary with different compression ratios. Here, a bit rate corresponding to each quantization constant is defined at some "ideal" compression ratio, with which the quantization threshold is "complete" in the sense that all coefficients are truncated by the same threshold and all coefficients which are no less than 2^m are sent.

A sequence $q(i)$ ($i = 0$ to k) is defined to describe the quantization constants. $q(i) = 2^{k_0 - i}$ where k_0 is an integer usually less than n_0. Since we are only interested in those $q(i)$ with which watermark engraving is possible, 2^{k_0} is the largest quantization constant when at least one watermark sample can be inserted. An example of $q(i)$ is: 64, 32, 16, 8, 4, 2, 1, and $k_0 = 6$. We use a sequence $c(i)$ ($i = 0$ to k) to denote those "ideal" compression ratios corresponding to each $q(i)$. The bit rate for the compression ratios other than $c(i)$ is bounded by the bit rates of the two $c(i)$s closest to it. We also define sequence $B(i)$ and $b(i)$ to be the bit rate with compression $c(i)$ using multibit engraving and binary engraving respectively.

We derive an approximate relation between the bit rate sequence of multibit and binary engraving. The approximate capacity equivalence is indicated by the symbol $\mathcal{O}(\cdot)$.

Theorem 4.2: The bit rate of binary engraving is related to the bit rate of multiple bit engraving by

$$\mathcal{O}(b(i+1)) = \mathcal{O}(B(i+1)) - \mathcal{O}(B(i)) \tag{10}$$

$$\mathcal{O}(B(i+1)) = \mathcal{O}(b(i+1)) + \mathcal{O}(b(i)). \tag{11}$$

The proof is attached in Appendix B. Note that the second relation in Theorem 4.2, i.e., (11), can also be obtained from (10) if we assume that $b(i) \approx B(i)$ at $i+1$, i.e., $B(i)$ and $b(i)$ is far less than $B(i+1)$. These equations can be used to predict $B(i)$ and $b(i)$. We can predict $B(i)$ if we know $b(i)$, or we can predict $b(i)$ if we know $B(i)$. The latter is particularly useful since we have derived an upper bound for the $B(i)$. This assumption plus the other assumptions used in the proof were verified by the experiments, especially in the case of large compression ratios.

C. Effect of α on the Capacity

In the Appendixes, the impact of α on bit rate by multibit engraving and binary engraving is studied. We find that the bit rate by multiple engraving arrives its maximum when $\alpha = \alpha_0$. α_0 is an approximate threshold of α with which the range of $(b_{(3)}, b_{(1)})$ is fully exploited for watermarking (see Appendix B-1 for the definition of α_0). At that point, the capacity upper-bound follows Theorem 4.1. In addition, a sequence $\alpha(i)$ ($i = 0$ to l) is defined to describe the tuning parameter α. $B_\alpha(i)$ and $b_\alpha(i)$ are defined to be the bit rate at a particular compression ratio using multibit engraving and binary engraving, respectively, and a set of equations similar to those in Theorem 4.2

$$\mathcal{O}(b_\alpha(i+1)) = \mathcal{O}(B_\alpha(i+1)) - \mathcal{O}(B_\alpha(i)) \tag{12}$$

$$\mathcal{O}(B_\alpha(i+1)) = \mathcal{O}(b_\alpha(i+1)) + \mathcal{O}(b_\alpha(i)) \tag{13}$$

is obtained.

D. Simulations

Here, we present experimental results illustrating the information capacity as a function of quantization. Our experimental results confirm a logarithm relation between the bit rate and the quantization level. The results are presented in Figs. 14–16 and Table I. Fig. 14 plots the watermark bit rate for an eight-bit 512×512 image versus $\log_2(1 + (S/q^2))$. S and α remain a constant in the plot. The horizontal axis is inversely related to the compression ratio. As q increases, the quantization increases, the compression ratio increases, but $\log_2(1 + (S/q^2))$ decreases. The bit rate results include: experimental results using multibit engraving, experimental result using binary engraving scheme and the upper bound. The signal power S is estimated by the image's variance that is measured in the spatial domain. Sequence $q(i) = 2^{6-i}$ ($i = 0, 1, \ldots, 6$). The upper bound is obtained by measuring the image's bandwidth in the FFT transformed domain to approximate the bandwidth W[18]. The bandwidth is obtained as the frequency where the power spectrum density (psd) decays 30 db from the DC level.

In Fig. 15, we plot the bit rate at various quantization levels vs α. The same image is used as in Fig. 14. It is clear that the maximum bit rate is attained for $\alpha = 2$ while $\alpha_0 = 1.79$.

In Table I, we depict the experimental results of the number of bits using multibit and binary engraving methods, where B stands for the experimental result and B' stands for the prediction by sequence b. Similarly, b is the experimental result and b' the predictions by sequence B. The estimation and the experimental data are very close especially when i is small where the quantization constant is large. To show the impact imposed by the floor function $\lfloor (f(\cdot)) \rfloor$, we also calculated the bit rate of multibit engraving not using the floor function and obtained the sequence B''. b'' are the predictions made using B''.

The final experiment performed on the information bit rate was the measurement of the watermark bit rate in bits per pixel and PSNR for images with different sizes. As Fig. 16 shows, the bit rates for the 512×512 image and the 256×256 image are comparable at the same PSNR, especially when the PSNR is low ($\alpha = 0.125$). Fig. 16 illustrates the relation between the SNR and the bit rate.

V. Experimental Results

Under the wavelet compression scheme, several experiments were run to evaluate the robustness of the signature and the image quality change that resulted from the signature engraving. Our experimental results show that the signature achieves our initial goal of robustness and transparency.

A. Authentication Under Image Compression

The impact of image compression on the watermarks is studied. We repeatedly compressed the image with the signature using different compression ratios and extracted the embedded information out each time. Let's denote C_0 as the compression ratio applied when the signature was engraved. Our experimental results show that the signature stayed in the image if the compression ratio used in later compression was lower than or equal to C_0. In the experiment, we compressed the watermarked image more than ten times and the signature

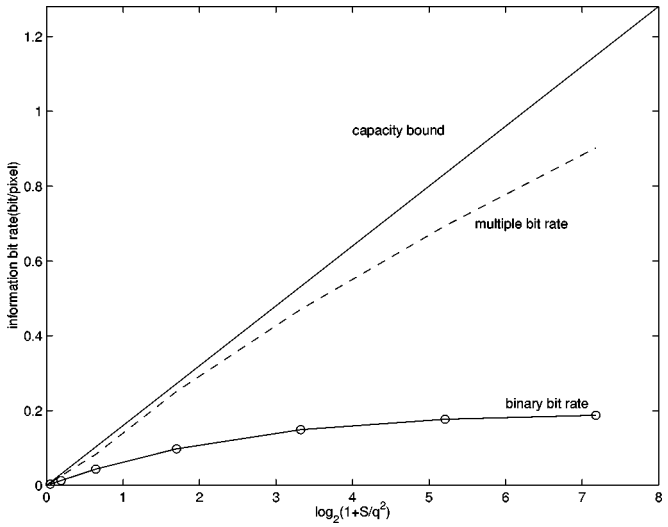

Fig. 14. Relation of bit rate and quantization using binary and multibit engraving at α_0.

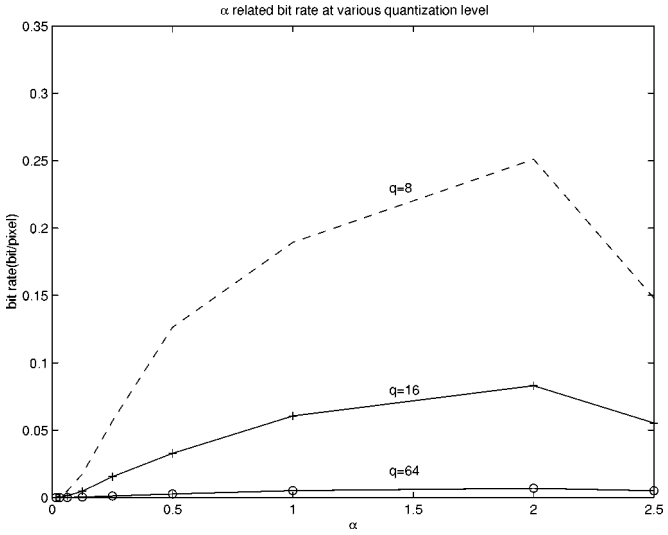

Fig. 15. α related bit rate at various quantization level.

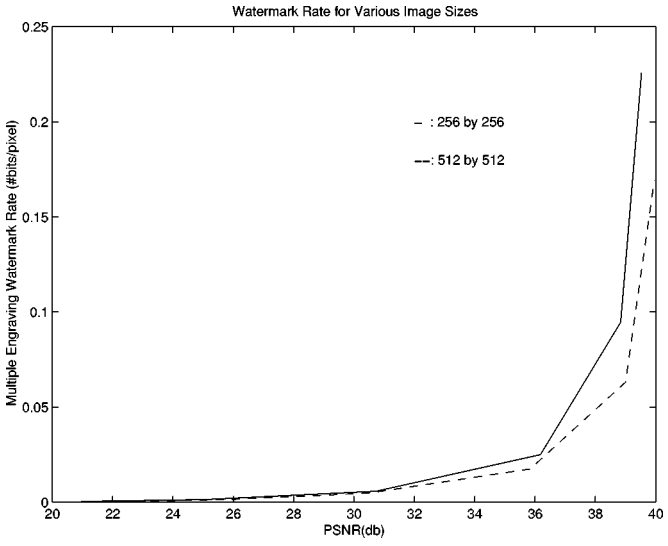

Fig. 16. Logarithm relation between the information rate and PSNR.

TABLE I
EXPERIMENTAL DATA AND THE PREDICTED CAPACITY USING MULTIBIT AND BINARY ENGRAVING AT $\alpha = 0.125$

i	B	b	B'	b'	B''	b''
1	84	74	94			
2	337	292	366	247	369	255
3	1234	1059	1351	903	1366	997
4	4577	3896	4955	3343	5034	3768
5	16533	14101	17997	12043	18304	13270
6	44297	35017	49118	27764	50397	32093
7	85300	55184	90201	41003	99452	49035

was not altered. These experiments show that the signature is more robust if engraved at high compression ratios. Since compression is a low pass filtering process, the result agrees with arguments by Cox [19] that robust watermarks must be put in the most significant components.

In all, the experimental results show that the signature embedded at higher compression ratios is more robust than signatures embedded at lower compression ratios. However, it is expected that fewer bits can be engraved when the compression ratio is higher. This is understood since higher compression smoothes the image and reduces the possibilities of data hiding. Therefore, there is a trade-off between the information bit rate and the robustness of the signature.

B. Visual Impact of the Signature on an Image

"Transparency" refers to the visual impact of the signature on the image. We use the MSE and PSNR to measure the visual impact. A comparison of the MSE and PSNR of the original image versus the compressed image with and without the signature was computed and it is observed that there is little difference brought by signature engraving. Particularly, the difference gets smaller when the compression ratio is higher.

VI. CONCLUSION

Digital signatures for secure transmission and distribution of digitized images are becoming important with the rapid development of information technology. In this paper, we propose a content based digital image signature system for image authentication using digital watermarking techniques. We introduce a blind watermarking digital signature for the purpose of authentication. Thus anyone, with access to the embedded public cryptographic keys, wishing to authenticate watermarked images is able to do it, provided the watermarking retrieval mechanism is available. An edge based message digest is developed which is capable of detecting image tampering. The signature survives in a lossy environment provided the signature is robustly embedded. The information capacity was studied. For multibit engraving, the capacity is shown to be bounded by the rate given by Shannon's channel capacity theorem where the noise is the effective quantization. An approximate relation between the bit rate sequence of multibit and binary engraving was derived. Experimental results illustrate the signature's robustness and its perceptual visual impact on the image, and confirm the theoretical results in the information bit rate.

APPENDIX A
PROOF OF THEOREM 4.2

Proof: At $c(i)$, we index all the significant areas by m, $m = 1, \ldots, b(i)$ and name them A_m. We also define the sequence $d_m = \alpha((|b_{(3)}| + |b_{(1)}|)/2)$ [$m = 1$ to $b(i)$] for each A_m. Let $G_i = \{A_1, A_2, \ldots, A_{b(i)}\}$. Note that d_m must be larger or equal than $q(i)$. The number of bits we can engrave by the multibit engraving method is

$$B(i) = \sum_{m=1}^{b(i)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i)} \right\rfloor\right) \tag{14}$$

where the function $\lfloor \cdot \rfloor$ is the lowest integer truncating "floor" function. Similarly at $i+1$

$$B(i+1) = \sum_{m=1}^{b(i+1)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor\right). \tag{15}$$

And $G_{i+1} = \{A_1, \ldots, A_{b(i)}, A_{b(i)+1} \ldots, A_{b(i+1)}\}$ where $A_{b(i)+1}, \ldots, A_{b(i+1)}$ are the new admitted areas. It can be seen that $G_i \subseteq G_{i+1}$. Define $G_{i+1|i}$ such that $G_{i+1} = G_i \cup G_{i+1|i}$ and $G_i \cap G_{i+1|i} = \emptyset$ so $G_{i+1|i} = \{A_{b(i)+1}, \ldots, A_{b(i+1)}\}$. It follows that $B(i+1)$ is equal to

$$\sum_{m=1}^{b(i)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor\right) + \sum_{m=b(i)+1}^{b(i+1)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor\right). \tag{16}$$

Since $q(i) = 2q(i+1)$, $B(i+1) - B(i)$ is equal to

$$\underbrace{\sum_{m=1}^{b(i)} \log_2 \frac{1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor}{1 + \left\lfloor \frac{d_m}{2q(i+1)} \right\rfloor}}_{I_1} + \underbrace{\sum_{m=b(i)+1}^{b(i+1)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor\right)}_{I_2}. \tag{17}$$

Let I_1 be the first term on the right side of (17). Denoting $\kappa = d_m/2q(i+1)$ and $\alpha(\kappa) = \kappa/(1+\kappa)$, we have

$$I_1 - b(i) \approx \sum_{m=1}^{b(i)} (\log_2(1 + \alpha(\kappa)) - 1) \tag{18}$$

where the approximation $x \approx \lfloor x \rfloor$ was made to remove the "lowest integer" operation. Using the Taylor expansion on $\ln(1+x) = x - (x^2/2) + (x^3/3) + \cdots$, we obtain

$$I_1 - b(i) \approx \sum_{m=1}^{b(i)} (1.44\alpha(\kappa) - 1 + o(\alpha(\kappa))) \tag{19}$$

where $o(\alpha)$ are the higher order terms compared with $\alpha(\kappa)$. Since $d_m \geq 2q(i+1)$ for those $A_m \in G_i$, $\kappa \geq 1$ and $0.5 \leq \alpha(\kappa) < 1$, then discarding the higher order terms, we have

$$-0.28b(i) \leq I_1 - b(i) < 0.44b(i). \tag{20}$$

The second term in (17), I_2, is

$$I_2 = b(i+1) - b(i) \tag{21}$$

since $q(i+1) \leq d_m \leq 2q(i+1)$ for those $A_m \in G_{i+1|i}$, $\lfloor d_m/q(i+1) \rfloor = 1$. Furthermore, we assume that $b(i) \ll b(i+1)$. Using (20), we conclude that $I_1 - b(i)$ is small compared to $b(i+1)$. If we omit the small term and combine (17) and (21), $B(i+1) - B(i)$ is simplified to

$$\mathcal{O}(B(i+1)) - \mathcal{O}(B(i)) = \mathcal{O}(b(i+1)) \tag{22}$$

proving the first result of Theorem 4.2. The proof for the second result in Theorem 4.2 follows a similar development and is not presented here.

APPENDIX B
EFFECT OF α ON THE CAPACITY

A. Impact of α by Multibit Engraving

In our scheme, the range of $(b_{(3)}, b_{(1)})$, denoted by $d_{1,3}$, is split into intervals with the size of S_α

$$S_\alpha = \alpha \frac{|b_{(1)}| + |b_{(3)}|}{2}. \tag{23}$$

The maximum bit rate corresponds to the multibit engraving method, by which S_α is fully exploited. However, S_α is bounded by

$$q \leq S_\alpha \leq d_{1,3}. \tag{24}$$

$S_\alpha \leq d_{1,3}$ implies that information will not be embedded in a very smooth area and $S_\alpha \geq q$ implies no watermarks in a weak area where the multiplication of the tuning parameter (α) and the overall magnitude of the local area $(|b_{(1)}| + |b_{(3)}|)/2$ is less than the quantization constant q.

For those locations where S_α satisfies (24), when α increases, S_α increases (S_α is proportional with α), the bit rate of the watermarks increases since more levels are obtained when a larger S_α is split by q at the refining stage (see Fig. 11). Also, by (24), there is an upper-bound for S_α, so there exists α_0 such that $S_\alpha = d_{1,3}$ with which the maximum number of bits can be engraved in the local area. However, $S_{\alpha_0} = d_{1,3}$ in one window does not mean S_α will be equal to $d_{1,3}$ using the same α_0 in other windows. We estimate an α_0 for the entire image at the point that the following approximation holds in average

$$S_{\alpha_0} \approx d_{1,3} \tag{25}$$

i.e., α_0 is estimated using the average of $d_{1,3}$ over the average of the overall magnitude $(|b_{(1)}| + |b_{(3)}|)/2$ in all local areas. One example value of α_0 is $\alpha_0 = 1.76$ (an 8 bit 512×512 image is tested). When $\alpha = \alpha_0$, the capacity problem goes back to what we have discussed in Section IV because $S_{\alpha_0} \approx d_{1,3}$ and splitting S_α by q is equivalent to splitting $d_{1,3}$ by q. Therefore, the capacity upper-bound follows Theorem 4.1.

For those locations where S_α is beyond the bound provided by (24), there is no watermark. The smaller α, the smaller S_α is and there are more locations that $S_\alpha < q$, the bit rate will decrease. When $\alpha > \alpha_0$, $S_\alpha \geq S_{\alpha_0}$ while $S_{\alpha_0} \approx d_{1,3}$. No watermarks will be inserted at places where $S_\alpha \geq d_{1,3}$, the bit rate will also decrease.

As a result, the bit rate arrives its maximum when $\alpha = \alpha_0$. At that point, the capacity upper-bound follows the logarithm function in (9) with parameters W, P and variable q.

B. α and the Bit Rate Of Binary Engraving

A sequence $\alpha(i)$ ($i = 0$ to l) is defined to describe the tuning parameter α. Define $B_\alpha(i)$ and $b_\alpha(i)$ to be the bit rate at a particular compression ratio using multibit engraving and binary engraving respectively. In this paper, a sequence $\alpha(i)$: 0.0156, 0.0312, 0.0625, 0.125, 0.25, 0.5, 1, 2, $0 \le i \le 7$ is examined. $\alpha(0) = 0.0156$ and $\alpha(i+1) = 2\alpha(i)$.

Similar to the proof of Theorem 4.2, we index all the significant areas at $\alpha(i)$ by m, $m = 1, \ldots, b_\alpha(i)$ and name them A_m. Thus, the number of bits we can engrave by the multibit engraving method is

$$B_\alpha(i) = \sum_{m=1}^{b_\alpha(i)} \log_2 \left(1 + \left\lfloor \frac{S_{\alpha(i)}}{q} \right\rfloor\right) \tag{26}$$

and,

$$B_\alpha(i+1) = \sum_{m=1}^{b_\alpha(i+1)} \log_2 \left(1 + \left\lfloor \frac{S_{\alpha(i+1)}}{q} \right\rfloor\right) \tag{27}$$

where $S_{\alpha(i+1)} = 2S_{\alpha(i+1)}$. Comparing the above with the two equations that start the proof of Theorem 4.2

$$B(i) = \sum_{m=1}^{b(i)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i)} \right\rfloor\right) \tag{28}$$

$$B(i+1) = \sum_{m=1}^{b(i+1)} \log_2 \left(1 + \left\lfloor \frac{d_m}{q(i+1)} \right\rfloor\right) \tag{29}$$

where $q(i+1) = 2q(i)$, we find that the settings of the two problems are almost the same. As it turns out, a similar set of equations

$$\mathcal{O}(b_\alpha(i+1)) = \mathcal{O}(B_\alpha(i+1)) - \mathcal{O}(B_\alpha(i)) \tag{30}$$

$$\mathcal{O}(B_\alpha(i+1)) = \mathcal{O}(b_\alpha(i+1)) + \mathcal{O}(b_\alpha(i)) \tag{31}$$

can be obtained if we make similar approximations and assume $b_\alpha(i) \ll b_\alpha(i+1)$. The proof of it is analogous to the proof of Theorem 4.2.

REFERENCES

[1] A. Furche and G. Wrightson, *Computer Money*. Heideberg:dpunkt: Verlag fur Digitale Technologie, 1996.

[2] B. Schneider, *Applied Cryptography: Protocols, Algorithms, and Source Code*. New York: Wiley, 1996.

[3] R. L. Rivest, "RFC1321: The md5 message-digest algorithm," Internet Activities Board, Apr. 1992.

[4] S. Craver, N. Memon, B. Yeo, and M. Yeung, "Can invisible watermarks resolve rightful ownerships?," IBM Res. Tech. Rep. RC 20509, July 1996.

[5] R. E. Blahut, *Digital Transmission of Information*. Reading, MA: Addison-Wesley, 1990.

[6] M. Scheneider and S. Chang, "A robust content based digital signature for image authentications," in *Proc. 1996 IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, Atlanta, GA, May 1996.

[7] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

[8] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, Mar. 1996.

[9] L. Xie and G. R. Arce, "A blind content based digital image signature," in *Proc. 2nd Annu. Fedlab Symp. Advanced Telecommunications/Information Distribution*, vol. 1, College Park, MD, Feb. 1998.

[10] ——, "A blind wavelet based digital signature for image authentication," in *Proc. Eur. Signal Image Processing Organization (EUSIPO) Conf.*, Sept. 1998.

[11] J. Fridrich, "Robust bit extraction from images," in *Proc. 1999 IEEE Int. Conf. Multimedia Computing Systems*, vol. 2, Florence, Italy, June 1999.

[12] ——, "Visual hash for oblivious watermarking," *Proc. SPIE*, vol. 3971, Jan. 2000.

[13] G. R. Arce, L. Xie, and R. F. Gravemen, "Approximate image authentication codes," in *Proc. 4th Annu. Fedlab Symp. Advanced Telecommunications/Information Distribution*, vol. 1, College Park, MD, Mar. 2000.

[14] L. Xie, G. R. Gonzalo, A. Lewis, and B. Basch, "Methods for soft image/video authentication," in *Proc. 4th Annu. Fedlab Symp. Advanced Telecommunications/Information Distribution*, vol. 1, College Park, MD, Mar. 2000.

[15] D. Kundur and D. Hatzinakos, "Digital watermarking using multiresolution wavelet decomposition," in *Proc. 1998 IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, Seattle, WA, May 1998.

[16] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding—A survey," *Proc. IEEE*, vol. 87, July 1999.

[17] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vol. 37, Jan. 1949.

[18] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[19] I. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," in *Proc. 1996 IEEE Int. Conf. Image Processing*, vol. 3, Lausanne, Switzerland, Sept. 1996.

Liehua Xie was born in Hunan, China. She received the B.S. degree in electrical engineering from Beijing University, Beijing, China, in 1995, and the M.S. and Ph.D. degree in electrical engineering from the University of Delaware, Newark, in 1998 and 2000, respectively.

From September 1996 to June 2000, she was a Research Assistant in the Department of Electrical and Computer Engineering, University of Delaware. She has consulted with industry in the areas of image and video processing. Her research interests include image and video processing, image authentication, and secure multimedia communication.

Gonzalo R. Arce (M'82–SM'93–F'00) was born in La Paz, Bolivia. He received the B.S.E.E. degree with the highest honors from the University of Arkansas, Fayetteville, in 1979, and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1980 and 1982, respectively.

Since 1982, he has been with the Department of Electrical and Computer Engineering at the University of Delaware, Newark, where he is currently Professor and Chair, and a Fellow in the Center for Advanced Studies. He has consulted for several industrial organizations in the general areas of signal and image processing and digital communications. His research interests include robust signal processing and its applications, communication theory, image processing, and electronic imaging. He holds two U.S. patents. He was Guest Editor for the Optical Society of America's *Optics Express*. He is a Senior Editor of EURASIP's *Applied Signal Processing Journal*.

Dr. Arce is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and Guest Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a member of the Digital Signal Processing Technical Committee of the Circuits and Systems Society and the Board of Nonlinear Signal and Image Processing.