Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 56904, Pages 1–15
DOI 10.1155/ASP/2006/56904

A Secure Watermarking Scheme for Buyer-Seller Identification and Copyright Protection

Fawad Ahmed, Farook Sattar, Mohammed Yakoob Siyal, and Dan Yu

School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798

Received 6 April 2005; Revised 17 January 2006; Accepted 29 January 2006

Recommended for Publication by Mauro Barni

We propose a secure watermarking scheme that integrates watermarking with cryptography for addressing some important issues in copyright protection. We address three copyright protection issues—buyer-seller identification, copyright infringement, and ownership verification. By buyer-seller identification, we mean that a successful watermark extraction at the buyer's end will reveal the identities of the buyer and seller of the watermarked image. For copyright infringement, our proposed scheme enables the seller to identify the specific buyer from whom an illegal copy of the watermarked image has originated, and further prove this fact to a third party. For multiple ownership claims, our scheme enables a legal seller to claim his/her ownership in the court of law. We will show that the combination of cryptography with watermarking not only increases the security of the overall scheme, but it also enables to associate identities of buyer/seller with their respective watermarked images.

1. INTRODUCTION

With rapid growth of the Internet, security of digital images is becoming a great concern. It has now become very easy to illegally copy, modify, and retransmit a digital image. Digital watermarking is a technique that provides a way to protect digital images from illicit copying and manipulation. A digital watermark is an imperceptible signal added to digital data, called cover work, which can be detected later for buyer/seller identification, ownership proof, and so forth [1]. A digital watermarking scheme can either be *symmetric* or *asymmetric*. A symmetric watermarking scheme uses identical keys for watermark embedding and detection [2]. This possesses a security weakness as the information used to detect a watermark can be used to remove it. This restricts the use of symmetric watermarking, as the number of authorized detectors has to be strictly controlled. To solve this problem, asymmetric watermarking schemes have been proposed that use different keys for watermark embedding and detection [3–6]. This makes the use of watermarking possible for public domain applications where any one with the detection key can check the embedded watermark. However, the practical use of asymmetric watermarking requires careful considerations [7]. It is worth noting that merely using a watermarking algorithm does not completely address the issues of copyright protection. To devise a secure watermarking scheme, it is necessary that a watermarking algorithm is well integrated with a secure protocol [8, 9]. For example, in [10], an interactive buyer-seller protocol is proposed that prevents a seller from knowing the exact watermarked copy he/she creates for a buyer. Therefore, the seller cannot create copies of the original content that contains the buyer's watermark. The protocol further allows the seller to identify a buyer from whom an unauthorized copy has originated and prove this fact to a third party.

Our primary aim in this paper is to devise cryptographic protocols and integrate them with some of the existing watermarking techniques in order to address the issues related to buyer/seller identification and copyright protection. To further elaborate our motivation, we present a few scenarios. Suppose Alice sells a watermarked image to Bob. Later in time, Bob starts selling Alice's watermarked image using his fake watermarks. How will Alice prevent Bob from doing this? If the watermarked image consists of both Alice's and Bob's watermarks, how will the actual owner (Alice) be identified? If Bob somehow removes Alice's watermark from the image in dispute, is there any way for Alice to claim her genuine ownership? Consider another scenario. Alice wants to sell a watermarked image I_w to Bob such that the extraction of the watermark from I_w is a legal proof that Bob has indeed purchased I_w from Alice. How will such a watermark be designed whose extraction reveals identities of the buyer/seller?

In this paper, we will show that the combination of cryptography with watermarking not only increases the security of the overall scheme but it also enables to associate identities of the buyer/seller with their respective watermarked images. Specifically, we will focus on three issues of copyright protection, that is, buyer-seller identification, copyright infringement, and verification of ownership. By buyer-seller identification, we mean that a successful watermark extraction at the buyer's end will reveal the identities of the buyer and seller of the watermarked image. In case of copyright infringement from a buyer, the proposed scheme enables the seller to identify the specific buyer from whom an illegal copy of a watermarked image has originated, and further prove this fact to a third party. By ownership verification, we mean that the seller of a watermarked image should be able to prove his/her legal ownership in case of multiple ownership claims.

The rest of the paper is organized as follows. Section 2 presents an overview of some of the terminologies used in this paper and describes certain assumptions. In Sections 3 and 4, we describe the watermark embedding and extraction processes, respectively. In Sections 5 and 6, we present details of the copyright protection protocols. Section 7 concludes the paper.

2. PRELIMINARIES

Before we describe our watermarking scheme and related protocols, we give an overview of some of the terminologies and describe certain assumptions made in the paper. We assume that there exists a certification trusted authority (CTA) whose purpose is to generate watermarks and issue them to any user upon request. The CTA is memory-less and does not keep a track record of the watermarks issued to different users. At any instant in time, the CTA can issue watermarks to a single seller. It is further assumed that each time a seller requests for watermarks, the CTA issues unique watermarks. We represent the seller of a watermarked image as Alice. For encryption/decryption and digital signatures, we use the RSA public key cryptosystem [11]. We denote encryption and decryption with the functions $E_K(\cdot)$ and $D_K(\cdot)$, respectively. The subscript K is used to represent the cryptographic key used for encryption/decryption. For the purpose of illustration, assume $(K_C{}^{\text{pub}}, K_C{}^{\text{pri}})$ to be the respective public and private key pair of the CTA. Let $(K_A^{\text{pub}}, K_A^{\text{pri}})$ be the respective public and private key pair of Alice. We represent digital signature by the function $S_S(\cdot)$. The subscript S represents the signer's identity. For example, for a message X, the digital signature of the CTA will be represented by $S_C(X)$. We now give a brief overview of hash function, digital signature, and blind source separation.

2.1. Hash function

Suppose a message is to be sent that contains "p'' symbols and we would like to reduce the length of the message to say "k'' symbols. A cryptographic hash function [12] $H(x)$ maps the set of "p'' symbols to a set of "k'' symbols if $H(x)$ is easy to compute from x, however,

(i) it is computationally difficult to find two different values of x that gives the same $H(x)$, that is, a hash function is *collision free*;
(ii) given y in the image of $H(\cdot)$, no one can feasibly find an x such that $H(x) = y$, that is, a hash function is *preimage resistant*.

There are a number of hash functions proposed in the literature. The two famous ones are SHA and MD5 that give 160-bit and 128-bit hash values, respectively, for any length of a message [13]. Hash functions are also called message-digest algorithms.

2.2. Digital signature

A digital signature of a message is a number dependent on some secret known only to the signer, and additionally on the content of the message being signed. It provides a way to protect the integrity of a digital document and to verify who signed it. One way to implement a digital signature scheme is to use a one-way hash function and the RSA public key cryptosystem [14].

2.2.1. Signature generation

Suppose Alice wants to send a digitally signed message m to Bob. Alice will calculate the digital signature as follows.

(i) Transform the message m to a message digest $H(m)$.
(ii) Encrypt $H(m)$ with her private key to get the digital signature $S_A(m) : S_A(m) = E_{K_A^{\text{pri}}}(H(m))$.
(iii) Send the pair $[m, S_A(m)]$ to Bob.

2.2.2. Signature verification

At the receiving end, Bob will verify Alice's signature as follows.

(i) Decrypt $S_A(m)$ with Alice's public key to obtain $H(m)$: $H(m) = D_{K_A^{\text{pub}}}(S_A(m))$.
(ii) Compute the hash $\overline{H(m)}$ of the message m (for the purpose of clarity, the notation $\overline{H(X)}$ is used in the signature verification stage to represent the computed hash of a message X).
(iii) If $\overline{H(m)} = H(m)$, the signature will be considered valid.

2.3. Blind source separation using independent component analysis

Independent component analysis (ICA) is probably the most widely used method for performing blind source separation (BSS). It is a very general-purpose statistical technique to recover the independent sources given only sensor observations that are linear mixtures of independent source signals [15, 16]. ICA model consists of two parts: the mixing process and the unmixing process. In the mixing process, the observed linear mixtures $x_1, \ldots, x_m$ of n number of independent

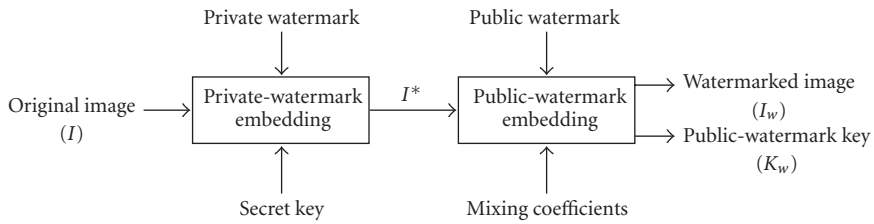

Figure 1: Block diagram of watermark embedding.

components are defined as

$$x_j = a_{j1}s_1 + a_{j2}s_2 + \cdots + a_{jn}s_n, \quad 1 \le j \le m, \tag{1}$$

where $\{s_k,\ k = 1,\ldots,n\}$ denote the source variables, that is, the independent components, and $\{a_{jk},\ j = 1,\ldots,m;\ k = 1,\ldots,n\}$ are the mixing coefficients. In vector-matrix form, the above mixing model can be expressed as

$$x = As, \tag{2}$$

where

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix} \tag{3}$$

is the mixing matrix, $x = [x_1 x_2 \cdots x_m]^T$, $s = [s_1 s_2 \cdots s_n]^T$, and T is the transpose operator. The unmixing process [15, 16] can be formulated by computing the separation/unmixing matrix Q so that the independent components can be obtained as

$$s = Qx. \tag{4}$$

The simplest BSS model assumes that there is the same number of linear mixtures as the independent components or sources. The objective of BSS is to find a linear representation in which the components are statistically independent. For performing BSS, techniques such as principal component analysis (PCA) [17] are not feasible as they give components that are uncorrelated. However, there are many uncorrelated representations of signals that are actually not independent. As a matter of fact, independence is a much stronger property than uncorrelatedness. Independence implies uncorrelatedness, however, the opposite is not true. The goal of ICA is much broader than PCA as it gives components that are not only uncorrelated but statistically independent as well. This makes ICA suitable for performing BSS. It is to be noted that for the ICA model in (2), two major ambiguities exist. The first ambiguity is that we cannot determine the variances or energies of the extracted independent components as both the mixing matrix and the original independent components are unknown. This may also create ambiguity in the sign of the extracted components. The second ambiguity is that we cannot determine the original order of the independent components as both the mixing matrix and the original independent components are unknown. In Section 4, we will show how these ambiguities are addressed in our watermarking scheme.

The use of ICA in watermarking application is not new. Noel and Szu [18] were among the first to introduce ICA in watermarking application. Likewise, Yu and Sattar [19] have proposed a blind watermarking technique using ICA. In this paper, we have used ICA for extracting the public watermark from the watermarked image. The basic idea behind our work is to use some specific image pattern as the public watermark, for example, see Figure 2. In Sections 4 and 5, we demonstrate how such patterns can be used within a cryptographic framework to represent the identities of the buyer and seller of a watermarked image. The image to be watermarked is linearly mixed with the public watermark to get the watermarked image. Hence, watermark extraction can be viewed as a blind source separation problem. To perform BSS, we have used ICA to extract the public watermark from the watermarked image.

3. WATERMARK EMBEDDING

In this section, we describe the procedure for watermark embedding. Let the seller's original image be denoted by I and the watermarked image by I_w. To address buyer-seller identification and copyright protection, our proposed watermarking scheme uses two different watermarks. The first watermark is used to reveal the identity of the buyer and seller of the watermarked image. We name this watermark as the *public watermark* W^{pub}. The second watermark serves two purposes. Firstly, it enables a legal seller to prove his/her ownership in case of multiple ownership claims. Secondly, in case of copyright infringement by a buyer, the extraction of this watermark will enable the seller to identify the malicious buyer from whom an illegal copy of the watermarked image has originated and further prove this fact to a third party. We call this watermark as the *private watermark* W^{pri}. Figure 1 shows the block diagram for watermark embedding. The private watermark W^{pri} is first embedded into the original image I to get an *intermediate-watermarked image* I^*. The image I^* is then further watermarked with the public watermark W^{pub} to get the final watermarked image I_w and the public-watermark key. We will show in the watermark extraction procedure that embedding the public watermark after embedding the private does not have any significant

degradation on the private-watermark extraction. In the following sections, we present the details of private- and public-watermark embedding.

3.1. Private-watermark embedding

The private watermark is required to be very robust because of three main reasons. Firstly, it is used to resolve copyright infringement and multiple ownership claims. Secondly, since the public watermark is embedded after embedding the private watermark, the private watermark should withstand the distortions introduced due to public-watermark embedding. The third reason for the private watermark to be robust is because it is the only means through which a genuine owner can prove his/her ownership in case the public watermark is destroyed. As Mintzer and Braudaway [20] have pointed out, in case of multiple watermark embedding, different watermarks might have different robustness requirements. Secondly, the order of embedding the watermarks is also very important. Mintzer and Braudaway suggest that the ownership watermark should be the most robust and should be embedded first; the most fragile watermark should be embedded last, while moderately robust watermark(s) should be inserted in between. For successful multiple watermark embedding, the robust watermark that is embedded first should be able to withstand all the subsequent watermark insertions [20]. In our work, since the private watermark is the most important, it is embedded first which is then followed by public-watermark embedding. We have used the spread spectrum watermarking technique proposed by Cox et al. [2] to embed the private watermark. This technique is very robust against a number of attacks as discussed in [2]. In addition, the watermark pattern can be detected even if an image is watermarked multiple number of times.

The private watermark W^{pri} is a sequence of real numbers:

$$W^{\text{pri}} = \{w_1, w_2, \ldots, w_n\}, \tag{5}$$

where each w_i is chosen independently according to a normal distribution with zero mean and unit variance. The DCT of the original image I is taken and the watermark sequence w_i is embedded in the 1000 (N=1000) highest-valued AC coefficients using the following relation [2]:

$$\hat{z}_i = z_i(1 + \alpha_1 w_i), \tag{6}$$

where z_i is the ith highest-valued AC DCT coefficient of I and α_1 controls the strength of the watermark. The modified DCT coefficients $\hat{z}_i$ are then inserted back in place of z_i and an inverse DCT is taken to get the intermediate-watermarked image I^*.

3.2. Public-watermark embedding

The purpose of embedding the public watermark is to enable anyone with the knowledge of the *public-watermark key* to extract the public watermark. The public watermark is used to identify the buyer and seller of the watermarked image.

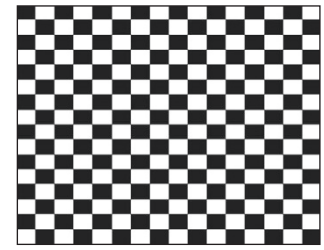

Figure 2: Public watermark.

An important question that arises is how to design a public watermark that can be associated with some information. For example, if the public watermark is required to reveal the identities of the buyer and seller of a watermarked image, how can this be achieved? We address this issue by using a watermark that portrays the hash of the information that is required to be associated with the watermark. Some commonly used hash functions are MD5 and SHA1 that give 128-bit and 160-bit hash values, respectively [14]. Suppose M is a piece of information that uniquely identifies the buyer and seller of a watermarked image. In our proposed scheme, we use the public keys of the buyer and seller for identification purpose. It should be noted that in a real-world scenario, public keys are certified by some trusted certification authority and therefore can be used for identification purpose. We obtain M by concatenating the public keys of the buyer and seller of the watermarked image. The seller calculates the hash of M to get $H(M)$ and generates a watermark that portrays $H(M)$. The reason for using a cryptographic hash function is that no matter how long is M, the hash output will be compressed to 128 bits in case of MD5, or 160 bits in case of SHA1. For the purpose of illustration, suppose we have a hash sequence $H(M) = 0101010\ldots10$. The public watermark for such a sequence is shown in Figure 2. The watermark pattern can accommodate 256 bits of information. The box in black represents a "0" while the box in white represents a "1." In case the hash function used is MD5, the hash output will be 128 bits. Since our watermark pattern can accommodate 256 bits of information, the remaining blocks in the watermark can be zero-padded or the hash pattern can be tiled to cover the entire image area of the watermark. For the sake of convenience, we use the public-watermark pattern shown in Figure 2 in our discussion to follow. In our experiments, a black pixel in Figure 2 is represented by a gray value of zero, while a white pixel is represented by a gray value of 255. We segment the public watermark into $W^{\text{pub}\,1}$ and $W^{\text{pub}\,2}$ as shown in Figures 3 and 4, respectively:

$$W^{\text{pub}} = W^{\text{pub}\,1} + W^{\text{pub}\,2}. \tag{7}$$

A third-level wavelet decomposition of the intermediate-watermarked image I^* is performed and the public watermark is embedded in the LL subband. If the image I^* is of dimension $N \times N$, then the size of the public watermark will be $(N/8 \times N/8)$. As pointed out in [2], for a watermark to be robust, it should be embedded in the perceptually most significant components of the image spectrum. We embed

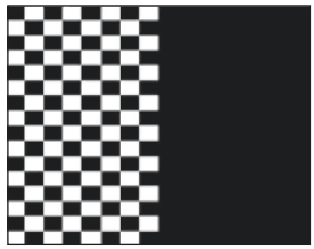

Figure 3: Segment 1 of the public watermark.

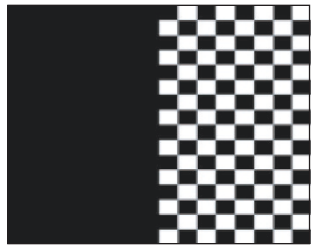

Figure 4: Segment 2 of the public watermark.

Figure 5: Original image.

Figure 6: Intermediate watermarked image.

the public watermark in the *LL* subband for increased robustness as the *LL* subband contains the most important information of an image. The embedding coefficients should however be carefully chosen keeping in view that for a particular value of the embedding coefficient, the *LL* subband is more susceptible to perceptual distortion as compared to the other subbands. Let us denote the third-level *LL* subband wavelet coefficient of I^* by Y_{LL3}. The following are the steps for public-watermark embedding.

(1) Perform the third-level discrete wavelet decomposition of I^*. Embed $W^{\text{pub}\,1}$ and $W^{\text{pub}\,2}$ separately in Y_{LL3} to the following rules:

$$\hat{Y}1_{LL3} = Y_{LL3} + \alpha_2 \cdot W^{\text{pub}\,1}, \tag{8}$$

$$\hat{Y}2_{LL3} = Y_{LL3} + \alpha_2 \cdot W^{\text{pub}\,2}, \tag{9}$$

where α_2 controls the watermark embedding strength and $\hat{Y}1_{LL3}$, $\hat{Y}2_{LL3}$ are the modified *LL* subband wavelet coefficients after embedding the watermark.

(2) The watermarked image I_w is obtained by replacing the Y_{LL3} coefficients of I^* by the modified $\hat{Y}1_{LL3}$ coefficients and then taking the inverse discrete wavelet transform. The inverse discrete wavelet transform takes into account all the frequency subbands.

(3) The modified wavelet coefficients $\hat{Y}2_{LL3}$ are then scaled and rounded off into an n-bit integer to obtain the public-watermark key K_w. The purpose of scaling and rounding off is to compress the size of K_w. The coefficient of $\hat{Y}2_{LL3}$ that has the minimum value is always mapped to zero while the coefficient of $\hat{Y}2_{LL3}$ that has the maximum value is mapped to 2^n. The remaining coefficients are linearly mapped between the values zero and 2^n using the equation of a straight line. The mapped wavelet coefficients are then rounded off to the nearest integer. For an 8-bit gray-level image having 256×256 pixels, there will be a total of 1024 $\hat{Y}2_{LL3}$ coefficients. By scaling and rounding off these coefficients to 8 bits, the size of K_w will be compressed to 1024 bytes. By choosing different values of n, the size of K_w can be controlled. We have experimentally observed that scaling and rounding off to a 10-bit integer gives good extraction results. In this case, the size of K_w will be 1280 bytes.

Figure 5 shows the original cameraman image I, while Figure 6 shows the intermediate-watermarked image I^* obtained by following the steps outlined in Section 3.1. Figures 7 and 8 show the watermarked image I_w and the corresponding public-watermark key K_w obtained by the steps outlined above.

4. WATERMARK EXTRACTION

4.1. Public-watermark extraction

The public watermark is extracted from the watermarked image using the public-watermark key K_w. Figure 9 shows the block diagram of the public-watermark extraction. A third-level discrete wavelet decomposition of the watermarked image is first performed to get the *LL* subband coefficients, Y_w. Note that the dimensions of Y_w and K_w are the same. The matrix Y_w is a linear mixture of Y_{LL3} and $W^{\text{pub}\,1}$ (8), while the matrix K_w is a linear mixture of Y_{LL3} and $W^{\text{pub}\,2}$ (9). We therefore have a total of three sources Y_{LL3}, $W^{\text{pub}\,1}$, and $W^{\text{pub}\,2}$ in the two mixtures Y_w and K_w. To extract the public watermark, we have used blind source separation as discussed in Section 2.3. We have used Cardoso's JADE ICA algorithm [21] for watermark extraction. The mixtures Y_w and

Figure 7: Watermarked image.

Figure 8: Public-watermark key.

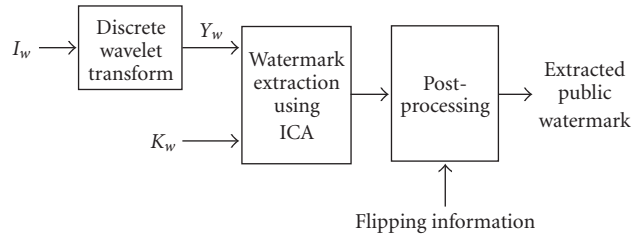

Figure 9: Block diagram of public-watermark extraction.

Figure 10: Extracted output 1.

K_w are treated as inputs to the blind source separation process. Since we are using two mixtures, the BSS process will give us two outputs, as shown in Figures 10 and 11. The first output consists of a distorted version of Y_{LL3}. We call this as the residue output. The second output consists of two parts. The left half is similar to the left half of $W^{\text{pub}\,1}$ (Figure 3). The right half is however exactly the opposite of the right half of $W^{\text{pub}\,2}$ (Figure 4). This change in sign is because of the sign ambiguity present in the ICA algorithm as discussed in Section 2.3. By scaling the pixels values in Figure 11 to gray-level range between 0 and 255 and flipping the right half, we get the extracted watermark $\widehat{W}^{\text{pub}}$ as shown in Figure 12.[1] The binary pattern of the extracted watermark shown in Figure 12 is similar to the watermark embedded (Figure 2). In some cases, it might be required to flip the left half. Since the seller of the watermarked image knows the exact pattern of the public watermark, he/she can carry out BSS to see which half of the extracted output shown in Figure 11 is required to be flipped. This information can then be conveyed to the recipient. We have used ICA for watermark extraction because of the scaling and rounding off as performed in step (3) of the public-watermark embedding. Because the $\hat{Y}2_{LL3}$ coefficients obtained from (9) are scaled and rounded off into an n-bit integer, a simple subtraction of Y_w and K_w will not work.

4.2. Private-watermark extraction

The private-watermark extraction is nonblind and requires the original image I. To extract the private watermark, the DCT of the watermarked image I_w and the original image I is taken and the watermark sequence is extracted from the embedding locations using (6):

$$\bar{w}_i = \frac{1}{\alpha_1}\left(\frac{\bar{z}_i}{z_i} - 1\right), \tag{10}$$

where $\bar{w}_i$ is the extracted watermark sequence and $\bar{z}_i$ are the DCT coefficients of I_w. The extracted watermark is then compared with the original watermark using some similarity measure. We use the normalized correlation coefficient [1] as our similarity measure. For our experiments, the values of α_1 and α_2 were chosen as 0.08 and 0.07, respectively. Since we are using the normalized correlation coefficient as our similarity measure, the value of α_1 is not required to be known for comparing the extracted watermark with the reference watermark.

It is important to note that the embedding of the public watermark should not cause any significant degradation in the private watermark. Interestingly, due to the robustness property of the spread-spectrum watermarking technique, the public watermark introduces a slight decrease in the correlation value of the private watermark from 1.00 to 0.96. This interference can be further minimized by carefully choosing the domain where both watermarks are embedded.

[1] The JADE algorithm that we have used in this paper is based on linear mixing model, fourth-order statistics, and noniterative approach. Although this algorithm works well in the wavelet domain; like other BSS algorithms, there are ambiguities present with this algorithm, like scaling and sign change. Because of this reason, the hash bits of the right half of the extracted public watermark (Figure 11) are toggled, that is, a binary "one" becomes "zero" and a binary "zero" becomes "one." This will therefore give an incorrect value of the hash. To compensate this problem, the portion of the extracted output that has been inverted is flipped. Furthermore, the left half of Figure 11 appears different from the left half of Figure 12. This is due to the high contrast between the left and the right half of Figure 11.

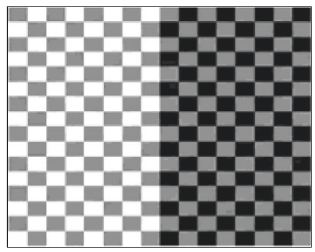

Figure 11: Extracted output 2.

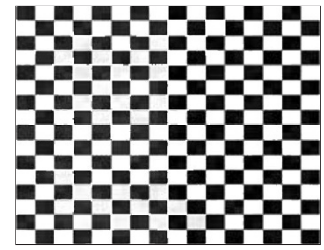

Figure 12: Extracted public watermark.

For example, if we embed the public watermark in the LH or HL subband instead of the LL-subband, it will cause less interference with the private watermark. However, there will be a loss in robustness of the public watermark.

5. COPYRIGHT PROTECTION PROTOCOLS

Our proposed watermarking scheme consists of the following protocols to deal with copyright protection issues:

(I) watermarked image generation and distribution protocol;
(II) buyer-seller identification protocol;
(III) copyright infringement protocol.

In Section 6, we discuss a few more protocols that can be used for resolving ownership claims in case of multiple ownership disputes.

5.1. Watermarked image generation and distribution protocol

Suppose Alice wants to sell a watermarked image to Bob. This protocol will enable Alice to acquire a watermark certificate Cer from the CTA that contains a valid private watermark and digital signatures. Let $(K_B^{\text{pub}}, K_B^{\text{pri}})$ be the respective public and private key pair of Bob. Figure 13 shows the flow diagram of the watermarked image generation and distribution protocol. The protocol proceeds as follows.

(1) Alice hashes her original image I to get $H(I)$. She then sends $H(I)$, her public key K_A^{pub}, and certificate of her identity to the CTA along with a request for issuing a watermark.
(2) CTA verifies Alice's identity. It then generates the private watermark W_A^{pri} for Alice. The private watermark is a pseudorandom noise sequence as described by (5).
(3) CTA calculates the hash $H(W_A^{\text{pri}}, H(I), T_1)$. The parameter T_1 indicates the time stamp that is used to resolve ownership disputes. CTA encrypts $H(W_A^{\text{pri}}, H(I), T_1)$ with its private key K_C^{pri} to get digital signatures for $H(I)$ and W_A^{pri}:

$$S_C(W_A^{\text{pri}}, H(I), T_1) = E_{K_C^{\text{pri}}}(H(W_A^{\text{pri}}, H(I), T_1)). \quad (11)$$

(4) A tuple X_A is formed as shown by (12). A digital signature $S_{CA}(X_A, T_1)$ is obtained by encrypting $H(X_A, T_1)$ with CTA's private key and then with Alice's public key:

$$X_A = \{W_A^{\text{pri}}, S_C(W_A^{\text{pri}}, H(I), T_1), T_1\}, \quad (12)$$

$$S_{CA}(X_A, T_1) = E_{K_A^{\text{pub}}}(E_{K_C^{\text{pri}}}(H(X_A, T_1))). \quad (13)$$

(5) CTA sends the watermark certificate Cer_A to Alice:[2]

$$\text{Cer}_A = \{X_A, S_{CA}(X_A, T_1)\}. \quad (14)$$

(6) Alice verifies Cer_A by first decrypting $S_{CA}(X_A, T_1)$ with her private key and then further decrypting the result with CTA's public key to get $\overline{H(X_A, T_1)}$. She then hashes X_A and T_1 to get $\overline{H(X_A, T_1)}$. If $\overline{H(X_A, T_1)} = H(X_A, T_1)$, it will be verified that Cer_A has been generated by the CTA and that it has not been tampered. Alice then uses the watermark W_A^{pri} obtained from Cer_A to generate the intermediate-watermarked image I_A^* using the steps outlined in Section 3.1.
(7) Alice hashes K_A^{pub}, K_B^{pub} and uses the hash bits to generate the public watermark W_A^{pub}. She then segments W_A^{pub} into $W_A^{\text{pub}\,1}$ and $W_A^{\text{pub}\,2}$ using (7). Using the steps outlined in Section 3.2, she generates the watermarked image I_{Aw} and the public-watermark key K_{Aw}. She then encrypts K_{Aw} with her private key to get CK_{Aw}:

$$CK_{\text{Aw}} = (E_{K_A^{\text{pri}}}(K_{\text{Aw}})). \quad (15)$$

(8) Alice calculates $H(W_A^{\text{pri}})$ and sends it along with I_{Aw} and CK_{Aw} to Bob.[3]
(9) In this step, Bob will verify the genuine buyer-seller transaction between him and Alice. Bob performs the following steps.
    (I) Decrypt CK_{Aw} with Alice's public key to get K_{Aw}. Using I_{Aw} and K_{Aw}, extract the public watermark $\widehat{W}_A^{\text{pub}}$ according to the procedure outlined in Section 4.1.

[2] Instead of using W_A^{pri} and its corresponding digital signature in Cer_A, the CTA can also use a seed (that can be used with a secure publicly known pseudorandom number generator) and its corresponding digital signature. This will save bandwidth. For further security, the CTA can also encrypt Cer_A with Alice's public key and then transmit the encrypted version of Cer_A to Alice.
[3] Although not mentioned, the flipping information for postprocessing of the public watermark will also be transmitted.

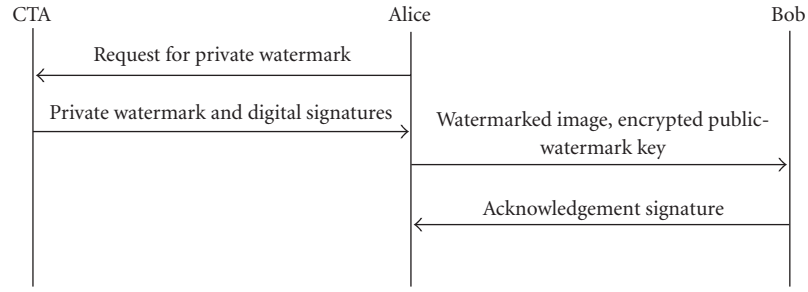

Figure 13: Flow diagram of watermarked image generation and distribution protocol.

(II) Hash K_A^{pub}, K_B^{pub} and compare the output of the hash function with the binary pattern obtained from $\widehat{W}_A^{\text{pub}}$.

After performing step (I), Bob will only be successful in extracting a genuine watermark pattern (like the binary pattern shown in Figure 2), if the public-watermark key has been encrypted with Alice's private key. This will also prove that the extracted watermark has been embedded by Alice as no one else is supposed to know Alice's private key other than herself. Furthermore, if step (II) is successful, Bob will be convinced that $\widehat{W}_A^{\text{pub}}$ reflects his and Alice's identities.

(10) After positive verification in step (9), Bob sends the following to Alice:

$$S_B\left(I_{\text{Aw}}, H\left(W_A^{\text{pri}}\right), K_A^{\text{pub}}\right) = E_{K_B^{\text{pri}}}\left(H\left(I_{\text{Aw}}, H\left(W_A^{\text{pri}}\right), K_A^{\text{pub}}\right)\right). \tag{16}$$

(11) Alice verifies $S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ and stores Cer_A, I_{Aw}, CK_{Aw}, and $S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ as a record of this transaction with Bob.[4]

5.2. Buyer-seller identification protocol

Suppose Alice makes a selling transaction with Bob as discussed in Section 5.1. The protocol discussed in this section can be used by Bob or any other party to show that Bob is a genuine buyer of the watermarked image I_{Aw} sold to him by Alice. The protocol requires I_{Aw} and CK_{Aw} that Bob obtained from Alice in the watermarked image generation and distribution protocol along with Alice's and Bob's public keys.

Figure 14 shows the block diagram of the proposed buyer-seller identification protocol. The protocol proceeds as follows.

(1) Decrypt CK_{Aw} with Alice's public key to get K_{Aw}:

$$K_{\text{Aw}} = \left(D_{K_A^{\text{pub}}}\left(CK_{\text{Aw}}\right)\right). \tag{17}$$

(2) Using I_{Aw} and K_{Aw}, extract the public watermark $\widehat{W}_A^{\text{pub}}$ according to the procedure outlined in Section 4.1.

(3) Hash the public keys of Alice and Bob to get HPub of length L bits. For example, if the hash function used is SHA1, then L will be 160 bits:

$$\text{HPub} = H\left(K_A^{\text{pub}}, K_B^{\text{pub}}\right). \tag{18}$$

(4) Compare the binary bit sequence of HPub with the bit pattern obtained from $\widehat{W}_A^{\text{pub}}$. If all the bits are compared successfully, then it will be proved that Bob is the legal buyer of I_{Aw} sold to him by Alice.

Remark 1. Is it possible for Bob to embed the binary sequence HPub in any arbitrary image J and then claim that he is the legal buyer of J sold to him by Alice? It is easy for Bob to generate the pattern HPub shown by (18) since it only requires the knowledge of Alice's and Bob's public keys that are available in the public domain. However, Bob cannot obtain (15) (step (7) of the watermarked image generation and distribution protocol), since it requires the knowledge of Alice's private key. If the correct private key of Alice is not used in this step, then the result of decryption in step (1) of the buyer-seller identification protocol will not be correct. As a result, the extracted watermark will be gibberish. This shows that it is not possible for Bob to insert the pattern HPub in any arbitrary image J and then claim that he is the legal buyer of J sold to him by Alice.

5.3. Copyright infringement protocol

Suppose Alice finds an illegal copy $\hat{I}_{\text{Aw}}$ of the watermarked image I_{Aw} that she had previously sold to Bob. Using this protocol, a judge can check whether $\hat{I}_{\text{Aw}}$ has originated from

[4] It is not necessary that Alice stores I_{Aw} as this will add an extra storage overhead. Instead she can regenerate the watermarked image I_{Aw} when required using step (7) of the watermarked image generation and distribution protocol. This will require some extra storage requirements like the watermark embedding strength parameters and any secret key used in watermark embedding, and so forth. This storage however will be quite less as compared to storing the entire watermarked image. In order to make sure that the watermarked image regenerated in the future is 100% similar to the one that was generated in the past, Alice can store the cryptographic hash of I_{Aw}.

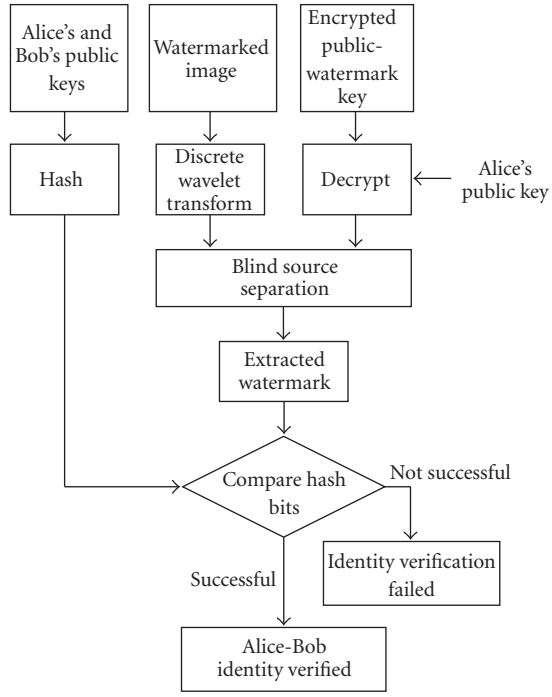

Figure 14: Block diagram of buyer-seller identification protocol.

the watermarked image I_{Aw}.[5] Figure 15 shows the block diagram of this protocol. This protocol requires either one or two stages to complete.

Stage 1. In this stage, the judge will follow the steps outlined in the buyer-seller identification protocol (Section 5.2) to extract the public watermark $\widehat{W}_A^{\text{pub}}$ from $\hat{I}_{\text{Aw}}$ using CK_{Aw} (supplied by Alice from Bob's transaction record). If the extracted watermark depicts Alice's and Bob's identities, then Bob will be liable for copyright infringement. Bob can however be smarter. Since he knows the public watermark, he can subtract its scaled version from $\hat{I}_{\text{Aw}}$ such that $\widehat{W}_A^{\text{pub}}$ is not detected in $\hat{I}_{\text{Aw}}$. In such a case, Stage 2 of the protocol will be used.

Stage 2. In this stage, the judge will extract an estimation of the private watermark $\widehat{W}_A^{\text{pri}}$ from $\hat{I}_{\text{Aw}}$ to check whether Bob is guilty or not. In this stage, Alice will supply the judge with I_{Aw}, W_A^{pri}, and $S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ from Bob's transaction record along with her original image I. The protocol proceeds as follows.

(1) Use Alice's original image I to extract the private watermark $\widehat{W}_A^{\text{pri}}$ from $\hat{I}_{\text{Aw}}$ using the procedure outlined in Section 4.2.

(2) Decrypt $S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ with Bob's public key to get $H(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$:

$$H(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}}) = D_{K_B^{\text{pub}}}(S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})). \tag{19}$$

(3) Use I_{Aw}, W_A^{pri} (supplied by Alice from her transaction record for Bob) and K_A^{pub} to get $\overline{H(I_{\text{Aw}})}$, $\overline{H(W_A^{\text{pri}}), K_A^{\text{pub}}}$.

(4) If $\overline{H(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})} = H(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$, it will be proved that Bob had purchased the watermarked image I_{Aw} from Alice that contains the private watermark W_A^{pri}. The reason for this is because in step (2), $H(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ is obtained by decrypting $S_B(I_{\text{Aw}}, H(W_A^{\text{pri}}), K_A^{\text{pub}})$ using Bob's public key and that it contains Alice's public key as an argument.

(5) Bob will be considered guilty of copyright infringement if the following are true:

(i) $\widehat{W}_A^{\text{pri}}$ and W_A^{pri} match with high correlation;
(ii) $\hat{I}_{\text{Aw}}$ and I_{Aw} match with high correlation.

Remark 2. If Alice has sold different watermarked versions of the same cover image to different customers, how will she identify the particular customer from whom an illegal copy has originated? This task may become complicated, especially if the number of clients grows huge. Before reporting the case to the judge, Alice will first have to find out the identity of the buyer from whom the illegal copy has originated. For example, for each cover work, I_1, I_2, I_3, and so on that she watermarks and sells, she can maintain a separate database of all the private watermarks that she has embedded into that particular cover work. Now if she finds, for example, an illegal image I', first she will sort out that for which cover work I' belongs. This can be done by using a number of image processing techniques that are available for efficient and effective comparison of images. Once I' is matched with a particular cover work, say I_2, Alice will then narrow her search by extracting the number of possible watermarks she has embedded in I_2 for different clients. In case Alice uses the same secret locations to embed private watermarks, she will have to extract only a single watermark from I'. The extracted watermark will then be compared with all the watermarks that she has stored with respect to the cover work I_2. The match with the highest correlation will enable her to decide about the buyer. After this she may report that particular buyer to the judge.

6. RESOLVING MULTIPLE OWNERSHIP CLAIMS

In this section, we discuss problems that arise in case of multiple ownership claims over a watermarked image. In particular, we illustrate the following three attacks and show how our proposed scheme can resist such attacks:

(I) multiple watermarked image attack;
(II) invertible watermark attack;
(III) watermark removal attack.

[5] In this paper, we have not considered the case in which an unauthorized copy of a watermarked image is distributed by a malicious seller or due to a security breach in the buyer/seller system. This problem has been addressed in [10].

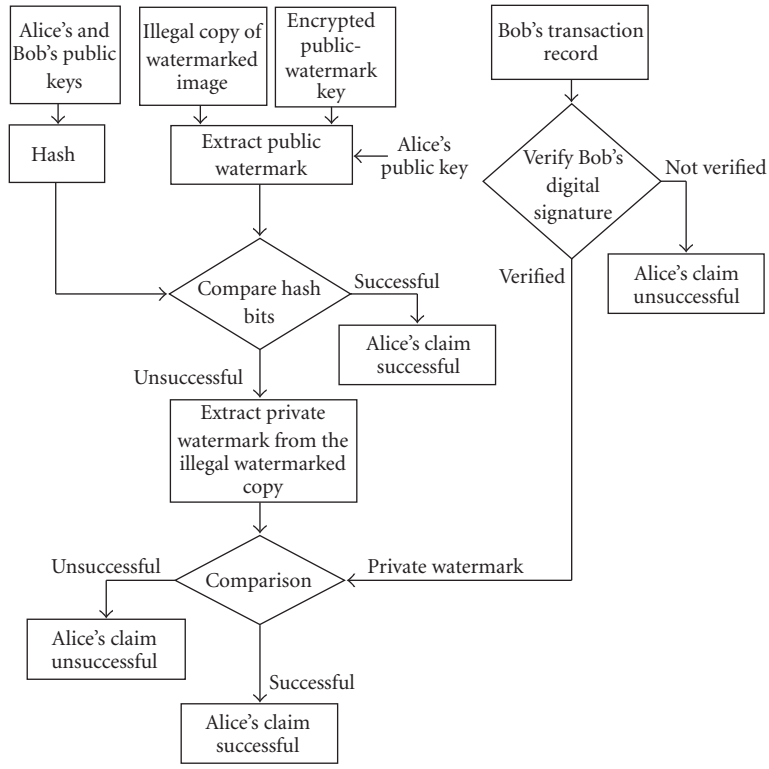

Figure 15: Block diagram of copyright infringement protocol.

Throughout the discussion to follow, assume that Alice watermarks her original image I using the private watermark W_A^{pri} obtained from the CTA's watermark certificate given by (14) to get a watermarked image I_{Aw}.

6.1. Multiple watermarked image attack

Suppose Bob obtains a copy of I_{Aw} and further watermarks it by using his private watermark W to get the watermarked image I_{ABw}, for which he claims to be the legal owner. Resolving an ownership dispute between Alice and Bob over I_{ABw} is quite straightforward if the watermarking technique is robust [2, 22]. For example, in case of the spread-spectrum technique that we have used in this paper, both Alice's and Bob's watermarks can be detected in the disputed image I_{ABw}. Bob with his fake original I_{Aw} can show the presence of his watermark W in I_{ABw}. However, he cannot show the presence of W in Alice's original image I. Alice on the other hand can show the presence of her watermark W_A^{pri} both in Bob's fake original I_{Aw} and as well as in the disputed image I_{ABw}. In this way, Alice can prove her legal ownership of I_{ABw}. To show a numerical example, we watermarked the cameraman image I shown in Figure 5 with a PN sequence W_A^{pri} to represent I_{Aw}. We then watermarked I_{Aw} with another PN sequence W to get another watermarked image that we represent byI_{ABw}. In both cases, we kept the embedding strength of the watermark α_1 as 0.08. Using I as the original image, the watermark W_A^{pri} was detected in I_{Aw} and I_{ABw} with a normalized correlation coefficient of 0.998 and 0.691, respectively. Similarly, using I_{Aw} as the original image, the watermark W was detected in I_{ABw} with a normalized correlation coefficient of 0.995. However, the normalized correlation coefficient for W in I was only -0.0316. These results confirm our above discussion.

6.2. Invertible watermark attack

The scenario depicted in Section 6.1 enables Alice to claim her legal ownership because Bob cannot show the presence of his watermark in Alice's original image I. What if Bob is able to show the presence of his fake watermark in Alice's original image? This might lead to an ownership deadlock. In fact, Craver et al. [22] were the first to show such a scenario. The attack proposed in [22] works for the decoding strategy shown by (10) in which the extraction of the private watermark is nonblind. The idea is quite simple. In contrast to what we showed in Section 6.1, Bob does something smarter. Instead of embedding a watermark W in I_{Aw}, he subtracts W from I_{Aw} to get an image $\tilde{I}_{\text{Aw}}$ which he calls his original. Let us denote the watermark embedding and subtraction operators by $\oplus$ and Θ, respectively. With this notation, Alice's watermarked image and Bob's fake original are represented

by (20) and (21), respectively:

$$I_{\mathrm{Aw}} = I \oplus W_A^{\mathrm{pri}}, \tag{20}$$

$$\bar{I}_{\mathrm{Aw}} = I_{\mathrm{Aw}} \Theta W = (I \oplus W_A^{\mathrm{pri}}) \Theta W. \tag{21}$$

In terms of I_{Aw} and I, (21) can be written as follows:

$$\begin{aligned} I_{\mathrm{Aw}} &= \bar{I}_{\mathrm{Aw}} \oplus W, \\ I &= (\bar{I}_{\mathrm{Aw}} \Theta W_A^{\mathrm{pri}} \oplus W). \end{aligned} \tag{22}$$

From (22) we can see that by using his fake original $\bar{I}_{\mathrm{Aw}}$, Bob can show the presence of his watermark W both in Alice's watermarked version I_{Aw} and Alice's original image I. Bob can therefore accuse Alice that I_{Aw} and I are indeed his copies of the watermarked image I_{Aw}. Similarly, Alice can also show the presence of her watermark W_A^{pri} in both $\bar{I}_{\mathrm{Aw}}$ and I_{Aw}. Craver et al. [22] have termed such a scheme as *invertible*, which can actually lead to ownership deadlock. To solve this problem, Craver et al. [22] suggested the following.

(i) Hash the original image I to generate a seed S.
(ii) The seed S is used by a fixed pseudorandom number generator to generate the watermark W_s.
(iii) The watermarked image $I_w = I \oplus W_s$.

Now an attacker cannot generate a fake original because generating a fake original requires subtraction of a watermark that is derived from the original itself. However, Ramkumar and Akansu have shown that this method is still invertible [23]. Their proposed attack requires about 10^9 trials to find the watermark for which the probability of error in watermark detection is bounded by 10^{-9}. This indicates that the attack is computationally feasible. The authors in [23] have also proposed some remedies to ensure noninvertibility. With our proposed protocol, however, an invertible watermark attack does not seem possible because the watermarks are issued by the CTA. Suppose Bob wants to create a fake original $\bar{I}_{\mathrm{Aw}}$ by subtracting a watermark W from I_{Aw}. However, with our proposed scheme, Bob does not have the liberty to choose W because the watermarks are issued by the CTA. Suppose the CTA issues Bob the watermark W_B^{pri}. Can Bob subtract W_B^{pri} from I_{Aw} to get $\bar{I}_{\mathrm{Aw}}$ and then claim $\bar{I}_{\mathrm{Aw}}$ to be his fake original? To answer this question, refer to step (1) of the watermarked image generation and distribution protocol. Before Bob can request for a private watermark from the CTA, he is required to send the hash of his original image to the CTA. Since Bob can only obtain his fake original $\bar{I}_{\mathrm{Aw}}$ once he subtracts W_B^{pri} from I_{Aw}, he cannot perform step (1) of the watermarked image generation and distribution protocol.

We now examine the security of our scheme against invertibility from another perspective. Bob calculates $H(I_{\mathrm{Aw}})$ and sends it to the CTA along with a request for a private watermark. Suppose CTA generates a private watermark W_B^{pri} and sends the following watermark certificate to Bob:

$$\begin{aligned} \mathrm{Cer}_B &= \{X_B,\ S_{CB}(X_B, T_2)\}, \\ X_B &= \{W_B^{\mathrm{pri}}, S_C(W_B^{\mathrm{pri}}, H(I_{\mathrm{Aw}}), T_2), T_2\}, \\ S_{CB}(X_B, T_2) &= E_{K_B^{\mathrm{pub}}}(E_{K_C^{\mathrm{pri}}}(H(X_B, T_2))). \end{aligned} \tag{23}$$

It is very realistic to assume that the time stamp $T_2 > T_1$, since Bob can only send I_{Aw} after Alice generates I_{Aw} (using Cer_A with the time stamp T_1). If Bob subtracts W_B^{pri} from I_{Aw} to get a fake original $\bar{I}_{\mathrm{Aw}}$, can he accuse Alice in the court of law that I_{Aw} and I are actually his copies of watermarked image I_{Aw}? We now show a protocol that will prevent Bob from claiming such false ownership. The protocol proceeds as follows.

(1) Alice and Bob present their original images I and $\bar{I}_{\mathrm{Aw}}$ along with their respective watermark certificates Cer_A and Cer_B to the judge.

(2) The judge will check whether the identities of Alice and Bob are associated with their respective watermark certificates Cer_A and Cer_B. In addition, the judge will also check the authenticity and integrity of Cer_A and Cer_B. For example, in case of Alice, the identity of Alice and the authenticity and integrity of Cer_A are verified as follows.

(a) Alice will first decrypt $S_{CA}(X_A, T_1)$ with her private key K_A^{pri}. The result is then further decrypted with the CTA's public key K_C^{pub} to get $H(X_A, T_1)$:

$$H(X_A, T_1) = D_{K_C^{\mathrm{pub}}}(D_{K_A^{\mathrm{pri}}}(S_{CA}(X_A, T_1))). \tag{24}$$

Note 1. The CTA obtains $S_{CA}(X_A, T_1)$ in step (4) (Section 5.1) by first encrypting $H(X_A, T_1)$ with CTA's private key and then with Alice's public key. Therefore, $S_{CA}(X_A, T_1)$ has to be first decrypted with Alice's private key and then with the CTA's public key to get back $H(X_A, T_1)$. However, in a real-world scenario, Alice cannot release her private key to the judge as this may compromise the security of the public key cryptosystem. As an alternative, she can decrypt $S_{CA}(X_A, T_1)$ with her private key and present the result to the judge. The judge can then again encrypt this result with Alice's public key. If the judge is able to get back $S_{CA}(X_A, T_1)$, the judge will be convinced that Alice has correctly performed the decryption. Using CTA's public key, the judge then further decrypts the decrypted result given by Alice to get $H(X_A, T_1)$. This is possible because in the RSA public key cryptosystem, which we have used in this paper, if the public key is used for encryption, then the corresponding private key is used for decryption and vice versa.

(b) Retrieve X_A from Cer_A and T_1 from X_A. Hash X_A and T_1 to get $\overline{H(X_A, T_1)}$. If $\overline{H(X_A, T_1)} = H(X_A, T_1)$, the following will be proved:

(i) Cer_A has been assigned to Alice by the CTA;
(ii) S_{CA} has been generated by the CTA at the instant T_1 and it has not been tampered by Alice or any one else.

(3) The judge will then check whether the hash of Alice's and Bob's supplied original images along with their respective private watermarks and time stamp provided in their watermark certificates matches with CTA's signatures.

For example, in case of Alice, the judge will do the following.

(a) Decrypt $S_C(W_A^{\text{pri}}, H(I), T_1)$ with the CTA's public key to get $H(W_A^{\text{pri}}, H(I), T_1)$:

$$H(W_A^{\text{pri}}, H(I), T_1) = D_{K_C^{\text{pub}}}(S_C(W_A^{\text{pri}}, H(I), T_1)). \quad (25)$$

(b) Using W_A^{pri} and T_1 from Cer_A and the original image I supplied by Alice (step (1)), calculate the hash $H(W_A^{\text{pri}}, H(I), T_1)$. If $H(W_A^{\text{pri}}, H(I), T_1) = H(W_A^{\text{pri}}, H(I), T_1)$, the judge will be convinced that the image I that Alice presented as her original is indeed the image that was used by Alice to obtain the watermark certificate Cer_A.

(4) The judge will also check the visual similarity/correlation between I and I_{Aw}.

(5) Similarly, for Bob, the judge will perform the following steps.

(a) Decrypt $S_C(W_B^{\text{pri}}, H(I_{\text{Aw}}), T_2)$ with CTA's public key to get $H(W_B^{\text{pri}}, H(I_{\text{Aw}}), T_2)$:

$$H(W_B^{\text{pri}}, H(I_{\text{Aw}}), T_2) = D_{K_C^{\text{pub}}}(S_C(W_B^{\text{pri}}, H(I_{\text{Aw}}), T_2)). \quad (26)$$

(b) Using W_B^{pri} and T_2 from Cer_B and the original image $\bar{I}_{\text{Aw}}$ supplied by Bob (step (1)), calculate the hash $H(W_B^{\text{pri}}, H(\bar{I}_{\text{Aw}}), T_2)$. It is interesting to note that Bob obtained Cer_B using I_{Aw}. However, the fake original that Bob has presented in step (1) is $\bar{I}_{\text{Aw}}$. Since $H(I_{\text{Aw}}) \neq H(\bar{I}_{\text{Aw}})$, therefore $H(W_B^{\text{pri}}, H(\bar{I}_{\text{Aw}}), T_2) \neq H(W_B^{\text{pri}}, H(I_{\text{Aw}}), T_2)$. Therefore, Bob's claim for false ownership will not be successful.

To make the scheme invertible, Bob has to tweak the image $\bar{I}_{\text{Aw}}$ to get another image $\widetilde{I}_{\text{Aw}}$ such that

(I) $H(\widetilde{I}_{\text{Aw}}) = H(I_{\text{Aw}})$;

(II) there should be a high visual similarity/correlation between I_{Aw} and $\widetilde{I}_{\text{Aw}}$.

Finding an image $\widetilde{I}_{\text{Aw}}$ which satisfies condition (I) requires breaking the security of the cryptographic hash function. In case if the hash function used is SHA1, there are 2^{160} different combinations from which only one will satisfy condition (I). This is computationally infeasible.

6.3. Watermark removal attack

Consider a scenario in which Bob performs some operation on Alice's watermarked image I_{Aw} to get another image $\widehat{I}$ such that the strength of Alice's watermark W_A^{pri} in $\widehat{I}$ is significantly reduced to an extent that W_A^{pri} is rendered undetected in $\widehat{I}$. Next Bob sends $H(\widehat{I})$ to the CTA along with a request for a genuine watermark. Suppose the CTA sends Bob the watermark certificate:

$$\text{Cer}_B = \{X_B, S_{CB}(X_B, T_2)\},$$
$$X_B = \{W_B^{\text{pri}}, S_C(W_B^{\text{pri}}, H(\widehat{I}), T_2), T_2\}, \quad (27)$$
$$S_{CB}(X_B, T_2) = E_{K_B^{\text{pub}}}(E_{K_C^{\text{pri}}}(H(X_B, T_2))).$$

Again, it is realistic to assume that the time stamp $T_2 > T_1$, since Bob can only obtain $\widehat{I}$ after Alice generates I_{Aw} (using Cer_A with the time stamp T_1). Bob watermarks $\widehat{I}$ with W_B^{pri} to get a watermarked image $\widehat{I}_{Bw}$. If Alice accuses Bob that $\widehat{I}_{Bw}$ originated from I_{Aw}, can Bob prove his fake ownership of $\widehat{I}_{Bw}$? Despite the fact that W_A^{pri} cannot be detected in $\widehat{I}_{Bw}$, we will show a protocol that will prevent Bob from his false claim. For this protocol to work, Alice should have a copy of her watermarked image I_{Aw}. We now outline the main steps of this protocol.

(1) Alice and Bob present their original images I and $\widehat{I}$ along with their respective watermark certificates Cer_A and Cer_B to the judge.

(2) As discussed in step (2) (Section 6.2), the judge will check whether the identities of Alice and Bob are associated with their respective watermark certificates Cer_A and Cer_B. The judge will then check the authenticity and integrity of Cer_A and Cer_B. In addition, the judge will also check whether the hash of Alice's and Bob's supplied original images along with their respective private watermarks and time stamp provided in their watermark certificates matches with the CTA's signatures.

(3) The judge will then check the visual similarity between I, I_{Aw} and $\widehat{I}$, $\widehat{I}_{Bw}$. Since the underlying watermarking scheme is invisible, there should be a very high visual similarity/correlation between the original image and its watermarked version. For a particular claimant, if visual similarity/correlation is not found between the original image and its watermarked version, then that person will not be considered as a candidate for true ownership. It should be noted that for Alice's claim to be true, there should be a high visual similarity/correlation between $\widehat{I}_{Bw}$ and I_{Aw}.

(4) The judge will then check for the presence of Alice's and Bob's private watermarks W_A^{pri} and W_B^{pri} in their respective watermarked images I_{Aw} and $\widehat{I}_{Bw}$.

(5) In our specific illustration, W_A^{pri} will be detected in I_{Aw} with a high correlation. Similarly W_B^{pri} will also be detected in $\widehat{I}_{Bw}$ with a high correlation. The judge will then decide about the actual owner by looking at the time stamp in Alice's and Bob's watermark certificates. Since Alice got Cer_A before Bob, therefore $T_1 < T_2$. As

a result, Alice will be considered the true owner and Bob will fail to prove his false claim of ownership.

7. CONCLUSION

A watermarking algorithm alone cannot completely address the complex issues involved in copyright protection. For a reliable and secure watermarking scheme, it is necessary that the watermarking algorithm being used is well integrated with a secure protocol. In this paper, we have proposed a secure watermarking scheme that is aimed at addressing some of the important issues in copyright protection. Specifically, we have focused on three issues of copyright protection, that is, buyer-seller identification, copyright infringement, and verification of ownership. By buyer-seller identification, we mean that a successful watermark extraction at the buyer's end will reveal the identities of the buyer and seller of a watermarked image. In case of copyright infringement, our proposed scheme will enable the seller to identify the specific buyer from whom an illegal copy of a watermarked image has originated and further prove this fact to a third party. By verification of ownership, we mean that the seller of the watermarked image should be able to prove his or her genuine ownership in case of multiple ownership claims.

In conventional watermarking schemes, the seller of a watermarked image embeds some information, for example, a logo, in a cover image to associate his/her ownership with the image. But the question which arises is that how can we associate some kind of information (i.e., a watermark) with a particular person? To resolve this issue, there has to be some legal binding of the extracted watermark with the identity of the person who claims to be the buyer/seller of the watermarked image. We have addressed this issue by associating the identities of the buyer/seller with their respective public-private key pairs. In a real-world situation, public key infrastructure (PKI) provides a framework in which a public-private key pair is associated with the identity of a person, for example, by issuing digital certificates [14]. With this idea in mind, we have devised a secure watermarking scheme that uses public-private keys and digital signatures to bind the identities of the buyer/seller with the watermarked image. Using public-private keys and digital signatures, our scheme enables a buyer to prove that he/she has legally purchased a watermarked image from a specific seller. In case of copyright infringement, the cryptographic primitives allow a seller to identify the specific buyer from whom an illegal copy has originated and further prove this fact to a third party. Issues pertaining to multiple ownership claims have also been addressed by using watermarks and their respective time-stamped digital signatures issued from a trusted certification authority. The security analysis shown in the appendix demonstrates the security of our scheme against tampering of the original image, the time stamp, or the watermarks. The cryptographic security of the hash function and the RSA public key cryptosystem ensure the security of our proposed watermarking scheme against such attacks.

Although the spread-spectrum technique which we have used for embedding the private watermark is quite robust, however, it is a nonblind technique. For example, in case of copyright infringement and ownership verification, the legal seller is required to produce the original image to resolve the ownership dispute. This may not be desirable in some applications as it potentially leaks the secrecy of the original unwatermarked image. Using a blind watermarking technique for private-watermark extraction seems a good solution. However, issues related to ambiguity in dispute resolving or ownership deadlocks due to weaknesses in watermarking algorithms or the cryptographic protocols have to be thoroughly investigated.

APPENDIX

SECURITY ANALYSIS

In this appendix, we present the security analysis of our proposed scheme against tampering of the watermark certificate or the original image. We introduce an attack called the *time stamp attack* and show that our scheme can resist such an attack. In addition, we will also show that it is computationally infeasible for an attacker to modify or tamper the contents of the watermark certificate or the original image. As discussed in Section 6.3, in case of watermark removal attack, the true seller will be identified by looking at the CTA's time stamp provided in the watermark certificate. Consider again the scenario discussed in Section 6.3. A question that arises is whether it is possible for Bob to watermark Alice's watermarked image I_{Aw} using a watermark obtained from the CTA at time T_0 ($T_0 < T_1$). If Bob somehow manages to launch this attack, he will be considered as the true seller in case of an ownership dispute with Alice. We call this attack as the time stamp attack (TSA), and show that our proposed set of copyright protocols can prevent Bob from launching such an attack. To demonstrate this fact, assume that for some image J that Bob possesses, CTA issues a watermark certificate Cer_{B0} to Bob at time T_0. Let the watermark certificate Cer_{B0} be defined as

$$\mathrm{Cer}_{B0} = \{X_{B0}, S_{CB0}(X_{B0}, T_0)\}, \tag{A.1}$$

$$X_{B0} = \left\{W_{B0}^{\mathrm{pri}}, S_C\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right), T_0\right\}, \tag{A.2}$$

$$S_{CB0}(X_{B0}, T_0) = E_{K_B^{\mathrm{pub}}}\left(E_{K_C^{\mathrm{pri}}}(H(X_{B0}, T_0))\right). \tag{A.3}$$

Suppose, later in time, Alice watermarks her original image I using the private watermark W_A^{pri} obtained from the CTA's watermark certificate Cer_A given by (14) to generate a watermarked image I_{Aw}. Let us assume that Bob gets a copy of Alice's watermarked image I_{Aw}, and some how removes W_A^{pri} from I_{Aw} to get another image $\widehat{I}$. He then watermarks $\widehat{I}$ using the private watermark W_{B0}^{pri} obtained from Cer_{B0} to get a watermarked image I_{Bw0}. Now suppose there is an ownership dispute between Alice and Bob over the image I_{Bw0}. Alice accuses Bob that I_{Bw0} originated from I_{Aw}. The question that arises is whether Bob can prove his fake ownership. The protocol discussed in Section 6.3 can be used to decide about the actual seller of I_{Bw0}. As discussed in Section 6.3,

before checking the presence of Alice's and Bob's private watermarks in the respective watermarked images I_{Aw} and I_{Bw0}, the judge will check the following:

(I) association of Alice and Bob with their respective watermark certificates;
(II) authenticity and integrity of Cer_A and Cer_{B0};
(III) the perceptual similarity/correlation between (I, I_{Aw}), $(\hat{I}, \hat{I}_{Bwo})$, and (I_{Aw} and $\hat{I}_{Bwo}$);
(IV) the authenticity and integrity of the claimant's supplied original image and its respective digital signature by the CTA in the watermark certificate.

Theorem A.1. *To prove his fake ownership of I_{Bw0}, Bob has to find a fake original image P such that the following conditions are satisfied.*

(I) *The image P should be visually similar to I_{Bw0}, that is, the normalized correlation coefficient between P and I_{Bw0} should approach unity.*
(II) *The fake original image P, the watermark, and the time stamp in the watermark certificate* Cer_{B0} *should satisfy the following:*

$$H\left(W_{B0}^{\mathrm{pri}}, H(P), T_0\right) = H\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right). \tag{A.4}$$

Proof. As discussed in Section 6.3, in case of Bob, the judge will check the perceptual similarity/correlation between I_{Bw0} and the image that Bob will present to the judge as his original. Since Bob has obtained his watermark certificate Cer_{B0} before he had the knowledge of I_{Aw}, it is reasonable to assume that the original image J that Bob had used to get Cer_{B0} will be different from I_{Aw}. Therefore, Bob has to fool the judge by presenting a fake image P that is visually similar to I_{Bw0} instead of J. This proves condition (I). We now prove condition (II) by proving (A.4). Since Bob had provided $H(J)$ to the CTA for obtaining Cer_{B0}, therefore the CTA's digital signature will be

$$S_C\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right) = E_{K_C^{\mathrm{pri}}}\left(H\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right)\right). \tag{A.5}$$

During the ownership verification protocol discussed in Section 6.3, the judge will perform the following:

$$H\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right) = D_{K_C^{\mathrm{pub}}}\left(S_C\left(W_{B0}^{\mathrm{pri}}, H(J), T_0\right)\right). \tag{A.6}$$

To keep the CTA's digital signature intact, Bob's fake original P should satisfy (A.4). □

Corollary A.1. *It is computationally infeasible for Bob to find an image P that satisfies* (A.4).

Proof. One way that Bob may use to start finding a fake image P is to make small changes in I_{Bw0} until (A.4) is satisfied. The changes made in I_{Bw0} should be such that P is visually similar to I_{Bw0}. However, for Bob to be successful, he has to try all unique combinations of the hash functions before he can get one that satisfies (A.4). In addition, the changes that Bob makes to get P from I_{Bw0} should still keep a high visual similarity/correlation between P and I_{Bw0}. Finding an image P that satisfies (A.4) requires breaking the security of the cryptographic hash function. In case the hash function used is SHA1, there are 2^{160} different combinations from which only one will satisfy (A.4). This is computationally infeasible to do so. □

Remark A.1. Let us assume a hypothetical scenario in which Bob is successful in finding a fake original P that satisfies (A.4). However, to generate an authentic watermark certificate, Bob will have to forge the digital signature of the CTA to generate the second element of the watermark certificate. This is computationally infeasible as it implies breaking the RSA public key cryptosystem. Therefore, the cryptographic security of the hash function and the RSA cryptosystem ensures the security of our proposed watermarking protocols against TSA and any other attack that involves any modification or fake insertion of the original image (whose hash was presented to the CTA for obtaining the watermark certificate) or the private watermark present in the watermark certificate.

REFERENCES

[1] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*, Morgan Kaufmann, San Francisco, Calif, USA, 2001.
[2] I. J. Cox, J. Kilian, F. T. Leighton, and T. G. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.
[3] T. Furon, I. Venturini, and P. Duhamel, "A unified approach of asymmetric watermarking schemes," in *Security and Watermarking of Multimedia Contents III*, vol. 4314 of *Proceedings of SPIE*, pp. 269–279, San Jose, Calif, USA, January 2001.
[4] T. Furon and P. Duhamel, "An asymmetric watermarking method," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 981–995, 2003.
[5] J. J. Eggers, J. K. Su, and B. Girod, "Public key watermarking by eigenvectors of linear transforms," in *Proceedings of the European Signal Processing Conference (EUSIPCO '00)*, Tampere, Finland, September 2000.
[6] H. Choi, K. Lee, and T. Kim, "Transformed-key asymmetric watermarking system," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 251–254, 2004.
[7] M. L. Miller, "Is asymmetric watermarking necessary or sufficient?" in *Proceedings of the European Signal Processing Conference (EUSIPCO '02)*, Toulouse, France, September 2002.
[8] M. Ramkumar and A. N. Akansu, "A robust protocol for proving ownership of multimedia content," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 469–478, 2004.
[9] S. Katzenbeisser, "On the integration of cryptography and watermarks," in *International Workshop on Digital Watermarking*, vol. 2939 of *Springer Lecture Notes in Computer Science*, pp. 50–60, Seoul, Korea, October 2003.
[10] N. Memon and P. W. Wong, "A buyer-seller watermarking protocol," *IEEE Transactions on Image Processing*, vol. 10, no. 4, pp. 643–649, 2001.
[11] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[12] N. Koblitz, *Algebraic Aspects of Cryptography*, Springer, Berlin, Germany, 1998.
[13] B. Schneier, *Applied Cryptography*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1996.

[14] S. Burnett and S. Paine, *RSA Security's Official Guide to Cryptography*, Osborne/McGraw-Hill, Emeryville, Calif, USA, 2001.
[15] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.
[16] T.-W. Lee, *Independent Component Analysis—Theory and Applications*, Kluwer Academic, Dordrecht, The Netherlands, 1998.
[17] I. T. Jolliffre, *Principal Component Analysis*, Springer, Berlin, Germany, 2002.
[18] S. E. Noel and H. H. Szu, "Multimedia authenticity with ICA watermarks," in *Wavelet Applications VII*, vol. 4056 of *Proceedings of SPIE*, pp. 175–184, Orlando, Fla, USA, April 2000.
[19] D. Yu and F. Sattar, "A new blind watermarking technique based on independent component analysis," in *International Workshop on Digital Watermarking*, vol. 2613 of *Springer Lecture Notes in Computer Science*, pp. 51–63, Seoul, Korea, October 2003.
[20] F. Mintzer and G. W. Braudaway, "If one watermark is good, are more better?" in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2067–2069, Phoenix, Ariz, USA, March 1999.
[21] J.-F. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, no. 1, pp. 157–192, 1999.
[22] S. Carver, N. Memon, B.-L. Yeo, and M. M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 573–586, 1998.
[23] M. Ramkumar and A. N. Akansu, "Image watermarks and counterfeit attacks: some problems and solutions," in *Proceedings of Content Security and Data Hiding in Digital Media*, Newark, NJ, USA, May 1999.

Fawad Ahmed received the B.E. degree in industrial electronics from the Institute of Industrial Electronics Engineering, NED University of Engineering and Technology, Karachi, Pakistan, in 1995, and the M.Eng.S. degree in electrical engineering from the University of New South Wales, Sydney, Australia, in 1998. He is currently pursuing the Ph.D. degree at the Nanyang Technological University, Singapore. From 1998 to 2002, he was a lecturer at the Department of Electronics and Power Engineering, Pakistan Navy Engineering College, Karachi. His research interests include digital watermarking, image authentication using robust hashing, biometrics, and cryptography.

Farook Sattar has received his Technical Licentiate and Ph.D. degrees in signal and image processing from Lund University, Sweden. He is currently an Assistant Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His current research interests include digital watermarking, blind source separation, speech/audio segmentation, and image enhancement. He had been involved in a number of signal-processing-related projects sponsored by Swedish National Science and Technology Board (NUTEK) and Singapore Academic Research Funding (AcRF) Scheme. His research has been published in a number of leading journals and conferences.

Mohammed Yakoob Siyal got his B.E. degree from Mehran University in electronic engineering, his M.S. and Ph.D. degrees from the University of Manchester Institute of Science and Technology (UMIST), England, in computer engineering, and an M.B.A. degree from Surrey European Management School (SEMS), Surrey University, England, in IT. Before joining Nanyang Technological University, Singapore, in early 1993, he was with the University of Newcastle upon Tyne, England. He is a Chartered Engineer in England, a Member of the Institute of Electrical Engineers, UK, and a Senior Member of IEEE, USA. Currently he is a Tenured Associate Professor of information engineering at Nanyang Technological University, Singapore.

Dan Yu received her B.Eng. (first-class honors) in communication engineering and her Ph.D. degree in information security and multimedia signal processing from Nanyang Technological University, Singapore, in 2000 and 2005, respectively. She has been involved in various research projects in digital watermarking, information hiding, license plate recognition, and video surveillance system. She is currently holding a postdoctoral research position in Institut National de Recherche en Informatique et en Automatique (INRIA), France, with special interests in texture modeling for image segmentation, classification, and high-level understanding for image retrieval.