RC 21885 (98464) 17 November 2000 **Electrical Engineering**

IBM Research Report

Limitations and Requirements of Content-Based Multimedia Authentication Systems

Chai Wah Wu
IBM T. J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - T. J. Watson - Tokyo - Zurich

Limitations and requirements of content-based multimedia authentication systems

Chai Wah Wu

IBM Research Division, Thomas J. Watson Research Center
P. O. Box 218, Yorktown Heights, NY 10598, U.S.A.

ABSTRACT

Recently, a number of authentication schemes have been proposed for multimedia data such as images and sound data. They include both label based systems and semifragile watermarks. The main requirement for such authentication systems is that minor modifications such as lossy compression which do not alter the content of the data preserve the authenticity of the data, whereas modifications which do modify the content render the data not authentic. These schemes can be classified into two main classes depending on the model of image authentication they are based on. One of the purposes of this paper is to look at some of the advantages and disadvantages of these image authentication schemes and their relationship with fundamental limitations of the underlying model of image authentication. In particular, we study feature-based algorithms which generate an authentication tag based on some inherent features in the image such as the location of edges. The main disadvantage of most proposed feature-based algorithms is that similar images generate similar features, and therefore it is possible for a forger to generate dissimilar images that have the same features. On the other hand, the class of hash-based algorithms utilizes a cryptographic hash function or a digital signature scheme to reduce the data and generate an authentication tag. It inherits the security of digital signatures to thwart forgery attacks. The main disadvantage of hash-based algorithms is that the image needs to be modified in order to be made authenticatable. The amount of modification is on the order of the noise the image can tolerate before it is rendered inauthentic.

The other purpose of this paper is to propose a multimedia authentication scheme which combines some of the best features of both classes of algorithms. The proposed scheme utilizes cryptographic hash functions and digital signature schemes and the data does not need to be modified in order to be made authenticatable. Several applications including the authentication of images on CD-ROM and handwritten documents will be discussed.

Keywords: Data authentication, cryptography, digital signatures, hash functions, digital watermarks, digital notary.

1. INTRODUCTION

Several authentication schemes have been proposed in the last few years for authenticating multimedia data such as images and sound data. These schemes include both label based systems[1] and semifragile watermarks.[2] Authenticating images and multimedia content in general differs from the traditional problem of authentication in cryptography. The goal in image authentication is to authenticate the content and not the specific representation of the image. As a result, a requirement of such authentication systems is that minor modifications such as lossy compression which do not alter the content of the data preserve the authenticity of the data, whereas modifications which do modify the content render the data inauthentic. This requirement is difficult to formalize as the notion of content is difficult to specify precisely. Furthermore, as images can be considered as points in a continuous space, there is not a sharp boundary between authentic and inauthentic data since a sharp boundary would mean that there are authentic and inauthentic images which are similar to each other. More realistically is the following picture, where the region of surely authentic images is separated from the surely inauthentic images by a fuzzy region where the authenticity of the images is difficult to ascertain (Fig. 1). In the figure, these regions are illustrated as spheres in some suitable metric space to facilitate characterization, although in general they can have more complicated forms. Thus there are three answers when authenticating an image: authentic, inauthentic and don't know. We will mainly discuss multimedia authentication systems which either answer authentic or inauthentic when given a dataset to authenticate.

Email: chaiwah@watson.ibm.com

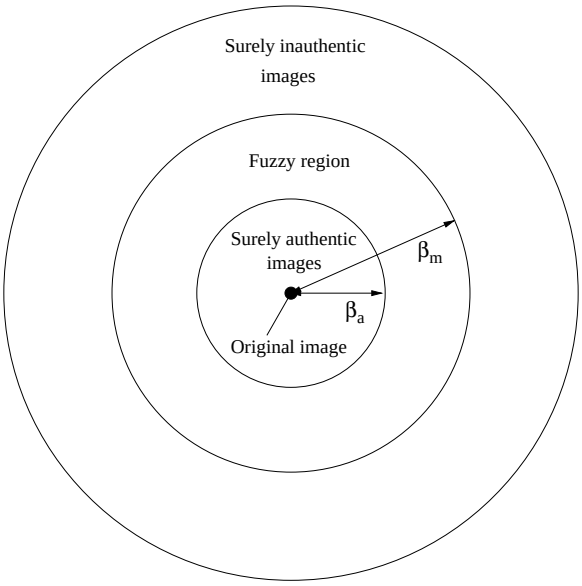

Figure 1. Diagram illustrating how the original image is surrounded by a set of images which are surely to be authentic and separated from the set of surely inauthentic images by a fuzzy region where the authenticity of the image is uncertain. β_a and β_m indicate radii in the case these regions are spheres in the underlying space.

The image authentication schemes in the literature can be classified into two main classes depending on the model of image authentication they are based on. One of the purposes of this paper is to look at some of the advantages and disadvantages of these schemes and their relationship with fundamental limitations and requirements of the underlying model of image authentication. The class of feature-based algorithms generate an authentication tag based on some inherent features in the image which changes smoothly with the image and is discussed in Section 3. The main disadvantage of feature-based algorithms is that since similar images generate similar features, it is possible for a forger to generate dissimilar images which have the same features. In cryptography, this lack of diffusion is generally avoided in constructing secure encryption and signature schemes. The class of hash-based algorithms utilize a cryptographic hash function to reduce the data and to generate the authentication tag and is discussed in Section 4. This class of algorithms inherits the security of cryptographic hash functions and digital signatures to thwart forgery attacks. The main disadvantage of hash-based algorithms is that the image needs to be modified in order to be made authenticatable. The amount of modification is on the order of the noise the image can tolerate before it is rendered inauthentic. We propose in Section 5 a multimedia authentication scheme which addresses these disadvantages and aims to combine the best features of both classes of algorithms. Section 6 gives a specific implementation of this scheme for authenticating images and Section 7 extends the algorithm to include authenticability distortion. Finally, in Section 8 we present examples and discuss applications such as the digital notary where this algorithm can be used.

2. IMAGE AUTHENTICATION SCHEMES

Almost all image authentication schemes in the literature have the following form. Given a source image I, an authentication tag T is derived from I. Generally *data reduction* occurs in generating the tag, resulting in an authentication tag that is much smaller than the source image I. The source data I is then processed to I' and the authentication tag is then inserted or appended to I' resulting in authentication data I''. If the authentication tag is appended onto I', as is the case in label-based approaches, then the image portion of I'' is equal to I' and we denote this as $I'' = (I', T)$. If the authentication tag is inserted into I' via a data hiding algorithm, as is the case in semifragile watermarks, then we denote this as $I'' = I' \circ T$. The source image I is changed to I'' in order to make it authenticatable. The *authenticability distortion* is defined as the change that is necessarily made to the image in order to make it authenticatable. It is the difference between I' and I in the case $I'' = (I', T)$ and it is the difference between I'' and I in the case $I'' = I' \circ T$. We will mainly study the case $I'' = (I', T)$ (i.e. label-based schemes) in this paper. I'' will be referred to as an *authenticatable image*. To authenticate the authenticatable image (I_a, T_a)

(or $I_a \circ T_a$), the tag T_a is extracted from the image, and a second tag T'_a is computed from I_a. The two tags T_a and T'_a are then compared. If they compare favorably, the image is considered authentic, otherwise it is considered inauthentic. In order to facilitate analysis, we make the assumption that the images I, I' and I_a are expressed in suitable continuous normed spaces such as $\mathcal{R}^n$. The comparison of the two tags is done by a binary relation R: $R(T_a, T'_a)$ is true if and only if the image is considered authentic. A flow diagram of this general model of image authentication is shown in Figure 2. Figure 2a shows the steps of generating an authenticatable image I'' whereas Figure 2b shows the steps of authenticating an authenticatable image I''.

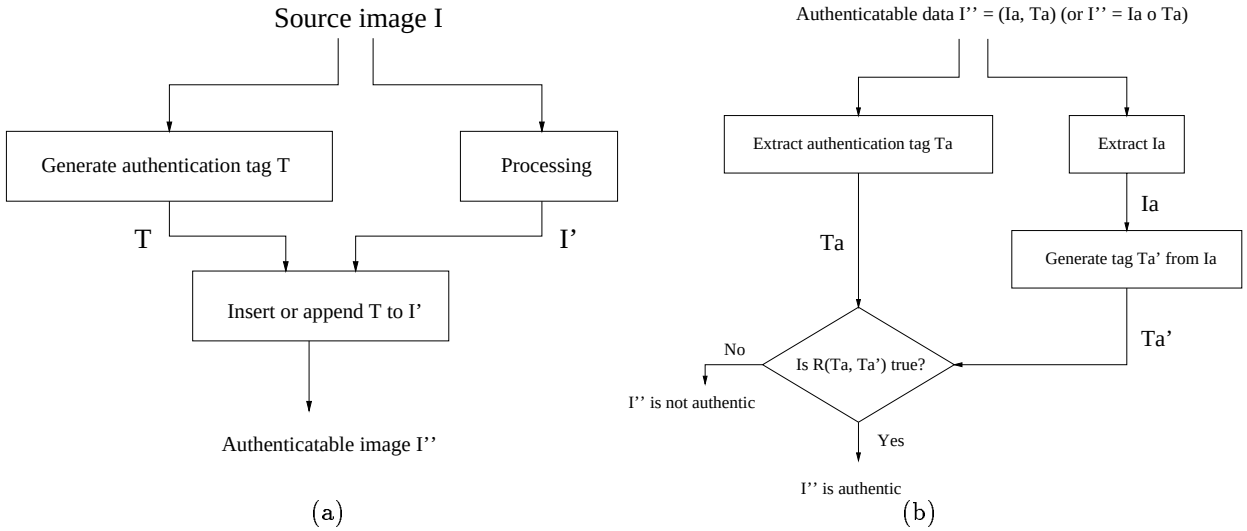

Figure 2. Flow diagram of a general image authentication system. (a) generation of authenticatable data. (b) authentication of authenticatable data.

We use the following parameters to characterize the performance of a multimedia authentication scheme:

- The maximum authenticability distortion is D.
- The size of the authentication tag as a percentage of the source data is α.
- The parameters $\beta_a \leq \beta_m < \infty$ are defined as follows: If $|x| \leq \beta_a$, then the authentication algorithm considers $(I' + x, T)$ as authentic. If $|x| \geq \beta_m$, then the authentication algorithm considers $(I' + x, T)$ as inauthentic. These parameters are illustrated in Figure 1.
- The parameter $\Delta\beta$ is defined as $\beta_m - \beta_a$.

An objective of multimedia authentication system design is to minimize D, α and $\Delta\beta$ as they trade off against each other. For instance, reference[3] presents a scheme where $D = \beta_a = \beta_m$ whereas in reference[1] $D = 0$. We discuss some applications in Section 8 where it is imperative that D is zero or small. In Section 5 we present a novel multimedia authentication scheme which allows the parameters D, α and $\Delta\beta$ to be traded off against each other.

3. EXTRACT FEATURES AND CHECK FOR SIMILARITY

In this model, data reduction is performed by extracting some relevant features of the image and the two tags are compared for similarity.[4,5,1] Generally the similarity relation is of the form: $R(T1, T2)$ is true if and only if $d(T1, T2) < \epsilon$ for some metric d. In this model, the data reduction that is performed invariably results in very different images generating similar tags. In other words, there will always exist forged images, i.e. images which are far away from the original, but generate similar tags and therefore are considered authentic. This is seen as follows:

Suppose that the images lies in the n-dimensional unit cube $[0,1]^n$ and the tags lies in the m-dimensional unit cube $[0,1]^m$. Data reduction means that m is much smaller than n. Let us denote the map which generates a tag from an image as $F : [0,1]^n \to [0,1]^m$. We have the following result:

THEOREM 3.1. *For each function $F : [0,1]^n \to [0,1]^m$ where $n > m$ and for each $\delta > 0$, k a nonzero integer, there exists x, y such that $d(x,y) > k^{-\frac{m}{n}} - \delta$ and $d(F(x),F(y)) \leq \frac{1}{k}$ where $d(x,y) = \max_i(x_i - y_i)$ is the L_∞ norm.*

Since $k^{-\frac{m}{n}}$ approaches 1 for $m << n$, this theorem implies that for n much larger than m, there will be forged images which are very different from the original image, but generate similar tags.

Proof. Partition $[0,1]^m$ into k^m cubes which are of length $\frac{1}{k}$ on each side. We will call these cubes $\{C_i\}$. The preimages of $\{C_i\}$ are nonoverlapping sets. To reach a contradiction assume that the diameter of these preimages are all less than or equal to $\mu = k^{-\frac{m}{n}} - \delta$. Thus each $F^{-1}(C_i)$ is enclosed in a cube of side μ. Since there are only k^m of them, the volume of these cubes will be less than $k^m(k^{-\frac{m}{n}} - \delta)^n < 1$ and thus they will not cover $[0,1]^n$. Pick a point x not in these cubes. x will be mapped by F into some cube C_i which is a contradiction since x is not in $F^{-1}(C_i)$. □

Note that the theorem is false for $k = \infty$. This is because $[0,1]^n$ and $[0,1]^m$ have the same cardinality and thus there exists a one-to-one map between $[0,1]^n$ and $[0,1]^m$.

Theorem 3.1 is also true for $F : D \to [0,1]^m$, as long as D is a dense enough subset of $[0,1]^n$. In particular, we have:

THEOREM 3.2. *Consider a function $F : D \to [0,1]^m$ where $D \subset [0,1]^n$ and $n > m$, and $\sup_{x \in [0,1]^n} d(x,D) < \gamma$ and k a nonzero integer. Then there exists x, y such that $d(x,y) > k^{-\frac{m}{n}}(1-\gamma)$ and $d(F(x),F(y)) \leq \frac{1}{k}$ where $d(x,y) = \max_i(x_i - y_i)$.*

Proof. Essentially the same as Theorem 3.1, except that we now choose $\delta = k^{-\frac{m}{n}}\gamma$. The volume of $[0,1]^n$ not covered by the cubes is more than $1 - k^m(k^{-\frac{m}{n}} - \delta)^n \geq \gamma^n$, which means that there exists a point of D in this set, which completes the proof. □

There are two disadvantages to this model. First of all, since the similarity of the tags are used to indicates similarity between the images they correspond to, there is a smoothness in generating the tags, i.e. small changes in the image results in small changes in the tag. This lack of diffusion may allow forged images to be constructed easily which have the same tag as the original image. In the language of cryptography, the function which generates the tag from the original image may not be preimage resistant.[6] Other examples of this problem are discussed in reference.[3] Secondly, the similarity relation on the tags with the smoothness condition in generating the tag makes it difficult to define exactly the amount of modifications the source image can be tolerated and still be considered authentic, i.e. concrete values for β_a and β_m are hard to determine.

4. HASH AND CHECK FOR EQUALITY

In this model, I' is hashed by a cryptographic hash function and the hash is used as the authentication tag. Since small changes in the images result in large changes in the hash, the two tags are compared for equality to determine authenticity, i.e. R is the equality relation. Examples of such image authentication schemes are given in reference.[3]

A generalization for this type of authentication systems is as follows: the authenticability distortion is applied to image I resulting in $I' = f(I)$, and the tag T is generated from I' as $T = g(I')$. This is illustrated in Figure 3. The authentication steps are the same as in Figure 2b. Note that this class of authentication systems is large enough to encompass the schemes described in Section 3.

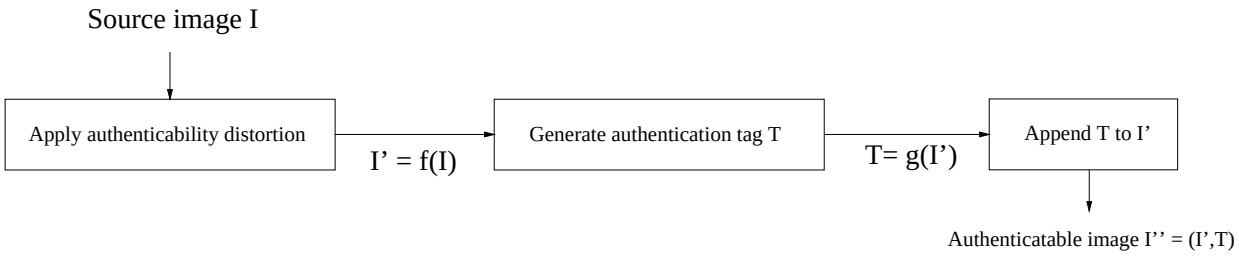

Figure 3. Flow diagram of the generation of authenticatable data in a hash-and-check-for-equality type image authentication system. The authentication step is the same as in Figure 2b.

With the model as described above, we have the following result:

THEOREM 4.1. *If R is transitive* then $D \geq \beta_a$. If particular, there exists an image I such that $\|f(I) - I\| \geq \beta_a$.*

Proof. Proof by contradiction. Suppose $D < \beta_a$. Consider an image I. Pick μ such that $\|\mu\| = \beta_a$. Define recursively $I_1 = f(I)$, $I_2 = f(I_1 + \mu)$, $I_n = f(I_{n-1} + \mu)$, etc. Since $R(g(I + \delta), g(I))$ is true for all δ such that $\|\delta\| \leq \beta_a$, this implies that $R(g(I_n + \mu), g(I_n))$ is true. Since the maximum authenticability distortion D is defined as $\sup_I \|I - f(I)\|$, $\|I_n - I_{n-1} - \mu\| \leq D < \beta_a$, and thus $R(g(I_n), g(I_{n-1} + \mu))$ is also true. Transitivity of R implies that $R(g(I_j), g(I_k))$ for all j and k.

By induction we get that $\|I_n - I_1 - (n-1)\mu\| \leq (n-1)D$. This implies that $\|I_n - I_1\| \geq \|(n-1)\mu\| - \|(n-1)\mu - (I_n - I_1)\| \geq (n-1)(\beta_a - D)$ which diverges as $n \to \infty$. This along with the fact that $R(g(I_n), g(I_1))$ is true contradicts the fact that $\beta_m < \infty$. □

In particular, if $\beta_a > 0$, then $D > 0$. In other words, if we want an authentication system of this class to tolerate minor modifications, we need to change the image in order to make it authenticatable. Furthermore, the amount of modification the system is willing to tolerate is less than the maximum authenticability distortion. This is one of the main disadvantages of this type of image authentication systems. In reference,[3] the two tags are tested for equality which is a transitive relation and therefore the image is changed to enable authentication. In fact, for this case, $D = \beta_a = \beta_m$. On the other hand, an advantage of the scheme in reference[3] is that the parameters β_a and β_m can be set explicitly and $\Delta\beta = 0$.

In reference[1] (Sect. 3) the relation is not transitive and therefore we can have $D = 0$. There are several reasons why a small or a zero D is desirable. For instance, a large D is a disadvantage for applications where the authenticatable images are of high quality, but lower quality images are still acceptable for authentication. For example, the images are printed as high-quality images, but lower quality scans are submitted for authentication. Furthermore, depending on the space where the feature vector is located, $D \geq \beta_a$ can distort the source image too much. For instance, in reference,[3] the image pixels are quantized with a quantization step of k to ensure that a change of up to $\frac{k}{2}$ at each pixel is tolerated. Consider the grayscale Lena image shown in Fig. 4a, denoted as I_1. We call I_2 the Lena image after JPEG compression. In particular, I_2 is obtained by compressing I_1 using JPEG with quality factor 5 (medium quality) in Adobe Photoshop 5.0. The maximum difference between the pixel values ofI_1 and I_2 is 29 and therefore a quantization step of 58 will ensure that both I_1 and I_2 quantize to the same image. However I_1 quantized with a quantization step of 58 result in posterization as shown in Fig. 4b and is not acceptable as authenticatable data.

(a) (b)

Figure 4. (a) Lena image. (b) Lena image after quantization with step 58. Posterization and contouring appear due to the quantization.

*i.e. if $R(a, b)$ and $R(b, c)$ are true then so is $R(a, c)$.

Other applications require that the authenticability distortion D be zero. For instance, when the image resides on a recordable CD-ROM (CD-R) which can not be rewritten, and the authentication tag is added afterwards. In Sect. 8 we present another application where $D = 0$ is a requirement.

5. A CRYPTOGRAPHY BASED LOW DISTORTION MULTIMEDIA AUTHENTICATION SYSTEM

In this section we introduce an authentication system for multimedia data which do not have the disadvantages of the two types of image authentication systems discussed earlier. In particular, the system utilizes digital signatures and cryptographic hash functions to thwart forgery attacks while at the same time D can be much smaller (even zero) than β_a. Furthermore, β_a and β_m can be explicitly determined.

The general scheme is shown in block diagram form in Figure 5. A set of quantization functions $\{q_j\}$ is chosen a priori for this scheme.

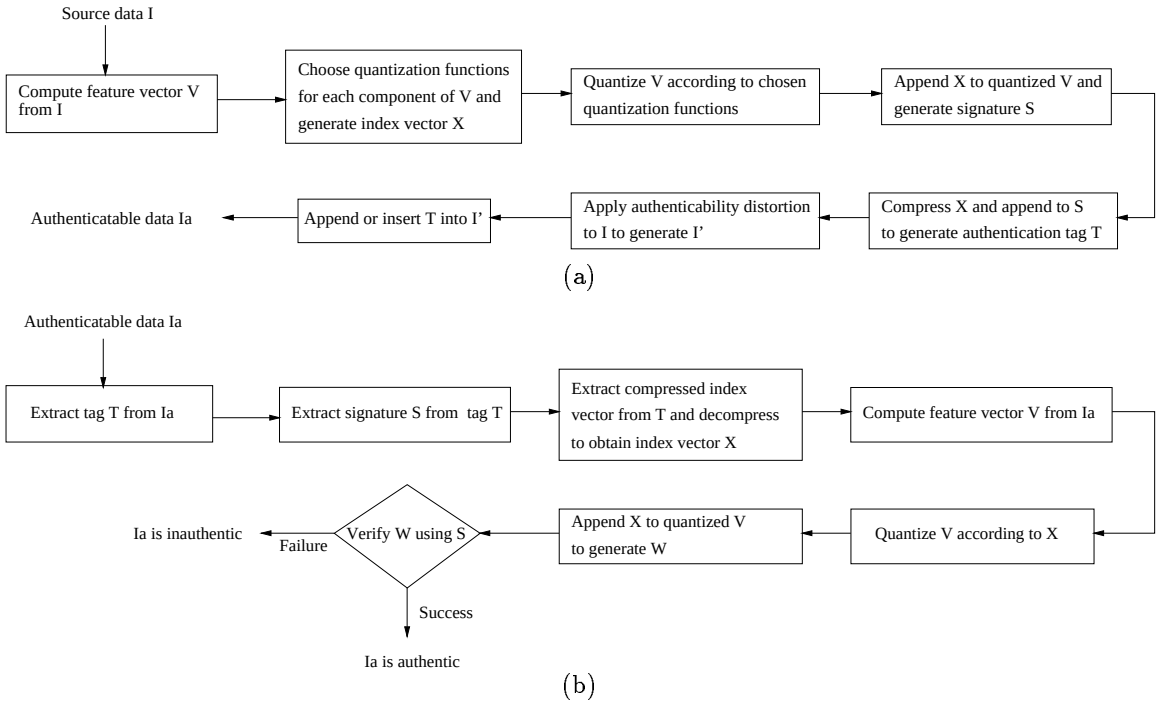

Figure 5. Flow diagram of multimedia authentication scheme utilizing cryptographic hash functions with low authenticability distortion D. (a) generation of authenticatable data (b) authentication of authenticatable data.

Figure 5a shows the process of generation of authenticatable data. First, a feature vector $V = v(I)$ is computed from I. Preferably, v should be smooth. Furthermore, v should be invertible or nearly invertible to avoid the problems associated with the methods in Section 3. In other words, we want v to be pre-image resistant, which it is if it is invertible. For each component V_i of the n-dimensional vector V, a quantization function q_{j_i} is chosen. q_{j_i} is chosen such that the quantization of V_i using q_{j_i} result in a small amount (or the least amount) of quantization error[†]. The indices j_i are stored in the index vector X. The feature vector V is quantized according to q_{j_i} and the index vector X is appended to the quantized V resulting in W. A digital signature algorithm is used to sign W resulting in a digital signature S. X is then compressed with a lossless compression algorithm and appended to S resulting in an authentication tag T. Authenticability distortion is applied to I resulting in I'. T is appended or inserted into I' resulting in authenticatable data I_a. Besides the indices j_i, the vector X can contain other information about the data such as the date, owner, recipient, quantization tables, etc.

Figure 5b shows the authentication process. To authenticate I_a, an authentication tag T is first extracted from I_a. The signature S is removed from T and what remains of T is the compressed index vector. This is then decompressed

[†]The quantization error for V_i is defined as $V_i - q_{j_i}(V_i)$.

to obtain the index vector X. Next a feature vector V is derived from I_a. Using the indices j_i in X, the components of V are quantized using q_{j_i}. X is appended to the quantized V resulting in W. W is then verified using the corresponding signature verification algorithm against the signature S. If the signature S verifies with W, the data is authentic. Otherwise, it is not authentic.

Let us introduce at this point some terminology regarding quantization functions. The range of a quantization function is called the *quantized values*. For each element x of the set of quantized values of q, the set $q^{-1}(x)$ is called the *quantization region* of x for q. This is illustrated in Figure 6.

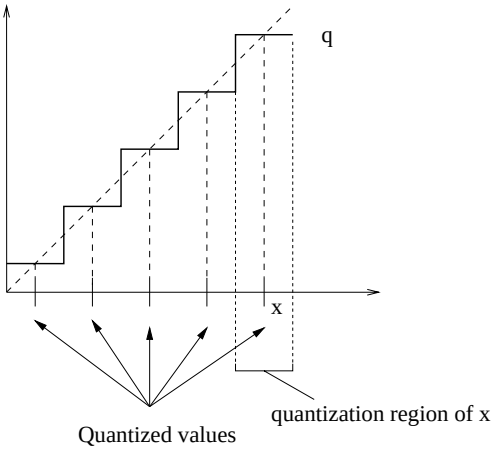

Figure 6. A quantization function q. The quantized values is the range of the quantization function and the set of points which maps to x is the quantization region of x.

The above description is quite general and several design requirements are needed in practical implementations. By suitably selecting the set of quantization functions q_j, for each V_i the corresponding q_{j_i} is such that V_i is not close to the boundaries of the quantization regions. In other words, small changes in V_i does not affect the value of $q_{j_i}(V_i)$. This allows the authenticability distortion D to be zero while $\beta_a > 0$. A choice of such q_j is given in Section 6.

For example, assume that V is a 2-dimensional vector in the plane. Consider two quantization function q_α and q_β whose quantization regions partition the plane as shown in Fig. 7 by the solid and dashed lines respectively. For point A, we should choose q_β since it is close to the boundary of the quantization regions of q_α. Thus q_β maps small changes to A to the same quantized value as A whereas q_α can map small changes to A to a different quantized value than $q_\alpha(A)$. The opposite is true for point B. The point C is close to the boundaries of the quantization regions of both q_α and q_β. Therefore neither quantization function is a good choice for this point and additional quantization functions are needed.

The choice of the feature vector should be such that the feature vector lies in a space where the distance corresponds to perceptual differences or malicious changes. One such space for images is properly scaled DCT space (Sect. 6).

There are three main features of this multimedia authentication scheme. First, unlike the authentication schemes in Section 4, D can be much smaller than β_a. In particular, it can be zero. Second, the parameters β_a and β_m can be explicitly determined. The parameters β_a and β_m describe where the scheme operates properly. If the distortion has norm less than β_a, the system determines the distorted image as authentic. If a forger creates a forged images with a distortion with norm larger than β_m, the systems determines the image to be inauthentic, with the same security as the underlying digital signature scheme. The third main feature is that it uses cryptographic hash functions and digital signature schemes to combat forgery attacks. It is easy to see that finding forged data which are β_m apart from the original data is at least as hard as forging the signature of the underlying digital signature scheme.

There are two places in this scheme where data reduction takes place; in the quantization step and in the digital signature generation step. The data reduction in the quantization step is to ensure that small changes will not destroy authenticity whereas the signature generation step is used to generate a small authentication tag and cryptography is used to prevent forgery attacks.

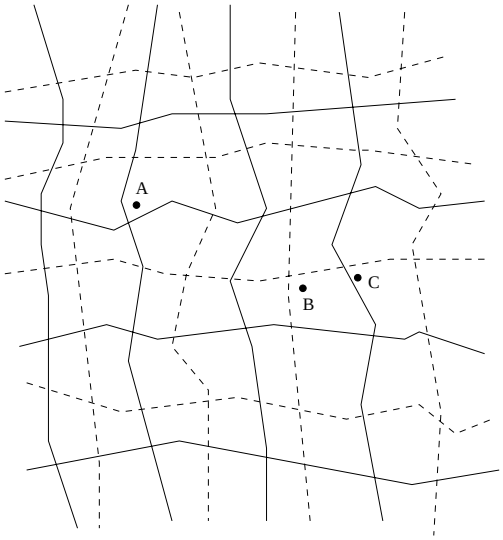

Figure 7. Example of quantization functions on the plane. The quantization regions of the two quantization function q_α and q_β partition the plane by the solid and dashed lines respectively.

6. A SPECIFIC IMPLEMENTATION OF THE PROPOSED AUTHENTICATION SCHEME

In this section, we illustrate the authentication scheme in Section 5 with a more concrete algorithm for images. The feature space we use will be the scaled DCT space as used in the JPEG compression algorithm. Consider the image authentication system in Figure 8.

Figure 8a shows the process of generation of an authenticatable image. We will first consider the case of a grayscale image. The image is separated into 8x8 pixel blocks. For the case where the image cannot be partitioned into 8x8 pixel blocks, rows and columns of zeros are added to the image. Another possibility is to reflect some rows and columns of pixels. For each block a 2-dimensional Discrete Cosine Transform (DCT) is applied. Each DCT coefficient is then scaled by dividing it by a corresponding scaling value[‡]. Next, for each of the resulting scaled DCT coefficient, one of two quantization functions is chosen. The two quantization functions, denoted by q_0 and q_1, are shown in Figure 9.

q_0 and q_1 can be expressed as $q_0(x) = \text{round}(x)$ and $q_1(x) = \text{round}(x + 0.5) - 0.5$ where $\text{round}(x)$ is the integer closest to x. The quantized values of q_0 are the integers $\{..., 0, 1, 2, 3, ...\} = Q_0$ while the quantized values of q_1 is $\{..., 0.5, 1.5, 2.5, 3.5, ...\} = Q_1 = Q_0 + 0.5$. The quantization regions of q_0 and q_1 are of the form $[a - 0.5, a + 0.5)$ and $[a, a + 1)$ respectively for a an integer. The quantization function q chosen for each DCT coefficient is the one which minimizes the quantization error, i.e. if x is the DCT coefficient, then choose q such that $|q(x) - x|$ is minimal. Another way to choose the quantization function is to choose q_t where $t = \arg\min_{i \in \{0,1\}} d(Q_i, x)$ and $d(Q_i, x)$ denotes the distance from x to the set Q_i in the space of real numbers[§]. For the set of quantization functions q_0 and q_1 described above, these two methods give the same result.

For each DCT coefficient, a single bit of the index vector X is assigned to determine which of the two quantization functions is chosen, i.e. a "0" bit is assigned if q_0 is chosen and a "1" bit is assigned if q_1 is chosen. As these bits form the index vector X, there are as many bits in X as there are pixels in the image. For the case of a color image, the feature vector V is derived from the DCT coefficients of 8 by 8 blocks in all the three (or more) color planes. In this case, the number of bits in X is three times the number of pixels in the image. The DCT coefficients are quantized according to the chosen quantization functions. X is then appended to the quantized DCT coefficients to form W. W is then signed by a digital signature algorithm such as DSA (digital signature algorithm)[6] resulting in

[‡]e.g. from a quantization table such as Tables 4-1 and 4-2 in reference.[7]

[§]In case of a tie, a quantization function is randomly chosen.

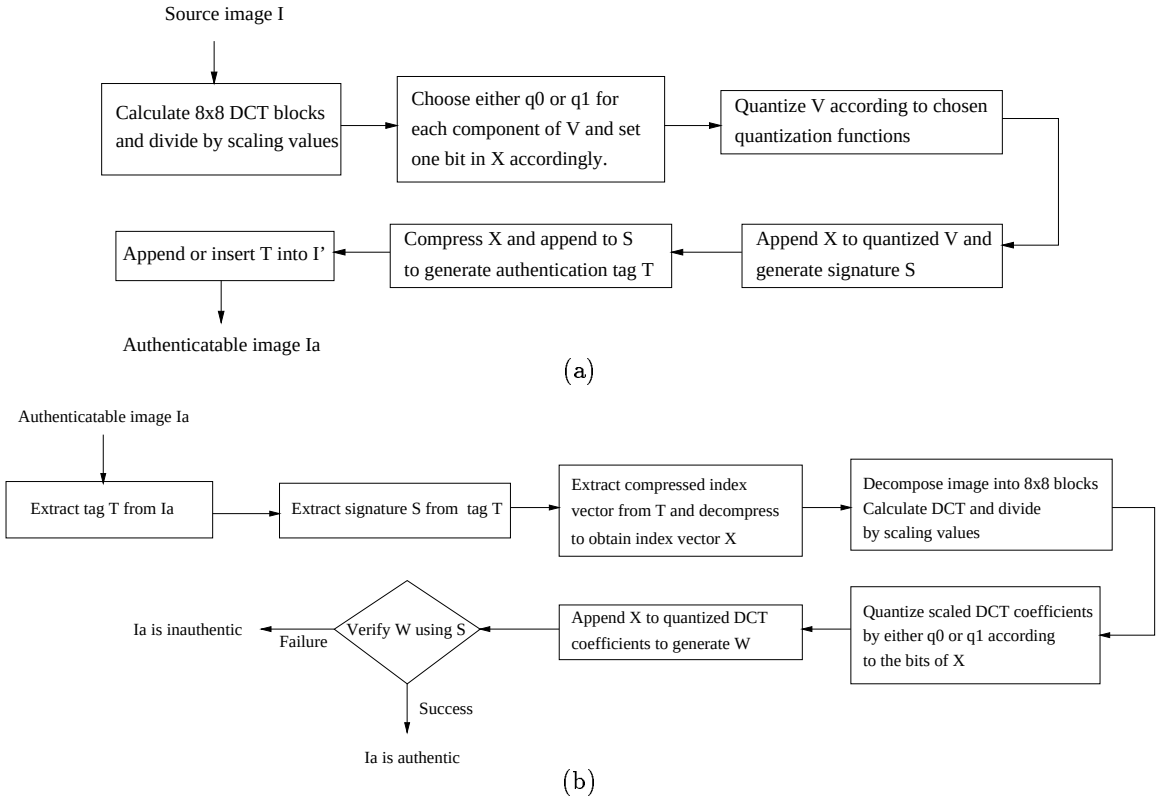

Figure 8. Flow diagram of a specific implemen tation of proposed authentication scheme for images. (a) generation of authenticatable image. (b) authentication of image.

a signature S[¶]. Practical digital signature algorithms typically include a cryptographic hash function to reduce the data and generate a relatively small signature.

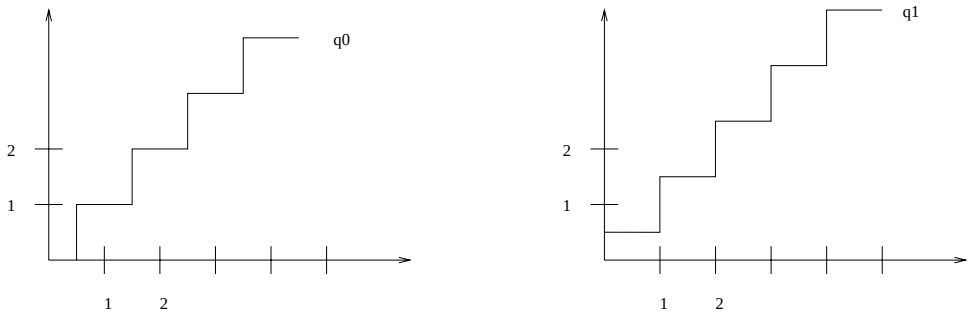

Figure 9. Quantization functions q_0 and q_1.

The index vector X is then compressed using a lossless compression algorithm suc h as Huffman encoding or LZW encoding.[8] To facilitate compression of X, the bits which form X are ordered as follows. Consider the zig-zag ordering of the DCT coefficients in each block as described in Figure 10-5 in reference.[7] First the bits corresponding to the first DCT coefficient in each block are collected, then follows the bits corresponding to the second DCT coefficient in each block, etc. This is illustrated in Figure 10a. In Figure 10a, a_j is the j-th DCT coefficients of block a and xa_j is the bit corresponding to the quantization function chosen for DCT coefficient a_j. If a color image is considered, first the bits corresponding to the first 8 DCT coefficients of the first color dimension (i.e. R

[¶]For the purpose of generating S, $q_1'(x) = \text{round}(x + 0.5)$ can be used instead of $q_1(x)$ in computing the quantized DCT coefficients. This ensures that the quantized DCT coefficients are integers.

in RGB space, L in LAB space) in each block are collected, then followed by the bits corresponding to the first 8 DCT coefficients of the second color dimension in each block, etc. This is illustrated in 10b. a_{ij} is the j-th DCT coefficients of block a of color plane i and xa_{ij} is the bit corresponding to the quantization function chosen for DCT coefficient a_{ij}.

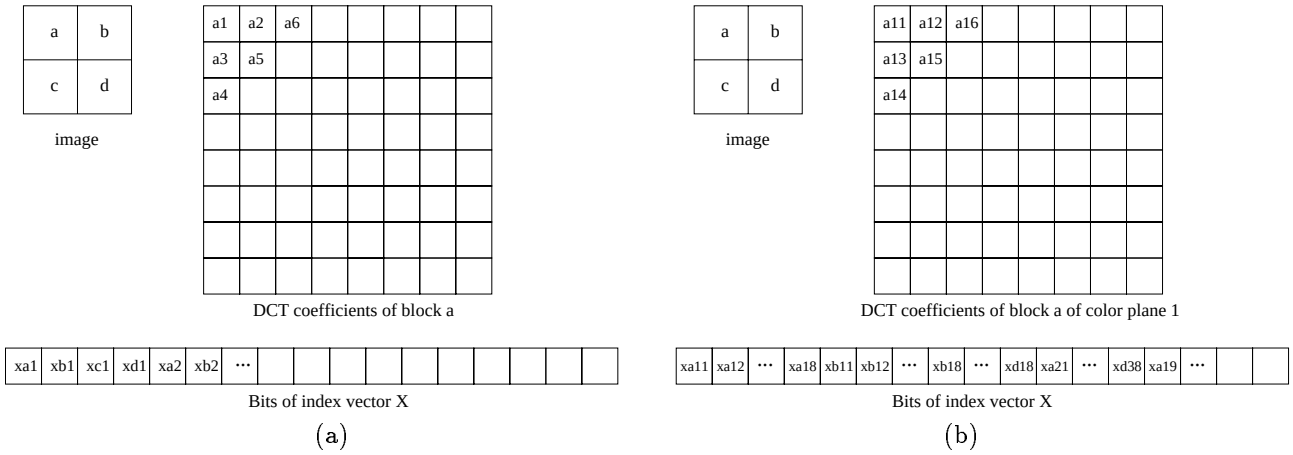

Figure 10. Ordering of the bits in index vector X for an image with 4 blocks. (a) grayscale image. (b) color image.

Since we know exactly how many bits are in X (e.g. it equals the number of 8x8 blocks in a grayscale image), we can remove the trailing zeros in X before compression. In the authentication phase, again we know how many bits are in X, so X is retrieved by decompression and adding the right amount of trailing zeros.

The compressed form of X is appended to the signature S to form an authentication tag T. This authentication tag is then appended onto or inserted into the image I. The tag T can be appended into I by writing it into the comment field of the image format. Image formats which support such fields include JPEG and TIFF. For instance, the tag T can be appended into I by writing T into the COM (Comment) marker segment or the ImageDescription Tag when the JPEG image format or the TIFF image format are used respectively. The tag T can be inserted into I by means of a robust data hiding scheme.[9,10] The robust data hiding schemes should be robust enough such that the tag T can be recovered from the image exactly even under minor modifications to the image.

Figure 8b shows the authentication process. To authenticate an authenticatable image, the authentication tag T is extracted. After the signature S is removed from T, the remainder of T forms the compressed index vector. This is decompressed and trailing zeros are added to obtain X. The image is then decomposed into 8x8 blocks and a DCT operation is applied to each block and scaled by dividing the DCT coefficients by scaling values. The scaled DCT coefficients are then quantized according to the quantization functions given by the bits in X. X is then appended to the resulting quantized DCT coefficients and the result is verified with the signature S by the corresponding signature verification algorithm. If it is verified, the image is authentic. Otherwise, the image is not authentic. The use of a digital signature algorithm can be replaced with message authentication codes or modification detection codes, depending on the type of application.[6]

7. EXTENSIONS TO $D > 0$

So far the implementation described in Section 6 does not introduce any authenticability distortion ($D = 0$). Furthermore, β_a and β_m can be defined to satisfy $\beta_m = 3\beta_a$ using the L_∞ norm. In this section we show how authenticability distortion can be added in two ways. In the first method, authenticability distortion is added to reduce the size of the authentication tag and therefore reduce α. In the second method, authenticability distortion is added to increase the tolerance to distortion (i.e. β_a is larger) and thereby reduce $\Delta\beta$. Referring back to Figure 5, these extensions are applicable when the generation of the feature vector is an invertible process, i.e. $V = v(I)$ is invertible. The modified scheme is shown in Fig. 11.

In the first method, the dataset is distorted as follows. Without loss of generality, assume that for the given feature vector V, the number of zeros in the bits of X is larger than the number of ones. For the components x_i of the feature vector V which are closer towards the quantized values of q_1 than to those of q_0, they are moved

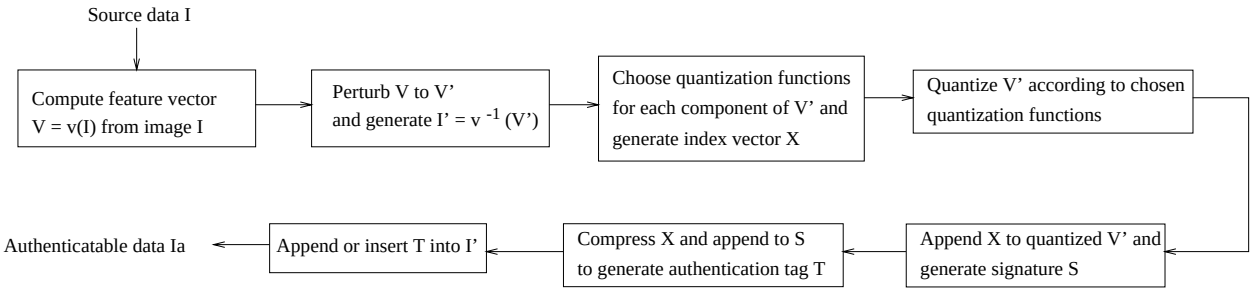

Figure 11. Flow diagram of proposed authentication scheme with authenticability distortion.

closer towards the quantized values of q_0. Thus if $d(x_i, Q_0) \geq d(x_i, Q_1)$ then x_i is moved towards y_i, where y_i is the closest point to x_i such that $d(y_i, Q_0) < d(y_i, Q_1)$. This is shown in Figure 12a where α is moved to α' whereas β is not moved since $d(\alpha, Q_0) > d(\alpha, Q_1)$ and $d(\beta, Q_0) < d(\beta, Q_1)$. Depending on how much these components are moved, this result in the index vector having even more zeros and thus more compressible, resulting in a smaller authentication tag T. In particular, if x_i is changed to y_i, then the resulting index vector consists solely of zeros and can be compressed into a single bit after removing trailing zeros.

In the second method, the components of the feature vector are distorted by moving them closer to the nearest quantized values among the quantized values of q_0 and q_1. This is shown in Figure 12b where α is moved to α' and β is moved to β'. This allows the dataset to tolerate more changes before it is deemed not authentic, i.e. β_a is larger. Note also that β_m becomes smaller and thus $\Delta\beta$ becomes smaller. In particular, by moving the components all the way to the quantized values of q_0 and q_1, we have $\beta_a = \beta_m = 2D$. This has less authenticability distortion when compared with the scheme in reference[3] where $\beta_a = \beta_m = D$.

In both methods only the authenticability distortion is changed, and the authentication step remains the same as before. After the feature vector V is distorted, a new dataset I' is constructed from V. The rest of the scheme remains the same and the tag is appended or inserted into I' to form the authenticatable data. Note that both of these modifications can be applied simultaneously or in different parts of the dataset. For instance, applying the first method to make X consists solely of zeros and then applying the second method to move the feature vector components to the closest quantized values results in an image authentication scheme equivalent to reference.[3]

8. EXAMPLES AND APPLICATIONS

In this section we illustrate how this scheme is used with an example. Consider the Lena RGB image which in TIFF LZW compressed form occupies 646KB. Authenticatable data is generated with no authenticability distortion according to Sect. 6 with the scaling values being 9 times the values in Tables 4-1 and 4-2 in reference.[7] The authentication tag consists of 5626 bytes (5466 bytes for the compressed index vector after LZW compression and 160 bytes for the digital signature assuming DSA is used) resulting in α being less than 1 percent.

We consider two modifications to the authenticatable image: JPEG lossy compression (using quality factor 5 in Adobe Photoshop 5.0) and slightly brightening of the image. Both modifications do not destroy the authenticity of the image. We also created a tampered image by adding some more purple strands to the hat. The algorithm determined this image to be inauthentic.

Another example of the proposed authentication scheme is in the application of a digital notary. In this application, original documents which are not produced by the authenticating device are made authenticatable. Example of such documents include original documents printed on special paper or using special inks, handwritten documents or previously printed documents. In this case, the area containing the essential information of the document is scanned in as an image and an authentication tag is generated. The tag is then printed onto an area of the document which does not cover the essential information in a machine readable format, such as 1-D or 2-D barcodes, or OCR fonts. The tag can also be attached to the document as magnetic strips or radio frequency identification tags (RFID). As the original document is not modified, the authenticability distortion is required to be zero. To authenticate the document, the area containing the essential information is scanned and the authentication tag is read in by the appropriate reader (which can be the scanner) and verified.

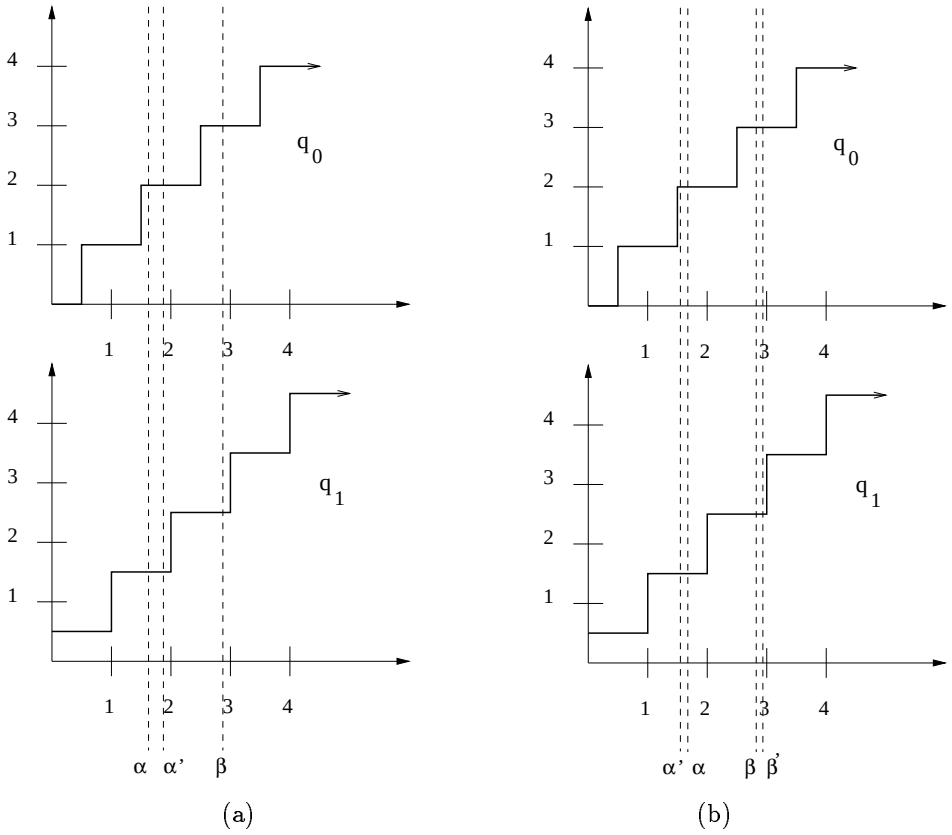

Figure 12. Authenticability distortion is applied by modifying the components of feature vector V. (a) Trade off of authenticability distortion versus tag size. (b) Trade off of authenticability distortion versus tolerated benign modification.

REFERENCES

1. M. P. Queluz, "Content-based integrity protection of digital images," in *Proceedings of SPIE*, vol. 3657, pp. 85–93, 1999.
2. E. T. Lin, C. I. Podilchuk, and E. J. Delp, "Detection of image alterations using semi-fragile watermarks," in *Proceedings of SPIE*, vol. 3971, pp. 152–163, 2000.
3. N. Memon, P. Vora, B.-L. Yeo, and M. Yeung, "Distortion bounded authentication techniques," in *Proceedings of SPIE*, vol. 3971, pp. 164–174, 2000.
4. S. Bhattacharjee and M. Kutter, "Compression tolerant image authentication," in*Proceedings of ICIP'98*, 1998.
5. J. Fridrich, "Image watermarking for tamper detection," in *Proceedings of ICIP'98*, 1998.
6. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
7. W. P. Pennabaker and J. L. Mitchell, *JPEG: still image data compression standard*, Van Nostrand Reinhold, 1993.
8. K. Sayood, *Introduction to Data Compression*, Morgan Kaufmann, 1996.
9. J. R. Hernandez, J.-F. Delaigle, and B. Macq, "Improving data hiding by using convolutional codes and soft-decision decoding," in *Proceedings of the SPIE*, vol. 3971, pp. 24–47, 2000.
10. B. Chen and G. W. Wornell, "Preprocessed and postprocessed quantization index modulation methods for digital watermarking," in *Proceedings of the SPIE*, vol. 3971, pp. 48–59, 2000.