Multimed Tools Appl (2008) 39:1–46
DOI 10.1007/s11042-007-0154-3

Methods for image authentication: a survey

Adil Haouzia · Rita Noumeir

Published online: 1 August 2007

Abstract Image authentication techniques have recently gained great attention due to its importance for a large number of multimedia applications. Digital images are increasingly transmitted over non-secure channels such as the Internet. Therefore, military, medical and quality control images must be protected against attempts to manipulate them; such manipulations could tamper the decisions based on these images. To protect the authenticity of multimedia images, several approaches have been proposed. These approaches include conventional cryptography, fragile and semi-fragile watermarking and digital signatures that are based on the image content. The aim of this paper is to present a survey and a comparison of emerging techniques for image authentication. Methods are classified according to the service they provide, that is strict or selective authentication, tamper detection, localization and reconstruction capabilities and robustness against different desired image processing operations. Furthermore, we introduce the concept of image content and discuss the most important requirements for an effective image authentication system design. Different algorithms are described and we focus on their comparison according to the properties cited above.

Keywords Image authentication · Image content · Cryptography · Fragile watermarking · Semi-fragile watermarking · Digital image signature

1 Introduction

Information had a paramount role during history at all times [15]. Its control is synonymous of ability and power. It can represent battle plans, secret negotiations or current events and television news. Exploitation of information can bring richness. Information used to be transmitted by manuscript or by voice; now it can travel thousands of kilometres in some tenths of second thanks to waves and cables. These fast technological developments make

A. Haouzia · R. Noumeir (✉)
Electrical Engineering Department, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada, H3C 1K3
e-mail: rita.noumeir@etsmtl.ca

information extremely important in our life. However, this powerful information is now more volatile and can be easily intercepted or reproduced with all the consequences which we can imagine such as false medical diagnostic, false military targets or false proof of events [22]. Image authentication is important in many domains: military target images, images for evidence in court, digital notaries documents, and pharmaceutical research and quality control images. All these images have to be protected in order to avoid false judgements.

The wide availability of powerful digital image processing tools allows extensive access, manipulations and reuse of visual materials. In fact, lot of people could now easily make unauthorized copies and manipulate images in such a way that may lead to big financial or human lives losses. These problems can be better understood with a simple example. A patient with a serious illness, discovered from medical diagnostic images, may eventually get better due to medical treatments. The medical follow-up of that patient involves the interpretation of historic images to evaluate the progression of the illness in time. A possible false diagnosis can jeopardize the patient life, if the stored image underwent malevolent manipulations, storage errors or compression, such that the resulted distortions cannot be detected by the doctor. This is an example where modifications are not tolerated. However, in many other applications we need to tolerate some image processing operations for transmission, enhancement or restoration while we still need to detect at the same time any significant changes in the image content.

Therefore, there is an ambiguity since some changes must be tolerated while others not. Consequently, image authentication can be divided in two groups: strict and selective authentication. Strict authentication is used for applications where no modifications in the protected image are allowed. On the other hand, selective authentication is used especially when some image processing operations must be tolerate such as compression, different filtering algorithms and/or even some geometrical transformations...[92, 158]. For strict authentication, solutions including conventional cryptography and fragile watermarking provide good results that satisfy users, even though some researches still need to be done in order to enhance localization and reconstruction performances of the image regions that were tampered. Selective authentication on the other hand, uses techniques based on semi-fragile watermarking or image content signatures to provide some kind of robustness against specific and desired manipulations. Results are satisfying, but the problem is far from being solved. Researches are now more concentrated in the area of image content signatures and the number of proposed solutions has increased rapidly in last years due to the large number of applications. Nevertheless, more sophisticated solutions that allow combinations of several desired modifications are still to be discovered.

In this paper, we present, discuss, classify and compare different algorithms that provide solutions for both strict and selective authentication services. Comparisons are based on different criterions such as detection, localization, restoration and tolerance. The properties of each group of methods are provided with references to algorithms.

The rest of this paper is organised as follows. First, we tend to clarify the definition of image content. Second, we thought that some definitions would help some readers; therefore, a brief introduction to some mathematical tools and vocabulary that are used in image authentication is provided. Third, we present, classify, discuss and compare approaches that have been proposed for image authentication such as conventional cryptography, fragile/semi-fragile watermarking and content-based image signatures. Our approach focuses on comparing algorithms within each group and subgroup. Methods are divided into two groups: strict and selective authentication. Strict authentication methods are further divided into conventional cryptography and fragile watermarking subgroups. Selective authentication methods are further divided into semi-fragile watermarking and digital signature based

algorithms subgroups. Algorithms based on conventional cryptography are compared between each others; algorithms based on fragile watermarking are compared between each others; conventional cryptography and fragile watermarking methods are then compared. Similar comparisons are carried on for selective authentication subgroups. A summary paragraph (Table 3 and Fig. 6) is provided with references to the performances of each method. Finally, recommendations are given for specific practical applications and future researches.

1.1 Image content definition challenge

Strict image authentication considers an image as non-authentic when just an image pixel or even one bit of data has been changed. There are applications that need such service. However, this is not the desired authentication method for most practical cases [89]. Ideally, we wish to compress an image in order to save memory space or bandwidth; we may want to enhance an image and restore it for better perceptual quality or even to convert its format.

In this context, we need an authentication service that tolerates specific image processing operations. These image processing operations change pixel values without modifying the image content. Therefore, the real problem of selective image authentication is related to the problem of image semantic content definition. In other words, we need to detect only changes that generate a modification in the image visualization or an error in its interpretation like an object disappearance or the appearance of a new object. Consequently, to develop appropriate selective image authentication approaches, it is necessary to distinguish between manipulations that change the image content and those that preserve it [22]. Unfortunately, this distinction is not easy to realize technically. Moreover, this distinction could change with images, applications and even within a single image. However, in the current literature many innovative characteristics and features have been proposed to describe the image content and identify content modifications. Several image processing operations are listed in Tables 1 and 2. Operations presented in Table 1 preserve image content in most cases and therefore authentication methods need to tolerate them. Table 2 lists manipulations that change the image content and therefore they must be detected by selective authentication methods [20].

1.2 Mathematical background and tools

Even though image authentication is a recent field, it benefits to some extent from results obtained during the last decades in conventional cryptography and information security [61]. Mathematical tools that are useful for message authentication are valid for image authentication while requiring some adaptations. Thus, a presentation of these tools and a discussion of methods to adapt them for image authentication is presented hereinafter.

Table 1 Manipulations that preserve the image content

Transmission error
Transmission noise
Storage error
Quantization and compression
Geometrical transformations (rotation, scaling...)
Enhancement techniques (spatial and frequency filtering, histograms and grey level processing...)
Restoration techniques (de-noising, deconvolution...)
Image formats conversion

Table 2 Manipulations that change the image content

Deletion of objects from the image
Addition of objects to the image
Position change of objects in the image
Change in image characteristics (texture, edges, colors...)
Change in image background (daytime...)
Change of luminance conditions (shadows...)

Cryptography is the study of principles, methods and mathematical techniques related to information security such as confidentiality, data integrity and data authentication [135]. Cryptography enables significant information to be stored or transmitted over non-secure networks, so that only authorized recipients can read it [9]. Message authentication techniques are used in image integrity and authentication systems. Hash functions, private or public key systems and digital signatures are also used. A brief introduction to these mathematical tools is presented as this is necessary to evaluate the performances of image authentication systems. However, image-processing techniques such as discrete cosine transform (DCT), discrete wavelet transform (DWT), edge detection, image enhancement and restoration and image compression are not covered here. The reader can refer to [41] for more information.

A conventional signature on a paper document usually engages the responsibility of the signer. Signatures are very common in our life. Signatures are used when letters are written, when we withdraw money at the bank, or when we sign a contract. Conversely, a digital signature aims to sign a document in its electronic form [9, 102]. So, a digital signature can be transmitted electronically with the signed document.

Many problems are encountered with digital signatures. The first problem is related to the concept of a signed document. In fact, a conventional signature is physically attached to the signed document, which is not the case for digital signatures. A digital signature algorithm is needed to attach, in some way, the signature to the electronic document. The second problem is related to the verification of the signature authenticity. A conventional signature is authenticated by comparing it with a certified one. For example, when one signs an act of purchase by credit card, the salesman compares the signature with the one on the back of the card. This authentication method is obviously not very reliable, since it is easy to imitate the signature of someone else. An electronic signature however, can be verified by any person that knows the verification algorithm. Lastly, a fundamental difference resides between conventional and digital signature: Any copy of an electronic document is identical to its original while a signed paper document copy can usually be distinguished from its original. This difference introduces a new fundamental problem related to the conceptual definition of an original electronically signed document and methods that forbid its reuse.

Digital signature algorithms that ensure message authenticity and integrity have been well studied in cryptography for four decades [102]. Several algorithms have been proposed in the literature such as the ElGamal signature [14, 27, 45], the Digital Signature Standard (DSS) [23, 39], the Van Heyst–Pedersen signature and the Dementi signature. Moreover, digital signatures based on cryptosystems such as the Rivet, Shamir, Adleman (RSA) [91, 113, 137] and Digital Signature Algorithm (DSA) [3, 95, 121] have also been proposed. Digital signature algorithms are applied either directly to the message to be authenticated or to its hash value to generate a tag that is used for the authenticity verification. Therefore, digital signature algorithms that are specific for images are not necessary. Traditional digital signatures can be used for images. However, a specific image signature problem exists. It

resides in the information to be signed: the image data or the image content? In fact, applying a digital signature algorithm directly to the image representation may result in situations where, even though the image content has not been modified, the algorithm declares the image as non-authentic. In consequence, modifications to existing digital signature schemes must be carried out by defining the information that needs to be signed. That is, traditional digital signature algorithms can be used, but the image content must be signed rather than the image data itself.

There are different tools that help implementing image authentication algorithms. The most important one is based on the hash function.

A hash function generally operates on a message X of arbitrary length to provide a fixed size hash value h, called the message digest:

$$h = H(X) \tag{1.1}$$

Where, H denotes the hash function. The size of h is usually much smaller than the size of X.

Several algorithms were proposed for hash calculation [85, 122]. The dilemma always consists in finding a compromise between security and computational time [46, 109, 112, 123, 153]. Private key encryption systems, also called symmetric encryption systems, use a single key for data encryption and decryption. The most used symmetric systems are Data Encryption Standard (DES) [1] and Advanced Encryption Standard [115].

Public key systems, also called asymmetric systems, use a pair of keys: a public key and a private key. The public key is published in directories and thus is known to everyone whereas only the person who creates the pair knows the private key. The most used public key systems are Rivest, Shamir, Adleman (RSA) [24, 110] and Pretty Good Privacy (PGP) [19, 75]. Some new protocols such as digital envelopes [65, 144] tend to get the best of both systems, private and Public, to improve performances.

2 State of the art

Before presenting and discussing various methods, we start by defining the general requirements that are essential for any authentication system. These requirements are:

Sensitivity: The authentication system must be able to detect any content modification or manipulation. For strict authentication algorithms, detection of any manipulation is required and not only content modification.
Robustness: Also called tolerance. The authentication system must tolerate content preserving manipulations. This property is valid just for algorithms that provide a selective authentication service.
Localization: The authentication system must be able to locate the image regions that have been altered.
Recovery: The authentication system must be able to partially or completely restore the image regions that were tampered.
Security: The authentication system must have the capacity to protect the authentication data against any falsification attempts.
Portability: The authentication system must be able to carry the signature with the protected image during any transmission, storage or processing operation.
Complexity: The authentication system must use real-time implemented algorithms that are neither complex nor slow.

2.1 Strict image authentication

Strict image authentication methods do not tolerate any changes in the image data. These methods can be further separated in two groups according to the techniques that are used: methods based on conventional cryptography and methods that use fragile watermarking.

2.1.1 Methods based on conventional cryptography

Image authentication methods based on cryptography compute a message authentication code (MAC) from images using a hash function [46, 85, 109, 112, 122, 123, 153]. The resulting hash (h) is further encrypted with a secret private key S of the sender and then appended to the image. For a more secure exchange of data between subjects, the hash can be encrypted using public key K1 of the recipient [141] (Fig. 1a). The verification process is depicted in Fig. 1b. The receiver computes the hash from the received image. The hash that was appended to the received image is extracted and decrypted using private key K1. The extracted hash and the calculated one are then compared.

Techniques that are based on the hash computing of image lines and columns are known as line–column hash functions [22]. Separate hashes are obtained for each line and each column of an image. These hashes are stored, and compared afterwards with those obtained for each line and each column of the image to be tested. If any change in the hashes is found, the image is declared manipulated otherwise it is declared authentic.

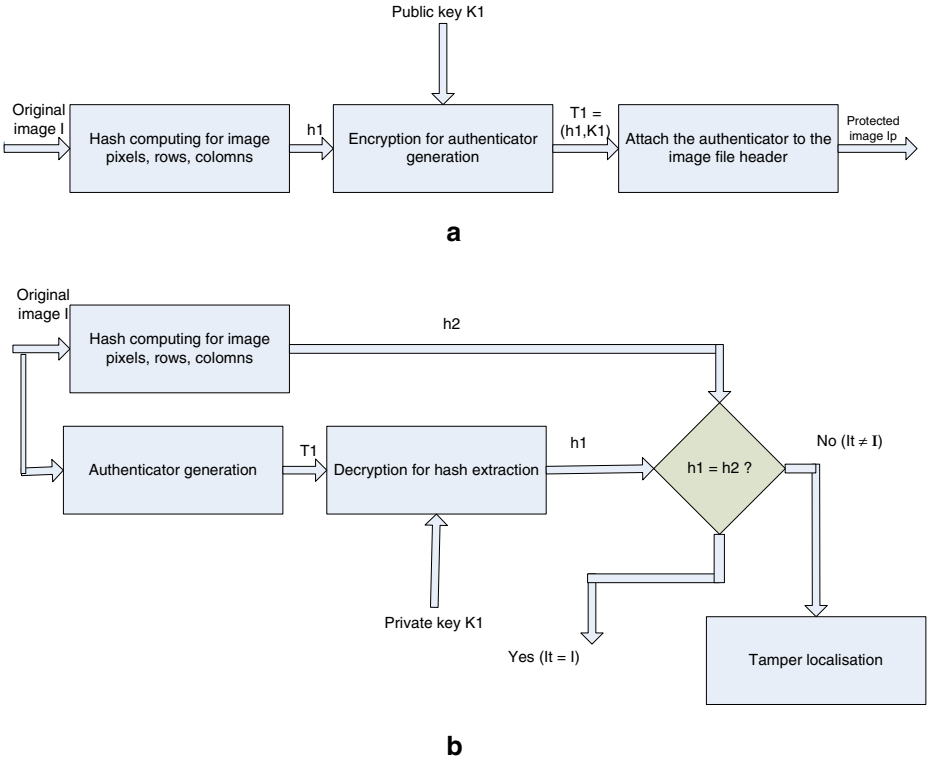

Fig. 1 Strict authentication system by conventional cryptography; **a** generation of authenticator; **b** verification of authenticity

Distortions localization can be achieved by identifying lines and columns for which the hashes are different. Unfortunately, the localization of changes can be easily lost if more than one region of the image was corrupted. This is called the ambiguity problem of the line–column hash function. To solve this problem, another approach has been proposed by Wolfgang and Delp [145]. This technique consists in obtaining the hash of image blocks, separately. If an image is to be tested, the user calculates the hashes for each block using the same block size, and compares the results with the hashes from the original image to decide whether the image is authentic. Blocks for which hashes are different enable tamper localization. The computation of hashes for each block separately had increased the localization capabilities. However, these techniques are not able to restore image regions that were tampered.

Conventional cryptography was developed to solve the problem of message authentication, and had a great success since its appearance. Algorithms based on conventional cryptography show satisfying results for strict image authentication with high tamper detection. Localization performances are not very good but may be acceptable for some applications. Hash functions are very sensitive to any small change in the image pixels or even in the binary image data. In consequence the image is classified as manipulated, when just only one bit of this image is changed; this is very severe for most of applications.

Recently, many teams of researchers have published works where they try to use hash functions while introducing some errors into the images in order to achieve methods that tolerate some desired manipulations. The type of introduced errors automatically determines the kind of manipulations tolerated such as compression [67], and histogram equalization [154]. These methods however, cannot tolerate a combination of several allowed manipulations in the same image. Moreover, they are vulnerable to attacks against the hash functions [54, 111].

2.1.2 Fragile watermarking

Watermarking consists in calculating a watermark, hiding it in the image, and then extracting it when it is necessary. In this paper, we choose fragility as the basic criterion for algorithms classification. Fragile watermarking belongs to the strict authentication class, while semi-fragile watermarking to the selective authentication class.

Some authors define reversible watermarking, also called erasable or invertible [34], as a subgroup of fragile watermarking. The idea behind reversible watermarks is to reconstruct the exact original image when the image is declared as authentic. Thus, it reconstructs the information that was lost during watermarking. Usually, it is a lossless compressed version of the space where the watermark was embedded. This lossless compressed version is thereafter concatenated with the watermark, inserted within the image and extracted for reconstruction purposes only when the image is declared authentic. However, in most image watermarking algorithms, modifications caused by embedding functions are really insignificant. Therefore, reversible watermarks are desired only for specific applications such as for high sensitive images. Moreover, their main goal is to eliminate the distortion artifacts caused by the embedding functions. Interested readers could consult references [34, 40, 64, 133] on this subject.

Throughout this paper we compare the restoration capabilities of each algorithm, which is somehow different from reversibility. Restoration is the ability of an algorithm to restore the damaged data. When an algorithm detects and localises a region with some undesired manipulations, we wish that this algorithm could restore the original data. This requirement is desirable for wide range of applications.

Additionally, we classify an algorithm as symmetric or asymmetric. This mainly depends on the security key model. The asymmetric algorithms are generally more secure as they provide different private and public keys for encoding and decoding. However, they are much slower than symmetric algorithms [112, 123, 153].

The basic idea behind fragile watermarking techniques is to generate a watermark and to insert it in the image to be protected in such a way that any modification made to the image is also reflected in the inserted watermark. Simply verifying the presence of the inserted watermark allows the image authenticity verification and eventually localization of tampered regions. This type of watermarking does not tolerate any image distortion. The image is considered authentic if and only if all its pixels remain unchanged.

The first algorithms of fragile watermarking were based on watermark generation from image information only [142] as shown in Fig. 2a. The watermark is computed from a set of image pixels. The computation of the watermark differs between the various authentication methods. The set of pixels may be chosen with the help of a secret key

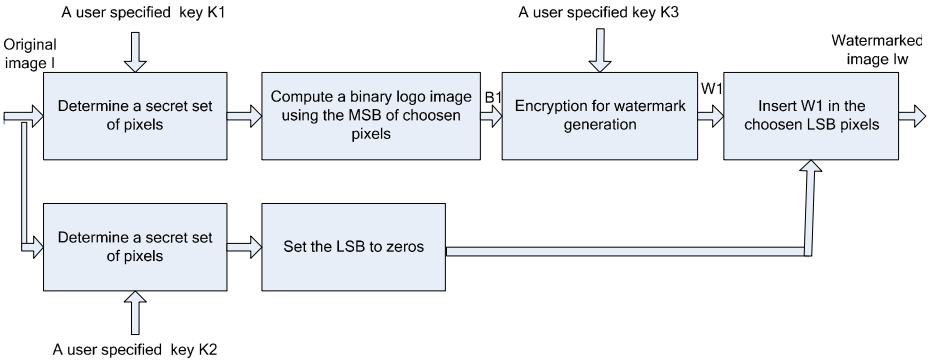

Fig. 2 Strict authentication system by fragile watermarking using image information; **a** generation of authenticator; **b** verification of authenticity

K1. The computed watermark may be encrypted with a key K3. It is then inserted in the least significant bits of another set of pixels. In order to increase the algorithm security, the set of pixels where the watermark is embedded may be determined with another secret key K2. Similarly, the verification schema is shown in Fig. 2b. The secret keys must be known to the receiver, as well. The receiver uses the same key K2 to determine the set of pixels where the watermark is dissimulated in order to extract it. Also, the receiver uses the same algorithms to calculate the watermark from the received image and then compares the calculated watermark with the dissimulated one to decide whether the image is authentic or not.

One of the first techniques that used image authentication by fragile watermarking was proposed by Walton [142]; it used only image information to generate the watermark. This technique is based on the insertion, in the least significant bits (LSB), the checksum calculated with the grey level of the seven most significant bits of pseudo-randomly selected pixels. This method was able to detect and localize manipulations but with no restoration capabilities. Various algorithms were proposed for the realization of this technique [5, 16, 31]. The algorithm that attracted most attention was proposed by Fridrich [31]. A sufficient large number N is chosen to be used for the calculation of the checksums. The size of N directly impacts the probability of detecting manipulations. The original image is first subdivided into blocks of size 8×8; in each block, a pseudo-random walk through its 64 pixels is generated. The checksum S is calculated by the following equation:

$$S = \sum_{j=1}^{64} \left(a_{\mathrm{i}}.g\left(p_{\mathrm{j}}\right)\right) \bmod N \tag{2.1}$$

Where $g\ (p_j)$ is the grey level of the pixel p_j obtained with the seven most significant bits only and a_j is a pseudo random sequence of 64 integers. Then the binary format of S is encoded using a coding algorithm, and inserted into the LSB of the block pixels. To increase the security of the system, the coefficients a_{j} and the pseudo random walk can be dependent on the blocks by using secret keys. The procedure of checking an image authenticity consists in extracting the inserted checksums, recalculating the checksums in a similar way, and finally comparing the two checksums to decide about the image authenticity. This method has the advantage of being very simple and fast. Moreover, it detects and localizes tampering. However, the algorithm cannot detect the manipulation if blocks from the same position of two different images, which are protected with the same key were exchanged. To avoid this type of attack, several improvements were made to this method by extracting more robust bits [16]. The method is not able to restore the damaged data.

In a more general schema, the watermark that is inserted in the image to be authenticated is obtained by combining information from the image with a predefined logo as depicted in Fig. 3a and b. A secret key K1 can be used to extract specific image information from the image. In order to generate the watermark, the extracted image information is combined with a binary logo by using another secret key K2. The computed watermark may be encrypted with a key K4. It is then inserted in the least significant bits of a set of pixels that may be determined with a secret key K3. The secret keys must be known to the receiver, as well. The receiver uses the appropriate key to determine the set of pixels where the watermark was dissimulated in order to extract it. Also, the receiver uses the same algorithms to calculate the

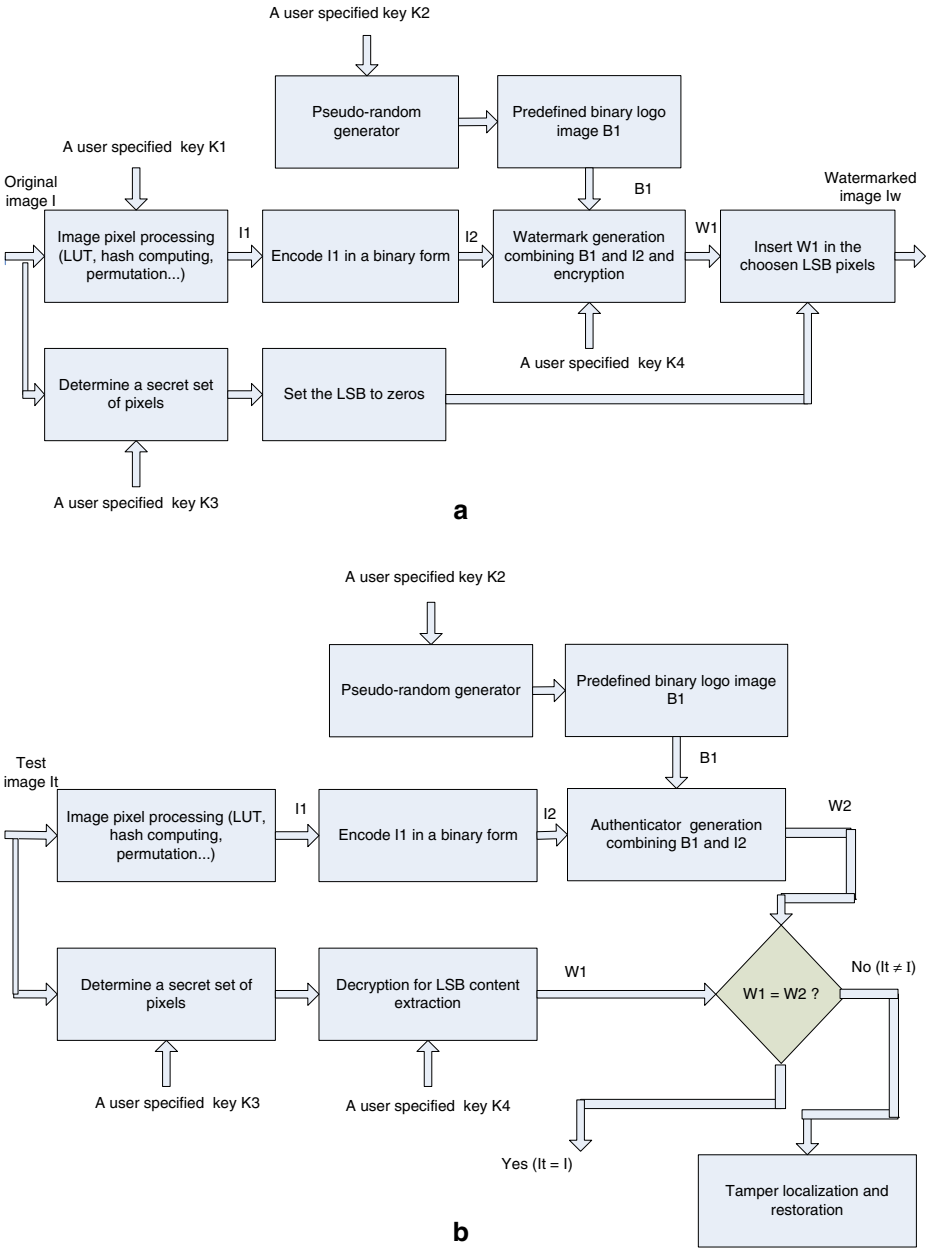

Fig. 3 Strict authentication system by fragile watermarking where the watermark is obtained from the image and a logo; **a** generation of authenticator; **b** verification of authenticity

watermark from the received image and then compares the computed watermark with the dissimulated one to decide whether the image is authentic or not.

Yeung and Mintzer proposed a method [159] based on a secret key that is used to generate a binary function f_g. This binary function f_g maps integers from the interval {0, 1,

2,..., 255} towards the binary values 0 or 1. For color images, three similar functions, f_R, f_G, f_B, are generated for each color channel. These binary functions are used to code a logo L. For each pixel (i, j) the following expression is calculated for the grey levels g_{ij}:

$$L_{ij} = f_g(g_{ij}) \tag{2.2}$$

To verify the authenticity of an image, the user can easily check for each pixel (i, j) the relationship $L_{ij}=f_g(g_{ij})$. This method has the advantage of allowing visual inspection of the image integrity and localization of the tampered areas. The logo L can be inserted more deeply by using more grey levels to increase the security of the method. However, it was shown by Fridrich and Memon [86, 88] that a malevolent person can easily estimate the inserted logo or the binary function, if the key used to generate the binary function was used for several images. The attack is more complex for color images but remains threatening since the algorithms are symmetric. Furthermore, the algorithm has no restoration capabilities. This type of attack is known as the vector quantification attack [48]. To prevent it, Fridrich proposed in his work [33], to make the logo dependable on image indexes. These indexes are inserted in the original image several times into various blocks to exclude all attempts to remove them. However this technique remains vulnerable to this attack because the used method is not very reliable. To avoid the vector quantification attack, another asymmetric method based on public or private key systems, was proposed by Wong [147, 148] and was improved afterwards through collaboration with Memon [150]. An image $X(M, N)$ to be protected, and a logo B, are subdivided into blocks of size $I \times J$. For each block X_r of the original image, a corresponding block $X_{r'}$ is formed, where each element of $X_{r'}$ is equal to its corresponding element in X_r except for the LSB that is zero. A hash d is calculated for each new block $X_{r'}$ according to the following equation:

$$d = H(K, I_x, M_x, N_x, r, X_{r'}) \tag{2.3}$$

Where H is MD5 hash function, K is a secret key, I_x is an identification code of the image and r is the block index. The resulting hash d is either truncated or padded to the block dimension. A new block C_r is formed by combining d with its corresponding block B_r from the logo according to the following expression:

$$C_r = B_r \, XOR \, d \tag{2.4}$$

Finally, each element of the resulting block C_r is inserted in the LSB of the corresponding element in the block $X_{r'}$. This procedure is repeated for the whole image and the resulting blocks are grouped to form the watermarked image. The verification procedure starts with the extraction of the LSB for each block G_r of the image G to be tested. A new hash f for each block of the image G is generated. Finally, the operation of exclusive OR is applied between the block G_r and the new hash f. The result of the exclusive OR operation gives the logo if and only if the arguments of the hash function (the key, ID, dimensions, index of the block) are unchanged, otherwise, the result will look like a random noise. This method seems to prevent the vector quantification attack and to provide in the same time tampering localisation capabilities. However its security depends on the security of the used keys. In the case where the used keys are private, it is necessary to carry out the key exchange over a secure channel, which complicates the practical use of this technique, but ensures a high level of security. Furthermore, the problem of restoration capability was not solved.

A more recent method was proposed by Byun and Lee [10] to authenticate color images by fragile watermarking. The original color image $I(M, N)$ to be protected is decomposed into its three color components I_R (red), I_G (green), I_B (blue). As frequency response of the

blue component is much smaller than the red and green frequency responses, the blue component is often used to hide information [117]. The components I_R and I_G are used as the data for authentication. Each LSB is extracted from the two components I_R and I_G to form a vector. A key is used to carry out a permutation of these vector elements. The hash function MD5 is calculated on the permuted vector to generate a sequence h of 128 bits. A logo W and the sequence h are resized so that both of them have the same dimensions as the original image (M, N). The exclusive or operation is calculated, element by element, between the resulting logo W_0 and the resulting hash h_0:

$$F = W_0 \, XOR \, h_0 \tag{2.5}$$

The result of this operation is encoded by a private or public key system. The LSB of the component I_B of each pixel are put to zero, and F is dissimulated in these bits. The verification procedure starts with the extraction of the LSB of the blue component I_B, the decoding of these bits with the corresponding secret key. A hash is obtained from the red and green components I_R and I_G in a similar way as for the insertion. The operation of exclusive OR is applied, element by element, between the hash and the result of the I_B component LSB decoding. The result of this operation will be the inserted logo if and only if the checked image is identical to the original. One can notice that this method has the same problem of key security as the method described previously. Moreover, the algorithm cannot restore the damaged data as the watermark is lost if any changes are made to the watermarked image. This is the irreversibility property of this algorithm. However it is an adequate method for strict color image authentication and it has an acceptable capability of localizing modifications made to an image.

Then again, Fridrich and Goljan proposed a method [35, 36] that translates the grey levels of the original image into the interval [−127, 128]. Then it breaks up the image into 8×8 blocks and transforms each block separately using the discrete cosine transform (DCT). The DCT is applied to the most significant bits (MSB) of the pixels. The DCT coefficients of each block are arranged according to the zigzag order used in JPEG compression. The first 11 coefficients of each block are quantified by the JPEG quantization table that results in a 50% reduced quality. These quantified values are then binary encoded on 64 bits and inserted into the LSB of the pixels from another block. The block that is used as support for the dissimulated data is chosen sufficiently far away from the source block. The use of the first 11 coefficients guarantees that the result of their encoding would be a binary sequence of 64 bits. This method has the advantage of being the only method seen until now that has some restoration capabilities because sufficient and significant information is hidden. In fact, the DC and the first ten AC coefficients of the DCT contain sufficient information to restore a block with an acceptable quality. Even though, the quality of the restored regions is lower than 50%, it is sufficient to inform the user about the original content. Moreover, the localization capabilities are very satisfying since 8×8 blocks are used as authenticators. In order to improve the quality of the restored regions, the authors proposed to use two LSB to hide 11 DCT coefficients hiding thus more significant information while allowing a poor quality watermarked image. This method is vulnerable to the whole image attacks that destroy the capacity of restoration such as setting all LSB bits to zeros. Other methods based on ideas similar to those presented in this section were also proposed [25, 79]. They are based on fragile watermarking and use LSB to hide information.

Strict image authentication is appropriate for many applications. For example, a modification of just one or two pixels in some medical or military images can dramatically change the decisions of doctors or war strategists, respectively, and can result in costly

damages. Most existing image applications use image processing operations that preserve the content in order to save memory space and bandwidth or to enhance image quality: compression, filtering, geometrical transformations and image enhancement techniques. Therefore, some tolerant image authentication algorithms are needed.

2.2 Content-based image authentication or selective authentication

We defined a content modification as an object appearance or disappearance, a modification to an object position, or changes to texture, color or edges (see Section 1.1). We have also listed the image processing operations that preserve the image content. Thus, lot of applications that base their decisions on images need authentication methods that can tolerate content preserving manipulations while at the same time detect any manipulation that change the image content. This leads to new watermarking methods known as semi-fragile watermarking, and to new approaches known as content-based signatures. In this section we will present and compare semi-fragile techniques and content-based signatures approaches that provide selective image authentication service.

2.2.1 Semi-fragile watermarking

Robust watermarking is designed to resist all attempts to destroy the watermark. Its main application includes the intellectual property protection and owner identification. The robustness of the embedded watermark is crucial to resist any intentional and even unintentional manipulation. The goal of these techniques is not the verification of the image authenticity, but rather the verification of their origins. Conversely, fragile watermarking, presented in Section 2.1.2, is designed to easily destroy the embedded watermark following any kind of manipulations of the protected image. It is useful for applications where strict authentication is needed, that is where the main objective is to determine whether the image has been modified or not, with the possibility of locating and reconstructing image regions that have been tampered. On the other hand, semi-fragile watermarking [29, 30, 37] combines characteristics of fragile and robust watermarking techniques.

Basically, the idea of semi-fragile watermarking is to insert a watermark in the original image in such a way that the protected image can undergo some specific image processing operations while it is still possible to detect malevolent alterations and to locate and restore image regions that have been altered. For image authentication purposes watermarking algorithms should be invisible. Visible watermarking algorithms are applied for on-line content distribution, transaction tracking or owner identification. The procedures of generating a watermark and embedding it into the image can be dependent on a private or public, symmetric or asymmetric, key system in order to increase the overall system security. This is a trade-off between security and computational time [46, 109, 112, 123, 153]. Generally, symmetric key systems are less secure than asymmetric ones, and asymmetric key systems consume more resources and consequently need more computing time.

The general schema for semi-fragile watermarking methods is shown in Fig. 4. The watermark is computed from the result of an image-processing algorithm applied on the image pixels. The computation of the watermark varies as different image processing algorithms can be used. A secret key K1 can be used to extract specific information from the image. In order to generate the watermark, the extracted image information is often combined with a binary logo using another secret key K2. Usually, the generated watermark is then inserted in a set of frequency coefficients that are in the middle range. The set of

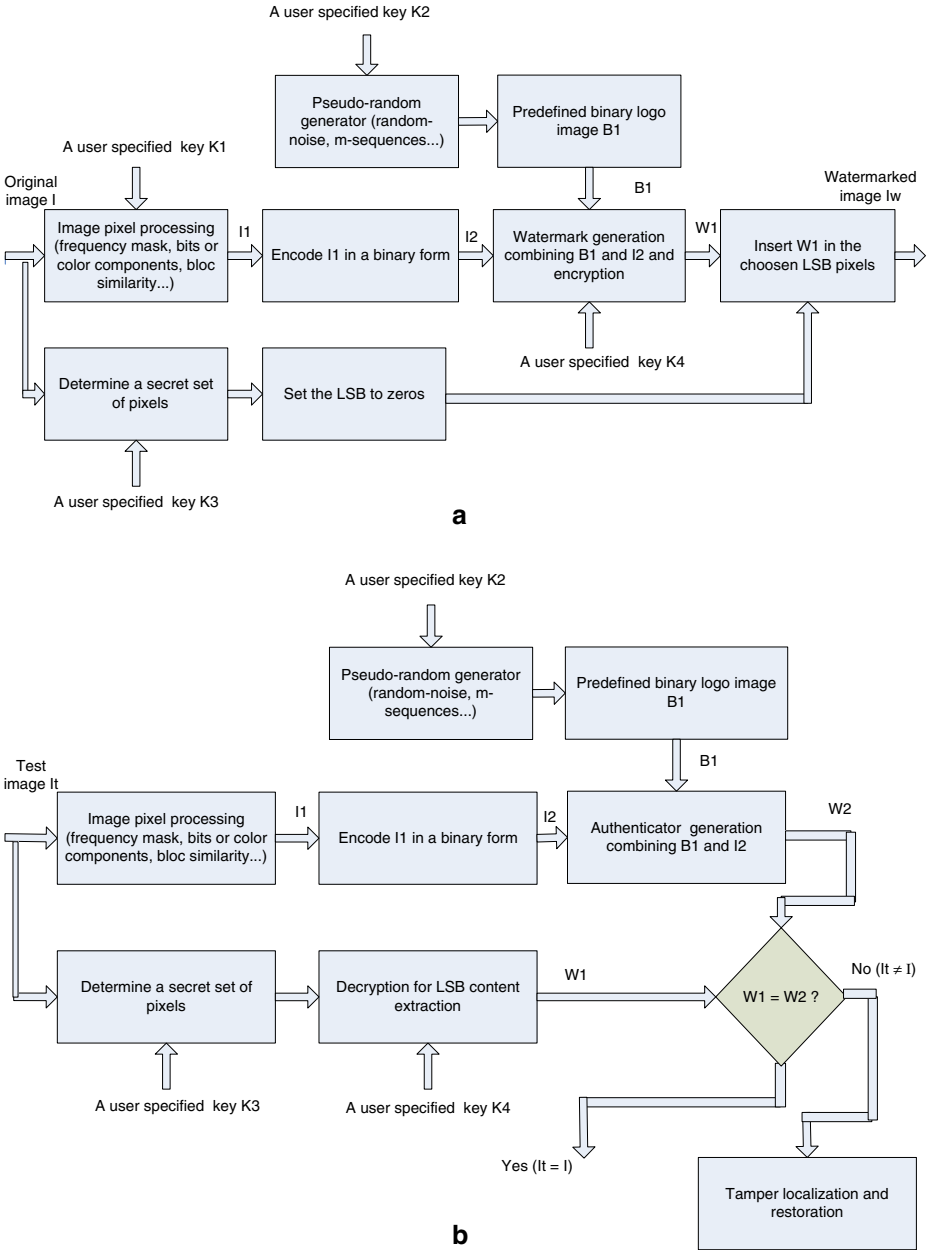

Fig. 4 Selective authentication system by semi-fragile watermarking; **a** generation of authenticator; **b** verification of authenticity

coefficients where the watermark is inserted may be determined with the help of a secret key K3. The computed watermark may be encrypted with a key K4. Similarly, the general verification schema is shown in Fig. 4b. The secret keys must be known to the receiver, as well. The receiver uses the same key to determine the set of pixels where the watermark is dissimulated in order to extract it. Also, the receiver uses the same algorithms to compute

the watermark from the received image and then compares the computed watermark with the dissimulated one to decide whether the image is authentic or not.

van Schyndel, Tirkel and Osborne proposed a method [140] that exploits the maximum length shift register sequence, called the *m*-sequences. The *m*-sequences are often used to represent random sequences. These sequences are of length $L=(2^n-1)$ bits, where n is the number of echelons in a register. They are generated from m shift registers with looping. The registers mainly depend on primer polynomials coefficients. These *m*-sequences have a period L and each one of them contains (2^{n-1}) elements equal to (+1) and $(2^{n-1}-1)$ elements equal to (−1). The most important characteristic of the *m*-sequences is the auto-correlation function $R_{x,x}(q)$:

$$R_{x,x}(q)=\sum_{i=0}^{L-1}x_i x_{i+q}=\begin{cases}L, if & q=0\\ -1, if & q\neq 0\end{cases} \tag{2.6}$$

Where $0\leq q<L$ is a shift between two sequences.

For large i, where i is the number of bits, the following property $R_{x,x}(0)>>R_{x,x}(q)$ is obtained, $q\neq 0$. This property allows the localization of the random sequence even in the presence of additive noise. The method suggested by the authors modifies the LSB by adding extended *m*-sequences to the lines of the original image. The *m*-sequences can be generated recursively by the relation of Fibonacci. The auto-correlation function and the spectral distribution of these *m*-sequences are similar to those of a Gaussian noise [116]. For a 512×512 image, coded with 8 bits, a sequence of 512 bits length is randomly changed and coded line by line in the LSB of each pixel. A simple cross-correlation [96] is used to test for the presence of the watermark by comparing the content of the LSB of each pixel with a watermark that is generated with the same parameters. The security of this system, which is based on the security of the generated *m*-sequences, was tested and approved in several works [37]. To exploit the unique and optimal auto-correlation function of the *m*-sequences, the authors proposed another technique in the same work [134] which adds the bits of the watermark to the LSB of each pixel instead of replacing it. The detection capabilities are good but localization capabilities are not optimal. Robustness against noise addition and compression based on adaptive histogram manipulation and JPEG standard are also demonstrated. However, the algorithm is not able to recover the damaged data in an image. In fact, if an image is declared to be non-authentic, there is no information that can be used for restoration since the *m*-sequences used as a watermark are not dependent on the image. Moreover, the algorithm needs high computationally time which makes it almost impractical.

This method has been further improved by Wolfgang and Delp [146] with small modifications that enhance its robustness and its capability to localize the corrupted regions. To generate the watermark, a binary sequence is mapped from {0, 1}to {−1,1}, rearranged in desired blocks of dimensions 8×8, and added to the pixels value of the correspondent 8×8 blocks in the original image with the following expression:

$$Y(\mathrm{b})=X(\mathrm{b})+W(\mathrm{b}) \tag{2.7}$$

Where $Y(b)$ is the watermarked image, $X(b)$ is the original image and $W(b)$ is the watermark. The presence of the watermark can be simply verified by:

$$\delta(b)=Y(b)W(b)-Z(b)W(b) \tag{2.8}$$

The values of $\delta(b)$ are then compared with a threshold to decide about the authenticity of the tested image $Z(b)$. The threshold represents a compromise between the robustness of the system and its capacity to detect any malevolent manipulation. The authors proposed to

compute it from the number of elements in the watermark block. Another proposition may be an adaptation of the threshold to different image regions. This could help tolerate different operations dependently on the image information in each region. The main advantage of this method consists in allowing an authorized user who knows the watermark to reconstruct the original image. Moreover, since the correlation properties cannot be affected without the knowledge of the embedded sequence, this method is secure. However, an attacker can compute the watermark in a block, knowing a limited number of consecutive hidden bits. To avoid this problem, the authors proposed to use other codes, such as the nonlinear codes of gold [38, 114, 129] or Kasami codes [43, 47]. This algorithm preserves robustness against noise addition and compression. Moreover, it is able to survive some filtering operations and can localize malevolent manipulations with acceptable precision. However, some additional tests still need to be carried out with other manipulations that preserve the content.

Alternatively, Zhu and Tewfik [163] proposed two techniques that use a mask in the spatial or in the frequency domain. Their watermark can detect errors until half of acceptable manipulations for each pixel or each frequency coefficient according to whether the mask is used in space or in the frequency domain. Generally, the effects of space or frequency masking are often used to form sequences of pseudo noise in order to maximize the energy of a watermark while maintaining the watermark itself invisible. The authors used the masking values obtained by the model with visual threshold from their work on image binary rate low coding [162]. In fact, the spatial or frequency masking is inspired by the human visual system model (HVS). This model is used to determine the maximum invisible distortion that can be applied to each pixel or each frequency coefficient. The original image is subdivided into blocks and for each block a secret random signature (a pseudo-random sequence uniformly distributed in the interval [0, 1]) is multiplied by the mask values of this same block. The resulting signal depends on the image block, and it is added to the original block. The resulting values are then quantified by the same mask values. The authors apply this technique to blocks of dimensions 8×8. The blocks are transformed thereafter using the DCT, and the mask values $M(i, j)$ for each DCT coefficient $P(i, j)$ are computed by the frequency mask model. The values $M(i, j)$ are the maximal changes that do not introduce perceptible distortions. The DCT coefficients $P(i, j)$ are modified by the following expression:

$$P_s(i,j) = M(i,j)\left\{ \left\lfloor \frac{P(i,j)}{M(i,j)} \right\rfloor + sign(P(i,j))r(i,j) \right\} \tag{2.9}$$

Where $r(i, j)$ is a key dependent noise signal, $\lfloor x \rfloor$ round x to zero and sgn(x) is the sign of x defined as:

$$\text{sign}(x) = \begin{cases} 1, \text{ if } x \geq 0 \\ -1, \text{ if } x < 0 \end{cases}.$$

The error introduced by the operation below is smaller than the threshold of imperceptibility. This means that: $|P_s(i,j) - P(i,j)| \leq M(i,j)$, which implies that the modifications made to the DCT coefficients are invisible. For a test image with DCT coefficients $P_s(i, j)$, the mask values $M'(i, j)$ are computed. The error e' is estimated by:

$$e' = P'_s - M'\left\{ sign(P'_s)r + \left\lfloor \frac{P'_s}{M'} - \left(r - \frac{1}{2}\right)\text{sgn}(P'_s) \right\rfloor \right\} \tag{2.10}$$

Where all the values are evaluated at the same frequency location (i, j). This technique can detect malevolent manipulations and can localise them with acceptable precision. The algorithm is robust against JPEG compression with small ratios. However, it detects only errors smaller than half of the acceptable distortion for each pixel. In other words, this method gives good results when the distortions in the tested image are relatively small. Moreover, its robustness against other manipulations that preserve the image content was not shown. Frequency masking method is more powerful than the space masking method, which is very sensitive to the noise introduced by the watermark shape. The algorithm did not show any restoration capabilities.

Lin and Delp proposed a method [71], which uses the DCT domain to generate a smooth watermark that resists to damages caused by JPEG compression. Gaussian distributions of pseudo-random numbers, with a zero average and unit variance, are used to generate the watermark that is localized in each DCT block. In consequence, each block contains a different watermark, but the distribution of the watermark in all blocks is similar. Since the watermark is dissimulated in the middle frequencies DCT coefficients it is invisible and resists JPEG compression. In fact, if a watermark was embedded in high frequencies, it could easily be destroyed by JPEG compression and if it was embedded in low frequencies, it could become visible. For color images, the same procedure is applied to the brightness component only. Once the watermark is built in the DCT domain, the inverse transform is taken to produce a watermark in the spatial domain. This watermark is embedded in the original image by:

$$Y = X + sW \tag{2.11}$$

Where s is the watermark strength, X the original image and Y the watermarked image.

Detecting the presence of the watermark is based on the difference between adjacent pixels in the spatial domain. Most natural images contain large areas with relatively smooth characteristics and few edges, therefore the energy in a difference signal must be substantially less than the energy of the signal itself. Thus, in the absence of edges, the difference between two adjacent pixels represents the embedded signal plus a small random energy component from the image itself. Block's authenticity is decided by comparing the difference with a predefined threshold. This method is robust against JPEG compression even with large ratios. The detection and localization capabilities of this algorithm are very acceptable, but its performance could be affected by the block size. In fact, the block size must be large enough to permit a good detection of alterations. At the same time this could affect its localization performances. The algorithm is not able to recover the damaged data if they are detected. Moreover, the presence of an image with many edges and textures could dramatically decrease the algorithm performances since it is based on the differences between adjacent pixels in the spatial domain. To solve this problem, regions with less edges and textures could be watermarked with small strength s and regions with more textures and edges could be strongly watermarked (larger value of watermark strength s).

To generate a robust watermark, Chen and Wang [13] used the method of robust bits extraction proposed by Fridrich [32]. The original image is divided into 16×16 blocks. It is supposed that there are N blocks after the division of the image; K bits are extracted from each block to form a watermark of $K\times N$ bits. Using a secret key, $K\times N$ random matrices that have entries uniformly distributed in the interval [0, 1], are generated. A low-pass filter is applied to each random matrix to obtain $K\times N$ smooth random shapes. After subtracting the average of each shape, each block is projected on K forms successively, to obtain $K\times N$

scalar values Vi. Finally, their absolute values are compared with a threshold T, to obtain $K \times N$ bits Bi:

$$\mathrm{Bi} = \begin{cases} -1, \text{if } |\mathrm{Vi}| < T \\ 1, \text{ if } |\mathrm{Vi}| \geq T \end{cases} \tag{2.12}$$

The results obtained by Fridrich [32] show that these bits are robust against some image processing operations. They are used by the authors as a watermark that is embedded in the image. The method used to embed the watermark in the original image is based on the work of Inoue [50] who applies a DWT to the image, inserts the watermark in low frequency subbands and carries out an inverse DWT to obtain the watermarked image. This method allows the embedding of the watermark bits in the same block from which they were extracted, which help enabling good detection and localization of corrupted regions. Verifying an image authenticity is achieved by extracting the watermark with the inverse operations that have been used to embed it. The obtained watermark is then compared with the bits extracted for each block. If the number of different bits exceeds a predefined threshold, the corresponding block is considered altered. Otherwise the block is authentic. This method is robust against compression, but its robustness was not tested against other manipulations that preserve image content such as filtering, scaling or contrast adjustment. Moreover, this method depends on a threshold to decide about an image authenticity. This threshold may vary from one application to another and depends on the image content. Therefore, the threshold is supposed to be adapted to a specific image and within a specific region. The algorithm did not show any restoration capabilities.

Then again, Paquet and Ward proposed a method based on the DWT [98]. This method was also simulated by Kundur and Hatzinakos with some modifications in [57]. An identification key of 64 bits length is secretly produced. A 64 bits length is considered 'sufficient' to guarantee the uniqueness of the key. The key is used to secretly select the parameters, the wavelet and scaling functions of DWT decomposition. The DWT is applied on the original image according to the selected parameters. The same key is used again to choose the level of details as well as the position in each level of the coefficients that are used to embed the watermark. The same and unique key is used again, but this time as a watermark. The zeros are inserted by odd quantization of the selected coefficients, whereas the ones are embedded by even quantization. The modified coefficients and those that remained unchanged are used to apply the inverse DWT in order to obtain a protected image. The same key is used in watermark extraction by following the same procedure as for the embedding. The extracted watermark is compared then with the user key in order to verify the authenticity of the image. The comparisons are carried out between the frequency bands and within each frequency band to detect manipulations both in the spatial and in frequency domains. The proposed algorithm can detect and localize any tampering with acceptable precision. It is robust against JPEG2000 compression that is based on DWT. However, its robustness against other content preserving manipulations has not been tested yet. Moreover, its security is based on the key security that must be transmitted separately through a secure channel, which is not always easy to realize in practice. The algorithm did not provide any reconstruction capabilities if the image was corrupted since it uses only the key as the watermark.

Conversely, in [80], Lu and Liu proposed a method based on vector quantification coding (VQ). In vector quantification coding, the original image X is subdivided into small blocks x_i of dimension $K=1 \times h$; each block is rearranged as a vector. For each input vector x_j, the VQ coder finds a codeword c_j, in the codebook $C=\{c_1, c_2,...,c_{N-1}\}$ that best

matches it, and only the code index j is transmitted through the channel. The codebook is generated in advance with the Linde, Buzo, Gray (LBG) algorithm [73]. The VQ decoder uses the same codebook $C=\{c_1, c_2,...,c_{N-1}\}$ to find the vector c_j that corresponds to the received index j.

To embed a bit in each transmitted index, the authors proposed an approach based on VQ coding with an index constraint. The principal idea behind this approach is that since each index contains n bits, there are n candidate positions where a bit can be embedded. For example, the position 'm' can be selected to insert a bit, $0<m<n-1$; contrary to normal VQ coding, the process of watermark bit dissimulation is carried out by looking for the codeword c_p who is the most similar to the vector x_p under the constraint that the mth bit of the index p is equal to the watermark bit which we want to dissimulate. The extraction procedure starts with the subdivision of the tested image to small blocks of equal dimensions $K=1\times h$. The codeword c_p, corresponding to the index p is found for each block x_p in the reception codebook C. The reconstruction of the watermark is done by regrouping all the dissimulated bits in each index knowing their positions. This method is robust against JPEG as well as VQ compression, which is becoming increasingly used in practice. Moreover, it is robust against some geometrical transformation such as rotation but with small angles. The algorithm is able to detect and localize malevolent manipulations. However the user must have the codebook in order to verify the authenticity of an image; consequently, the codebook has to be securely exchanged in advance. Furthermore, this method introduces some distortion that could eventually affect the watermarked image quality. The algorithm did not provide any reconstruction or restoration capabilities.

In a recent work [81] the authors proposed a novel method based on a multipurpose scheme that offers both robust and semi-fragile watermarking. The robust watermarking consists in ensuring a copyright service which is not the main object of this paper. The semi-fragile watermarking part is similar to the one described above. However, some improvements were added to decrease distortions. The authors proposed to add another constraint for watermark bits dissimulation. It consists in embedding the watermark bit in the position that introduces the smallest possible extra distortion. The performances of the new algorithm are very similar to those described above with additional tolerance against rotation of relatively large angles and some filtering algorithms. The new algorithm has not proposed any improvement to restore or reconstruct the damaged data.

On the other hand, a method for color image authentication by semi-fragile watermarking was proposed by Kostopoulos, Gilani, Skodras [55, 56]. The original color image of dimensions $M\times N$ is first transformed into the YC_bC_r domain. The brightness component is used as an authentication watermark; this enables the restoration of modified regions. The brightness that is coded on 8-bits per pixel is processed to reject the two LSB in order to reduce the required storage space. The watermark is embedded in the three color components, in such a way that an information bit affects in average four of the LSB at each color intensity value. Given $C(i, j)$, $i=0, 1,..., M-1$, $j=0, 1,..., N-1$. The value of an 8 bits color component at pixel (i, j) is mapped towards x such as:

$$|\, C(i,j) - x| < 10 \tag{2.13}$$

and

$$f(x) = \begin{cases} 0, \text{ if } ((x \bmod 100)\text{div}10) \bmod 2 = 0 \\ 1, \text{ if } ((x \bmod 100)\text{div}10) \bmod 2 = 1 \end{cases} \tag{2.14}$$

Where mod is modulo function and div is an integer division. The function f maps the values [0, 255] towards [0, 1]. If the bit to be inserted is 0, then the intensity value $C(i, j)$ is

mapped towards x so that $f(x)=0$, and x is modified so that it is equal to the center of the interval $[x1, x2]$. Where:

$$\begin{cases} x1 \bmod 10 = 0 \\ x2 \bmod 10 = 0 \end{cases} \text{ and } x2 - x1 = 10.$$

Inserting a bit of information equal to 1 is achieved in a similar way. Hiding the watermark is completed by the use of a secret key K that map the position p_i of the pixel from which the brightness information is taken, towards the position p_k of the pixel where information is dissimulated. The choice of the key K is important since it affects the system security. To verify an image authenticity, the brightness of each pixel is computed and compared with the dissimulated watermark information of the corresponding pixel using the same secret key K. The main advantages of this approach are its usefulness for color images and its capacity to approximately restore the corrupted regions. In fact, once a region is identified as manipulated the algorithm can localize the tampered region and try to restore it. The restoration is based on the brightness information used as a watermark which provides some acceptable quality. Only grey scale approximation of the damaged regions is restored. Moreover, this method has an acceptable dissimulation capacity for color images, but it is not tested against some content preserving manipulations such as geometrical transformations and brightening.

Marvel and Hartwig [84] proposed a method, which exploits a system of data hiding, called spread spectrum image steganography (SSIS). SSIS uses error correction coding to encode the watermark and employs techniques of spreading spectrum to dissimulate information in a signal that appears like a Gaussian white noise [83]. This white noise is added thereafter to a low-resolution version of the original image to form the watermarked image. The white noise appears to be caused by an image acquisition instrument and by consequence, it is imperceptible. For an original image I of size $N \times N$, a DWT is applied to obtain the thumbnail T.

T contains the lower order coefficients at a level n of the wavelet transform W: $T=W(I, n)$. The user must adequately choose the scale n to ensure that all the important regions of the image, those for which the detection of malevolent manipulations is required, are visible in the thumbnail. The watermarked image G is built by the coder of SSIS, which inserts T in the original image using a secret key K: $G=\text{SSIS}(I, T, K)$. The verification of a suspect image G' is similar to the method of inserting the watermark. The original thumbnail that represents the inserted watermark is extracted from the image G' using the inverse decoder SSIS^{-1} with the same secret key K: $T=\text{SSIS}^{-1}(G', K)$. The new thumbnail T' is extracted from the image G' using the same procedure and the same wavelet parameter n: $T'=W(G', n)$. The two thumbnail T and T' are compared taking the absolute difference of the inverse wavelet transform:

$$D = \left| W^{-1}(T, n) - W^{-1}\left(T', n\right) \right| \tag{2.15}$$

In the regions where D is large, one can assume that malevolent manipulations were carried out. If D is small, one can suppose that it is the effect of compression or transmission errors. For the image areas with $D=0$, manipulations are not significant. The algorithm can detect and localize any malevolent manipulation with acceptable precision. Furthermore, results obtained by this method show its robustness against JPEG compression. The authors did not provide any clarification on how to choose the scale n. This step could be very delicate since it depends on the image regions that need to be

protected. Other experiences should be carried out to evaluate the algorithm performance against additional content preserving manipulations such as geometrical transformations and filtering. The algorithm is not able to restore the damaged data.

In their recent work Sherif and Mansour [120] proposed a new technique to authenticate images using a semi-fragile watermarking based on the principle of similarity in the image itself. This is done by representing the image with finite-state machines. The basic idea behind these automata is to locate parts of the image that are identical or very similar to the whole image or to other parts of it, and then to build the graph of automata that shows these relations [17]. For a given image I, the representation in the automata domain is carried out to determine the degree of similarity between its different parts. The resulting graph is analyzed. A shape, based on the number of edges associated with each state is generated and used as an authentication watermark for this image. This watermark is thereafter dissimulated in the original image using methods for data dissimulation. The approximation between the different parts of the image is determined using a parameter α that controls the approximation degree and by consequence controls the system fragility. Given two multi-resolution images $f1$ and $f2$ of the automata graph, the difference $d_k(f1, f2)$ between two image blocks is computed by the formula:

$$d_k(f1,f2) = \frac{\sum_{w\in\sum^k} \left| f1(w)_k - f2(w)_k \right|}{2^k.2^k} \leq \alpha \tag{2.16}$$

Where, k is a positive integer. To verify a suspect image, the automata graph is first generated. The verification is done by the inspection of the difference between the resulting watermark and the stored one. This new technique showed interesting results since it was able to detect and localize malevolent manipulations and at the same time resist some compression ratios. Furthermore, by varying the parameter α, this method became very sensitive and fragile at the limits. This could be used to resist other content preserving manipulations by using a large value for the parameter α. For additional improvements, another type of automats based on generalized finite-state machines [18] can be used. The main advantage of this improved method is that it results in a more compact watermark as the automata requires fewer transition states. However, the algorithm is not able to restore the damaged data. Moreover, this method could become less interesting when an image presents few self similarities, which is more likely in practice.

Recently, Lee and Jang [63] proposed a method that is robust against JPEG compression. The proposed algorithm is based on random mapping, which randomizes in a secret way an image to prevent the VQ attack. The image to be authenticated is divided into 4×4 blocks. The block's order is randomized with a secret key. A predefined watermark is embedded in the reordered blocks to increase the system security. The largest single value (SV) of an 8×8 image block is modified with some quantization step in order to embed a watermark bit. The authors proposed adjusting and dithering quantized value in order to prevent histogram analysis attack. Dithering is achieved by adding image dependent uniformly distributed random noise. The blocks are then rearranged adequately to obtain a watermarked image. Results show high tamper detection and localization capabilities. Moreover, some robustness against JPEG compression was demonstrated. However, the algorithm has no restoration capabilities and robustness against other content preserving manipulations is not provided.

Image authentication by semi-fragile watermarking retained the attention of researchers because the developed methods could tolerate some content preserving manipulations while

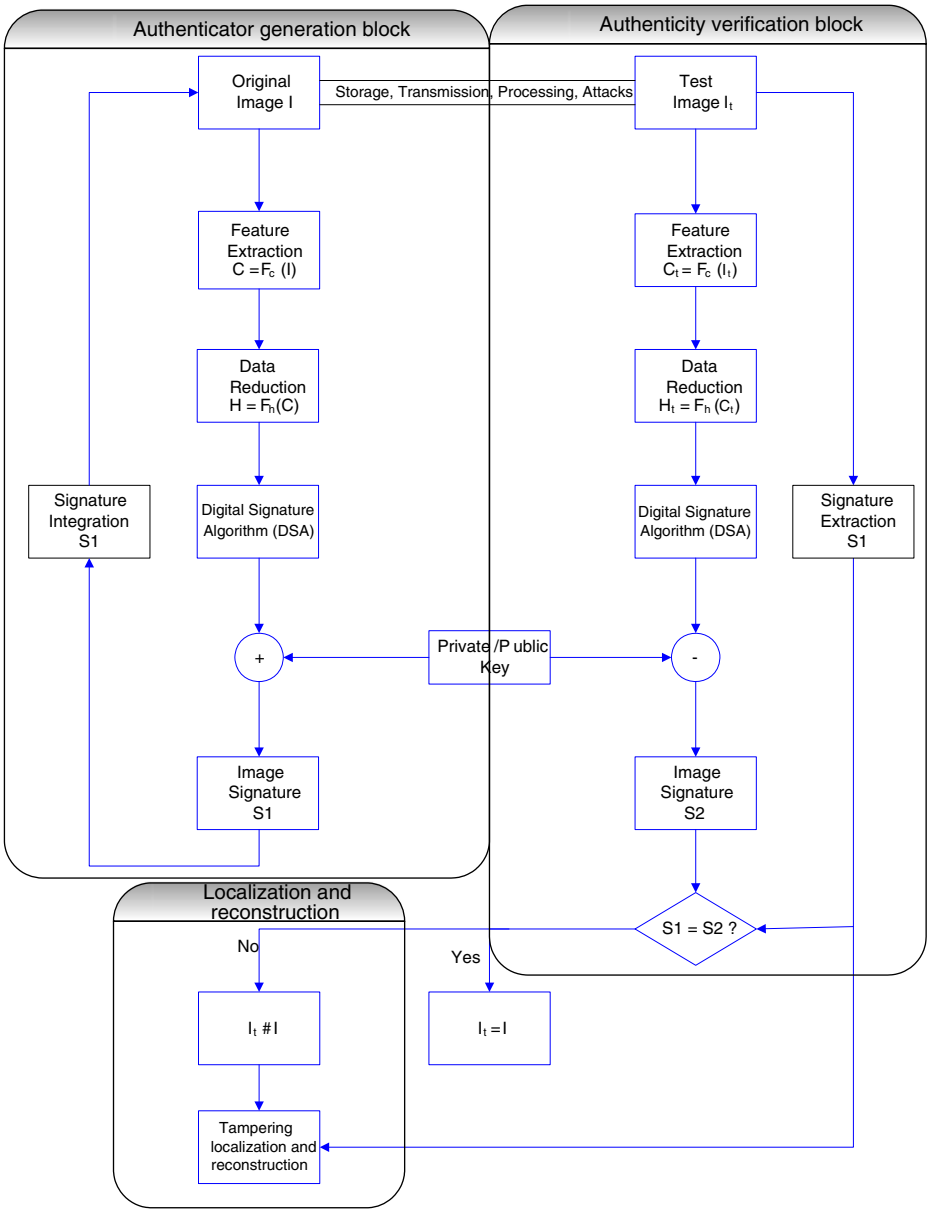

Fig. 5 Selective authentication system by a digital signature that is based on the image content

detecting malevolent operations. This flexibility is the result of two factors. First, the majority of these watermarking algorithms are of targeted robustness, as the resistance of the proposed techniques is specific to predefined manipulations. Second, the recently proposed methods use 'relatively' high level authentication watermarks that are rather based on the image content than on pixel intensities. The use of the content information for authentication increases the robustness of the system and the possibility of corrupted regions restoration. However, the main disadvantage of semi-fragile techniques is their

limited tolerance against combinations of content preserving manipulations. In fact, the majority of these methods are successful even in the presence of image processing operations that preserve the image content such as JPEG compression, filtering such as mean and median, noise addition and rotation. However, other manipulations that are typically used in image processing such as transmission or storage errors, geometrical transformations, and gamma or brightness correction need to be tolerated by authentication methods as well. Therefore, other techniques were investigated. They are known as image authentication algorithms by digital signatures that are based on the image semantic content.

2.2.2 Image authentication by digital signatures based on the image content

Most recent investigations in the domain of image authentication were concentrated on digital signatures applied to the image content; these approaches offer high performance and promise additional breakthroughs in the near future.

Generic diagram of an authentication system based on image content Image authentication systems that use a digital signature based on the semantic content of images could be described in a generic diagram (Fig. 5). Such systems consist in (1) extracting specific high-level characteristics from the original image; (2) applying a hash function to these characteristics in order to reduce their size; (3) digitally signing the hash value using an existing digital signature algorithm such as a private or public key system to increase the overall security; (4) attaching the signature to the original image or inserting it in the image using techniques for data dissimulation. Likewise, the verifying procedure of an image authenticity consists in (1) generating the image signature using the same algorithm; (2) extracting the attached or dissimulated signature; (3) comparing these two signatures using a comparison algorithm to decide whether the image was altered or not; (4) determining the image regions that were manipulated. When the image is declared as not authentic, information from the original signature could be used to partially or even completely restore the regions that were corrupted.

Several parameters directly affect the performance of an image authentication system based on image content signature. These parameters include the choice of the appropriate characteristics, the choice of the hash function and the digital signature algorithm, the choice of the data dissimulation method in images as well as the choice of the algorithm that compares the signatures to decide about the authenticity of an image. Among these parameters, the image features that represent the image content and the data dissimulation method mostly affect the performance of image authentication methods. In fact, sensitivity, robustness, recovery, portability, safety and complexity (defined in Section 2) are directly affected by the choice of the characteristics that are used to generate a content-based signature; they are affected as well by the choice of the data dissimulation method. The hash function and the digital signature algorithms are almost the same for all techniques. The algorithm used to compare the signatures directly depends on the selected characteristics and the dissimulation method. Therefore, we will use these two parameters, the choice of the appropriate characteristics and the data dissimulation algorithm, to classify and compare existing image authentication systems based on image content signatures.

Content-based characteristics extraction Signatures are generated from significant characteristics that represent the image semantic content. This is a great challenge since there is no unique explicit definition of the image semantic content. In the literature, image

characteristics are usually selected to maintain invariance to desired modifications in an image and to break invariance to malevolent manipulations. These characteristics typically include edges, colors or grey levels, histograms, DWT or DCT coefficients, textures, statistical measurements and some combination of them. The basic problem resides in the fact that any single characteristic, considered alone, cannot describe the semantic content of an image completely.

One of the first efforts to exploit image content signature was proposed by Schneider and Chang [118]. The authors used the image histogram to represent the image content as this information is likely to change with the content. However the histogram of the whole image is not very useful, because it does not contain local spatial information of image intensities. Therefore, the original image was subdivided into blocks of equal dimensions and the histogram of each block was computed separately. This permits the incorporation of spatial information in the signature, since the localisation of these blocks is fixed. The proposed approach resulted in a long signature that required a long computational time. An improvement of this method has also been proposed. It consists in varying the block dimensions according to the detail distribution within the image. The regions with finer details were protected with small-size blocks, whereas the regions with large details were subdivided into blocks of bigger sizes. This improvement decreased the computational time but preserved the detection and localization capabilities of the algorithm. However, the regions identified as corrupted could not be restored.

The histogram approach suffers from two major disadvantages: (1) it is robust against compression of small rates only; (2) histogram is not very representative of image content because image content can be changed without changing the image histogram. Despite its disadvantages, this is the first proposed method to generate a signature based on image content.

Image edges give relatively good information about the image content because they allow the identification of object structures [6, 11]. Several researchers used edges to generate image content signatures. Queluz proposed in her work [103], a method that consists in obtaining a binary image describing the pixels with or without edges. To carry out this process, the gradient was first computed in each pixel using the Sobel operator [6], and compared with a threshold determined from the gradient histogram [97]. The use of Canny operator [11] provided better results with the advantage of being less sensitive to noise but with a higher computational cost. The bitmap of the binary image, resulting from the edge detection is compressed thereafter. Depending on its size, the bitmap could be down-sampled. This bitmap was then encoded with one of the lossless compression techniques for binary images such as the Modified READ [105, 128] or JBIG [28, 44, 66]. The final result forms a bit sequence that was used to generate the image content signature. This algorithm allows good detection of image malevolent manipulations with relatively high localization performances. Unfortunately, the algorithm is not able to reconstruct damaged data. Edges are relatively a good choice for image content authentication, but methods based on edges suffer from three major problems: (1) the generated signature is long; (2) the signature depends on the edge detector; different edge detectors give different results for the same image; (3) the invariance against color manipulations. In fact, image color can change without affecting its edges. Moreover, it is noteworthy to mention that JPEG compression introduces a well-known edge distortion the ‘mosquito noise’ effect [60] that results in edge orientation ambiguity after compression. To avoid this problem, the author suggested using this information as an a priori knowledge in order to tolerate some JPEG compression ratios.

Similarly, Dittmann, Stabenau and Steinmetz [20] have proposed a method based on determining the image edges and transforming them into a characteristic code to generate

the image content signature. The authors used the Canny detector to compute the edges. The result, a new image C, called edge characteristics, was then transformed into binary edge characteristics, called a binary characteristic shape, which was compressed with the variable coding length to produce a code ready to be signed by a digital signature algorithm. The authors did not detail the binary characteristics shape generation procedure. Therefore, the restoration capability of the algorithm could not be well evaluated. Detection and localization performances are very satisfying. However, likewise all other methods based on edges, this method suffers from the same problems mentioned above. Moreover, its robustness against compression was not well demonstrated.

Most recent efforts in feature extraction were dedicated to methods operating in the frequency domain, more precisely in the DCT domain. This high interest can be explained by the important role the DCT transform plays in many image compression standards, and by the fact that it is easy to compute and understand, and gives good frequency representation of the image content [107]. Several authors used the DCT coefficients to generate characteristics able to survive specific desired manipulations while allowing for alteration detection.

One of the first attempts based on the DCT domain for characteristic extraction was proposed by Storck [124]. In his method, the author exploited the fact that low and middle frequency DCT coefficients were generally affected by changes to the original image content, whereas the high frequency coefficients were less affected. The author proposed to use the DC and the first five AC coefficients, obtained according to the zigzag order used in JPEG compression, to generate the image signature. In order to tolerate image scaling, the author added the original image dimensions to the signature information. The selected coefficients for each DCT block were encoded and a hash value was generated by applying MD4 or MD5 hash function. The dimensions of the original image were also encoded and appended to the code before applying the hash function. A public key system, such as the RSA, was used to generate the signature from the hash result. The authentication results obtained with this method were robust against compression based on the DCT that is used in the JPEG standard. Moreover, high detection and localization of image malevolent manipulation were possible. However, this method robustness against other content preserving manipulations such as filtering and gamma correction was not shown nor tested. The algorithm didn't propose any reconstruction performances since it is not able to restore image regions that were detected and localized as tampered. The use of traditional hash functions increased the system security but introduced the risk of increasing the system sensitivity to small modifications that preserve the image content.

Similarly, Wu proposed an algorithm [151, 152] that separated the original image I, into blocks of 8×8 pixels and calculated the DCT transform for each block. DCT coefficients were then quantized by applying the JPEG quantization table [131]. This table exploits the human psycho-visual model to minimize the visual distortions. The quantized DCT coefficients were then grouped to form a characteristics vector V. For each value in the vector V, a quantization function q_0 or q_1 is selected by minimizing the quantization error $|q(x) - x|$ where x is an element of V. These two quantization functions q_0 and q_1 are:

$$\begin{aligned} q_0(x) &= \text{round}(x) \\ q_1(x) &= \text{round}(x + 0.5) - 0.5 \end{aligned} \tag{2.17}$$

Where round (x) denotes the nearest integer to x. An index vector X is generated. For each DCT coefficient from vector V, an index bit is assigned in X; a bit '0' is assigned if q_0

is selected whereas a '1' is assigned if q_1 is selected. The vector X, contains as many bits as there are pixels in the original image I. For color images, the vector V is computed from the DCT coefficients of the color components, and the number of bits in the vector X is three times the number of pixels in the color image. Vector X is added to the quantified coefficients of characteristic vector V to form a new vector W. Vector W is signed with a digital signature algorithm such as the DSA to generate the signature S. The index vector X is compressed by a lossless compression algorithm like the Huffman coding. This compressed version of X is appended to the signature S to form the final authentication tag T that is dissimulated or attached to the original image. The use of the DCT coefficients and the quantization table ensures high robustness against JPEG compression even with big compression ratios. The algorithm shows also a tolerance against slightly brightening processing. The possibility to use this method for color images is yet another advantage. Image manipulated regions could be well detected and localized by the proposed algorithm. The reconstruction performances were not highlighted by the authors. The security of this method is based on the security of the signature algorithm. However the robustness of this method against other content preserving manipulations such as filtering and affine transformations was not demonstrated.

Then again, Lin proposed a very interesting method in [68] that was further improved in [69, 70, 72, 106]. This method divides the image in 8×8 blocks; the DCT of each block is calculated separately. The blocks are then separated in two groups P1 and P2; for example: even blocks '2, 4, 6...' in a group and odd blocks '1, 3, 5...' in the other group. The coefficients of each DCT block are arranged according to the zigzag order used in JPEG standard compression [107]. A subset ($b1$) of the DCT coefficients is then chosen in each block. These coefficients have the same positions in all blocks and are used to generate N sets of feature codes Z_n, $n=1, 2..., N$ in order to build the signature. To generate the first set Z_1 the differences between coefficient pairs from the same positions for all possible combination of block pairs are computed. Z_1 contains the result of comparing the differences with a predefined threshold k_1. If, for example, the computed difference dx, is smaller than k_1, a bit '1' is assigned in the Z_1 vector, otherwise a bit '0' is assigned. For each block pair a feature code of length b_1 is obtained, and this procedure is repeated for all possible combination of block pairs from groups P1 and P2. To generate the second set of feature codes Z_2 the same algorithm is repeated with different parameters: b_2 different from b_1 and k_2 different from k_1. In other words, for each set Z_n, the number of selected coefficients b_n and the threshold k_n are different. After generating all sets Z_n, they are grouped to form the final vector Z. The author suggests selecting the coefficients b_n in the low and middle frequencies since these coefficients have larger values than those in the high frequencies and their corresponding quantization values are smaller so they usually survive a JPEG compression. Thresholds k_n can be arbitrarily selected. Moreover, other information can be recorded in the vector Z such as the mean value of DCT coefficients in each selected position for all blocks in order to defeat a constant change to DCT coefficients in the same position for all blocks. This vector Z is finally signed using a public or private key system digital signature algorithm. The author exploited the fact that all DCT coefficient blocks of an image are divided by the same quantization table during the process of JPEG compression [131], so the relation between two coefficients of the same position in two different blocks is not changed after quantization. This simple remark ensures a very great robustness against JPEG compression. Furthermore, it ensures a great sensitivity against content changing manipulations, since any addition or removal of an object is likely to be reflected in the DCT coefficients. The localization performances of the algorithm are very satisfying. Moreover, the algorithm is

able to reconstruct the damaged data using the dissimulated information that is significant. The security of this method can be increased by keeping secret the mapping function used to separate the blocks in two groups and the criteria used for the selection of the coefficients subsets b_n. However, the robustness of this method against other content preserving manipulation was not shown. Moreover, this method could have a problem of localization ambiguity which results from the combination of separates image blocks to generate the signature.

Alternatively, the DWT has been successfully used in various image processing applications including filtering, multi-resolution analysis and image compression [2, 42, 90]. This is due to its capacity to characterise the image content both in the spatial and the frequency domains. Recently, the interest in the DWT for extracting content-based features has largely increased. One technique proposed by Sun and Chang [126] starts with a filtering algorithm called Bayes Shrink in order to stabilize the features that will be extracted with respect to some desired manipulations. The proposed filtering algorithm, described in details in [12], gives a stable content representation of the image to protect by reducing from 1 up to 8 dB the distortions that are caused by content preserving manipulations such as filtering or compression with various compression standards and ratios. After filtering, a threshold is applied to all DWT coefficients to generate a 'binary significant map'. This threshold affects the sensitivity of the system since a lower threshold leads to a sensitive system. A morphological conditional dilation [119] is applied to the binary significant map according to: $(S \oplus B) \backslash B$, where S represents the significant map, B represents the structuring filter and $\oplus$, $\backslash$ denote the dilation and difference operations, respectively. The morphological filtering is applied to reduce the noise level, and to increase the robustness against acceptable manipulations. To generate the signature, the author proposed to take into account not only the significant coefficients, but all their children as well, resulting in a binary significant map that contains more details. Error correcting code (ECC) encoding [74, 125, 143] is applied on the binary data to obtain signature sets, one per block; blocks enable good localization of manipulated regions. This method is robust against compression of different standards and ratios, as well as against filtering. Detection and localization of content changes had been demonstrated. However, algorithm robustness against other content preserving manipulations needs be demonstrated. Moreover, the algorithm is not able to reconstruct damaged data.

The authors recently proposed a new scheme [127] capable of resisting all distortions that may be introduced by JPEG2000 compression. The feature extraction is applied after the EBCOT (embedded block coding with optimized truncation) process. EBCOT is the last encoding process in JPEG2000 compression standard. The idea is that the resulting information from the EBCOT process will not undergo any modifications in the compression process. Furthermore, this information is significant and contains image content details. In fact, this information is taken generally from different levels of DWT and not only from one level. This encoded information is encrypted with ECC algorithm to form a digital content-based signature. The detection and localization performances are very satisfying. Tolerance against JPEG2000 standard is very high even for color images. The main disadvantage is tolerance against other content preserving manipulations. The restoration capability of this algorithm is still to be demonstrated.

In another work, Lu and Liao [78] proposed a technique that exploits the relationship between the coefficients of an orthogonal DWT at various scales. If $w_{s,o}(x, y)$ is a DWT coefficient at scale s, orientation o (horizontal, vertical or diagonal), and at position (x, y), when another scale J is generated, where $0<s<J$, the inter-scale relationship between the

DWT coefficients for the parent node $w_{s+1,o}(x, y)$ and its four children nodes $w_{s,o}(2x+i, 2y+j)$ is given by:

$$\left|w_{s+1,o}(x,y)\right| \geq \left|w_{s,o}(2x+i,2y+j)\right| \text{ or } \left|w_{s+1,o}(x,y)\right| \leq \left|w_{s,o}(2x+i,2y+j)\right| \quad (2.18)$$

Where $0 \leq s < J$, $0 \leq i, j \leq 1$, $1 \leq x \leq N$, $1 \leq y \leq M$ and M, N are the image dimensions. The basic idea behind this technique is that the inter-scale relationship between the coefficients is difficult to break by manipulations that preserve the image content, and hard to preserve by content changing manipulations, because these inter-scale relationships depend on the image structure. In this method, the structural digital signature (SDS) is made up of parent/children relationships that satisfy the following relation:

$$\left\|w_{s+1,o}(x,y)\right\| - \left\|w_{s,o}(2x+i,2y+j)\right\| > \rho \quad (2.19)$$

Where $\rho>0$ is a predefined threshold. To generate the SDS only significant parent–descendant pairs are considered. Pairs that have small differences are very sensitive to content preserving manipulations and are not considered for the signature generation. The relationships are encoded with an encoding algorithm such as the RSA and used as the signature. The length of this signature depends on the threshold ρ that also plays the trade-off role between system robustness and fragility. This method is robust against compression, mean and median filtering, and even noise addition. Moreover, it is fragile against any image content alteration. The algorithm is able to detect and localize any image alteration. However, it is not able to restore the data that was declared as corrupted. The fragility and the robustness depend heavily on the signature length. Better detail preservation is attained with longer signature that retains more features. Therefore, it is not obvious how to automatically identify the threshold ρ for any image.

In order to increase the robustness against compression and noise addition, Feng and Liu presented an algorithm [26] that generates a SDS in a more straightforward manner. This method, called segmentation derived attribute graph, is based on the spatial distribution of homogeneous regions. This algorithm is considered to be used by the human eye in its first-perception of the image content. While defining the graph, frequency features such as DCT or DWT coefficients can be integrated to form a robust SDS. The authors proved that the derived attribute graph could resist compression and rotation in a robust manner for large ratios and angles. At the same time, it could eventually detect and localize image tampering. The problem of restoration disability however, was not solved.

Bhattacharjee and Kutter [7] proposed a technique that extracts the salient feature points from a given image. The extraction procedure was inspired by Manjunath and Chellappa [82, 161] who utilized it for face recognition and movement correspondence using the Gabor wavelet. This technique defines a features detection function P_{ij}:

$$P_{ij} = \left|M_i(x) - \gamma M_j(x)\right| \quad (2.20)$$

Where $M_i(x)$ and $M_j(x)$ represent the Mexican-Hat wavelet response at position x and scales i and j, respectively. The normalisation constant γ is given by $\gamma = 2^{-(i-j)}$. Instead of

using the Gabor wavelet, the authors proposed to use the Mexican-Hat wavelet, which is isotropic and is given by:

$$\phi(x) = \left(2 - |x|^2\right) \exp\left(-\frac{x^2}{2}\right) \tag{2.22}$$

The local maxima points of the P_{ij} function are computed. These maxima correspond to potential feature points. The local maxima of P_{ij} are calculated within a circular neighbourhood having a radius of 5 pixels. A local maximum is accepted as a feature if and only if the variance of its neighbours is larger than a given threshold. The variance is calculated within a circular region that may have a radius different from the one that was used for local maxima computing. The resulting feature points are arranged by line and column in order to facilitate the localization of image distortions. A sequence of bits is built by the concatenation of the line and column positions of the maxima. This resulting sequence is finally ready to be signed by a digital signature algorithm such as the RSA. This technique can be used with color images by applying the described procedure on each color component. This method is robust against compression, detects and localizes any content change with acceptable precisions. However its robustness against other content preserving manipulations has not been demonstrated. Moreover, the algorithm is not able to restore the altered regions in images.

Monga and Vats tested the same algorithm with another wavelet function [94]. The results were very similar. Some improvements are related to the robustness against different content preserving manipulations. This includes some geometrical transformations and some image enhancement operations. The new proposed algorithm preserves its capabilities of detecting and locating any intentional manipulation.

In a more recent work Liu and Hu proposed a method based on DWT coefficients quantization [76]. In their work they apply the just noticeable distortion values (JND) [2] to the image DWT coefficients. The result is quantified and rounded to obtain a {0, 1} sequence. This sequence is then encoded to form a digital signature. The algorithm is valid with 2 and 3 decomposition levels but yields to a trade-off between security, complexity and time processing. The algorithm is based on a threshold to decide whether the image was manipulated or not. The threshold is defined with the help of probability. The authors demonstrated the detection and localization performances of the algorithm. Moreover, a good robustness against compression was attained. However, other content preserving manipulations were not tested and the algorithm is not able to restore the damaged data.

Another recent algorithm, proposed by Yu and Hu [49, 160], uses a three level Haar DWT. The transform is applied to a $M \times N$ image I. A de-noising algorithm is applied to LL3 component coefficients. The Sobel operator is then applied to the coefficients to get a $1/8M \times 1/8N$ binary Sobel edge map WL. Md5 is applied as a hush function on the binary map. The result is a bit array of length 128. The output is then mapped to a $1/8M \times 1/8N$ binary matrix Wa using a secret key. To increase the system security a private key could be used to encrypt Wa and produce We. For verification, the extracted WL and Wa watermarks are both compared to the original ones. Using a predefined threshold, one can decide about the authenticity of the tested image. This technique shows very interesting results. It could tolerate both JPEG and JPEG2000 compression with various ratios, filtering and histogram equalization. Moreover, detection and localization of manipulated regions are guaranteed with acceptable precision. However, the use of the threshold depends on images and applications. Thus, the choice of the threshold could be image adaptive. Moreover, tolerance against other content preserving manipulations as

geometrical transformations is not discussed. The algorithm is not able to restore the data that was altered.

Another interesting and intuitive technique that exploits the image statistics exhibits good results. Kailasanathan, Safavi-Naini, and Ogunbona [53] proposed to divide the original image I_0 of dimensions $m \times n$ into $a \times b$ blocks and to compute the following statistical features for each block:
Mean of each block:

$$\overline{x} = \frac{\sum_1^n x_i}{n} \tag{2.23}$$

Standard deviation of each block:

$$D = \frac{\sum_1^n (x_1 - \overline{x})^2}{n} \tag{2.24}$$

Kurtosis of each block:

$$K = \frac{m_4}{m_2^2} \tag{2.25}$$

Skewness of each block:

$$S = \frac{m_3}{\sqrt{m_2^3}} \tag{2.26}$$

Where $m_i = \frac{\sum (x_j - \overline{x})^i}{n}$ and n is the pixels number in each block.

These statistics are computed for an image that has been altered with acceptable transformations such as JPEG compression, filtering, and scaling. The magnitude of the differences between the statistics computed for the modified image and those computed for the original image are grouped, and the maximum of all these differences is identified. This maximum represents the threshold for the authenticity verification, or in other words the threshold for acceptable manipulations. The feature code for a given image is made up of the statistical features computed for each block, the block dimension, the type of statistics and the computed threshold. The block dimensions (a, b) must be chosen such that a compromise between a good image representation and an acceptable feature code length is reached. The signature is thereafter generated from this feature code using a digital signature algorithm. The main advantage of this method is its acceptable performance in restoring the regions that have been altered. Moreover, it can tolerate a set of acceptable predefined manipulations. Detection and localization of undesirable manipulation are also shown. However, it hardly tolerates various simultaneous manipulations for the same image.

Another method, which exploits the image statistics to generate a signature, was proposed by Lou and Liu [77]. To obtain the statistical characteristics, the original image is subdivided into 8×8 blocks; the grey level mean is computed within each block separately and is used as the authentication information. To decrease the characteristics code length that is composed of all block means, the authors proposed to quantify them and to add the threshold T to the code. A public key system is used to generate the image digital signature. Robustness to compression is obtained by determining the threshold T for a compressed image with a maximal allowed compression ratio. The procedure starts by selecting the tolerated compression ratio. This compression ratio is used to compress the original image. The block means for the compressed image are computed. For each pair of compressed image block mean and quantified mean, the difference T_{ij} is computed. The threshold is chosen such: $T = \max |T_{ij}|$.

The principal advantage of this method is its restoration capabilities of the corrupted image regions. Corrupted region recovery is achieved by replacing the altered pixels with their block mean value. This recovery method is simple and intuitive. Moreover, detection and localization of corrupted regions is guaranteed. However the robustness of this method against other manipulations has not been proved.

On the other hand, Tzeng and Tsai proposed in their work [138] a method to generate an image digital signature using both edges and statistical characteristics. For a given image, decomposition into blocks is carried out and a method for detecting significant edges is applied. The main goal of significant edge detection is to classify each block according to two types: smooth block or edge block. Edge detection is accomplished with a Sobel operator to obtain a binary edge image. If the Sobel gradient in a pixel is larger than a predefined threshold T, its corresponding location in the binary image is set to '1', otherwise to '0'. The threshold T can be computed automatically with histogram thresholding [119]. Both the binary edge image and the original image are subdivided into blocks of 8×8 pixels. A block B_k is classified as edge block if it contains at last one connected element larger than 4 pixels; otherwise, B_k is classified as a smooth block. Thereafter, smooth blocks are presented by their means and standard deviations whereas edge blocks are transformed into binary blocks using a threshold technique that preserves image moments [136]. Each binary block is further subdivided into four sub-blocks that are compared with a predefined edge form of the same size, to generate a bit sequence that represents the edge information. The edges bit sequence, the means of the smooth block and the standard deviations form the characteristics code are used to generate the image digital signature. For color images, the method can be used to generate characteristics code from the brightness and the chrominance components. The algorithm shows good results. The detection and localization of alterations are very acceptable. Moreover, the altered regions can be recovered with acceptable quality but only for blocks that had been classified as smooth. Robustness of the algorithm against lossy image compression is guaranteed only for a specific range of ratios. Robustness against other content preserving manipulations is still to be explored.

Another approach based on structural and statistical features of an image was recently proposed by Li and Tang [130]. The principal component analysis (PCA) is applied to block DCT coefficients in order to compute the Hotelling's T-square statistics (HTS). HTS are values that represent some distance metric between DCT coefficients within each block and the whole image. Each element of the HTS matrix is quantized by applying an algorithm based on the maximum and minimum values in that matrix. The quantized matrix is then grouped to form a statistical structural fingerprint (SSF). As most HTS elements are 0 or 1 after quantization, more compact SSF is obtained by the Huffman coding. Results show a high detection and localization of most known tampering operations. Moreover, a good robustness against JPEG compression and some filtering algorithms was demonstrated. The algorithm is simple to implement. However, the robustness against others content preserving manipulations was not discussed. Moreover, the algorithm did not show any restoration capabilities.

In her work [104], Queluz, proposed a technique based on the invariance of the image moment against geometrical transformations. In fact, image moments are invariant to translation, scaling and rotation [8, 51, 108]. For digital images, moments are computed by:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y) \tag{2.27}$$

Where x, y are the image pixel coordinates, $f(x, y)$ denotes the intensity and p, q are the moment order. Usually, an image moment is normalized by dividing its value by the image total mass: $\sum\sum f(x,y)$.

The lowest-order moment that depends simultaneously on grey levels and their spatial positions is the second-order moment m_{11}. This moment computation is easy and requires N^2 multiplications for an $N\times N$ image. To increase moments uniqueness and their discrimination capability as well as to decrease the collision probability, moments are calculated for separate and overlapping blocks. With half block overlapping, each pixel contributes simultaneously to four different blocks, with various weights [different coordinates (x, y)]. Thus, after subdividing the original image into overlapped blocks, the second-order moments m_{11} and their relative variance m_{var} (%) are computed for each block. The variance is used as a distance measure between the original image moments, and those extracted for the verification of authenticity. Additionally, a threshold T to decide about the authenticity of the image according to that distance is defined. This method is robust against geometrical transformations because image moments are invariant to these content preserving manipulations. Moreover, the algorithm shows good detection and localization performances of the corrupted regions. However the robustness of this method against compression still needs improvements and the reconstruction of corrupted regions is not defined.

Data dissimulation in images Signature dissimulation is an important issue that differentiates image authentication systems. In fact, the digital image signature computed with one of the previously presented techniques needs to be attached or dissimulated in the image in order to verify its authenticity afterwards or to provide information for reconstruction purposes. Thus, an overview of the data dissimulation methods is necessary. Detailed information on data hiding can be found in [52, 101].

Data hiding is a general term encompassing a wide range of problems beyond embedding messages in content. In fact, it includes watermarking and data security. Watermarking algorithms seen in Section 2.2.1 can be used to dissimulate signatures in images for selective authentication proposes. In fact, fragile watermarking techniques do not tolerate content preserving manipulations, and the dissimulated signature risk to be destroyed only by a compression or some filtering algorithm for example. Thus, algorithms that dissimulate data in images in a fragile way are helpful for strict authentication and not for selective authentication. In this section we present other existing methods and algorithms which offer this service and we compare their performances. Data dissimulation methods and algorithms can be separated in two groups: methods that insert the signature in the image in a way to avoid any attempt to remove it by malevolent or by desired manipulations, and methods that attach the signature to the image or to a separate file. Signature dissimulation in images is achieved in the spatial or the frequency domain.

The simplest and most intuitive method for data dissimulation in images is to use the spatial domain [52]. Grey levels or the color image brightness are usually coded on 8 bits for 256 possible values. Spatial domain data hiding is based on the fact that the human eye cannot differentiate between two consecutive levels. That is the change in the least significant bits (LSB) is not noticeable or imperceptible to the human eye. Consequently, the LSB are considered not very important. So a binary image, a logo for example or a signature in a binary format, can be hidden in the LSB. However, the human eye reacts differently according to contrast. Human perception is very sensitive to contrast for low intensities and much less sensitive for intensities close to white. Therefore, some methods adapt the number of LSB used for data hiding according to the local contrast. Spatial data hiding methods suffer from many problems: Hidden data is very easy to remove by putting

all LSB to '0' for example. Moreover, all spatial filtering operations are drastic; they may erase the hidden data almost completely. Compression, for example, does not leave any survival chance to data hidden with this method. An improvement is achieved with the use of a pseudo random number generator based on 'a secret sequence' to determine the pixels that are used to dissimulate the data. For some color images, GIF for example, this method can damage all the image colors since a color lookup table defines the pixel color.

Kutter, Jordan, and Bossen proposed a technique to dissimulate the signature in color images [58] by inserting it in the spatial domain of the blue color component because the human eye is less sensitive to blue. Furthermore, in order to increase the security of this method, the authors proposed to use a pseudo random sequence of pixel positions where to hide the data. For each pixel at the selected position the following transformation is applied:

$$\mathrm{Blue}(i,j) = \mathrm{Blue}(i,j) + (2s-1).L(i,j).q \tag{2.28}$$

With $L(i,j) = 0.299\,\mathrm{Red}(i,j) + 0.587\,\mathrm{Green}(i,j) + 0.114\,\mathrm{Blue}(i,j)$, s is the bit to dissimulate, and q is a constant that represents the dissimulation strength. This method works well but can only be applied on RGB images.

The main advantage of spatial domain methods is their acceptable dissimulation capacity. However, they are not robust because they are vulnerable to various attacks [101] and are not secure. In consequence, methods that exploit the frequency domain to dissimulate data have been investigated. The most popular frequency transforms to dissimulate data in images are the FFT, DCT and DWT. The DCT allows an image to be decomposed into low, middle and high frequencies. The middle frequency bands are usually used to dissimulate information because low frequencies contain visually sensitive parts of the image and high frequencies are changed or removed by compression and noise [62]. One of the data dissimulation techniques in the DCT domain is based on the comparison of the DCT coefficients in the middle frequency band of each block. In order to achieve robustness against compression, two locations Bi($u1$, $v1$) and Bi($u2$, $v2$) are selected according to their quantization values from the quantization table used in compression standards [99] so that any change to one coefficient by a factor will change the other coefficient by a similar factor, preserving thus their relative value. '1' is encoded if Bi($u1$, $v1$)>Bi($u2$, $v2$); otherwise '0' is encoded. The method robustness can be improved by introducing a constant K, such as: Bi($u1$, $v1$)−Bi($u2$, $v2$)>K. By increasing K the probability of error detection is reduced. Contrary to methods based on spatial domain, this method shows a big robustness against JPEG compression and noise addition. However the information that can be dissimulated in the coefficients of middle frequency is limited and the security is not high.

Another domain to dissimulate data is the wavelet domain. The DWT decomposes an image in a low-resolution (LL) approximation, horizontal (HL), vertical (LH) and diagonal (HH) details. The process can be repeated to compute multiple "scales". One advantage of the DWT is that it models the human vision system better than other transforms. Data is dissimulated in regions where the human perception is less sensitive. A technique, which is proposed, by Xie and Arce [156], uses the low frequency band of the DWT to ensure the robustness of the method. A window of dimension 3×1 is slipped throughout the low frequency band of the image transform. The elements of the window are noted by $b1$, $b2$, $b3$ and represent the values of the coefficients with the coordinates $(i-1,j)$ (i,j) $(i+1,j)$. The coefficients are ordered after, in such way so that $b1 < b2 < b3$. A non-linear transformation [155, 156] is applied to change the median value of these coefficients while leaving remaining coefficients unchanged. The non-linear transformation is done by the following

equation: $b'2=f(b1, b3, x)$. Where x is the bit to be dissimulated. Another technique hides data in frequency bands with more details following the expression:

$$I_{W_{u,v}} = \begin{cases} W_i + \alpha|Wi|x_i; u, v \in HL, LH \\ W_i; u, v \in LL, HH \end{cases} \tag{2.29}$$

Where W_i is the DWT transform coefficient, x_i is the dissimulated bit, and α is the strength of hiding.

All data dissimulation methods suffer from three major problems: robustness, security and capacity. These methods must be robust against all manipulations, including content preserving and changing manipulations, since the dissimulated signature travels with the image and must remain unchanged in order to verify the image authenticity and eventually help in the restoration of image corrupted regions. Manipulations that result from standards image processing operations such as compression, rotation, filtering, scaling, and sampling [21, 100] form a category of attacks called distortion attacks. It seems that there is no data dissimulation algorithm that can resist all combinations of these attacks. The proof is given by the software StirMark [4, 149] created in 1997 and available for free on the Web. StirMark is dedicated to test any method of data dissimulation in images; it can simulate a large number of distortion attacks [139]. There is no data dissimulation diagram that could pass all StirMark tests [4]. Another type of attacks cause the image authentication system to take the wrong authentication decision [59, 87], is to believe that an image is authentic when its content has been modified or the opposite. Fortunately, this type of attacks cannot be always successful with digital signature based on image content. Nevertheless, perfect robustness of data dissimulation methods is still not reached.

On the other hand, information hiding capacity is a major concern because it is limited by the insertion algorithm and the dissimulation domain. The number of frequency coefficients or the number of LSB that can be used to hide information, without causing visual effects in the image, is limited. To avoid these problems, alternatives were proposed to ensure high signature security and to offer large hiding capacity. These solutions are based on attaching the signature to the image file or to a separate file. These solutions often use public or private key signing systems. Consequently, their security is directly affected by the security of the signature algorithms that is well studied in cryptography. Usually, when the signature is attached to the image, it is written in the comments zone of the image format. However, few image formats, used for image storage and transmission, have comment zone such as the comment marker segment for JPEG, the Image Tag Description for TIFF, or the application extension block for GIF [93, 132]. These zones can be used to attach the image signature but it is not guaranteed that image processing tools will preserve it once the data is stored back. Therefore, image authenticity can be lost, because the signature was not recorded by an application, even though the image did not undergo any content modification. An alternative solution consists in storing the signature in a separate file from the image. This technique is often used with a private key system to ensure a secure signature file transmission. It offers a quasi-unlimited dissimulation capacity but is far from being the best solution as two new constraints are to be taken into account: (1) ensuring the security of the signature which is separated from the image; (2) ensuring the couple image/signature authenticity, or in other words proving that a specific signature belongs to a specific image.

Table 3 Comparison between image authentication categories

Category	Authentication tag	Authentication tag dependency on image	Authentication service type	Localization capabilities	Tolerant t	Reconstruction capabilities
Conventional Cryptography	Lines and columns hashes [22]	Yes	Strict	With ambiguity	Not applicable	No
	Block hashes [67, 145, 157]	Yes	Strict	Yes	Not applicable	No
Fragile water-marking	Check sum [5, 17, 31, 142]	Yes	Strict	Yes	Not applicable	No
	Predefined logo [33, 147, 148, 150, 159]	No	Strict	Yes	Not applicable	No
	Color components [10]	Yes	Strict	Yes	Not applicable	No
	DCT [35]	Yes	Strict	Yes	Not applicable	Yes
Semi-fragile water-marking	*m*-sequences [134, 140, 146]	No	Selective	Yes but with some ambiguity	JPEG compression and noise addition	No
	Random-noise [13, 32, 57, 71, 84, 98]	No	Selective	Yes	JPEG compression	No
	Predefined mark [63, 163]	No	Selective	Yes	JPEG compression with small ratios	No
	Luminance [55, 56]	Yes	Selective	Yes	JPEG compression with small ratios	Yes
	Block similarity [120]	Yes	Selective	Yes	JPEG compression with small ratios	No
	VQ quantization [80, 81]	Yes	Selective	Yes	VQ based and JPEG compression, rotation	No
Content-based signatures	Histogram Edges DCT Wavelet Moments Statistics	See Fig. 6 for more details				

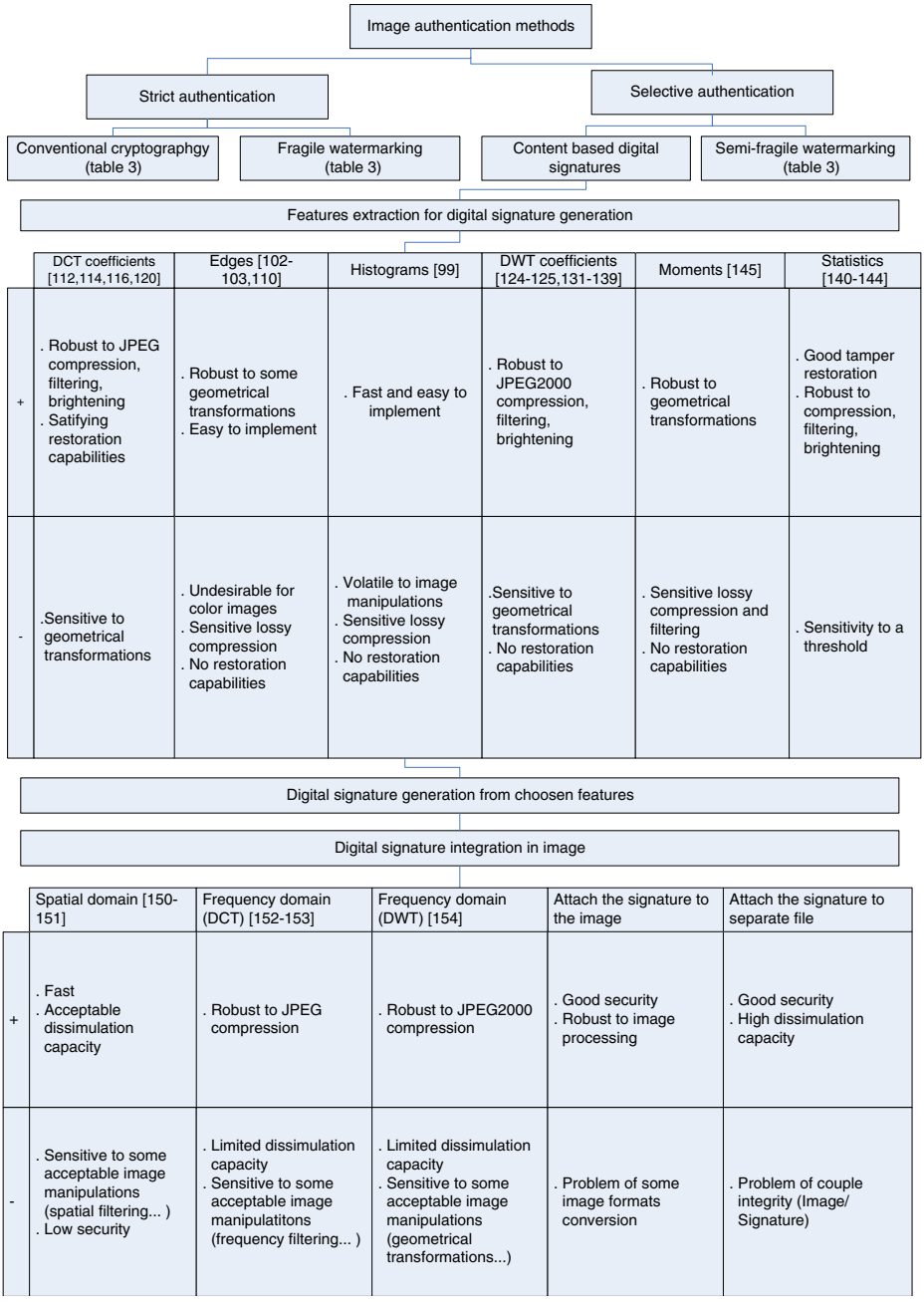

Fig. 6 Classification of image authentication methods; *plus sign* indicates advantages; *minus sign* indicates disadvantages

3 Summary

Table 3 presents a summarized comparison of image authentication methods discussed in this paper: methods based on conventional cryptography, fragile watermarking, semi-fragile watermarking and on image content signatures.

For each group of methods we have shown the type of the authentication tag, the dependency of this authentication tag on the image, the type of the authentication service provided, that is: strict or content-based (selective) image authentication service, the localization capacity of the altered regions, as well as the possibility of restoration of image corrupted regions. Algorithms are also grouped according to the authentication tag that is used, and references are included. It can be noticed that one principal property of an image authentication system, the detection of malevolent manipulations, is not included in this table for the following reason: All described methods can detect malevolent manipulations. Moreover, the robustness against content preserving manipulations is not offered by the first two categories since they provide a strict authentication services and do not tolerate any modification to the original image.

According to this summary table, algorithms performances are very similar. In fact, most of algorithms offer acceptable detection and localization of image manipulations while restoration performances still need to be improved. For strict authentication applications, where no modification to the original image is allowed, fragile watermarking algorithms perform better than algorithms based on conventional cryptography. Fragile watermarking algorithms offer high detection and localization capabilities. Moreover, some of them could provide an acceptable restoration level of damaged regions. On the other hand, selective authentication methods tolerate some desired manipulations while detecting any malevolent operations. Semi-fragile algorithms show good results for detecting and locating any malevolent manipulations while providing acceptable reconstruction performances. Unfortunately, their tolerance against desired manipulations includes mainly compression, noise addition and rotation by small angles, whereas, many of the desired manipulations need to be tolerated in practice. Since algorithms based on digital signature show more interesting results, we present them and compare their performances along with references in Fig. 6.

Figure 6 presents a classification of image authentication methods with a detailed comparison of signature content-based methods. The comparison is made according to two important properties: the domain from which features are extracted to provide a content-based signature and the domain used to dissimulate or attach this signature. Moreover, for the sake of simplicity, only the most important weakness and strength for each group are highlighted.

Every image-extracted feature used to generate the image signature has its weakness and force. The comparison of these features, their weaknesses and forces, help choosing the right method for a specific application. For example, if an application needs to tolerate compression with JPEG or JPEG2000 standard, the DCT domain or DWT domain, respectively, are best suited to generate the signature. If geometrical transformations need to be tolerated, the use of moments would be the best choice. If restoring the damaged data is important, statistical features could help well. Moreover, they are able to survive lossy image compression and a predefined set of content preserving manipulations (filtering, brightening...). On the other hand, using edges for content-based signature is undesirable for color images since one may change colors without affecting edges. This could result in an error where an image is declared authentic while some undesirable changes were introduced to it. Dissimulating signatures or attaching them to the image depends on the application and user requirements. A big dissimulation capacity and a high security can be achieved by attaching the signature to the image or to a separate file. However, the latter solution suffers from the problem of ensuring the couple image-signature integrity.

4 Conclusion

Internet allows people to communicate between various locations by exchanging very large volume of information. Moreover, access to this network is relatively possible for anyone thanks to the widespread of telecommunication technologies. Therefore, the volume of numerical data exchange is continuously increasing and new information security requirements are thus needed. Digital images do not escape from this phenomenon. Users expect solutions that protect the intellectual properties and ensure the authenticity and the integrity of multi-media documents circulating over the network. This requirement becomes evermore urgent since image falsification examples are raising because image manipulation tools are increasingly sophisticated and accessible to the public.

At first, researchers turned towards conventional cryptography that provided powerful solutions for digital message protection. The results for digital image authentication were not very satisfying as for digital messages. In fact, conventional cryptography provides a strict authentication service which declares an image authentic if and only if all pixels or the image binary representation did not undergo any modifications. Detection performances are good but localization of the regions that were manipulated is not the satisfactory. Furthermore, these techniques do not provide any reconstruction capabilities.

Therefore, researchers were attracted by fragile watermarking for strict image authentication. It consists in inserting a mark in the image to be protected in a fragile way. The mark would be destroyed if the image is altered. Fragile watermarking techniques show interesting results for strict image authentication. Detection of corrupted image regions was well demonstrated as for conventional cryptography algorithms. However, some additional improvements were added by fragile watermarking. The localization capabilities are more satisfactory and reconstruction of altered regions with acceptable quality is possible.

In order to provide image authentication methods that tolerate image manipulations while detecting content changes, efforts were oriented towards semi-fragile watermarking. The authentication mark inserted in the image would be fragile against content changing manipulations and robust against content preserving manipulations. This was achieved by generating marks from the image to be protected as well as by targeting the robustness of semi-fragile watermarking methods with respect to predefined manipulations. Semi-fragile watermarking methods allow an acceptable tolerance against some content preserving manipulations such as compression and brightening. Moreover, their detection and localization performances are very interesting. Semi-fragile watermarking algorithms also offer some restoration capabilities that are very encouraging. More investigations are yet needed to propose methods capable of tolerating various content preserving manipulations such as different filtering algorithms, image enhancement and restoration techniques and hopefully geometrical transformations. Moreover, restoration capabilities need additional improvements.

Recently, to selectively authenticate images, researchers have been focusing on the content representation of images instead of pixel representation. Digital signatures based on features that describe the image content have been investigated. These signatures would be modified if the image content was changed. Image authentication methods based on image content digital signatures promise a great success since they are able to detect any malevolent manipulation, to locate the altered regions with high precision and eventually to restore the original data using the significant information included in the signature. Moreover, they could tolerate some desirable image processing operations. However, the problem of tolerance against preserving manipulations is not solved completely.

Authentication methods based on image content digital signatures are robust against many content preserving manipulations but not all of them at once. Despite this weakness, hardware and software offering authentication services for digital images and documents are proliferating. Kodak DSS (http://all.net/books/forensics/digitalphoto/www.kodak.com/global/en/digital/acrobat/software/Authentication_whitepaper.PDF; DIGITAL Signature Standard, recognized by the National Institute of Standards and Technology) and Epson IAS system (http://www.epson.co.uk; Image Authentication System) are directly integrated in the digital cameras provided by these two companies. These commercial systems protect images and video sequences for a proof of acquisition. Also, Veridata of Signum Technologies (http://www.signumtech.com) and PhotoCheck of Alp Vision (http://www.alpvision.com) offer security services for digital document authentication such as passport photographs or badges that can be used to access sensitive zones. Nevertheless, more powerful solutions, especially in term of robustness against content preserving manipulations, are yet to be discovered.

By analyzing the various image authentication methods that were presented in this paper, it is clear that the authentication method is specific to the application (industry, medicine, military, copyright...). For applications where strict integrity is needed, such as medical, military or document images, algorithms that do not tolerate any modification to the image such as strict authentication are very satisfactory. The results show a high level of strict protection allowing detection and localization of pixels or even bits that underwent corruption with some acceptable restoration.

But in practice, most applications need to tolerate some content preserving manipulations such as compression, filtering, brightness correction and geometrical transformations. Therefore, the new challenge awaiting the image authentication community is robustness against content preserving manipulations rather than detection of manipulations that change the image content. In fact, some algorithms offer high tolerance performances only against one or two content preserving manipulations at the same time. As content preserving manipulations are specific to applications, practical solutions that tolerate a defined set of content preserving manipulations such as compression, geometrical transformations filtering and brightness correction...are still to be proposed. A flexible algorithm that allows the user to specify the list of desirable and malevolent manipulations does not exist yet. Therefore, Table 3 and Fig. 6 can be used to identify the features that are best suited for a specific application. In fact, the list of acceptable and the list of not allowable image manipulations can be easily defined for a specific application. These two lists can then be used to help identify the suitable features from Table 3 and Fig. 6.

In consequence, we can expect that a future more intelligent definition of the image content would help extract higher-level characteristics to be used in authentication methods that would be at the same time robust against important manipulations that preserve the image content and sensitive against changes that modify this content. An improvement of hash functions sensitivity would provide an alternative solution based on combining already known characteristics to provide a compact signature, less sensitive to changes and at the same time more robust against a predefined set of desired manipulations. A promising combination of characteristics would include DCT or DWT coefficients to provide robustness against compression and filtering, image moments to ensure robustness against geometrical transformations, and statistical characteristics such as means and variances of blocks to provide a good restoration capacity.

Acknowledgments This work was supported by the Natural Sciences and Engineering Research Council of Canada.

References

1. Akkar ML, Giraud C (2001) An implementation of DES and AES, secure against some attacks. In: Cryptographic hardware and embedded systems—CHES 2001. Proceedings on the third international workshop. Lecture notes in computer science, vol 2162. Paris, France, pp 309–318
2. Antonini M, Barlaud M, Mathieu P, Daubechies I (1992) Image coding using wavelet transform. IEEE Trans Image Process 1:205–220
3. Arjen K (1996) Generating standard DSA signatures without long inversion. In: Advances in cryptology—ASIACRYPT'96. Proceedings on the international conference on the theory and applications of cryptology and information security, Kyongju, Korea. Lecture notes in computer science, vol 1163. Springer, pp 57–64
4. Aucsmith D (ed) (1998) Information hiding. Proceedings of the second international workshop, Portland, Oregon, USA. Lecture notes in computer science, vol 1525. Springer, pp 306–318
5. Baldoza A, Sieffert M (2000) Methods for detecting tampering in digital images. AFRL Technology Horizons® 1(1):15–17
6. Bernd J (1993) Digital image processing: concepts, algorithms, and scientific applications. Springer, Berlin, Allemagne
7. Bhattacharjee S, Kutter M (2000) Compression tolerant image authentication. In: Proceedings of the ICIP-98, vol 1. Chicago, pp 435–439
8. Bhattacharya D, Sinha S (1997) Invariance of stereo images via the theory of complex moments. Pattern Recogn 30(9):1373–1386
9. Brassard G (1993) Cryptologie contemporaine. Masson, Paris
10. Byun SC, Lee IL, Shin TH (2002) A public key based watermarking for color image authentication. In: Proceedings of the IEEE international conference on multimedia and expo, vol 1. Piscataway, NJ, USA, pp 593–600
11. Canny JF (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698
12. Chang G, Yu SB, Vetterli M (2000) Adaptive wavelet thresholding for image de-noising and compression. IEEE Trans Image Process 9(9):1532–1546
13. Chen T, Wang J, Zhou Y (2000) Combined digital signature and digital watermark scheme for image authentication. In: Proceedings of the ICII2001, vol 5, pp 78–82
14. Choonsik P, Kaoru K (1996) New ElGamal type threshold digital signature scheme. IEICE Trans Fundam Electron Commun Comput Sci 1:86–93
15. Cover TM, Thomas JA (1991) Elements of information theory. Wiley, New York, NY
16. Cox IJ, Linnartz MG (1997) Public watermarks and resistance to tampering. In: Proceedings of the ICIP'97, Santa Barbara, California, USA
17. Culik K, Kari J (1997) Finite state transformations of images. Computer and Graphics 34:151–166
18. Culik K, Valenta V (1997) Generalized finite automata and transducer. J Autom Lang Comb 2:3–17
19. Darnell D (1999) PGP or PKI? The future of Internet security. EDI Forum: The Journal of Electronic Commerce 12(1):59–62
20. Dittmann J, Steinmetz A (1999) Content-based digital signature for motion pictures authentication and content-fragile watermarking. In: Proceedings of the IEEE international conference on multimedia computing and systems, vol II. Florence, Italy, pp 209–213
21. Dodgson NA (1997) Quadratic interpolation for image resampling. IEEE Trans Image Process 6(9): 1322–1326
22. Dugelay J-L, Rey C (2002) Un panorama des Méthodes de Tatouage Permettant d'Assurer un Service d'Intégrité. Revue Traitement du Signal 18(4), France
23. Eierman M, Niederman A, Adams FC (1995) DSS theory: a model of constructs and relationships. Decis Support Syst 14(1):1–26
24. Evertse J-H, Van-Heyst E (1992) Which new RSA-signatures can be computed from certain given RSA-signatures? J Cryptol 5(1):41–52
25. Faical A, Mersereau R (2001) Secure fragile digital watermarking technique for image authentication. In: Proceedings of the IEEE international conference on image processing, vol 3. Thessaloniki, pp 1031–1034
26. Feng W, Liu Z-Q (2004) Bayesian structural content abstraction for image authentication. In: Proceedings of the third international conference on machine learning and cybernetics. Shanghai, 2004
27. FIPS PUB XX (1993) Digital signature standard
28. Fowler B, Arps R, ElGamal A (1995) Quadtree based JBIG compression. In: Proceedings of the IEEE data compression conference, pp 102–111

29. Fridrich J (1998a) Image watermarking for tamper detection. In: Proceedings of the ICIP'98, Chicago, USA
30. Fridrich J (1998b) Methods for tamper detecting in digital images. In: Proceedings of the 6th IEEE international workshop on intelligent signal processing and communication systems (ISPACS '98), Melbourne, Australia
31. Fridrich J (1999a) Methods for tamper detection in digital images. In: Proceedings of the multimedia and security workshop at ACM multimedia '99, ACM, Orlando, FL, USA, pp 29–33
32. Fridrich J (1999b) Robust bit extraction from images ICMCS'99, Florence, Italy
33. Fridrich J, Goljan M, Baldoza AC (2000) New fragile authentication/watermark for images. In: Proceedings of the ICIP'2000, Vancouver, Canada
34. Fridrich J, Goljan M, Du R (2001) Invertible authentication. In: Proceedings of the SPIE, security and watermarking of multimedia contents, San Jose, California, January 23–26, 2001
35. Fridrich J, Goljan M (1999a) Protection of digital images using self embedding. In: Proceedings of the symposium on content security and data hiding in digital media, Institute of Technology, New Jersey, USA
36. Fridrich J, Goljan M (1999b) Images with self-correcting capabilities. In: Proceedings of the IEEE international conference on image processing, vol 3. Kobe, Japan, pp 792–796
37. Friedman G (1993) The trustworthy digital camera: restoring credibility to the photographic image. IEEE Trans Consum Electron 39:905–910
38. Gao Q, Li R, Wang H (2004) Spread spectrum digital watermarking algorithm using gold codes. Journal of Xi'an Jiaotong University 38(2):119–123
39. Gennaro R, Jarecki S, Krawczyk H (1996) Robust threshold DSS signatures. Advances in cryptology—EUROCRYPT '96. In: Proceedings on the international conference on the theory and application of cryptographic techniques. Lecture notes in computer science, vol 1070. Springer, pp 354–71
40. Goljan M, Fridrich JJ, Du R (2002) Distortion-free data embedding for images. In: Proceedings of the 4th international workshop on information hiding, April 25–27, 2001, pp 27–41
41. Gonzalez RC, Woods RE (2002) Digital image processing. Prentice-Hall, Upper Saddle River, NJ
42. Grossmann A, Morlet J (1987) Decomposition of function into wavelets of constant shape and related transforms. In: Streit L (ed) Mathematics and physics, lectures on recent results. World Scientific
43. Hai Pang H, Sweeney P, Paffett J (1998) Extended Kasami algorithm for cyclic codes. In: Proceedings of the IEEE GLOBECOM 1998 (Cat. no. 98CH36250), vol 5. Sydney, Australia, pp 2834–2839
44. Hampel H, Arps R, Chamzas C (1992) Technical features of the JBIG standard for progressive bi-level image compression. Signal Process Image Commun 4(2):103–111
45. Harn L (1994) New digital signature scheme based on discrete logarithm. Electron Lett 30(5):396–398
46. Harry A (1992) VDM specification of the MD4 message digest algorithm. Nat Phys Lab Teddington, UK, NPL DITC 204/92
47. Helleseth T, Kumar V (1994) The weight hierarchy of the Kasami codes. In: Proceedings of the 1994 IEEE international symposium on information theory (Cat. no. 94CH3467-8), p 308
48. Holliman M, Memon N (1997) Counterfeiting attacks on oblivious block-wise independant invisible watermarking schemes. IEEE Trans Image Process 6:432–441
49. Hu Y, Han D-Z (2005) Using two semi fragile watermark for image authentication. In: Proceedings of the international conference on machine learning and cybernetics, vol 9. Guangzhou, pp 5484–5489
50. Inoue H, Miyazaki A, Katsura T (2000) A digital watermark for images using the wavelet transform. Integr Comput Aided Eng 7(2):105–115
51. Inoue T, Murakami T, Inomoto K (1997) Evaluation of image symmetry using 3rd order moments. Bulletin of Aichi Institute of Technology Part B 32:177–82
52. Johnson NF, Katezenbeisser SC (1999) A survey of steganographic techniques. In: Katzenbeisser SC et al. (eds) Information techniques for steganography and digital watermarking. Artec House, Northwood, MA, pp 43–75
53. Kailasanathan C, Safavi-Naini R, Ogunbona P (2001) Image authentication surviving acceptable modifications. IEEE-EURASIP, workshop on nonlinear signal and image processing
54. Knudsen LR, Xuejia L (1998) Attacks on fast double block length hash functions. J Cryptol 11(1):59–72
55. Kostopoulos I, Christodou-lakis D, Skodras AN (2001) Self-authentication of colour images. In: Proceedings of the European conference on electronic imaging and visual arts, Florence, Italy
56. Kostopoulos I, Gilani SAM, Skodras AN (2002) Colour image authentication based on a self-embedding technique. In: Proceedings of the 14th international conference on digital signal processing (DSP2002) vol 2, Santorini, Greece, pp 733–736
57. Kundur D, Hatzinakos D (1999) Digital watermarking for telltale tamper proofing and authentication. Proc IEEE 87(7):1167–1180
58. Kutter M, Jordan F, Bossen F (1998) Digital watermaking of color images using amplitude modulation. J Electron Imaging 7(2):326–332

59. Kutter M, Voloshynocskiy S, Herrigel, A (2000) The watermark copy attack. In: Proceedings of SPIE security and watermarking of multimedia content II, vol 3971, San Jose, California, USA
60. Lambrecht C (1996) Perceptual models and architectures for video coding applications. Ph.D. thesis no 1520, Ecole Polytecnique Federale De Lausanne, Switzerland
61. Lampson B, Rivest R (1997) Cryptography and information security group research project: a simple distributed security infrastructure. Technical report, MIT
62. Langelaar G, Setyawan I, Lagendijk RL (2000) Watermarking digital image and video data. IEEE Signal Process Mag 17:20–43
63. Lee S, Jang D, Yoo CD (2005) An SVD-based watermarking method for image content authentication with improved security. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, vol 2, pp 525–528
64. Leest A, Veen M, Bruekers F (2003) Reversible image watermarking. In: Proceedings of the ICIP'03, vol 2, September 2003, pp 731–734
65. Li N, Wenliang D, Boneh D (2003) Oblivious signature-based envelope. In: Proceedings of the Annual ACM Symposium on Principles of Distributed Computing, vol 22, pp 182–189
66. Liang S, Rangayyan RM (1997) Lossless compression of continuous-tone images by combined inter-bit-plane decorrelation and JBIG coding. J Electron Imaging 6(2):198–207
67. Liehua X, Gonzalo, R, Lewis AA (2000) Image enhancement towards soft image authentication. In: Proceedings of the IEEE international conference on multimedia and expo (I), pp 497–500
68. Lin CY, Chang SF (1997) A robust image authentication method distinguishing JPEG compression from malicious manipulation. CU/CTR Technical Report 486-97-19
69. Lin CY, Chang SF (2000) Semi-fragile watermarking for authenticating JPEG visual content. In: Proceedings of the SPIE security and watermarking of multimedia content
70. Lin CY, Chang SF (2001) SARI: self-authentication-and-recovery image watermarking system. ACM Multimedia, Ottawa, Canada
71. Lin ET, Christine I, Podilchuk B, Delp EJ (2000) Detection of image alterations using semi-fragile watermarks. In: Proceedings of the SPIE international conference on security and watermarking of multimedia contents II, vol 3971, San Jose, CA, USA
72. Lin CY, Sow D, Chang SF (2001) Using self-authentication-and-recovery for error concealment in wireless environments. In: Proceedings of SPIE, vol 4518
73. Linde Y, Buzo A, Gray RM (1980) An algorithm for vector quantizer design. IEEE Trans Commun 28:84–95
74. Liu B, Wang X (2000) ECC performance evaluation for MDFE read channel. Chin J Electron 9(2): 144–148
75. Liu H, Zhang Q, Wu J (2003) Trust issues in PGP and solutions. Journal of Beijing University of Aeronautics and Astronautics 29(3):278–282
76. Liu T, Hu X, Dai Y (2004) Semi fragile watermarking for image content authentication. In: Proceedings on the international conference on signal processing, vol 3, pp 2342–2345
77. Lou DC, Liu JL (2000) Fault resilient and compression tolerant digital signature for image authentication. IEEE Trans Consum Electron 46:31–39
78. Lu C-S (2000) Structural digital signature for image authentication: an incidental distortion resistant scheme. In: Proceedings of the ACM multimedia workshops, pp 115–118
79. Lu H, Shen R, Chung F-L (2003) Fragile watermarking scheme for image authentication. Electron Lett 39(12):898–900
80. Lu Z-M, Liu C-H, Xu D-G (2003) Semi-fragile image watermarking method based on index constrained vector quantization. Electron Lett 39(1):35–36
81. Lu Z-M, Xu D-G, Sun S-H (2005) Multipurpose image watermarking algorithm based on multistage vector quantization. IEEE Trans Image Process. 14(6):822–831
82. Manjunath BS, Shekhar C, Chellappa R (1996) A new approach to image feature detection with applications. Pattern Recogn 31:627–640
83. Marvel LM, Boncelet CG (1999) Spread spectrum image steganography. IEEE Trans Image Process 8:1075–1083
84. Marvel LM, Hartwig GW (2000) Compression compatible fragile and semi fragile tamper detection. In: Proceedings of the SPIE international conference on security and watermarking of multimedia contents, vol 2, p 3971
85. Matsuo T, Kaoru K (2004) On parallel hash functions based on block-ciphers. In: Proceedings of the IEICE transactions on fundamentals of electronics, communications and computer sciences, pp 67–74
86. Memon N, Fridrich, J (2000) Attack on a fragile watermarking scheme. In: Proceedings of the SPIE international conference on security and watermarking of multimedia contents, San Jose, California

87. Memon ND, Fridrich J (2000) Further attacks on the Yeung–Mintzer fragile watermark. In: Proceedings of the SPIE international conference on security and watermarking of multimedia content II, vol 3971, San Jose, California, USA
88. Memon N, Fridrich J, Goljan M (2000) Further attacks on Yeung–Mintzer watermarking scheme. In: Proceedings of the SPIE international conference on electronic imaging 2000, San Jose, USA
89. Memon N, Poorvi V, Boon-Lock Y, Yeung M (2000) Distortion bounded authentication techniques. In: Proceedings of the SPIE international conference on security and watermarking of multimedia contents II, vol 3971, pp 164–174
90. Meyer Y (1990) Ondelette, in ondelette et operateurs. Hermann
91. Min-Shiang H, Jui-Lin L, Iuon-Chang L (2003) A practical (t, n) threshold proxy signature scheme based on the RSA cryptosystem. IEEE Trans Knowl Data Eng 15(6):1552–1560
92. Mintzer F, Braudaway G, Yeung M (1998) Effective and ineffective digital watermarks. In: Proceedings of the ICIP'97, Santa Barbara, CA, USA
93. Mohand M, Mesbah A (1997) Apprendre et Maîtriser MATLAB, Springer
94. Monga V, Vats D, Evans BL (2005) Image authentication under geometric attacks via structure matching. In: Proceedings of the IEEE international conference on multimedia and expo (ICME 2005), pp 229–232
95. Naccache D, M'raïhi D, Vaudenay S, Raphaeli D (1995) Can DSA be improved? Complexity trade-offs with the digital signature standard. In: Proceedings of advances in cryptology—EUROCRYPT '94. Workshop on the theory and application of cryptographic techniques, pp 77–85
96. Niho Y (1972) Multi-valued cross-correlation functions between two maximal linear recursive sequence. Ph.D. dissertation, Department of Electrical Engineering, University of Southern California
97. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):62–66
98. Paquet AH, Ward RK (2002) Wavelet-based digital watermarking for image authentication. In: Proceedings of the IEEE Canadian conference on electrical and computer engineering, vol I. Winnipeg, Manitoba, Canada, pp 879–884
99. Pennebarker WP, Mitchell JL (1993) JPEG: still image data compression standard. Van Nostrand Reinhold
100. Petitcolas FAP, Anderson RJ, Kuhn MG (1993) Attacks on copyright marking systems. In Aucsmith, pp 218–238
101. Petitcolas FAP, Anderson RJ, Kuhn MG (1999) Information hiding. A survey. In: Proceedings of the IEEE special issue on protection of multimedia content, vol 87, pp 1062–1078
102. Pfitzmann B (1996) Digital signature schemes: general framework and fail-stop signatures. Lecture notes in computer science, vol 1100. Springer, Berlin, Allemagne
103. Queluz MP (1998) Towards robust content based techniques for image authentication. In: Proceedings of IEEE signal processing society 1998 workshop on multimedia signal processing
104. Queluz MP (1999) Content-based integrity protection of digital images. In: Proceedings of the SPIE conference on security and watermarking of multimedia contents, vol 3657. San Jose California, USA, pp 85–93
105. Queluz MP, Salema C (1990) Compression factor and error sensitivity of the modified READ method. Communication, control and signal processing. In: Proceedings of the 1990 Bilkent international conference on new trends in communication, control and signal processing, vol 2, pp 1446–1452
106. Radhakrishnan R, Memon N (2002) On the security of the SARI image authentication system. IEEE Trans Circuits Syst Video Technol 2:440–443
107. Rao K, Yip RP (1990) Discrete cosine transform: algorithms, advantages, applications. Academic, New York
108. Raveendran P, Ornatu S, Chew P (1997) New technique to derive invariant features for unequally scaled images. In: Proceedings of the IEEE international conference on systems, man and cybernetics, vol 4, pp 3158–3163
109. Rivest R (1992) The MD4 message digest algorithm. RFC 1320, MIT and RSA Data Security, Inc
110. Rivest RL, Shamir A, Adleman LM (1978) A method for obtaining digital signatures and public-key cryptosystems. Commun ACM 21(2):120–126
111. Roe M (1994) Performance of symmetric ciphers and one-way hash functions. Fast software encryption. In: Proceedings of the Cambridge security workshop, pp 83–89
112. Rogier N, Chauvaud P (1997) MD2 is not secure without the checksum byte. Designs Codes Cryptogr 12(3):245–251
113. Rosario G, Krawczyk H (2000) RSA-based undeniable signatures. J Cryptol 13(4):397–416

114. Rothaus O (1993) Modified gold codes. IEEE Trans Inf Theory 39(2)654–656
115. Sanchez-Avila C, Sanchez-Reillol R (2001) The Rijndael block cipher (AES proposal): a comparison with DES. In: Proceedings of the IEEE 35th annual 2001 international Carnahan conference on security technology, pp 229–234
116. Sarwate DV, Pursley MB (1980) Crosscorrelation properties of pseudorandom and related sequences. Proc IEEE 68:593–619
117. Sayrol EJ, Cabanillas VS (1999) Optimum watermark detection for color images. In: Proceedings of the IEEE international conference on image processing, vol 2, pp 231–235
118. Schneider M, Chang S-F (1996) A robust content based digital signature for image authentication. In: Proceedings of the IEEE international conference on image processing, pp 227–230
119. Sergio D, Servetto KR, Orchard MT (1999) Image coding based on morphological representation of wavelet data. IEEE Trans Image Process 8(9):1161–1174
120. Sherif, NE-D, Mansour M (2002) Fragile and semi-fragile image authentication based on image self-similarity image processing. In: Proceedings of the international conference on image processing, vol 2, pp 897-900
121. Shin JB, Lee K, Shim K (2002) New DSA-verifiable signcryption schemes. In: Proceedings of the 5th international conference on information security and cryptology—ICISC 2002. Lecture notes in computer science, vol 2587. Springer, pp 35–47
122. Skala V, Kucha M (2001) The hash function and the principle of duality. In: Proceedings of the computer graphics international, vol 200, pp 167–174
123. Stallings W (1994) SHA: the secure hash algorithm. Dr. Dobb's Journal of Software Tools 19(4):32–34
124. Storck D (1996) A new approach to integrity of digital images. In: Proceedings of the IFIP conference on mobile communication, pp 309–316
125. Su C, Wang J (1993) ECCSyn. A synthesis tool for ECC circuits. In: Proceedings of the IEEE international symposium on circuits and systems, vol 3, pp 1706–1709
126. Sun Q, Chang SF (2002) Semi-fragile image authentication using generic wavelet domain features and ECC. In: Proceedings of the ICIP
127. Sun Q, Chang SF (2005) A secure and robust digital signature scheme for JPEG 2000 image authentication. IEEE Trans Multimedia 7(3):480–494
128. Tanaka K, Nakamura Y, Matsui K (1990) Duplex modified-READ coding scheme for facsimile transmission of documents. Electron Commun Jpn Part 1 Commun 73(5):46–56
129. Tancevski L, Bazgaloski L, Andonovic I (1994) Incoherent asynchronous optical CDMA using gold codes. Electron Lett 30(9):721–723
130. Tang S, Li J-T, Zhang Y-D (2005) SSF fingerprint image authentication: an incidental distortion resistant scheme. In: Proceedings of ACM international conference on multimedia, November 2005
131. Taubman D, Marcellin S, Michael W (2002) JPEG2000: image compression fundamentals, standards, and practice. Kluwer, Boston
132. The Mathworks Inc. image processing toolbox. User's guide. Version 3
133. Tian J (2002) Reversible watermarking by difference expansion. In: Proceedings of workshop on multimedia and security, December 2002, pp 19–22
134. Tirkel AZ, Rankin GA, Schyndel van RG, Osborne CF (1993) Electronic watermark. DICTA-93. Macquarie University, Sydney, Australia, pp 666–672
135. Trappe W, Washington LC (2002) Introduction to cryptography: with coding theory. Prentice-Hall, Upper Saddle River, NJ
136. Tsai WH (1985) Moment-preserving thresholding: a new approach. Comput Vis Graph Image Process 29:377–393
137. Tsuyoshi T (2004) A fast RSA-type public-key primitive modulo $p^k q$ using Hensel lifting. IEICE Trans Fundam Electron Commun Comput Sci E87-A(1):94–101
138. Tzeng CH, Tsai WH (2001) A new technique for authentication of image/video for multimedia applications. In: Proceedings of ACM multimedia workshops—multimedia and security: new challenges, Ottawa, Ontario, Canada
139. van Renesse RL (1998) Optical security and counterfeit deterrence techniques II, vol 3314, San Jose, California, USA
140. van Schyndel RG, Tirkel AZ, Osborne CF (1994) A digital watermark. In: Proceedings of the IEEE international conference on image processing, vol 2, pp 86–90
141. Venkatesan R, Koon S, Jakubowski M (2000) Robust image hashing. In: Proceedings of the IEEE international conference on image processing, vol 3, pp 664–666
142. Walton S (1995) Information authentication for a slippery new age. Dr Dobb's J 20(4):18–26
143. Wang X, Liu B (2002) A new ECC/RLL coding scheme. Chin J Electron 11(2):186–191

144. Wigderson A (1994) Wonders of the digital envelope—a crash course in modern cryptography. In: Proceedings of the IFIP transactions A: computer science and technology, n A-51, technology and foundations, pp 235–238
145. Wolfgang RB, Delp EJ (1997) Techniques for watermarking digital imagery: further studies. In: Proceedings of the international conference on imaging science, systems, and technology, vol 1. Las Vegas, Nevada, USA, pp 279–287
146. Wolfgang RB, Delp EJ (1996) A watermark for digital images. In: Proceedings of the IEEE international conference on image processing, vol 3, pp 219–222
147. Wong PW (1998a) A watermark for image integrity and ownership verification. In: Proceedings of the IS&T PIC conference, Portland, OR, USA
148. Wong PW (1998b) A public key watermark for image verification and authentication. In: Proceedings of the ICIP, Chicago, IL, USA
149. Wong PW, Delp EJ (1999) Security and watermarking of multimedia contents, vol 3657. San Jose, California, USA, pp 204–213
150. Wong PW, Memon N (2001) Secret and public key image watermarking schemes for image authentication and ownership verification. IEEE Trans Image Process 10:1593–1601
151. Wu CW (2001) Limitations and requirements of content based multimedia authentication systems. In: Proceedings of the IS&T/SPIE's international symposium on electronic imaging: science and technology, San Jose, CA, vol 4314, pp 241–252
152. Wu CW (2002) On the design of content based multimedia authentication systems. IEEE Trans Multimedia 4(3):385–393
153. Xiaotie D, Chan L, Huafei Z (1999) A proposal for secure hash algorithm. In: Proceedings of the 1999 international workshop on cryptographic techniques and e-commerce, pp 254–258
154. Xie L, Arce GR (1998a) A blind content based digital image signature. In: Proceedings of the second annual fedlab symposium on ATIRP
155. Xie L, Arce GR, (1998b) A blind wavelet based digital signature for image authentication. In: Proceedings of the EUSIPCO-98
156. Xie L, Arce GR (1998c) Joint wavelet compression and authentication watermarking. In: Proceedings of the ICIP'98
157. Xie L, Lewis A (2000) Methods for soft image/video authentication. In: Proceedings of the 4th annual fedlab symposium on ATIRP
158. Yeung MM (1998) Digital watermarking introduction. Commun ACM 41(7):31–33
159. Yeung M, Mintzer F (1997) An invisible watermarking technique for image verification. In: Proceedings of the ICIP'97, Santa Barbara, CA
160. Yu S, Hu Y, Zhou J (2004) Content based watermarking scheme for image authentication. In: Proceedings of the control, automation, robotics and vision conference 2004, vol 2, pp 1083–1087
161. Zheng Q, Chellappa R (1993) A computational vision approach to image registration. IEEE Trans Image Process 2(3):311–326
162. Zhu B, Tewfik AH, Gerek ON (1995) Low bit rate near-transparent image coding. SPIE Conf on Wavelet Application II 2491:173–184
163. Zhu B, Swanson MD, Tewfik AH (1996) Transparent robust authentication and distortion measurement technique for images. IEEE digital signal processing workshop (DSP 96)

Adil Haouzia received his B.Sc. degree in Electrical Engineering in 2000 from the Polytechnic Institut of Kiev, Ukraine and his Master degree in Electrical Engineering from the École de technologie supérieure

(ETS) in Montreal, Quebec, Canada. He worked as a Research Associate with ETS before joining in the Institut Supérieur d'Informatique - ISI Conseil in Montreal, as a Professor. His research interests include digital image processing, image authentication and networks security analysis and tests.

Rita Noumeir is a Professor in the Department of Electrical Engineering, École de Technologie Supérieure, Université du Quebec in Montreal. Her Ph.D. and Masters degrees in Biomedical Engineering, specialising in Medical Imaging, are from the École Polytechnique Engineering School in Montreal. A Professional Engineer, in the Ordre des ingénieurs du Québec, she is a member of numerous Canadian and international scientific associations. A founding member of the Imaging, Vision and Artificial Intelligence Laboratory (LIVIA), her main research interest is healthcare information technology, specifically interoperability, electronic patient record, security, information confidentiality and image processing. A member of both the Technical and Planning International IHE Radiology Committees, Dr. Noumeir has been involved in developing many IHE integration profiles in radiology, and in organizing integration demonstrations. Cofounder of IHE Canada. Dr. Noumeir has contributed to many R&D projects with several Canadian and international companies in medical imaging and healthcare information.