US 20030070075A1

(19) **United States**

(12) **Patent Application Publication**

(10) **Pub. No.: US 2003/0070075 A1**

(43) **Pub. Date: Apr. 10, 2003**

Deguillaume et al.

(54) **SECURE HYBRID ROBUST WATERMARKING RESISTANT AGAINST TAMPERING AND COPY-ATTACK**

(76) Inventors: **Frederic Deguillaume**, Geneva (CH); **Sviatoslav Voloshynovskiy**, Geneva (CH); **Thierry Pun**, Geneva (CH)

Correspondence Address:
Prof. Thierry Pun
University of Geneva
Dept. of Computer Science
24, rue du General Dufour
Geneva 1211 GENVA 4 (CH)

(21) Appl. No.: **10/194,278**

(22) Filed: **Jul. 15, 2002**

Related U.S. Application Data

(60) Provisional application No. 60/327,097, filed on Oct. 4, 2001.

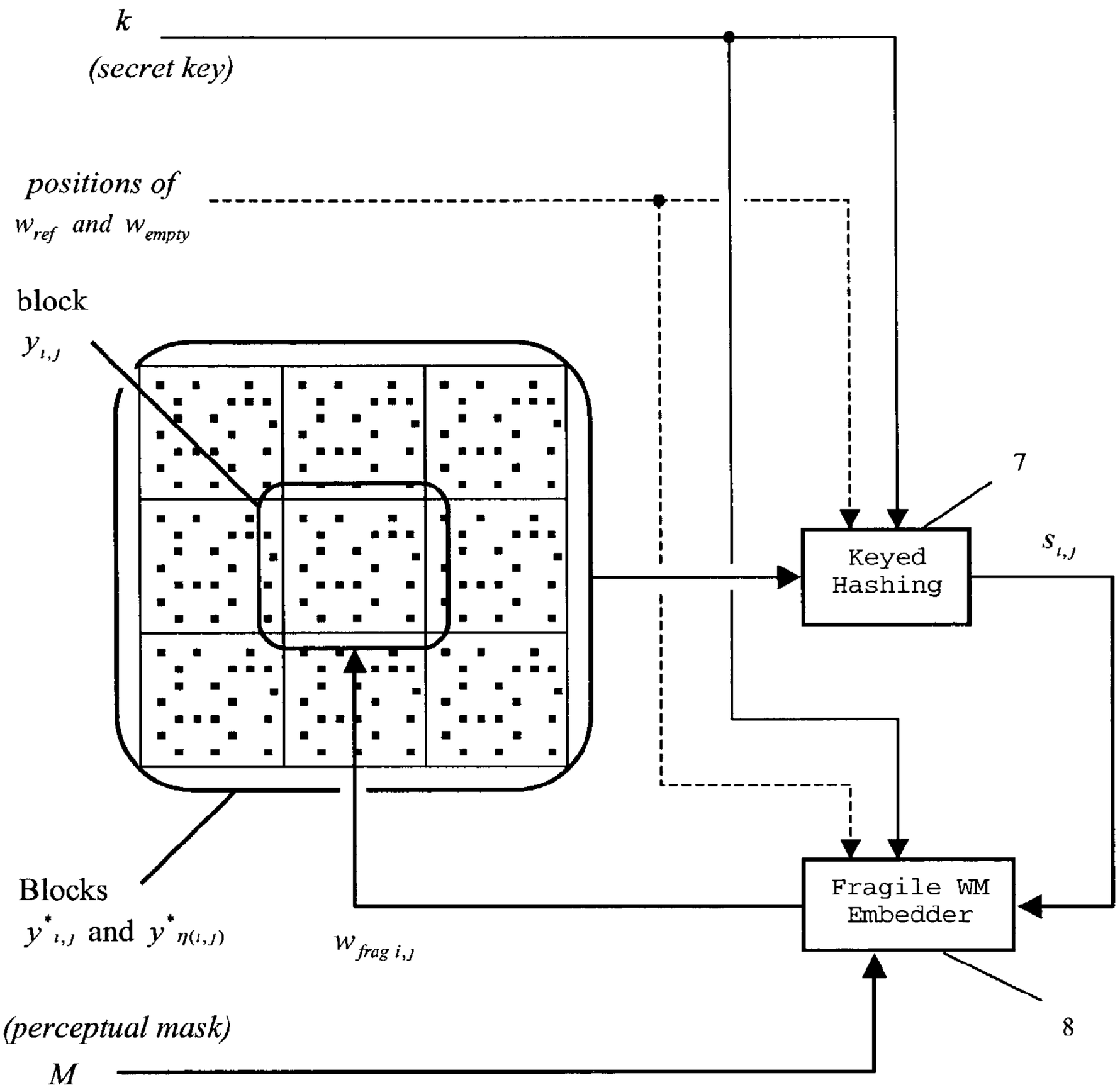
Publication Classification

(51) **Int. Cl.**[7] **H04L 9/00**

(52) **U.S. Cl.** **713/176**

(57) **ABSTRACT**

The present invention relates to the methods for hybrid watermarking method joining a robust and a fragile watermark, and thus combining copyright protection, authentication and tamperproofing. As a result this approach is at the same time resistant against the copy attack. In addition, the fragile information is inserted in a way which preserves the resistance and reliability of the robust part.



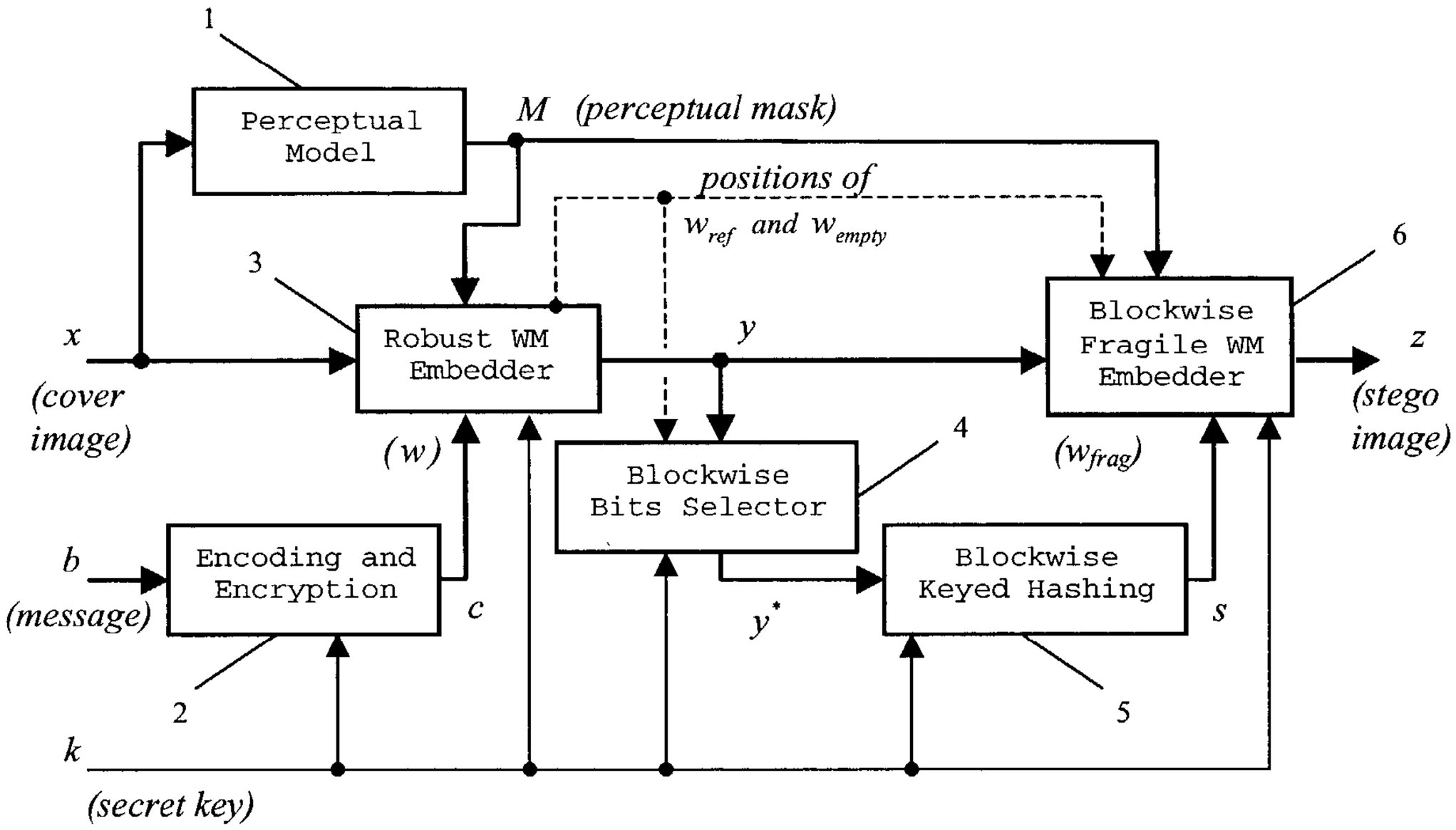

1
Perceptual Model
M (perceptual mask)
positions of
w_{ref} and w_{empty}
3
6
x
(cover image)
Robust WM Embedder
y
Blockwise Fragile WM Embedder
z
(stego image)
(w)
4
Blockwise Bits Selector
(w_{frag})
b
(message)
Encoding and Encryption
c
y^*
Blockwise Keyed Hashing
s
2
5
k
(secret key)

FIG. 1

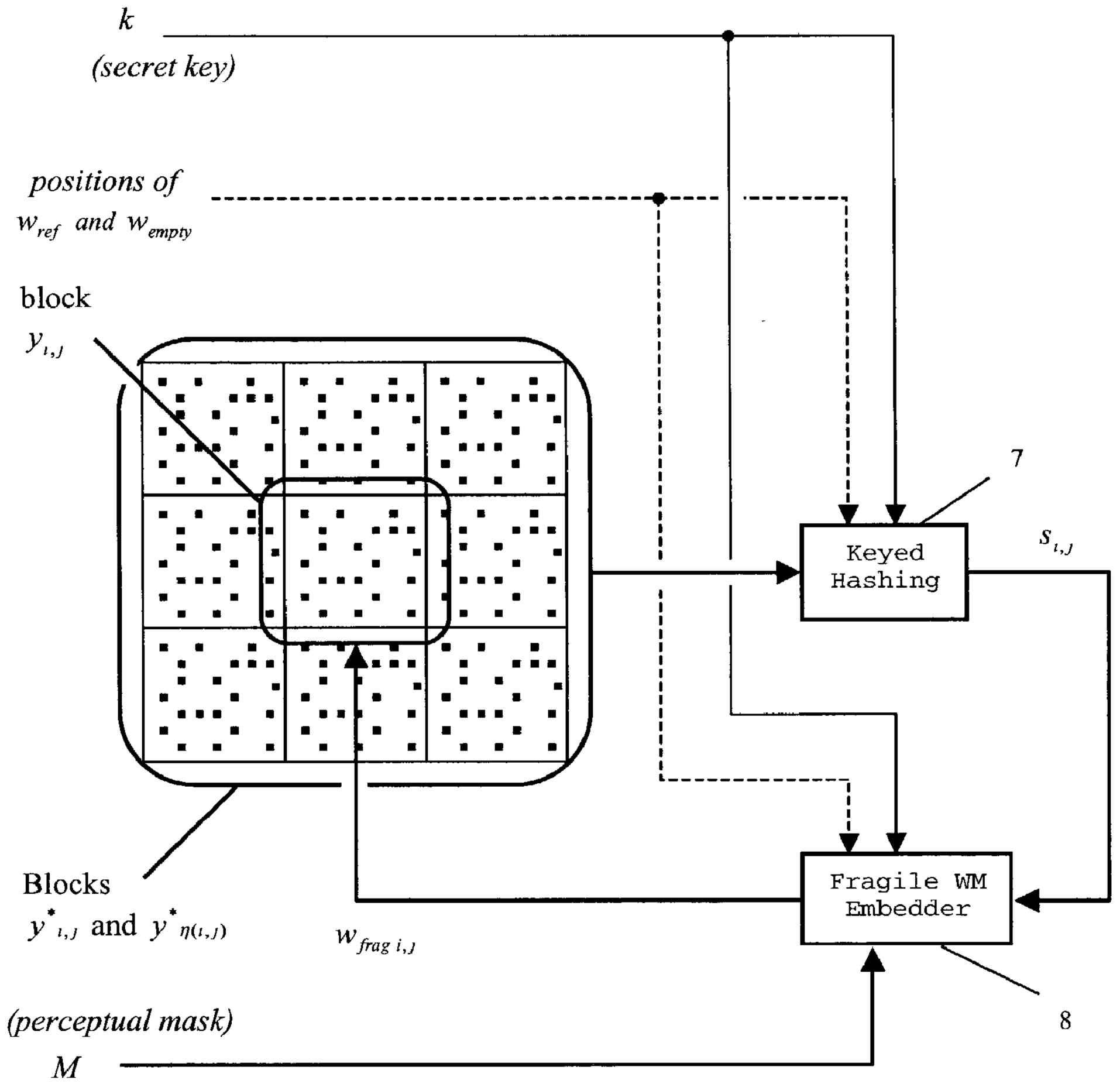

k
(secret key)
positions of
w_{ref} and w_{empty}
block
$y_{i,j}$
Blocks
$y^*_{i,j}$ and $y^*_{\eta(i,j)}$
(perceptual mask)
M
Keyed Hashing
7
$s_{i,j}$
Fragile WM Embedder
$w_{frag\ i,j}$
8

FIG. 2

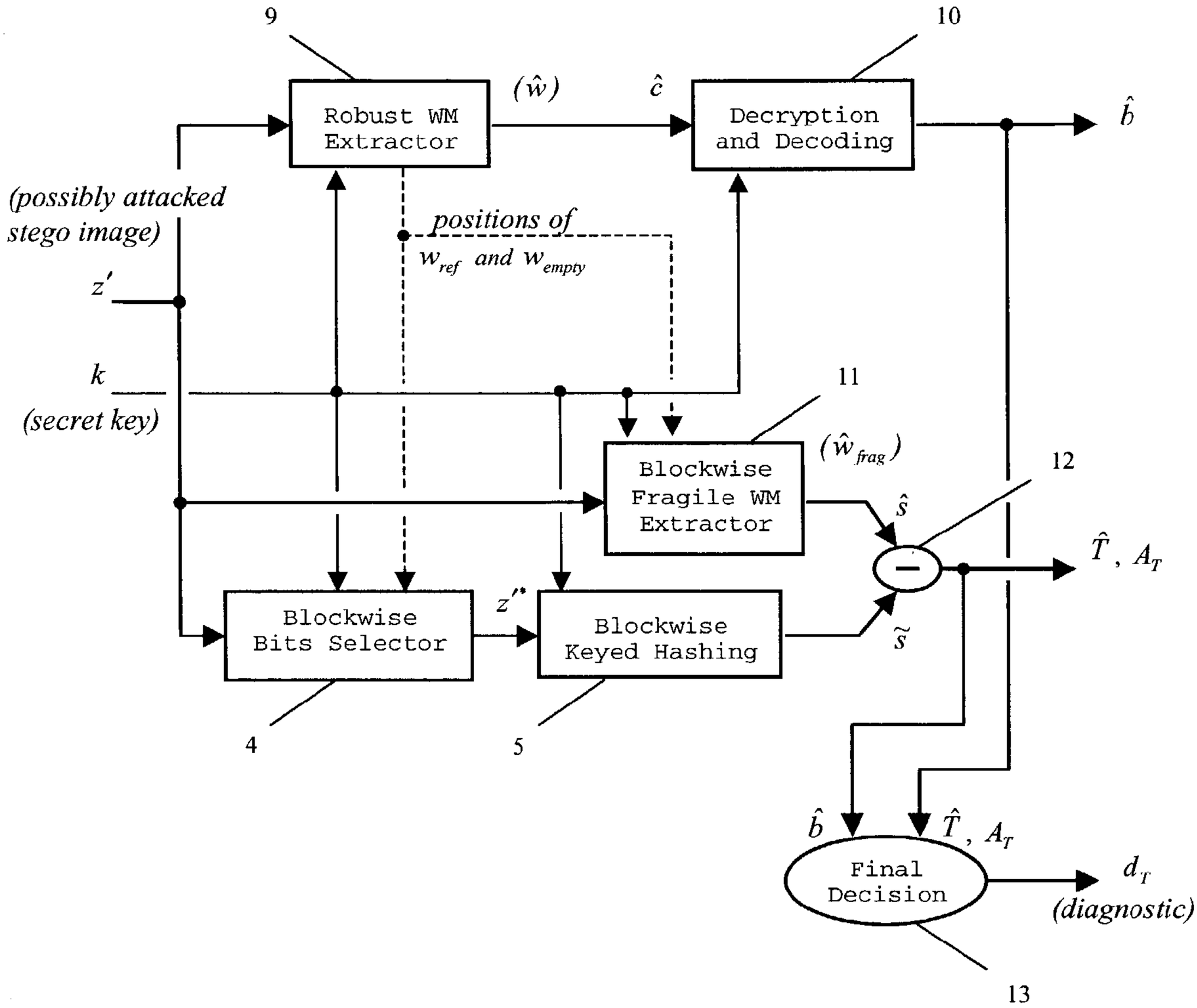

9
10
Robust WM Extractor
$(\hat{w})$
$\hat{c}$
Decryption and Decoding
$\hat{b}$
(possibly attacked stego image)
z'
positions of w_{ref} and w_{empty}
k
(secret key)
11
$(\hat{w}_{frag})$
Blockwise Fragile WM Extractor
12
$\hat{s}$
$\hat{T}$, A_T
Blockwise Bits Selector
z''
Blockwise Keyed Hashing
$\tilde{s}$
4
5
$\hat{b}$
$\hat{T}$, A_T
Final Decision
d_T
(diagnostic)
13

FIG. 3

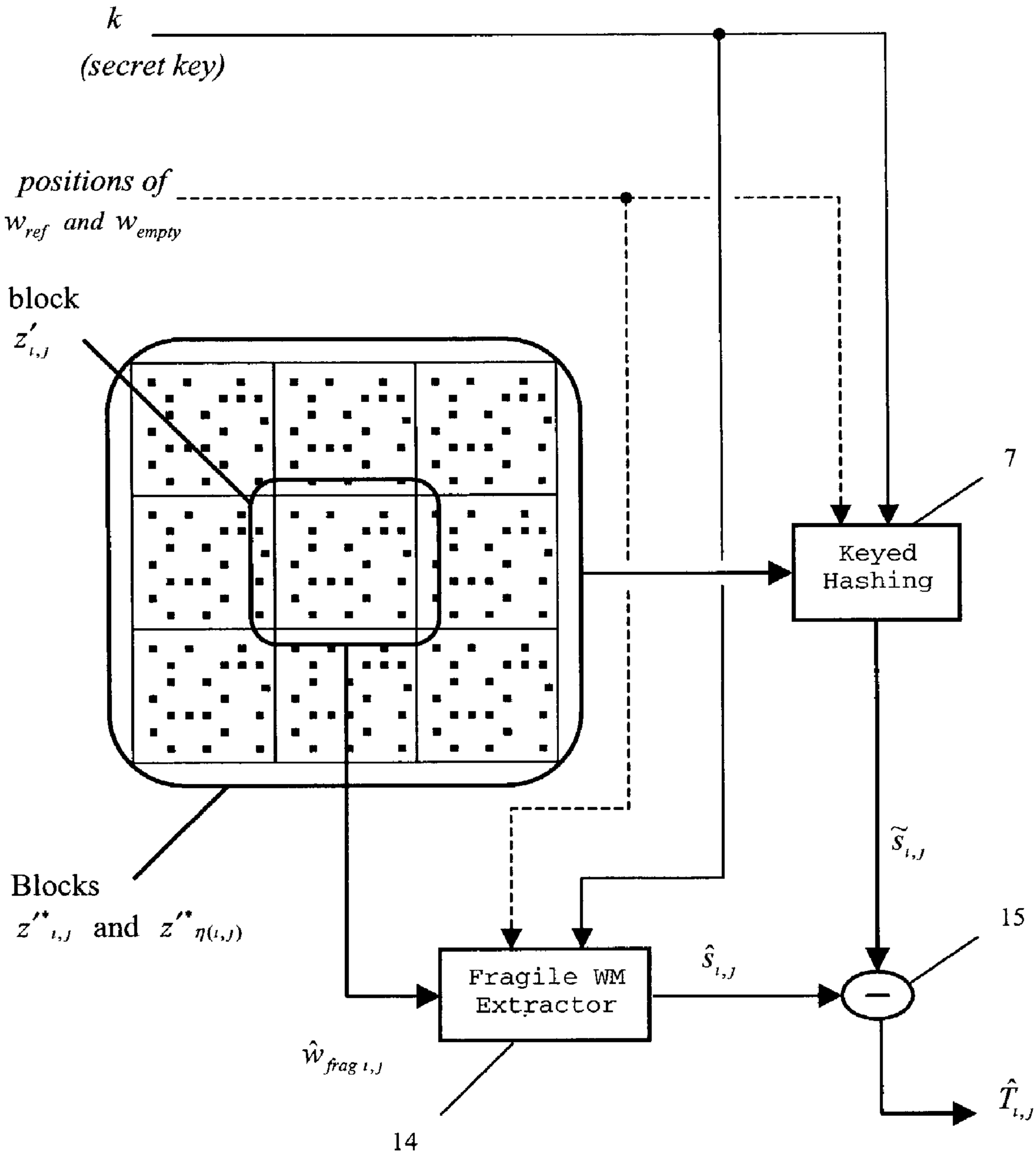

k
(secret key)
positions of
w_{ref} and w_{empty}
block
$z'_{i,j}$
Blocks
$z'^{*}_{i,j}$ and $z'^{*}_{\eta(i,j)}$
14
Keyed Hashing
7
$\tilde{s}_{i,j}$
Fragile WM Extractor
$\hat{w}_{frag\ i,j}$
$\hat{s}_{i,j}$
15
$\hat{T}_{i,j}$

FIG. 4

1. for $i = 1$ to M/m
2. for $j = 1$ to N/n
3. $y^*_{i,j}, \; y^*_{\eta(i,j)} = BitsSelector(y_{i,j}, \; y_{\eta(i,j)})$
4. $s_{i,j} = KeyedHash_k(y^*_{i,j}, \; y^*_{\eta(i,j)}, \ldots)$
5. $z_{i,j} = FragileEmbed_k(y_{i,j}, \; s_{i,j})$
6. end for
7. end for

FIG. 5

1. for $i = 1$ to M/m
2. for $j = 1$ to N/n
3. $z'^*_{i,j}, \; z'^*_{\eta(i,j)} = BitsSelector(z'_{i,j}, \; z'_{\eta(i,j)})$
4. $\tilde{s}_{i,j} = KeyedHash_k(z'^*_{i,j}, z'^*_{\eta(i,j)}, \ldots)$
5. $\hat{s}_{i,j} = FragileExtract_k(z'_{i,j})$
6. $\hat{T}_{i,j} = \hat{s}_{i,j} \ominus \tilde{s}_{i,j}$
7. end for
8. end for

FIG. 6

SECURE HYBRID ROBUST WATERMARKING RESISTANT AGAINST TAMPERING AND COPY-ATTACK

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] In some embodiment this application refers to the following robust watermarking related patents:

[0002] Self-reference multi-resolution watermarking in the European Patent Application PCT/IB00/01089 filed by Sviatoslav Voloshynovskiy, Frédéric Deguillaume, Shelby Pereira, Alexander Herrigel and Thierry Pun on Aug. 3, 2000 and entitled "Method for adaptive digital watermarking robust against geometric transform", accepted in May 2001 [1];

[0003] Recovering of local non-linear distortions on the U.S. patent application USPTO 60/327,097 filled by Sviatoslav Voloshynovskiy, Frédéric Deguillaume and Thierry Pun in Oct. 4, 2001 and entitled "A Method for Digital Watermarking Robust Against Local and Global Geometrical Distortions and Projective Transforms"[2];

[0004] Recovering of global affine transforms on the U.S. patent application USPTO 10/051,808 filled by Frédéric Deguillaume, Sviatoslav Voloshynovskiy and Thierry Pun in Jan. 17, 2002 and entitled "A Method for the Estimation and Recovering of General Affine Transform"[3].

BACKGROUND OF THE INVENTION

[0005] These last years, the rapidly growing multimedia market and use of digital technologies in general has revealed an urgent need for securing documents. Numerous threats have been identified yet, but one of the first to be pointed out was the incredible ease with which exact copies could be done without any authorization. Classical protection such as cryptography soon appeared not to be a solution, since once a document has been decrypted, even by an authorized customer, this customer could always distribute the document in plain form without any restriction. Therefore more sophisticated document security methods have been proposed, aiming first at solving the copyright protection problem, based on watermarking technologies.

[0006] Copyright Protection

[0007] The main requirements for copyright-protection watermarking algorithms are robustness (denoting how the watermark can survive any kind of malicious or unintentional transformations), visibility (does the watermark introduce perceptible artifacts), and capacity (the amount of information which can be reliably hidden and extracted from the document after certain attacks). For copyright applications, robustness should be as high as possible, visibility as low as possible in order to preserve the value of the marked document. Note however that capacity can be low since copyright information generally requires a rather small amount of information, which can be an index inside a database holding copyright information. Other requirements can be outlined, which are: security (from the cryptographic point of view), and that the scheme should be oblivious (the original image is not needed for the extraction process).

[0008] Many robust watermarking schemes have been proposed, consisting in either spatial domain, or transform domain watermarks. Currently two main issues can be pointed out: first, interference cancellation, which can be performed either at the encoder side by embedding the watermark using quantization as Quantization Index Modulation (QIM) [4] or using product codebooks of dithered uniform scalar quantizers in the Scalar Costa Scheme (CSC) [5], or at the decoder side based on the robust prediction of the embedded watermark as in our previous approach [1,6]. Secondly, geometrical synchronization, aiming at compensating geometrical distortions which desynchronize the embedded signal and make it unreadable.

[0009] Solutions against geometrical transform can use either a transform invariant domain watermark like the Fourier-Mellin transform [7], or an additional template for resynchronization [8], or a self-reference watermark based on the Autocorrelation Function (ACF) of a repetitive watermark [9]. Self-reference watermarks have been shown to have as main advantage over other methods the fact that they exploit the redundancy of the regular structure of the watermark in order to robustly estimate the undergone geometrical distortions. We previously proposed a method based on this concept, which is robust to general affine transforms [6,10] as well as to non-linear distortions and to the Random Bending Attack (RBA) [2,11]; our approach uses the ACF or magnitude spectrum of a periodical watermark, at the global level to recover from affine transforms, and at the local level to recover from the RBA.

[0010] Tamperproofing and Authentication

[0011] Other important threats have recently been identified with respect to multimedia document, the most important of them being the ease offered by today technologies for tampering or counterfeiting. Digital cameras are constantly growing in quality while becomming widely available, and softwares such as Paintshop Pro or Addobe Photoshop make it very easy to perform complex modifications without visible artifact. Although this is useful for artistic applications, this is a serious problem for legal applications such as evidences in trials, for insurances in medical imaging, for counterfeiting, etc. Classical analysis techniques used for authenticating analog photographs are ineffective. Another important issue is the ability to authenticate the originator of a visual document.

[0012] Of course global cryptographic signatures can detect tampering and authenticate documents, but are unable either to highlight which areas have been modified, or to assess the severity of the alteration; moreover, format conversion kills this meta-data. Such a global authentication has been proposed by Friedman in his trusted digital camera [12]. Therefore one proposed solution to both tamperproofing and authentication is again watermarking, which is used here to attach check-codes of local areas inside the image itself, in order to achieve the ability to localize altered regions. Such watermarks do not need the same level of robustness than for copyright protection, since in case of removal or cancellation the image can just be considered as non authentic. Two cases can be distinguished: the watermark can be either fragile, meaning that any modification, even a limited change of a small set of pixels, is detected, or

semi-fragile, offering a level of tolerance to some "acceptable" alterations such as low-level lossy compression or slight contrast adjustment.

[0013] For fragile watermarking, the image is generally first divided into small blocks for locality, and a key-dependent hash function is applied to each of them, and the obtained hash-codes are embedded into their corresponding blocks, usually in the least significant bits (LSB) of pixels. Tampering is then detected where the recomputed codes do not match the stored codes. Wong [13] proposed such a blockwise approach. At the opposite, semi-fragile watermarks are more tolerant, and can even be used to measure the severity of the alteration; a robust watermarking scheme has sometimes been proposed for this, however this approach is insecure since robust watermarks are usually additive, making them vulnerable to the so-called copy attack: the signal can be easily estimated using denoising techniques and copied to another image [14]. Note that the same attack can be applied to LSB-based technologies too. Another possibility is to compute robust or visual hashes which are tolerant to slight modifications, and to embed them robustly. We can mention also self-embedding watermarks where a low resolution version of the visual content is embbedded into the image itself; Wu and Liu [15] propose such a scheme which embeds the visual content using the look-up table (LUT) of the frequency domain coefficients, and Fridrich [16] proposes to embed the visual content in the bit representation of chosen discrete Cosine transform (DCT) coefficients. Self-embedding watermarks not only can detect tampered areas by locally analyzing mismatches between the stego image and the actually extracted visual information, but can even reconstruct these areas.

[0014] A Hybrid Solution

[0015] While robust watermarks are typically required for copyright protection, the fragile or semi-fragile watermarks have been proposed to solve tamperproofing and authentication. Watermarking methods above are either robust schemes, or fragile/semi-fragile schemes; however approaches combining both robust and fragile/semi-fragile schemes for copyright and tamperproofing/authentication application are rarely proposed. Fridrich [17] proposed such an hybrid method, but uses a watermark with relatively low robustness. Further, most of robust watermarking schemes are vulnerable to the copy attack, which allows copying a watermark from one document to another without need for any a priori knowledge [14]. Therefore, to this extent no real working scheme for hybrid robust watermarking, tamperproofing and authentication has been proposed yet.

[0016] The present invention describes a method for hybrid robust watermarking which: first, joins a highly robust watermark (which we will call w) with a fragile authentication watermark (called w_{frag}) for combined copyright protection, authentication and tamperproofing; secondly, which embeds the authentication watermark w_{frag} in a way which preserves the resistance and the reliability of the robust watermark w. The robust watermark w mainly consists in two parts which are: an informative watermark carrying the embedded message itself (called w_{inf}), and a key-dependent only reference watermark used as a pilot signal for synchronization as well as for channel state estimation purpose (called w_{ref}) at the decoder side. Therefore the authentication watermark w_{frag} could be embedded orthogonally with respect to the informative watermark w_{inf}, using the positions of the reference watermark w_{ref} only. In the case where the density of the robust watermark w is less than 1, positions still remain which contain no robust watermark information at all, called w_{empty}, and which could be used for the embedding of w_{frag} too. We further address the cryptography and security aspects of blockwise hash-coding. As a result this approach is at the same time resistant against local or global tampering, and against the copy attack which aims at copying a watermark from an image to another one without knowing the key.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] The drawings shown in:

[0018] **FIG. 1**: An embodiment for the proposed hybrid embedding algorithm, including both the robust and the authentication parts at the global level, is shown in this block-diagram, each block being identified by a unique number in parenthesis:

[0019] Robust part: a Perceptual Model M is computed from the input cover image x (block **1**) in order to achieve low visual impact; the message b to be embedded is encoded and encrypted (**2**) using a user secret key k, resulting into a codeword c; the codeword c is then spatially allocated (i.e. into k key-dependent positions) and embedded into x by the Robust Watermark (WM) Embedder (**3**) as a robust watermark w, using the perceptual mask M, to form the robustly marked image y. Blocks **2** and **3** use the secret key k.

[0020] Fragile part: spatial k key-dependent positions and bits are retained for the embedding of the fragile watermark w_{frag}; these positions can fit those used by w for the reference watermark (w_{ref}) as well as positions not containing any robust watermark information (w_{empty}), in order to achieve the orthogonality with respect to the informative watermark (w_{inf}); the bits retained can be the least significant bits (LSB) of the selected pixels; the Blockwise Bits Selector block (**4**) then clears these selected bits (i.e. set them to zero) inside y which may be modified by w_{frag}, resulting into y*; then a Blockwise Keyed Hashing (**5**) generates hash codes from y*, resulting into a set of signatures s which are embedded by the Blockwise Fragile WM Embedder (**6**) into y, using the perceptual mask M, to get the final stego image z. Blocks **4**, **5** and **6** use the secret key k; blocks **4** and **6** also use the w_{ref} and w_{empty} positions transmitted from the robust part as shown by the dashed arrows.

[0021] **FIG. 2**: An embodiment for the proposed algorithm is shown for the fragile part embedding at the block level, this part corresponding to blocks **4**, **5** and **6** of **FIG. 1** but for one local block $y_{i,j}$ indexed by i,j, after division of the robustly marked image y into contiguous and non-overlapping blocks as $y=\{y_{i,j}\}$. The current block $y^*_{i,j}$ and its neighbors $y^*_{\eta(i,j)}$, obtained after setting to zero the bits where w_{frag} will be embedded in order to exclude them from the hash function input, are hashed together by the k key-dependent Keyed Hashing function (block **7**) resulting into the signature $s_{i,j}$ for this block i,j ($s=\{s_{i,j}\}$); $s_{i,j}$ is then embedded by the Fragile WM Embedder (**8**), using the perceptual mask M, into the current block $y_{i,j}$; the positions

where the fragile corresponding watermark block $w_{frag\ i,j}$ is embedded are shown by the black square dots, these k key-dependent positions also fitting those corresponding to w_{ref} and w_{empty}

[0022] of the robust part and being transmitted to the fragile algorithm (dashed arrows).

[0023] **FIG. 3**: An embodiment for the proposed hybrid extraction and verification algorithm, including both the robust and the authentication parts at the global level, is shown in this block-diagram:

[0024] Robust part: the possibly attacked stego image z' is processed by the Robust WM Extractor (block **9**) which estimates the robust watermark $\hat{w}$ and extracts an estimated codeword $\hat{c}$; $\hat{c}$ is decrypted and decoded (**10**) using the secret key k, to get the estimated message $\hat{b}$.

[0025] Fragile part: the Blockwise Fragile WM Extractor (**11**) estimates the embedded fragile watermark $\hat{w}_{frag}$ and get the embedded signatures $\hat{s}$; the Blockwise Bits Selector (**4**) clears from z' all bits reserved for w_{frag} to get z'* on which the Blockwise Keyed Hashing (**5**) is performed, resulting into the recomputed set of signatures $\tilde{s}$; then $\hat{s}$ and $\tilde{s}$ are blockwise compared (**12**) (schematically denoted as "-") to get a tampered-blocks map $\hat{T}$ of changed blocks, with values 1 where tampering occurred—i.e. 1 for different signatures, and 0 where no modification occurred—matching signatures; a global authenticity value A_T ($0 \leqq A_T \leqq 1$) can then be computed which counts the ratio of authentic blocks over all blocks (i.e. the ratio of 0's in $\hat{T}$).

[0026] Hybrid diagnostic: finally, by combining the robust message $\hat{b}$ and a decoding diagnostic (i.e. is $\hat{b}$ correctly decoded or not, based for example on some integrity check-code applied to the message and including into the binary string b), the tampered-blocks map $\hat{T}$, and the global authenticity value A_T, a Final Decision (**13**) is taken about the authenticity or the possible tampering of z', resulting into the tampering/authenticity diagnostic message d_T.

[0027] **FIG. 4**: An embodiment for the proposed algorithm is shown for the fragile part extraction at the local block level, this part corresponding to blocks **4**, **5**, **11** and **12** of **FIG. 3** but for one block $z'_{i,j}$ indexed by i,j, the marked and possibly attacked image z' being divided into contiguous and non-overlapping blocks as $z'=\{z'_{i,j}\}$. From the current block $z'_{i,j}$ the fragile watermark block $w_{frag\ i,j}$ is extracted and the local signature $\hat{s}_{i,j}$ (as $\hat{s}=\{\hat{s}_{i,j}\}$) given by the Fragile WM Extractor (block **14**); the current block $z'_{i,j}$ and its neighbors $z'_{\eta(i,j)}$ are taken, inside them bits reserved for w_{frag} are set to zero as for the embedding stage, resulting into $z'^*_{i,j}$ and $z'^*_{\eta(i,j)}$ which are hashed by the k key-dependent Keyed Hashing function (**7**) to get the recomputed local signature $\tilde{s}_{i,j}$, (as $\tilde{s}=\{\tilde{s}_{i,j}\}$); then $\hat{s}_{i,j}$ and $\tilde{s}i,j$ are matched together (**15**) (denoted as "-") to get the local tampering value $\hat{T}_{i,j}$ in order to build the tampered-blocks map $\hat{T}=\{\hat{T}_{i,j}\}$, with $\hat{T}_{i,j}=0 \Leftrightarrow \hat{s}_{i,j}=\tilde{s}_{i,j}$.

[0028] **FIG. 5**: Pseudo-code describing the fragile part embedding for all blocks: the robustly marked image y is divided into blocks $y_{i,j}$, and the blocks processed for each index i,j (lines **1**, **2**); from the current block $y_{i,j}$ and its neighbors $y_{\eta(i,j)}$ the bits which are to be used for the fragile watermark embedding are cleared by the BitsSelector function, resulting into $y^*_{i,j}$ and $y^*_{\eta(i,j)}$ (line **3**); then $y^*_{i,j}$ and $y^*_{\eta(i,j)}$ are hashed by the KeyedHash function, together with additional information if needed (denoted by the " . . . "), giving the local signature $s_{i,j}$ (line **4**); at the end $s_{i,j}$ is embedded into $y_{i,j}$ by FragileEmbed to form the final stego block $z_{i,j}$ (line **5**); both keyedHash and FragileEmbed depend on the secret key k.

[0029] **FIG. 6**: Pseudo-code describing the fragile part verification for all blocks: the possibly attacked image z' is divided into blocks $z'_{i,j}$, and the blocks processed for each index i,j (lines **1**, **2**); from the current block $z'_{i,j}$ and its neighbors $z'_{\eta(i,j)}$ the bits used for the fragile watermark embedding are cleared by the BitsSelector function, resulting into $z'^*_{i,j}$ and $z'^*_{\eta(i,j)}$ (line **3**); then $z'^*_{i,j}$ and $z'^*_{\eta(i,j)}$ are hashed by the KeyedHash function, together with the same additional information as for the embedding stage if needed (denoted by " . . . "), to recompute the local signature $\tilde{s}_{i,j}$ (line **4**); the embedded local signature is estimated too as $\hat{s}_{i,j}$ by the FragileExtract function (line **5**); at the end $\tilde{s}_{i,j}$ and $\hat{s}_{i,j}$ are compared (symbolized by "-") to get the local authenticity value $\hat{T}_{i,j}$, with $\hat{T}_{i,j}=1$ if $\hat{s}_{i,j}=\tilde{s}_{i,j}$, or 0 otherwise; both keyedHash and FragileExtract depend on the secret key k.

[0030] In the drawings identical parts are designated by identical reference numerals. The pseudo-codes are given for the purpose to describe the algorithms as clearly as possible, and are not optimised.

DETAILED DESCRIPTION OF THE INVENTION

[0031] We propose to join the highly robust watermarking method that we previously developed (Voloshynovskiy et al. [6]) with a blockwise fragile algorithm based on cryptographically secure hash-codes similar to Wong's approach [13], but with various improvements for security reasons that are discussed later in this document. The robust watermarking scheme our hybrid technology is based on is a content adaptive multi-resolution algorithm with channel state estimation, exploiting a self-reference watermark in order to resist against geometrical transformations. The principles of this scheme are also explained in more details in our previous publications [6,10,11,18] and patents [1,2,3].

[0032] Hybrid Watermark Embedding

[0033] The block diagram **FIG. 1** shows the hybrid embedding process at the image level. This is a symmetrical tamperproofing/authentication scheme, that means that both the signature embedding and verification require the same user key k, which should be kept secret. The robust watermark w further consists in the following two non-overlapping, i.e. orthogonal, components: the informative watermark w_{inf} holding the copyright message b, encoded and encrypted to a codeword c with the secret key k (**FIG. 1**, block **2**); and the reference watermark w_{ref} only depending on k, used as a pilot e.g. for translation/cropping determination and for other side information which can be used for the decoding step. The allocated positions within each block also depend on k. Further, if w is embedded with a density less than 1, free positions still remain that contain no robust information and which we call w_{empty}. Then w is embedded by the robust watermarking algorithm to the cover image x

(**FIG. 1**, block **3**), taking into account the perceptual model M (**FIG. 1**, block **2**) computed from x to ensure low visual distortions.

[**0034**] Obviously, the fragile component has to be applied after the robust one, in order to hash the robust watermark with the image. The fragile watermark, called w_{frag}, is then based on a key-dependant blockwise cryptographically secure hash function (**FIG. 1**, block **5**), of which input key is derived from k. The resulting code is then embedded as a local signatures s (note that from the cryptographic point of view, we should talk about message digest code (MDC), however in this document we will use the term of secret key signature) is then embedded in a fragile way within each block (**FIG. 1**, block **6**): a set of positions is pseudo-randomly selected in y based on k, and the bits of the signature embedded at these positions into the bits reserved for w_{frag} in y (e.g. the LSB). In order to keep hash codes valid, the hash function takes as input y*, a version of y where all bits (e.g. LSB) selected for the embedding of w_{frag} have been cleared (i.e. set to 0) by the "Bits Selector" block (**FIG. 1**, block **4**). The "Keyed Hashing" block could be any keyed hash algorithm, or an unkeyed one encrypted afterwards. The hash function requirements could be summarized as:

$$I=I' \Rightarrow H_k(I)=H_k(I')$$
$$I\neq I' \Rightarrow H_k(I)\neq H_k(I') \quad (1)$$

[**0035**] where I and I' are any input (not necessary visual data), and H_k is a hash function depending on a random key k. Moreover, when $I\neq I'$ even for a single bit, $H_k(I)$ and $H_k(I')$ are completely uncorrelated. Finally we obtain z, the stego image containing both robust and fragile watermarks.

[**0036**] The robustly marked image I (containing w) is divided by the fragile algorithm into contiguous and non-overlapping blocks of indexes i,j, and result into a final stego image z (containing both w and w_{frag}); therefore y and z can be written in term of these blocks as follows:

$$y=\{y_{i,j}\},\ z=\{z_{i,j}\},\ \text{with}\ \ i=1,\ldots,\frac{M}{m}\ \ \text{and}\ \ j=1,\ldots,\frac{N}{n} \quad (2)$$

[**0037**] where M,N is the image size and m,n the block size in number of pixels (width and height respectively). Note that if the image size is not an exact multiple of the block size, one can actually take the lower integer bounds of

$$\frac{M}{m}$$

[**0038**] and

$$\frac{N}{n}.$$

[**0039**] The embedding of the fragile part w_{frag} is detailed for the block level in the pseudo-code in **FIG. 5**, and illustrated in **FIG. 2**.

[**0040**] In contrast to Wong's approach where blocks are independently hashed, our hash function takes as input the current i,j-block itself as well as some neighboring blocks (**FIG. 2**, block **7**), but the resulting code being then embedded into the i,j-block only (**FIG. 2**, block **8**): this has a crucial impact on the security of the method. Such hashing of the current block and neighboring blocks together is a first step to introduce local contextual dependencies, and could be called hash-code block chaining (HBC). In the pseudo-code of **FIG. 5**, for each block of indexes i,j, the neighboring indexes are denoted by $\eta(i,j)$, with the possible configurations examples:

$\eta(i,j)=((i-1),j-1),(i-1,j),(i-1,j+1),(i,j-1),(i,j+1),(i+1,j-1),(i+1,j),(i+1,j+1))$ (the 8 neighbors)

$\eta(i,j)=((i-1,j),(i,j-1),(i,j+1),(i+1,j))$ (4 neighbors)

$\eta(i,j)=((i,j-1),(i,j+1))$ (2 neighbors)

$\eta(i,j)=((i,j-1))$ (only 1 neighbor)

[**0041**] For those blocks which are along the image borders (i.e. $i\in\{1,M/m\}$ or $j\in\{1,N/n\}$) and for which the neighboring blocks fall outside of the image, one can just consider that the image is infinitely padded with the value 0 or just ignore out-of-range neighbors from the hash input.

[**0042**] In addition to HBC, other local or global contextual information can be included in the input of hash functions, such as current block indexes (i,j), the image size (M,N), owner-related data like in the case of robust watermarking, date and time, place, unique image identification name or number, etc. Such hashed additional information is denoted by the " . . . " in pseudo-code in **FIG. 5** (line **4**). Linking individual block hashing with both local and global contextual information is important from the security point of view, in order to defeat a large class of substitution attacks dedicated to fragile watermarking schemes.

[**0043**] Note that w_{frag} fragile blocks may or may not coincide with w robust blocks; actually fragile blocks may be sub-blocks from robust blocks for better locality in the tamper detection. However an important issue is to preserve the original robustness of the robust watermark: first, embedding the fragile part by LSB modulation of selected pixels ensures very limited modification, which is very unlikely to destroy the robust watermark which has larger amplitude; secondly, we propose to embed the fragile watermark in selected positions not belonging to the robust watermark copyright information component w_{inf}, i.e. we embed w_{frag} in positions of the reference watermark w_{ref} and in positions containing no watermark at all w_{empty}), thus fully preserving w_{inf}. This characteristic is shown by the dashed arrows transmitting the w_{inf} and w_{empty} positions in **FIG. 1** and **FIG. 2**, and by the squared points inside the image blocks in diagrams (**FIG. 2**). Thus w_{inf} is untouched, and on average at most 50% of positions in w_{ref} are altered by +1 or −1 due to the LSB modulation. Since w_{ref} and w_{empty} usually cover not more than 20% of the area of w in practical cases, this makes w and w_{frag} almost orthogonal. At the same time the visual impact of the fragile part is much lower than the visual distortions of the robust part.

[**0044**] Hybrid Watermark Extraction And Verification

[**0045**] The block diagram **FIG. 3** shows the extraction and authentication part. At the extraction stage, the robust extractor (**FIG. 3**, block **9**) first estimates the robust watermark $\hat{w}$ from the possibly attacked and tampered stego

image z', and decodes an estimate of the copyright message $\hat{b}$ (**FIG. 3**, block **10**); the possibly applied global (affine) and local geometrical distortions (RBA) are compensated for in this part.

[**0046**] The authentication part takes z' as input; re-computes signatures (**FIG. 3**, block **5**) $\tilde{s}$ from z'* (a version of z' where the LSB used for the embedding of w_{frag} have been cleared, i.e. set to 0—**FIG. 3**, block **4**); extract w_{frag} from z' and get the estimated embedded signatures $\hat{s}$ (**FIG. 3**, block **11**); outputs a tamper map $\hat{T}$ by comparing the signatures $\tilde{s}$ and $\hat{s}$ for each block (**FIG. 3**, block **12**); and finally takes a final decision d_T based on the validity of $\hat{b}$, the authentication map $\hat{T}$ and the global authenticity value A_T (the ratio of authentic blocks over all blocks, Equation 5). The embedding positions of w_{inf} and w_{empty} are transmitted as for the embedding stage to the fragile part (dashed arrows in **FIG. 3** and **FIG. 4**, and squared points in **FIG. 4**).

[**0047**] For the authentication the input image z' is divided into blocks of the same size and same positions as for the embedding process as:

$$z' = \{z'_{i,j}\}, \text{ with } i = 1, \ldots, \frac{M}{m} \text{ and } j = 1, \ldots, \frac{N}{n} \tag{3}$$

[**0048**] The block diagram **FIG. 4** and pseudo-code in **FIG. 6** show the extraction and verification of the fragile signature at the block level. We can then define an estimated authenticity value $\hat{T}_{i,j}\{0,1\}$ for each block index i,j as 1 if the block $z'_{i,j}$ and its neighbors $z'_{\eta(i,j)}$ are unmodified, and 0 otherwise, as given by the comparison operator "-" in **FIG. 4** and pseudo-code line **6** in **FIG. 6**. One possible definition of the comparison operator is:

$$\hat{T}_{i,j}=1-\delta(\tilde{s}_{i,j}-\hat{s}_{i,j}) \tag{4}$$

[**0049**] where $\delta(.)$ is the Kroneker symbol ($\delta(x)=1$ if $x=0$, and $\delta(x)=0$ otherwise) considering $\tilde{s}_{i,j}$ and $\hat{s}_{i,j}$ as binary encoded integers). At the end a global normalized authenticity measure A_T indicates the ratio of authentic blocks over the total number of blocks for the whole image, and could be defined for example as:

$$A_T = \frac{1}{\frac{M}{m}\frac{N}{n}} \sum_{i=1}^{M/m} \sum_{j=1}^{N/n} 1 - \hat{T}_{i,j} \tag{5}$$

[**0050**] with the following interpretation:

$A_T=1 \Rightarrow$authentic image

$0<A_T<1 \Rightarrow$partially tampered image

$A_T=0 \Rightarrow$non-authentic image (6)

[**0051**] Tamperproofing/Authentication Decision

[**0052**] At the end the generic following decision d_T can then be made concerning the authenticity or the tampering of the image z', based on the diagnostics of both robust and fragile watermarks:

[**0053**] 1. $\hat{b}$ is correctly decoded and $A_T=1$: the image is fully authenticated and has not been tampered.

[**0054**] 2. $\hat{b}$ is correctly decoded but $A_T<1$: if $A_T>0$ then only malicious local modification probably occurred: we partially authenticate the image and we point out modified regions (where $\hat{T}_{i,j}=1$); if $A_T=0$, we reject the image as globally non authentic, but since $\hat{b}$ is valid at the same time, we can claim that a copy attack may have occurred, and the origin of the copied watermark may be easily verified.

[**0055**] 3. $\hat{b}$ failed or multiple $\hat{b}_k$, k=1,2, . . . are decoded, and $A_T>0$: if $A_T=1$, then we can immediately claim that an advanced substitution attack may have been applied, such as the collage attack; if $A_T<0$, we can suspect the same for example if some of the $\hat{T}_{i,j}$ were 0 (i.e. authentic block) simultaneously for at least two regions containing distinct valid $\hat{b}_k$ and $\hat{b}_1$.

[**0056**] 4. $\hat{b}$ was not decoded and $A_T=0$: we reject the image as globally non authentic, and at the same time we can not claim any copyright.

[**0057**] Simple attacks are easily detected in items 1, 2, and 4 above. If the marked image has been simply replaced by another one, the input will obviously be rejected; any local modification in a valid image will destroy signatures in the altered blocks. A copy attack further corresponds to the second item when $A_T=0$: the copy of a robust watermark w from another image would make the robust message $\hat{b}$ still decodable, but all signatures would not match ($\hat{T}_{i,j}=1\ \forall i,j$); therefore by rejecting this case, our hybrid approach is resistant to the copy attack.

[**0058**] Item 3 above is a particular case: if the robust watermark is altered or is not coherent, then we could expect $\hat{T}_{i,j}=1$ at least in regions where the robust watermark w was destroyed or changed, resulting into signatures mismatches (since w is included in the hash functions input). However this situation can occur when different robust watermarks are present, all embedded with the same key; note that our robust watermarking algorithm, which works at the local level to achieve resistance to the RBA [2,11], can successfully decode different messages $\hat{b}_k$. This situation appears if a sophisticated substitution attack was applied, which we can name as collage attack: the composition of an image from various source images, all watermarked with the same key, can be constructed without being detected by the fragile algorithm if this latter was not designed properly.

[**0059**] In general the analysis of the $\hat{T}_{i,j}$ locally, with respect to blocks from which $\hat{b}$ or $\hat{b}_k$ were correctly decoded, can be useful for both items 2 and 3 in order to get more detailed diagnostics about what probably happened to the image.

[**0060**] Security Of Hybrid Watermarking

[**0061**] Many attacks or malicious changes can be mounted against hybrid-watermark documents, targeting the robust watermark and the fragile watermark, as well as interactions or relationship between both parts. Since attacks on robust watermarking have been already widely discussed, here we will mainly focus on intentional attacks specific to the fragile part. Unlike those dedicated to robust watermarks, the general goal of attacks on fragile watermark is not to remove the information (otherwise the host data would be invalidated), but rather to perform tampering or manipulations without being detected at the verification stage.

[0062] In a fragile approach, any change is in theory detected, since the change of one pixel would result into the mismatch of embedded and recomputed hash codes for the corresponding block. However, the retained method for the generation of signatures and their embedding should be carefully designed in order to keep resistance to various tampering attack. It has been noticed very soon that simple schemes based on the hashing of non-overlapping and independent blocks like in Wong's approach were vulnerable to various tampering attacks, and especially to substitutions attacks described by Holliman et Memon [19], and Barreto et al. [20]. Other weaknesses could result from the design of the used cryptographic primitives and the way they are implemented, the signatures lengths, etc. Many of these attacks have been pointed out and advanced solutions proposed, in particular by Barreto et al. [20]. Further, in a hybrid approach, the information given by the joint use of robust and fragile watermarks can be exploited in order to increase the security. Below we describe the most significant attacks, and then propose countermeasures against them, covered by the scope of this invention.

[0063] Substitution Attacks

[0064] The most simple of these attacks could consist in exchanging color planes in color images, in the case where each plane is hashed separately. Therefore an obvious solution would be to hash the three color planes together. Generally, the hashing and marking of independent blocks, without any other contextual information, is vulnerable to simple copy and paste inside the same watermarked image: a few valid blocks copied from a suitable area can be pasted in another place in order to hide or to replace an object, without visible artifact; the only restriction for this attack to succeed is to respect blocks synchronization, which is not difficult when the block size is publicly known. The knowledge of the key is not required, since each block is independently authenticated by itself. If the copied area comes from another image, two cases can be distinguished: either the other image is not watermarked or is watermarked with a different key, and the copied object will be detected as tampered; or the other image is watermarked with the same key, and the copied area can be seen as authentic. Therefore the problem arises when the images used are all watermarked using the same key. By this technique, it is even possible to construct a fake image by pasting together areas coming from different images. This type of attack, aiming at replacing parts or the entire image, are known as substitution attacks. The different variants above could be named copy-and-paste attack when an object is pasted into a valid image, or the already mentioned collage attack when a composite image is generated from several marked source images.

[0065] In the same framework, an advanced version of the substitution attack can be mounted using vector-quantization (VQ) techniques [19], which is known as the vector quantization attack, or the Holliman-Memon attack. This is an enhancement of the collage attack, which is able to construct an arbitrary composite image using the smallest possible areas—the blocks themselves. For this purpose the attacker first needs to gather a set of watermarked images, all marked with the same key. These blocks are sorted in order to regroup together blocks corresponding to the same embedded logo or the same block-synchronization used for the fragile watermark embedding; this is actually the case for all blocks having the same index i,j, if the division is made in the same order for all images. Then the attacker can reconstruct a completely new image by picking up, for each block synchronization, a block from the group corresponding to the same synchronization, which is visually the closest to the image to be constructed. This approach is merely the same as vector quantization, where we can think of a "code book" as the collection of all blocks that would be correctly decoded. The gathering of a sufficient number of set of images marked with the same key is quite realistic, for example from a database; actually a small number of images (i.e. less than 10) is often sufficient to apply this attack, with very little visual artifact. This attack can also be named the cut-and-paste attack [20].

[0066] Cryptographic Attacks

[0067] The underlying cryptographic primitives are obviously important too. Secure and well-studied cryptographic algorithms should be used, using keys of sufficient lengths. However since the fragile watermarking is based on hash codes and signatures, one important point to mention is the lengths of such hash-codes. Wong's scheme uses hash-codes of 64 bits length. It could be believed 64 bits are secure enough, since an exhaustive search would take $2^{64} \approx 1.84 \times 10^{19}$ tries to find an input resulting into a given hash code.

[0068] However the possible weakness here rather consists in the possibility to find hash-code collisions, i.e. two blocks from different images (watermarked with the same key) which result into the same hash-code—which would help for generating a faked image. Here the problem is not to find input which result into one particular hash-code, but to find two arbitrary codes which collide. Collision search can be performed on a set of images assuming they are all watermarked with the same key, without knowing the actual key by comparing the bits used for the embedding (the LSB selected positions in our case). This problem is subject to the anniversary paradox [21], which states that for hash-codes of n bits, the probability to obtain a collision is already equal to about 50% when only $\sqrt{n}$ random blocks are gathered. With hash-codes of 64 bits, only $2^{32} \approx 4.29 \times 10^9$ block samples are needed to have already 1 chance over 2 to get a collision. In a concrete situation, an image of 1000×1500 pixels can be divided into about 5766 blocks of size 16×16; therefore 744879 images would contain the 2^{32} blocks needed to mount an anniversary attack with 64 bits hash-codes. The possible availability of large databases of images all protected with the same key would make this attack almost realistic, therefore Wong's scheme is vulnerable to the so called anniversary attack. Then to achieve higher security level, it is recommended to use hash-codes of at least 128 bits: in this case the anniversary attack would actually require 2^{64} block samples as previously expected.

[0069] We therefore discuss the following countermeasures to defeat all known cryptographic attacks described above on joint robust and fragile watermarking.

[0070] Hash-Code Block Chaining (HBC)

[0071] Substitutions attacks are made possible mainly due to the independence of blocks. The solution is therefore to introduce local dependencies as well as other local contextual information. First hashing the three planes together in color image prevents from color swapping. Secondly, hashing each block with some of its neighbors (HBC) makes substitution attacks more difficult to mount; HBC is equiva-

lent to the overlapping blocks proposed by Coppersmith et al. [22]. Thirdly, we propose to hash additional global and local contextual information with each block, including the image size, the current block indexes, and other unique random information for each image. Fourthly, the anniversary attack could be simply defeated by using signatures of sufficient lengths.

[0072] Un-Deterministic Hash-Code Chaining (HBC)

[0073] Barreto et al. [20] further show that even with HBC, a fragile watermarking algorithm is not secure against a more sophisticated substitution attack which consider groups of chained blocks together instead of single blocks. They call this attack, which is an enhancement of the cut-and-paste attack, the transplantation attack; increasing the number of chained blocks does not help, since this attack would just need to consider larger groups of chained blocks. Therefore they proposed to enhance HBC by chaining previous hash-codes too as hash-code block chaining version 2 (HBC2), combined with un-deterministic signatures: first, the hash function takes as input not only the neighboring blocks, but also neighboring (and already computed) signatures; secondly, "un-deterministic signature" means that two strictly identical input hashed using the same key produce two randomly different signatures: consequently the assumption that images are all watermarked with the same key does not help anymore, since signatures always look random to an attacker. Note that any deterministic hash function may be turned into an un-deterministic one by using a random salt, taken as input and appended to the signature. The salt consists in a random string r which is appended to the hash-code h or the signature s; at the embedding stage r is included in the input of the hash function as:

$$h=H_k(r, \ldots)$$

$$s=S_k(r, \ldots) \quad (7)$$

[0074] and both r and h (or s) are embedded as (r,h) (or (s,h), since this salt r is needed for the verification stage (k being the user key).

[0075] Global And Local Contextual Information Hashing

[0076] Unfortunately, the previously given solutions are still not enough to ensure full resistance against the collage attack mentioned above, when areas large enough are copied and pasted: only the boundaries between areas coming from different images are detected as tampered, but nothing can tell us that these different areas come from different sources. We could then think of hashing the binary representation of blocks indexes (i,j), or the image size (M,N) as well. However the collage attack is still possible by preserving the blocks original positions and by using images of the same size.

[0077] A second solution we can think of is then to hash some global additional information, chosen unique for each image; an identification number (ID) could be used for this purpose. The consequence of this method is that given an image ID, only the corresponding areas will be authenticated, but the pasted areas will be rejected. Any additional global and local information hashed is then represented by the " . . . " in pseudo-codes in **FIG. 6** (line **4**).

[0078] Embedded Hashed Unique Stamps

[0079] Since any hashed additional information is also needed at the verification stage, it should be stored with its corresponding key, which could make the images ID method above inconvenient for many applications. We therefore propose a third solution, which is to store such additional information within the hash code—an unique stamp for each image, in encrypted form. In this case such stamp can be random and does not need to be stored separately from the image anymore, and just acts as an additional salt (equation 10) which is the same for all blocks of one image, but is different from one image to another. Moreover this stamp can even carry useful information, its only requirement being to be unique for each image. We actually propose to use a time-stamp indicating the date and time of embedding, plus other specific information if necessary. The time-stamp is included in the input of the hash functions, and at the verification stage is used before recomputing the signature. In this approach, signatures will be authenticated again in every copied area again, but the extraction of different time-stamps can alerts us that a collage attack probably occurred. With this method it is even possible to count the number of copied areas and to localize them.

[0080] Jointly Exploiting The Robust Watermark

[0081] Finally, the proposed hybrid watermarking scheme gives us an opportunity which current state-of-art fragile only schemes do not have. From our part, we propose to use the extraction result from both the robust part and the fragile part, in addition to every countermeasure detailed above. Consequently a more precise diagnostic can be given.

[0082] First, the collage attack detection can be enhanced, since the robust algorithm could either fail, or decode different independent messages correctly when the RBA-resistant version of our robust method [2,11] is used (due to the fact that it extracts the watermark at the local level). This feature corresponds to the item 3. of the decision enumeration given in the "Tamperproofing/authentication decision" paragraph. When used jointly with the stamp/time-stamp approach, we have then another criteria to detect such attacks; further, if the same robust message was embedded in all parts (resulting into only one decoded message), the embedded stamp approach can still distinguish the different parts.

[0083] Secondly, as we concluded in the "Tamperproofing/authentication decision" paragraph, joint robust and fragile watermarking is resistant to the copy attack: it is generally easy to estimate the robust watermark, and to copy it into another unmarked image. The robust watermark will still be correctly decoded from the new image, but the fragile watermark will fail. Even if the fragile part is also copied to the destination image (e.g. by copying the LSB), the signatures would not match since the input of the hash functions are changed.

[0084] Summary Of Security Measures

[0085] Consequently, we can summarize the main security measures that could be implemented by the following items:

[0086] 1. use hash-codes of sufficient lengths: hash-codes of at least 128 bits should be used, and we propose the MD5 (128 bits) or the SHA (160 bits), in order to defeat the anniversary attack.

[0087] 2. chain blocks in hash-coding (HBC): for each block compute the hash-code of this blocks plus

neighboring blocks, in order to defeat simple substitution attacks and the cut-and-paste attack.

[0088] 3. chain signatures in hash-coding (HBC2): in addition to HBC, make hash-codes also dependent from at least one previously computed signature.

[0089] 4. use un-deterministic hash-coding: un-deterministic hash-codes or signatures, jointly used with HBC2 above, in order to defeat advanced attacks such as the transplantation attack.

[0090] 5. hash extra global and local information: hashing the indexes i,j of the current block makes block synchronization necessary for an attack to succeed; hashing the image's size M,N restrict attacks to images of the same size; hashing an unique ID for each image makes substitutions attacks merely infeasible, but may be not practicable in many applications (this ID should be stored separately).

[0091] 6. hash and embed an unique stamp: hash an unique stamp for each image (e.g. a random ID), which is embedded beside the signatures, to defeat the collage attack, and to allow to distinguish and localize pasted areas; can also carry useful information such as a time-stamp. This method can be used in place of the unique ID approach of item 5.

[0092] 7. use jointly information from the robust and fragile parts: analyzing the decoding of both parts gives us a more powerful diagnostic, in order to confirm the detection of the collage attack, and to defeat the copy attack.

[0093] Therefore, using first countermeasures suggested for the fragile part, and secondly by taking advantage of the hybrid approach by exploiting the additional information coming from the robust part, we can expect a highly robust and secure approach for both copyright protection, tamper-proofing and authentication.

[0094] Conclusion

[0095] This patent presents a hybrid robust watermarking scheme for visual data, which combines copyright protection, detection of tampering, and authentication. For this purpose we jointly used the highly robust watermarking scheme we previously developed, and a fragile watermark based on local signatures. Note that little work has been done today on such hybrid robust and fragile.

[0096] The robust part exhibits high robustness to signal processing attacks, geometrical transforms as shown by the Stirmark [23] results, as well as robustness to printing and rescanning. The algorithm is resistant against random local geometrical distortions too as well as to projective and non-linear transforms, and can also defeat collage attack by extracting and decoding the copyright information locally.

[0097] The fragile part does not decrease the robustness of the robust part, due to its nearly orthogonal embedding with respect to the robust information. Exploiting the diagnostics from both the robust and the fragile parts, the algorithm is resistant against different kinds of attacks, including the copy attack and the collage attack.

REFERENCES

[0098] 1. S. Voloshynovskiy, F. Deguillaume, S. Pereira, A. Herrigel and Thierry Pun, "Method for adaptive digital watermarking robust against geometric transform", European Patent Application PCT/IB00/01089, Aug. 3, 2000, accepted in May 2001.

[0099] 2. S. Voloshynovskiy, F. Deguillaume and T. Pun, "A Method for Digital Watermarking Robust Against Local and Global Geometrical Distortions and Projective Transforms", U.S. patent application USPTO 60/327,097, Oct. 4, 2001.

[0100] 3. F. Deguillaume, S. Voloshynovskiy and T. Pun, "A Method for the Estimation and Recovering of General Affine Transform", U.S. patent application USPTO 10/051,808, Jan. 17, 2002.

[0101] 4. B. Chen and G. W. Wornell, "Quantization Index Modulation: A class of provably good methods for digital watermarking and information embedding", Proceedings of IEEE International Symposium on Information Theory, vol. 47 num. 3, pp. 1423-1443, May 2001.

[0102] 5. J. J. Eggers and J. K. Su and B. Girod, "A blind watermarking scheme based on structured codebooks", IEE Conference on Secure Images and Image Authentication Proceedings, pp. 4/1-4/6, Apr. 10, 2000, London, UK.

[0103] 6. S. Voloshynovskiy, F. Deguillaume and T. Pun, "Content adaptive watermarking based on a stochastic multiresolution image modeling", Tenth European Signal Processing Conference EUSIPCO'2000, September 2000, Tampere, Finland.

[0104] 7. J. J. K. Ó Ruanaidh and T. Pun, "Rotation, Scale and Translation Invariant Spread Spectrum Digital Image Watermarking", Signal Processing, vol. 66 num. 3, pp. 303-317, 1998.

[0105] 8. S. Pereira, J. J. K. Ó Ruanaidh, F. Deguillaume, G. Csurka and T. Pun, "Template Based Recovery of Fourier-Based Watermarks Using Log-polar and Log-log Maps", Int. Conference on Multimedia Computing and Systems, Special Session on Multimedia Data Security and Watermarking, June 1999.

[0106] 9. M. Kutter, "Digital image watermarking: hiding information in images", EPFL, August 1999, Lausanne, Switzerland.

[0107] 10.F. Deguillaume, S. Voloshynovskiy and T. Pun, "Method for the estimation and recovering of general affine transforms in digital watermarking applications", IS&T/SPIE's 14th Annual Symposium, Electronic Imaging 2002: Security and Watermarking of Multimedia Content IV, vol. 4675, Jan. 20-25, 2001, San-Jose, Calif., USA.

[0108] 11.S. Voloshynovskiy, F. Deguillaume and T. Pun, "Multibit Digital Watermarking Robust Against Local Nonlinear Geometrical Distortions", IEEE ICIP2001, pp. 999-1002, October 2001, Thessaloniki, Greece.

[0109] 12.G. L. Friedman, "The trustworthy digital camera: restoring credibility to the photographic image", IEEE Transactions on Consumer Electronics, vol. 39, pp. 905-910, November 1993.

[0110] 13.P. W. Wong, "A Public Key Watermark for Image Verification and Authentication", IEEE International Conference on Image Processing '98 (ICIP'98) Proceedings, vol. 1, MA11.07, 1998.

[0111] 14.M. Kutter, S. Voloshynovskiy and A. Herrigel, "Watermark copy attack", Wah Wong, Ping and Edward J. Delp, IS&T/SPIE's Electronic Imaging 2000 SPIE Proceedings, vol. 3971, January 2000, San Jose, Calif., USA.

[0112] 15.M. Wu and B. Liu, "Watermarking for image authentication", IEEE International Conference on Image Processing '98 (ICIP'98) Proceedings, TA10.11, Focus Interactive Technology Inc., October 1998, Chicago, Ill., USA.

[0113] 16.J. Fridrich and M. Goljan, "Protection of Digital Images Using Self Embedding", Symposium on Content Security and Data Hiding in Digital Media, May 1999, New Jersey Institute of Technology, USA.

[0114] 17.J. Fridrich, "A Hybrid Watermark for Tamper Detection in Digital Images", ISSPA'99 Conference, August 1999, Brisbane, Australia.

[0115] 18.S. Voloshynovskiy, A. Herrigel, N. Baumgaertner and T. Pun, "A Stochastic Approach to Content Adaptive Digital Image Watermarking", Lecture Notes in Computer Science: Third International Workshop on Information Hiding, Springer, vol. 1768, pp. 211-236, September/October 1999, Dresden, Germany.

[0116] 19.M. Holliman and N. Memon, "Couterfeting attacks on oblivious block-wise independent invisible watermarking schemes", IEEE Transactions on Image Processing, vol. 9, num. 3, pp. 432-441, March 2000.

[0117] 20.P. S. L. M. Barreto, H. Y. Kim and V. Rijmen, "Toward a Secure Public-key Blockwise Fragile Authentication Watermarking", IEEE ICIP2001, pp. 494-497, October 2001, Thessaloniki, Greece.

[0118] 21.A. J. Menezes and P. C. van Oorschot and S. A. Vanstone, "Handbook of Applied Cryptography", CRC Press, ISBN 0-8493-8523-7, October 1996.

[0119] 22.D. Coppersmith, F. Mintzer, C. Tresser, C. W. Wu and M. M. Yeung, "Fragile imperceptible digital watermark with privacy control", IS&T/SPIE Electronic Imaging'99, Session: Security and Watermarking of Multimedia Contents", January 1999, San Jose, Calif., USA.

[0120] 23.F. A. P. Petitcolas, "Stirmark benchmark 4.0", http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/, 2002.

We claim:

1. A method for generating watermarked data z based on some original data x, wherein said robust watermark w contains multi-bit informative w_{inf} and reference w_{ref} watermarks encoded and embedded in such a way as to resist against attacks, and an authentication watermark w_{frag} contextually encoded, encrypted and embedded in orthogonal or almost orthogonal positions with respect to the robust watermark w, comprising the steps of:

(a) encoding said multi-bit message b, possibly using any error correction code (ECC),

(b) generating said w as a function of said key k and said message b encoded and/or encrypted as codeword c, where said w consists in a said informative watermark w_{inf} and said reference watermark w_{ref},

(c) generating said authentication watermark w_{frag} as a function of said key k and said contextual information,

(d) embedding said robust watermark w into said original data x to get said robustly marked data y,

(e) embedding said authentication watermark w_{frag} into said robustly marked data y in orthogonal manner to said informative watermark w_{inf}, resulting into the final marked data z

whereby said reference watermark w_{ref} assists amongst others in the estimation and recovering from local and global geometrical image alterations, channel state estimation, verification of reliability, fast detection of the said robust watermark w presence in said z and synchronized decoding of said error correction codes (ECC), and said authentication watermark w_{frag} assists in tampering detection, authentication and prevention of estimation of robust watermark w or informative watermark w_{inf} with following copying to another target media known as the copy attack, and identification of the reliability of said informative watermark w_{inf}, and furthermore whereby said z is visually indistinguishable from said x.

2. The method of claim 1 wherein said function uses perceptual masking M while adding said watermark w and said authentication watermark w_{frag} to said x in the spatial domain or some transform domain.

3. The method of claim 1 wherein said authentication watermark w_{frag} contains global and/or local encrypted contextual information about data x such as data size, unique data ID, name, index or random unique stamp that is the same for one data but is different to another data to resist against collage attack and copy attack, and produces local data dependent signatures or local message digest code (MDC).

4. The method of claim 1 and **3** wherein said unique stamp additionally carries information about date, time and other specific information identifying the particularities of embedding process that even enables the identification of copied areas and their localization.

5. The method of claim 1 wherein said encoded and encrypted authentication watermark w_{frag} uses as input global and local information about the original data x and/or the robustly watermarked data y, including local blocks of the data, local blocks indexes, data global size, etc., and wherein all key dependent positions and bit planes where said authentication watermark w_{frag} is to be embedded are excluded from the information used as input for the generation of this w_{frag}.

6. The method of claim 1 wherein said encoded and encrypted authentication watermark w_{frag} comprises a key-dependent regular (such as square blocks), or any irregular spatial allocation structure.

7. The method of claims **1**, **5** and **6** wherein said blocks are hashed using information about neighboring blocks to defeat simple substitution attacks and cut-and-paste attacks.

8. The method of claims **1** wherein said authentication watermark w_{frag} is embedded into contiguous and non-overlapping blocks with predefined indexes which may be generally key-dependent.

9. The method of claim 1 wherein said informative w_{inf} and authentication watermarks w_{frag} are used jointly to detect the collage and the copy attacks that can not be achieved only based on their independent usage.

10. The method of claim 1 and **9** wherein said informative w_{inf} and authentication w_{frag} watermarks are used jointly on the local blockwise level to detect the sequence of the applied attacks and to distinguish the multiple watermarks embedded with the same technology and the same key but possibly with different messages, and to identify the original informative message b initially embedded into said media x.

11. The method of claim 1 wherein said informative watermark w_{inf} and/or authentication watermark w_{frag} is used to detect the tampered regions or the boundaries of objects, which have been copied from other medias even with the preservation of their block indexes and positions.

12. The method of claim 1 wherein said original data x is video, audio or image data.

13. The method of claim 1 applied to video data, wherein a plurality of watermarked video frames is generated.

14. The method of claim 1 wherein said function operates in the spatial domain, Discrete Cosine Transform (DCT) domain, Discrete Fourier Transform (DFT) domain, wavelet domain, or any other transform domain, or some combination thereof.

* * * * *