Tailored Reversible Watermarking Schemes for Authentication of Electronic Clinical Atlas

[1]Feng Bao, [2]Robert H. Deng, [3]Beng Chin Ooi, [2,3]Yanjiang Yang
[1]Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
[2]School of Information Systems, Singapore Management University, Singapore 259756
[3]School of Computing, National University of Singapore, Singapore 117543

***Abstract*—It is accepted that digital watermarking is quite relevant in medical imaging. However, due to the special nature of clinical practice, it is often required that watermarking do not introduce irreversible distortions to medical images. Electronic clinical atlas has such a need of "lossless" watermarking. We present two tailored reversible watermarking schemes for clinical atlas by exploiting its inherent characteristics. We have implemented the schemes and our experimental results look very promising.**

***Index Terms*—Electronic Clinical Atlas, Authentication, Reversible Watermarking, Security.**

I. Introduction

EFFICIENT maintenance of medical data in electronic format is crucial in enhancing the quality and efficacy of healthcare provision through efficient information sharing. However, along with the benefits is the growing concern about the security of digital medical information. In a broad sense, security issues pertaining to digital medical data are categorized into the following aspects [1], [2]:

1) *Confidentiality*: individual privacy of patients as well as physicians implies that medical data must be protected from inappropriate disclosure. Only authorized users with appropriate rights are offered access to the data.
2) *Authentication*: authentication (reliability in [1], [2]) of medical data can be further classified into (1) integrity: medical information must be assured of its intactness; (2) authenticity: credibility must be given to the users that the underlying data are up to what they are claimed to be.
3) *Availability*: medical data must be guaranteed to be readily available to the authorized uses.

Medical imaging constitutes an important part of digital medical data. Clearly, the attacks that threaten digital medical information as a whole (see e.g., [3], [4]) would also apply to medical images. In this paper, we are interested in exploring the authentication aspect of medical images.

Multimedia authentication inherits many characteristics of generic data authentication using cryptographic primitives, such as integrity verification, authenticity verification and nonrepudiation [5]. However, multimedia authentication has its own unique features that make the techniques for generic data authentication not suffice and sometimes undesirable. For example, an image changing from one format to another without losing visual content should be deemed authentic in multimedia authentication, whereas this turns out to be hard to achieve by applying generic data authentication that uses Message Authentication Code (MAC) or digital signature [5]. As a result, multimedia authentication is normally accomplished by digital watermarking [6], [7]. Digital watermarking can be classified into *copyright watermarking* and *authentication watermarking*, based on the purposes it is intended for. We note watermarking can also be used to establish a channel for carrying, e.g., meta-data (e.g., [8], [9], [10]). In a strict sense, this application of watermarking belongs to the area of steganography since the objective is to hide medical data in a host image for data secrecy purposes. In copyright watermarking, the inserted mark, upon extraction, asserts ownership of the underlying data. To achieve this end, copyright watermarking must be robust so that the inserted mark cannot be easily removed. In contrast, authentication watermarking is designed to prove the integrity and authenticity of the underlying data. Authentication watermarking can be further classified into *hard authentication* and *soft authentication* [11]. Hard authentication detects and rejects any modification to the multimedia content except for lossless compression and format conversion with equivalent visual content, while soft authentication allows for *admissible* content modifications while rejects *malicious* manipulations. The lack of a clear distinction between admissible and malicious operations is one of the main challenges for soft authentication.

Clearly, authentication watermarking represents a viable solution to authentication of medical images in our context. We note that medical practice is very strict with the management of medical data for the clinical, ethical and legislative reasons [1], [12]. Thus in many cases it is often desirable that watermarking itself does not introduce any distortion to the medical images for the purpose of data authentication. For example, images used for court proof or for data archiving are strictly forbidden to be altered. However, it is well known that normally, watermarking including hard authentication watermarking introduces perpetual distortions to the original data. This suggests that most of the existing authentication watermarking techniques are not satisfactory for medical imaging. To authenticate medical images, a watermarking scheme must possess *reversibility* - the ability to recover the exact original content from a watermarked image. Watermarking working in such a reversible way is referred to

Part of the work has been published in IEEE. International Conference on Acoustics, Speech, and Signal Processing, 2003.

as reversible (lossless, invertible, distortion-free, erasible, etc.) watermarking [13], [14], [15], [16], [17], [18] in the literature. Reversible watermarking is regarded as a special form of hard authentication [11]. In this paper, we develop reversible watermarking schemes that are tailored for authentication of electronic clinical atlas [19], [20], [21], a particular type of medical images in palette format. Our motivations of studying reversible watermarking for electronic clinical atlas are (1) this is in conformance with the strict medical practice in general; (2) The atlas, in combination with other medical imaging modalities, has a wide spectrum of applications in functional neurosurgery, brain mapping, neuroeducation, and so on (see e.g., [20], [21], [22]). Usability of these applications, many of which run in an online interactive mode, is determined largely by the accuracy of the underlying atlas and the brain data. This highlights the need for the authentication of the atlas by reversible watermarking; (3) as we shall discuss shortly, the atlas has some unique features, which make the existing reversible watermarking schemes inadequate.

The remainder of this paper is organized as follows. In Section II, we provide a brief background on multimedia authentication, and review related work on reversible watermarking and authentication of medical images, respectively. In Section III, we propose two tailored reversible watermarking schemes for authentication of electronic clinical atlas, by exploiting the very nature of the atlas. We present experimental results in Section IV and Section V concludes the paper.

II. Background and Related Work

A. Background

Authentication of generic data has been well studied in cryptography [5], where either a Message Authentication Code (MAC) or a digital signature is computed on and appended to a message [5]. The MAC or the signature is treated as accessory data, semantically distinguishable from the message itself. Though the generic approach does not in any way affect the fidelity of the data to be authenticated, it has some intrinsic limitations when applied for authentication of multimedia content: (1) a malicious intruder can easily separate and ruin the ancillary authentication payload, thereby disabling the authentication functionality; (2) the tagged payload is susceptible to normal file format conversion, so much so that even a simple re-save operation will render it useless; and (3) the generic data authentication technique lacks the property of localization, i. e., the ability to determine the locations where modifications were made.

These limitations can be overcome using authentication watermarking which embeds a watermark payload into multimedia data, so that the payload is semantically combined with the data. The ability to localize tampering or determine the severity of tampering is another highly desirable feature of authentication watermarking [11], [13], [23]. Moreover, depending on the integrity criteria, authentication watermarking authenticates multimedia content in such a flexible way that hard authentication rejects every alteration while soft authentication passes admissible operations. A stronger requirement in some scenarios e.g., in military or healthcare, is the recovery of the exact original content from the watermarked data. Regular watermarking does not possess such an ability. This calls for a special type of authentication watermarking, reversible watermarking, which allows for the recovery of the original data from the watermarked data. The fundamental principle of reversible watermarking is demonstrated in Figure 1. Note that the "recovery" process is where reversible watermarking differs from regular authentication watermarking. By "recovery", reversible watermarking can completely remove the introduced distortion and reconstruct the exact content of the original data, provided that the watermarked image is authentic.

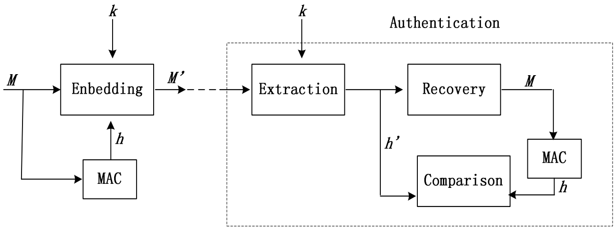


Fig. 1. Flow chart for reversible watermarking.

B. Reversible Watermarking

Reversibility of digital watermarking has been explored for quite some time [24], [25], and progress within recent years proved its viability for authentication multimedia. The first known reversible watermarking scheme was due to [14], and the method in [13] that followed employed the same idea of modulo addition to extend the classic patchwork algorithm. However, the embedding capacity of these methods is limited and annoying artifacts may be caused in the underlying images. For these reasons, they are not very satisfactory for practical use. To overcome these limitations, Fridrich *et al.* [15], [16], [17] developed an elegant method for achieving reversible watermarking which works as following: a bit plane that can be randomized without causing noticeable artifacts is losslessly compressed, so that the original bit stream B is shortened to B' and the newly created space of $|B| - |B'|$ bits is used to accommodate the authentication payload h. Then the original bit plane is replaced by $B'||h$ for the purpose of watermarking, where $||$ denotes concatenation. During the authentication verification, B' is decompressed to obtain B, which is in turn used to recover the original bit plane for the reconstruction of the original image. Practical schemes for virtually any commonly used file format including raw, uncompressed formats (BMP), lossy or transform formats (JPEG), and palette formats (GIF, PNG) were presented in [17]. As we will see later however, the methods for palette format used by Fridrich *et al.* cannot be used to reversibly watermark electronic clinical atlas due to the unique features of atlas. Fridrich *et al.*'s method is a general approach, whereas it is not always optimal in certain aspects such as embedding capacity and image quality preservation.

Reversible watermarking proposed in [18], [26], [27] often provide higher embedding capacity. In particular, the method in [18] creates extra space for the authentication payload by virtue of Difference Expansion, in conjunction with Generalized Least Significant Bit Embedding. In contrast, the scheme in [26] adopts a totally different method to achieve reversibility, that is, circular interpretation of bijective transformations. Compression on the original data is avoided in [18], [26]. An added feature of [26] is that the watermarked content can endure certain lossy processing, and virtually none of the other methods has this property.

Localization is quite useful in some circumstances. The method in [28] is the only known reversible watermarking scheme that provides localization.

C. Authentication of Medical Images

The demand for authentication of medical images is enormous, due to the increasing needs of data exchange within medical community and with external parties such as research institutes. The work in [29] highlighted the perspective of applying generic data authentication techniques such as one-way hash function and digital signature for the authentication of medical images, while [12] demonstrated the applicability of digital signature in medical imaging.

The relevance of authentication watermarking in medical imaging for reversible watermarking and integrity control has been extensively discussed [1]. The work of [30] was based on the fact that a medical image is normally allowed to be separated into Regions of Interest (ROIs) and Regions of Non-Interest (RONIs); the ROIs must be strictly preserved while the RONIs can be allowed for some modifications; consequently, the authentication payload of ROIs is inserted into the RONIs. A possible weakness of this method is that if an adversary can identify the RONIS, then the embedded authentication payload can be totally erased, thereby disabling the authentication functionality. The work in [31] made improvements over [30] by repeatedly inserting the payload closely around the ROIs. Moreover, the payload can be used to repair the ROIs in the case of minor alterations to the ROIs. A watermarking scheme for DICOM format images was proposed in [32] to guarantee the genuine link between a delivered image with the root part of its header. Specifically, the authentication payload on the information of the referring physician is inserted in the image content by a watermarking scheme.

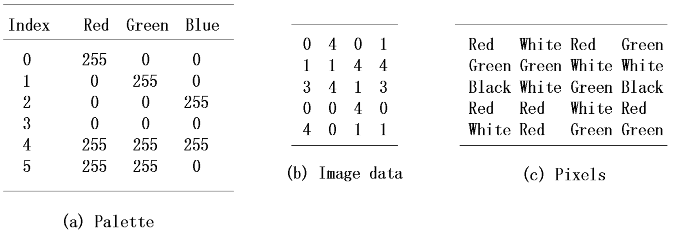
III. Tailored Reversible Watermarking schemes for Electronic Clinical Atlas

In this section, we first give a brief introduction to electronic clinical atlas, discussing its uniqueness and explaining why the existing reversible watermarking schemes are not effective in watermarking the atlases. We then propose two tailored schemes to solve the problem.

A. An Introduction to Electronic Clinical Atlas

Electronic clinical atlas in palette format [19], [20], [21] is a special type of medical images: each structure contained in an atlas is filled completely with a single color. Figure 8 (left) shows a typical atlas with dimension of 705×820.

The image file of an atlas semantically includes two parts: one is the *palette* (file header) that lists color indices and their corresponding RGB components, e.g., Figure 2(a); the other part is the *image data* (file body) that represent each pixel by a color index, e.g., Figure 2(b). In Figure 2, the atlas containing a palette as shown in 2(a) and the image data as in 2(b) actually represents pixel values as shown in 2(c). It is clear that some regular embedding operations such as LSB modification of the image data is not appropriate for watermarking electronic clinical atlas, because the image data of an atlas are color indices rather than RGB values, so indices of similar values may refer to quite different colors.



Index	Red	Green	Blue
0	255	0	0
1	0	255	0
2	0	0	255
3	0	0	0
4	255	255	255
5	255	255	0

(a) Palette

0	4	0	1
1	1	4	4
3	4	1	3
0	0	4	0
4	0	1	1

(b) Image data

Red	White	Red	Green
Green	Green	White	White
Black	White	Green	Black
Red	Red	White	Red
White	Red	Green	Green

(c) Pixels

Fig. 2. Semantic content of an atlas file. (a) The palette. (b) The image data. (c) The pixel values.

We next summarize in the following the uniqueness of electronic clinical atlas.

- **U1.** Most of the atlases are not "natural" images as opposed to the "natural" images that are commonly seen in daily life. A "natural" image normally presents smooth scene. In contrast, every structure in an atlas is of pure color (e.g., structures in the atlas in Figure 8 (left) are homochromous), so that (1) the boundary along two adjoining structures is clear-cut; (2) any alteration within a structure by a different color would appear definitely noticeable.
- **U2.** An atlas normally does not use up 256 colors, which is the upper bound that most palette images assume. On average, 30-100 colors are used to label the structures in an atlas.
- **U3.** While the content of an atlas should be strictly protected, the palette of it can be permuted provided that the content (piexels) of the atlas is not affected. For example, the following scenario should be allowed: in Figure 2(a), the entries of (0, 255, 0) and (0, 0, 255) are permuted in the palette, and 1's in the image data (Figure 2(b)) are changed to 2's and 2's are changed to 1's. This does not change the actual pixels.

In principle, we can exploit the existing reversible watermarking schemes that were designed for palette images to fulfil the task of reversibly watermarking electronic clinical atlas. Unfortunately, the only known such schemes that were proposed by Fridrich *et al.* in [17] actually turn out to be unable to meet the needs due to the very nature of the atlases as listed earlier. To see this, we check respectively the two cases distinguished in [17] with electronic clinical atlas as follows.

Case 1. *Palette with fewer than 256 distinct colors*: The basic idea is to make at least two entries in the palette for color c , a most frequently occurring color in the image to be watermarked. If c already has two or more entries, nothing needs to be done. Otherwise, simply allocating an empty entry for c in the palette. This is feasible as the palette is not used up. We then suppose the indices of two entries of c in the palette are i and j, respectively. Then 0 is associated with i and 1 is associated with j. In embedding, the image is scanned in a defined pattern, e.g., row by row. When a pixel with color c is encountered, its color index is changed to i if a bit 0 is to be embedded and changed to j otherwise. In extraction, the image is scanned in the same pattern, and the embedded binary stream is obtained depending on the indices i, j of color c. Embedding in this way does not alter the pixels at all, so no further reconstruction is needed.

Due to **U2**, theoretically it seems this scheme can be used for electronic atlas. However, in practice it is not the case because this method requires adding an entry for a most frequent color c. This is not allowed since further processing of the atlases, e.g., automatic extraction of the structure boundaries in an atlas [22] would be interfered: in this processing, structures represented by different color indices would be recognized as different. While the palette of an atlas can tolerate some manipulations as suggested in **U3**, we should avoid causing interference to other uses of the atlas and we are interested in the watermarking schemes that are compatible with the existing processing algorithms.

Case 2. *Full palette with 256 distinct colors*: In case the palette is exhausted by 256 distinct colors, two colors c_i (index is i) and c_j (index is j) are chosen, such that $|c_i - c_j|$ is small. Then 0 is associated with i and 1 is associated with j. In embedding, the image is scanned in a defined manner. If a pixel with c_i or c_j is encountered, its color index is changed to i if 0 is to be embedded and changed to j otherwise. In extraction, the image is scanned in the same pattern and the embedded binary stream is obtained depending on the indices i, j of color c_i and c_j. It is easy to see that this scheme is also applicable to the first case, provided that an appropriate pair of similar colors is found.

Clearly, this method is a good tool for watermarking "natural" palette images. However, as suggested in **U1** electronic clinical atlas cannot be assumed "natural", so a color pair with minor difference does not always exist. Furthermore, even such a desired color pair is found, color flipping within a homochromous structure would be readily detectable.

We have shown the reasons why the reversible watermarking schemes for palette images in [17] cannot be used for watermarking electronic clinical atlas. In the next two subsections, we shall propose two reversible watermarking schemes that are tailored for electronic clinical atlas, and the proposed schemes circumvent all these limitations. The first scheme is designed exclusively for the atlases that comprise homochromous structures, and the second scheme can be applied to any atlas, be it "natural" or not.

B. Our First Reversible Watermarking Scheme

Intuitively, the main reason for Fridrich *et al*'s method in case 2 fails in our scenario lies in the fact that the structures contained in an atlas are homochromous. We thus tailor our first scheme exclusively for the atlases with homochromous structures, and this scheme is an extended version of [33].

For the purpose of watermarking, we are faced to find the right embedding channel and the way to insert the authentication payload in an atlas, while without causing perceptible artifacts. We observe that when a boundary point of a homochromous structure revises its color to its neighboring color, the effect caused is visually unnoticeable. On the contrary, if a point within a homochromous structure changes its color to even a similar color, the difference would be quite clear. Motivated by this, we shall exploit the boundary points of the homochromous structures for embedding. Meanwhile, to avoid generating accumulative artifacts, the points to be revised for watermarking should be uniformly diffused over the whole set of boundary points across an atlas. Moreover, we shall follow Fridrich *et al*'s general methodology for achieving reversibility (see Section II). We next elaborate our ideas on these issues, and they are building blocks of the actual scheme.

1) Basic ideas: For ease of understanding, we take Figure 3 as an example, which represents virtually all possible cases of how structures bordering one another along a row. In particular, along row i, (1) structure A stands alone without neighbors; (2) two structures B_1 and B_2 border each other; (3) more than two structures in turn border each other (three structure C_1, C_2 and C_3 are used in the figure, and generalization to the scenario of more than three structures is straightforward).

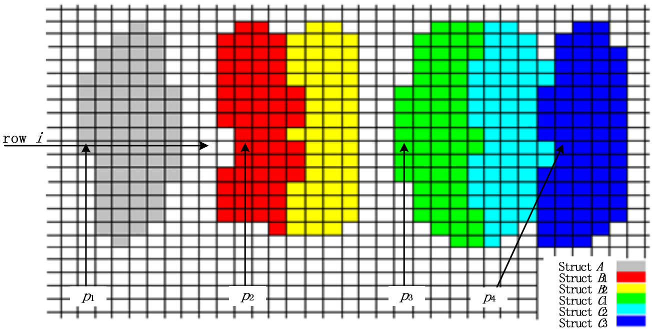


Fig. 3. Illustration of embedding

Embedding channel

Along a row (row i in Figure 3), we in turn pick up p_1 in A, p_2 in B_1, p_3 in C_1 and p_4 in C_3 as *channelling points* for embedding, and the corresponding structures A, B_1, C_1 and C_3 as *embedding structures*. Note that *every other* structures in each group of neighboring structures are chosen as embedding structures. In our example, A, B_1, C_1 and C_3 are chosen as embedding structures, while B_2, C_2 are not embedding structures. As we shall see shortly, this is crucial in preventing interference in encoding between two neighboring structures if otherwise every structure is chosen as embedding structures. The embedding channel for embedding then comprises the channelling points collected by scanning all rows across an entire atlas.

Encoding primitive

For an embedding structure, we associate 0 to the even number of points the structure has along a row, and 1 to the odd number of points. For example, along row i in Figure 3, structure A has 6 points, B_1 has 4 points, C_1 has 5 points and C_3 has 4 points, so the path $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_4$ is encoded as 0010.

Embedding method

The insertion of a bit-stream into the embedding channel is equivalent to encoding it along the channel. Rules on how to embed a bit at a channelling point p are specified in Table I.

Bit to be embedded	Number of points	Action taken
0	Odd	Change color
	Even	No action
1	Odd	No action
	Even	Change color

TABLE I

RULES FOR EMBEDDING.

In Table I, let the channelling point be p and the embedding structure that p lies in be $Struct$, "Number of points" denotes the number of points $Struct$ has along the row that traverses p. By "No action", we do nothing to p; by "Change color", we revise the color of p to its *left* neighboring color. We include the case that the left neighboring color is the background color as long as $Struct$ has no neighboring structure along the row.

To make it clearer, let us see an example of inserting a bit-stream by executing the above rule. Suppose 1001 is to be embedded along the path determined by $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_4$ in Figure 3, the embedding is shown in Table II.

Channelling point	Bit to be embedded	Number of points	Action
p_1	1	6 (Even)	Change color
p_2	0	4 (Even)	No action
p_3	0	5 (Odd)	Change color
p_4	1	4 (Even)	Change color

TABLE II

AN EXAMPLE OF EMBEDDING.

According to Table II, p_1 is changed to be the background color; p_2 keeps unchanged; p_3 is changed to be the background color; p_4 revises its color to be the color of C_2.

The reason for choosing every other structures in a group of neighboring structures is now clear: otherwise, adjoining structures might mutually spoil each other's embedding.

Recovering method

We are faced with recovering the original channelling points by given the original coded stream. The original coded stream can be obtained because we assume Fridrich *et al*'s methodology for reversibility in our scheme. It must be noted that the set of channelling points gathered from an watermarked atlas might be different from the original one due to watermarking. This can be easily seen from the above example of embedding 1001 along row i. In particular, the channelling points of A, C_1, C_3 will change to be the respective right neighboring point of p_1, p_3, p_4, and we denote them as p_1', p_3', p_4', respectively. This is however in no way preventing us from reconstructing the original structures. Let us continue with the above example: for structure A, we know the current channelling point p_1' is actually the right neighboring point of the original point p_1. From the watermarked atlas, we know the current code of p_1' along row i is 1. Once we are given the original code 0, since $0 \neq 1$, we change the color of the *left* neighboring point of p_1' to be the color of p_1' in order to recover the structure A. Likewise, other embedding structures are recovered to their original content following the same rationale. Note that if the current code = the original code, the current channelling point is the original one, e.g., p_2 in our example.

With these ideas, we are ready to give our reversible watermarking scheme.

2) The scheme: Similar to a common authentication watermarking scheme, ours consists of two parts: embedding algorithm and authentication algorithm.

The basic steps of the embedding algorithm are demonstrated in Figure 4.

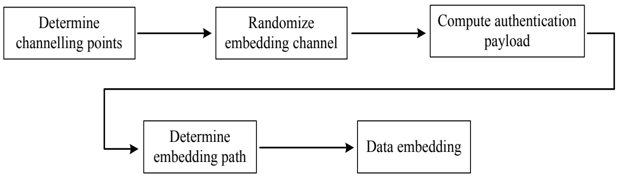

Determine channelling points
Randomize embedding channel
Compute authentication payload
Determine embedding path
Data embedding

Fig. 4. Flow chart for embedding algorithm.

In particular, the embedding algorithm works as follows:

Embedding algorithm

[a] Scan the atlas $\mathcal{M}$ row by row in a fixed pattern, e.g., from top to bottom. Along each row, pick up the channelling points p_i by the method described in embedding channel. An ordered set is eventually formed by these points $\mathcal{S} = \{p_1, p_2, \cdots p_N\}$, where N is the total number of the channelling points that are picked up.

[b] Reorder $\mathcal{S}$ with a secret key k_1 by the following program.

```
$j = N$
from $i := 1$ $to$ $N$
begin
    $idx = H(i, k_1) \bmod j$
    $\overline{\mathcal{S}}[i] = \mathcal{S}[idx]$
    $\mathcal{S} = \mathcal{S} - \{\mathcal{S}[idx]\}$ and shift left each of
    the items after $idx$ in $\mathcal{S}$
    $j = j - 1$
end
$\mathcal{S} = \overline{\mathcal{S}}$
```

i, j, idx and $\overline{\mathcal{S}}$ are temporary variables in the program. Note that the reordering of $\mathcal{S}$ serves to evenly diffuse the positions that embedding is to occur. Otherwise, accumulative effect is possible if we embed row by row. In addition, this randomization reduces the likelihood that an attacker can figure out the exact embedding path, thereby increasing security.

[c] Compute the authentication payload as $h = H(\mathcal{M}, k_2)$, where we let $\mathcal{M}$ denote the pixels (RGB values) of the atlas, $H(.)$ is a cryptographic one-way hash function, e.g., MD5 or SHA1 [5] and k_2 is another secret authentication key. Note that different keys are used to rule out possible correlation.

[d] In a gradual way, encode the path along the ordered points in $\mathcal{S}$ into a bitstream B by the encoding primitive. In the meantime, run an adaptive lossless arithmetic compression algorithm $C(.)$ to compress B as $\overline{B} = C(B)$. Check the difference between B and $\overline{B}$. Once there is enough space to accommodate the authenticating payload, stop the compression algorithm. In particular, let $S_j = \{p_1, p_2, \cdots, p_j\}$ be the path of j steps along $\mathcal{S}$ and B_j correspond to the bit-stream by encoding the path of S_j, the following program achieves the above process:

```
while $(|B_i| - |C(B_i)|) < |h|$
    $i = i + 1$
```

i is a fixed value when the above program halts.

[e] Embed $h||C(B_i)$ into the path of S_i by the embedding method, where $||$ denotes concatenation.

The basic steps of the authentication algorithm are shown in Figure 5.

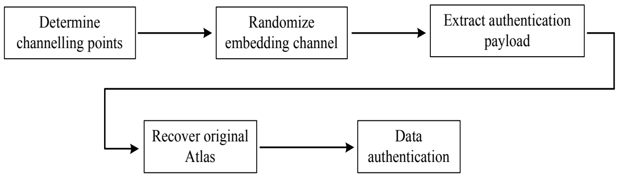

Determine channelling points
Randomize embedding channel
Extract authentication payload
Recover original Atlas
Data authentication

Fig. 5. Flow chart for authentication algorithm.

In particular, the authentication algorithm works as follows:

Authentication algorithm

[a] Scan the watermarked atlas $\mathcal{M}'$ row by row in the same fixed pattern as in the embedding algorithm, and generate an ordered set $\mathcal{S} = \{p'_1, p'_2, \cdots p'_N\}$ of channelling points as in the embedding algorithm.

[b] Reorder $\mathcal{S}$ with the secret key k_1 as in the embedding algorithm.

[c] In a gradual way, encode the path along $\mathcal{S}$ into a bit stream B, and decompress B to be $\overline{B}$ by running the lossless arithmetic decompression algorithm $D(.)$ that corresponds to $C(.)$. Stop the algorithm $D(.)$ once $|\overline{B} - B| \geq |h|$, where $|h|$ is the bit length of $H(.)$. Let S_j, B_j be defined as in the embedding algorithm, the following program achieves the above process:

```
while $(|D(B_i)| - |B_i|) < |h|$
    $i = i + 1$
```

i is a fixed value when the program halts.

[d] Sequentially get $|h|$ bits from the position i afterwards along $\mathcal{S}$ as the authentication payload h'. Next, use the bit-stream $\overline{B}_i = D(B_i)$ to reconstruct the original atlas by the recovering method.

[e] Compare h' with $H(\mathcal{M}, k_2)$, where $\mathcal{M}$ is the pixels of the recovered atlas. If $h' = H(\overline{\mathcal{M}}, k_2)$, then the recovered atlas is authentic, otherwise it has been tampered with.

3) Discussions: In this scheme, we try to recover the original image without first considering whether the watermarked atlas had ever been tampered with. We can actually make some enhancements over this aspect. The basic idea is to insert an extra piece of authentication data of some areas in an atlas that are not affected by watermarking. We can determine such areas before watermarking occurs, because the length of the authentication data suffices help us decide the channelling points that will be affected by watermarking (see step [d] of the embedding algorithm). With such an extra piece of authentication data inserted, in step [d] of the authentication algorithm, we first extract this authentication data together with h', then use this authentication data to determine whether the not-to-be-affected areas has been tampered with. If no tamper is found, we proceed to recover the original image, otherwise, we simply reject. This improvement is of particular importance when data tamper indeed occurred, since in such cases there is no need and in fact, it is not possible to reconstruct the original content.

It is clear that the embedding capacity of this scheme is determined by the total number of channelling points as well as the effectiveness of the lossless arithmetic compression algorithm. Once the total number of channelling points is fixed, more compression effectiveness of the compression algorithm means larger embedding capacity.

We now examine security of this scheme. Specifically, without knowing the secret authentication key k_1 and k_2, an adversary is faced with forging an arbitrary atlas $\mathcal{M}''$ (with respect to a watermarked $\mathcal{M}'$) in an attempt to pass the authentication. It is equivalent to the case that given a fixed hash value h, the value of $H(\mathcal{M}'', k_2)$ happens to be h. Evidently, the probability for this is $1/2^{|h|} = 2^{-128}$ when $|h| = |H(.)| = 128$, according to the *collision free* property of the cryptographic one-way hash function. As a final note, we point out the compression algorithm does not affect security of the watermarking scheme.

C. Our Second Reversible Watermarking Scheme

Our second scheme is applicable to any atlas in palette format, including those with homochromous structures. We develop this scheme by extending the zero-distortion method proposed in [34] to attain increased computational efficiency while without compromising its actual security. This second scheme does embedding by merely manipulating the palette of an atlas, thereby causing no distortion to the image pixels. The main difference between our scheme with Fridrich *et al*'s method in case 1 is that our scheme only involves permutation of the entries in the palette, whereas Fridrich *et al*'s scheme involves adding a new entry in the palette. Consequently, our scheme can be used for any palette image, whereas Fridrich *et al*'s scheme can only be applied to images with palette having fewer than 256 distinct colors.

1) The scheme: The basic idea of our scheme is to arrange the palette according to the authentication payload of the atlas, such that a correlation between the palette and the image content is established. In authentication, the image is authentic if the correlation still holds. In particular, the scheme works as follows.

Embedding algorithm

[a] Suppose there are N colors in the palette of the atlas $\mathcal{M}$ to be watermarked, and each color has the form (R, G, B). Sort these colors into a *virtual* list $Vlist$ according to the values of $R.256^2 + G.256 + B$.

[b] Compute the authentication payload for $\mathcal{M}$ as $h = H(\mathcal{M}$, $k)$, where we let $\mathcal{M}$ denote the pixels (RGB values) of the atlas. Partition h into l segments of equal length as $h = h_0$ $h_1 ... h_{l-1}$, where each h_i satisfies $0 \leq h_i \leq N$. Note that the length of h_i should be determined as $|h_i| = m$ for some m that satisfies $2^m \leq N < 2^{m+1}$. The intuition for this is that we want m (i.e., the length of each h_i) is as large as possible while the value of each h_i is less than N.

[c] Rearrange the palette of $\mathcal{M}$ by the following steps. (1) Move the h_0^{th} color in $Vlist$ to the 0^{th} entry in the palette. (2) Move the h_i^{th} $(1 \leq i < l)$ by the following program.

```
j = 1
from i := 1 to l − 1
begin
    if $h_i \notin \{h_0, ..., h_{i-1}\}$ /* $h_i$ has not ever occurred */
    begin
        move the $h_i^{th}$ color in $Vlist$ to the $j^{th}$ position
        in the palette
        $j = j + 1$
    end
end
```

i, j are temporary variables in the program. (3) Sequentially move the rest colors in $Vlist$ to the remaining entries after the first j entries in the palette (j is a fixed value when step (2) halts).

Note that step (2) is the tricky part of the rearrangement process. The basic idea is that we simply ignore h_i if h_i is equal to any proceding h_n, where $0 \leq n < i$. See the example in Figure 6, supposing h is partitioned into 6 segments. Since h_0 and h_4 refer to the same color in $Vlist$ (h_0 = h_4), so h_4 is ignored and the entry that corresponds to h_4 in the palette is occupied by the color referred to by h_5 (the h_5^{th} color in $Vlist$).

[d] Change the image data according to the new palette. For example, when a color in the i^{th} entry in the original palette is moved to the j^{th} entry in the new palette, all i in the image data need to be modified to be j.

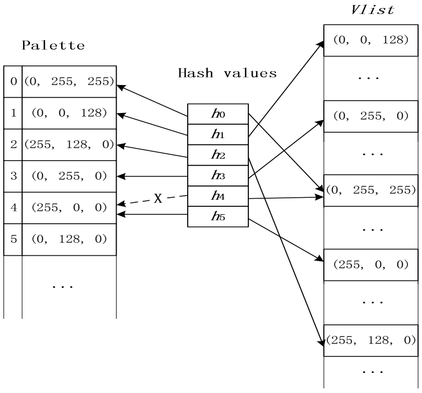


Fig. 6. Rearrangement of the first l colors

Authentication algorithm

[a] Suppose there are N' colors in the palette of the watermarked atlas $\mathcal{M}'$. Sort these colors into a *virtual* list $VList$ as in the embedding algorithm.

[b] Compute the authentication payload for the atlas as $h' = H(\mathcal{M}', k)$, where we let $\mathcal{M}'$ denote the pixels of the atlas. Partition h' into l segments of equal length as $h' = h'_0 h'_1 ... h'_{l-1}$ as in the embedding algorithm

[c] Sequentially authenticate h'_i, $0 \leq i \leq l-1$, by the following program. Suppose PC_i is the color in the i^{th} entry in the palette and C_i is the color in $Vlist$ referred to by h'_i:

```
if $PC_0 \neq C_0$
return INVALID
$j = 1$
from $i := 1$ to $l - 1$
begin
    if $h'_i \notin \{h'_0, ..., h'_{i-1}\}$ /* $h'_i$ has not ever occurred */
    begin
        if $PC_j \neq C_i$
        return INVALID
        $j = j + 1$
    end
end
return AUTHENTIC
```

i, j are temporary variables. Note that in this program, $\mathcal{M}'$ is deemed authentic only when all $h'_i, i \in [0, l)$ pass the testing. In this authentication procedure, h'_i will be ignored if it is equal to any preceding h'_n, where $n \in [0, i)$

2) Discussions: Our extension to the method in [34] (Wu's method for short) lies mainly in the following: the l segments of the authentication payload h in Wu's method must be distinct to one another. As a result, Wu's method may need to repeatedly feed a distinct random seed to the one-way hash function in computing the authentication payload until a *suitable* hash value is found. By contrast, the l segments of h in our scheme are not necessarily distinct, so we achieve higher computational efficiency. In addition, we do not need to remember the seed during data authentication.

In the sequel, we assume $2^m \leq N < 2^{m+1}$, where N is the number of colors in the palette of an atlas. Clearly, $m \leq 8$. Applicability of this scheme to electronic clinical atlas is discussed as follows. As each segment h_i is used as a color index in our scheme, to guarantees that $h_i \leq N$ and h_i is as large as possible, $|h_i|$ should be equal to m. As a result, the total number of segments that the hash value h is partitioned is $l = |h|/m$. Moreover, clearly l should be less than or equal to N, that is, $l \leq N \Rightarrow |h|/m \leq N \Rightarrow |h|/m \leq 2^m$. Assume $|h| = 128$, then $m \geq 5$. This suggests that for an atlas to be watermarkable, it has to have at least 32 colors in the palette.

Next, we shall examine the actual security of our scheme scheme. In particular, we aim to determine the upper bound for the probability of successful forgeries by an adversary without knowing the secret authentication key k. That is, we want to answer the following question: *given an watermarked atlas $\mathcal{M}'$, in what probability can an arbitrarily forged $\mathcal{M}''$ ($\mathcal{M}'' \neq \mathcal{M}'$) by the adversary pass the authentication process?* As we just discussed, it should hold that $5 \leq m \leq 8$ and $2^m \leq N < 2^{m+1}$, where N is the number of colors in the palette of an atlas. Therefore, an 128-bit hash value would be accordingly partitioned into l segments, where $16 \leq l \leq 26$.

We start with a simple example by assuming $l = 4$. Consequently, $h = H(\mathcal{M}', k)$ would be partitioned into $h = h_0h_1h_2h_3$, where $\mathcal{M}'$ denote the pixels of a watermarked atlas $\mathcal{M}'$ and k is the secret authentication key. Without loss of generality, suppose $h_0 = h_2$, so upon watermarking the first 4 entries (0^{th}, 1^{th}, 2^{nd}, 3^{rd}) in the palette of $\mathcal{M}'$ would be the colors referred to by $h_0, h_1, h_3, \overline{h}_3$, respectively, where $\overline{h}_3$ refers to the random color that occupies the 3^{rd} entry. This can be shown in Figure 7a. According to the authentication algorithm, a forged $\mathcal{M}''$ would be deemed authentic as long as $h' = H(M'', k)$ satisfies any of the 4 cases shown in Figure 7b: (1) each of segment h'_1, h'_2 and h'_3 of h' is the same as h_0; (2) each of segment h'_2 and h'_3 is the same as either h_0 or h_1; (3) segment h'_3 is the same as either of h_0, h_1 and h_3; (4) $h' = h_0h_1h_3\overline{h}_3$. In total, there would be 9 possible values for h' that could pass the authentication. Specifically, (1) $h' = h_0h_0h_0h_0$; (2) $h' = h_0h_1h_0h_0$, $h' = h_0h_1h_0h_1$, $h' = h_0h_1h_1h_0$, $h' = h_0h_1h_1h_1$; (3) $h' = h_0h_1h_3h_0$, $h' = h_0h_1h_3h_1$, $h' = h_0h_1h_3h_3$; (4) $h' = h_0h_1h_3\overline{h}_3$.

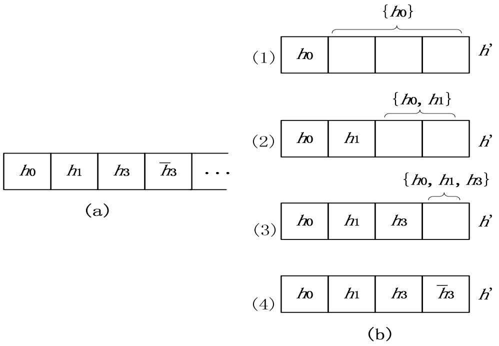


Fig. 7. All possible cases of forgery

We proceed to generalize the above simple example to the general case of any value of l. That is, given the first l entries of the palette that correspond to $h = h_0h_1 \cdots h_{l-1}$, we discuss what values h' can take so as to pass the authentication. Generalizing the cases in Figure 7b, we know that h' taking value as $h' = h_0h_1 \cdots h_{i-1}h'_i \cdots h'_{l-1}$ for a particular i could succeed in the authentication, where each $h'_j \in \{h_0, h_1, \cdots, h_{i-1}\}$. The number of possible values of h' in such a case can be computed as i^{l-i}. Considering all cases of i ($1 \leq i \leq l-1$), we then get the total number of values that h' can take as $Num = \sum_{i=1}^{l-1} i^{(l-i)} + 1 = \sum_{i=1}^{l} i^{(l-i)}$. The probability Pr of an adversary forging an arbitrary atlas $\mathcal{M}''$ that could be deemed authentic is thus $Pr = \frac{Num}{2^{|h|}}$.

Apparently, Pr increases with Num as $|h|$ fixes. We are now going to determine the upper bound of Pr in the case of watermarking electronic clinical atlas. Clearly when $l = 26$ ($16 \leq l \leq 26$), Num takes the biggest value as $Num = \sum_{i=1}^{26} i^{(26-i)} < 2^{56}$. Hence $Pr = \frac{Num}{2^{|h|}} < 2^{56}/2^{128} = 2^{-72}$. We then have the following claim to answer the question raised earlier: *in the absence of the secret authentication key, the probability that an arbitrary bogus atlas by an adversary makes the authentication algorithm return AUTHENTIC cannot exceed* 2^{-72}.

IV. Experimental Results

We implemented and conducted extensive experiments on our proposed two schemes. Experiments were done on a PC with 2G CPU and 512M RAM. Source codes were written in Microsoft C++. The hash function $H(.)$ in our schemes was instantiated by MD5 [5] which offers 128-bit hash values.

A. Experiments with the First Scheme

In the experiments with the first scheme, we used truncated hash values of 64 bits. From the earlier security discussions, we know this achieves a confidence factor of 2^{-64}, which is a sufficiently small probability for forgery. For lossless compression, we used the LZW compression algorithm [35], which works well when the streams to be compressed have repeated patterns. In addition, in the experiments we authenticate the set of boundary points of structures contained in an atlas. The reasons are (1) in our current applications [22], we utilized the boundary points rather than the entire content of an atlas itself; (2) the set of boundary points is a compact representation of the corresponding atlas, and the integrity of the boundary points of an atlas is equivalent to the integrity of the atlas, since every change of the atlas pixels will be reflected in the set of boundary points gathered by the algorithms in [22]; (3) this is expected to accelerate the computation of authentication payload, because less data are feed to the hash function, although this acceleration is theoretically minor. We leverage on the algorithms proposed in [22] to extract the boundary points of atlases. Figure 8 shows an atlas and the set of boundary points within it.

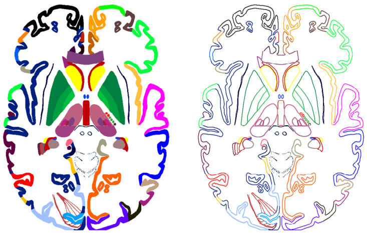


Fig. 8. An atlas (left) and the boundaries of all structures (right)

From the experiments, it is estimated that on average, 3000-4000 channelling points could be found in an atlas, and most of the atlases are watermarkable by our first scheme. Figure 9 shows an example of our experimental results: 9a is the original atlas, 9b shows the positions where embedding occurs and 9c shows the corresponding watermarked atlas. It can be seen from 9b that the revised points had been well diffused, which is due to the output randomness of the hash function used in the reordering of the channelling points. Since embedding occurs upon the boundary points of structures, so no apparent distinction is seen between 9a and 9c.

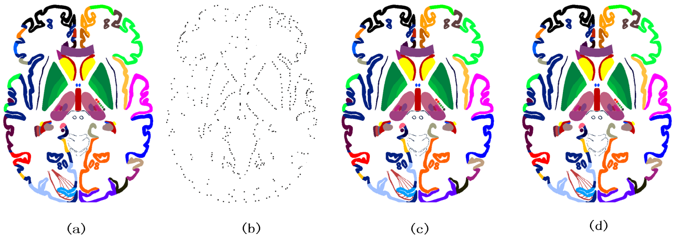


Fig. 9. Experiment results. (a) The original atlas. (b) Positions where watermarking occurs in the first scheme. (c) The watermarked atlas by the first scheme. (d) The watermarked atlas by the second scheme.

B. Experiments with the Second Scheme

Experiments with the second scheme focused on the program implementation and testing applicability of the scheme. Other factors such as image quality degradation is not a concern, since the scheme only involves permutation of the palette and the visual appearance of the atlases is not affected at all. Figure **??**d shows a watermarked atlas by our second scheme that corresponds to **??**a. As we have discussed earlier, an atlas is watermarkable if it has 32 or more colors in the palette. In fact, all the atlases we experimented have more colors than 32.

V. Conclusion

In this paper, we developed two tailored reversible watermarking schemes for authentication of electronic clinical atlas. The first scheme was designed exclusively for atlases with homochromous structures, and followed Fridrich *et al*'s general methodology for achieving reversibility; the second scheme can be applied to any atlas in palette format, and incurres zero-distortion to the watermarked atlas by simply manipulating the palette. A weakness of the second scheme is that while it attains the authentication functionality by detecting every tampering to the image content, it can generate false positives: an adversary can permute the entries in the palette, and modify the image data accordingly while without changing the image content. In such cases, the authentication algorithm will report tamperings. This weakness is actually the price must be paid by zero-distortion. We implemented the proposed schemes and obtained promising experimental results.

Acknowledgment

The fourth author is grateful to Prof. W. L. Nowinski, who supervised the fourth author's master thesis on modelling tumor growth with Electronic Clinical Atlas. The authors also would like to thank Dr. Yudong Wu for extensive discussions on the security of the proposed schemes.

References

[1] G. Coatrieux, H. Maitre, B. Sankur, Y. Rolland, R. Collorec, "Relevance of Watermarking in Medical Imaging", in *Proc. IEEE EMBS Conf. On Information Technology Applications in Biomedicine*, 2000, pp. 250-255.

[2] R. J. Anderson, "Security in Clinical Information Systems", British Medical Association, 1996

[3] H. Lehrer, "Potential Legal Problems with Digitalized Images", *Radiology*, pp. 190-902, 1994

[4] R. M. Frieddenberg, "Potential Problems Associated with PACS", *Radiology*, Vol. 189, pp. 55-57, 1993

[5] B. Schneier, *Applied Cryptography*, 2^{nd} Edi, John Wiley & Sons, 1996.

[6] I. Cox, J. Boom, and M. Miller, *Digital Watermarking*, Morgan Kaufmann, 2001.

[7] N. F. Johnson, Z. Duric, and S. Jajodia, *Information Hidding: Steganography and Watermarking - Attacks and Countermeasures*, Kluwer Academic Publishers, 2000.

[8] H. M. Chao, C. M. Hsu, S. G. Miaou, "A Data-Hidding Technique With Authentication, Integration, and Confidentiality for Electronic Patient Records", *IEEE Trans. on Information Technology in Biomedicine*, Vol. 6, No. 1, pp. 46-53, 2002.

[9] D. Anand, U. Niranjan, "Watermarking Medical Images with Patient Information", in *Proc. of IEEE/EMBS Conference*, 1998, pp. 703-7-6.

[10] S. Miaou, C. Hsu, Y. Tsai, H. Chao, "A Secure Data Hidding Technique with Heterogeneous Data-Combining Capacity for Electronic Patient Records", in *Proc. World Congree on Medical Physicas and Biomedical Engineering, Electronic Healthcare Records, IEEE-EMB*, 2000.

[11] B. B. Zhu, M. D. Swanson, A. H. Tewfik, "When Seeing Isn't Believing", *IEEE Signal Processing Magzine*, Vol. 21, No. 2, pp. 40-49, 2004.

[12] J. P. Smith, "Authentication of Digital Medical Images with Digital Signature Technology", *Radiology*, Vol.194, No.3, pp. 771-774, 1995.

[13] B. Macq, "Lossless Multiresolution Transform for Image Authenticating Watermarking", in *Proc. of EUSIPCO*, 2000.

[14] C. W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data", US Patent application, Docket No: 77102/E-D, 1999.

[15] J. Fridrich, M. Goljan and R. Du, "Invertible Authentication", in *Proc. SPIE Photonics West, vol. 3971, Security and Watermarking of Multimedia Contents III*, 2001, pp. 197-208.

[16] J. Fridrich, M. Goljan and R. Du, "Lossless Data Embedding - New Paradigm in Digital Watermarking", *Special Issue on Emerging Applications of Multimedia Data Hiding*, Vol. 2002, No.2, PDF Journal Editorial, pp. 185-196, 2002.

[17] J. Fridrich, M. Goljan and R. Du, "Lossless Data Embedding for All Image Formats", in *Proc. SPIE Photonics West, Vol. 4675, Electronic Imaging 2002, Security and Watermarking of Multimedia Contents*, pp. 572-583.

[18] J. Tian, "High Capacity Resersible Data Embedding and Content Authentication", in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2003.

[19] W. L. Nowinski, T. T. Yeo, A. Thirunavuukarasuu, "Microelectrode-Guided Functional Neurosurgery Assisted by Electronic Clinical Brain Atlas CD-ROM", *Computer Aided Surgery*, pp.115-122, 1998.

[20] W. L. Nowinski, A. Thirunavuukarasuu, "Electronic Atlases Show Value in Brain Studies", *Diagnostic Imaging Asia Pacific*, Vol. 8(2), pp.35-39, 2001.

[21] Available: http://www.cerefy.com/

[22] Y. J. Yang, "Fast Deformation of A Human Brain Atlas Against Tumor", Master Thesis submitted to National University of Singapore, 2001.

[23] M. Celik, G., harma, E. Saber, "A Hierarchical Image Authentication Watermark With Improved Localization And Security", in *Proc. ICIP 2001(CD-ROM version)*, paper ID 3532, 2001.

[24] S. Carver, N. D. Memon, B. L. Yeo, M. M. Yeung, "On the Invertibility of Invisible Watermarking Techniques", in *Proc. ICIP 97*, 1997, pp. 540-543.

[25] I. J. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia", Technique Report 95-10, NEC Research Institute, 1995.

[26] C. D. Vleeschouwer, J. F. Delaigle, B. Macq, "Circular Interpretation of Bijective Transformations in Lossless Watermarking for Media Asset Management", *IEEE Trans. Multimedia*, Vol. 5, No. 1, pp. 97-105, 2003.

[27] T. Kalker, F. M. Willems, "Capacity Bounds and Code Constructions for Reversible Data-Hiding", in *Proc. Security and Watermarking of Multimedia Contents, SPIE, Vol. 5020*, 2003, pp. 604-611.

[28] M. Celik, G. Sharma, A. M. Tekalp, E. Saber, "Localized Lossless Authencation Watermark", in *Proc. Security and Watermarking of Multimedia Contents, SPIE, Vol. 5020*, 2003, pp. 689-698.

[29] H. A. Wang, Y. Z. Wang, and S. Wang, "Digital Signature Technology for Health Care Applications", *Southern Medical Journal*, pp. 281-286, 2001.

[30] G. Coatrieux, H. Maitre, B. Sankur, "Strict Integrity Control of Biomedical Images", in *Proc. Security and Watermarking of Multimedia Contents III, SPIE Vol. 4314*, 2001, pp. 229-240.

[31] A. Wakatani, "Digital Watermarking for ROI Medical Images by Using Compressed Signature Image", in *Proc. of Annual Hawaii Internation Conference on System Sciences*, 2002.

[32] B. Macq, F. Dewey, “Trusted Headers for Medical Images”, in *Proc. DFG VIII-DII Watermarking Workshop*, 1999.

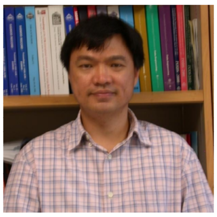
[33] Y. J. Yang, F. Bao, “An Invertible Watermarking Scheme for Authentication of Electronic Clinical Atlas”, in *Proc. IEEE International Conference on Acoustic, Speech, and Signal Processing*, 2003, pp.533-536.

[34] Y. D. Wu, “Zero-Distortion Authentication Watermarking”, in *Proc. International Conference on Information Security*, LNCS 2851, 2003, pp. 325-337.

[35] K. Sayood, *Introduction to Data Compression*, Morgan Kaufmann Publishers, 1996.

Feng Bao Feng Bao received his BS in mathematics, MS in computer science from Peking University and his PhD in computer science from Gunma University in 1984, 1986 and 1996 respectively. He was an assistant/associate professor of the Institute of Software, Chinese Academy of Sciences from 1987 to 1993 and a visiting scholar of Hamberg University, Germany from 1990 to 1991. Since 1996 he has been with the Institute for Infocomm Research, Singapore. Currently he is a Lead Scientist and the head of Infocomm Security Department and Cryptography Lab of the institute. His research areas include algorithm, automata theory, complexity, cryptography, distributed computing, fault tolerance and information security. He has over 100 publications and 16 patents.

Robert H. Deng Robert H. Deng received his B.Eng from National University of Defense Technology, China, in 1978, and his M.Sc and PhD from Illinois Institute of Technology, USA, in 1983 and 1985, respectively. He is Professor, school of Information Systems, Singapore Management University. He has more than 140 publications in the areas of error-control coding, digital communications, computer networks and information security. He served on many advisory and program committees of international conferences. He served as Program Chair of the 2002 International Conference on Information and Communications Security and the General Chair of the 2004 International Workshop on Practice and Theory in Public Key Cryptography.



Beng Chin Ooi Beng Chin is Professor of Computer Science and Vice Dean (Academic Affairs and Graduate Studies) at School of Computing, National University of Singapore, and a fellow of Singapore-MIT Alliance Programme.

Beng Chin obtained his BSc (1st Class Honors) and PhD from Monash University, Australia, in 1985 and 1989 respectively. His research interests include database performance issues, indexing techniques, XML, P2P/grid/parallel systems and advanced applications. He has served as a PC member for international conferences including SIGMOD, VLDB, ICDE, EDBT, DASFAA, CIKM and Vice PC Chair for ICDE’00,04,06 and PC Chair for SSD’93 and DASFAA’05. He is an editor of GeoInformatica, Journal of GIS, VLDB Journal and IEEE Transactions on Knowledge and Data Engineering. He is a co-founder and director of Thothe Technologies (formerly known as GeoFoto), a company providing imaging and digital asset management solutions.

Yanjiang Yang Mr. Yanjiang Yang received his BS and MS in computer science and engineering from Nanjing University of Aeronautics and Astronautics in 1995 and 1998, MS in Biomedical Imaging from National University of Singapore in 2001, respectively. He is now a PhD candidate in School of Computing, National University of Singapore, attached to Institute for Infocomm Research, A*STAR, Singapore. His research areas include Information Security, Biomedical Imaging.