Reversibility, Deceptions, and Counteractions in Adaptive Digital Watermarking*

ZHAO Xian-feng, WANG Wei-nong, CHEN Ke-fei

(*Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai* 200030, *China*)

E-mail: zhao-xf@cs.sjtu.edu.cn

http://www.cs.sjtu.edu.cn

Received October 17, 2001; accepted January 8, 2002

Abstract: To further enhance the security of the present digital watermarking, the reversibility of widely researched adaptive watermarking is investigated. First, watermarking schemes are classified and generalized. Then, on the assumption that adaptive watermarking places no constraint on the formation of watermarks and scaling factors, the reversibility and quasi-reversibility, together with their resulting reverse and quasi-reverse engineering attacks, which could disturb or even overturn the ownership verification, are defined, analyzed and illustrated. Finally, the necessity of placing constraints on the formation of watermarks and scaling factors is concluded, and the essential irreversibility of some adaptive technologies, which can be used to enhance the security, is pointed out. Making watermarks and scaling factors one-way dependent on original data, and exploiting the human perceptual system, help watermarking become resistant to the above attacks and more reliable in ownership verification.

Key words: reversibility; digital watermarking; digital copyright protection; information hiding; steganalysis

While images, videos, audios etc are evolving into their digital forms, the ease of duplicating perfect products results in the spread of unauthorized copies. Research on digital copyright management shows that digital watermarking is a feasible copyright control technique. Balancing between perceptual transparency and robustness, the technology embeds copyright information into original digital works without perceptually degrading the quality of the released version, and tries to preserve the information in case of intentional or unintentional attacks[1,2]. Attacks on watermarking have to maintain the perceptual quality of attacked copies, though they can just aim at damaging watermarks rather than replacing or deciphering them. Therefore, typical attacks are moderate active attacks, which mainly include image processing, lossy compression, geometric transformation, additive noise, optic copy etc[3~5].

Research on watermarking often concentrates on the robustness of additive signal based watermarking[1,2]. Nevertheless, if it meets our requirements so that active attacks might hardly succeed, could watermarking be reliable enough to verify ownership of various multimedia? In cryptography, Kerckhoff's desiderata require that the security of algorithms should be built on keys, and their publication should do no harm to them[6]. Unfortunately, some researche has shown that watermarking is not secure or convincing enough in this sense. On the assumption that scaling factors for adjusting the embedding intensity are constant and watermarking affects the attackers' embedding domain slightly, the

* Supported by the National Natural Science Foundation of China under Grant No.60073034 (国家自然科学基金)

ZHAO Xian-feng was born in 1969. He is a Ph.D. candidate at the Department of Computer Science and Engineering, Shanghai Jiaotong University. His research interests are information security and multimedia. **WANG Wei-nong** was born in 1949. He is a professor and doctoral supervisor at the Department of Computer Science and Engineering, Shanghai Jiaotong University. His current research areas are information security and network security. **CHEN Ke-fei** was born in 1959. He is a professor and doctoral supervisor at the Department of Computer Science and Engineering, Shanghai Jiaotong University. His current research areas are information security, network security, and e-commerce.

most significant conclusion drawn by the research indicates that watermarking should be irreversible, and that the security should be built on original data besides the key-stream or pseudorandom noise (PN)[7,8].

With the development of adaptive watermarking[9~11], however, we think that the above rudimentary irreversible watermarking also does not comply with Kerckhoff's desiderata strictly because the 2 underlying assumptions are untenable now. First, scaling factors are variable in adaptive watermarking. Second, having exploited the human perceptual system (HPS), anyone can affect the embedding domains more heavily. So the ownership verification based on it is worth further consideration. With the questions of how the new changes influence the security of watermarking and who is the beneficiary, this paper investigates the security of adaptive watermarking without considering the above 2 assumptions. In Section 1, it generalizes watermarking schemes and their typical methods. In Sections 2 and 3, reversibility, quasi-reversibility, together with their resulting attacks and the counteractions, are investigated for private watermarking schemes and public ones respectively. We draw the conclusions in Section 4.

1 Generalized Watermarking Schemes and Their Typical Methods

Like cryptosystems, watermarking schemes define the frameworks of steps and methods in watermarking applications without considering the specific features of algorithms used. In spite of the similarity among embeddings, watermarking schemes are usually divided into private and public watermarking schemes according to whether or not to use the original data in extraction[1,2,4,5,9~11].

Algorithm 1. Generalized embedding: Let $\boldsymbol{h}=(h_1,h_2,\ldots,h_n)$ be original data or one of its transform domains. Let $\boldsymbol{w}=(w_1,w_2,\ldots,w_m)$ be a coding unit of the copyright information, and $c=g(\boldsymbol{w})=(c_1,c_2,\ldots,c_n)$ be the code word coded by a channel coding algorithmg $g(\cdot)$, which may be repetition, linear block, or spread spectrum coding etc. Let $\boldsymbol{k}=(k_1,k_2,\ldots,k_n)$ be a key-stream or PN, and $\boldsymbol{s}=r(\boldsymbol{c},\boldsymbol{k})=(s_1,s_2,\ldots,s_n)$ be the watermark code randomized by a stream cipher algorithm $r(\cdot,\cdot)$. Let $\boldsymbol{a}=v(\boldsymbol{h})=(a_1,a_2,\ldots,a_n)$ be the scaling factor generated by a perceptually adaptive algorithm $v(\cdot)$. Then, the embedding can be expressed as

$$\boldsymbol{h}'=e(\boldsymbol{h},v(\boldsymbol{h})\cdot r(g(\boldsymbol{w}),\boldsymbol{k}))=e(\boldsymbol{h},\boldsymbol{a}\cdot r(\boldsymbol{c},\boldsymbol{k}))=e(\boldsymbol{h},\boldsymbol{a}\cdot\boldsymbol{s}) \tag{1}$$

where $e(\cdot,\cdot)$ denotes the embedding algorithm, $\boldsymbol{h}'$ denotes the released version of $\boldsymbol{h}$, and $\cdot$ represents the operation of direct product of 2 vectors, which could be defined as $\boldsymbol{a}\cdot\boldsymbol{b}=(a_1,a_2,\ldots,a_n)\cdot(b_1,b_0,\ldots,b_n)=(a_1b_1,a_2b_2,\ldots,a_nb_n)$.

Algorithm 2. Generalized extraction in private schemes. Let $\boldsymbol{h}''$ be the possibly attacked version of $\boldsymbol{h}'$. At the extraction end, where the original data $\boldsymbol{h}$ is available, the hidden watermark $\boldsymbol{w}'$ can be extracted by means of subtracting $\boldsymbol{h}$ from $\boldsymbol{h}''$ followed by some decoding and recognition operations, which can be expressed as

$$\boldsymbol{w}'=g^{-1}(r^{-1}(v^{-1}(\boldsymbol{h})\cdot e^{-1}(\boldsymbol{h}'',\boldsymbol{h}),\boldsymbol{k}))=g^{-1}(r^{-1}(\boldsymbol{a}^{-1}\cdot e^{-1}(\boldsymbol{h}'',\boldsymbol{h}),\boldsymbol{k})) \tag{2}$$

$$t=s(\boldsymbol{w}',\boldsymbol{w}),\qquad c_T(t)=\begin{cases}0\ (\text{nonexistence}), & t<T,\\ 1\ (\text{existence}), & t\geq T\end{cases} \tag{3}$$

where, with t for the similarity between $\boldsymbol{w}'$ and $\boldsymbol{w}$, and T for the recognition threshold, $c_T(t)$ draws the conclusion about the existence of $\boldsymbol{w}$. t is usually the normalized or non-normalized correlation between 2 vectors represented by

$$t=[\boldsymbol{w}',\boldsymbol{w}]\Big/\sqrt{[\boldsymbol{w}',\boldsymbol{w}']\cdot[\boldsymbol{w},\boldsymbol{w}]}=[\boldsymbol{w}',\boldsymbol{w}]/(|\boldsymbol{w}'|\cdot|\boldsymbol{w}|), \tag{4}$$

$$t=[\boldsymbol{w}',\boldsymbol{w}]\Big/\sqrt{[\boldsymbol{w},\boldsymbol{w}]}=[\boldsymbol{w}',\boldsymbol{w}]/|\boldsymbol{w}|, \tag{5}$$

respectively, where $[\cdot,\cdot]$ denotes the inner product of 2 vectors, and $|\cdot|$ denotes the length of a vector.

Algorithm 3. Generalized extraction in public schemes: Suppose the addition of a watermark changes a statistical characteristic of released data, which can be tested by a test statistic t. At the extraction end, where original data is unavailable, t is computed and compared to a threshold T, and the conclusion is drawn by

$$t=e^{-1}(\boldsymbol{h}'',\boldsymbol{a},\boldsymbol{k}),\qquad c_T(t)=\begin{cases}0\ (\text{nonexistence}) & t<T\\ 1\ (\text{existence}) & t\geq T\end{cases}. \tag{6}$$

Here, the scaling factor $\boldsymbol{a}$ must be either available or computable in the extraction.

Most additive watermarking schemes can be further simplified for our research purposes without the loss of their generality[7,8]. In most cases, we do not care channel coding and stream cipher, which are beyond the scope of our research, so we often disregard $g(\boldsymbol{w})$ and $r(\boldsymbol{c},\boldsymbol{k})$, and think that $\boldsymbol{s}=\boldsymbol{c}=\boldsymbol{w}$. The following 2 samples, whose simple adaptive technique conforms to Weber's law[1], convey our thought well.

Example 1. A simple adaptive private watermarking scheme in DCT coefficient domain[5]:

(1) Embedding: Compute the DCT coefficients of an image. Sort the middle segment of them in zig-zag order and get $\boldsymbol{h}$. Let $\boldsymbol{a}=v(\boldsymbol{h})=a\boldsymbol{h}=(ah_1,ah_2,\ldots,ah_n)$, where α is a constant about 0.1. Then embed the watermark $\boldsymbol{w}$ by

$$\boldsymbol{h}'=e(\boldsymbol{h},\boldsymbol{a}\cdot\boldsymbol{w})=(h_1+\alpha h_1w_1,h_2+\alpha h_2w_2,...,h_n+\alpha h_nw_n) \tag{7}$$

(2) Extraction: Extract the watermark $\boldsymbol{w}'$ from the possibly attacked version $\boldsymbol{h}''$ by

$$\boldsymbol{w}'=v^{-1}(\boldsymbol{h})\cdot e^{-1}(\boldsymbol{h}'',\boldsymbol{h}) \tag{8}$$

(3) Verification: Draw the conclusion according to Eqs.(3) and (5).

Example 2. A simple adaptive public watermarking scheme in spatial domain[9]:

(1) Embedding: Suppose that $\boldsymbol{h}$ is the luminance component of an image. Let $\boldsymbol{a}=v(\boldsymbol{h})=a\boldsymbol{h}=(ah_1,ah_2,\ldots,ah_n)$, where α is 1/20, and let $\boldsymbol{w}$ be a PN composed of 1s and 0s. Then embed the watermark $\boldsymbol{w}$ by $\boldsymbol{h}''=e(\boldsymbol{h},\boldsymbol{a}\cdot\boldsymbol{w})$.

(2) Extraction: In fact, $\boldsymbol{w}$ divides $\boldsymbol{h}$ or $\boldsymbol{h}'$ into 2 subsets, which can be represented by $X=\{h_i|w_i=1\}$ and $Y=\{h_i|w_i=0\}$, where $1\le i\le n$. Therefore, the test statistic t can be computed by

$$t=e^{-1}(\boldsymbol{h}'',\boldsymbol{a})=\frac{\mu_X-\mu_Y}{\sqrt{(\sigma_X^2+\sigma_Y^2)/n}}=\frac{\mu_X-\mu_Y}{\sigma_{XY}}\sim\begin{cases}N(\alpha\,\mu_X/\sigma_{XY},1) & \text{existance}\\ N(0,1) & \text{nonexistance}\end{cases}, \tag{9}$$

where μ_X, μ_Y, σ_X and σ_Y denote the mean values and the standard variations of pixels in X and Y respectively.

(3) Verification: Draw the conclusion according to

$$c_T(t)=\begin{cases}0\,(\text{nonexistence}) & t<T\\ 1\,(\text{existence}) & t\ge T\end{cases},\quad T=\alpha\cdot\frac{\mu_X}{2\sigma_{XY}}. \tag{10}$$

To more strictly and feasibly discuss typical additive watermarking, we introduce the following definition[12].

Definition 1. (Direct product vector space) Suppose $(V_n,\oplus)$ is an additive group of n-dimension, and $(A_n,\hat{+},\times)$ is a field of n-dimension, whose multiplicative identity is $\boldsymbol{I}$. For $\forall\boldsymbol{x},\boldsymbol{y}\in V_n$, $\forall\boldsymbol{u},\boldsymbol{v}\in A_n$, if

(1) addition $\oplus$ is define by $x\oplus y=(x_1+y_1,x_2+y_2,\ldots,x_n+y_n)\in V_n$,

(2) numeric multiplication $\otimes$ is define by $\boldsymbol{u}\otimes\boldsymbol{x}=(u_1x_1,u_2x_2,\ldots,u_nx_n)\in V_n$, and

(3) $\boldsymbol{u}\otimes\boldsymbol{x}=\boldsymbol{x}\otimes\boldsymbol{u},\boldsymbol{u}\otimes(\boldsymbol{x}\oplus\boldsymbol{y})=(\boldsymbol{u}\otimes\boldsymbol{x})\oplus(\boldsymbol{u}\otimes\boldsymbol{y}),(\boldsymbol{x}\oplus\boldsymbol{y})\otimes\boldsymbol{u}=(\boldsymbol{x}\otimes\boldsymbol{u})\oplus(\boldsymbol{y}\otimes\boldsymbol{u}),\boldsymbol{u}\otimes(\boldsymbol{v}\otimes\boldsymbol{x})=(\boldsymbol{u}\times\boldsymbol{v})\otimes\boldsymbol{x},\boldsymbol{I}\otimes\boldsymbol{x}=\boldsymbol{x}$,

then V_n is called a direct product vector space over A_n, which is denoted by $V_n|A_n$.

In a $V_n|A_n$, where $\boldsymbol{a}\in A_n$ and $\boldsymbol{h},\boldsymbol{h}',\boldsymbol{k}\in V_n$, one can discuss additive watermarking with simple vector operations. For example, suppose we disregard $g(\boldsymbol{w})$, or both $g(\boldsymbol{w})$ and $r(\boldsymbol{c},\boldsymbol{k})$ in embedding, we have $\boldsymbol{h}'=\boldsymbol{h}\oplus(\boldsymbol{a}\otimes(\boldsymbol{w}\oplus\boldsymbol{k}))$ or $\boldsymbol{h}'=\boldsymbol{h}\oplus(\boldsymbol{a}\otimes\boldsymbol{w})$. We also have $\boldsymbol{w}=(\boldsymbol{a}^{-1}\otimes(\boldsymbol{h}''\oplus(-\boldsymbol{h})))\oplus(-\boldsymbol{k})=(\boldsymbol{a}^{-1}\otimes(\boldsymbol{h}''-\boldsymbol{h}))-\boldsymbol{k}$ or $\boldsymbol{w}=\boldsymbol{a}^{-1}\otimes(\boldsymbol{h}''\oplus(-\boldsymbol{h}))=\boldsymbol{a}^{-1}\otimes(\boldsymbol{h}'-\boldsymbol{h})$ correspondingly for extraction in private schemes, with $-$ for either the unary operation of getting the additive negative or the subtraction based on it, and the superscript -1 over vectors for the unary operation of getting the multiplicative inverse. Extraction in public schemes is more difficult to be given in these operations, but it usually can be expressed as a set of statistical functions defined in $V_n|A_n$.

With the above general watermarking schemes, which we think are widely supported [1,2,4,5,9~11], we are going to investigate the security of adaptive watermarking. In the following, we call the real owner of digital works Alice, and the deceiver Bob. Their measurements are marked by subscripts of A and B for Alice and Bob respectively.

2 Reversibility, Deceptions and Counteractions in Private Watermarking Schemes

Before our further discussion, 2 facts in the state-of-the-art watermarking are worth noting[1,2]. First, most schemes

assume registered algorithms are used in embedding, extraction and verification, but they often impose no restriction on the formation of watermarks and scaling factors. Second, to facilitate the applications, no one wants an authority center to process every case of digital ownership online, such as a timestamp server[6].

Table 1 Possible verification conclusions in private watermarking schemes
($\boldsymbol{h}^x$ represents $\boldsymbol{h}_A'$ or $\boldsymbol{h}_B'$, whose ownership is in dispute, and $\in$ represents 'exists in')

No.	$\boldsymbol{w}_A \in \boldsymbol{h}^x$	$\boldsymbol{w}_B \in \boldsymbol{h}^x$	$\boldsymbol{w}_A \in \boldsymbol{h}_B$	$\boldsymbol{w}_B \in \boldsymbol{h}_A$	conclusion
1	0	1	0/1	0/1	Bob
2	1	1	0	1	Bob or Alice
3	1	1	0	0	Alice and Bob
4	1	1	1	1	Alice and Bob
5	1	1	1	0	Alice or Bob
6	1	0	0/1	0/1	Alice

We use Eq.(7), Eq.(8) and Eq.(5) as the basic watermarking steps, and use a triple (e, e^{-1}, c_T) to represent a watermarking scheme. Then, $\boldsymbol{w}_A \in \boldsymbol{h}^x$, $\boldsymbol{w}_B \in \boldsymbol{h}^x$, $\boldsymbol{w}_A \in \boldsymbol{h}_B$, and $\boldsymbol{w}_B \in \boldsymbol{h}_A$ in Table 1 correspond to

$$c_{T_A}(s(\boldsymbol{a}_A^{-1} \cdot e^{-1}(\boldsymbol{h}^x, \boldsymbol{h}_A), \boldsymbol{w}_A)) = 1 \quad (11)$$

$$c_{T_B}(s(\boldsymbol{a}_B^{-1} \cdot e^{-1}(\boldsymbol{h}^x, \boldsymbol{h}_B), \boldsymbol{w}_B)) = 1 \quad (12)$$

$$c_{T_A}(s(\boldsymbol{a}_A^{-1} \cdot e^{-1}(\boldsymbol{h}_B, \boldsymbol{h}_A), \boldsymbol{w}_A)) = 1 \quad (13)$$

$$c_{T_B}(s(\boldsymbol{a}_B^{-1} \cdot e^{-1}(\boldsymbol{h}_A, \boldsymbol{h}_B), \boldsymbol{w}_B)) = 1 \quad (14)$$

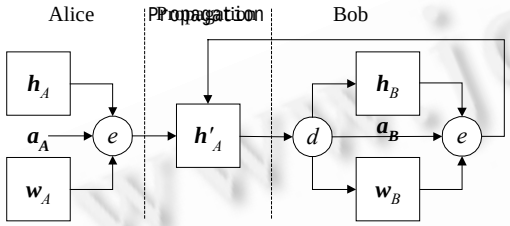


Fig.1 Reversibility in private schemes

Definition 2. (Reversibility of private schemes) A private watermarking scheme (e, e^{-1}, c_T) is reversible if there exists a decomposition $d(\boldsymbol{h}'_A) = (\boldsymbol{h}_B, \boldsymbol{w}_B, \boldsymbol{a}_B)$, which validates

(1) $\boldsymbol{h}'_A = e(\boldsymbol{h}_B, \boldsymbol{a}_B \cdot \boldsymbol{w}_B)$ (15)

(2) $\boldsymbol{w}_B = \boldsymbol{a}_B^{-1} \cdot e^{-1}(\boldsymbol{h}'_A, \boldsymbol{h}_B)$ (16)

Otherwise, (e, e^{-1}, c_T) is irreversible.

Theorem 1. A reversible private watermarking scheme (e, e^{-1}, c_T) in $V_n \mid A_n$ can be exploited to deceive itself in verifying the ownership of $\boldsymbol{h}$ and $\boldsymbol{h}'$. $\varphi_{\boldsymbol{w}_B - z_1, \boldsymbol{w}_B}$ $\varphi_{\boldsymbol{w}_B - z_1, \boldsymbol{w}_B}$ $\varphi_{\boldsymbol{w}_A - z_2, \boldsymbol{w}_A}$ $\boldsymbol{w}_B - z_1$ $\boldsymbol{w}_B - z_1$ $\boldsymbol{w}_A - z_2$ z_1 z_2

Proof. In (e, e^{-1}, c_{T_A}), when $\boldsymbol{h}^x = \boldsymbol{h}'_A$, Eq.(11) already holds, and Eq.(12) is implied by Eq.(15) and Eq.(16). Furthermore, to prove $\boldsymbol{w}_A \in \boldsymbol{h}_B$, Alice can extract $\boldsymbol{w}'_A$ from $\boldsymbol{h}_B$ through

$$\begin{aligned} \boldsymbol{w}'_A &= \boldsymbol{a}_A^{-1} \cdot e^{-1}(\boldsymbol{h}_B, \boldsymbol{h}_A) = \boldsymbol{a}_A^{-1} \otimes (\boldsymbol{h}_B \oplus (-\boldsymbol{h}_A)) = a_A^{-1} \otimes (\boldsymbol{h}'_A \oplus (-(\boldsymbol{a}_B \otimes \boldsymbol{w}_B)) \oplus (-\boldsymbol{h}_A)) \\ &= \boldsymbol{w}_A \oplus (-(\boldsymbol{a}_A^{-1} \times \boldsymbol{a}_B \otimes \boldsymbol{w}_B)) = \boldsymbol{w}_A - (\boldsymbol{a}_A^{-1} \times \boldsymbol{a}_B \otimes \boldsymbol{w}_B) \end{aligned} \quad (17)$$

On the other hand, to prove $\boldsymbol{w}_B \in \boldsymbol{h}_A$, Bob can extract $\boldsymbol{w}'_B$ from $\boldsymbol{h}_A$ in the same way through

$$\boldsymbol{w}'_B = \boldsymbol{a}_B^{-1} \cdot e^{-1}(\boldsymbol{h}_A, \boldsymbol{h}_B) = \boldsymbol{a}_B^{-1} \otimes (\boldsymbol{h}_A \oplus (-\boldsymbol{h}_B)) = \boldsymbol{w}_B - (\boldsymbol{a}_B^{-1} \times \boldsymbol{a}_A \otimes \boldsymbol{w}_A) \quad (18)$$

Because the measurements of Alive and Bob are symmetrical between Eq.(17) and Eq.(18), they are also symmet+rical between $t_A = s(\boldsymbol{w}'_A, \boldsymbol{w}_A)$ and $t_B = s(\boldsymbol{w}'_B, \boldsymbol{w}_B)$. If Alice has the fortune to make $t_A \geq t_B$, Bob has the same fortune to make $t_A \leq t_B$. Therefore, whether Eq.(13) or Eq.(14) is valid or not, Alice does not have any advantage over Bob. □

The significance of Theorem 1 is that it discloses the existence of the reverse engineering on $\boldsymbol{h}'_A$, which does not affect any perceptual quality. This paper calls these deceptions reverse engineering attacks.

Reversibility is a general phenomenon in many cases. For example, if one knows $\boldsymbol{s}$ in $\boldsymbol{s} = r(\boldsymbol{w}, \boldsymbol{k})$, he can fabricate a valid $(\boldsymbol{w}, \boldsymbol{k})$ pair[6]. If one knows $\boldsymbol{c}$ in $\boldsymbol{c} = g(\boldsymbol{w})$, he already has $\boldsymbol{w}$. That is one of the reasons that we simplify our research model

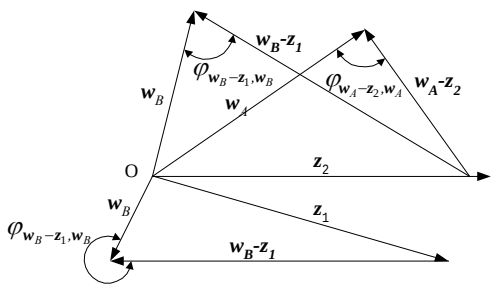


Fig.2 An illustration of Corollary 1

by disregarding channel coding and stream cipher, and presume that $\boldsymbol{w}$ has already been channel-coded and randomized.

Corollary 1. If a correlation between 2 watermark vectors in $V_n|A_n$ shows their similarity, and if a reversible private (e, e^{-1}, c_T) imposes no restriction on the formation of watermarks and scaling factors, Bob could prove that

(1) $\boldsymbol{w}_B$ exists in Alice's released version $\boldsymbol{h}'_A$.

(2) $\boldsymbol{w}_B$ exists in Alice's original version $\boldsymbol{h}_A$, and that $\boldsymbol{w}_B \in \boldsymbol{h}_A$ is more likely to happen than $\boldsymbol{w}_A \in \boldsymbol{h}_B$.

Proof. Because the private watermarking scheme is reversible, (1) is already established on Theorem 1. To prove (2), we first introduce the general cosine in V_n. For $\forall \boldsymbol{x}, \boldsymbol{y} \in V_n$, the general cosine is defined by[13]

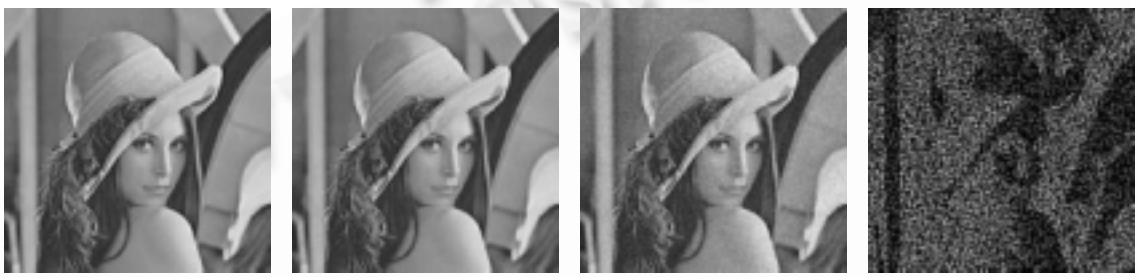
$$\cos(\phi_{x,y}) = [x, y]/(|x| \cdot |y|). \tag{19}$$

So, $t = s(x, y) = \cos(\phi_{x,y})$. Let $\boldsymbol{z}_1 = \boldsymbol{a}_B^{-1} \times \boldsymbol{a}_A \otimes \boldsymbol{w}_A$ and $\boldsymbol{z}_2 = \boldsymbol{a}_A^{-1} \times \boldsymbol{a}_B \otimes \boldsymbol{w}_B$. By Eq.(17) and Eq.(18), we have

$$t_B = s(\boldsymbol{w}'_B, \boldsymbol{w}_B) = s(\boldsymbol{w}_B - (\boldsymbol{a}_B^{-1} \times \boldsymbol{a}_A \otimes \boldsymbol{w}_A), \boldsymbol{w}_B) = s(\boldsymbol{w}_B - \boldsymbol{z}_1, \boldsymbol{w}_B) = \cos(\phi_{w_B - z_1, w_B}),$$

$$t_A = s(\boldsymbol{w}'_A, \boldsymbol{w}_A) = s(\boldsymbol{w}_A - (\boldsymbol{a}_A^{-1} \times \boldsymbol{a}_B \otimes \boldsymbol{w}_B), \boldsymbol{w}_A) = s(\boldsymbol{w}_A - \boldsymbol{z}_2, \boldsymbol{w}_A) = \cos(\phi_{w_A - z_2, w_A}).$$

Let us see Bob's opportunities to make $t_B \geq t_A$. He can change every element of $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ on his will by adjusting $\boldsymbol{a}_B$ and $\boldsymbol{w}_B$. Especially, he only needs to validate $\cos(\phi_{w_B - z_1, w_B}) \geq \cos(\phi_{w_A - z_2, w_A})$ that has many solutions (Fig.2). □



(a) Original image $\boldsymbol{h}_A$ (b) Released image $\boldsymbol{h}_A'$ (c) Forged original $\boldsymbol{h}_B$ (d) Forged $\boldsymbol{a}_B \cdot \boldsymbol{w}_B$

Fig.3 A group of related images acquired by inverse DCT in a reverse engineering attack on Example 1

Example 3. A reverse engineering attack on the private scheme in Example 1(Fig.3~4):

(1) Reverse engineering: Compute the DCT coefficients of a released image. Sort the middle n coefficients in zig-zag order and get $\boldsymbol{h}_A'$. Subtract an assumed $\Delta h'_A$ from $\boldsymbol{h}_A'$, and get $\boldsymbol{h}_B$. Fabricate an arbitrary $\boldsymbol{a}_B$ and deduce the forged $\boldsymbol{w}_B$ through $w_B = a_B^{-1} \cdot \Delta h'_A$. Or assume α_B and $\boldsymbol{w}_B$ first. Then, get $\boldsymbol{a}_B$ and $\boldsymbol{h}_B$ similarly.

(2) Verification: To prove $\boldsymbol{w}_B$ exists in $\boldsymbol{h}_A'$ or $\boldsymbol{h}_A$, t_B defined by Eq.(5) is computed and compared to the threshold T. To prove $\boldsymbol{w}_A$ exists in $\boldsymbol{h}_A'$ or $\boldsymbol{h}_B$, t_A defined by Eq.(5) is also computed and compared to T. Here, t_B might be larger than t_A.

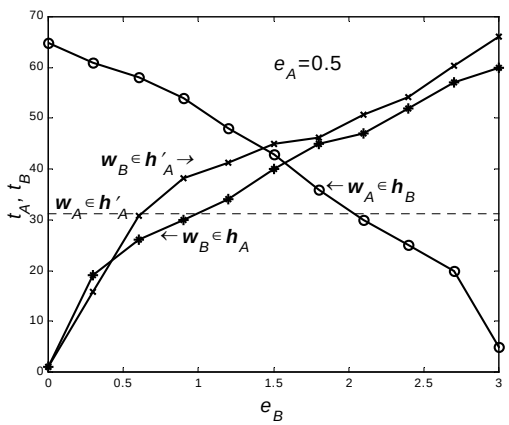


Fig.4 10 reverse engineering attacks on Example 1 (e=[$\boldsymbol{w}$,$\boldsymbol{w}$]/|$\boldsymbol{w}$|. Dash line represents the normal case)

In the above attacks, Bob has to forge much claimed data to validate Eq.(15). Bob's another simpler but less elegant way to exploit the reversibility, called quasi-reverse engineering attack, will be clarified below.

Definition 3. (Quasi-reversibility of private schemes) A private watermarking scheme (e, e^{-1}, c_T) is quasi-reversible if there exists an algorithm $d(\boldsymbol{h}'_A) = (\boldsymbol{h}_B, \boldsymbol{w}_B, \boldsymbol{a}_B)$, which validates

(1) $\boldsymbol{h}'_B = e(\boldsymbol{h}_B, \boldsymbol{a}_B \cdot \boldsymbol{w}_B)$ (20)

(2) $c_{T_A}(s(\boldsymbol{a}_B^{-1}\cdot e^{-1}(\boldsymbol{h}'_A,\boldsymbol{h}_B),\boldsymbol{w}_B))=1$ (21)

where $\boldsymbol{h}_B$ and $\boldsymbol{h}'_B$ are perceptually similar to $\boldsymbol{h}'_A$ like $\boldsymbol{h}_A$. Otherwise, (e,e^{-1},c_T) is non-quasi-reversible.

Theorem 2. A quasi-reversible private watermarking scheme (e,e^{-1},c_T) in $V_n \mid A_n$ can be exploited to deceive itself in verifying the ownership of $\boldsymbol{h}_A$, $\boldsymbol{h}'_A$, $\boldsymbol{h}_B$ and $\boldsymbol{h}'_B$.

Proof. Omitted (In fact, Bob has more flexibilities now. And reversibility is just a particular case of quasi-reversibility. By Definition 3, Theorem 1 and Corollary 1, the proof is easy to give). □

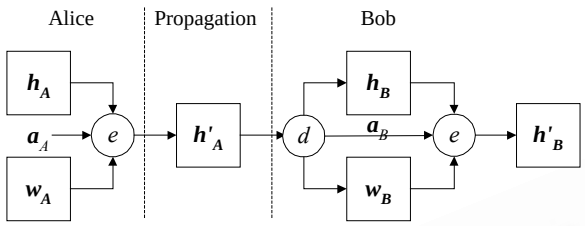


Fig.5 Quasi-Reversibility in private schemes

Through the above analysis, we find that both reversibility and quasi-reversibility originate from the fact that the watermarking schemes do not restrict the formation of both watermarks and scaling factors. But how to make restrictions, and how many restrictions to be made? Let us see Algorithm 4 before investigating the questions.

Algorithm 4. An irreversible and anti-quasi-reversible private watermarking scheme:

(1) Embedding: (a) Let $s=hash(\boldsymbol{h})=(s_1,s_2,\ldots,s_m)$, where $hash(\cdot)$ represents an one-way hash function. (b) Let $\boldsymbol{a}=v(\boldsymbol{h})$, where $v(\cdot)$, which exploits the HPS to maximizing the embedding intensity and the recognition threshold T, represents a perceptually adaptive function generating the scaling factor $\boldsymbol{a}$. (c) Let $\boldsymbol{r}=lfsr(\boldsymbol{s})=(r_1,r_2,\ldots,r_n)$, where $lfsr(\cdot)$ represents the processing of a linear feedback shift register (LFSR), $\boldsymbol{r}$ denotes a m-sequence grouped into n elements, and $\boldsymbol{s}$ is the seed of the LFSR. Optionally, the exclusive-OR operation could be performed between bit-streams of meaningful information and $\boldsymbol{r}$. (d) Let $\boldsymbol{w}=\boldsymbol{r}$, and then let $\boldsymbol{h}'=e(\boldsymbol{h},\boldsymbol{a}\cdot\boldsymbol{w})$.

(2) Extraction: Regenerate $\boldsymbol{s}$, $\boldsymbol{r}$, and $\boldsymbol{a}$ from $\boldsymbol{h}$. Extract the watermark through $\boldsymbol{w}'=\boldsymbol{a}^{-1}\cdot e^{-1}(\boldsymbol{h}',\boldsymbol{a}\cdot\boldsymbol{r})$.

(3) Verification: Draw the conclusion according to Eq.(3).

Let us investigate the validation of Algorithm 4. Its irreversibility lies in the fact that on one hand, Alice can easily embed her watermark, on the other hand, Bob has great difficulty dividing $\boldsymbol{h}'_A$ into $\boldsymbol{h}_B$ and $\boldsymbol{a}_B\cdot\boldsymbol{w}_B$, and validating both $\boldsymbol{w}_B=lfsr(hash(\boldsymbol{h}_B))$ and $\boldsymbol{a}_B=v(\boldsymbol{h}_B)$. If Bob launches attacks by adjusting $\boldsymbol{h}_B$, he has to solve the equation $v(\boldsymbol{h}_B)\cdot lfsr(hash(\boldsymbol{h}_B))=\boldsymbol{h}'_A-\boldsymbol{h}_B$, which can easily be proven difficult to solve with the one-way function's attribute in cryptography[6]. Similarly, the anti-quasi-reversibility of Algorithm 4 lies in the fact that after Alice has exploited most channel capacity of the HPS[1,13], Bob has great difficulty validating $\boldsymbol{w}_B=lfsr(hash(\boldsymbol{h}_B))$, $\boldsymbol{a}_B=v(\boldsymbol{h}_B)$ and Eq.(21), and keeping $\boldsymbol{h}_B$ and $\boldsymbol{h}'_B$ perceptually similar to $\boldsymbol{h}'_A$ like $\boldsymbol{h}_A$. Because no one is able to prevent Bob from forging $\boldsymbol{h}'_B$ by means of Eq.(20) in an active attack, we regard Algorithm 4 as an anti-quasi-reversible scheme instead of a non-quasi-reversible one, which will rely on advanced adaptive technology at last.

In spite of its obvious irreversibility, Algorithm 4 seems a little complicated. So we present 2 simplified versions here in brief. In the first version, step (a) is not used, and $\boldsymbol{a}$ or part of it becomes the seed of LFSR. Now, Bob has to crack $v(\boldsymbol{h}_B)\cdot lfsr(v(\boldsymbol{h}_B))=\boldsymbol{h}'_A-\boldsymbol{h}_B$ to deduce his $\boldsymbol{h}_B$. Fortunately, many perceptual analysis technologies are nonlinear and not one-to-one mapping, so the equation either is insolvable or could only be given homogeneous solutions[10,11]. In fact, even $v(\boldsymbol{h}_B)\cdot\boldsymbol{w}_B=\boldsymbol{h}'_A-\boldsymbol{h}_B$, where $\boldsymbol{w}_B$ is an arbitrary vector composed of either 1 or 0, might be insolvable. We call them irreversible adaptive technologies so as to differ from the rudimentary ones used in Example 1 and 2. The second revised version, which is somewhat obsolete, allows $\boldsymbol{a}$ to be a constant and neglects step (b).

3 Reversibility and Deceptions in Public Watermarking Schemes

Reversibility and quasi-reversibility have their corresponding forms in public watermarking schemes. All possible verification conclusions of public schemes are listed in Table 2. Because the case No.2 in Table 2 corresponds to the

situation of lacking robustness that we presume, we only discuss the case No.3.

Table 2 Possible verification conclusions in public schemes
($\boldsymbol{h}^x$ represents $\boldsymbol{h}_A'$ or $\boldsymbol{h}_B'$, whose ownership is in dispute, and $\in$ represents 'exists in')

No.	$\boldsymbol{w}_A \in \boldsymbol{h}^x$	$\boldsymbol{w}_B \in \boldsymbol{h}^x$	conclusion
1	1	0	Alice
2	0	1	Bob
3	1	1	Alice and Bob

Definition 4. (Reversibility of public schemes) A public watermarking scheme (e, e^{-1}, c_T) is reversible if there exists a decomposition $d(\boldsymbol{h}_A') = (\boldsymbol{h}_B, \boldsymbol{a}_B, \boldsymbol{w}_B)$, which validates Eq.(15) and

$$c_{T_A}(e^{-1}(\boldsymbol{h}_A', \boldsymbol{a}_B, \boldsymbol{w}_B)) = 1. \tag{22}$$

Otherwise, (e, e^{-1}, c_T) is irreversible.

Definition 5. (Quasi-Reversibility of public schemes) A public watermarking scheme (e, e^{-1}, c_T) is quasi-reversible if there exists an algorithm $d(\boldsymbol{h}_B') = (\boldsymbol{h}_B, \boldsymbol{a}_B, \boldsymbol{w}_B)$, which validates Eq.(20) and

$$c_{T_A}(e^{-1}(\boldsymbol{h}_B', \boldsymbol{a}_B, \boldsymbol{w}_B)) = 1, \tag{23}$$

where $\boldsymbol{h}_B'$ is perceptually similar to $\boldsymbol{h}_A'$. Otherwise, (e, e^{-1}, c_T) is non-quasi-reversible.

Theorem 3. Any public watermarking scheme (e, e^{-1}, c_T) is reversible. And it is quasi-reversible if the embedding has not enough intensity compared to what the HPS allows.

Proof. Because $\boldsymbol{h}$ is unavailable in the extraction, $\boldsymbol{a}$ and T are either constant or deducible from $\boldsymbol{h}'$, and $\boldsymbol{w}$ bears no relation to any other data except the claimed key or PN. Then, Bob could launch the following attacks:

(1) Reverse engineering: Fabricate or deduce $\boldsymbol{w}_B$ and $\boldsymbol{a}_B$ from $\boldsymbol{h}_A'$. Let $t_B = e^{-1}(\boldsymbol{h}_A', \boldsymbol{a}_B, \boldsymbol{w}_B)$ and make t_B as large as possible. If $\boldsymbol{w}_B$ and $\boldsymbol{w}_A$ have the similar statistical characteristic, which is often implied by the scheme itself, Alice does not gain any advantage over Bob in verifying the ownership of $\boldsymbol{h}_A'$ by just computing t_A.

(2) Quasi-Reversible engineering: Deduce a perceptually similar $\boldsymbol{h}_B$ from $\boldsymbol{h}_A'$, or just let $\boldsymbol{h}_B$=$\boldsymbol{h}_A'$. Fabricate or deduce $\boldsymbol{w}_B$ and $\boldsymbol{a}_B$ from $\boldsymbol{h}_A'$. Compute $\boldsymbol{h}_B'$ by Eq.(20). Then, $\boldsymbol{h}_B'$ contains both $\boldsymbol{w}_A$ and $\boldsymbol{w}_B$, but $\boldsymbol{w}_A$ might be somewhat damaged. If Alice's embedding has not enough intensity compared to what the HPS allows, Bob's embedding could exploit most channel capacity so that Eq.(23) might be validated, and the perceptual similarity among $\boldsymbol{h}_A'$, $\boldsymbol{h}_B$, and $\boldsymbol{h}_B'$ might be maintained[13]. □

Some papers have discussed the second attack in the above proof, which is also called multi-watermark attack[3,4]. Here, an example exploiting both the reversibility and the quasi-reversibility is given.

Example 4. Reverse and quasi-reverse engineering attacks on the public scheme in Example 2 (Figs.6, 7):

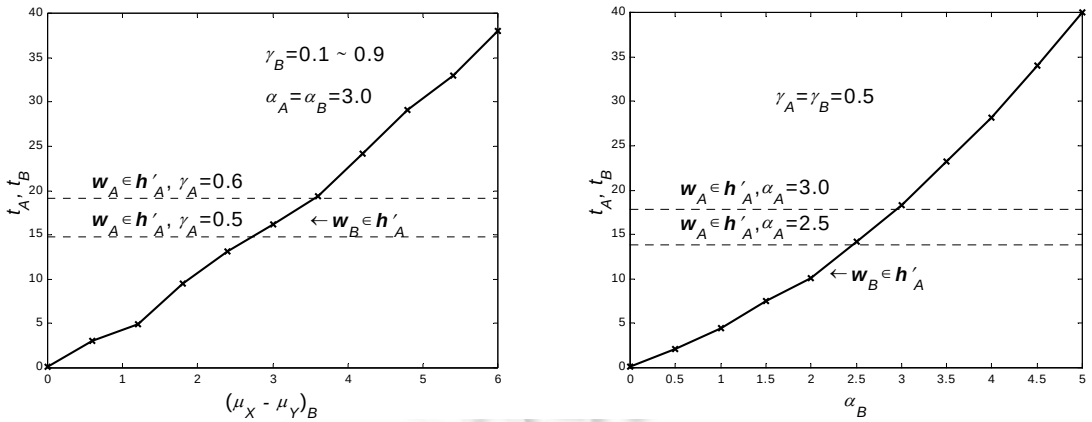
(a) Original image $\boldsymbol{h}_A$ (b) Released image $\boldsymbol{h}_A'$ (c) Multi-Watermarked $\boldsymbol{h}_B'$ (d) Forged watermark $\boldsymbol{w}_B$

Fig.6 Some related images in a reverse engineering attack and a multi-watermark attack on Example 2

(1) Reverse engineering: At the beginning, Bob assumes an arbitrary $\boldsymbol{w}_B$. He then keeps adjusting the area of subset X and subset Y to change μ_X, μ_Y, and σ_{XY} until t_B defined by Eq.(9) is large enough. He finally records the last $\boldsymbol{w}_B$, and claims it to be his watermark in $\boldsymbol{h}_A'$.

(2) Multi-Watermark: Having exploited the HPS, Bob derives $\boldsymbol{h}_B$ from $\boldsymbol{h}_A'$ to enlarge the channel capacity. Bob also

generates a properly distributed $\boldsymbol{w}_B$ and a just intense $\boldsymbol{a}_B$. Then, he gets $\boldsymbol{h}_B'$ by means of Eq.(20). In verification, Bob's test statistic is apparently larger than Alice's if he can embed more energy.



(a) 10 attacks by adjusting areas and value distribution (b) 10 attacks by embedding multiple watermarks

Fig.7 10 reverse engineering attacks and 10 multi-watermark attacks on Example 2
(γ is the ratio of X's area to Y's area. Dash lines represent the normal cases)

4 Conclusions

Having investigated the reversibility and quasi-reversibility of adaptive watermarking schemes, we think we can answer the questions presented at the beginning now. In general, the reversibility and quasi-reversibility also exist in additive adaptive watermarking schemes, which impose no restriction on the formation of watermarks and scaling factors, or use less advanced adaptive technologies. A fabricated watermark together with an arbitrary scaling factor gives an attacker more flexibility in reverse engineering attacks. A rough adaptive technology leaves too much channel capacity to quasi-reverse engineering attackers. These cases are threats to the security of watermarking.

Reversible or quasi-reversible watermarking schemes can be revised to enhance their security. The demand for original data in extraction makes private schemes more inconvenient, but it can be used to enhance their security and verify ownership in a more convincing way. Imposing restrictions on the formation of watermarks and scaling factors is an applicable way of counteracting the reversibility and quasi-reversibility. We also find that some adaptive technologies are essentially irreversible so that only the formation of watermarks should be loosely restricted. Our algorithm that makes watermarks be the hash value and scaling factors be the one-way HPS analysis result of original data forces the attackers to solve the difficult problems in cryptography, algebra or signal processing. Public schemes are more feasible because they do not need to provide original data in extraction. But the feasibility results in their essential reversibility and less convincing ownership verification. As is the cases with private schemes, advanced adaptive technologies exploiting watermarking channel well help resist quasi-reversibility in public schemes.

We find that an authority center for every ownership case, which exists in the timestamp protocol[6], is not needed, and that just an organization for regulating algorithms is needed, although the revised scheme seems more complicated. Therefore, we believe that the applications of an irreversible scheme will be carried out at an acceptable cost.

References:

[1] Langelaar, G.C., Setyawan, I., Lagendijk, R.L. Watermarking digital image and video data, a state-of-the-art overview. IEEE Signal Processing Magazine, 2000,17(5):20~46.

[2] Sun, S.H., Lu, Z.M. Digital watermarking technology. Chinese Journal of Electronics, 2000,28(8):85~90 (in Chinese).

[3] Cox, I.J., Linnartz, J.M.G. Some general methods for tempering with watermarks. IEEE Journal on Selected Areas in Communications, 1998,16(4):587~593.

[4] Kutter, M., Petitcolas, F. A fair benchmark for image watermarking systems. In: Wong, P.W., Delp, E.J., eds. SPIE Proceedings of the Security and Watermarking of Multimedia Contents. Bellingham: SPIE Press, 1999. 226~239.

[5] Cox, I.J., Kilian, J., Leighton, T., Shamoon, T. Secure spread spectrum watermarking for images, audio and video. In: Kunt, M., ed. Proceedings of the IEEE International Conference On Image Processing (ICIP'96). New York: IEEE Press, 1996. 243~246.

[6] Feng, D.G., Qing, S.H. Information Security—Key Theories and Applications. Beijing: Defense Industry Press, 2000 (in Chinese).

[7] Craver, S., Memon, N., Yeo, B.L., Yeung, M. Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications. IEEE Journal on Selected Areas in Communications, 1998,16(4):573~586.

[8] Qiao, L., Nahrstedt, K. Watermarking methods for MPEG encoded video: towards resolving rightful ownership. In: Cobb, G., Ichikawa, T., eds. Proceedings of the IEEE International Conference on Multimedia Computing and Systems. New York: IEEE Press, 1998. 276~285.

[9] Nikolaidis, N., Pitas, I. Robust watermarking in the spatial domain. Signal Processing, 1998,66:385~403.

[10] Podilchuk, C.I., Zeng, W. Image-Adaptive watermarking using visual models. IEEE Journal on Selected Areas in Communications, 1998,16(4):525~539.

[11] Swanson, M.D., Zhu, B., Tewfik, A. H., Boney, L. Robust audio watermarking using perceptual masking. Signal Processing, 1998,66:337~355.

[12] Jacobson, N. Lectures in Abstract Algebra. Berlin: Springer-Verlag, 1980.

[13] Huang, Y., Chen, K.F. Channel capacity of digital watermarking channel. In: Chen, K.F., Feng, D.G., Wu, W.L., eds. Proceedings of the 2nd Conference of Chinese Information and Communication Security. Beijing: Science Press, 2001. 122~127 (in Chinese).

附中文参考文献：

[2] 孙圣和,陆哲明.数字水印处理技术.电子学报,2000,28(8):85~90.

[6] 冯登国,卿斯汉.信息安全——核心理论与实践.北京:国防工业出版社,2000.

[13] 黄旸,陈克非.数字水印信道的信道容量.见:陈克非,冯登国,吴文玲编,第 2 届中国信息和通信安全学术会议论文集.北京:科学出版社,2001.122~127.

自适应数字水印中的可反向性以及相关的欺骗和对策

赵险峰, 汪为农, 陈克非

(上海交通大学 计算机科学与工程系,上海 200030)

摘要: 为进一步加强当前数字水印的安全性,对得到广泛研究的自适应水印中的可反向性问题进行了探讨.首先对水印体制进行了分类和抽象.随后基于自适应水印对植入水印和调节因子的形成没有约束的前提,对存在的可反向性和半可反向性问题,以及由此引起的反向工程攻击和半反向工程攻击进行了定义、分析和实验,指出了它们对相关数字所有权验证的干扰甚至否定作用.最后得出了对植入水印和调节因子的形成都必须进行约束的结论,指出了一些自适应技术本身的不可反向性对安全性的增强作用.让植入水印和调节因子的形成单向依赖于原始媒体,并充分利用人类感知系统,使水印体制对上述攻击具备抵抗性,增强了数字所有权验证的可靠性.

关键词: 可反向性;数字水印;数字版权保护;信息隐藏;隐秘分析

中图法分类号: TP309 文献标识码: A