A Method To Prevent Attackers From Counterfeiting Valid Rightful Ownership Proof In Watermarking

Zhiyi Qu
Information Building, Room 220 #,
School of Information Science and Engineering,
Lanzhou University, Gansu, Peoples Republic of China.
Zip-Code: 730000
Phone: (+86) 0931-891-3843
E_Mail: quzy@vip.163.com

Weilin Wang
Graduate School of Computer Science Application,
School of Information Science and Engineering,
Lanzhou University, Gansu, Peoples Republic of China.
Zip-Code:730000
Phone: (+86)0931-891-4000-8220
E_Mail: wang_weilin@sohu.com

Huanxia Wu
Graduate School of Computer Science Application,
School of Information Science and Engineering,
Lanzhou University, Gansu, Peoples Republic of China,
Zip-Code:730000
Phone:(+86)0931-862-8085
E_Mail: whx_81@163.com
Abstract: Digital watermarks have been proposed in recent literature as a means for copyright protection of multimedia data. In this paper, we will address two types of existing watermarking attacks for copyright protection and propose our technique of resolving the attacks. Counterfeiting the allowable watermarks and “original” images by the inverse operation on the watermark contained features will bring forth multiple ownership claims for the watermarked images and invalidate the rightful ownership proof of the watermarking. This inverse attack can be foiled by ensuring the validity of the inserted watermarks and wrapping the prior knowledge of extracting the watermarking features. Employing the secret feature extraction operators can prevent the watermark contained features exposing to attackers and thus hide the prior knowledge. The one-way mapped watermarking key can ensure the vindicability and validity of the watermarking scheme. That multiple allowable ownership labels in an image are forward introduced by a unique watermarking scheme is another intractable problem for rightful ownership settling. Time-stamp marked watermarks can provide us the help of resolving this problem. In this paper, the inverse attack and forward attack have been settled in our discussion, and the detected watermarks can provide us the correct proof of the ownership dispute resolving.

Index Terms — Feature-based watermarking, inverse counterfeiting attack, forward watermarking attack, multimedia copyright protection.

1 INTRODUCTION

The rapid growth of digital imagery has called upon the needs for effective copyright protection tools. Various watermarking schemes and software products have been introduced recently in an attempt to address this growing concern. It may have appeared from some ensuing work that the most important property of the watermarking schemes for providing unique proof of ownership was their robustness — their ability to survive despite malicious attempts at removal, or common image process operations. Watermarking schemes have been proposed and shown to be robust. The idea has focused on the means to label an image invisibly and robustness of the inserted labels against malicious attacks. As a result, the concerns regarding what watermarks can achieve or fail to achieve may not have been properly addressed. It was demonstrated that the ability to embed robust watermarks does not necessarily imply the ability to establish ownership in [1], unless certain requirements are imposed on the schemes used in inserting the watermarks. Considering that we are in a distributed system similar to the World Wide Web, we have a number of users who create digital images and make them available to many other users to view and potentially copy. If we have no central authority actively monitoring, maintaining, or enforcing ownership rights to information, a user wants to be able to retain ownership rights to an image is impossible only by using one of the robust watermarking techniques reported in the literature. And the watermarking techniques can not prove beyond a reasonable doubt that an image in dispute has actually been derived from a user’s original. So the watermarking techniques need some standardization or specification of requirements to resolve rightful ownership of an image watermarked with multiple ownership labels. There have been presented the Non-Invertible Watermarking Schemes in [2] and the Non-Quasi-invertible watermarking schemes in [3] to resolving this concern. In these schemes, the watermarking techniques employed some specifications and could invalidate some attacks, such as SWICO attack.
or TWICO attack and so on. While in these schemes, the authors had not limited the means to achieve the end. The enforcement of some requirements and hiding some prerequisites in the watermarking technique can also foil the attacks. Based on the conception, we will address our idea how to keep the watermarks safe and valid for resolving the ownership dispute.

In this paper, we will address the fault of exposing some pivotal watermarking information in depth and present why exposing some pivotal watermarking information is vulnerable for existing watermarking algorithm. Then we propose the improved idea. In section 2, we will review the inverse attacks and find out the necessary factors for the attacks and propose the improved conception of settling the problem. In section 3, we will discuss our new means of settling the inverse attack and forward attack. The forward attack imposed by introducing multiple ownership labels via a unique watermarking scheme can be resolved by associating the watermarks with a time-stamp. And in section 4 we will analyze our test data for our watermarking scheme. Finally, we have a conclusion for our work in section 5.

2 CLASSICAL MEANS TO COUNTERFEIT RIGHTFUL ONERSHIP PROOF

It has been generally assumed that invisible watermarking schemes may be used to protect the rights of copyright owners; at the very least, the labels extracted from watermarked images can be used to identify the rightful owner. However, the researchers have recently found out that the ending does not work as what we hoped originally. Suppose Alice and Bob use the same digital watermarking technique to watermark their images. This means that there is one unique decoding scheme to extract the labels embedded in the images. If the label we detected in the watermarked image matches the particular watermark labels of Alice, then it is deemed that the image belongs to her. Similarly, if the label matches the claimed watermark labels of Bob, then we believe that the image belong to him. In fact, the hypothesis only works correctly in case 1 and case 2 as showed in table 1. Of course there are the so called original images (un-watermarked images). Let \( I_{\text{alice}}^o \) and \( I_{\text{alice}}^w \) represent the original image and watermarked image of Alice respectively; similarly, \( I_{\text{bob}}^o \) and \( I_{\text{bob}}^w \) represent the original image and watermarked image of Bob. In the case 1 and case 2 in table 1, we can extract or detect only one valid label from the watermarked image. While more than one watermark labels are extracted from the watermarked images, for example, we can extract both Alice’s and Bob’s watermark labels from a watermarked image, then the hypothesis can not work. The scenario really exists. For example, if the watermarking scheme has no limit, due to the limitless copy or modification to the published images, all users can add another watermark label in a watermarked image. When more than one watermark labels are inserted in an image, we may detect them all and the watermark labels used to claim the unique ownership have no help for the ownership dispute resolving. When we can not determine the rightful ownership of an image only depending on the watermarked images, we can look to the original images of the owners. Suppose both Alice and Bob have detected their marks in a watermarked image, and both claim they should own the image. If Alice keeps her original image locked away, she can ask Bob for his original image and check if it contains her watermark. Similarly, Bob can ask Alice for her original image and check for his watermark. If Bob obtained Alice’s watermarked image and inserted his
watermark label into it to fabricate his “original” image, then both Bob’s “original” and watermarked images contain Alice’s mark. Alice’s original image does not contain Bob’s. Thus, by keeping her original image locked away with the details of the watermark label, Alice can ensure that any copy of $I^w_{alice}$ will contain her mark and it can easily foil such ex post facto watermarking attack. The Case 3 and Case 4 in table 1 show the scenario and resolving result.

Table 1. The determination of ownership from watermark presence detection. ‘OK’ indicates the presence of anticipant watermarks, ‘NO’ indicates the absence, and ‘—’ represents do not care’s.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Alice</th>
<th>Bob</th>
<th>Derived Ownership</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>$I^o_{alice}$</td>
<td>$I^w_{alice}$</td>
<td>$I^o_{bob}$</td>
</tr>
<tr>
<td>Case 2</td>
<td>—</td>
<td>OK</td>
<td>—</td>
</tr>
<tr>
<td>Case 3</td>
<td>NO</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>Case 4</td>
<td>OK</td>
<td>OK</td>
<td>NO</td>
</tr>
<tr>
<td>Case 5</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
</tbody>
</table>

However, the above hypothesis only works correctly under the presupposition that we can not detect the counterfeit watermarks from the true original images. If both owners can detect their anticipant watermarks from the opposing party’s original images, as case 5 showed in table, the detection result is also helpless to resolve the ownership dispute. Suppose Bob is the attacker and his original image is derived form Alice’s watermarked image. In case 5, Bob can also extract his watermark labels from Alice’s original image even he has not accessed Alice’s original image in the attack. In the follow, we will introduce that how Bob carries out the attack.

Given only watermarked image $I^w_{alice}$ of Alice, Bob intends to counterfeit an “original” to implement the scenario in the case 5 in table 1, in which both Alice and Bob can extract their anticipant watermark labels from their “modified version” images. A variety of encoding and corresponding decoding processes established in the literature are feature-based. In these schemes, we embed a watermark $S=\{s_1, s_2, \ldots\}$ into a set of derived features $F(I) = \{f_1(I), f_2(I), \ldots\}$. The embedding process is achieved by an insertion operation that we denote by the symbol $\oplus$, i.e., $f^*_i = f_i \oplus s_i$. The extraction operation in the decoding process can be denote by $\bigcirc$, i.e., $f^*_i \bigcirc f_i = s_i$. Usually, the feature set $\{f_1(I), f_2(I), \ldots\}$ is chosen such that slight modification of individual features does not perceptually degrade image $I$. In the transform-domain watermarking technique, the transform-domain coefficients that contain significant energy are chosen. When Bob obtains the watermarked image $I^w$ of Alice, he can construct a counterfeit “original” image by extracting a chosen watermark $T = \{t_1, t_2, \ldots\}$ from some feature set $D(I^w) = \{f^*_i(I^w)\}$ to generate an image $I^{wo}$ such that

$$f^*_i(I^{wo}) = f^*_i(I^w) \bigcirc t_i$$

When Bob detects his watermarks in Alice’s original image $I$, he first extracts $T = \{t_1, t_2, \ldots\}$ as follows:
\[ t_i' = f_i'(I) \cup f_i'(I^{\text{wo}}) \]  \hspace{1cm} (2)

Because of the robustness of the set \( D'(I^{\text{w}}) \) against perceptually insignificant modification, we can expect that

\[ f_i'(I) \approx f_i'(I^{\text{wo}}) \]  \hspace{1cm} (3)

Combining (1)-(3), we have \( t_i \approx t_i' \), so that the correlation between \( T \) and \( T' \) is large and implies that Bob can extract his watermark label from Alice’s original image. Then Bob can claim that the true original \( I \) contains his watermark \( T \) and that \( I \) is a modified version of \( I^{\text{wo}} \). Conversely, the robustness of the watermarking scheme can ensure that the watermarked image and original image of Bob contain Alice’ watermark label, then Alice can argue that \( I^{\text{wo}} \) is a modified version of \( I \). So the case 5 happens in table 1. Usually, we call the instance of such attack which involves only one watermarked image as **SWICO** (Single-Watermarked-Image-Counterfeit-Original) attack.

Another watermarking scheme was introduced to avoid the SWICO attack. We first produce an one-way hash bit \( \{b_1,b_2,\ldots\} \) of the original image before computing the feature set. We then use two slightly different equations for inserting the watermark vector elements in formula (4) as follows:

\[ f_i' = \begin{cases} f_i \oplus s_i, b_i = 0; \\ f_i \Theta s_i, b_i = 1. \end{cases} \]  \hspace{1cm} (4)

For each feature \( f_i \) to be modified, we choose one of the two formulas depending on the value of the hash bit \( b_i \). Anticipating a possible attack involving rearranges watermark elements to match the required hash values. The above scheme is non-invertible and appears to foil the SWICO attack. However, it is not effective against the TWICO attack. In TWICO attack, we construct any watermark works by mix-and-match method. Suppose Alice is the true owner and Bob is the attacker. When Bob obtains Alice’s watermarked image, he can compute a watermark \( S \) with a bit string “1111...1” and another watermark \( T \) using the string “0000...0”. Then the two resulting images are averaged to yield an image \( I^{\text{wo}} \) which he claims to be his “original” image. This image is then hashed, and the attack watermark \( S' \) is computed thus: if the ith hash bit of the average is 0, Bob uses the ith vector element of the second watermark; else he uses the ith element of the first, to build the watermark vector \( S' \). Finally, he watermarks \( I^{\text{wo}} \) with \( S' \) to produce a watermarked image \( I^{\text{ww}} \), which he claims to be the watermarked version of his “original”. \( S' \) is a vector composed of elements from S and T so as to match the hash bit of \( I \). We call such a counterfeit-attack involving two watermarked versions of an image as **TWICO** (Twin-Watermarked-Images-Counterfeit-Original) attack.

The above attacks invalidate the rightful evidence of watermarking. In fact, both the watermarking and attack are to ask for a solution under some prerequisites. If the prerequisites do not fit the requirement, it will fail to achieve the end. Usually, the watermarks are embedded on the features in feature-based algorithm. Attackers can extract the anticipant features and then perform the corresponding modification on the features as the SWICO attack and TWICO attack do. How to yield a secret watermark
feature set and keep it safe are the key for the security of watermarks. This is very difficult in watermarking, because different watermarking schemes can extract their anticipant features and then the attacks are implemented on these features. So we only discuss our improved ideas under a unique watermarking scheme for all users. If attackers can not extract the anticipant features from the watermarked image, then it has less menace to the inserted watermarks. At the same time, the operators which are used to extract the features and the inserted watermarks are one-way decided by the watermarking key. The one-way function ensures the validity of the algorithm and avoids the effective attacks by counterfeiting allowable watermarks and images. In the insertion process, we reinforce the correlation for the inserted marks and make it is impossible to detach valid watermarks out. In the next section, we will introduce the new watermarking scheme in depth, which has higher security and can foil the inverse attack as above introduced.

3 NEW IDEAS TO FOIL THE INVERSE ATTACK AND FORWARD ATTACK

We obtain the feature set primarily in many feature-based watermarking schemes and the operators used to compute the features are public. For example, in the DCT transform-based watermarking scheme, we usually embed a set of independent and identically distributed samples drawn from a Gaussian distribution into the perceptually most significant frequency components of the data. The open operators of the DCT transform may provide attackers the prior knowledge to extract the watermark contained features from watermarked images, thus the crucial watermarking information is exposed to the attackers. The vision imperceptible characteristic of the invisible watermarking can also provide attackers the prior knowledge to orient the watermark contained features exactly from the transform-domain coefficients of the watermarked images. In many transform-based invisible watermarking schemes, the above two terms are exposed to attackers factually. Thus the inverse attacks happen easily. In order to avoid the awful plight, we can let the feature generating be decided by a secret key and wrap the prior knowledge to extract the feature set in the watermarking scheme. Only the man who has the correct watermarking key can obtain the watermark contained features. Figure 1 has illustrated an effective watermarking scheme to hold the watermarking information and foil the reverse attack similar to SWICO attack.

![Diagram](image)

Figure 1. Encoding and Decoding signatures in an image (a) the builder of the feature extraction operators and watermarks (b) encoder (c) decoder and correlating comparator.
In the watermarking scheme as figure 1(a) showed, we first yield the secret feature extraction operator \(B = \{b_1, b_2, \ldots\}\) and a set of independent and identically distributed samples \(S = \{S_i\}\) drawn from a Gaussian distribution by the one-way function \(F\) with the watermarking key as seed. Then we can compute the anticipant feature set \(F(I) = \{f_1, f_2, \ldots\\}\) of the image using the formula (5) as follows.

\[
f_i = \langle I^n, b_i \rangle
\]

(5)

Where the operation \(\langle \cdot, \cdot \rangle\) is the dot production of two vectors and \(I^n\) is the n-dimensional image vector.

In the watermarking encoding process as figure 1(b) showed, we transform the image \(I\) into the image vector \(I^n\) first and then insert watermarks onto the vector with the encoding function \(E\). Finally the watermarked vector \(I^n_w\) is transformed back to the watermarked image \(I_w\). The encoding function is represented in formula (6). In the encoding process, the watermarking signature is not embedded onto the watermarking features directly, however, we can obtain it from the watermarking features in the decoding process as discussed later. In formula (6), the signature distributes on all the image block points and the modification to the image is relative to several factors: image feature \(f_i\), feature extraction operator \(b_i\) and watermark \(S_i\).

\[E(I^n) = I^n + \sum_i \alpha_i f_i b_i S_{ii}\]

(6)

In the watermarking detection process, we should detach the watermarking signature from the test image first and then compare the extracted signature with the anticipant signature to decide whether the test image contains the claimed watermarks. Figure 1 (c) shows the watermarking detection process. As in the watermarking encoding process, the test image \(X\) is transformed to the n-dimensional vector \(X^n\) first, then the decoding function \(D\) extracts the watermark contained features with the feature extraction operators. The decoding function is showed in formula (7), where the test image is usually noised and we deem it has been added a noise signature \(N\). From the formula (7), we can see that the watermarks hide in all image block points and can be reconstructed only by the corresponding operators. After the watermark contained features have been obtained, we can design a correlating comparator to determine the existence of the anticipant signature. Usually, it needs to detach the watermarks from the extracted features in the correlation detector, while it is no need in the statistic detector.

\[
f'_i = \langle X^n, b_i \rangle = f_i(1 + \alpha S_{ii}) + N_i
\]

(7)

Why can we deem that the watermarking scheme is more secure to the inverse attacks brought forward in section 2? The inserted watermarks are distributed on all image block points and the information losing of partial watermarked image points can not affect the reconstruction of the watermarks. The modification to the image has more correlation factors and the factors are one-way key-decided by the function in the encoding process, which can ensure the validity of the watermarks and foil the counterfeiting action by the inverse operations on the watermarked image. The one-way function is very important to the security of the watermarking scheme, because it ensures that it is almost impossible to counterfeit a watermark set and a matched feature extraction operator set at the same time under the limited watermarking key restriction. When attackers can not carry out the inverse action in which they map a set of
watermarks and a set of matched feature extraction operators to a valid watermarking key, thus the inverse attack is noneffective.

The watermarking scheme can keep the inverse counterfeiting attack away, however, it is incapable of preventing the attack which was introduced by embedding multiple valid watermark labels with the same watermarking scheme. Because with another valid watermarking key as seed, the watermarking scheme can obtain another set of feature extraction operators and matched watermarks. If attackers implement this attack to the watermarked image, the case 5 in table 1 occurs too. To resolve this attack involved by forward introducing multiple valid watermark labels, now no man has established an effective scheme. In [4] the paper had reported a method, which employed the time-stamp server in the system, to mark the creating time of the works. If the watermarking key associates a watermarking time-stamp obtained from the time-stamp server in the watermarking system, we can determine that the watermarked image marked later time-stamp is the counterfeit one. It is possible in the watermarking scheme.

Now we have discussed the inverse attack and forward attack we have met in the watermarking attack and proposed the corresponding solution. In the scheme, the security of the watermarking key is also important to keep the watermarking information safe in the watermarked image. To foil the inverse attack, the watermarking key can not be mapped from the counterfeit watermarks and corresponding feature extraction operators. The one-way function is responsible for this work. The validity of the watermarking key has to be provable. Some proposals in the published literature suggested that the watermarking key must be the mapping of an owner’s registered information. Only depending on this registered information obtained from the registration institution, it is also dangerous and means that we have high security requirement for the registration institution. If the watermarking key depends on a certifiable information and another secret information, we then have less security requirement for the registration institution.

As the above paragraph analyzed, the watermarking key has to bind a time-stamp information. Thus the watermarking key is the mapping of a function associating with the public key $K_{public}$, the secret key $K_{secret}$ and the time-stamp $T$. It is showed as follows.

$$Key = Bind(K_{secret}, K_{public}, T)$$

The binding function in formula (8) must be one-way. Then all the watermarking process is one-way and can be provable. The proposed prerequisite as above is feasible in the watermarking scheme and the detection result of the image watermarks is helpful for the rightful ownership resolving.

**4 THE TEST ON THE PROPOSED WATERMARKING SCHEME**

We can divide the watermarking scheme into two steps both in the watermarking encoding process and decoding process. The both processes have a same preparation step in which we yield the watermarking signature and feature extraction operators and this step must be finished before we carry out the other work. Other than the preparation step, the encoding process includes another watermarking insertion process and the decoding process includes another watermarking detection process.

The preparation step is responsible for generating the watermarking signature and feature extraction operators. As stated in the above section, the generating of the watermarking signature can be represented in formula (9) and the feature extraction
operators in formula (9) respectively.

\[ S = F_s(Bind(K_{secret}, K_{public}, T)) \]
\[ B = F_b(Bind(K_{secret}, K_{public}, T)) \]

In the statistic detector, the inserted signature is a set of samples drawn from some distribution similar to Gaussian distribution. For the detection validity, the size of the set must be large enough. In order to obtain the private feature extraction operators, there are two different types of means for us. We can reorder a set of public well performance orthogonal vectors and let the order be decided by the watermarking key. Another method is to obtain a set of random vectors with the watermarking key as seed and then orthogonalize them with the **Gram-Schmidt** algorithm.

The watermarking insertion had been discussed explicitly enough in section 3. Now we have a review on the watermarking detection. After we extract the watermark contained features in formula (7), we can detach the watermarks from the features and then determine the existence of the anticipant watermarks by the correlation output computed in formula (10).

\[ cor(s', s) = \frac{< s', s >}{\| s' \| \| s \|} \]  \hspace{1cm} (10)

The above formula is usually used in the correlation detector and it needs the participation of the original image to detach the watermarks from the watermark contained features. While in the statistic detector, it is no need to look to the original un-watermarked image. But the watermarking scheme must be amended. Figure 2 illustrates the amended watermarking scheme, in which the insertion process and detection process have employed the reference mode \( G(\cdot) \) and the correlating comparator has introduced the optimized sequence \( S_2 \). The encoding function in figure 2 (a) can be represented as follows:

\[ E(I') = I' + \sum_i G_i(I) b_i S_i \]  \hspace{1cm} (11)

Where the reference mode \( G(\cdot) \) could be represented as \( G_i(I) = \alpha f_i \).

![Figure 2](image)

**Figure 2.** The encoding process and correlating comparator employing the reference mode \( G(\cdot) \)
(a)encoder  \hspace{1cm} (b)correlating comparator.

In order to verify the existence of the anticipant watermarks from the extracted features \( f'_i \) in figure 2(b), we must design a correlating comparator. A statistic detector reported in [5] is showed as follows:
\[ q = \frac{\sum_{i=1}^{n} Y_i}{V_y \sqrt{n}} \]  \hspace{1cm} (13)

Where \( Y_i = f_i S_{zi} \), \( n \) is the size of the feature set, \( M_y \) and \( V_y^2 \) are the sample mean and the sample variance of \( Y_i \), given respectively by

\[ M_y = \frac{\sum_{i=1}^{n} Y_i}{n} \quad V_y^2 = \frac{\sum_{i=1}^{n} (Y_i - M_y)^2}{n-1} \]  \hspace{1cm} (14)

The detector output is compared to a threshold \( T \) to determine if the test image contains the claimed watermarks. if \( q > T \), the test image is declared to have been signed with the claimed signature \( \{S_{ii}\} \). The detection analysis of the watermarks is accomplished via the following hypothesis.

\[ H_0 : f_i = f_i + N_i \quad \text{not contain the claimed watermark} \]
\[ H_1 : f_i = f_i + G_i(I)S_{ii} + N_i \quad \text{contain the claimed watermark} \]  \hspace{1cm} (15)

Where \( N_i \) is noise, possibly resulted from some signal processing such as compression, lowpass filtering, etc.

Usually, the inserted watermarks may distort the image vision quality. To improve the invisibility of watermarks, we have some tips here. Firstly, we divide the watermark set into many subsets and segment the image into many non-overlapped blocks, then make a one-to-one mapping between the watermark sets and image blocks for watermarking insertion. The nonzero and smoothness of the random created vectors are poor. So we should wash out the malformed vectors and smooth the reserved vectors in the watermarking preparation step. The smoothed vectors are more jarless and effective for the watermarking invisibility and information holding. For colorful images, the color component of taking watermarks is also important for the watermarking invisibility. In [6] the author pointed out that the green component is more robust for the watermarking invisibility and information holding. With the watermarking scheme and tips, we have the tests on the 24bits color bitmaps and figure 3 illustrates the watermarking results.

Figure 3. Watermarked images (a)Boboon (b)Fishingboat (c)Lina (d)Pepper Where watermarks are inserted on the green component, the scalar \( \alpha = 0.1 \), image sub-block size is \( 8 \times 8 \), the quantity of the Gaussian distribution watermarks showed in (e) is between -3 and 3 and the size of the watermark set is 1000.

The quantity of the inserted signature is float in our watermarking scheme, while the gray scale of the image is integer. The quantification may discard the watermarking information. Under the quantification distortion rate 28, in table 2 we obtain the
correlation detector outputs of the watermarked images showed in figure 3. The obtained results are litter bad comparing to the DCT-based watermarking scheme.

Table 2. Correlation detector outputs

<table>
<thead>
<tr>
<th>Images</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boboon</td>
<td>0.9439543</td>
</tr>
<tr>
<td>FishingBoat</td>
<td>0.9238863</td>
</tr>
<tr>
<td>Lina</td>
<td>0.9604834</td>
</tr>
<tr>
<td>Peppers</td>
<td>0.9581465</td>
</tr>
</tbody>
</table>

The size of the watermark set and the characteristic of the detection sequence $S_2$ affect the statistic detector output evidently. In [5], the paper reported that the optimized detection sequence is $S_2 = kG_i S_{1i}$, where $k$ is a positive constant and the $G_i$ and $S_{1i}$ is independent of I. We obtain the statistic detector outputs in table 3 for several optimized detection sequence.

Table 3. The statistic detector outputs under different detection sequence. The results is obtained under the condition that the size of the watermark set is 5000 and the watermarking signature value is between -3 and 3.

<table>
<thead>
<tr>
<th>Image / Detection Sequence</th>
<th>$S_2 = \text{Sign}(S_{1i})$</th>
<th>$S_2 = S_{1i}$</th>
<th>$S_2 = \alpha S_{1i}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boboon</td>
<td>5.0585219</td>
<td>5.3401167</td>
<td>5.3401167</td>
</tr>
<tr>
<td>FishingBoat</td>
<td>4.9260381</td>
<td>5.4468283</td>
<td>5.4468283</td>
</tr>
<tr>
<td>Lina</td>
<td>5.5885254</td>
<td>5.7641024</td>
<td>5.7641024</td>
</tr>
<tr>
<td>Peper</td>
<td>5.1104161</td>
<td>5.4688220</td>
<td>5.4688220</td>
</tr>
</tbody>
</table>

In order to obtain anticipant statistic detector outputs, it requires that the samples of the extracted features follow the imitate normal distribution perfectly, and this is based on the large watermark set. If the watermark set is not large enough, the detector output may be not perfect. On the other hand, if the watermark set is too large, this will lead to inconsiderable computation time and unacceptable image vision quality. So we need to consider the balance among the watermarking invisibility, detection validity and computing complexity when we determine the size of the watermark set. In figure 4, we show the variety curve of the statistic detector outputs at different watermark set sizes for the four watermarked images illustrated in figure 3. At the beginning, the detector outputs increase obviously in the collectivity; while the size of the watermark set reaches at a critical point, the detector outputs increase imperceptibly, because the watermarks embedded in the image are in saturation and the image vision will be affected seriously.

![Figure 4. The statistic detector outputs for different watermark set sizes.](image)
In order to verify the ability of resisting the traditional watermarking attacks, we have taken the tests on the watermarked image Lina (resolution 512×512, bit depth 24, illustrated in figure 5(a)) with StirMark BenchWork tool of version 4.0. The results show that the watermarking scheme performs perfectly in resisting the PSNR attack and Self-Similarity attack, because we can obtain large detection outputs from the test images after the attacks. On JPEG compression attack, we show the statistic detector outputs under different image vision quality scalars Q in figure 6. Figure 6 illustrates that the detector outputs are small at small image vision quality scalars, but the detector outputs become larger with the image vision quality scalars increasing. It means that the detection error rate becomes litter when the image vision quality scalar increases. While the image vision quality scalar Q is less than 50 or 60, we can distinguish the true original image from the counterfeit one from the image vision point, because the counterfeit one is seriously distorted by the attack. So the scheme is also robust to the JPEG compression attack. What is more interesting is that the best choice of $S_2$ is not $S_2=\alpha S_1$ but $S_2=\text{Sign} (S_1)$ after the JPEG compression attack.

![Figure 5. Watermarked Lina image (a) and the resulted image (b) after Guassian filtering attack.](image)

In fact, the watermarking signature is also a noise signature over the original un-watermarked image. In order to hold the watermarking information after the mean smooth attack or filtering attack, it requires that the inserted watermarking signature must be as quite smooth as possible, or the attacks may remove the signature from the watermarked image, thus leading to the watermarking detection failing. Some filtering attack may distort the image vision quality badly, similar to the Gaussian filtering attack showed in figure 5(b), and we can distinguish which one is the true original from the image vision quality for the ownership dispute resolving.

These are other attacks, which do not change the value of a point in the image but would change the image size or figure, such as remove lines, add lines, cropping, scale and so on. After these kinds of attacks, the relative position of the points which carry the watermarks changes, thus leading to the feature extracting and detecting failing. But these attacks can be discovered from the image physical feature and can not affect claiming the rightful ownership of an image.
5 DISCUSSIONS AND CONCLUSIONS

Invalidating claiming the rightful ownership is the limit of the watermarking usage in copyright protection. If this problem can not be resolved, the watermarking for the rightful ownership controversy is insignificant. In the paper, we mainly focus on two types of attacks. In the inverse attack, attackers counterfeit a valid watermark set and attack images from the published watermarked image. In different watermarking scheme, the watermarking decoding is different. If the users employ different decoding schemes, they can always obtain their anticipant watermarking features, and the reverse attack is possible. It means that attackers can compute a set of feature extraction operators and a set of matched watermarks from the watermarked image and then map them back to a valid watermarking key. It is very difficult to foil the attack with different decoding schemes. So our discussion in the paper is only based on the unique decoding scheme and then the inverse attack is impossible in the circumstance.

Another type of attacks is the forward attack by imposing multiple valid watermark labels in an image with a unique watermarking scheme. This attack will also lead to detecting multiple valid watermark labels from the watermarked image and the anticipant watermarks from opposing party’s original image. Almost all the watermarking schemes are helpless to determine the true owner when we face this issue. The time-stamp associated watermarking key is a means to resolve the problem.

The means of foiling both types of attacks is by imposing some restriction for the watermarking process. For the inverse attack, we employ the one-way function and secret matched feature extraction operators to increase the weight of making the valid watermarks and images. For the forward attack, the time-stamp associated watermarking key is important for the validity of the watermarking scheme. The information transmission between the user end and time-stamp server may discover the time-stamp information. If the transmission process employs the zero-knowledge commit mode, then
the time-stamp transmission is also safe.

CITED REFERENCES