Cryptographic Copyright Protection for Digital Images based on Watermarking Techniques ⋆

Joseph Ó Ruanaidh [a], Holger Petersen [b], Alexander Herrigel [c], Shelby Pereira [a] and Thierry Pun [a]

[a] *University of Geneva, CUI - Vision Group,24, rue du Général- Dufour, CH-1211 Geneva, {ORuanaidh,Pereira,Pun}@cui.unige.ch*

[b] *Entrust Technologies Europe, Glatt Tower, CH-8301 Glattzentrum Holger.Petersen@entrust.ch*

[c] *Digital Copyright Technologies, Stauffacherstr. 149, CH-8023 Zürich, Herrigel@usa.net*

Abstract

In this paper [1] we present a new approach for the secure and robust copyright protection of digital images. We describe a system for generating digital watermarks and for trading watermarked images. The system is based on a new watermarking technique, which is *robust* against image transformation techniques such as *compression, rotation, translation, scaling* and *cropping*. It uses modulation of the magnitude components in Fourier space to embed a watermark and an accompanying template and, during watermark extraction, reads a template in the log polar transform of the frequency domain. The template is used for analyzing scaling and rotation suffered by the watermarked stego-image. The detection of the watermarks is also possible without any need for the original cover-image. In addition, the system applies asymmetric cryptographic protocols for different purposes, namely embedding/detecting the watermark and transferring watermarked data. The public key technique is applied for the construction of a one-way watermark embedding and the verification function to identify and prove the uniqueness of the watermark. Legal dispute resolution is supported for the multiple watermarking of a digital image without revealing the confidential keying information.

Key words: Copyright protection, digital images, digital watermark, asymmetric cryptography, robustness

[1] This work has been founded by the Swiss National Science Foundation under the SPP program (Grant. 5003-45334)

1 Introduction

The current rapid development and deployment of new IT technologies for the fast provision of commercial multimedia services has resulted in a strong demand for reliable and secure *copyright protection* techniques for multimedia data. Copyright protection of digital images is defined as the process of proving the intellectual property rights to a court of law against the unauthorized reproduction, processing, transformation or broadcasting of a digital image. Depending on the law in various countries, this process may be based on a prior registration of the copyright with a trusted third party. After successful registration, the copyright ownership is legally bound by a copyright notice, which is required to notify and prove copyright ownership.

Digital watermarking is a method for marking data sets, such as images, sound or video. A stego data set consists of the original data, the cover data set and a digital watermark that does not affect the data set's usability but that can be detected using dedicated analysis software or systems. Watermarking can, for example, be used for marking authorship or ownership of a data set.

Quite a number of different approaches [5,6,8–11,13,18,19,25,26,30–37,39,40] to digital watermarking have been proposed but only some of them implemented in commercial products. Due to the very short time and minimal effort needed for copying and distributing digital multimedia data, protection against copyright infringements is an important issue for the copyright owner and should form an integral part of the exploitation process for Internet based trading services. Today, the Internet community has not identified or accepted adequate copyright protection techniques. This is in direct contrast to the provision of secure transaction protocols, such as SSL [12].

2 State-of-the-art

Digital watermarking can be seen as being fundamentally a problem in digital communications [5]. Early methods of encoding watermarks consisted of no more than incrementing an image component to encode a binary '1' and decrementing to encode a '0' [6]. Tirkel et al. [34] and van Schyndel et al. [35] have applied the properties of $m-$sequences to produce oblivious watermarks resistant to filtering, cropping and reasonably robust to cryptographic attack. Matsui and Tanaka [19] have applied linear predictive coding for watermarking. Their approach to hide a watermark is to make the watermark resemble

⋆ All methods, procedures and schemes presented in this paper are based on the European patent application No. 97 810 708.4

quantization noise. Tirkel and Osborne [34] were the first to note the applicability of spread spectrum techniques to digital image watermarking. Spread spectrum has several advantageous features. It offers cryptographic security (see [34]) and is capable of achieving error free transmission of the watermark at the limits given by the maximum channel capacity [30]. Fundamental information theoretic limits to reliable communication have been discussed by some authors (see [30]). The shorter is the payload of a watermark, the better are the chances of it being communicated reliably.

Spread spectrum is an example of a symmetric key cryptosystem [31]. System security is based on proprietary knowledge of the keys (or pseudo random seeds) which are required to embed, extract or remove an image watermark. One provision in the use of a spread spectrum system is that it is important that the watermarking be non-invertible because only in this way can true ownership of the copyright material be resolved [8]. Ó Ruanaidh et al. [25] and Cox et al. [5] have developed perceptually adaptive transform domain methods for watermarking. In contrast to previous approaches the emphasis was on embedding the watermark in the most significant components of an image. The general approach used in these papers is to divide the image into blocks. Each block is mapped into the transform domain using either the Discrete Cosine Transform (DCT) [28], the Hadamard Transform [7] or the Daubechies Wavelet Transform [29]. Information has been embedded using the DCT [26] or FFT magnitude and phase, wavelets (see refs. of [26]), Linear Predictive Coding [19] and fractals [10]. J.-F. Delaigle at al. [11] have applied signature labelling techniques for the copyright protection of digital images.

The industrial importance of digital copyright protection has resulted in a number of products, either based on specific watermark techniques or additional registration services. They include the PictureMarc system by Digimarc, SureSign (former FBI's Fingerprint) by HighWater Signum, IP2 system by Intellectual Protocols, the Argent system by Digital Information Commodities Exchange and the Tigermark system from NEC. Further some prototypes have been developed among which are the PixelTag system by the MIT Media Lab and the SysCop system from Zhao and Koch of the Frauenhofer-Institut für Graphische Datenverarbeitung [39,40]. Many of these systems have been subject to attacks, that destroy the marking or at least reveal significant limitations of the marking techniques [2,3,17,24,27].

3 Overview

We envision the watermark system operating an open environment like the Internet with different interconnected computers. Users can be located anywhere and can sell or buy images. If legal dispute resolution for multiple watermarks

is needed the Copyright Holder (H) sends copyright information and authentic image information to the Copyright Certificate Center (C). After having received a copyright certificate from C, the copyright holder can sell his digital images, for example, via an image shopping mall, to an image buyer (B). The Public Key Infrastructure (PKI) supports the distribution of authentic public keys between all parties which are needed for mutual authentication, non-repudiation and confidentiality. The communication channels between the parties are shown in figure 1.

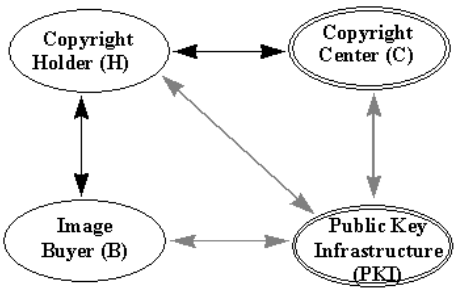

Fig. 1. Communication channels between identified parties

Our approach enables the secure generation and transmission of watermarked data using an asymmetric key pair like applied in public-key cryptography. The cover data set is watermarked, while the watermark is encoded using one or both of these keys. The resulting stego data set is then transmitted to a second party, while the same keys are used for establishing a secure transmission between the parties.
During the trading process, the involved parties use asymmetric key pairs and a key agreement protocol for establishing a symmetric key between them. The party creating the watermark can embed a *private*, a *detection* and a *public watermark* in the data set. The public watermark can be detected by third parties while the private and detection watermark can only be detected by the copyright holder.
After embedding the digital watermark into the image, the information describing it, such as the image identifier, a textual description and related information is transmitted authentically to a registration party that permanently stores a record relating to this stego data set and issues a copyright certificate which it stores and transmits to the copyright holder.
A template pattern is added to the Fourier transform of an image to be watermarked. For checking the watermark, the Fourier transform of the stego-image is calculated. ¿From this Fourier transform, the log polar mapping transform is generated, which is then searched for the modulation pattern. Using the log polar transform of the Fourier transform has the advantage that scaling and rotation of the stego-image are expressed as translations. This allows an easy search for rotation and scaling using cross-correlation techniques. The magnitude components of the Fourier transform of each image block is modulated using the same pattern in each block. This method provides robustness against

cropping of the stego-image because the magnitude spectrum is invariant to circular translations (cyclic shifts) and cropping leads to a circular translation of the watermark in each block.

4 Threat model

For the reader's convenience we describe different scenarios with typical people, such as Alice, Bob and Mallet. Alice is the copyright holder. She has taken a very valuable photograph, scanned it and would like to sell it to Bob in digital form. Mallet would also like to sell or distribute the image himself. He has a lot of computing power and he can listen and intercept/change any information that Alice and Bob transmitted on the Internet. Figure 2 shows a basic commercial scenario. We list the encountered risks for each party in table 1.

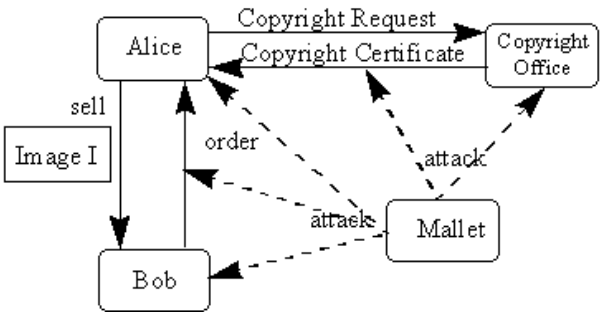

Fig. 2. Main transactions during watermarking

It is beyond the scope of the paper to discuss all possible threats in detail. We have listed them in table 1 along with the exploitation assessment. We now analyse some of the risks for Alice and Bob. Alice's main interest is to protect her cover-image, register it and receive a legally bound copyright certificate from C. She wants to have guarantee that nobody can access the cover-image and that the copyright certificate is saved permanently by C. When registering the image, Alice wants to be certain that she receives a copyright certificate for a novel image and that the legal binding is guaranteed by C. She can, therefore, successfully sue fraudulent behaviour. Bob is particularly interested to receive the images he wants and he has paid for. Mallet on the other hand is very antisocial and tries to hurt Alice or Bob wherever possible. Mallet tries to steal information and sell it himself. We discuss some of the important threats that have been identified:

(1) *Mallet buys the image but distributes it further*
Mallet has bought an image from Alice. Without Alice's approval, he sells it himself.

(2) *Mallet embeds his own watermark and registers the image himself*
Mallet buys an image from Alice and embeds himself a watermark . He then fills in his copyright information and registers the image at his C.

(3) *Mallet removes the watermark*
Mallet removes Alice's watermark, embeds his own and sells the image.

(4) *Counterfeit original attack*
In [9] it was shown that if an invertible watermark embedding function is used, an attack called the counterfeit original, is possible.

(5) *Statistical removal attack*
Mallet first embeds a large number of watermarks in the image, using the same method as Alice. He now starts to apply image transformations until all of his embedded watermarks disappear. With a calculable confidence, the watermark embedded by Alice also vanished.

(6) *Reduced colour image attack*
Alice tries to watermark a logo that consists only of very few colours. The watermarking algorithm adds more colours and they become visible in the image. Mallet only needs to apply a "de speckle" algorithm to successfully remove the watermark.

(7) *Mallet performs a known plaintext attack*
With any watermarking scheme, an attacker could apply a known plaintext attack to find out Alice's private key. Mallet could steal the cover-image and the watermark message if Alice is not cautious enough in hiding them. Then he can try to derive Alice's key from the given information. This is known as a known plaintext attack.

(8) *Mallet destroys Alice's image or copyright certificate*
Mallet wants to sell Alice's image himself. He therefore attacks Alice's system and destroys the cover-image, the private key and the copyright certificate.

(9) *Mallet steals watermarked image from Alice or Bob during transmission.*
Mallet tries to obtain the watermarked image without paying. He tries to attack through a security hole in Alice's or Bob's computer or read the image while it is transmitted.

(10) *Watermark collision*
Mallet found out that when he extracts a watermark with his key out of Alice's image he gets a meaningful watermark message. He could then claim that Alice stole his image and embedded her watermark on top of his.

5 Copyright Protection and Image Owner Authentication

Depending on the proof-level to be provided for the protection, our approach provides three increasing levels of reliability, namely: individual copyright pro-

Table 1
Identified threats and exploitation assessment

Threat	Description	Exploitation
1	Mallet buys the image and distributes it further	High
2	Mallet embeds his own watermark and registers the mage himself	High
3	Mallet removes the watermark	High
4	Counterfeit original attack	High
5	Statistical removal attack	High
6	Reduced colour image attack	High
7	Mallet performs the known plaintext attack	High
8	Mallet destroys Alice's image or copyright certificate	High
9	Mallet steals the watermarked image (from Alice or Bob) during transmission	High
10	Watermark collision	High
11	Mallet steals Alice's cover-image and/or secret key	Medium
12	Mallet intercepts and changes the image sent to Bob	High
13	The watermark was removed and Alice cannot discover fraud	High
14	Denial-of-service attacks such as Bob paid for the image but did not receive it	Low
15	C loses Alice's information	Low
16	The issued copyright certificate is not legally bound	Low
17	Mallet destroys or alters the copyright certificate	High
18	Mallet impersonates Alice and registers images at C	High
19	Alice loses her secret key	Low
20	Mallet sets up an image server in a country that does not support copyright laws and distributes "stolen" images	Low
21	Bob paid, but did not receive the correct image	Low
22	Mallet can find out which images Bob is buying	Low
23	Mallet impersonates Bob	Medium
24	Alice denies to have sent a copyright request to C	Low

tection, copyright protection with registered cryptographic keys and copyright protection with C on the basis of registered cryptographic keys. The present method is based on an image owner authentication technique, described below, which embeds and detects the Image Authentication Data (IAD) as the payload of a watermark. The applied image owner authentication technique is based on a perceptually adaptive spread spectrum technique. This technique provides reliable means of embedding robust watermarks. Such a technique will be discussed in section 6. In addition, a spread spectrum techniques is a form of symmetric cryptosystem. In order to embed or extract a watermark, it is necessary to know the exact values of the seed used to produce pseudo random sequences used to encode a watermark. The seeds are considered to be cryptographic keys for watermark generation and verification. System security is therefore based on proprietary knowledge of private keys, which provide in addition the necessary security parameters needed for a secure communication (mutual authentication, integrity, confidentiality, non-repudiation) in the trading process of digital images. Because spread spectrum signals are statistically independent (and therefore virtually orthogonal), the present method and apparatus encodes more than one watermark in an image at the same time, namely a private, a detection and a public watermark. The *detection watermark* is embedded under a fixed random seed which serves H to efficiently search for his images on the internet. He has to check only one embedding per picture to find the locations of his pictures. The *public watermark* indicates that the image is copyright material and provide information on true ownership. At the same time there is a secure *private watermark* whose secrecy depends on the private key of H. Since the public key of H is registered, H can prove that he is the only person in the possession of the adequate private key and therefore the generator of the private watermark. The system also provides the secure registration (mutual authentication, integrity, non-repudiation) of watermark encoded images (data sets) at C. Derived data of the stego image is registered at C and a signed digital copyright certificate is generated by C and transmitted to H. If an unauthorized third party has also encoded watermarks in the same image, conflicting claims in copyright disputes can be resolved, as only one of the two parties has a copyright certificate of the image containing only its watermark. The other party, who redistributed the original watermarked image, has only a certificate on the image where both watermarks are embedded and thus can be identified as the cheating party. Watermark protection with registered cryptographic keys and C based copyright protection are based on a PKI. The PKI issues on request public key certificates, e.g. X.509 certificates, containing the public key of the party, its distinguished name and a time stamp. Every certificate is signed with the PKI's private key and the trust is built on the validity of the authentic copy of the PKI's public key (we assume that the public key of the PKI is accessible, authentically distributed and verifiable by every party).

5.1 Cryptographic mechanisms

The following cryptographic mechanisms are used in the description [21]:

- $(\mathrm{SigMR}_G(x, m, \sigma),\ \mathrm{SigMR}_V(y, \sigma, m))$: A probabilistic digital signature scheme with message recovery, where (x, y) denotes the key pair of the signer, m the input data to be signed and σ the signature,
- $(\mathrm{SigAP}_G(x, m, \sigma),\ \mathrm{SigAP}_V(y, m, \sigma))$: A probabilistic digital signature scheme with appendix, where (x, y) denotes the key pair of the signer, m the input data to be signed and σ the signature,
- *AKAP*$(A, B, K_{AB}, x_A, y_A, x_B, y_B)$*:* Asymmetric key agreement protocol (e.g. [16]) with entity A's keypair (x_A, y_A), entity B's keypair (x_B, y_B), between the entities A and B. After the protocol, the two entities have agreed on a symmetric key K_{AB}.
- *OIAE(X, Y, CI, SI)*: Oblivious image owner authentication embedding algorithm with seed X, payload Y, cover image CI and res. stego image SI.
- *OIAV(X, SI, Y)*: Oblivious image owner authentication detection algorithm (cf. section 6) with seed X, stego image SI and resulting payload Y.
- h_1, h_2: collision resistant hash functions with hash value of appropriate length.

In principle, all known signature schemes and authentic key exchange protocols might be used with our protocols. Nevertheless to obtain an efficient variant, we propose to choose the Nyberg-Rueppel signature scheme with message recovery in a small mode, where both signature parameters are computed in a small subgroup of order q [23,15] and q has about 160 bit. For the siganture scheme with appendix one might choose an efficient variant of the Meta-ElGamal signature scheme [14] in small mode. As asymmetric key agreement protocol, we propose to choose a three-pass scheme with mutual implicit authentication, as proposed in [16] as class 7. The hash function h_1 should return a hash value of 64 bit and might e.g. be defined as $a :=$RIPEMD-$128(m), a = a_1||a_2, h_1(m) = a_1 \oplus a_2$. The hash function h_2 should return a hash value of 160 bit (e.g. SHA-1 or RIPEMD-160).

5.2 Individual Copyright Protection

During individual copyright protection only the copyright holder H with distinguished $name_H$ and asymmetric key pair (x_H, y_H) is involved. The following steps are applied:

(1) H retrieves the cover image CI, generates a unique image identifier $ID_I := h_1(name_H)||SN_I$, where SN_I is a serial number, and stores ID_I.
(2) *Embedding of private watermark:*

(a) H generates the stego image SI_0 applying the transformation $OIAE(h_2(\sigma), ID_I, CI, SI_0)$, where CI denotes the cover image, SI_0 denotes the resulting stego image and $SigMR_G(x_H, ID_I||(ID_I \pmod{2^{63}+1}), \sigma)$.
(b) H stores σ together with CI in a protected database.

(3) *Embedding of detection watermark:*
H generates $OIAE(h_2(x_H), SN_I||SN_I, SI_0, SI^*)$.

(4) *Embedding of public watermark:*
(a) H generates a public IAD_I := "Copyright by" $||CDSig||$ "All Rights Reserved" applying $SigMR_G(x_H, CD, CDSig)$ where $CD := Initials||year||(Initials||year) \pmod{2^{31}+1}$.
(b) H partitions IAD_I into blocks $BL_i, 1 \leq i \leq P$ of length 128 bit.
(c) H generates the stego image SI applying for every $i, 1 \leq i \leq P$, the transformation $OIAE(h_2(y_H||i||y_H), BL_i, CI_i, SI_i)$, where $CI_i := SI_{i-1}$ is the cover image (stego image from the previous iteration), $CI_1 = SI^*$ and SI_i is resulting stego image after iteration i.
The resulting stego image is $SI := SI_P$.

(5) H stores SI and might generate a signed copyright certificate applying $\text{SigAP}_G(x_H, SI, TS, SigSI)$, with SigSI as the signature of the stego image and TS a time stamp.

5.2.1 Cryptographic properties

Besides the robustness of the watermarks against various image transformations, which is discussed in section 6.4, the embedding of the private watermark offers useful cryptographic properties:

- The seed for embedding the private watermark is *probabilistic*, as it depends on the output of a probabilistic signature scheme. Thus even if the private watermark in one image is detected this doesn't allow an attacker to find immediately the private watermarks in other protected images.
- The seed for embedding is *image-dependent*. Thus an attacker who knows a valid image ID can't embed this under a different seed, as this wouldn't fit with the recovered ID from the signature with message used to generate the seed. The same is true for an attacker who knows a valid signed seed and wants to use this to embed his own image ID.
- The private watermark offers *non-repudiation*, as it can only be generated by the copyright holder, who knows the corresponding private key. This allows the proof of ownership to a judge.
- The signature σ with message recovery remains *secret* until H has to prove his ownership to a court. In this case, he doesn't have to reveal a secret to the court, which would enable it to generate valid private watermarks instead of H afterwards.
- The payload is *very short*. One could think of embedding the signed image

ID under a random seed instead of the described method. This leads to a longer message, which is not as robust as the chosen one, if we assume that the image ID consists of e.g. 12 byte. The shortest known signature scheme with message recovery [23,15] produces already a signature of length 20 byte. i.e. it is 80% longer.

5.3 Trading of digital images

Two parties are involved in the trading of digital images: the H and the image buyer B with distinguished $name_B$. Suppose (x_H, y_H) is the asymmetric key pair of H and (x_B, y_B) is the asymmetric key pair of B. Suppose H has an authentic copy of y_B and B has an authentic copy of y_H before they start any communication. The following steps are applied during the trading of digital images:

(1) H and B execute $AKAP(H, B, K_{HB}, x_H, y_H, x_B, y_B)$ for the generation of a shared symmetric session key K_{HB}.

(2) B generates the Trading Request Envelope $TRE :=< TD||SigTD >$, with $TD := \{ID_I||ExpTime||name_B||name_H\}$ and $\text{SigAP}_G(x_B, h_2(TD), SigTD)$. $ExpTime$ is the expiry time of the TRE, which avoids replay of the same envelope. B transmits TRE to H.

(3) H receives TRE and verifies TD, applying $\text{SigAP}_V(y_B, SigTD, IVR)$ where IVR denotes the intermediate verification result. If $IVR = h_2(TD)$, with $TD := ID_I||ExpTime||name_B||name_H$, then TD has been successfully verified and the next step is executed. In any other case, the processing and communication between H and B is stopped.

(4) If the verification was successful, H retrieves with ID_I the corresponding stego image SI and generates the Trading reSponse Envelope $TSE :=< TD||SigTD >$, with $TD := K_{HB}[SI]||ExpTime||name_H||name_B$ and $\text{SigAP}_G(x_H, h_2(TD), SigTD)$. H transmits TSE to B.

(5) B receives TSE and verifies TD by applying $\text{SigAP}_V(y_H, SigTD, IVR)$, where IVR denotes the intermediate verification result. If $IVR = h_2(TD)$, with $TD := K_{HB}[SI]||ExpTime||name_H||name_B$, then TD has been successfully verified.

B then deciphers $K_{HB}[SI]$ and checks IAD applying for every $i, 1 \leq i \leq P$, the following transformation: $OIAV(h_2(y_H||h_2(i)||y_H), SI, PL_i)$, where SI_i denotes the stego image and PL_i the detected payload of the i-th public watermark. (If P is not known, the procedure is iterated until no more public watermarks can be detected).

IAD_I is then generated by concatenating PL_i, i.e. $IAD_I := PL_1||\ldots||PL_N, 1 \leq i \leq P$. IAD_I should be of the format "Copyright by" $||CDSig||$ "All Rights Reserved". $CDSig$ is then verified applying $\text{SigMR}_V(y_H, CDSig, OD)$, with OD as the output data. If OD is

$Initials||year||(Initials||year) \pmod{2^{31}+1}$, B has verified H as the copyright holder.

Remark
In the case of a legal copyright dispute, H can retrieve IAD_I and construct the corresponding unique image ID_I. Since the generation of the same asymmetric key pair by two distinguished entities is very unlikely, the construction of the unique image ID_I provides a high level of proof against copyright infringement. In the case of watermark protection with registered keys, the generation of the same asymmetric key pair by two distinguished entities can be prevented.

5.4 Copyright Protection with Registered Keys

The copyright protection with registered cryptographic keys needs three parties, namely H with $name_H$, B with $name_B$ and the PKI with $name_I$. Suppose (x_H, y_H), (x_B, y_B), (x_I, y_I) are the unique key pairs of H, B, and I respectively. Suppose, H has an authentic and actual copy of $Cert_B$ which signature was verified with the authentic copy of y_I and B has an authentic and actual copy of $Cert_H$ which signature was verified with the authentic copy of y_I. Then the same steps as for the individual watermark protection are applied.

Remark:
Since the generated asymmetric key pairs are unique, H can be uniquely identified if no additional watermarks by unauthorized persons have been encoded into a given SI. The O based watermark protection described below provides the necessary countermeasures to prevent this threat.

5.5 Copyright Protection based on a Copyright Center

The system for copyright protection based on a Copyright Center has four participants, namely H with $name_H$, B with $name_B$, the PKI with $name_I$ and C with $name_C$. Suppose (x_H, y_H), (x_B, y_B), (x_I, y_I) and (x_C, y_C) are the unique key pairs of H, B, I and C, respectively. H has an authentic copy of $Cert_B$ and $Cert_C$ whose signatures were verified with the authentic copy of y_I, B has an authentic copy of $Cert_H$ and $Cert_C$ whose signatures were verified with the authentic copy of y_I and C has an authentic copy of $Cert_H$ and $Cert_B$ whose signatures were verified with the authentic copy of y_I. The following steps are applied:

(1) Steps 1.-4. of the individual watermark protection are applied, where the unique image ID_I is computed as $ID_I := h_1(name_H||name_C)||SN_I$.

(2) H generates a thumbnail $Thumb$ from SI and stores it together with the stego image SI.

(3) H and C execute the following steps for the secure registration and the generation of copyright certificates:

   (a) H generates the Copyright Request Data $CRD := \{h_2(SI)|| h_2(Thumb)|| ID_I\}$ and the Copyright Request Envelope $CRE :=< TD||SigTD >$, with $TD := \{CRD||ExpTime||name_H||name_C\}$ and $\text{SigAP}_G(x_H, h_2(TD), SigTD)$. H transmits CRE to C.

   (b) C receives CRE and verifies it. For this C requests the certificate $Cert_H$ that belongs to $name_H$ in CRE and obtains the authentic public key y_H. C checks the signature on TD by applying $\text{SigAP}_V(y_H, SigTD, h_2(TD))$, where $TD = CRD||ExpTime||name_H||name_C$. Then C checks the semantical correctness of TD. If all verifications are passed, then CRE has been successfully verified and the next step is executed.

   (c) C generates the digital *Copyright Certificate* by executing $\text{SigAP}_G(x_C, CCD, SigCCD)$, where $CCD := \{name_C||name_H|| SN|| ID_I||h_2(SI)||h_2(Thumb)||UCCN\}$ and $UCCN :=$"(Copyright'' $||year||name_H||$ "All rights reserved''. C stores the copyright certificate $CC := CCD||SigCCD$ together with $h_2(Thumb)$ in its database, generates the Copyright Certificate Envelope $CCE :=< TD||SigTD >$, with $TD := \{CC||ExpTime||name_C||name_H\}$ and $\text{SigAP}_G(x_C, TD, SigTD)$ and transmits it to H.

   (d) H receives CCE and verifies it by requesting the certificate $Cert_C$ that belongs to $name_C$ in CCE. H obtains the authentic public key y_C which he uses to check $SigTD$ on TD by applying $\text{SigAP}_V(y_C, SigTD, h_2(TD))$, where $TD := \{CC||ExpTime||name_C||name_H\}$. H checks the semantical correctness of TD and if all verifications are correct then H stores CC in its database.

(4) H and B might execute the image trading protocol described in section 5.3

Remark

B may check the copyright certificate requesting C to transfer an authentic copy of the copyright certificate for a given image identifier ID_I. Except the data transfer, the applied protocol is similar to the one described above. If B would like to transfer a specific copyright of a CI to another legal party, he may initiate a copyright revocation request with C. The different phases of this request are analogue to the copyright request.

5.6 Watermark detection

The system for private watermark detection is executed by a single process, the H with distinguished name $name_H$. H must be able to detect if an image GI in a list of images is protected with its copyright. Therefore he has to check, if its detection watermark is embedded in GI.

(1) H applies to GI the following transformation: $OIAV(h_1(x_H), GI, PL)$ and checks if $PL = SN||SN$.
(2) If PL is of suitable format H looks in its database for the corresponding $CC = CCD||SigCCD$, where $CCD = \{name_C||name_H||SN||ID_I||h(SI)||UCCN\}||SigCCD$, and SN appears in ID_I. There it finds the corresponding physical address of SI. If no entry in the database exists, H read a random string PL, which happens only with probability 2^{-32}.
(3) H gets the stego image SI.
(4) H checks if its private watermark is embedded in SI.

5.7 Watermark verification

In the case of a dispute, a judge must be able to check private watermarks in stego-images. The system for private watermark verification is partitioned into two processes, namely H with distinguished $name_H$ and the judge with distinguished $name_J$.

(1) H forwards $name_H, SI, SigCCD$ and $Sig(ID_I)$ to J.
(2) J verifies $CC = \{name_C||name_H||SN||ID_I||h_2(SI)||UCCN\}||SigCCD$ by the following steps:
    (a) J checks the signature $SigCCD$ by $SigAP_V(y_0, SigCCD, CCD)$.
    (b) J checks $name_C$, $name_H$, ID_I, if $h_2(SI)$ corresponds to SI and if if UCCN contains $name_H$.
(3) J verifies the public key y_H of H by requesting the certificate $Cert_h$ that corresponds to $name_H$.
(4) J checks that $Sig(ID_I)$ was issued by H with distinguished $name_H$.
(5) J checks the appearance of ID_I in the image applying the transformation $OIAV(h_2(Sig(ID_I)), SI, PL)$, where PL denotes the detected private watermark. If $PL = ID_I$ then J has checked that SI contained a private watermark of H. This proves together with $SigCCD$ that H is the legal owner of SI.

6 Oblivious Image Owner Authentication

The watermarking technique comprises the following components:

(1) An error-control coding technique for the payload to be transmitted in the watermark.
(2) An encoding technique to encode the resulting message.
(3) A reliable method for embedding the encoded message in the image without introducing visible artefacts.

Components 1 and 2 apply to embed watermarks in any type of data while component 3 is specific to the embedding of watermarks in images.

6.1 Error control coding

Error control coding is applied to the message prior to encoding in order to include redundancy, which makes the scheme more robust. Symbol based Reed Solomon (RS) codes are applied for this purpose. The advantages are:

- RS codes are capable of correcting symbol errors rather than bit errors and
- RS decoders can correct erasures as well as errors.

The erasure correcting capability is particularly useful for channels, which provide reliability information on the received symbols. Symbols with low reliability are treated as erased symbols. This approach is efficient in terms of error correction since one error can be traded against two erasures in the decoding process.

RS codes have fixed lengths, which depend on the size of the symbol alphabet. For 8-bit symbols, RS codes have a length of 255 and they are convenient for byte oriented data. In watermarking one wishes to use codes of shorter lengths than 255. Such codes are easily obtained from the original RS codes by puncturing or shortening (cf. [4]).

6.2 Encoding the message

During encoding, the message to be transmitted in the watermark is transformed into a form suitable for use in the modulation of image components. At the same time, it is encrypted using a suitable key.

One can easily combine spread spectrum based watermarking with the cryptographic key distribution techniques described earlier. The encoding procedure

has access to the cryptographic keys x_H and y_H (or their hash values), which are used to generate the seeds for the pseudo-random sequences as described below. Knowledge of the corresponding key is required for recovering the message from the watermark.

A watermark can be embedded or extracted by the key owner using the same key. ¿From the point of view of embedding watermarks in images given the cryptographic keys the sequences themselves can be generated. A good spread spectrum sequence is one which combines desirable statistical properties such as uniformly low cross correlation with cryptographic security.

Suppose we are given a message M (for example, that was provided with error coding). The message has the binary form $b_1 b_2 \ldots b_L$, where b_i are its bits. This can be written in the form of a set of symbols $s_1 s_2 \ldots s_M$ – most generally by a change in a number base from 2 to B. The next stage is to encode each symbol s_i in the form of a pseudo random vector of length N, wherein each element of this vector either takes the value 0 or 1. N is, for example, in the order of 1000 to 10000 (typically in the order of 10% of the total number of Fourier components that can, theoretically, be modulated).

To encode the first symbol a pseudo random sequence v of length $N + B - 1$ is generated. To encode a symbol of values where $0 < s < B$ the elements $v_s, v_{s+1} \ldots v_{s+N-1}$ are extracted as a vector r_1 of length N. For the next symbol another independent pseudo random sequence is generated and the symbol encoded as a random vector r_2. Each successive symbol is encoded in the same way. Alternatively, one can use any the N cyclic shifts of the pseudo random sequence v. Note that even if the same symbol occurs in different positions in the sequence, no collision is possible because the random sequences used to encode them are different — in fact they are statistically independent. Finally the entire sequence of symbols is encoded as the summation: $m = \sum_{i=1..M} r_i$

The pseudo-random vector m has N elements, each varying between 0 and M. In a next step, the elements of m are offset to make their mean zero. When decoding the watermark, a vector m' (read-out message) is derived from the stego-image. In oblivious watermarking, m' corresponds to the modulated Fourier coefficients. Due to distortions suffered by the stego-image due to image processing, in general m' will not be equal to but will be "statistically similar" to m. To decode s from m', the elements of m' are first offset to make their mean zero. Then, starting from the (known) seed, the first random sequence v of length $N + B - 1$ is generated and the correlation of v with m' is calculated. The peak of the correlation indicates the offset s_1 in the random sequence that was used for generating r_1. Then, the next random sequence v is generated and cross-correlated with m' to retrieve s_2, etc. Reliable communications of the system are best accommodated by using $m-$sequences that possess minimum cross correlation with each other. This is the same as

maximizing the Euclidean distance between vectors $v_1, v_2, v_3 \ldots$. If M is sufficiently large, the statistical distribution of the message m should approach a Gaussian distribution (according to the Central Limit Theorem). A Gaussian distributed watermark has the advantage that it is somewhat more difficult to detect. The variance increases with order $M^{1/2}$; in other words, the expected peak excursion of the sequence is only order $M^{1/2}$.

6.3 Oblivious Image Authentication Algorithm

In this section, we describe how to embed the encoded message m in the image in the form of a watermark. The method is designed for robustness to operations generally applied to images such as translation, cropping, rotating and scaling. (The method is not designed for other types of data such as sound or text.) In order to achieve robustness against circular translation, each image block is first subjected to a Fourier transform and the same watermark is embedded in each block so the watermark tiles the entire image. Then, message m modulates the Fourier components. In addition to this, a template is embedded in the image, which can be used for detecting a rotation and scaling of the image when reading the watermark. Given a cover image the steps for embedding a watermark in the image are as follows:

(1) If the image is a colour image, then compute the luminance component (for example, by simply replacing each pixel by $g/2 + r/3 + b/6$, where g, r and b are its green, red and blue components) and use these values for the following calculations.
(2) Divide the image into adjacent blocks of size 128 $\times$128 pixels.
(3) Map the image luminance levels (or grey levels for a black and white image) to a perceptually "flat" domain by replacing them with their logarithm. The logarithm is a good choice because it corresponds to the Weber-Fechner law which describes the response of the human visual system to changes of luminance. This step ensures that the intensity of the watermark is diminished in the darker regions of the image where it would otherwise be visible.
(4) Compute the FFT (Fast Fourier Transform) of each block. From the real and imaginary components obtained in this way, calculate the corresponding magnitude and phase components. The magnitude components are translation invariant and will therefore be used in the following modulation steps. (However, it is possible to derive translation invariants from the phase spectrum as well, which could also be modulated).
(5) Select the magnitude components to be modulated. To encode a message m of length N, a total number of N components are modulated. In non-oblivious watermarking, any components can be modulated. For oblivious watermarking, because of the interference of the cover-image

with the watermark, the largest magnitude components are avoided and only a band of frequencies are used. These mid band frequencies are chosen because they generally give a good compromise between robustness and visibility of the watermark. There are two methods for selecting the components to be modulated:

- The selection of the components to be modulated does not depend on the given image. Rather, the same components are selected for every image. The author as well as the reader of the watermark know the positions of the components to be selected in advance.
- The largest components (inside the allowable frequency range) are used for modulation using a perceptually adaptive approach.

(6) Add a template by a second modulation of the magnitude components. This is described in more detail below.

(7) Compute the inverse FFT using the phase components and the modulated magnitude components.

(8) Compute the inverse of the perceptual mapping function of step 3. For Weber-Fechner law mapping, the inverse function is an exponential.

(9) Replace each watermarked block in the image to obtain the stego-image.

(10) If the image is a colour image, then re-scale the red, green and blue components by the relative change in luminance introduced by embedding a watermark. Typically, the red, green and blue pixels occupy a byte each in program memory. If overflow or underflow occurs then the pixel is set to the upper bound 255 or lower bound 0 respectively.

When selecting the components to be modulated, care must be taken to preserve the symmetry imposed on the Fourier components $F(k_1, k_2)$ by the fact that the image block is real valued: $F(k_1, k_2) = F^*(N_1 - k_1, N_2 - k_2)$, where N_1, N_2 designate the size of the image block. Once the magnitude components $(M_1, ...M_N)$ to be modulated are chosen, the corresponding value m_i of message m is added to or subtracted from the corresponding selected magnitude component M_i. Addition is applied, if the difference between the corresponding phase component P_i and a "reference phase" is between 0 and π. Subtraction is applied if the difference is between π and 2π. Note that the reference phase for each watermark component should be the same for each block. This provides robustness against translation and cropping (see below). Before adding/subtracting the values m_i to/from M_i, the vector m can be scaled to adjust the magnitude of its elements to those of the components M_i. Generally, the elements m_i should be in the same order of magnitude as the components M_i. The depth of modulation or amplitude of the embedded signal should depend on the objective measure of the perceptual significance. The lower the perceptual significance, the higher should be the amplitude of the watermark. However, for simplicity, the amplitude for all components is usually kept constant.

6.3.1 Template

As mentioned above, a template is added to the image in step 6. The following steps have to be executed:

(1) Apply a log-polar map to the magnitude components, i.e. transform them into a polar coordinate system $(\Theta, log - r)$ with an angular and a logarithmic radius axis respectively. In this representation, a scaling of the image leads to an offset of the components along the $log - r$ axis. A rotation of the image leads to an offset along the Θ axis. Preferably, low pass filtering is used for interpolating the frequency space components during this mapping. The magnitude components belonging to very low or high frequencies are not mapped. The following modulation is only applied to components in midband frequency range.
(2) Select the magnitude components in the log-polar coordinate system to be modulated. Typically, as few as 10 components can be modulated. The pattern T formed by the selected components in log polar space should be such that its auto-correlation under translation is weak. There should be as little ambiguity as possible in matching the shifted stego-image with the template. For this purpose, the indices of the selected components should be coprime or at least be derived from a two-dimensional random sequence.
(3) Map the modulated points by a change of coordinates back into frequency space. The pattern T formed by the selected components in log polar space is predefined and known to the reader of the watermark. It must be noted that the calculation of the log-polar transform is not required for embedding the template. The pattern T of the components to be modulated in log-polar space can be mapped back to the corresponding components in frequency space.

As will be explained below, the template is not required for non-oblivious watermarking (since the original image can be used as the "template", in other words, the stego-image can be registered with the original image). Also if the image is a colour image then the luminance image is used during the following operations:

(1) Divide the image into adjacent blocks of size 128×128.
(2) Map the image luminance levels (or gray levels) to the perceptually "flat" domain by replacing them with their logarithm.
(3) For each block compute the FFT.
(4) Determine the rotation and scaling that the image suffered by finding the template in log-polar space and compensate for the rotation and scale.
(5) Read the modulated components to generate message m'.
(6) Once that the message m' is recovered, it is demodulated and error corrected using the methods described earlier.

Finding the template
The steps for finding the template are as follows:

(1) Apply a log-polar mapping to the magnitude components of the Fourier transform. The magnitude components in the very low or high frequency range are not mapped. The log-polar mapping is only applied to components in midband frequency range.
(2) For oblivious watermarking, calculate the normalized cross correlation of the components in log-polar space with the template pattern T that was used for generating the template in step 6 and find the point of best correlation. If the image has neither been rotated or scaled, this point is at the origin. Scaling leads to a corresponding offset along the $log - r$ axis, rotation to a corresponding offset along the Θ axis.
(3) For non-oblivious watermarking, the log polar transform of the Fourier components of the cover-image can be used instead of template pattern T for retrieving scaling and rotation. The cross correlation can be calculated efficiently using conventional Fourier techniques.

6.4 Properties of the watermark

In the following, some of the properties of the watermark generated using the steps described above are discussed. We first review some of the fundamental properties of the Fourier transform which lead to robustness against cropping, scaling and translation and rotation.

6.4.1 Definition

Let the image be a real valued continuous function $f(x_1, x_2)$ defined on an integer-valued Cartesian grid $0 \leq x_1 < N_1, 0 \leq x_2 < N_2$.

The Discrete Fourier Transform (DFT) is defined as follows:

$$F(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(x_1, x_2) e^{-j2\pi x_1 k_1/N_1 - j2\pi x_2 k_2/N_2} \tag{1}$$

The inverse transform is

$$f(x_1, x_2) = \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} F(k_1, k_2) e^{j2\pi k_1 x_1/N_1 + j2\pi k_2 x_2/N_2} \tag{2}$$

The DFT of a real image is generally complex valued. This leads to magnitude

and phase representation for the image:

$$A(k_1, k_2) = [F(k_1, k_2)] \tag{3}$$

$$\Phi(k_1, k_2) = \angle F(k_1, k_2) \tag{4}$$

6.4.2 General Properties of the Fourier Transform

It is extremely instructive to study the effect of an arbitary linear transform on the spectrum of an image. From this study we will conclude that it is possible to undo the effect of any linear transformation on an image, even if the image is cropped.

Once $N_1 = N_2$ (i.e. square blocks) the kernel of the DFT contains a term of the form:

$$x_1 k_1 + x_2 k_2 = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \tag{5}$$

If we compute a linear transform on the spatial coordinates:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \mathbf{T} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{6}$$

then one can see that the value of the DFT will not change [2] if:

$$\begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \rightarrow \left(\mathbf{T}^{-1}\right)^T \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \tag{7}$$

6.4.3 FFT: Rotation

Consider a rotation matrix

$$\mathbf{T} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{8}$$

[2] The DFT will be invariant except for a scaling factor which depends on the Jacobian of Transformation, namely the determinant of the transformation matrix T.

Therefore,

$$\left(\mathbf{T}^{-1}\right)^T = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{9}$$

Rotating the image through an angle θ in the spatial domain causes the Fourier representation to be rotated through the same angle.

$$\begin{aligned} &F(k_1\cos\theta - k_2\sin\theta, k_1\sin\theta + k_2\cos\theta) \\ &\quad \leftrightarrow f(x_1\cos\theta - x_2\sin\theta, x_1\sin\theta + x_2\cos\theta) \end{aligned} \tag{10}$$

Note that the grid is rotated so the value of the image at the new grid points may not be defined. The value of the image at the nearest valid grid point can be estimated by interpolation.

6.4.4 FFT: Scale

Consider a scaling matrix

$$\mathbf{T} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{11}$$

Therefore,

$$\left(\mathbf{T}^{-1}\right)^T = \begin{bmatrix} \frac{1}{\lambda_1} & 0 \\ 0 & \frac{1}{\lambda_2} \end{bmatrix} \tag{12}$$

Hence, scaling the axes in the spatial domain causes an inverse scaling in the frequency domain.

$$\frac{1}{\rho}F(\frac{k_1}{\rho}, \frac{k_2}{\rho}) \leftrightarrow f(\rho x_1, \rho x_2) \tag{13}$$

6.4.5 FFT: Translation

Shifts in the spatial domain cause a linear shift in the phase component.

$$F(k_1, k_2)\exp\left[-j(ak_1 + bk_2)\right] \leftrightarrow f(x_1 + a, x_2 + b) \tag{14}$$

Note that both $F(k_1, k_2)$ and its dual $f(x_1, x_2)$ are periodic functions so it is implicitly assumed that translations cause the image to be "wrapped around". We shall refer to this as a *circular translation* or a cyclic shift. From property 14 of the Fourier transform it is clear that spatial shifts affect only the phase representation of an image. This leads to the well known result that the magnitude of the Fourier transform is a circular translation invariant.

6.4.6 Robustness to cropping

One feature of translation invariants developed using the Fourier transform is that they are invariant to circular translations (or cyclic shifts). This is used to construct watermarks that are invariant to cropping. As mentioned above, the image is split into blocks and the watermark is applied to each block. In other words, the same modulation pattern is applied to the Fourier components of each block, wherein the modulation pattern is given by the corresponding encoded messages m. Suppose that the watermark in a standard size block will be of the form: $T = [AB; CD]$ where the submatrices A, B, C and D are of arbitrary size. A circular translation of such a watermark is of the form: $S = [DC; BA]$. The original stego-image is tiled with watermarks in the pattern $[TTTT; TTTT; TTTT]$. A little thought demonstrates that a cropped section of the matrix will carry a watermark in the form $[SSSS; SSSS; SSSS]$. When reading the watermark of the cropped image, each block carries the watermark S. Since S is a circular transform of T, it can be read directly in the Fourier domain using the steps outlined above. Note, however, that the cover-image is not tiled, only the watermark is. Therefore, while cropping merely induces a circular translation of the watermark in each block, the change of image in each block is not a circular translation.

The optimum size of block depends on a number of different factors. A size that is a power of two is useful because the FFT can be used. The block size also must be small enough to withstand cropping but large enough to comfortably contain a watermark. Heuristically, the best compromise for block size is 128.

6.4.7 Robustness to scaling and rotation

As mentioned earlier, reading the template in log-polar space allows to detect and measure any scaling and/or rotation that was applied to the image. This information can then be used for reading the watermark. Since the reader knows the pattern that was used for modulating the magnitude components, he can identify the modulated components in the scaled and rotated image and extract the message m'. Note that the apparatus does not explicitly use a rotation and scale invariant watermark but instead searches the parameter space of rotations and scales. Since searching the space of rotation and scales

in the frequency or space domain is quite complicated, the log-polar map is used to map these parameters to Cartesian coordinates where searching can be carried out using efficient correlation techniques.

6.4.8 Robustness to translations

By translation, we mean zero padding of the image such as would occur if an image were placed on a scanner and scanned. In this case the effect on the watermark blocks may be understood in terms of simple signal and image processing theory. Effectively, zero padding is a multiplication by a rectangular window function in the spatial domain. In the frequency domain this approximates to a convolution with a cardinal sine function. Generally, this blurring of frequency space is not severe. If more than about one third of an watermark block is present the watermark can still be decoded.

6.4.9 Robustness to compression

The watermark is also resistant to compression. It is well known that transform based image compression techniques favour low frequencies in the sense that low frequency content in the original image is better preserved in the compressed image. For this reason, it would seem that a low frequency watermark would be better. However, as mentioned earlier in oblivious watermarking it is necessary to avoid low frequencies for embedding information because the image interferes with the watermark at those frequencies. Fortunately, a compromise is possible: judicious selection of a band of frequencies leads to a watermark that is both oblivious and is sufficiently resistant to lossy image compression. One helpful factor is that there are relatively few low frequency components in which to embed a spread spectrum signal. Using midband frequencies actually improves the robustness of the mark because of the increased redundancy of the encoding of the payload.

6.4.10 Robustness of the template pattern

The template pattern is essential to determine the rotation and scaling of the image. Thus its robustness is highly important. There is a good case for arguing that the template is somewhat more robust than the watermark. The reason for this is, that the template actually has to carry less information than the watermark. For example, suppose it is only possible to recover the watermark if the rotation angle and scaling factor recovered using the template are both 99.9% accurate. To give 99.9% accuracy one needs $\log_2(1000) \approx 9$ bit. Specifying both the angle and the scaling factor therefore needs around 18 bit which is considerably less than the amount of information contained in a typical watermark (about 128 bit).

6.4.11 Redundancy

The watermark is embedded in blocks of a fixed size with exactly the same watermark embedded in each block. This means that the watermark can be recovered from a single block only. However the chance of extracting the watermark correctly increases if one has more than one block. Therefore, the reliability of watermark extraction increases with the size of image, as one might expect.

7 Implementation

To demonstrate the feasibility of the approach, a Java/C++ based copyright protection and authentication environment for digital images has been implemented. The PKI, H, C and B application processes all implement a Graphical User Interface and a server, supporting both console users and other requests through a socket interface.

The communication between the application and the dispatching watermarking proxy process is achieved through a socket interface, giving us a good control over communication security and application location. Flexibility is just one advantage of this technique: the database, watermarking engine and GUI can run on different and distant machines. For the first version of our implementation, we have assumed that all parts of the application are run on the same secure machine. It is, therefore, not necessary to protect the privacy of the connections used by the application internally.

The copyright holder can load images, enter the copyright information, contact C, embed the watermark into the cover-image, save all data encrypted in a database and sell the digital image. Figure 3 illustrates an example of the loaded cover-image together with the copyright information.

We have seen previously that C is important to solve many of the possible attacks of fraudulent people. The C is even indispensable for having legally bound copyright certificates. Today's copyright laws require that C is in possession of a copy of the cover-image. We can therefore list the following necessary functionality's of C application:

- Legally embedded institution backed by the government
- Provide a server to answer incoming copyright requests
- Verify consistency of all received copyright requests
- Save all information persistently and securely, since the data has to be recovered after any restart of the server.

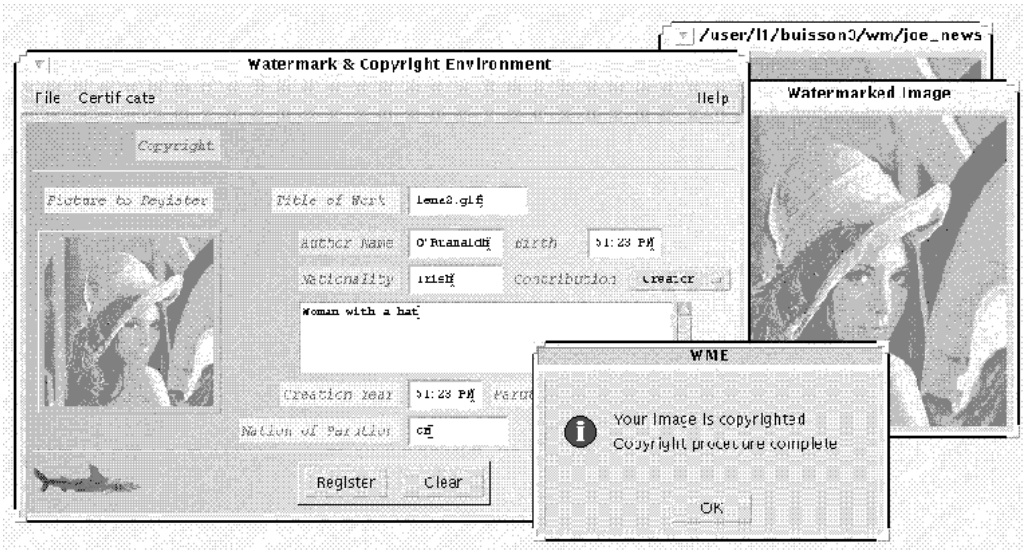

Fig. 3. The Copyright Holder Application

Because all transactions are executed electronically, it is obvious that all the requests must be authenticated, secure (integrity, confidentiality) and transactional. The legal binding further requires non-repudiation. In particular, the persistence property is necessary in order to leave the copyright certificate valid, even if the legal dispute is 20 years later. The data integrity is needed so nobody could claim that the copyright certificate was forged by C. We can note that the trust that H puts into C is substantial. Therefore C should be backed by some public authority such as the chamber of commerce.

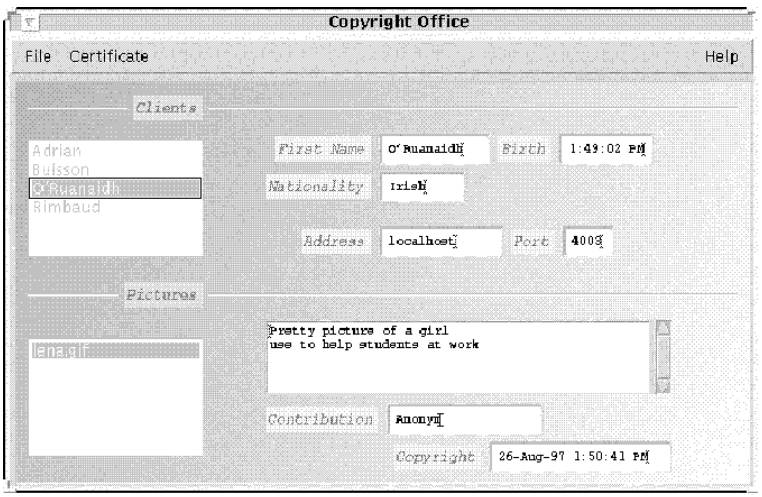

Fig. 4. The Copyright Certificate Center Server Application

Figure 4 shows a screen shot of C GUI. We can note on the console window that the application has created C server on port 4001 and is waiting for incoming copyright requests. The user can list all the copyrights issued. The PKI, implemented as a Trusted Third Party (TTP) Server, issues all certificates for all users. On the basis of the authenticly distributed public key from the TTP, a public/private key-pair along with its associated certificate may be

generated and then passed back as ciphered information to the client. These operations are based on a specific request from H. The request includes a password for online key generation, key distribution and certificate handling. The request is protected on the basis of the TTP's public key which was authentically distributed before. The buyer application can contact a copyright holder image server residing per default at port 4003. The buyer can enter the name of the desired image and if the image was already registered at C, H image server sends back the image.

8 Conclusion

We have presented a new approach for the copyright protection of digital images. This approach is based on asymmetric cryptographic protocols and techniques. In contrast to any other scheme, the combination of a spread spectrum based technique in conjunction with an asymmetric cryptographic technique allows the construction of a one-way function, since only H is able to verify the private watermark. In addition, he may prove that he has the adequate key by verifying the signature of the seed for the payload data. Even if the different phases of the approach are known in the public, the security of our approach is not compromised. Compared to other approaches, the following new properties have been identified:

(1) Different security services for the communication, such as *mutual authentication, integrity, confidentiality* and *non-repudiation* are supported along with the protection against copyright infringement by the system with one asymmetric cryptographic key pair.
(2) The present technique enables a strong binding relation between the image ID, the image and H if H registers his copyright at C. If an image is watermarked later by an unauthorized person, the time stamp in the copyright certificates resolves the copyright ownership.
(3) H does not have to reveal his private cryptographic key if ownership verification has to be applied by a different legal party.
(4) The present technique supports the *transferral of copyrights*. If a copyright is transferred to another legal party, corresponding copyright revocation certificates may be generated.
(5) Digital signatures are used for the security of the communication between different parties and for the authenticity of the data embedded in a public watermark of an image. No signature labelling techniques of the complete image are applied by the system.
(6) Circular translation invariants are used as a means of constructing digital watermarks that are invariant to cropping.
(7) In contrast to some known techniques, the system does not require a database of watermarks since only the keys are required to embed or

extract a watermark.

(8) Information is retrieved from the log polar domain of the Fourier transform. Frequency components are modulated which are oblivious to the cover-image but which also have the property that they form an unambiguous non-repeated pattern in log-polar space. They are used for determining the degree of rotation and scaling suffered by a stego-image in the absence of the cover-image. Co-prime frequencies are useful for generating such a pattern or template. Uniform random sampling of log polar space is another method that can be applied.

The approach presented for the copyright protection of digital images can also be extended to other data such as video, audio, and binary data. We are actually investigating new spread spectrum techniques for the watermarking of audio and binary data.

Acknowledgements

We wish to thank our colleague Thomas Mittelholzer for helpful comments on error control coding.

References

[1] G.M. Acken, "How watermarking adds value to digital contents'', Communications of the ACM, July 1998, Vol.41, No.7, pp. 75 - 77.

[2] R.J. Anderson, "Stretching the limits of steganography'', Workshop on Information Hiding, LNCS 1174, Springer, 1996, pp. 39-48.

[3] R.J. Anderson and F.A. Petitcolas, "On the limits of steganography'', IEEE Journal of Selected Areas in Communications, Special Issue on Copyright & Privacy Protection, 1998, to appear.

[4] R.E. Blahut, "Theory and Practice of Error Control Codes'', Chap. 3.6, Addison-Wesley, 1984.

[5] C. Cox, J. Killian, T. Leighton and T. Shamoon, "Secure spread spectrum communication for multimedia'', Technical report, N.E.C. Research Institute, 1995.

[6] G. Caronni "Assuring Ownership Rights for Digital Images", Reliable IT Systems, VIS '95, Vieweg, Germany, 1995, pp. 251 - 265.

[7] W.G. Chambers, "Basics of Communications and Coding", Oxford Science Publications, Clarendon Press Oxford, 1985.

[8] S. Craver, N. Memon, B. Yeo and M. Yeung, "Can invisible marks resolve rightful ownerships ?'', IS&T/SPIE Electronic Imaging '97 : "Storage and Retrieval of Image and Video Databases'', 1997, pp. 310 – 321.

[9] S. Craver, N. Memon, B. Yeo and M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks and implications'', IEEE Journal of Selected Areas in Communications, 1998, to appear.

[10] P. Davern and M. Scott, "Fractal based image steganography'', Proceedings International Workshop in Information Hiding, LNCS 1173, Springer, 1996, pp. 279 – 294.

[11] J.F. Delaigle, J.M. Boucqueau, J.J. Quisquater and B. Macq, "Digital Images protection techniques in a broadcast framework: An overview'', Laboratoire de Télécommunications et de Télédéction, Université Catholique de Louvain, 1996.

[12] A. Freier, P. Karlton and P. Kocher, "SSL Version 3.0'', Netscape Communications, Version 3.0, November 1996.

[13] F. Goffin, J.F. Delaigle, C. De Vleeschouwer, B. Macq and J.-J. Quisquater, "A low cost perceptive digital picture watermarking method'', IS&T/SPIE Electronic Imaging '97 : "Storage and Retrieval of Image and Video Databases'', 1997, pp. 264 – 277.

[14] P. Horster, H. Petersen, M. Michels, "Meta-ElGamal signature schemes'', Proc. 2. ACM conference on Computer and Communications security, ACM Press, November, 1994, pp. 96 – 107.

[15] P. Horster, M. Michels, H. Petersen, "Meta-Message recovery and Meta-blind signature schemes based on the discrete logarithm problem and their applications'', LNCS 917, Advances in Cryptology: Proc. Asiacrypt '94, Springer, 1995, pp. 224 – 237.

[16] ISO/IEC 11770-3, "Information technology - Security techniques - Key management, Part 3: Mechanisms using asymmetric techniques'', 1996.

[17] M.G. Kuhn, "StirMark'', http://www.cl.cam.ac.uk/~fapp2/watermarking /image_water marking/stirmark/, November 1997.

[18] G.C. Langelaar, J. van der Lubbe and J. Biemond, "Copy protection for multimedia data based on labelling techniques'', 17th Symposium on Information Theory in the Benelux, May 1996.

[19] K. Matsui and K. Tanaka, "Video-Steganography : How to secretly embed a signature in a picture'', IMA Intellectual Property Project Proceedings, January, 1994, pp. 187 – 206.

[20] N. Memon, P.W. Wong, "Protecting Digital Media Content'', Communications of the ACM, July 1998, Vol.41, No.7, pp. 35 - 43.

[21] A.J. Menezes, P.C. Van Oorschot and S.A. Vanstone, "Handbook of Applied Cryptography'', CRC Press, 1996.

[22] F. Mintzer, G.W. Braudaway, A.E. Bell, "Opportunities for Watermarking Standards'', Communications of the ACM, July 1998, Vol.41, No.7, pp. 57 - 64.

[23] K. Nyberg, R. Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem'', LNCS 950, Advances in Cryptology: Proc. Eurocrypt '94, Springer, 1994, pp. 182 – 193.

[24] F.A. Petitcolas, "Weakness of existing watermarking schemes'', http://www.cl.cam.ac.uk/~fapp2/watermarking/image_watermarking/, October 1997.

[25] J.K. Ó Ruanaidh, W.J. Dowling and F.M. Boland, "Phase watermarking of images'', IEEE International Conference on Image Processing, September, 1996.

[26] J.K. Ó Ruanaidh, W.J. Dowling and F.M. Boland, "Watermarking digital images for copyright protection'', IEEE Proceedings on Vision, Image and Signal Processing, Vol. 143, No. 4, August 1996, pp. 250 – 256.

[27] F.A. Petitcolas, R.J. Anderson and M.G. Kuhn, "Attacks on Copyright Marking Systems'', Proc. Workshop on Information Hiding, Springer, 1998.

[28] W.B. Pennebaker and J.L. Mitchell, "JPEG Still Image Compression Standard'', Van Nostrand Reinhold, New York, 1993.

[29] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, "Numerical Recipes in C'', Cambridge University Press, second edition, 1992.

[30] J. Smith and B. Comiskey, "Modulation and information hiding in images'', in Ross Anderson, editor, Proceedings of the First International Workshop in Information Hiding, LNCS 1173, Springer, 1996, pp. 207 - 226.

[31] B. Schneier, "Applied Cryptography'', Wiley, 2nd edition, 1995.

[32] M.D. Swanson, B. Zhu and A.H. Tewfik. Robust data hiding for images. In 7th Digital Signal Processing Workshop (DSP 96), pp. 37 – 40.

[33] M.D. Swanson, B. Zhu and A.H. Tewfik, "Transparent robust image watermarking'', International Conference on Image Processing, volume III, 1996, pp. 211 – 214.

[34] A.Z. Tirkel, G.A. Rankin, R.G. van Schyndel, W.J. Ho, N.R.A. Mee and C.F. Osborne, "Electronic watermark'', Dicta-93, December 1993, pp. 666 - 672.

[35] A.Z. Tirkel, R.G. van Schyndel and C.F. Osborne, "a two-dimensional digital watermark'', Proc. ACCV'95, 1995, pp. 378-383.

[36] R.B. Wolfgang and E.J. Delp, "A watermark for digital images'', International Conference on Images Processing, September 1996, pp. 219 - 222.

[37] R.B. Wolfgang and E.J. Delp, "A watermarking technique for digital imagery: further studies'', International Conference on Imaging, Systems and Technology, 1997, pp. 279 – 287.

[38] M.M. Yeung, "Digital Watermarking'', Communications of the ACM, July 1998, Vol.41, No.7, pp. 31 - 33.

[39] J. Zhao and E. Koch, "Embedding robust labels into images for copyright protection'', Technical report, Fraunhofer Institute for Computer Graphics, Darmstadt, Germany, 1994.

[40] J. Zhao, "A WWW Service To Embed And Prove Digital Copyright Watermarks'', Proc. Of the European Conference on Multimedia Application, Services and Techniques, May, 1996, pp. 695 – 710.

[41] J. Zhao, E. Koch, C.Luo, "In Business Today and Tomorrow'', Communications of the ACM, July 1998, Vol.41, No.7, pp. 67 - 72.