WATERMARKING AND SECURE DISTRIBUTION FOR ENCRYPTED VIDEO

T. Amornraksa, D. R. B. Burgess and P. Sweeney
CCSR, University of Surrey, Guildford, GU2 5XH, U.K.

Abstract We propose a technique for watermarking encrypted video which has been encoded using block-based video coding techniques e.g. MPEG. By dividing 8×8 DCT coefficients of the coded video into DC and AC coefficient parts, the encrypting process is applied first but only to the DC coefficient so that the watermarking process can be applied later to the AC coefficients. Although the video signal is not entirely encrypted, in some video applications such as broadcasting services, the quality of this selectively encrypted video is considered to be sufficiently degraded. The technique enables the original copy to be kept in encrypted form with local distributors who sell watermarked encrypted copies while the original owner distributes the decryption keys. After the decrypting and decoding process, the video signal still contains the watermark inside. We describe such a scheme which applies this technique for the purpose of secure distribution of copyright data. The scheme incorporates public keys for which the RSA algorithm is assumed.

1. INTRODUCTION

Copyright protection for multimedia data is an important issue since the digital data can be copied repeatedly without loss of quality and it is difficult to differentiate illegal copies from the original or legal one. One method of copyright protection is the addition to the multimedia data of a *watermark* which carries some information e.g. the copyright owner or the sender or receiver. Therefore, watermarking enables identification and tracing of different copies of distributed data. In some multimedia

B. Preneel (ed.), *Secure Information Networks*

applications such as video distribution over the World Wide Web, the watermark is a digital code embedded in the video signal which typically indicates the copyright owner. In such cases where watermarks are applied to individual copies of the video, each is referred to as a *fingerprint* i.e. the identity of the receiver of the copy.

In the distribution process, the video sequence may be transmitted in encrypted form from a local distributor (merchant) to an end-user (buyer). The video sequence should at least contain fingerprint information to enable the merchant to identify the original buyer of any redistributed copy. Several fingerprinting schemes have been proposed for this kind of application to ensure that the deception will not occur during the transaction process between the merchant and buyer. In classical fingerprinting schemes, e.g. [1], the merchant sees the fingerprinted copy before the encrypting process, that is, both parties know the fingerprinted copy. This leads to the problem of proving to a third party that it was the buyer who made the illegal copy, not the merchant. Using the asymmetric fingerprinting technique proposed in [2], in principle the fingerprinted copy is known by the buyer only. If the merchant finds an illegal copy, he can identify and prove whose copy it was. An advanced technique based on [2] is anonymous fingerprinting [3], where the buyers can act anonymously, but can still be identified if they redistribute the information illegally. Recently, an anonymous fingerprinting scheme with automatic identification of redistributors [4] was proposed with the advantage that the merchants need no help from a registration authority to identify the dishonest buyer.

All these schemes are based on a secure two-party computation and work on an assumption that both parties execute the scheme honestly. When there is no deception, these schemes output the fingerprinted copy to the buyer. However, in our opinion, they do not achieve the requirement that the fingerprinted copy must not be seen by the merchant. When an illegal copy is found, the schemes assume one of the buyers must be responsible for it and hence they protect the merchants from cheating by buyers. However, they give no protection to buyers from cheating by merchants. Although the merchants cannot create the fingerprinted copy in isolation, there are ways for them to obtain such a copy. This is because they have access to a non-encrypted copy. The possibility of this kind of fraud still remains, even if the secure two-party computation is performed at some other place, e.g. a trusted host.

In this paper, an alternative approach based on the technique of watermarking encrypted video is proposed. Using this technique, the original copy will be encrypted by the original owner and kept in encrypted

form with the merchant. The merchant performs the watermark process and distributes the watermarked copy to the buyer. Then the original owner will send the decryption key directly to the buyer. We describe a scheme which applies this technique and has the advantage of not needing to assume the honest application of a protocol. In other words, the need for a secure two-party computation protocol in the fingerprinting schemes can be avoided. The scheme incorporates public keys. One advantage of this is that certain choices of public key schemes (for example RSA) also provide a digital signature scheme by using the secret key to sign the message and the public key to verify the signature. The scheme is such that, if an illegal copy is found, the source can be identified without doubt.

2. THE TECHNIQUE FOR WATERMARKING ENCRYPTED VIDEO

The video signal under consideration is assumed to have been compressed using block-based video coding techniques, which apply some sort of transform coding such as Discrete Cosine Transform (DCT). MPEG (Moving Picture Expert Group) is an industrial standard for video processing and chosen for use in our experiments. In MPEG encoding, the video sequence is compressed frame-by-frame using block-based motion compensation, to take advantage of interframe temporal redundancy, and DCT-based compression, to take advantage of intraframe spatial redundancy [5]. Each frame in the sequence is broken into 8×8 pixel blocks for intraframe DCT compression and 16×16 pixel macroblocks for interframe motion compensation. A 16×16 macroblock also includes two 8×8 chrominance blocks in addition to the four luminance blocks for a total of six blocks per macroblock. There are three types of frames: intracoded frames (I), motion-estimated forward predicted frames (P) and motion-estimated bi-directional predicted frames (B).

I-frames are encoded block by block, without regard to previous or future frames. The encoder calculates the DCT of each 8×8 block, transforming that block into its frequency-based representation, and then applies the processes of thresholding, quantization, zig-zag-scan, run-level-coding and entropy coding. Note that, in the 8×8 transformed block, the coefficient in the top left corner (co-ordinates:0,0) which represents null horizontal and vertical frequencies is called the *DC coefficient*, whereas the rest of the coefficients are called *AC coefficients*. After the quantization step, the DC coefficient is coded separately by a predictive Differential Pulse Code

Modulation (DPCM) technique while the AC coefficients are coded using the *run-level* symbol structure and referring to the Variable Length Code (VLC) tables defined in MPEG. Forward predicted (P) frames are encoded with reference to the most recent previous I or P-frame. Each macroblock undergoes a motion-estimation search to find the so-called best-fit vector relative to a macroblock from the reference frame. This vector is then transmitted along with the error between the macroblocks from the two frames, encoded using the DCT. Macroblocks with no motion and no error are marked as skipped blocks, and no further coding is performed. The MPEG standard allows encoders to include I-blocks in the P-frames. Encoders can choose to encode certain blocks as I-blocks when the block cannot be adequately motion-compensated. Bi-directional predicted (B) frames are coded with reference to both the previous and next I or P-frames. The motion-estimation and encoding procedure for B-frames is similar to that for P-frames.

From the structure above, it can be seen that there is no need for the whole video stream to be encrypted. For instance, we could encrypt only small important parts of the video stream such as I-frames. When such encrypted video is decoded without decrypting first, the coding error will propagate to P and B-frames so these frames too will be in corrupted form. Based on this idea, a selective encryption scheme [6] was proposed with the aim of reducing the computation required in the encrypting process. The experimental results of decoding images without decrypting in [6], with I-frame and I-block in P and B-frame encrypted, are quite satisfactory, although they sometimes reveal a certain amount of information such as the general content in the images. This technique may not be suitable for secure-purpose applications but it is secure enough in some multimedia applications such as video-on-demand. Our method is also based on the idea of selective encryption but the encryption is applied to all the DC coefficients (after the quantization step), leaving all the AC coefficients unchanged for the watermarking process.

In the watermarking process, we use a simple scheme for the watermarking of MPEG video which is presented in [7] but do not watermark the DC coefficient. In [7], the watermark signal is embedded by the addition to the video stream of a pseudo-random signal, which is below the threshold of perception and cannot be removed without the knowledge of the parameters of the watermarking algorithm. This method is an extension of ideas from direct-sequence spread spectrum communications. The watermark signal is obtained by spreading the information bits over many frequency bins (so that the signal energy present in any single

frequency bin is very small and almost certainly undetectable) and modulating them with a binary pseudo-noise sequence.

According to [7], a sequence of information bits $m_j \in \{-1, 1\}$ to be hidden is first spread by a large factor cr, called the chip-rate, to obtain the spread sequence b_i: $b_i = m_j$, $j.cr \le i < (j+1).cr$. This spread sequence b_i is then modulated with a pseudo-noise sequence $p_i \in \{-1, 1\}$ to obtain the watermark signal $p_i.b_i$. The watermark signal may be amplified, subject to consideration of the human visual system (HVS), before finally adding it to the pixels of the line-scanned video sequence. The recovery of the embedded watermark signal can be accomplished by correlating the watermarked video signal with the same pseudo-noise sequence that was used in the process of constructing the watermark signal. Correlation here is demodulation followed by summation over the width of the chip-rate. If the peak of the correlation is positive (or, respectively, negative), the recovered information bit is a +1 (or -1).

To watermark the video encrypted as above, the watermark signal just described is transformed using the DCT and added coefficient-by-coefficient (except the DC coefficient) to the coded video. Each VLC codeword for the coded video must be decoded first in order that each non-zero AC coefficient can be added to the corresponding AC coefficient of the watermark signal. The encrypted DC coefficient remains unchanged. We can control the bit-rate of the watermarked video signal as follows. If the resultant VLC codeword has more bits than the original (before watermark), then the original will be output instead. Thus the bit-rate of the video stream is not increased.

In [7], it is shown that typically around 15-30 % of the DCT coefficients are altered, depending on scene structure and bit-rate, although in our approach, the DC coefficient will not be watermarked. The authors indicated that should there be insufficient DCT coefficients to recover the watermark signal, it may be possible to compensate by increasing the chip-rate (although this results in a decrease in the data rate for the watermark). We have carried out experiments based on our technique and the results are shown in Figure 1 and 2.

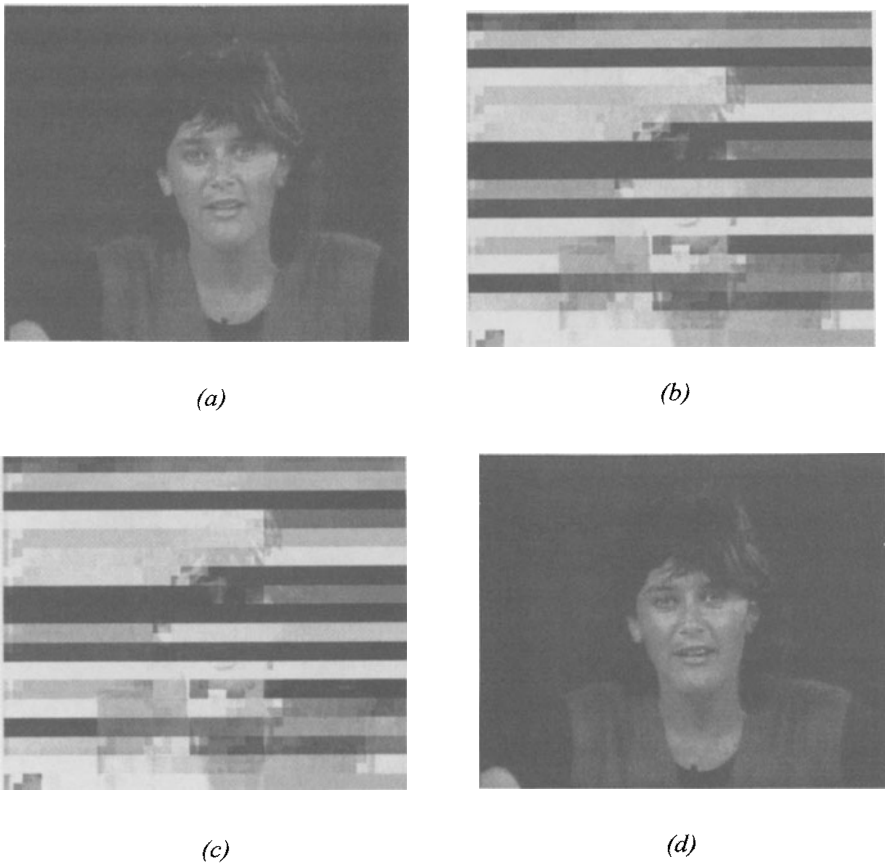
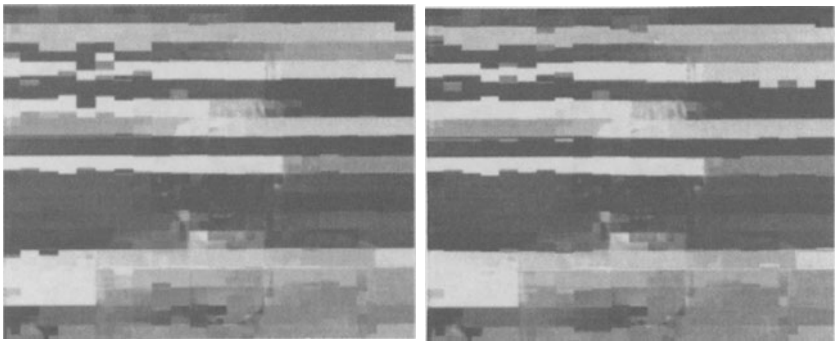


Figure 1. First (I-) frame of 'Miss America' sequence: (a) original frame (b) encryption of DC coefficients (c) watermark signal is embedded in AC coefficients (d) watermarked frame after decrypting.



(a) *(b)*

Figure 2. (a) A P-frame and (b) a B-frame from 'Miss America' sequence after encryption of DC coefficients and propagation of coding errors from other frames.

3. IMPLEMENTATION AND FINGERPRINTING SCHEME

The general model of the scheme is that a set of individual copies of the DC coefficient of the original video is sent to the merchant, each copy encrypted with a unique encryption key *Ki*, and each having a corresponding serial number. Before the encrypting process, the original owner will embed in each copy a fingerprint to allow later identification both of the merchant and of that particular copy (in the manner of [8]). A single copy of the remainder of the video is also sent to each of the merchants who are in different locations e.g. other countries. (See step 1 of Figure 3.) In our experiments, the amount of the DC coefficient turns out to be only around 1% of the whole video. So each merchant needs to store only the (small) individually encrypted copies of the DC coefficient and one copy of the rest of video. The fingerprinting scheme we propose for implementation of the watermarking process above draws highly on the use of public keys. We assume that each party has a key pair (*Pk* and *Sk*) under the RSA algorithm, so that the public key can serve as a digital signature.

After a registering process, when a customer requests to buy/watch the video (step 2), the merchant sends the original owner an authenticated message, secured by the owner's public key, containing the request information (step 3) who then sends the corresponding decryption key encrypted by the customer's public key directly to the customer (step 4).

The merchant then makes another copy of the non-encrypted part of the video and embeds a unique watermark (i.e. a fingerprint) into it. This is combined with one of the individually encrypted copies of the DC coefficient and the resultant encrypted and fingerprinted copy of the video is sent to the customer (step 5).

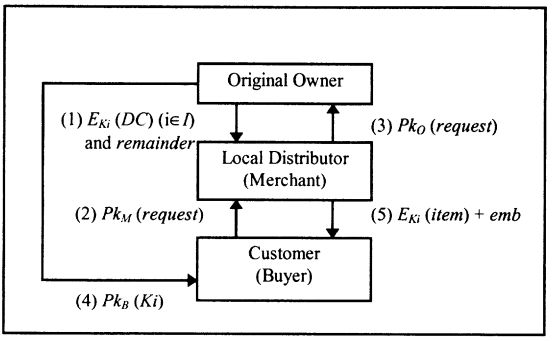


Figure 3. General model of the fingerprinting scheme

In practice, there will be an additional body, namely a trusted party, who the original owner will allow to manage the key distribution for the customers. The details of the registering and fingerprinting processes are as follows. (Note that we introduce a registration authority to achieve the registering process.)

PROTOCOL FOR REGISTERING PROCESS

i. The buyer B encrypts his identity (B_i), public key (Pk_B) and signature ($Sk_B[h_{B_i}]$) by using the public key of the registration authority R (Pk_R) and sends Pk_R (B_i, Pk_B, $Sk_B[h_{B_i}]$) to R, where h_{B_i} is the hash value of B_i derived from a one-way hash function such as MD5. The buyer's identity may contain some verifiable data to protect against impersonation fraud e.g. credit card number.

ii. R verifies B's signature by applying Pk_B to $Sk_B[h_{B_i}]$ and comparing with $h[B_i]$, then checks his identity with some trusted party e.g. a bank. If everything is correct, R records the information and acknowledges the registration by signing Pk_B with his secret key (Sk_R) and sends back Pk_B ($Sk_R[Pk_B]$) to B.

iii. B authenticates the acknowledgement from R's signature. B now uses his public key (Pk_B) and R's signature ($Sk_R[Pk_B]$) every time he contacts the

merchant. $Sk_R[Pk_B]$ is used as a certificate *cert* to prove that B's public key is valid. Pk_B can be considered as a pseudonym used in the anonymous fingerprinting scheme, and B can register several times so he may have different pseudonyms.

PROTOCOL FOR FINGERPRINTING PROCESS

i. B sends Pk_M *(Text,* Pk_B, $Sk_B[h_T]$, *cert,* R_i*)* to the merchant M, where *Text* is a string identifying the purchase, h_T is the hash value of *Text* and R_i is R's identity.
ii. M decrypts B's request by using his secret key (Sk_M) and finds Pk_R that corresponds to R_i. Then he checks the validity of the public key by verifying *cert* $(Sk_R[Pk_B])$ with Pk_R and Pk_B. Also, M authenticates the key pair by verifying B's signature by applying Pk_B to $Sk_B[h_T]$ and comparing with h_T. If both results are true, M sends Pk_O *(serial number, Text,* Pk_B, $Sk_B[h_T]$, $Sk_M[Pk_B]$, *M)* to the original owner O, where $Sk_M[Pk_B]$ acts as a certificate issued by M to guarantee B.
iii. O decrypts M's request by using his secret key (Sk_O) and finds Pk_M that corresponds to *M*. Then he checks the validity of the public key by verifying M's signature $(Sk_M[Pk_B])$ with Pk_M and Pk_B. He also authenticates B's request by verifying B's signature in the same manner as did M. If both results are true, O sends back a secured acknowledgement Pk_M $(Sk_O$*[serial number], O)* to M and the secured decryption key Pk_B *(Ki)* to B.
iv. When M has all parameters as previously described, he uses the proposed watermarking technique to embed a fingerprint *emb* into E_{Ki}*(item)*. He thus forms the final version of the whole video and then sends this copy to B.

The fingerprint information (*emb*) to be embedded in E_{Ki}*(item)* is *Text*\\ $Sk_B[h_T]$\\ Pk_B\\ *cert*, and the algorithm given in section 5 of [3] allows the embedding and extraction of relatively large amounts of information.

4. SECURITY ANALYSIS

If M finds an illegal copy, he extracts *emb* from that copy and verifies the buyers' signature ($Sk_B[h_T]$) by using Pk_B. This process is for proving that the owner of this public key received the original fingerprinted copy and he

made an illegal copy. Then the merchant sends this proof to R and asks for the identity of this person. With *cert* ($Sk_R[Pk_B]$) in *emb*, R must know B's identity.

Since M puts the fingerprint information into the request to O, O can log, for each purchase, the fingerprint identifying the merchant and the particular copy, the fingerprint identifying the buyer and purchase, the merchant, the buyer and the decryption key issued. M is blamed if O receives more than one request for the same decryption key or if a pirate copy turns up which bears invalid (or no) fingerprints. (In the latter case M's fingerprint in the DC coefficient of that copy would identify M as the culprit.) Otherwise any pirate copy has a correct fingerprint corresponding to some buyer B who is thus responsible.

According to the scheme, M uses Pk_R and the signature issued by R to prove the validity of B before sending him the fingerprinted encrypted copy of the video, whilst O uses Pk_M and the signature issued by M to prove the validity of the transaction between M and B before sending the decryption key to B. Therefore, it can be considered that the security of the whole system relies on the difficulty of breaking the public key algorithm. That is, if the secret key of any party can be derived, that party's signature can be forged. For the problem of public key authentication, for instance, B and M can obtain Pk_R in several ways such as they can get it directly from R or from a public key database. If it is necessary for R's public key to be taken from a database then all public keys must be signed by some trusted party known to B and M.

The security of the scheme clearly also depends on the watermarking technique used being robust. In other words, we are assuming that the information in the watermarks cannot be affected by attacks which do not significantly impair the quality of the video. The potential attacker is the merchant in the case of the watermark applied to the DC coefficient and the buyer in the case of the watermark applied to the remainder of the video (AC coefficients). The debate on the robustness of the various watermarking techniques is ongoing with existing techniques being refined to meet previously unconsidered attacks [9]. The technique we have used for watermarking the AC coefficients in our scheme is certainly believed to be robust to current known attacks when applied to the entire video [7]. The same technique could be used for the watermark for the DC coefficient since the robustness of such a watermark has been argued by previous authors who have also suggested selective watermarking [8].

5. CONCLUSIONS

We have proposed a technique of watermarking encrypted video where a DCT has been performed. It is based on the idea of encrypting the DC coefficient and watermarking the AC coefficients. Our experimental results have shown that the encrypted images are sufficiently degraded for general applications such as video distribution over Internet or video-on-demand.

We have also presented a scheme for implementing the proposed technique which gives secure distribution of copyright compressed video stream data. The scheme allows the encrypted video to be sent to merchants in different locations prior to fingerprinting. The encrypted copy of the video is no use to the merchant since he does not have the decryption key. Thus there is protection from a dishonest merchant who tries to deceive other parties in the system. Another advantage of the scheme is that it does not need a secure two-party computing protocol and hence means there is a reduction in complexity compared to the existing schemes.

Finally, the scheme can also be applied to some programmed broadcast applications such as pay-per-view by distributing the decryption keys to the customers before the broadcasting begins. However, since the fingerprinted parts (i.e. AC coefficients) are a very large proportion of the whole video and differ between customers, the bandwidth required for the broadcasting channel will be too large in practice. Nevertheless, we cannot discount the possibility of the future discovery of an efficient watermarking technique which fulfils all necessary requirements but only requires a small amount of the information to be watermarked. In fact, whatever future advances there may be in watermarking techniques, we can expect them to make the proposed scheme even more benefitial to any of the applications.

REFERENCES

[1] D. Boneh and J. Shaw, 'Collusion-secure fingerprinting for digital data', Advances in Cryptology-CRYPTO'95, LNCS 963, (Springer-Verlag, Berlin, 1995), pp. 452-465.

[2] B. Pfitzmann and M. Schunter, 'Asymmetric fingerprinting', Advances in Cryptology-EUROCRYPT'96, LNCS 1070, (Springer-Verlag, Berlin, 1996), pp. 84-95.

[3] B. Pfitzmann and M. Waidner, 'Anonymous fingerprinting', Advances in Cryptology-EUROCRYPT'97, LNCS 1233, (Springer-Verlag, Berlin, 1997), pp. 88-102.

[4] J. Domingo-Ferrer, 'Anonymous fingerprinting of electronic information with automatic identification of redistributors', Electronics Letters, vol. 34, no. 13, June 1998, pp. 1303-1304.

[5] J. Mitchell, W. Pennebaker, C. Fogg and D. Legall, 'MPEG Compression Standard', Chapman & Hall, New York, 1997.

[6] I. Agi and L. Gong, 'An Empirical Study of Secure MPEG Transmission', In ISOC Symposium on Network and Distributed System Security, San Diego, CA, February 1996.

[7] F. Hartung and B. Girod, 'Watermarking of Uncompressed and Compressed Video', Signal Processing, Vol. 66, no. 3 (Special issue on Watermarking), pp. 283-301, May 1998.

[8] T-L. Wu and S. F. Wu, 'Selective Encryption and Watermarking of MPEG Video', International Conference on Image Science, Systems, and Technology, CISST'97, 1997.

[9] F. Hartung, J. K. Su and B. Girod, 'Spread Spectrum Watermarking: Malicious Attacks and Counterattacks', Security and Watermarking of Multimedia Contents, Vol. 3657 of SPIE Proceedings Series, January 1999.