A Robust and Secure Media Signature Scheme for JPEG Images

Qibin Sun, Qi Tian
Media Engineering
Laboratories for Information Technology
119613, Singapore
{qibin, tian}@lit.a-star.edu.sg

Shih-Fu Chang
Department of Electrcal Engineering
Columbia University
New York City, 10027, USA
sfchang@ee.columbia.edu

***Abstract*—In [1, 2, 3], we have introduced a robust and secure digital signature solution for multimedia content authentication, by integrating content feature extraction, error correction coding (ECC), watermarking and cryptographic hashing into a unified framework. We have successfully applied it to JPEG2000 as well as generic wavelet transform based applications. In this paper, we shall introduce a new JPEG-compliant solution under our proposed framework but with different ECC and watermarking methods. System security analysis as well as system robustness evaluation will also be given to further demonstrate the practicability of our method.**

Keywords—Digital signature; authentication; watermarking; ECC

I. INTRODUCTION

Our objective is to design a digital signature scheme that allows two parties to exchange images while guaranteeing content integrity and non-repudiation from content owner, in a semi-fragile way. Integrity protection means that the content is not allowed to be modified in a way such that the content meaning is altered. Non-repudiation prevention means that once a content owner generates the content signature, he cannot subsequently deny such a signing if both the signature and the content have been verified as being authentic.

State-of-the-art work shows that the above objective can be achieved in a fragile way (i.e., even one bit change is not allowable) either by watermarking [4] or by cryptographic digital signature scheme such as RSA or DSA [5]. However, some applications demand the same security solution on a semi-fragile level, i.e., some manipulations on the content will be considered acceptable (e.g. lossy compression) while some are not allowable (e.g. content copy-paste). However, at the semi-fragile level, watermarking-based approaches only work well in protecting the integrity of the content [6] but are unable to solve the non-repudiation issue caused by the use of a symmetric key for watermark embedding and extracting. Once the key or watermark is compromised, attackers can use the key or watermark to fake other images as authentic. Signature based approaches can work on both the integrity protection of the content and the repudiation prevention of the owner, but a shortcoming exists. The generated signature is unavoidably very large because its size is usually proportional to the image size. Some recent work can be found in [7, 8, 9].

In [1, 2, 3], we have already introduced a robust and secure digital signature solution for multimedia content-based authentication, by integrating content-based feature extraction, error correction coding, watermarking and crypto hashing into a unified framework. The proposed scheme is *efficient* by generating only one crypto signature (hundreds of bits) per image regardless of image size, in a semi-fragile way. System *robustness* (i.e., the ability to tolerate incidental distortions from some predefined acceptable manipulations such as lossy compression) is achieved through an effective method based on error correction coding (ECC) techniques. System *security* (i.e., the ability to prevent attacked images from passing authentication) is obtained by adopting crypto hashing and signing. In addition, watermarking is used for storing ECC check information and locating change locations.

The above framework is compliant with traditional digital signature system structures. In the content signing procedure, the content owner uses his private key to sign on the hash value of the extracted features, embed the signature to the image, and send watermarked content to the recipients. In the content verification procedure, the recipient can verify the received content's authenticity by using its owner's public key and the associated signature. The watermarking is done in such a way that it can indicate the locations of attacks on the content if the authentication procedure fails. Such a capability is important because it helps to visually convince users of the authentication result. In order to differentiate our framework from traditional digital signature schemes such as RSA, we name our proposed solution as **media signature** hereafter.

Considering compliance with the JPEG standard encoding and decoding procedure, in next section we shall introduce a JPEG-compliant solution whose idea derives from our previous solutions [1, 2, 3] but realizes a different ECC and watermarking implementation. System security and robustness will be analyzed and evaluated in Section III and Section IV respectively. Conclusion and future work are presented in Section V.

II. PROPOSED JPEG-COMPLIANT MEDIA SIGNATURE SCHEME

A typical JPEG compression procedure includes block formation, DCT, quantization and lossless entropy coding. In this paper, we select DCT coefficient as the feature. Denote a DCT coefficient before quantization as D, the quantization step size specified in the quantization table is Q, and the output of quantizer is quotient F (integer rounding) and remainder R respectively. We have

$$D / Q = F, D \% Q = R = D - F * Q \qquad (1)$$

For JPEG compression, the F will be losslessly compressed and R will be discarded. Suppose the incidental distortion introduced by acceptable manipulations can be modeled as noise whose maximum absolute magnitude is denoted as N, we can then use R to correct the errors of F caused by corruption from added noise.

Refer to Figure 1, assuming $Q>4N$, N is the maximum range of added noise, we can see that if the original DCT value is located at the point nQ, then no matter how this value is corrupted, the distorted value will still be in the range $((n-0.5)Q,(n+0.5)Q)$, and the quantized DCT value will remain unchanged as nQ before and after noise addition. However, if the original DCT value drops into the range of $((n-0.5)Q,nQ)$ (the point *P* in Figure 1, its quantized value is still nQ before adding noise, but there is also a possibility that the noisy DCT value could drop at the range $((n-1)Q,(n-0.5)Q)$ and will be quantized as $(n-1)Q$, not nQ, after adding noise. Thus the noise corruption will cause a different quantization result. To avoid such a case, we propose a simple ECC-like procedure to record the sign of *R*. We want to push the points away from the quantization decision boundaries and create a margin of at least $Q/4$ so that the DCT value when contaminated later will not exceed the quantization decision boundaries.

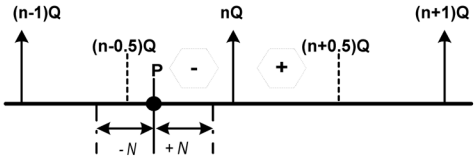

Figure 1. Illustration on the concept of error correction

In the ECC procedure, let's record a 0 bit if the original DCT value drops between $((n-0.5)Q,nQ)$ (i.e., $R<0$). In the authentication procedure, assume this DCT value has been corrupted. Add the value $0.25Q$ from the corrupted value before quantizing it if we retrieve a 0 bit indicating R<0. Then we can obtain the same quantization value as nQ. Similarly, for the case that the original DCT value is in $(nQ,(n+0.5)Q)$ (i.e., $R>0$), we record a 1 bit and we should subtract $0.25Q$ from the corrupted DCT value before the quantization. We can still obtain the same quantized value as nQ. Based on such an error correction concept, the crypto hashed value on all quantized DCT values will stay the same before and after distortion. This forms the basis of our solution.

In [8], the authors presented a content-based watermarking solution for JPEG image authentication. Two quantization step sizes are used: Q_a is for generating features and watermarking while Q_c is for actual JPEG compression. They proved that as long as Q_c is less than than Q_a, the robustness of generated features as well as embedded watermarks can be guaranteed. We shall use this concept in our solution. The whole media signature signing/verification algorithm is depicted as follows.

Signature Generation

Initialization

Random sequence *S* for feature selection / watermarking (*S* is used to decide which DCT values are used for feature extraction and which are used for watermarking and it will be included in the signature for verification purpose)

Input

Owner's private key *Pri*.

Original image I_o to be protected.

Authentication quantization step size Q_a.

JPEG compression quantization step size Q_c. Note $Q_c < Q_a$

Begin

Normal JPEG compression processing such as blocking, DCT, obtaining a number of 8x8 DCT blocks in zig-zag scan denoted as: $\{D_{ij}^O : 0 \le i < 64;\ 0 \le j < N\}$.

For $j = 0 : N - 1$ **Do**

Take DC and other 3 AC coefficients randomly selected by *S* and form feature set *F* containing 4 elements:

$$F_j = \{D_{0j}^O; D_{lj}^O : l = \{i : 1 \le i \le 20\}_S^{(3)}\}.$$

Compute rounding quantization on F_j, obtain quantized value $\overline{F_j}$ and remainders R_j according to (1) with Q_a.

Generate watermark W_j with $W_j = \begin{cases} 0, & if\ R_j < 0 \\ 1, & if\ R_j \ge 0 \end{cases}$.

Embed W_j into the same block by using other AC coefs labeling $1 \le i \le 20$ but excluding those which have been used for feature set, refer to [8] for detailed watermarking scheme.

De-quantize all processed DCT coeffs with Q_a.

End

Crypto hash all concatenated $\overline{F_j}$: $H^O = h(\cup \overline{F_j})$, $0 \le j < N$.

Sign on H^O by *Pri* and obtain signature *G*.

Compress watermarked image I_w with Q_c quality (i.e., quantize all DCT coeffs with Q_c) and obtain I_w.

End

Output

Compressed watermarked image I_w.

Content based signature *G*.

Signature Verification

Same random sequence *S* obtained from the signature.

Input

Image I_w to be authenticated.

Owner's public key *Pub*

Associated signature *G*.

Authentication quantization step size Q_a.

JPEG compression quantization step size Q_c. Note $Q_c < Q_a$

Begin

Normal JPEG decoding such as Huffman, de-quantizing with Q_c, obtaining a number of 8x8 DCT blocks in zig-zag scan denoted as: $\{D_{ij}^W : 0 \le i < 64;\ 0 \le j < N\}$.

For $j = 0 : N - 1$ **Do**

Take DC and other 3 AC coefficients randomly selected by *S* and form feature set F^W containing 4 elements:

$$F_j^W = \{D_{0j}^W; D_{lj}^W : l = \{i : 1 \le i \le 20\}_S^{(3)}\}.$$

Extract W_j from the same block, refer to [8] for detailed watermark extraction procedure.

Modify F_j^W according to the extracted W_j:

$$F_j^W = \begin{cases} F_j^W + 0.25Q_a, & if\ W_j = 0 \\ F_j^W - 0.25Q_a, & if\ W_j = 1 \end{cases}$$

Compute rounding quantization on F_j^W and obtain quantized values $\overline{F}_j^W$ and remainders R_j^W according to (1) with Q_a.

Correlate W_j with R_j^W, obtain difference image I_d.

End

Crypto hash all concatenated $\overline{F}_j^W$: $H^W = h(\cup \overline{F}_j^W)$, $0 \le j < N$.

Decrypt signature G by owner's public key *Pub,* obtain hash H^O.

Bit-wise comparison between H^O and H^W: $A = xor\left(H^O, H^W\right)$

End

Output

If $A > 1$, then report image unauthentic and display I_d to indicate possible attacked locations.

If $A = 0$, the image is authentic.

Note that sometimes difference image I_d fails to indicate the modification locations although the image is verified as unauthentic based on bit-wise comparison between H_o and H_w. For example, DCT values originally closed to $n*Q$ will be pushed to the opposite side and thus the sign changes due to acceptable manipulations. Or similarly, when there are attacks, there might not be sign changes, e.g., the change to the DCT value is large, pushing the value from "-" side of $n*Q$ to "-" side of $(n+1)*Q$, even after $Q/4$ adjustment. However, the crypto hash value definitely will be changed, because the quantized values are changed.

III. System Security Analysis

In this subsection, we investigate the security of our media signature solution. Since the whole media signature scheme is compliant with crypto digital signature, we only need to analyze the security of generating content-based hash. Refer to Figure 2, we can see that actually the procedure of generating content-based hash consists of three processing modules. In addition to crypto hashing, the last step to forming content-based hash, the other two processing modules are feature extraction and error correction coding. Therefore, if we denote security in terms of the possibility being attacked, the security for our scheme comprises of three possibilities: P_F in feature extraction, P_E in error correction coding and P_C in crypto hashing. (The security for crypto hashing function (e.g., 160 bits in SHA-1) is [5]: $P_C \approx 2^{-80}$ under well-known "birthday attack"). Since they are mutually independent and very small, the final security of our proposed scheme could be approximated as: $P = P_F + P_E + P_C$. This represents the possibility of finding a faked image which can pass the authentication test. Obviously it is much larger than P_C, which is usually deemed as nearly perfectly secure based on current computational capabilities. Therefore, we only need to study P_F and P_E which impair the system security in different ways, as shown in Figure 2. A good feature descriptor should represent as much entropy of the original source as possible. Differing from feature extraction which functions as "removing" redundancy from original source, ECC functions as "adding" redundancy in order to tolerate incidental distortions. Thus, a good feature extraction method and a proper ECC scheme are the key factors in system security. Let's check P_E first. Originated from [10], we have:

Lemma: Let H_C be our proposed content-based hash scheme based on an ECC with error correction ability t (e.g., $t = 0.25Q$). For any D' which satisfies $\|D - D'\| \le t$, we have $H_C(F) = H_C(F')$.

Note that the notations in the above lemma are the same as media signature algorithm. We skip its proof here due to the paper size limit. Basically, it states that as long as the difference between original DCT values and corrupted DCT values are not greater than t, their corresponding hash values are the same as by using ECC. Clearly, ECC diminishes system security in some sense as ECC does provide the property of fuzziness. This goes back to the issue of how to select a proper ECC scheme to balance between system robustness and security. This issue is application dependent. In our proposed scheme for JPEG (refer to the algorithm described above), we take the remainder of quantization of DCT coefficient for error correction while hashing quantized DCT values. Since the maximum magnitude of the remainder is less than half of quantization step size Q_a and it is usually discarded in JPEG compression, we argue that the security risk caused by this particular ECC scheme should be ignored. We mean that changing and attacking on remainder of DCT values will not affect the system security while attacking quantized DCT values is computationally impossible because a crypto hashing works on protecting all quantized DCT values.

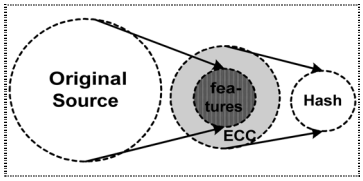

Figure 2. Illustration of the security of proposed scheme

Before analysing P_F, we make the following assumption:

Assumption: Lossy compression (e.g, JPEG/JPEG2000) is the approximate representation of its original source in terms of an optimal rate-distortion measure.

Intuitively, it implies there is no security gap between the original source and its compressed version under a given targeted bit-rate. Therefore, we can argue that if a content-based hashing function could make use of all compressed information (e.g., all quantized DCT coefficients in all blocks), it should be considered secure at this targeted bit rate. In other words, if an attacker intends to modify the content in the spaces between original source and its compressed version, this attack should not be deemed harmful to the content. However, because we need to store ECC check information back into the content, we may have to pick up only some DCT coefficients for hash generation. In such a case, the system security will suffer. Since the low frequency DCT coefficients are more important than high frequency DCT coefficients, selecting DCT coefficients from a low frequency band for hash generation, as we did in our proposed scheme, will gain more security than selecting from a high frequency band. The worst security performance of our proposed scheme can then be estimated by simply assuming all DCT coefficients own the same importance to system security. As we described before, we select one DC and three AC components for feature formation and embed them back into other AC components again, all in the range of the first 20 DCT coefficients which are from the low to middle frequency band. Therefore, analyzing P_F in one 8x8 block is equivalent to computing the possibility in finding 7 balls (3 from AC features and 4 for embedding) with a fixed order from 20 balls: $P_F = (20-7)!/20! \approx 2.56 \times 10^{-9}$. This represents the probability of the attacker guessing correctly the locations

of the 7 coefficients in an exhaustive attempt. If he can determine the correct locations, he can then keep the DCT values at these locations unchanged, and try to cause harmful misunderstanding by exhaustively changing other DCT values in this block. Although it is still very difficult, especially when we take the contextual property (i.e., the values of its surrounding pixels are very similar to the central one) of image into account, we have to analyze the worst case scenario from the viewpoint of system security. The final security could then be: $P = P_F + P_E + P_C \approx P_F = \left(2.56 \times 10^{-9}\right)^N$, where N is the total number of blocks in an image. We can see P is not so high from the viewpoint of cryptography, especially in the case of attacking locally (i.e., N may be 1). However, such security performance should be able to satisfy the requirements of real applications, considering that strong contextual property in image will also increase the difficulties in attacking.

The last issue related to security is watermarking. In our proposed scheme, we do not need to pay more attention to watermarking security except by using S to randomly select watermarking locations. The main reason is that watermarking here only functions to store ECC check information. Since the distribution of remainder R is independent from $\overline{F}$, and it does not reveal any information about $\overline{F}$, leaving R in the public will not affect the security of hash function [10].

IV. System Robustness Evaluation

For semi-fragile authentication, defining acceptable manipulations is a key and first step in order to differentiate malicious attacks from acceptable manipulations. But to date, no effective measures can successfully accomplish this. In our proposed scheme, We simply define the maximum allowable distortions as acceptable manipulations which is set to 1/4 of the quantization step size. Figure 3 shows the authentication result on an attacked image and Figure 4 shows the result on a noisy image. We use RSA to sign and verify the generated media signature whose length is 1024 bits.

Figure 3. Authentication result on attacked image

In Figure 3, the original image on the left is used for media signature generation, the center windowcontains the modified watermarked image compressed with JPEG quality factor 50, and right-most window is the signature verification result showing the possible locations of modification. In Figure 4, the image to the leftis the original and one to the right is the noisy watermarked image with a zero mean and 0.01 variance Gaussion noise. We can see that when we use the same settings as in Figure 3, this noisy image can be verified successfully. We also adopt some image pre-processing techniques such as low pass filtering and histogram equalization to help to stabilize signature verification [9]. As an implementation issue, the watermarking method we adopted is the same as [8] except for blocks where watermarks cannot be embedded due to visual quality consideration. For these blocks, we simply substitute a random sequence generated by S and block location for the features extracted from this block to crypto hashing. More extensive experiments will be conducted as our future work both in security tests as well as robustness tests.

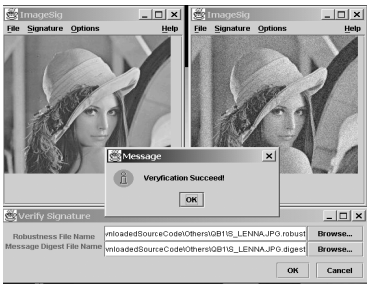

Figure 4. Authentication result on noised image

V. Conclusion and Future Work

In this paper, we have proposed a new semi-fragile image authentication solution combining ECC and watermarking. By using ECC, we provide a mechanism allowing minor variations of content features caused by acceptable manipulations (such as lossy compression, low-pass filtering). In summary, our media signature scheme not only eliminates the signature size issue in previous signature-based methods but also eliminates the security issue in previous watermarking based methods. The whole solution is compliant with Public Key Infrastructure (PKI) while retaining promising system robustness as well as security performance. Future work includes more rigorous testing and analysis and extending this process to other media such as audio and video.

References

[1] Qibin Sun, Shih-Fu Chang, Maeno Kurato and Masayuki Suto, A new semi-fragile image authentication framework combining ECC and PKI infrastructure, *ISCAS 2002*, Phoneix, USA, 2002.

[2] Qibin Sun, Shih-Fu Chang, Maeno Kurato and Masayuki Suto, A quantitive semi-fragile JPEG2000 image authentication system, *ICIP 2002*, Rochester, USA, 2002.

[3] Qibin Sun and Shih-Fu Chang, Semi-fragile image authentication using generic wavelet domain features and ECC, *ICIP 2002*, Rochester, USA.

[4] P. W. Wong and N. Memon, Secret and public image watermarking schemes for image authentication and ownership verification, *IEEE Transactions on Image Processing*, Vol.10, No.10, pp.1593-1601, 2001

[5] B. Schneier, *Applied Cryptography*, New York: Wiley, 1996.

[6] D. Kundur and D. Hatzinakos, Digital Watermarking for Telltale Tamper-Proofing and Authentication, *Proceedings of the IEEE Special Issue on Identification and Protection of Multimedia Information*, vol. 87, no. 7, pp. 1167-1180, July 1999

[7] C.-Y. Lin and S.-F. Chang, A robust image authentication method surviving JPEG lossy compression, *SPIE Security and Watermarking of Multimedia Content*, Vol.3312, pp.296-307, 1998

[8] C.-Y. Lin and S.-F. Chang, Semi-Fragile Watermarking for Authenticating JPEG Visual Content, *SPIE Security and Watermarking of Multimedia Contents II EI '00*, SanJose, CA, Jan. 2000

[9] L. Xie, G. R. Arce and R. F. Graveman, Approximate image message authentication codes, *IEEE Trans Multimedia*, Vol.3, No.2, pp.242-252, 2001

[10] A Juels and M. Wattenberg, A fuzzy commitment scheme, Proceedings of ACM Conference on Computer and Communications Security'99, Singapore, 1999.