A Codestream Domain Authentication and Tamper Localization Scheme for JPEG 2000

Marielena PALACIOS PEREZ, Masaaki FUJIYOSHI, and Hitoshi KIYA
Tokyo Metropolitan University, Japan
E-mail: palaciosperez-marielena@sd.tmu.ac.jp, mfujiyoshi@ieee.org, kiya@sd.tmu.ac.jp
Tel: +81–42–585–8454, +81–42–585–8419

***Abstract*—This paper proposes a tamper detection and localization scheme for JPEG 2000 coded images. The proposed scheme signs a JPEG 2000 compressed codestream without decompressing it, i.e. in the codestream domain. The generated signature is directly inserted to the codestream so that the sign is not needed to transmit separately. The signature insertion in the proposed scheme complies with codestream structure, so a standard JPEG 2000 decoder simply neglects the signature and the decoded image is not distorted. Tamper detection compares the signature extracted from the signed codestream and the signature regenerated from the codestream. Since the signature is generated in encoded codeblocks, tampered regions are spatially localized as well as tampered positions are localized in the codestream domain. Experimental results show the effectiveness of the proposed scheme.**

I. Introduction

Digital contents are not as secure as they should be. Images, for example, are vulnerable to tampering and it is often difficult for humans to detect tamper when original and tampered images are compared, so image authentication schemes are required to ensure image integrity [1]. An image authentication scheme detects intentional modifications using digital signature [2], robust hash [3] or non-intrusive schemes that do not process images when they are created [1].

In digital signature-based image authentication schemes, a digital signature is generated from an image. An original signature is compared with a reference signature that is computed from the image that could have been tampered. If the image has been tampered, the reference signature will differ from the original. The signature is hidden in the image itself in fragile watermarking [4], and the encrypted signature is transmitted along with the image in others [4]. This paper focuses on the latter.

Meanwhile, digital images are often compressed for efficient transmission and/or storing. JPEG 2000 (JP2) [5], [6] is an international standard for multimedia compression based on discrete wavelet transformation (DWT), and it is known for its set of useful features as well as its superior compression performance. Authentication schemes for JP2 coded images have been reported [7]–[11], and they are classified into two groups; one generates and hides a signature in the DWT domainin [7], [8], and the other works without decompressing the compressed codestream [9]–[11]. The former should decode the codestream for processing and the latter does not localize the tampered region.

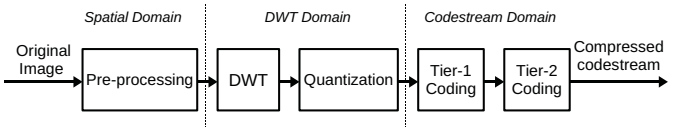


Fig. 1. JPEG 2000 Encoder.

This paper proposes a new tamper detection and localization scheme for JP2 coded images. The proposed scheme generates a signature directly from a JP2 compressed codestream and directly inserts the signature into the codestream without decoding the codestream. The signature is generated for a codeblock, the minimal coding unit in JP2, in each DWT subband. The signature insertion in the proposed scheme complies with JP2 codestream structure, a standard decoder simply neglects the inserted sign and no distortion is introduced into the decoded image. Mismatched signatures indicate the tampered positions not only in the codestream domain but also in the DWT and spatial domains.

The organization of this paper is as follows. Section II overviews the JP2 standard. Section III presents the proposed authentication and tamper localization scheme, and the experimental results are shown in Section IV. Finally, conclusions are drawn in Section V.

II. JPEG 2000 Overview

This section outlines the JP2 standard [5], [6]. A JP2 encoder is illustrated in Fig. 1. An original image, first, is spatially divided into blocks referred to as tiles which will be independently encoded. Furthermore, a level shifting of pixel values and a color transformation are applied to one or more tile(s) before computing the DWT. In case of lossy compression, quantization is applied to DWT coefficients. DWT coefficients are further divided into blocks referred to as codeblocks. Here, a codeblock is a unit for encoding. Finally, the adaptive binary arithmetic encoding known as Tier-1 encoding and the codestream organizing (Tier-2 coding) are adopted. Subsequent sections further mention the DWT, codeblock, and codestream structure.

A. DWT

The DWT is widely known for its multiresolution image representation. It is able to decompose an image into subbands as shown in Fig. 2 and to compute reversible (lossless) and

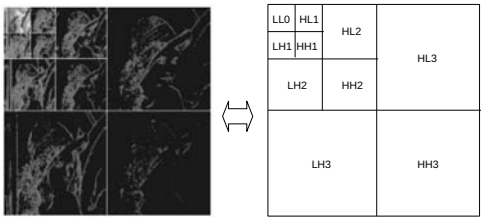


Fig. 2. DWT Level Decomposition.

TABLE I
FREQUENCY-SPATIAL MODIFICATION PROPAGATION (ONE COEFFICIENT IN THE SUBBAND CORRESPONDS TO PIXELS MENTIONED IN THIS TABLE)

Level of Decomposition	Subband			
	LL	LH	HL	HH
1	49	63	63	81
2	289	341	341	401
3	1229	1395	1395	1577

irreversible (lossy) transformations by using different filters. Part 1 of JP2 standard has adopted two transformations for lossy and lossless compression. Each transformation consist of a low pass and a high pass filter pair known as the analysis filter banks for forward DWT and the synthesis filter banks for inverse DWT.

In Fig. 2, three decomposition levels exist: LL_0 represents the lowest resolution subband that is a degraded approximation of the original image, and LH_3, HL_3, and HH_3 are the highest resolution subbands that correspond to vertically, horizontally, and diagonally oriented edges and lines, respectively.

When one DWT coefficient in a subband is modified in the DWT domain, its effect propagates to several pixels in the spatial domain. The range of the propagation is according to the subband, the subband level, and the synthesis filter bank. Table I shows an example of the propagation for two dimensional DWT. A modification of only one coefficient in HH1 spreads its effect to 81 pixels in the spatial domain.

B. Codeblocks

Each subband is equally partitioned into rectangular referred to as codeblocks (Fig. 3). Codeblocks are used as an independently unit for arithmetic coding. Typical sizes of codeblocks are 32×32 and 64×64.

C. Codestream Organization

Figure 4 shows the JP2 codestream structure. A JP2 codestream begins with a main header followed by a sequence

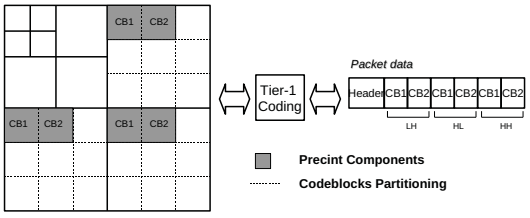


Fig. 3. Precinct and Packets Partitioning.

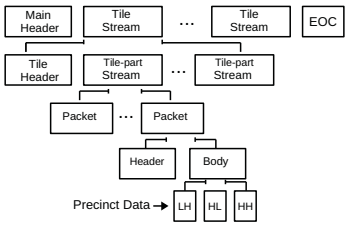


Fig. 4. JPEG 2000 Codestream Syntax.

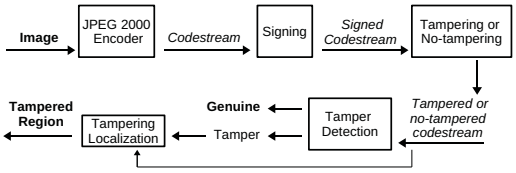


Fig. 5. The Proposed System.

of tile streams and a marker segment called EOC (end of codestream) at the end. Each tile stream is composed of a tile header and a set of part streams which group the packets. A header and body data are included in each packet, and the body data form a precint that is composed of arithmetic encoded codeblocks from a resolution level.

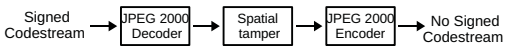
Figure 3 summarizes the relation between codeblocks, precincts, and packets. In this figure, three subbands in the highest decomposition level are divided into codeblocks. Shadowed rectangles represent a set of codeblocks from the same spatial region that are used to construct a precinct. Though codeblocks are independently encoded by Tier-1 encoder, codeblocks are organized in a packet according to the precinct partitioning.

The next section proposes a tamper detection and localization scheme for JP2 encoded images. The proposed scheme signs codestreams and detects tamper in the codestream domain. The scheme localizes tampered area in codeblocks.

III. THE PROPOSED SCHEME

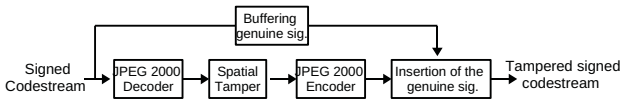
This section proposes a codestream domain authentication and tamper localization scheme for JP2 coded images. The proposed system is shown in Fig. 5. An original image is compressed by a standard JP2 encoder, and the compressed codestream is signed without being decompressed. A suspected codestream is examined in the codestream domain, and tamper localization is performed after the codestream is found inauthentic.

Tampers can be applied in the spatial, frequency, or codestream domains. Additionally, signature can be lost, preserved, or imitated in order to perform tampering. Hence, nine types of tampering can be applied. Figure 6 shows the types of tampering assumed in this paper: tampering in the spatial domain, tampering in the spatial domain with an imitated signature, tampering in the spatial domain preserving the signature, tampering in the frequency domain, tampering in the frequency domain with an imitated signature, tampering in the

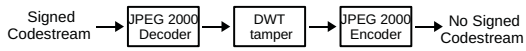


(a) Spatial Tamper.

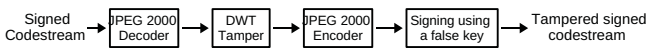
(b) Spatial Tamper using an imitated sign.



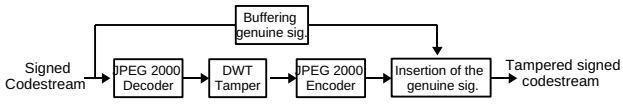
(c) Spatial Tamper preserving the original sign.



(d) DWT Tamper.



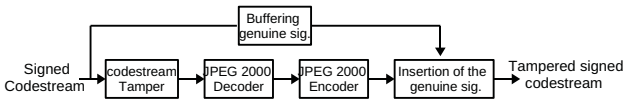
(e) DWT Tamper using an imitated sign.



(f) DWT Tamper preserving the original sign.

(g) Codestream Tamper.

(h) Codestream Tamper using an imitated sign.



(i) Codestream Tamper preserving the original sign.

Fig. 6. Type of Tampering Performed during Simulations.

frequency domain preserving the signature, tampering in the codestream domain, tampering in the codestream domain with an imitated signature, and tampering in the codestream domain preserving the signature. The proposed scheme countervails all these tampering types.

Subsequently three sections describe signing, authentication, and tamper localization mechanisms, respectively, and the last section summarizes the feature of the proposed scheme.

A. Signing

Figure 7 illustrates the concept of the signing in the proposed scheme. The proposed scheme applies a hash function to arithmetic encoded codeblock data in a codestream, and it encrypts the obtained hash values with a key.

The encrypted hashes are gathered to form a signature and the signature is inserted into the codestream. The proposed scheme inserts termination marker, that are two bytes value

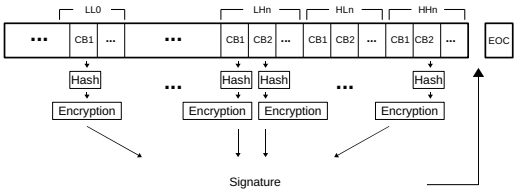


Fig. 7. Sign of Codestreams (CB denotes a codeblock data).

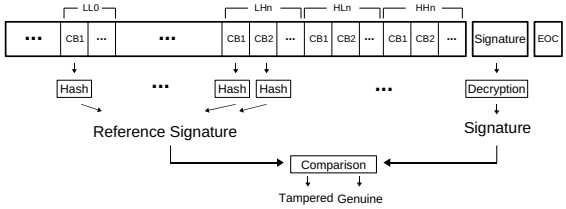


Fig. 8. Verification of Signature.

greater than 0xFF8F, and the encrypted signature after the last packet but before the EOC marker segment.

B. Authentication

The proposed scheme firstly search an inserted signature with a termination marker. When no signature is found, the proposed scheme inmediately determines that the codestream is inauthentic.

The codestream with the signature, then, is further examined as shown in Fig. 8. The proposed scheme extracts the inserted signature and decrypts it with the same key used in the signing process so that it can be compared with a reference signature. The reference signature is computed by hashing each of the codeblocks in the codestream. When the orignal and reference signatures are the same, the proposed scheme determines that the codestream is genuine.

Otherwise, the codestream is tampered, and the proposed scheme proceeds to the tamper localization process described in the next section.

C. Tamper Localization

For an inauthentic codestream with the signature, the proposed scheme separates the original and reference signatures according to its codeblocks. Then, the original hash value and its corresponding reference hash value are compared. Tampered regions are codeblocks that have their hash values mismatched.

Furthermore, the proposed scheme maps the encoded codeblocks, that are determined to be tampered, from the codestream domain to the DWT domain so that the proposed scheme localizes the tampered regions in the DWT domain without decompressing the codestream.

The proposed scheme further estimates the tampered areas in the spatial domain, without decoding the codestream, by mapping tampered codeblocks into the spatial domain using DWT filter information. Filter information for the synthesis

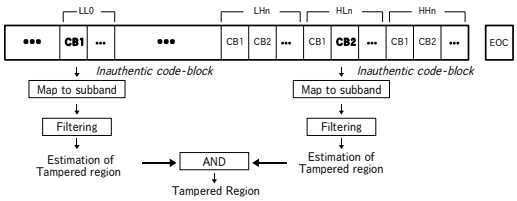


Fig. 9. Localization of Tampered Region.

filter is obtained in the codestream from either the coding style default (COD) or the coding style component (COC) marker segments, which are located in the main and/or tile headers, respectively. Combining the areas estimated by codeblocks belonging to several subband, the proposed scheme finally localizes the tampered areas. This procedure is summarized in Fig. 9.

D. Features

This section describes three main features of the proposed scheme; processing in the codestream domain, JP2 codestream structure compliant, and efficient spatial tamper localization ability.

1) Codestream Domain Processing: For processing JP2 coded images in the codestream domain, the proposed scheme focuses on the fact that each codeblock is arithmetic encoded individually in JP2 coding standard and the scheme takes into account the JP2 codestream structure.

The proposed scheme easily accesses a codestream segment corresponding to a target codeblock without decompressing the codestream. A hash function is also directly applied to the codestream segment to form a signature. The signature is directly inserted into the codestream according to the JP2 codestream structure. Consequently, the signing process of the proposed scheme runs in the codestream domain.

The authentication and tamper localization are also done in the codestream domain. The original signature is extracted from the codestream and a reference signature is directly obtained from the codestream. Tamper localization works in the codestream domain naturally. Mapping tampered codestream segments to inauthentic codeblocks in the DWT domain does not require to decode the codestream to the DWT domain. Moreover, spatial tamper localization using the mapped codeblocks and the filter information does not require codestream decoding.

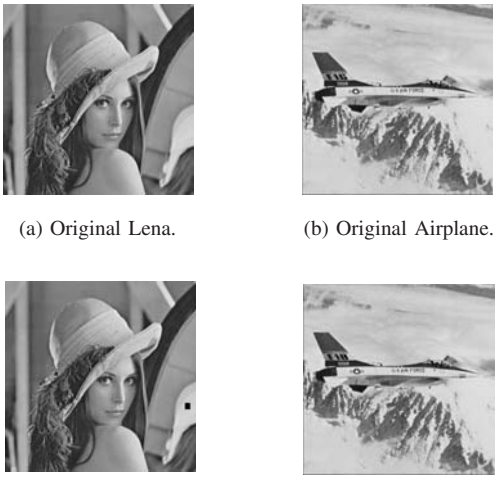
It is concluded that the proposed scheme works without decoding codestreams to the DWT or spatial domain for processing, in other words, it works in the codestream domain.

2) Codestream Structure Compliance: The proposed scheme takes the codestream structure for compliance with JP2 standard. The proposed scheme applies a hash function to the codestream segment corresponding to a codeblock, and it does not change the segment itself. A signature composed of hash values is inserted after the last packet but before the EOC marker segment. In the signature insertion, termination marker is also inserted just before the signature so that a standard JP2 decoder skips the signature rather than decoding it. That is, decoding a signed codestream with a standard decoder restores the original decoded image without any distortion.

Consequently, the proposed scheme is compliant with JP2 codestream structure.

3) Efficient Spatial Tamper Localization Ability: Based on codeblocks, the proposed scheme localizes the tampered areas in the spatial domain as well as in the DWT and codestream domain. A tampered codestream segment corresponding to a codeblock is mapped to the codeblock of a subband in the DWT domain. Thanks to the spatial-frequency analysis property of the DWT, the codeblock position in a subband corresponds to the position in the spatial domain. For same tampering, tamper localization using the highest resolution, i.e. LH_n, HL_n and HH_n, is enough because of its highest spatial resolution.

It is concluded that the proposed scheme has an efficient spatial tamper localization ability.



(a) Original Lena. (b) Original Airplane.

(c) Tampered Lena. (d) Tampered Airplane.

Fig. 10. Images for Experiments (original images are tampered by adding a black square in lena and replacing 6 with 8 in the vertical stabilizer of airplane).

IV. Experimental Results

The effectiveness of the proposed scheme is verified in this section from the perspective of tamper detection ability and tamper localization accuracy. Two of 256 levels of grayscale images with 512×512 pixels, Lenna and Airplane as shown in Fig. 10, are used for evaluation.

A standard JP2 encoder implemented by Kakadu v6.0 compresses two images with three levels of decomposition. Lossy compression is performed. The codestreams are signed using SHA-1 hash function [12], [13] and AES encryption [13] in cipher feedback block cipher mode. The signature lengths are summarized in Table II. The signature size depends on the number of codeblocks, in other words, it depends on the image size and codeblock.

TABLE II
SIGNATURE SIZE FOR LENA AND AIRPLANE

Codeblock size	No. of codeblocks	Signature Size (KB)
16×16	1024	20
32×32	256	5
64×64	64	1.25

(a) Lena (Codeblock Size 16×16).

(b) Airplane (Codeblock Size 16×16).

(c) Lena (Codeblock Size 64×64.

(d) Airplane (Codeblock Size 64×64.

Fig. 11. Localization Region of Tampered Images

Tamper types considered in this section are the same as shown in Fig. 6., codestreams are directly modified and the decoded images are altered. Figs. 10 (c) and (d) are images that are spatially tampered after decoding the codestreams. The tampered images are, then, encoded again by the same standard encoder with the same encoding parameters.

All genuine codestreams should have an appended signature in the proposed system, so the proposed scheme determines any unsigned codestreams inauthentic. Moreover, the proposed scheme also detected tampers for codestreams with a preserved signature and an imitation signature. It is noted that all nontampered codestreams were verified as authentic.

For inauthentic codestreams with a signature, the proposed scheme performs tamper localization. Tampered positions were localized in the codestream domain. Mapping tampered positions to the DWT domain localized tampered regions in the DWT domain. By using filter information, the tampered areas were localized in the spatial domain as shown in Fig. 11 for codestreams with a preserved signature. The proposed scheme localized tampered areas in codeblocks, the accuracy of localization depending on the codeblock size. Table III presents localization accuracy by showing pixels progagation of codeblock coefficients in the spatial domain. Small codeblock is good for localization as shown in Fig. 11, but it decreases coding efficiency. The codeblock size and the localization accuracy depend on applications.

TABLE III
TAMPER LOCALIZATON ACCURACY

Codeblock size	Subband		
	LH_3	HL_3	HH_3
16×16	37×39	39×37	39×39
64×64	133×135	135×133	133×133

V. CONCLUSIONS

This paper has proposed an authentication and tamper localization scheme for JPEG 2000 coded images. In the proposed scheme, signing, tamper detection, and tamper localization are done in the codestream domain, i.e., without decompressing the compressed codestream. The proposed scheme generates signed codestream compliant with the JP2 codestream structure, and a standard decoder is able to decompress the signed codestream without introducing any distortion to the decoded image. By using the positions of tampered codeblocks and the filter information, the proposed scheme localizes the tampered areas in the spatial domain without decopmressing the codestream.

REFERENCES

[1] S. Lion, D. Kanellopus and G. Ruffo, "Recent advances in multimedia information system security," *Informatica (Ljubljana)*, vol.33, pp.3–24, Mar. 2009.

[2] A. Haouzia and R. Noumeir, "Methods for image authentication: a survey," *Multimedia Tools and Applications*, vol.39, no.1, pp.1–46, Aug. 2008.

[3] C. Rey and J.L. Dugelay, "A survey of watermarking algorithms for image Authentication," *EURASIP J. Applied Signal Process.*, vol.2002, pp.613–621, Jun. 2002.

[4] I.J. Cox, M.L. Miller, J.A. Bloom, J. Fridrich, and T. Kalker, "Digital watermarking and steganography," 2nd ed., Morgan Kufmann, 2008.

[5] D.S. Taubman and M.W. Marcellin, "JPEG 2000: image compression fundamentals, standards and practice," Kluver Academic Publishers, 2002.

[6] P. Schelkens, A. Skodras, and T. Ebrahimi, "The JPEG 2000 suite," Wiley, 2009.

[7] S. Sun and Z. li, "Distributed block-dependent watermarking method for JPEG2000 image authentication," in *Proc. IEEE DBTA*, 2009, pp.119–122.

[8] H.H. Song, Y. Zhang, and C. Zou, "A semi-fragile wavelet transform watermarking scheme for content authentication of images," *J. Shanghai Jiaotong Univ. (Science)*, vol.14, pp.569–573, 2009.

[9] R. Grosbois, P. Gerbelot, and T. Ebrahimi, "Authentication and access control in the JPEG 2000 compressed domain," in *Proc. SPIE,* vol.4472, pp.95–104, 2001.

[10] A. Haggag, M. Ghoneim, J. Lu, and T.Yahagi, "Scalable authentication and nonrepudiation technique for JPEG 2000 images using JPSEC protection tools," *IEICE Trans. Fundamentals*, vol.E89-A, pp.2945–2954, Nov. 2006.

[11] Y. Wu, D. Ma, and R. H Deng, "Progressive protection of JPEG 2000 codestream," in *Proc. IEEE ICIP*, 2004.

[12] X. Wang, Y.L. Yin, and H. Yu, "Finding collisions in the full SHA-1," *LNCS*, vol.3621, 2005, pp.17–36.

[13] W. Trapple and L.C. Washington, "Introduction to criptography with coding theory," Pearson Prentice Hall, 2006.