HASH BASED IDENTIFICATION OF JPEG 2000 IMAGES WITH DIFFERENT CODING PARAMETERS

Osamu WATANABE†, Takahiro FUKUHARA , and Hitoshi KIYA‡*

†Takushoku University, Dpet. of Electronics & Computer Systems, 815–1, Tatemachi, Hachioji-shi, Tokyo, JAPAN
* Sony Corporation, CPDG PSG, 4–14–1, Asahi-cho, Atsugi-shi, Kanagawa, JAPAN
‡Tokyo Metropolitan University, Dept. of Info. and Commun. Systems Eng., 6–6 Asahigaoka, Hino-shi, Tokyo, JAPAN

ABSTRACT

A hash based identification method of JPEG 2000 images with different coding parameters is presented. The method correctly identifies the compressed JPEG 2000 images encoded even if there are difference in the coding parameters, such as coding-rate and quantization step sizes. A new algorithm is introduced to calculate hash values from the number of zero-bit-planes that can be extracted from the JPEG 2000 codestream by only parsing the header information. Because the proposed approach uses the hash values, the length of data to identify images is fixed. Using the fixed length data, processing speed of the proposed approach should be fast. This feature is not provided by conventional methods. The experimental results revealed the effectiveness of the processing speed of the proposed method.

Index Terms— JPEG 2000, Image identification, Hash function, Digital Cinema

1. INTRODUCTION

The use of digital images and video sequences has greatly increased recently because of the rapid growth of the Internet and multimedia systems. It is often necessary to identify a certain image in a database that has a large number of digital images [1–6] in various types of the applications of digital images/videos. The image database generally consists of images in compressed form to reduce the amount of data. Several international standards for searching still or moving images and retrieval systems have been developed [7–10] in connection to this. "Identification" in this work is defined as an operation for finding an image that is identical to a given original image from an image database.

JPEG 2000 [11] has been officially selected as the standard compression/decompression technology for digital cinema by the Digital Cinema Initiatives consortium [12]. There is need to identify a certain frame in some operations such as editing and re-encoding in digital cinema applications. The identification system used for digital cinema systems must be able to handle a large number of frames encoded with JPEG 2000 in a sufficiently short processing time. Several methods have been developed for identifying compressed images [1,2,5]. The method described in Ref. [1] is for JPEG images and uses the signs of the discrete cosine transform (DCT) coefficients of the images. One method for JPEG 2000 [2] , uses the signs of the discrete wavelet transform (DWT) coefficients. An algorithm for both JPEG 2000 and JPEG was proposed in Ref. [5]. Although these methods are for compressed images, they use transformed coefficients, which are not available without decoding.

Codestream-based identification methods for JPEG 2000 images [3, 4, 13] have also been proposed. These methods achieve fast and precise image identification because they use

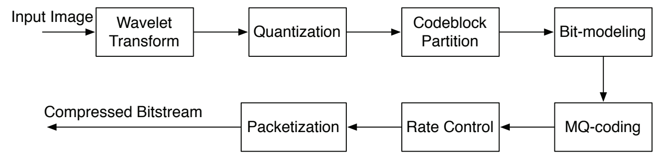


Fig. 1. Block diagram of JPEG 2000 encoder

the number of zero-bit-planes (NZBP), which is obtained by only parsing the header part of a JPEG 2000 codestream, as the features for the identification. The methods work well for the images which have the exactly same parameters of JPEG 2000 coding, however, the difference in the parameters causes the error of the identification [3, 4]. The identification error occurs with the fact that the values of the NZBP are not equal between two codesreams (they have the same original image) with the different coding parameters. This problem is solved with the method written in Ref. [13].

Another problem is revealed that the length of feature, which is the number of zero-bit-planes, increases with increasing number of code-blocks. In digital cinema [12], a frame has the resolution of $4,000 \times 2,000$ and its JPEG 2000 codestream has over 20,000 code-blocks with typical size of code-block. The length of feature per code-block is 36 bit in Ref. [13]. Therefore the length of feature for a frame is over 720k bit. It is necessary to shorten the length of feature for faster processing.

A new algorithm for calculation of hash values from the number of zero-bit-planes is introduced to accelerate the processing speed of the identification and a hash based identification method of JPEG 2000 images with different coding parameters is presented in this paper. The algorithm is designed to be robust against the changes of the inputs caused by the difference of the coding parameters and to be sensitive to the difference of the original image. Using the hash values, the length of feature data to identify images for the proposed approach is fixed. The processing speed of the proposed approach should be fast because the length of feature does not depend on the number of code-blocks. The experimental results of image identification based on the proposed approach are presented. The results revealed the effectiveness of the proposed method.

2. NUMBER OF ZERO-BIT-PLANES IN JPEG 2000

2.1. Definition

Fig. 1 has a block diagram of the JPEG 2000 encoder. JPEG 2000 uses a bit-plane coding architecture summarized as follows. As outlined in Fig. 2, quantized DWT coefficients are represented in sign-magnitude form. The sign bit-plane is at the MSB level, and the magnitude bit-planes are beneath it.

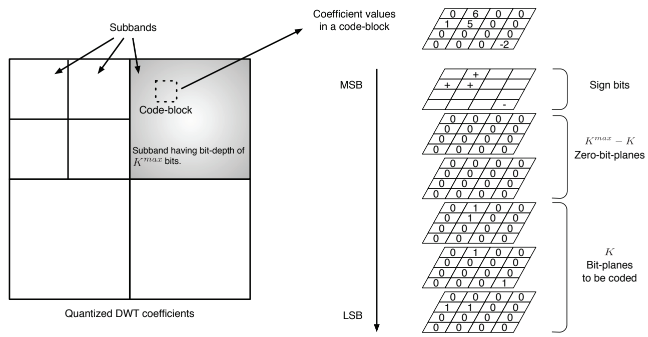


Fig. 2. Bit-plane decomposition and sign-magnitude representation of DWT coefficients in a code-block: A zero-bit-plane is a special bit-plane in which samples are all zeros. Zero-bit-planes are arranged from the MSB to the LSB level.

The number of samples in a bit-plane is equal to that in the code-block, and all the samples in a bit-plane are either 0 or 1. Let the K^{max} denotes the number of bits required to represent all quantized coefficients in a subband. The block coder for JPEG 2000 first determines the number of bits, $K \leq K^{max}$, that are needed to represent the quantized magnitudes. The encoder ideally finds the smallest such K. The difference between $K^{max} - K$ is called "number of zero-bit-planes" and defined as follows.

$$K^{msbs} = K^{max} - K \tag{1}$$

The NZBP, K^{msbs}, represents the number of most significant magnitude bits that is skipped to encode with the encoder. The decoder will take this to be zero for all samples. The remaining K magnitude bits must be explicitly coded. A code-block which all the bit-planes are zero-bit-planes in the JPEG 2000 standard is defined as "not included" because the code-block does not have any data to be encoded.

2.2. Effect of changes in coding parameters on NZBP

Since the coding-rate in JPEG 2000 is normally controlled by discarding MQ-encoded codestreams from LSB to MSB, there is fundamentally no effect from the NZBP even if the coding rate changes. However, the numbers of zero-bit-planes in images compressed with different quantization step sizes may differ, even if the compressed images are generated from one original image and the other coding parameters are the same. The NZBP is part of the header information of JPEG 2000 codestreams. This information is easily obtained by parsing the main header of JPEG 2000without heavy EBCOT decoding.

3. PROPOSED METHOD

A new identification method for JPEG 2000 images which uses hash values calculated by number of zero-bit-planes in a subband is proposed in this section. Note that "Identification" in this work means an operation for finding an image that is identical to a given original image. The proposed method consists of three parts. The first part defines order of scanning code-blocks in a subband. An algorithm to produce hash values following the scan order is described in the second part. The goal of the algorithm is to be robust against the changes of the inputs caused by the difference of the coding parameters and to be sensitive to the difference of the original image.

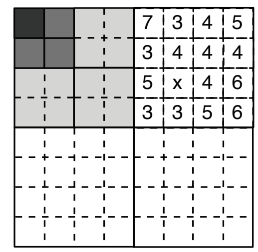


Fig. 3. Definition of subband, code-block, where DWT resolution level is 3. Solid line means boundaries of subbands and Dashed line means boundaries of code-blocks. Subbands filled by the same gray level are in the same DWT level. The NZBP is on the code-block. Symbol "x" means "not included."

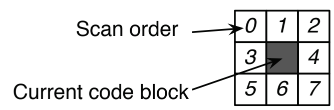


Fig. 4. Scan window for current code-block which has 8 neighbors. Current code-block is filled in gray and each number means scan order for neighboring code-blocks.

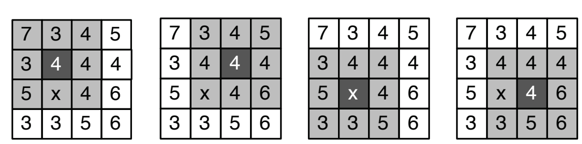
The last part compares the hash values of a pair of JPEG 2000 images.

3.1. Definition of scan order

Fig. 3 shows an example of code-blocks, subbands, and DWT levels for JPEG 2000 images. For simplicity, the NZBP of code-blocks in only one subband is depicted in Fig. 3. For a current code-block in a subband, a scan window is defined as Fig. 4. The current code-block is filled in gray and numbers are the order of scanning. Letting q_n denotes the scan order, it is written as:

$$q_n = n, \; n = 0, 1, \cdots, 7. \tag{2}$$

This window is defined for code-blocks which has its 8 neighbors. In other words, this window is not defined for code-blocks which lie the boundaries of a subband. The number of code-blocks having the scan window is written as M. The NZBP of neighboring code-blocks is scanned following this order. The examples of the code-block scanning in a subband are shown in Fig. 5. Letting M denotes the number of code-blocks having the scan window, $M = 4$. The scan window moves in raster order (Fig. 5(a)→Fig. 5(b)→Fig. 5(c)→Fig. 5(d)).



(a) 1st scan (b) 2nd scan (c) 3rd scan (d) 4th scan

Fig. 5. Example of code-block scanning in a subband. The number of code-blocks having the scan window is four ($M = 4$). The window is moves in raster order (1st→2nd→3rd→4th).

3.2. Algorithm to produce hash values

Step1. With the scanned value of the NZBP, a vector is formed for a current code-block as follows.

$$\boldsymbol{v} = \{(p_0, q_0), (p_1, q_1), \cdots, (p_7, q_7)\}. \quad (3)$$

An element of $\boldsymbol{v}$ consists of a ordered pair (p_n, q_n) and let v_n denotes the element. p_n means the value of the NZBP for the code-block scanned in the q_n th order.

Step2. Next, the elements of $\boldsymbol{v}$ are sorted in ascending order. Note that there are two sort keys. The primary sort key is the value of p_n and the secondary sort key is the value of q_n. The sorted version of $\boldsymbol{v}$ denoted by $\boldsymbol{v'}$:

$$\boldsymbol{v'} = \{(p'_0, q'_0), (p'_1, q'_1), \cdots, (p'_7, q'_7)\}, \ p'_n \leq p'_{n+1}. \quad (4)$$

Step3. Letting i means the index of code-blocks having the scan window, a binary string $\boldsymbol{s_i}$ is formed with the first N elements of $\boldsymbol{v'}$:

$$\boldsymbol{s_i} = c_0 c_1 \cdots c_7,$$
$$c_n = \begin{cases} 11 & n = q'_k, \ k < N \\ 00 & otherwise, \end{cases} \quad (5)$$

Step4. The hash value of a subband b belongs to DWT resolution level r is defined as

$$H_{r,b} = h(s_0 s_1 \cdots s_{M-1}), \quad (6)$$

where $h(\cdot)$ means a hash function. Note that any hash functions with fixed-length outputs (such as MD5, SHA-1, etc.) are applicable here.

For example, the vector $\boldsymbol{v}$ for Fig. 5(a) is formed as follows.

$$\boldsymbol{v} = \{(7, 0), (3, 1), (4, 2), (3, 3), (4, 4), (5, 5), (x, 6), (4, 7)\} \quad (7)$$

The vector $\boldsymbol{v}$ is sorted in ascending order.

$$\begin{aligned} \boldsymbol{v'} &= \{(3, 1), (3, 3), (4, 2), (4, 4), (4, 7), (5, 5), (7, 0), (x, 6)\} \\ &= \{(p_1, q_1), (p_3, q_3), (p_2, q_2), (p_4, q_4), \\ &(p_7, q_7), (p_5, q_5), (p_0, q_0), (p_6, q_6)\}. \end{aligned} \quad (8)$$

For the value of $N = 2$, the string $\boldsymbol{s}_0$ is formed as

$$\boldsymbol{s}_0 = 00\ 11\ 00\ 11\ 00\ 00\ 00\ 00 \quad (9)$$

In a similar way, $\boldsymbol{s}_1$, $\boldsymbol{s}_2$ and $\boldsymbol{s}_3$ are obtained. Finally, the hash value for subband b is derived as

$$H_{r,b} = h(s_0 s_1 s_2 s_3) \quad (10)$$

3.3. Hash-based identification

The hash values are calculated for all subbands and attached as extra information of a JPEG 2000 image. The image database is built with these hash-attached JPEG 2000 images. Let I^Q denotes a query image and I^D denotes a database image. Note that it is assumed that I^Q is also encoded as JPEG 2000 image. An identification system based the proposed algorithm takes DWT resolution level r as its parameter and the system returns a decision whether or not I^Q is identical with I^D. The decision D is made by following equations:

$$D = \begin{cases} Negative & H^Q_{r,b} \neq H^D_{r,b} \text{ for all } b \\ Positive & otherwise, \end{cases} \quad (11)$$

where b is the index of the subband belongs to a given DWT resolution level r.

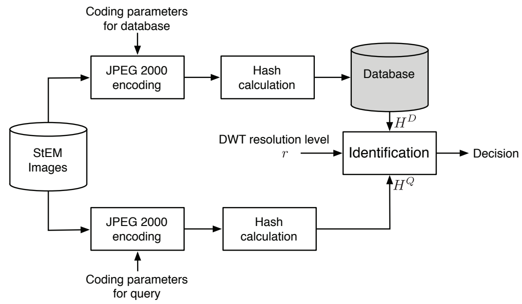


Fig. 6. Procedure used in identification experiment

Table 1. Specifications for standard evaluation material

Number of frames	14,964
Spatial resolution	4,096 (H)× 1,740 (V)
Color format	RGB(4:4:4) 12 bits/component

4. EXPERIMENTAL RESULTS

The performance of the identification methods in terms of their precision and processing speed of the image identification were evaluated to verify the effectiveness of the proposed method.

4.1. Conditions and Procedure

The Standard Evaluation Material (StEM) [14] was used as test sets. The original StEM includes 17,239 frames; however, visually full black frames were rejected from the test sequence used in the experiments. As a result, the test sequence contained 14,964 frames. The specifications for the test sequences are summarized in Table 1. First, sequences of JPEG 2000 compressed images were built for the database of hash values. The encoding parameters for the sequences are listed in Table 2. Kakadu [15] version 6.4.1 was used as a JPEG 2000 codec. The hash values for all the encoded frames in the sequences were calculated by the proposed algorithm in offline and the calculated hash values were stored into a database. Hash values for a query image were calculated followed by JPEG 2000 encoding with the coding parameters for query in Table2. Note that base step size of quantization for query was different from that for database. To confirm that the proposed method correctly identifies the compressed frames with the difference of the quantization step sizes, identification experiments were carried out for all possible combinations of the codestreams. The number of the combinations was $14,964 \times 14,964$. The parameters for the identification, such as DWT resolution level r and N are set to $r = 2, 3, 4, 5$ and $N = 2$. Because there were no code-blocks having 8 neighbors in any subband, $r = 0$ and 1 were excluded from the experiment. Fig. 7 outlines the procedure for these experiments.

4.2. Results and remarks

(A) Performance of identification

The number of true-positive (TP), true-negative (TN), false-positive (FP), and false-negative (FN) decision obtained from the experiment is shown in Tab. 3. This table also has the false-positive-rate (FPR) and true-positive-rate (TPR) [16] with the results. From Tab. 3 and Fig. 7, the result with DWT resolution level $r = 2$ is the best. The result with $r = 2$ is a

Table 2. JPEG 2000 encoding parameters

Codec	Kakadu version 6.4.1
Coding-rate (VBR)	174 Mbps
DWT Filter	9×7
DWT Level	5
Base step size (for database)	1/128
Base step size (for query)	1/256
Code-block size	32×32
Tile decomposition	No

Table 3. Number of true-positive (TP), true-negative (TN), false-positive (FP), and false-negative (FN) results obtained from the experiment.

r	TP	TN	FP	FN	FPR	TPR
2	14964	211554472	12351860	0	0.055	1.0
3	14947	211932683	11973649	17	0.053	0.99
4	14436	131341757	92564575	528	0.41	0.97
5	12983	75317792	148588540	1981	0.66	0.87

match for that of the conventional method [13]. An Receiver operating characteristic (ROC) graph [16] built with the FPRs and the TPRs is shown in Fig. 7. ROC graphs depict relative trade-offs between benefits (true-positives) and costs (false-positives). Informally, one point in ROC space is better than another if it is the northwest (TPR is higher, FPR is lower, or both) of the first.

Note that the base step size (1/256 for query images and 1/128 for database images) is the Kakadu parameters used for scalar quantization. As the actual quantization step size is determined based on the base step size, the change in the base step size means a change in the quantized coefficients, that is, the NZBP will change; Therefore, It can be confirmed that identification based on the proposed method could identify the query frames regardless of the difference in base step size.

(B) Complexity of Identification

The previous reported method [13] forms a relation vector from the NZBP. Although the procedure to form this vector is very simple, there is a problem that the data length of the vectors used for identification is not fixed. The length depends on the total number of code-blocks in a JPEG 2000 codestream. If the size of image becomes large, the number of code-blocks will increase. For example, with 32×32 sized code-blocks, the number of code-blocks are over 20,000 for an image has its resolution of $4,000 \times 2,000$. The length of feature for a frame is over 720k bit in Ref. [13] because the length of feature per code-block is 36 bit.

By contrast, the proposed method does not use the NZBP directly. By applying a hash function, the length of feature for a subband is fixed. For example, with the MD5 hash function, the length of feature for a subband is 128 bit. With an image decomposed 5 level DWT, there are 16 subbands. Thus, the total length of feature is $128 \times 16 = 2,048$ bit. Clearly, the space complexity of the proposed method is relatively smaller than that of the conventional method.

5. CONCLUSION

A hash based identification method of JPEG 2000 images with different coding parameters has been presented in this paper. A new algorithm for installing hash function to codestream-based identification has been described. The proposed method based on the algorithm has the following features. One is that it correctly identifies the JPEG 2000 codestreams even if there are difference of the coding parameters. The other is the fixed length of features. In the most cases, the length of feature is shorter than that of the conventional method. The experimental results confirmed that the performance of the proposed method is a match for that of the conventional method with much shorter length of the identification features.

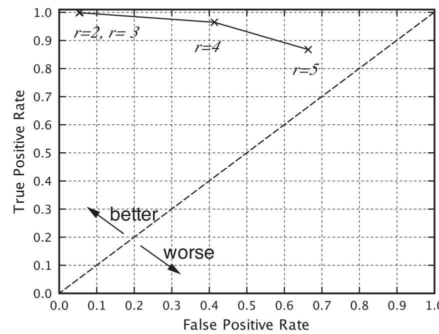


Fig. 7. ROC graph of identification experiments: Each point was depicted with the values of FPR and TPR in Tab. 3 (Note that the values FPR and TPR for $r = 2$ and $r = 3$ were very close). All points obtained with the results in Tab. 3 are in the "better" area. The dashed-line in the figure means the performance of random guess.

6. REFERENCES

[1] F. Arnia, I. Iizuka, M. Fujiyoshi and H. Kiya, "Fast and Robust Indentification Methods for JPEG Images with Various Compression Ratios," in *Proc. ICASSP 2006*, May 2006, vol. 2, pp. 397–400.

[2] O. Watanabe, A. Kawana and H. Kiya, "An Identification Method for JPEG2000 Images Using the Signs of DWT Coefficients," *Technical Report of IEICE*, vol. 106, no. IE2006-217, pp. 177–181, Jan. 2007.

[3] T. Fukuhara, K. Hosaka and H. Kiya, "Accurate identifying method of JPEG2000 images for digita cinema," in *Proceedings of The 14th International Multimedia Modeling Conference (MMM'08)*. Lecture Notes in Computer Science, Jan. 2008, vol. 4903, pp. pp.380–390.

[4] T. Fukuhara, K. Hosaka and H. Kiya, "Identifying method of JPEG2000 images in the codestream level for digital cinema (in Japanese)," *IEICE Trans.*, vol. J91-D, no. 9, pp. 2305–2313, 2008.

[5] K. O. Cheng, N. F. Law, W. C. Siu, "A fast approach for identifying similar features in retrieval of JPEG and JPEG 2000 images," in *Proc. APSIPA ASC 2009*, Oct. 2009, number MP-P2-3.

[6] Y. Uchida and S. Sakazawa, "Near-Duplicate Video Detection Considering Temporal Burstiness of Local Features," in *Proc. The 2011 IEICE General Conference*, Mar. 2011 (in Japanese), number D-12-93, p. 196.

[7] "Information technology – JPSearch – Part 1: System framework and components," International Standard ISO/IEC TR-24800-1, Dec. 2007.

[8] "Compact Descriptors for Visual Search: Applications and Use Scenarios," ISO/IEC JTC1/SC29/WG11/N11529, July 2010.

[9] "Compact Descriptors for Visual Search: Context and Objectives," ISO/IEC JTC1/SC29/WG11/N11530, July 2010.

[10] "Compact Descriptors for Visual Search: Requirements," ISO/IEC JTC1/SC29/WG11/N11531, July 2010.

[11] "Information technology — JPEG 2000 image coding system – Part 1: Core coding system," International Standard ISO/IEC IS-15444-1, Dec. 2000.

[12] "Digital Cinema System Specification V1.2," Digital Cinema Initiatives, LLC Technology Committee, Mar. 2008.

[13] O. Watanabe, T. Fukuhara and H. Kiya, "Fast Identification of JPEG 2000 Images for Digital Cinema Profiles," in *Proc. ICASSP 2011*, May 2011, pp. 881–884.

[14] Digital Cinema Initiatives, LLC Technology Committee, "StEM Access Procedures," http://www.dcimovies.com/StEM/, Sept. 2010.

[15] "Kakadu software," http://www.kakadusoftware.com/.

[16] Tom Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861 – 874, 2006.