Fast Accurate Identifying Method of JPEG 2000 Images with Different Coding Parameters for Digital Cinema

Osamu WATANABE* and Takahiro FUKUHARA† and Hitoshi KIYA‡

* Takushoku University, Dpt. of Electronics & Computer Systems, 815–1, Tatemonchi, Hachioji-shi, Tokyo, JAPAN
† Sony Corporation, B2B Solution Dept., 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—This paper proposes a method for identifying JPEG 2000 images with different code-block sizes and resolution levels. The proposed method does not produce false negative matches regardless of the compression rate, code-block size, or Discrete Wavelet Transform (DWT) resolution levels, or quantization step sizes. This feature is not obtained in conventional methods. Moreover, the proposed method is fast because it uses the number of zero-bit-planes extracted without Embedded Block Coding with Optimized Truncation (EBCOT) decoding, which requires a large amount of calculation time. The experimental results show the effectiveness of the proposed method.

I. INTRODUCTION

Recently, the use of digital images and video sequences has greatly increased 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 amount of digital images. “Identification” is defined as an operation for finding from an image that is identical to a given original image among compressed images in this paper. Database images are generally stored in compressed forms to reduce the amount of data.

JPEG 2000 [1] was officially selected as the standard compression/decompression technology for digital cinema by the Digital Cinema Initiatives (DCI) consortium [2], and standards for identifying and retrieving JPEG 2000 images are now being developed [3]. The identification system used for digital cinema systems must be able to handle a large number of JPEG 2000-encoded frames in a sufficiently short processing time.

Several methods have been developed for identifying compressed images [4]–[6]. The method described in [4] is for JPEG images and uses the signs of the discrete cosine transform (DCT) coefficients. One for JPEG 2000 [5], uses the signs of the discrete wavelet transform (DWT) coefficients. For both JPEG 2000 and JPEG, an algorithm was proposed in [6]. Although these methods are for compressed images, they use transformed coefficients, which are not available without decoding. Code-stream-based identification methods for JPEG 2000 images [7], [8] have been proposed. Code-stream-based means that there is no need for EBCOT decoding, which is the most time consuming process in a JPEG 2000 decoder; therefore they feature fast identification.

The problem of these code-stream-based methods are that they cannot identify images compressed with different coding parameters of JPEG 2000, such as code-block sizes, DWT resolution levels, and quantization step sizes. This disadvantage is not trivial because the most simple and reasonably fast way to identify the JPEG 2000 images which have completely the same coding parameters is binary comparison of the compressed data.

To solve it, a novel method for identifying JPEG 2000 images with different code-block sizes, DWT resolution levels, and quantization step sizes is proposed in this paper. The proposed method runs fast since the method uses the number of zero-bit-planes, which is obtained by parsing only the header part of JPEG 2000 codestream. In addition, the proposed method does not produce false negative matches, regardless of differences on the JPEG 2000 coding parameters. Simple binary comparison of the compressed data cannot achieve this feature.

II. BRIEF OVERVIEW OF JPEG 2000 CODE-STREAM

Figure 1 shows a block diagram of a JPEG 2000 encoder and Figure 2 shows an example of an image I analyzed using DWT. \( R^I \) denotes the number of DWT resolution levels, which is three in Fig. 2, and \( r^I = 1, 2, ..., R^I \) is the index of the resolution level. \( b \) denotes the index of subband, where 1, 2, 3, and 4 are for LL, HL, LH, and HH, respectively. The subbands are divided into \( X^I \times Y^I \)-sized code-blocks, and \( S^I(r^I) \times
where, \( S_{\min}(l) = \min_{I \in \{Q, D\}} S_{I}(l) \) and \( T_{\min}(l) = \min_{I \in \{Q, D\}} T_{I}(l) \).

Each quantized coefficient is separated into its sign and absolute magnitude, as shown in Fig. 3, where the absolute magnitudes are factorized as bit-planes from the Most Significant Bit (MSB) to the Least Significant Bit (LSB). All the samples in the bit-planes are either zero or one.

A special bit-plane, in which all samples are zero, frequently appears, as shown in Fig. 3. This special bit-plane is referred to as a “zero-bit-plane.” When all bit-planes are zero-bit-planes, as shown as the third code-block in Fig. 3, the code-block is defined as “not included,” because it does not include any data to be encoded. It should be noted that the numbers of zero-bit-planes in images compressed with different quantization step sizes may be different, even if the compressed images are generated from one original image.

Hereafter, \( Z_{t,b}^{r}(s,t) \) represents the number of zero-bit-planes in the code-block at \((s,t)\), where \( s = 1, 2, \ldots, S_{t}^{r}(r) \) and \( t = 1, 2, \ldots, T_{t}^{r}(r) \). Note that the number of zero-bit-planes in a code-block can be obtained by only parsing the packet header, i.e., without heavy EBCOT decoding.

### III. Proposed Method

The proposed method consists of three algorithms. The first algorithm defines regions to be compared. In the second algorithm, the numbers of zero-bit-planes in the compared regions are derived. The third algorithm compares the query and database images by using the numbers of zero-bit-planes in the regions.

It is assumed that query image \( Q \) has \( X_{Q} \times Y_{Q} \)-sized code-blocks and is decomposed to \( R_{Q} \) resolution levels, whereas database image \( D \) has \( X_{D} \times Y_{D} \)-sized code-blocks and consists of \( R_{D} \) resolution levels.

#### A. Definition of Regions to Be Compared

To normalize or unify the regions to be compared, this algorithm defines the size of compared regions and the compared depth of resolution levels as shown in Fig. 4. The algorithm is as follows:

1. By Eqs. (1) and (2), the numbers of code-blocks in a compared region are derived.

   \[
   (M_{Q}, M_{D}) = \begin{cases} 
   \left(\frac{X_{Q}}{M}, \frac{Y_{Q}}{M}\right) & X_{Q} \leq X_{D} \\
   \left(1, \frac{Y_{Q}}{M}\right) & \text{others} \end{cases}
   \]

   \[
   (N_{Q}, N_{D}) = \begin{cases} 
   \left(\frac{Y_{Q}}{N}, 1\right) & Y_{Q} \leq Y_{D} \\
   \left(1, \frac{Y_{Q}}{N}\right) & \text{others} \end{cases}
   \]

2. The compared depth of resolution levels is given by Eq. (3).

   \[
   L = \begin{cases} 
   R_{Q} & R_{Q} = R_{D} \\
   R_{Q} - 1 & R_{Q} < R_{D} \\
   R_{Q} - 1 & R_{Q} > R_{D} 
   \end{cases}
   \]

   where \( L \) is the compared depth. It is noted that the comparable depth decreases by one without decoding under the condition that \( R_{Q} \neq R_{D} \). The number of the compared regions, \( c \), is given as

   \[
   c = \begin{cases} 
   S_{\min}(L)^{T_{\min}(L)} + \sum_{b=2}^{L-1} \sum_{l=1}^{b-1} S_{\min}(l)^{T_{\min}(l)} & R_{Q} = R_{D} \\
   \sum_{b=2}^{L-1} \sum_{l=1}^{b-1} S_{\min}(l)^{T_{\min}(l)} & \text{others} \end{cases}
   \]

   where, \( S_{\min}(l) = \min_{I \in \{Q, D\}} S_{I}(l) \) and \( T_{\min}(l) = \min_{I \in \{Q, D\}} T_{I}(l) \).
B. Derivation of the Numbers of Zero-Bit-Planes

This algorithm derives the number of zero-bit-planes in all compared regions, i.e., in each compared region in subband $b$ at $l$-th resolution level, where $b = 1, 2, 3,$ and $4$, and $l = 1, 2, ..., L$. The algorithm is as follows:

1. $l \leftarrow 1$.
2. $b \leftarrow 2$. If $R^Q = R^D$ and $l = L$, then $b \leftarrow 1$.
3. $u_1 \leftarrow 1$.
4. $v_1 \leftarrow 1$.
5. The number of zero-bit-planes at the $(u_1, v_1)$-compared region in the $b$-th subband of the $l$-th resolution levels of image $I$, $W^l_{I,b}(u_1, v_1)$, is given as

$$W^l_{I,b}(u_1, v_1) = \min_{m^l, n^l} Z^l(u_1-1) M^l + m^l, (v_1-1) N^l + n^l$$

where $m^l = 1, 2, ..., M^l$, $n^l = 1, 2, ..., N^l$.

It should be noted that Eq. (5) should satisfy

$$1 \leq (u_1-1)M^l + m^l \leq S^l(l), 1 \leq (v_1-1)N^l + n^l \leq T^l(l).$$

Note that if at least one $Z^l_{I,b}(u_1-1) M^l + m^l, (v_1-1) N^l + n^l$ among $m^l$'s and $n^l$'s is “not included,” then $W^l_{I,b}(u_1, v_1)$ is also defined as “not included,” as shown in Fig. 5.

6. $v_1 \leftarrow v_1 + 1$. If $v_1 \leq T^{\min}(l)$, then go to Step 5.
7. $u_1 \leftarrow u_1 + 1$. If $u_1 \leq S^{\min}(l)$, then go to Step 4.
8. $b \leftarrow b + 1$. If $b = 3$ or $b = 4$, then go to Step 3. Otherwise, go to Step 2.
10. The numbers of zero-bit-planes among all $c$ compared regions are obtained.

C. Image Identification For Different Quantization Step Size

According to differences of the quantization step size between query and database, the proposed method switches two sub algorithms. That is, for images having the same quantization step size based on Fukuhara et al. [7], [8], and for those having different step sizes. In the following, the sub algorithm for images having different step sizes is described.

This sub algorithm uses the relation among the focused-compared region and $K$ of its neighboring regions to compare images, as shown in Fig. 6. This sub algorithm consists of two parts: relation derivation and image identification. The former part is as follows:

1. Define the neighboring regions by the coordinate difference. That is,

$$g = (g_1, g_2, ..., g_K), h = (h_1, h_2, ..., h_K),$$

where $g_k$ and $h_k$ indicate the horizontal and vertical distances between the focused region and the $k$-th neighboring region, respectively, c.f., Fig. 6.

2. $l \leftarrow 1$.
3. $b \leftarrow 2$. If $R^Q = R^D$ and $l = L$, then $b \leftarrow 1$.
4. $u_1 \leftarrow 1$.
5. $v_1 \leftarrow 1$.
6. $k \leftarrow 1$.
7. Relation between the focused and the $k$-th neighboring regions, $e^Q_{I,b,k}$ is

$$e^Q_{I,b,k}(u_1, v_1) = \begin{cases} 1 & W^l_{I,b}(u_1, v_1) - W^l_{I,b}(u_1 + g_k, v_1 + h_k) > 0 \\ 0 & W^l_{I,b}(u_1, v_1) - W^l_{I,b}(u_1 + g_k, v_1 + h_k) = 0 \\ -1 & W^l_{I,b}(u_1, v_1) - W^l_{I,b}(u_1 + g_k, v_1 + h_k) < 0 \end{cases}$$

Note that if either $W^l_{I,b}(u_1, v_1)$ or $W^l_{I,b}(u_1 + g_k, v_1 + h_k)$ is “not included,” and if either $u_1 + g_k \leq 0, u_1 + g_k > S^{\min}(l), v_1 + h_k \leq 0$, or $v_1 + h_k > T^{\min}(l)$, then $e^Q_{I,b,k} = 0$.

8. $e^Q_{I,b,k}(u_1, v_1) = e^Q_{I,b,k}(u_1, v_1), e^Q_{I,b,k}(u_2, v_1), ..., e^Q_{I,b,k}(u_1, v_1)$ is derived. $k \leftarrow k + 1$. If $k < K$, then go to Step 7.
9. $v_1 \leftarrow v_1 + 1$. If $v_1 \leq T^{\min}(l)$, then go to Step 6.
10. $u_1 \leftarrow u_1 + 1$. If $u_1 \leq S^{\min}(l)$, then go to Step 5.
11. $b \leftarrow b + 1$. If $b = 3$ or $b = 4$, then go to Step 4. Otherwise, go to Step 12.
12. $l \leftarrow l + 1$. If $l < L$, then go to Step 3. Otherwise, go to Step 13.
13. The relations of $c$-compared regions are obtained. The latter part (Fig. 7) is as follows:

1. $l \leftarrow 1$.
2. $b \leftarrow 2$. If $R^Q = R^D$ and $l = L$, then $b \leftarrow 1$.
3. $u_1 \leftarrow 1$.
4. $v_1 \leftarrow 1$.
5. $k \leftarrow 1$.
6. If $e^D_{I,b,k}(u_1, v_1) \neq 0$ and $e^D_{I,b,k}(u_1, v_1) \neq 0$, then proceed to Step 7. Otherwise, go to Step 8.
7. If $e^D_{I,b,k}(u_1, v_1) = e^D_{I,b,k}(u_1, v_1)$, then proceed to Step 8. Otherwise, $D$ is determined as different from $Q$.
8. $k \leftarrow k + 1$. If $k < K$, then go to Step 6.
9. $u_1 \leftarrow u_1 + 1$. If $u_1 \leq T^{\min}(l)$, then go to Step 5.
10. $u_1 \leftarrow u_1 + 1$. If $u_1 \leq S^{\min}(l)$, then go to Step 4.
11. $b \leftarrow b + 1$. If $b = 3$ or $b = 4$, then go to Step 3. Otherwise, go to Step 12.
12. $l \leftarrow l + 1$. If $l < L$, then go to Step 2. Otherwise, $D$ is defined as $Q$. 

300
TABLE I
TEST SEQUENCE AND JPEG 2000 ENCODING PARAMETERS FOR QUERY IMAGES

<table>
<thead>
<tr>
<th>Test sequence</th>
<th>StEM(DCI standard), 14,964 frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>5,096(H) × 1,764(V)</td>
</tr>
<tr>
<td>Format</td>
<td>RGB(4:4:4) 12bits/component</td>
</tr>
<tr>
<td>Rate Control</td>
<td>VBR(Variant Bit Rate)</td>
</tr>
<tr>
<td>Code-stream</td>
<td>DCT Compliant(JPEG 2000 Part-1)</td>
</tr>
<tr>
<td>DWT Level</td>
<td>5</td>
</tr>
<tr>
<td>Base Step Size</td>
<td>1/256</td>
</tr>
<tr>
<td>Code-block Size</td>
<td>32 × 32</td>
</tr>
</tbody>
</table>

IV. EXPERIMENTAL RESULTS

D. Features

The features of the proposed method is as follows. (A) Robustness against the difference in coding parameters: the algorithm described in 3.1 makes it possible to identify images having different code-block sizes and/or DWT resolution levels. Images having different quantization step sizes can be identified with the algorithm described in 3.3. These algorithms are easily combined. Moreover, the method does not produce false negative matches regardless of the differences in quantization step size, the code-block size, and the number of resolution levels between images. This feature cannot be obtained by the conventional methods. (B) Fast processing: because the number of zero-bit-planes can be extracted without EBCOT decoding, the proposed method is very fast.

CONCLUSION

A novel zero-bit-plane-based identification method for JPEG 2000 images with different JPEG 2000 coding parameters has been proposed in this paper. The proposed method is fast since it uses the extracted number of zero-bit-planes without EBCOT decoding. Moreover, the method does not produce false negative matches, regardless of the compression rate, code-block size, and/or DWT resolution levels.

REFERENCES