An Image Retrieving Scheme Using Salient Features and Annotation Watermarking

Jenq-Haur Wang[1], **Chuan-Ming Liu**[1], **Jhih-Siang Syu**[1] **and Yen-Lin Chen***[1]
[1] Department of Computer Science and Information Engineering,
National Taipei University of Technology, Taipei 106, Taiwan, ROC.
[e-mail: jhwang@csie.ntut.edu.tw, cmliu@csie.ntut.edu.tw, taurus870tom@yahoo.com.tw,
ylchen@csie.ntut.edu.tw]
*Corresponding author: Yen-Lin Chen

Received August 23, 2013; revised November 22, 2013; accepted December 16, 2013; published January 29, 2014

Abstract

Existing image search systems allow users to search images by keywords, or by example images through content-based image retrieval (CBIR). On the other hand, users might learn more relevant textual information about an image from its text captions or surrounding contexts within documents or Web pages. Without such contexts, it's difficult to extract semantic description directly from the image content. In this paper, we propose an annotation watermarking system for users to embed text descriptions, and retrieve more relevant textual information from similar images. First, tags associated with an image are converted by two-dimensional code and embedded into the image by discrete wavelet transform (DWT). Next, for images without annotations, similar images can be obtained by CBIR techniques and embedded annotations can be extracted. Specifically, we use global features such as color ratios and dominant sub-image colors for preliminary filtering. Then, local features such as Scale-Invariant Feature Transform (SIFT) descriptors are extracted for similarity matching. This design can achieve good effectiveness with reasonable processing time in practical systems. Our experimental results showed good accuracy in retrieving similar images and extracting relevant tags from similar images.

Keywords: Image Annotation, CBIR, Annotation watermarking, SIFT, QR code

A preliminary version of this paper appeared in ICETI 2012, Nov. 2-6, Kaohsiung, Taiwan. This version includes more experimental results and detailed analysis. This research was sponsored by a research grant from National Science Conucil, Taiwan under the grant number NSC102-2219-E-027-005.

http://dx.doi.org/10.3837/tiis.2014.01.013

1. Introduction

Images often contain rich semantics for human beings as the saying goes: "A picture is worth a thousand words." However, it is still a challenge for computer systems to automatically obtain semantic meanings directly from an image. Existing image search systems such as Google Image Search allow users to submit either text or image queries to get similar images. But users are usually not familiar with formulating exact queries [1]. To obtain more related information, users often need to browse through related images within documents and Web pages to get the overall picture of what they really want. On the one hand, text queries are usually very short, with multiple diverse meanings. Relevance feedback techniques are helpful to provide more related features from initial search results. On the other hand, images might be alternative useful resource which contains rich meanings. Although it might not be easy to directly extract the semantic meanings from image content, if relevant textual descriptions can be effectively embedded into images, we can utilize content-based image retrieval (CBIR) methods to collect similar images, from which the corresponding annotations can be extracted for more related textual information.

In this paper, we propose an annotation watermarking system for image tagging to help users embed and retrieve relevant textual information. The goal of annotation watermarking is to embed relevant semantic information into images as an invisible watermark that can be successfully extracted later. First, given an image and its associated tags, discrete wavelet transform (DWT) is used to transform the image into frequency domain, and the associated tags are converted by two-dimensional code and embedded into the mid-frequency subband of the image based on simple modulo arithmetic. Then, global features such as color ratios and dominant sub-image colors, and local features such as Scale-Invariant Feature Transform (SIFT) descriptors are extracted as the major features for similarity matching. Finally, given a query image, the same features are extracted to find similar images from which the embedded annotations can be extracted to provide useful relevant information for users. From our experimental results on VOC Challenge 2007 data set [2], a high percentage of similar images and relevant tags can be retrieved. The annotation embedding and extraction processes are simple but effective, without affecting image visual quality. Further investigation is needed to verify the performance in larger scales.

The remainder of this paper is organized as follows. In Section 2, we survey the related work in image annotation and CBIR. In Section 3, our proposed approach is described in details. Section 4 shows the experimental results and discussions. In Section 5, we list our conclusions.

2. Related Work

CBIR [3] has been a popular research field in multimedia retrieval. Image feature extraction, similarity estimation, and relevance feedback are among the major topics that have been extensively studied. Among many possible choices of image features such as colors, textures, and shapes, we apply CBIR techniques by using selected global color features such as color ratios and dominant sub-image colors and local features such as SIFT descriptors in finding relevant images from which additional semantic information can be embedded and extracted.

Image annotations [4] are usually given as descriptions in a separate metadata field in files or databases. However, separating metadata from image contents can cause problems of management and security. Annotations in metadata fields can be vulnerable to information loss or modification. For example, descriptions in TIFF images can be removed by converting into JPEG format. Retrieving relevant images also involves obtaining related annotations. When modifying images or annotations, we also need to update the associated counterpart information. To resolve these issues, a special application in digital watermarking [5] called *annotation watermarking* (or *caption watermarking*) [6] is used to embed supplementary information directly into the media contents. The integration of metadata with content prevents the metadata from being easily modified or destroyed.

In recent years, digital watermarking has been applied in many applications, including copyright owner verification (*copyright watermarking*), integrity and authenticity check (*integrity watermarking*). For example, a semi-fragile watermark-based image content authentication technique was proposed [7] to verify the integrity of received image based on secure hash in frequency domain. Since watermarked images might undergo malicious attacks such as geometric deformation and content-preserving manipulations such as JPEG compression, a fragile watermarking scheme with restoration capability was proposed [8] to recover the tampered regions after tampering localization. To enable reversible data hiding for compressed images, Qin et al. [9] used index set construction strategy for vector quantizaiton (VQ) index tables. In contrast to these applications that aim at protecting the security of ownership on copyrighted materials and the integrity information of digital content, *annotation watermarking* focuses on correctly embedding and extracting relevant information for multimedia contents. Several types of annotation watermarking have been proposed for images [10-13], audios [14], and videos [15]. For digital images, illustration watermarking [10] provides an object-based annotation, and they further extend their framework by including hierarchical object relations using hypergraph model [11] and image hashing and cover pre-conditioning [12]. In this paper, we focus on imperceptible and oblivious whole-image annotation watermarking since our major goal is to effectively embed and extract relevant annotations. Different from existing approaches to annotation watermarking, relevant semantic descriptions are converted using two-dimensional codes such as QR code since more textual information can be embedded. Then, they are embeded into the mid-frequency subband of wavelet coefficients, which balances the imperceptibility and robustness to attacks [16, 17].

Local invariant features such as SIFT [18] have shown promise in image retrieval tasks [19]. The major advantages of SIFT include scale- and rotation-invariance, and resistance to noises, occlusions, and affine transforms. In contrast to global features such as color histogram or ordinal feature [20], SIFT is more effective in finding near-duplicate images that can resist image processing attacks [21]. Since the computational costs of SIFT are higher than global features, in this paper, we try to reduce the number of candidates by a filtering step with selected simple global color features such as color ratios and dominant sub-image colors.

3. The Proposed Approach to Annotation Watermarking

The proposed approach can be divided into two phases: indexing and searching. There are four major components in the system architecture: annotation embedding, feature extraction, similarity estimation, and annotation extraction, as shown in **Fig. 1**.

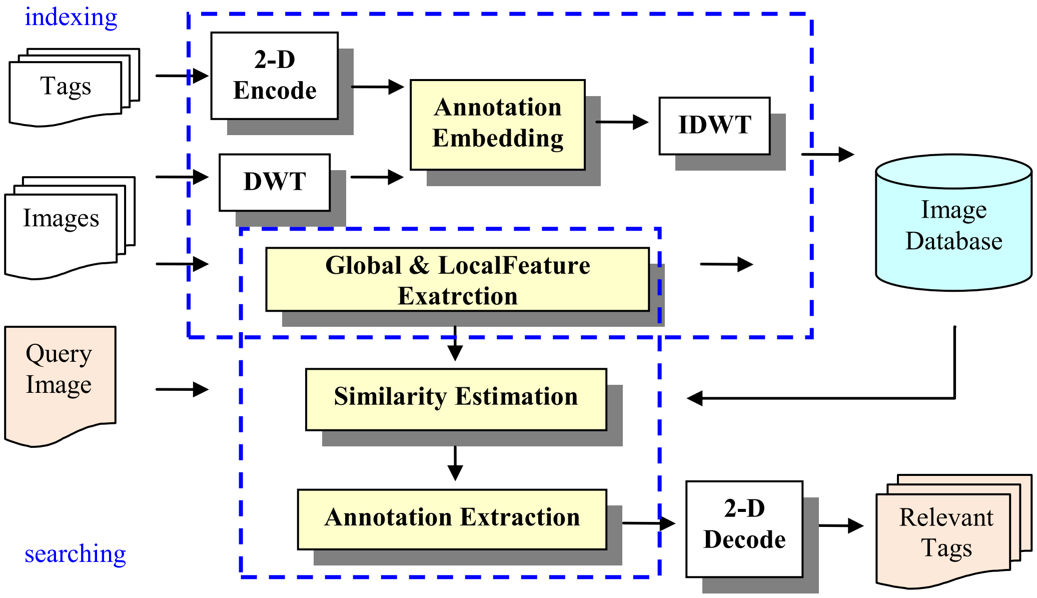


Fig. 1. System architecture of the proposed approach

In the indexing phase, both global and local features are extracted from images, and stored in image database for better searching efficiency. Then, the associated tags are converted by two-dimensional code such as QR Code, and combined into DWT transformed images by annotation embedding module, and also stored in image database. In the searching phase, given an input query image, the same features are extracted and similarity with images in database is estimated. The most similar images are retrieved and the corresponding annotations are extracted by annotation extraction module. Finally, noises in QR code are filtered and the relevant tags can be extracted as needed. Next, we describe each component in details.

3.1 Annotation Embedding

To embed textual annotations, we first convert annotations into two-dimensional codes such as QR codes. QR code (abbreviated from Quick Response Code) [22] is an ISO standard of machine-readable barcodes for automatic identification and data capture. It is widely used in item identification, product tracking and convenient mobile tagging due to its fast readability and larger storage capacity. Generally, it's a content encoding scheme, which puts different types of data (including alphanumeric symbols, Kanji characters and binary data) into two-dimensional square in a zig-zag way. Then, error correction codes based on Reed-Solomon coding are included to detect and correct varying amount of random errors according to different error-correction levels. Depending on the number of symbols to be encoded, there are 40 versions of QR code with symbol size 21x21 to 177x177. The maximum capacity of QR code is 4,296 alphanumeric symbols, 2,953 binary bytes, or 1,813 Kanji characters.

After generating the QR code, it is embedded into mid-frequency wavelet coefficients of the target image by modular arithmetic. Before performing DWT, we convert images from RGB to YCbCr color space as in Eq. (1):

$$\begin{aligned}
Y &= 0.299\times R+0.587\times G+0.114\times B \\
C_b &= (R-Y)\times 0.731+128 \\
C_r &= (B-Y)\times 0.564+128
\end{aligned} \tag{1}$$

Since human eyes are more sensitive to luminance changes in Y component, we can embed two dimensional codes into Cr or Cb components. After performing single-level 2D DWT on Cr component, we obtain four blocks in different frequency channels: HH, HL, LH, LL. Among these blocks, changes in high frequency block (HH) are not easily perceptible, but they are susceptible to image processing attacks; on the other hand, embedding watermarks in low frequency subbands (LL) is more robust to common attacks, but they are easily perceived by humans since the energy of most images are concentrated in low frequencies [17]. Since most watermarking studies choose to embed in mid-frequencies (HL, LH) [16], in this paper, the annotations are embedded in HL blocks. Then, we use a simple modular arithmetic in our embedding algorithm based on the ideas from [23, 24], as in **Fig. 2**:

```
Annotation_Embedding($v$, $w$, $R$, $\beta$)
    $r = |v| \bmod R$
    $\Delta (v) = v - r$
    if $w=1$ then $d=(R-1)*\beta$
    else if $w=0$ then $d=(R-1)*(1-\beta)$
    $v' = \Delta (v) + d$
```

Fig. 2. The annotation embedding algorithm

where v is a wavelet coefficient, w is the pixel value in two-dimensional code, R is a predefined modulus, and β is the parameter to control differences in wavelet coefficients. According to our preliminary experiments and observations, R is set to 20, and β is set to 3/4. After embedding the two-dimensional code into wavelet coefficients, we perform an inverse DWT (IDWT) and convert the embedded image from YCbCr back to RGB color space as in Eq.(2).

$$\begin{aligned}
R &= Y+1.403\times (C_r-128) \\
G &= Y-0.344\times (C_r-128)-0.714\times (C_b-128) \\
B &= Y+1.773\times (C_b-128)
\end{aligned} \tag{2}$$

An example of annotation embedding process is illustrated as in **Fig. 3**.

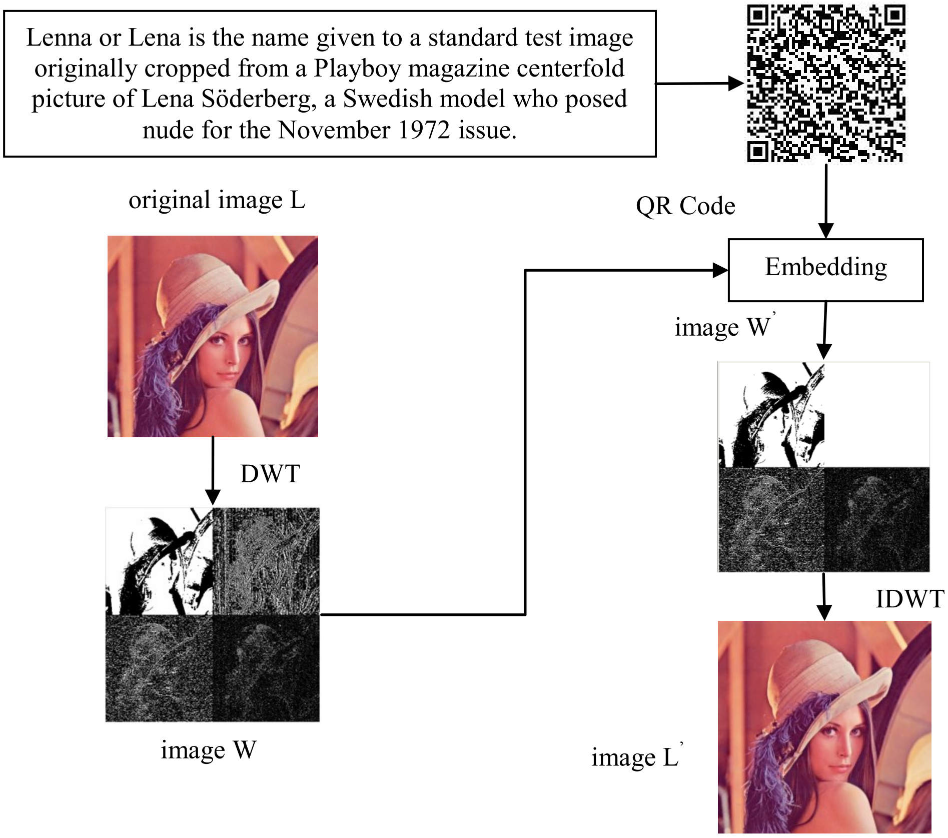

Lenna or Lena is the name given to a standard test image originally cropped from a Playboy magazine centerfold picture of Lena Söderberg, a Swedish model who posed nude for the November 1972 issue.
original image L
QR Code
Embedding
image W'
DWT
IDWT
image W
image L'

Fig. 3. An example of annotation embedding process

3.3 Feature Extraction

To effectively retrieve similar images in different sizes and scales with robustness, local invariant features such as SIFT are often adopted. However, the calculation of SIFT features can be time-consuming, which impedes the practical application for large numbers of images. Thus, we use selected simple global color features as a filtering step to reduce unnecessary SIFT calculation for very dissimilar images.

Global Color Features

Global features are often used in image retrieval to obtain visually similar images. Since different types of images usually have different color distributions, we use color ratios as the major feature. However, there are cases when two images have the same color distribution in different parts of the image. Thus, we further include dominant sub-image colors to reflect the distribution in different parts of an image. First, we convert an image from RGB to HSV color space since it's close to human perceptions. In order to reduce the number of distinct colors for efficient processing, we perform color quantization as follows. By dividing Hue and Saturation components into 13 and 8 levels, respectively, and removing 12 redundant colors for the most saturated of all hues, we obtain 92 colors in HSV space. After further clustering these colors into 7 categories by visual perceptions, we obtain: red, orange, yellow, green, blue, purple, and white, as shown in Fig. 4.

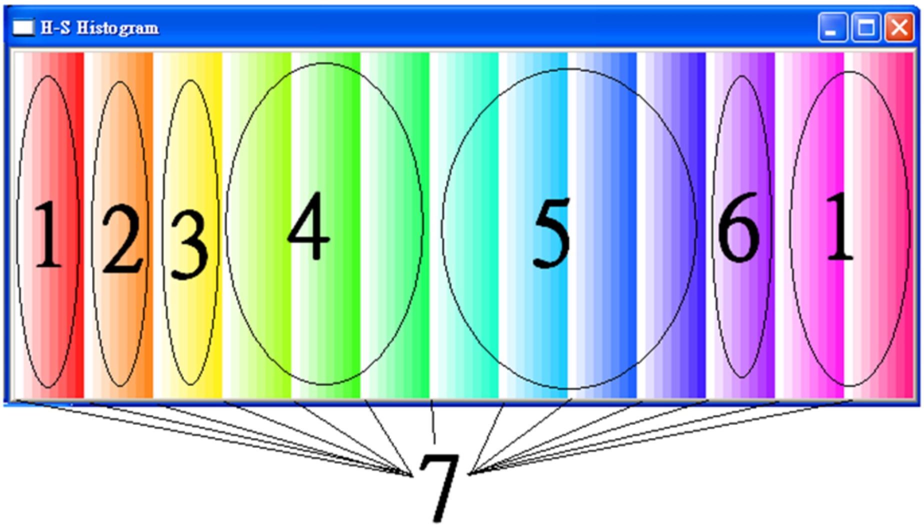


Fig. 4. Quantization of 92 colors into 7 categories in HSV color space

Then, we obtain the color histogram of the image and calculate the color ratio in each category. The *color ratio* $cr_i(I)$ means the percentage of pixels in image I that belongs to category c_i rounded to integers. Although the sum of all color ratios might not be 100%, the performance will not be affected since we use the same method in feature extraction and retrieval. Finally, we divide each image into 6 sub-images b_i, and identify the *dominant color* $dc_i(I)$ in sub-image b_i of the image I.

Since we do not consider the Value component in HSV color space, black and white colors will be grouped into the same category "white". Thus, we will evaluate the performance with and without "white" color ratio features in our experiments. **Table 1** shows the color ratios and dominant sub-image color features for the example image Lena.

Table 1. Color ratios and dominant sub-image colors for the example image Lena

Color	Color Ratio (%)	Block#	Dominant Color
Red	93	b_1	Red
Orange	3	b_2	Red
Yellow	0	b_3	Red
Green	0	b_4	Red
Blue	0	b_5	Red
Purple	2	b_6	Red
White	0		

Local SIFT Features

Local features such as SIFT has been successfully utilized in image retrieval that can resist common image processing attacks. In this paper, we adopt the original method of SIFT [18] to extract the scale-invariant keypoints that might more likely help to identify similar interesting components in images. Since the calculation of SIFT features is time-consuming, we only calculate SIFT descriptors after verifying the color ratios and sub-image dominant colors. Specifically, we obtain the keypoints of an image and record the corresponding descriptors in our database. For any given query image, we compare it with each image in the database, and obtain the keypoints for similarity estimation. The assumption is that: the more SIFT descriptors extracted and matched between two images, the more likely they might be similar in terms of SIFT features.

3.3 Similarity Estimation and Ranking

With information on color features and SIFT features, we estimate the similarity between two images *Q* and *I* as follows. First, based on the overall color distributions in an image, the similarity score in color is calculated as a linear combination of the corresponding scores for color ratios and dominant sub-image colors as follows:

$$S_{color}(Q,I) = \alpha \sum_{i=1}^{6} CR_i(Q,I) + (1-\alpha) \sum_{i=1}^{6} DC_i(Q,I) \tag{3}$$

where α is the weight between the two scores CR_i and DC_i, which are the similarity scores for the color ratio of category c_i and the dominant color of sub-image b_i, respectively, as follows:

$$CR_i(Q,I) = \begin{cases} 1 & \textit{if } |cr_i(Q) - cr_i(I)| < \theta \\ 0 & \textit{otherwise} \end{cases} \tag{4}$$

$$DC_i(Q,I) = \begin{cases} 1 & \textit{if } dc_i(Q) == dc_i(I) \\ 0 & \textit{otherwise} \end{cases} \tag{5}$$

where θ is the threshold for determining if two color ratios are regarded as similar. Generally, if two images have more color categories within similar percentages, and more sub-image blocks with the same dominant color, they are more likely to be similar. After this preliminary filtering by global color features, we can filter out those images that are very dissimilar to the query image. Then, only for images that have higher ranks in color scores, we further calculate their SIFT scores as follows:

$$S_{SIFT}(Q,I) = \begin{cases} 1 & \textit{if } SIFT(Q,I) > SIFT(I,I) * \gamma \\ 0 & \textit{otherwise} \end{cases} \tag{6}$$

where *SIFT(I,I)* is the number of SIFT descriptors extracted from *I*, and *SIFT(Q,I)* is the number of extracted SIFT descriptors matched between *Q* and *I*, and γ is the threshold for determining if they are similar. Since we utilize the approximate nearest-neighbor search for feature matching, the more SIFT descriptors matched between images, the more likely they are similar. We further combine the two types of scores in a simple linear combination as follows:

$$S_{total}(Q,I) = \beta * S_{color}(Q,I) + (1-\beta) * S_{SIFT}(Q,I) \tag{7}$$

Since the number of color categories and the number of sub-image blocks are both 6, the maximum color score $S_{color}(Q,I)$ will be 6 according to Eq.(3). Thus, to balance the effects of color features and SIFT features, we assign a value of 1/6 to β if they are potential similar images in terms of SIFT features. Finally, the images are ranked according to the total score, and the top-*k* images are returned. The corresponding tags can then be extracted as needed. We will determine the appropriate values of various parameters α, θ, and γ in our experiments.

3.4 Annotation Extraction

The annotation extraction process works on wavelet coefficients of an image in a similar way to annotation embedding. Given an image, we first convert it from RGB to YCbCr color space, and perform single-level DWT. Next, we need to extract the embedded watermark by the following algorithm in **Fig. 5**.

```
Annotation_Extraction(v', R)
    $r' = |v'| \bmod R$
    if $r' >= (R-1)/2$ then $w'=1$
    else $w'=0$
```

Fig. 5. The annotation extraction algorithm

where *v'* is an extracted wavelet coefficient, and *R* is a predefined modulus. Then, since images might be subject to possible image attacks or processing, before extracting annotations from two-dimensional codes, we have to filter the possible noises in the QR code image. Since QR codes are composed of blocks in 3x3 pixels, noises can be easily filtered and corrected by checking the dominant pixel in each 3x3 block. Specifically, 3x3 blocks with more than 4 black pixels are all set to black, otherwise they are all set to white. Finally, we can extract the annotation by converting QR codes into texts. An example of annotation extraction process is illustrated in **Fig. 6**.

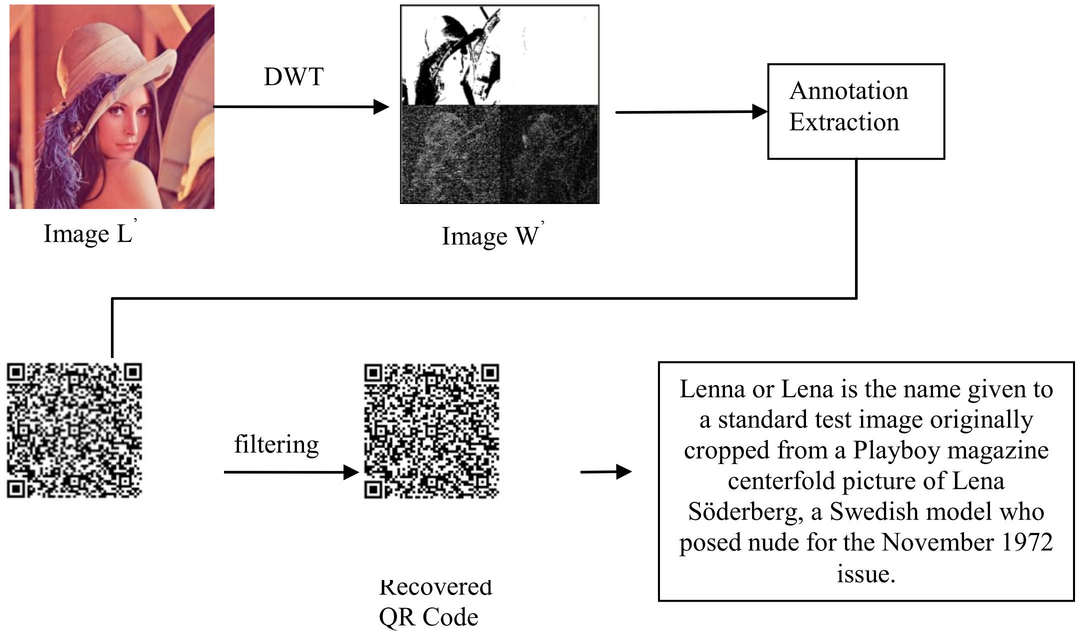

DWT
Image L'
Image W'
Annotation Extraction
filtering
Recovered QR Code
Lenna or Lena is the name given to a standard test image originally cropped from a Playboy magazine centerfold picture of Lena Söderberg, a Swedish model who posed nude for the November 1972 issue.

Fig. 6. An example of annotation extraction process

4. Experiments and Discussions

To evaluate the performance of our proposed approach, 4,952 JPEG images in the VOC Challenge 2007 dataset [2] were used in our experiments. Since the goal of annotation watermarking is to embed supplementary information into the image, the major issues include the imperceptibility of watermarks and the correct embedding and extraction of relevant semantic information from similar images. First, we checked the effects of various parameters on relevant image retrieval in terms of recalls since we want as many relevant images to be included, from which more tags could be retrieved. Then, we evaluated the effects of annotation watermark embedding on image quality by Peak Signal-to-Noise-Ratio (PSNR). Finally, we checked the overall system performance for the percentage of relevant annotations that can be successfully retrieved. Since the test images are in JPEG format, the effects of JPEG compression will also be evaluated.

4.1 Threshold γ for SIFT Features

Since the threshold γ determines the number of SIFT descriptors matched between two images to be considered similar, we first evaluated the distribution of the number of SIFT descriptors for relevant and nonrelevant images. First, we randomly selected three images and utilized XnView to rotate each image in 90 degrees clockwise and counterclockwise every 10 degrees, and scaled in 4 different sizes, respectively. This generated 23 variations for each image including 18 rotated and 4 scaled variations. The remaining 4,949 images in the dataset are considered nonrelevant, which constitutes a total of 5,018 images in this experiment. The percentage of SIFT descriptors for relevant and nonrelevant images are shown in Fig. 7.

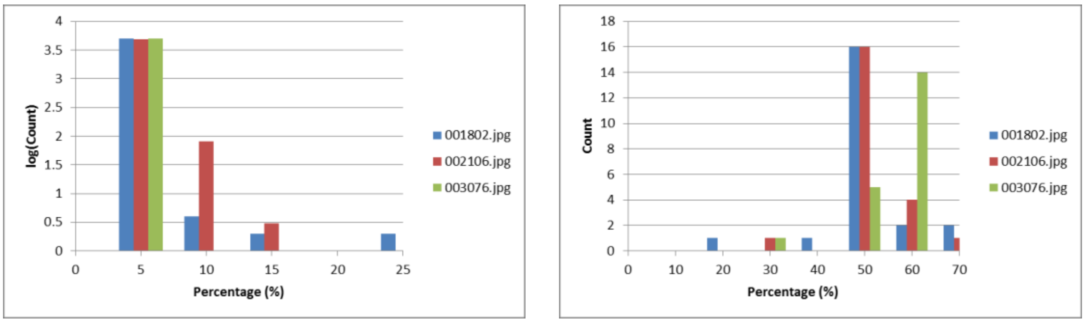


Fig. 7. The percentage of SIFT descriptors in nonrelevant images and relevant images

As shown in **Fig. 7**, we can see most relevant images have more than 20% of SIFT descriptors, and most nonrelevant images have less than 15% of SIFT descriptors. Thus, we chose the threshold γ as 20% in distinguishing relevant and nonrelevant images.

4.2 Effects of White Color Feature

In our global feature extraction method, the "white" color feature might contain both black and white colors since we do not consider the Value component in HSV color space. To compare the effects with and without white color feature, we randomly selected 100 images from the dataset and rotated in 90 degrees clockwise and counterclockwise, respectively. Thus we obtained 1,900 images, each with two versions of features: with and without white color. Under the configurations of various parameters of k as 100, α from 0.1-0.9, and θ from 1-5, we randomly selected 100 images as the query image, and evaluated the performance in terms of recall.

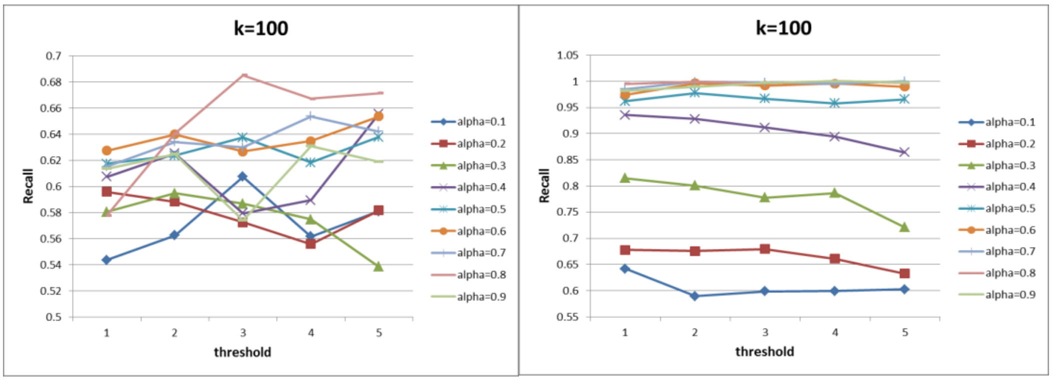
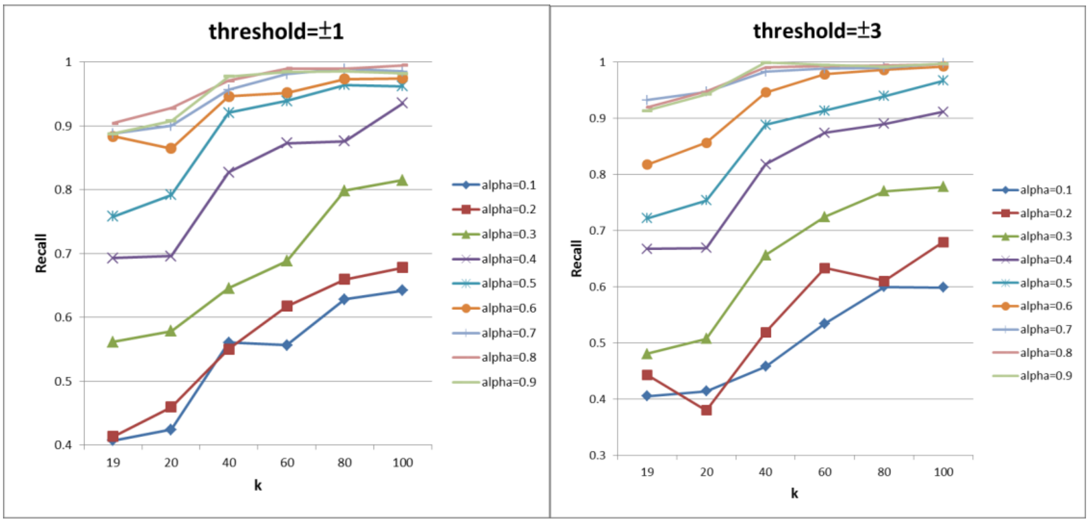


Fig. 8. The recalls with and without white color feature

As shown in **Fig. 8**, the recalls without white color feature are significantly better than with white color feature in all configurations of threshold values and α. The best recall can be obtained when α=0.6-0.9. This shows the disadvantage of using white in our color features. For the remaining experiments, white color features will not be used.

4.3 Effects of θ, α, and *k*

To further verify the effects of θ, α, and k on the performance in terms of average recalls, we checked one parameter at one time: the value of θ for better recalls, the value of k for better efficiency, and the value of α for better effectiveness. First, since θ determines if the color ratios of two images are considered similar, a large value of θ means a loose threshold, while a small value represents a strict threshold. Given the value of θ from 1-5, we can obtain the recall levels as follows.



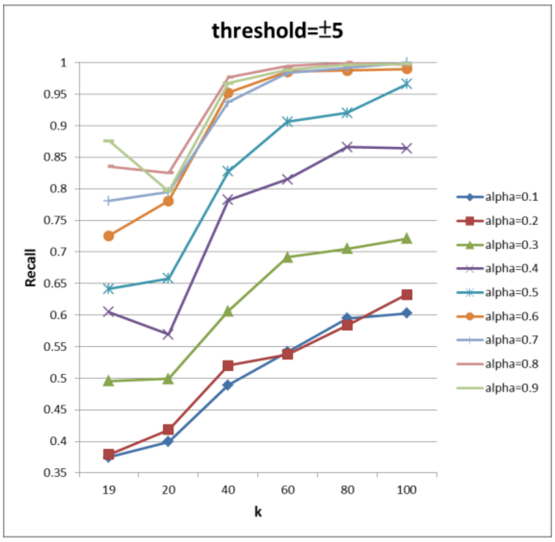


Fig. 9. The recalls for different combinations of k and α, at various threshold θ

As shown in **Fig. 9**, we observed an inferior performance for either a strict threshold (of 5) or a loose threshold (of 1) since the recall will be close to 1.0 only after *k*>=60 in fewer combinations of *α*. Second, the value of *k* affects the number of images that requires SIFT score calculation, and the efficiency will be greatly affected. When *k* is lower than 40, higher threshold *θ* will yield lower recalls. Thus, considering both effectiveness and efficiency, we chose *θ* as 2, as shown in **Fig. 10**.

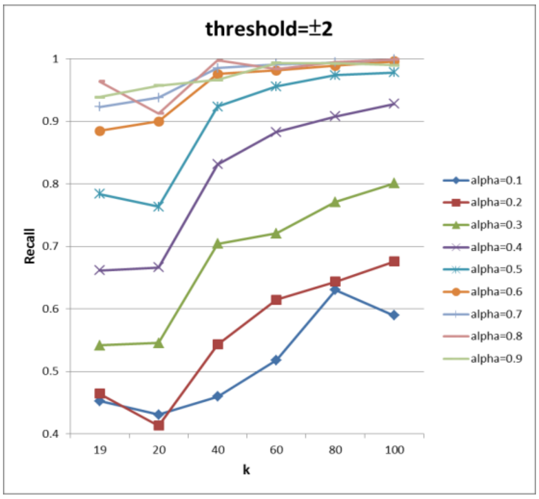


Fig. 10. The recalls for different combinations of k and α (when θ=2)

As shown in **Fig. 10**, recalls generally increase with higher *k*, in which the best performance occurs when *α*=0.6-0.9. But the precision and the efficiency both decrease since the SIFT computational cost is too expensive. Thus, we set *k* as 19, which still gives good recalls when *α*=0.7-0.9.

Finally, considering the overall performance in terms of F-measure, the best performance can be obtained when *α*=0.8. This configuration is used for the overall system.

4.4 Evaluations on Annotated Image Quality

Since transparency or imperceptibility is an important aspect for the acceptance of watermarked media in digital watermarking [15], we next evaluated the impact of annotation watermarks on annotated image quality. To evaluate the degradation in image quality, we adopted the well-known Peak Signal-to-Noise Ratio (PSNR) as commonly used in digital watermarking as in Eq. (8),

$$PSNR = 20 * \log_{10}(\frac{255}{\sqrt{MSE}}) \tag{8}$$

where MSE denotes the mean square error as defined in Eq.(9),

$$MSE = \frac{1}{m*n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\| I(i,j) - E(i,j) \|^2 \tag{9}$$

where *I(i,j)* is the grey level of pixel (*i,j*) in the original image, and *E(i,j)* is the grey level of pixel (*i,j*) in the watermarked image, *m* and *n* are the width and height of the image. A higher PSNR usually indicates higher similarity between the watermarked image and the original. In general, with a PSNR larger than 30dB, it is difficult for human eyes to tell the differences.

Since the test images are in JPEG format, we further investigated the effects of JPEG compression on the annotated image quality. For each of the test images in VOC 2007 dataset, we generated four different versions of images in various qualities. In addition to the original images, we set the JPEG quality factor parameter in OpenCV package as: 10, 30, 50, and 70. The results of the average PSNR for all test images after annotation embedding are shown in **Table 2**.

Table 2. The average PSNR for test images after annotation embedding

JPEG quality factor	PSNR (RGB) Average / Stddev	PSNR (Cr in YCbCr) Average / Stddev
10	44.66dB / 1.12	43.59dB / 0.99
30	44.49dB / 1.06	43.62dB / 1.00
50	44.54dB / 1.07	43.63dB / 1.00
70	44.62dB / 1.06	43.63dB / 0.99
95 (default)	44.61dB / 1.03	43.69dB / 0.99

Note that the PSNR values are calculated in two different ways: one for the average MSE in RGB color space, and the other for the average MSE in Cr channel of YCbCr color space. As shown in **Table 2**, we can see almost no significant difference among the various JPEG quality factors. This shows high average PSNR can be obtained for the test images.

The resulted annotated images for selected images from VOC 2007 and well-known example images after annotation embedding are shown in **Fig.11**.

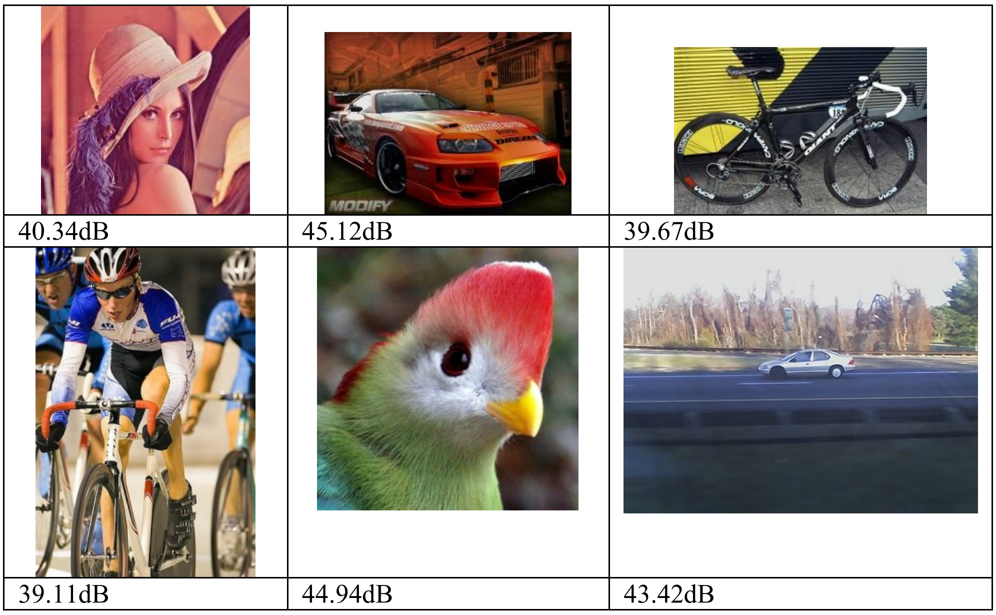


Fig. 11. The PSNR for selected images after annotation embedding

In the case of the well-known example image Lena, our approach obtains a PSNR of 40.34dB. This is comparable to the performances of 20-40dB given different capacity levels in a previous study [10], and 37.5-42.28dB given different embedding strengths in a recent study [13]. This experimental result validates the performance of our proposed approach in annotation embedding without degrading the image quality.

4.5 Evaluations on Relevant Annotation Retrieval

Finally, we further checked the performance of relevant image annotation retrieval under image processing attacks and JPEG compression. Using the configuration as determined in the previous experiments, we utilized the same test images in different JPEG quality factors, and evaluated the performance in terms of the percentage of annotations that can be successfully retrieved (or the *success rate*). The results are shown in Table 3.

Table 3. The percentage of successfully retrieved annotations in various JPEG quality factors

JPEG quality factor	Success Rate
10	96.57%
30	97.17%
50	97.25%
70	97.31%
95 (default)	97.33%

As shown in **Table 3**, the success rates in retrieving annotations under various JPEG quality factors are quite high. The degradation in JPEG quality only slightly degrades the success rate. This validates the performance of our image retrieval method with global and local features.

To show the effects of JPEG compression on the QR code recovery, we further applied different levels of error correction L, M, Q, and H in QR codes. For each of the test images, we checked if the different levels of JPEG compression will affect the successful extraction and recovery of annotations in QR codes with various error correction levels. The results are show in Fig. 12.

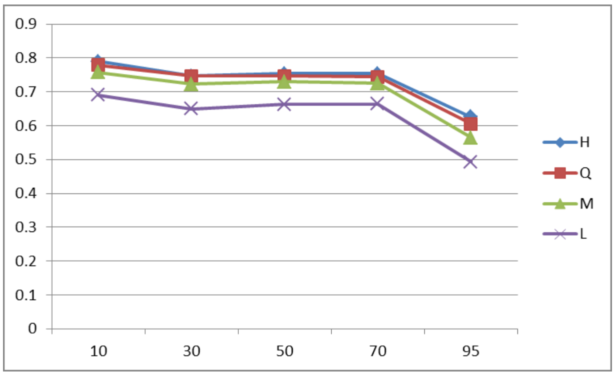


Fig. 12. The performance of QR code recovery rate in various error correction levels and JPEG quality factors

As shown in Fig. 12, we can see a better recovery rate when the quality factor decreases. This is due to the less details in a lower quality compressed image. When QR codes are embedded into the image, there will be less damage to the QR code. As the error correction level increases for QR codes, the corresponding recovery rate also increases. But when error correction reaches Q and H levels, the improvement becomes limited. And the data size for QR code will greatly increase for higher error correction levels. Thus, for a better balance between recovery rate and data size, it would be satisfactory to use error correction level M.

For the QR codes that cannot be successfully recovered, we further analyzed the reasons as follows. First, some test images are very dark in most of the pixels. Second, the color distribution is biased towards a particular color, in which there are more images with such color ratios than *k* (with the value of 19). Third, black-and-white photos stored in RGB formats could be another major sources of errors. The reason is the global color feature extraction for these images will be either black or white, which are not distinguishable in our global color ratios and sub-image dominant colors. Thus, they will be unable to pass the filtering step.

Finally, to be practically useful in real systems, we further evaluated the system efficiency under various configurations. Among the different modules in the system, the time spent in the following modules can be ignored as compared to other modules: modular arithmetic for annotation embedding and extraction cost $7.4*10^{-5}$s and $6.7*10^{-5}$s, respectively; QR encoding and decoding cost $2.7*10^{-3}$s and $3.3*10^{-4}$s, repesctively. The remaing results are shown in Table 4.

Table 4. The system efficiency in various configurations

JPEG quality factor	SIFT	Global feature	DWT	IDWT	Matching
10	0.152s	0.0026s	0.131s	0.114s	0.0016s
30	0.134s	0.0025s	0.131s	0.114s	0.0016s
50	0.126s	0.0025s	0.130s	0.114s	0.0016s
70	0.122s	0.0025s	0.130s	0.114s	0.0016s
95 (default)	0.130s	0.0025s	0.131s	0.114s	0.0016s

As shown in Table 4, the most time consuming parts are SIFT calculation and DWT/IDWT. In the indexing phase, both global and local image features are extracted offline. This can greatly reduce the searching time. In the searching phase, only the feature extraction and the matching time for query image are necessary. This implies an average of about 0.134s per image query in the test collection. This performance is feasible for practical system use. Since we limited the number of potential candidates with global color features inlcuding color ratios and dominant sub-image colors, the unnecessary processing overhead for SIFT calculations can be effectively reduced. The results also show the good performance in effectiveness and efficiency.

From these experimental results, we list several observations and discussions as follows:

1. SIFT features are effective for CBIR even under image processing attacks such as rotations and scaling.
2. To effectively obtain color ratios, we converted the color space from RGB to HSV and clustered colors into 7 categories according to visual perceptions. From our experimental results, removing white color features can avoid the confusion between white and black in the absence of Value component in HSV color space.
3. From the experimental results on annotated image quality, the proposed annotation embedding algorithm can obtain comparable results for the perceived image quality in terms of PSNR.
4. To balance the effectiveness and efficiency, we tested various settings of parameter configurations. With the filtering of color ratios and dominant sub-image colors, the system can retrieve a high percentage (more than 96%) of the relevant images and their annotations for relevant information retrieval.
5. JPEG compression ratios only have a limited effect on the retrieval performance. And the quality of annotated images is not greatly degraded. This shows that annotation watermarking can preserve high image quality (an average PSNR of more than 43.0dB).

In this paper, we focus on the annotations for whole images. Partial regions or objects in an image cannot be effectively identified without considering semantic visual features. Since the effectiveness of our proposed approach highly depends on SIFT features, the higher computational costs can be further reduced with better data structures such as *k*-d tree. Further investigations are needed for larger image datasets.

5. Conclusion

We have proposed an annotation watermarking system for retrieving relevant tags from similar images. Two-dimensional codes and DWT are utilized to embed and extract image annotations. We also included global color features such as color ratios and dominant sub-image colors as a simple filtering step, and local SIFT features for effectively estimating similarity among images. The experimental results showed good performance for retrieving similar images and extracting relevant textual information from similar images for users.

References

[1] Christopher D. Manning, Prabhakar Raghavan and Hinrich Scheutze, *Introduction to Information Retrieval*, Cambridge University Press, 2008. Article (CrossRef Link)

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007. Article (CrossRef Link)

[3] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta and Ramesh Jain, "Content-Based Image Retrieval at the End of the Early Years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.22, no.12, pp.1349-1380, December, 2000. Article (CrossRef Link)

[4] Minxian Li, Jinhui Tang and Chunxia Zhao, "Active Learning on Sparse Graph for Image Annotation," *KSII Transactions on Internet and Information Systems*, vol.6, no.10, pp.2650-2662, Oct. 2012. Article (CrossRef Link)

[5] V. M. Potdar, S. Han and E. Chang, "A Survey of Digital Image Watermarking Techniques," in *Proc. of 3rd International IEEE Conference on Industrial Informatics*, pp.709-716, 2005. Article (CrossRef Link)

[6] J. Dittmann, M. Steinebach, T. Kunkelmann and L. Stoffels, "H204M — Watermarking for Media: Classification, Quality Evaluation, Design Improvements," in *Proc. of 2000 ACM Workshops on Multimedia*, pp. 107-110, 2000. Article (CrossRef Link)

[7] P. D. Sheba Kezia Malarchelvi, "A Semi-Fragile Image Content Authentication Technique based on Secure Hash in Frequency Domain," *International Journal of Network Security*, vol. 15, no. 5, pp. 365-372, 2013. Article (CrossRef Link)

[8] C. Qin, C. C. Chang, P. Y. Chen, "Self-Embedding Fragile Watermarking with Restoration Capability Based on Adaptive Bit Allocation Mechanism", *Signal Processing*, vol. 92, no. 4, pp. 1137-1150, 2012. Article (CrossRef Link)

[9] C. Qin, C. C. Chang, Y. C. Chen, "A Novel Reversible Data Hiding Scheme for VQ-Compressed Images Using Index Set Construction Strategy," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 8, pp. 2027-2041, 2013. Article (CrossRef Link)

[10] T. Vogel and J. Dittmann, "Illustration Watermarking: An Object based Approach for Digital Images," in *Proc. of SPIE 2005*, pp. 578-589, 2005. Article (CrossRef Link)

[11] C. Vielhauer and M. Schott, "Image Annotation Watermarking: Nested Object Embedding using Hypergraph Model," in *Proc. of MM&Sec 2006*, pp. 182-189, 2006. Article (CrossRef Link)

[12] M. Schott, J. Dittmann and C. Vielhauer, "AnnoWaNO: An Annotation Watermarking Framework," in *Proc. of 6th International Symposium on Image and Signal Processing and Analysis*, pp. 483-488, 2009. Article (CrossRef Link)

[13] P. Korus, J. Bialas and A. Dziech, "A New Approach to High-Capacity Annotation Watermarking based on Digital Fountain Codes," *Multimedia Tools and Applications*, 2012. Article (CrossRef Link)

[14] P. Dymarski and R. Markiewicz, "Informed Algorithms for Watermarking and Synchronization Signal Embedding in Audio Signal," in *Proc. of 20th European Signal Processing Conference* (EUSIPCO 2012), pp. 2699-2703, 2012. Article (CrossRef Link)

[15] P. C. Su and C. Y. Wu, "A Join Watermarking and ROI Coding Scheme for Annotating Traffic Surveillance Videos," *EURASIP Journal on Advances in Signal Processing*, pp. 1-14, 2010. Article (CrossRef Link)

[16] Gin-Der Wu and Pang-Hsuan Huang, "Image Watermarking Using Structure based Wavelet Tree Quantization," in *Proc. of 6th IEEE International Conference on Computer and Information Science (ICIS 2007)*, pp. 315-319, 2007. Article (CrossRef Link)

[17] P. Tay and J. P. Havlicek, "Image Watermarking using Wavelets," in *Proc. of 45th Midwest Symposium on Circuits and Systems (MWSCAS 2002)*, pp.258-261, 2002. Article (CrossRef Link)

[18] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal on Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004. Article (CrossRef Link)

[19] Congxin Liu, Jie Yang and Deying Feng, "PPD: A Robust Low-Computation Local Descriptor for Mobile Image Retrieval," *KSII Transactions on Internet and Information Systems*, vol.4, no.3, pp.305-323, Jun. 2010. Article (CrossRef Link)

[20] D. Bhat and S. Nayar, "Ordinal Measures for Image Correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 415–423, 1998. Article (CrossRef Link)

[21] Jenq-Haur Wang, Hung-Chi Chang and Jen-Hao Hsiao, "Protecting Digital Library Collections with Collaborative Web Image Copy Detection," in *Proc. of 11th International Conference on*

Asian Digital Libraries (ICADL 2008), pp. 332-335, 2008. Article (CrossRef Link)

[22] Denso-Wave, QR Code – About 2D Code. Article (CrossRef Link)

[23] Y. Yuan, D. Huang and D. Liu, "An Integer Wavelet based Multiple Logo-Watermarking Scheme," in *Proc. of 1st International Multi-Symposiums on Computer and Computational Sciences*, pp. 175-179, 2006. Article (CrossRef Link)

[24] M. Tsai, K. Yu and Y. Chen, "Joint Wavelet and Spatial Transformation for Digital Watermarking," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 241-245, 2000. Article (CrossRef Link)

Jenq-Haur Wang received his B.S. and Ph.D degrees in computer science and information engineering from National Taiwan University, Taiwan, in 1994 and 2002, respectively. From Oct. 2002 to Feb. 2007, he was a postdoctoral fellow in Institute of Information Science, Academia Sinica, Taiwan. Since Mar. 2007, he is an assistant professor at the Dept. of Computer Science and Information Engineering, National Taipei University of Technology, Taiwan. Dr. Wang is a member of the ACM and IEEE. His research interests include social Web mining and knowledge discovery, network and information security, peer-to-peer retrieval and cloud computing.

Chuan-Ming Liu is an associate professor in the Department of Computer Science and Information Engineering, National Taipei University of Technology (NTUT), TAIWAN. He received his Ph. D. in Computer Sciences from Purdue University in 2002 and B.S. and M.S. degrees both in Applied Mathematics from National Chung-Hsing University, Taiwan, in 1992 and 1994, respectively. In the summer of 2010 and 2011, he has held visiting appointments at Auburn University and Beijing Institute of Technology, respectively. Dr. Liu’s research interests include data management and data dissemination in various emerging computing environments, query processing in moving objects, location-based services, ad-hoc and sensor networks, parallel and distributed computation, and analysis and design of algorithms.

Jhih-siang Syu is a software engineer in system software engineering department, ASUS Cloud Corporation. He received his M.S. degree in Computer Science from National Taipei University of Technology, Taiwan, in 2010. His research interests include multimedia information retrieval, digital watermarking, and cloud computing.

Yen-Lin Chen received the B.S. and Ph.D. degree in electrical and control engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2000 and 2006, respectively. From Feb. 2007 to Jul. 2009, he was an Assistant Professor at the Dept. of Computer Science and Information Engineering, Asia University, Taichung, Taiwan. From Aug. 2009 to Jan. 2012, he was an Assistant Professor at the Dept. of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan, and since Feb. 2012, he is now an Associate Professor in the same institute. Dr. Chen is a Senior Member of the IEEE, and a member of ACM, IAPR, and IEICE. In 2012, he earned the best annual paper award sponsored by the Intelligent Transportation Society of Taiwan. In 2003, he received Dragon Golden Paper Award sponsored by the Acer Foundation and the Silver Award of Technology Innovation Competition sponsored by the AdvanTech. He also earned the Silver Invention Award from Ministry of Economic Affairs of Taiwan, and received best annual ITS paper awards two times in 2011 and 2012. His research interests include image and video processing, embedded systems, pattern recognition, intelligent vehicles, and intelligent transportation system. His research results have been published on over 80 journal and conference papers.