J. Vis. Commun. Image R. 25 (2014) 963–969

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Content-based image retrieval using H.264 intra coding features

Ren-Jie Wang, Ya-Ting Yang, Pao-Chi Chang *

Department of Communication Engineering in National Central University, Jhong-Li 320, Taiwan

ARTICLE INFO

Article history:
Received 1 June 2013
Accepted 21 February 2014
Available online 6 March 2014

Keywords:
Content-based image/video retrieval
H.264
Intra prediction
Video coding
Compression domain
Geometrical verification
Image search
Texture features

ABSTRACT

Efficient multimedia retrieval has become a vital issue because more audio and video data are now available. This paper focuses on content-based image retrieval (CBIR) in the compression domain (CPD). The retrieval features are extracted based on I-frame coding information in H.264. This paper proposes using a local mode histogram as the texture feature to match images and applying the residual coefficients to filter non-confident modes. The geometrical correspondence between two images is also considered. The experimental results show that the proposed method can substantially reduce computational and memory resource consumption, and provides similar performance compared with methods that extract features from decompressed images.

1. Introduction

Multimedia retrieval applications have received substantial attention. With the rapid growth of digital multimedia data, the effective retrieval of visual information has become a challenging research issue. Instead of using text, content-based image retrieval (CBIR) [1] provides a more flexible method to extract visual features, such as color [2,3] or shape [4,5], to represent the characteristics of the query image and other images in a database. Traditional CBIR methods extract the visual features from raw pixels; this is referred to as the pixel domain (PXD) approach. Because almost all images on the Internet or in local storage are stored in a compressed form, such as JPEG or H.264 intra coding, a decompression operation is necessary for the PXD approach. A more efficient retrieval approach for real-time implementation is to operate in the compression domain (CPD) [6,7]; that is, extracting features from the encoded bitstream without fully decoding the images. This approach generally achieves better time and memory efficiency than the PXD approach.

Numerous related studies on feature extraction and matching in the JPEG CPD have been conducted. A discrete cosine transform (DCT) is performed on image blocks based on the JPEG standard. The luminance and the shape in a block can be extracted from the DC [8] and AC coefficients [9,10], respectively. In addition to applying DCT coefficients to image retrieval, the feature extractions obtained from the DCT coefficients can also be applied to facial recognition [11] and pornographic image detection [12].

Another CPD that various studies have addressed is the intra coded frame in video codec. Because of the fast and independent decoding of intra frames, these methods consider the intra coded frame as the key frame in a video for performing feature extraction and image matching. For MPEG coding, intra coding uses a similar coding structure as JPEG coding. The feature extraction technique using DCT coefficients can be applied to intra frames in MPEG. Related applications, including caption localization [13] and shot change detection [14,15], have been proposed.

Novel video coding standards have recently been released and have been extensively used. More information in the bitstream is available because of sophisticated predictions employed in these standards. Predictive intra frame coding is an innovation in H.264, in which the DCT is applied to the residual signal instead of the original frame. In addition to transform coefficients, a prediction mode must be transmitted and be available as a content feature in predictive coding. The use of prediction modes as features has been effectively demonstrated in motion object extraction [16] and shot detection [17].

Intra prediction in H.264 can also be applied to single image coding because it has higher coding efficiency than JPEG does [18]. Zargari et al. [19] proposed an image retrieval method for intra coded images in H.264 standard. It uses a direction histogram of intra prediction mode as the feature that matches similar images and achieves strong results. However, the experimental results presented in the study only show the performance of specific

* Corresponding author. Fax: +886 3 4229187.
E-mail address: pcchang@ce.ncu.edu.tw (P.-C. Chang).

http://dx.doi.org/10.1016/j.jvcir.2014.02.016
1047-3203/

textures. Performance degrades when it is directly applied to general consumer images, as shown in the experimental results in Section 4. Hence, the direction histogram of intra prediction mode must be investigated further before it can be applied to a general image retrieval system.

In this paper, employing a local direction histogram as the texture feature to match images and the application of residual coefficients to filter non-confident modes are presented. Moreover, the geometrical correspondence between two images is also considered. The experimental results show that the proposed method can reduce computational and memory resource consumption, as well as achieve similar performance compared to methods that extract features in fully decompressed images.

The rest of this paper is organized as follows. Section 2 introduces image retrieval in the CPD. In Section 3, the proposed CBIR method is described. Section 4 shows performance evaluations and comparisons with related works. Finally, a conclusion is offered in Section 5.

2. Image retrieval in the compression domain

In Fig. 1, the path with solid lines represents the retrieval process in the PXD. The retrieval process requires full decoding and feature extraction processing to obtain features for matching. It is interesting to observe that in order to improve the coding efficiency the operations in modern video encoders are similar to content analysis although the objective is different. The coding information, such as the prediction mode or transform coefficients in a bitstream, contains the high-level information of an image. The retrieval process in the CPD, as shown by the dotted lines in Fig. 1, extracts the coding information and refines it to directly produce the features. Here, only partial decoding is required to decode the binary code back into the corresponding values. Although this coding information might not be more meaningful than the original frames, it still provides rich texture and color information for retrieval.

The structure of a retrieval system can be realized by using either one of the two methods [20]. In the first method, which is more direct, the end user sends a query image in a compression form to the server, and feature extraction and matching are then performed in the server. Alternatively, the second method, by taking advantage of the privacy and loading reduction of the server, is to perform feature extraction at the user end in advance, and then the compact features are sent to the server to be matched to database images.

The two mobile visual search scenarios, extracting features on server-side (the first method) and extracting features on user-side (the second method), are compared as follows. (1) To consider the user-side computational complexity, the second method has to pay computational overhead on client devices, while the first method frees this overhead. (2) Transmission overhead is a JPEG file in the first method, while the transmission overhead in the second is the extracted feature vector that is usually significantly less than a whole JPEG file. (3) Response time includes the transmission delay and the processing delay for feature extraction and matching. The second method results in more processing delay than the first method because of the computational power limitation on client devices. While the transmission delay for feature vector is less than JPEG file. The overall response time for both methods thus will depend on the client device computation power and Internet speed.

Regardless of which structure is selected to implement the retrieval system, the approach in the CPD can be used to improve the time efficiency, as demonstrated in Fig. 2. For feature extraction in the server, the response time for a query image might be acceptable. However, the feature extraction for a database continues to consume substantial time, especially with the enormous amount of images and videos that are uploaded each minute. Hence, faster feature extraction is critical. Regarding the second method, efficient feature extraction is indispensable when a mobile device is used at the user end because of the limited computational power and battery life of mobile devices.

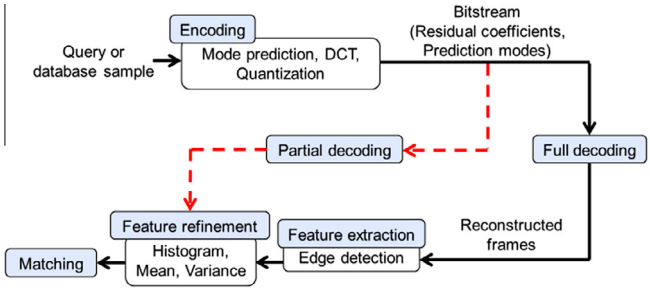


Fig. 1. Retrieval system in the pixel or compression domain.

3. Proposed method

The proposed method is based on H.264 I-frame coding. Picture features include not only prediction modes but also residual coefficients that are extracted from the bitstream. Moreover, local mode histogram with a patch size instead of global histogram is proposed for image matching. Geometrical correspondence is also considered in the proposed method.

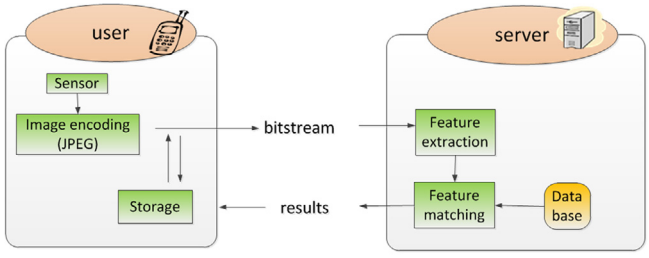
3.1. Analysis of intra prediction mode

The edge direction of texture has been extensively used as a feature for matching patterns in other studies. The coding mode can represent the edge direction of an original block and, hence, can be used for feature matching, as described in the following section.

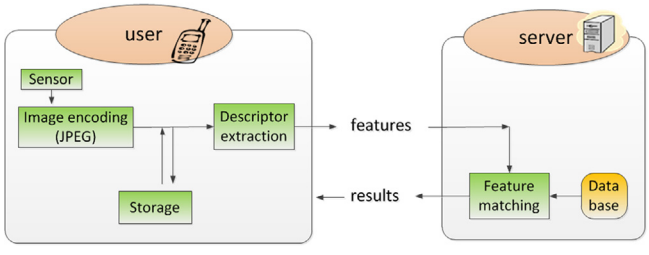
The intra mode decision in H.264 [21] selects the optimal coding mode by minimizing the RD cost function as

$$J(\theta) = D(\theta) + \lambda_L(QP)R(\theta) \quad (1)$$

where D and R are the distortion and rate, respectively, and both depend on the prediction direction θ in H.264; λ_L is the Lagrange



(a) Feature extraction in server end



(b) Feature extraction in user end

Fig. 2. The structures of the retrieval systems.

parameter, which is a function of the quantization parameter *QP*. Because the rate and distortion are both dominated by the residual variation [22], the RD minimization in Eq. (1) can be approximated to minimize the mean square of the residuals

$$\sigma_r^2(\theta) = \int_y \int_x (f_{x,y} - p_{x,y}(\theta))^2 dxdy \tag{2}$$

in which f and p are the original block and the predicted block, and x and y are the position indexes, respectively. According to the intra prediction in H.264, the predicted block can be represented as

$$p_{x,y}(\theta) = \begin{cases} f'_{(x-\frac{y+1}{\tan\theta}),-1}, & \tan\left(\frac{y}{x}\right) < \theta \\ f'_{-1,(y-(x+1)\tan\theta)}, & \text{o.w.} \end{cases} \tag{3}$$

where f′ is the reconstructed block. The prediction signal is obtained from the nearest reconstructed pixels with a prediction direction. In general, the prediction on the right side of the block comes from the top side; the prediction on the left side comes from the left as shown in Fig. 3.

Conversely, the original block can be represented with an edge direction of original block θ_0 and the original pixels of the upper and left boundaries:

$$f_{x,y} = \begin{cases} f_{(x-\frac{y+1}{\tan\theta_0}),-1}, & \tan\left(\frac{y}{x}\right) < \theta_0 \\ f_{-1,(y-(x+1)\tan\theta_0)}, & \text{o.w.} \end{cases} \tag{4}$$

If constant f_u and f_l are used to represent the pixels of the top and left boundaries, respectively, and the quantization error is ignored, that is,

$$f'_{x,-1} = f_{x,-1} = f_u, \quad x = 0\text{—}3 \tag{5}$$

$$f'_{-1,y} = f_{-1,y} = f_l, \quad y = 0\text{—}3 \tag{6}$$

then the residual variance can be approximated as in Eq. (7) when the rectangular block is approximated to a circle, as shown in Fig. 4.

$$\sigma_r^2(\theta) = \frac{(f_u - f_l)^2|\theta_0 - \theta|}{(\pi/2)} \tag{7}$$

Consequently, the residual variance is proportional to the square of the difference between f_u and f_l, and the absolute difference between θ_0 and θ. The minimum of the residual variance falls on $\theta = \theta_0$. Hence, the prediction direction determined by the video encoder can approximate the edge direction of the original block.

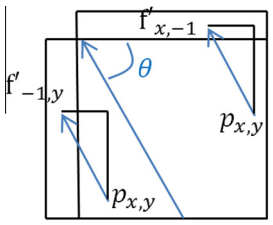


Fig. 3. A case of H.264 intra prediction.

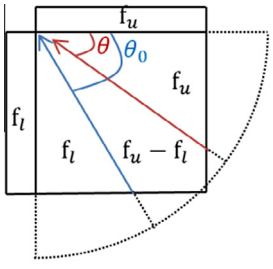


Fig. 4. Simplified intra prediction modeling.

3.2. Local mode histogram

The histogram of the mode index is employed to describe the texture characteristics in this study. To map the coding mode into the corresponding bin of the histogram, nine direction modes of I4 × 4 blocks in H.264 are assigned to feature bins 0–8, and four direction modes for I16 × 16 blocks are assigned to Bins 9–12.

Because a global mode histogram could suffer from the problem of different viewpoints, using local histograms with a suitable patch size can improve the matching performance. In this work, the image is divided into a fixed number of patches, N_p^2, as shown in Fig. 5. The corresponding patch size can be obtained as

$$\text{Patch Size} = \frac{W}{N_p} \times \frac{H}{N_p} \tag{8}$$

where W and H are the image width and height, respectively. The performances with different N_p are shown in Section 3; $N_p = 8$ provides the best performance in the test data set.

To evaluate the similarity between the two images, the matching score is defined as

$$S_i = \max_j \{H_q(i) \cap H_d(j)\} \tag{9}$$

where H_q and H_d are the mode histogram of the query image and the database image, respectively; i and j are the patch indexes. By starting from the first patch in the query image to search each patch of the database image, the matched patch that has the maximum intersection can be found. Each patch can locate a corresponding patch with a similarity value of S_i. The average score S for all patches is used to calculate the similarity between the two images. A high S indicates that two images are similar.

Note that, it is possible to have many-to-one mapping, but it is not needed to avoid in advance. This many-to-one mapping usually happens in the estimation of two irrelevant images, and thus can be used as an indication to show whether the two images are matched well. In practice, many-to-one mapping results in high variation of estimated displacement vector, which has been considered and incorporated into the similarity calculation as described in Section 3.4.

3.3. Residual information

In H.264 I-frame coding, the original signal is decomposed into the direction mode and the residual. In addition to direction mode, the residual signal also provides rich information on the texture feature.

According to Eq. (7), if the residual coefficient power σ_r^2 is low, then the edge exists in the prediction direction ($\theta_0 = \theta$). However, if the residual coefficient power is high, then an edge deviation from the prediction direction exists ($\theta_0 \neq \theta$). This analysis matches the phenomenon practically observed in [23]. Hence, the reliability of the prediction mode can be decided according to the corresponding residual power. By setting the mean of the residual power at threshold T, non-confident modes can be removed when the residual power is higher than the threshold. In addition to mode filtering, a high residual power also indicates a high variation of the texture content $(f_u - f_l)^2$. The texture variation is recognized as a significant feature for matching.

The local mode histogram is incorporated with the residual information in this work. The filtered mode histogram, in which the residual power σ_r^2 is below the mean value, is applied to feature bins 0–13. Moreover, the histogram of residual power that is greater than threshold T is appended to feature bins 14–23, as

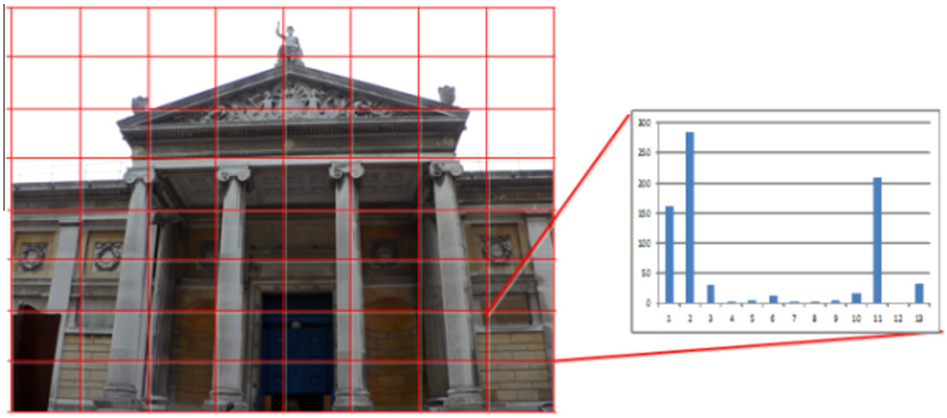


Fig. 5. Patch partition and local mode histogram.

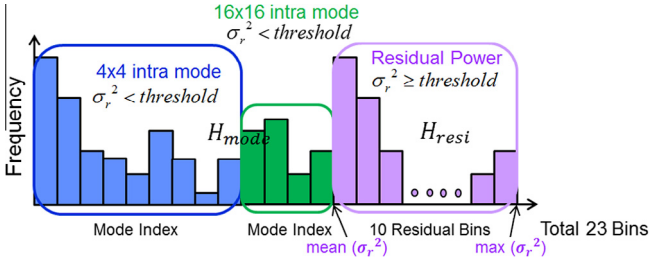


Fig. 6. Proposed feature vector for matching.

shown in Fig. 6. In summary, the proposed feature vector is defined as follows:

$$H_{total} = [H_{mode}\ H_{resi}] \tag{10}$$

where

$$H_{mode} = hist(mode|\sigma_r^2 < T)$$

$$H_{resi} = hist(\sigma_r^2|\sigma_r^2 \geqslant T)$$

3.4. Geometrical verification

A patch based search may suffer from incorrect matching resulted from conditions such as different viewpoint, different image size, and object movement. The geometrical correspondence of the matching results should be verified. For time efficiency, a simple yet efficient approach is proposed as an alternative to homography for geometrical verification. If two images are captured from the same place but different viewpoints, the estimated displacement between two images should be much consistent within certain regions. In contrast, the estimated displacement becomes complicated in a wrong image pair. Thus we use the local variation of the displacement to check the geometric issue.

V_i^x and V_i^y are defined as the displacement vectors (DV) of the ith patch in the horizontal and vertical directions, respectively, which is the exact distance of the patch positions between the two corresponding patches in the query and database images. A high variation of the DV in the same object region tends to result in a poor match. In other words, lower variations of the DV indicate a strong geometrical correspondence. The local variation of the DV is used to evaluate the geometrical correspondence to refine the matching score. The difference of the displacement vector (DDV) is defined as the absolute difference between the DV and average DV

$$D_i = |V_i^x - V_i^{xm}| + |V_i^y - V_i^{ym}| \tag{11}$$

where V_i^{xm} and V_i^{ym} are the average DV of the eight DVs around the ith patch in the horizontal and vertical directions, respectively. Hence, the DDV is introduced to restrict the texture similarity S_i. The matching score with geometrical correspondence S_i^{GV} is

$$S_i^{GV} = S_i - \lambda \times \frac{D_i}{2 \times N_p} \tag{12}$$

where λ is a weight for two normalized terms that can be empirically set to a constant 0.05 for good retrieval performance. If two DVs for two adjacent patches are close, the DDV is small and does not significantly affect S_i^{GV}. However, if the two DVs are distant from each other, then the DDV is large and the similarity will decrease.

3.5. Overall framework

An overall block diagram of the proposed system is shown in Fig. 7. In a compressed bitstream for the input, the intra-predicted portion is partially decoded. The prediction modes and residual coefficients are then obtained to construct the proposed feature vector for patch matching. Finally, a geometrical verification is used to obtain the final similarity score. After searching each image in the database, the recommended list is produced according to the similarity scores.

4. Experimental results

To evaluate the performance of the proposed method, the Oxford 5k database [24] and INRIA Holiday [30] was used. The Oxford Buildings Dataset consists of 5062 images collected from Flickr by searching for particular Oxford landmarks. The collection has been manually annotated to generate a comprehensive ground truth for 11 different landmarks, each represented by 5 possible queries. This gives a set of 55 queries over which an object retrieval system can be evaluated. The holidays dataset is a set of images which mainly contains some of personal holiday photos. The remaining ones were taken on purpose to test the robustness to various attacks: rotations, viewpoint and illumination changes, blurring, etc. The dataset includes a very large variety of scene types (natural, man-made, water and fire effects, etc.) The dataset contains 500 image groups, each of which represents a distinct scene or object. In the H.264 coding configuration, the reference software, JM17.2 [25], was used; the mode decision method was set to high complexity (RDO on), and the CABAC was applied for entropy coding. The QP was set to 30, which provided a suitable image quality on the Internet, and each image was encoded as one H.264 I-frame.

First, the patch size decision was made. The performance was tested by choosing N_p as equal to 4, 8, 16, and 32, as shown in

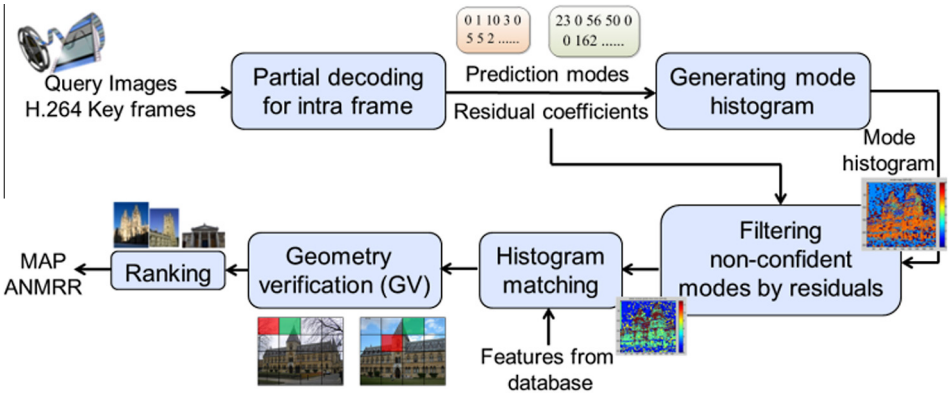


Fig. 7. The overall proposed system.

Table 1
Experimental results of suitable numbers of patches.

						
	MAP	ANMRR	MAP	ANMRR	MAP	ANMRR
4	0.092	0.823	0.157	0.793	0.421	0.502
8	0.128	0.773	0.214	0.629	0.515	0.396
16	0.104	0.822	0.155	0.746	0.507	0.424
32	0.051	0.918	0.076	0.852	0.333	0.574

Table 1. An N_p equal to 8 achieves the best retrieval performance based on two measures: mean average precision (MAP) [26] and the average normalized modified retrieval rank (ANMRR) [27].

The performance criterions MAP and ANMRR are briefly described as follows. Average precision (AP) is first defined as follows, to represent the retrieval performance of one query image

$$\mathrm{AP} = \int_0^1 P(R)\mathrm{d}R \tag{13}$$

where precision P(t) is the number of hit images h(t) over the number of retrieved images t, that is P(t) = h(t)/t; recall R(t) is the number of hit h(t) over the number of relevant images in database π, that is R(t) = h(t)/π. MAP is then the mean of AP for all query images. Higher MAP is, more accurate the system archives.

For ANMRR, average retrieval rank (AVR) is first defined as

$$\mathrm{AVR} = \frac{1}{\pi}\sum_{k=1}^{\pi}\mathrm{Rank}^*(k) \tag{14}$$

where $\mathrm{Rank}^*(k)$ is the weighted rank of the #k hit image, Then the normalized modified retrieval rank (NMRR) is the normalized score calculated from AVR, which falls into the range [0 1]. Finally ANMRR is the average NMRR for all query images, where lower ANMRR means the higher performance. The specific definition can be found in [27].

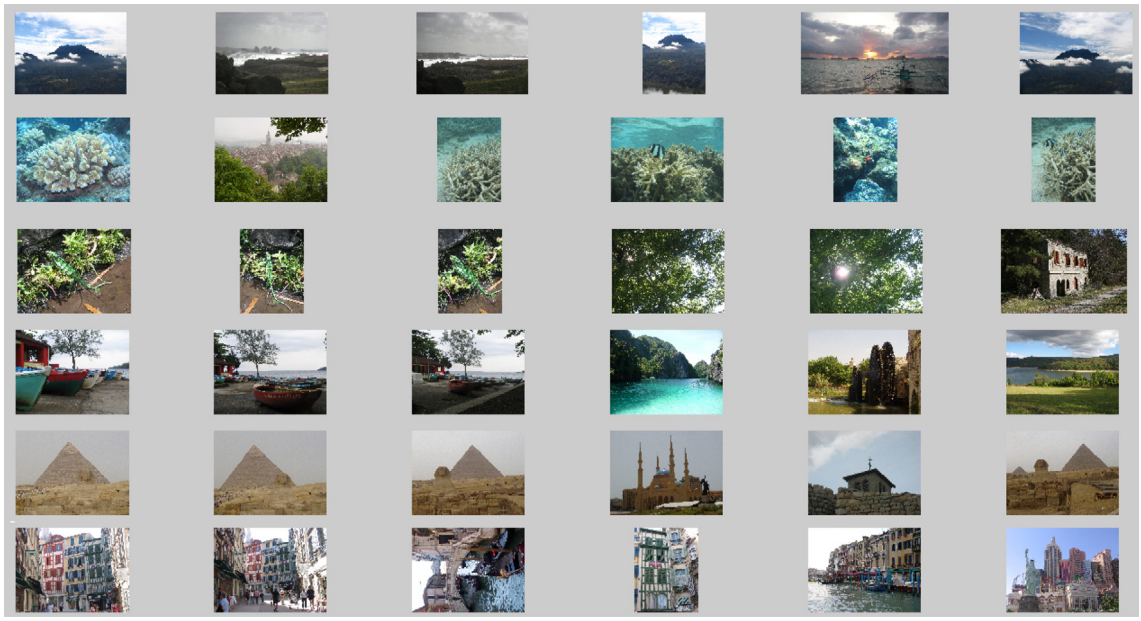
The performance was compared with the existing methods in the PXD and CPD. As shown in Table 2, the local mode histogram, H_{local}, and the feature vector consisting of residual information, H_{mode} and H_{total} as shown in Eq. (10), can gradually improve the performance. Finally, the proposed feature vector with the geometrical verification (GV) achieved a good accuracy of 0.3 (Oxford 5k) and 0.4 (INRIA Holiday) in the MAP. The proposed method has better performance than the related work [19] in CPD because the residual coefficient and local descriptor with geometrical verification are further concerned, and achieves an accuracy similar to the SIFT+k-means method [28][30] in the PXD.

We have provided subjective results for the proposed method H_{total} with GV. The samples of retrieval results are shown in Fig. 8, in which the query images include different kinds of scenes such as landscapes, buildings, and objects. It can be observed that relevant images can be effectively found in the first five ranks. Although some retrieved images may not be in the relevant set, they still look like the query image.

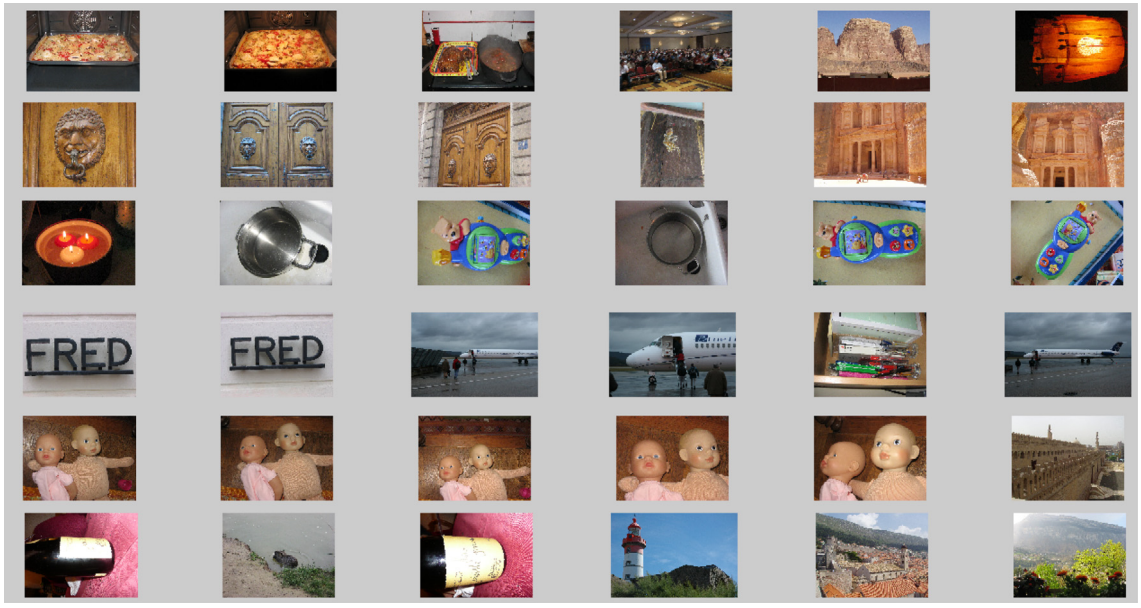
The proposed retrieval method in the CPD reduces the computational complexity in two aspects: (1) partial decoding and (2) direct feature extraction. According to computational complexity analysis of the decoding time in H.264 [29], 70% of the decoding time was eliminated by using the proposed method. The feature extraction in the PXD generally includes interest point detection, edge detection, and transformation. It requires heavy computational complexity that can be released by the proposed method.

Table 2
Performance comparisons with related works.

Method	Oxford 5K		INRIA Holiday	
	MAP	ANMRR	MAP	ANMRR
Color Hist.[2]	0.12	0.81	0.48	0.47
SIFT+k-means[28][30]	0.35		0.45	
H_{global}[19]	0.17	0.76	0.22	0.73
H_{local}	0.21	0.71	0.41	0.53
H_{mode}	0.25	0.66	0.38	0.56
H_{total}	0.27	0.66	0.41	0.54
H_{total} with GV	0.30	0.63	0.44	0.50



(a) Samples with outdoor scenes



(b) Samples with indoor scenes

Fig. 8. The samples of retrieval results, where the first image in each row is the query image; the others are the retrieved first five images.

For the memory requirements, the SIFT method produces 2.1×10^9 feature vectors for the Oxford 5k database [28]. However, the proposed method only generates 7.5×10^6 feature vectors and, hence, is considerably more suitable for real-time implementation. The details are presented in Table 3.

Table 3
Descriptor requirements for Oxford 5k in the proposed method and the method with PXD in [28].

	# of bins /feature	# of features /image	# of images in Oxford 5k	Total # of bins
SIFT+k-means [28]	128	3300	5062	2.1×10^9
Proposed	23	64		7.5×10^6

5. Conclusions

In this paper, the mode and residual information from H.264 coded images are used to extract picture features for image retrieval. The experiment results show that the performance of the proposed method is adequate for real-world applications and the memory requirements are considerably lower than those of the PXD approach. In the future, the proposed method can also be extended for the application in the retrieval of H.264 coded videos.

References

[1] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content-based image retrieval at the end of the early years, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 1349–1380.

[2] M. Swain, D. Ballard, Color indexing, Int. J. Comput. Vision 7 (1991) 11–32.

[3] M. Ferman, A.M. Tekalp, R. Mehrotra, Robust color histogram descriptors for video segment retrieval and identification, IEEE Trans. Image Process. 11 (2002) 497–508.

[4] S. Junding, X. Heli, Contour-shape recognition and retrieval based on chain code, in: Proceedings of International Conference on Computational Intelligence and Security, 2009, pp. 349–352.

[5] X.S. Zhou, T.S. Huang, Edge-based structural features for content-based image retrieval, Pattern Recogn. Lett. 22 (2001) 457–468.

[6] H. Wang, A. Divakaran, A. Vetro, S. Chang, H. Sun, Survey of compressed-domain features used in audio–visual indexing and analysis, J. Vis. Commun. Image Represent. 14 (2003) 150–183.

[7] G. Schaefer, D. Edmundson, Performance comparison of JPEG compressed domain image retrieval techniques, in: Processing of IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC), 2012, pp. 587–592.

[8] G. Schaefer, D. Edmundson, DC stream based JPEG compressed domain image retrieval, Lect. Notes Comput. Sci. 7669 (2012) 318–327.

[9] B. Baharudin, Effective content-based image retrieval: combination of quantized histogram texture features in the DCT domain, in: Proceeding of International Conference on Computer & Information Science (ICCIS), 2012, pp. 425–430.

[10] X.H. Zhang, G.C. Bian, W.B. Xu, A shape feature based image retrieval in DCT compressed-domain, in: Proceeding of Fifth International Conference on Computer and Information Technology, 2005, pp. 629–633.

[11] D. Zhong, I. Defe'e, DCT histogram optimization for image database retrieval, Pattern Recogn. Lett. 26 (2005) 2272–2281.

[12] L. Zhuo, J. Zhang, Y. Zhao, S. Zhao, Compressed domain based pornographic image recognition using multi-cost sensitive decision trees, Signal Process. 93 (2012) 2126–2139.

[13] A.K. Jain, Automatic caption localization in compressed video, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 385–392.

[14] C. Yeo, K. Ramchandran, Compressed domain video processing of meetings for activity estimation in dominance classification and slide transition detection, Technical Report Electrical Engineering and Computer Sciences University of California at Berkeley, 2008.

[15] J. Bescos, Real time shot change detection over online MPEG-2 video, IEEE Trans. Circuits Syst. Video Technol. 1 (2004) 475–484.

[16] F.-P. Wang, W.-H. Chung, G.-K. Ni, I.-Y. Chen, S.-Y. Kuo, Moving object extraction using compressed domain features of H.264 INTRA frames, in: Proceeding of IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance, 2012, pp. 258–263.

[17] W. Zeng, W. Gao, Shot change detection on H.264/AVC compressed video, in: Proceeding of IEEE International Symposium on Circuits and Systems (ISCAS), 2005, pp. 3459–3462.

[18] F.D. Simonea, M. Ouareta, F. Dufauxa, A.G. Tescherb, T. Ebrahimia, A comparative study of JPEG 2000, AVC/H.264, and HD photo, in: Proceeding of SPIE Optics and Photonics, Applications of Digital Image Processing, vol. 6696, 2007.

[19] F. Zargari, M. Mehrabi, M. Ghanbari, Compressed domain texture based visual information retrieval method for I-frame coded pictures, IEEE Trans. Consum. Electron. 56 (2010) 728–736.

[20] B. Girod, V. Chandrasekgar, R. Grzeszczuk, Y.A. Reznik, Mobile visual search: architectures technologies, and the emerging MPEG standard, IEEE Multimedia 18 (2011) 86–94.

[21] T. Wiegand, G.J. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H.264/AVC video coding standard, IEEE Trans. Circuits Syst. Video Technol. 13 (2003) 560–576.

[22] I. Richardson, The H.264 Advanced Video Compression Standard, second ed., Wiley, 2010.

[23] Y. Wang, An improved image edge detection algorithm based on H.264 intra prediction, in: Proceeding of International Conference on Intelligence Science and Information, Engineering, 2011, pp. 450–453.

[24] Oxford Buildings Dataset, available on: <http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/>.

[25] H.264/AVC Reference Software: Joint Model (JM), available on: <http://iphome.hhi.de/suehring/tml/index.htm>.

[26] M. Zhu, Recall, precision, and average precision, Dept. Stat. Actuarial Sci., Univ. Waterloo, CA, Tech. Rep. 9, 2004.

[27] P. N.-Nya, J. Restat, T. Meiers, J.-R. Ohm, A. Seyferth, R. Sniehotta, Subjective evaluation of the MPEG-7 retrieval accuracy measure (ANMRR), in: ISO/WG11 MPEG Meeting, Geneva, Switzerland, Doc. M6029, May 2000.

[28] J. Phillbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, in: Proceeding of Conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 18–23.

[29] Q. Xe, J. Liu, S. Wang, J. Zhao, H.264/AVC baseline profile decoder optimization on independent platform, in: Proceeding of International Conference on Wireless Communications, Networking, and Mobile, Computing, 2005, pp. 1253–1256.

[30] H. Jegou, M. Douze, C. Schmid, Hamming embedding and weak geometric consistency for large scale image search, Lect. Notes Comput. Sci. 5302 (2008) 304–317.