Rate-Distortion Optimized Sparse Coding With Ordered Dictionary for Image Set Compression

Xinfeng Zhang, *Member, IEEE*, Weisi Lin, *Fellow, IEEE*, Yabin Zhang, Shiqi Wang, *Member, IEEE*, Siwei Ma, *Member, IEEE*, Lingyu Duan, *Member, IEEE*, and Wen Gao, *Fellow, IEEE*

***Abstract*— Image set compression has recently emerged as an active research topic due to the rapidly increasing demand in cloud storage. In this paper, we propose a novel framework for image set compression based on the rate-distortion optimized sparse coding. Specifically, given a set of similar images, one representative image is first identified according to the similarity among these images, and a dictionary can be learned subsequently in wavelet domain from the training samples collected from the representative image. In order to improve coding efficiency, the dictionary atoms are reordered according to their use frequencies when representing the representative image. As such, the remaining images can be efficiently compressed with sparse coding based on the reordered dictionary that is highly adaptive to the content of the image set. To further improve the efficiency of sparse coding, the number of dictionary atoms for image patches is further optimized in a rate-distortion sense. Experimental results show that the proposed method can significantly improve the image compression performance compared with JPEG, JPEG2000, and the state-of-the-art dictionary learning-based methods.**

***Index Terms*— Image set compression, sparse coding, dictionary learning, rate-distortion optimization.**

I. Introduction

THE exponentially increasing demand of digital image and video services has been creating an ever stronger demand for image compression techniques, which target to achieve highly compact representation for images and videos by exploiting various types of redundancies [1]–[6]. Typical image coding methods usually compress images individually by reducing three types of redundancies within images, i.e., spatial redundancy, visual redundancy and statistical redundancy, such as JPEG [7], [8] and JPEG-2000 [9]. These redundancies can be referred as to intra-image redundancy. With the development of cloud storage [10]–[12], a huge amount of images are upload and stored in servers, and many of them are classified into different categories based on their content similarity by different image management applications, e.g., Google Photos and iCloud Photo Library. Besides intra-image redundancy, set redundancy [13] or inter-image redundancy exists among images in image set as well. Since the traditional compression methods only focus on achieving compact representation for each image individually by reducing intra-image redundancy, the inter-image redundancy has been largely ignored, which leads to a waste of storage space obviously.

Considering the cloud storage of images and videos, numerous cloud-based image and video processing and compression methods are proposed. Wang *et al.* [14] inferred the best contrast level by taking advantage of the retrieved images from cloud to guide image contrast enhancement process. Liu *et al.* [15] made full use of the similarity in both the low-resolution image itself and the cloud images to facilitate image super-resolution. Yue *et al.* [16] propose to describe an input image based on its down-sampled version and local feature descriptors, and the high resolution image can be reconstructed via retrieved similar image patches from cloud. This method can effectively reduce the bandwidth for image transmission.

In order to reduce set redundancy, various image set compression techniques also have been proposed in the literatures [13], [17]–[27], which have attracted more and more attentions recently. According to the prediction structures, these methods can be roughly classified into two categories, *i.e.,* central prediction and sequential prediction methods. Given a group of similar images, the methods with central prediction structure first select or construct one or more representative image(s) from them, and compress the representative image(s) independently with traditional image compression methods. In this manner, the remaining images can be compressed by referring to the decoded representative images and only the prediction residuals are entropy coded. By contrast, the sequential prediction structure methods take advantage of the video coding framework by reorganizing similar images into a sequence according to the prediction costs, such that each image should be decoded sequentially.

Karadimitriou proposed two construction methods for representative image, *i.e.*, max-min differential (MMD) method [17] and centroid method [18], to reduce the set redundancy.

Manuscript received June 23, 2016; revised January 20, 2017 and May 5, 2017; accepted August 25, 2017. Date of publication September 1, 2017; date of current version December 4, 2018. This work was carried out at the Rapid-Rich Object Search (ROSE) Laboratory, Nanyang Technological University, Singapore. The ROSE Laboratory was supported by the National Research Foundation, Singapore, under its Interactive Digital Media Strategic Research Programme. This work was also cooperated with Peking University, and in part supported by the National Natural Science Foundation (61632001) and National Basic Research Program of China (973 Program) under Grant 2015CB351800. This paper was recommended by Associate Editor J. Ostermann. (*Corresponding author: Weisi Lin.*)

X. Zhang, W. Lin, and Y. Zhang are with the Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore (e-mail: xfzhang@ntu.edu.sg; wslin@ntu.edu.sg; zhan0398@e.ntu.edu.sg).

S. Wang is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: shiqwang@cityu.edu.hk).

S. Ma, L. Duan, and W. Gao are with the School of Electronic Engineering and Computer Science, Institute of Digital Media, Peking University, Beijing 100871, China (e-mail: swma@pku.edu.cn; lingyu@pku.edu.cn; wgao@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2017.2748382

The MMD method constructs two representative images, 'min' image and 'max' image, which are generated by choosing the minimum and maximum pixel values for every pixel position from all the similar images. For each image, the original pixel values are replaced by the difference from either the 'min' image or the 'max' image, and then the difference values are compressed with entropy coding. However, this method needs to compress extra two representative images. The centroid method constructs only one representative image by averaging the pixel values in the same position among all the images, and the difference images between the representative image and other similar images are compressed individually. Yeung *et al.* [19] created a new low frequency template as the representative image by averaging the low frequency components of all images to capture most of the common patterns among similar images while discarding variation content. Li *et al.* [21] further extended the low frequency template method by decomposing original images and the low frequency template into different resolutions and predict the images at certain resolution. However, these methods cannot handle images with large scale geometric deformations, and also need to compress extra representative image(s). Therefore, the above methods are mainly limited to compress medical images, which are well aligned.

To deal with motion among similar images, sequential prediction structure for image set compression is proposed in the literatures [22], [24]–[27]. Nielsen and Li [22] utilized the root mean square error as the inter-image redundancy measure to construct a minimum spanning tree (MST) [23], and derived the image coding order by traversing the MST. Subsequently, these images are compressed sequentially by coding the difference between images and their previously encoded images using JPEG2000. Zou *et al.* [24] further improved the inter-image redundancy measure by performing motion estimation among images and used HEVC codec to compress image sequence. Considering that irregular and large scale motions among similar images make traditional local motion estimation inefficient, Shi *et al.* [25]–[27] utilized the distance of matched local feature descriptors between any two images to measure the inter-image redundancy, and introduced the geometric and photometric transform to improve inter-prediction efficiency. Though significant improvement has been achieved compared to individually coding methods, these approaches still aim at finding similar content directly from reference images in the same set by brute force searching, which may be inefficient. When multiple objects with different large scale motions, they are difficult to find the redundancy parts from other images efficiently [28]. Especially, the luminance changes and resolution diversity make the similar content retrieval more difficult.

Recently, image sparse coding with learned overcomplete dictionaries shows promising results on image compression [29]–[32] by representing images with dictionary atoms compactly. In this novel compression framework, a general dictionary is firstly learned from a lot of images, and an image can be compressed by representing its non-overlapping image patches as linear combination of very few dictionary atoms, which is called sparse coding. Since the atoms in dictionaries are learned from training samples by approximating them with coefficients as few as possible, the similar content with that in training images can be compressed more efficiently. Therefore, sparse coding with dictionary learning provides a promising solution to image set compression by learning specific dictionaries for different image sets. Bryt and Elad [33] took advantage of K-SVD method [34] to training a dictionary offline from image patches in pixel domain, and applied it to compress facial images, which significantly improves the compression performance compared with JPEG and JPEG2000. Skretting and Engan [35] compared the compression efficiency of pixel domain dictionary and wavelet domain dictionary, and showed that the wavelet domain dictionary achieved better compression performance.

In this paper, we propose a novel framework with dictionary learning based sparse coding to tackle the image set compression problem. In the proposed framework, a specific dictionary is learned from a representative image for every image set, and the other images are compressed by sparse coding with the corresponding dictionary. In order to improve the compression performance, there are two main contributions in this paper. Firstly, considering the similarity among images in the same set, we reorder the dictionary atoms according to their use frequencies in sparse coding for the representative image by making the frequently used atoms centralized to the front. With the reordered dictionary, the *run-level* coding can be more efficient for compressing the sparse coefficients. Second, considering different image patches with different sparsity, we propose a rate-distortion based orthogonal matching pursuit (RD-OMP) method to assign different numbers of atoms to different image patches according to their rate-distortion costs. The basic idea of this paper has been introduced in our conference version [36], and the motivation and rationality of these contributions are analyzed detailedly in this paper, and more analysis and experiments are provided in this paper to show the philosophy and the efficiency of the proposed method.

The remainder of this paper is organized as follows. In Section II, we briefly review the background of image compression with dictionary learning, and then introduce the proposed framework for image set compression. Section III gives the detailed introduction for the proposed dictionary reordering, rate-distortion based orthogonal matching pursuit method and the corresponding entropy coding method, respectively. Experimental results are reported in Section IV and Section V concludes the paper.

II. The Proposed Framework for Image Set Compression

A. Image Compression With Dictionary Learning

Image compression with a learned overcomplete dictionary utilizes sparse coding to approximate every image patch with linear combinations of few dictionary atoms. Considering the good adaptability of learned overcomplete dictionaries, many related image compression methods have been proposed in the literatures, *e.g.*, [33], [37]. In traditional methods, the overcomplete dictionary is learned from a collection of

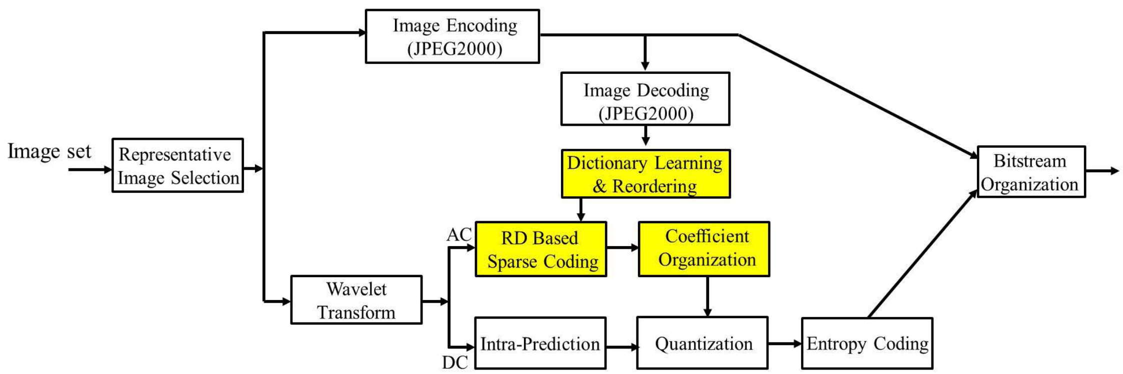


Fig. 1. The proposed framework for image set compression with adaptive dictionary learning.

image patches, $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_L\}$, by solving the following optimization problem [38],

$$f(\mathbf{D}, \boldsymbol{\alpha}) = \min_{\mathbf{D}, \boldsymbol{\alpha}} \sum_{i=1}^{L} \|\boldsymbol{x}_i - \mathbf{D}\vec{\alpha}_i\|_2^2 + \lambda \|\vec{\alpha}_i\|_0, \quad s.t.\ \|\boldsymbol{d}_i\|_2 \le 1. \tag{1}$$

Here $\boldsymbol{x}_i \in R^M$ is the vectorization of the i^{th} image patch and $\mathbf{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_N] \in R^{M \times N} (M < N)$ is the learned dictionary with N atoms. $\vec{\alpha}_i \in R^N$ is the coefficient vector corresponding to image patch $\boldsymbol{x}_i$, and it has very few non-zero values, and $\boldsymbol{\alpha} = [\vec{\alpha}_1, \vec{\alpha}_2, \cdots, \vec{\alpha}_L]$. λ is the regularization parameter to trade off the data fitting term and the regularized sparsity term corresponding to the first and the second term respectively.

The joint optimization problem in Eqn.(1) can be solved by iteratively optimizing the sparse coding coefficient matrix $\boldsymbol{\alpha}$ and dictionary $\mathbf{D}$, respectively, *i.e.*, solving the sub-optimization problems in Eqn.(2) and Eqn.(3),

$$\boldsymbol{\alpha}^{(k)} = \min_{\boldsymbol{\alpha}} \sum_{i=1}^{L} \|\boldsymbol{x}_i - \mathbf{D}^{(k-1)}\vec{\alpha}_i\|_2^2 + \lambda \|\vec{\alpha}_i\|_0, \tag{2}$$

$$\mathbf{D}^{(k)} = \min_{\mathbf{D}} \sum_{i=1}^{L} \|\boldsymbol{x}_i - \mathbf{D}\vec{\alpha}_i^{(k)}\|_2^2, \quad s.t.\ \|\boldsymbol{d}_i\|_2 \le 1, \tag{3}$$

where $\boldsymbol{\alpha}^{(k)}$ and $\mathbf{D}^{(k)}$ represent the k^{th} iteration results. Many optimization algorithms have been proposed to solve the above problems in Eqn.(2) and Eqn.(3), *e.g.*, orthogonal matching pursuit (OMP) [39] for the sparse coding problem in Eqn.(2), and RLS-DLA [40] for the dictionary update problem in Eqn.(3).

A typical image compression method with learned dictionary in wavelet domain is proposed in [35]. The dictionary is learned from the image patches $\{\boldsymbol{x}_i\}$, which are transformed into wavelet domain and subtracted their DC components. For an image $\mathcal{I}$, it is firstly divided into non-overlapping patches, $\{\boldsymbol{x}_i\}$, which are also transformed into wavelet domain. The DC components of image patches are compressed with DPCM method, while the AC components are represented by sparse approximation with the learned dictionary, and the non-zero coefficients are further quantized and entropy coded to reduce the psychovisual redundancy and statistical redundancy, respectively. A given error threshold, ϵ, is usually utilized as stopping criterion for the sparse coding of every image patch independently,

$$\min \|\alpha_i\|_0, \quad s.t.\ \|\boldsymbol{x}_i - \mathbf{D}\alpha_i\|_2 \le \epsilon,\ i = 1, 2, \ldots, L, \tag{4}$$

where L is the number of image patches. Due to the sparsity of the coefficients, $\{\alpha_i\}$ are usually coded with the *run-level* method.

B. The Proposed Framework for Image Set Compression With Dictionary Learning

In this paper, we proposed a new framework for image set compression utilizing specific dictionaries learned from image sets, as illustrated in Fig.1. For an image set, one representative image is firstly selected from them according to its similarity with others, and it should share similar content with others as much as possible. The representative image is compressed with traditional image compression methods, e.g., JPEG or JPEG2000. Then, a specific dictionary for the given image set is learned from the image patches extracted from the decoded representative image. In this paper, we compress the representative images with JPEG2000 at high bitrate, about 0.8 bpp. Considering the better performance of wavelet domain dictionary [35], the image patches extracted from the representative image are first transformed into 9/7 wavelet domain and subtracted their DC components. Then, the dictionary can be learned for the AC components by solving the optimization problem in Eqn.(1) with the method in [40]. Finally, the dictionary atoms are further reordered according to the number of times they are used in sparse coding for the representative image.

Except for the representative image, the other images are compressed based on the reordered dictionary. Firstly, an image is divided into non-overlapping patches, which are further transformed into wavelet domain. The DC components of image patches are compressed by simple intra-prediction, while the AC components of image patches are compressed by sparse coding with the reordered dictionary. During the sparse coding, the proposed rate-distortion based orthogonal matching pursuit (RD-OMP) method is utilized to adaptively assign different number of atoms to image patches. At the decoder side, the representative image should be first decoded, and the dictionary is retrained from it. Then, the other images can be decoded with the learned dictionary. This method can avoid compressing dictionary for every image set, but it also incurs extra computational burden for dictionary learning. Fortunately, most of the images in cloud are rarely accessed, even never accessed, but they must be preserved, e.g., medical images. Therefore, the storage efficiency is more important for them than computational complexity.

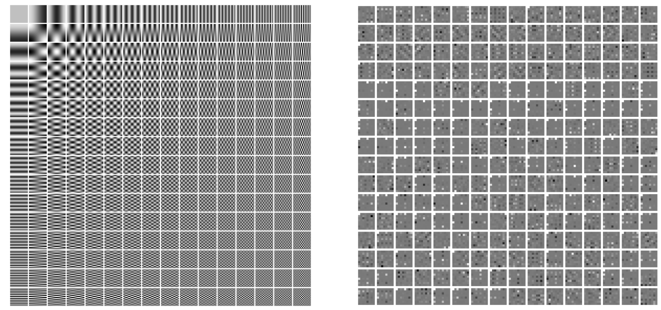
In order to improve the efficiency of the dictionary in representing other images, the representative image should share the common content or structures among other images as much as possible. There are numerous methods proposed to determine or construct a representative image [22], [24], [26], [41]. In this paper, we take the distance of matched Scale-Invariant Feature Transform (SIFT) [42] features to measure image similarity considering its good performance in visual description for images, even though other visual descriptors, e.g., low complexity descriptor CDVS [43], [44], are also applicable. The image with the minimum distance to all the other images is selected as the representative one of the image set. In [42], Lowe utilized the scale-space extremum in the difference of Gaussian filter convolved with images as the key points, and calculated SIFT descriptor to describe image local characteristic with a 128-dimensional (128-D) vector, which is composed of the histogram of gradient directions in the 16×16 area around the key point. Therefore, the k^{th} SIFT descriptor in one image can be formulated as $f(k) = \{x(k), y(k), s(k), o(k), \boldsymbol{u}(k)\}$, where $x(k)$ and $y(k)$ are the horizontal and vertical coordinates of the SIFT point, $s(k)$ and $o(k)$ are the scale and dominant direction of the local region around SIFT point, and $\boldsymbol{u}(k)$ is the 128-D SIFT descriptor vector. For an image set, $G = \{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_n\}$, we extract the SIFT features for every image, $F = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_n\}$. The similar content between any two images can be retrieved by identifying the corresponding SIFT points. Specially, for two images, $\mathcal{I}_i$ and $\mathcal{I}_j$, when we match one SIFT descriptor $f_i(k_t)$ in image $\mathcal{I}_i$ to SIFT descriptors in image $\mathcal{I}_j$, the nearest neighbor and the second-closest neighbor for $f_i(k_t)$ in image $\mathcal{I}_j$ are denoted as $f_j(k_m)$ and $f_j(k_n)$. If $f_i(k_t)$ and $f_j(k_m)$ is a pair of matched SIFT descriptors, they should conform to the following requirement,

$$d\left(f_i(k_t), f_j(k_m)\right) \leq \beta d\left(f_i(k_t), f_j(k_n)\right), \tag{5}$$

where β is a constant and $d(\cdot)$ is the Euclidean distance of two SIFT descriptors. The dissimilarity of two images can be measured by the average of all the matched SIFT descriptors,

$$\bar{d}\left(\mathcal{I}_i, \mathcal{I}_j\right) = \sum_{(k_t,k_m)\in\Omega_{i,j}} d\left(f_i(k_t), f_j(k_m)\right). \tag{6}$$

where $\Omega_{i,j}$ is the set of matched SIFT points between image $\mathcal{I}_i$ and $\mathcal{I}_j$. The image in one set has the minimum distance to all the other images are selected as representative image. In the following section, we will introduce the dictionary learning based compression method for non-representative images.



(a) (b)

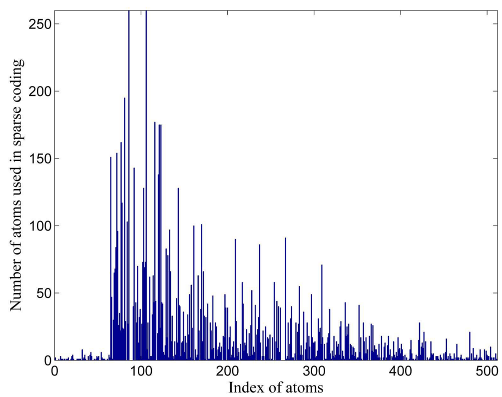
Fig. 2. Atoms in different transform matrix, (a) DCT, (b) learned dictionary.

III. Dictionary Learning Based Image Set Compression

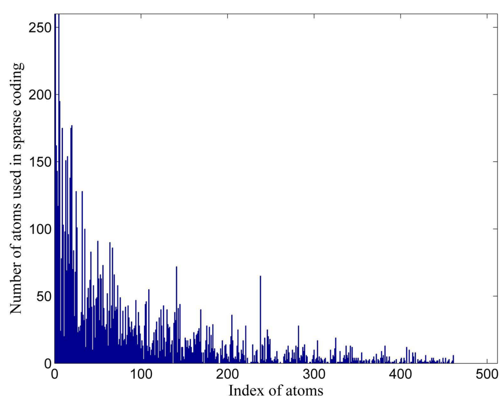
A. Dictionary Reordering

The atoms in traditional orthogonal transformation are usually arranged according to the frequency variation, e.g., atoms in DCT matrix illustrated in Fig.2(a). The non-zero coefficients of the transformed images with DCT are concentrated at the low frequency bands, which makes the coefficients can be coded efficiently with *run-level* method. However, such neat property does not hold for the atoms in an adaptively learned dictionaries. Since the dictionary atoms do not have obvious regular variations in frequency bands, especially for the dictionary learned from AC components of wavelet transform, e.g., Fig.2(b) illustrating the atoms learned from 16×16 patches only with AC components. Although the learned dictionaries are more powerful than DCT in representing image with sparse coefficients, the irregularly distributed non-zero coefficients are more difficult to compress compared with DCT coefficients, which can be efficient compressed by scanning them into 1-D array according to Zig-Zag order.

The main difference between our proposed framework and traditional dictionary learning based image compression is that the images to be coded share similar content or structures with the representative image, which has been compressed with traditional image compression method. Although the atoms are irregularly distributed, they still preserve different utilization frequencies in sparse representation for a special kind of images. Fig.3(a) illustrates the histogram of the atoms used in representing an image in image set *MailRoom*, where some atoms utilized more frequently than others obviously. Based on the assumption that images with similar contents usually share similar atom distribution in sparse coding, we propose to reorder the atoms in the learned dictionary according to the number of times they are used in sparse coding of the representative image to improve the coding efficiency for the non-zero coefficients. The atoms with more use times utilized in sparse coding of the representative image are arranged at the front of the dictionary, while the ones with fewer use times are placed at the back of the dictionary. Furthermore, we directly remove the atoms when its use time is smaller than a threshold, τ, in sparse coding of the representative image to shorten the length of the coefficient vectors. Fig.3(a) and Fig.3(b) show the atom use times for sparse coding the same image with



(a)



(b)

Fig. 3. (a) The histogram of the atoms used in sparse coding image, *MailRoom*, with unordered dictionary, the variance of the *run* in *run-level* coding of non-zero coefficients, 1.13×10^4, (b) the histogram of the atoms used in sparse coding with reordered dictionary, the variance of the *run* of non-zero coefficients, 0.92×10^4.

unordered dictionary and reordered dictionary respectively. We can see that when applying the reordered dictionary to other images, the non-zero coefficients are centralized to the front atoms of the dictionary, which can benefit the *run-level* coding. To verify this conclusion, we calculate the variance of the variable, *run*, in *run-level* coding. The variance of *run* with the reordered dictionary significantly decreases compared with that generated from unordered dictionary, which implies improvement of the coding efficiency.

B. Rate-Distortion Based Orthogonal Matching for Image Sparse Coding

In existing sparse coding methods, e.g., OMP [39] and least angle regression [45], they solve the optimization problem in Eqn.(4), by minimizing the number of non-zero coefficients with the same distortion threshold for every image patch, without considering rate-distortion costs for different patches. Since the same dictionary has different efficiencies in representing image patches, it will lead to different amount of distortion reduction when assigning the same amount atoms to different image patches [46]. Therefore, a global parameter to control the distortion or number of non-zero coefficients for all the image patches equally is not optimal to image compression. Fig.4 shows the relationship between distortions

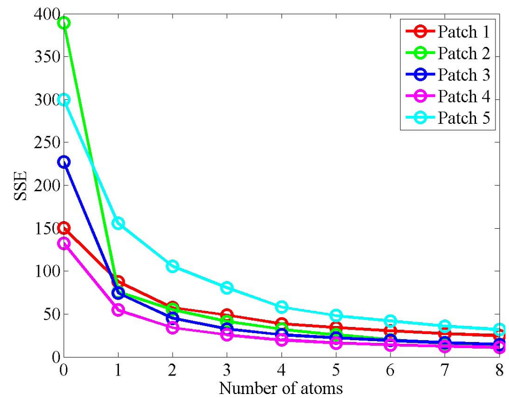


Fig. 4. The relationship between distortion reduction and the number of atoms used in sparse coding.

of reconstructed image patches and the amount atoms used in sparse coding. Although the distortion of image patches decreases along with the number of atoms increasing, the distortion reduction rate is different for image patches, which makes it possible to improve the compression performance by assigning different number of dictionary atoms for image patches adaptively.

In order to achieve better compression performance, we propose a rate-distortion optimized OMP method to assign different number of non-zero coefficients to different patches according to the rate-distortion costs of image compression. The objective function of the proposed sparse coding method is given by

$$\min R, \quad s.t. \sum_{i=1}^{L} \|\boldsymbol{x}_i - \mathbf{D}\vec{\alpha}_i\|_2^2 \leq \Upsilon, \tag{7}$$

where R is the bits used in coding all the non-zero coefficients, L is the amount of image patches to code, and $\boldsymbol{x}_i$ is the transformed image patch without DC component. Υ is the given distortion for the whole image. The proposed objective function is different from the traditional one in Eqn.(4) by constraining the whole distortion of image instead of the equal distortion for every patch in sparse coding.

The exact number of bits utilized to code non-zeros coefficients in entropy coding can only be obtained after the coding process, which is impossible for solving the problem in Eqn.(7). Fortunately, the coding rate can be well estimated based on coefficient statistic characteristics [47]. Therefore, we first formulate a rate model based on the L0 norm of non-zero coefficients to predict the amount of coding bits. In this paper, we take the *run-level* coding to organize the non-zero coefficients, and utilize the improved Huffman coding with recursive splitting method [48] to compress the (*run, level*) pairs. Fig.5 illustrates the relationship between the L0 norm of non-zero coefficients and the corresponding bits consumed in real entropy coding process for test images. We can see that there are almost linear correlation between non-zero coefficients and consumed bits image compression, which indicates that the number of non-zero coefficients can well predict the coding bits. With the L0 norm instead of coding bits,

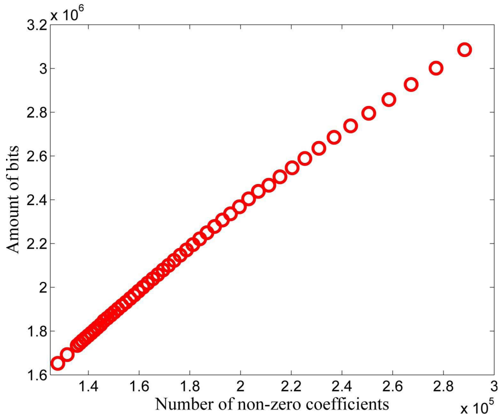


Fig. 5. The relationship between the amounts of atoms used in sparse coding and the compression bits for the image *MailRoom*.

the objective function can be rewritten as,

$$\min \sum_{i=1}^{L} |\vec{\alpha}_i|_0, \quad s.t. \sum_{i=1}^{L} \|\boldsymbol{x}_i - \mathbf{D}\vec{\alpha}_i\|_2^2 \le \Upsilon. \tag{8}$$

Traditional OMP algorithm is widely used to solve the sparse coding problem in Eqn.(4). It first selects the atom with the highest correlation to the input signal, then the signal is orthogonally projected to the atom to calculate the residual. At each step after that, a new atom with the highest correlation to the current residual is selected, and the input signal is also orthogonally projected to the spanned space of all the selected atoms. The residual is recomputed, and the process repeats until achieving the given threshold for distortion or number of atoms for an image patch.

In this paper, we propose a greedy algorithm based on OMP method to solve the optimization problem in Eqn.(8). The algorithm can be divided into three stages sequentially, i.e., initialization, OMP sparse coding and heap update. At the first stage, we find the highest correlated atom, $\boldsymbol{d}_{\boldsymbol{x}_i,0}$, for every image patch, $\boldsymbol{x}_i$, and calculate the corresponding residual $\boldsymbol{r}_i^{(0)}$ as follows,

$$\boldsymbol{r}_i^{(0)} = \boldsymbol{x}_i - \vec{\alpha}_i^{(0)} \boldsymbol{d}_{\boldsymbol{x}_i,0}. \tag{9}$$

The amount of distortion reduction, $DistReduction[i]$, for the i^{th} image patch is calculated and organized into a maximum heap structure. At each step after that, we only process the image patch corresponding to the first element of the heap with OMP sparse coding by increasing one atom each time, and update the maximum heap. In the OMP sparse coding stage, we select the highest correlated atom with current signal at the top of the heap, $\boldsymbol{r}_i^{(k)}$, and calculate the corresponding coefficient vector with all the selected atoms, $\vec{\alpha}_i^{(k)}$. In the heap update stage, the image patch signal at the top of heap is first projected to the space spanned by all the selected atoms at present to calculate the new residual. Then, we update the distortion reduction for the first element of the heap, and adjust the heap to a maximum heap again. The two stages, OMP sparse coding and heap update, can be performed iteratively until achieving the given amount of distortions or number of non-zero coefficients for the whole image. The proposed rate-distortion optimized sparse coding algorithm is formulated in **Algorithm 1**.

Algorithm 1 Rate-Distortion Optimized Orthogonal Matching Pursuit

Input:
Dictionary $\mathbf{D}$={$\boldsymbol{d}_i$};
Image patches: {$\boldsymbol{x}_i$};
Distortion threshold: Υ;
Variable: $DistReduction[i]$, representing the distortion reduction of image patches, $\boldsymbol{x}_i$, when increasing one atom.
Initialization:
1. Select one highest correlated atom for every image patch and calculate the coefficients;
2. Calculate the residual of every image patch with the selected atom, $r_i^{(0)}$;
3. Calculate the distortion reduction for every image patch with the selected atom, $DistReduction[i]$;
4. Build a maximum heap for the image patches according to $DistReduction[i]$, H;
while ($\sum_{i=1}^{L} \|\boldsymbol{x}_i - \boldsymbol{D}\vec{\alpha}_i\|_2^2 \le \Upsilon$) **do**
OMP sparse coding:
1. Select the highest correlated atom, $\boldsymbol{d}_{\boldsymbol{x}_i,k}$, for residual signal, $\boldsymbol{r}_i^{(k)}$, which is at the top of heap, H;
2. Calculate the corresponding coefficient vector, $\vec{\alpha}_i^{(k)}$;
Heap Update:
1. Project the image patch signal corresponding to the top element of heap to the space spanned by all selected the atoms orthogonally, update the residual signal, $\boldsymbol{r}_i^{(k+1)}$;
2. Update the distortion reduction when increasing the new atom, $DistReduction[i]$;
3. Adjust the heap H into maximum heap with the updated $DistReduction[i]$, and $k = k + 1$;
end
Output: Sparse coefficient matrix of image patches, $\boldsymbol{\alpha}$

Compared with the traditional OMP method, we only increase a heap building and adjust procedure, the complexity of which is $O(LlogL)$ on average. Specifically, the extra computation is a heapsort operation with about $logL$ comparison operations for each heap adjustment process on average. An extra buffer with size of L is needed to store the values of distortion reduction for L image patches. Therefore, the proposed RD-OMP method only entails very few extra computation complexity and limited buffers, which is similar with that of OMP in computation complexity and buffer consumption.

C. Quantization and Entropy Coding for Sparse Coefficients

To efficiently compress the signals in DC and AC components, the uniform quantization is applied to them to further improve the compression ratio. For a given quality level, we determine the quantization step for DC and AC components respectively to achieve the given quality for them respectively.

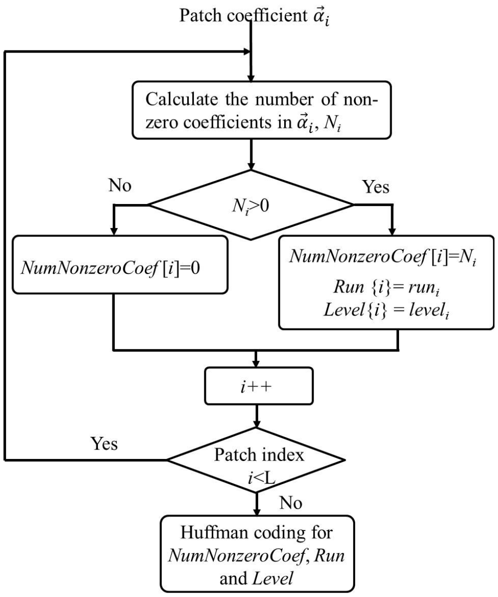


Fig. 6. The syntax construction in entropy coding for sparse coefficients.

Since the signals in DC and AC have different distribution characteristics, the quantization steps in DC and AC are usually different. In the proposed method, for an 8 bit image, it is first subtracted 128 for each pixel, and then transformed with 9/7 wavelet into 3 layers. Therefore, the signal range of DC component is $[-1024, 1016]$, and the range for AC component is $[-510, 510]$. The maximum bits for symbols are 11 and 10 for DC and AC components respectively, but most of the signals are concentrated around zero, which only need very short code words to be represented.

Since image patches are approximated by very few atoms, the *run-level* coding method is suitable to compress these sparse coefficients, which codes a run-length of zeros followed by a nonzero level. In this paper, we also utilize *run-level* coding to organize sparse coefficients into two 1-D arrays with *Run* and *Level*, respectively. Since the non-zero coefficients are few and different in image patches based on the proposed RD-OMP method, there may be too many consecutive zeros in some neighboring image patches, leading to very large value in run-length. Therefore, we introduce a new syntax element, *NumNonzeroCoef*[i], to indicate the number of non-zero coefficients for image patch, $\boldsymbol{x}_i$. The $(run_i, level_i)$ pair for the i^{th} image patch is compressed only when the value of *NumNonzeroCoef*[i] is not equal to 0.

Fig.6 shows the basic flow chart for the syntax construction in entropy coding for sparse coefficients. For quantized sparse coefficient vector $\vec{\alpha}_i$, if there are not non-zero coefficients, we only need to store 0 in the corresponding variable *NumNonzeroCoef*[i], which indicates that there is no coefficient to encode. If there are N_i non-zero coefficients in the coefficient vector, we first store the number of non-zero coefficients in the variable *NumNonzeroCoef*[i], and then store the run and $level$ in $Run\{i\}$ and $Level\{i\}$ respectively. After all the image patches are processed, we construct three Huffman tables for *NumNonzeroCoef*, *Run* and *Level* respectively according to the symbol frequencies, and then compress them using Huffman coding method [48].

TABLE I

INFORMATION OF IMAGE SETS USED IN EXPERIMENTS

Image sets	Resolution	Number	Scene
CastleEntry	3072×2048	30	Building
MallRoom	1024×576	7	Indoor
RockBoat	2560×1728	20	Lake and boat
WadhamCollege	1024×768	5	Building and grass
Fountain	3072×2048	11	Building
Herzjesu	3072×2048	25	Building
StudentHousing	1072×712	8	Building
Lakes	2560×1920	8	Lake and mountain
CoralReef	2048×1536	9	Coral reef and fish

Fig. 7. Example images in each image set, *CastleEntry*, *MallRoom*, *RockBoat*, *WadhamCollege*, *Fountain*, *Herzjesu*, *StudentHousing*, *Lakes*, *CoralReef*, from left-top to right-down, respectively.

IV. EXPERIMENTAL RESULTS

In order to verify the performance of the proposed method, we collect some different types of image sets, including buildings, indoor scene, mountains, lakes and grass, which have been used in other image set compression works [26], [27]. The detailed information about these test images are listed in TABEL I. Fig.7 shows one representative image for every image set. These images are transformed into gray ones, and compressed by popular image compression methods and the proposed method.

We compare our method with the most popular image compression methods, JPEG[1] and JPEG2000[2] and the representative dictionary learning based image compression method RLS-DLA [35], which is the most similar with our method utilizing wavelet domain dictionary. In order to verify the efficiency of our contributions, RLS-DLA utilize two kinds of dictionaries. One dictionary is offline learned from 8×8 image patches in wavelet domain by removing DC components, which are extracted from 50 different kinds of images. We denote this compression method as *RLS-DLA-D1*. The other kind of dictionaries is learned from the representative image for each image set. We denote this compression method

[1]JPEG codec, http://www.ijg.org

[2]JPEG2000 codec, http://www.openjpeg.org

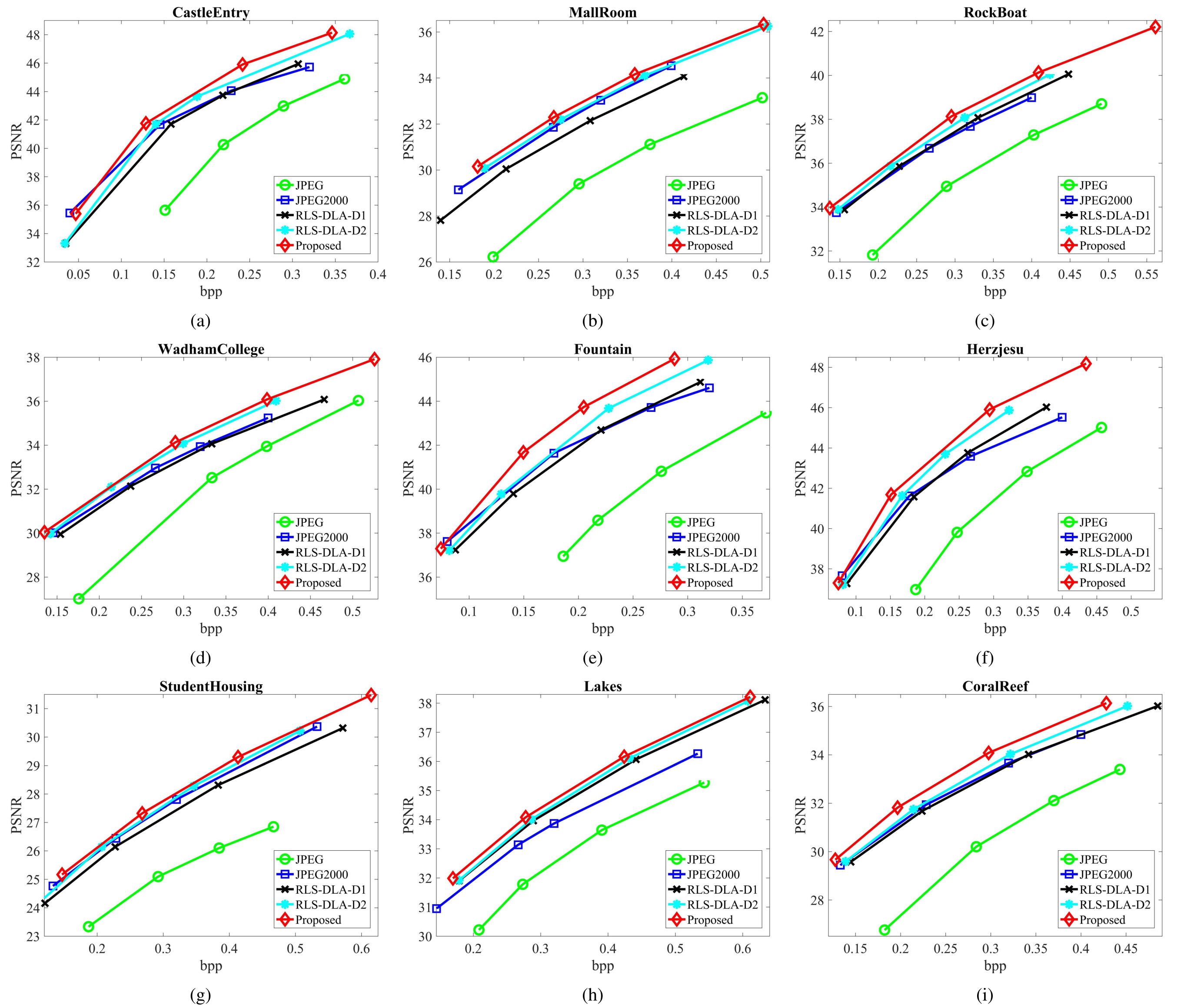


Fig. 8. Performance comparison on different image sets, (a) *CastleEntry*, (b) *MallRoom*, (c) *RockBoat*, (d) *WadhamCollege*, (e) *Fountain*, (f) *Herzjesu*, (g) *StudentHousing*, (h) *Lakes*, (i) *CoralReef*.

as *RLS-DLA-D2*. Here, $D1$ and $D2$ represent different dictionaries. All the dictionaries used in different methods have 512 atoms, i.e., 64×512.

Fig.8 shows the curve of the average PSNR and bit-rate for all the images in every image set. Our proposed method achieves up to 4 dB and 1 dB improvement at the same bit-rate compared with JPEG and JPEG2000, which are the popular image compression standards. From the comparison with *RLS-DLA-D1* and *RLS-DLA-D2*, we can see that the compression performance with dictionary learned from representative image is much better than that learned from some general images. This verifies that the dictionary from representative image can efficiently reduce the inter-image redundancy and improve the compression performance of traditional dictionary learning based image compression methods. The proposed method with dictionary reordering and RD-OMP further outperforms *RLS-DLA-D2* at relative large bit-rate range, which directly verifies that the proposed dictionary reordering and RD-OMP sparse coding are useful in improving coding efficiency. The results also show that the learnt dictionary based image compression methods achieve significant bitrate saving at middle and high bitrate compared with JPEG2000, while at low bitrate case, the improvement is not so significant as that at middle and high bitrate cases. This is mainly because that at low bitrate, most of the image patches are with very few non-zero coefficients, even only with the DC components. Therefore, the high efficiency dictionary cannot show its performance, and the proposed method dictionary reordering and RD-OMP also play less effect. In addition, the EBCOT in JPEG2000 is a more efficient entropy coding method than Huffman for wavelet coefficients, especially at low bit-rate case [49].

In Table II, we show a more detailed numerical comparison results for different methods, and also compare the proposed

TABLE II

PERFORMANCE COMPARISON FOR DIFFERENT IMAGE COMPRESSION METHODS COMPARED WITH JPEG

Image sets	JPEG2000		RLS-DLA-D1		RLS-DLA-D2		RLS-DLA-D3		Proposed	
	BD-RATE	BD-PSNR	BD-RATE	BD-PSNR	BD-RATE	BD-PSNR	BD-RATE	BD-PSNR	BD-RATE	BD-PSNR
CastleEntry	-34.36%	3.22	-33.53%	3.20	-46.10%	4.01	-49.83%	4.29	-55.57%	4.68
MallRoom	-38.08%	3.27	-27.44%	2.68	-39.70%	3.36	-40.14%	3.40	-44.57%	3.68
RockBoat	-30.36%	2.33	-31.29%	2.39	-36.51%	2.68	-38.93%	2.87	-43.71%	3.11
WadhamCollege	-26.82%	2.35	-22.76%	2.13	-32.67%	2.69	-33.57%	2.73	-37.41%	2.96
Fountain	-46.67%	3.63	-42.04%	3.43	-51.41%	3.96	-52.18%	4.05	-63.98%	4.86
Herzjesu	-36.90%	3.06	-38.47%	3.22	-48.54%	3.87	-48.51%	3.85	-57.29%	4.50
StudentHousing	-47.03%	2.42	-38.77%	2.10	-48.60%	2.49	-48.60%	2.49	-52.62%	2.67
Lakes	-26.35%	1.32	-37.78%	1.87	-37.97%	1.82	-38.20%	1.89	-43.45%	2.15
CoralReef	-39.34%	3.01	-36.92%	2.90	-42.49%	3.18	-42.98%	3.21	-51.48%	3.69
Average	-36.21%	2.73	-34.33%	2.66	-42.67%	3.12	-43.66%	3.20	-50.01%	3.59

TABLE III

PERFORMANCE COMPARISON BETWEEN HEVC AND THE PROPOSED METHOD (THE ANCHOR IS THE PROPOSED METHOD)

Image sets	HEVC AI		HEVC LDP	
	BD-RATE	BD-PSNR	BD-RATE	BD-PSNR
CastleEntry	-29.08%	1.72	-34.84%	2.28
MallRoom	-40.67%	3.64	-47.13%	4.28
RockBoat	-29.76%	2.12	-29.88%	2.09
WadhamCollege	-30.74%	2.44	-33.60%	2.67
Fountain	-20.67%	1.41	-25.08%	1.74
Herzjesu	-20.48%	1.36	-22.28%	1.47
StudentHousing	-32.72%	2.36	-35.82%	2.69
Lakes	-26.96%	1.70	-31.11%	2.04
CoralReef	-13.76%	0.87	-21.22%	1.37
Average	-27.21%	1.96	-31.22%	2.29

dictionary reordering and RD-OMP sparse coding respectively based on the BD-PSNR and BD-Rate [50], which compute the average distance in PSNR and bitrate between two RD-curves respectively. The proposed image set compression method only with dictionary reordering is denoted as *RLS-DLA-D3*. We can see that the proposed dictionary reordering achieves bit-rate reduction compared with that using unordered dictionary at the same quality level, about 1% bitrate saving on average compared with the method with unordered dictionary *RLS-DLA-D2*. The RD-OMP achieves much obvious bit-rate reduction further on the basis of reordered dictionary, more than 7% bitrate saving compared with *RLS-DLA-D2*. Furthermore, the proposed RD-OMP is not only suitable for image set compression problem, but also can be applicable to common dictionary learning based image compression frameworks.

In Table III, we further compare the state-of-the-art video coding standard, HEVC, with the proposed method. Two coding configurations, all intra coding (AI) and low delay P coding (LDP), are utilized, and they achieves 27.21% and 31.22% bitrate saving on average compared with the proposed method. Since the significant changes between images, e.g., scale changes and illuminance changes, the traditional motion compensation using local translation model is not as efficient as that for natural videos, which makes the HEVC inter coding improvement against intra coding is not so significant as that for natural videos. HEVC is a well-designed image/video compression system with many elaborately tuned coding tools, e.g., rate-distortion based coding unit decision, 35 intra prediction modes, multiple inter-prediction unit partitions, sup-pixel motion compensation, different transforms, context-adaptive binary arithmetic coding, in-loop filters and so on. These modules jointly achieve significant compression performance improvements. However, the proposed method is based on the sparse coding framework, which is different from HEVC and is still in the exploratory phase. Although the proposed method is inferior to the HEVC coding, it is a completely different framework compared with HEVC, and it may be a potential solution to improve the image/video coding performance by improving the different modules of sparse coding.

V. CONCLUSION

In this paper, we have proposed a new image set compression framework based on over-complete dictionary, which is learned from one representative image of the target image set. Since the dictionary is learned from the image that shares the most common similar content with other images, it can represent images in set more efficiently with fewer coefficients. Another two contributions, dictionary reordering and RD-OMP sparse coding algorithm, further improve the compression performance. The dictionary reordering is a special design for the image set compression problem, which adjusts the order of dictionary atoms according to their use frequency in representing representative images. It makes non-zero coefficients concentrate at the front of coefficient vector with higher probabilities. The proposed RD-OMP algorithm can efficiently optimize the atom distribution among image patches according to the rate-distortion costs. It can be also apply to the common over-complete dictionary based image compression problem. Experimental results show that our proposed method can efficiently reduce storage space and performs significantly better compared with popular image compression methods.

REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[2] X. Zhao, X. Zhang, L. Zhang, S. Ma, and W. Gao, "Low-complexity and sampling-aided multi-view video coding at low bitrate," in *Proc. Pacific-Rim Conf. Multimedia*, 2010, pp. 319–327.

[3] X. Zhang, R. Xiong, S. Ma, and W. Gao, "Adaptive loop filter with temporal prediction," in *Proc. Picture Coding Symp. (PCS)*, 2012, pp. 437–440.

[4] X. Zhang *et al.*, "Low-rank based nonlocal adaptive loop filter for high efficiency video compression," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[5] S. Ma, X. Zhang, J. Zhang, C. Jia, S. Wang, and W. Gao, "Nonlocal in-loop filter: The way toward next-generation video coding?" *IEEE Multimedia*, vol. 23, no. 2, pp. 16–26, Jun. 2016.

[6] X. Zhang, R. Xiong, S. Ma, and W. Gao, "New image coding scheme with hierarchical representation and adaptive interpolation," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Nov. 2011, pp. 1–4.

[7] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 68–84, Feb. 1992.

[8] X. Zhang, S. Wang, K. Gu, W. Lin, S. Ma, and W. Gao, "Just-noticeable difference-based perceptual optimization for JPEG compression," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 96–100, Jan. 2017.

[9] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of JPEG-2000," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2000, pp. 523–541.

[10] W. Zeng, Y. Zhao, K. Ou, and W. Song, "Research on cloud storage architecture and key technologies," in *Proc. 2nd Int. Conf. Interaction Sci., Inf. Technol., Culture Human*, 2009, pp. 1044–1048.

[11] Y. Wu, Z. Zhang, C. Wu, Z. Li, and F. C. M. Lau, "CloudMoV: Cloud-based mobile social TV," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 821–832, Jun. 2013.

[12] Y. Zhou, T. Z. J. Fu, D. M. Chiu, and Y. Huang, "An adaptive cloud downloading service," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 802–810, Jun. 2013.

[13] K. Karadimitriou, "Set redundancy, the enhanced compression model, and methods for compressing sets of similar images," Ph.D. dissertation, Dept. Comput. Sci., Louisiana State Univ., Baton Rouge, LA, USA, Aug. 1996.

[14] S. Wang, K. Gu, S. Ma, W. Lin, X. Liu, and W. Gao, "Guided image contrast enhancement based on retrieved images in cloud," *IEEE Trans. Multimedia*, vol. 18, no. 2, pp. 219–232, Feb. 2016.

[15] J. Liu, W. Yang, X. Zhang, and Z. Guo, "Retrieval compensated group structured sparsity for image super-resolution," *IEEE Trans. Multimedia*, vol. 19, no. 2, pp. 302–316, Feb. 2017.

[16] H. Yue, X. Sun, J. Yang, and F. Wu, "Cloud-based image coding for mobile devices—Toward thousands to one compression," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 845–857, Jun. 2013.

[17] K. Karadimitriou and J. M. Tyler, "Min-max compression methods for medical image databases," *SIGMOD Rec.*, vol. 26, no. 1, pp. 47–52, Mar. 1997.

[18] K. Karadimitriou and J. M. Tyler, "The Centroid method for compressing sets of similar images," *Pattern Recognit. Lett.*, vol. 19, no. 7, pp. 585–593, May 1998.

[19] C.-H. Yeung *et al.*, "Compressing similar image sets using low frequency template," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2011, pp. 1–6.

[20] J. Jiang, B. Guo, and S. Y. Yang, "Revisiting the JPEG-LS prediction scheme," *IEE Proc.-Vis., Image Signal Process.*, vol. 147, no. 6, pp. 575–580, Dec. 2000.

[21] S. Li, O. C. Au, R. Zou, L. Sun, and W. Dai, "Similar images compression based on DCT pyramid multi-level low frequency template," in *Proc. IEEE 14th Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2012, pp. 255–259.

[22] C. Nielsen and X. Li, "MST for lossy compression of image sets," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2006, p. 1.

[23] C.-P. Chen, C.-S. Chen, K.-L. Chung, H.-I. Lu, and G. Y. Tang, "Image set compression through minimal-cost prediction structures," in *Proc. ICIP*, 2004, pp. 1289–1292.

[24] R. Zou, O. C. Au, G. Zhou, S. Li, and L. Sun, "Image set modeling by exploiting temporal-spatial correlations and photo album compression," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Dec. 2012, pp. 1–4.

[25] Z. Shi, X. Sun, and F. Wu, "Multi-model prediction for image set compression," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2013, pp. 1–6.

[26] Z. Shi, X. Sun, and F. Wu, "Feature-based image set compression," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2013, pp. 1–6.

[27] Z. Shi, X. Sun, and F. Wu, "Photo album compression for cloud storage using local features," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 4, no. 1, pp. 17–28, Mar. 2014.

[28] Z. Tu, R. Poppe, and R. C. Veltkamp, "Weighted local intensity fusion method for variational optical flow estimation," *Pattern Recognit.*, vol. 50, pp. 223–232, Feb. 2016.

[29] R. Neff and A. Zakhor, "Dictionary approximation for matching pursuit video coding," in *Proc. Int. Conf. Image Process.*, vol. 2. Sep. 2000, pp. 828–831.

[30] G. Monaci, P. Jost, and P. Vandergheynst, "Image compression with learnt tree-structured dictionaries," in *Proc. IEEE 6th Workshop Multimedia Signal Process.*, Oct. 2004, pp. 35–38.

[31] M. Xu, S. Li, J. Lu, and W. Zhu, "Compressibility constrained sparse representation with learnt dictionary for low bit-rate image compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1743–1757, Oct. 2014.

[32] L. Xie, X. Zhang, S. Wang, S. Wang, X. Zhang, and S. Ma, "Perceptually optimized sparse coding for HDR images via divisive normalization," in *Proc. Vis. Commun. Image Process. (VCIP)*, 2016, pp. 1–4.

[33] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," *J. Vis. Commun. Image Represent.*, vol. 19, no. 4, pp. 270–282, May 2008.

[34] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[35] K. Skretting and K. Engan, "Image compression using learned dictionaries by RLS-DLA and compared with K-SVD," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 1517–1520.

[36] X. Zhang, W. Lin, S. Ma, S. Wang, and W. Gao, "Rate-distortion based sparse coding for image set compression," in *Proc. Vis. Commun. Image Process. (VCIP)*, Dec. 2015, pp. 1–4.

[37] Y. Sun, X. Tao, Y. Li, and J. Lu, "Dictionary learning for image coding based on multisample sparse representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 11, pp. 2004–2010, Nov. 2014.

[38] J. Mairal, F. Bach, and J. Ponce, "Sparse modeling for image and vision processing," *Found. Trends Comput. Graph. Vis.*, vol. 8, nos. 2–3, pp. 283–285, 2014.

[39] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput.*, vol. 1. Pacific Grove, CA, USA, Nov. 1993, pp. 40–44.

[40] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2121–2130, Apr. 2010.

[41] X. Zhang, Y. Zhang, W. Lin, S. Ma, and W. Gao, "An inter-image redundancy measure for image set compression," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1274–1277.

[42] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[43] L.-Y. Duan *et al.*, "Overview of the MPEG-CDVS standard," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 179–194, Jan. 2016.

[44] W. Sun, X. Zhang, S. Wang, J. Chen, and L. Duan, "GPU based fast MPEG-CDVS encoder," in *Proc. Int. Conf. Image Process.*, Sep. 2017, pp. 1122–1126.

[45] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.

[46] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.

[47] J. Si, S. Ma, X. Zhang, and W. Gao, "Adaptive rate control for High Efficiency Video Coding," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2012, pp. 1–6.

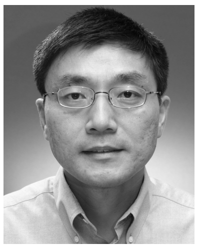
[48] K. Skretting, J. H. Husøy, and S. O. Aase, "Improved Huffman coding using recursive splitting," in *Proc. Norwegian Signal Process. (NORSIG)*, 1999, pp. 1–4.

[49] B. Shi, L. Liu, and C. Xu, "Comparison between JPEG2000 and H.264 for digital cinema," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jun. 2008, pp. 725–728.

[50] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33, ITU-Telecommunications Standardization Sector, STUDY GROUP 16 Question 6, Video Coding Experts Group (VCEG), Mar. 2001.

Xinfeng Zhang (M'16) received the B.S. degree in computer science from the Hebei University of Technology, Tianjin, China, in 2007, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2014.

He is currently a Research Fellow with Nanyang Technological University, Singapore. His research interests include image and video processing, image and video compression.



Weisi Lin (M'92–SM'98–F'16) received the B.Sc. degree in electronics and the M.Sc. degree in digital signal processing from Zhongshan University, Guangzhou, China, in 1982 and 1985, respectively, and the Ph.D. degree in computer vision from the King's College London, London University, London, U.K., in 1992.

He was involved in teaching and research with Zhongshan University, Shantou University, Shantou, China, Bath University, Bath, U.K., the National University of Singapore, the Institute of Microelectronics, Singapore, and the Institute for Infocomm Research, Singapore. He was the Laboratory Head of Visual Processing and the Acting Department Manager of Media Processing, Institute for Infocomm Research. He is currently a Full Professor with the School of Computer Engineering, Nanyang Technological University, Singapore. His current research interests include image processing, perceptual modeling, video compression, multimedia communication, and computer vision.

Yabin Zhang received the B.E. degree in electronic information engineering in the Honors School, Harbin Institute of Technology, in 2013. He is currently pursuing the Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests include video coding, image/video processing, and computer vision.

Shiqi Wang (M'15) received the B.S. degree in computer science from the Harbin Institute of Technology in 2008 and the Ph.D. degree in computer application technology from Peking University in 2014. From 2014 to 2016, he was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada. From 2016 to 2017, he was with the Rapid-Rich Object Search Lab, Nanyang Technological University, Singapore, as a Research Fellow. He is currently an Assistant Professor with the Department of Computer Science, City University of Hong Kong. He has proposed over 30 technical proposals to ISO/MPEG, ITU-T and AVS standards. His research interests include video compression, image/video quality assessment, and image/video search and analysis.

Siwei Ma (M'12) received the B.S. degree from Shandong Normal University, Jinan, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005.

From 2005 to 2007, he held a post-doctoral position with the University of Southern California. Then he joined the Institute of Digital Media, EECS, Peking University, where he is currently a Full Professor. He has authored over 100 technical articles in refereed journals and proceedings in the areas of image and video coding, video processing, video streaming, and transmission.

Lingyu Duan (M'06) received the M.Sc. degree in automation from the University of Science and Technology of China, China, in 1999, the M.Sc. degree in computer science from the National University of Singapore, Singapore, in 2002, and the Ph.D. degree in information technology from The University of Newcastle, Australia, in 2008. He was a Research Scientist with the Institute for Infocomm Research (I2R), Singapore, from 2003 to 2008. He is currently a Full Professor with the National Engineering Laboratory of Video Technology, School of Electronics Engineering and Computer Science, Peking University (PKU), China, and has been serving as the Associate Director of the Rapid-Rich Object Search Lab, a joint lab between Nanyang Technological University, Singapore, and PKU, China, since 2012. His research interests include multimedia indexing, search, and retrieval, mobile visual search, visual feature coding, and video analytics. He has authored over 130 research papers in international journals and conferences.

Wen Gao (M'92–SM'05–F'09) received the Ph.D. degree in electronics engineering from The University of Tokyo, Tokyo, Japan, in 1991.

He was a Professor of computer science with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He is currently a Professor of computer science with the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing, China. He has authored extensively including five books and over 600 technical articles in refereed journals and conference proceedings in the areas of image processing, video coding and communication, pattern recognition, multimedia information retrieval, multimodal interfaces, and bioinformatics. He is a member of the China Engineering Academy. He served or serves on the Editorial Board of several journals, such as the IEEE Transactions on Circuits and Systems for Video Technology, the IEEE Transactions on Multimedia, the IEEE Transactions on Autonomous Mental Development, the *EURASIP Journal of Image Communications*, and the *Journal of Visual Communication, and Image Representation*. He has been the Chair of a number of prestigious international conferences on multimedia and video signal processing, such as the IEEE International Conference on Multimedia and Expo and the ACM Multimedia, and served on the advisory and technical committees of numerous professional organizations.