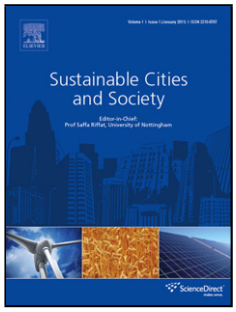

Title: A sparse representation-based image resolution improvement method by processing multiple dictionary pairs with latent Dirichlet allocation model for street view images

Authors: Hu Li, Xiaomin Yang, Lihua Jian, Kai Liu, Yuan Yuan, Wei Wu

PII: S2210-6707(17)31099-5
DOI: https://doi.org/10.1016/j.scs.2017.12.020
Reference: SCS 892

To appear in:

Received date: 19-8-2017
Revised date: 22-11-2017
Accepted date: 13-12-2017

Please cite this article as: Li, Hu., Yang, Xiaomin., Jian, Lihua., Liu, Kai., Yuan, Yuan., & Wu, Wei., A sparse representation-based image resolution improvement method by processing multiple dictionary pairs with latent Dirichlet allocation model for street view images.*Sustainable Cities and Society* https://doi.org/10.1016/j.scs.2017.12.020

A sparse representation-based image resolution improvement method by processing multiple dictionary pairs with latent Dirichlet allocation model for street view images

Hu Li[a], Xiaomin Yang[a], Lihua Jian[a], Kai Liu[b], Yuan Yuan[a], Wei Wu[a,*]

[a] College of Electronics and Information Engineering, Sichuan University, Chengdu, Sichuan, 610064, China
[b] College of Electrical and Engineering Information, Sichuan University, Chengdu, Sichuan, 610064, China

* Corresponding author. Wei Wu, wuwei@scu.edu.cn (Wei Wu)

Research highlight

- The proposed method clusters patches at the semantic level by utilizing latent Dirichlet allocation model.
- The proposed method represents word patches and document patches by sparse representation coefficients.

Abstract

Street view applications are widely used in many situations. However, the resolution of the street view image is not high enough. Users always desire high resolution street view images. Image resolution improvement methods can effectively generate a high resolution street view image from a single low resolution street view image. The sparse representation-based image resolution improvement method is a promising way to improve the resolution of an image. However, only one dictionary pair, which fails to represent the diverse structures in images, is used in conventional sparse representation-based methods. This may lead to poor performances in many circumstances. In this paper, we propose a new sparse representation-based method with multiple dictionary pairs. To capture the various structures at the semantic level, our method adopts latent Dirichlet allocation model to divide the patches into clusters. Then we learn a dictionary pair for each cluster. Finally, these dictionary pairs are used to reconstruct high resolution images. Experimental results validate that our method is superior over the compared methods in both visual perception and objective quantitation.

Keywords: image resolution improvement method, latent Dirichlet allocation, semantic information, sparse representation.

1 Introduction

Street view applications, which can make users enjoy scenes from a driver's point of view, are very popular. Street view applications develop fast in recent years [1,2], and play an important role in smart cities due to their massive advantages. Home buyers can use a street view application to see the surrounding area of house being sold. People can even exploit a lava cave just in their own home by using a street view application. Up to now, Baidu street view has covered 372 cities in China, and Google street view is available in more than 70 countries worldwide.

Street view applications use street view images to reconstruct particular scenes. The higher resolution the street view images have, the higher resolution the reconstructed scenes are. Users always prefer high resolution scenes when using a street view application. However, contemporary street view applications suffer from the lacking of high resolution street view images.

Generally，the resolution of an image can be improved by two ways: changing image sensors and using an image resolution improvement method. Changing image sensors is a straightforward way to improve image resolution. However, it is very expensive. Thus, image resolution improvement methods become very active techniques, since they can improve image resolution at low cost. In addition, image resolution improvement methods are also very attractive to many other industries, which need high resolution (HR) images for interpretation, such as medical industry [3,4] and aerospace [5,6].

As shown in Fig. 1, image resolution improvement methods can be broadly classified into three classes [7]: interpolation-based methods [8,9], reconstruction-based methods [10,11,12], and learning-based methods [7,13,14,15,16,17,18,19,20].

Interpolation-based methods, which include nearest-neighbor interpolation, bilinear interpolation, cubic splines interpolation [8], and cubic convolution interpolation [9], reconstruct HR images with its own information. Although these methods are simple and easy to be implemented, they are very likely to cause chessboard effects or other artifacts.

Reconstruction-based methods use multiple low resolution (LR) images with different sub-pixel shifts from the same scene to reconstruct the HR image. Irani et al. [10] present a method which is similar to back-projection in tomography to reconstruct a HR image. Hardie et al. [11] reconstruct a HR image by estimating registration parameters based on maximum a posteriori (MAP). Elad et al. [12] reconstruct a HR image from LR images by using projection onto convex sets (POCS), maximum likelihood (ML), and MAP. Generally, reconstruction-based methods obtain better results than interpolation-based methods. However, reconstruction-based methods are constrained by three factors. First, it is hard to obtain sufficient different images with different sub-pixel shifts of the same scene. Second, reconstruction-based methods apply image registration to merge the LR images. However, image registration is very difficult in the practical application. Third, the reconstructed HR image deteriorates rapidly under a big upscaling factor.

Learning-based methods use a priori information to reconstruct a HR image. According to the ways of exploiting a priori information, we can categorize learning-based methods into three classes: neighbor embedding-based methods [13,14],

regression-based methods [15,16], sparse representation-based methods [7,17], and deep-network-based methods [18,19,20].

Neighbor embedding-based methods combine neighbor training HR patches linearly to reconstruct HR patches. In [13], Chang et al. propose a reconstruction strategy inspired by locally linear embedding (LLE). In [14], Zhang et al. present a partially supervised neighbor embedding-based algorithm. Regression-based methods reconstruct HR patches by exploiting relationships of LR-HR patch pairs. Wang et al. [15] use eigentransformation for face hallucination. Wu et al. [16] solve the one-to-many mapping problem by using the kernel partial least squares regression model. However, both neighbor embedding-based methods and regression-based methods are constrained by the complicated statistical models.

Sparse representation-based methods are widely studied due to the development of sparse coding techniques in recent years. By learning a priori information between HR images and their corresponding LR images, sparse representation-based techniques construct a LR-HR dictionary pair for reconstructing HR images. The general process can be illustrated as Fig. 2 [21]. Yang et al. [7] learn a LR-HR dictionary pair by iteratively solving two convex optimization problems. Zeyde et al. [17] use K-SVD [22] to learn the LR-HR dictionary pair.

Deep-network-based methods learn a nonlinear mapping between LR images and their corresponding HR images to reconstruct HR images. Super-resolution convolutional neural network (SRCNN) [18] reconstructs HR images by utilizing deep conventional network. Kim et al. [19] propose an image resolution improvement method by utilizing deeply-recursive convolutional network (DRCN). Dong et al. [20] propose a compact hourglass-shape CNN structure to reconstruct HR images faster and better (FSRCNN).

The Sparse representation-based methods mentioned above use a single LR-HR dictionary pair to reconstruct HR images. They fail in extracting the various structures in images, because only one LR-HR dictionary pair cannot represent all image patterns. Thus, the reconstructed HR images may lack sufficient high-frequency details. To overcome this shortcoming, image resolution improvement methods with multiple dictionary pairs are proposed.

In essence, generating a LR patch from a HR patch is a many-to-one mapping [23]. Thus, for two same LR patches, they may come from two different HR patches. For instance, the zoomed areas in Fig. 3 are very similar. However, they are actually derived from an automotive light picture and a human eye picture respectively. It is not reasonable to reconstruct eye picture with a priori information of automotive light picture, vice versa. Conventional image resolution improvement methods learn a priori information by utilizing low-level features. These low-level features can be used in many areas. For example, Zhou et al. [24] proposed a novel method to extract information by utilizing low-level features like image gradient magnitudes and orientations for implementation of copy detection. Wang et al. [25] propose an interesting multi-watermarking method by utilizing the discrete wavelet transform (DWT) coefficients. Xiong et al. [26] propose a novel reversible data hiding (RDH) method by taking advantage of the Laplacian-like distribution of integer wavelet high-frequency coefficients. However, utilizing low-level features cannot identify the kind of difference shown in Fig. 3. Thus, much noise may be induced in the reconstructed HR images.

To solve this problem, we present a new image resolution improvement method by utilizing latent Dirichlet allocation (LDA) model to learn a priori information semantically. The advantage of using semantic information is that we can predict the higher level semantic in which the patches possibly belong to [23]. In our work, a semantic relationship of different patches is obtained by using semantic information. The relationship helps us to cluster patches in a training set, and helps us to classify patches for input test patches. The reason for using LDA model instead of other models is that LDA model performs best among many models in the domain of text modeling [27], such as unigram model, mixture of unigrams, latent semantic indexing (LSI) and probabilistic latent semantic indexing (pLSI).

In fact, the semantic information is widely used in many applications. In [28], semantic information is well adopted to construct conceptual graphs as knowledge representation tool to make search in encrypted data more precisely. In [29], Li et al. propose a novel method to find the copy-move forgery within an image by using semantical information among different patches. In [30], semantic information between concepts in concept hierarchy are well used for effective and context-aware search method in encrypted cloud data. In [31], an effective content-based search strategy by processing semantic information in users' retrieval is proposed to precisely match users' search intention.

Based on above consideration, we proposed an image resolution improvement method by utilizing semantic information. Specifically, our method divides patches into clusters by utilizing LDA model. Then we learn a LR-HR dictionary pair for each cluster. These dictionary pairs are used to reconstruct HR images. Experimental results validate that our method is superior over the compared methods in both visual perception and objective quantitation.

The rest of this paper is organized as follows. In section 2, we illustrate related works. In section 3, we elaborate the proposed method. We show our experimental results in section 4. Finally, we make a conclusion of this paper in section 5.

2 Related Works

2.1 Sparse representation-based methods

Given a signal $\boldsymbol{X} \in R^N$, its sparse representation coefficients $\boldsymbol{\alpha} \in R^K$ in terms of an over-complete dictionary $D \in R^{N \times K}$ [32] are shown in Fig. 4. N is the size of $\boldsymbol{X}$, and K is the size of $\boldsymbol{\alpha}$. The column of an over-complete dictionary is called an atom. Accordingly, we can sparsely represent a patch in terms of an over-complete dictionary formed by patches.

Image resolution improvement methods reconstruct a HR image $\boldsymbol{X} \in R^N$ from a given LR image $\boldsymbol{Y} \in R^M$. The mathematic relationship between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is:

$$\boldsymbol{Y} = HB\boldsymbol{X}, \tag{1}$$

where $H \in R^{M \times N}$ is a down-sampling operation, $B \in R^{N \times N}$ denotes a blurring filter, and M is the size of $\boldsymbol{Y}$.

Sparse representation-based methods assume that sparse representation coefficients of a LR patch is the same as that of its corresponding HR patch. A HR patch can be reconstructed by exploiting this assumption. Sparse representation-based

methods contain four main steps [33]: 1. feature extraction; 2. patch partition; 3. dictionary learning; 4. HR image reconstruction. These four steps are illustrated as follows.

2.1.1 Feature extraction

We extract features from a LR image as:

$$\boldsymbol{y}_f = F_L(\boldsymbol{y}), \tag{2}$$

where $\boldsymbol{y} \in R^M$ is a LR image in the training set, $F_L(\cdot)$ is an operation which extracts features from a LR image, and $\boldsymbol{y}_f \in R^M$ is the LR feature image.

2.1.2 Patch partition

We partition each LR feature image and its corresponding HR image into patches as:

$$\boldsymbol{y}_f^i = Patch^i(\boldsymbol{y}_f), \tag{3}$$

$$\boldsymbol{x}^i = Patch^i(\boldsymbol{x}), \tag{4}$$

where $Patch^i(\cdot)$ is an operation to extract a patch from an image, i refers to the position where the patch is extracted from, and $\boldsymbol{x}$ is a HR image. $\boldsymbol{y}_f^i \in R^m$ and $\boldsymbol{x}^i \in R^n$ are the extracted patches. m is the size of LR patch $\boldsymbol{y}_f^i$ and n is the size of HR patch $\boldsymbol{x}^i$.

2.1.3 Dictionary learning

Given LR-HR patch pairs $\{\boldsymbol{y}_f^i, \boldsymbol{x}^i\}(i \in [1, P])$, where P is the number of LR-HR patch pairs. By learning a priori information in $\{\boldsymbol{y}_f^i, \boldsymbol{x}^i\}$, we can construct the LR-HR dictionary pair (D_l, D_h). A LR dictionary D_l and sparse representation coefficients $\boldsymbol{\alpha}_l^i$ of $\boldsymbol{y}_f^i$ can be obtained as:

$$D_l, \{\boldsymbol{\alpha}_l^i\} = \arg\min_{D_l, \{\boldsymbol{\alpha}_l^i\}} \sum_{i=1}^{P} \left\| D_l \boldsymbol{\alpha}_l^i - \boldsymbol{y}_f^i \right\|_2 + \lambda \left\| \boldsymbol{\alpha}_l^i \right\|_0, \tag{5}$$

where $D_l \in R^{n \times k}$ is the LR dictionary, k is the number of dictionary atoms, and λ decides the degree of sparsity.

Since we assume sparse representation coefficients of a HR patch $\boldsymbol{\alpha}_h^i$ are the same as that of its corresponding LR patch: $\boldsymbol{\alpha}_h^i = \boldsymbol{\alpha}_l^i$, the HR dictionary D_h is obtained as:

$$\begin{aligned} D_h &= \arg\min_{D_h} \sum_{i=1}^{P} \left\| \boldsymbol{x}^i - D_h \boldsymbol{\alpha}_h^i \right\|_2 \\ &= \arg\min_{D_h} \sum_{i=1}^{P} \left\| \boldsymbol{x}^i - D_h \boldsymbol{\alpha}_l^i \right\|_2, \end{aligned} \tag{6}$$

where $D_h \in R^{m \times k}$.

2.1.4 HR image reconstruction

Given a LR image $\boldsymbol{Y}$, its feature patches $\boldsymbol{Y}_f^j$ ($j \in [1, J]$) can be extracted via Eq. (2) and Eq. (3), where J is the number of LR patches. Then we obtain the sparse representation coefficients $\hat{\boldsymbol{\alpha}}_l^j$ of the LR patch by:

$$\hat{\boldsymbol{\alpha}}_l^j = \arg\min_{\hat{\boldsymbol{\alpha}}_l^j} \sum_{i=1}^{P} \left\| D_l \hat{\boldsymbol{\alpha}}_l^j - \boldsymbol{Y}_f^j \right\|_2 + \lambda \left\| \hat{\boldsymbol{\alpha}}_l^j \right\|_0 . \tag{7}$$

Since we assume a HR patch has the same coefficients as its corresponding LR patch, we can reconstruct a HR patch as:

$$\hat{\boldsymbol{X}}^j = D_h \hat{\boldsymbol{\alpha}}_l^j . \tag{8}$$

The final HR image $\hat{\boldsymbol{X}}$ can be reconstructed by merging the HR patches $\{\hat{\boldsymbol{X}}^j\}$.

2.2 Latent Dirichlet allocation model

We use a $M \times V$ co-occurrence table to denote a corpus. This table stores the frequency of occurrences $n(\boldsymbol{w}_i, \boldsymbol{d}_j)$ for word $\boldsymbol{w}_i$ in document $\boldsymbol{d}_j$. M is the number of documents, and V is the size of the vocabulary. Assume that the corpus has K hidden latent topics $\{\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_K\}$, and each document in this corpus has a certain topic probability distribution. Fig. 5 shows the LDA model [27], where $\boldsymbol{\alpha} \in R^K$ is the Dirichlet prior. $\beta \in R^{K \times V}$ is the probability distribution of words and topics. $\boldsymbol{\theta} \sim Dir(\boldsymbol{\alpha})(\boldsymbol{\theta} \in R^K)$ is a Dirichlet random variable.

Given the parameter $\boldsymbol{\alpha}$, the probability density can be expressed as:

$$p(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = \frac{\Gamma(\sum_{i=1}^{K} \boldsymbol{\alpha}_i)}{\prod_{i=1}^{K} \Gamma(\boldsymbol{\alpha}_i)} \boldsymbol{\theta}_1^{\boldsymbol{\alpha}_1 - 1} \cdots \boldsymbol{\theta}_K^{\boldsymbol{\alpha}_K - 1}, \tag{9}$$

where $\Gamma(\cdot)$ is the Gamma function, and $\boldsymbol{\theta}$ is the topic mixture. The joint distribution of $\boldsymbol{\theta}$, topics $\boldsymbol{z}$, and words $\boldsymbol{w}$ for the given parameters $\boldsymbol{\alpha}$ and β is [27]:

$$p(\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w} \mid \boldsymbol{\alpha}, \beta) = p(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \prod_{n=1}^{N} p(\boldsymbol{z}_n \mid \boldsymbol{\theta})\, p(\boldsymbol{w}_n \mid \boldsymbol{z}_n, \beta), \tag{10}$$

where N is the number of topics. For a given document, we can obtain its marginal distribution as:

$$p(\boldsymbol{w} \mid \boldsymbol{\alpha}, \beta) = \int p(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \left(\prod_{n=1}^{N} \sum_{\boldsymbol{z}_n} p(\boldsymbol{z}_n \mid \boldsymbol{\theta})\, p(\boldsymbol{w}_n \mid \boldsymbol{z}_n, \beta) \right) d\boldsymbol{\theta} . \tag{11}$$

To estimate the topic distribution $\boldsymbol{z}$ for a given document, we need to compute the posterior distribution as:

$$p(\boldsymbol{\theta}, \boldsymbol{z} \mid \boldsymbol{w}, \boldsymbol{\alpha}, \beta) = \frac{p(\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w} \mid \boldsymbol{\alpha}, \beta)}{p(\boldsymbol{w} \mid \boldsymbol{\alpha}, \beta)}, \tag{12}$$

where $p(\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w} \mid \boldsymbol{\alpha}, \beta)$ is obtained by Eq. (10) and $p(\boldsymbol{w} \mid \boldsymbol{\alpha}, \beta)$ is obtained by Eq. (11).

To apply LDA model in the image processing field, we need to organize an image in a text form. The general terms, such as topics, documents and words that are mostly

used in text literature. In the context of image resolution improvement methods, 'Document' is defined as cropped portion of an image. 'Word' is defined as cropped portion of a 'Document'. The difference between a text and an image is that a text has different composing elements from an image. 'Word' is the composing element of both a text and an image. However, 'Word' in a text usually has a specific codebook, but "Word" in an image does not have a specific codebook. Thus, we use the sparse representation coefficients of a patch to denote a "Word" in an image. The different methods of representing 'Word' cause the difference between a text and an image.

3 Proposed method

The proposed method contains two stages: the learning stage and the reconstructing stage. In the learning stage, patches are clustered at the semantic level. Then dictionary pairs are learned for each cluster of patches. In the reconstructing stage, we assign a topic to each LR patch based on semantic information. Then a HR patch is reconstructed with its corresponding topic dictionary pair. Finally, the HR image is obtained by merging the HR patches.

3.1 The learning stage

The learning stage can be summarized as Fig. 6. We divide the learning stage into five steps: 1. extracting feature images; 2. partitioning images; 3. learning trigger dictionary; 4. assigning topics; 5. learning topic dictionary pairs. Each step is elaborated as follows.

3.1.1 Extracting feature images

LR-HR image pairs are denoted as $\{\boldsymbol{Y}^q, \boldsymbol{X}^q\}(q \in [1,Q])$, where Q is the number of image pairs. Their corresponding feature images are obtained via Eq. (2). In our experiments, we use derivative filters ($\boldsymbol{f}_1 = [1,0,-1]$, $\boldsymbol{f}_2 = [1,0,-1]'$, $\boldsymbol{f}_3 = [1,0,-2,0,1]$,and $\boldsymbol{f}_4 = [1,0,-2,0,1]'$) to extract features of a LR image. Thus the Eq. (2) can be rewritten as:

$$\begin{aligned} \boldsymbol{Y}_f^q &= F_L(\boldsymbol{Y}^q) \\ &= \boldsymbol{f} * \boldsymbol{Y}^q, \end{aligned} \tag{13}$$

where $*$ denotes a convolution operation. We denote LR feature images and their corresponding HR image as $\{\boldsymbol{Y}_{f_1}^q, \boldsymbol{Y}_{f_2}^q, \boldsymbol{Y}_{f_3}^q, \boldsymbol{Y}_{f_4}^q, \boldsymbol{X}^q\}$.

3.1.2 Partitioning images

Images in $\{\boldsymbol{Y}_{f_1}^q, \boldsymbol{Y}_{f_2}^q, \boldsymbol{Y}_{f_3}^q, \boldsymbol{Y}_{f_4}^q, \boldsymbol{X}^q\}$ are partitioned into two different size patches: word patches and document patches. Word patches are denoted as $\{\boldsymbol{y}_{f_1,w}^s, \boldsymbol{y}_{f_2,w}^s, \boldsymbol{y}_{f_3,w}^s, \boldsymbol{y}_{f_4,w}^s, \boldsymbol{x}_w^s\}$ $(s \in [1,S])$. The subscript w denotes a word patch. The capital S is the num-

ber of the word patches. A document patch contains word patches. The document patches are denoted as $\{\boldsymbol{y}^l_{f_1,d}, \boldsymbol{y}^l_{f_2,d}, \boldsymbol{y}^l_{f_3,d}, \boldsymbol{y}^l_{f_4,d}, \boldsymbol{x}^l_d\}(l \in [1, L])$. The subscript *d* denotes a document patch. *L* is the number of document patches. Both word patches and document patches have overlaps in our experiments. For notational convenience, we use $\{\boldsymbol{y}^s_{f,w}\}$ to denote $\{\boldsymbol{y}^s_{f_1,w}, \boldsymbol{y}^s_{f_2,w}, \boldsymbol{y}^s_{f_3,w}, \boldsymbol{y}^s_{f_4,w}\}$ and, $\{\boldsymbol{y}^l_{f,d}\}$ to denote $\{\boldsymbol{y}^l_{f_1,d}, \boldsymbol{y}^l_{f_2,d}, \boldsymbol{y}^l_{f_3,d}, \boldsymbol{y}^l_{f_4,d}\}$.

3.1.3 Learning trigger dictionary

Before we cluster word patches by using LDA model, we need to represent a word patch by a feature vector. A sparse dictionary is learnt by using all word patches. This dictionary is used to obtain the sparse representation of each patch. The coefficients of the sparse representation are used as the feature vector of a patch. The feature vector is used for discovering a topic for each word patch in a training set, and is also used for assigning a topic to the input test word patch. Thus, we call the dictionary, which is used to generate the feature vector, as a 'Trigger dictionary'. The number of atoms in the trigger dictionary is equal to the vocabulary size in LDA model. Trigger dictionary D_l^T can be obtained as:

$$D_l^T, \{\boldsymbol{\alpha}_l^s\} = \arg\min_{D_l^T, \{\boldsymbol{\alpha}_l^T\}} \sum_{s=1}^{S} \left\| D_l^T \boldsymbol{\alpha}_l^s - \boldsymbol{y}^s_{f,w} \right\|_2 + \lambda \left\| \boldsymbol{\alpha}_l^s \right\|_0, \tag{14}$$

where $\{\boldsymbol{\alpha}_l^s\}$ are side products which denote the sparse representation coefficients. The optimization toolbox sparse modeling software (SPAMS) is used for dictionary learning by solving Eq. (14).

3.1.4 Assigning topics

To utilize LDA model, the sparse representation coefficients are used to denote a patch. For a word patch $\boldsymbol{y}^p_{f,w}$ in a document patch $\boldsymbol{y}^l_{f,d}$, we can obtain the coefficients $\boldsymbol{\alpha}^p_{l,w}$ in terms of D_l^T as:

$$\boldsymbol{\alpha}^p_{l,w} = \arg\min_{\boldsymbol{\alpha}^p_{l,w}} \sum_{i=1}^{P} \left\| D_l^T \boldsymbol{\alpha}^p_{l,w} - \boldsymbol{y}^p_{f,w} \right\|_2 + \lambda \left\| \boldsymbol{\alpha}^p_{l,w} \right\|_0, \tag{15}$$

where *P* is the number of word patches in the document patch. Then the document patch can be represented as:

$$\boldsymbol{d}_d = \sum_{p=1}^{P} \boldsymbol{\alpha}^p_{l,w}, \tag{16}$$

where $d \in [1, L]$ denotes the *d*-th document patch.

Given a corpus $\{\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_L\}$, we need to find parameters $\boldsymbol{\alpha}$ and β to maximize the log likelihood as:

$$l(\boldsymbol{\alpha}, \beta) = \sum_{d=1}^{M} \log p(\boldsymbol{d}_d \mid \boldsymbol{\alpha}, \beta), \tag{17}$$

where $\boldsymbol{\alpha}$ is the Dirichlet prior, and β is the probability distribution of word patches and topics. The $\boldsymbol{\alpha}$ and β are obtained by using the variational expectation maximization (EM) algorithm proposed in [27]. Then we can compute the topic distribution of each document patch in the corpus $\{\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_L\}$ via Eq. (12). We denote the topic distribu-

tion of the corpus as $\boldsymbol{Z} \in R^{L\times K}$, where $\boldsymbol{Z}_{l,k}(l \in [1,L], k \in [1,K])$ is the weight of topic k in the document patch $\boldsymbol{d}_l$. The topic with the maximum weight in a document patch is assigned to this document patch. Assume that we assign the topic *k* to a document patch, then the topic *k* is assigned to every word patch in this document patch. Then the word patches are divided into *K* (the number of topics we set in LDA model) clusters according to the topics assigned to these word patches. Thus, we can obtain *K* sets of LR-HR patch pairs $\{\boldsymbol{y}_{f,w,k}^{c_k}, \boldsymbol{x}_{f,w,k}^{c_k}\}(c_k \in [1,C_k], k \in [1,K])$, where C_k is the number of word patches under topic *k*.

3.1.5 Learning topic dictionary pairs

A dictionary pair is learned for each cluster of word patches. Since word patches are clustered by topics, we call the learned dictionary pairs 'Topic dictionary pairs'. The starting point of each topic dictionary pair learning is the LR word patches $\{\boldsymbol{y}_{f,w,k}^{c_k}\}$. The LR dictionary D_l^k for word patches under topic *k* is learned as:

$$D_l^k, \{\boldsymbol{\alpha}_l^{c_k}\} = \arg\min_{D_l^k, \{\boldsymbol{\alpha}_l^{c_k}\}} \sum_{c_k=1}^{C_k} \left\| D_l^k \boldsymbol{\alpha}_l^{c_k} - \boldsymbol{y}_{f,w,k}^{c_k} \right\|_2 + \lambda \left\| \boldsymbol{\alpha}_l^{c_k} \right\|_0 , \tag{18}$$

where $\{\boldsymbol{\alpha}_l^{c_k}\}$ are side products which denote the sparse representation coefficients. The next step is to construct the HR dictionary for topic *k*. The HR dictionary D_h^k is learned as:

$$D_h^k = \arg\min_{D_h^k} \sum_{c_k=1}^{C_k} \left\| \boldsymbol{x}^{c_k} - D_h^k \boldsymbol{\alpha}_l^{c_k} \right\|_2 . \tag{19}$$

The same as learning trigger dictionary, SPAMS is used to learn topic dictionary pairs.

3.2 The reconstructing stage

A HR image is reconstructed from a LR image in the reconstructing stage. The reconstructing stage can be summarized as Fig. 7. We divide the reconstructing stage into four steps: 1. extracting feature images; 2. partitioning LR feature images; 3. assigning topics; 4. reconstructing HR image. Each step is elaborated as follows.

3.2.1 Extracting feature images

Given a LR image $\boldsymbol{L}$, it is upscaled by bicubic interpolation with an upscaling factor u. Then we use the same feature extraction operation as the learning stage to obtain LR feature images $\{\boldsymbol{L}_{f_1}, \boldsymbol{L}_{f_2}, \boldsymbol{L}_{f_3}, \boldsymbol{L}_{f_4}\}$.

3.2.2 Partitioning LR feature images

The document patches $\{\boldsymbol{l}_{f,d}^t\}(t \in [1,T])$ are obtained from LR feature images the same as the step 2 in the learning stage. *T* is the number of document patches.

3.2.3 Assigning topics

We assign a topic to each word patch the same as the step 4 in the learning stage. Then the word patches are divided into *K* (the number of topics we set in the learning stage) clusters. Each cluster of word patches is denoted as $\{\boldsymbol{l}_{f,w,k}^{q_k}\}(q_k \in [1, Q_k])$, where Q_k is the number of word patches under the topic *k*.

3.2.4 Reconstructing HR image

Sparse representation coefficients $\{\boldsymbol{\alpha}_{l,k}^{q_k}\}$ of a word patch $\boldsymbol{l}_{f,w,k}^{q_k}$ are obtained in terms of a LR dictionary D_l^k via Eq. (7). Then its corresponding HR word patch $\boldsymbol{h}_{f,w,k}^{q_k}$ is obtained as:

$$\boldsymbol{h}_{f,w,k}^{q_k} = D_h^k \boldsymbol{\alpha}_{l,k}^{q_k} . \tag{20}$$

A HR document patch $\boldsymbol{h}_{f,d,k}^{q_k}$ is obtained by merging the HR word patches in this document patch. The values in the overlapped regions among HR word patches are averaged to get smooth edges. We get the reconstructed HR image $\boldsymbol{H}$ by merging the HR document patches. The values in the overlapped regions between HR document patches are averaged too.

4 Experimental results

We perform a series of experiments to show that our method is practical and efficient. Experimental results are elaborated in both visual perception and objective quantitation. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) are adopted in our experiments to evaluate the objective quantitation.

Our method uses the same optimization toolbox as [23] for dictionary learning. Our training images are the same as that of Yang et al. [7]. Our experiments contain two kinds of test images: widely used images in image processing area and street view images. The widely used images are selected from Set5 [34] and Set14 [17]. Fig. 8 shows these images. The street view images are obtained from Google street view. Figs. 17-18 show the street view images. The values of key parameters for our method are illustrated as Table 1.

4.1 The effect of the number of topics

The number of topics plays a significant role in our experiments. Thus, we perform a comparative experiment to validate that how the number of topics effect our method. We upscale image 'Baby' by 3 times with different numbers of topics. Experimental results are shown as Figs. 9-10.

As we can see from Figs. 9-10, both the curve of PSNR and the curve of SSIM rise first, then fall down. This can be explained by that the number of the inherent topics for a specific training image set is fixed at the semantic level. Thus, the closer the number of topics we set to this fixed number, the better our algorithm will perform. The best experimental result is obtained at around 10 topics.

4.2 Reconstruction results

To illustrate the effectiveness of our method, we compare our method with bilinear interpolation, bicubic interpolation, and Yang's method [7]. Figs. 11-16 show the visual comparisons. To show the details after upscaling, the regions in red boxes are magnified individually. The edges of the HR image reconstructed by using bilinear interpolation and bicubic interpolation are blurred, and have obvious jaggy effects. Lots of details in the reconstructed images are lost too. This is because they cannot use a priori information to reconstruct HR images. HR images reconstructed by Yang's method have sharper edges and more details than bilinear interpolation and bicubic interpolation. However, there exist many ringing artifacts near the edges or textures in the reconstructed images obtained by Yang's method. This is because Yang's method uses only one LR-HR dictionary pair, which cannot capture various structures in images, to reconstruct HR images. HR images reconstructed by our method, by contrast, have sharper edges but fewer artifacts than Yang's method. This is because our method can cluster patches at the semantic level. Then LR-HR dictionary pairs can be constructed to represent the various structures. By using these dictionary pairs, we can reconstruct HR patches more precisely and reconstruct clearer HR images.

Besides visual comparison, we also perform comparative experiments with objective quantitation. The experimental results are illustrated in Table 2. Average PSNR obtained by our method improve 0.338 (dB) over Yang's method, improve 0.970(dB) over bicubic interpolation, and improve 1.587(dB) over bilinear interpolation. Average SSIM obtained by our method improve 0.001 over Yang's method, improve 0.019 over bicubic interpolation, and improve 0.036 over bilinear interpolation. These datums demonstrate that our method performs better than other three methods. In brief, in both visual perception and objective quantitation, our method performs best among the compared methods.

4.3 Experiments with street view images

To validate the effectiveness of our method in a practical street view application, street view images are used as the test images in our experiments. The test street view images, which are taken at Times Square, Manhattan, New York, USA, are obtained from Google street view. Figs. 17-18 show the experimental results. We can see that the edges in the images obtained by our method are sharper and clearer than the edges obtained by bilinear interpolation and bicubic interpolation. These experiments demonstrate that our method can reconstruct HR street view images effectively.

5. Conclusion

A new image resolution improvement method by processing multiple dictionary pairs with latent Dirichlet allocation model is proposed in this paper. We divide patches into multiple clusters by using semantic information. Then LR-HR dictionary pairs are constructed for each cluster to represent various structures in images. Our method contains two stages: the learning stage and the reconstructing stage. We cluster patches at the semantic level, then learn a topic dictionary pair for each cluster in the learning stage. In the reconstructing stage, we assign a topic to each LR patch with semantic information, then reconstruct a HR patch with its corresponding topic dictionary pair. The eventual HR image is obtained by merging the HR patches. Experimental results validate our method is superior over the compared methods in both visual perception and objective quantitation.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (#61701327, #61711540303, and #61473198), National Research Foundation of Korea (#NRF-2017K2A9A2A06013711), Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) Fund, Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

References

[1] Anguelov D, Dulong C, Filip D, et al. Google street view: Capturing the world at street level[J]. Computer, 2010, 43(6): 32-38.

[2] Li X, Zhang C, Li W, et al. Assessing street-level urban greenery using Google Street View and a modified green view index[J]. Urban Forestry & Urban Greening, 2015, 14(3): 675-685.

[3] Zheng H, Qu X, Bai Z, et al. Multi-contrast brain magnetic resonance image super-resolution using the local weight similarity[J]. Bmc Medical Imaging, 2017, 17(1):6.

[4] Huang Y, Shao L, Frangi A F. Simultaneous Super-Resolution and Cross-Modality Synthesis of 3D Medical Images using Weakly-Supervised Joint Convolutional Sparse Coding[J]. arXiv preprint arXiv:1705.02596, 2017.

[5] Li L, Wang W, Luo H, et al. Super-Resolution Reconstruction of High-Resolution Satellite ZY-3 TLC Images[J]. Sensors, 2017, 17(5): 1062.

[6] Ducournau A, Fablet R. Deep learning for ocean remote sensing: an application of convolutional neural networks for super-resolution on satellite-derived SST data[C].Pattern Recogniton in Remote Sensing (PRRS), 2016 9th IAPR Workshop on. IEEE, 2016: 1-6.

[7] Yang J, Wright J, Huang T S, et al. Image Super-Resolution Via Sparse Representation[J]. IEEE Transactions on Image Processing. 2010, 19(11):2861-2873.

[8] Hou H, Andrews H. Cubic splines for image interpolation and digital filtering[J]. IEEE Transactions on acoustics, speech, and signal processing, 1978, 26(6): 508-517.

[9] Keys R. Cubic convolution interpolation for digital image processing[J]. IEEE Transactions on Acoustics Speech & Signal Processing, 2003, 29(6):1153-1160.

[10] Irani M, Peleg S. Improving resolution by image registration[J]. CVGIP: Graphical models and image processing, 1991, 53(3): 231-239.

[11] Hardie R C, Barnard K J, Armstrong E E. Joint MAP registration and high-resolution image estimation using a sequence of undersampled images[J]. IEEE Transactions on Image Processing, 1997, 6(12):1621.

[12] Elad M, Feuer A. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images[J]. IEEE transactions on image processing, 1997, 6(12): 1646-1658.

[13] Chang H, Yeung D Y, Xiong Y. Super-Resolution through Neighbor Embedding[C]. Computer Vision and Pattern Recognition, 2004: I-275- I-282 Vol.1.

[14] Zhang K, Gao X, Li X, et al. Partially supervised neighbor embedding for example-based image super-resolution[J]. IEEE Journal of Selected Topics in Signal Processing, 2011, 5(2): 230-239.

[15] Wang X, Tang X. Hallucinating face by eigentransformation[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2005, 35(3): 425-434.

[16] Wu W, Liu Z, He X. Learning-based super resolution using kernel partial least squares[J]. Image and Vision Computing, 2011, 29(6): 394-406.

[17] Zeyde R, Elad M, Protter M. On Single Image Scale-Up Using Sparse-Representations[M]. Curves and Surfaces. Springer Berlin Heidelberg, 2010:711-730.

[18] Dong C, Loy C C, He K, et al. Learning a deep convolutional network for image super-resolution[C]. European Conference on Computer Vision. Springer, Cham, 2014: 184-199.
[19] Kim J, Kwon Lee J, Mu Lee K. Deeply-recursive convolutional network for image super-resolution[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1637-1645.
[20] Dong C, Loy C C, Tang X. Accelerating the super-resolution convolutional neural network[C]. European Conference on Computer Vision. Springer International Publishing, 2016: 391-407.
[21] Wu W, Yang X, Liu K, et al. A new framework for remote sensing image super-resolution: sparse representation-based method by processing dictionaries with multi-type features[J]. Journal of Systems Architecture, 2016, 64: 63-75.
[22] Aharon M, Elad M, Bruckstein A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation[J]. IEEE Transactions on Signal Processing, 2006, 54(11):4311-4322.
[23] Purkait P, Chanda B. Image upscaling using multiple dictionaries of natural image patches[C]. Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012: 284-295.
[24] Zhou Z, Yang C N, Chen B, et al. Effective and efficient image copy detection with resistance to arbitrary rotation[J]. IEICE Transactions on information and systems, 2016, 99(6): 1531-1540.
[25] Wang J, Lian S, Shi Y Q. Hybrid multiplicative multi-watermarking in DWT domain[J]. Multidimensional Systems and Signal Processing, 2017, 28(2): 617-636.
[26] Xiong L, Xu Z, Shi Y Q. An integer wavelet transform based scheme for reversible data hiding in encrypted images[J]. Multidimensional Systems and Signal Processing, 2017: 1-12.
[27] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan): 993-1022.
[28] Fu Z, Huang F, Ren K, et al. Privacy-Preserving Smart Semantic Search Based on Conceptual Graphs Over Encrypted Outsourced Data[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(8): 1874-1884.
[29] Li J, Li X, Yang B, et al. Segmentation-based image copy-move forgery detection scheme[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(3): 507-518.
[30] Fu Z, Ren K, Shu J, et al. Enabling personalized search over encrypted outsourced data with efficiency improvement[J]. IEEE transactions on parallel and distributed systems, 2016, 27(9): 2546-2559.
[31] Fu Z, Huang F, Sun X, et al. Enabling semantic search based on conceptual graphs over encrypted outsourced data[J]. IEEE Transactions on Services Computing, 2016.
[32] Mallat S, Zhang Z. Adaptive time-frequency transform[C]. Acoustics, Speech, and Signal Processing, 1993, 3: 241-244.
[33] Yang X, Wu W, Liu K, et al. Multi-sensor image super-resolution with fuzzy cluster by using multi-scale and multi-view sparse coding for infrared image[J]. Multimedia Tools & Applications, 2017:1-32.
[34] Bevilacqua M, Roumy A, Guillemot C, et al. Low-complexity single-image super-resolution based on nonnegative neighbor embedding[J]. BMVC, 2012: 1-10.

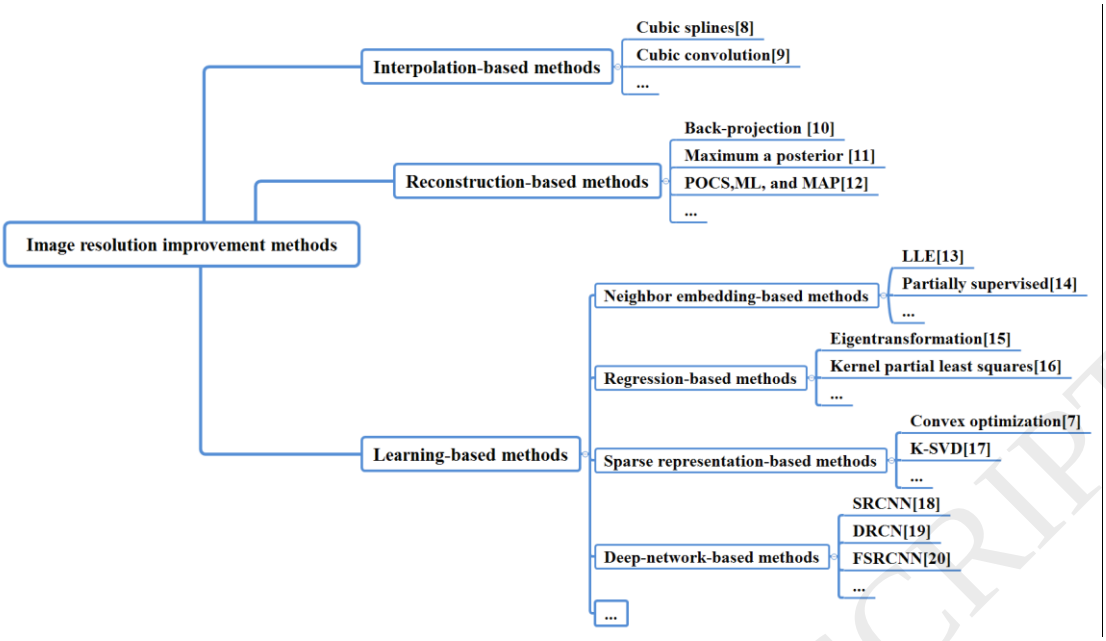

Figure 1. Summary of image resolution improvement methods.

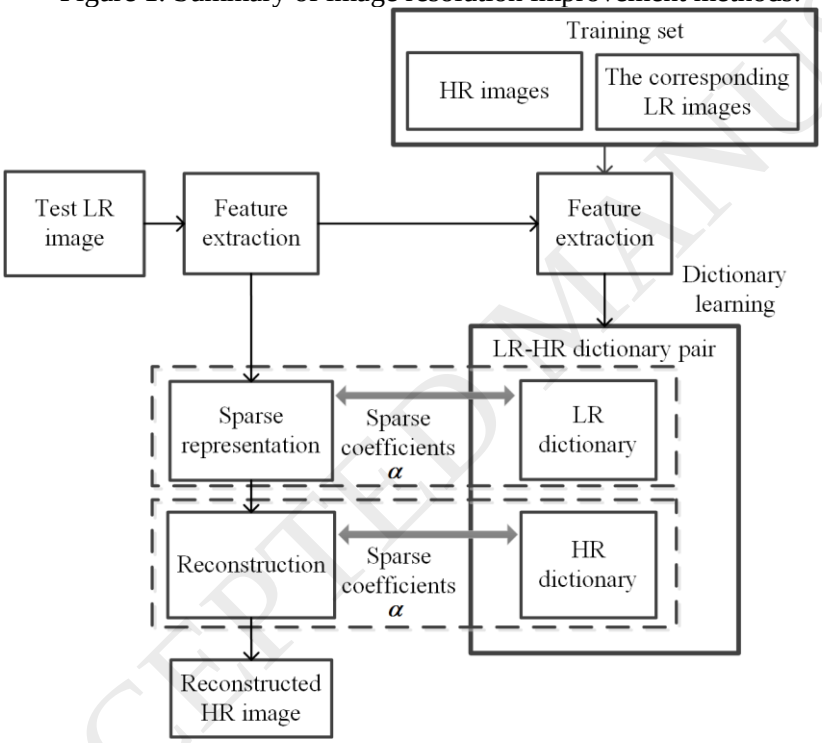

Figure 2. Framework of sparse representation-based methods.

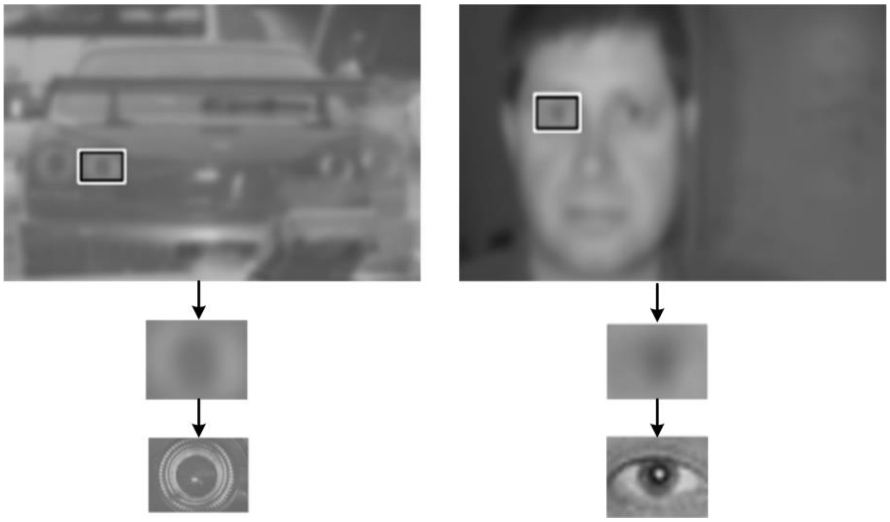

Figure 3. An illustration that two similar LR patches may come from two different HR patches.

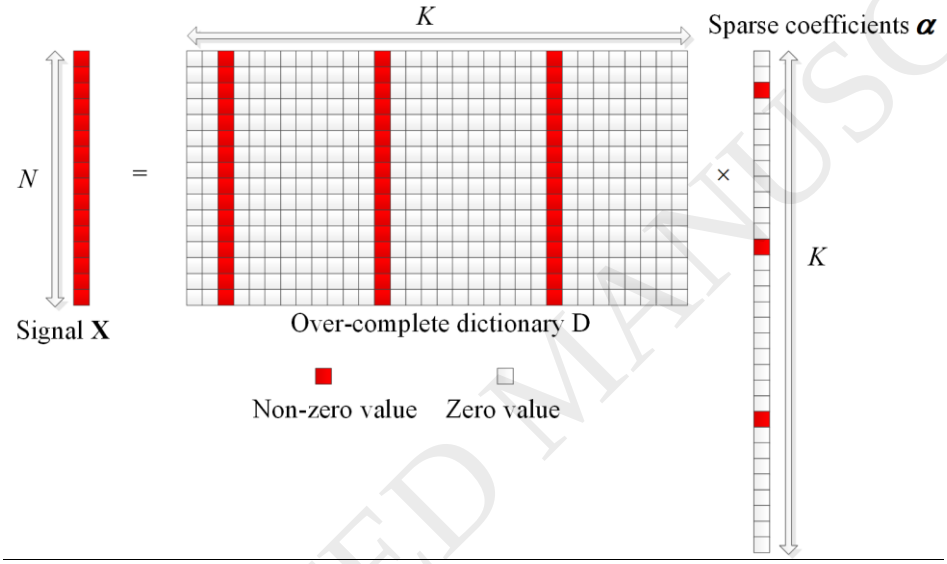

Figure 4. Sparse representation of a signal.

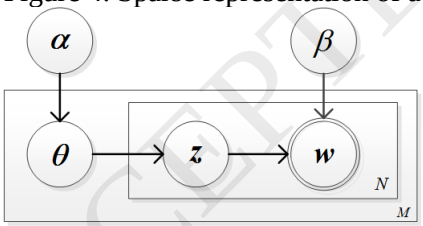

Figure 5. Illustration of LDA model. The number in the bottom right corner indicates the repetition times of nodes inside the box.

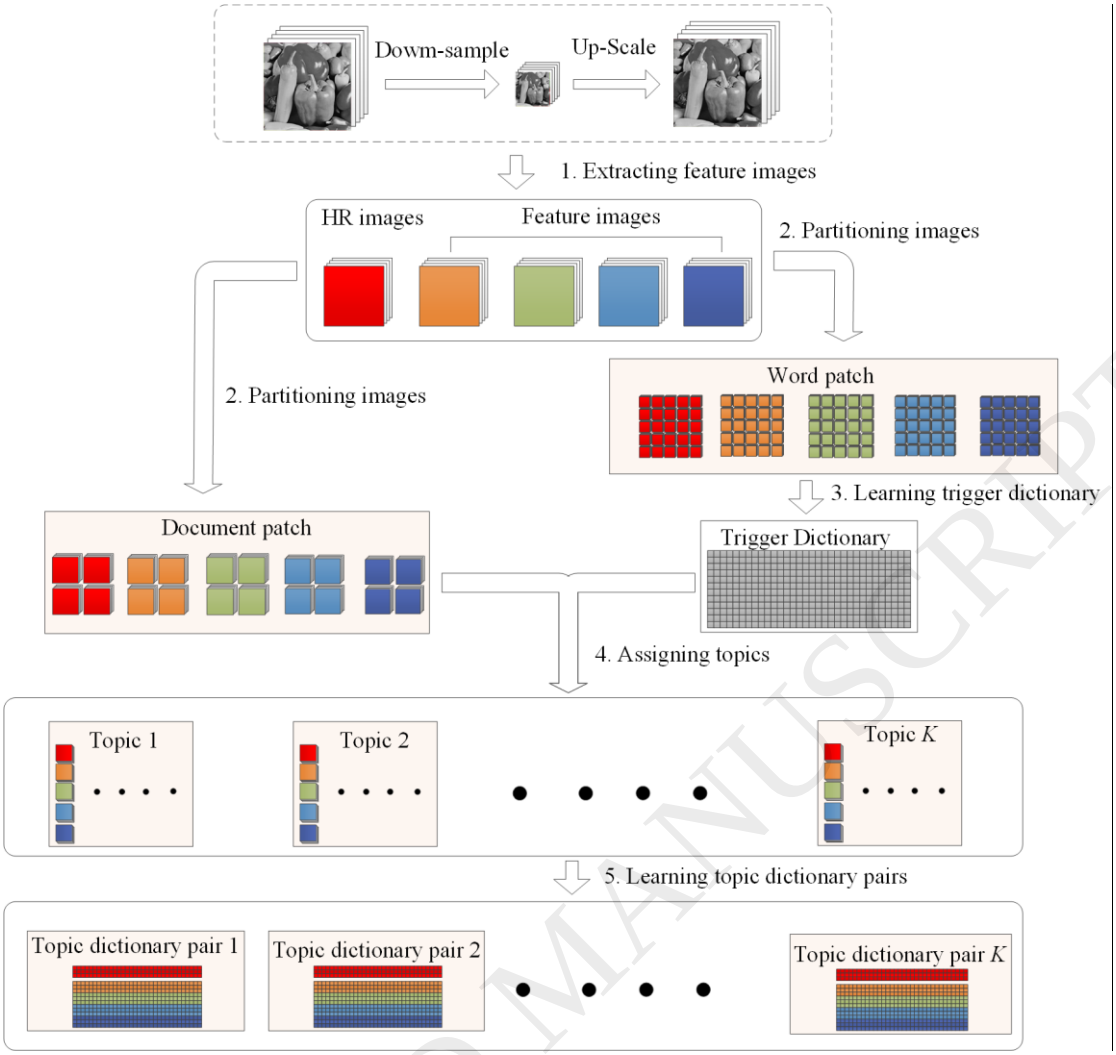

Figure 6. Summary of the learning stage.

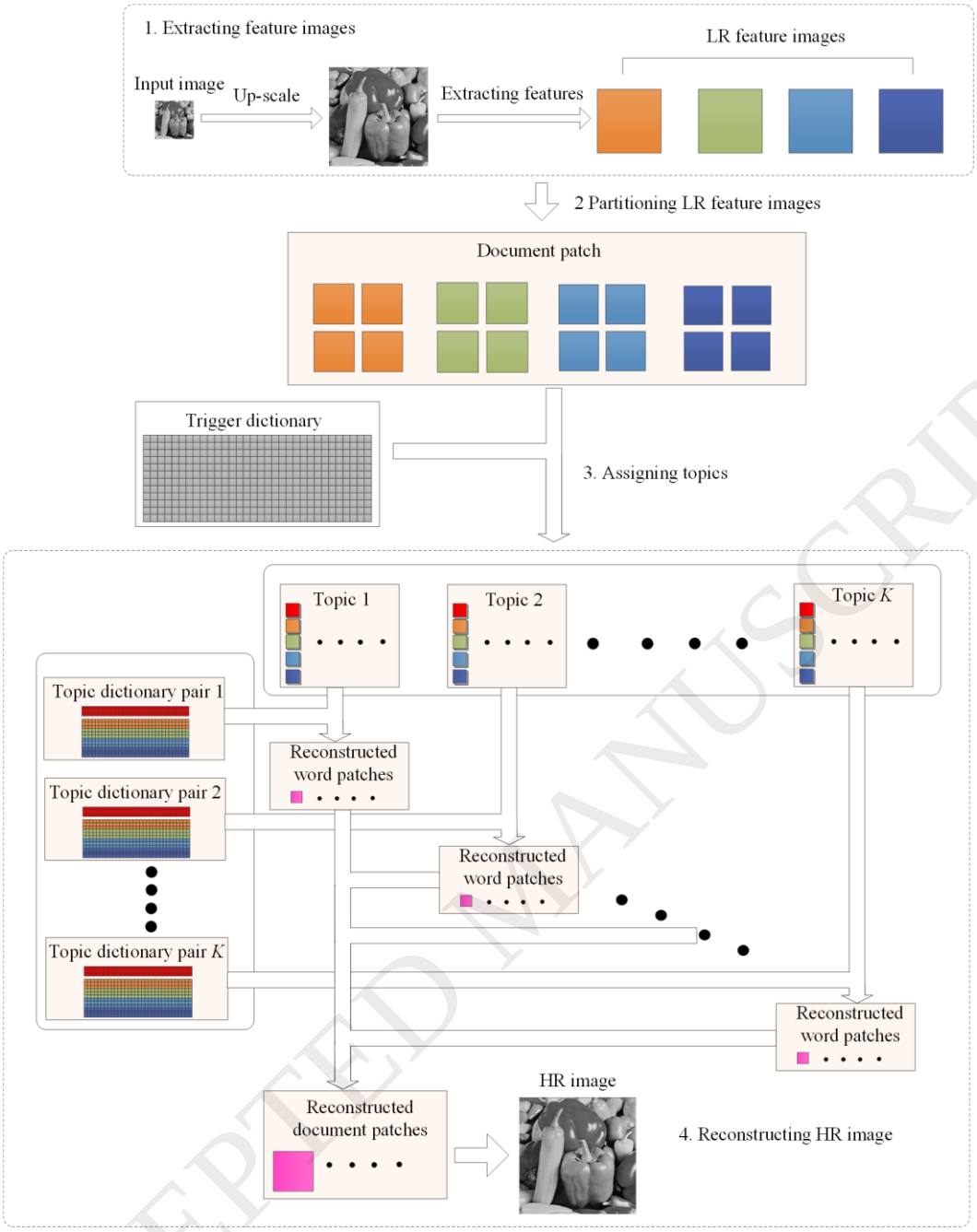

Figure 7. Summary of the reconstructing stage.


Baby


Barbara


Coastguard


Foreman


Monarch


Pepper

Figure 8. The test images selected from Set5 and Set14.

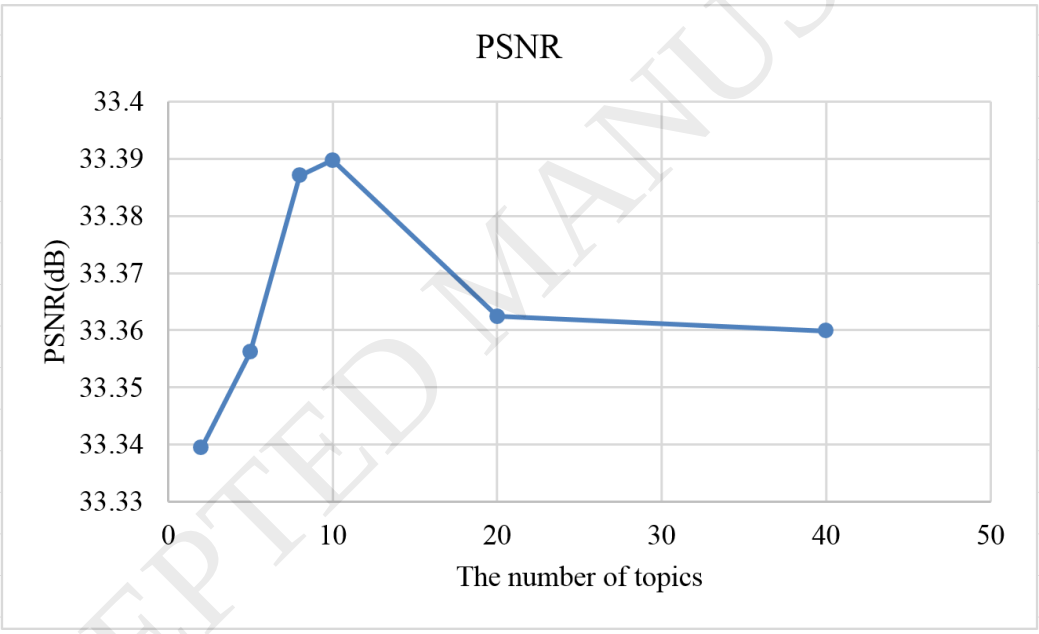

Figure 9. PSNR(dB) obtained with different numbers of topics.

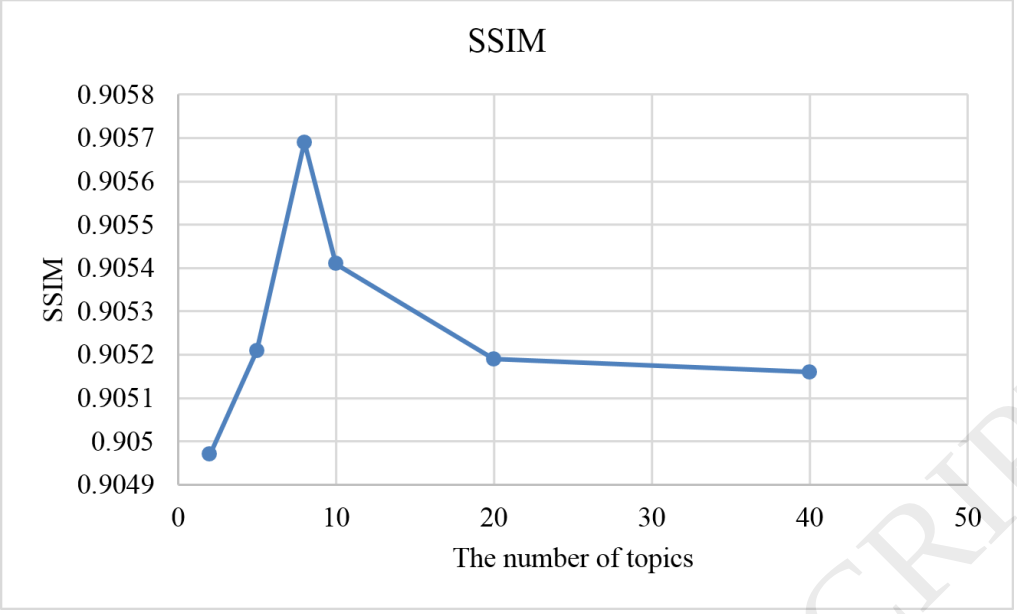

Figure 10. SSIM obtained with different numbers of topics.

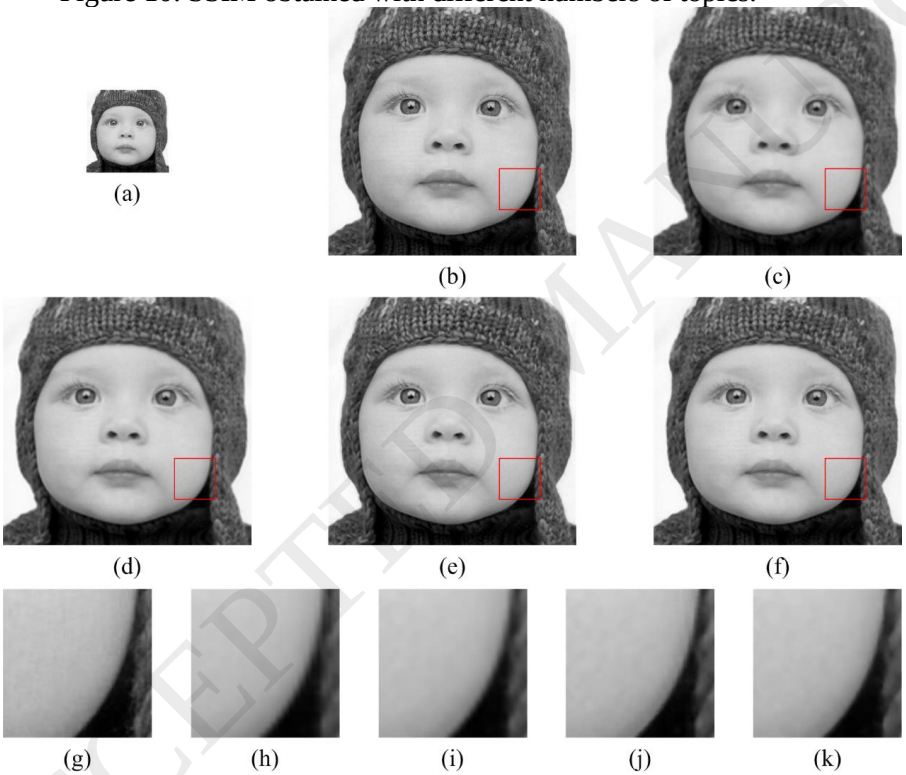

Figure 11. Visual comparison of image 'Baby': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

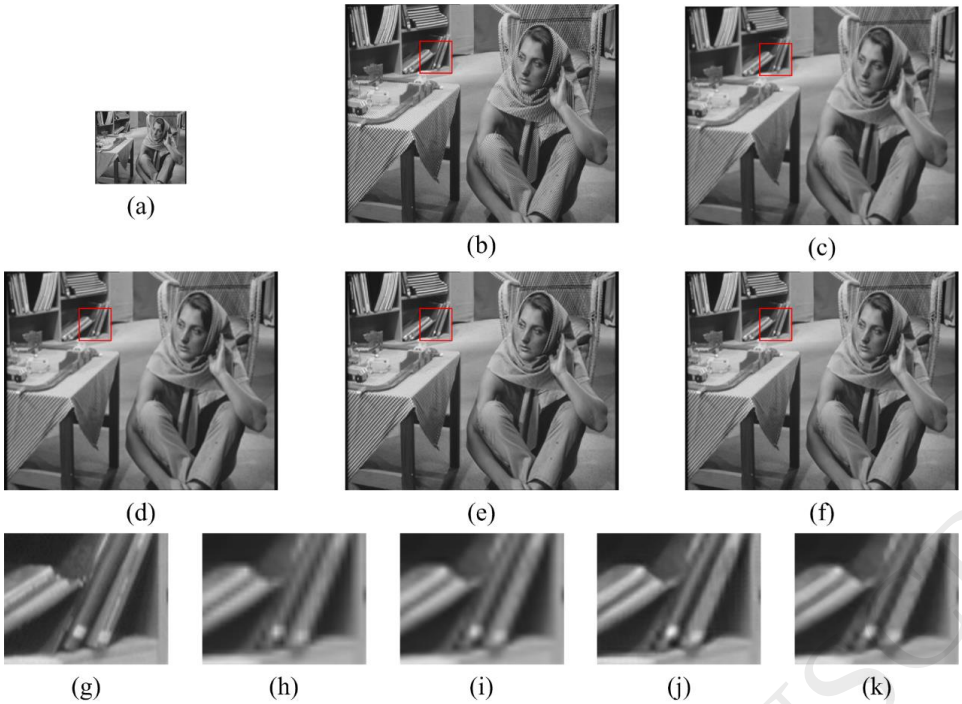

Figure 12. Visual comparison of image 'Barbara': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

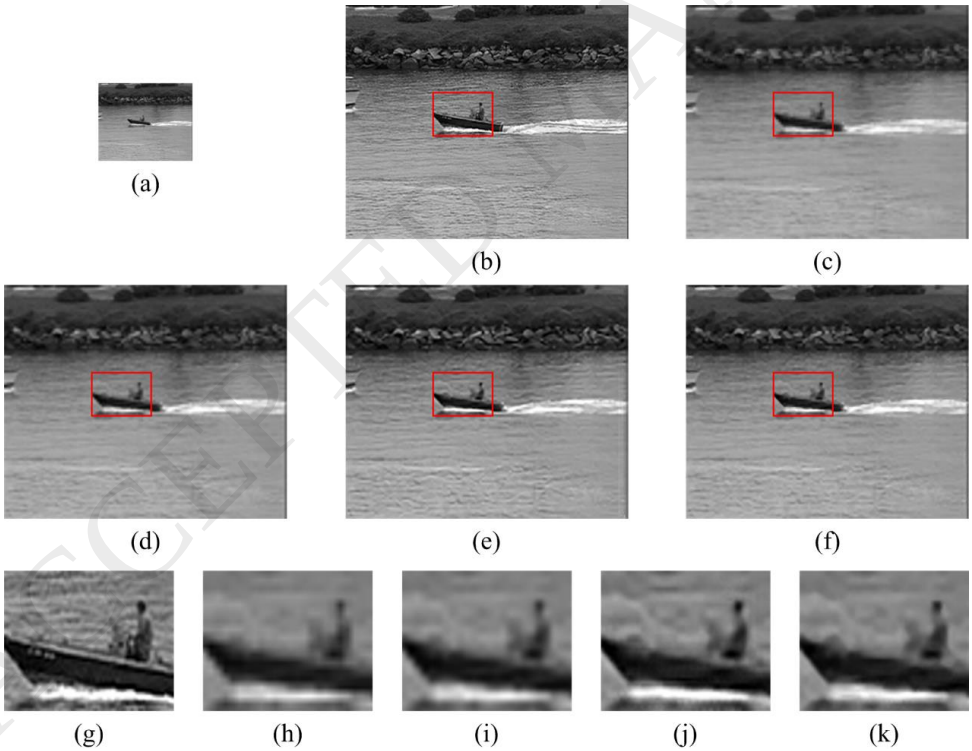

Figure 13. Visual comparison of image 'Coastguard': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

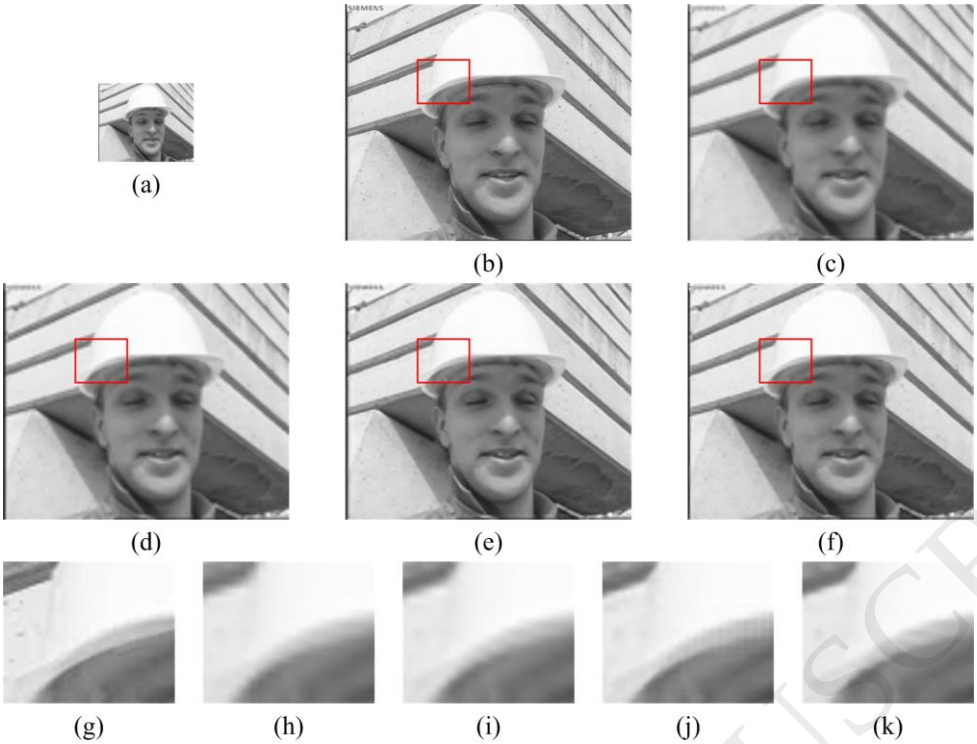

Figure 14. Visual comparison of image 'Foreman': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

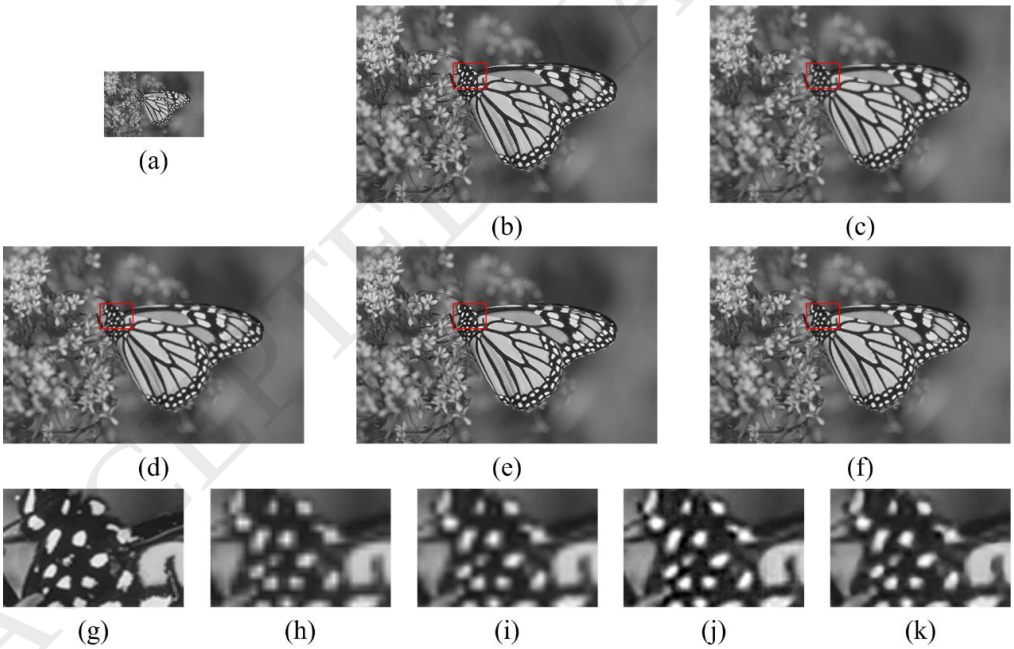

Figure 15. Visual comparison of image 'Monarch': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

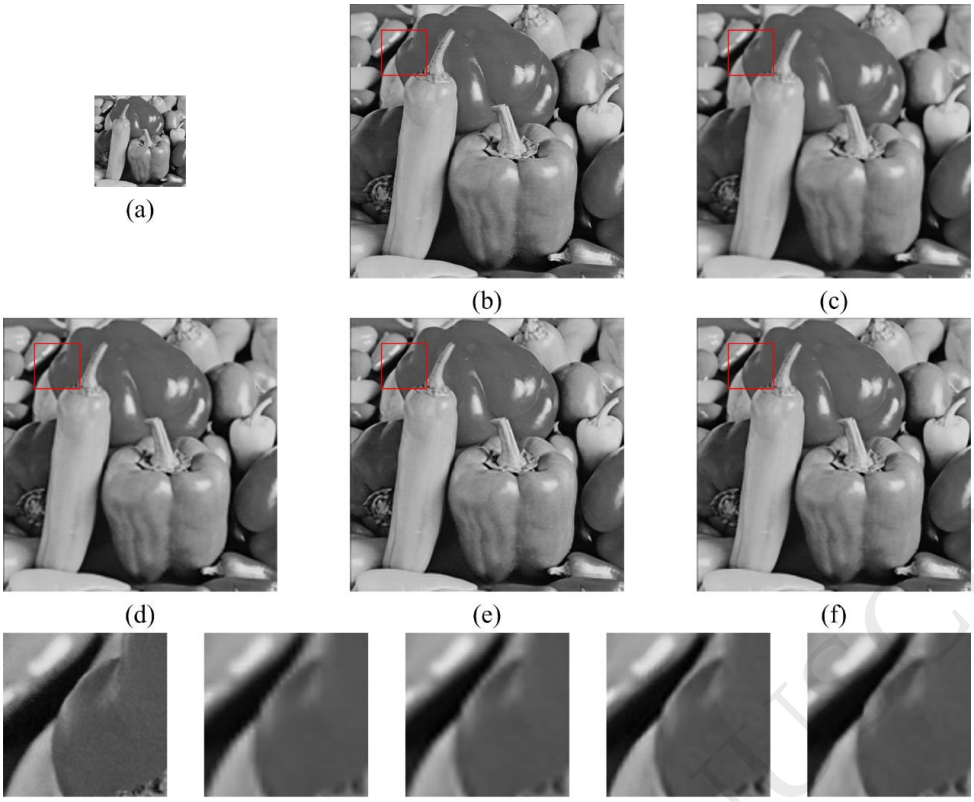

Figure 16. Visual comparison of image 'Pepper': (a) LR image. (b) original HR image. (c) HR image obtained by bilinear interpolation. (d) HR image obtained by bicubic interpolation. (e) HR image obtained by Yang's method. (f) HR image obtained by our method. (g)-(k) the magnified version of the marked area of (b)-(f).

Figure 17. Visual comparison of a street view image: (a) original street view image. (b) the marked area in (a). (c) the upscaled image of (b) obtained by bilinear interpolation. (d) the

upscaled image of (b) obtained by bicubic interpolation. (e) the upscaled image of (b) obtained by our method. (f)-(h) the magnified version of the marked area of (c)-(e).

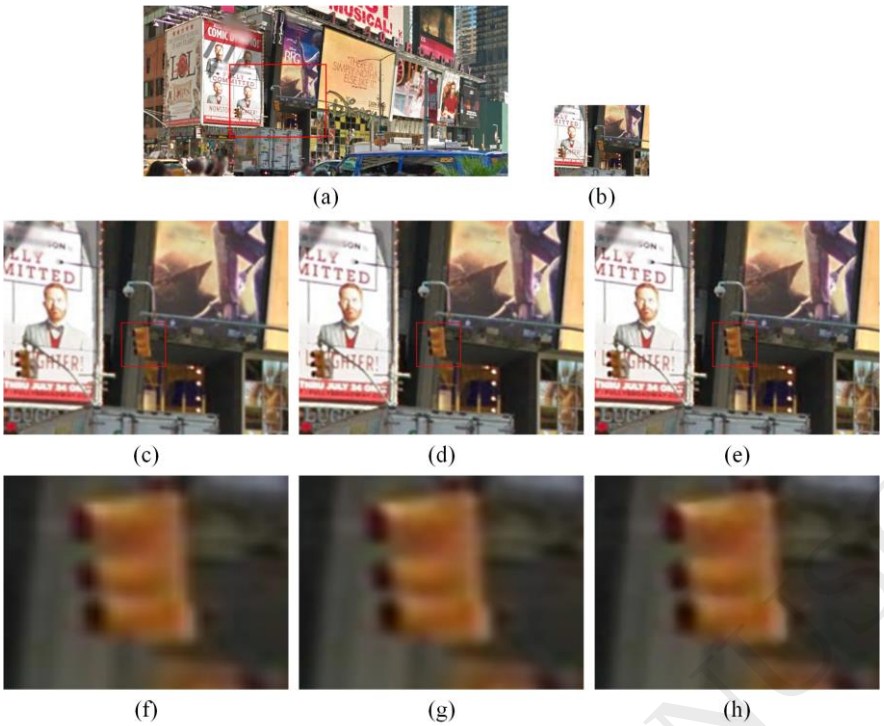

Figure 18. Visual comparison of a street view: (a) original street view image. (b) the marked area in (a). (c) the upscaled image of (b) obtained by bilinear interpolation. (d) the upscaled image of (b) obtained by bicubic interpolation. (e) the upscaled image of (b) obtained by our method. (f)-(h) the magnified version of the marked area of (c)-(e).

Table 1. The key parameters of our experiment.

Key parameters	Value
Upscaling factor	3
Word patch size	3×3
Overlap for word patch	1
Overlap for document patch	8
The number of topics	10
The number of word patches in a document patch	49
Trigger dictionary size	500
The size of LR dictionary in a topic dictionary pair	500
The size of HR dictionary in a topic dictionary pair	500
The number of document patches extracted from the training set in the learning stage	10000

Table 2. Numerical results of Figs. 11-16.

Images	Bilinear interpolation	Bicubic interpolation	Yang's method	Our method
	PSNR SSIM	PSNR SSIM	PSNR SSIM	PSNR SSIM
Baby	31.654 0.875	32.564 0.892	33.096 0.904	**33.390** **0.905**
Barbara	24.514 0.718	24.877 0.739	25.049 **0.759**	**25.229** **0.759**
Coastguard	24.705 0.561	24.982 0.583	25.337 **0.627**	**25.444** 0.612
Foreman	27.634 0.877	28.336 0.891	28.946 0.906	**29.519** **0.913**
Monarch	27.260 0.899	28.127 0.912	29.528 0.925	**29.755** **0.929**
Pepper	28.779 0.835	29.357 0.846	30.081 0.855	**30.729** **0.860**
Average	27.424 0.794	28.041 0.811	28.673 0.829	**29.011** **0.830**