Single image super-resolution reconstruction using multiple dictionaries and improved iterative back-projection*

ZHAO Jian-wen (赵建雯), **YUAN Qi-ping** (袁其平)**, **QIN Juan** (秦娟), **YANG Xiao-ping** (杨晓苹), **and CHEN Zhi-hong** (陈志宏)

Tianjin Key Laboratory of Film Electronic and Communication Devices, School of Electrical and Electronic Engineering, Tianjin University of Technology, Tianjin 300384, China

(Received 23 August 2018; Revised 11 November 2018)

In order to improve the super-resolution reconstruction effect of the single image, a novel multiple dictionaries learning via support vector regression (SVR) and improved iterative back-projection (IBP) are proposed. To characterize the image structure, the low-frequency dictionary is constructed from the normalized brightness of low-frequency image patches in a discrete-cosine-transform (DCT) domain. Pixels determined by Gaussian weighting are added to the input vector to restore more high-frequency information when training the high-frequency image patch dictionary in the space domain. During post-processing, the improved IBP is employed to reduce regression errors each time. Experiment results show that the peak signal-to-noise ratio (*PSNR*)and structural similarity (*SSIM*) of the proposed method are enhanced by 1.6%—5.5% and 1.5%—13.1% compared with those of bicubic interpolation, and the proposed method visually outperforms several algorithms.

Document code: A **Article ID:** 1673-1905(2019)02-0156-5

DOI https://doi.org/10.1007/s11801-019-8138-x

Image super-resolution (SR) reconstruction aims to recover high-frequency information from one or more low-resolution (LR) images[1]. Because it includes more image details, high-resolution (HR) images are widely used in medicine, remote sensing and video surveillance as well as by the military[2]. However, the cost of improving hardware to obtain HR images is expensive, so the research on image reconstruction algorithm is essential.

SR reconstruction algorithms are broadly classified into three categories: interpolation, reconstruction-based methods and example-based methods. The first type of algorithm is simple but tends to produce artifacts at the edges of the image[3]. The second one utilizes the rich information obtained from multiple complementary LR images, but it requires prior knowledge of the image for reconstruction[4,5]. The learning method can obtain more high-frequency details by learning the relationships between the LR and HR images[6-9]. Timofte et al[10] proposed an algorithm that combined anchored neighborhood regression (ANR) with simple functions to improve the quality of ANR. It was based on image features and anchored regression instead of using a dictionary to learn regression. Wang et al[11] proposed a method based on support vector regression (SVR) and self-similarity of image patches. Their method produced better reconstruction effect that has highly similar textural structure. Lin et al[12] proposed an algorithm that applied cyclical-scan actions and SVR to single-image reconstruction and found that it was more competitive than other methods. However, it did not make full use of the characteristics of the high- and low-frequency image patches because it adopted the same feature vector to learn from the high- and low-frequency dictionaries. To avoid this problem, a novel method that uses different feature vectors to create different dictionaries is presented. First, the Log algorithm is adopted to distinguish high- and low-frequency image patches, and different feature vectors are extracted to train the different dictionaries. The SVR is applied to construct two dictionaries using the high- and low-frequency image patches in space and discrete-cosine-transform (DCT) domains, respectively. It is noteworthy that some of the pixel used for training the high-frequency dictionary are determined by Gaussian weighting. Moreover, the normalized brightness of the image patches in a DCT domain is employed as the input vector to train the low-frequency dictionary because the energy of the image is mainly concentrated in the low-frequency components in the DCT domain. Then, the improved iterative back-projection (IBP) is used to

* This work has been supported by the Tianjin Applied Basic and Frontier Technology Research Program of Youth Fund Funding Project (No.14JCQNJC00900), and the Tianjin Education Commission Project (No.2018kj132).

** E-mail: Yqp1962@163.com

reduce the regression errors in the regression image. Experimental findings show that the proposed method performs more accurately than several SR algorithms.

The SVR linearly estimates the output of the nonlinear input in a higher dimensional feature space. SVR[13] is used to solve the following optimization problem:

$$\min_{\omega,b,\zeta,\zeta^*} \frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+c\sum_{i=1}^{k}\left(\zeta_i+\zeta_i^*\right),$$

$$s.t.\begin{cases} \boldsymbol{y}_i-\left(\boldsymbol{\omega}^{\mathrm{T}}\phi\left(\boldsymbol{x}_i\right)+b\right)\le\varepsilon+\zeta_i \\ \left(\boldsymbol{\omega}^{\mathrm{T}}\phi\left(\boldsymbol{x}_i\right)+b\right)-\boldsymbol{y}_i\le\varepsilon+\zeta_i^*, \\ \zeta_i,\zeta_i^*\ge 0, i=1,...,k \end{cases} \tag{1}$$

where $\boldsymbol{x}_i$ represents the feature input vector, $\boldsymbol{y}_i$ is the pixel label, and $\boldsymbol{\omega}$ is the norm vector of a nonlinear mapping function. The trade-off c is a constant that lies between the upper and lower training error bounds ζ_i and ζ_i^*, respectively, which are subject to threshold ε. The number of training samples is k, and $\boldsymbol{x}_i$ is mapped into high dimensional space by $\phi(\boldsymbol{x}_i)$. In this method, $\boldsymbol{y}_i$ is formed from the central pixel value of the high-resolution image patch, and b is the offset of the regression model. The kernel function in the proposed method is a radial basis kernel function (RBF), with the parameters c=22, σ=0.01 and ε=0.1, which can map data onto a high-dimensional feature space.

In the learning phase, the proposed method treats the normalized brightness of the low-frequency image patches as an input vector in the DCT domain. Some pixels, which are determined by Gaussian weighting in the high-frequency image patches, are extracted to form the input vector in the space domain. The central pixel of the HR image patch is added to the label vector. According to Eq.(1), this procedure generates two dictionaries.

The basic principle of IBP is that it minimizes the errors between two LR images by back-projecting the residual to obtain the final HR image, which is shown as

$$\boldsymbol{X}^{m+1}=\boldsymbol{X}^m+\boldsymbol{M}^{\mathrm{BP}}(\boldsymbol{Y}-\boldsymbol{Y}^m), \tag{2}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the HR and LR images, respectively, and m represents the number of iterations. The observed LR image is $\boldsymbol{Y}^m$, and $\boldsymbol{X}^m$ is the reconstructed image. The quantity $\boldsymbol{M}^{\mathrm{BP}}$ is the back-projection matrix and is used in image reconstruction. Xu et al[14] proposed the use of bicubic interpolation (BI) to replace $\boldsymbol{M}^{\mathrm{BP}}$ in IBP.

The dictionary-learning algorithm mainly learns a mapping between LR and HR images. However, it is difficult to uniquely determine $\boldsymbol{M}^{\mathrm{BP}}$ in the traditional IBP. To avoid this problem and reduce regression errors, an improved IBP is used in the post-processing. It is defined as

$$\begin{cases} \boldsymbol{X}_R^m=\boldsymbol{X}_r^m+BI\left(\boldsymbol{Y}-\boldsymbol{Y}^m\right) \\ \boldsymbol{X}_r^m=R\left(\boldsymbol{X}_R^m\right) \end{cases}, \tag{3}$$

where BI represents bicubic of the errors between $\boldsymbol{Y}$ and $\boldsymbol{Y}^m$. In order to obtain $\boldsymbol{X}_R^m$, these errors are added to $\boldsymbol{X}_r^m$, where $\boldsymbol{X}_r^m$ is got by SVR using two dictionaries. Each reconstruction process is regarded as R, so $\boldsymbol{X}_r^m$ is different from $\boldsymbol{X}^m$ of the traditional IBP, and this method improves the quality of the reconstruction.

The proposed method comprises three phases: training, prediction and post-processing. The training phase is shown in Fig.1, and the specific steps are as follows.

(i) To generate a training set, the degradation model shown in Eq.(4) is used to obtain a LR image:

$$\boldsymbol{Y}=DB\boldsymbol{X}+\boldsymbol{n}, \tag{4}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ represent the HR and LR images, respectively, D is the downsampling operator, B is the blurring of the HR image, and $\boldsymbol{n}$ represents noise pollution. Many denoising algorithms can effectively reduce the noise[15], so the noise is not added to the LR image in the pre-processing phase of this method.

The HR image is blurred and downsampled with a scale factor of 2 to create an LR image. BI is adopted to produce an image of the desired size from this LR image, and the resulting image is termed as image Ib.

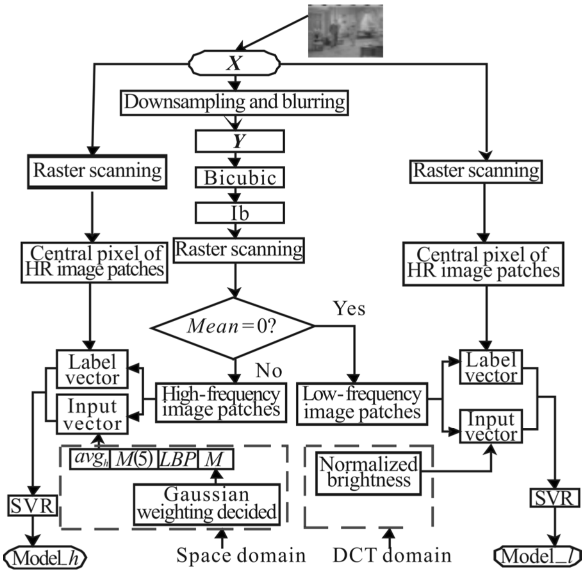


Fig.1 Training process

(ii) 3×3 image patches are extracted after using the raster scan of image Ib. The Log operator is adopted to determine whether the patches contain low- or high-frequency information, which depends on the mean of the edge image patch[12]. The edge image is obtained by Log operator. When the mean is zero, a corresponding patch of the image Ib mainly contains low-frequency information, and the normalized brightness that better reflects the low-frequency characteristics in the DCT domains is regarded as the input vector. The input vector is expressed as

$$\boldsymbol{x}_{li}=\Big[\boldsymbol{S}_{l1}\left(DCT\left(P\left(Ib\right)\right)-avg_l\right);...;$$
$$\boldsymbol{S}_{lj}\left(DCT\left(P\left(Ib\right)\right)-avg_l\right)\Big], \tag{5}$$

where P indicates the raster scan of the image, and avg_l is the mean of the DCT-transformed image patch. Subscript i runs over the number of raster scans, and subscript j is the number of low-frequency image patches. $\boldsymbol{S}_{li}$ represents the *j*th low-frequency input vector after the *i*th raster scan of the image, and it contains nine elements that are formed of the normalized brightness. The input vector $\boldsymbol{x}_{li}$ is comprised of $\boldsymbol{S}_{li}$, and the label vector $\boldsymbol{y}_{li}$ is constituted by the central pixel of the corresponding HR image patch.

(iii) When the mean of the edge image patch is greater than zero, a patch of the image Ib mainly contains high-frequency information. The feature value is extracted in the space domain, and the input vector is represented as

$$\boldsymbol{x}_{hi}=\left[\boldsymbol{S}_{h1}\left(avg_h, M(5), LBP, M\right);...;\right.$$
$$\left.\boldsymbol{S}_{hj}\left(avg_h, M(5), LBP, M\right)\right], \qquad (6)$$

where the local binary pattern (*LBP*) has invariance of rotation and grayscale, the *M*(5) is the central pixel, and *M* includes some pixels determined by Gaussian weighting. Gaussian weighting depends on the distances between the central and neighboring pixels, which defines the relationship between the pixels. The average of the image patch pixel is the quantity avg_h in the space domain. Here, $\boldsymbol{S}_{hi}$ represents the *j*th high-frequency input vector after the *i*th raster scan of the image Ib, and the length of $\boldsymbol{S}_{hi}$ is seven, which contains avg_h, *M*(5), *LBP* and *M*. The input vector $\boldsymbol{x}_{hi}$ is formed of $\boldsymbol{S}_{hi}$. The central pixel of the HR image patches is included in the label vectors $\boldsymbol{y}_{hi}$.

The resulting input vectors $\boldsymbol{x}_l$ and $\boldsymbol{x}_h$ and the label vectors $\boldsymbol{y}$ and $\boldsymbol{y}_h$ are shown as

$$\begin{cases}\boldsymbol{x}_l=\left[\boldsymbol{x}_{l1};\boldsymbol{x}_{l2};...;\boldsymbol{x}_{li};...;\boldsymbol{x}_{ln}\right]\\ \boldsymbol{x}_h=\left[\boldsymbol{x}_{h1};\boldsymbol{x}_{h2};...;\boldsymbol{x}_{hi};...;\boldsymbol{x}_{hn}\right]\end{cases},\left(i=1,2,...,9\right), \qquad (7)$$

$$\begin{cases}\boldsymbol{y}_l=\left[\boldsymbol{y}_{l1};\boldsymbol{y}_{l2};...;\boldsymbol{y}_{li};...;\boldsymbol{y}_{ln}\right]\\ \boldsymbol{y}_h=\left[\boldsymbol{y}_{h1};\boldsymbol{y}_{h2};...;\boldsymbol{y}_{hi};...;\boldsymbol{y}_{hn}\right]\end{cases},\left(i=1,2,...,9\right), \qquad (8)$$

and they are constituted after nine raster scans.

(iv) Two dictionaries of model_*l* and model_*h* are trained by SVR using the optimized vector pairs.

The prediction and post-processing phases are shown in Fig.2.

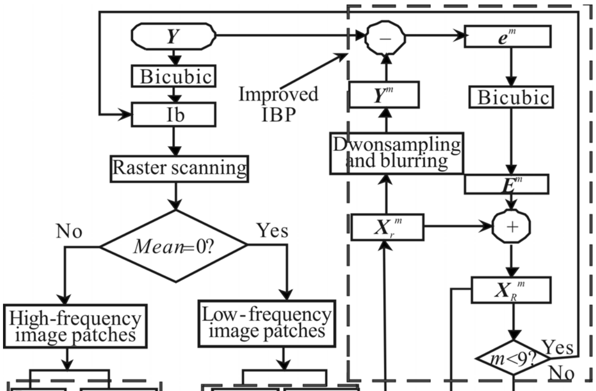


Fig.2 Prediction and post-processing processes

In the prediction phase, the input vector is formed by using the same method as training process, and the predicted label pixels replace the central pixels of image patches to obtain the regression image.

The post-processing phase mainly adopts the improved IBP algorithm to reduce the regression errors and ensure consistency between the reconstructed and LR images. According to Eq.(3), represent errors of two LR images. *BI* is adopted to obtain the desired residuals $\boldsymbol{E}^m=BI(\boldsymbol{X}-\boldsymbol{X}^m)$ in the improved IBP. The quantity $\boldsymbol{E}^m$ is added to the regression image $\boldsymbol{X}_r^m$ which is reconstructed using two dictionaries. The quantity *m* is the number of raster scans.

By the experiment, the proposed method is compared with four other SR algorithms using different images extracted from the USC-SIPI image database. The training image size is 512×512, and two image sizes of 256×256 and 512×512 are used for the testing set. In the training phase, the LR image is obtained from a blurred and downsampled version of the HR image. The scale factor of 2 is applied for the decimation operator, and image Ib is obtained from the LR image via *BI*. 3×3 patches are utilized, which are obtained after segmenting image Ib using raster scans, and the corresponding 3×3 HR image patches are also extracted. The LIBSVM[16] is applied for the SVR model in which the kernel function is an RBF. To test this method, HR images with different sizes are used in the testing set, which contain people, scenes and animals. They are shown in Fig.3 and are numbered as No.1—No.8.

Fig.3 Images in testing, from left to right, top row: No.1—No.4, bottom row: No.5—No.8

The peak signal-to-noise ratio (*PSNR*) and the structural similarity (*SSIM*) are selected as the evaluation criteria. The *PSNR* is defined as

$$PSNR=10\log_{10}\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I\left(i,j\right)-I'\left(i,j\right)\right)^2}, \qquad (9)$$

where *I*(*i*, *j*) and *I'*(*i*, *j*) represent the pixel values at coordinate of (*i*, *j*). The size of the image is *M*×*N*, and the maximum grayscale value of the image is 255. The *SSIM* between two images is defined as

$$SSIM=\frac{\left(2\mu_x\mu_y+c_1\right)+\left(2\sigma_{xy}+c_2\right)}{\left(\mu_x^2+\mu_y^2+c_1\right)\left(\sigma_x^2+\sigma_y^2+c_2\right)}, \qquad (10)$$

where μ and σ are the mean and variance of image x or y, respectively, and σ_{xy} is the covariance of the images. c_1 and c_2 are constants that are used to maintain stability.

Comparisons of the values of *PSNR* and *SSIM* for bicubic (BI), SCSR[6], A+[10], Lin[12], CNN[17] and the proposed method (Pro) are shown in Tabs.1 and 2.

Tab.1 Performance in terms of *PSNR* (dB)

No	BI	CNN	A+	SCSR	Lin	Pro
1	29.473 8	29.869 7	29.887 7	29.918 8	30.467 2	**30.876 4**
2	29.770 5	29.797 5	29.986 7	29.989 2	29.777 4	**30.252 2**
3	23.074 0	23.485 3	23.420 8	23.493 5	24.047 0	**24.340 9**
4	21.626 4	21.831 4	21.851 8	21.826 5	22.064 8	**22.287 8**
5	27.140 1	27.447 4	27.617 5	27.624 4	27.671 6	**28.114 9**
6	26.180 3	26.601 8	26.650 4	26.656 2	27.275 1	**27.409 6**
7	26.305 4	26.718 7	26.745 6	26.731 2	27.291 5	**27.643 2**
8	22.523 5	22.882 1	22.942 2	22.982 6	23.452 7	**23.707 4**

Tab.2 Performance in terms of *SSIM*

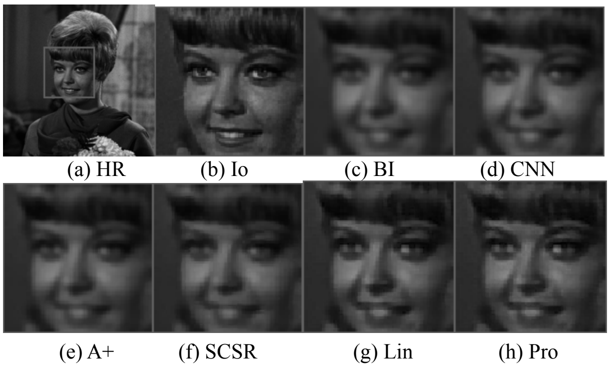
No	BI	CNN	A+	SCSR	Lin	Pro
1	0.866 8	0.876 1	0.877 4	0.875 8	0.885 8	**0.886 3**
2	0.948 2	0.951 2	0.952 0	0.953 4	0.961 2	**0.962 2**
3	0.762 8	0.781 5	0.779 0	0.778 6	0.803 6	**0.797 3**
4	0.767 1	0.786 6	0.789 0	0.792 4	0.861 2	**0.867 9**
5	0.947 6	0.953 1	0.954 0	0.956 2	0.972 2	**0.972 7**
6	0.925 9	0.933 5	0.934 7	0.938 1	0.962 0	**0.962 4**
7	0.897 5	0.908 0	0.909 8	0.913 0	0.946 7	**0.948 8**
8	0.694 1	0.725 6	0.725 2	0.726 8	0.777 3	**0.780 0**

Tabs.1 and 2 summarize that the proposed method enhances *PSNR* and *SSIM* by 1.6%—5.5% and 1.5%—13.1%, respectively, in comparison with BI. Moreover, it produces 1.6% improvement in *PSNR* for image No.2 and enhances *SSIM* by 0.8% for image No.4 in comparison with Lin's algorithm. According to these quantitative comparisons, the proposed method yields better results for different images.

In terms of the visual perception, we compare image No.1 and image No.5, which contain rich details and contour information, respectively. The HR image, a magnified detail of the HR image called Io, and magnified details produced by different algorithms are shown in Figs.4 and 5.

Among these images, the BI method has the most ambiguity due to the loss of a large amount of high-frequency information. CNN and A+ algorithms can recover some of the detailed information, but blurring still exists. SCSR and Lin algorithms also recover some high-frequency information, e.g., the black digit area in Fig.5(e) and (f). However, the proposed method can restore more edge information and provides a better visual appearance than other algorithms, e.g., the eye area in Fig.4(g) and the white digit area in Fig.5(g) are clearer than those observed by other methods.

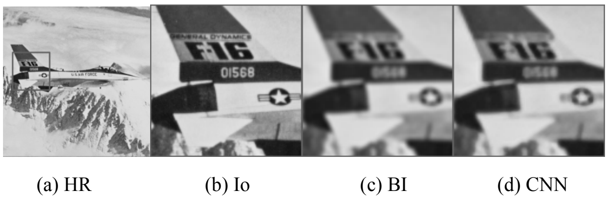
Thus, the method not only improves the objective effect in *PSNR* and *SSIM* but also significantly outperforms several methods visually.



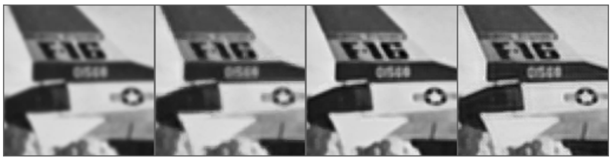
(a) HR (b) Io (c) BI (d) CNN

(e) A+ (f) SCSR (g) Lin (h) Pro

Fig.4 (a) The HR image, (b) a magnified detail and magnified details produced by (c) BI, (d) CNN[17] (e) A+[10], (f) SCSR[6], (g) Lin[12] and (h) the proposed method for the image people



(a) HR (b) Io (c) BI (d) CNN



(e) A+ (f) SCSR (g) Lin (h) Pro

Fig.5 (a) The HR image, (b) a magnified detail and magnified details produced by (c) BI, (d) CNN[17] (e) A+[10], (f) SCSR[6], (g) Lin[12] and (h) the proposed method for the image plane

In this work, a novel method is adopted which trains and uses multiple dictionaries to learn the mapping between the LR and HR image patches and employs an improved IBP to process the regression image. It takes full advantage of the image information to recover missing high-frequency details. In contrast to conventional methods, low- and high-frequency image patch dictionaries are learned via SVR from the LR and HR image patches using different feature vectors in the DCT and space domains, respectively. Furthermore, the accuracy of the recovered image is improved, and the regression errors are reduced by employing the improved IBP. The experimental results show this method outperforms several SR methods from both the objective and subjective standpoints.

References

[1] Liu Chanzi, Chen Qingchun and Li Hengchao, Multimedia Tools and Applications **76**, 14759 (2017).

[2] Yang Qi, Zhang Yanzhu, Zhao Tiebiao and Chen Yangquan, ISA Transactions **82**, 163 (2017).

[3] Zhang Xiang-jun and Wu Xiao-lin, IEEE Transactions on Image Processing **17**, 887 (2008).

[4] Kourosh Jafari-Khouzani, IEEE Transactions on Medical Imaging **33**, 1969 (2014)

[5] Dai Shao-sheng, Liu Jin-song, Xiang Hai-yan, Du Zhi-hui and Liu Qin, Optoelectronics Letters **10**, 313 (2014).

[6] Yang J.,Wang Z., Lin Z., Cohen S. and Huang T., IEEE Transactions on Image Processing **21**, 3467 (2012).

[7] Wang Zhang-yang, Yang Ying-zhen, Wang Zhao-wen,Chang Shi-yu,Han Wei, Yang Jian-chao and Thomas S. Huang, Self-Tuned Deep Super Resolution, IEEE Conference on Computer Vision and Pattern Recognition Workshops, 1 (2015).

[8] Huang Yuan-fei, Li Jie, Gao Xin-bo, He Li-huo and Lu Wen, IEEE Transactions on Image Processing **27**, 5904 (2018).

[9] Huang De-tian,Huang Wei-qin,Huang Hui and Zheng Li-xin, Optoelectronics Letters **13**, 439 (2017).

[10] Radu Timofte, Vincent De Smet and Luc Van Gool, A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution, Asian Conference on Computer Vision, 111 (2014).

[11] Wang Hong, Lu Fang-fang and Li Jian-wu, Journal of Image and Graphics **21**, 986 (2016). (in Chinese)

[12] Yuan Qi-ping, Lin Hai-jie,Chen Zhi-hong and Yang Xiao-ping, Optics and Precision Engineering **24**, 2302 (2016). (in Chinese)

[13] Ni K. S. and Nguyen T. Q., IEEE Transactions on Image Processing **16**, 1596 (2007).

[14] Liu Zhi-zhou, Dictionary Learning Based Super-Resolution Image Reconstruction, Xian University of Electronic Technology, 2011. (in Chinese)

[15] Liu Feng-lain, Sun Meng-yao and Cai Wen-na, Optoelectronics Letters **13**, 237 (2017).

[16] Chang Chih-chung and Lin Chih-jen, ACM Transactions on Intelligent Systems and Technology **2**, 27 (2011).

[17] Chao Dong, Chen Change Loy, Kaiming He and Xiaoou Tang, IEEE Transactions on Pattern Analysis and Machine Intelligence **38**, 295 (2016).