

Single Image Super-Resolution Using Sparse Representation on a K-NN Dictionary

Liu Ning^(✉) and Liang Shuang

School of Mathematical, Peking University, Science Building in Peking University,
No 5. Yiheyuan Road, Beijing, People's Republic of China
liuning19880928@gmail.com, liangshuang12@pku.edu.cn

Abstract. This paper presents a new method of generating a high-resolution image from a low-resolution image. We use a sparse representation based model for low-resolution image patches. We use large patches instead of small ones of existing methods. The size of the dictionary must be large to guarantee its completeness. For each patch in the low-resolution image, we search for similar patches in the dictionary to obtain a sub-dictionary. To define the similarity and to speed up the searching process, we present a Restricted Boltzmann Machine (RBM) based binary encoding method to get binary codes for the low-resolution patches, and use Hamming distance to describe the similarity. With the KNN dictionary of each low-resolution patch, we use a sparse representation method to get its high-resolution version. Experimental results illustrate that our method outperforms other methods.

Keywords: Super-resolution · Sparse representation · Restricted Boltzmann machine · Binary encoding

1 Introduction

Single image super-resolution (SR) refers to the task of generating a high-resolution (HR) image from a low-resolution (LR) image. It is useful in many applications, such as medical imaging, video surveillance and remote sensing imaging. Since many HR images can produce the same LR images, and SR is the reverse process, so SR is inherently ill-posed. Various SR methods have been proposed to stabilize the inversion of this ill-posed problem.

SR methods can be roughly classified into three categories: interpolation based, reconstruction based and learning based methods. Interpolation based methods (e.g., [1, 2]) are simple and fast, while they tend to smooth edges and cause blurring problems. Reconstruction based methods (e.g. [4]) apply various smoothness priors, and the enforced priors are typically designed to reduce edge artifacts. The performance of reconstruction based SR methods degrades rapidly when the number of available input images is small or the desired magnification factor is large. Learning based methods are the most popular ones. These methods either exploit internal similarities of the same image [5, 13], or reconstruct

the high frequency details from a training set of LR and HR patch pairs [6,7]. However, these methods use a unique dictionary for all the patches, and the patch size is relatively small, which may limit the SR result.

In this paper, we propose a novel SR algorithm which is based on sparse representation. Unlike existing sparse representation methods, ours employ an RBM based binary encoding method for nearest neighbor searching and uses KNN as the dictionary of each patch to conduct sparse representation.

The contributions of this paper are summarized as follows: (1) An RBM based binary encoding method is used to retrieve the KNN of each patch. (2) Using KNN as dictionary for each patch minimizes the reconstruction error in sparse representation, which leads to better SR result. (3) Large patches are employed instead of small ones as the atoms of dictionaries. Since a large patch contains more meaningful information than a small one, more high frequency details are retained in SR results.

2 Methods

In this section, we present our sparse representation based super-resolution method. In Sect. 2.1, we will introduce dictionary preparation related work. RBM based binary encoding method is introduced in Sect. 2.2. We use this method to encode each LR patch in D_l and the input LR image. We use the KNN of each LR patch retrieved by the binary codes to conduct sparse representation reconstruction, which is introduced in Sect. 2.3. The flowchart of our method is shown in Fig. 1.

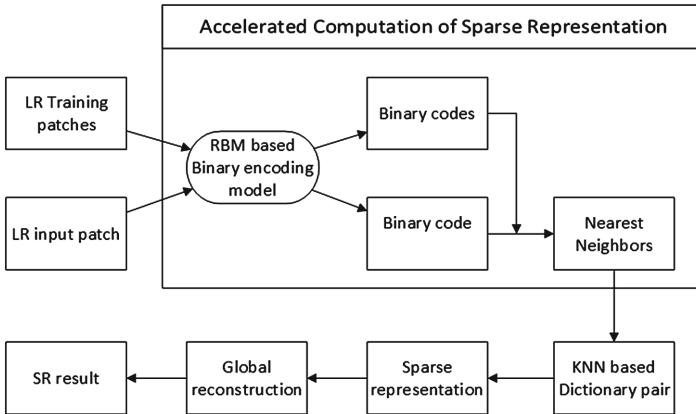


Fig. 1. The flowchart of our super-resolution algorithm.

2.1 Dictionary Preparation

Sparse representation based methods are widely used in learning-based super-resolution field [16,17]. An over-complete dictionary pair is important for these methods. Many SR methods focus on learning an efficiency dictionary [3,15]. To prepare the dictionary pair, we download HR images from Internet as our training set. For each HR training image, we generate the corresponding LR image by blurring and down-sampling. For LR images, we conduct bicubic interpolation to ensure that their sizes are the same as their HR images. We are interested in the patches with ample texture. For this purpose, we use Canny operator to get edges in the LR images, and extract the LR patches that center at edges. HR patches are extracted at the same position in the corresponding HR images.

We use large patches instead of small ones to combine the dictionary pair D_l and D_h , and set the patch size as 19×19 . To ensure the completeness of the dictionary with large patches, we set its size as 10^8 . Since the dictionaries are huge, (for instance, the LR dictionary D_h , with the feature dimension $d = 19 \times 19 = 361$, and patch number $N = 10^8$), the total integer number is over than 10^{10} , which needs about tens of Gigabytes. Obviously, computer memory can not load them all at once, so we divide them into 100 blocks respectively and handle them in block order.

2.2 RBM Based Binary Encoding Model

RBM based binary encoding model was proposed in [10]. Compact binary codes are learned via a stack of RBMs, each of which is trained by using the hidden activities of the previous RBM as its training data. This type of network is capable of capturing higher order correlations between different layers of the network. Each time a new RBM is added to this stack, the model has a better variational lower bound on the log probability of the data.

An RBM is a fully connected bipartite graph with one visible input (feature) layer $\mathbf{v} \in \mathbb{R}^{I \times 1}$ and one hidden (coding) layer $\mathbf{h} \in \mathbb{R}^{J \times 1}$, in which I and J represent the number of units of feature layer and coding layer respectively. Each unit has an input bias: represented as c_i for the feature layer and b_j for the coding layer. The layers are connected via undirected weights $\mathbf{W} \in \mathbb{R}^{I \times J}$. The activation probabilities of units are computed by sampling from the opposite layer with a sigmoid activation function, formulated as follows:

$$P(h_j|\mathbf{v}) = \text{sigmoid}(b_j + \sum_{i \in I} \omega_{ij}v_i), \quad (1)$$

$$P(v_i|\mathbf{h}) = \text{sigmoid}(c_i + \sum_{j \in J} \omega_{ij}h_j), \quad (2)$$

where v_i and h_j represent the state of units in the visible and hidden layers respectively.

The energy function of the RBM [10] is formulated as:

$$\begin{aligned} E(\mathbf{v}, \mathbf{h}) &= -\log P(\mathbf{v}, \mathbf{h}) \\ &= -\sum_{i \in I} \sum_{j \in J} v_i \omega_{ij} h_j - \sum_{i \in I} c_i v_i - \sum_{j \in J} b_j h_j \end{aligned} \quad (3)$$

The parameters $\mathbf{W} \in \mathbb{R}^{I \times J}$, $\mathbf{b} \in \mathbb{R}^{J \times 1}$ and $\mathbf{c} \in \mathbb{R}^{I \times 1}$ are updated with the objective of minimizing the energy function. The optimization method is called contrastive divergence (CD). A detailed introduction of the optimization process is given in [11]. With this optimization method, the update rules of parameters are formulated as:

$$\Delta w_{ij} = \epsilon(\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{recon}), \quad (4)$$

$$\Delta b_j = \epsilon(\langle h_j \rangle_{data} - \langle h_j \rangle_{recon}), \quad (5)$$

$$\Delta c_i = \epsilon(\langle v_i \rangle_{data} - \langle v_i \rangle_{recon}), \quad (6)$$

where ϵ is the learning rate, $\langle \cdot \rangle_{data}$ represents the sample result, and $\langle \cdot \rangle_{recon}$ represents the distribution of the model.

We implement this model with four layers: an input layer, two hidden layers and an output layer. The unit numbers used in each layer are 1444 – 1444 – 722 – 64. The input unit number is determined by the high-frequency feature number of an LR patch, which is with the patch size as 19 and 4 directional high-frequency features. The output unit number 64 is the length of binary codes. The unsupervised training phase is executed layer by layer from input to output. We use the trained parameters to generate binary codes for patches in D_l and LR image. To reduce noise, we use the probabilities rather than the stochastic binary states in the generate process. The binary codes are the binarization results of output values of output units.

Figure 2 shows an example of similar patches retrieved in the LR dictionary D_l . From this figure, we get that the proposed binary encoding method preserves the local characteristics of patches.

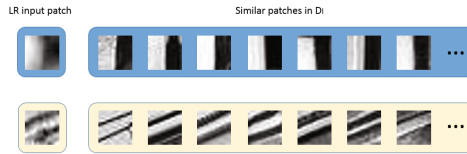


Fig. 2. Example of similar patches retrieved in the LR dictionary D_l .

2.3 Sparse Representation Based Super Resolution

In this section, we present our sparse representation based super-resolution method. Unlike existing sparse representation methods, ours uses KNN as dictionary for each LR patch. And it is realized by the RBM based binary encoding

method presented in the previous section. We first conduct binary encoding to D_l and the LR patch x_i , and search the nearest neighbors of x_i in D_l using the hamming distance of their binary codes. The nearest neighbors compose the sub-dictionary $D_l(i)$. Since the atoms in the dictionaries are similar to their corresponding patches, the dictionaries are efficient enough, so that we do not need any dictionary learning method.

For each LR patch x_i in the original LR image, we get its LR dictionary $D_l(i)$ by KNN and the corresponding HR dictionary $D_h(i)$. The patch x_i can be represented as a sparse linear combination in dictionary $D_l(i)$, and the problem of finding the sparsest representation of x_i can be formulated as

$$\min \|\alpha\|_0 \quad \text{s.t.} \quad \|FD_l(i)\alpha - Fx_i\|_2^2 \leq \epsilon, \quad (7)$$

where F is a high-frequency (HF) feature extraction operator, which is composed of the following high-pass operators:

$$f_1 = \begin{pmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}, \quad f_2 = \begin{pmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{pmatrix},$$

$$f_3 = \begin{pmatrix} 0 & +1 & +2 \\ -1 & 0 & +1 \\ -2 & -1 & 0 \end{pmatrix}, \quad f_4 = \begin{pmatrix} +2 & +1 & 0 \\ +1 & 0 & -1 \\ 0 & -1 & -2 \end{pmatrix}.$$

We concatenate the four different directional HF features as one vector as the HF feature extraction operator F .

Since the optimization problem (7) is NP-hard, we approximate the ℓ_0 norm with the ℓ_1 norm, which can be formulated as

$$\min \|\alpha\|_1 \quad \text{s.t.} \quad \|FD_l(i)\alpha - Fx_i\|_2^2 \leq \epsilon \quad (8)$$

Theoretical study results [8] prove that as long as the coefficient α is sufficiently sparse, the solutions of optimization problems (7) and (8) are equivalent. Lagrange multipliers offer an equivalent formulation to (8), which is formulated as

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \lambda \|\alpha\|_1 + \frac{1}{2} \|FD_l(i)\alpha - Fx_i\|_2^2, \quad (9)$$

where the parameter λ balances the fidelity of the approximation to x_i and sparsity of the solution. We recommend to set $\lambda \in [0.1, 0.3]$. In statistical literature, this problem is called LASSO [9].

Another important issue is to guarantee the compatibility of adjacent patches. We solve this problem by a one-pass algorithm similar to the method in [14]. The basic idea is that the reconstruction result $D_h(i)\alpha$ should be close to the previously computed adjacent HR patches in the overlapping areas. With this constraint, the optimization problem can be formulated as

$$\min \|\alpha\|_1 \quad \text{s.t.} \quad \begin{aligned} \|FD_l(i)\alpha - Fx_i\|_2^2 &\leq \epsilon_1, \\ \|PD_h(i)\alpha - \omega\|_2^2 &\leq \epsilon_2, \end{aligned} \quad (10)$$

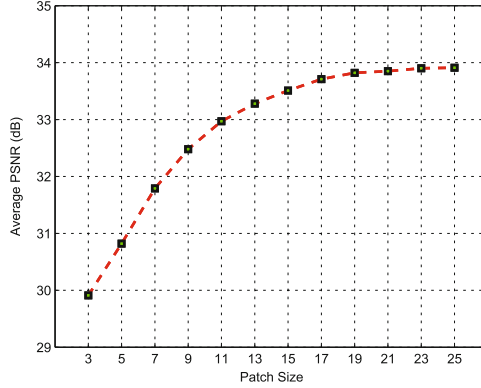


Fig. 3. Average PSNR results of ten benchmark images with different patch sizes.

where the matrix P extracts the overlap region between the previously reconstructed HR result and the current patch, and ω contains the previously reconstructed HR image on the overlap. The optimization problem (10) can be rewritten as

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \lambda \|\alpha\|_1 + \frac{1}{2} \|\hat{D}\alpha - \hat{x}\|_2^2, \quad (11)$$

where $\hat{D} = \begin{bmatrix} FD_l(i) \\ PD_h(i) \end{bmatrix}$ and $\hat{x} = \begin{bmatrix} Fx_i \\ \omega \end{bmatrix}$. After computing the optimal solution α^* of (11), the HR patch can be reconstructed as $x_i^* = D_h(i)\alpha^*$.

After patch-based reconstruction, a global reconstruction constraint is enforced to get final super-resolution result. The global constraint can be formulated as:

$$Y^* = \underset{Y}{\operatorname{argmin}} \|Y - Y_{sr}\| \quad \text{s.t.} \quad X = DBY, \quad (12)$$

where Y_{sr} is the SR result of patch-based sparse representation method, X represents the LR version of Y , B represents the blurring filter, and D represents the down-sampling operator.

Formula (12) is solved using back-projection method by an iterative strategy, which is formulated as

$$Y_{t+1} = Y_t + ((X - DBY_t) \uparrow s) * p, \quad (13)$$

where Y_t is the estimated high-resolution image after the t -th iteration, $\uparrow s$ represents up-sampling by the factor of s , and p is the “back-projection” filter. The details of this global reconstruction constraint is explained in [6].

3 Experiments

We apply our algorithm to benchmark images including Pepper, Lena, Foreman etc., and compare the results with those of the bicubic, NE [12], Freeman and



Fig. 4. SR results of image Lena. From top to bottom, left to right: ground truth, bicubic interpolation, NE based [12], Freeman and Liu [7], Yang et al. [6], Peleg and Elad [18], SRCNN [19] and proposed method. The magnification factors is 2.



Fig. 5. SR results of image Foreman. From top to bottom, left to right: ground truth, bicubic interpolation, NE based [12], Freeman and Liu [7], Yang et al. [6], Peleg and Elad [18], SRCNN [19] and proposed method. The magnification factors is 4.

Liu [7], Yang et al. [6], Peleg and Elad [18] and SRCNN [19] methods. We test with different magnification factors respectively, and use PSNR to measure and compare these methods.

3.1 Parameter Settings

There are three important parameters in this algorithm: patch size, dictionary size and sub-dictionary size (the KNN number of a patch). To investigate the impact of different patch sizes, we fix all other parameters of this algorithm and record the average PSNR indicators of ten benchmark images. Experimental results in Fig. 3 show that larger patches result in better SR results. While the calculation time increases rapidly with an increase of patch size. To balance the

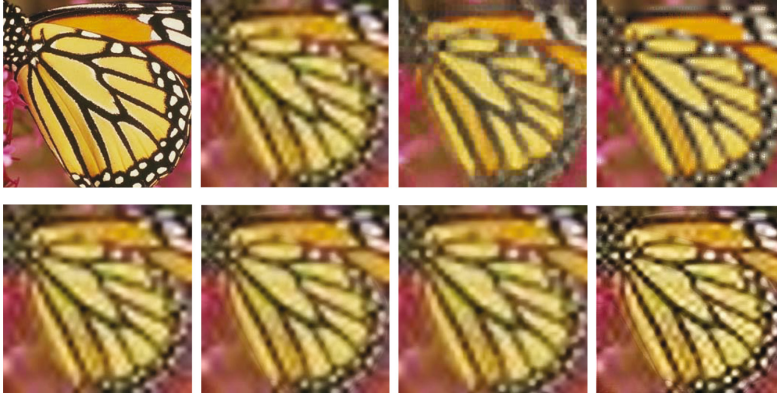


Fig. 6. SR results of image Butterfly. From top to bottom, left to right: ground truth, bicubic interpolation, NE based [12], Freeman and Liu [7], Yang et al. [6], Peleg and Elad [18], SRCNN [19] and proposed method. The magnification factors is 8.

efficiency and performance, we choose the patch size as 19 in our experiments (Fig. 4).

Theoretically, larger dictionary size and sub-dictionary size result in better SR results, while larger dictionary means more retrieve time and storage space.

Table 1. Experimental results of PSNR index on benchmark images

	MF	Lena	Girl	Baboon	Butterfly	Flowers	Foreman	Zebra	Pepper	Starfish	Average
Bicubic	2	31.35	36.86	24.36	26.51	29.03	31.39	29.25	32.04	28.75	29.95
	4	27.66	34.23	22.06	21.79	24.93	26.78	23.63	28.44	24.48	26.01
	8	24.72	32.18	21.01	17.68	21.96	23.10	19.57	25.24	21.49	22.99
NE [12]	2	31.17	36.90	24.28	26.30	29.14	31.52	29.09	32.15	28.69	29.92
	4	27.55	34.17	22.17	21.61	24.82	26.91	23.98	28.49	24.59	26.03
	8	24.61	32.01	21.21	17.83	21.82	23.18	19.79	25.19	21.61	23.03
Freeman and Liu [7]	2	31.48	36.87	24.28	26.49	29.18	31.54	29.17	32.72	28.72	30.05
	4	27.61	34.61	22.29	21.91	25.02	27.01	23.71	28.61	24.61	26.15
	8	24.92	32.42	21.51	18.17	22.01	23.72	19.92	25.64	21.94	23.36
SC [6]	2	32.80	36.58	24.42	27.33	29.61	31.91	29.50	32.34	29.21	30.41
	4	27.66	34.18	22.06	22.02	25.01	26.66	23.99	28.45	24.53	26.06
	8	24.72	32.16	21.01	17.98	22.18	23.09	19.84	25.26	21.56	23.07
Peleg and Elad [18]	2	32.19	36.25	23.83	26.83	29.71	31.26	29.61	32.73	29.31	30.19
	4	27.86	34.19	21.12	21.87	24.82	26.15	24.01	28.38	24.72	25.90
	8	24.71	32.01	19.89	17.10	22.19	22.98	19.71	25.01	21.46	22.79
SRCNN [19]	2	32.79	36.54	24.29	26.50	28.48	31.28	28.13	32.01	29.02	29.89
	4	27.12	33.61	22.70	22.20	24.23	27.16	22.62	27.78	23.67	25.68
	8	24.91	32.71	21.81	18.01	22.81	23.27	20.19	25.53	21.71	23.44
Our method	2	33.28	37.13	25.31	27.49	30.17	32.58	30.19	33.17	30.62	31.10
	4	28.11	34.89	22.98	22.38	25.19	27.28	24.20	28.78	25.01	26.54
	8	25.17	32.81	22.01	18.22	23.17	24.11	20.28	26.16	21.90	23.76

Limited by the capability of our computer, we set dictionary size as 10^8 , and sub-dictionary size as 2000.

3.2 Experimental Results and Comparison

We compare our algorithm with other algorithms under different magnification factors (MF). Figure 6 shows the SR results with different magnification factors. From the first row to the third row, the magnification factors are 2, 4 and 8 respectively. Table 1 shows the PSNR results tested on several benchmark images. From this figure and table, we get that our algorithm outperforms others. And the advantage is obvious when the magnification factor is large (Fig. 5).

4 Conclusion

Experimental results demonstrate the effectiveness of our SR method, especially when the magnification factor is large. We adopt a RBM based binary encoding model for super-resolution, and use the KNN of a patch to combine its sub-dictionary. Because the KNN dictionary can minimize the reconstruction error, and the large patch size contains more information, the super-resolution result is better than the state-of-art of super-resolution methods.

Further improvement can be realized through a property hashing method, which can accelerate the KNN searching process. On the other hand, a fast dictionary learning method may be added to make the dictionary more compact, which may further improve the super-resolution result.

References

1. Keys, R.: Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process.* **29**(6), 1153–1160 (1981)
2. Li, X., Orchard, M.T.: New edge-directed interpolation. *IEEE Trans. Image Process.* **10**(10), 1521–1527 (2001)
3. Juefei-Xu, F., Savvides, M.: Single face image super-resolution via solo dictionary learning. In: *IEEE International Conference on Image Processing*, pp. 2239–2243 (2015)
4. Morse, B., Schwartzwald, D.: Image magnification using level-set reconstruction. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 333–340 (2001)
5. Freedman, G., Fattal, R.: Image and video upscaling from local self-examples. *TOG* **30**(2), 12 (2011)
6. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution via sparse representation. *IEEE Trans. Image Process.* **19**(11), 2861–2873 (2010)
7. Freeman, W.T., Liu, C.: Markov random fields for super-resolution and texture synthesis (Chap. 10). In: Blake, A., Kohli, P., Rother, C. (eds.) *Advances in Markov Random Fields for Vision and Image Processing*. MIT Press, Cambridge (2011)
8. Donoho, D.L.: For most large underdetermined systems of linear equations, the minimal ℓ_1 -norm solution is also the sparsest solution. *Comm. Pure Appl. Math.* **59**(6) (2006)

9. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B* **58**(1), 267–288 (1996)
10. Hinton, G., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507 (2006)
11. Hinton, G.: Training products of experts by minimizing contrastive divergence. *Neural Comput.* **14**(8), 1771–1800 (2002)
12. Chang, H., Yeung, D.-Y., Xiong, Y.: Super-resolution through neighbor embedding. In: *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1 (2004)
13. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5197–5206 (2015)
14. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example based super-resolution. *IEEE Comput. Graphics Appl.* **22**(2), 56–65 (2002)
15. Xie, J., Chou, C.C., Feris, R., et al.: Single depth image super resolution and denoising via coupled dictionary learning with local constraints and shock filtering. In: *IEEE International Conference on Multimedia and Expo*, pp. 1–6 (2014)
16. Rao, A.B., Rao, J.V.: Super resolution of quality images through sparse representation. In: Satapathy, S.C., Avadahani, P.S., Udgata, S.K., Lakshminarayana, S. (eds.) *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of CSI - Volume II. AISC*, vol. 249, pp. 49–56. Springer, Heidelberg (2014)
17. Wang, Y.-H., Fu, P.: Sparse representation based medical MR image super-resolution. *Int. J. Adv. Comput. Technol.* **4**(19) (2012)
18. Peleg, T., Elad, M.: A statistical prediction model based on sparse representations for single image super-resolution. *IEEE Trans. Image Process.* **23**(6), 2569–2582 (2014)
19. Dong, C., Loy, C.C., He, K., et al.: Image super-resolution using deep convolutional networks (2015)