Adaptive Local Nonparametric Regression for Fast Single Image Super-Resolution

Yulun Zhang [1,3], Yongbing Zhang [1], Jian Zhang [2], Haoqian Wang [1], Xingzheng Wang [1], and Qionghai Dai [1,3]

[1] *Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China*
[2] *Institute of Digital Media, Peking University, Beijing 100871, China*
[3] *Department of Automation, Tsinghua University, Beijing 100084, China*

***Abstract*—We propose a fast single image super-resolution algorithm based on adaptive local nonparametric regression. Making use of dictionary learning and regression, we learn multiple projection matrices mapping low-resolution features to their corresponding high-resolution ones directly. Different from previous linear regression that needs some constant parameters, our method would not use extra parameters for regression. We use the mutual coherence between dictionary atom and low-resolution feature as a label to reconstruct more sophisticated high-resolution feature. As we use the same form of mutual coherence as labels in both training and testing phases, our method would lead to an adaptive local linear regression model. Moreover, we investigate the statistical property of the dictionary atoms from the training features. Utilizing the learned statistical priors, our method would not only obtain more useful dictionary atoms, but also further decrease the computational time. As shown in our experimental results, the proposed method yields high-quality super-resolution images quantitatively and visually against state-of-the-art methods.**

***Index Terms*—Adaptive local linear regression, dictionary learning, mutual coherence, projection matrix, super-resolution.**

I. Introduction

Single image super-resolution (SR), a classical and important computer vision problem, aims at generating a high-resolution (HR) image from its degraded low-resolution (LR) measurement, while minimizing visual artifacts as far as possible. It is still a challenging task, as we have to predict pixel intensities from very limited input data. To alleviate this ill-posed problem, most popular SR methods exploit additional information (e.g. exemplar priors [1]–[7], and internal similarities [8]–[10]) using machine learning (ML) techniques.

Anchored neighborhood regression (ANR) [2] and simple functions (SF) [11] are the two representative ML-based fast image SR algorithms. In ANR [2], Timofte et al. computed a set of anchored regressors mapping LR to HR features by grouping nearest neighbors within a learned LR-HR dictionary pair. Soon after, Timofte et al. proposed A+ [6], an improved variant of ANR, which used the full training samples instead of the dictionary pair to learn regressors. However, both ANR [2] and A+ [6] need to adjust a constant parameter in regression and also assume strictly equal encodings in the LR and HR domains. On the other hand, in SF [11], Yang et al. proposed to tackle such a complex regression by splitting the feature space and learning simple functions mapping LR to HR features directly for each subspaces. While SF [11] bounds its functions to the cluster features and treats each training feature equally in the regression, which is not true in the practical problem and also lacks flexibility for the input signal.

Motivated by the observation above, in this paper, we propose a simple yet efficient fast image SR model called adaptive local nonparametric regression (ALNR). In this model, we use the mutual coherence between dictionary atom and LR feature as a label in the training and reconstruction phases. Moreover, we investigate the statistical property of the dictionary atoms, which would be resorted for more efficient image SR.

Overall, the main contributions of this work can be summarized in three aspects. **First**, we directly train mappings from LR to HR domain with mutual coherence between atoms and features serving as labels for regression, avoiding tedious parameter tweaks. **Second**, we can obtain more sophisticated HR features with adaptive mutual coherence between dictionary atoms and the input LR features. **Third**, using the learned statistical priors from the training features, we resort the dictionary atoms to obtain far fewer atoms for regression and reconstruction.

II. Related Work

As the ML-based SR methods have been achieving superior and challenging results, we limit the related work to its recent three types of methods: dictionary learning, regression, and deep learning based methods.

Dictionary learning based SR approaches typically build upon sparse coding (SC) [12]. Yang et al. [13] used a SC formulation to learn LR and HR dictionaries by assuming that LR and HR features share the same reconstruction coefficients. This work was further improved by Zeyde et al. [1], who utilized PCA and orthogonal matching pursuit (OMP) [14] to reduce the dimensionality of LR features and solve sparse representation respectively. More sophisticated SC formulations were proposed recently, for example, a Bayesian method using a beta process prior was applied to learn the over-complete dictionaries in [3] and a statistical prediction model based on sparse representation was used in [4]. However, all these approaches are still quite slow in either training or reconstruction phases because of requiring a sparse encoding.

As mentioned in the introduction, a fast SR approach is to directly learn mapping relations from LR to HR domain based on regression. Timofte et al. learned regressors anchored to the dictionary atoms within a LR-HR dictionary pair [2] or the

IEEE VCIP'15, Dec. 13 - Dec. 16, 2015, Singapore

whole training features [6]. Yang et al. [11] clustered the input data space and learned simple regressors for each subspace. Following a similar approach, Dai et al. [7] jointly learned the separation into cells and regressors. Schulter *et al.* [15] upscaled image with SR forests taking advantage of random forests [16].

Very recently, deep learning shows its power in image SR by learning hierarchical representations of high dimensional data. Cui et al. [17] proposed a deep network cascade (DNC) to gradually upscale LR images layer by layer. Because independent optimization of the self-similarity search process and auto-encoder was needed in each layer of the cascade, DNC [17] failed to obtain an end-to-end solution. This problem was addressed by Dong et al. [5], who proposed a model named super-resolution convolutional neural network (SRCNN) learning structure with different mapping layers.

In our work, we combine dictionary learning and regression for fast image SR by learning adaptive nonparametric projection matrices mapping from LR to HR domain. Moreover, we investigate the statistical property of dictionary atoms for more efficient image SR. To the best of our knowledge, the image SR problem has not witnessed such an investigation.

III. Proposed Method

In this section, we first introduce the learning and reconstruction phases in our ALNR algorithm respectively. Finally, we present dictionary atoms resorting with the learned statistical priors. In the following discussion, we denote $\boldsymbol{I}_{L/H}, \boldsymbol{p}_{L/H}$, and $\boldsymbol{y}_{L/H}$ as an LR/HR image, patch, and feature, respectively.

A. Learning Nonparametric Projection Matrices

Let $\left\{\boldsymbol{y}^i_{Ls}, \boldsymbol{y}^i_{Hs}\right\}_{i=1}^{N_s}$ be the LR-HR feature sets extracted from LR-HR training images, and the dimensionality of these LR features have been reduced by a PCA projection matrix $\mathbf{P}$. We train an LR dictionary $\boldsymbol{D} = \{\boldsymbol{d}_k\}_{k=1}^{K}$ in the LR feature space. Then a larger training LR-HR feature set $\left\{\boldsymbol{y}^i_{Lw}, \boldsymbol{y}^i_{Hw}\right\}_{i=1}^{N_w}$ from the same training images are used for neighborhoods sampling, similar as A+ [6]. For each LR dictionary atom $\boldsymbol{d}_k$, we group its LR anchored neighborhood $\boldsymbol{N}_{L,k}$ by searching max nearest training LR features and the corresponding HR features are used to form HR neighborhood $\boldsymbol{N}_{H,k}$. The distance measure we utilize is the mutual coherence $c_{k,i}$ between dictionary atom $\boldsymbol{d}_k$ and LR feature $\boldsymbol{y}^i_{Lw}$ via

$$c_{k,i} = \left| {\boldsymbol{d}_k}^{\mathrm{T}} \boldsymbol{y}^i_{Lw} \right|, \tag{1}$$

where $\boldsymbol{d}_k$ is k-th atom in the learned dictionary $\boldsymbol{D} = \{\boldsymbol{d}_k\}_{k=1}^{K}$ and $\boldsymbol{y}^i_{Lw}$ is an LR training sample in $\left\{\boldsymbol{y}^i_{Lw}, \boldsymbol{y}^i_{Hw}\right\}_{i=1}^{N_w}$. Furthermore, we also use $c_{k,i}$ to compute adaptive values $\boldsymbol{L}_k = \left\{l^i_k\right\}_{i=1}^{max}$ for nonparametric regression via

$$l^i_k = \exp\left(c_{k,i}\right), \tag{2}$$

where l^i_k is used as a label indicating a more sophisticated mapping from $\boldsymbol{y}^i_{Lw}$ to $\boldsymbol{y}^i_{Hw}$. Finally, the regression problem becomes

$$\min_{\boldsymbol{F}_k} \left\| \boldsymbol{N}_{H,k} - \boldsymbol{F}_k \begin{bmatrix} \boldsymbol{N}_{L,k} \\ \boldsymbol{L}_k \end{bmatrix} \right\|_F^2. \tag{3}$$

This linear least-squares problem has a closed-form solution given by

$$\boldsymbol{F}_k = \boldsymbol{N}_{H,k} \boldsymbol{M}_k^{\mathrm{T}} \left(\boldsymbol{M}_k \boldsymbol{M}_k^{\mathrm{T}} \right)^{-1}, \tag{4}$$

where $\boldsymbol{M}_k = \begin{bmatrix} \boldsymbol{N}_{L,k} \\ \boldsymbol{L}_k \end{bmatrix}$ and the nonparametric projection matrice $\boldsymbol{F}_k$ can be computed offline for fast image SR.

B. Adaptive Local Nonparametric Regression for Image SR

After obtaining a set of mapping models $\{\boldsymbol{d}_k, \boldsymbol{F}_k\}_{k=1}^{K}$ in the learning phase, we can perform patch-wise ALNR based SR from LR input to the desired HR output directly.

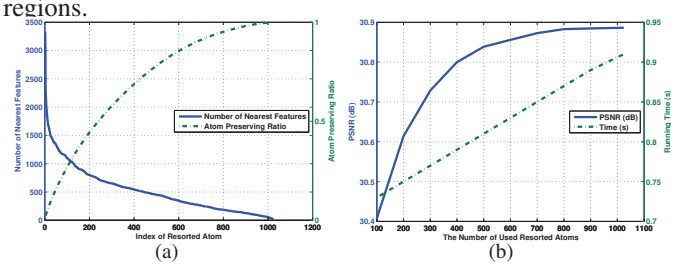
For an LR image $\boldsymbol{I}_L$, we first upscale it to the size of the desired HR image by Bicubic interpolation. The interpolated image $\boldsymbol{I}_B$ is then partitioned into a set of LR patches $\left\{\boldsymbol{p}^i_L\right\}_{i=1}^{N}$ and features $\left\{\boldsymbol{y}^i_L\right\}_{i=1}^{N}$. For each LR feature $\boldsymbol{y}^i_L$, we search its nearest atom $\boldsymbol{d}_k$ from the LR dictionary $\boldsymbol{D}$ with highest mutual coherence $c_{k,i}$. In order to recover a more sophisticated HR feature and also comply with adaptive regression in the training phase, we compute its labeled value l^i_k via (2). The desired HR feature $\boldsymbol{y}^i_H$ can be recovered via

$$\boldsymbol{y}^i_H = \boldsymbol{F}_k \begin{bmatrix} \boldsymbol{y}^i_L \\ l^i_k \end{bmatrix}, \tag{5}$$

from which we can see that there is an adaptive labeled value l^i_k for each input LR feature. Then, the HR patch $\boldsymbol{p}^i_H$ can be obtained by adding the HR feature to the corresponding interpolated LR image patch $\boldsymbol{p}^i_L$ by

$$\boldsymbol{p}^i_H = \boldsymbol{y}^i_H + \boldsymbol{p}^i_L. \tag{6}$$

Finally, we combine these HR patches to a whole HR image $\boldsymbol{I}_H$ by averaging the intensity values over the overlapping regions.



(a) (b)

Fig. 1. (a) Histogram of the number of nearest LR features to each resorted dictionary atom. Atom preserving ratio shows a statistical prior that most of training features relate to a relatively small number of popular atoms. (b) Average performance of ALNR on PSNR and running time values with different values of K' (the number of mapping models used for SR) on 10 testing images.

C. Dictionary Atoms Resorting

In this sub-section, we learn the statistical priors about dictionary atoms in the training phase and use far fewer mapping models for more efficient image SR. For each LR feature $\boldsymbol{y}^i_{Ls}$ in the LR training set $\left\{\boldsymbol{y}^i_{Ls}\right\}_{i=1}^{N_s}$, we search its nearest dictionary atom and also treat $\boldsymbol{y}^i_{Ls}$ as the nearest LR feature of the atom. Then we calculate $|\Omega_k|$, the number of nearest LR features, for each dictionary atom $\boldsymbol{d}_k$. Ω_k stands for the specified index set of LR features, whose nearest atom is $\boldsymbol{d}_k$. And $|\cdot|$ denotes the cardinality of a set.

With the learned statistical priors mentioned above, we conduct to resort the dictionary atoms to form a new dictionary $\boldsymbol{D}_R = \{\boldsymbol{d}_{R_i}\}_{i=1}^{K}$ via

$$|\Omega_{R_1}| \geq |\Omega_{R_2}| \geq \cdots \geq |\Omega_{R_K}|. \tag{7}$$

Fig. 1(a) shows the distribution of $|\Omega_{R_i}|$ to $\boldsymbol{d}_{R_i}$ in the sorted dictionary and the atom preserving ratio r, which is defined as

$$r = \left(\sum_{i=1}^{K'} |\Omega_{R_i}| \right) / N_s, \tag{8}$$

where K' denotes the number of atoms or projection matrices used in the SR phase. The atom preserving ratio r is a useful statistical prior, from which we know which atoms are more populous and make the proposed method faster with fewer mapping models.

The effectiveness of resorting dictionary atoms is briefly shown in Fig. 1(b), where we investigate how the number of used mapping models would affect the performance (e.g. PSNR and running time) of ALNR. More details and discussions are shown in Section IV-D.

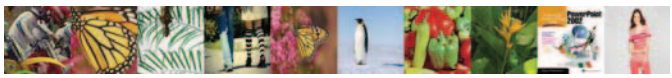


Fig. 2. 10 test images from left to righ: **bike**, **butterfly**, **leaves**, **legs**, **monarch**, **penguin**, **pepper**, **plants**, **ppt3**, and **stripes**.

IV. Experimental Results

In this section, we evaluate our direct regression-based method, ALNR, for fast single image SR. After outlining our experimental settings including details on datasets, evaluation metrics, we present the performance comparisons with other state-of-the-art methods. To provide more insights into ALNR, we also investigate the effectiveness of resorting dictionary atoms in more detail.

A. Experimental Settings

In our experiments, for a fair comparison, we use 91 images proposed by Yang et al. [13] as the training dataset. $3\times$ magnification is conducted on 10 commonly used LR natural images shown in Fig. 2, including humans, animals, and plants. Peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [18], and visual information fidelity (VIF) [19] are employed to evaluate the quality of SR results by different methods [1]–[7]. The implementations are all from the publicly available codes provided by the authors[1]. All of these evaluation metrics are performed between the luminance channel of the original HR and the reconstructed image.

We down-sample the original HR images to generate LR images for training and testing on the luminance channel by Bicubic interpolation. The parameters of ALNR are set as $K = 1024$, $max = 2048$, and N_w is 5 million. The LR and HR features are similar to Zeyde's methods in [1].

[1]The source code of **ALNR** will be available after this paper is published.

B. Performance

Table I shows numerical evaluations in terms of PSNR, SSIM, and VIF. Our method achieves the best results on average for each evaluation metric. An obvious average PSNR, SSIM, and VIF gains of our algorithm over the second best method A+ [6] are 0.13dB, 0.001, and 0.005 respectively. Such improvements are notable, since A+ [6] has been one of the best learning-based SR methods so far. This indicates that adaptive local nonparametric regression mapping LR to HR features directly contributes to improving image SR results. Table II shows the running time comparisons of numerous methods in the SR phase. We profile the running time of all the methods in a Matlab 2012a environment using the same machine (3.20 GHz Core(TM) i5-3470 with 16 Gb RAM). According to Table II, our ALNR ranks one of the fastest methods. Furthermore, ALNR can save more running time for fast image SR with fewer mapping projections, which would be detailed in Section IV-D.

TABLE I
Quantitative Comparisons for $3\times$ Magnification. For Each Image, There Are Three Rows: PSNR (dB), SSIM, and VIF. The Best Result for Each Image is **Highlighted**.

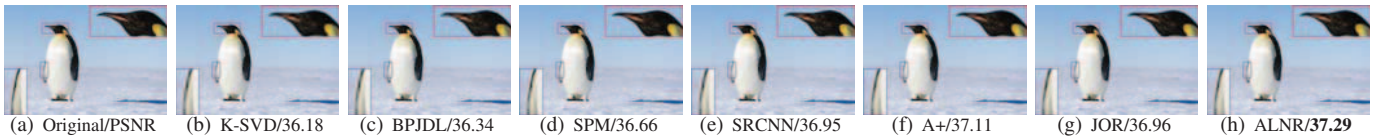
Images	K-SVD [1]	ANR [2]	BPJDL [3]	SPM [4]	SRCNN [5]	A+ [6]	JOR [7]	ALNR
bike	23.89	23.96	24.23	24.45	24.51	24.66	24.62	**24.74**
	0.765	0.768	0.781	0.788	0.787	0.797	0.796	**0.800**
	0.382	0.388	0.404	0.409	0.397	0.424	0.421	**0.427**
butterfly	25.94	25.85	26.52	26.78	**27.59**	27.26	27.08	27.46
	0.876	0.870	0.887	0.899	0.901	0.909	0.904	**0.912**
	0.466	0.468	0.500	0.507	0.516	0.529	0.522	**0.537**
leaves	25.50	25.38	25.96	26.09	26.49	26.47	26.54	**26.59**
	0.876	0.869	0.887	0.894	0.899	0.906	0.905	**0.909**
	0.489	0.490	0.519	0.514	0.500	0.542	0.541	**0.547**
legs	30.65	30.15	30.66	31.17	31.41	31.59	31.52	**31.73**
	0.891	0.885	0.891	0.893	0.895	0.901	0.901	**0.902**
	0.544	0.538	0.552	0.550	0.553	0.584	0.584	**0.589**
monarch	31.10	31.08	31.59	31.93	**32.39**	32.20	32.06	32.30
	0.938	0.937	0.941	0.944	0.945	0.947	0.946	**0.948**
	0.546	0.553	0.565	0.564	0.569	0.584	0.581	**0.587**
penguin	36.18	35.73	36.34	36.66	36.95	37.11	36.96	**37.29**
	0.962	0.961	0.961	0.962	0.963	0.966	0.965	**0.966**
	0.606	0.606	0.606	0.583	0.609	0.634	0.631	**0.637**
pepper	34.13	33.88	34.17	34.36	34.42	34.80	34.74	**34.85**
	0.888	0.887	0.888	0.888	0.889	0.894	0.893	**0.894**
	0.592	0.592	0.604	0.588	0.598	0.627	0.625	**0.629**
plants	32.82	32.83	33.13	33.14	33.52	33.81	33.77	**33.95**
	0.904	0.906	0.910	0.907	0.911	0.921	0.920	**0.923**
	0.583	0.593	0.605	0.590	0.599	0.635	0.634	**0.639**
ppt3	25.47	25.28	25.55	25.87	26.32	26.33	**26.43**	26.41
	0.920	0.913	0.921	0.926	0.929	0.939	0.938	**0.940**
	0.461	0.451	0.473	0.481	0.482	0.520	0.523	**0.528**
stripes	32.39	32.16	32.59	33.13	33.13	33.40	33.32	**33.54**
	0.950	0.948	0.951	0.954	0.954	0.959	0.959	**0.960**
	0.490	0.485	0.506	0.516	0.507	0.546	0.543	**0.552**
Average	29.81	29.63	30.07	30.36	30.67	30.76	30.70	**30.89**
	0.897	0.894	0.902	0.905	0.907	0.914	0.913	**0.915**
	0.516	0.516	0.534	0.530	0.533	0.562	0.560	**0.567**

TABLE II
Running Time (Seconds) Comparisons of Different Methods

Methods	K-SVD [1]	ANR [2]	BPJDL [3]	SPM [4]	SRCNN [5]	A+ [6]	JOR [7]	ALNR
Time	2.93	0.94	598.70	29.65	6.50	1.02	6.69	0.91

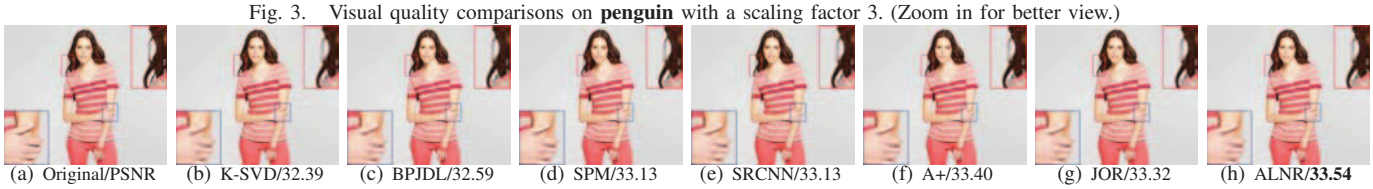
C. Visual Results

To further demonstrate the effectiveness of our proposed method, in Figs. 3 and 4, we compare our visual results with those of recent state-of-the-art methods. As we can see from these visual results, K-SVD [1] and BPJDL [3] would always generate obvious jaggy artifacts along the edges



(a) Original/PSNR (b) K-SVD/36.18 (c) BPJDL/36.34 (d) SPM/36.66 (e) SRCNN/36.95 (f) A+/37.11 (g) JOR/36.96 (h) ALNR/**37.29**

Fig. 3. Visual quality comparisons on **penguin** with a scaling factor 3. (Zoom in for better view.)



(a) Original/PSNR (b) K-SVD/32.39 (c) BPJDL/32.59 (d) SPM/33.13 (e) SRCNN/33.13 (f) A+/33.40 (g) JOR/33.32 (h) ALNR/**33.54**

Fig. 4. Visual quality comparisons on **stripes** with a scaling factor 3. (Zoom in for better view.)

and visually displeasing blurred textural details. Figs. 3(d) and 4(d) show the results obtained by SPM [4], which includes ringing artifacts around the over-sharped edges. Although SRCNN [5] achieves relatively high PSNR values, it generates results (e.g. Fig. 4(e)) possessing unpleasing artifacts along the dominant edges. A+ [6] and JOR [7] perform well in synthesizing fine details, but they can also generate blurred edges (e.g. Figs. 3(f) and 3(g)) and fail to recover more detailed textures (e.g. Figs. 4(f) and 4(g)). As can be observed in Figs. 3(h) and 4(h), our proposed ALNR would obtain high-quality results more faithful to the original HR images with sharper edges and finer details.

D. Investigation of Dictionary Atoms Resorting

As we have obtained the resorted dictionary $\boldsymbol{D}_R$, where an atom $\boldsymbol{d}_{R_i}$ with higher $|\Omega_{R_i}|$ is thought to be more populous. We present more results to investigate the influence of K' (the number of dictionary atoms used in SR) on the performance of ALNR on the 10 test images. In Table III, as K' becomes larger, the values of PSNR, SSIM, VIF, and running time would all increase, which demonstrates that it is reasonable for us to resorting the dictionary atoms for faster image SR. Moreover, when $K' = 400$, much smaller than 1024 used in A+ [6], our ALNR outperforms all of other competing methods. Even though we set K' as 100, the average performance of ALNR can also be better than many state-of-the-art methods (e.g. K-SVD [1], ANR [2], BPJDL [3], and SPM [4]). Using the resorted dictionary, we can use fewer atoms to compute projection matrices for image SR, saving running time and memory utilization.

TABLE III
AVERAGE PERFORMANCE OF ALNR ON PSNR (DB), SSIM, VIF, AND RUNNING TIME (S) WITH DIFFERENT VALUES OF K'.

K'	100	200	300	400	500	600	700	800	900	1024
PSNR	30.41	30.61	30.73	30.80	30.84	30.86	30.87	30.88	30.88	30.89
SSIM	0.908	0.911	0.913	0.914	0.915	0.915	0.915	0.915	0.915	0.915
VIF	0.547	0.556	0.561	0.564	0.566	0.566	0.567	0.567	0.567	0.567
Time	0.73	0.75	0.77	0.79	0.81	0.83	0.85	0.87	0.89	0.91

V. CONCLUSIONS

In this work, we propose an adaptive local nonparametric regression model for fast single image super-resolution. We use the mutual coherence between dictionary atoms and LR features as labeled values in the regression and SR phases, resulting an adaptive nonparametric regression model. Moreover, we learn statistical priors in the training phase and then resort dictionary atoms for high-quality image SR while saving much more running time and memory utilization. Extensive experimental results validate the effectiveness and robustness of our approach quantitatively and visually.

VI. ACKNOWLEDGEMENT

This work was partially supported by the National High-tech R&D Program of China (863 Program, 2015AA015901), the National Natural Science Foundation of China under Grant 61170195, U1201255, U1301257, and Guangdong Natural Science Foundation 2014A030313751.

REFERENCES

[1] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. 7th Int. Conf. Curves Surf.*, Jun. 2010, pp. 711–730.

[2] R. Timofte, V. De Smet, and L. V. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *ICCV*, Dec. 2013, pp. 1920–1927.

[3] L. He, H. Qi, and R. Zaretzki, "Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution," in *CVPR*, Jun. 2013, pp. 345–352.

[4] T. Peleg and M. Elad, "A statistical prediction model based on sparse representations for single image super-resolution." *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2569–2582, Jun. 2014.

[5] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *ECCV*, Sep. 2014, pp. 184–199.

[6] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *ACCV*, Nov. 2014.

[7] D. Dai, R. Timofte, and L. Van Gool, "Jointly optimized regressors for image super-resolution," in *Eurographics*, 2015.

[8] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *ICCV*, Sep./Oct. 2009, pp. 349–356.

[9] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *CVPR*, Jun. 2015, pp. 5197–5206.

[10] C. Fernandez-Granda and E. Candes, "Super-resolution via transform-invariant group-sparse regularization," in *ICCV*, Dec. 2013, pp. 3336–3343.

[11] C.-Y. Yang and M.-H. Yang, "Fast direct super-resolution by simple functions," in *ICCV*, Dec. 2013, pp. 561–568.

[12] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?" *Vision research*, vol. 37, no. 23, pp. 3311–3325, Dec. 1997.

[13] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *CVPR*, Jun. 2008, pp. 1–8.

[14] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[15] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *CVPR*, Jun. 2015, pp. 3791–3799.

[16] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[17] Z. Cui, H. Chang, S. Shan, B. Zhong, and X. Chen, "Deep network cascade for image super-resolution," in *ECCV*, Sep. 2014, pp. 49–64.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[19] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.