Multi-task Gaussian Process Regression-based Image Super Resolution

Xinwei Jiang
xinwei.jiang@ia.ac.cn
Jie Yang
yangjie@nlpr.ia.ac.cn
Lei Ma
lei.ma@ia.ac.cn
Yiping Yang
yiping.yang@ia.ac.cn

Institute of Automation
Chinese Academy of Sciences
Beijing, China

Abstract

This paper presents a novel framework for image super resolution (SR) based on the multi-task gaussian process (MTGP) regression. The core idea is to treat each pixel prediction using gaussian process regression as one single task and cast recovering a high resolution image patch as a multi-task learning problem. In contrast to prior gaussian process regression-based SR approaches, our algorithm induces the inter-task correlation for considering image structures. We demonstrate the efficiency and effectiveness of the proposed method by applying it to the classic image dataset and experimental results show our approach is competitive with even outperforms the related and state-of-the-art methods.

1 Introduction

Image super resolution (SR) aims at recovering the missing high frequency details from single image or multiple images. It has been studied over decades and many great methods are proposed to improve the SR performance gradually. Existing SR methods can be divided into three categories: interpolation-based, reconstruction-based and example learning-based.

Interpolation-based SR are usually implemented fast but they also suffer from some blurring and not being able to recover real details [2, 3, 12].

Reconstruction-based SR [10, 11, 13, 16] assumed the LR image is generated from HR image through blurring, down-sampling and noising. Based on this assumption, one pixel in LR image corresponds to multiple pixels in HR image, so that SR is an ill-posed problem. To alleviate the ill-posedness, various natural images prior are proposed and incorporated into the MAP optimization framework. However, its recovery performance drops badly when upscale factor is large.

Example learning-based SR methods utilize the LR-HR image pair to infer the missing high-frequency details in the LR image and achieve state-of-the-art performance. [4] first proposed to infer the HR image patch based external training LR/HR pairs and used the Markov Random Fields (MRF) to model the relationships between HR and LR patches, as well as between neighboring HR patches. [15] proposed to learn the HR/LR patch dictionary simultaneously with assumption that the sparse representation of LR patch and corresponding HR patch are the same with respect to the LR/HR dictionary. With the help of external dataset, example learning-based SR could recover more details than interpolation- and reconstruction-based methods, especially when zooming factor is large. The most significant step in example learning-based SR is to learn the relationship between LR-HR image explicitly or implicitly using the known LR-HR image pairs. These image pairs may be constructed from the image internal structures or external large database. Recently, in the field of example learning-based SR, more and more researchers resort to learn the LR-HR relationship directly, i.e. $y = f(x)$, where x is the input LR image feature, y is the targeted HR image and f is the mapping function that transforms the LR feature into HR image. To achieve the satisfactory SR result, we should find the unique proper mapping function for each test patch, however, this data-driven method undertakes high computational complexity [7, 8]. To alleviate the computation burden, many methods conduct clustering on training dataset and assume that patches in the same cluster share the same mapping function [14, 17]. In test stage, each LR patch first finds the most closest cluster and then uses the corresponding mapping function to recover the HR patch. This strategy reduces the computational complexity largely. [14] assume f is linear mapping, namely, $y = Fx$, where F is the regression matrix. [14] made use of large scale patch pairs (2.2 million) and cluster them into 4096 clusters. For each cluster, the desired F is computed by simple least square regression. Though [14] is simple to implement and takes short test time, it depends heavily on great number of training pairs and large number of cluster. There still exist some clusters lacking of enough training data, making the recovered HR image contain unsatisfactory details.

Instead of commonly used parametric models, non-parametric methods [6, 9], especially gaussian process regression (GPR)-related methods [5, 7, 8] begin to emerge in the SR field. He and Siu [5] firstly proposed to apply GPR to SR problem and predicted the center pixel using its eight neighbor pixels through GPR without any external database. Unfortunately, it suffers from long processing time and performance remains to be improved when zooming factor is over 3. Li *et al.* [8] proposed to learn multiple local GPR for mapping LR feature to HR image patch with the help of external database. Two methods, data-driven gaussian process regression (DDGPR) and prototype-based gaussian process regression (PGPR) are proposed. DDGPR attempted to learn specific GPR model for each test image patch. DDGPR first found nearest neighbors for each LR test patch and used the special LR-HR pairs to train GPR for this patch. Though achieved the promising performance, it took a long time to super-resolve an image. Instead of learning unique GPR for each test patch, PGPR assumed if two patches are similar, they share the same mapping function, so PGPR conducted clustering on training dataset and learnt GPR for each cluster. This method reduced the training time largely compared to DDGPR, however, at the cost of performance decreasing. Though some GPR-based SR methods are proposed, main obstacle for application to SR is its high computational complexity. To overcome the high computational cost, [7] proposed a semi-local GPR (SLGPR) framework for learn-based image enhancement and applied it to SR. SLGPR improved the sparse GP through inducing nearest neighbors as specific inducing inputs for the given test input and this reduced the training time and made it possible for large-scale GPR.

As for GPR, since it predicts a scalar value based on the observations, it is common to learn multiple GPR models together to predict an image patch, where each GPR corresponds to a pixel prediction. However, previous GPR-based SR methods simply learn all the GPR

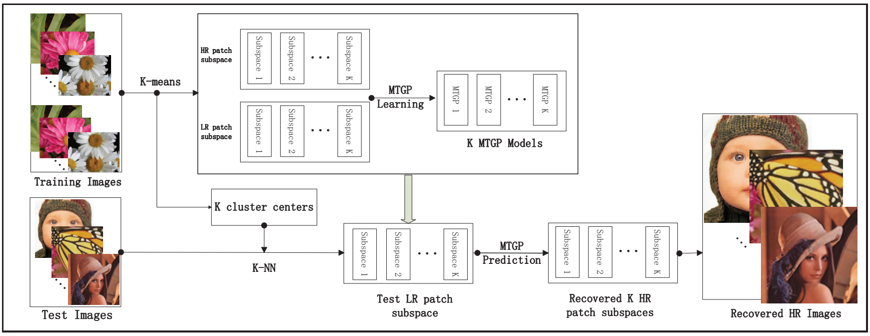


Figure 1: Flowchart of proposed Multi-task Gaussian Process Super Resolution.

models independently and ignore the correlation between them. On the other hand, each pixel prediction can be treated as a task, so that inferring a HR patch can be regarded as a multi-task problem. To model the correlation between GPR and make the predictive patch more accurate and natural, we propose a multi-task gaussian process framework for image super resolution (MTGPSR). To the best of our knowledge, it is the first time to apply the MTGP to the image super resolution problem. The key ingredient of MTGPSR is to take pixel prediction correlation into consideration and induce inter-task similarity to model this correlation, so that MTGPSR can be adaptive to image local structures.

The remainder of this paper is organized as follows: we give a brief overview of multi-task gaussian process prediction proposed in [1] in Section 2; Analysis of how SR problem corresponds to MTGP and details of proposed algorithm are presented in Section 3; Experimental validation and comparisons are provided in Section 4; Section 5 concludes the whole paper.

2 Multi-Task Gaussian Process

In this section, we give a brief introduction to multi-task gaussian process prediction model described in [1]. MTGP tries to solve the following problem: Given N distinct inputs $x_1,\dots,x_N$ we define the complete set of responses for M tasks as $\mathbf{y} = (y_{11},\dots,y_{N1},\dots,y_{12},\dots,y_{N2},\dots,y_{1M},\dots,y_{NM})^T$, where y_{ij} is the response for the j^{th} task on the i^{th} input x_i. We also denote the $N \times M$ matrix Y such that $\mathbf{y} = \mathrm{vec}Y$. Given a set of observations $\mathbf{y_o}$, which is a subset of $\mathbf{y}$, we wish to predict the unobserved values of $\mathbf{y_u}$ of some input points for some tasks. MTGP wishes to learn M related latent functions $\{f_l\}$ by placing a GP prior over $\{f_l\}$ and directly induce correlations between tasks. Assuming that the GPs have zero mean we define

$$\left\langle f_l(x) f_k(x') \right\rangle = K^f_{lk} k^x(x,x') \qquad (1)$$

$$y_{ij} \sim N(f_l(x_i), \sigma_j^2) \qquad (2)$$

where K^f is a positive semi-definite (**PSD**) matrix that specifies the inter-task similarities, k^x is a covariance function over inputs, and σ_j^2 is the noise variance for the j^{th} task. The key property of multi-task gaussian process model is the introduction of inter-task correlation matrix K^f, so that observations of one task can affect the predictions on another task. Inference in the MTGP model can be carried out by using the standard GP formulation for the predictive mean and variance. The predictive mean on a test point x_* for task j is obtained by

$$f_j(x_*) = (k_j^f \otimes k_*^x)^T \Sigma^{-1} y, where \Sigma = K^f \otimes K^x + D \otimes I \tag{3}$$

where $\otimes$ denotes the Kronecker product, k_j^f is the j^{th} column of K^f, k_*^x represents the vector consisting of covariances between the test point x_* and all the training points, K^x stands for the covariance matrix obtained by computing covariances between all the training points pairs, D is an $M \times M$ diagonal matrix with σ_j^2 in the $(j,j)^{th}$ position, and Σ is an $MN \times MN$ matrix. In learning stage, given the observations $\mathbf{y_o}$, we wish to optimize the hyper-parameters l_x of k^x and matrix K^f to marginal likelihood $p(y_o|X, l_x, K^f)$. [1] exploited the Kronecker strcture of covariance matrix and proposed to use an expectation-maximization (EM) algorithm to decouple the learning of l_x and K^f and optimize alternately. More details about hyper-parameters optimization can be found in [1].

3 MTGP for Single Image Super Resolution

In this section, we will present our proposed MTGP-based super resolution (MTGPSR) method in detail. First, we will show the MTGP can be exploited to solve the SR problem. Next, we give the algorithm details of MTGPSR.

3.1 How Super-resolution corresponds to MTGP

Example learning-based SR methods aim to recover the HR image by utilizing the external database of LR-HR image pair. The core ingredient is to learn the relationship $y = f(x)$, f is the targeted LR-HR mapping function. Example learning-based SR methods are mostly conducted patch-wise and algorithms consist of four steps which can be summarized as feature extraction, patch clustering, LR-HR regression and result aggregation. Different SR methods adopt different regression algorithms to learn the mapping function between LR patch feature and corresponding HR patch. In this subsection, we will illustrate that MTGP can be used for inferring the missing HR details.

In gaussian process regression, for a test input one GPR model generates a scalar output, however, targeted predictive HR image patch in SR requires the vector-value output. To break this limitation and apply GRP to image SR, previous GPR-based image SR [7, 8] simply utilize M GPR models altogether to predict a HR patch (size of the patch is $\sqrt{M} \times \sqrt{M}$), specifically, each GPR generates a pixel value and all these outputs form the desired patch. But these GPR predictions are independent, so that it ignores the correlation between these outputs and make the super-resolved image bear some unnatural details.

From another viewpoint, each pixel value prediction by one gaussian process regression can be regarded as one task, so that the whole patch prediction is a multi-task gaussian process prediction problem. In the context of image SR, assumed the size of target patch is $\sqrt{M} \times \sqrt{M}$, we define independent zero mean GP prior over all the latent mapping functions $\{f_i\}_{i=1}^M$, one for each pixel prediction

$$\langle f_i(x) f_j(x') \rangle = K_{ij}^f k^x(x,x') \tag{4}$$

With this prior, the GP prior over the observations $y_j(\cdot)$ is given by

$$\langle y_j(x) y_{j'}(x') \rangle = K_{jj'} k_j^x(x,x') \tag{5}$$

where $y_j(\cdot)$ is the observation of f_j. What's more, this model also constrains that all the mapping functions for a given cluster share the same covariance function. This completes the correspondence with the multi-task GP model. Based on the above analysis, we can apply MTGP to image super resolution.

Figure 1 gives the flowchart of our proposed MTGPSR method. In the training stage, we first construct HR/LR patch pairs by downsampling and then conduct K-means clustering on LR patch dataset. For each cluster, we learn one MTGP model to fit the training data. In the predicting stage, given an LR image, we first overlap sample the image getting test patch dataset and classify the data using the K-NN algorithm based on the cluster centers obtained from training stage. Next, each HR patch subspace corresponding to the LR patch subspace are recovered through learned MTGP regression. Finally, all the predicted patches are reconstructed into a HR image using average weighting scheme.

3.2 Proposed Algorithm

Based on the processing pipeline analyzed in the above subsection, if we have C patch clusters, then we learn C MTGP models, one for each cluster. Then learned MTGP model is applied to the test LR patch to predict the targeted HR patch. Inference and hyper-parameter learning can be done separately for each cluster. What's more, the covariance function plays the center role in the gaussian process and we adopt the squared exponential covariance function

$$k(x,y) = exp\left(-\frac{\|x-y\|^2}{l^2}\right) \tag{6}$$

where l is the hyper-parameter that defines the characteristic length scale.

In learning stage, the MTGP hyper-parameters are optimized as follows: Let f be the vector of function values corresponding to y, and similarity for F for Y. Further, let $y_{.j}$ denote the vector $(y_{1j},...,y_{Nj})^T$ and similarly for $f_{.j}$. Given the missing data, which in this case is f, the complete-data log-likelihood is

$$\begin{aligned} L_{comp} = -\frac{N}{2}log|K^f| - \frac{M}{2}log|K^x| - \frac{1}{2}tr\left[(K^f)^{-1}F^T(K^x)^{-1}F\right] - \frac{N}{2}\sum_{j=1}^{M} log\sigma_l^2 \\ -\frac{1}{2}tr[(Y-F)D^{-1}(Y-F)^T] - \frac{MN}{2}log2\pi \end{aligned} \tag{7}$$

from which we have following updates:

$$\hat{l}_x = arg\min_{l_x}\left(Nlog|\langle F^T(K^x(l_x))^{-1}F\rangle| + Mlog|K^x(l_x)|\right) \tag{8}$$

$$\hat{K}^f = N^{-1}\langle F^T(K^x(l_x))^{-1}F\rangle \tag{9}$$

$$\hat{\sigma}_j^2 = N^{-1}\langle(y_{.j} - f_{.j})^T(y_{.j} - f_{.j})\rangle \tag{10}$$

where the expectations $\langle\cdot\rangle$ are taken with respect to $p(f|y_o,l_x,K^f)$, and $\hat{\cdot}$ denotes the updated parameters.

Assumed we have obtained the hyper-parameters of all the C MTGP models, in inference stage, i.e. super-resolving the LR patch, the predictive HR patch corresponding to the input patch could be inferred according to equation. (3). The proposed MTGPSR algorithm is summarized in **Algorithm 1**.

Algorithm 1 Multi-task Gaussian Process-based Single Image Super Resolution

```
Input:
    training pairs $\{(x_i,y_i)\}_{i=1}^{N_{train}}$, testing LR patch set $\{y_j\}_{j=1}^{N_{test}}$
Output:
    Estimated HR image.
Conduct $K$-means on $\{x_i\}_{i=1}^{N}$ and get cluster centers $\{c_k\}_{k=1}^{K}$;
for $i$= 1 to $C$ do
    optimize hyper-parameters of $i$th MTGP model using equation. from (8) to (10);
end for
for $j$ = 1 to $N_{test}$ do
    assign cluster label $c$ of $j$th patch $y_j$ using $k$-nn;
    predict the corresponding HR patch using $c$th MTGP model according to equation. (3);
end for
convert the estimated HR patch set to the HR image through average aggregation;
return Estimated HR image
```

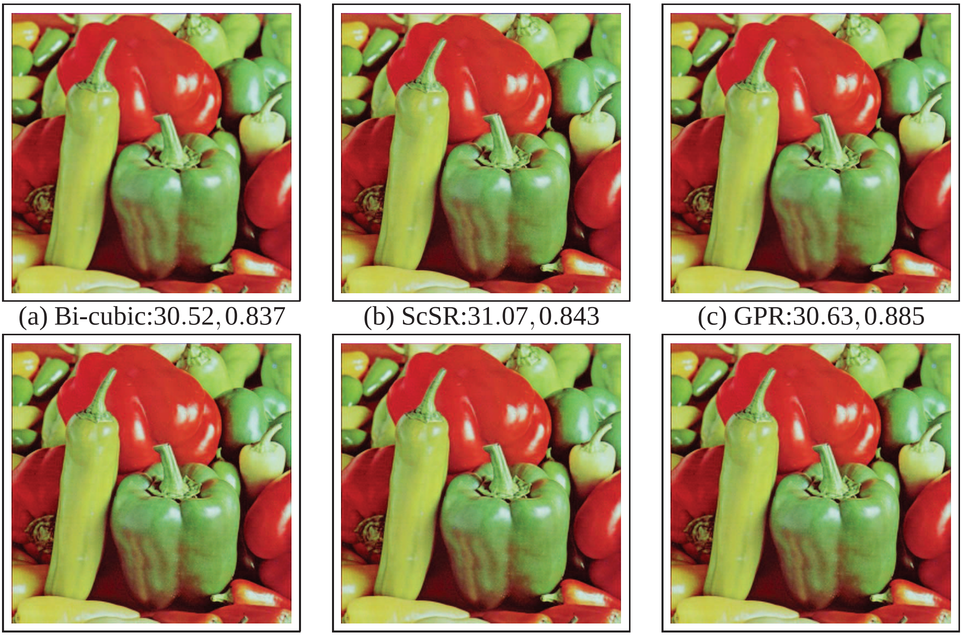
4 Experiment

In this section, to validate the effectiveness of proposed MTGPSR, we compare our method with following SR methods: bi-cubic interpolation and sparse coding-based SR [15](ScSR) as baselines and three gaussian process regression-based methods: [5](GPR), [8](PGPR), [7](SLGPR). ScSR and GPR are implemented using the original code provided by their authors respectively. PGPR and SLGPR are implemented by ourself according to the experiment description presented in the original paper. We evaluate the SR performance in terms of peak-signal-noise-ratio(PSNR) and structural similarity(SSIM). The HR training images are provided by [15]. The classic **Set5** and **Set14** datasets are used as the test images. The parameters in our experiment are set as follows: the patch size is 3, the number of cluster C is 500, total number of training pairs is 20000. Note that although increasing patch size can improve the SR performance, but training time of MTGPSR will be prohibitively long, so we set it 3.

4.1 Analysis

Figure 2 and 3 show the representative SR results and figure 4 gives the comparison results of $\times 2$ super-resolution on classic test images in terms of PSNR and SSIM. From the above figures, we can see that MTGPSR achieves better performance on some images both in PSNR and SSIM. Furthermore, for all the test images, we can see that though PSNRs of our MTGPSR are lower than that of SLGPR [7], but are higher than other comparisons. On other other hand, our MTGPSR outperforms all the related methods in terms of SSIM. It means the recovered HR image by our approach can be more pleasant to human subjectiveness and achieve the comparable numerical results. The above results prove the effectiveness of our proposed MTGPSR framework and show that it can be more adaptive to image structures.

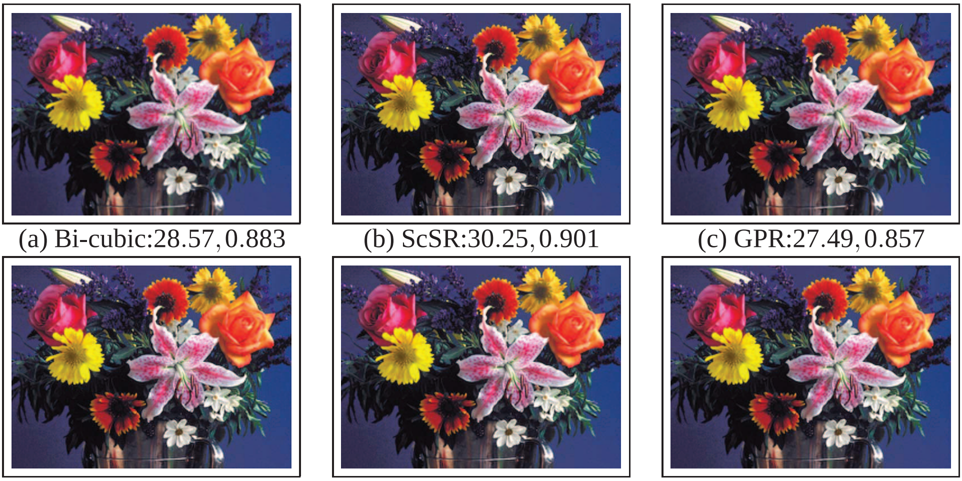
As for PSNRs of MTGPSR are lower than SLGPR, we try to empirically analyze the result in terms of patch size. For patch-based SR methods, the patch size has a great influence on the final SR performance and proper patch size usually improves the result. In our experiment, we set patch size 3 during the training and test stage. The reason we adopt



(a) Bi-cubic:30.52, 0.837 (b) ScSR:31.07, 0.843 (c) GPR:30.63, 0.885

(d) PGPR:32.19, 0.893 (e) SLGPR:33.66, 0.911 (f) Ours:**34.29**, **0.914**

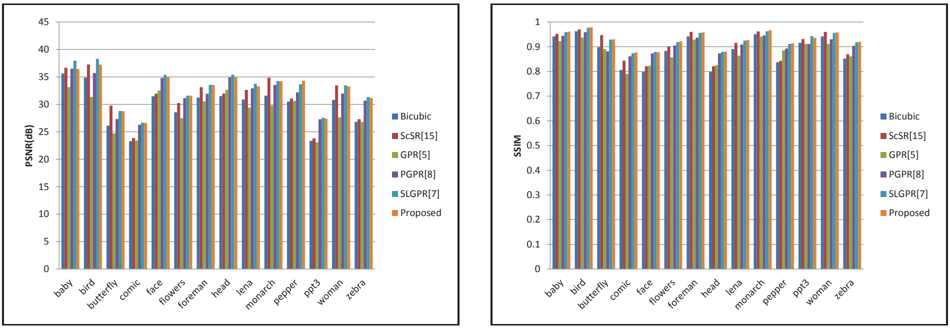
Figure 2: Comparison of SR results (×2) on *Pepper* image. (a) Bi-cubic interpolation. (b) ScSR method [15]. (c) GPR method [5]. (d) PGPR method [8]. (e) SLGPR method [7]. (f) The proposed method. The two numbers below each figure are PSNR and SSIM respectively.



(a) Bi-cubic:28.57, 0.883 (b) ScSR:30.25, 0.901 (c) GPR:27.49, 0.857

(d) PGPR:31.13, 0.905 (e) SLGPR:**31.59**,0.919 (f) Ours:**31.59**, **0.922**

Figure 3: Comparison of SR results (×2) on *flowers* image. (a) Bi-cubic interpolation. (b) ScSR method [15]. (c) GPR method [5]. (d) PGPR method [8]. (e) SLGPR method [7]. (f) The proposed method. The two numbers below each figure are PSNR and SSIM respectively.



(a) (b)

Figure 4: Comparison of ×2 SR results on **Set5** and **Set14** datasets. (a) PSNR (b) SSIM

the patch of size as 3 rather than larger size like 7 or 9, which is usually taken in the SR experiment, is that we observe empirically bigger size patch even 7 would make the training time prohibitively long (more than one week using CPU). Meanwhile, other comparative methods adapt the patch as follows: ScSR:5, SLGPR: 7, LGPR:17. The former two are set according to the original algorithm, however, patch size in the LGPR is taken 17, not original setting 5 because in our implementation, small size patch would make the recovered HR image contain obviously unpleasant artifacts, so we increase the patch size gradually and find 17 is appropriate. So, even if patch size of our proposed MTGPSR is smaller than all the comparisons, as analyzed above, MTGPSR still achieves comparable PSNR and prevails all the comparisons in terms of SSIM.

5 Conclusion

In this paper we propose a super resolution framework based on the multi-task gaussian process regression and study how the models are optimized and inferred effectively. The proposed MTGPSR framework makes the pixel prediction correlation into consideration based on the image local structure. Experimental results show that the proposed algorithm achieves the comparative performance and makes the super-resolved image more accurate and natural. In future work, we will focus on reducing training time of MTGP and apply it to other image processing tasks, such as denoising and enhancement.

6 Acknowledgment

We thank Christopher K. I. Williams for suggesting the related paper [1]. This work is supported by the National Natural Science Foundation of China under Grant 61203239.

References

[1] Edwin V Bonilla, Kian M. Chai, and Christopher Williams. Multi-task gaussian process prediction. In *Advances in Neural Information Processing Systems 20*, pages 153–160. 2008.

[2] Raanan Fattal. Image upsampling via imposed edge statistics. In *ACM Transactions on Graphics (TOG)*, volume 26, page 95. ACM, 2007.

[3] Gilad Freedman and Raanan Fattal. Image and video upscaling from local self-examples. *ACM Transactions on Graphics (TOG)*, 30(2):12, 2011.

[4] William T Freeman, Thouis R Jones, and Egon C Pasztor. Example-based super-resolution. *Computer Graphics and Applications, IEEE*, 22(2):56–65, 2002.

[5] He He and Wan-Chi Siu. Single image super-resolution using gaussian process regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 449–456, 2011.

[6] Li He, Hairong Qi, and Russell Zaretzki. Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 345–352, 2013.

[7] Younghee Kwon, Kwang In Kim, Jin Hyung Kim, and Christian Theobalt. Efficient learning-based image enhancement: Application to super-resolution and compression artifact removal. In *Proceedings of the British Machine Vision Conference(BMVC)*, pages 1–12, 2012.

[8] Jianmin Li, Yanyun Qu, Cuihua Li, Yuan Xie, Yang Wu, and Jianping Fan. Learning local gaussian process regression for image super-resolution. *Neurocomputing*, 154(0): 284–295, 2015.

[9] Gungor Polatkan, David Blei, Ingrid Daubechies, Lawrence Carin, and Mingyuan Zhou. A bayesian nonparametric approach to image super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):346–358, 2014.

[10] Matan Protter, Michael Elad, Hiroyuki Takeda, and Peyman Milanfar. Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on Image Processing*, 18(1):36–51, 2009.

[11] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2008.

[12] Lingfeng Wang, Huaiyu Wu, and Chunhong Pan. Fast image upsampling via the displacement field. *IEEE Transactions on Image Processing*, 23(12):5123–5135, Dec 2014.

[13] Hongteng Xu, Guangtao Zhai, and Xiaokang Yang. Single image super-resolution with detail enhancement based on local fractal analysis of gradient. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(10):1740–1754, Oct 2013.

[14] Chih-Yuan Yang and Ming-Hsuan Yang. Fast direct super-resolution by simple functions. In *IEEE International Conference on Computer Vision (ICCV)*, pages 561–568, Dec 2013.

[15] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873, 2010.

[16] Kaibing Zhang, Xinbo Gao, Dacheng Tao, and Xuelong Li. Single image super-resolution with non-local means and steering kernel regression. *IEEE Transactions on Image Processing*, 21(11):4544–4556, Nov 2012.

[17] Kaibing Zhang, Dacheng Tao, Xinbo Gao, Xuelong Li, and Zenggang Xiong. Learning multiple linear mappings for efficient single image super-resolution. *IEEE Transactions on Image Processing*, 24(3):846–861, March 2015.