Open access • Journal Article • DOI:10.1109/TMM.2012.2232646

A Self-Learning Approach to Single Image Super-Resolution — Source link

Min-Chun Yang, Yu-Chiang Frank Wang

Institutions: National Taiwan University, Center for Information Technology

Published on: 01 Apr 2013 - IEEE Transactions on Multimedia (IEEE)

Topics: Bicubic interpolation, Sparse approximation and Support vector machine

A Self-Learning Approach to Single Image Super-Resolution

Min-Chun Yang and Yu-Chiang Frank Wang, *Member, IEEE*

***Abstract*—Learning-based approaches for image super-resolution (SR) have attracted the attention from researchers in the past few years. In this paper, we present a novel self-learning approach for SR. In our proposed framework, we advance support vector regression (SVR) with image sparse representation, which offers excellent generalization in modeling the relationship between images and their associated SR versions. Unlike most prior SR methods, our proposed framework does not require the collection of training low and high-resolution image data in advance, and we do not assume the reoccurrence (or self-similarity) of image patches within an image or across image scales. With theoretical supports of Bayes decision theory, we verify that our SR framework learns and selects the optimal SVR model when producing an SR image, which results in the minimum SR reconstruction error. We evaluate our method on a variety of images, and obtain very promising SR results. In most cases, our method quantitatively and qualitatively outperforms bicubic interpolation and state-of-the-art learning-based SR approaches.**

***Index Terms*—Self-learning, sparse representation, super-resolution, support vector regression.**

I. Introduction

SUPER-RESOLUTION (SR) has been an active research topic in the areas of image processing and computer vision. It is a process to produce a high-resolution (HR) image from one or several low-resolution (LR) images. Conventional methods are based on the reconstruction of multiple LR images, and they approach SR as solving an inverse problem, i.e., they recover the HR image as a linear operation of multiple LR patches. Recently, learning-based SR approaches which focus on modeling the relationship between training low and high-resolution images have also attracted researchers, while the existence of the aforementioned relationship is typically seen in natural images [1], [2]. However, the difficulty of learning-based SR methods lies on the selection of proper training data and proper learning models for SR from an unseen target image.

Manuscript received November 16, 2011; revised February 22, 2012; accepted July 28, 2012. Date of publication December 11, 2012; date of current version March 13, 2013. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Feng Wu.

M.-C. Yang is with the Department of Computer Science and Information Engineering, National Taiwan University, and the Research Center for Information Technology Innovation, Academia Sinica, Taiwan (e-mail: d96922009@ntu.edu.tw).

Y.-C. F. Wang is with the Research Center for Information Technology Innovation and the Institute of Information Science, Academia Sinica, Taipei, Taiwan (e-mail: ycwang@citi.sinica.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2012.2232646

In machine learning, support vector regression (SVR) [3] is considered as an extension of support vector machine (SVM), which exhibits excellent generalization ability in predicting functional outputs without any prior knowledge or assumption on the training data (e.g., data distribution, etc.). SVR is capable of fitting data via either linear or nonlinear mapping, and the use of SVR has been applied in applications of data mining, bioinformatics, financial forecasting, etc. Previously, SVR has been shown to address SR problems in [1], [4]; however, these SVR-based SR approaches require the collection of training low and high-resolution image pairs in advance, and this might limit their practical uses. In this paper, we propose a self-learning framework for SR. We not only present quantitative and qualitative SR results to support our method, we will also provide theoretical backgrounds to verify the effectiveness of our learning framework.

The remainder of this paper is organized as follows. Prior SR works are discussed in Section II. Section III details the proposed self-learning method, including theoretical backgrounds of Bayesian decision theory for our SVR model selection. Section IV provides empirical results on a variety of images with different magnification factors, including comparisons with several SR methods. Finally, Section V concludes this paper.

II. Related Work

A. Reconstruction-Based SR

Typically, reconstruction-based SR algorithms require image patches from one or several images (frames) when synthesizing the SR output. This is achieved by registration and alignment of multiple LR image patches of the same scene with sub-pixel level accuracy [5]–[7]. For single-image reconstruction-based SR methods, one needs to exploit self similarity of patches within the target LR image. With this property, one can thus synthesize each patch of the SR image by similar patches in the LR version. However, reconstruction-based methods are known to suffer from ill-conditioned image registration and inappropriate blurring operator assumptions (due to an insufficient number of LR images) [8]. Moreover, when an image does not provide sufficient patch self-similarity, single-image reconstruction based methods are not able to produce satisfying SR results [9]. Although some regularization based approaches [5], [7], [10] were proposed to alleviate the above problems, their SR results will still be degraded if only a limited number of low-resolution images/patches are available or if a larger image magnification factor is needed. According to [8], [11], the magnification factor of reconstruction-based SR approaches is limited to be less than 2 for practical applications. A recent

approach proposed in [12] alleviates this limitation by learning image prior models via kernel principal component analysis from multiple image frames.

Since single-image SR does not require multiple LR images as inputs, it attracts the interest from researchers and engineers due to practical applications. As discussed above, methods assuming the existence of image patch self-similarity need to search for similar patches from an input image when synthesizing the SR output. However, the assumption of self-similarity might not always hold, and the associated SR performance varies with the similarity between different categories of image patches. The nonlocal means (NLM) method [13] is one of the representatives which advocates such a property in image related applications.

B. Learning-Based SR

In the past few years, much attention has been directed to learning (or example) based SR methods (e.g., [1], [14], [15]), which can be considered as single image SR approaches utilizing the information learned/observed from training image data. With the aid of training data consisting of low and high resolution image pairs, learning-based methods focus on modeling the relationship between the images with different resolutions (by observing priors of specific image or context categories). For example, Chang *et al.* [15] applied the technique of locally linear embedding (LLE) for SR purposes. They collected a training data set with multiple low and high-resolution image pairs. For each patch in an input LR image, they proposed to search for similar patches from LR training images, and they used the corresponding training HR patches to linearly reconstruct the SR output (using the weights determined by LLE). Ni *et al.* [1] proposed to use support vector regression (SVR) to fit LR image patches and the pixel value of the corresponding HR images in spatial and DCT domains. It is not surprisingly that, however, the performance of typical learning-based methods varies significantly on the training data collected. As a result, in order to achieve better SR results, one needs to carefully/manually select the training data. In such cases, the computation complexity of training and difficulty of training data selection should be both taken into consideration.

Recently, Glasner *et al.* [2] proposed to integrate both classical and example-based SR approaches for single image SR. Instead of collecting training image data beforehand, they searched for similar image patches across multiple down-scaled versions of the image of interest. It is worth noting that this single image SR method advocates the reoccurrence of similar patches across scales in natural images, so that their approach simply downgrades the resolution of the input image and perform example-based SR. In other words, once similar patches are found in different scaled versions, classical SR methods such as [7], [16], [17] can be applied to synthesize the SR output. Although very promising SR examples were shown in [2], there is no guarantee that self-similarity always exists within or across image scales, and thus this prohibits the generalization of their SR framework for practical problems. Moreover, it is not clear what is the preferable magnification factor when applying their approach (SR images with different magnification factors were presented in [2]).

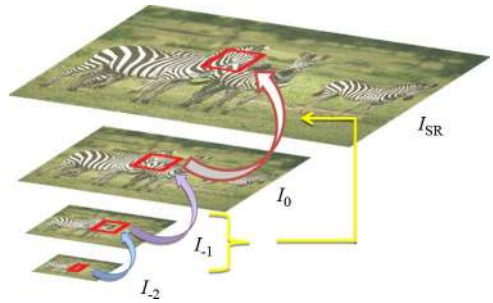

Fig. 1. Overview of our learning-based single image SR approach. Note that I_0 represents the original input low-resolution image, and I_{-1}, I_{-2}, etc. are the down-scaled versions of I_0. I_{SR} is the SR result.

C. Sparse Representation for SR

Originally applied to signal recovery, sparse coding [18] has shown its success in image related applications such as image denoising [19], and it was first applied to SR by Yang *et al.* [20], [21]. They considered the image patch from HR images as a sparse representation with respect to an over-complete dictionary composed of signal-atoms. They suggested that, under mild conditions, the sparse representation of high-resolution images can be recovered from the low-resolution image patches [20], [21]. They used a small set of randomly chosen image patches for training, and implied that their SR method only applies to images with similar statistical nature. Kim and Kwon [22], [23] proposed an example-based single image SR for learning the mapping function between the low and high-resolution images by using sparse regression and natural image priors. However, blurring and ringing effects near the edges exist in their SR results, and additional post-processing techniques are still needed to alleviate this problem. Recently, Yang *et al.* [24] extended the framework of [2]. Based on the assumption of image patch self-similarity, they concatenated high and low-resolution image pairs from the image pyramid and jointly learned their sparse representation. When super-resolve an LR input patch, they searched for similar patches from the image pyramid and use the associated sparse representation (the HR part) to predict its final SR version.

D. Our Contributions

The overview of our single image SR approach is shown in Fig. 1. Image I_0 is the LR input image, and I_{SR} is the SR output. Instead of searching for similar patches from training image data or from different down-scaled versions of the original image (e.g., [2], [15]), we aim to learn the relationships between image patches across scales. Once these relationships are observed, we can apply the proper model to synthesize the final SR image from I_0. It is worth noting that, instead of working directly at pixel, etc. domains as [1], [15] did, we apply image sparse representation as a robust and effective feature for learning ([20], [21] also did this). Furthermore, our method does not require collection of training data, and we do not require the assumption of image patch self-similarity in our framework.

In our method, there are two major issues to be addressed. The first task is the modeling of relationships between image patches from different image scales. Once these models are observed, the following task (and probably the most challenging one) is

TABLE I
THE DESCRIPTIONS OF THE SYMBOLS USED IN THIS PAPER

SYMBOL	DESCRIPTION
I_{SR}	The high-resolution SR image to be synthesized
I_i	The down-sampled LR image at the i-th scale. Note that I_0 is the input LR image.
S_i	The up-sampled version of I_{i-1} at the i-th scale (i.e., a blurred version of I_i). Note that S' is the up-sampled version of I_0.
E_i	The difference (error) image $E_i = I_i - S_i$ at the i-th scale
I_SVR_i	The SVR model observing the relationship between S_i and I_i at the i-th scale
E_SVR_i	The SVR model observing the relationship between S_i and E_i at the i-th scale
x	The image patch of interest
α	The sparse coefficient vector of x
D_i	The dictionary for image sparse representation at the i-th scale
y	The pixel value at the same location as the center of x to be observed/predicted
b	The offset of the SVR model
$\phi(\alpha_i)$	The sparse image representation in the transformed space
w	The norm vector of the non-linear SVR mapping function
C	The tradeoff between the complexity of the learned SVR model and its training error
ξ, ξ^*, ϵ	The upper/lower bounds of the training errors, subject to a margin ϵ
$P(x\|I\_SVR_i, D_i)$	The category-conditional probability function (i.e., the likelihood) of x given models observed at the i-th scale
$P(I\_SVR_i)$	The prior probability of I_SVR_i
$P(I\_SVR_i, D_i\|x)$	The posterior probability of I_SVR_i given the observed patch x
I_SVR^*	The SVR model selected for the purpose of SR for the patch x
Err_{emp}	The predicted SR error of patches x using E_SVR models

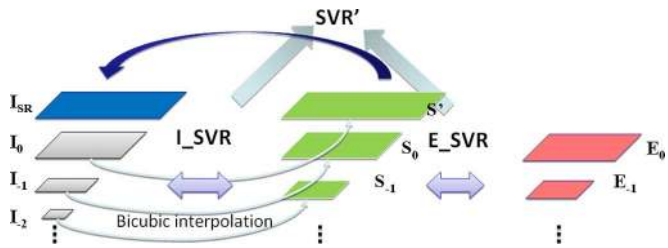

Fig. 2. Our proposed self-learning framework for single image SR. Note that no training low and high-resolution image data is required in this framework.

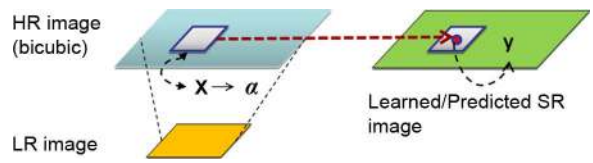

Fig. 3. Refinement of an interpolated high-resolution image (by bicubic interpolation) for SR.

to determine the proper model for SR. Detailed discussions and solutions will be presented in Section II and III, respectively. Finally, the contribution of our proposed SR method is twofold. First, our method focuses on the *learning of proper SR models from different image scales* when producing higher resolution images, and thus our approach does not search for similar image patches for SR reconstruction. This avoids the problem of insufficient self-similarity of patches from different training images (e.g., [5], [7]), within the image of interest (e.g., [25]), or across different scaled versions of that image (e.g., [2]). In other words, there is no need to manually and carefully select training image data, which is not practical to real-world SR applications. The second contribution of our approach is to support the learning of proper image sparse representation for SVR and SR purposes. The use of sparse representation not only can be viewed as an effective image feature, such a compact image representation also accelerates the learning and prediction processes of SR images (due to very few non-zero attributes in a feature vector), which makes the SR with larger magnification factors more computationally feasible.

III. THE PROPOSED SELF-LEARNING APPROACH TO SINGLE IMAGE SUPER-RESOLUTION

This section presents our proposed single image SR method, which is an extension of our previous work [4], and is also inspired by the recent success of the method of Glasner *et al.* [2]. Fig. 2 shows the detailed framework of our approach, where image I_0 is the low-resolution input image, and I_{SR} is the final SR output. Instead of searching for similar patches from images $\{I_{-1}, I_{-2}, \ldots\}$ as [2] did, we focus on modeling the relationships between image patches from each scale. Once these models are observed from each scale, we select the best ones to *refine* each patch in S' into the final output I_{SR}. Using such a *self-learning and data-driven* framework, we need not collect training data from other low and high-resolution image pairs, and we do not require the reoccurrence of image patches either. We now discuss the details of our method. In addition, the descriptions of the symbols used in this paper are listed in Table I.

A. An Image Refinement Process for SR via SVR

In our prior SR method in [4], we approached the SR problem as a refinement (or denoising) process which converts an interpolated HR image into a final SR output, as illustrated in Fig. 3. Like most prior learning-based SR works, our approach in [4] also required the training of SR algorithms from pre-collected training data, which not only limits its practical uses, and the associated SR performance would also be dependent on the training data of use. Since the image refinement process for SR previously proposed in [4] motivates the self-learning approach in this paper, it is necessary to review this work for the sake of completeness.

1) Learning of Image Sparse Representation: As shown in Fig. 3, our work in [4] uses bicubic interpolation to synthesize an LR input image into its HR version. After all 5 x 5 pixel patches are extracted from this synthesized image, proper features need to be selected to describe each patch for learning the SR models. Instead of using raw pixels as image features, we advocate the

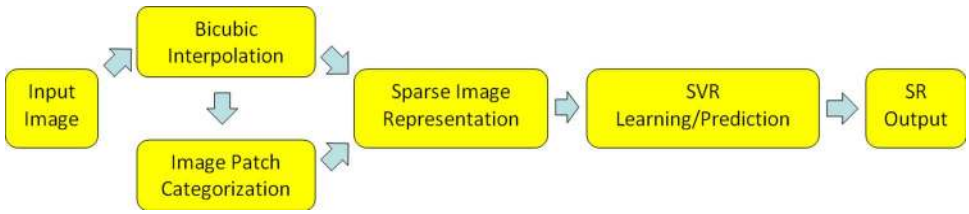

Fig. 4. The flowchart of our previous approach [4] for synthesizing an SR image. Note that the learning of SVR models in this work requires the collection of training low and high-resolution image data.

use of sparse representations as the image features. As suggested by [4], [13], [21], [23],[24], sparse representation has been observed as a preferable type of features to use especially for SR applications. To determine the sparse representation for image patches, an over-complete dictionary will be learned from the extracted patches, and the resulting spare coefficients for each patch will be the features of interest. It is worth noting that, another advantage of using sparse representation as image features is that most feature attributes will be zeroes (due to the enforced sparsity), and thus its use as image features would further decrease the training and computation time when learning or producing SR images (as we confirmed in [4]).

In [4] and our proposed work in this paper, we apply the tool developed by [26] to learn the dictionary D and the associated sparse coefficient vectors α for each patch. This can be formulated as following optimization problem:

$$\min \frac{1}{2}\|D\alpha - x\|_2^2 + \lambda\|\alpha\|_1, \tag{1}$$

where x is the image patch of interest (in terms of pixels), D is the over-complete dictionary to be determined, and α is the corresponding sparse coefficient vector. Note that the Lagrange multiplier λ balances the sparsity of α and the l_2-norm reconstruction error. In our implementation, we divide the image patches into high and low spatial frequency ones and learn their dictionaries accordingly. From our observation (and as verified in [4]), this patch categorization strategy moderately improves the averaged PSNR performance, and computation/training time will be significantly reduced due to a smaller training set for the SR learning model (i.e., SVR in our proposed framework). We note that, one can apply other patch categorization methods like [27] and can expect to obtain context specific representation for the image patches. When integrating such more advanced feature selection techniques into our proposed self-learning framework, further improved SR performance can be expected. Nonetheless, from our observations and the above recent SR works, we consider the use of sparse representation as the features, and we choose to learn distinct sparse representation dictionaries for high and low spatial frequency patches. In the following step, we will discuss how we integrate sparse image representation and SVR for self-learning and predicting the final SR output.

2) Support Vector Regression for SR: Support vector regression (SVR) [3] is an extension of support vector machine, which is able to fit the data in a high-dimensional feature space without assumptions of data distribution. Similar to SVM, the generalization ability makes the SVR very powerful in predicting unknown outputs, and the use of SVR has been shown to produce effective SR outputs in [1], [4]. As illustrated in Fig. 3, the final task of our previous work in [4] aims at modeling the relationship between input data α and the associated SR pixel values y. In training, our SVR solves the following problem

$$\begin{aligned} \min_{w,b,\xi,\xi^*} \quad & \frac{1}{2}w^T w + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ s.t. \quad & y_i - (w^T\phi(\alpha_i) + b) \le \epsilon + \xi_i, \\ & (w^T\phi(\alpha_i) + b) - y_i \le \epsilon + \xi_i^*, \\ & \xi_i, \xi_i^* \ge 0, i = 1, \ldots, n. \end{aligned} \tag{2}$$

In (2), y is the pixel value at the same location as the center of the patch of interest in the high-resolution image. The parameter b is the off-set of the regression model, n is the number of training patches, $\phi(\alpha_i)$ is the sparse image representation in the transformed space, and w is the norm vector of the nonlinear mapping function. Similar to SVM, C is the tradeoff between the generalization and the upper/lower bounds of training errors ξ_i and ξ_i^*, subject to a margin ϵ. In our implementation, we subtract the mean value of each patch from its pixel values x before calculating the sparse coefficient α, and the same value is also subtracted from the corresponding pixel label y in the high resolution image. This is because our approach suggests the learning and prediction of local pixel value variations, not the absolute pixel value output. In testing, the mean value of each patch will be added to the predicted output pixel value y.

Once the training of SVR is complete, the observed SVR will be applied to predict the final SR output for a test LR input. Using the flowchart shown in Fig. 4, our work in [4] first synthesizes the HR version from the test input using bicubic interpolation. To refine this HR image into the final SR output, we derive the sparse representation of each patch and update the centered pixel value for each using the learned SVR models. From the experimental results shown in [4], we achieved better and comparable SR results than some prior learning-based SR approaches did. While this refinement process exhibits impressive capability in producing SR images, its need to collect training LR/HR image data in advance is not desirable. In this paper, a novel self-learning approach for single image SR based on this refinement strategy is presented, as discussed in the following subsections.

B. Self-Learning of In-Scale SVR for Image SR

To address the single-image SR problem with a self-learning strategy, We first downgrade the resolution of an input LR image I_0 into several lower-resolution versions $\{I_{-1}, I_{-2}, \ldots\}$. Next, we interpolate these images to synthesize the corresponding

higher-resolution images $\{S_0, S_{-1}, \ldots\}$ using bicubic interpolation. The entire framework is shown in Fig. 2. From this figure, it is clear that two image pyramids $\{I_i\}$ and $\{S_i\}$ will be constructed, and each image I_i can be considered as a ground truth image of its interpolated version S_i. It is worth noting that, *we do not limit this framework to the use of bicubic interpolation when constructing the image pyramids*. Later in our experiments, we will apply other state-of-the-art SR techniques to produce the up-sampled images S_i, and we will show that our proposed method is still able to produce effective SR results.

Once the ground truth image I_i and its interpolated version S_i are obtained, we extract the sparse image representation of each patch in S_i. Similar to our work in [4], we apply SVR to model the relationship between S_i and I_i, and we have the set $I\_SVR = \{I\_SVR_0, I\_SVR_{-1}, \ldots\}$ as the collection of SVR models indicating the learning models observed from different image scales. As discussed in Section III-A, SVR is subject to training errors ξ_i and ξ_i^* with precision ϵ. As a result, the predicted result of S_i using I_SVR_i does not guarantee to be exactly the same as I_i, and the resulting error image E_i will be obtained. Thus, each pixel e_{ij} in this error image E_i can be calculated as follows:

$$e_{ij} = \|I_{ij} - I\_SVR_i(\alpha_{ij})\|, i = 0, -1, \ldots. \tag{3}$$

In the above equation, I_{ij} is the j-th pixel value in image I_i (at the ith scale), and α_{ij} is the sparse representation of the jth image patch in S_i. Once the error image E_i is obtained, we choose to fit another SVR regression model between S_i and E_i, and this model is denoted as E_SVR_i. In practice, the precision ϵ for E_SVR_i will be smaller than that for the I_SVR_i. This is because, given an input patch (in terms of its sparse representation), the output of E_SVR_i will be used as a cue that how well the associated I_SVR_i is for the purpose of image SR. In other words, *a smaller E_SVR_i output indicates that the corresponding model I_SVR_i fits the patch relationship between images S_i and I_i better*.

Let us take I_{-1} in Fig. 2 for example. Using image sparse representation as the input feature, we first learn the SVR model I_SVR_{-1}, which refines the interpolated image S_{-1} into the ground truth image I_{-1}. The predicted output image of S_{-1} is the orange image shown (i.e., the learned HR image) in Fig. 5. Since this predicted SVR output will not be the same as the ground truth I_{-1} due to the precision ϵ and training errors ξ, the induced error image E_{-1} (the red image in Fig. 5) can be thus calculated as the difference between the predicted image and the ground truth I_{-1}, and the error SVR model E_SVR_{-1} can be trained to associate E_{-1} with S_{-1}. For an input image patch in the synthesized HR image S' in Fig. 2, if this E_SVR_{-1} produces a smaller output than other E_SVR_i models at other scales, this implies that the corresponding I_SVR_{-1} better fits the model between S' and I_{SR}, and thus I_SVR_{-1} will be applied to refine that image patch into its SR output. Similar remarks apply to other SVR models at different scales.

To summarize our proposed framework, we construct the image pyramids from an LR input, as depicted in Fig. 2. Both types of SVR models I_SVR and E_SVR will be observed from different image scales. To refine the patches in S' into I_{SR} as the final output, we predict the errors for each patch using different E_SVR models. Once the smallest E_SVR_i output is determined for a patch, we choose the associated I_SVR_i for the prediction/refinement purpose (recall that, a smaller E_SVR_i output indicates a better I_SVR_i for that patch to fit in). In the following subsection, we will discuss the background theories for this SVR model selection process. Moreover, experimental results in Section IV will also verify the effectiveness of this proposed selection method.

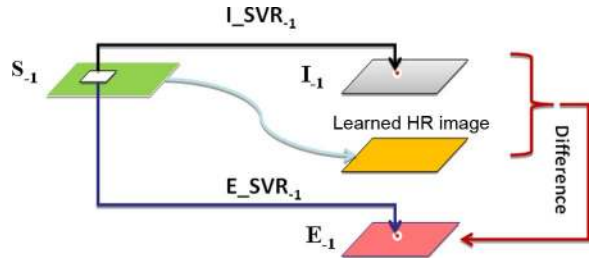

Fig. 5. Illustration of SVR models I_SVR_{-1} and E_SVR_{-1} for the ground truth I_{-1}, interpolated image S_{-1}, and the corresponding error E_{-1}.

C. SVR Model Selection via Bayesian Decision Theory

As discussed above, when refining each patch in the target high-resolution image S' into the final SR output, our proposed method aims at selecting the most desirable SVR model which is learned from the subsequent image scales. More precisely, for the E_SVR_i producing the smallest output, we consider the associated I_SVR_i to be the best in terms of refining the patch x in S' into that in I_{SR}.

We observe that this SVR model selection process corresponds to a minimum-error-rate classification rule based on Bayesian decision theory. While the traditional Bayes classification rule aims to minimize the classification error, our SVR selection process focuses on minimizing the image reconstruction error for SR purposes. To be more specific, for a patch x in S', its preferable I_SVR_i model is the category label to be predicted, and the difference between that patch in the resulting SR output I_{SR} and that in the ground truth HR image is the prediction error. We now discuss why the proposed SVR model selection process is associated with the Bayes decision theory, and why a minimum SR reconstruction error can be achieved using our decision rule.

Let $P(x|I\_SVR_i, D_i)$ be the category-conditional probability function (or likelihood) of x, where D_i is the dictionary for image sparse representation in image scale i and x is the input patch. Under the assumption of equal prior (i.e., the prior probabilities $P(I\_SVR_i)$ are the same across image scales, which means the SVR models learned from each scale are equally informative), the posterior probability $P(I\_SVR_i, D_i|x)$ will be proportional to $P(x|I\_SVR_i, D_i)$ due to the Bayes rule. For each image scale i, we compute the posterior probability $P(I\_SVR_i, D_i|x)$ for the patch x as follows

$$P(x|I\_SVR_i, D_i) = 1 - \frac{|E\_SVR_i(x, D_i)|}{c}, \tag{4}$$

where $c = |\max(E\_SVR_l(x, D_l))|, \forall l$. It can be seen from (4) that the likelihood of x is inversely proportional to the E_SVR_i

output modeling the reconstruction/prediction error at image scale i (i.e., one minus the normalized regression error output), and the above equation allows us to calculate this likelihood accordingly. Using (4), a smaller E_SVR_i output for the patch x will result in a larger $P(x|I\_SVR_i, D_i)$. As derived above, a large likelihood $P(x|I\_SVR_i, D_i)$ value corresponds to a large posterior probability $P(I\_SVR_i, D_i|x)$, and it implies that the patch x is more likely to occur in the ith scale (and thus the model I_SVR_i is the most preferable among all). We note that this observation is consistent with our proposed SVR model selection rule. Later in Section IV, we will verify the validity of our SR method based on this probability model, and we will show that the use of our proposed SVR model selection method consistently outperforms the random selection one for all test images.

To produce the final I_{SR}, we calculate the posterior probability $P(I\_SVR_i, D_i|x)$ for each patch x in the target high-resolution image S'. For the I_SVR_i model resulting in the maximum $P(I\_SVR_i, D_i|x)$ output, this SVR model will be used for refining the patch x of S' into the I_{SR} output. This SVR model selection process can be determined as follows

$$\begin{aligned} I\_SVR^* &= \arg\max_i P(I\_SVR_i, D_i|x) \\ &= \arg\max_i P(x|I\_SVR_i, D_i)P(I\_SVR_i, D_i) \\ &= \arg\max_i (1 - \frac{|E\_SVR_i(x, D_i)|}{c}) \\ &= \arg\min_i (\frac{|E\_SVR_i(x, D_i)|}{c}). \end{aligned} \tag{5}$$

We note that the above process corresponds to the Bayes decision rule in minimizing the prediction error when assigning the category I_SVR_i to the input x. The last term in (5) indicates our selection rule, i.e., the model I_SVR_i is preferable for patch x if its E_SVR_i produces a smallest output. For example, we select the model $I\_SVR^* = I\_SVR_{-1}$ for the patch x in image S', if its $E\_SVR^* = E\_SVR_{-1}$ is the smallest among all E_SVR_i.

We now derive the expected reconstruction error using the aforementioned rule, and we will show that why our method is able to achieve a preferable SR output with a *smallest reconstruction error* (and thus a large PSNR value). Let N be the number of pixels (patches) in the target high-resolution image S', we approximate the predicted SR error Err_{emp} in the final SR output I_{SR} by

$$\begin{aligned} Err_{emp} &\approx \sum_{j=1}^{N} E\_SVR^*(x_j, D^*) \\ &\leq \sum_{j=1}^{N} E\_SVR_i \\ &\quad \times (x_j, D_i),\ \forall E\_SVR_i \neq E\_SVR^*. \end{aligned} \tag{6}$$

From (6), we conclude that if one does not choose proper I_SVR_i models to obtain the final SR output as we do, the resulting reconstruction error will always be larger than that produced by our SR method. In other words, using such a self-learning framework, our selection rule for choosing proper learning models in predicting the final SR output will result in the *minimum* SR reconstruction error.

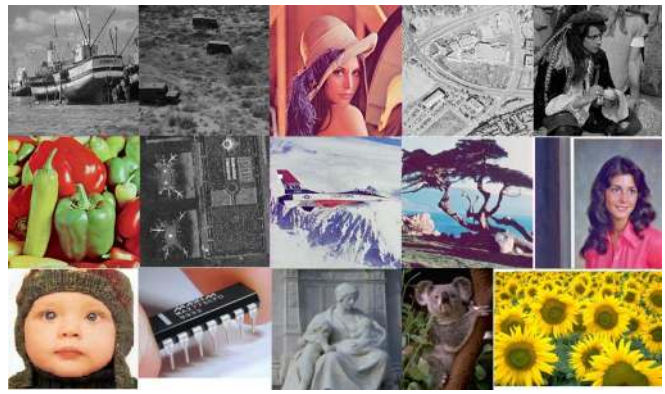

Fig. 6. Example images for SR in our experiments.

IV. Experimental Results

In Section IV-A, we first evaluate our proposed SR method with different magnification factors on several grayscale or color images. Besides presenting the SR images and their PSNR values, we also compare our results with those produced by state-of-the-art learning-based SR methods. In Section IV-B, we replace bicubic interpolation in our framework by other up-sampling or SR algorithms. This is to verify that our proposed framework is *not* limited to the use of any specific type of image up-sampling techniques when constructing the image pyramids. These two parts of experiments are to verify the effectiveness and robustness of our proposed method, respectively.

A. Quantitative and Qualitative Comparisons of SR Results

We test our SR approach on a variety of images, which are collected from the USC-SIPI database (http://sipi.usc.edu/database) and by Fattal *et al.* [28]. Some example images are shown in Fig. 6. We do not particularly consider any specific category of images for SR (e.g., natural scenes, artificial objects, etc.), and we do not require the collection of training data as other learning-based SR approaches did. For color images, we only perform SR on the illuminance channel in the YUV color space, and the remaining color channels are up-sampled by bicubic interpolation. When synthesizing an SR output, we start with the original image as the ground-truth HR image, and we degrade it into LR versions by nearest neighbor interpolation or Gaussian blur kernels.

In the implementation of our SR framework, we train our SVR models using LIBSVM [29]. Since only linear SVR models are considered, the computation time of our SR algorithm is actually *less* than or comparable to state-of-the-art learning-based SR methods (as shown later). To learn the image sparse representation, we construct the dictionaries and calculate the corresponding sparse coefficients using the software developed by [26], in which the sparsity constraint (i.e., λ) is set to 0.25 and the size of the dictionary is 100. With these parameter choices, the sparsity of the derived sparse representation can be observed. We have the average number

TABLE II
COMPARISONS OF SR IMAGES PRODUCED BY DIFFERENT METHODS WITH A MAGNIFICATION FACTOR OF 2, INCLUDING THE AVERAGE TRAINING AND TESTING TIME (T_{Tr} AND T_{Te}) FOR EACH. NOTE THAT THE RUN-TIME ESTIMATES (IN MINUTES) ARE OBTAINED ON AN INTEL QUAD CORE PC WITH 2.23 GHZ PROCESSORS AND 2G RAM, AND THE INPUT IMAGES ARE DOWNSAMPLED FROM THEIR GROUND TRUTH VERSIONS USING NEAREST NEIGHBOR INTERPOLATION

	boat	cars	skyview	lena	person	fruit	station	airplane	tree	susan	T_{Tr} / T_{Te}	Training Image
Bicubic	26.59	27.46	23.69	29.70	24.35	32.56	22.82	27.65	27.00	33.44	- / -	N/A
LLE [15]	25.25	26.59	22.30	27.43	23.45	29.84	21.63	25.26	26.87	30.79	5 / 15	Yes
Yang et al. [21]	26.27	27.63	23.49	28.86	**25.32**	32.47	23.57	27.58	27.09	32.72	5500 / 5	Yes
Wang et al. [4]	26.81	27.75	23.83	29.89	24.69	31.83	22.73	**28.34**	27.13	34.19	170 / 20	Yes
Glasner et al. [2]	25.02	26.03	22.05	28.31	23.22	**34.42**	21.34	25.03	24.58	31.49	- / 250	**No**
Ours (random SVR sel.)	28.18	28.11	24.92	32.15	24.19	31.97	23.44	28.10	27.02	33.75	- / 160	**No**
Ours	**28.45**	**28.50**	**25.10**	**32.63**	24.60	32.63	**23.78**	28.28	**27.31**	**34.21**	- / 160	**No**

TABLE III
COMPARISONS OF SR IMAGES PRODUCED BY DIFFERENT METHODS WITH A MAGNIFICATION FACTOR OF 2. NOTE THAT THE INPUT IMAGES ARE DOWNSAMPLED FROM THEIR GROUND TRUTH VERSIONS USING GAUSSIAN BLUR KERNELS

	boat	cars	skyview	lena	person	fruit	station	airplane	tree	susan
Bicubic	30.21	30.87	27.24	34.60	27.45	31.97	25.12	25.64	26.87	33.74
LLE [15]	25.61	26.87	22.73	28.63	24.05	27.52	22.46	23.24	22.16	28.61
Yang et al. [21]	29.66	30.34	27.84	32.92	27.83	32.09	25.71	25.25	27.59	33.44
Wang et al. [4]	30.36	30.89	27.52	34.54	27.60	32.48	25.44	25.91	27.50	34.38
Glasner et al. [2]	**31.23**	30.10	28.06	35.62	27.92	**35.39**	25.61	**26.35**	26.28	33.86
Ours (random SVR sel.)	30.46	31.32	27.56	35.34	28.08	33.06	25.63	25.93	27.22	33.77
Ours	31.01	**31.52**	**28.20**	**35.94**	**28.32**	32.98	**25.76**	26.22	**27.85**	**35.26**

of non-zero entries as 2.93 out of 100 (more specifically, 5.54 for the high-frequency patches and 1.99 for the low-frequency ones). In addition, we limit the number of levels in the image pyramid of our framework less than or equal to three. This choice avoids producing images with insufficient resolution (e.g., 32×32 pixels or less), which might result in over-fitting during the learning of SVR.

To quantitatively compare our results with those produced by prior SR methods, Table II lists the PSNR values of SR images using different approaches: bicubic interpolation, locally linear embedding (LLE) for SR [15], sparse representation for SR by Yang *et al.* [21], our previous SVR method for SR (Wang *et al.*) [4], the recent work of Glasner *et al.* [2], our framework using randomly-chosen SVR models, and our proposed method (i.e., with the selection of SVR via Bayesian decision theory). We note that we apply the Matlab code implemented by Yang *et al.* [24] when comparing to the method of [2]. Their code is available at the website http://faculty.ucmerced.edu/mhyang/code/accv10_super.zip. All the input LR images are down-sampled from their ground truth versions using nearest neighbor interpolation. As can be seen in the last column of Table II, all the learning-based SR approach except for ours and that of Glasner *et al.* [2] need to collect/select training image data. For [21], the training images are included in their software, and thus we use their designed dictionary and algorithms for SR directly. For [4] and [15], we randomly select an image *bridge* from the USC-SIPI dataset for training. We note that the implementation of [4] and [15] requires only one training image, while several hundred images are needed for [21] (as provided by their software package). From Table II (and later summarized in Table IV), we observe PSNR improvements from 0.07 dB to 2.93 dB, or 1.02 dB (3.77%) in average, for our method over bicubic interpolation.

In additional, we repeat the above experiments but replacing the nearest neighbor interpolation by a Gaussian blur kernel (we set $\sigma = 1$ as [2] did) for degrading the resolution of input images, which is considered as a more general way in degrading the image resolution in practice. Table III lists the PSNR comparisons, and we observe an averaged 0.93 dB (or 3.14%) improvement using our proposed approach for this scenario. From the last two rows in Tables II and III, it is clear that the proposed probability-based SVR model selection strategy consistently outperformed the random selection one in all SR image outputs, which thus verified the validity of the proposed method (including the SVR probability models). To make the comparisons more complete, we list the PNSR improvements of different SR methods with different magnification factors in Table IV. Besides, we consider a larger magnification factor of 4 and repeat the above experiments. The PSNR values of different methods are also listed in Table IV. From this summary table, it can be observed that our approach consistently achieved the best PSNR improvements using different down-sampling techniques and with different magnification factors.

TABLE IV
AVERAGE PSNR IMPROVEMENTS (IN dB) OF DIFFERENT SR METHODS OVER BICUBIC INTERPOLATION WITH DIFFERENT MAGNIFICATION FACTORS M. NOTE THAT INPUT IMAGES GENERATED BY GAUSSIAN BLUR OR NEAREST NEIGHBOR TECHNIQUES ARE CONSIDERED FOR $M = 2$

M	2 (Gaussian blur)	2 (NN)	4
Wang et al. [4]	0.29 ± 0.24	0.19 ± 0.42	0.42 ± 0.96
LLE [15]	-4.18 ± 1.10	-1.58 ± 0.88	-0.14 ± 0.59
Yang et al. [21]	-0.10 ± 0.74	-0.34 ± 1.16	0.83 ± 1.39
Glasner et al. [2]	0.77 ± 0.48	-1.65 ± 1.73	-0.57 ± 1.54
Our method	0.93 ± 0.30	1.02 ± 0.86	1.57 ± 1.16

As for the computation time, Table II lists actual runtime for different SR methods. The average computation time (in minutes) for training and testing, i.e., T_{Tr} and T_{Te}, are listed. For the first three learning-based methods in Table II requiring training image data, LLE [15] reported the least computation time (with only one training LR/HR image pair) but produced slightly poorer or comparable PSNR results than bicubic interpolation. While neither the methods of Glasner *et al.* nor ours

Fig. 7. Example SR results (with a magnification factor 2) and their PSNR values. Top row: ground truth HR image, bicubic interpolation (PSNR: 29.70), LLE [15] (PSNR: 27.43). Bottom row: Yang *et al.* [21] (PSNR: 28.86), Glasner *et al.* [2] (PSNR: 28.81), and ours (PSNR: 32.63). Note that nearest neighbor interpolation is applied to degrade the image resolution in this example.

Fig. 8. Example SR results (with a magnification factor 4) and their PSNR values. Top row: ground truth HR image, bicubic interpolation (PSNR: 28.19), LLE [15] (PSNR: 26.85). Bottom row: Yang *et al.* [21] (PSNR: 26.12), Glasner *et al.* [2] (PSNR: 24.87), and ours (PSNR: 28.66). Note that nearest neighbor interpolation is applied to degrade the image resolution in this example.

require training image data (and thus *no* training time is reported in the table), our average time to produce an SR image is only 64% of that required by Glasner *et al.* (160 min. vs. 250 min.) and thus is computationally more preferable. However, it is worth noting that, since we did not particularly perform any optimization techniques for the SR methods considered in this paper (including ours), we simply reported the training/test time using the Matlab code either implemented by ourselves or released by the authors (such as [2], [15], [21]).

Fig. 7 shows example SR results of different SR approaches, in which a portion of the SR image is enlarged for detailed comparisons. Comparing the SR images produced by different methods, we see that our approach obtained a satisfactory result without blurring, etc. artifacts along the edges, while the highest PSNR value was achieved. Using a larger magnification factor of 4, Fig. 8 shows another SR example with a larger magnification factor of 4, and the effectiveness of our proposed method is again verified.

Fig. 9. Example SR results applying Gaussian blur kernels for degrading the image resolution. Top row: Ground truth image, bicubic interpolation (PSNR: 27.45), LLE [15] (PSNR: 24.05). Bottom row: Yang *et al.* [21] (PSNR: 27.83), Glasner *et al.* [2] (PSNR: 27.92) and ours (PSNR: 28.32).

Fig. 10. Example SR results (with a magnification factor 4) and the corresponding PSNR values. Top row: Ground truth image, SR image produced by Genuine Fractals (software available at www.ononesoftware.com), Freeman *et al.* [14] (PSNR: 21.34), Fattal *et al.* [31] (PSNR: 21.36), Wang *et al.* [4] (PSNR: 23.86). Bottom row: Kim and Kwon [22], [23] (PSNR: 20.93), Glasner *et al.* [2] (PSNR: 20.94), LLE [15] (PSNR: 29.8), Yang *et al.* [21] (PSNR: 24.43), and ours (PSNR: 31.14).

Finally, we replace nearest neighbor interpolation by Gaussian blur kernels (as [2] did) when down-sampling images in our experiments; this is to confirm that our method applies to practical image processing scenarios. More precisely, to synthesize the input LR images from their ground truth HR versions, we apply Gaussian blur kernels before down-sampling the images. The same down-sampling technique is also applied to construct the image pyramid in our proposed framework for self-learning purposes. We show example SR results on three different images in Figs. 9–11. From these figures (and the listed PSNR values), one can see the advantages of our method in producing qualitatively and quantitatively satisfactory SR results.

B. Our SR Framework With Different Up-Sampling Techniques

As shown in Fig. 2, we apply bicubic interpolation to up-sample the images $\{I_{-1}, I_{-2}, \ldots\}$ into $\{S_0, S_{-1}, \ldots\}$ in the proposed framework for self-learning purposes. While very promising SR results were reported in Section IV-A, we need to address the concern that whether the use of bicubic interpolation is critical in this framework. In other words, when other up-sampling or SR techniques are deployed in our framework, we need to verify whether the proposed self-learning approach is still able to achieve satisfactory SR results. While similar remarks have been concluded in [30], it is necessary to confirm the robustness of our proposed SR framework, and to show that it is not limited to the use of bicubic interpolation during the self-learning process.

We now consider the methods of Lanczos, LLE [15], Yang *et al.* [21], and Glasner *et al.* [2] as the up-sampling techniques. We replace bicubic interpolation in Fig. 2 by these methods, and the resulting SR performance is shown in Table V. The first row of Table V lists the average PSNR improvement of our proposed self-learning framework, in which the default choice of bicubic interpolation is used to construct the image pyramid from the input LR image. The remaining entries in Table V list the PSNR

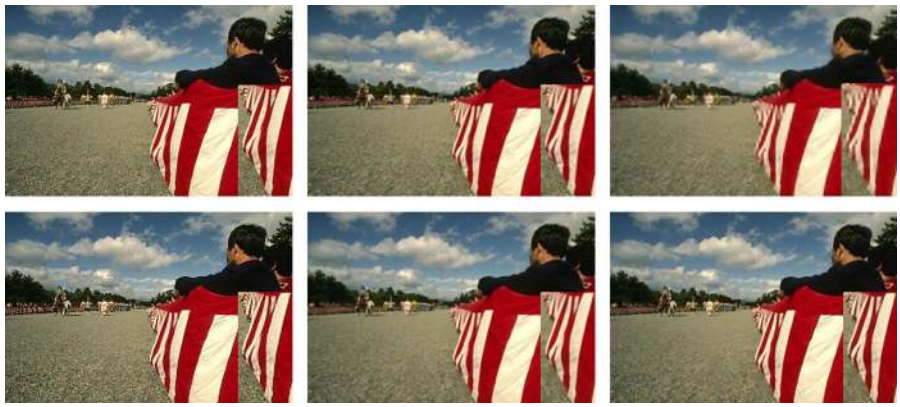

Fig. 11. Example SR results applying Gaussian blur kernels for degrading the image resolution. Top row: Ground truth image, bicubic interpolation (PSNR: 27.08), LLE [15] (PSNR: 24.23). Bottom row: Yang *et al.* [21] (PSNR: 25.99), Glasner *et al.* [2] (PSNR: 24.41) and ours (PSNR: 27.48).

Fig. 12. Example SR results using different up-sampling techniques. SR image produced by Glasner *et al.* [2] (PSNR: 21.34), error image between the ground truth and the SR image of Glasner *et al.*; our SR output using the approach of Glasner *et al.* [2] to up-sample the intermediate images in the proposed framework (PSNR: 23.42) and error image between the ground truth and the SR image using Glasner *et al.* approach to up-sample the intermediate images. It can be seen that, while not using bicubic interpolation, our approach still performs better than that of Glasner *et al.* Note that the error images are scaled into the same display range for detail comparisons.

TABLE V
AVERAGE PERFORMANCE IMPROVEMENTS USING DIFFERENT UP-SAMPLING TECHNIQUES IN OUR SR FRAMEWORK

Approach	PSNR improvement (in dB)
Bicubic (default choice)	1.02 ± 0.86
Lanczos	0.82 ± 1.02
LLE [15]	1.67 ± 1.07
Yang et al. [21]	1.37 ± 0.38
Glasner et al. [2]	2.41 ± 0.34

improvements when applying the above techniques for up-sampling, and it can be seen that all the values are positive and larger than those reported by other learning-based SR methods shown in Table IV. We also show an SR example in Fig. 12, in which the left image was produced by Glasner *et al.* [2], while the right one was obtained by our approach using the SR method of Glasner *et al.* [2] as the up-sampling technique. Comparing the two images (and the corresponding error images) in Fig. 12, we see that our proposed self-learning framework indeed achieved visually improved SR results (especially with less noise and artifacts along the edges). As a result, the robustness of our SR method is successfully verified.

V. CONCLUSION

This paper proposed a novel in-scale self-learning framework for single image SR. We advanced the learning of support vector regression and image sparse representation in our SR method, which exhibited excellent generalization in refining an up-sampled image into its SR version. Different from most prior learning-based approaches, our approach is very unique since we do not require training low and high-resolution image data. We do not assume context, edge, etc. priors when synthesizing SR images, nor do we expect reoccurrence of image patches in images as many prior learning-based methods did. Supported by the Bayes decision rule, our method produced excellent SR results on a variety of images, and we achieved a significant improvement in PSNR when comparing with state-of-the-art SR approaches. Moreover, by deploying different types of interpolation or SR techniques to up-sample images at intermediate scales in our framework, we confirmed the robustness and effectiveness of our proposed SR framework. Future research will be directed at the extension of our current SVR models to the use of multiple kernel learning for adaptive kernel selection.

REFERENCES

[1] K. S. Ni and T. Q. Nguyen, "Image superresolution using support vector regression," *IEEE Trans. Image Process.*, vol. 16, no. 6, pp. 1596–1610, Jun. 2007.
[2] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE Int. Conf. Comput. Vision*, 2009.
[3] A. J. Smola and B. Schölkopf, A Tutorial on Support Vector Regression Statistics and Computing, 2003.
[4] M.-C. Yang, C.-T. Chu, and Y.-C. F. Wang, "Learning sparse image representation with support vector regression for single-image super-resolution," in *Proc. IEEE Int. Conf. Image Processing*, 2010.

[5] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint map registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1621–1633, 1997.
[6] N. Nguyen, P. Milanfar, and G. H. Golub, "A computationally efficient superresolution image reconstruction algorithm," *IEEE Trans. Image Process.*, vol. 10, no. 4, pp. 573–583, Apr. 2001.
[7] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multi-frame super-resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.
[8] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1167–1183, Sep. 2002.
[9] J. Sun, J. Zhu, and M. F. Tappen, "Context-constrained hallucination for image super-resolution," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2010.
[10] M. E. Tipping and C. M. Bishop, "Bayesian image super-resolution," *Adv. Neural Inf. Processing Syst.*, 2002.
[11] H. Y. Shum and Z. C. Lin, "Fundamental limits of reconstruction-based superresolution algorithms under local translation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 83–97, Jan. 2004.
[12] A. Chakrabarti, A. N. Rajagopalan, and R. Chellappa, "Super-resolution of face images using kernel PCA-based prior," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 888–892, 2007.
[13] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 27–35, Jan. 2009.
[14] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, 2002.
[15] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2004.
[16] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graph. Models Image Process.*, vol. 53, no. 3, 1991.
[17] D. Capel, *Image Mosaicing and Super-Resolution (Cphc/Bcs Distinguished Dissertations)*. New York, NY, USA: Springer-Verlag, 2004.
[18] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
[19] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. ICCV*, , pp. 2272–2279.
[20] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2008.
[21] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
[22] K. I. Kim and Y. Kwon, Example-Based Learning for Single-Image Super-Resolution and jpeg Artifact Removal, Max-Planck-Institute for Biological Cybernetics, Tübingen, Germany, 2008.
[23] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 6, pp. 1127–1133, Jun. 2010.
[24] C.-Y. Yang, J.-B. Huang, and M.-H. Yang, "Exploiting self-similarities for single frame super-resolution," in *Proc. Asian Conf. Comput. Vision*, 2010, pp. 497–510.
[25] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the non-local-means to super-resolution reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 36–51, Jan. 2009.
[26] J. Mairal, F. R. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, 2010.
[27] J. Sun, N.-N. Zheng, H. Tao, and H.-Y. Shum, "Image hallucination with primal sketch priors," in *Proc. IEEE Int. Conf. Comput. Vision and Pattern Recognition*, 2003.
[28] Y. HaCohen, R. Fattal, and D. Lischinski, "Image upsampling via texture hallucination," in *Proc. IEEE Int. Conf. Computational Photography*, 2010, pp. 1–8.
[29] C.-C. Chang and C.-J. Lin, LIBSVM: a Library for Support Vector Machines, 2001.
[30] Z. Xiong, X. Sun, and F. Wu, "Image hallucination with feature enhancement," in *Proc. IEEE Int. Conf. Comput. Vision and Pattern Recognition*, 2009.
[31] R. Fattal, "Upsampling via imposed edges statistics," *ACM Trans. Graph.*, vol. 261, no. 3, 2007.

Min-Chun Yang received B.S. and M.S. degrees in Computer Science and Information Engineering from National Chung Cheng University, Chia-Yi, Taiwan in 2006 and 2008, respectively. He is currently pursuing his Ph.D. degree in Computer Science and Information at National Taiwan University, Taipei, Taiwan.

During the Ph.D. study, he works as a research assistant and is supported in part by the Research Center for Information Technology Innovation, Academia Sinica, Taiwan. His research interests are in the areas of image processing, focusing on biomedical image processing, image super-resolution, and image sparse representation.

Yu-Chiang Frank Wang (M'09) received the B.S degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan in 2001. From 2001 to 2002, he worked as a research assistant at the National Health Research Institutes, Taiwan. He received his M.S. and Ph.D. degrees in Electrical and Computer Engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2004 and 2009, respectively.

Since 2009, he has been with the Research Center for Information Technology Innovation (CITI) of Academia Sinica, Taiwan, where he holds the position as a tenure-track assistant research fellow. He leads the Multimedia and Machine Learning Lab at CITI, and works in the fields of signal and image processing, computer vision, and machine learning. From July to December 2010, he is a visiting scholar of the Department of Computer Science and Information Engineering at National Taiwan University Science and Technology, Taiwan.