A Statistical Prediction Model Based on Sparse Representations for Single Image Super-Resolution

Tomer Peleg, *Student Member, IEEE*, and Michael Elad, *Fellow, IEEE*

***Abstract*— We address single image super-resolution using a statistical prediction model based on sparse representations of low- and high-resolution image patches. The suggested model allows us to avoid any invariance assumption, which is a common practice in sparsity-based approaches treating this task. Prediction of high resolution patches is obtained via MMSE estimation and the resulting scheme has the useful interpretation of a feedforward neural network. To further enhance performance, we suggest data clustering and cascading several levels of the basic algorithm. We suggest a training scheme for the resulting network and demonstrate the capabilities of our algorithm, showing its advantages over existing methods based on a low- and high-resolution dictionary pair, in terms of computational complexity, numerical criteria, and visual appearance. The suggested approach offers a desirable compromise between low computational complexity and reconstruction quality, when comparing it with state-of-the-art methods for single image super-resolution.**

***Index Terms*— Dictionary learning, feedforward neural networks, MMSE estimation, nonlinear prediction, single image super-resolution, sparse representations, statistical models, restricted Boltzmann machine, zooming deblurring.**

I. Introduction

IMAGE super-resolution (SR) are techniques aiming at resolution enhancement of images acquired by low-resolution (LR) sensors, while minimizing visual artifacts. These techniques offer the promise of overcoming the inherent limitations of LR imaging (e.g. surveillance cameras) and better utilization of the growing capability of high-resolution (HR) displays (e.g. HD LCDs). The classical image SR problem is cast as fusing multiple LR images of the same scene obtained at sub-pixel misalignments into one HR image (for a comprehensive review see [1] and [2]). However, this problem is typically severely ill-conditioned due to the insufficient number of observations and the unknown registration parameters, limiting classical SR to small scale-up factors (less than 2). Recently, the field of *single* image SR has drawn considerable attention [3]–[24]. In this setup, image recovery is cast as a severely underdetermined inverse problem, regularized by utilizing an image model or prior.

The baseline method for approximating a solution to this problem is through conventional linear interpolators, of which the bicubic interpolator is highly popularized. In recent years a wide range of approaches has been suggested that outperform bicubic interpolation. Models that are commonly exploited in single image SR methods include image smoothness and geometric regularity of image structures [3]), gradient profile priors [4], self-similarity of image patches within and across different scales in the same image [6], [7], [24], and sparsity – either in the wavelet domain [8], [11], through Gaussian mixture models [14] or through an analysis operator [20]. The methods leading to the best performance incorporate a combination of several such ideas. Zhang et al. [18] take advantage of sparsity of the reconstructed image through a multi-scale dictionary, along with local and non-local image priors, while Dong et al. [12], [22] combine the ideas of data clustering, adaptive PCA-based sparse representations and non-local self-similarity of image patches within a given image.

A different line of work, which will be most relevant to this paper, is based on sparse representation of low and high resolution (LHR) patch-pairs over a dictionary pair $\mathbf{D}_l, \mathbf{D}_h$. The core idea underlying this approach is that each LHR patch-pair is sparsely represented over the dictionaries $\mathbf{D}_l$ and $\mathbf{D}_h$ and the resulting representations $\boldsymbol{\alpha}_l, \boldsymbol{\alpha}_h$ are assumed to have a pre-specified correspondence. A common assumption in this context is sparse representation invariance [9], [10], [15], where patches in a LHR pair have the same sparse representation over the LHR dictionary pair, namely $\boldsymbol{\alpha}_h = \boldsymbol{\alpha}_l$. To allow for a meaningful recovery of the HR patch, this calls for joint learning of the dictionary pair. The authors of [9], [10] suggest to first learn a dictionary $\mathbf{D}_l$ that best fits the LR patches, and then learn a dictionary $\mathbf{D}_h$ that works best with the resulting coefficients $\boldsymbol{\alpha}_l$ in recovering the HR patches. Yang et al. [15] suggest a fully-coupled learning approach for the dictionary pair via a bi-level optimization problem.

While sparse representation invariance is indeed a simplifying assumption, it suffers from two inherent drawbacks. First, to ensure alignment of the dictionary pair, both dictionaries should have the same number of atoms. In order to allow for a rich recovery model this number should be very large with respect to the patch dimension (typically $1,000$ atoms for patches of size 9×9). Second, since LR patches consist of low energy in coefficients corresponding to edges and textures,

Manuscript received September 7, 2013; revised December 23, 2013; accepted February 7, 2014. Date of publication February 12, 2014; date of current version May 7, 2014. This work was supported in part by the European Research Council under EU's 7th Framework Program, in part by ERC under Grant 320649, and in part by Intel Collaborative Research Institute for Computational Intelligence. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Brendt Wohlberg.

T. Peleg is with the Department of Electrical Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel (e-mail: tomerfa@tx.technion.ac.il).

M. Elad is with the Computer Science Department, Technion – Israel Institute of Technology, Haifa 32000, Israel (e-mail: elad@cs.technion.ac.il).

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/TIP.2014.2305844

a pre-processing stage in the form of high-pass filtering is required to allow for good recovery of HR detail through sparse representation invariance.

Recently, several attempts have been made to go beyond the invariance assumption, aimed at improving the stability of the recovery. Wang et al. [17] learn linear mappings between $\boldsymbol{\alpha}_l$ and $\boldsymbol{\alpha}_h$, combined with ℓ_1 sparsity. Jia et al. [19] and He et al. [23] suggest restricting the invariance only to the supports[1] and assume a linear map between the non-zero representation coefficients. The latter addresses the joint dictionary learning task through a Bayesian approach that utilizes a Beta process prior. This approach allows inferring the number of dictionary atoms non-parametrically, which turns out to be smaller with respect to previous approaches. Note that two of these methods [17], [19] utilize data clustering and learn a set of linear mappings – one for each cluster. While the number of sets in [17] is small (32), it becomes huge (32, 104) in [19]. All three approaches overcome the need for pre-processing. However, the two dictionaries are obliged to have the same number of atoms, which remains large with respect to the patch dimension (the dictionaries are highly redundant). Thus, they involve a computationally expensive sparse coding stage for the low-resolution patches.

Our work is also based on sparse representation of patch-pairs over a dictionary pair. However, we make no invariance assumption. Instead we suggest a parametric model which captures the statistical dependencies between the sparsity patterns of the LHR coefficients and between the corresponding nonzero coefficients. Prediction of $\boldsymbol{\alpha}_h$ from $\boldsymbol{\alpha}_l$ is carried out by the MMSE estimator arising directly from the model, which is shown to have a closed form formula. Our model does not require the dictionary pair $\mathbf{D}_l, \mathbf{D}_h$ to be strictly aligned; in fact they do not even have to be of the same size. Thus we have no restrictions on the LR dictionary $\mathbf{D}_l$. More specifically, we utilize very small orthogonal dictionaries for the LR patches, which lead to a low cost scale-up scheme. To further enhance performance, we use data clustering and learn a set of prediction models. Furthermore, we design a multi-level scale-up scheme and explain how to naturally extend the suggested approach so that it will exploit maximally overlapping patches for image reconstruction. These extensions are shown to improve the outcome quality. Our contribution is twofold. First, our model goes beyond linear prediction models and thus offers more representation power. Second, the suggested single image SR scheme has the useful interpretation of a feedforward neural network (NN), leading to a very efficient implementation.

The rest of the paper is organized as follows. In Section II we formalize the single image SR problem. Section III presents the suggested model and the basic scale-up scheme. Training the parameters of the resulting neural net is discussed in Section IV. In Section V we describe how to construct the final scale-up scheme by data clustering and cascading several levels of the basic scheme. We evaluate the performance of our algorithm in Section VI, demonstrating its advantages over existing methods based on a LHR dictionary pair, both in visual appearance and numerical criteria, and comparing it with state-of-the-art methods for single image SR.

[1]The term 'support' refers to the locations of the non-zeros in the representation vector.

II. Problem Formulation

To formulate the single image SR problem, we first present several notations. The LR and HR images are represented as vectors $\mathbf{z}_l \in \mathbf{R}^{N_l}$ and $\mathbf{y}_h \in \mathbf{R}^{N_h}$, where $N_h = q^2 N_l$ and $q > 1$ is some integer scale-up factor. We further denote by $\mathbf{H} \in \mathbf{R}^{N_h \times N_h}$ the blur operator and by $\mathbf{Q} \in \mathbf{R}^{N_l \times N_h}$ the decimation operator for a factor q in each axis, which discards rows/columns from the input image. Two acquisition models are commonly used in the literature to describe how a LR image is generated from a HR image, and each of them has a different rationale. The first assumes that prior to decimation, a known low-pass (anti-aliasing) filter is applied on the image,

$$\mathbf{z}_l = \mathbf{QHy}_h + \mathbf{v}, \tag{1}$$

where $\mathbf{v}$ is an additive noise/error in the acquisition process. The corresponding problem of reconstructing $\mathbf{y}_h$ from $\mathbf{z}_l$ is also referred to in the literature as zooming deblurring [14]. The second acquisition model assumes that there is no blur prior to decimation, namely $\mathbf{z}_l = \mathbf{Qy}_h$, so that image reconstruction is cast as a pure interpolation problem – filling out the missing pixels between the original pixels in the input LR image, which remain unaltered in the recovered HR image.

In this work we shall consider only the first acquisition model, leading to the zooming deblurring setup. Note that various choices for the blur kernel (associated with the operator $\mathbf{H}$) exist in the literature for this setup. The two most popular choices are a bicubic low-pass filter[2] and a Gaussian low-pass filter. In our experiments we will focus on these two choices. Fig. 1 illustrates the absolute difference image between $\mathbf{y}_h$ and $\mathbf{y}_l$ for this setup with a bicubic filter. We can see that most of the energy in this image is concentrated at edges and textures.

Similar to previous approaches that work with sparse representations of patch-pairs, we would like to learn the correspondence between LHR patches of the same dimensions. Thus we apply bicubic interpolation[3] on the input LR image and obtain an image $\mathbf{y}_l \in \mathbf{R}^{N_h}$. Since we are addressing the zooming deblurring setup we aim at recovering the difference image $\mathbf{y}_{hl} = \mathbf{y}_h - \mathbf{y}_l$ and then apply $\hat{\mathbf{y}}_h = \hat{\mathbf{y}}_{hl} + \mathbf{y}_l$ to obtain the final recovery, thus we retain the LR detail and predict only the missing HR detail.

We now turn to describe the idea of patch-based image reconstruction. Let $\mathbf{p}^k = \mathbf{R}_k \mathbf{y}$ be an image patch of size $\sqrt{n} \times \sqrt{n}$ centered at location k and extracted from the image $\mathbf{y}$ of size N_h by the linear operator $\mathbf{R}_k$. In the sequel we shall suggest a local model which predicts a HR patch $\mathbf{p}_h^k = \mathbf{R}_k \mathbf{y}_h$ from a LR patch $\mathbf{p}_l^k = \mathbf{R}_k \mathbf{y}_l$. Once we have obtained all HR patch predictions, the HR image is recovered by averaging the overlapping recovered patches on their overlaps. Note that

[2]This is the filter used by the MATLAB function *imresize* for down-scaling an image with the option 'bicubic'.

[3]There is some ambiguity in the literature regarding the term *bicubic interpolation*. To be concise, for zooming deblurring the MATLAB function *imresize* is used, and for interpolation the function *interp2* is used.

Fig. 1. Different image resolutions – an illustration for part of the image 'Lena', scale factor $q = 3$ and a bicubic filter. From left to right: $\mathbf{y}_h$, $\mathbf{z}_l$, $\mathbf{y}_l$ and $|\mathbf{y}_{hl}|$ (the latter image is displayed in the dynamic range $[0, 60]$).

there is an inherent tradeoff between run time and the quality of image reconstruction in the choosing the size of overlap between adjacent patches. In image processing applications such as denoising and deblurring, patch-based approaches typically work with maximally-overlapping patches (an overlap of $\sqrt{n} \times (\sqrt{n}-1)$ pixels between adjacent patches in the horizontal and vertical directions) to achieve the best reconstruction quality. However, the decimation operator taking part in the image degradation process calls for some additional attention. Since decimation is invariant only to shifts of the form rq (in the horizontal or vertical directions), where r is some integer,[4] patches corresponding to such shifts undergo the same degradation process, whereas other patches are degraded in a different fashion. Therefore, we cannot expect one set of model parameters to predict the missing HR detail in all overlapping patches, but only for a set of patches whose locations oblige to the shift requirement mentioned above.

In light of this observation, it is assumed hereafter that we extract only patches for which the center pixel is located on the sampling grid – this choice leads to a set of partially-overlapping patches (an overlap of $\sqrt{n} \times (\sqrt{n} - q)$ pixels between adjacent patches in the horizontal and vertical directions). The corresponding set of locations $\{k\}$ is denoted by Ω. We shall revisit this point in Section V-C where we will explain how to extend the basic scheme so that it will exploit maximally overlapping patches for image reconstruction.

Finally, we briefly mention the sparsity-based synthesis model (for a comprehensive review see [25]). The core idea in this model is that a signal $\mathbf{x} \in \mathbf{R}^n$ can be represented as a linear combination of a *few* atoms (signal prototypes) taken from a dictionary $\mathbf{D} \in \mathbf{R}^{n \times m}$, namely $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha} + \boldsymbol{\eta}$, where $\boldsymbol{\alpha} \in \mathbf{R}^m$ is the sparse representation vector and $\boldsymbol{\eta}$ is noise or model error. Similar to previous approaches, we assume that each LR patch can be represented over a dictionary $\mathbf{D}_l \in \mathbf{R}^{n \times m_l}$ by a sparse vector $\boldsymbol{\alpha}_l \in \mathbf{R}^{m_l}$, and similarly a HR patch is represented over $\mathbf{D}_h \in \mathbf{R}^{n \times m_h}$ by $\boldsymbol{\alpha}_h \in \mathbf{R}^{m_h}$.

III. The Statistical Model and the Basic Single Image SR Scheme

The main motivation for the model we are about to suggest is the desire to predict for each LR patch the missing HR detail via a pair of LR and HR dictionaries with different number of atoms. This is mainly for two reasons: First, each dictionary aims at characterizing signals of different qualities, so it seems natural to use fewer atoms for the lower quality content. Second, working with a small and orthogonal (possible only in the complete and under-complete cases) dictionary for the LR patches allows for avoiding typical high-complexity sparse-coding computations, utilized in previous work. We begin by describing the low-cost pursuit stage for obtaining the LR coefficients $\boldsymbol{\alpha}_l$. We then suggest utilizing a statistical parametric model for predicting the HR representation vector $\boldsymbol{\alpha}_h$ of each patch from its corresponding LR coefficients $\boldsymbol{\alpha}_l$, and explain how to perform inference with the model, assuming its parameters are already known (through a training stage addressed in Section IV). Finally, we present the basic single image SR scheme resulting from the suggested model and inference.

A. Low-Cost Pursuit

We begin with the basic observation that LR patches have few detail, so that even an undercomplete dictionary ($m_l < n$) suffices to sparsely represent these patches. In light of this observation and to allow for a low-cost scale-up scheme, we assume hereafter that $\mathbf{D}_l$ is an *undercomplete orthonormal* dictionary. Therefore, the LR coefficients are readily computed by the inner products of the LR patch with the dictionary atoms,

$$\boldsymbol{\alpha}_l = (\mathbf{D}_l)^T \mathbf{p}_l. \tag{2}$$

Computing the LR coefficients for all overlapping patches $\left\{\mathbf{p}_l^k\right\}$ can be implemented using a convolutional network [26]. To enforce sparsity on these representations we perform hard-thresholding, so that the sparsity pattern $\mathbf{s}_l \in \{-1, 1\}^{m_l}$ is computed as

$$s_{l,j} = \begin{cases} 1, & |\alpha_{l,j}| > \lambda, \\ -1, & otherwise \end{cases} \quad \forall j = 1, \ldots, m_l, \tag{3}$$

where λ is set adaptively for each LR patch based on a residual error criterion. Specifically, λ is the maximal threshold satisfying

$$\sum_{j=1}^{m_l} \left|\alpha_{l,j}\right|^2 \mathbf{1}\left(\left|\alpha_{l,j}\right| \leq \lambda\right) \leq n\tau^2, \tag{4}$$

[4] This is in contrast to the blur operator, which is invariant to any integer shift (ignoring border affects).

where τ is some pre-specified parameter indicating the desired accuracy of the LR sparse representation.

B. The Model

As mentioned in the previous subsection, the LR dictionary $\mathbf{D}_l$ is an undercomplete orthogonal dictionary. The HR dictionary $\mathbf{D}_h$ however is assumed to be complete or overcomplete to allow for sufficient representation power. Due to the different number of atoms in $\mathbf{D}_l$ and $\mathbf{D}_h$, assuming the same sparsity pattern for the LR and HR representations as in all previous work that consider a dictionary pair [9], [10], [15], [23] is no longer a valid option. Therefore, a model that captures the relations between the two different sparsity patterns – $\mathbf{s}_l \in \{-1,1\}^{m_l}$ for the LR patch and $\mathbf{s}_h \in \{-1,1\}^{m_h}$ for the HR patch – is required.

Recently, the Boltzmann machine prior,

$$\Pr(\mathbf{s}) = \frac{1}{Z}\exp\left(\mathbf{b}^T\mathbf{s} + \frac{1}{2}\mathbf{s}^T\mathbf{W}\mathbf{s}\right), \tag{5}$$

where $\mathbf{b} \in \mathbf{R}^m$ is a bias vector and $\mathbf{W} \in \mathbf{R}^{m\times m}$ is an interaction matrix, has been used to capture statistical dependencies within the sparsity pattern $\mathbf{s} \in \{1,-1\}^m$ of a single representation vector [27]–[30]. In our setup we would like to capture the dependencies between the sparsity patterns of the LHR pair, so we turn to a variant of the BM, termed restricted Boltzmann machine (RBM) [31]–[33] and given by the conditional probability,

$$\begin{aligned}\Pr(\mathbf{s}_h|\mathbf{s}_l) &= \frac{1}{Z_1}\exp\left(\mathbf{b}_h^T\mathbf{s}_h + \mathbf{s}_h^T\mathbf{W}_{hl}\mathbf{s}_l\right)\\ &= \prod_{j=1}^{m_h}\Phi\left((b_{h,j} + \mathbf{w}_{hl,j}^T\mathbf{s}_l)\mathbf{s}_{h,j}\right),\end{aligned} \tag{6}$$

where $\mathbf{b}_h \in \mathbf{R}^{m_h}$ is a bias vector for the HR sparsity pattern, $\mathbf{W}_{hl} \in \mathbf{R}^{m_h\times m_l}$ is an interaction matrix connecting between the LR and HR sparsity patterns, and $\Phi(z) = (1+\exp(-2z))^{-1}$ is the well-known sigmoid function. The last equality in Eq. (6) holds since the entries of $\mathbf{s}_h$ are statistically independent given $\mathbf{s}_l$.

The RBM is a simple exponential model relating two binary vectors,[5] and leading to tractable inference problems. Specifically, this choice leads to a simple closed-form formula for the conditional marginal probability of each entry in $\mathbf{s}_h$ given $\mathbf{s}_l$,

$$\Pr(s_{h,j} = 1|\mathbf{s}_l) = \Phi\left(b_{h,j} + \mathbf{w}_{hl,j}^T\mathbf{s}_l\right), \quad \forall j = 1,\ldots,m_h, \tag{7}$$

which aligns with the well-known sigmoid unit in NNs. Next, we address the HR coefficients $\boldsymbol{\alpha}_h$. Given the sparsity pattern $\mathbf{s}_h$ and the LR coefficients $\boldsymbol{\alpha}_l$, we suggest the following model,

$$\alpha_{h,j} = \begin{cases} u_j, & s_{h,j} = 1,\\ 0, & s_{h,j} = -1\end{cases} \quad \forall j = 1,\ldots,m_h, \tag{8}$$

where $\mathbf{u} \in \mathbf{R}^{m_h}$ is assumed to be Gaussian distributed given $\boldsymbol{\alpha}_l$, so that $\mathbf{u}|\boldsymbol{\alpha}_l \sim N(\mathbf{C}_{hl}\boldsymbol{\alpha}_l, \boldsymbol{\Sigma}_{hl})$ with $\mathbf{C}_{hl} \in \mathbf{R}^{m_h\times m_l}$ and $\boldsymbol{\Sigma}_{hl} \in \mathbf{R}^{m_h\times m_h}$. Straightforward considerations lead to the following conditional expectation,

$$\mathrm{E}\left[\alpha_{h,j}|s_{h,j} = 1, \boldsymbol{\alpha}_l\right] = \mathbf{c}_{hl,j}^T\boldsymbol{\alpha}_l, \quad \forall j = 1,\ldots,m_h. \tag{9}$$

Note that Eq. (8) and (9) imply a different mapping from $\boldsymbol{\alpha}_l$ to $\boldsymbol{\alpha}_h$ for each sparsity pattern $\mathbf{s}_h$. However, all 2^{m_h} possible mappings are described through the same matrix – $\mathbf{C}_{hl}$. This is different from Jia et al. [19] which trains a different linear mapping for each sparsity pattern under consideration (there are $32,104$ such patterns in their setup). Notice that the prediction implied by our model is linear only when the sparsity pattern $\mathbf{s}_h$ is known, and as we will see in the next subsection, the final estimator for $\alpha_{h,j}$ given $\boldsymbol{\alpha}_l, \mathbf{s}_l$ is nonlinear.

C. Inference

Now that we have presented a statistical model for $\boldsymbol{\alpha}_h, \mathbf{s}_h$ given $\boldsymbol{\alpha}_l, \mathbf{s}_l$, we are ready to perform inference. We suggest using an MMSE estimator for predicting each entry in $\boldsymbol{\alpha}_h$ from $\mathbf{s}_l$ and $\boldsymbol{\alpha}_l$,

$$\begin{aligned}\hat{\alpha}_{h,j} &= \mathrm{E}\left[\alpha_{h,j}|\mathbf{s}_l, \boldsymbol{\alpha}_l\right] = \sum_{\mathbf{s}_h\in\Gamma_j}\mathrm{E}\left[\alpha_{h,j}|\mathbf{s}_h, \mathbf{s}_l, \boldsymbol{\alpha}_l\right]\Pr\left(\mathbf{s}_h|\mathbf{s}_l, \boldsymbol{\alpha}_l\right)\\ &= \sum_{\mathbf{s}_h\in\Gamma_j}\mathrm{E}\left[\alpha_{h,j}|s_{h,j} = 1, \boldsymbol{\alpha}_l\right]\Pr\left(\mathbf{s}_h|\mathbf{s}_l\right)\\ &= \mathrm{E}\left[\alpha_{h,j}|s_{h,j} = 1, \boldsymbol{\alpha}_l\right]\Pr\left(s_{h,j} = 1|\mathbf{s}_l\right)\\ &= \left(\mathbf{c}_{hl,j}^T\boldsymbol{\alpha}_l\right)\Phi\left(b_{h,j} + \mathbf{w}_{hl,j}^T\mathbf{s}_l\right),\end{aligned} \tag{10}$$

where $\Gamma_j = \left\{\boldsymbol{\gamma} \in \mathbf{R}^{m_h} : g_j = 1\right\}$ and we have used two independence assumptions for deriving the third equality:

1) $\alpha_{h,j} \perp \left(\mathbf{s}_l, \left\{s_{h,i}\right\}_{i\neq j}\right) \mid \left(s_{h,j} = 1, \boldsymbol{\alpha}_l\right), \quad \forall j$
2) $s_{h,j} \perp \boldsymbol{\alpha}_l \mid \mathbf{s}_l, \quad \forall j.$

Just as in other sparsity-based MMSE estimators [34], the resulting estimate for $\boldsymbol{\alpha}_h$ is not sparse. However, it leads to a better signal recovery compared to any sparse estimator. Note that in contrast to previous works that suggest linear mappings between the nonzeros in $\boldsymbol{\alpha}_h$ and $\boldsymbol{\alpha}_l$ [17], [19], [23], the estimator in Eq. (10) is constructed as a product of a linear term with respect to $\boldsymbol{\alpha}_l$ and a nonlinear term with respect to $\mathbf{s}_l$ – the sparsity pattern of $\boldsymbol{\alpha}_l$. This leads to a much richer family of predictors.

D. The Basic Scheme

The proposed scheme for single image SR is summarized in Algorithm 1. Fig. 2 depicts a signal flow diagram, demonstrating how this scheme can be interpreted as a feedforward NN, leading to a very fast and simple implementation. Patch extraction in line 4 and the inner products with the dictionary atoms in line 6 can be implemented using a convolutional network [26]. Turning to the local processing, line 7 can be implemented by a layer of hard-thresholding decision units and line 8 by an inner product of two outputs – one comes from a linear layer operating on $\boldsymbol{\alpha}_l$ and the second from a linear-then-sigmoid layers operating on $\mathbf{s}_l$. Line 9 constitutes as additional linear layers. Finally, averaging the local patch predictions on their overlaps for image reconstruction in line 11 is a close variant of the well-known average pooling layer. Note that

[5] Since RBM works with binary vectors, it is most natural to harness it for the supports, rather than the representation coefficients.

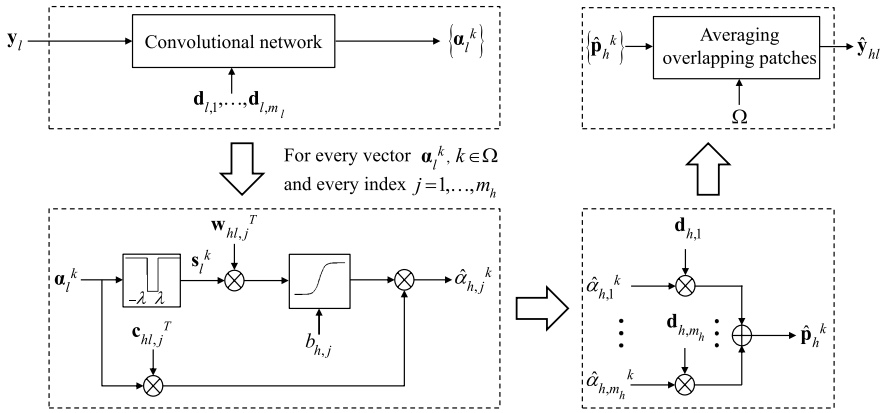

Fig. 2. The proposed single image SR basic scheme – interpretation as a feedforward NN: Top left – low-cost pursuit of LR representation coefficients. Bottom left – prediction of HR representation coefficients. Bottom right – synthesis of HR patches. Top right – reconstruction of the HR image.

Algorithm 1 Basic Scheme for Single Image SR

1: **Input:** LR image $\mathbf{z}_l$, scale-up factor q and model parameters $\mathbf{D}_l, \mathbf{D}_h, \mathbf{C}_{hl}, \mathbf{b}_h, \mathbf{W}_{hl}$
2: **Output:** Estimation for the HR image $\hat{\mathbf{y}}_h$
3: **Bicubic interpolation** with a scale-up factor q to generate the image $\mathbf{y}_l$ from $\mathbf{z}_l$
4: **Extract overlapping patches** $\left\{\mathbf{p}_l^k\right\}$ centered at locations $k \in \Omega$ from the image $\mathbf{y}_l$
5: **for** $k \in \Omega$ **do**
6: **Compute the LR representation** $\boldsymbol{\alpha}_l^k$ from the LR patch $\mathbf{p}_l^k$ using Eq. (2)
7: **Compute the LR sparsity pattern** $\mathbf{s}_l^k$ from the LR representation $\boldsymbol{\alpha}_l^k$ using Eq. (3)
8: **Compute the MMSE estimator for the HR representation** $\hat{\boldsymbol{\alpha}}_h^k$ using Eq. (10)
9: **HR patch recovery:** $\hat{\mathbf{p}}_h^k = \mathbf{D}_h \hat{\boldsymbol{\alpha}}_h^k$
10: **end for**
11: **Recover the LHR diff. image** $\hat{\mathbf{y}}_{hl}$ from the patches $\left\{\hat{\mathbf{p}}_h^k\right\}_{k\in\Omega}$ by averaging on their overlaps
12: **HR image recovery:** $\hat{\mathbf{y}}_h = \hat{\mathbf{y}}_{hl} + \mathbf{y}_l$

the architecture of the suggested neural network is well-reasoned as it follows directly from solid ideas such as efficient sparse coding of LR patches and a statistical prediction model for the HR representations. This is quite different from the common trend for determining the architecture of feedforward networks.

IV. Learning the Model Parameters

In Section III-D we described the feedforward neural network resulting from the suggested model and inference. In this section our goal is to find the network parameters leading to the best prediction from the LR patches to the corresponding HR ones, in terms of the squared error with respect to the HR patches. The suggested network consists of the following parameters,

$$\Theta = \left\{\mathbf{D}_l, \mathbf{D}_h, \mathbf{C}_{hl}, \mathbf{b}_h, \mathbf{W}_{hl}\right\}. \tag{11}$$

As in previous work on sparsity-based single image SR [9], [10], [15], [17], [19], we suggest learning the model parameters offline, using a set of LHR image pairs. Image patches are extracted at the same locations (indexed in the set $\boldsymbol{\Omega}$) from each image pair $\mathbf{y}_l, \mathbf{y}_{hl}$, resulting in a training set consisting of N paired LHR patches $\left\{\mathbf{p}_l^k, \mathbf{p}_h^k\right\}$.

Training the model parameters Θ can be formulated through the following optimization problem,

$$\underset{\Theta}{\text{Argmin}} \sum_{k=1}^{N} \left\| \mathbf{D}_h \left(\left[\Phi \left(\mathbf{b}_h + \mathbf{W}_{hl} \mathbf{s}_l^k \right) \right] \circ \left[\mathbf{C}_{hl} (\mathbf{D}_l)^T \mathbf{p}_l^k \right] \right) - \mathbf{p}_h^k \right\|_2^2, \tag{12}$$

where $\circ$ is the Hadamard product.

Learning all the model parameters together by solving the joint optimization problem of (12) seems to be very challenging; and thus, we looked for a way around it. To remove some of the complexity we start from both ends, setting initial estimates for the dictionaries $\mathbf{D}_l$, $\mathbf{D}_h$ using well-known approaches – directional PCAs [14], K-SVD [35]. Given these two estimates and knowing the true sparsity patterns for each LR and HR patch pair over these dictionaries, setting an initial estimate for the covariance matrix $\mathbf{C}_{hl}$ is quite straightforward by solving a least-squares problem. After these initial estimates have been made, we can update $\mathbf{D}_h$ and $\mathbf{C}_{hl}$ together, so that they we be well-tuned. At this point we reach the innermost layer of the network and we update the RBM parameters $\mathbf{W}_{hl}, \mathbf{b}_h$, while the rest of the parameters remain fixed to their current estimates. Finally, we perform another last tuning of the dictionary $\mathbf{D}_h$ so that it will work best (in terms of HR patch error) with the predicted representation vectors that are generated from the all the previous layers of the network. The overall learning algorithm is summarized in Algorithm 2. In the rest of this section we provide the justification and details for each of the steps in this algorithm.

We begin with the LR dictionary $\mathbf{D}_l$, assumed to be undercomplete and orthonormal. Note that there are two main practical reasons for working with undercomplete orthogonal dictionaries rather than complete ones: (i) reducing the overall amount of computations; and (ii) reducing the number of parameters in the matrices $\mathbf{W}_{hl}$, $\mathbf{C}_{hl}$ and thus avoiding possible

Algorithm 2 Learning the Parameters of the Basic Scheme

1: **Input:** A training set of paired LHR patches $\{\mathbf{p}_l^k, \mathbf{p}_h^k\}$ and an orthogonal dictionary $\mathbf{U}$
2: **Output:** Learned model parameters $\mathbf{D}_l, \mathbf{D}_h, \mathbf{C}_{hl}, \mathbf{b}_h, \mathbf{W}_{hl}$
3: **Set the LR dictionary** $\mathbf{D}_l$ as a subset of atoms from the given orthogonal dictionary $\mathbf{U}$, covering most of the energy of the LR patches $\{\mathbf{p}_l^k\}$.
4: **Set the LR representations and their sparsity patterns** $\{\boldsymbol{\alpha}_l^k, \mathbf{s}_l^k\}$ from the LR patches $\{\mathbf{p}_l^k\}$ and the estimate for $\mathbf{D}_l$ using Eq. (2)-(3).
5: **Initialize the HR dictionary** $\mathbf{D}_h$ from the HR patches $\{\mathbf{p}_h^k\}$ using the K-SVD algorithm.
6: **Compute the HR representations and their sparsity patterns** $\{\boldsymbol{\alpha}_h^k, \mathbf{s}_h^k\}$ from $\{\mathbf{p}_h^k\}$ and the current estimate for $\mathbf{D}_h$ using the OMP algorithm.
7: **Initialize the low-to-high prediction matrix** $\mathbf{C}_{hl}$ from $\{\boldsymbol{\alpha}_l^k, \boldsymbol{\alpha}_h^k, \mathbf{s}_h^k\}$ using Eq. (14).
8: **Update the HR dictionary and low-to-high prediction matrix** $\mathbf{D}_h, \mathbf{C}_{hl}$ from $\{\boldsymbol{\alpha}_l^k, \mathbf{s}_h^k, \mathbf{p}_h^k\}$ using Eq. (18).
9: **Initialize the RBM parameters** $\mathbf{b}_h, \mathbf{W}_{hl}$ from $\{\mathbf{s}_h^k\}$ using Eq. (21).
10: **Update the RBM parameters** $\mathbf{b}_h, \mathbf{W}_{hl}$ from $\{\boldsymbol{\alpha}_l^k, \mathbf{s}_l^k, \mathbf{p}_h^k\}$ and the current estimates for $\mathbf{D}_h, \mathbf{C}_{hl}$ using the conjugate gradient descent algorithm to approximate the solutions to Eq. (19).
11: **Compute the MMSE estimator for the HR representations** $\{\hat{\boldsymbol{\alpha}}_h^k\}$ from $\{\boldsymbol{\alpha}_l^k, \mathbf{s}_l^k\}$ and the current estimates for $\mathbf{D}_h, \mathbf{C}_{hl}, \mathbf{b}_h, \mathbf{W}_{hl}$ using Eq. (10).
12: **Update the HR dictionary** $\mathbf{D}_h$ from $\{\hat{\boldsymbol{\alpha}}_h^k, \mathbf{p}_h^k\}$ using Eq. (22).

over-fitting issues. We suggest choosing an orthogonal transform that is known to perform well on natural image patches, such as DCT or directional PCAs [14], and setting $\mathbf{D}_l$ to be a subset of atoms from the given unitary dictionary, covering most of the energy of the LR patches.

Next, we set initial estimates for $\mathbf{D}_h$ and $\mathbf{C}_{hl}$. The first is learned via the K-SVD algorithm, which finds a dictionary to sparsely represent the HR patches. Using the learned dictionary, we apply the OMP algorithm [36] to evaluate the target HR representations $\{\boldsymbol{\alpha}_h^k\}$ and their corresponding sparsity patterns $\{\mathbf{s}_h^k\}$. Then $\mathbf{C}_{hl}$ is set as the matrix describing best the linear mappings between the LHR nonzero coefficients,

$$\hat{\mathbf{C}}_{hl} = \underset{\mathbf{C}_{hl}}{\text{Argmin}} \sum_{k=1}^{N} \left\| \boldsymbol{\Lambda}_h^k \left(\mathbf{C}_{hl} \boldsymbol{\alpha}_l^k - \boldsymbol{\alpha}_h^k \right) \right\|_2^2, \tag{13}$$

where $\boldsymbol{\Lambda}_h^k = \text{diag}\left[\frac{1}{2}\left(\mathbf{s}_h^k + 1\right)\right]$. This is a least-square (LS) problem with a closed-form solution,

$$\left(\hat{\mathbf{C}}_{hl}\right)^{\text{cs}} = \left[\sum_{k=1}^{N} \left(\boldsymbol{\alpha}_l^k (\boldsymbol{\alpha}_l^k)^T \right) \otimes \boldsymbol{\Lambda}_h^k \right]^{-1} \left(\sum_{k=1}^{N} \boldsymbol{\Lambda}_h^k \boldsymbol{\alpha}_h^k (\boldsymbol{\alpha}_l^k)^T \right)^{\text{cs}}, \tag{14}$$

where $\otimes$ is the Kronecker product and $\mathbf{F}^{\text{cs}}$ is the column stacked version of a matrix $\mathbf{F}$. Note that the matrix inversion in Eq. (14) can be computed efficiently despite its large dimensions (this is a $m_l m_h \times m_l m_h$ matrix) due to its strong sparsity.

Next, we update the estimates for $\mathbf{D}_h$ and $\mathbf{C}_{hl}$ to work best together in terms of patch recovery,

$$\hat{\mathbf{D}}_h, \hat{\mathbf{C}}_{hl} = \underset{\mathbf{D}_h, \mathbf{C}_{hl}}{\text{Argmin}} \sum_{k=1}^{N} \left\| \mathbf{D}_h \boldsymbol{\Lambda}_h^k \mathbf{C}_{hl} \boldsymbol{\alpha}_l^k - \mathbf{p}_h^k \right\|_2^2. \tag{15}$$

Taking a similar approach as in the atom update rule of the K-SVD algorithm, we update one column in $\mathbf{D}_h$ and one row in $\mathbf{C}_{hl}$ at a time, using only 'relevant' examples,

$$\begin{aligned} \hat{\mathbf{d}}_{h,j}, \hat{\mathbf{c}}_{hl,j}^T = \underset{\mathbf{d}_{h,j}, \mathbf{c}_{hl,j}^T}{\text{Argmin}} & \left\| \mathbf{d}_{h,j} \mathbf{c}_{hl,j}^T \mathbf{A}_l^J - \mathbf{E}_h^J \right\|_F^2 \\ & \text{Subject To} \quad \|\mathbf{d}_{h,j}\|_2 = 1, \ \forall j = 1, \ldots, m_h, \end{aligned} \tag{16}$$

where $\mathbf{A}_l$ and $\mathbf{E}_h$ are matrices consisting of the vectors $\{\boldsymbol{\alpha}_l^k\}$ and $\{\mathbf{e}_h^k\}$ respectively in their columns,

$$\mathbf{e}_h^k = \mathbf{p}_h^k - \sum_{i \neq j, \ s_{h,i}^k = 1} \hat{\mathbf{d}}_{h,i} \left(\hat{\mathbf{c}}_{hl,i}^T \boldsymbol{\alpha}_l^k \right) \tag{17}$$

is the residual vector for the kth patch example, and $J = \{k \in \{1, \ldots, N\} : \ s_{h,j}^k = 1\}$ is the index set of 'relevant' examples. Each column in $\mathbf{D}_h$ is constrained to have a unit norm to avoid ambiguity and ensure uniqueness of the solution. Eq. (16) admits a closed-form solution,

$$\begin{aligned} \hat{\mathbf{d}}_{h,j} &= \text{maximal eigenvector of} \ \ \mathbf{E}_h^J (\mathbf{A}_l^J)^\dagger \mathbf{A}_l^J (\mathbf{E}_h^J)^T, \\ \hat{\mathbf{c}}_{hl,j}^T &= \hat{\mathbf{d}}_{h,j}^T \mathbf{E}_h^J (\mathbf{A}_l^J)^\dagger, \quad \forall j = 1, \ldots, m_h. \end{aligned} \tag{18}$$

Note that the matrix $\mathbf{E}_h$ consisting of the residual vectors is re-computed at each iteration, based on the most recent updates of $\mathbf{D}_h$ and $\mathbf{C}_{hl}$ (at the first iteration the initial estimates are utilized).

We now turn to estimate the parameters $\mathbf{b}_h, \mathbf{W}_{hl}$, which play their role in patch recovery through the MMSE formula of Eq. (10). Our goal is to find the parameters $\mathbf{b}_h, \mathbf{W}_{hl}$ that lead to the best performance in terms of patch recovery from the LR coefficients using the MMSE formula and the already estimated parameters $\mathbf{D}_h$ and $\mathbf{C}_{hl}$. As the MMSE estimator of the jth coefficient in $\boldsymbol{\alpha}_h$ depends only on $b_{h,j}, \mathbf{w}_{hl,j}^T$, we estimate $\mathbf{b}_h, \mathbf{W}_{hl}$ one element and one row at a time, respectively,

$$\begin{aligned} &\hat{b}_{h,j}, \hat{\mathbf{w}}_{hl,j}^T \\ &= \underset{b_{h,j}, \mathbf{w}_{hl,j}^T}{\text{Argmin}} \left\| \mathbf{d}_{h,j} \left(\Phi \left(b_{h,j} + \mathbf{w}_{hl,j}^T \mathbf{S}_l \right) \right) \circ \left(\mathbf{c}_{hl,j}^T \mathbf{A}_l \right) - \tilde{\mathbf{E}}_h \right\|_F^2, \\ &\qquad\qquad \forall j = 1, \ldots, m_h, \end{aligned} \tag{19}$$

where

$$\tilde{\mathbf{e}}_h^k = \mathbf{p}_h^k - \sum_{i \neq j} \hat{\mathbf{d}}_{h,i} \Phi \left(b_{h,j} + \mathbf{w}_{hl,j}^T \mathbf{s}_l^k \right) \left(\hat{\mathbf{c}}_{hl,i}^T \boldsymbol{\alpha}_l^k \right) \tag{20}$$

is the residual vector for the kth patch example. The optimization problem in Eq. (19) is convex and we approximate its

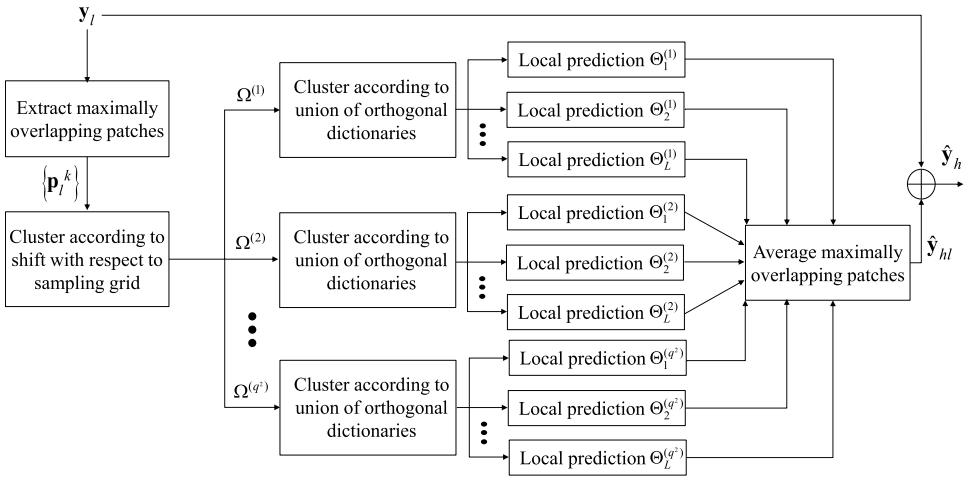

Fig. 3. One level of the proposed single image SR enhanced scheme. Each local prediction block applies lines 6-9 of Algorithm 1 on a different set of patches with a different set of model parameters that were trained using Algorithm 2.

solution using 20 iterations of the conjugate gradient descent algorithm, using the following initialization,

$$\hat{b}_{h,j} = \frac{1}{2}\ln\left(\frac{|J|}{N-|J|}\right), \quad \hat{\mathbf{w}}_{hl,j}^T = \mathbf{0}, \quad \forall j = 1,\ldots,m_h. \tag{21}$$

As before, the matrix $\tilde{\mathbf{E}}_h$, consisting of the vectors $\left\{\tilde{\mathbf{e}}_h^k\right\}$ in its columns, is updated at each iteration.

Now that the parameters $\mathbf{b}_h, \mathbf{W}_{hl}, \mathbf{C}_{hl}$ have been estimated, we can predict the HR coefficients from $\left\{\boldsymbol{\alpha}_l^k, \mathbf{s}_l^k\right\}$ using the MMSE formula in Eq. (10). Finally, we update $\mathbf{D}_h$ as the dictionary working best (in terms of patch recovery) with the predicted HR coefficients, using the same dictionary update rule as in [10],

$$\hat{\mathbf{D}}_h = \underset{\mathbf{D}_h}{\text{Argmin}} \left\|\mathbf{D}_h\hat{\mathbf{A}}_h - \mathbf{P}_h\right\|_F^2 = \mathbf{P}_h(\hat{\mathbf{A}}_h)^\dagger, \tag{22}$$

where $\hat{\mathbf{A}}_h, \mathbf{P}_h$ are matrices consisting of the vectors $\left\{\hat{\boldsymbol{\alpha}}_h^k\right\}, \{\mathbf{p}_h^k\}$ respectively in their columns.

V. Improvements to the Basic Scheme

To further enhance the performance of the suggested scheme, we suggest incorporating three additional principles which have been found to be useful in previous work. First, to allow for better representation capability, the single prediction model is replaced by a union of models and each LR patch is assigned to the model that fits it best, similar to previous work [14], [17], and [19]. Second, to stabilize the reconstruction process and reduce hallucination artifacts, we suggest a gradual increase in resolution. This is achieved through a multi-level scale-up scheme, where each level takes as input the output of the previous one and further improves its resolution. A methodology of the same spirit was practiced in [6]. However, as we explain in the sequel, our implementation is very different. Finally, we explain how to extend the suggested approach so that it will benefit from image reconstruction with maximally overlapping patches. Fig. 3 shows one level of the enhanced single image SR scheme. The multi-level scheme is constructed by cascading M such levels where the output of each level serves as an input to the next level.

A. Using a Union of Prediction Models

So far we have addressed both prediction and learning with a single model. To extend our work to a union of models, data clustering is essential. Once the data has been clustered, we can treat each cluster separately, thus we return to the single model setup. Instead of achieving the clustering through the widely used Gaussian Mixture Model (GMM), we suggest a union of sparsity-based models for the LR patches, based on orthogonal dictionaries. This model is closely related to the GMM, and generalizes it by incorporating sparsity directly into the representations. Specifically, we utilize a union of L orthogonal dictionaries $\left\{\mathbf{U}^j\right\}$, learned offline on a set of HR image patches, using an iterative algorithm like the one suggested by Yu et al. [14], which is initialized with a union of directional PCAs and a DCT dictionary.

Then for each LR patch $\mathbf{p}_l^k$ and each dictionary $\mathbf{U}^j$ we compute the following representation error,

$$\epsilon_j^k = \sum_{i\in I_j}\left|\alpha_{l,i}^{j,k}\right|^2, \tag{23}$$

where $\boldsymbol{\alpha}_l^{j,k} = \left(\mathbf{U}^j\right)^T\mathbf{p}_l^k$ and I_j is the index set of the 3 largest absolute values in $\boldsymbol{\alpha}_l^{j,k}$. Finally, we choose for each LR patch $\mathbf{p}_l^k$ the dictionary $\mathbf{U}^j$ with the lowest error ϵ_j^k, namely the one that best represents the patch $\mathbf{p}_l^k$ using only 3 atoms (with the largest coefficients). Note that $\mathbf{U}^j$ is also used for setting the LR dictionary $\mathbf{D}_l$ in the jth prediction model – as explained in Section IV the LR dictionary will consist of a subset of atoms from $\mathbf{U}^j$. Fig. 4 shows on the left the leading atoms in each of the L orthogonal dictionaries and demonstrates on the right the clustering of LR patches according to this union of dictionaries.

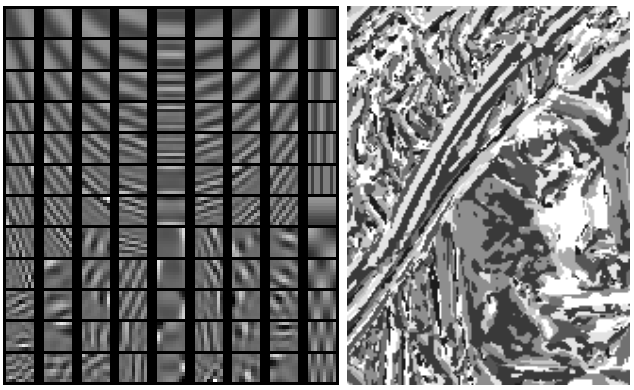

Fig. 4. Clustering the LR patches according to a union of $L = 9$ orthogonal dictionaries. Left - each column shows the 12 leading atoms in each of these dictionaries. Right - the cluster labels assigned to the center pixel of each LR patch for part of the image 'Lena' shown in Fig. 1.

B. Constructing a Multi-Level Scheme

Due to the great loss of detail resulting from the LR acquisition process, our suggested prediction model is prone to numerical instabilities. Therefore, we expect that using a multi-level scheme, each responsible for only part of the scale-up and taking into account the previous levels, will offer a more stable recovery. Glasner et al. [6] suggest constructing a cascade of M images corresponding to scale factors $1 < q_1 < q_2 < \cdots < q_M = q$ with respect to the input LR image $\mathbf{z}_l$, namely the jth image has q_j^2 more pixels than $\mathbf{z}_l$. Similarly, we consider an M-level scale-up scheme aimed at generating a cascade of images at increasing resolution. However, the images in our cascade are all of the same size in terms of pixel count, and the increase in resolution is essentially an increase in quality.

We suggest an M-level feedforward cascade taking as input the interpolated LR image $\mathbf{y}_l$, where each level is designed with the same form as our basic scale-up scheme, but trained to achieve a different goal. Specifically, in the training stage we generate from each HR image a cascade of images in increasing quality (the specific details depend on the blur kernel under consideration and will be provided in the sequel). The jth level takes as input the output of the previous one and is trained to minimize the patch recovery error with respect to the patches of the jth image in the aforementioned image cascade. As the level j increases (and so does the quality of the target patches), we use higher values for both m_h (the number of atoms in $\mathbf{D}_h$) and τ (determines the threshold λ in Eq. (3)), since more atoms are required to sparsely represent the patches at the output of the level and a higher threshold λ is required to obtain a sparse representation for the patches at the input of the level.

We now provide the details for generating a cascade of images in increasing quality. For the setup with a bicubic filter, we generate from the HR image an image pyramid consisting of M resolutions by downsampling it with scale factors $q/q_1 > q/q_2 > \cdots > q/q_M = 1$. All these images are then scaled-up again to the original size (HR) by bicubic interpolation.[6] For the setup with a Gaussian filter with standard deviation σ, we generate from the HR image a set of blurred images (no downsampling) using Gaussian filters with decreasing standard deviations $\sigma_1 = \sigma/q_1 > \sigma_2 = \sigma/q_2 > \ldots > \sigma_{M-1} = \sigma/q_M > \sigma_M = 0$. Specifically, a scale-up factor of $q = 2$ is achieved by the sequence $q_1 = 4/3, q_2 = 2$ and a scale-up factor of $q = 3$ is achieved by $q_1 = 1.5, q_2 = 2, q_3 = 2.5$, $q_4 = 3$.

[6]For down-scaling or bicubic interpolation with non-integer scale factors we use the MATLAB function *imresize*.

C. Exploiting Maximally Overlapping Patches

So far image reconstruction is performed by averaging the predictions for a set of partially overlapping patches on their overlaps. Specifically, there is an overlap of $\sqrt{n} \times (\sqrt{n} - q)$ pixels between adjacent patches (in the horizontal and vertical directions) in this set, so that we have at most n/q^2 estimates for a given pixel in the HR image. Working with maximally overlapping patches is much desired as it increases the number of estimates per pixel to n and thus allows for a more reliable reconstruction with fewer artifacts. Note however that such an extension evidently increases the computational complexity by a factor of q^2 as the number of patches to process is increased by the same factor.

In light of the discussion appearing in Section II we can deduce that working with maximally overlapping patches is possible as long as we train and apply the prediction model for patches undergoing a similar degradation model, namely patches with center pixels located at a given shift with respect to the sampling grid. Specifically, this means that for each level in our scheme we treat separately q^2 groups, each consisting of partially overlapping patches. Combining the patch predictions from all these groups yields a set of maximally overlapping patches, which in turn are utilized to generate the output image of the current level (see the rightmost block in Fig. 3).

Note that treating separately patches at different shifts with respect to the sampling grid has indeed been considered in previous work (see for example [7]). However, to the best of our knowledge, training a different sparsity-based prediction model for each of q^2 groups of LR patches, each undergoing a different degradation model, has not been considered before.

VI. Experiments

We consider 3 typical single image SR scenarios, all corresponding to a zooming deblurring setup with a known blur kernel.

1) A bicubic filter followed by downsampling by a scale factor of $q = 2$.
2) A bicubic filter followed by downsampling by a scale factor of $q = 3$.
3) A Gaussian filter of size 7×7 with standard deviation 1.6 followed by downsampling by a scale factor of $q = 3$.

For each scenario we train offline (on a set of natural images provided in [9]) both a single-level scheme and an M-level scheme, where each level corresponds to a union of $L = 9$ prediction models. The multi-level scheme is trained separately for reconstruction with partially and maximally overlapping patches (for the latter see Section V-C). Table I provides the design parameters used for each scenario:

TABLE I
SINGLE IMAGE SR, DESIGN PARAMETERS FOR THE DIFFERENT SCENARIOS

Scenario	n	M	m_h	τ
1) Bicubic filter, $q=2$	64 64	1 2	200 [100,200]	2 [2,3]
2) Bicubic filter, $q=3$	81 81	1 4	200 [81,120,160,200]	2 [2,2.5,3,3.5]
3) Gaussian filter, $\sigma=1.6$, $q=3$	81 81	1 5	200 [81,81,120,160,200]	2 [2,2,2.5,3,3.5]

TABLE II
SINGLE IMAGE SR, AVERAGE RUN TIMES (SECONDS PER 10^4 INPUT LR PIXELS)

Ours p/o 1-level	Ours p/o 2-level	Zeyde et al. [10]	Ours p/o 4-level	Ours p/o 5-level
0.8	1.4	1.9	2.8	3.4
Ours m/o 2-level	Ours m/o 4-level	Ours m/o 5-level	Yang et al. [9], $q=3$	Yang et al. [15], $q=2$
5.5	30.3	34.5	72.6	164.2
He et al. [23], $q=2$	ASDS-AR -NL [12]	ASDS-AR [12]	NCSR [22]	He et al. [23], $q=3$
322.6	511.4	580.5	855.1	1498.3

the patch dimension n, the number of levels M, the number of atoms m_h in $\mathbf{D}_h$, and the parameter τ for determining the threshold λ (the two latter parameters are given for a single level scheme and for each level of the multi-level scheme).

For each scenario the trained scale-up schemes are applied on a set of 10 distinct test images and their performance is compared with bicubic interpolation and with leading previous single image SR algorithms. For color images we apply the single image SR algorithm only on the luminance channel, so that the performance is evaluated only on that channel. The results of previous algorithms were obtained using the corresponding software packages that are publicly available.[7] All experiments were performed on an Intel Core i5-460M 2.53GHz laptop PC under MATLAB R2012a environment. This work does not address the pure interpolation setup, thus we cannot compare our approach with previous work that considered such a setup [3], [8], [14], [17], [21] since they consider only that setup.

We begin by comparing the average run times (seconds per 10^4 input LR pixels) for all the methods considered throughout this section – these are summarized in Table II. This table shows that our single level scheme is clearly the fastest method. Note that it runs at least 100 times faster than all previous methods apart from [10]. We can also see that the run time of our method increases linearly with the number of levels and with the number of overlapping patches taking part in the image reconstruction (this is determined by the size of the overlap).

Let us first focus on the two first scenarios, corresponding to zooming deblurring with a bicubic blur kernel. All previous work that considered this setup are based on sparse representations of LHR patch-pairs, assume full sparse representation invariance [9], [10], [15] or sparsity pattern invariance [23], and suggest to learn offline a local prediction model. Note that in most of these works [9], [15], [23] a global post-processing stage in the form of back-projection is performed on the intermediate result obtained via the local prediction model. To allow for a proper assessment of the contribution of the different local prediction models and the suggested post-processing we run two versions of these algorithms – with and without the back-projection. We also test the effect of this post-processing on our most elaborate method (multi-level, maximal overlaps) and on the method by Zeyde et al. We noticed that for the second scenario back-projection is not useful for these two methods. Thus, we show the effect of this post-processing on the two methods mentioned above only for the first scenario.

Tables III and IV compare the image reconstruction performance (image PSNR and SSIM index) of the various methods for these scenarios. We can see that when we do not allow any post-processing stages for all methods, our single-level scheme leads to superior or comparable performance with respect to previous methods based on a LHR dictionary pair with some sparsity invariance assumption [9], [10], [15], [23]. When performing the scale-up gradually via the suggested multi-level scheme we achieve an average gain of $0.25[dB]$ with respect to the single-level scheme. Exploiting maximal overlapping patches for image reconstruction in each of these levels leads to an additional average gain of $0.1[dB]$ for $q=2$ and $0.2[dB]$ for $q=3$. When including the post-processing stage, our best method outperforms Yang et al. [9] (for $q=3$) by an average gain of $0.36[dB]$ and Zeyde et al. [10] by $0.45dB$. Our best method achieves comparable performance to Yang et al. [15] (for $q=2$) and it is slightly inferior to He et al. [23] by an average loss of $0.17[dB]$, at a lower computational cost – it reduces the run time by a factor of 5.4 compared to [15] and by 10.6-49.4 compared to [23] (for $q=2$ and $q=3$ respectively).

Figs. 5 and 6 demonstrate that our multi-level scheme achieves results that are visually more pleasing compared to Zeyde et al. [10], Yang et al. [9] and the suggested single-level scheme. Specifically, we obtain sharper edges – see for example the diagonal lines at the background of the image 'Foreman' (Fig. 5) and the stripes at the hat of the image 'Lena' (Fig. 6). When comparing our best visual result with that of He et al. [23], we can see that we obtain sharper edges, but this comes at the price of having more artifacts due to larger prediction errors at the vicinity of some edges.

Turning to the third scenario, we compare our single and multi-level schemes to the method by Zeyde et al. [10], where we re-learned the dictionary pair to allow for adequate behavior for this specific scenario. We also compare to several methods by Dong et al. – ASDS-AR, ASDS-AR-NL [12] and NCSR [22]. All these three methods utilize sparse representations over PCAs adapted to the given image; the two latter methods also exploit non-local self-similarities between patches within the image to allow for better image recovery. The NCSR method leads to state-of-the-art performance in this scenario. Table V compares the image reconstruction performance (image PSNR and SSIM index) of the various methods

[7]We would like to thank the authors of [9], [10], [12], [15], [22], [23] for sharing their codes.

TABLE III

SINGLE IMAGE SR, SCENARIO NUMBER 1 – NUMERICAL RESULTS

Image	Butterfly	Comic	Flowers	Foreman	Girl	Lena	Man	Pepper	Starfish	Zebra	Avg.
Bicubic interp.	27.44 0.916	26.01 0.850	30.37 0.899	35.55 0.950	34.77 0.865	34.09 0.991	29.25 0.981	34.98 0.993	30.17 0.908	30.64 0.987	31.33 0.934
He et al. [23], no BP	29.87 0.946	27.43 0.893	31.96 0.922	37.39 0.958	35.15 0.880	35.11 0.991	30.21 0.986	35.88 0.992	32.05 0.933	31.90 0.990	32.70 0.949
Zeyde et al. [10]	30.06 0.949	27.44 0.893	32.02 0.924	37.79 0.962	35.46 0.883	35.77 **0.996**	30.30 0.992	36.46 **0.997**	31.81 0.932	32.98 **0.996**	33.01 0.952
Yang et al. [15], no BP	30.54 0.955	27.38 0.891	32.16 0.924	38.39 **0.963**	35.33 0.880	35.51 0.994	30.33 0.989	36.42 **0.995**	32.09 0.935	32.48 0.993	33.06 0.952
Ours p/o 1-level	30.22 0.949	27.68 0.899	32.16 0.927	37.73 0.961	**35.53** 0.883	35.95 **0.997**	30.41 **0.994**	36.40 **0.997**	32.23 0.937	32.89 **0.997**	33.12 0.954
Zeyde et al. [10] + BP	30.42 0.950	27.76 0.902	32.34 0.930	38.04 **0.964**	**35.66** **0.890**	36.09 **0.998**	30.50 **0.995**	**36.56** **0.998**	32.21 0.938	33.35 **0.998**	33.29 **0.956**
Ours p/o 2-level	30.64 0.953	27.92 0.904	32.43 0.929	38.08 **0.963**	**35.58** 0.884	**36.19** **0.997**	30.57 **0.994**	**36.65** **0.997**	32.60 0.940	33.16 **0.997**	33.38 **0.956**
Ours m/o 2-level	30.87 0.956	28.03 0.906	32.52 0.930	38.10 **0.963**	**35.59** 0.884	**36.24** **0.997**	**30.60** **0.994**	**36.71** **0.997**	32.81 **0.942**	33.24 **0.997**	33.47 **0.957**
Ours m/o 2-level + BP	31.00 **0.957**	**28.10** **0.909**	32.64 **0.933**	38.31 **0.965**	**35.72** **0.891**	**36.37** **0.998**	**30.67** **0.996**	**36.76** **0.998**	**32.99** **0.945**	**33.35** **0.998**	**33.59** **0.959**
Yang et al. [15]	**31.31** **0.960**	27.99 0.908	32.70 **0.933**	**38.90** **0.966**	**35.63** **0.889**	36.12 **0.997**	**30.68** **0.995**	**36.70** **0.997**	32.73 **0.943**	**33.35** **0.998**	**33.61** **0.959**
He at al. [23]	**31.51** **0.959**	**28.27** **0.912**	**32.92** **0.934**	**38.71** **0.965**	**35.69** **0.890**	**36.30** **0.997**	**30.78** **0.995**	**36.68** **0.997**	**33.17** **0.945**	**33.55** **0.997**	**33.76** **0.959**

In each cell 2 results are reported: Top – image PSNR [dB], Bottom – SSIM index. The methods are organized according to their average image PSNR (from the worse to the best). Note that BP stands for back-projection, p/o stands for partial overlaps, and m/o for maximal overlaps. PSNR/SSIM results that are less than $0.2[dB]$/0.003 (respectively) away from the best result for each image are highlighted.

TABLE IV

SINGLE IMAGE SR, SCENARIO NUMBER 2 – NUMERICAL RESULTS

Image	Butterfly	Comic	Flowers	Foreman	Girl	Lena	Man	Pepper	Starfish	Zebra	Avg.
Bicubic interp.	24.04 0.822	23.12 0.699	27.23 0.802	32.62 0.910	32.67 0.800	30.75 0.952	27.01 0.909	32.40 0.969	26.90 0.811	26.64 0.912	28.34 0.859
He et al. [23], no BP	25.73 0.881	23.90 0.759	28.15 0.836	34.01 0.922	32.88 0.818	31.48 0.959	27.69 0.928	33.04 0.969	27.74 0.844	27.67 0.928	29.23 0.884
Yang et al. [9], no BP	25.59 0.862	23.90 0.758	28.27 0.836	34.33 0.924	33.30 0.822	31.94 0.965	27.79 0.934	33.54 0.975	27.83 0.847	27.97 0.939	29.45 0.886
Ours p/o 1-level	25.89 0.871	24.04 0.761	28.49 0.839	34.58 0.929	**33.41** 0.821	32.31 **0.967**	27.91 0.935	33.85 **0.978**	28.09 0.849	28.42 **0.942**	29.70 0.889
Zeyde et al. [10]	25.97 0.878	23.97 0.756	28.44 0.839	34.68 0.930	**33.40** 0.822	32.21 **0.967**	27.91 0.934	34.08 **0.978**	27.95 0.847	28.52 0.941	29.71 0.889
Yang et al. [9]	26.05 0.868	24.11 0.764	28.54 0.841	34.77 0.927	**33.49** **0.826**	32.33 **0.968**	27.99 **0.938**	33.92 **0.978**	28.16 0.852	28.45 **0.944**	29.78 0.891
Ours p/o 4-level	26.38 0.885	24.13 0.767	28.67 0.844	35.00 0.931	**33.51** 0.824	32.55 **0.969**	28.08 0.937	34.13 **0.978**	28.32 0.855	28.52 **0.943**	29.93 0.893
Ours m/o 4-level	26.74 **0.900**	**24.31** 0.776	28.83 0.847	35.52 **0.936**	**33.52** 0.823	**32.66** **0.969**	**28.19** **0.939**	**34.36** **0.979**	**28.64** **0.863**	28.67 **0.943**	**30.14** **0.898**
He et al. [23]	**27.04** **0.903**	**24.45** **0.782**	**29.04** **0.853**	**35.75** **0.936**	**33.59** **0.829**	**32.76** **0.970**	**28.30** **0.941**	**34.53** **0.979**	**28.65** **0.864**	**29.03** **0.945**	**30.31** **0.900**

In each cell 2 results are reported: Top – image PSNR [dB], Bottom – SSIM index. The methods are organized according to their average image PSNR (from the worse to the best). Note that BP stands for back-projection, p/o stands for partial overlaps, and m/o for maximal overlaps. PSNR/SSIM results that are less than $0.2[dB]$/0.003 (respectively) away from the best result for each image are highlighted.

for this scenario. We can see that our single-level scheme outperforms Zeyde et al. [10] by an average gain of $0.23[dB]$, and the multi-level scheme with partial overlaps obtains an additional average gain of $0.44[dB]$. When exploiting maximal overlaps for image recovery we obtain a further average improvement of $0.18[dB]$, leading to an overall average gain of $0.85[dB]$ with respect to Zeyde et al. [10].

We now focus on our best result – multi-level, maximally overlapping patches, which will be referred to as "our approach" from this point on. Our approach is comparable in PSNR performance to ASDS-AR, while reducing the run time by a factor of 14.8. However, it is still $0.26[dB]$ (on average) away from ASDS-AR-NL and $0.54[dB]$ away from the state-of-the-art results of Dong et al. [22]. Note that our approach is much faster than the two leading methods mentioned above – it reduces the run time by a factor of 16.8 compared to ASDS-AR-NL and by 24.8 compared to NCSR. We can conclude that our suggested approach leads to a reconstruction quality that is as good as the leading method that does not exploit non-local self-similarities (ASDS-AR)

Fig. 5. Single image SR – visual results for the second scenario with part of the image 'Foreman'. Top (from left to right): reconstructed HR image by bicubic interpolation, Zeyde et al. [10], Yang et al. [9] and our single-level scheme. Bottom (from left to right): reconstructed HR image by our 4-level scheme with partially and maximally overlapping patches, and by He et al. [23], and ground truth HR.

Fig. 6. Single image SR – visual results for the second scenario with part of the image 'Foreman'. Top (from left to right): reconstructed HR image by bicubic interpolation, Zeyde et al. [10], Yang et al. [9] and our single-level scheme. Bottom (from left to right): reconstructed HR image by our 4-level scheme with partially and maximally overlapping patches, and by He et al. [23], and ground truth HR.

and offers a desirable compromise between low computational complexity and reconstruction quality, when comparing it to the state-of-the-art method in this scenario (NCSR).

Fig. 7 shows a visual comparison between the various methods for this scenario. We can see that the reconstructed HR images obtained by our approach are of higher visual

TABLE V
SINGLE IMAGE SR, SCENARIO NUMBER 3 – NUMERICAL RESULTS

Image	Butterfly	Comic	Flowers	Foreman	Girl	Lena	Man	Pepper	Starfish	Zebra	Avg.
Bicubic interp.	22.41 0.771	22.10 0.622	25.92 0.749	31.05 0.887	31.65 0.767	29.16 0.924	26.04 0.865	30.92 0.952	25.52 0.754	24.71 0.852	26.95 0.814
Zeyde et al. [10]	25.41 0.862	23.59 0.730	27.97 0.821	33.81 0.923	33.02 0.810	31.68 0.961	27.62 0.926	33.59 0.975	27.48 0.829	27.93 0.931	29.21 0.877
Ours p/o 1-level	25.58 0.858	23.91 0.751	28.32 0.834	33.86 0.924	33.21 0.820	32.15 0.967	27.80 0.934	33.49 **0.977**	27.90 0.843	28.19 0.940	29.44 0.885
Ours p/o 5-level	26.31 0.883	24.19 0.768	28.72 0.842	34.70 0.930	**33.44** 0.822	32.46 0.968	28.06 0.937	34.11 **0.978**	28.31 0.854	28.46 0.942	29.88 0.892
ASDS-AR [12]	26.54 0.887	24.22 0.769	28.78 0.843	35.21 0.932	33.32 0.816	32.79 0.969	28.10 0.936	34.13 **0.976**	28.45 0.857	28.72 0.943	30.03 0.893
Ours m/o 5-level	26.73 0.897	24.26 0.772	28.81 0.845	35.24 0.934	**33.42** 0.821	32.54 0.969	28.16 **0.938**	34.23 **0.979**	28.56 0.860	28.65 0.942	30.06 0.896
ASDS-AR -NL [12]	27.17 0.901	**24.51** **0.782**	29.04 0.849	35.27 0.933	**33.45** 0.818	**32.99** **0.970**	**28.30** **0.939**	34.39 **0.977**	**29.02** **0.870**	29.09 **0.946**	30.32 0.899
NCSR [22]	**28.15** **0.917**	**24.53** **0.785**	**29.18** **0.855**	**35.94** **0.940**	**33.60** **0.827**	**33.11** **0.973**	**28.39** **0.941**	**34.60** **0.977**	**29.12** **0.872**	**29.39** **0.948**	**30.60** **0.904**

In each cell 2 results are reported: Top – image PSNR [dB], Bottom – SSIM index. The methods are organized according to their average image PSNR (from the worse to the best). Note that p/o stands for partial overlaps and m/o for maximal overlaps. PSNR/SSIM results that are less than $0.2[dB]$/0.003 (respectively) away from the best result for each image are highlighted.

Fig. 7. Single image SR – visual results for the third scenario with part of the image 'Butterfly'. Top (from left to right): reconstructed HR image by bicubic interpolation, Zeyde et al. [10] and ASDS-AR [12]. Bottom (from left to right): reconstructed HR image by our 5-level scheme with maximally overlapping patches and NCSR [22], and ground truth HR.

quality compared to Zeyde et al. [10] and comparable to the one achieved by the ASDS-AR method. The NCSR method [22] offers yet another improvement in reconstruction quality, enjoying the benefits of non-local self-similarities. Nevertheless, the visual quality of our results is very pleasing – the reconstructed edges are sharp and there are few artifacts.

VII. CONCLUSION

In this work we have suggested a statistical prediction model based on sparse representations of LR and HR image patches for single image SR, which goes beyond the standard assumption of sparse representation invariance over a low and high resolution dictionary pair. Inference with our model leads to a low-complexity scale-up scheme that has the useful

interpretation of a feedforward neural network. We have found it constructive to utilize a union of prediction models and to perform the image scale-up gradually by a multi-level scheme. Our algorithm operates locally on overlapping image patches and outperforms previous methods based on local predictions with sparse representation invariance over a dictionary pair.

Neural networks have recently shown remarkable performance in image classification and object recognition applications. This is mainly due to the concept of deep learning [32], [33]. However, a similar impact of neural networks on image processing applications is yet to be seen. Recently, a neural network has been suggested for the problem of image denoising [37], showing excellent performance, even though deep learning was not utilized there. The desirable compromise between run time and recovery quality for our approach implies that neural networks can be the right choice for treating very challenging image processing applications, such as single image SR, and are not limited to denoising.

We believe that exploiting self-similarities within the image by incorporating a non-local term into our single image SR scheme can further enhance performance. The proposed approach for training the network does not exploit the full capabilities of current learning strategies in the machine learning community, and there is much room for improvement in this aspect. One immediate step would be utilizing the suggested approach as an initialization for the network and then performing a cross training of the different levels by standard back-propagation. Two other directions left for future research are taking into account prediction errors and applying a post-processing stage to further reduce ringing artifacts.

ACKNOWLEDGMENT

T. Peleg is grateful to the Azrieli Foundation for the award of an Azrieli Fellowship.

REFERENCES

[1] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Process. Mag.*, vol. 20, no. 3, pp. 21–36, May 2003.

[2] S. Farisu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *Int. J. Imag. Syst. Technol.*, vol. 14, no. 2, pp. 47–57, 2004.

[3] X. Zhang and X. Wu, "Image interpolation by adaptive 2D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.

[4] J. Sun, Z. Xu, and H. Shum, "Image super-resolution using gradient profile prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[5] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[6] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Oct. 2009, pp. 349–356.

[7] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Trans. Graph.*, vol. 30, no. 2, p. 12, 2011.

[8] S. Mallat and G. Yu, "Super-resolution with sparse mixing estimators," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2889–2900, Nov. 2010.

[9] 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.

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

[11] A. Adler, Y. Hel-Or, and M. Elad, "A shrinkage learning approach for single image super-resolution with overcomplete representations," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 622–635.

[12] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1838–1857, Jul. 2011.

[13] X. Gao, K. Zhang, D. Tao, and X. Li, "Joint learning for single-image super-resolution via a coupled constraint," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 469–480, Feb. 2012.

[14] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2481–2499, May 2012.

[15] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467–3478, Aug. 2012.

[16] K. Zhang, X. Gao, D. Tao, and X. Li, "Single image super-resolution with non-local means and steering kernel regression," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4544–4556, Nov. 2012.

[17] S. Wang, L. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2216–2223.

[18] K. Zhang, X. Gao, D. Tao, and X. Li, "Multi-scale dictionary for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1114–1121.

[19] K. Jia, X. Wang, and X. Tang, "Image transformation based on learning dictionaries across image spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 2, pp. 367–380, Feb. 2013.

[20] S. Hawe, M. Kleinsteuber, and K. Diepold, "Analysis operator learning and its application to image reconstruction," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2138–2150, Mar. 2013.

[21] W. Dong, L. Zhang, G. Shi, and X. Li, "Sparse representation based image interpolation with nonlocal autoregressive modeling," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1382–1394, Apr. 2013.

[22] W. Dong, L. Zhang, R. Lukac, and G. Shi, "Nonlocal centralized sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620–1630, Apr. 2013.

[23] L. He, H. Qi, and R. Zaretzki, "Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 345–352.

[24] J. Yang, Z. Lin, and S. Cohen, "Fast image super-resolution based on in-place example regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1059–1066.

[25] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, 2009.

[26] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech and time series," in *The Handbook of Brain Theory and Neural Networks*, M. A. Arbib, Ed. Cambridge, MA, USA: MIT Press, 1995, pp. 255–258.

[27] P. J. Wolfe, S. J. Godsill, and W. J. Ng, "Bayesian variable selection and regularization for time-frequency surface estimation," *J. R. Statist. Soc. B*, vol. 66, no. 3, pp. 575–589, 2004.

[28] P. J. Garrigues and B. A. Olshausen, "Learning horizontal connections in a sparse coding model of natural images," in *Proc. Adv. NIPS*, 2008, pp. 505–512.

[29] V. Cevher, M. F. Duarte, C. Hedge, and R. G. Baraniuk, "Sparse signal recovery using Markov random fields," in *Proc. Adv. NIPS*, 2009, pp. 257–264.

[30] T. Peleg, Y. C. Eldar, and M. Elad, "Exploiting statistical dependencies in sparse representations for signal recovery," *IEEE Trans. Signal Process.*, vol. 60, no. 5, pp. 2286–2303, May 2012.

[31] P. Smolensky, *Information Processing in Dynamical Systems: Foundations of Harmony Theory*. Cambridge, MA, USA: MIT Press, 1986.

[32] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[33] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[34] M. Elad and I. Yavneh, "A plurality of sparse representations is better than the sparsest one alone," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4701–4714, Oct. 2009.
[35] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
[36] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. Signals, Systems, Comput.*, 1993, pp. 40–44.
[37] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 4321–4328.

Tomer Peleg (S'12) received the B.Sc. degree in electrical engineering (*summa cum laude*) and the B.Sc. degree in physics (*summa cum laude*) from the Technion, Haifa, Israel, in 2009. He is currently pursuing the Ph.D. degree in electrical engineering from the Technion. From 2007 to 2009, he was with RAFAEL Research Laboratories, Israel Ministry of Defense. His current research interests include statistical signal processing, sparse representations, image processing, inverse problems, and graphical models. Since 2012, he holds a fellowship in the Azrieli program for outstanding Israeli graduate students.

Michael Elad (M'98–SM'08–F'12) received the B.Sc., M.Sc., and D.Sc. degrees from the Department of Electrical Engineering, Technion, Israel, in 1986, 1988, and 1997, respectively. Since 2003, he has been a faculty member with the Computer Science Department, Technion, and since 2010, he has held a full-professorship position.

Michael Elad works in the field of signal and image processing, specializing in particular on inverse problems, sparse representations, and super-resolution. Michael received the Technion's Best Lecturer Award six times, and he is the recipient of the 2007 Solomon Simon Mani Award for excellence in teaching, the 2008 Henri Taub Prize for academic excellence, and the 2010 Hershel-Rich prize for innovation. He is serving as an Associate Editor for SIAM SIIMS, the IEEE-TIT, and ACHA. Michael is serving as a Senior Editor for the IEEE SPL.