Image Super-Resolution via Sparse Representation

Jianchao Yang, *Student Member, IEEE,* John Wright, *Member, IEEE* Thomas Huang, *Life Fellow, IEEE* and Yi Ma, *Senior Member, IEEE*

***Abstract*—This paper presents a new approach to single-image superresolution, based on sparse signal representation. Research on image statistics suggests that image patches can be well-represented as a sparse linear combination of elements from an appropriately chosen over-complete dictionary. Inspired by this observation, we seek a sparse representation for each patch of the low-resolution input, and then use the coefficients of this representation to generate the high-resolution output. Theoretical results from compressed sensing suggest that under mild conditions, the sparse representation can be correctly recovered from the downsampled signals. By jointly training two dictionaries for the low- and high-resolution image patches, we can enforce the similarity of sparse representations between the low resolution and high resolution image patch pair with respect to their own dictionaries. Therefore, the sparse representation of a low resolution image patch can be applied with the high resolution image patch dictionary to generate a high resolution image patch. The learned dictionary pair is a more compact representation of the patch pairs, compared to previous approaches, which simply sample a large amount of image patch pairs [1], reducing the computational cost substantially. The effectiveness of such a sparsity prior is demonstrated for both general image super-resolution and the special case of face hallucination. In both cases, our algorithm generates high-resolution images that are competitive or even superior in quality to images produced by other similar SR methods. In addition, the local sparse modeling of our approach is naturally robust to noise, and therefore the proposed algorithm can handle super-resolution with noisy inputs in a more unified framework.**

***Index Terms*—Image super-resolution, sparse representation, sparse coding, face hallucination, non-negative matrix factorization.**

I. Introduction

Super-resolution (SR) image reconstruction is currently a very active area of research, as it offers the promise of overcoming some of the inherent resolution limitations of low-cost imaging sensors (e.g. cell phone or surveillance cameras) allowing better utilization of the growing capability of high-resolution displays (e.g. high-definition LCDs). Such resolution-enhancing technology may also prove to be essential in medical imaging and satellite imaging where diagnosis or analysis from low-quality images can be extremely difficult. Conventional approaches to generating a super-resolution image normally require as input *multiple* low-resolution images of the same scene, which are aligned with sub-pixel accuracy. The SR task is cast as the inverse problem of recovering the original high-resolution image by fusing the low-resolution images, based on reasonable assumptions or prior knowledge about the observation model that maps the high-resolution image to the low-resolution ones. The fundamental reconstruction constraint for SR is that the recovered image, after applying the same generation model, should reproduce the observed low-resolution images. However, SR image reconstruction is generally a severely ill-posed problem because of the insufficient number of low resolution images, ill-conditioned registration and unknown blurring operators, and the solution from the reconstruction constraint is not unique. Various regularization methods have been proposed to further stabilize the inversion of this ill-posed problem, such as [2], [3], [4].

However, the performance of these reconstruction-based super-resolution algorithms degrades rapidly when the desired magnification factor is large or the number of available input images is small. In these cases, the result may be overly smooth, lacking important high-frequency details [5]. Another class of SR approach is based on interpolation [6], [7], [8]. While simple interpolation methods such as Bilinear or Bicubic interpolation tend to generate overly smooth images with ringing and jagged artifacts, interpolation by exploiting the natural image priors will generally produce more favorable results. Dai *et al.* [7] represented the local image patches using the background/foreground descriptors and reconstructed the sharp discontinuity between the two. Sun *et. al.* [8] explored the gradient profile prior for local image structures and applied it to super-resolution. Such approaches are effective in preserving the edges in the zoomed image. However, they are limited in modeling the visual complexity of the real images. For natural images with fine textures or smooth shading, these approaches tend to produce watercolor-like artifacts.

A third category of SR approach is based on machine learning techniques, which attempt to capture the co-occurrence prior between low-resolution and high-resolution image patches. [9] proposed an example-based learning strategy that applies to generic images where the low-resolution to high-resolution prediction is learned via a Markov Random Field (MRF) solved by belief propagation. [10] extends this approach by using the Primal Sketch priors to enhance blurred edges, ridges and corners. Nevertheless, the above methods typically require enormous databases of millions of high-resolution and low-resolution patch pairs, and are therefore computationally intensive. [11] adopts the philosophy of Locally Linear Embedding (LLE) [12] from manifold learning, assuming similarity between the two manifolds in the high-resolution and the low-resolution patch spaces. Their algorithm maps the local geometry of the low-resolution patch space to

Jianchao Yang and Thomas Huang are with Beckman Institute, University of Illinois Urbana-Champaign, Urbana, IL 61801 USA (email: jyang29@ifp.uiuc.edu; huang@ifp.uiuc.edu). John Wright is with the Visual Computing Group, Microsoft Research Asia (email: jnwright@uiuc.edu). Yi Ma is with the Visual Computing Group, Microsoft Research Aisa, as well as Coordinated Science Laboratory, University of Illinois Urbana-Champaign, Urbana, IL 61801 USA (email: yima@uiuc.edu).

This work was supported in part by the U.S. Army Research Laboratory and the U.S. Army Research Office under grant number W911NF-09-1-0383. It was also supported by grants NSF IIS 08-49292, NSF ECCS 07-01676, and ONR N00014-09-1-0230.

the high-resolution one, generating high-resolution patch as a linear combination of neighbors. Using this strategy, more patch patterns can be represented using a smaller training database. However, using a fixed number K neighbors for reconstruction often results in blurring effects, due to over- or under-fitting. In our previous work [1], we proposed a method for adaptively choosing the most relevant reconstruction neighbors based on sparse coding, avoiding over- or under-fitting of [11] and producing superior results. However, sparse coding over a large sampled image patch database directly is too time-consuming.

While the mentioned approaches above were proposed for generic image super-resolution, specific image priors can be incorporated when tailored to SR applications for specific domains such as human faces. This *face hallucination* problem was addressed in the pioneering work of Baker and Kanade [13]. However, the gradient pyramid-based prediction introduced in [13] does not directly model the face prior, and the pixels are predicted individually, causing discontinuities and artifacts. Liu *et al.* [14] proposed a two-step statistical approach integrating the global PCA model and a local patch model. Although the algorithm yields good results, the holistic PCA model tends to yield results like the mean face and the probabilistic local patch model is complicated and computationally demanding. Wei Liu *et al.* [15] proposed a new approach based on TensorPatches and residue compensation. While this algorithm adds more details to the face, it also introduces more artifacts.

This paper focuses on the problem of recovering the super-resolution version of a given low-resolution image. Similar to the aforementioned learning-based methods, we will rely on patches from the input image. However, instead of working directly with the image patch pairs sampled from high- and low-resolution images [1], we learn a compact representation for these patch pairs to capture the co-occurrence prior, significantly improving the speed of the algorithm. Our approach is motivated by recent results in sparse signal representation, which suggest that the linear relationships among high-resolution signals can be accurately recovered from their low-dimensional projections [16], [17]. Although the super-resolution problem is very ill-posed, making precise recovery impossible, the image patch sparse representation demonstrates both effectiveness and robustness in regularizing the inverse problem.

a) Basic Ideas: To be more precise, let $\boldsymbol{D} \in \mathbb{R}^{n\times K}$ be an overcomplete dictionary of K atoms ($K > n$), and suppose a signal $\boldsymbol{x} \in \mathbb{R}^n$ can be represented as a sparse linear combination with respect to $\boldsymbol{D}$. That is, the signal $\boldsymbol{x}$ can be written as $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}_0$ where where $\boldsymbol{\alpha}_0 \in \mathbb{R}^K$ is a vector with very few ($\ll n$) nonzero entries. In practice, we might only observe a small set of measurements $\boldsymbol{y}$ of $\boldsymbol{x}$:

$$\boldsymbol{y} \doteq L\boldsymbol{x} = L\boldsymbol{D}\boldsymbol{\alpha}_0, \tag{1}$$

where $L \in \mathbb{R}^{k\times n}$ with $k < n$ is a projection matrix. In our super-resolution context, $\boldsymbol{x}$ is a high-resolution image (patch), while $\boldsymbol{y}$ is its low-resolution counter part (or features extracted from it). If the dictionary $\boldsymbol{D}$ is overcomplete, the equation $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$ is underdetermined for the unknown coefficients $\boldsymbol{\alpha}$. The equation $\boldsymbol{y} = L\boldsymbol{D}\boldsymbol{\alpha}$ is even more dramatically underdetermined. Nevertheless, under mild conditions, the sparsest solution $\boldsymbol{\alpha}_0$ to this equation will be unique. Furthermore, if $\boldsymbol{D}$ satisfies an appropriate near-isometry condition, then for a wide variety of matrices L, any sufficiently sparse linear representation of a high-resolution image patch $\boldsymbol{x}$ in terms of the $\boldsymbol{D}$ can be recovered (almost) perfectly from the low-resolution image patch [17], [18].[1] Fig. 1 shows an example that demonstrates the capabilities of our method derived from this principle. The image of the raccoon face is blurred and downsampled to half of its original size in both dimensions. Then we zoom the low-resolution image to the original size using the proposed method. Even for such a complicated texture, sparse representation recovers a visually appealing reconstruction of the original signal.

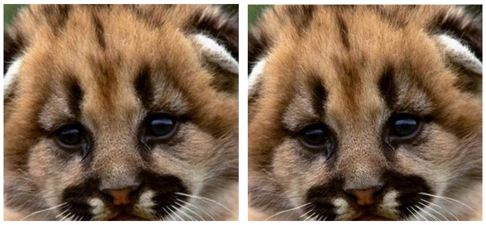

Fig. 1. Reconstruction of a raccoon face with magnification factor 2. Left: result by our method. Right: the original image. There is little noticeable difference visually even for such a complicated texture. The RMSE for the reconstructed image is 5.92 (only the local patch model is employed).

Recently sparse representation has been successfully applied to many other related inverse problems in image processing, such as denoising [19] and restoration [20], often improving on the state-of-the-art. For example in [19], the authors use the K-SVD algorithm [21] to learn an overcomplete dictionary from natural image patches and successfully apply it to the image denoising problem. In our setting, we do not directly compute the sparse representation of the high-resolution patch. Instead, we will work with two coupled dictionaries, $\boldsymbol{D}_h$ for high-resolution patches, and $\boldsymbol{D}_l$ for low-resolution ones. The sparse representation of a low-resolution patch in terms of $\boldsymbol{D}_l$ will be directly used to recover the corresponding high-resolution patch from $\boldsymbol{D}_h$. We obtain a locally consistent solution by allowing patches to overlap and demanding that the reconstructed high-resolution patches agree on the overlapped areas. In this paper, we try to learn the two overcomplete dictionaries in a probabilistic model similar to [22]. To enforce that the image patch pairs have the same sparse representations with respect to $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$, we learn the two dictionaries simultaneously by concatenating them with proper normalization. The learned compact dictionaries will be applied to both generic image super-resolution and face hallucination to demonstrate their effectiveness.

Compared with the aforementioned learning-based methods, our algorithm requires only two compact learned dictionaries, instead of a large training patch database. The computation, mainly based on linear programming or convex optimization,

[1] Even though the structured projection matrix defined by blurring and downsampling in our SR context does not guarantee exact recovery of $\boldsymbol{\alpha}_0$, empirical experiments indeed demonstrate the effectiveness of such a sparse prior for our SR tasks.

is much more efficient and scalable, compared with [9], [10], [11]. The online recovery of the sparse representation uses the low-resolution dictionary only – the high-resolution dictionary is used to calculate the final high-resolution image. The computed sparse representation adaptively selects the most relevant patch bases in the dictionary to best represent each patch of the given low-resolution image. This leads to superior performance, both qualitatively and quantitatively, compared to the method described in [11], which uses a fixed number of nearest neighbors, generating sharper edges and clearer textures. In addition, the sparse representation is robust to noise as suggested in [19], and thus our algorithm is more robust to noise in the test image, while most other methods cannot perform denoising and super-resolution simultaneously.

b) Organization of the Paper: The remainder of this paper is organized as follows. Section II details our formulation and solution to the image super-resolution problem based on sparse representation. Specifically, we study how to apply sparse representation for both generic image super-resolution and face hallucination. In Section III, we discuss how to learn the two dictionaries for the high- and low-resolution image patches respectively. Various experimental results in Section IV demonstrate the efficacy of sparsity as a prior for regularizing image super-resolution.

c) Notations: $\boldsymbol{X}$ and $\boldsymbol{Y}$ denote the high- and low-resolution images respectively, and $\boldsymbol{x}$ and $\boldsymbol{y}$ denote the high- and low-resolution image patches respectively. We use bold uppercase $\boldsymbol{D}$ to denote the dictionary for sparse coding, specifically we use $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$ to denote the dictionaries for high- and low-resolution image patches respectively. Bold lowercase letters denote vectors. Plain uppercase letters denote regular matrices, i.e., S is used as a downsampling operation in matrix form. Plain lowercase letters are used as scalars.

II. Image Super-Resolution from Sparsity

The single-image super-resolution problem asks: given a low-resolution image $\boldsymbol{Y}$, recover a higher-resolution image $\boldsymbol{X}$ of the same scene. Two constraints are modeled in this work to solve this ill-posed problem: 1) reconstruction constraint, which requires that the recovered $\boldsymbol{X}$ should be consistent with the input $\boldsymbol{Y}$ with respect to the image observation model; and 2) sparsity prior, which assumes that the high resolution patches can be sparsely represented in an appropriately chosen overcomplete dictionary, and that their sparse representations can be recovered from the low resolution observation.

1) Reconstruction constraint: The observed low-resolution image $\boldsymbol{Y}$ is a blurred and downsampled version of the high resolution image $\boldsymbol{X}$:

$$\boldsymbol{Y} = SH\boldsymbol{X} \tag{2}$$

Here, H represents a blurring filter, and S the downsampling operator.

Super-resolution remains extremely ill-posed, since for a given low-resolution input $\boldsymbol{Y}$, infinitely many high-resolution images $\boldsymbol{X}$ satisfy the above reconstruction constraint. We further regularize the problem via the following prior on small patches $\boldsymbol{x}$ of $\boldsymbol{X}$:

2) Sparsity prior: The patches $\boldsymbol{x}$ of the high-resolution image $\boldsymbol{X}$ can be represented as a sparse linear combination in a dictionary $\boldsymbol{D}_h$ trained from high-resolution patches sampled from training images:

$$\boldsymbol{x} \approx \boldsymbol{D}_h\boldsymbol{\alpha} \quad \text{for some } \boldsymbol{\alpha} \in \mathbb{R}^K \text{ with } \|\boldsymbol{\alpha}\|_0 \ll K. \tag{3}$$

The sparse representation $\boldsymbol{\alpha}$ will be recovered by representing patches $\boldsymbol{y}$ of the input image $\boldsymbol{Y}$, with respect to a low resolution dictionary $\boldsymbol{D}_l$ co-trained with $\boldsymbol{D}_h$. The dictionary training process will be discussed in Section III.

We apply our approach to both generic images and face images. For generic image super-resolution, we divide the problem into two steps. First, as suggested by the sparsity prior (3), we find the sparse representation for each local patch, respecting spatial compatibility between neighbors. Next, using the result from this local sparse representation, we further regularize and refine the entire image using the reconstruction constraint (2). In this strategy, a local model from the sparsity prior is used to recover lost high-frequency for local details. The global model from the reconstruction constraint is then applied to remove possible artifacts from the first step and make the image more consistent and natural. The face images differ from the generic images in that the face images have more regular structure and thus reconstruction constraints in the face subspace can be more effective. For face image super-resolution, we reverse the above two steps to make better use of the global face structure as a regularizer. We first find a suitable subspace for human faces, and apply the reconstruction constraints to recover a medium resolution image. We then recover the local details using the sparsity prior for image patches.

The remainder of this section is organized as follows: in Section II-A, we discuss super-resolution for generic images. We will introduce the local model based on sparse representation and global model based on reconstruction constraints. In Section II-B we discuss how to introduce the global face structure into this framework to achieve more accurate and visually appealing super-resolution for face images.

A. Generic Image Super-Resolution from Sparsity

1) Local model from sparse representation: Similar to the patch-based methods mentioned previously, our algorithm tries to infer the high-resolution image patch for each low-resolution image patch from the input. For this local model, we have two dictionaries $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$, which are trained to have the same sparse representations for each high-resolution and low-resolution image patch pair. We subtract the mean pixel value for each patch, so that the dictionary represents image textures rather than absolute intensities. In the recovery process, the mean value for each high-resolution image patch is then predicted by its low-resolution version.

For each input low-resolution patch $\boldsymbol{y}$, we find a sparse representation with respect to $\boldsymbol{D}_l$. The corresponding high-resolution patch bases $\boldsymbol{D}_h$ will be combined according to these coefficients to generate the output high-resolution patch $\boldsymbol{x}$. The problem of finding the sparsest representation of $\boldsymbol{y}$ can be formulated as:

$$\min \|\boldsymbol{\alpha}\|_0 \quad \text{s.t.} \quad \|F\boldsymbol{D}_l\boldsymbol{\alpha} - F\boldsymbol{y}\|_2^2 \leq \epsilon, \tag{4}$$

where F is a (linear) feature extraction operator. The main role of F in (4) is to provide a perceptually meaningful constraint[2] on how closely the coefficients $\boldsymbol{\alpha}$ must approximate $\boldsymbol{y}$. We will discuss the choice of F in Section III.

Although the optimization problem (4) is NP-hard in general, recent results [23], [24] suggest that as long as the desired coefficients $\boldsymbol{\alpha}$ are sufficiently sparse, they can be efficiently recovered by instead minimizing the ℓ^1-norm [3], as follows:

$$\min \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \|F\boldsymbol{D}_l\boldsymbol{\alpha} - F\boldsymbol{y}\|_2^2 \le \epsilon. \tag{5}$$

Lagrange multipliers offer an equivalent formulation

$$\min_{\boldsymbol{\alpha}} \ \|F\boldsymbol{D}_l\boldsymbol{\alpha} - F\boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1, \tag{6}$$

where the parameter λ balances sparsity of the solution and fidelity of the approximation to $\boldsymbol{y}$. Notice that this is essentially a linear regression regularized with ℓ^1-norm on the coefficients, known in statistical literature as the Lasso [27].

Solving (6) individually for each local patch does not guarantee the compatibility between adjacent patches. We enforce compatibility between adjacent patches using a one-pass algorithm similar to that of [28].[4] The patches are processed in raster-scan order in the image, from left to right and top to bottom. We modify (5) so that the super-resolution reconstruction $\boldsymbol{D}_h\boldsymbol{\alpha}$ of patch $\boldsymbol{y}$ is constrained to closely agree with the previously computed adjacent high-resolution patches. The resulting optimization problem is

$$\begin{aligned} \min \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad & \|F\boldsymbol{D}_l\boldsymbol{\alpha} - F\boldsymbol{y}\|_2^2 \le \epsilon_1, \\ & \|P\boldsymbol{D}_h\boldsymbol{\alpha} - \boldsymbol{w}\|_2^2 \le \epsilon_2, \end{aligned} \tag{7}$$

where the matrix P extracts the region of overlap between the current target patch and previously reconstructed high-resolution image, and $\boldsymbol{w}$ contains the values of the previously reconstructed high-resolution image on the overlap. The constrained optimization (7) can be similarly reformulated as:

$$\min_{\boldsymbol{\alpha}} \ \|\tilde{\boldsymbol{D}}\boldsymbol{\alpha} - \tilde{\boldsymbol{y}}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1, \tag{8}$$

where $\tilde{\boldsymbol{D}} = \begin{bmatrix} F\boldsymbol{D}_l \\ \beta P\boldsymbol{D}_h \end{bmatrix}$ and $\tilde{\boldsymbol{y}} = \begin{bmatrix} F\boldsymbol{y} \\ \beta\boldsymbol{w} \end{bmatrix}$. The parameter β controls the tradeoff between matching the low-resolution input and finding a high-resolution patch that is compatible with its neighbors. In all our experiments, we simply set $\beta = 1$. Given the optimal solution $\boldsymbol{\alpha}^*$ to (8), the high-resolution patch can be reconstructed as $\boldsymbol{x} = \boldsymbol{D}_h\boldsymbol{\alpha}^*$.

Algorithm 1 (Super-Resolution via Sparse Representation).

```
Input: training dictionaries $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$, a low-resolution image $\boldsymbol{Y}$.
For each $3 \times 3$ patch $\boldsymbol{y}$ of $\boldsymbol{Y}$, taken starting from the upper-left corner with 1 pixel overlap in each direction,
    • Compute the mean pixel value $m$ of patch $\boldsymbol{y}$.
    • Solve the optimization problem with $\tilde{\boldsymbol{D}}$ and $\tilde{\boldsymbol{y}}$ defined in (8): $\min_{\boldsymbol{\alpha}} \|\tilde{\boldsymbol{D}}\boldsymbol{\alpha} - \tilde{\boldsymbol{y}}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$.
    • Generate the high-resolution patch $\boldsymbol{x} = \boldsymbol{D}_h\boldsymbol{\alpha}^*$. Put the patch $\boldsymbol{x} + m$ into a high-resolution image $\boldsymbol{X}_0$.
End
Using gradient descent, find the closest image to $\boldsymbol{X}_0$ which satisfies the reconstruction constraint:
    $\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}} \|SH\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + c\|\boldsymbol{X} - \boldsymbol{X}_0\|_2^2.$
Output: super-resolution image $\boldsymbol{X}^*$.
```

2) Enforcing global reconstruction constraint: Notice that (5) and (7) do not demand exact equality between the low-resolution patch $\boldsymbol{y}$ and its reconstruction $\boldsymbol{D}_l\boldsymbol{\alpha}$. Because of this, and also because of noise, the high-resolution image $\boldsymbol{X}_0$ produced by the sparse representation approach of the previous section may not satisfy the reconstruction constraint (2) exactly. We eliminate this discrepancy by projecting $\boldsymbol{X}_0$ onto the solution space of $SH\boldsymbol{X} = \boldsymbol{Y}$, computing

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}} \|SH\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + c\|\boldsymbol{X} - \boldsymbol{X}_0\|_2^2. \tag{9}$$

The solution to this optimization problem can be efficiently computed using gradient descent. The update equation for this iterative method is

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t + \nu[H^T S^T(\boldsymbol{Y} - SH\boldsymbol{X}_t) + c(X - X_0)], \tag{10}$$

where $\boldsymbol{X}_t$ is the estimate of the high-resolution image after the t-th iteration, ν is the step size of the gradient descent.

We take result $\boldsymbol{X}^*$ from the above optimization as our final estimate of the high-resolution image. This image is as close as possible to the initial super-resolution $\boldsymbol{X}_0$ given by sparsity, while respecting the reconstruction constraint. The entire super-resolution process is summarized as Algorithm 1.

3) Global optimization interpretation: The simple SR algorithm outlined in the previous two subsections can be viewed as a special case of a more general sparse representation framework for inverse problems in image processing. Related ideas have been profitably applied in image compression, denoising [19], and restoration [20]. In addition to placing our work in a larger context, these connections suggest means of further improving the performance, at the cost of increased computational complexity.

Given sufficient computational resources, one could in principle solve for the coefficients associated with all patches *simultaneously*. Moreover, the entire high-resolution image $\boldsymbol{X}$ itself can be treated as a variable. Rather than demanding that $\boldsymbol{X}$ be perfectly reproduced by the sparse coefficients $\boldsymbol{\alpha}$, we can penalize the difference between $\boldsymbol{X}$ and the high-resolution image given by these coefficients, allowing solutions that

[2] Traditionally, one would seek the sparsest $\boldsymbol{\alpha}$ s.t. $\|\boldsymbol{D}_l\boldsymbol{\alpha} - \boldsymbol{y}\|_2 \le \epsilon$. For super-resolution, it is more appropriate to replace this 2-norm with a quadratic norm $\|\cdot\|_{F^TF}$ that penalizes visually salient high-frequency errors.

[3] There are also some recent works showing certain non-convex optimization problems can produce superior sparse solutions to the ℓ^1 convex problem, e.g., [25] and [26].

[4] There are different ways to enforce compatibility. In [11], the values in the overlapped regions are simply averaged, which will result in blurring effects. The greedy one-pass algorithm [28] is shown to work almost as well as the use of a full MRF model [9]. Our algorithm, not based on the MRF model, is essentially the same by trusting partially the previously recovered high resolution image patches in the overlapped regions.

are not perfectly sparse, but better satisfy the reconstruction constraints. This leads to a large optimization problem:

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X},\{\boldsymbol{\alpha}_{ij}\}} \Big\{ \|SH\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + \lambda \sum_{i,j} \|\boldsymbol{\alpha}_{ij}\|_0 + \gamma \sum_{i,j} \|\boldsymbol{D}_h\boldsymbol{\alpha}_{ij} - P_{ij}\boldsymbol{X}\|_2^2 + \tau\rho(\boldsymbol{X}) \Big\}. \tag{11}$$

Here, $\boldsymbol{\alpha}_{ij}$ denotes the representation coefficients for the $(i,j)_{th}$ patch of $\boldsymbol{X}$, and P_{ij} is a projection matrix that selects the $(i,j)_{th}$ patch from $\boldsymbol{X}$. $\rho(\boldsymbol{X})$ is a penalty function that encodes additional prior knowledge about the high-resolution image. This function may depend on the image category, or may take the form of a generic regularization term (e.g., Huber MRF, Total Variation, Bilateral Total Variation).

Algorithm 1 can be interpreted as a computationally efficient approximation to (11). The sparse representation step recovers the coefficients $\boldsymbol{\alpha}$ by approximately minimizing the sum of the second and third terms of (11). The sparsity term $\|\boldsymbol{\alpha}_{ij}\|_0$ is relaxed to $\|\boldsymbol{\alpha}_{ij}\|_1$, while the high-resolution fidelity term $\|\boldsymbol{D}_h\boldsymbol{\alpha}_{ij} - P_{ij}\boldsymbol{X}\|_2$ is approximated by its low-resolution version $\|F\boldsymbol{D}_l\boldsymbol{\alpha}_{ij} - F\boldsymbol{y}_{ij}\|_2$.

Notice, that if the sparse coefficients $\boldsymbol{\alpha}$ are fixed, the third term of (11) essentially penalizes the difference between the super-resolution image $\boldsymbol{X}$ and the reconstruction given by the coefficients: $\sum_{i,j} \|\boldsymbol{D}_h\boldsymbol{\alpha}_{ij} - P_{ij}\boldsymbol{X}\|_2^2 \approx \|\boldsymbol{X}_0 - \boldsymbol{X}\|_2^2$. Hence, for small γ, the back-projection step of Algorithm 1 approximately minimizes the sum of the first and third terms of (11).

Algorithm 1 does not, however, incorporate any prior besides sparsity of the representation coefficients – the term $\rho(\boldsymbol{X})$ is absent in our approximation. In Section IV we will see that sparsity in a relevant dictionary is a strong enough prior that we can already achieve good super-resolution performance. Nevertheless, in settings where further assumptions on the high-resolution signal are available, these priors can be incorperated into the global reconstruction step of our algorithm.

B. Face super-resolution from Sparsity

Face image resolution enhancement is usually desirable in many surveillance scenarios, where there is always a large distance between the camera and the objects (people) of interest. Unlike the generic image super-resolution discussed earlier, face images are more regular in structure and thus should be easier to handle. Indeed, for face super-resolution, we can deal with lower resolution input images. The basic idea is first to use the face prior to zoom the input to a reasonable medium resolution, and then to employ the local sparsity prior model to recover details. To be precise, the solution is also approached in two steps: 1) global model: use reconstruction constraint to recover a medium high-resolution face image, but the solution is searched only in the face subspace; and 2) local model: use the local sparse model to recover the image details.

a) Non-negative matrix factorization: In face super-resolution, the most frequently used subspace method for modeling the human face is Principal Component Analysis (PCA), which chooses a low-dimensional subspace that captures as much of the variance as possible. However, the PCA bases are holistic, and tend to generate smooth faces similar to the mean. Moreover, because principal component representations allow negative coefficients, the PCA reconstruction is often hard to interpret.

Even though faces are objects with lots of variance, they are made up of several relatively independent parts such as eyes, eyebrows, noses, mouths, checks and chins. Nonnegative Matrix Factorization (NMF) [29] seeks a representation of the given signals as an additive combination of local features. To find such a part-based subspace, NMF is formulated as the following optimization problem:

$$\begin{aligned} &\arg\min_{U,V} \|X - UV\|_2^2 \\ &s.t. \quad U \geq 0, V \geq 0, \end{aligned} \tag{12}$$

where $X \in \mathbb{R}^{n\times m}$ is the data matrix, $U \in \mathbb{R}^{n\times r}$ is the basis matrix and $V \in \mathbb{R}^{r\times m}$ is the coefficient matrix. In our context here, X simply consists of a set of pre-aligned high-resolution training face images as its column vectors. The number of the bases r can be chosen as $n*m/(n+m)$ which is smaller than n and m, meaning a more compact representation. It can be shown that a locally optimum of (12) can be obtained via the following update rules:

$$\begin{aligned} V_{ij} &\longleftarrow V_{ij} \frac{(U^T X)_{ij}}{(U^T UV)_{ij}} \\ U_{ki} &\longleftarrow U_{ki} \frac{(XV^T)_{ki}}{(UVV^T)_{ki}}, \end{aligned} \tag{13}$$

where $1 \leq i \leq r$, $1 \leq j \leq m$ and $1 \leq k \leq n$. The obtained basis matrix U is often sparse and localized.

b) Two step face super-resolution: Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ denote the high resolution and low resolution faces respectively. $\boldsymbol{Y}$ is obtained from $\boldsymbol{X}$ by smoothing and downsampling as in Eq. 2. We want to recover $\boldsymbol{X}$ from the observation $\boldsymbol{Y}$. In this paper, we assume $\boldsymbol{Y}$ has been pre-aligned to the training database by either manually labeling the feature points or with some automatic face alignment algorithm such as the method used in [14]. We can achieve the optimal solution for $\boldsymbol{X}$ based on the Maximum *a Posteriori* (MAP) criteria,

$$\boldsymbol{X}^* = \arg\max_{\boldsymbol{X}} p(\boldsymbol{Y}|\boldsymbol{X})p(\boldsymbol{X}). \tag{14}$$

$p(\boldsymbol{Y}|\boldsymbol{X})$ models the image observation process, usually with Gaussian noise assumption on the observation $\boldsymbol{Y}$, $p(\boldsymbol{Y}|\boldsymbol{X}) = 1/Z\exp(-\|SHU\boldsymbol{c} - \boldsymbol{Y}\|_2^2/(2*\sigma^2))$ with Z being a normalization factor. $p(\boldsymbol{X})$ is a prior on the underlying high resolution image $\boldsymbol{X}$, typically in the exponential form $p(\boldsymbol{X}) = \exp(-c\rho(\boldsymbol{X}))$. Using the rules in (13), we can obtain the basis matrix U, which is composed of sparse bases. Let Ω denote the face subspace spanned by U. Then in the subspace Ω, the super-resolution problem in (14) can be formulated using the reconstruction constraints as:

$$\boldsymbol{c}^* = \arg\min_{\boldsymbol{c}} \|SHU\boldsymbol{c} - \boldsymbol{Y}\|_2^2 + \eta\rho(U\boldsymbol{c}) \quad \text{s.t.} \quad \boldsymbol{c} \geq 0, \tag{15}$$

where $\rho(U\boldsymbol{c})$ is a prior term regularizing the high resolution solution, $\boldsymbol{c} \in \mathbb{R}^{r\times 1}$ is the coefficient vector in the subspace Ω

Algorithm 2 (Face Hallucination via Sparse Representation).

```
Input: sparse basis matrix $U$, training dictionaries $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$, a low-resolution aligned face image $\boldsymbol{Y}$.
Find a smooth high-resolution face $\hat{\boldsymbol{X}}$ from the subspace spanned by $U$ through:
    • Solve the optimization problem in (16): $\arg\min_{\boldsymbol{c}} \|SHU\boldsymbol{c} - \boldsymbol{Y}\|_2^2 + \eta\|\Gamma U\boldsymbol{c}\|_2 \quad s.t. \quad \boldsymbol{c} \geq 0$.
    • $\hat{\boldsymbol{X}} = U\boldsymbol{c}^*$.
For each patch $\boldsymbol{y}$ of $\hat{\boldsymbol{X}}$, taken starting from the upper-left corner with 1 pixel overlap in each direction,
    • Compute and record the mean pixel value of $\boldsymbol{y}$ as $m$.
    • Solve the optimization problem with $\tilde{\boldsymbol{D}}$ and $\tilde{\boldsymbol{y}}$ defined in (8): $\min_{\boldsymbol{\alpha}} \|\tilde{\boldsymbol{D}}\boldsymbol{\alpha} - \tilde{\boldsymbol{y}}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$.
    • Generate the high-resolution patch $\boldsymbol{x} = \boldsymbol{D}_h\boldsymbol{\alpha}^* + m$. Put the patch $\boldsymbol{x}$ into a high-resolution image $\boldsymbol{X}^*$.
Output: super-resolution face $\boldsymbol{X}^*$.
```

for estimated the high resolution face, and η is a parameter used to balance the reconstruction fidelity and the penalty of the prior term. In this paper, we simply use a generic image prior requiring that the solution be smooth. Let Γ denote a matrix performing high-pass filtering. The final formulation for (15) is:

$$\boldsymbol{c}^* = \arg\min_{\boldsymbol{c}} \|SHU\boldsymbol{c} - \boldsymbol{Y}\|_2^2 + \eta\|\Gamma U\boldsymbol{c}\|_2 \quad \text{s.t.} \quad \boldsymbol{c} \geq 0. \tag{16}$$

The medium high-resolution image $\hat{\boldsymbol{X}}$ is approximated by $U\boldsymbol{c}^*$. The prior term in (16) suppresses the high frequency components, resulting in over-smoothness in the solution image. We rectify this using the local patch model based on sparse representation mentioned earlier in Section II-A1. The complete framework of our algorithm is summarized as Algorithm 2.

III. Learning the Dictionary Pair

In the previous section, we discussed regularizing the super-resolution problem using sparse prior which states that each pair of high- and low-resolution image patches have the same sparse representations with respect to the two dictionaries $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$. A straightforward way to obtain two such dictionaries is to sample image patch pairs directly, which preserves the correspondence between the high resolution and low resolution patch items [1]. However, such a strategy will result in large dictionaries and hence expensive computation. This section will focus on learning a more compact dictionary pair for speeding up the computation.

A. Single Dictionary Training

Sparse coding is the problem of finding sparse representations of the signals with respect to an overcomplete dictionary $\boldsymbol{D}$. The dictionary is usually learned from a set of training examples $X = \{x_1, x_2, ..., x_t\}$. Generally, it is hard to learn a compact dictionary which guarantees that sparse representation of (4) can be recovered from ℓ_1 minimization in (5). Fortunately, many sparse coding algorithms proposed previously suffice for practical applications. In this work, we focus on the following formulation:

$$\begin{aligned} \boldsymbol{D} = \arg\min_{\boldsymbol{D},Z} &\|X - \boldsymbol{D}Z\|_2^2 + \lambda\|Z\|_1 \\ \text{s.t. } &\|D_i\|_2^2 \leq 1, i = 1, 2, ..., K. \end{aligned} \tag{17}$$

where the ℓ_1 norm $\|Z\|_1$ is to enforce sparsity, and the ℓ_2 norm constraints on the columns of $\boldsymbol{D}$ remove the scaling ambiguity [5]. This particular formulation has been studied extensively [30], [22], [31]. (17) is not convex in both $\boldsymbol{D}$ and Z, but is convex in one of them with the other fixed. The optimization performs in an alternative manner over Z and $\boldsymbol{D}$:

1) Initialize $\boldsymbol{D}$ with a Gaussian random matrix, with each column unit normalized.
2) Fix $\boldsymbol{D}$, update Z by
$$Z = \arg\min_{Z} \|X - \boldsymbol{D}Z\|_2^2 + \lambda\|Z\|_1, \tag{18}$$
which can be solved efficiently through linear programming.
3) Fix Z, update $\boldsymbol{D}$ by
$$\begin{aligned} \boldsymbol{D} = \arg\min_{\boldsymbol{D}} &\|X - \boldsymbol{D}Z\|_2^2 \\ \text{s.t.} &\|D_i\|_2^2 \leq 1, i = 1, 2, ..., K, \end{aligned} \tag{19}$$
which is a Quadratically Constrained Quadratic Programming that is ready to be solved in many optimization packages.
4) Iterate between 2) and 3) until converge. In our implementation, we used a Matlab package developed in [22].

B. Joint Dictionary Training

Given the sampled training image patch pairs $P = \{X^h, Y^l\}$, where $X^h = \{x_1, x_2, ..., x_n\}$ are the set of sampled high resolution image patches and $Y^l = \{y_1, y_2, ..., y_n\}$ are the corresponding low resolution image patches (or features), our goal is to learn dictionaries for high resolution and low resolution image patches, so that the sparse representation of the high resolution patch is the same as the sparse representation of the corresponding low resolution patch. This is a difficult problem, due to the ill-posed nature of super-resolution. The individual sparse coding problems in the high-resolution and low-resolution patch spaces are

$$\boldsymbol{D}_h = \arg\min_{\{\boldsymbol{D}_h, Z\}} \|X^h - \boldsymbol{D}_h Z\|_2^2 + \lambda\|Z\|_1, \tag{20}$$

and

$$\boldsymbol{D}_l = \arg\min_{\{\boldsymbol{D}_l, Z\}} \|Y^l - \boldsymbol{D}_l Z\|_2^2 + \lambda\|Z\|_1, \tag{21}$$

respectively. We combine these objectives, forcing the high-resolution and low-resolution representations to share the same codes, instead writing

$$\begin{aligned} \min_{\{\boldsymbol{D}_h, \boldsymbol{D}_l, Z\}} &\frac{1}{N}\|X^h - \boldsymbol{D}_h Z\|_2^2 + \frac{1}{M}\|Y^l - \boldsymbol{D}_l Z\|_2^2 \\ &+ \lambda(\frac{1}{N} + \frac{1}{M})\|Z\|_1, \end{aligned} \tag{22}$$

[5] Note that without the norm constraints the cost can always be reduced by dividing Z by $c > 1$ and multiplying $\boldsymbol{D}$ by $c > 1$.

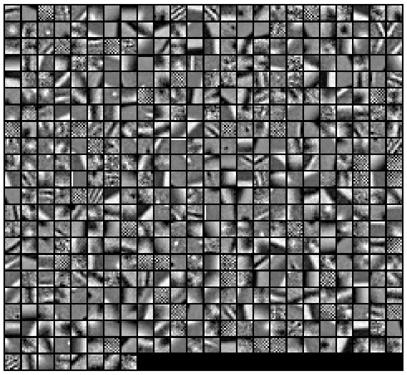

Fig. 2. The high resolution image patch dictionary trained by Eq. 24 using 100,000 high resolution and low resolution image patch pairs sampled from the generic training images. Totally 512 dictionary atoms are learned with each atom of size 9×9.

where N and M are the dimensions of the high resolution and low resolution image patches in vector form. Here, $1/N$ and $1/M$ balance the two cost terms of (20) and (21). (22) can be rewritten as

$$\min_{\{\boldsymbol{D}_h, \boldsymbol{D}_l, Z\}} \|X_c - \boldsymbol{D}_c Z\|_2^2 + \lambda(\frac{1}{N} + \frac{1}{M})\|Z\|_1, \tag{23}$$

or equivalently

$$\min_{\{\boldsymbol{D}_h, \boldsymbol{D}_l, Z\}} \|X_c - \boldsymbol{D}_c Z\|_2^2 + \hat{\lambda}\|Z\|_1, \tag{24}$$

where

$$X_c = \begin{bmatrix} \frac{1}{\sqrt{N}} X^h \\ \frac{1}{\sqrt{M}} Y^l \end{bmatrix}, \quad \boldsymbol{D}_c = \begin{bmatrix} \frac{1}{\sqrt{N}} \boldsymbol{D}_h \\ \frac{1}{\sqrt{M}} \boldsymbol{D}_l \end{bmatrix}. \tag{25}$$

Thus we can use the same learning strategy in the single dictionary case for training the two dictionaries for our super-resolution purpose. Note that since we are using features from the low resolution image patches, $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$ are not simply connected by a linear transform, otherwise the training process of (24) will depend on the high resolution image patches only (for detail, refer to Section III-C). Fig. 2 shows the dictionary learned by (24) for generic images.[6] The learned dictionary demonstrates basic patterns of the image patches, such as orientated edges, instead of raw patch prototypes, due to its compactness.

C. Feature Representation for Low Resolution Image Patches

In (4), we use a feature transformation F to ensure that the computed coefficients fit the most relevant part of the low-resolution signal, and hence have a more accurate prediction for the high resolution image patch reconstruction. Typically, F is chosen as some kind of high-pass filter. This is reasonable from a perceptual viewpoint, since people are more sensitive to the high-frequency content of the image. The high-frequency components of the low-resolution image are also arguably the most important for predicting the lost high-frequency content in the target high-resolution image.

In the literature, people have suggested extracting different features for the low resolution image patch in order to boost the prediction accuracy. Freeman et al. [9] used a high-pass filter to extract the edge information from the low-resolution input patches as the feature. Sun et. al. [10] used a set of Gaussian derivative filters to extract the contours in the low-resolution patches. Chang et. al. [11] used the first- and second-order gradients of the patches as the representation. In this paper, we also use the first- and second-order derivatives as the feature for the low-resolution patch due to their simplicity and effectiveness. The four 1-D filters used to extract the derivatives are:

$$\begin{aligned} \boldsymbol{f}_1 &= [-1, 0, 1], & \boldsymbol{f}_2 &= \boldsymbol{f}_1^T, \\ \boldsymbol{f}_3 &= [1, 0, -2, 0, 1], & \boldsymbol{f}_4 &= \boldsymbol{f}_3^T, \end{aligned} \tag{26}$$

where the superscript "T" means transpose. Applying these four filters yields four feature vectors for each patch, which are concatenated into one vector as the final representation of the low-resolution patch. In our implementation, the four filters are not applied directly to the sampled low resolution image patches; instead, we apply the four filters to the training images. Thus, for each low resolution training image, we get four gradient maps, and we extract fours patches from these gradient maps at each location, and concatenate them to become the feature vector. Therefore, the feature representation for each low resolution image patch also encodes its neighboring information, which is beneficial for promoting compatibility among adjacent patches in the final super-resolution image.

In practice, we find that it works better to extract the features from the upsampled version of the low-resolution image instead of the original one. That is, we first upsample the low resolution image by factor of two [7] using Bicubic interpolation, and then extract gradient features from it. Since we know all the zoom ratios, it is easy to track the correspondence between high resolution image patches and the upsampled low resolution image patches both for training and testing. Because of the way of extracting features from the low resolution image patches, the two dictionaries $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$ are not simply linearly connected, making the joint learning process in Eq. 24 more reasonable.

IV. Experimental Results

In this section, we first demonstrate the super-resolution results obtained by applying the above methods on both generic and face images. We then move on to discuss various influential factors for the proposed algorithm including dictionary size, noise with inputs, and the global reconstruction constraints.

In our experiments, we magnify the input low resolution image by a factor of 3 for generic images and 4 for face images, which is commonplace in the literature of single frame super-resolution. In generic image super-resolution, for the low-resolution images, we always use 3×3 low-resolution

[6] We omit the dictionary for the low resolution image patches because we are training on features instead the patches themselves.

[7] We choose 2 mainly for dimension considerations. For example, if we work on 3-by-3 patches in the low resolution image, by upsampling the image by ratio of 2, the final feature for the 9 dimensional low resolution patch will be $6 \times 6 \times 4 = 144$.

patches (upsampled to 6×6), with overlap of 1 pixel between adjacent patches, corresponding to 9×9 patches with overlap of 3 pixels for the high-resolution patches. In face super-resolution, we choose the patch size as 5×5 pixels for both low- and high-resolution face images. For color images, we apply our algorithm to the illuminance channel only, since humans are more sensitive to illuminance changes. We therefore interpolate the color layers (Cb, Cr) using plain Bicubic interpolation. We evaluate the results of various methods both visually and qualitatively in Root Mean Square Error (RMSE). Even though RMSE is a common criterion in image processing for recovery, it is not quite reliable for rating visual image quality [32], as we will see in the following parts. Note that since we only work on illuminance channel, the RMSE reported is carried out only on the illuminance channel.

A. Single Image Super-Resolution

1) Generic image super-resolution: We apply our methods to generic images such as flowers, human faces and architectures. The two dictionaries for high resolution and low resolution image patches are trained from 100,000 patch pairs randomly sampled from natural images collected from the internet. We preprocess these images by cropping out the textured regions and discard the smooth parts [8]. Unless otherwise explicitly stated, we always fix the dictionary size as 1024 in all our experiments, which is a balance between computation and image quality (Section IV-B will examine the effects of different dictionary sizes). In the super-resolution algorithm Eq. 8, the choice of λ depends on the level of noise in the input image, which we will discuss further in Section IV-C. For generic low-noise images, we always set $\lambda = 0.1$ in all our experiments, which generally yields satisfactory results.

Fig. 3 and 4 compare the outputs of our method with those of the neighborhood embedding (NE) [11]. The NE method is similar to ours in the sense that both methods use the linear combination weights derived from the low resolution image patch to generate the underlying high resolution image patch. Unlike our method, the NE method uses fixed k nearest neighbors to find the reconstruction supports directly from sampled training patches and does not including a dictionary training phase. To make a fair comparison, we use the same 100,000 patch pairs for the NE method and try different $k's$ to get the most visually appealing results. Using a compact dictionary pair, our method is much faster and yet generates shaper results. As the reconstructed images show in Fig. 3 and 4, there are noticeable differences in the texture of the leaves: the fuzz on the leaf stalk, and also the freckles on the face of the girl. In the captions of both figures, we list the RMSEs in parentheses following each method. As seen, our method can achieve lower RMSE than both Bicubic interpolation and NE. An interesting observation is that, although NE generates visually more appealing images than Bicubic, its RMSE is actually higher than Bicubic, indicating that RMSE is not a reliable criterion for visual image quality.

In Figure 5, we compare our method with several more state-of-the-art methods on an image of the Parthenon used in [7], including back projection (BP) [33], NE [11], and the recently proposed method based on a learned soft edge prior (SE) [7]. The result from back projection has many jagged effects along the edges. NE generates sharp edges in places, but blurs the texture on the temple's facade. The SE method gives a decent reconstruction, but introduces undesired smoothing that is not present in our result. We also give the RMSEs for all the methods in the followed parentheses in the caption. Again, besides best visual quality, our method achieves the lowest RMSE among these methods as well.

2) Face super-resolution: In this part, we evaluate our proposed super-resolution algorithm on frontal views of human faces. The experiments are conducted on the face database FRGC Ver 1.0 [34]. All these high resolution face images were aligned by an automatic alignment algorithm using the eye positions, and then cropped to the size of 100×100 pixels. To obtain the face subspace Ω spanned by W, we select 540 face images as training, covering both genders, different races, varying ages and different facial expressions (Figure 6). These high resolution face images are blurred and downsampled to 25×25 pixels to form the low-resolution counterparts. To prepare the coupled dictionaries needed for our sparse representation algorithm, we also sample 100,000 patch pairs from the training images and learn the dictionaries of size 1024. 30 new face images (from people not in the training set) are chosen as our test cases, which are blurred and downsampled to the size of 25×25 pixels in the same procedure as preparing the training set. These low-resolution input faces are aligned by manually labeling the eyeball positions.

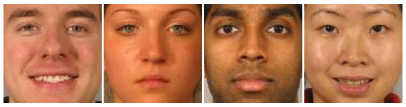

Fig. 6. Example training faces for the face super-resolution algorithm. The training images cover faces of both genders, different ages, different races and various facial expressions.

As mentioned earlier, face image super-resolution can handle more challenging tasks than generic image super-resolution due to the regular face structure. Indeed, it is not an easy job to zoom the 25×25 low resolution face image by 4 times using the method for generic image super-resolution. First, the downsampling process loses so much information that it is difficult to predict well a 12×12 high resolution patch given only a 3×3 image patch. Second, the resolution of the face image is so low that the structures of the face that are useful for super-resolution inference (such as corners and edges) collapses into only a couple of pixels. The two-step approach for face super-resolution, on the other hand, can compensate for the lost information in the first step using the redundancy of the face structures by searching the solution in the face subspace respecting the reconstruction constraints. The local model from sparse representation then can be further employed to enhance the edges and textures to achieve shaper results.

[8] Other authors prepare the training patches by extracting the image edges and sample patches around the edge regions to get the patch primitives.

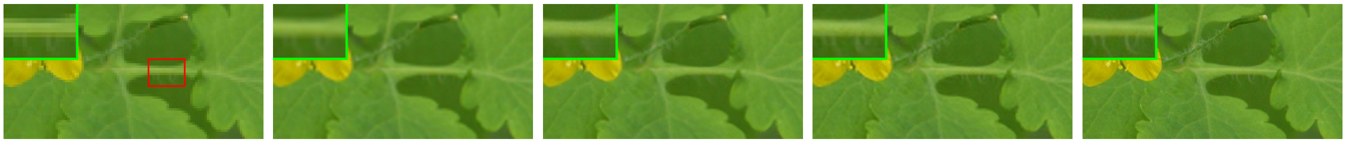

Fig. 3. Results of the flower image magnified by a factor of 3 and the corresponding RMSEs. Left to right: input, Bicubic interpolation (RMSE: 4.066), NE [11] (RMSE: 4.888), our method (RMSE: **3.761**), and the original.

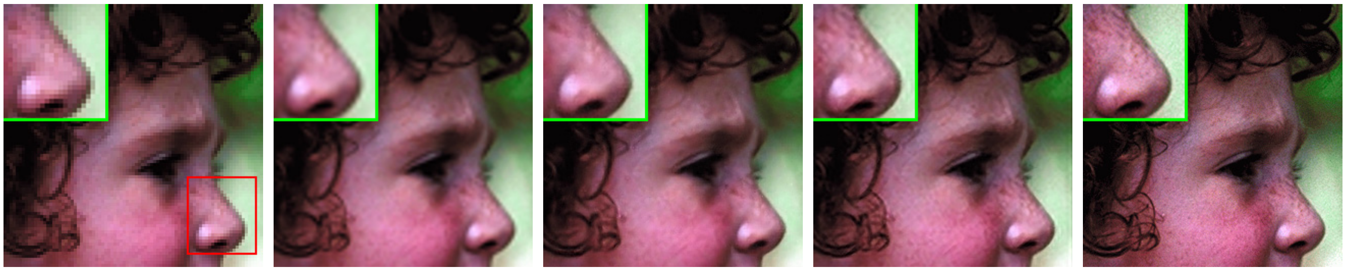

Fig. 4. Results of the girl image magnified by a factor of 3 and the corresponding RMSEs. Left to right: input, Bicubic interpolation (RMSE: 6.843), NE [11] (RMSE: 7.740), our method (RMSE: **6.525**), and the original.

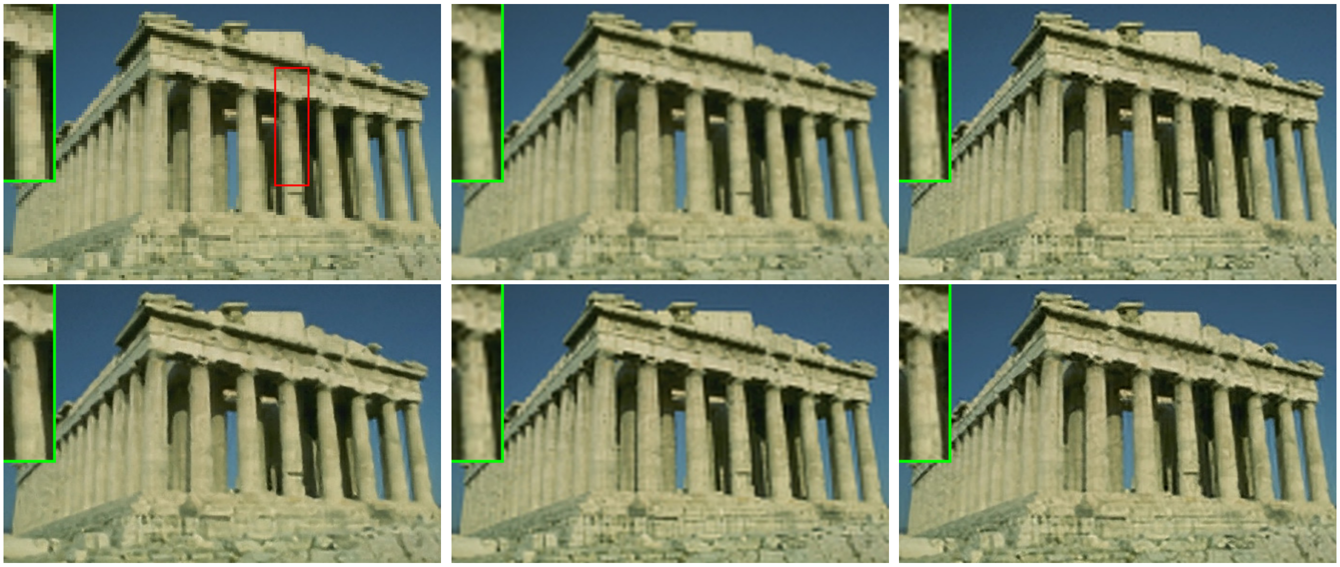

Fig. 5. Results on an image of the Parthenon with magnification factor 3 and corresponding RMSEs. Top row: low-resolution input, Bicubic interpolation (RMSE: 12.724), BP (RMSE: 12.131). Bottom row: NE (RMSE: 13.556), SE [7] (RMSE: 12.228), and our method (RMSE: **11.817**).

In Fig. 7, we compare the proposed two-step approach with the direct sparse representation method for generic images. Since the resolution of the input face image is so low, a direct application of the generic approach does not seem to generate satisfying results.

In our experiments with face images, we also set $\lambda = 0.1$ for sparsity regularization. We compare our algorithm with Bicubic interpolation [6] and BP [33]. The results are shown in Fig. 8, which indicate that our method can generate much higher resolution faces. Column 4 shows the intermediate results from the NMF global modeling and column 5 demonstrates the results after local sparse modeling. Comparing the two columns, the local sparse modeling further enhances the edges and textures , and also reduces RMSE.

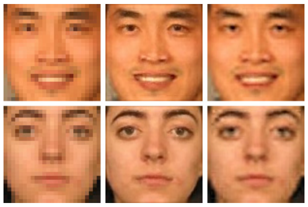

Fig. 7. The comparison between the two-step face super-resolution algorithm with the generic image super-resolution algorithm applied to low resolution face images. From left to right: input image, super-resolution result using the two step approach, and super-resolution result using the generic approach. The two-step face super-resolution algorithm generates visually much better results.

From columns 4 and 5, we can also see that the local patch method based on sparse representation further enhances the edges and textures.

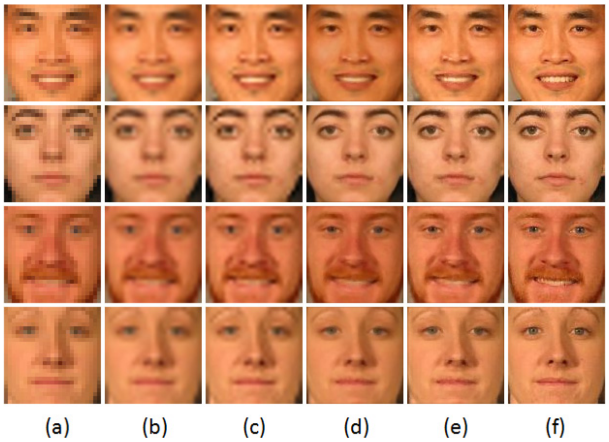

Fig. 8. Results of our algorithm compared to other methods and the corresponding average RMSEs. From left to right columns: (a) low resolution input; (b) Bicubic interpolation (RMSE: 8.024); (c) back projection (RMSE: 7.474); (d) global NMF modeling followed by bilateral filtering (RMSE: 10.738); (e) global NMF modeling and Sparse Representation (RMSE: **6.891**); (f) Original.

B. Effects of Dictionary Size

The above experimental results show that the sparsity prior for image patches is very effective in regularizing the otherwise ill-posed super-resolution problem. In those results, we fix the dictionary size to be 1024. Intuitively, larger dictionaries should possess more expressive power (in the extreme, we can use the sampled patches as the dictionary directly as in [1]) and thus may yield more accurate approximation, while increasing the computation cost. In this section, we evaluate the effect of dictionary size on generic image super-resolution. From the sampled 100,000 image patch pairs, we train four dictionaries of size 256, 512, 1024 and 2048, and apply them to the same input image. We also use the 100,000 image patches directly as the dictionary for comparison. The results are evaluated both visually and quantitatively in RMSE.

Fig. 9 shows the reconstructed results for the Lena image using dictionaries of different sizes. While there are not many visual differences for the results using different dictionary sizes from 256 to 2048 and the whole sampled patch set, we indeed observe the reconstruction artifacts will gradually diminish with larger dictionaries. The visual observation is also supported by the RMSEs of the recovered images. In Table IV-B, we list the RMSEs of the reconstructed images for dictionaries of different sizes. As shown in the table, using larger dictionaries will yield smaller RMSEs, and all of them have smaller RMSEs than those by Bicubic interpolation. However, the computation is approximately linear to the size of the dictionary; larger dictionaries will result in heavier computation. Fig. 10 shows the computation time in seconds with "Girl" as the test image. The algorithm is written in Matlab without optimization for speed, and ran on a laptop of Core duo @ 1.83G with 2G memory. To compare with [1], the computation time is almost an hour, much slower than our current solution with trained compact dictionaries. In practice, one chooses the appropriate dictionary size as a trade-off between reconstruction quality and computation. We find that dictionary size 1024 can yield decent outputs, while allowing fast computation.

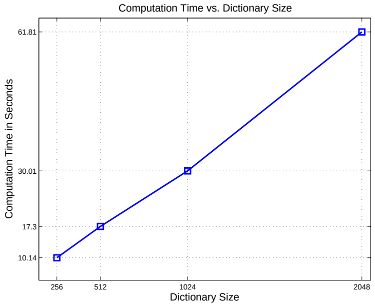

Fig. 10. The computation time on "Girl" image with dictionaries of different sizes (in seconds).

C. Robustness to Noise

Most single input super-resolution algorithms assume that the input images are clean and free of noise, an assumption which is likely to be violated in real applications. To deal with noisy data, previous algorithms usually divide the recovery process into two disjoint steps: first denoising and then super-resolution. However, the results of such a strategy depend on the specific denoising technique, and any artifacts during denoising on the low-resolution image will be kept or even magnified in the latter super-resolution process. Here we demonstrate that by formulating the problem into our sparse representation model, our method is much more robust to noise with input and thus can handle super-resolution and denoising simultaneously. Note that in (6) the parameter λ depends on the noise level of the input data; the noisier the data, the larger the value of λ should be. Fig. 11 shows how λ influences the reconstructed results given the same noiseless input image. The larger λ, the smoother the result image texture gets. This is obvious by formulating Eq. 8 into Maximum a Posterior (MAP) problem:

$$\boldsymbol{\alpha}^* = \arg\max \ P(\boldsymbol{\alpha}) \cdot P(\tilde{y}|\boldsymbol{\alpha}, \tilde{\boldsymbol{D}}). \tag{27}$$

where

$$\begin{aligned} P(\boldsymbol{\alpha}) &= \frac{1}{2b}\exp(-\frac{\|\boldsymbol{\alpha}\|_1}{b}) \\ P(\tilde{y}|\boldsymbol{\alpha}, \tilde{\boldsymbol{D}}) &= \frac{1}{2\sigma^2}\exp(-\frac{1}{2\sigma^2}\|\tilde{\boldsymbol{D}}\boldsymbol{\alpha} - \tilde{\boldsymbol{y}}\|_2^2), \end{aligned} \tag{28}$$

where b is the variance of the Laplacian prior on $\boldsymbol{\alpha}$, and σ^2 is the variance of the noise assumed on the data $\tilde{y}$. Taking the negative log likelihood in Eq. 27, we get the exact optimization problem in Eq. 8, with $\lambda = \sigma^2/b$. Suppose the Laplacian variance b is fixed, the more noisy of the data (σ^2 is larger), the larger of the value λ should be. On the other hand, given the input image, the larger value of λ we set, the more noisy the model will assume of the data, and thus tends to generate smoother results.

To test the robustness of our algorithm to noise, we add different levels of Gaussian noise to the low resolution input image. The standard deviation of the Gaussian noise ranges from 4 to 10. The regularization parameter λ is empirically set to be one tenth of the standard deviation. In Fig. 12, we

Images	Bicubic	D256	D512	D1024	D2048	Raw Patches
Girl	5.912	5.606	5.603	5.491	**5.473**	**5.483**
Flower	3.530	3.266	3.271	3.212	**3.164**	**3.139**
Lena	7.360	6.587	6.572	6.359	**6.232**	**6.029**
Statue	9.873	8.826	8.777	8.342	**8.237**	**8.255**

TABLE I
THE RMSES OF THE RECONSTRUCTED IMAGES USING DICTIONARIES OF DIFFERENT SIZES, AND USING THE RAW IMAGE PATCHES DIRECTLY FROM WHICH THE DICTIONARIES ARE TRAINED.

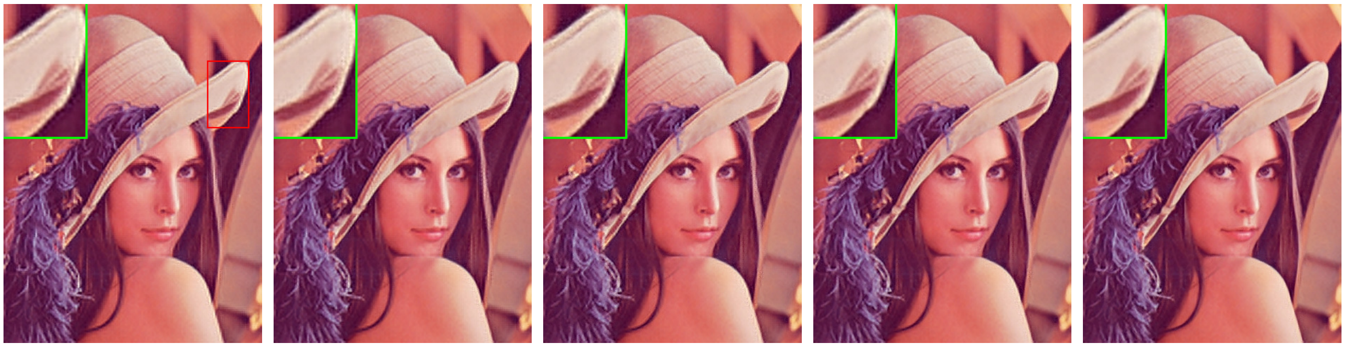

Fig. 9. The effects of dictionary size on the super-resolution reconstruction of Lena. From left to right: dictionary size 256, 512, 1024, 2048 and the whole sampled patch set respectively.

Noise Levels / Gaussian σ	0	4	6	8
Bicubic	9.873	10.423	11.037	11.772
Neighbor Embedding	9.534	10.734	11.856	13.064
Our method	**8.359**	**9.240**	**10.454**	**11.448**

TABLE II
THE RMSES OF THE RECONSTRUCTED IMAGES FROM DIFFERENT LEVELS OF NOISY INPUTS.

Methods	Flower	Girl	Parthenon	Lena	Statue
Bicubic	3.530	5.912	12.724	7.360	9.873
Local Model	3.365	5.669	12.247	6.775	8.902
Plus Global	3.212	5.491	11.875	6.359	8.237

TABLE III
THE GLOBAL CONSTRAINT IN THE SECOND STEP FURTHER REFINES THE RESULTS FROM LOCAL SPARSE MODEL IN THE FIRST STEP AND REDUCES RMSES.

show the results of our algorithm applying to the Liberty statue image with different levels of Gaussian noise. For comparison, we also show the results of using Bicubic and NE [11]. As expected, the results of Bicubic is both noisy and blurred. For NE, the number of neighbors is chosen as decreasing as the noise becomes heavier to get better results. As shown, the NE method is good at preserving edges, but fails to distinguish the signal from noise, and therefore generates unwanted noisy results. Our algorithm is capable of performing denoising and super-resolution simultaneously more elegantly. Table II shows the RMSEs of the reconstructed images from different levels of noisy data. In terms of RMSE, our method outperforms both Bicubic interpolation and NE in all cases.

D. Effects of Global Constraints

The global reconstruction constraint enforced by Eq. 9 is employed to refine the local image patch sparse model, ensuring the recovered high-resolution image to be consistent with its low-resolution observation. In our experiments, we observe that the sparsity prior is very effective and contribute the most, while the global constraint in the second step reduces RMSE by removing some minor artifacts which are hardly seen from the first step. Tab. III shows the RMSEs of the results from local sparse model only and local model combined with the global model. The RMSEs of Bicubic interpolation are again given as references. As shown, the local sparse model can achieve better RMSEs than Bicubic interpolation, and the global constraint further reduces the RMSEs of the recovered images. These experiments are carried out with dictionary size 1024.

V. CONCLUSION

This paper presented a novel approach toward single image super-resolution based on sparse representations in terms of coupled dictionaries jointly trained from high- and low-resolution image patch pairs. The compatibilities among adjacent patches are enforced both locally and globally. Experimental results demonstrate the effectiveness of the sparsity as a prior for patch-based super-resolution both for generic and face images. However, one of the most important questions for future investigation is to determine the optimal dictionary size for natural image patches in terms of SR tasks. Tighter connections to the theory of compressed sensing may yield conditions on the appropriate patch size, features to utilize and also approaches for training the coupled dictionaries.

REFERENCES

[1] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image super-resolution as sparse representation of raw image patches,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

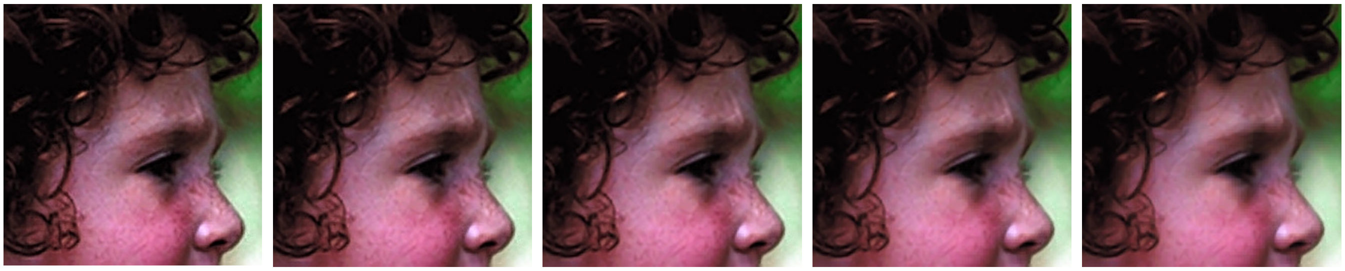

Fig. 11. The effects of λ on the recovered image given the input. From left to right, $\lambda = 0.01, 0.05, 0.1, 0.2, 0.3$. The larger λ is, the smoother the result image gets. Note that the results are generated from the local model only.

[2] R. C. Hardie, K. J. Barnard, and E. A. Armstrong, “Joint map registration and high-resolution image estimation using a sequence of undersampled images,” *IEEE Transactions on Image Processing*, vol. 6, pp. 1621–1633, 1997.

[3] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, “Fast and robust multiframe super-resolution,” *IEEE Transactions on Image Processing*, vol. 13, pp. 1327–1344, 2004.

[4] M. E. Tipping and C. M. Bishop, “Bayesian image super-resolution,” in *Advances in Neural Information and Processing Systems 16 (NIPS)*, 2003.

[5] S. Baker and T. Kanade, “Limits on super-resolution and how to break them,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1167–1183, Sep. 2002.

[6] H. S. Hou and H. C. Andrews, “Cubic spline for image interpolation and digital filtering,” *IEEE Transactions on Signal Processing*, vol. 26, pp. 508–517, 1978.

[7] S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong, “Soft edge smoothness prior for alpha channel super resolution,” in *IEEE Conference on Computer Vision and Pattern Classification (CVPR)*, 2007, pp. 1–8.

[8] J. Sun, Z. Xu, and H. Shum, “Image super-resolution using gradient profile prior,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[9] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, “Learning low-level vision,” *International Journal of Computer Vision*, vol. 40, no. 1, pp. 25–47, 2000.

[10] J. Sun, N. N. Zheng, H. Tao, and H. Shum, “Image hallucination with primal sketch priors,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2003, pp. 729–736.

[11] H. Chang, D.-Y. Yeung, and Y. Xiong, “Super-resolution through neighbor embedding,” in *IEEE Conference on Computer Vision and Pattern Classification (CVPR)*, vol. 1, 2004, pp. 275–282.

[12] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[13] S. Baker and T. Kanade, “Hallucinating faces,” in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2000, pp. 83–88.

[14] C. Liu, H. Y. Shum, and W. T. Freeman, “Face halluciantion: theory and practice,” *International Journal of Computer Vision*, vol. 75, no. 1, pp. 115–134, 2007.

[15] W. Liu, D. Lin, and X. Tang, “Hallucinating faces: tensorpatch super-resolution and coupled residue compensation,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2006, pp. 478–484.

[16] *Compressive sensing*, vol. 3, 2006.

[17] D. L. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[18] H. Rauhut, K. Schnass, and P. Vandergheynst, “Compressed sensing and redundant dictionaries,” *IEEE Transactions on Information Theory*, vol. 54, no. 5, May, 2008.

[19] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, pp. 3736–3745, 2006.

[20] J. Mairal, G. Sapiro, and M. Elad, “Learning multiscale sparse representations for image and video restoration,” *Multiscale Modeling and Simulation*, vol. 7, pp. 214–241, 2008.

[21] M. Aharon, M. Elad, and A. Bruckstein, “K-svd: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transaction on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[22] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” in *Advances in Neural Information Processing Systems (NIPS)*, 2007.

[23] D. L. Donoho, “For most large underdetermined systems of linear equations, the minimal ℓ^1-norm solution is also the sparsest solution,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[24] ——, “For most large underdetermined systems of linear equations, the minimal ℓ^1-norm near-solution approximates the sparsest near-solution,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 7, pp. 907–934, 2006.

[25] D. Wipf and S. Nagarajan, “A new view of automatic relevance determination,” in *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[26] ——, “Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions,” 2008, uC San Francisco technical report.

[27] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of Royal Statistical Society, Series B*, vol. 58, no. 1, 1996.

[28] W. T. Freeman, T. R. Jones, and E. C. Pasztor, “Example-based super-resolution,” *IEEE Computer Graphics and Applications*, vol. 22, pp. 56–65, 2002.

[29] D. D. Lee and H. S. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature 401*, pp. 788–791, 1999.

[30] B. Olshausen and D. Field, “Sparse coding with an overcomplete basis set: A strategy employed by v1?” *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[31] J. F. Murray and K. Kreutz-Delgado, “Learning sparse overcomplete codes for images,” *The Journal of VLSI Signal Processing*, vol. 45, pp. 97–110, 2007.

[32] Z. Wang and A. C. Bovik, “Mean squared error: love it or leave it? -a new look at signal fidelity measures,” *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, Jan. 2009.

[33] M. Irani and S. Peleg, “Motion analysis for image enhancement: resolution, occlusion and transparency,” *Journal of Visual Communication and Image Representation*, vol. 4, no. 4, pp. 324–335, 1993.

[34] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of face recognition grand challenge,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 947–954.

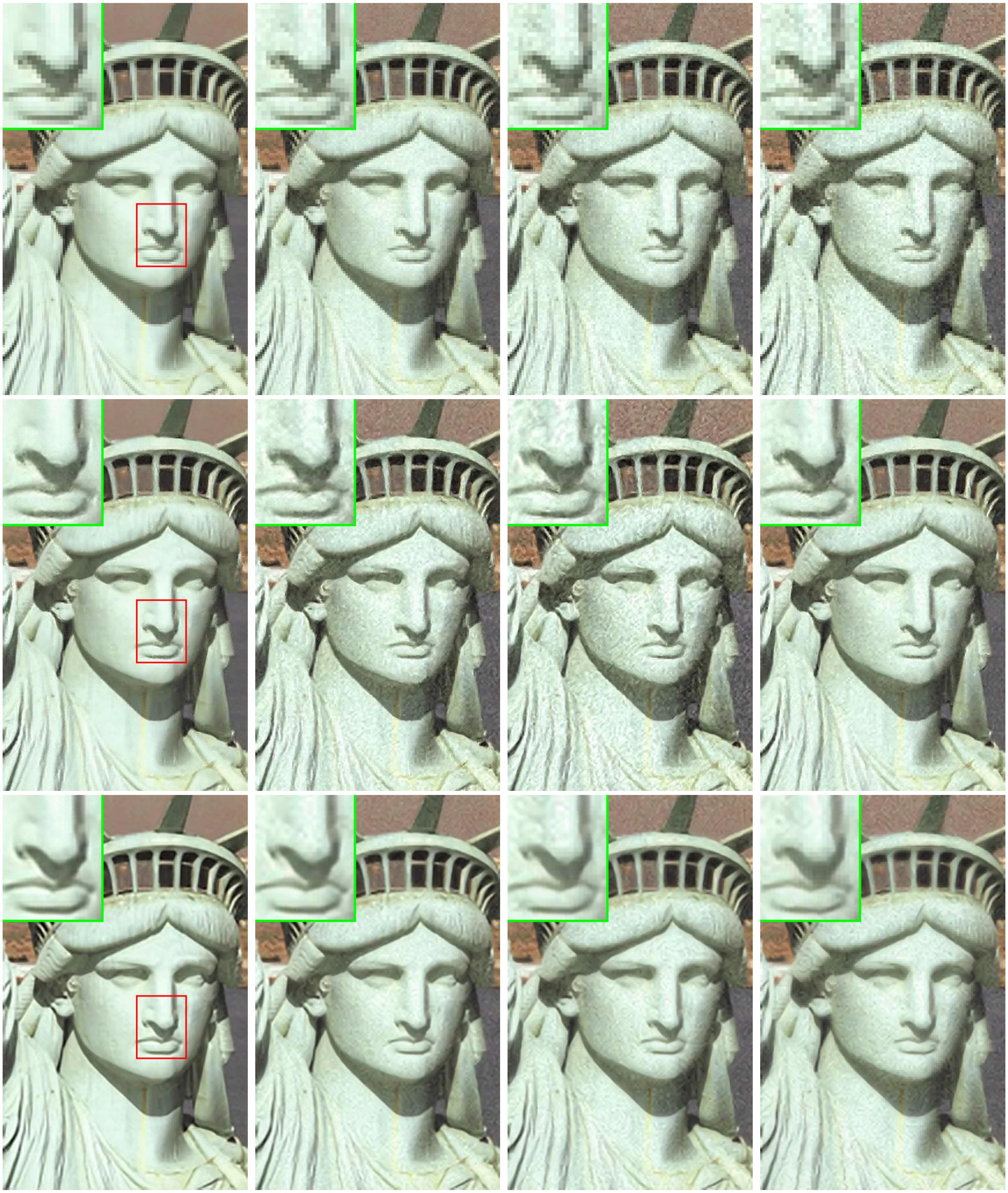

Fig. 12. Performance evaluation of our proposed algorithm on noisy data. Noise level (standard deviation of Gaussian noise) from left to right: 0, 4, 6 and 8. Top row: input images. Middle row: recovered images using NE [11] (k = 13, 12, 9, 7). Bottom row: recovered images using our method ($\lambda = 0.1, 0.4, 0.6, 0.8$).