Multi-Scale Weighted Nuclear Norm Image Restoration

Noam Yair and Tomer Michaeli
Technion - Israel Institute of Technology
{noamyair@campus, tomer.m@ee}.technion.ac.il

Abstract

*A prominent property of natural images is that groups of similar patches within them tend to lie on low-dimensional subspaces. This property has been previously used for image denoising, with particularly notable success via weighted nuclear norm minimization (WNNM). In this paper, we extend the WNNM method into a general image restoration algorithm, capable of handling arbitrary degradations (*e.g. *blur, missing pixels, etc.). Our approach is based on a novel regularization term which simultaneously penalizes for high weighted nuclear norm values of* all *the patch groups in the image. Our regularizer is isolated from the data-term, thus enabling convenient treatment of arbitrary degradations. Furthermore, it exploits the fractal property of natural images, by accounting for patch similarities also across different scales of the image. We propose a variable splitting method for solving the resulting optimization problem. This leads to an algorithm that is quite different from "plug-and-play" techniques, which solve image-restoration problems using a sequence of denoising steps. As we verify through extensive experiments, our algorithm achieves state of the art results in deblurring and inpainting, outperforming even the recent deep net based methods.*

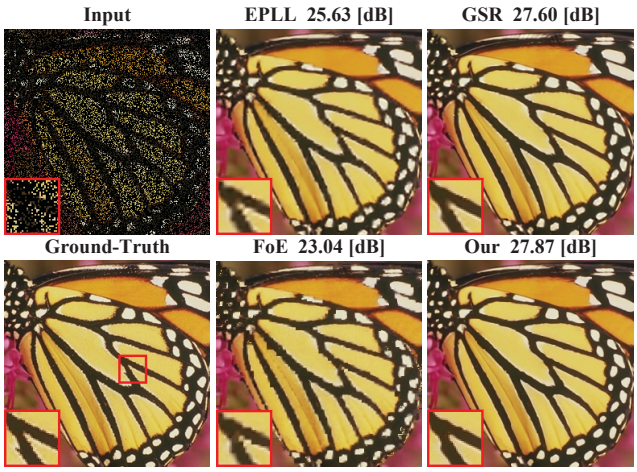

Figure 1. **Inpainting with** 75% **missing pixels.** Our algorithm handles arbitrary degradations within a single simple framework. It relies on a novel regularization term which encourages similar patches within and across scales of the image to lie on low-dimensional subspaces. This leads to state-of-the-art results in tasks like inpainting and deblurring. Note how our algorithm produces a naturally looking reconstruction with sharp edges and no distracting artifacts. This is also supported by the high PSNR values it attains w.r.t. competing approaches.

1. Introduction

Removing undesired degradations from images (*e.g.* blur, noise, missing parts) is important in a wide range of applications, and also serves as an ideal test bed for natural image statistics models. In recent years, this field is seeing a paradigm shift, as discriminative methods based on convolutional neural nets (CNNs) [7, 48, 49, 15, 44, 27, 14, 13, 30, 37, 29] push aside generative and regularization-based algorithms [12, 18, 19, 52, 39, 16, 31, 40, 36, 35]. However, while direct end-to-end training of a CNN is particularly suitable for image denoising, it is not equally practical for *all* image restoration tasks. For example, in deblurring or inpaining one would need to train a different net for every possible blur kernel or missing pixels mask. Recent works suggested to overcome this limitation by using iterative algorithms, which involve a denoising operation in each step [42, 38], thus requiring training only a denoising net [49]. Yet, this "plug-and-play" approach does not directly target the minimization of the mean-square error (MSE) through end-to-end training, and thus does not exploit the full power of discriminative methods.

In this work, we demonstrate that a simple regularization based algorithm can achieve state-of-the-art results in image restoration, improving over all existing methods by a significant margin (including those based on CNNs). Figure 1 shows an example result of our algorithm in the task of inpainting. We join together under a single framework several features, which have been previously shown to be very effective. First, we rely on the tendency of small patches to recur abundantly within natural images [6, 17, 12, 50, 45]. More specifically, we use the fact groups of similar patches typically span low-dimensional subspa-

ces [19, 31, 41, 8, 10, 9, 20, 28]. We do this by adopting the weighted nuclear norm minimization (WNNM) framework of [24], which has been shown to lead to excellent results in image denoising. Second, we use the fact that small patches tend to recur not only within the same scale but also across different scales in natural images [22, 50]. This phenomenon has been successfully used for super-resolution [22, 21, 25] and for blur kernel estimation [33, 32]. Lastly, rather than formulating an independent reconstruction problem for each patch group, as in [24], we propose a regularization term that takes into account *all* the patch groups simultaneously, by using the expected patch log-likelihood (EPLL) approach of [52]. This allows us to isolate the regularizer from the data term, thus enabling convenient treatment of arbitrary degradations (*e.g*. noise, blur, downsampling) by a single algorithm.

To solve our optimization problem, we propose a unique variable splitting approach. The resulting algorithm turns out to be quite different from plug-and-play techniques, as it does not involve explicit steps of denoising. The differences are also confirmed in experiments, where we show that our method achieves state-of-the-art results in deblurring and inpainting. An important conclusion from our work, is that regularization-based approaches are still relevant for image restoration, even in the era of deep-nets.

2. Related work

Internal patch-based methods Many image restoration algorithms exploit the tendency of small patches to repeat within natural images. Since its first successful use in image denoising [6], many methods relied on refined versions of this property, taking into account also small variations between recurring patches [12, 19, 18, 47, 31, 17, 45, 46]. In the context of image denoising, a particularly effective approach is the WNNM algorithm [24, 23, 43], which encourages groups of similar patches to form low-rank matrices. However, similarly to many other patch-based methods, the WNNM algorithm processes each group of patches independently while averaging the denoised overlapping patches. Therefore, it cannot be trivially extended to treat spatial degradations, like blur, where each patch is also affected by its surrounding environment.

From patches to whole images To extend the WNNM technique to arbitrary degradations, we formulate an optimization problem with a data-term and a prior-term, which both apply to *the whole image* rather than to independent patches. For our prior term, we follow the successful EPLL method [52] for combining single-patch models into whole image priors. The formalism underlying this method has been given various interpretations [26]. The original EPLL approach was used with parametric (Gaussian mixture) models. Later, it has also been applied with nonparametric patch recurrence models [33]. Here, we apply the approach with patch groups rather than with single patches. This leads to a complex model which captures long-range dependencies, as each patch can participate in several different groups.

Cross-scale patch recurrence Small patterns recur not only within the same scale, but also across different scales of the image [22, 50]. This property has been shown very effective for image compression [3], super-resolution [22, 21, 25], deblurring [2], blind super-resolution [32], blind deblurring [33], and denoising [51]. Here, we exploit this phenomenon in our regularizer, allowing us to boost the performance in *any* image restoration task within a single framework.

Image restoration by denoising Recently, it has been shown that image restoration problems can be solved using a sequence of denoising operations [42, 38, 5, 49]. The key idea is that objectives comprising a data term and a prior (regularization) term, can be solved iteratively using variable splitting techniques like half quadratic splitting (HQS) and alternating direction method of multipliers (ADMM). Each iteration then involves a sub-problem that can be interpreted as a (regularization-based) denoising step. This observation has motivated researchers to plug-in state-of-the-art denoisers, *e.g*. based on CNNs [49], in order to obtain high quality image restoration results. It should be noted, however, that this formulation does not guarantee that the better the denoiser's performance, the better the performance of the entire plug-and-play prior (PPP) scheme. Indeed, as we demonstrate experimentally, our algorithm outperforms PPP with a CNN denoiser [49] as well as the regularization-by-denoising (RED) approach of [38] with the TNRD [11] denoiser. This is despite the fact that those denoisers outperform the WNNM denoiser, upon which we rely. Furthermore, we also outperform RED with a WNNM denoiser, indicating that plug-and-play formulations do not necessarily make best use of the prior underlying their denoiser.

3. Problem formulation

Our goal is to recover an image x from its degraded version

$$y = Hx + n, \tag{1}$$

where n is noise and H is some known matrix. This formulation can account for many types of degradations, including blur (uniform or nonuniform), missing pixels, downsampling, etc. In most cases of interest, this inverse problem is severely ill-posed. Thus, any attempt to provide an accurate estimate of x must rely on some prior knowledge regarding the typical behavior of natural images.

In this work, we rely on the property that groups of similar patches tend to span low-dimensional subspaces [24]. Specifically, let x_i be some $\sqrt{m}\times\sqrt{m}$ patch in the image x, and let X_i be the $m\times k$ matrix whose columns contain the k nearest neighbor patches of x_i (stacked as column vectors) within some search window around x_i. Then, as shown in [24], any such constructed X_i is usually very close to be a low-rank matrix. Namely, only a few of its singular values $\{\sigma_\ell(X_i)\}$ are large, while the rest are close to zero.

This property has been exploited in [24] for image denoising, *i.e.* where H is the identity matrix. The WNNM method operates on each patch-group independently and then averages the results obtained for overlapping patches. The denoised version of the patch-group Y_i, is obtained as the solution to

$$\min_{X_i} \frac{1}{2\sigma_n^2}\|Y_i - X_i\|_{\mathrm{F}}^2 + \|X_i\|_{w,*}, \tag{2}$$

where σ_n^2 is the variance of the noise (assumed white and Gaussian), and $\|X_i\|_{w,*}$ is the weighted nuclear norm of X_i, defined as $\sum_\ell w_\ell \sigma_\ell(X_i)$ for some set of non-negative weights $\{w_\ell\}$. This term promotes solutions with few non-zero singular values. As shown in [24], assuming the singular values are ordered from large to small and the weights are non-descending, the solution of (2) is given by

$$\hat{X}_i = U S_w(\Sigma) V^T, \tag{3}$$

where $Y_i = U\Sigma V^T$ is the SVD of Y_i, and $S_w(\Sigma)$ is a generalized soft-thresholding operator that shrinks the values on the diagonal of Σ as

$$[S_w(\Sigma)]_{\ell\ell} = \max(\Sigma_{\ell\ell} - 2\sigma_n^2 w_\ell, 0). \tag{4}$$

It was specifically proposed in [24] to penalize the small singular values more than the large ones. This can be done in an iterative fashion, where in each iteration the weights are taken to be inversely proportional to the singular values of the solution from the previous iteration. In [23], it was shown that this iterative procedure possesses a closed form solution, which we adopt in our algorithm as well.

Extending the WNNM method to handle the general image restoration problem (1) involves several challenges. First, when the degradation H is spatial (*e.g.* blur), each patch in y is also affected by pixels outside the corresponding patch in x. Therefore, it is sub-optimal to work on each patch-group independently. Second, the low-rank phenomenon is a property of x. However in WNNM, the nearest neighbors are determined based on the noisy image y (as there is no access to x). While this may be a good approximation in the case of denoising, it may be highly inaccurate when y is a blurry version of x, especially when the blur is not isotropic.

To overcome those limitations, we propose to construct a single cost function for the entire image x. Specifically, we would like to find an image x such that: (i) all its patch groups satisfy the low-rank assumption, and (ii) it conforms to the measured degraded image y. To this end, we suggest the objective

$$\min_x \frac{1}{2\sigma_n^2}\|Hx - y\|^2 + \lambda \sum_{i\in\Omega} \|X_i^{\mathrm{ms}}\|_{w,*}, \tag{5}$$

where Ω is the set of all patch indices in the image, and X_i^{ms} is a matrix whose columns contain the k nearest neighbor patches of the patch x_i from both the input image *and its coarser scaled-down versions* ('ms' stands for multi-scale). The parameter λ weighs the contribution of the prior term w.r.t. the data term (and is not a function of σ).

Note that since both the data term and the prior term are functions of the whole image x, we can treat arbitrary spatial degradations. Furthermore, our prior term explicitly takes into account all overlapping patches in the image through an EPLL-like formulation [52] (note that each patch can be a member of several groups). This is in contrast to the WNNM denoiser which simply averages inconsistent estimates of overlapping patches. Our prior also exploits recurrence of patches across scales, which as we show, provides a significant boost of performance.

4. Algorithm

To simplify the exposition, we first discuss the single-scale case, *i.e.* when scaled-down versions of x are not used. Let N_i denote the set of indices of the k nearest neighbor patches of x_i (the indices of the patches comprising X_i^{ms}). Note that both the patch groups $\{X_i^{\mathrm{ms}}\}$ and the index sets $\{N_i\}$ are functions of the unknown image x. We therefore alternate between updating the index sets $\{N_i\}$ based on the current x (using a nearest-neighbor search for each patch), and updating the image x with the current nearest-neighbor groups.

To update the image x, we use HQS, but with two types of auxiliary variables instead of one. Specifically, we associate an auxiliary image z with the image x, and also an auxiliary matrix Z_i with each of the patch groups X_i^{ms}. We then aim at solving the optimization problem

$$\begin{aligned}\min_{x,z,\{Z_i\}} \frac{1}{2\sigma_n^2}\|Hx - y\|^2 + \lambda \sum_{i\in\Omega} \|Z_i\|_{\omega,*} \\ + \frac{\mu_1}{2}\|z - x\|^2 + \frac{\mu_2}{2}\sum_{i\in\Omega}\|Z_i - \mathcal{R}_i\{z\}\|_{\mathrm{F}}^2,\end{aligned} \tag{6}$$

where $\mathcal{R}_i$ is the operator which extracts the patch group associated with patch x_i. Note that as μ_1 and μ_2 become larger, z approaches x, and each Z_i approaches $\mathcal{R}_i\{z\}$, which approaches $\mathcal{R}_i\{x\}$. Therefore, the solution of (6) approaches that of (5). The idea in HQS is to start with small values for μ_1 and μ_2, and gradually increase them while

updating x, z and $\{Z_i\}$. Specifically, as summarized in Algorithm 1, we repeatedly apply the steps:

1. solve for $\{Z_i\}$, while keeping x and z fixed,
2. solve for z, while keeping x and $\{Z_i\}$ fixed,
3. solve for x, while keeping z and $\{Z_i\}$ fixed,
4. increase μ_1 and μ_2,

until μ_1 and μ_2 reach certain predetermined high values.

Updating $\{Z_i\}$ Retaining only the terms that depend on the patch group Z_i in (6), we obtain the optimization problem

$$\min_{Z_i} \frac{\mu_2}{2\lambda}\|Z_i - \mathcal{R}_i\{z\}\|_{\mathrm{F}}^2 + \|Z_i\|_{w,*}, \tag{7}$$

where we divided all terms by λ. This problem is the same as (2), so that its solution is as in (3) and (4), with Z_i instead of X_i, $\mathcal{R}_i\{z\}$ instead of Y_i, and λ/μ_2 instead of σ_n^2. This step is in fact a denoising of the patch groups in z. However, in contrast to PPP techniques, here each group is processed independently, without constructing a whole image.

Updating z Retaining only the terms that depend on z in (6), we obtain the objective

$$\min_{z} \frac{\mu_1}{2}\|z - x\|^2 + \frac{\mu_2}{2}\sum_{i\in\Omega}\|Z_i - \mathcal{R}_i\{z\}\|_{\mathrm{F}}^2. \tag{8}$$

This is a quadratic program in z (note that $\mathcal{R}_i$ is a linear operator). Let Z_i^j denote the jth column in the matrix Z_i (*i.e.* the jth patch in the ith group), let N_i^j denote the index of where this patch belongs in the image, and let $R_{N_i^j}$ denote the matrix which extracts this patch from the image. Then, as we show in the Supplementary Material, the solution to (8) is given by

$$z = (\mu_1 I + \mu_2 W)^{-1}(\mu_1 x + \mu_2 \tilde{z}), \tag{9}$$

where I is the identity matrix, W is the diagonal matrix

$$W = \sum_{i\in\Omega}\sum_{j=1}^{k} R_{N_i^j}^T R_{N_i^j}, \tag{10}$$

and $\tilde{z}$ is the image

$$\tilde{z} = \sum_{i\in\Omega}\sum_{j=1}^{k} R_{N_i^j}^T Z_i^j. \tag{11}$$

This expression has a simple interpretation. The matrix R_ℓ^T takes a patch and places it in the ℓth location in the image. Therefore, the image $\tilde{z}$ is constructed by taking each patch from each of the groups $\{Z_i\}$ and putting it in its place in the image, while accumulating the contributions from overlapping patches. Similarly, the matrix W corresponds to a map storing the accumulated number of overlaps in each pixel. Note that this step *does not* construct a denoised version of z by averaging the denoised patch groups $\{Z_i\}$ (that option would correspond to $W^{-1}\tilde{z}$). Rather, it directly merges the denoised patch groups with x.

Algorithm 1 Image Restoration

```
Set x = y
while stopping criterion not satisfied do
    Update nearest neighbor index groups {N_i}
    Set z = x, initialize μ1, μ2 to small values
    while stopping criterion not satisfied do
        Update {Zi} according to (7) using (3),(4)
        Update z using (9)-(11)
        Update x using (13)
        Increase μ1 and μ2
    end while
end while
return x                    ▷ The restored image is x
```

Updating x Retaining only the terms that depend on x in (6), we get the problem

$$\min_{x} \frac{1}{2\sigma_n^2}\|Hx - y\|^2 + \frac{\mu_1}{2}\|z - x\|^2. \tag{12}$$

This is a simple quadratic program whose solution is

$$x = \left(\frac{1}{\sigma_n^2}H^T H + \mu_1 I\right)^{-1}\left(\frac{1}{\sigma^2}H^T y + \mu_1 z\right). \tag{13}$$

When H is a diagonal matrix, as in inpaining, this step corresponds to a per-pixel weighted average between y and z. When H is a convolution operator, as in deblurring, this expression can be efficiently calculated in the Fourier domain.

4.1. Extension to multi-scale

The algorithm described above can be easily extended to the multi-scale case. In this setting, each patch in x searches for nearest neighbors both within the image and within its scaled-down version. Therefore, in each patch group, typically some of the patches are from the original scale and some from the coarser scales. This has an effect when updating the patch groups $\{Z_i\}$ according to Eq. (7). In particular, the presence of patches from the coarser scale (which are less blurry and less noisy), typically improves the denoising of the patches from the original scale.

In principle, the update of z should also be affected by the multi-scale formulation. Specifically, if ℓ is the index of a patch from a coarse scale of the image, then the associated matrix R_ℓ appearing in (10) and (11) performs both

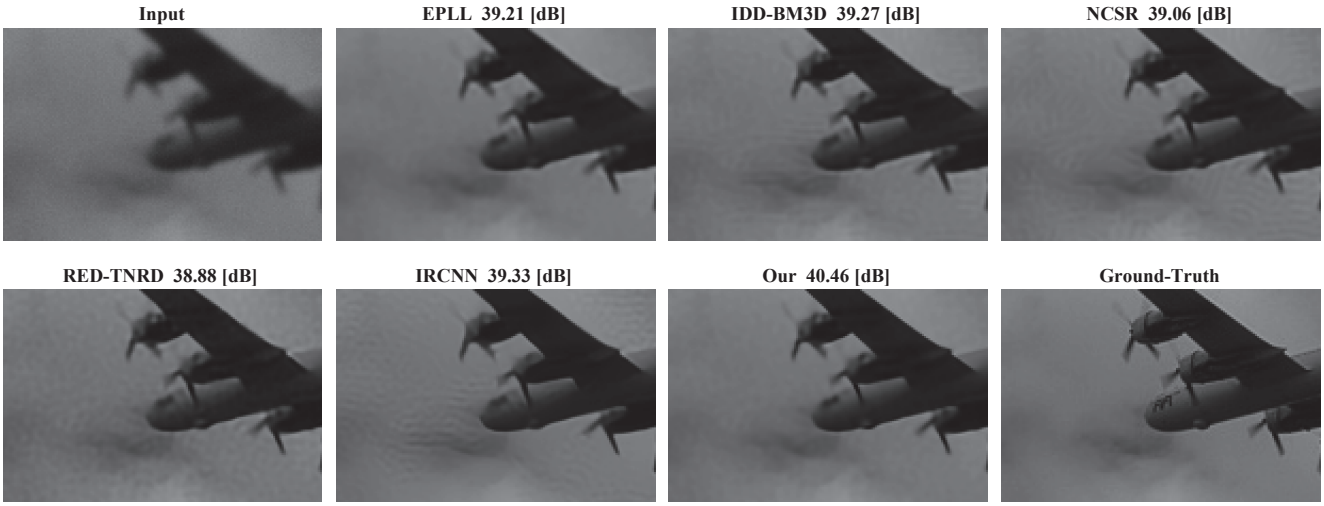

Figure 2. **Visual comparison of deblurring algorithms.** A crop from a degraded input image from the BSD dataset is shown on the top left. It suffers from Gaussian blur with standard deviation 1.6 and additive noise with $\sigma_n = 2$. As can be seen, while all state-of-the-art deblurring methods produce artifacts in the reconstruction, our algorithm produces sharp results without annoying distortions. Its precision is also confirmed by the very high PSNR it attaines w.r.t. the other methods.

down-sampling and patch extraction. Similarly, the matrix R_ℓ^T takes a patch, performs up-sampling, and places it in its location in the image (which now has a larger support). However, we found that ignoring the coarse-scale patches during the z-update step still leads to excellent performance, while reducing computations. Therefore, in our implementation we use the coarse scale patches only in (7).

5. Experimental results

We study the effectiveness of our method in non-blind deblurring and in inpainting, which are two cases where discriminative methods cannot be trained end-to-end. In both applications we use 6×6 patches at a stride of 2 pixels, and $k = 70$ nearest neighbors for each patch. By default, we use scales 1 and 0.75, with a search window of 30×30 around each patch in both scales (see analysis of the effect of multi-scale in Sec. 5.3). We perform 5 HQS iterations, where we initialize $\mu_1 = 1.5 \times 10^{-3}, \mu_2 = 10^{-3}$ and increment them by factors of 2 and 1.5, respectively.

To conform to the comparisons in previous works, we report results on gray-scale images. However, our algorithm can restore color images as well (see *e.g.* Fig. 1). This is done by converting the image to the YCbCr color-space and applying the restoration algorithm only on the luminance channel. In deblurring, we do not process the chrominance channels. In inpaining, we use simple interpolation to fill in the missing pixels in the chrominance channels. We then convert the result back to the RGB domain.

5.1. Deblurring

We compare to two state-of-the-art "restoration by denoising" approaches (see Sec. 2): **IRCNN** [49] which uses a CNN denoiser, and **RED** [38] with **TNRD** [11] as the denoising engine. We also compare to the more classic **IDD-BM3D** [12], **NCSR** [18] and **EPLL** [52] methods. To be consistent with previous works, we follow the non-blind deblurring experiments conducted in [18] and [38] (as well as in other papers). Namely, we study two blur kernels and two Gaussian noise levels. For the blur, we use a uniform 9×9 kernel and a Gaussian 25×25 kernel with standard deviation 1.6. For the noise, we use $\sigma_n = \sqrt{2}$ and $\sigma_n = 2$. We always run our algorithm for 300 iterations and use $\lambda = 0.01$.

Figure 2 shows an example result of our algorithm in the setting of Gaussian blur and noise level $\sigma_n = 2$. As can be seen, our approach produces reconstructions that are substantially better than the competing methods, both visually and in terms of PSNR.

For our quantitative analysis, we begin with two popular small-scale datasets: Set5 from [4], and the 10 test images from the NCSR paper [18] (see Fig. 3). Table 1 summarizes the PSNR results for each of the images in Set5. As can be seen, our algorithm significantly outperforms the competitors on the vast majority of the images with all four combinations of blur and noise settings. A similar behavior can be seen in Table 2, which shows the average PSNR results on the NCSR dataset (the results for the individual images can be found in the Supplementary). In both datasets, the PSNR of our algorithm is higher by 0.18dB on average than the second best method, which is IRCNN.

Next, we perform a comparison on the larger-scale

Figure 3. The Set5 dataset from [4] (left) and the NCSR dataset from [18] (right) used in our evaluations.

	Image	Input	EPLL	IDD-BM3D	NCSR	RED+TNRD	IRCNN	Our
Gaussian blur, $\sigma_n = \sqrt{2}$	Baby	30.10	35.06	35.01	34.47	34.73	34.83	**35.21**
	Bird	28.81	36.20	**36.75**	35.44	35.88	36.64	36.56
	Butterfly	21.48	28.46	29.28	28.77	29.63	29.96	**30.20**
	Head	28.93	32.88	32.94	32.64	32.76	32.68	**33.01**
	Woman	25.88	31.85	32.40	31.94	32.13	32.36	**32.71**
	Average	27.03	32.89	33.27	32.65	33.031	33.30	**33.54**
Gaussian blur, $\sigma_n = 2$	Baby	29.96	34.56	34.57	34.60	34.29	34.43	**34.74**
	Bird	28.71	35.39	**36.04**	35.79	35.34	36.01	35.70
	Butterfly	21.46	27.86	28.77	28.70	29.06	29.49	**29.73**
	Head	28.83	32.58	32.62	32.62	32.46	32.45	**32.70**
	Woman	25.83	31.29	31.92	31.82	31.62	31.89	**32.23**
	Average	26.95	32.33	32.78	32.70	32.56	32.85	**33.02**
Uniform blur, $\sigma_n = \sqrt{2}$	Baby	26.31	32.76	32.98	32.81	32.91	32.85	**33.14**
	Bird	24.64	32.49	33.56	33.32	33.70	33.90	**34.14**
	Butterfly	17.74	26.03	27.77	27.90	28.60	**28.93**	28.83
	Head	26.16	31.37	31.65	31.55	31.74	31.74	**31.81**
	Woman	22.14	29.05	30.49	30.68	30.49	31.08	**31.23**
	Average	23.39	30.34	31.29	31.25	31.49	31.70	**31.83**
Uniform blur, $\sigma_n = 2$	Baby	26.25	32.10	32.36	32.43	32.32	32.22	**32.54**
	Bird	24.60	31.60	32.64	32.86	32.75	32.95	**33.24**
	Butterfly	17.74	24.91	26.83	27.31	27.41	**28.05**	27.98
	Head	26.10	30.93	31.24	31.28	31.38	31.39	**31.41**
	Woman	22.11	28.06	29.51	30.07	29.57	30.24	**30.47**
	Average	23.36	29.53	30.52	30.79	30.69	30.97	**31.13**

Table 1. **Deblurring comparison on Set5.** Our method is compared to the state-of-the-art deblurring methods on Set5 [4] with four different degradations. The best results are shown in bold.

BSD100 dataset [1]. In this case, to accelerate our method's convergence, we initialize it with the fast IRCNN method and then run it for only 5 iterations (see Sec. 5.4 for an analysis of the effect of this approach). As can be seen in Table 3, again our method outperforms the second best method by 0.18dB on average. Furthermore, it attains the best PSNR among all methods on 97% of the images.

5.2. Inpainting

In the task of inpainting, we compare our method to the state-of-the-art algorithms **GSR** [47], **LINC** [34], **FoE** [39] and **EPLL** [52]. We use Set5 and Set NCSR with 25%, 50% and 75% missing pixels. The LINC method failed to operate with 75% missing pixels, and so was not evaluated for this setting. Here, we set λ = 0.015, and use 200, 300 and 400 iterations for the settings of 25%, 50% and 75% blank pixels, respectively. In general, the stronger the degradation the more iterations our algorithm requires to converge.

Figure 4 shows a visual comparison of the inpainging results produced by EPLL, FoE, GSR and our algorithm for 75% missing pixels. As can be seen, our reconstruction is sharp and does not suffer from artifacts. The GSR reconstruction is the only one which is visually similar to ours, albeit suffering from a few mild distortions. This method is also based on the low-rank property of patch groups, which provides further support to the effectiveness of this model for image restoration.

The results on Set5 and Set NCSR are summarized in Tables 4 and 5, respectively. As can be seen, our method outperforms all other methods. The only method that sometimes comes within 0.1[dB] to ours, is the GSR algorithm. This confirms again that the internal low-rank prior (exploited by both GSR and our algorithm) can provide a strong cue for image reconstruction. More comparisons and details are available in the Supplementary Material.

	Input	EPLL	IDD-BM3D	NCSR	RED+TNRD	IRCNN	Our
Gaussian blur, $\sigma_n = \sqrt{2}$	24.65	29.83	30.45	30.37	30.53	30.76	**31.06**
Gaussian blur, $\sigma_n = 2$	24.61	29.26	29.88	30.07	29.92	30.22	**30.38**
Uniform blur, $\sigma_n = \sqrt{2}$	21.47	28.34	29.63	29.73	29.76	30.10	**30.22**
Uniform blur, $\sigma_n = 2$	21.45	27.43	28.74	29.04	28.84	29.21	**29.34**

Table 2. **Deblurring comparison on Set NCSR.** Our method is compared to the state-of-the-art deblurring methods on Set NCSR [18]. The average PSNR [dB] is reported for four different degradations. The best results are shown in bold. Results on the individual images in this set can be found in the Supplementary Material.

	Input	EPLL	IDD-BM3D	NCSR	RED+TNRD	IRCNN	Our
Gaussian blur $\sigma_n = \sqrt{2}$	25.13	28.56	28.81	28.71	28.90	28.98	**29.18**
Gaussian blur $\sigma_n = 2$	25.07	28.12	28.39	28.48	28.50	28.58	**28.75**
Uniform blur $\sigma_n = \sqrt{2}$	22.66	27.42	27.93	28.05	28.10	28.25	**28.43**
Uniform blur $\sigma_n = 2$	22.62	26.74	27.25	27.48	27.44	27.57	**27.74**

Table 3. **Deblurring comparison on BSD100.** Our method is compared to the state-of-the-art deblurring methods on Set BSD100 [1]. The average PSNR [dB] is reported for four different corruptions. The best average results are shown in bold. In this experiment we initialized our algorithm with IRCNN and ran it for only 5 iterations (see Sec. 5.4).

5.3. The contribution of multiple scales

There are two key features, which contribute to the success of our algorithm. The first is the use of a coherent formulation that takes into account all patches in the image. The second is the use of patches from multiple scales of the image. To study the contribution of each of these two properties, we report in Table 6 the deblurring results of the RED algorithm with WNNM as its denoising engine, as well as the results attained by our algorithm with and without using multiple scales. All comparisons are carried out on Set5.

It can be seen that using multiple scales significantly improves the results over the single-scale setting. This supports the observation reported in many previous works, that the fractal property of natural images can provide a strong cue for image restoration. We can also learn from the table that even without using multiple scales, our method commonly outperforms RED-WNNM. Specifically, while for Gaussian blur it is inferior by roughly 0.1dB, for uniform blur it is superior by more than 0.3dB. This suggests that our formulation makes better use of the low-rank prior in natural images.

	Image	EPLL	FoE	GSR	LINC	Our
25% blank pixels	Baby	41.33	42.19	43.23	42.90	**43.52**
	Bird	44.11	43.58	48.02	47.08	**48.09**
	Butterfly	35.03	30.26	36.08	35.60	**37.35**
	Head	38.27	37.09	**38.59**	38.69	38.56
	Woman	40.09	37.27	41.62	40.74	**41.71**
	Average	39.77	38.08	41.51	41.00	**41.85**
50% blank pixels	Baby	37.11	37.72	38.45	38.30	**38.60**
	Bird	39.03	38.50	42.10	**42.24**	41.68
	Butterfly	29.63	26.85	31.60	30.87	**32.00**
	Head	34.88	34.23	35.12	**35.16**	34.94
	Woman	34.46	33.10	36.49	35.86	**36.60**
	Average	35.02	34.08	36.75	36.49	**36.77**
75% blank pixels	Baby	32.99	33.12	33.95	-	**34.15**
	Bird	33.14	33.02	**36.30**	-	36.01
	Butterfly	24.34	20.86	26.32	-	**26.67**
	Head	31.85	31.39	**32.13**	-	32.09
	Woman	28.61	27.81	30.90	-	**31.05**
	Average	30.19	29.24	31.92	-	**32.00**

Table 4. **Inpainting comparison on Set5.** Our method is compared to the state-of-the-art inpaining methods on Set5 [4] with three different missing pixel ratios. The best results are shown in bold.

5.4. Accelerated convergence and run-time

A key limitation of our algorithm is that it takes hundreds of iterations to converge. Furthermore, each iteration can take on the order of tens of seconds with brute-force nearest neighbor search (depending on the image size, search window, number of nearest neighbors, etc.). However, our method can be accelerated if initialized with a reasonable reconstruction. For example, in the context of deblurring, combining our approach with the fast IRCNN [49] method, leads to a significantly accelerated convergence. Figure 5 shows the progression of the PSNR along the iterations, when using the default initialization $x = y$, and when using IRCNN for initialization. As can be seen, IRCNN provides a solution which is already quite close to to our final optimum. Therefore, this allows running our algorithm for only a small number of iterations to get to the same final PSNR.

Running our algorithm on 256×256 images using an Intel Xeon CPU with un-optimized Matlab code (no parallelism or GPU used), each iterations takes about 1.1 minutes, whereas the IRCNN method takes about 0.5 minute (again,

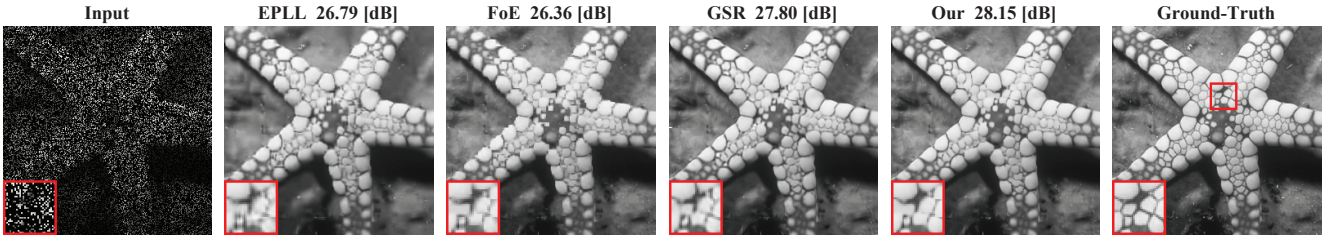

Figure 4. **Visual comparison of inpaining algorithms.** An input image with 75% missing pixels (left) was inpainted using several state-of-the-art methods. As can be seen, our algorithm and GSR, which both rely on the internal low-rank property of patch groups, produce the best visual results. However, our reconstruction suffers from less artifacts than GSR.

	EPLL	FoE	GSR	LINC	Our
25% blank	38.59	34.75	40.55	39.90	**41.29**
50% blank	33.19	31.28	35.44	34.94	**35.77**
75% blank	27.84	26.18	30.01	-	**30.06**

Table 5. **Inpainting comparison on Set NCSR.** Our method is compared to the state-of-the-art inpainting methods on Set NCSR [18]. The average PSNR [dB] is reported for three different missing pixel ratios. The best results are shown in bold. Results on the individual images in this set can be found in the Supplementary.

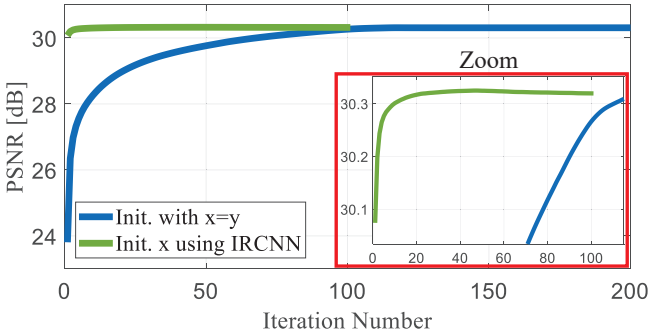

Figure 5. **Progression of PSNR along the iterations in deblurring.** When using the initialization $x = y$ (blue), our algorithm requires hundreds of iterations to converge. However, initializing x using the fast IRCNN method (green), allows running only a small number of iterations to get to the same final PSNR.

no GPU). Table 7 shows a run-time comparison for the deblurring experiment of Table 2 (first row). Note that when using IRCNN for initialization, our method attains a large PSNR improvement over the state of the art already from iteration 1 (96 seconds including the initialization).

6. Conclusion

We presented a method for image restoration, which generalizes the WNNM denoiser [24, 23] to treat arbitrary degradations. Our method is based on a regularization term, which is separate from the data term, thus allowing convenient treatment of different degradations with a single algorithm. This term simultaneously encourages all the groups

	Input	RED WNNM	Our w/o MS	Our w/ MS
Gaussian blur $\sigma_n = \sqrt{2}$	27.04	33.27	33.17	**33.54**
Gaussian blur $\sigma_n = 2$	26.96	32.78	32.65	**33.02**
Uniform blur $\sigma_n = \sqrt{2}$	23.40	31.25	31.57	**31.83**
Uniform blur $\sigma_n = 2$	23.36	30.38	30.79	**31.13**

Table 6. **The effect of multiple scales.** We compare deblurring performance on Set5 for RED with WNNM as its denoising engine, our method without multiple scales, and our method with multiple scales (1 and 0.75).

	Run-Time [minutes]	PSNR [dB]
EPLL	1	29.83
NCSR	2.5	30.37
IDD-BM3D	0.6	30.45
RED+TNRD	7	30.53
IRCNN	0.5	30.76
Our w/ IRCNN init. - 1 iteration	1.6	**30.99**
Our w/ IRCNN init. - 2 iterations	2.7	**31.03**
Our w/ naive init. - 300 iterations	330	**31.06**

Table 7. **Run-Time and PSNR Comparison.** Run-time and PSNR comparison for the deblurring experiment of Table 2 (first row). When initialized with IRCNN, our method significantly improves over the state-of-the-art already from iteration 1.

of similar patches in the image to lie on low dimensional subspaces. Moreover, it also takes into account repetitions of patches across scales of the image, which substantially improves the performance of our algorithm. We proposed a unique variable splitting method for solving our optimization problem, and showed that the resulting algorithm is quite different from existing plug-and-play approaches. We demonstrated through extensive experiments, that our algorithm leads to state-of-the-art deblurring and inpainting results, outperforming even methods based on CNNs.

Acknowledgements This research was supported in part by an Alon Fellowship and by the Ollendorf Foundation.

References

[1] The Berkeley segmentation dataset and benchmark. https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/.

[2] Y. Bahat, N. Efrat, and M. Irani. Non-uniform blind deblurring by reblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3286–3294, 2017.

[3] M. F. Barnsley and A. D. Sloan. Methods and apparatus for image compression by iterated function system, 1990. US Patent 4,941,193.

[4] M. Bevilacqua, A. Roumy, C. Guillemot, and M. Alberi. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. *British Machine Vision Conference*, 2012. https://github.com/titu1994/Image-Super-Resolution/tree/master/val_images/set5.

[5] A. Brifman, Y. Romano, and M. Elad. Turning a denoiser into a super-resolver using plug and play priors. *IEEE International Conference on Image Processing*, 2016.

[6] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2005.

[7] H. C. Burger, C. J. Schuler, , and S. Harmeling. Image denoising: Can plain neural networks compete with BM3D? *IEEE International Conference on Computer Vision and Pattern Recognition*, 2012.

[8] J.-F. Cai, E. J. Cands, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[9] E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3), 2011.

[10] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[11] Y. Chen and T. Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1256–1272, 2017.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.

[13] S. Diamond, V. Sitzmann, F. Heide, and G. Wetzstein. Unrolled optimization with deep priors. 2017.

[14] N. Divakar and R. V. Babu. Image denoising via CNNs: An adversarial approach. *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[15] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, 2016.

[16] W. Dong, G. Shi, and X. Li. Nonlocal image restoration with bilateral variance estimation: A low-rank approach. *IEEE Transactions on Image Processing*, 22(2):700–711, 2013.

[17] W. Dong, G. Shi, and X. Li. Nonlocal image restoration with bilateral variance estimation: A low-rank approach. *IEEE Transactions on Image Processing*, 22(2):700–711, 2013.

[18] W. Dong, L. Zhang, G. Shi, , and X. Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2013.

[19] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

[20] M. Fazel, H. Hindi, and S. Boyd. A rank minimization heuristic with application to minimum order system approximation. *American Control Conference*, 2001.

[21] G. Freedman and R. Fattal. Image and video upscaling from local self-examples. *ACM Trans. Graph.*, 28(3):1–10, 2010.

[22] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. *IEEE International Conference on Computer Vision*, 2009.

[23] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang. Weighted nuclear norm minimization and its applications to low level vision. *International Journal of Computer Vision*, 121(2):183–208, 2017.

[24] S. Gu, L. Zhang, W. Zuo, and X. Feng. Weighted nuclear norm minimization with application to image denoising. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2014.

[25] J.-B. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[26] G. Ji, M. C. Hughes, and E. B. Sudderth. From patches to images: A nonparametric generative model. 2017.

[27] Y. Kim, H. Jung, D. Min, and K. Sohn. Deeply aggregated alternating minimization for image restoration. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2017.

[28] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. *International Journal of Computer Vision*, 104(1):1–14, 2013.

[29] P. Liu and R. Fang. Learning pixel-distribution prior with wider convolution for image denoising. *Computing Research Repository (CoRR)*, 2017.

[30] P. Liu and R. Fang. Wide inference network for image denoising via learning pixel-distribution prior. 2017.

[31] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. *IEEE International Conference on Computer Vision*, 2009.

[32] T. Michaeli and M. Irani. Nonparmetric blind super-resolution. *IEEE International Conference on Computer Vision*, 2013.

[33] T. Michaeli and M. Irani. Blind deblurring using internal patch recurrence. *European Conference on Computer Vision*, pages 783–798, 2014.

[34] M. Niknejad, H. Rabbani, and M. Babaie-Zadeh. Image restoration using Gaussian mixture models with spatially constrained patch clustering. *IEEE Transactions on Image Processing*, 24(11):3624–3636, 2015.

[35] S. Osher and L. I. Rudin. Feature-oriented image enhancement using shock filters. *SIAM Journal on Numerical Analysis*, 27(4):919–940, 1990.

[36] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, 2003.

[37] P. Putzky and M. Welling. Recurrent inference machines for solving inverse problems. *Computing Research Repository (CoRR)*, 2016.

[38] Y. Romano, M. Elad, and P. Milanfar. The little engine that could: Regularization by denoising (RED). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

[39] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2005.

[40] L. I. Rudin, S. Osher, and E. Fatem. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992.

[41] N. Srebro and T. Jaakkola. Weighted low-rank approximations. *International Conference on International Conference on Machine Learning*, 2003.

[42] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg. Plug-and-play priors for model based reconstruction. *IEEE Global Conference on Signal and Information Processing*, 2013.

[43] J. Xu, L. Zhang, D. Zhang, and X. Feng. Multi-channel weighted nuclear norm minimization for real color image denoising. *IEEE International Conference on Computer Vision*, 2017.

[44] L. Xu, J. S. Ren, C. Liu, and J. Jia. Deep convolutional neural network for image deconvolution. *Advances in Neural Information Processing Systems 27*, pages 1790–1798, 2014.

[45] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873, 2010.

[46] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. *International Conference on Curves and Surfaces*, 2010.

[47] J. Zhang, D. Zhao, and W. Gao. Group-based sparse representation for image restoration. *IEEE Transactions on Image Processing*, 23(8):3336–3351, 2014.

[48] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[49] K. Zhang, W. Zuo, S. Gu, and L. Zhang. Learning deep CNN denoiser prior for image restoration. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2017.

[50] M. Zontak and M. Irani. Internal statistics of a single natural image. *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[51] M. Zontak, I. Mosseri, and M. Irani. Separating signal from noise using patch recurrence across scales. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1195–1202, 2013.

[52] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. *IEEE International Conference on Computer Vision*, 2011.