HAL
open science

Low-Rankness Transfer for Realistic Denoising

Hicham Badri, Hussein Yahia, Driss Aboutajdine

To cite this version:

Hicham Badri, Hussein Yahia, Driss Aboutajdine. Low-Rankness Transfer for Realistic Denoising. IEEE Transactions on Image Processing, 2016. hal-01361246

HAL Id: hal-01361246

https://inria.hal.science/hal-01361246v1

Submitted on 6 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Low-Rankness Transfer for Realistic Denoising

Hicham Badri, Hussein Yahia and Driss Aboutajdine

***Abstract*—Current state-of-the-art denoising methods such as non-local low-rank approaches have shown to give impressive results. They are however mainly tuned to work with uniform Gaussian noise corruption and known variance, which is far from the real noise scenario. In fact, noise level estimation is already a challenging problem and denoising methods are quite sensitive to this parameter. Moreover, these methods are based on shrinkage models that are too simple to reflect reality, which results in over-smoothing of important structures such as small-scale text and textures. We propose in this paper a new approach for more realistic image restoration based on the concept of low-rankness transfer (LRT). Given a training clean/noisy image pair, our method learns a mapping between the non-local noisy singular values and the optimal values for denoising to be transfered to a new noisy input. One single image is enough for training the model and can be adapted to the noisy input by taking a correlated image. Experiments conducted on synthetic and real camera noise show that the proposed method leads to an important improvement both visually and in terms of PSNR/SSIM.**

***Index Terms*—Denoising, low-rank, learning.**

I. Introduction

Image denoising is one of the most challenging problems in imaging science. It is also one of the most important long-standing problems because of its big commercial impact. Even-though camera devices become every year more powerful, noise is still present in pictures even using high-end devices. The ISO settings in each camera device permit to control the sensitivity of the sensor. Increasing the ISO leads to a better representation of the scene but produces more noise. High-end devices tend to capture less noise at higher ISO settings, which permits to capture better pictures but this comes at a much higher cost. The most simple way to model image corruption is by supposing that the noise is Gaussian uniform with known variance (AWGN model). This model is adopted to tune most of the current denoising methods. It is well known however that real camera noise is far from being Gaussian and uniform due to the complex camera imaging pipeline. For instance, the popular Charge-Couple Device (CCD) image sensors pipeline includes various perturbations such as atmospheric attenuation, lens/geometric distortion, CCD imaging/Bayer pattern, interpolation, white balancing and Gamma correction [26], [27]. In addition, at least five noise sources are added in the process including *fixed pattern noise*, *dark current noise*, *short noise*, *amlifier noise* and *quantization noise*. Due the complexity and non-linearity of the pipeline, one popular approach consists in simplifying the model to two noise corruptions of the irradiance where one is uniform and the other is signal-dependent [26]. The noisy image is then converted back with the camera response function (CRF) of the corrupted irradiance. From this model, a noise level function (NLF) can be calculated that maps the brightness to standard deviation and clearly reflects the non-uniformity of noise corruption. While this model is much richer than the standard AWGN model, it is still a simplification of a complex pipeline. Moreover, even if the NLF is well estimated, which is already a hard problem especially when fine textures are present in the picture, it is not clear how non-local methods can fully take advantage of it. As non-local methods manipulate non-local matrices, even if the noise levels of all the pixels are exactly known, the transformation from the non-local patch matrices to the clean ones so far takes into account only one noise level that controls the way the non-local patch matrices are processed in an appropriate domain.

On the other hand, working on the singular values of the non-local patches has various interesting properties in the context of non-local denoising. Using this approach, denoising each non-local patch matrix reduces to manipulating its singular values. Suppose that the patch size is 8×8, and we consider 70 similar patches, which results in a non-local patch matrix of size 64×70 and corresponds to 4480 unknown clean pixels to recover. The number of singular values to manipulate is 64 in this case. However, as the energy is concentrated in the first singular values, one can consider for example only the first 30 values, the rest can be set to zero. As a result, denoising all the 70 (8×8) patches reduces to estimating about 30 unknowns or less. This is a powerful property that makes learning denoising in this domain interesting due to the low dimensionality of the domain transformation instead of simply shrinking the values with sparse models [2], [4]. Based on this observation, we propose a method that efficiently learns the low-rankness for denoising based on a pair of clean/training image. Efficient training and transfer is however not straightforward in this domain as it is not clear what are the best singular values to map to. For that, we need first to derive a formulation of the *optimal* denoising singular values. This is achieved via an inverse problem that takes into account both the clean and noisy training non-local matrices. Secondly, we propose a fast and efficient method to learn the correspondence between the noisy and derived optimal denoising values. As the derived optimal values do not correspond to "true" singular values because they are not necessary in decreasing order, learning a shrinkage function is not suitable in this domain. We rather propose to learn various full *mapping* functions between the

H. Badri's PhD is funded by an INRIA (Direction of Research) CORDI-S grant. His PhD is in co-supervision with INRIA Bordeaux Sud-Ouest and Mohammed V-Agdal University - LRIT, Associated Unit to CNRST (URAC 29) - Rabat, Morocco. E-mail: hicham.badri@inria.fr.

H. Yahia is with INRIA Bordeaux Sud-Ouest. E-mail: hussein.yahia@inria.fr.

D. Aboutajdine is with Mohammed V-Agdal University - LRIT, Associated Unit to CNRST (URAC 29). E-mail: aboutaj@ieee.org.

values via robust optimization of multiple singular values clusters for a richer model. Our training model is simple and fast while being efficient compared to more advanced function mapping learning techniques such as neural networks. Finally, we evaluate our technique on synthetic and real noise corruptions. First, we use uniform and non-uniform Gaussian noise via the CCD-CRF model for synthetic evaluation. Secondly, we use real-world camera noise samples extracted with the help of professional photographers for a more realistic noise evaluation. Finally, we evaluate the proposed method on real noisy images with unknown camera settings. Experiments show that the proposed approach leads to better denoising results even in the presence of challenging small-scale textures and text structures.

The paper is organized as follows. Sec. II gives an overview of related work on image restoration and noise modeling. Sec. III shows how and why we derive a closed-form of the optimal denoising singular values. Sec. IV presents the proposed learning model and solver of the optimization problem. The proposed denoising method is evaluated in Sec. V against some state-of-the-art methods in the case of synthetic noise, real noise samples and real noisy images. A conclusion and discussion are presented in the final section.

II. Related Work

In this section, we give an overview of the most recent existing denoising methods in the literature. There are mainly three approaches to image denoising that are classified in the literature as follows : 1) *internal-based*, where only the image structures within the noisy image are used to perform restoration, 2) *learning-based*, where learning is used to improve the recovery of the latent clean image, 3) *external-based*, where external information from correlated images is directly used to improve recovery.

A. Internal-Based Denoising

Internal-based denoising is solely based on using information within the noisy image itself. Early methods perform pixelwise operations by exploring neighboring pixels via Gaussian filtering, bilateral filtering [6], total variation [7] or wavelet thresholding [8]. Preliminary work on non-local processing [9] has shown to lead to a dramatic improvement. The standard Non-Local Means (NLM) means approach [9], [10] consists in simply perfoming weighted averaging. The idea has been generalized by gathering similar patches in a matrix (a.k.a. *non-local matrix*), process this matrix in a domain and reconstruct the patches back. Various domains of processing have been proposed such as 3D collaborative filtering transform (BM3D) [12] and singular values [14], [2], [4]. Other internal-based methods have been proposed such as denoising by exploring patch recurrence accross scales [15], matching gradient histograms [16] or exploiting marginal histograms [5].

B. Learning-Based Denoising

Learning has significantly improved image restoration quality. Methods [17], [18] permit to learn a dictionary to perform denoising via Orthogonal Matching Pursuit (OMP) locally [18] or non-locally [19], [20]. Learning priors via high-order MRF models [22], Gaussian mixture models [3], [11] or shrinkage functions [23], [24] has shown to give interesting results in image denoising. Plain learning via neural networks has shown to give interesting results as well [25].

However, as these methods try to learn one single model based on various clean sources, performance compared to BM3D is not impressive and challenging structures such as textures (grass, trees, grainy wall,...) and small-scale texts are still not well restored. Moreover, both internal and external-based methods suppose that the noise level is known and most of the time, the noise is supposed to be uniform (AWGN), which is far from the real case senario. As a result, even if these methods can be tuned to perform well in this case, they lose their potential on real noise corruption when exact noise estimation is almost impossible and does not reflect the complex camera pipeline.

C. External-Based Denoising

External-based denosing is a relatively new approach that consists in exploiting directly external information using a set of correlated images. The approach by Burger *et al.* propose to use learning to combining denoising results from internal and external results [28]. The method in [31] uses web images to recover correlated images and use external patches in BM3D. Similarly, methods in [29], [30] combine internal and external patches extracted from correlated images in the BM3D framework. While these methods improve considerably restoration quality, they require that the external correlated images should be *too similar* to the input noisy image containing the same patterns, which is only possible in specific scenarios.

We propose a powerful learning-based method that manipulates only the *internal* patch correlations of the noisy image. Our method performs processing on the non-local singular values similar to internal-based denoising methods SAIST [2] and WNNM [4]. However, unlike these methods that use simple shrinkage operators via sparse priors, we *learn* a *mapping* between the noisy singular values and the derived optimal denoising values. Learning in our method does not consist in learning dictionaries [17], [19] or shrinkage functions [23]. We rather learn a *mapping* similar to neural networks [25], but in the singular values domain, which reduces considerably the number of latent variables to infer. Contrary to previous learning-based methods such as [17], [19], [3], [25], our method does not need a very large amount of patches to learn a model that can take a lot of time, sometimes days to train one single model ; the proposed approach uses only one *single* image pair for training. Moreover, training is fast as it consists in solving few linear systems. This *transfer* approach that consists in learning a denoising model on a pair of clean/noisy pair has many advantages. First of all, we do not consider the Gaussian noise assumption ; a model can be trained easily for various types of corruption. Secondly, the denoising quality can be improved by choosing a pair that is somehow similar to the noisy instance. Thirdly, as

various camera devices respond differently, our method can use noise samples from the camera device at hand and adapt the learned model to the device. To demonstrate the potential of the proposed approach, we evaluate it against some leading methods on various types of noise : uniform Gaussian noise, non-uniform noise, Poisson noise, speckle noise, salt & pepper noise and real noise sampled from a camera. Moreover, by learning a model based on a Nikon D-600, we show how our method produces high-quality results even if the noise corruption is not the same as that of the training pair.

III. Problem Formulation

Let x_c be a clean image and x_n its corrupted version. Non-local image restoration exploits the self-similarities in natural images. The approach consists in gathering similar patches of x_n, stack them in a matrix X_n where the columns correspond to the vectorized versions of the patches, applying a transformation, reconstruct the estimated clean matrices $\hat{X}_c$, and finally reconstruct the clean image $\hat{x_c}$ by aggregating all the transformed matrices. Methods such as SAIST [2] and WNNM [4] use low-rank estimation as a transformation by imposing sparsity models on the singular values. More precisely, the problem can be posed in the following general formulation

$$\hat{X}_c = \underset{X_c}{\text{argmin}} \frac{1}{2}||X_c - X_n||_2^2 + \lambda\psi_*(X_c), \tag{1}$$

where ψ_* models the sparsity of the singular values of the clean matrix X_c. ψ_* can take the form of the nuclear norm [2] or the reweighted-nuclear norm [4]. While these methods tend to model the singular values of the clean matrices X_c, it turns out that *the singular values of the matrices X_c in the clean natural image do not correspond to the optimal denoising singular values*. This is because we do not have the true orthogonal matrices of X_c for the complete decomposition. The values that need to be modeled should rather depend on the whole decomposition of the *noisy* patches. In fact, the actual optimal values do not even correspond necessary to a true low-rank transformation. As a result, a sparsity-based model that leads to a shrinkage operator cannot take fully advantage of this observation.

Optimal Singular Values Denoising

Given a pair of clean/noisy non-local patch matrices X_c/X_n, we would like to derive the best denoising singular values that we denote by Σ_o. In other words, Σ_o are the *optimal* singular values that permit to reconstruct X_c from X_n. Suppose the Singular Value Decomposition (SVD) of both X_c and X_n

$$X_c = U_c\text{diag}(\Sigma_c)V_c^T\ ,\ X_n = U_n\text{diag}(\Sigma_n)V_n^T, \tag{2}$$

where diag is the diagonal matrix operator that extracts the diagonal from a matrix or builds a diagonal matrix from a vector. Typically, sparsity-based methods [2], [4] try to denoise X_n by applying a shrinkage operator (shrink(Σ_n)). By supposing that the corruption is dense, the best singular values Σ_o to estimate X_c from X_n can be recovered by minimizing the error between X_c and its reconstruction

$$\underset{\Sigma_o}{\text{argmin}}\, ||U_n\text{diag}(\Sigma_o)V_n^T - X_c||_F^2. \tag{3}$$

By rewriting the diagonal operator as a linear operation :

$$\text{diag}(A) = A\mathbb{1}, \tag{4}$$

where $\mathbb{1}$ is an all-ones vector. The solution of the problem is given as follows

$$\Sigma_o = U_n^T X_c V_n \mathbb{1} = \text{diag}(U_n^T X_c V_n). \tag{5}$$

As there is no particular prior on the solution, Σ_o do not necessarily correspond to "true" singular values. In fact, they can be even negative and in a non-decreasing order. As the energy is concentrated in the first values, we consider only around half of the values for training, the rest is set to zero.

First, we would like to verify denoising results given by the optimal values Σ_o and compare with previous shrinkage-based methods SAIST and WNNM to see how far are these methods from the optimal bound. We compare also denoising results given by using the clean values Σ_c. The results are conducted on the popular *house* and *barbara* images with Gaussian noise of standard deviation 50. As can be clearly seen in Fig. 1, the derived optimal denoising values Σ_o lead to much better denoising results. One can clearly see that the clean values Σ_c do not correspond to the right singular values for denoising.

IV. Learning the Low-Rank Mapping

Given a pair of noisy/optimal singular values Σ_n/Σ_o, we would like to learn a mapping function F such that $F(\Sigma_n) \approx \Sigma_o$. The good thing about the proposed approach is that denoising each set of similar patches is reduced to estimating few singular values, which makes learning easier. One approach would be to learn a shrinkage function as adopted in some techniques such as [23]. However, as explained before, the mapping $F(\Sigma_n) \approx \Sigma_o$ does not correspond to a shrinkage as the optimal singular values Σ_o *do not* correspond to real singular values : they are not necessarily in decreasing order, are hard to fit and they can be even negative. A general model for a shrinkage shrink(.) can be formulated as follows

$$\Sigma_o \approx \text{shrink}(\Sigma_n) = max(0, \Sigma_n - \beta w(\Sigma_n)), \tag{6}$$

where β is a postive regularization term and w is a weight function. This shrinkage function is the solution (or first-order solution) of the proximal operator of the form

$$\underset{\Sigma_o}{\text{argmin}} \frac{1}{2}||\Sigma_o - \Sigma_n||_2^2 + \beta\psi(\Sigma_o), \tag{7}$$

where ψ is a prior. Learning a shrinkage function S is reduced to learning the weight function $w(.) \geq 0$. However, as can be clearly seen in Eq. (6), a shrinkage function is by definition monotonic. As a result, learning a shrinkage function is not appropriate in our case because of the nature of Σ_o. Instead, we learn an actual point-by-point *mapping*.

(a) Ground-truth (b) Noisy (14.17 dB) (c) SAIST (29.87 dB) (d) WNNM (30.47 dB) (e) Σ_c-den. (28.64 dB) (f) Σ_o-den. (**32.02** dB)

(g) Ground-truth (h) Noisy (14.15 dB) (i) SAIST (27.30 dB) (j) WNNM (27.78 dB) (k) Σ_c-den. (25.90 dB) (l) Σ_o-den. (**29.21** dB)

Fig. 1: Non-local low-rank denoising comparison ($\sigma = 50$). As can be seen, the derived optimal denoising singular values Σ_o lead to better results compared to shrinkage-based methods SAIST and WNNM. The singular values of the clean non-local patches Σ_c clearly do not correspond to the best values to reconstruct X_c from X_n.

A. Transfer Problem Formulation

A straightforward way to estimate this mapping F is to use a neural network (function fitting neural network). While this approach works well in practice, it is time consuming especially if the training image is large. The method that we propose is a simple yet efficient learning approach that is fast as it requires solving few linear systems and can benefit from sparsity for robust estimation. We formulate the mapping as a linear model with a latent matrix W that minimizes the fitting error in a high-dimensional space. The problem is formulated as follows

$$\begin{aligned} \underset{W}{\text{argmin}} \frac{1}{Mp} \sum_{j=1}^{M} ||W^T \phi(\Sigma_{n,j}) - \Sigma_{o,j}||_p^p + \lambda ||W||_F^2 \\ F(\Sigma_n) = W^T \phi(\Sigma_n), \end{aligned} \tag{8}$$

where $\lambda = 0.001$ is a small regularization term to prevent numerical problems, $p \leq 1$ represents the l_p-norm for robust fitting and $\phi(.)$ is a non-linear mapping to a high-dimensional space. $\Sigma_{n,j}$ and $\Sigma_{o,j}$ correspond to the singular values of the matrices $X_{n,j}$ and $X_{c,j}$ extracted from the training pair. The total number of such matrices is M. Typically, $\phi(.)$ is a basis function that can take any form as long as it is independent of the weights. In our case, we use a polynomial basis function of the following form

$$\phi(\Sigma_{n,j}) = \begin{bmatrix} 1 & \Sigma_{n,j}^T & (\Sigma_{n,j}^T)^2 & (\Sigma_{n,j}^T)^3 & \dots \end{bmatrix}^T. \tag{9}$$

Optimization

To estimate a solution to the problem (8), we use a Majorization-Minimization (MM) approach [1]. Training consists in estimating the weights W minimized over all the singular values pairs Σ_n / Σ_o in the training image. Due to the use of the $l_{p\leq 1}$-norm to prevent the influence of outliers, the energy cannot be minimized directly via Euler-Lagrange equations. The MM method consists in majorizing the non-convex term leading to a more tractable energy

$$\begin{aligned} \underset{W,Y}{\text{argmin}} \tfrac{1}{Mp} \textstyle\sum_{j=1}^{M} \sum_{i=1}^{N} Y_i^{p/2} \\ \text{s.t. } \left(W^T \phi(\Sigma_{n,j}) - \Sigma_{o,j}\right)_i^2 \leq Y_i, \\ i = 1, ..., N, \; ||W||_F^2 \leq \xi, \end{aligned} \tag{10}$$

where N is the number of optimal singular values per non-local matrix. The relaxation (10) is in the following general form

$$\underset{v}{\text{argmin}}\, h(v) \text{ s.t. } v \in \mathcal{C}, \tag{11}$$

where $\mathcal{C}$ is a convex set and $h(v) = v^{p/2}$ is a concave function that admits the following linearization

$$v^{(l+1)} = \underset{v}{\text{argmin}}\, h(v^{(l)}) + \nabla h(v^{(l)})(v - v^{(l)}). \tag{12}$$

The estimated solution thus corresponds to a reweighted-least squares problem (IRLS) [33], [34]

$$\begin{aligned} W^{(l+1)} &= A^{(l)-1} \textstyle\sum_{j=1}^{M} \left(\phi(\Sigma_{n,j}) Z_j^{(l)} \Sigma_{o,j}{}^T\right) \\ A^{(l)} &= \textstyle\sum_{j=1}^{M} (\phi(\Sigma_{n,j}) Z_j^{(l)} \phi(\Sigma_{n,j})^T) + \lambda I \\ Z_j^{(l+1)} &= \text{diag}\left(\tfrac{1}{|\phi(\Sigma_{n,j})^T W^{(l)} - \Sigma_{o,j}^T|^{2-p} + \epsilon}\right), \end{aligned} \tag{13}$$

where $\epsilon = 0.001$ is set for stability. However, this is a slow estimate as the weights Z should be evaluated for each patch matrix. Instead, we use the following estimation that computes directly one least-squares solution instead of performing M estimation of the weights Z_j

$$\begin{aligned} W^{(l+1)} &\approx \left(\phi(\bar{\Sigma_n}) Z^{(l)} \phi(\bar{\Sigma_n})^T + \lambda I\right)^{-1} \left(\phi(\bar{\Sigma_n}) Z^{(l)} \bar{\Sigma_o}^T\right) \\ Z^{(l+1)} &= \text{diag}\left(\tfrac{1}{\frac{1}{M} \sum_{j=1}^{M} (|\phi(\Sigma_{n,j})^T W^{(l)} - \Sigma_{o,j}^T)|^{2-p} + \epsilon}\right), \end{aligned} \tag{14}$$

where $\bar{\Sigma_n}$ and $\bar{\Sigma_o}$ correspond to the matrices containing all the singular values for all the M patch matrices along the columns

$$\bar{\Sigma_n} = \left[\begin{array}{ccc} \Sigma_{n,1} & \cdots & \Sigma_{n,M} \end{array} \right], \bar{\Sigma_o} = \left[\begin{array}{ccc} \Sigma_{o,1} & \cdots & \Sigma_{o,M} \end{array} \right]. \tag{15}$$

This estimation is based on the observation that (8) aims at minimizing the mean error over all the M patch matrices. We thus take the re-weighting that corresponds to the mean over all the M weights Z_j.

B. Fitting a Richer Model

Problem (8) attempts to learn a mapping between a pair of noisy singular values and their corresponding optimal denoising values over all the patch matrices of the training image pairs, which corresponds to only one denoising pass. Non-local low-rank denoising methods [2], [4] are implemented in an iterative framework that adds the filtered noise back to the denoised image as follows

$$x_n^{(k+1)} = x_n^{(k)} + \gamma(x_n - x_n^{(k)}), \tag{16}$$

where k denotes the current iteration and γ is a positive regularization term. The initial image is simply the noisy input $x_n^{(0)} = x_n$. This iterative regularization technique used in previous works has shown to significantly improve restoration quality. We use the same approach where a new model W_k is learned at each iteration k.

Additionally, in order to fit a more flexible model, we gather the singular values in clusters and train a separate model for each cluster and for each iteration. The number of clusters R that we use in the experiments is typically around 3, not too high to prevent over-fitting. The proposed training method is summarized in Algorithm 1 and an overview of the training approach is given in Fig. 2.

Inference consists in a similar workflow, with a difference that the singular values $\Sigma_{n,j}^{(k)}$ are first used to determine the right cluster r by minimizing the l_2 distance, then the optimal singular values $\hat{\Sigma_{o,j}}^{(k)}$ are estimated as follows

$$\hat{\Sigma_{o,j}}^{(k,r)} = W^{(k,r)^T} \phi({\Sigma_{n,j}}^{(k,r)}), \tag{17}$$

where $W^{(k,r)}$ denotes the trained model for cluster number r during iteration k.

C. Extension to Full Color Denoising

Color images contain important correlations along the channels that need to be taken into account. Our method is easily extended to full color image denoising. Instead of processing the non-local matrices of each channel separately, we simply concatenate the color patches in one vector. As a result, we only need to learn LRT for few singular values per non-local color matrix, which is faster both for training and denoising. We call this full color denoising approach C-LRT and the grayscale version G-LRT.

Algorithm 1: Proposed LRT training method.

Data: Pair of clean/corrupted image(s).
Result: Training model
$(\{W_{r,k}\}, r = 1, ..., R, k = 1, ..., K)$.
Init : Compute the location of similar patches;
Main processing :
for *k=1 to K* **do**
- Compute the current image with Eq. (16);
- Gather the non-local matrices $X_{c,j}^{(k)}$ and $X_{n,j}^{(k)}$;
- Calculate the SVD decomposition of $X_{n,j}^{(k)}$ and $\Sigma_{o,j}^{(k)}$ via Eq. (2) and Eq. (5);
- Cluster the singular values into R clusters and estimate $W^{(k,r)}$ for each one via Eq. (14);
- Reconstruct the estimated clean matrices $\hat{X_j}^{(k)}$ via Eq. (17) and Eq. (2);
- Aggregate all the estimated matrices to form the restored image and store it in $x_n^{(k)}$;

D. Processing Time

The main bottleneck in both non-local and learning-based denoising methods is the processing time. Our training model is fast compared to previous learning-based methods that can take hours or even days for large data [25], [3], [23]. For an image from the Kodak dataset (500×750), learning the model (Eq. (8)) at each iteration for 3 clusters takes only 1.50 seconds for full color filtering and 0.70 seconds for the grayscale version. Predicting the denoising singular values takes 0.13 seconds for the full color filtering and 0.08 for the grayscale version. We have implemented both training and denoising in parallel using the Matlab Parallel Computing Toolbox on an Intel Xeon E5-2609 CPU (8 cores). It takes a total of 378 seconds to denoise the full color version and 136 seconds to denoise the grayscale version for 8 iterations (the total training time is similar). Compared to WNNM [4], our implementation is more than 11 times faster.

V. Experimental Results

In this section, we perform extensive experiments to show the potential of the proposed method. Experiments are first conducted on the standard Kodak dataset. For test images, we select 10 images of the dataset. For training, half of the training images are selected from the dataset itself, and the other half consists of external images recovered by choosing one single result from Google Image Search. It turns out that Google Image Search is quite robust to noise, able to give a good correlated image even if the input is noisy. These images are presented in Fig. 3.

A. Synthetic Noise Experiments

The first experiments consist in evaluating the proposed denoising against previous methods using the standard Gaussian noise setup with fixed standard deviation. While this model is far from more realistic noise models that we use in the next subsections, it gives us an idea about the potential of

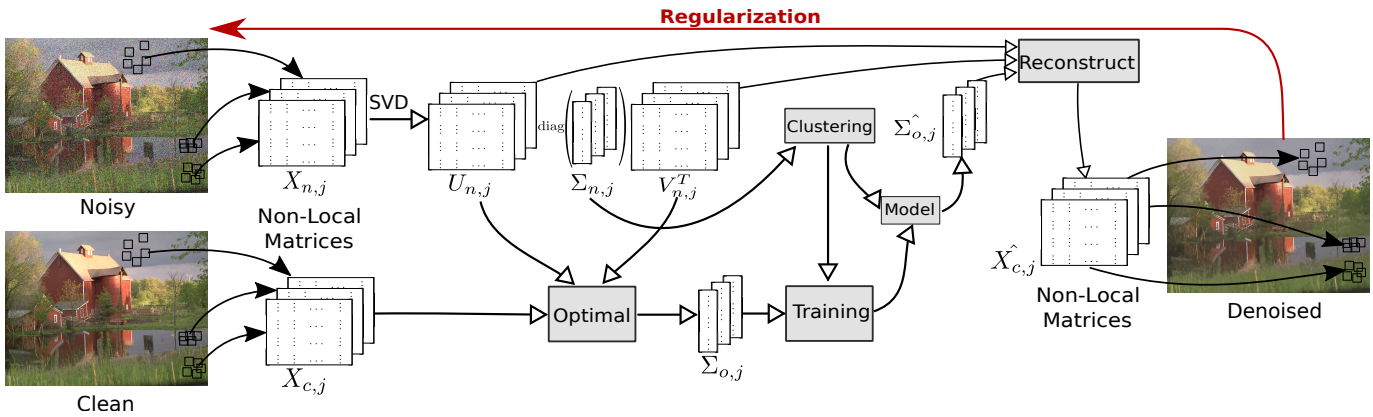


Fig. 2: Overview of the proposed training approach. First, the SVD decomposition of the noisy non-local matrices is used together with the clean non-local matrices to calculate the optimal singular values $\Sigma_{o,j}$ for denoising. The pair of noisy/optimal singular values is then used for training after clustering. Afterward, the denoised instance is used to regularize the noisy image and the process is repeated for several iterations.

the proposed method. In this case, the noisy image is generated as follows

$$x_n = x_c + \sigma\eta, \tag{18}$$

where η is here a Gaussian noise of zero mean and unit standard deviation. The noisy images are then truncated to fit values between 0 and 255 and the optimal noise level is re-estimated by calculating the standard deviation between the clean and noisy image. We compare with four leading methods : BM3D [12], EPLL [3], SAIST [2] and WNNM [4] for the grayscale case. For full color denoising, we compare with CBM3D [13]. The results are given in Table I. As can be seen, the proposed method leads to better empirical restoration in terms of both PSNR and SSIM [32] for higher noise level. It is worth noting that the proposed method successfully transfers denoising even if the training image is not correlated with the test image, which is not possible with external-based methods. Visual results are given in Fig. 5 that demonstrate the ability of the proposed method to better denoise textured regions such as grass, trees and water.

To demonstrate the effectiveness of the proposed learning-based method, we run experiments using 3 types of noise : Poisson, Speckle (with multiplicative factor 0.05) and Salt & Pepper (10% corruption). The results presented in Fig. 7 demonstrate the performance of the approach .

B. Synthetic Signal-Dependent Noise Experiments

The second set of experiments consists in evaluating the proposed method using a synthetic realistic noise level. In order to perform experiments with more realistic noise, we propose to use the CCD-CRF model that reflects better the non-uniformity of the corruption. This model is a simplification of the non-linear complicated imaging pipeline that includes various perturbations and noise corruptions. Following previous work [26], [27], the noise model is given as follows

$$x_n = f(L + n_s + n_c) + n_q, \tag{19}$$

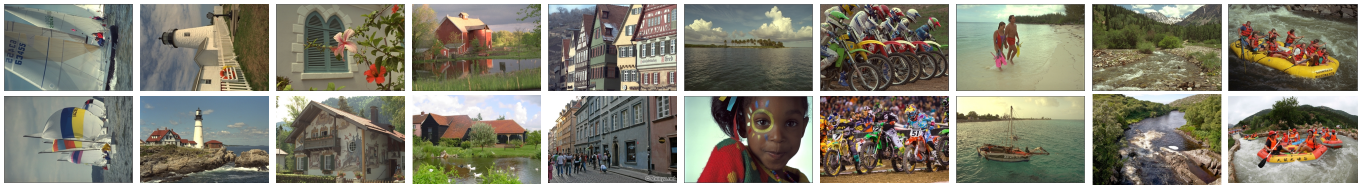
where L is the clean irradiance, x_n is the resulting corrupted image and $f(.) = \text{CRF}$ denotes the camera response function (CRF). This model considers mainly two types of noise : n_c that is independent of the signal before gamma correction and n_s represents all the noise components that depend on the irradiance L. n_q is the minimum camera noise that is ignored. As previous methods are tuned to work with Gaussian noise, the noises n_s and n_c are generated based on Gaussian noise as well. The noise n_s is zero-mean with variance $\text{Var}(n_s) = L\sigma_s^2$ and n_c is zero-mean with variance $\text{Var}(n_c) = \sigma_c^2$. This model is further simplified to take the signal-dependent variance model

$$x_n = x_c + \sigma_\eta(x_c)\eta, \tag{20}$$

where η is supposed a zero-mean independent random noise with unit standard deviation and the noise level σ_η depends on the intensity level x_c, described by the noise level function (NLF). For noise simulation, we use a simple noise synthesis technique proposed in previous works [26]. The CRF function is downloaded from the popular camera response database http://www.cs.columbia.edu/CAVE/databases/. Various methods to estimate the NLF are available [26], [27]. In this experiment however, we have access to the true CRF and the noise levels σ_s and σ_c, so we can recover the exact NLF using the technique in [26]. The CRFs and the corresponding NLFs (normalized to [0,1]) for two noise setups are given in Fig. 4. The empirical denoising results are given in Table I. As can be seen again, the proposed method leads to better denoising results, even in the case of non-uniform corruption. This demonstrates the ability of the proposed method to automatically adapt to internal structures of the image.

C. Real Noise Experiments

This experiment uses *real noise samples* directly instead of a synthetic noise model. We extract real camera noise sampled from a Canon 5D MK3, corrupt a ground-truth and denoise it. To perform this task, we first need a noise model to simulate the corrupted images. The sampled noise at a fixed ISO setting



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

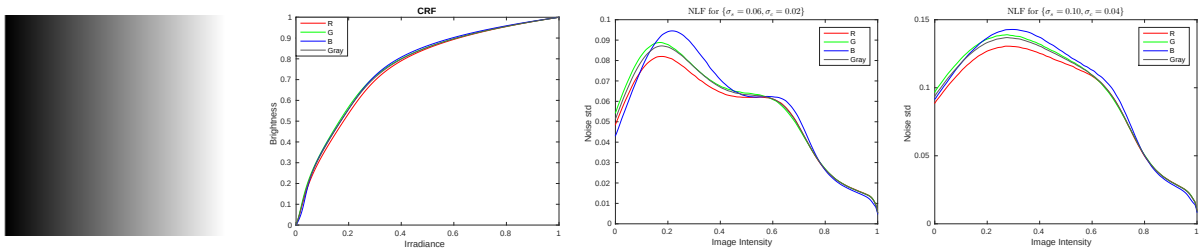
Fig. 3: Kodak dataset images used for the experiments. Top row are the test images and the bottom row are the training images.

	Uniform Gaussian $\sigma = 25$							Uniform Gaussian $\sigma = 50$						
Images	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL
(a)	33.56 dB	33.72 dB	30.33 dB	31.66 dB	31.72 dB	31.92 dB	31.34 dB	30.13 dB	29.92 dB	28.34 dB	28.14 dB	28.00 dB	28.30 dB	27.80 dB
	0.8859	0.8914	0.8459	0.8517	0.8507	0.8570	0.8406	0.8181	0.8195	0.7732	0.7652	0.7617	0.7715	0.7392
(b)	31.77 dB	31.66 dB	30.02 dB	29.99 dB	29.94 dB	30.20 dB	29.65 dB	28.69 dB	28.27 dB	27.33 dB	27.19 dB	27.15 dB	27.35 dB	26.49 dB
	0.8521	0.8518	0.8064	0.8016	0.7964	0.8069	0.8042	0.7786	0.7623	0.7384	0.7214	0.7229	0.7230	0.6990
(c)	33.76 dB	33.55dB	31.74 dB	31.71 dB	31.90 dB	32.11 dB	31.39 dB	29.65 dB	29.20 dB	28.11 dB	28.03 dB	27.96 dB	28.22 dB	27.54 dB
	0.9286	0.9272	0.9023	0.8998	0.9049	0.9084	0.8913	0.8702	0.8619	0.8232	0.8181	0.8255	0.8266	0.7890
(d)	30.93 dB	30.75 dB	29.38 dB	29.30 dB	29.31 dB	29.45 dB	29.44 dB	27.85 dB	27.56 dB	26.90 dB	26.78 dB	26.55 dB	26.85 dB	26.72 dB
	0.8261	0.8192	0.7558	0.7503	0.7469	0.7561	0.7610	0.7136	0.7029	0.6558	0.6491	0.6369	0.6536	0.6450
(e)	29.74 dB	29.53 dB	27.51 dB	27.43 dB	27.53 dB	27.81 dB	26.79 dB	26.01 dB	25.29 dB	23.91 dB	23.38 dB	23.54 dB	23.72 dB	22.98 dB
	0.8924	0.8953	0.8465	0.8487	0.8480	0.8584	0.8317	0.8075	0.8070	0.7381	0.7257	0.7296	0.7466	0.7003
(f)	32.36 dB	32.24 dB	30.28 dB	30.14 dB	30.15 dB	30.36 dB	30.10 dB	29.19 dB	28.79 dB	27.71 dB	27.45 dB	27.19 dB	27.49 dB	27.12 dB
	0.8661	0.8594	0.7995	0.7895	0.7835	0.7960	0.7858	0.7517	0.7449	0.6789	0.6699	0.6591	0.6706	0.6490
(g)	29.78 dB	29.54 dB	27.45 dB	27.31 dB	27.48 dB	27.57 dB	27.27 dB	25.74 dB	25.00 dB	23.99 dB	23.41 dB	23.45 dB	23.52 dB	23.46 dB
	0.8841	0.8837	0.8154	0.8151	0.8169	0.8240	0.8198s	0.7574	0.7399	0.6685	0.6467	0.6406	0.6486	0.6514
(h)	33.57 dB	33.36 dB	31.69 dB	31.88 dB	31.69 dB	32.02 dB	31.64 dB	30.51 dB	29.68 dB	29.62 dB	28.77 dB	28.74 dB	28.98 dB	28.33 dB
	0.8600	0.8623	0.8043	0.8217	0.8105	0.8223	0.8121	0.7835	0.7899	0.7564	0.7525	0.7504	0.7516	0.7319
(i)	27.57 dB	27.47 dB	25.13 dB	24.91 dB	25.12 dB	25.24 dB	25.28 dB	23.67 dB	23.55 dB	22.03 dB	21.71 dB	21.97 dB	22.02 dB	22.12 dB
	0.8281	0.8104	0.7295	0.6919	0.6932	0.7148	0.7323	0.6463	0.6138	0.5314	0.5007	0.5055	0.5282	0.5360
(j)	30.11 dB	29.88 dB	28.22 dB	28.05 dB	28.10 dB	28.25 dB	28.24 dB	26.68 dB	25.96 dB	25.34 dB	24.99 dB	24.93 dB	25.07 dB	25.08 dB
	0.8339	0.8273	0.7536	0.7466	0.7427	0.7553	0.7605	0.7052	0.6901	0.6253	0.6171	0.6068	0.6231	0.6250
Mean	**31.31** dB	31.17 dB	29.17 dB	29.24 dB	29.29 dB	**29.49** dB	29.11 dB	**27.81** dB	27.32 dB	**26.33** dB	25.99 dB	25.95 dB	26.15 dB	25.76 dB
	0.8657	0.8628	0.8059	0.8017	0.7994	**0.8099**	0.8039	**0.7632**	0.7532	**0.6989**	0.6866	0.6839	0.6943	0.6766

	Uniform Gaussian $\sigma = 70$							Uniform Gaussian $\sigma = 100$						
Images	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL
(a)	28.31 dB	27.73 dB	26.87 dB	26.41 dB	26.28 dB	26.54 dB	26.00 dB	26.47 dB	25.04 dB	24.99 dB	24.22 dB	23.79 dB	24.16 dB	23.70 dB
	0.7731	0.7710	0.7311	0.7113	0.7196	0.7262	0.6885	0.7216	0.7204	0.6755	0.6550	0.6701	0.6802	0.6366
(b)	26.94 dB	26.11 dB	25.82 dB	25.45 dB	25.38 dB	25.55 dB	24.38 dB	25.11 dB	23.31 dB	23.97 dB	23.22 dB	22.99 dB	23.32 dB	22.01 dB
	0.7361	0.7165	0.6989	0.6756	0.6795	0.6797	0.6398	0.6865	0.6717	0.6413	0.6198	0.6346	0.6480	0.5758
(c)	27.28 dB	26.62 dB	26.17 dB	26.09 dB	26.13 dB	26.15 dB	25.38 dB	24.86 dB	23.75 dB	24.09 dB	23.86 dB	23.40 dB	23.73 dB	23.02 dB
	0.8160	0.8104	0.7643	0.7606	0.7690	0.7733	0.7196	0.7457	0.7517	0.6746	0.6819	0.6977	0.7105	0.6374
(d)	26.47 dB	25.75 dB	25.62 dB	25.52 dB	25.25 dB	25.51 dB	25.23 dB	24.77 dB	23.22 dB	24.26 dB	23.66 dB	23.19 dB	23.55 dB	23.21 dB
	0.6594	0.6512	0.6039	0.6028	0.5998	0.6096	0.5935	0.5995	0.5947	0.5580	0.5523	0.5526	0.5637	0.5424
(e)	24.01 dB	22.97 dB	22.13 dB	21.42 dB	21.40 dB	21.59 dB	20.96 dB	21.79 dB	20.28 dB	20.11 dB	19.24 dB	19.07 dB	19.30 dB	18.67 dB
	0.7381	0.7386	0.6601	0.6425	0.6457	0.6603	0.6138	0.6343	0.6331	0.5512	0.5407	0.5344	0.5548	0.4993
(f)	28.19 dB	27.41 dB	26.64 dB	26.06 dB	25.88 dB	26.08 dB	25.53 dB	26.32 dB	23.77 dB	25.56 dB	23.94 dB	23.78 dB	23.95 dB	23.24 dB
	0.7047	0.6985	0.6317	0.6176	0.6140	0.6216	0.5992	0.6232	0.6205	0.5952	0.5711	0.5815	0.5888	0.5606
(g)	23.84 dB	22.65 dB	22.32 dB	21.57 dB	21.40 dB	21.53 dB	21.43 dB	21.78 dB	19.95 dB	20.76 dB	19.54 dB	19.27 dB	19.44 dB	19.21 dB
	0.6738	0.6472	0.5792	0.5571	0.5393	0.5505	0.5507	0.5574	0.5374	0.4855	0.4619	0.4401	0.4552	0.4486
(h)	29.22 dB	27.09 dB	28.37 dB	26.77 dB	26.71 dB	26.96 dB	26.03 dB	27.62 dB	23.65 dB	26.84 dB	23.96 dB	23.98 dB	24.27 dB	22.98 dB
	0.7541	0.7515	0.7305	0.7202	0.7215	0.7238	0.6950	0.7300	0.7188	0.7008	0.6843	0.7016	0.7095	0.6618
(i)	22.21 dB	21.68 dB	20.76 dB	20.47 dB	20.62 dB	20.65 dB	20.63 dB	20.49 dB	19.60 dB	19.73 dB	19.15 dB	19.05 dB	19.14 dB	18.98 dB
	0.5515	0.5155	0.4483	0.4215	0.4383	0.4442	0.4375	0.4296	0.4119	0.3540	0.3429	0.3372	0.3442	0.3388
(j)	25.06 dB	23.74 dB	24.10 dB	23.37 dB	23.20 dB	23.35 dB	23.23 dB	23.09 dB	21.00 dB	22.76 dB	21.36 dB	21.08 dB	21.29 dB	20.98 dB
	0.6361	0.6135	0.5611	0.5483	0.5381	0.5498	0.5488	0.5472	0.5245	0.5020	0.4781	0.4661	0.4789	0.4728
Mean	**26.15** dB	25.18 dB	**24.88** dB	24.31 dB	24.23 dB	24.39 dB	23.88 dB	**24.23** dB	22.36 dB	**23.31** dB	22.21 dB	21.96 dB	22.22 dB	21.60 dB
	0.7043	0.6914	**0.6409**	0.6258	0.6265	0.6339	0.6086	**0.6275**	0.6185	**0.5738**	0.5588	0.5616	0.5734	0.5374

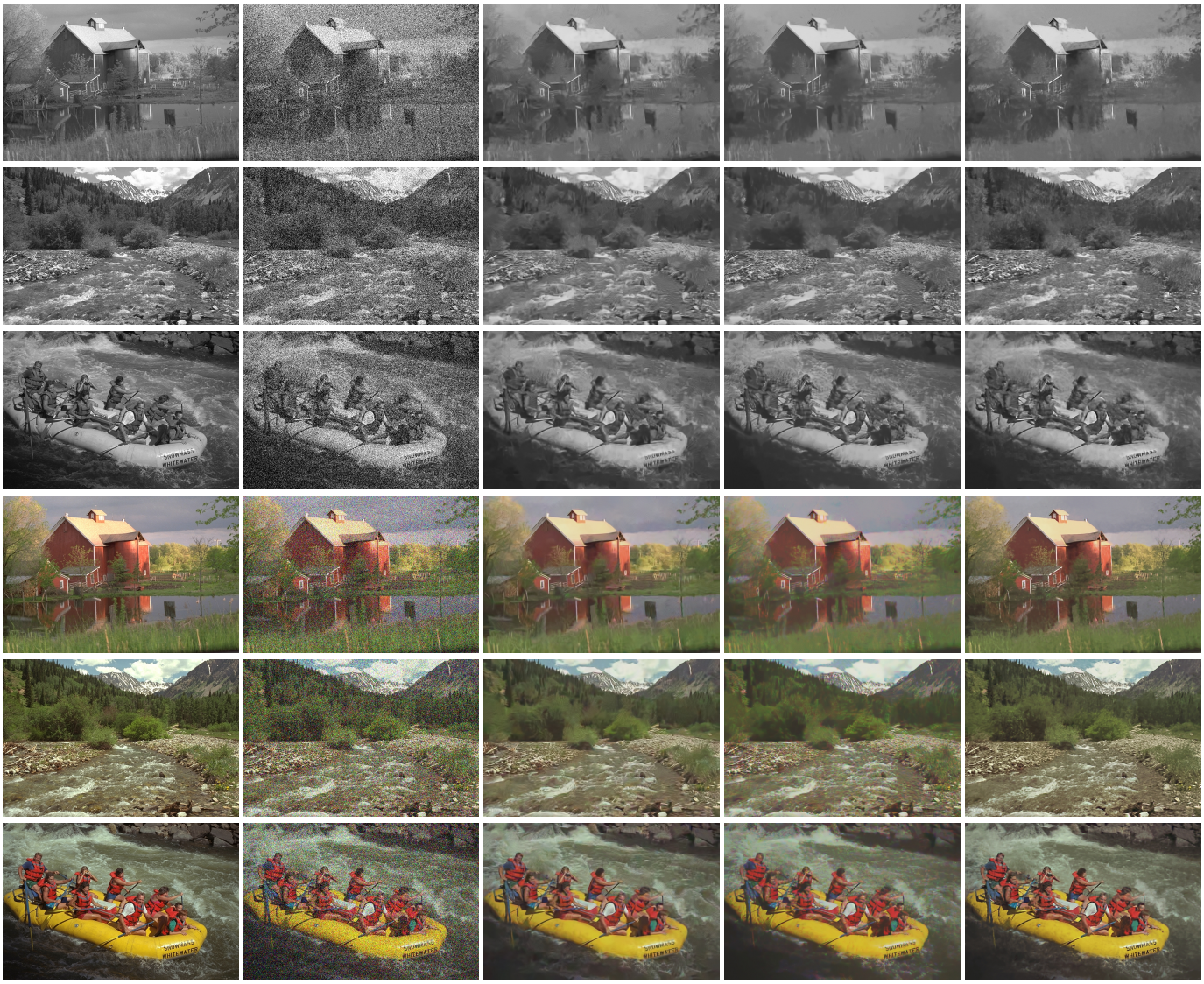
	Non-Uniform Gaussian ($\sigma_c = 0.02, \sigma_s = 0.06$)							Non-Uniform Gaussian ($\sigma_c = 0.04, \sigma_s = 0.10$)						
Images	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL	C-LRT	CBM3D	G-LRT	BM3D	SAIST	WNNM	EPLL
(a)	35.87 dB	35.30 dB	34.21 dB	33.66 dB	32.90 dB	33.56 dB	32.65 dB	33.15 dB	32.67 dB	31.19 dB	31.08 dB	30.78 dB	30.73 dB	30.17 dB
	0.9253	0.9101	0.9103	0.8927	0.8782	0.8866	0.8731	0.8948	0.8776	0.8658	0.8503	0.8487	0.8491	0.8253
(b)	33.364 dB	33.24 dB	31.51 dB	31.41 dB	30.96 dB	31.22 dB	30.64 dB	30.98 dB	30.85 dB	29.20 dB	29.31 dB	29.15 dB	29.09 dB	28.42 dB
	0.8845	0.8905	0.8431	0.8455	0.8422	0.8437	0.8410	0.8265	0.8244	0.7908	0.7793	0.7827	0.7745	0.7612
(c)	35.66 dB	35.10 dB	33.69 dB	33.16 dB	32.81 dB	33.45 dB	32.18 dB	32.72 dB	32.27 dB	30.66 dB	30.38 dB	30.44 dB	30.48 dB	29.76 dB
	0.9491	0.9414	0.9297	0.9204	0.9126	0.9195	0.9028	0.9192	0.9079	0.8839	0.8732	0.8777	0.8761	0.8560
(d)	32.40 dB	32.23 dB	31.08 dB	30.69 dB	30.50 dB	30.67	30.44 dB	29.87 dB	29.71 dB	28.64 dB	28.51 dB	28.40 dB	28.46 dB	28.27 dB
	0.8627	0.8557	0.8227	0.8008	0.7933	0.7977	0.7987	0.7861	0.7773	0.7260	0.7165	0.7112	0.7163	0.7137
(e)	32.07 dB	31.50 dB	29.96 dB	29.45 dB	29.35 dB	29.52	28.43 dB	28.97 dB	28.40 dB	26.99 dB	26.57 dB	26.56 dB	26.49 dB	25.50 dB
	0.9200	0.9082	0.8924	0.8783	0.8739	0.8752	0.8506	0.8704	0.8603	0.8323	0.8176	0.8202	0.8130	0.7794
(f)	33.75 dB	33.32 dB	31.72 dB	31.39 dB	31.04 dB	31.26 dB	30.61 dB	31.21 dB	30.96 dB	29.38 dB	29.25 dB	29.19 dB	29.33 dB	28.80 dB
	0.8954	0.8843	0.8470	0.8331	0.8195	0.8251	0.8150	0.8315	0.8233	0.7628	0.7557	0.7537	0.7611	0.7436
(g)	31.45 dB	31.31 dB	29.20 dB	28.89 dB	29.00 dB	29.12 dB	28.59 dB	28.48 dB	28.25 dB	26.36 dB	26.12 dB	26.28 dB	26.25 dB	26.01 dB
	0.9123	0.9125	0.8623	0.8568	0.8568	0.8618	0.8517	0.8488	0.8447	0.7753	0.7678	0.7728	0.7731	0.7687
(h)	35.87 dB	34.14 dB	34.58 dB	33.23 dB	31.98 dB	32.53 dB	31.83 dB	33.38 dB	32.034 dB	32.13 dB	30.78 dB	28.78 dB	28.75 dB	28.75 dB
	0.9055	0.8713	0.8842	0.8468	0.8159	0.8248	0.8145	0.8540	0.8181	0.8267	0.7773	0.7345	0.7453	0.7289
(i)	29.62 dB	29.58 dB	26.99 dB	26.93 dB	27.08 dB	27.07 dB	26.98 dB	26.57 dB	26.37 dB	24.28 dB	24.03 dB	24.27 dB	24.20 dB	24.29 dB
	0.8786	0.8735	0.8022	0.7830	0.7916	0.7910	0.8030	0.7766	0.7581	0.6757	0.6446	0.6520	0.6640	0.6777
(j)	31.62 dB	31.39 dB	29.77 dB	29.39 dB	29.31 dB	29.45 dB	29.20 dB	28.95 dB	28.69 dB	27.25 dB	27.08 dB	26.98 dB	27.01 dB	27.04 dB
	0.8722	0.8641	0.8137	0.7937	0.7870	0.7872	0.7901	0.7903	0.7841	0.7128	0.7035	0.6929	0.7044	0.7083
Mean	**33.18** dB	32.71 dB	**31.27** dB	30.82 dB	30.50 dB	30.79 dB	30.16 dB	**30.43** dB	30.02 dB	**28.61** dB	28.31 dB	28.08 dB	28.08 dB	27.70 dB
	0.9006	0.8912	**0.8608**	0.8451	0.8371	0.8413	0.8340	**0.8398**	0.8276	**0.7852**	0.7686	0.7646	0.7677	0.7563

TABLE I: PSNR and SSIM results on Kodak dataset for uniform and non-uniform Gaussian noise.



(a) Pattern (b) CRFs (c) NLFs (mid level) (d) NLFs (high level)

Fig. 4: CRFs and simulated NLFs on the pattern (a) used in the non-uniform Gaussian noise experiments for mid and high noise setups.



(a) Ground-truth (b) Noisy (c) BM3D/CBM3D (d) WNNM (e) Proposed

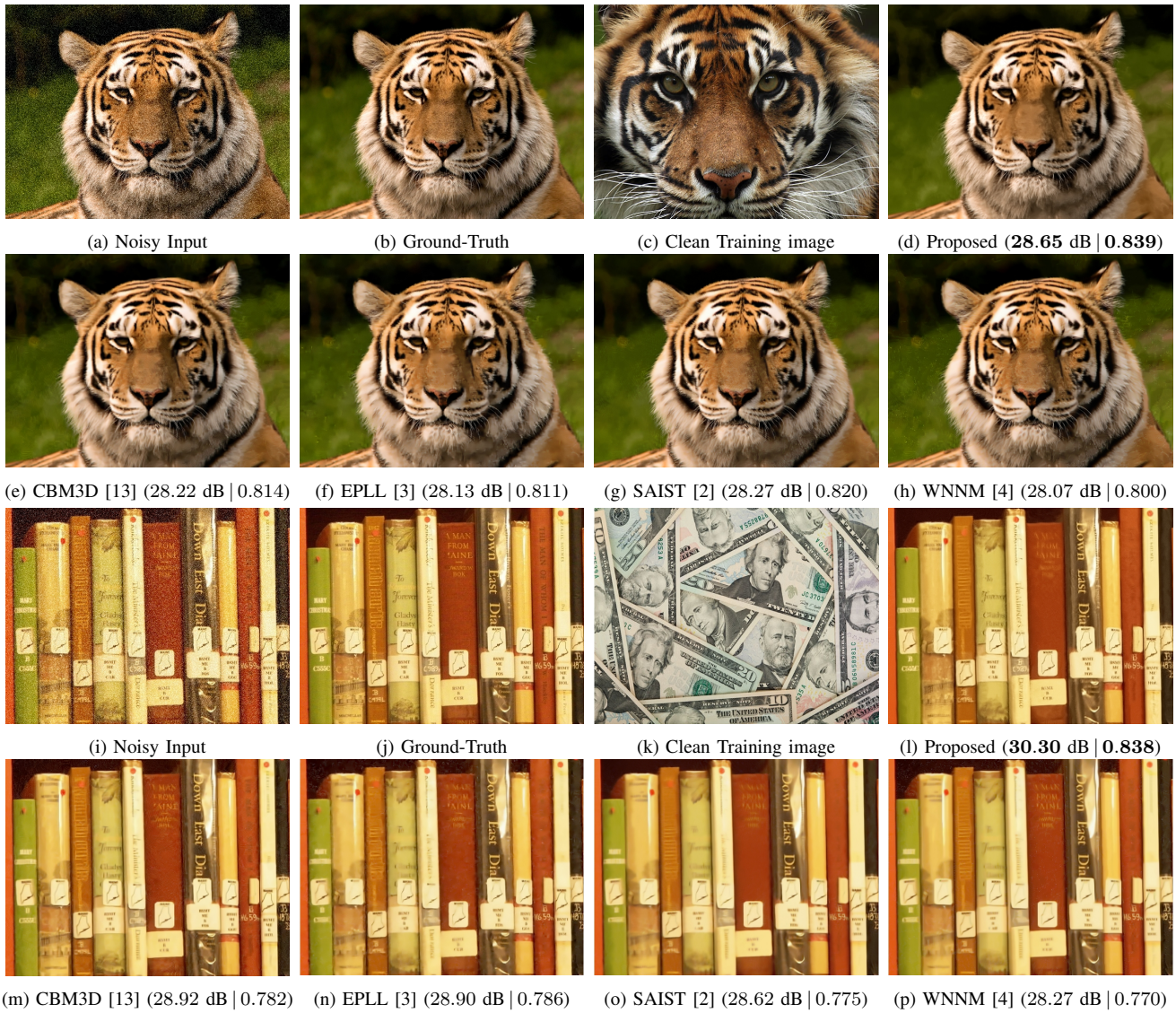
Fig. 5: Image denoising with high amount of uniform Gaussian noise ($\sigma = 70$). Note how the proposed method is able to recover challenging structures such as textures.

corresponds to blank frames with known unique background color (typically black) that we note η. We first zero-mean η to remove the background color and normalize the standard deviation to control the amount of corruption with a parameter σ_η. Given a clean color image x_c, we generate the noisy output x_n based on the noise level function (NLF) as follows

$$x_n = x_c + \sigma_\eta \text{NLF}(x_c/255, \eta). \tag{21}$$

Following previous works, the NLF can be estimated from the CRF and the variances $\text{Var}(n_c), \text{Var}(n_s)$. As we do not have access to the CRF of the Canon 5D MK3 device, we use a simple NLF that gives less weights to brighter pixels and more weights to darker pixels similar to standard NLFs. For each image, we fine-tune the noise level to get the best denoising result with methods BM3D, EPLL, SAIST and WNNM as the noise in this case is not uniform. In fact, each method required a different optimal noise level and standard methods for noise estimation failed to give an accurate estimation. To demonstrate the ability of our method to recover structures such as text and textures, we use two challenging images : one of a tiger with a fur and another one of books with various text sizes. The results are presented in Fig. 6. Note how the proposed method is able to recover the fur of the tiger and the text of the books. The other methods, even-though fine-tuned, result in over-smoothing. It is worth noting that, in the second example, the training image is not even visually close to the ground-truth, but contains text. Our training model is able to capture such structures and successfully transfer it to another noisy image.

Furthermore, to demonstrate the flexibility of the proposed method, we present denoising results using three different training images in Fig. 8. As can be seen, the results are of high-quality even-though the training image pair is not necessarily correlated with the latent image.



(a) Noisy Input (b) Ground-Truth (c) Clean Training image (d) Proposed (**28.65** dB | **0.839**)

(e) CBM3D [13] (28.22 dB | 0.814) (f) EPLL [3] (28.13 dB | 0.811) (g) SAIST [2] (28.27 dB | 0.820) (h) WNNM [4] (28.07 dB | 0.800)

(i) Noisy Input (j) Ground-Truth (k) Clean Training image (l) Proposed (**30.30** dB | **0.838**)

(m) CBM3D [13] (28.92 dB | 0.782) (n) EPLL [3] (28.90 dB | 0.786) (o) SAIST [2] (28.62 dB | 0.775) (p) WNNM [4] (28.27 dB | 0.770)

Fig. 6: Synthetic realistic image denoising experiments with (PSNR | SSIM). The proposed approach is able to to adapt to other types of noise (best viewed in full resolution).

D. Real Noisy Images Experiments

We evaluate in this section the proposed method on true real noisy images with unknown camera model and ISO settings. The images used in these experiments were taken with different devices. For training, we use one model trained on a Nikon D-600 device for a high ISO setting (see Fig. 10). To get the best results with previous methods, we generate various denoised outputs for various noise levels and choose the best result. We compare as well with the popular software NeatImage that is one of the best denoising softwares available. As can be seen in Fig. 9, our model trained on the real clean/noisy pair is able to produce high-quality results, preserving details such as text (see close-up). The other methods need noise estimation that is very hard when no uniform regions are available in the image (in which case NeatImage fails at automatically settings parameters). We run another experiment on a popular real noisy image instance *cave*. This image contains both rich textures and smooth regions, which is very challenging. Unfortunately, the clean/noisy training pair that we have (Fig. 10) does not contain enough rich textures to be transfered to this image. In this case, we sample directly the noise from the input, duplicate the noise pattern, and add it to a training image. The (cropped) results are given in Fig. 11. Note how the proposed method is able to recover the texture of the wall, which is very challenging in this case. The full resolution results are given in the supplementary material.

VI. Conclusion

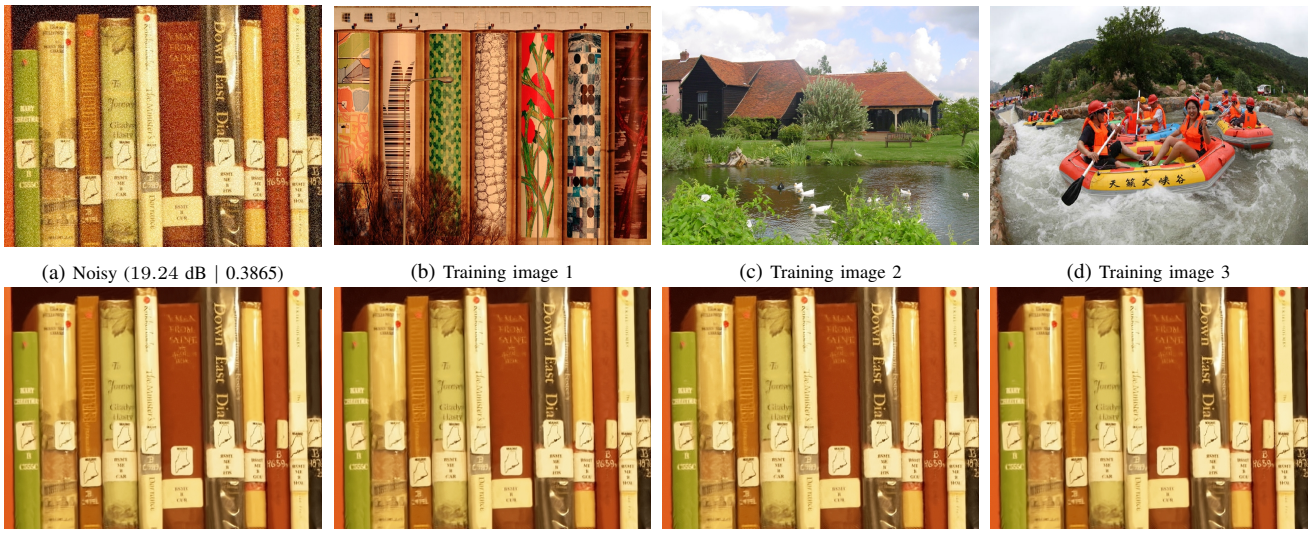
We present in this paper a new approach to image denoising based on the principle of non-local low-rankness transfer (LRT). Instead of using shrinkage operators to apply on the non-local singular values, we learn a mapping using a simple but efficient training model. This approach can support various types of corruption. Experiments on uniform and non-uniform noise as well as real camera noise show the ability of the proposed method to successfully recover challenging

(a) Ground-truth (b) Poisson (27.78 dB | 0.6577) (c) Speckle (19.97 dB | 0.3571) (d) Salt & Pepper (15.40 dB | 0.1912)

(e) Ground-truth (f) CBM3D (34.98 dB | 0.9141) (g) CBM3D (28.254 dB | 0.7218) (h) CBM3D (27.11 dB | 0.7001)

(i) Ground-truth (j) Proposed (35.42 dB | 0.9264) (k) Proposed (30.03 dB | 0.8270) (l) Proposed (27.72 dB | 0.7434)

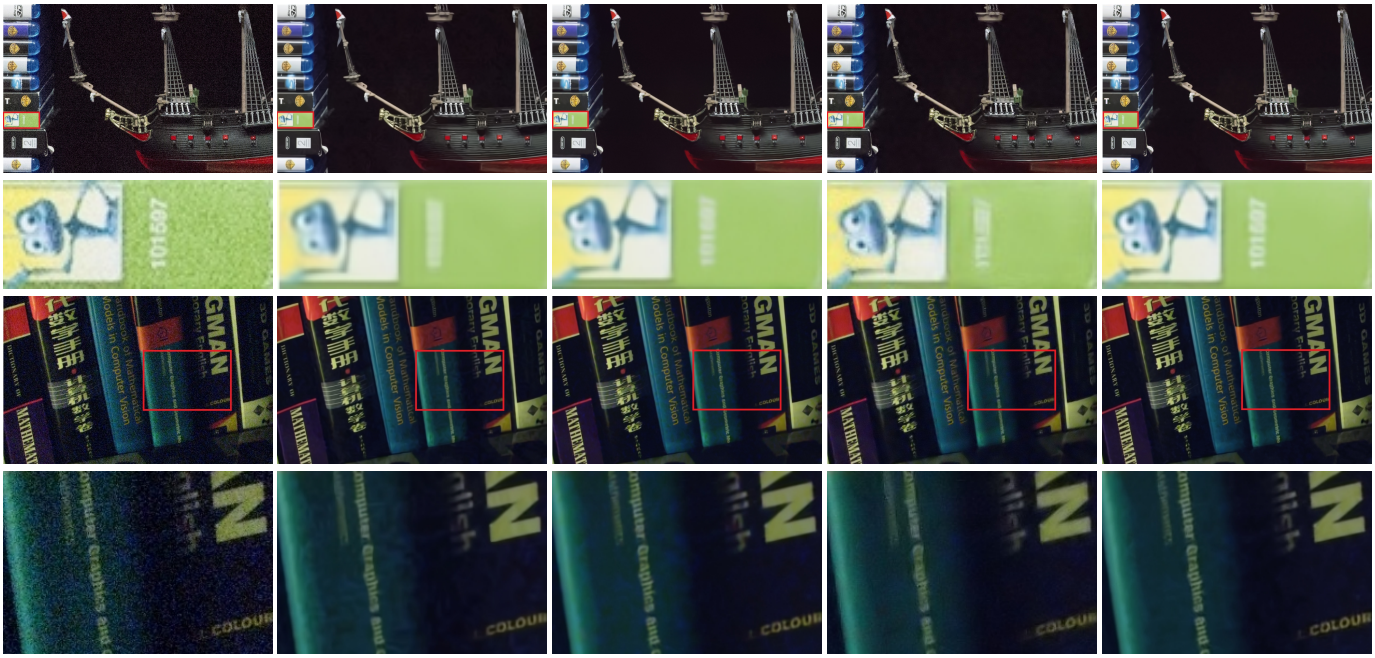
Fig. 7: Denoising results with different types of noise. The first line presents noisy results, the second and third lines represent respectively denoising results produced with CBM3D and the proposed method with PSNR and SSIM. As can be seen, the proposed approach produces high-quality restoration results.



(a) Noisy (19.24 dB | 0.3865) (b) Training image 1 (c) Training image 2 (d) Training image 3

(e) Oracle (30.70 dB | 0.8551) (f) Result using (b) (30.53 dB | 0.8506) (g) Result using (c) (30.47 dB | 0.8475) (h) Result using (d) (30.49 dB | 0.8493)

Fig. 8: Image denoising results with various training images. The oracle image is produced by using the clean ground-truth image as the training image (optimal). The proposed method produces fairly good results even using one single training pair that is not necessarily correlated with the latent image.



(a) Noisy input (b) CBM3D (c) SAIST (d) NeatImage (e) Proposed

Fig. 9: Real Image denoising using one single training model based on Nikon D-600 device (see Fig. 10). Without estimating the noise level, our method is able to produce high-quality denoising results.

(a) Clean (ISO-100) (b) Noisy (ISO-25600)

Fig. 10: Real clean/noisy training pair based on a Nikon D-600.

details and textures that are over-smoothed with previous methods even when fine-tuned. Note that in this paper, we use only one single training image and we do not fine-tune the training parameters. Our approach can be greatly improved by using better training images, automatically selecting the best training parameters via cross-validation and/or designing a new learning method such as neural networks. We believe the LRT approach can be useful not just for image restoration but for other low-rank based approaches as well that require low-rank estimation.

Acknowledgement

We would like to thank members of the blog http://photographylife.com/ for granting permission to use their images in this work. We also would like to thank photographers of http://publicdomainpictures.net/ and reaserchers for publicly providing image samples.

References

[1] K. Lange, "Optimization", in *Springer Texts in Statistics*, 2004.

[2] W. Dong, G. Shi, and X. Li, "Nonlocal Image Restoration with Bilateral Variance Estimation : A Low-Rank Approach", in *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 700-711, 2013.

[3] D. Zoran and Y. Weiss, "From Learning Models of Natural Image Patches to Whole Image Restoration", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 479 - 486, 2011.

[4] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted Nuclear Norm Minimization with Application to Image Denoising", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 2862-2869, 2014.

[5] X. Mei, W. Dong, B-G. Hu, and S. Lyu, "UniHIST: A Unified Framework for Image Restoration With Marginal Histogram Constraints", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 3753-3761, 2015.

[6] C. Tomasi, and R. Manduchi, "Bilateral Filtering for Gray and Color Images", in *IEEE Int. Conf. on Computer Vis.*, pp. 839-846, 1998.

[7] C. Rudin, S. Osher, and E. Fatemi, "Nonlinear Total Variation Based Noise Removal Algorithms", In *Physica D*, vol. 60, Issue 1-4, pp. 259-268, Nov. 1, 1992.

[8] J. Portilla, V. Strela, M.J. Wainwright, and E. P. Simoncelli, "Image Denoising Using Scale Mixtures of Gaussians in the Wavelet Domain", in *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338-1351, 2003.

[9] A. Buades, B. Coll, and J. M. Morel, "A Non-local Algorithm for Image Denoising", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 60-65, 2005.

[10] C-T. Huang, "Bayesian Inference for Neighborhood Filters with Application in Denoising", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 1657-1665, 2015.

[11] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng, "Patch Group Based Nonlocal Self-Similarity Prior Learning for Image Denoising", in *IEEE Int. Conf. on Computer Vis.*, to appear, 2015.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering", in *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080-2095, 2007.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color Image Denoising via Sparse 3D Collaborative Filtering with Grouping Constraint in Luminance-Chrominance Space", in *IEEE Int. Conf. on Image Process.*, pp. 313-316, 2007.

[14] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image Denoising Using the Higher Order Singular Value Decomposition", in *IEEE Trans. on Pattern Anal. and Machine Intel.*, vol. 35, no. 4, pp. 849-862, 2013.

[15] M. Zontak, I. Mosseri, and M. Irani, "Separating Sginal from Noise Using Patch recurrence across scales", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 1195-1202, 2013.

(a) Noisy Input (b) CBM3D (c) EPLL (d) SAIST (e) WNNM (f) Proposed

Fig. 11: Real image denoising example on a challenging example (close-up) that shows the details recovered with the proposed method.

[16] W. Zuo, L. Zhang, C. Song, and D. Zhang, "Texture Enhanced Image Denoising via Gradient Histogram Preservation", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 1203-1210, 2013.

[17] M. Aharon, L. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation", in *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311-4322, 2006.

[18] J. Mairal, M. Elad, and G. Sapiro, "Sparse Representation for Color Image Restoration", in *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53-69, 2008.

[19] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local Sparse Models for Image Restoration", in *IEEE Int. Conf. on Computer Vis.*, pp. 2272-2279, 2009.

[20] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally Centralized Sparse Representation for Image Restoration", in *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620-1630, 2013.

[21] H. Talebi, and P. Milanfar, "Global Image Denoising", in *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 755-768, 2014.

[22] S. Roth, and M. J. Black, "Fields of Experts", in *Inter. Journal of Computer Vis.*, vol. 82, no. 2, pp. 205-229, 2009.

[23] U. Schmidt and S. Roth, "Shrinkage Fields for Effective Image Restoration", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 2774-2781, 2014.

[24] Y. Chen, W. Yu, and T. Pock, "On learning optimized reaction diffusion processes for effective image restoration", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 5261-5269, 2015.

[25] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image Denoising : Can Plain Neral Networks Compete with BM3D?", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 2392-2399, 2012.

[26] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang, "Noise Estimation from a Single Image", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 901-908, 2006.

[27] J. Yang, Z. Gan, Z. Wu, and C. Hou, "Estimation of Signal-Dependent Noise Level Function in Transform Domain via a Sparse Recovery Model", in *IEEE Trans. Image Process.*, vol. 24, no. 5, 2015.

[28] B. C. Burger, C. Schuler, and S. Harmeling, "Learning How to Combine Internal and External Denoising Methods", in *Pattern Recognition*, pp. 121-130, 2013.

[29] H. Yue, X. Sun, J. Yang, and F. Wu, "CID: Combined Image Denoising in Spatial and Frequency Domains Using Web Images", in *Proc. IEEE Conf. Computer Vis. and Pattern Recognit.*, pp. 2933-2940, 2014.

[30] H. Yue, X. Sun, J. Yang, and F. Wu, "Image Denoising by Exploring External and Internal Correlations", In *IEEE Trans. Image Process.*, to appear.

[31] H. Yue, X. Sun, J. Yang, and F. Wu, "Image Denoising Using Cloud Images", in *Proc. SPIE 8856*, Applications of Digital Image Processing XXXVI, 2013.

[32] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity", in *IEEE Trans. Image Process.*, vol. 13, no. 4, 2004.

[33] E. J. Candes, M. B.Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted l_1 Minimization", in *Journal of Fourier Anal. and Appl.*, 14(5):877-905, special issue on sparsity, December 2008.

[34] A. Levin, R. Fergus, F. Durand, and W.T. Freeman, "Deconvolution using natural image priors", 2007.

Hicham Badri received the Master degree in computer sicence and telecomunications in 2012 from Mohammed V University, and the Ph.D. degree in 2015 from both Université de Bordeaux (INRIA) and Mohammed V University. His reserach interests include image processing, machine learning and sparse methods.

Hussein Yahia is the head of INRIA research team GEOSTAT (Geometry and statistics in acquisition data). Dr. H. Yahia received the Doctorat ode 3eme cycle from University Paris 11 (Orsay) and the HDR (Habilitation Diriger des Recherches) form University Paris 13. He is specialized in non-linear signal processing and the analysis of complex signals and systems using advanced nonlinear physics. Dr. H. Yahia has also made substantial contributions in Computer Graphics and Image Processing. Dr. H. Yahia is developing strong collaboration with the LEGOS Laboratory in Toulouse (UMR CNRS 55 66) as well ICM-CSIC in Barcelona, and also IIT Roorkee in India (Associated team OPTIC with Prof. Dharmendra Singh). He is also involved in many contract with the French spatial agency CNES and the European Spatial Agency (ESA). Dr. H. Yahia is the author or co-author of about 80 publications in international peer-reviewed journal and conferences including top-ranked conferences such as ACM SIGGRAPH, CVPR and ECCV. He is a member of the editorial board of journals in signal processing or complex systems and has been supervising more than 10 PhD students.

Driss Aboutajdine received the Doctorat de 3me Cycle and the Doctorat d'Etat-es-Sciences degrees in signal processing from the Mohammed V-Agdal University, Rabat, Morocco, in 1980 and 1985, respectively. He joined Mohammed V-Agdal University, Rabat, Morocco, in 1978, first as an assistant professor, then as an associate professor in 1985, and full Professor since 1990. Over 35 years, he developed teaching and research activities covering various topics of signal and image processing, wireless communication and pattern recognition which allow him to advise more than 50 Ph.D. theses and publish over 200 journal papers and conference communications. He succeeded to found and manage since 1993 and 2001 respectively the LRIT Laboratory and the Centre of Excellence of Information & Communication Technology (pole de competences STIC) which gathers more than 30 research laboratories from all Moroccan universities and including industrial partners as well. Prof. Aboutajdine has organized and chaired several international conferences and workshops. He was elected member of the Moroccan Hassan II Academy of Science and technology on 2006 and fellow of the TWAS academy of sciences on 2007. As an IEEE Senior Member, he co-founded the IEEE Morocco Section in 2005 and he is chairing the Signal Processing chapter he founded in December 2010. He is currently the director of the CNRST research center in Morocco, Rabat.