Category-Specific Object Image Denoising

Saeed Anwar, *Student Member, IEEE,* Fatih Porikli, *Fellow, IEEE and Cong Phuoc Huynh, Member, IEEE*

***Abstract*—We present a novel image denoising algorithm that uses external, category specific image database. In contrast to existing noisy image restoration algorithms that search patches either from a generic database or noisy image itself, our method first selects clean images similar to the noisy image from a database that consists of images of the same class. Then, within the spatial locality of each noisy patch, it assembles a set of "support patches" from the selected images. These noisy-free support samples resemble the noisy patch and correspond principally to the identical part of the depicted object. In addition, we employ a content adaptive distribution model for each patch where we derive the parameters of the distribution from the support patches. We formulate noise removal task as an optimization problem in the transform domain. Our objective function composed of a Gaussian fidelity term that imposes category specific information, and a low-rank term that encourages the similarity between the noisy and the support patches in a robust manner. The denoising process is driven by an iterative selection of support patches and optimization of the objective function. Our extensive experiments on five different object categories confirm the benefit of incorporating category-specific information to noise removal and demonstrates the superior performance of our method over the state-of-the-art alternatives.**

***Index Terms*—denoising, external datasets for denoising, category-specific denoising.**

I. Introduction

Our objective in this work is to remove noise from images (or image regions) that depict a single object of a known class. This goal perfectly complements the recent advancements in object detection and classification [1], [2]. Denoising of images with known classes is instrumental in various applications such as face image enhancement thus all image solutions tasks where face images are used, document image recovery, digital heritage, cell image analysis, and image aesthetics to count a few.

State-of-the-art techniques in image denoising [3], [4], [5], [6], [7], [8], [9] exploit repetitive local patterns that frequently occur in natural images, by selecting and grouping similar patches for collaborative denoising. Making use of this property, non-local means [3], [10] denoise images by computing a weighted average of non-local similar patches, with the weights set as the Euclidean distances between their pixel values. Dong *et al.* [7] proposed an $\ell_{p,q}$ norm constraint to promote patch similarity and derived a denoising solution via spatially adaptive iterative singular-value thresholding (SAIST).

S. Anwar and F. Porikli are with the Research School of Engineering, Australian National University, and CSIRO Data61. This research was done while C. P. Huynh was at National ICT Australia (NICTA), now CSIRO Data61. This research was supported under Australian Research Council's Discovery Projects funding scheme (project number DP150104645) and an Australian Government RTP Scholarship.
E-mail: (saeed.anwar, fatih.porikli, cong.huynh)@anu.edu.au

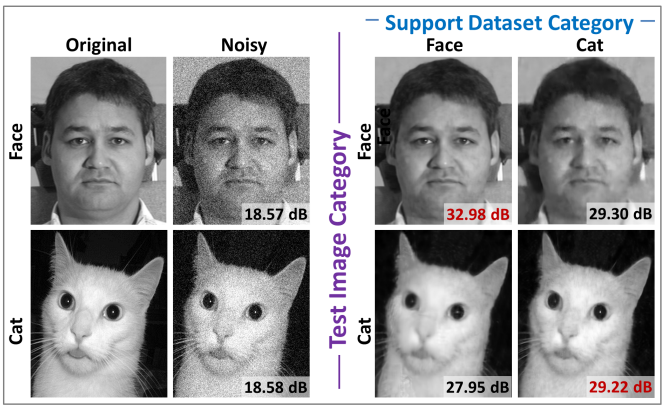


Fig. 1. Denoising results of two sample images from *face* and *cat* categories. As visible, by using the same category support dataset we generate higher PSNR scores - shown in red (best viewed in high-resolution).

Collaborative filtering with block matching (BM3D) and its variants [4], [8], [5] are prominent baseline methods that express patch similarity in a 2D transform domain. The core idea lies in the imposition of structural similarity among patches in each group by analyzing the subspace of the transform coefficients. The denoising of individual patches relies on the implicit assumption that insignificant coefficients correspond to the noise component and thus can be truncated via thresholding or attenuated via Wiener filtering. Similar to BM3D, Non-Local Bayes (NLB) [6] consists of a two-step process that estimates the values of a latent patch from the mean and covariance matrix of its similar patches.

The use of external image datasets for denoising has been around in recent years. This trend is motivated by several studies [11], [12] that show that theoretically minimal error can be achieved by using very large datasets. Furthermore, this approach can be made practical by applying efficient sampling techniques on large databases [13]. Many early works [14], [15], [16] learn an overcomplete dictionary of image patches from an external noise-free database and impose non-local self-similarity through a sparse representation. In a related work, Zha *et al.* [17] enforced a group sparsity residual constraint, to minimise the discrepancy between the sparse code of the noisy image and that of a pre-filtered image. As an alternative, many approaches [18], [19], [20] aim to learn a statistical prior of natural image patches, such as the Gaussian Mixture Model (GMM) of natural image patches or patch groups for patch reconstruction in a maximum likelihood framework.

Recently, several authors adapted Zoran *et al.* [18]'s patch prior to represent image-specific and class-specific semantics. Based on a Gaussian Mixture model (GMM), this generic

prior captures statistics of natural patches by performing the Expectation-Maximisation (EM) algorithm on a large dataset of clean patches. Luo *et al.* [21]'s work is aimed to adapt the generic patch prior to one that is is specific to the patch statistics of the input image. The core of the method is a modified version of the EM algorithm on the noisy image, or its pre-filtered version with an estimate of the noise. Teodoro *et al.* [22] proposed an approach to locally adapt the GMM prior [18] to the class of each individual patch. This approach enables patch-based image enhancement for multiple classes appearing in the same image.

Nevertheless, prior work research on external image denoising [14], [18], [23], [19], [24] only tackled the problem for generic natural images. None of them has considered how to denoise object images of a specific class by incorporating class-specific information from object image datasets. There has been effort in utilising class-specific priors for image deblurring [25], [26], but these approaches are not directly applicable to denoising. Yue *et al.* [24] proposed an ad-hoc denoising method where they imposed a restrictive assumption on the external images, which requires them to contain a significant similarity or overlap with the input image. Since they employed SIFT [27] as a keypoint localisation step for image registration, the method only works well if the external images are related to the original noisy image by a rigid transformation.

In this paper, we propose a denoising method for images of single objects using a noise-free external image dataset of the same category. Unlike the existing methods that ignore the relative locations of external patches with respect to the whole object window, our method considers the object part semantics during patch selection by limiting the search to a part-based locality in the most relevant external images, aiming to make the best use of the part-whole relationships.

Our formulation is unique and differs from previous approaches such as [23]. Our decision to express the denoising problem in the transform domain is strongly motivated by the need to establish a patch similarity metric that is invariant to the local pixel intensity. In natural images, it is rare that two patches have identical intensity values, but it is common that patches share similar local features such as uniform areas, smooth gradients, edges, corners, and textures. Such local patch features are closely related to the gradient responses and hence can be better represented by frequency coefficients in the transform domain.

We achieve robustness to object pose and scale variations by operating on the patch level, a similar analogy to the part-based models of object classification, and by creating copies of the dataset at various object scales and determining the correct scale for the input image, which can be provided by an object detector.

Figure 1 illustrates the imperative role of category-specific datasets in denoising. As visible, a significant improvement in image sharpness and PSNR is achieved when using the correct class dataset for denoising images of known classes. The novel contributions of our approach are as follows.

- A strategy for finding similar external patches to a given noisy patch within the same object part, which we term "support patches" hereafter.
- A formulation of the object category-specific patch denoising problem in a transform domain.
- A Gaussian model of the membership likelihood to a support patch group for a noisy patch.
- A low-rank constraint to enforce the similarity between the noisy patch and its support patches.

II. Denoising Problem Formulation

The noisy image model relates the true pixel value x to the noisy value y at the same pixel by

$$y = x + \eta, \tag{1}$$

where $\eta \sim \mathcal{N}(0, \sigma_n^2)$ is assumed to be Gaussian noise with a standard deviation σ_n.

We consider problem of recovering the latent (true) image, given the noisy image of an object and a dataset of noise-free images in the same object category. Let the matrices $\mathbf{X}$, $\mathbf{Y}$ represent the pixel values of the true and observed images, and the set of matrices $\{\mathbf{Z}_k : k = 1, \ldots, K\}$ denote the external dataset.

A. Support patch search

As mentioned earlier, image denoising by collaborative filtering relies on the similarity between patches and performs aggregation of their denoised version. Following this approach, we collect all the overlapping patches of the noisy image, and denote the intensity vector of the patch centered at the i-th pixel by $\mathbf{y}_i$ $i = 1, \ldots, M$,. Likewise, $\mathbf{x}_i$ denotes the patch intensity vector for the corresponding location in the latent image $\mathbf{X}$.

For a given noisy patch, we find the most similar "support patches" from the dataset of category-specific images. Due to the similarity in their local features, support patches facilitate noise suppression by enforcing a group sparsity constraint. Various approaches such as BM3D [4] and non-local means (NLM) denoising [3], [10], [6] have employed local and non-local patch similarity to separate the latent structure of a patch from its noise component.

In our algorithm, the support patch selection occurs in several stages. Firstly, we select a preset number (L) of external images that are structurally most similar to the noisy image based on the structural similarity (SSIM) index. Subsequently, from the l-th candidate image ($l = 1, \ldots L$), we obtain a pool $P_{i,l}$ of patches that are similar to a given noisy patch $\mathbf{y}_i$. We take into account the difference in resolution and aspect ratio between the input and the candidate image when determining the local search window. Suppose that H and W are the height and width of the input image, and H_l and W_l are the corresponding quantities of the candidate image. The center coordinates $[r_{i,l}, c_{i,l}]^T$ of the local search window in the l-th candidate image is a linear mapping of the location $[r_i, c_i]^T$ of the i-th pixel as follows, $r_{i,l} = \lfloor r_i \frac{H_l}{H} \rfloor$ and $c_{i,l} = \lfloor c_i \frac{W_l}{W} \rfloor$. The search window has a preset size of 51×51.

Finally, within each patch pool $P_{i,l}$ we only retain those that have a Euclidean distance from the input patch $\mathbf{y}_i$ that is below a threshold τ. We denote the resulting set of refined patches by

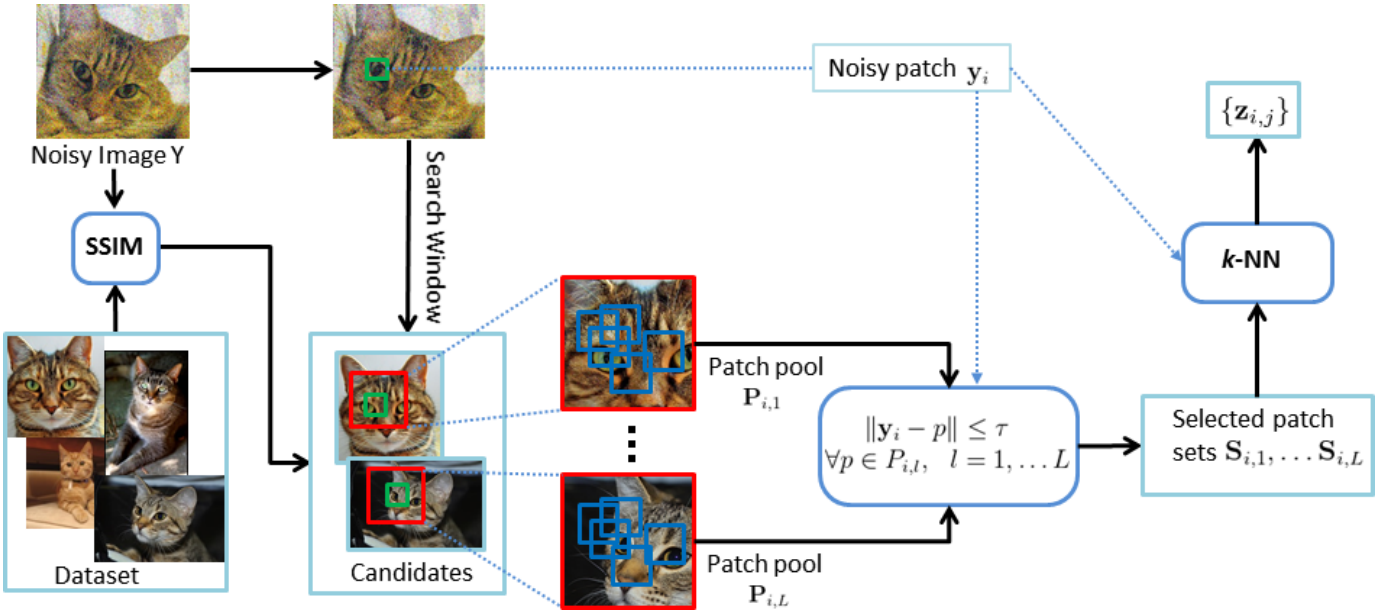


Fig. 2. Searching and selecting support patches for a given noisy patch $\mathbf{y}_i$. Candidate images similar to the noisy image (measured by SSIM) are selected from the given database. Subsequently, in each candidate image, we search for patches that are similar to the noisy patch, *i.e.* within a Euclidean distance of τ from $\mathbf{y}_i$. The search is restricted to a local window in each candidate image. Finally, among the remaining patches, only the nearest neighbors to $\mathbf{y}_i$ are retained for denoising.

$S_{i,l}$. Next, we aggregate the refined patch pools $S_{i,l}$ across the candidate images. Within the resulting collection, we perform a k-NN search for the most similar patches to $\mathbf{y}_i$. In the end, we obtain a set of support patches $\{\mathbf{z}_{i,j} : j = 1, \ldots, T_i\}$ resembling the noisy patch $\mathbf{y}_i$.

Figure 2 shows the procedure for searching and selecting support patches for a given noisy patch. In the figure, the noisy patch y_i is bounded by a green rectangle (top row) while the patches with blue boundaries illustrate members of the patch pools $P_{i,l}$. In both the noisy and candidate images, the search space is indicated by a red rectangular boundary.

B. Transform domain formulation

In our formulation, we opt to represent local patches in a transform domain, rather than the patch intensity domain. This is because matching patches in the original space of patch intensity vectors is susceptible to a bias in the overall patch intensity, such as local illumination. Representing patches in the transform domain encourages matching between those that have a various range of intensity but similar local structure.

To improve robustness to patch intensity, we subtract the mean patch intensity from the patch intensity vector before performing the domain transform. The per-patch mean subtraction effectively removes the zero-frequency (DC) bias, yielding patches lying in a $N-1$-dimensional subspace, where N is the number of patch pixel. Therefore, the latent patch can be represented by the remaining $D = N - 1$ transform coefficients. The Gaussian prior and the low-rank constraint in the space of non-zero-frequency coefficients effectively enforce patch similarity, and are less susceptible to variations in patch intensity.

With this intention, we introduce the notation for representing images and patches in the discrete transform domain. Here, we choose to use the DCT transform, although other popular transforms such as the wavelet, Fourier, DST and the Walsh-Hadamard transforms can be employed for the same purpose.

We denote the patch transform by $\mathcal{T} : \mathbb{R}^N \to \mathbb{R}^D$ that maps the original N-dimensional intensity vector to a vector of $D = N - 1$ non-zero frequency DCT coefficients, with N being the number of pixels per patch. Let $\Phi \in \mathbb{R}^{N \times D}$ denote the DCT basis spanning this D-dimensional subspace. Note that $\Phi^T\Phi = \mathbb{I}$. Let us also denote the mean subtracted versions of the patches $\mathbf{x}_i$ and $\mathbf{y}_i$ by $\bar{\mathbf{x}}_i$ and $\bar{\mathbf{y}}_i$, respectively. The intensity vector of these patches are related to the transform coefficient vectors α_i and $\beta_i \in \mathbb{R}^D$ by $\alpha_i = \Phi^T\bar{\mathbf{x}}_i$, $\beta_i = \Phi^T\bar{\mathbf{y}}_i$, and $\bar{\mathbf{x}}_i = \Phi\alpha_i$, $\bar{\mathbf{y}}_i = \Phi\beta_i$. Once the transformation coefficients α_i is estimated, we can compute the mean-subtracted latent patch by $\bar{\mathbf{x}}_i$ by an inverse DCT transform and recover the original patch $\mathbf{x}_i$ by adding its intensity mean to $\bar{\mathbf{x}}_i$.

Next, we will formulate the various components of our denoising problem.

C. Data fidelity

Assuming the independence of individual pixel values, the conditional likelihood of the noisy image given the original (noise-free) image is

$$p(\mathbf{Y}|\mathbf{X}) \propto \exp\left(-\frac{\|\mathbf{Y} - \mathbf{X}\|_2^2}{\sigma_n^2}\right), \tag{2}$$

where $\|\cdot\|_2$ stands for the ℓ_2 norm of a vector.

The reconstruction of the noise-free image $\mathbf{X}$ aims to maximise the conditional log likelihood in Equation 2, which is equivalent to minimising the data fidelity term $\|\mathbf{Y} - \mathbf{X}\|_2^2$,

which is a sum of squared errors over image pixels. Since each image pixel belongs an approximately equal number of overlapping patches, *i.e.* N, the term above can be approximated as a multiple of the sum of these terms evaluated on a per-patch basis, *i.e.* $\|\mathbf{Y}-\mathbf{X}\|_2^2 \approx \frac{1}{N}\sum_{i=1}^{M}\|\mathbf{y}_i-\mathbf{x}_i\|_2^2$. Furthermore, $\mathbf{y}_i-\mathbf{x}_i = \bar{\mathbf{y}}_i-\bar{\mathbf{x}}_i$ assuming that the mean intensity of the latent patch $\mathbf{x}_i$ is estimated from that of $\mathbf{y}_i$, and $\|\bar{\mathbf{y}}_i-\bar{\mathbf{x}}_i\|_2^2 = \|\Phi(\alpha_i-\beta_i)\|_2^2 = \|\alpha_i-\beta_i\|_2^2$ due to the orthonormality of the basis Φ. Expressing the data fidelity in terms of the transform coefficients, we obtain

$$\|\mathbf{Y}-\mathbf{X}\|_2^2 \approx \frac{1}{N}\sum_{i=1}^{M}\|\alpha_i-\beta_i\|_2^2. \tag{3}$$

D. Support patch group membership

Now we define an additional constraint that imposes the similarity between a noisy patch and those from an image dataset belonging to the same object category. Recall that the patch search described in Section II-A results in T_i support patches $\{\mathbf{z}_{i,j} : j = 1,\dots,T_i\}$ that resemble the local appearances of the noisy patch $\mathbf{y}_i$. Here, we rely on the statistics of the support patch group $\mathbf{z}_{i,j}$ in the transform domain in order to predict the latent patch $\mathbf{x}_i$ from $\mathbf{y}_i$. Let the transform coefficients of $\mathbf{z}_{i,j}$ be $\{\gamma_{i,j} : j = 1,\dots,T_i\}$, where T_i is the number of support (most similar) patches of patch $\mathbf{x}_i$, and μ_i and Σ_i be the mean and covariance matrix estimated from these transform coefficient vectors.

Assuming that similar patches belong to a Gaussian distribution in the transform domain, the most probable $\mathbf{x}_i$ is one that maximises its likelihood of belonging to the support patch group, *i.e.* $p(\alpha_i|\mu_i,\Sigma_i) \propto \exp\left(-\frac{1}{2}(\alpha_i-\mu_i)^T\Sigma_i^{-1}(\alpha_i-\mu_i)\right)$. This is equivalent to minimising the log-likelihood

$$\log p(\alpha_i|\mu_i,\Sigma_i) \propto \frac{1}{2}(\alpha_i-\mu_i)^T\Sigma_i^{-1}(\alpha_i-\mu_i), \tag{4}$$

which is the Mahalanobis distance from a noisy patch to the distribution of its support patches in the transform domain.

E. Low-rank constraint

We further formulate a low-rank constraint concerning a noisy patch and its support patches. The intuition behind this constraint is that the local structure of a patch can be sparsely represented by a basis with a low cardinality. Therefore, when similar patch vectors are stacked as columns of a matrix, the matrix should exhibit the low rank property and have sparse singular values. In [7], the authors derived this low-rank property directly from the common observation that the structural similarity between patches can be encoded as a group sparsity constraint, in terms of the $\ell_{p,q}$ norm of the above matrix.

However, the rank minimisation problem is NP-hard, and thus is intractable to solve directly. In their work, Candès and Recht [28] have provably derived the tightest convex relaxation of the rank minimisation problem in the form of a matrix nuclear norm minimisation problem. Under certain conditions, these two problems have exactly the same unique solution. Therefore, the low-rank approximating matrix can be recovered exactly by solving the nuclear norm minimisation (NNM) problem.

To formulate the NNM problem, for each latent patch $\mathbf{x}_i$, we form a data matrix $\mathbf{M}_i$ containing its transform coefficients and those of its support patches as its columns, as $\mathbf{M}_i = [\alpha_i, \gamma_{i,1}, \dots, \gamma_{i,T_i}]$. Here, we aim to minimise the matrix nuclear norm $\|\mathbf{M}_i\|_*$, which is the sum of its singular values.

III. Optimisation

In previous sections, we have described the data fidelity term in Equation 3, the patch group membership term in Equation 4 and the nuclear norm constraint on $\mathbf{M}_i$ for each noisy patch $\mathbf{y}_i$. Aggregating all these terms over all the image patches $\mathbf{y}_i, i = 1,\dots,M$, we formulate the overall minisation problem as

$$\mathcal{L} = \sum_{i=1}^{M}\mathcal{L}_i, \tag{5}$$

where the term $\mathcal{L}_i$ is related to only the i-th noisy patch as

$$\begin{aligned}\mathcal{L}_i = {} & \frac{1}{\sigma_n^2}\|\alpha_i-\beta_i\|_2^2 + \lambda_1(\alpha_i-\mu_i)^T\Sigma_i^{-1}(\alpha_i-\mu_i) \\ & + \lambda_2\|[\alpha_i,\gamma_{i,1},\dots,\gamma_{i,T_i}]\|_*,\end{aligned} \tag{6}$$

where $\{\gamma_{i,j} : j = 1,\dots,T_i\}$ are the transform coefficients of the support patches for the patch $\mathbf{x}_i$, $\|\cdot\|_*$ is the nuclear norm of a matrix and λ_1 and λ_2 are the weights of the support patch group likelihood and the nuclear norm terms.

A. Patch denoising

We can minimise the overall objective function in Equation 5 by minimising each of the term $\mathcal{L}_i$ independently. To this end, we introduce an auxiliary variable $\mathbf{M}_i \triangleq [\alpha_i, \gamma_{i,1}, \dots, \gamma_{i,T_i}]$ to Equation 6. Subsequently, we relax the equality constraint as minimising the squared Frobenius norm $\|\mathbf{M}_i - [\alpha_i, \gamma_{i,1}, \dots, \gamma_{i,T_i}]\|_F^2$ and incorporate it into the objective function.

In addition, we normalise the term by a Lagrange multiplier equal to $\frac{1}{(T_i+1)\sigma_n^2}$, which accounts for the image noise and the number of support patches. For a patch $\mathbf{x}_i$, we then minimise $\mathcal{L}_i$ with respect to the transform α_i and the variable $\mathbf{M}_i$

$$\begin{aligned}(\alpha_i^*,\mathbf{M}_i^*) = \underset{\alpha_i,\mathbf{M}_i}{\operatorname{argmin}} & \frac{1}{\sigma_n^2}\|\alpha_i-\beta_i\|_2^2 \\ & + \lambda_1(\alpha_i-\mu_i)^T\Sigma_i^{-1}(\alpha_i-\mu_i) \\ & + \frac{\|\mathbf{M}_i - [\alpha_i,\gamma_{i,1},\dots,\gamma_{i,T_i}]\|_F^2}{(T_i+1)\sigma_n^2} + \lambda_2\|\mathbf{M}_i\|_*.\end{aligned} \tag{7}$$

The relaxed objective function in Equation 7 is convex with respect to α_i and $\mathbf{M}_i$ separately, while the other variable is fixed. More specifically, when $\mathbf{M}_i$ is fixed, the non-constant terms, including the squared Frobenius norm, are quadratic functions of α_i. On the other hand, when α_i is fixed, the objective function involves a nuclear norm of $\mathbf{M}_i$ and a squared Frobenius norm. It is known that the nuclear norm is convex in the space of the matrix $\mathbf{M}_i$, and the squared Frobenius norm is regarded as a quadratic function of the matrix elements.

We employ an iterative procedure to minimise the cost function in Equation 7. Each iteration involves an alternating optimization scheme with respect to either α_i or $\mathbf{M}_i$, while fixing the other. Since each of these steps aims to solve a convex sub-problem with respect to its own variable, this scheme is guaranteed to converge to a global minimum in each step, with respect to either α_i or $\mathbf{M}_i$.

1) Update of α_i with fixed $\mathbf{M}_i$: With a fixed value of $\mathbf{M}_i^*$ at the current iteration, we solve the sub-problem

$$\alpha_i^* = \underset{\alpha_i}{\operatorname{argmin}} \frac{\|\alpha_i - \beta_i\|_2^2}{\sigma_n^2} + \lambda_1(\alpha_i - \mu_i)^T\Sigma_i^{-1}(\alpha_i - \mu_i) + \frac{\|\alpha_i - \mathbf{M}_i^*(:,1)\|_2^2}{(T_i+1)\sigma_n^2}, \tag{8}$$

where $\mathbf{M}_i^*(:,1)$ denotes the first column of the matrix $\mathbf{M}_i^*$. Since the problem above is quadratic in α_i, taking its derivative leads to the following linear equation, which can be solved by standard techniques.

$$\left(\frac{T_i+2}{T_i+1}\mathbb{I} + \lambda_1\sigma_n^2\Sigma_i^{-1}\right)\alpha_i^* = \beta_i + \lambda_1\sigma_n^2\Sigma_i^{-1}\mu_i + \frac{\mathbf{M}_i^*(:,1)}{T_i+1}. \tag{9}$$

2) Update of $\mathbf{M}_i$ with fixed α_i: With the values of α_i^* obtained from the previous step, we form a data matrix $\hat{\mathbf{M}}_i \triangleq [\alpha_i^*, \gamma_{i,1}, \ldots, \gamma_{i,T_i}]$ for each patch. The sub-problem to be solved with respect to $\mathbf{M}_i$ is then stated as

$$\mathbf{M}_i^* = \underset{\mathbf{M}_i}{\operatorname{argmin}} \|\mathbf{M}_i - \hat{\mathbf{M}}_i\|_F^2 + \tau\|\mathbf{M}_i\|_*, \tag{10}$$

where $\tau = \lambda_2(T_i+1)\sigma_n^2$.

The above problem is related to finding an approximation to a given matrix with a minimal nuclear norm. To solve the problem, we turn our attention to the singular value shrinkage operator developed by Cai *et al.* [29]. Suppose that we have $U\Lambda V^T$ as the singular value decomposition of $\hat{\mathbf{M}}_i$, with Λ_k being the k-th singular value. Theorem 2.1. in [29] derives the optimal solution to Equation 10 by soft-thresholding the singular values to obtain

$$\mathbf{M}_i^* = U\mathcal{S}_\tau(\Lambda)V^T, \tag{11}$$

where the soft-thresholding operator is defined as $\mathcal{S}_\tau(\Lambda) = \operatorname{diag}(\{(\Lambda_k - \tau)_+\})$ with $(x)_+ = \max(x, 0)$.

B. Recovering latent image

Once we have estimated the transform coefficients of individual patches, we recover them in the pixel domain by an inverse transform as $\mathbf{x}_i = \Phi^T\alpha_i, \forall i = 1, \ldots, M$ (assuming that Φ is orthonormal). To reconstruct the full image, we translate the patches to their original locations and average the values of overlapping patches at shared pixels. Let R_i denote the patch extraction matrix at the i-th pixel of an image, *i.e.* $\mathbf{x}_i = R_i\mathbf{X}$. With the known matrices R_i's, the latent image is the optimal solution to the problem

$$\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmin}} \lambda_0\|\mathbf{X} - \mathbf{Y}\|_2^2 + \sum_{i=1}^{M}\|R_i\mathbf{X} - \mathbf{x}_i\|_2^2. \tag{12}$$

Algorithm 1 Denoising with category-specific support patches

Input:
$\mathbf{Y}$: noisy input image.
σ_n: noise standard deviation.
$\lambda_0, \lambda_1, \lambda_2$: term weights in Equations 6 and 12.
ρ: relaxation factor in Equations 14.

1: $t \leftarrow 0$, $\mathbf{X}^{(0)} \leftarrow \mathbf{Y}$, $\mathbf{Y}^{(0)} \leftarrow \mathbf{Y}$, $\varsigma^{(0)} \leftarrow \sigma_n^2$.
2: **repeat**
3: **for** patch $\mathbf{y}_i^{(t)}$ in $\mathbf{Y}^{(t)}$ **do**
4: Update $\beta_i^{(t)} \leftarrow \Phi\mathbf{y}_i^{(t)}$.
5: $\{\mathbf{z}_{i,j}^{(t)} : j = 1, \ldots, T_i\} \leftarrow$ support patches of $\mathbf{x}_i^{(t)}$.
6: **for** $j = 1 \rightarrow T_i$ **do**
7: Support patch transform $\gamma_{i,j}^{(t)} \leftarrow \Phi\mathbf{z}_{i,j}^{(t)}$.
8: **end for**
9: $(\mu_i^{(t)}, \Sigma_i^{(t)}) \leftarrow$ mean and covariance matrix of $\{\gamma_{i,j}^{(t)} : 1 \le j \le T_i\}$.
10: **repeat**
11: Solve Equation 9 for $\alpha_i^{(t)}$.
12: Update matrix $\mathbf{M}_i^{(t)}$ by Equation 11.
13: **until** Convergence
14: Update $\mathbf{x}_i^{(t)} \leftarrow \Phi^T\alpha_i^{(t)}$.
15: **end for**
16: Reconstruct the image $\mathbf{X}^{(t)}$ by Equation 13.
17: Regularise the input image $\mathbf{Y}^{(t+1)}$ by Equation 14.
18: $t \leftarrow t + 1$.
19: Update noise variance $\varsigma^{(t+1)} \leftarrow \rho|\varsigma^{(t)} - \frac{1}{M}\|\mathbf{Y} - \mathbf{Y}^{(t+1)}\|^2|$
20: **until** $\|\mathbf{X}^{(t)} - \mathbf{X}^{(t-1)}\|_2 \le \epsilon$
21: **return** Latent (denoised) image $\mathbf{X}^{(t)}$.

where λ_0 is a positive constant. The least-squares solution to the above equation is

$$\mathbf{X}^* = \left(\lambda_0\mathbb{I} + \sum_{i=1}^{M} R_i^T R_i\right)^{-1}\left(\lambda_0\mathbf{Y} + \sum_{i=1}^{M} R_i^T\mathbf{x}_i\right) \tag{13}$$

The process of patch denoising and latent image recovery occurs iteratively until convergence. In addition, we apply the iterative input regularisation technique in [30]. Such an approach has been shown to be effective in denoising methods using total variation and wavelets [31] and spatially adaptive iterative singular value thresholding [7]. Specifically, in the t-th iteration, the algorithm takes input from a regularised noisy image $\mathbf{Y}^{(t)}$ computed as follows

$$\mathbf{Y}^{(t+1)} = \mathbf{X}^{(t)} + \rho\left(\mathbf{Y} - \mathbf{X}^{(t)}\right), \tag{14}$$

where $\mathbf{X}^{(t)}$ is the current latent image and ρ is a relaxation parameter.

C. Algorithm implementation

The overall denoising algorithm consists of interleaving steps of individual patch denoising and whole image restoration. The proposed iterative procedure is summarised in Algorithm 1, with the iteration number denoted by t. As the latent image $\mathbf{X}$ is updated in every iteration, so are the support patches (from the external image dataset) of its patches. Line 5

implements the support patch search procedure described in Section II-A. With the support patches in hand, individual patches in the input image are denoised by alternating the optimisation with respect to the variables α_i and $\mathbf{M}_i$ (in lines 10– 13). At the end of each iteration, the entire latent image $\mathbf{X}^{(t)}$ is reconstructed from the denoised patches and the input image $\mathbf{Y}^{(t+1)}$ to the next iteration is updated according to Equation 14. The noise variance $\varsigma^{(t+1)} \triangleq (\sigma^{(t+1)})^2$ is also updated according to the adjusted input as $\varsigma^{(t+1)} \leftarrow \lambda_n|\varsigma^{(t)} - \frac{1}{M}\|\mathbf{Y}-\mathbf{Y}^{(t+1)}\|^2|$, where M is the number of image pixels and $\lambda_n = 0.17$. The algorithm terminates when the change $\|\mathbf{X}^{(t)} - \mathbf{X}^{(t-1)}\|_2$ falls below a tolerance threshold ϵ.

To improve patch similarity, we follow Foi *et al.* [8] and perform a DCT transform on the mean-subtracted intensity of local patches and subsequently add the mean patch intensities back during patch reconstruction. This effectively means that we only involve the AC components $\gamma_{i,j}$ of the support patches for patch-wise denoising (Section III-A). This technique is based on the observation that subtracting the direct current (DC) component of each patch from its intensity values effectively increases the number of similar local patterns in each group, facilitating a more thorough selection of the most similar support patches to a noisy patch for collaborative filtering. Further, the per-patch mean subtraction improves the chance of finding a good match, which means a lower number of external images is required for patch-wise denoising.

IV. Experiments

In this section, we present a detailed performance evaluation of our method against a number of state-of-the-art internal and external image denoising algorithms. Firstly, we examine the influence of the number of category-specific images and support patches on the denoising accuracy. Subsequently, we report quantitative and qualitative results for all the methods under study.

A. Datasets and parameter settings

We performed experimental validation on the following datasets, including CMU PIE face dataset [32], Car dataset [33], Cat dataset [34], Gore face dataset [35] and the Multiview dataset [36][1]. For each dataset, we randomly selected half of the images to form a category-specific dataset and between 10 and 15 images from the remaining half as ground-truth images for denoising. It is to be noted here that we have disjoint image sets for the test and training *i.e.* neither the same people appear in the text and training images, nor the same objects with different scale and pose. To generate noisy images, we corrupt the test images by additive white Gaussian noise with standard deviations (std) of $\sigma_n = 30, 50, 70, 100$, similar to the practice employed elsewhere [24], [19], [23], [20], [37]. We also intend to demonstrate the effectiveness of our algorithm at the high noise std of 50 and beyond.

For evaluation purposes, we use Peak Signal-to-Noise Ratio (PSNR) index as the error metric. We compare our proposed method with numerous state-of-the-art methods, including BM3D [4], WNNM [38], NLM [3], SAPCA [5], TSID [9], EPLL [18], PCLR [19], PGPD [20] and TID [23]. To ensure a fair comparison, we modify the state-of-the-art internal denoising methods of NLM, BM3D, SAPCA and TSID to perform search on class-specific image datasets. We use the same settings as their original implementations.

Our method shares a number of common parameters with algorithms that exploit patch similarity, and inherits the parameter values from the prior works. Similar to BM3D [4] and WNNM [38], we choose a patch size of 8. When searching for support patches, we select $L = 16$ candidate images that are most similar to the noisy image, as described in the denoising method using targeted databases, *i.e.* TID [23]. In addition, we employ a search window with a size of 51×51 in each candidate image. In the last stage of support patch search, the number of nearest neighbors k is set to 16, similar to the external denoising methods of eBM3D, eSAPCA and eNLM.

Further, we set the parameters specific to our optimisation problem as follows, $\lambda_0 = 1$, $\lambda_1 = 0.5$, $\lambda_2 = 10$ and $\rho = 0.18$. The values of λ_0, λ_1 and λ_2 are determined by a sensitivity analysis such as that in Section IV-D, using a small number of noisy images as the validation set. Our algorithm inherits the value of ρ from the PCLR, WNNM and PGPD methods.

B. Influence of external dataset size

Here, we examine the influence of the size of the external dataset on the denoising performance, while fixing all the other parameters. To this end, we experiment with dataset sizes of 32, 64, 256 and 1024 by incrementally adding images to the clean image dataset. We choose 15 images among those not in the dataset and simulate noisy input by adding Gaussian noise with a standard deviation σ_n of 30 and 50. The left panel in Figure 3 demonstrates the robustness of our algorithm to the dataset size, showing that an increasing dataset size only slightly improves the denoising accuracy. Even with a small dataset size of 32, our algorithm can achieve an average PSNR of 32.3 dB for $\sigma_n = 30$ and 30 dB for $\sigma_n = 50$.

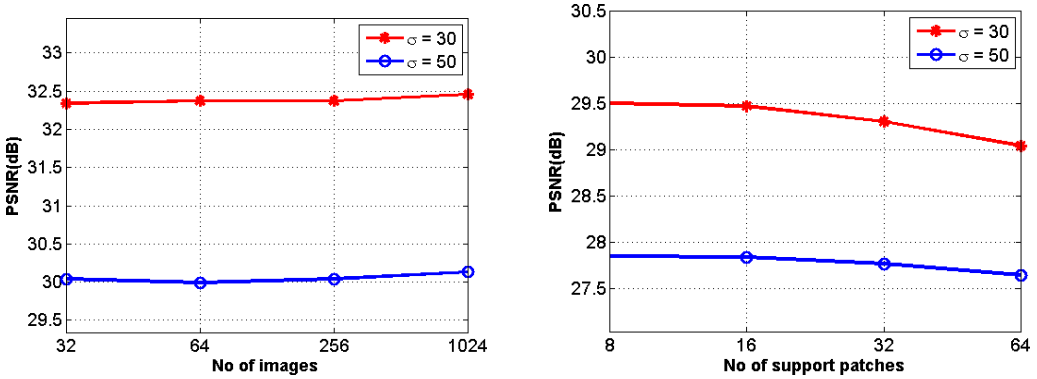
C. Influence of the number of support patches

Similarly, we test the robustness of our algorithm to the number of support patches required for denoising a single patch. For this purpose, we use 8, 16, 32 and 64 support patches per noisy patch. The plots on the right-hand side of Figure 3 shows that the average PSNR declines as the number of support patches increases. The main reason for this phenomenon is that the variation in appearance between the support patches is likely to increase with a larger number of support patches, and their aggregation would result in a loss of local details due to averaging.

D. Relative importance of priors

We assess the relative contribution of the Gaussian prior and the low rank term on the Gore dataset for $\sigma = 50$. The presence of both terms improves the PSNR compared to the scenario where one is absent. For example, when $\lambda_1 \in \{1, 10, 40\}$

[1]In practice, we can utilise images of particular object categories from publicly available datasets such as PASCAL VOC and ImageNet.



Influence of the dataset size | Impact of the number of support patches

Fig. 3. Denoising accuracy (in PSNR) at noise standard deviations $\sigma_n = 30$ and $\sigma_n = 50$. Left: Our method is robust to the changes in the dataset size, which has a low impact on the results. Right: Increasing the number of support patches slightly degrades the denoising results.

TABLE I
RUN-TIME COMPARISONS (IN SECONDS) ON A TEST IMAGE OF SIZE 304×228.

Method	BM3D	eBM3D	eNLM	eSAPCA	eTSID	PCLR	PGPD	TID	EPLL	WNNM	Ours
Time (s)	1.12	173.5	164.6	178.8	178.9	192.4	12.5	172.2	39.9	211.8	119.9

TABLE II
DENOISING PERFORMANCE (IN PSNR) WHEN USING DIFFERENT IMAGE CATEGORY DATASETS. THE PSNR IS MAXIMAL WHEN THE EXTERNAL DATASET CATEGORY MATCHES THE NOISY IMAGE CATEGORY.

	Dataset				
Noisy	Face	Cat	Texture	Text	Car
Face	**26.80**	24.79	22.79	17.03	24.89
Cat	25.01	**28.00**	25.24	19.57	25.87
Texture	24.09	24.67	**28.13**	19.33	24.62
Text	8.43	15.41	14.50	**21.09**	16.33
Car	18.41	19.80	18.85	16.93	**21.90**

and $\lambda_2 = 0$ the resulting PSNR are $\{27.58, 27.56, 27.56\}$, respectively. When $\lambda_1 = 0$ and $\lambda_2 \in \{1, 10, 40\}$ the results are $\{20.14, 26.33, 27.09\}$. When $\lambda_1 = 1$ and $\lambda_2 = 10$, the average PSNR increases to $\mathbf{27.82}$ dB.

E. Run-time comparisons

We have implemented our algorithm in MATLAB on an Intel Core^{TM} i7 machine with 16 GB of memory. In Table I, we show the running times for various methods including ours for an image of size 304×228 and an external dataset containing 10 images of similar sizes. The running time of our method, *i.e.* 119s, is shorter than the MATLAB implementations of various state of the art external image denoising methods *e.g.*, eNLM, eBM3D, eTSID, TID, WNNM and eSAPCA. We observe that our method spends most of its time on patch search. The speed of our algorithm can be improved by applying fast patch search algorithms *e.g.*, KD tree [40] and patch match [41], [42]. In addition, GPU implementations can be employed to parallelise the denoising of patches in independent threads.

F. Role of external image category

Now we illustrate the importance of choosing the correct external image category for denoising. To this end, we provide datasets of different object categories as input to our method for denoising the same noisy images. The categories involved in our experiment are Face (Gore dataset), Cat, Texture (from the Multiview dataset), Text [23] and Car. In Table II, we show the average PSNR of the denoised images for each pair of noisy image category and dataset category. Note that the PSNR values reported are averaged across all the mentioned noise levels ($\sigma_n = 30, 50, 70, 100$) and noisy images. Each row of the table corresponds to a noisy image category while each column represents a dataset category.

The overall trend is that the PSNR for each noisy image category reaches its maximum when the dataset belongs to the same category, as can be observed along the diagonal of Table II. On the other hand, the PSNR diminishes significantly when the dataset belongs to a different category, which confirms the benefit of category-specific information for denoising purposes. This observation also demonstrates the ability of our algorithm to extract useful category-specific information from the support patches.

G. Sensitivity to pose variations

We analyse the response of our method when the noisy test images contain significant pose variations including out-of-plane rotations, semi-profile views and different camera views. We present sample results in Figures 4 and 5.

As can be seen, our method attains qualitatively the most appealing results and quantitatively the best PSNR scores among the all considered methods thanks to its efficient scheme of support patch selection. It restores the different pose faces without visible deterioration of the facial details. Meanwhile, the competitive methods generate clearly visible artifacts induced by the noise distribution.

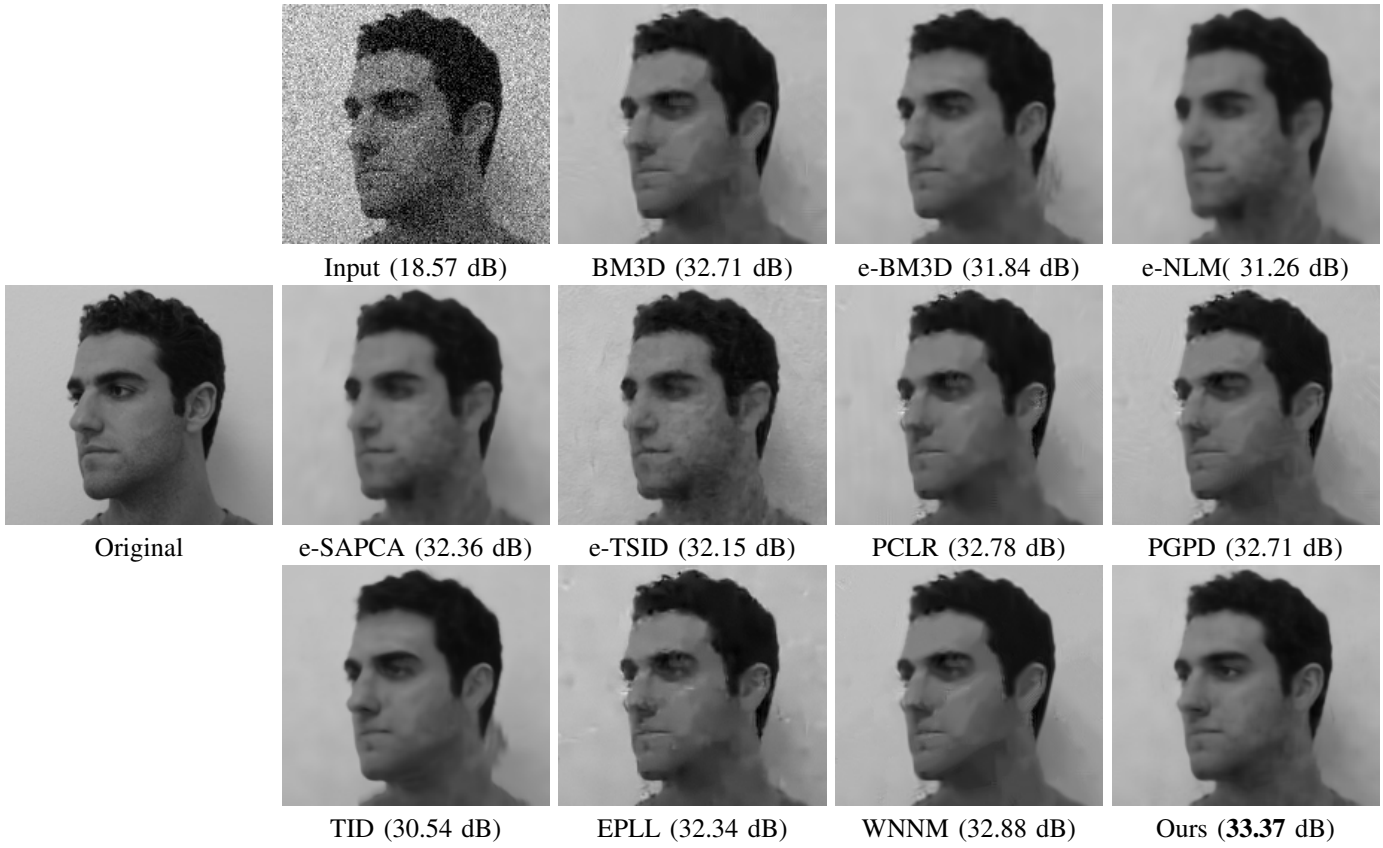


Fig. 4. Denoising results produced by different methods for a face image in a profile view from the FEI face dataset [39] when $\sigma_n = 30$. Our method is able to denoise the input image even with a different pose from those in the noise-free dataset (Differences are better viewed with high resolution display).

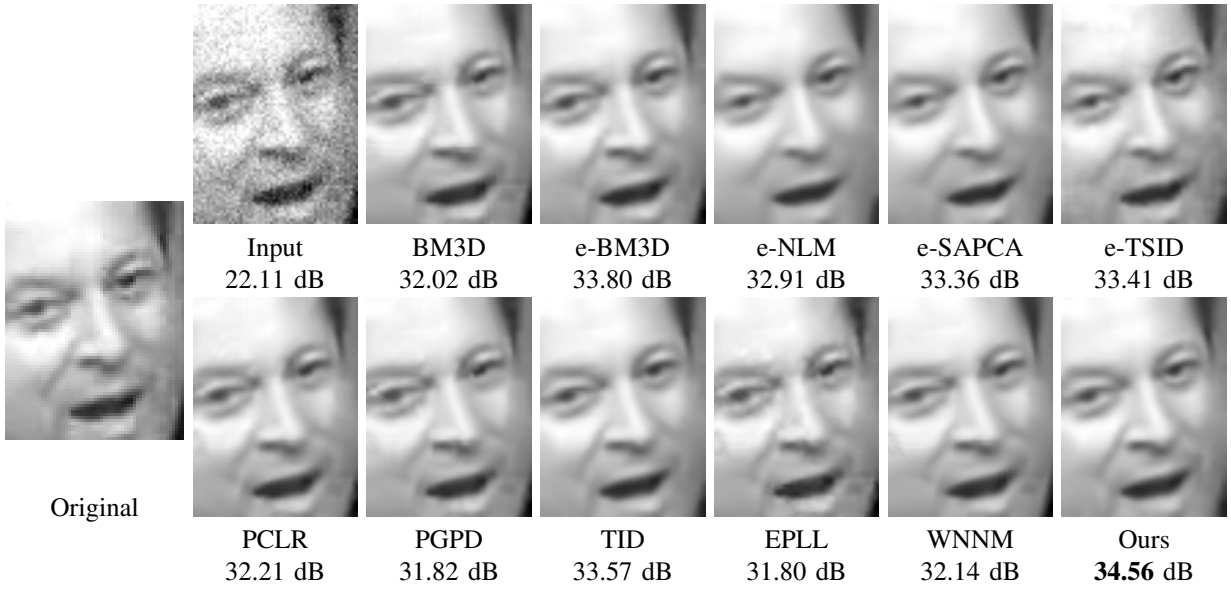


Fig. 5. Denoising results produced by different methods for a face image selected from the Gore dataset [35] when $\sigma_n = 20$. Our method is able to denoise the face image even with a different pose from those in the noise-free dataset.

TABLE III
PERFORMANCE COMPARISON BETWEEN OUR METHOD AND INTERNAL DENOISING TECHNIQUES ON SEVERAL DATASETS, IN TERMS OF PSNR (IN DB).

	BM3D	NLM	SAIST	SAPCA	TSID	DDID	PID	NLB	WNNM	AID	Ours
$\sigma_n = 30$											
Gore	29.28	27.59	29.77	29.00	28.21	29.23	29.21	29.04	29.35	29.32	**29.95**
Cat	30.08	27.08	29.86	30.03	29.45	29.97	29.95	29.86	30.11	29.96	**31.18**
CMU	32.56	30.37	32.48	32.61	31.92	32.65	32.73	32.33	32.63	32.45	**33.38**
View	28.35	27.08	28.20	28.43	28.82	28.19	28.34	28.26	28.46	28.31	**31.96**
$\sigma_n = 50$											
Gore	26.54	23.54	27.11	26.23	25.37	26.48	26.57	26.41	26.37	26.58	**27.82**
Cat	27.96	24.69	27.76	27.91	27.18	27.89	27.91	27.70	27.97	27.80	**28.79**
CMU	30.17	27.92	30.07	30.11	29.29	30.21	30.48	29.84	30.21	29.90	**30.64**
View	26.35	24.69	26.10	26.39	26.53	26.17	26.32	26.15	26.39	26.27	**28.64**
$\sigma_n = 70$											
Gore	24.94	21.30	24.01	25.13	23.43	24.68	24.88	24.64	24.32	24.81	**25.58**
Cat	26.56	23.21	26.24	26.21	25.64	26.45	26.59	26.15	26.52	26.36	**26.80**
CMU	28.45	26.00	28.36	27.90	27.52	28.51	28.89	27.95	28.58	28.16	**28.72**
View	25.14	23.21	24.94	24.92	25.09	24.92	25.07	24.83	25.16	25.00	**27.16**
$\sigma_n = 100$											
Gore	23.21	19.06	22.21	23.31	21.30	22.77	23.05	22.76	22.48	22.95	**23.86**
Cat	25.08	21.95	24.69	24.56	23.97	24.91	25.20	24.46	25.02	24.87	**25.21**
CMU	26.57	24.32	26.56	25.90	25.62	26.63	27.14	26.10	26.74	26.36	26.59
View	23.89	21.95	23.50	23.57	23.64	23.62	23.86	23.49	23.85	23.74	**24.79**

TABLE IV
PERFORMANCE COMPARISON BETWEEN OUR METHOD AND EXTERNAL DENOISING TECHNIQUES ON SEVERAL DATASETS, IN TERMS OF PSNR (IN DB).

	eBM3D	eNLM	eSAPCA	eTSID	PCLR	PGPD	TID	EPLL	Ours
$\sigma_n = 30$									
Gore	29.49	27.30	28.56	29.19	29.04	29.38	29.68	29.21	**29.95**
Cat	28.35	24.30	26.01	26.98	30.11	30.07	26.21	29.77	**31.18**
CMU	31.59	29.63	30.74	30.65	32.66	32.55	28.56	32.59	**33.38**
View	26.79	25.21	26.13	24.45	28.37	28.34	23.90	28.17	**31.96**
$\sigma_n = 50$									
Gore	26.95	26.21	26.79	26.64	25.80	26.70	27.55	26.59	**27.82**
Cat	26.42	23.57	25.07	25.68	27.87	28.01	24.77	27.74	**28.79**
CMU	29.27	28.12	29.06	28.35	30.26	30.18	27.12	29.51	**30.64**
View	24.88	24.47	24.71	23.42	26.32	26.39	23.01	26.13	**28.64**
$\sigma_n = 70$									
Gore	25.42	25.11	25.37	24.68	23.70	24.89	25.40	24.69	**25.58**
Cat	25.13	23.23	24.08	24.16	26.48	26.63	23.34	26.23	**26.80**
CMU	27.68	27.31	27.63	26.23	28.66	28.57	26.04	27.86	**28.72**
View	23.55	23.54	23.81	23.23	25.05	25.21	22.19	24.86	**27.16**
$\sigma_n = 100$									
Gore	23.38	23.26	23.32	22.13	21.93	23.00	23.30	22.85	**23.86**
Cat	23.90	22.20	23.10	22.44	25.09	25.12	23.06	24.82	**25.21**
CMU	25.88	25.65	25.95	23.84	**26.91**	26.71	24.49	26.13	26.59
View	22.28	22.67	21.84	22.63	23.80	23.93	21.25	23.61	**24.79**

H. Comparisons with internal denoising methods

We first present the quantitative comparisons with the state-of-the-art internal denoising methods in Table III. The scores are averaged across all test images in the datasets. Overall, our method is the best performer.

In Figure 6 it is visible that the proposed algorithms can restore high-frequency details with a closer resemblance to the ground truth than the existing internal denoising methods. Specifically, the highly-textured pattern is clearly reproduced by our method, while these details are highly distorted or smoothed out by the other methods. Upon close inspection, most of the other methods either smooth out the periodical variations of the background texture or introduce additional artifacts and artificial textures. This phenomenon explains the much inferior PSNR produced by the other methods.

I. Comparisons with external denoising methods

In Table IV, we present the average PSNR measured across the Gore, Cat, CMU-PIE and Multiview datasets. Among the considered methods, ours is the best overall performer (in terms of PSNR) across most combinations of datasets and noise levels.

In addition to the superior quantitative results, our method also delivers superior visual quality. As an example, we

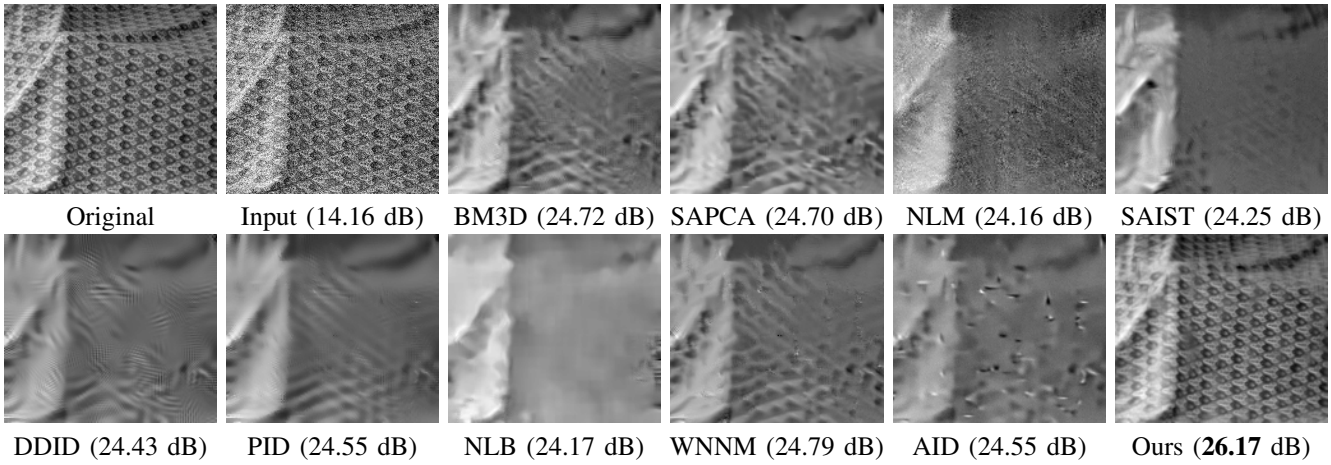


Fig. 6. Visual denoising results produced for $\sigma_n = 50$, by several methods for a sample texture image from the Multiview dataset [36]. Our method is able to recover much more texture details as compared to competing methods.

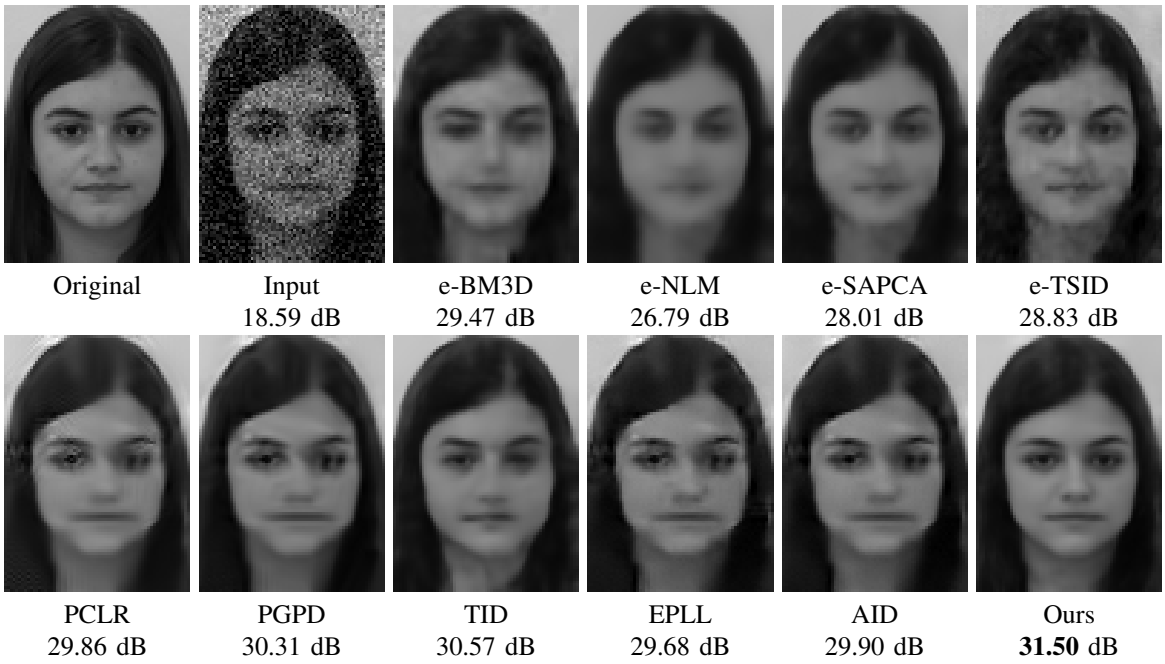


Fig. 7. Denoising results achieved by various methods for a sample image with a noise standard deviation $\sigma_n = 30$. The ground truth image is from the Gore dataset [35].

provide visual comparisons between the results generated by our method and the state of the art alternatives for a face image with the noise level $\sigma_n = 30$ and a texture image with the noise level $\sigma_n = 50$, as shown in Figures 7 and 8, respectively.

In Figure 7, the face image denoised by our method is indeed of higher visual quality than their counterparts. Within the face region, our algorithm can reproduce all the facial parts without distortion, whereas the other methods causes different kinds of artifacts. Furthermore, most of the other methods in this comparison introduce visible artifacts on the forehead and the chin. The lower performance of the other methods could be explained by their difficulty in finding correct matches for patch grouping due to high noise and high variance within each patch group. As a result of inaccurate grouping of patches, texture details are destroyed and incoherent patterns are generated.

In Figure 8, the proposed algorithms is able to restore high-frequency details with a closer resemblance to the ground truth than existing methods. Specifically, the highly-textured pattern is clearly reproduced by our method, while these details are highly distorted or smoothed out by the other methods. Upon close inspection, most of the other methods either smooth out the periodical variation of the background texture or introduce additional artifacts and artificial textures. This phenomenon implies a much inferior PSNR produced by others than our method.

J. Robustness to misalignment and rotation

In the top row of Figure 9, we show sample noise-free images in the Cat database. In the bottom row, we show a

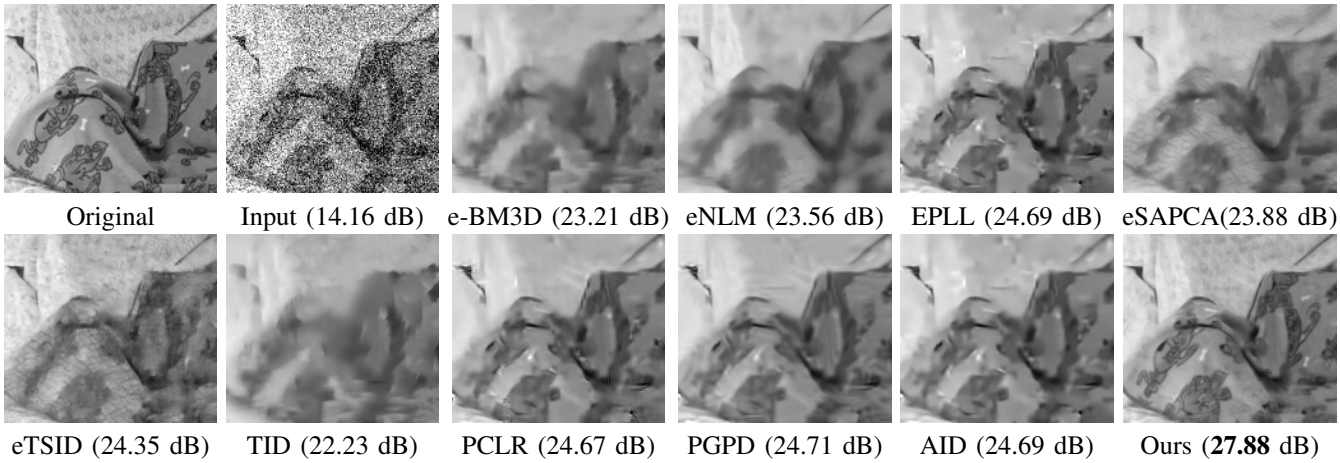


Fig. 8. Visual denoising results for a texture image selected from the Multiview dataset [36] where $\sigma_n = 50$. Our method is able to recover much more texture details than the others (please zoom-in to see details).

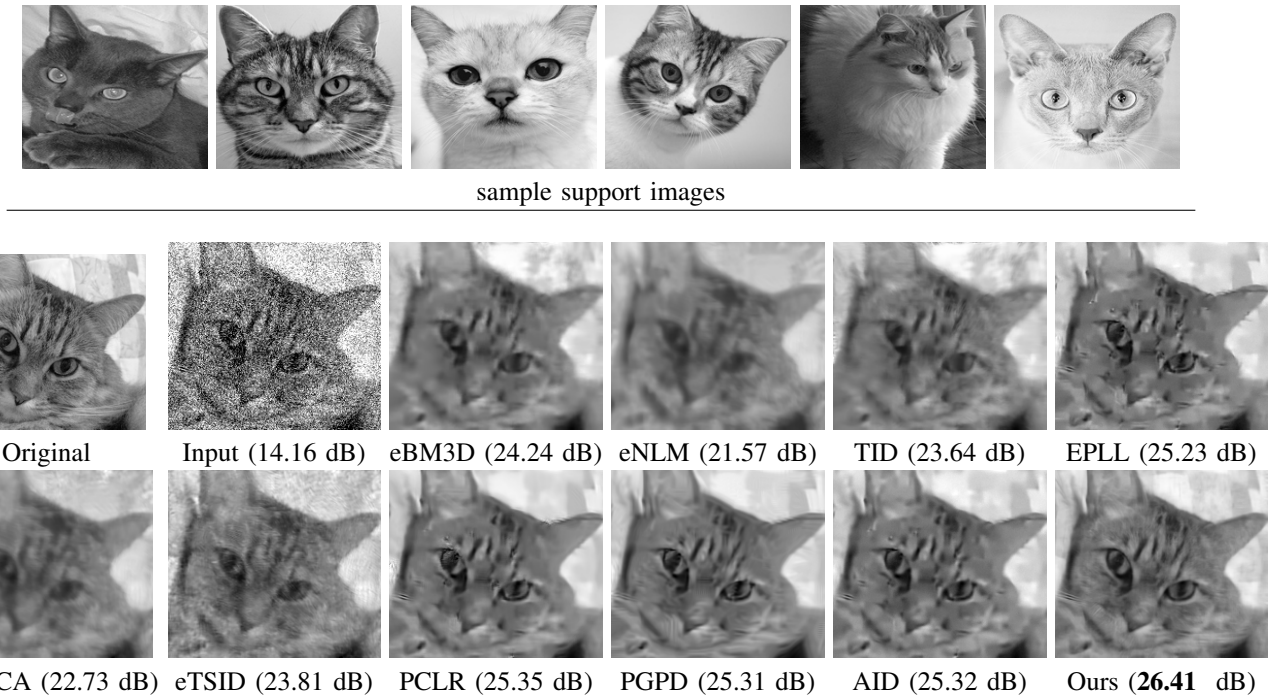


Fig. 9. Denoising results for different methods from the dataset in [34] when $\sigma_n = 50$. The top two rows show the candidate images from the dataset that are most similar to the noisy image.

noisy input image and the corresponding denoised one. We observe that while the appearances and expressions of cats in the support images significantly vary, and there are severe misalignments between them, our method still generates much higher PSNR than other methods.

K. Extension to color images

For noisy color images, we first perform a luminance-chrominance[2] transformation. Let Y denotes the luminance channel, and U and V denote the chrominance channels. Often, the luminance channel provides prominent texture information while the chrominance channels endure lower SNR [44]. We specifically deal with the high noise variance in the Y channel with our method, while simply applying BM3D to the chrominance channels. In Table V and Fig. 10, we present comparison with the current state-of-the-art color image denoising algorithms [6], [43]. One can observe that our method outperforms all existing methods on three benchmark datasets for five different noise levels.

[2]We consider opponent color models yet any other transformation such as *YCbCr, Lab* can be used.

V. Conclusion and Future Work

We have presented an effective algorithm for denoising object images using support patches from an image dataset of the same category. The patch selection strategy aims to draw support patches within a locality of the input patch from

TABLE V
DENOISING PERFORMANCE IN PNSR (DB) ON COLOR IMAGES FOR NOISE LEVELS $\sigma_n = 20, 50, 70, 80, 100$. BEST RESULTS ARE IN BOLD.

	Methods	CBM3D [43]					NLB [6]					Ours				
	σ_n	30	50	70	80	100	30	50	70	80	100	30	50	70	80	100
Datasets	FEI	35.59	33.23	31.55	30.88	29.68	34.99	33.32	31.60	30.91	29.65	**35.99**	**33.66**	**32.30**	**31.62**	**30.39**
	Views	29.28	27.19	25.94	25.46	24.60	29.06	27.12	25.44	24.95	24.02	**30.07**	**27.75**	**26.94**	**26.50**	**25.65**
	CMU	31.70	29.29	27.68	27.05	25.86	32.56	29.11	25.72	24.41	22.23	**32.84**	**30.90**	**29.56**	**29.15**	**28.30**

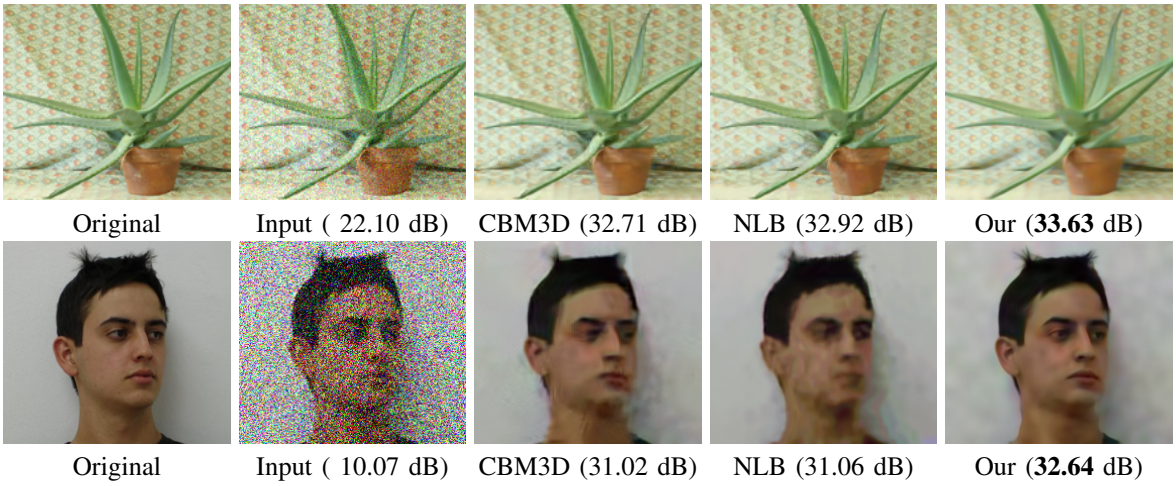


Fig. 10. Comparison of a few denoising methods on color images from the datasets in [36] and [39], where the noise standard deviations are $\sigma_n = 20$ and $\sigma_n = 80$, respectively. Our method is able to recover much more details than the others.

the best candidate images. The key difference from existing external denoising methods is the formulation of the denoising problem in a transform domain. In addition, we include novel terms to model support patch group membership and to promote the similarity between the noisy and the support patches. We have validated the robustness of our algorithm to the dataset size, the number of support patches, and verified the importance of choosing the appropriate dataset category. Overall, our algorithm outperforms all state-of-the-art methods included in our study, both numerically and visually.

An important question that requires more discussion is the behaviour and sensitivity of the algorithm even larger variations in pose, facial expressions, size, style, and view angle. Seeking an answer to this question will help us in improving the robustness of the algorithm. This aspect of our method will be studied in our future work.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014, pp. 580–587. 1

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105. 1

[3] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *CVPR*, 2005, pp. 60–65. 1, 2, 6

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *Image Processing, IEEE Transactions on*, pp. 2080–2095, 2007. 1, 2, 6

[5] ——, "BM3D image denoising with shape-adaptive principal component analysis," in *Signal Processing with Adaptive Sparse Structured Representations*, 2009. 1, 6

[6] M. Lebrun, A. Buades, and J.-M. Morel, "A nonlocal bayesian image denoising algorithm," *SIAM Journal on Imaging Sciences*, pp. 1665–1688, 2013. 1, 2, 11, 12

[7] W. Dong, G. Shi, and X. Li, "Nonlocal image restoration with bilateral variance estimation: a low-rank approach," *Image Processing, IEEE Transactions on*, pp. 700–711, 2013. 1, 4, 5

[8] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE transactions on image processing*, pp. 1395–1411, 2007. 1, 6

[9] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, pp. 1531–1549, 2010. 1, 6

[10] B. Goossens, H. Luong, A. Pizurica, and W. Philips, "An improved non-local denoising algorithm," in *Local and Non-Local Approximation in Image Processing, International Workshop, Proceedings*, 2008, p. 143. 1, 2

[11] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds," in *CVPR*, 2011, pp. 2833–2840. 1

[12] A. Levin, B. Nadler, F. Durand, and W. T. Freeman, "Patch complexity, finite pixel correlations and optimal denoising," in *ECCV*, 2012, pp. 73–86. 1

[13] S. H. Chan, T. Zickler, and Y. M. Lu, "Monte carlo non-local means: Random sampling for large-scale image filtering," *TIP*, pp. 3711–3725. 1

[14] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *Image Processing, IEEE Transactions on*, pp. 3736–3745, 2006. 1, 2

[15] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *ICCV*, 2009, pp. 2272–2279. 1

[16] W. Dong, X. Li, D. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *CVPR*, June 2011, pp. 457–464. 1

[17] Q. W. Y. B. Zhiyuan Zha, Xinggan Zhang and L. Tang., "Group sparsity residual constraint for image denoising," in *arXiv preprint arXiv:1703.00297,*, 2017. 1

[18] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *ICCV*, 2011, pp. 479–486. 1, 2, 6

[19] L. Z. F. Chen and H. Yu, "External Patch Prior Guided Internal Clustering for Image Denoising," in *ICCV*, 2015, pp. 1211–1218. 1, 2, 6

[20] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng, "Patch Group Based Nonlocal Self-Similarity Prior Learning for Image Denoising," in *ICCV*, 2015, pp. 1211–1218. 1, 6

[21] E. Luo, S. H. Chan, and T. Q. Nguyen, "Adaptive image denoising by mixture adaptation," *TIP*, vol. 25, no. 10, Oct 2016. 2

[22] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Image restoration with locally selected class-adapted models," in *MLSP*, Sept 2016. 2

[23] E. Luo, S. H. Chan, and T. Q. Nguyen, "Adaptive image denoising by targeted databases," *Image Processing, IEEE Transactions on*, pp. 2167–2181, 2015. 2, 6, 7

[24] H. Yue, X. Sun, J. Yang, and F. Wu, "Cid: Combined image denoising in spatial and frequency domains using web images," in *CVPR*, June 2014, pp. 2933–2940. 2, 6

[25] S. Anwar, C. Phuoc Huynh, and F. Porikli, "Class-specific image deblurring," in *ICCV*, 2015, pp. 495–503. 2

[26] L. Sun, S. Cho, J. Wang, and J. Hays, "Good image priors for non-blind deconvolution," in *ECCV*, 2014, pp. 231–246. 2

[27] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004. 2

[28] E. Cands and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, pp. 717–772, 2009. 4

[29] J.-F. Cai, E. J. Candès, and Z. Shen, "A Singular Value Thresholding Algorithm for Matrix Completion," *SIAM J. on Optimization*, pp. 1956–1982, 2010. 5

[30] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Modeling & Simulation*, pp. 460–489, 2005. 5

[31] J. Xu and S. Osher, "Iterative regularization and nonlinear inverse scale space applied to wavelet-based denoising," *Image Processing, IEEE Transactions on*, pp. 534–544, 2007. 5

[32] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," *Automatic Face and Gesture Recognition*, pp. 46–51, 2002. 6

[33] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D Object Representations for Fine-Grained Categorization," in *ICCVW*, 2013, pp. 554–561. 6

[34] W. Zhang, J. Sun, and X. Tang, "Cat head detection-how to effectively exploit shape and texture features," in *ECCV*, 2008, pp. 802–816. 6, 11

[35] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *TPAMI*, pp. 2233–2246, 2012. 6, 8, 10

[36] H. Hirschmüller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *CVPR*, 2007, pp. 1–8. 6, 10, 11, 12

[37] H. Yue, X. Sun, J. Yang, and F. Wu, "Image denoising by exploring external and internal correlations," *TIP*, pp. 1967–1982, 2015. 6

[38] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *CVPR*, 2014, pp. 2862–2869. 6

[39] C. E. Thomaz and G. A. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image and Vision Computing*, pp. 902–913, 2010. 8, 12

[40] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *TPAMI*, pp. 2227–2240, 2014. 7

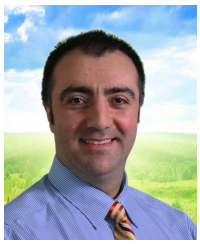
[41] M. Mahmoudi and G. Sapiro, "Fast image and video denoising via nonlocal means of similar neighborhoods," *Signal Processing Letters, IEEE*, pp. 839–842, 2005. 7

[42] R. Vignesh, B. T. Oh, and C.-C. J. Kuo, "Fast non-local means (nlm) computation with probabilistic early termination," *Signal Processing Letters, IEEE*, pp. 277–280, 2010. 7

[43] 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 *International Conference on Image Processing*, vol. 1. IEEE, 2007, pp. I–313. 11, 12

[44] O. Pirinen, A. Foi, and A. Gotchev, "Color high dynamic range imaging: The luminance–chrominance approach," *International journal of imaging systems and technology*, vol. 17, no. 3, pp. 152–162, 2007. 11

Saeed Anwar received Bachelor degree in Computer Systems Engineering with distinction from University of Engineering and Technology (UET), Pakistan, in July 2008, and Master degree in Erasmus Mundus Vision and Robotics (Vibot), jointly from Heriot watt University United Kingdom (HW), University of Girona Spain (UD) and University of Burgundy France in August 2010 with distinction. During his masters, he carried out his thesis at Toshiba Medical Visualization Systems Europe (TMVSE), Scotland. He has also been a visiting research fellow at Pal Robotics, Barcelona in 2011. Since 2014, he is a PhD student at the Australian National University (ANU) and Data61/CSIRO. He has also been working as a Lecturer and Assistant Professor at the National University of Computer and Emerging Sciences (NUCES), Pakistan. His major research interests are low-level vision, image enhancement, image restoration, computer vision, and optimization.



Fatih Porikli is an IEEE Fellow and a Professor in the Research School of Engineering, Australian National University (ANU). He is also managing the Computer Vision Research Group at Data61/CSIRO. He has received his PhD from New York University in 2002. Previously he served Distinguished Research Scientist at Mitsubishi Electric Research Laboratories. Prof. Porikli is the recipient of the R&D 100 Scientist of the Year Award in 2006. He won 4 best paper awards at premier IEEE conferences and received 5 other professional prizes. Prof. Porikli authored more than 150 publications and invented 66 patents. He is the co-editor of 2 books. He is serving as the Associate Editor of 5 journals for the past 8 years. He was the General Chair of AVSS 2010 and WACV 2014, and the Program Chair of WACV 2015 and AVSS 2012. His research interests include computer vision, deep learning, manifold learning, online learning, and image enhancement with commercial applications in video surveillance, car navigation, intelligent transportation, satellite, and medical systems.

Cong Phuoc Huynh is a Machine Learning scientist at Amazon Lab126, and concurrently an adjunct research fellow at the Australian National University. He has co-authored a book on imaging spectroscopy for scene analysis and over 20 journal articles and conference papers in computer vision and pattern recognition. He is an inventor of eight patents on spectral imaging. He is a co-recipient of a DICTA Best Student's paper Award in 2013. Previously, he was a computer vision researcher at National ICT Australia (NICTA). He received a B.Sc. degree (Hons) in Computer Science and Software Engineering from the University of Canterbury, New Zealand in 2006, and M.Sc. and Ph.D. degrees in Computer Science from the Australian National University (ANU) in 2007 and 2012.