

EFFICIENT NONLOCAL-MEANS DENOISING USING THE SVD

Jeff Orchard¹, Mehran Ebrahimi², Alexander Wong³

University of Waterloo

¹Cheriton School of Computer Science, ²Dept. of Applied Mathematics, ³Systems Design Engineering Dept.
Waterloo, Ontario, Canada N2L 3G1
{jorchard, m2ebrahi, a28wong}@uwaterloo.ca

ABSTRACT

Nonlocal-means (NL-means) is an image denoising method that replaces each pixel by a weighted average of all the pixels in the image. Unfortunately, the method requires the computation of the weighting terms for all possible pairs of pixels, making it computationally expensive. Some short-cuts assign a weight of zero to any pixel pairs whose neighbourhood averages are too dissimilar. In this paper, we propose an alternative strategy that uses the SVD to more efficiently eliminate pixel pairs that are dissimilar. Experiments comparing this method against other NL-means speed-up strategies show that its refined discrimination between similar and dissimilar pixel neighbourhoods significantly improves the denoising effect.

Index Terms— nonlocal-means, denoising, SVD

1. INTRODUCTION

Image denoising is the operation of removing unwanted noise from a noise-corrupted image, restoring the image to its un-degraded ideal.

Recently, the method of nonlocal-means (NL-means) has shown great promise [1, 2]. It involves replacing each pixel with a weighted average of other pixels with similar neighbourhoods. The idea is that images often contain self-similarity, and many structures show up more than once in the image. However, any noise in these similar neighbourhoods is uncorrelated if we assume an i.i.d. noise model. Hence, taking an average of these similar neighbourhoods yields a version that has less noise.

Unfortunately, the ideal implementation of NL-means is very computationally expensive. A number of recent papers on NL-means denoising focus on short-cuts to make the method computationally practical [1, 3, 4, 5, 6]. One of the most compelling strategies is to rule out many weight computations using a simple and computationally cheap test [6]. While the method is among the best short-cuts for NL-means, their approach is somewhat heuristic.

In this paper, we propose a different method for identifying dissimilar pixel pairs. Our method has its theoretical foundation in the singular value decomposition (SVD), and yields

the most efficient representation (on average) to quickly rule out dissimilar neighbourhoods.

2. NL-MEANS DENOISING

Nonlocal-means denoising [7, 8] replaces the intensity of each pixel x in the noisy image \mathbf{u} by a weighted average of all of the pixel intensities in the image (as a convention, we will refer to the pixel being denoised at any given time as the “pixel of interest” (POI), denoted x , and denote all other pixels as y). The weights $w(x, y)$ reflect the probability that the POI (x) has the same intensity value as the pixel it is being compared to (y). This probability is based on the similarity between the neighbourhoods around x and y . A small neighbourhood, or “patch”, around each pixel is used to compute the L2 norm. The weighting factor, $w(x, y)$, is then a normalized weighted Gaussian function of this L2 norm.

Consider a discrete noisy image $\mathbf{u} = \mathbf{f} + \mathbf{n}$, in which \mathbf{n} is additive white Gaussian noise. The NL-means filter is written

$$\mathbf{f}(x) \approx \sum_y w(x, y) \mathbf{u}(y), \quad (1)$$

where,

$$w(x, y) = \frac{1}{W(x)} \exp\left(\frac{-1}{h^2} \left\| \mathbf{u}(\mathcal{N}^d(x)) - \mathbf{u}(\mathcal{N}^d(y)) \right\|_2^2\right),$$

where $\mathcal{N}^d(x)$ represents the square patch of size $(2d + 1) \times (2d + 1)$ centered at x , and W is a normalizing term, $W(x) = \sum_y w(x, y)$. The parameter h will be referred to as the filter parameter that controls the decay of the exponential expression in the weighting scheme. This parameter is typically controlled manually in the algorithm. Choosing a very small h leads to noisy results identical to the input, while very large h gives an overly-smoothed image.

Note that the above computation is required for all pixel pairs in the image, $(x, y) \in [N \times N] \times [N \times N]$. For denoising an $N \times N$ image using patches of radius of d , each distance computation is of $\mathcal{O}(d^2)$, and hence the computational complexity of computing all the weights for all N^2 of the POIs is $\mathcal{O}(N^4 d^2)$.

3. ACCELERATING NL-MEANS

Buades *et al.* [1] proposed the idea of using only a subset of all possible pixels in the weighted average of (1), restricting the computation to a limited range of pixels in the neighbourhood of the POI, and effectively assigning a weight of zero to all other pixels. Consider a radius of s defining a neighbourhood with dimensions $(2s + 1) \times (2s + 1)$ centred on x . A total of $(2s + 1)^2$ patch comparisons (weights) are evaluated for every POI. Hence, the computational complexity of evaluating the weights for all N^2 POIs in an image is $\mathcal{O}(N^2 d^2 s^2)$. If s is much smaller than N , the time savings can be very significant. However, this method is not nonlocal in the strict sense, and a small s can defeat the purpose of using NL-means. However, combatting the issue with a larger s slows the method down. We will refer to this speed-up strategy as “limited-range” NL-means.

A different speed-up proposed by Mahmoudi and Sapiro [6] also computes only a fraction of the total set of weights. For a given POI, they choose the $(2s + 1)^2$ patches with patch average closest to that of the POI, terminating the list prematurely if they run out of patches with averages within about 10% the POI patch average. That list of candidate patches is further pruned by comparing the local average gradient vectors, excluding patches for which the gradient direction differs by an angle that exceeds a threshold. However, since noise can drastically affect the direction of a short gradient vector, the above pruning criterion is only applied if both gradient magnitudes are above a chosen threshold. Any pixel whose patch makes it through those two elimination rounds is used in the weighted average (see (1)) to denoise the POI. This whole process is repeated for each pixel to be denoised. For the remainder of this paper, we will refer to this method as “pre-filtering” NL-means.

Approaches similar to the pre-filtering method have recently been published. Coupé *et al.* [4] and Kervrann *et al.* [5] use patch average and patch variance to rule out dissimilar patches, although these methods also incorporate a limited search window in addition to pre-filtering. In [9], the search window radius is adjusted to optimize the patch fit.

A problem with these pre-filtering methods is that the measures by which pre-selection is based are chosen somewhat heuristically. In the next section, we propose a method for pre-filtering that is based on the statistical properties afforded by the singular SVD.

4. NL-MEANS USING THE SVD

If patches are $(2d + 1) \times (2d + 1)$ and we define D as $(2d + 1)^2$, then each patch can be represented as a point (or vector) in the space \mathbb{R}^D . While this space represents the full gamut of patches that are possible, we hypothesize that most natural images will place most of their patches in a relatively modest lower-dimensional manifold. If we seek only linear subspaces

of \mathbb{R}^D , then we can use the SVD [10] to extract the most significant subspaces (in the sense of minimizing the L2 norm).

Without loss of generality, order the N^2 pixels in the image lexicographically so that we can refer to every pixel with a single index variable, $i \in \{1, \dots, N^2\}$. Let \mathbf{z}_i be the column vector in \mathbb{R}^D representing the $(2d + 1) \times (2d + 1)$ patch centred on pixel i . Then we define \mathbf{M} as the matrix $[\mathbf{z}_1 \dots \mathbf{z}_{N^2}]^T$. Hence, \mathbf{M} contains all the patches in the image, one stored in each row. We then compute the SVD of \mathbf{M} , $\mathbf{U}\Sigma\mathbf{V}^T = \mathbf{M}$. Note that $\mathbf{U}^T\mathbf{U}$ yields the identity matrix, as does $\mathbf{V}^T\mathbf{V}$. Hence, the matrices \mathbf{U} and \mathbf{V} have orthonormal columns. The matrix Σ is a diagonal matrix with diagonal elements $\{\sigma_1, \sigma_2, \dots, \sigma_D\}$. The values $\{\sigma_k\}_{k=1}^D$ are called *singular values*, and are always in decreasing order, $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots \geq \sigma_D \geq 0$.

The remarkable property of the SVD is its statistical representation of \mathbf{M} in subspaces of decreasing importance. More precisely, one can reconstruct an approximation to \mathbf{M} by setting all of the singular values to zero except σ_1 . The resulting matrix has a rank of one. Likewise, it can be proven that \mathbf{M}_K , the rank- K approximation to \mathbf{M} using only the first K singular values, is the closest rank- K matrix to \mathbf{M} . That is, $\mathbf{M}_K = \arg \min_{\mathbf{A}} \|\mathbf{M} - \mathbf{A}\|_F$, where $\|\cdot\|_F$ is the Frobenius norm, the sum of the squares of the matrix elements.

In our application, the columns of \mathbf{V} (or rows of \mathbf{V}^T) can be used as a basis for the space of image patches, \mathbb{R}^D . Let $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_D\}$ be the basis comprised of singular vectors, and $\{\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \dots, \tilde{\mathbf{v}}_D\}$ be any other basis of \mathbb{R}^D . Consider any image patch \mathbf{z} , and denote its coordinates with respect to the two bases as $\{\alpha_1, \alpha_2, \dots, \alpha_D\}$ and $\{a_1, a_2, \dots, a_D\}$, respectively, such that $\mathbf{z} = \alpha_1\mathbf{v}_1 + \dots + \alpha_D\mathbf{v}_D$ and $\mathbf{z} = a_1\tilde{\mathbf{v}}_1 + \dots + a_D\tilde{\mathbf{v}}_D$. In approximating the patch \mathbf{z} , the SVD suggests that, on average,

$$\|\mathbf{z} - (\alpha_1\mathbf{v}_1 + \dots + \alpha_r\mathbf{v}_r)\| \leq \|\mathbf{z} - (a_1\tilde{\mathbf{v}}_1 + \dots + a_r\tilde{\mathbf{v}}_r)\|,$$

where $r \leq D$ is the number of basis vectors used in the approximation to \mathbf{z} , and $\|\cdot\|$ is the Euclidean distance (L2 norm). In other words, if we want to approximate image patches using fewer than D coordinates, we cannot do better on average than the basis of singular vectors contained in \mathbf{V} .

This line of reasoning suggests that we should do our patch selection based on each patch’s sequence of coordinates using the basis of singular vectors.

As with the other NL-means speed-ups, the method we propose computes the norms for only a subset of all patch pairs. The challenge is to efficiently find the most significant subset of patch pairs, the ones with the lowest norm (and hence highest weight).

Here, we describe our strategy. Consider a single POI, x . We proceed by estimating the L2 norm between the patch around x and all other patches using rank-1 approximations. That is, if α_1 is the first coordinate of our POI’s patch, $\mathbf{u}(\mathcal{N}^d(x))$, and β_1 is the first coordinate of another patch,

$\mathbf{u}(\mathcal{N}^d(y))$, then the norm between their rank-1 approximations is simply $(\alpha_1 - \beta_1)^2$. (More generally, since the basis is orthonormal, we can approximate the norm using the first r coordinates as $\sum_{k=1}^r (\alpha_k - \beta_k)^2$.)

Based on these rank-1 approximations, we eliminate the fraction of the pixels that yield the largest norm. With this smaller set of candidate pixels, we add to their norms the contribution from the second coordinate, $(\alpha_2 - \beta_2)^2$. Again, we threshold the new-and-improved norm estimates to eliminate the highest values. Continuing this process, we progressively narrow the set of candidate pixels. The nature of the SVD implies that our elimination scheme leaves us with a set of pixels with lower average norm than we could have achieved using any other basis.

5. EXPERIMENTS

In our experiments, we compare the performance of three NL-means implementations: the limited-range (LR) method of Buades *et al.* [1], the pre-filtering (PF) method of Mahmoudi and Sapiro [6], and our SVD-based approach (SVD). All three methods were implemented in MATLAB (Nattick, Massachusetts).

The three implementations are compared in two experiments. The first experiment is designed to see which method gives the highest PSNR (peak signal-to-noise ratio) for a fixed number of norm computations. The second one is designed to compare the quality of patches chosen by the different methods.

In the first experiment, we allow each method a fixed number of operations for computing norms. We measure the operations in units of “norm-term operations”. For example, computing the L2 norm between two vectors in \mathbb{R}^D involves adding up D terms. The limited-range method computes N^2SD norm terms, where $S = (2s + 1)^2$, $D = (2d + 1)^2$, and the image is $N \times N$. We choose parameters for the pre-filtering method so that approximately S patches make it through the elimination rounds. Hence, the pre-filtering method computes about the same number of norm terms as the limited-range method.

For the SVD method, approximately 95% of the patches are eliminated based on the rank-1 norm estimate, and 20% are eliminated for each subsequent norm increment after that (up to a maximum of 20 increments, for this experiment). As we bring in more coefficients, we keep track of the total number of norm terms computed. For example, we compute N^2 norm terms when computing the rank-1 estimates, but only $0.05N^2$ terms for the next increment. For each test image, the elimination rates are adjusted slightly to arrive at about N^2SD norm terms in total.

Given this fixed number of norm terms, each of the three methods is used to denoise three images that are artificially corrupted with additive Gaussian noise ($\sigma = 18$). We then compare the resulting peak signal-to-noise ratios (PSNR).

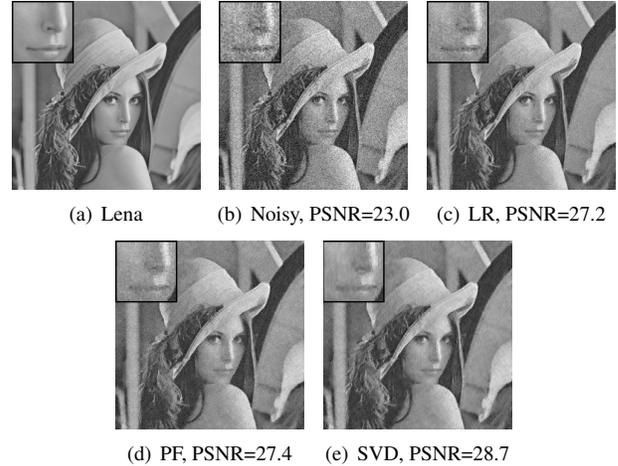


Fig. 1. Lena image (256×256) with $d = 3$ and $h = 40$.

The idea behind the second experiment is to choose a number, say S , and ask each of the three methods to choose the S patches that it would use to denoise each pixel. The lower the L2 norm of the chosen patch pairs, the better the denoising should be. To summarize the distribution of these patch-pair norms, we compute the minimum norm, mean norm, and maximum norm for each POI, and report their averages for each test image.

6. RESULTS

The outcome of experiment 1 is shown in Figs. 1 and 2. The PSNR for the limited-range method is usually slightly lower than that for the pre-filtering method, which itself is lower than the PSNR of the SVD method. In a third test image (denoted “weave”, not shown), the limited-range and pre-filtering methods yield very similar PSNR values. The improvement of the SVD method over the pre-filtering method ranges from 0.7dB to 1.3dB.

The first six singular vectors for the Lena image are displayed in Fig. 3. The first (most significant) singular vector is indistinguishable from a constant image, suggesting that patch average, as used by other pre-filtering methods [6, 4, 5], is actually a good first-order classifier.

The results for experiment 2 are shown in Fig. 4. While the minimum-norm value is very similar for all three methods, there is a marked difference in the mean and maximum norms. Recall that for this experiment each of the three methods is configured to arrive at the same number of candidate patches (on average) for each pixel. Thus, the lower distribution of the norms for the SVD method suggests that it was far more successful at finding close matches.

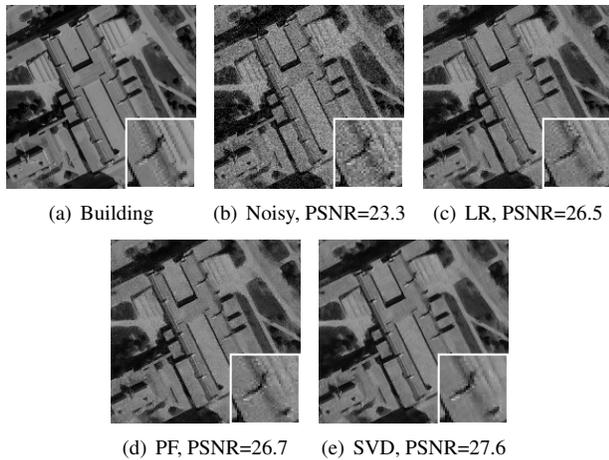


Fig. 2. Building image (182×177) with $d = 3$ and $h = 50$.

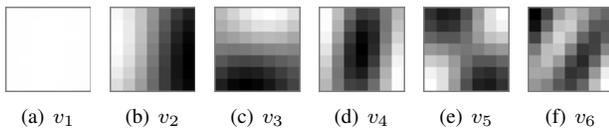


Fig. 3. First six singular vectors for Lena ($d = 3$)

7. CONCLUSIONS

In both experiments, the SVD method consistently selects patch pairs that are more similar than those selected by the limited-range and pre-filtering methods. The SVD method’s ability to efficiently extract the most significant patch pairs yields substantially higher PSNR values. We note, however, that the first criteria for eliminating candidate matches is essentially the same for the pre-filtering and SVD methods – similarity in patch average.

For experiment 1, the SVD method used a truncated set of 20 singular vectors. The effect of ignoring the remaining 29 singular vectors generally amounts to selective smoothing. This fact might contribute to the success of the SVD method on images with relatively smooth regions. More work is needed to elucidate the behaviour of the SVD method in these contexts.

8. REFERENCES

[1] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in *Proc. IEEE Intl. Conf. Comp. Vis. Pat. Recog.*, 2005, vol. 2, pp. 60–65.

[2] H. J. Seo, P. Chatterjee, H. Takeda, and P. Milanfar, “A comparison of some state of the art image denoising methods,” in *Proc. 41st Asilomar Conference on Signals, Systems, and Computers*, 2007.

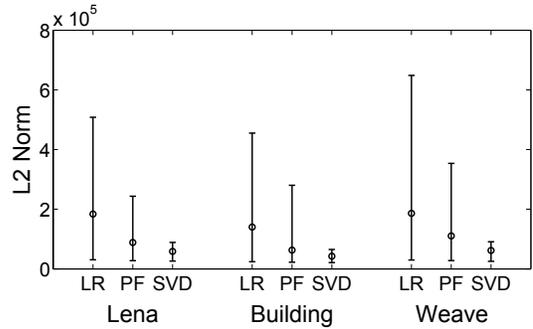


Fig. 4. Distribution of norms for patch pairs selected by the three methods. “LR” denotes the limited-range method [1], “PF” denotes the pre-filtering method [6], and “SVD” denotes the SVD method.

[3] R. C. Bilcu and M. Vehvilainen, “Fast non-local means for image de-noising,” in *Digital Photography III Proc. of SPIE IS&T*, Russel A. Martin, Jeffrey M. DiCarlo, and Nitin Sampat, Eds., February 2007, vol. 6502.

[4] P. Coupé, P. Yger, and C. Barillot, “Fast non local means denoising for 3D MR images,” in *Proc. MICCAI’06 (LNCS 2879)*, 2006, vol. 9, pp. 33–40.

[5] C. Kervrann, J. Boulanger, and P. Coupé, “Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal,” in *Proc. of SSVM*, F. Sgallari, A. Murli, and N. Paragios, Eds., Berlin, 2007, vol. LNCS 4485, pp. 520–532, Springer-Verlag.

[6] M. Mahmoudi and G. Sapiro, “Fast image and video denoising via nonlocal means of similar neighborhoods,” *IEEE Sig. Proc. Let.*, vol. 12, no. 12, pp. 839–842, December 2005.

[7] A. Buades, B. Coll, and J.-M. Morel, “A review of image denoising algorithms, with a new one,” *SIAM J. Multiscale Modeling and Sim.*, vol. 4, no. 2, pp. 490–530, 2005.

[8] A. Buades, B. Coll, and J.-M. Morel, “Nonlocal image and movie denoising,” *Intl. J. Comp. Vis.*, vol. (to appear), 2007.

[9] C. Kervrann and J. Boulanger, “Optimal spatial adaptation for patch-based image denoising,” *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2866–2878, October 2006.

[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Third Edition, Johns Hopkins University Press, Baltimore, MD, 1996.