Denoising an Image by Denoising Its Components in a Moving Frame

Gabriela Ghimpeţeanu[1], Thomas Batard[1], Marcelo Bertalmío[1], and Stacey Levine[2]

[1] Universitat Pompeu Fabra, Spain
[2] Duquesne University, USA

Abstract. In this paper, we provide a new non-local method for image denoising. The key idea we develop is to denoise the components of the image in a well-chosen moving frame instead of the image itself. We prove the relevance of our approach by showing that the PSNR of a grayscale noisy image is lower than the PSNR of its components. Experiments show that applying the Non Local Means algorithm of Buades et al. [5] on the components provides better results than applying it directly on the image.

Keywords: Image denoising, non-local method, differential geometry.

1 Introduction

Image denoising has been prevalent in the image processing literature for a number of decades. Anisotropic diffusion [14] and total variation based regularization [15] pioneered a rich line of research on edge preserving variational and PDE based methods. More recently, sparsity and self similarity have been used to develop state of the art denoising approaches in the form of patch based and nonlocal methods, e.g. [5, 8, 9]. In fact, it can be argued these techniques are on their way to approaching optimal results [7, 10–12].

In [4] the authors demonstrate that for sufficient noise levels, the unit normals of an image have higher PSNR than the image itself, and the curvature of its level lines have even higher PSNR still and thus theoretically should be easier to denoise. This was reflected in the results obtained from smoothing unit normals beginning with [13], as well as smoothing the curvature [4]. In this framework one generates a denoised image whose unit normals, resp. curvature, matches the smoothed results and whose average intensity along level lines matches that of the noisy image. Challenges do still exist however, mainly in developing an optimal reconstruction algorithm, as well as determining mathematically sound approaches for denoising curvature or normal vector field data.

In [1–3], a new approach was developed where one regularizes the components of a noisy image in a moving frame, the results of which are used to reconstruct a denoised image. An immediate benefit of this approach is that the denoised image is obtained from its components in the moving frame using a straightforward

A. Elmoataz et al. (Eds.): ICISP 2014, LNCS 8509, pp. 375–383, 2014.

invertible transform. Furthermore, denoising the components using a variational framework is mathematically sound and experimental results show improvement over comparable approaches.

In this work we demonstrate that patch based methods can also be used to denoise the components of the noisy image in a moving frame, yielding even better results. To this end, the paper is organized as follows. The moving frame approach is described in section 2. In section 3 we provide experimental evidence that the PSNR of the components in a moving frame is higher than that of the image itself, further justifying this approach. Section 4 contains experimental results demonstrating that patch based methods can be used to denoise the components in this framework, resulting in higher PSNR than comparable approaches. Section 5 contains conclusions and future work.

2 The Moving Frame Approach

Let I be a grey-level image defined on a domain Ω of $\mathbb{R}^2$. We construct a surface S embedded in $(\mathbb{R}^3, \|\,\|_2)$ parametrized by

$$\psi\colon (x,y) \longmapsto (x, y, \mu\, I(x,y)), \qquad \mu > 0. \tag{1}$$

Therefore, with the standard orthonormal frame (e_1, e_2, e_3), where $e_1 = (1,0,0)$, $e_2 = (0,1,0)$, $e_3 = (0,0,1)$, the surface S is given by $\psi(x,y) = xe_1 + ye_2 + \mu\, I(x,y)\, e_3$. Note that for $\mu = 1$ we obtain the graph of the function I.

We construct an orthonormal frame field (Z_1, Z_2, N) of $(\mathbb{R}^3, \|\,\|_2)$ over Ω, where $Z_1, Z_2 \in \Gamma(TS)$, i.e. Z_1, Z_2 are tangent vector fields of the surface S. It follows that N is normal to the surface.
In this paper we choose Z_1 as the unit vector field indicating the directions of the gradient and Z_2 as the unit vector field (up to a sign) indicating the directions of the level-lines. Fig. 1 illustrates the construction of the moving frame (Z_1, Z_2, N) for a simple image.

Denoting by P the matrix field encoding the coordinates of the moving frame (Z_1, Z_2, N) with respect to the fixed frame (e_1, e_2, e_3), we have

$$P(x,y) = \begin{pmatrix} \dfrac{I_x}{\sqrt{(I_x^2+I_y^2)(1+\mu^2(I_x^2+I_y^2))}} & \dfrac{-I_y}{\sqrt{I_x^2+I_y^2}} & \dfrac{-\mu\, I_x}{\sqrt{1+\mu^2(I_x^2+I_y^2)}} \\ \dfrac{I_y}{\sqrt{(I_x^2+I_y^2)(1+\mu^2(I_x^2+I_y^2))}} & \dfrac{I_x}{\sqrt{I_x^2+I_y^2}} & \dfrac{-\mu\, I_y}{\sqrt{1+\mu^2(I_x^2+I_y^2)}} \\ \dfrac{\mu(I_x^2+I_y^2)}{\sqrt{(I_x^2+I_y^2)(1+\mu^2(I_x^2+I_y^2))}} & 0 & \dfrac{1}{\sqrt{1+\mu^2(I_x^2+I_y^2)}} \end{pmatrix} \tag{2}$$

It should be noted that when we write I_x it is implied that we mean $I_x(x,y)$, and likewise for I_y. Note that Z_1, Z_2 are not defined on homogeneous regions of

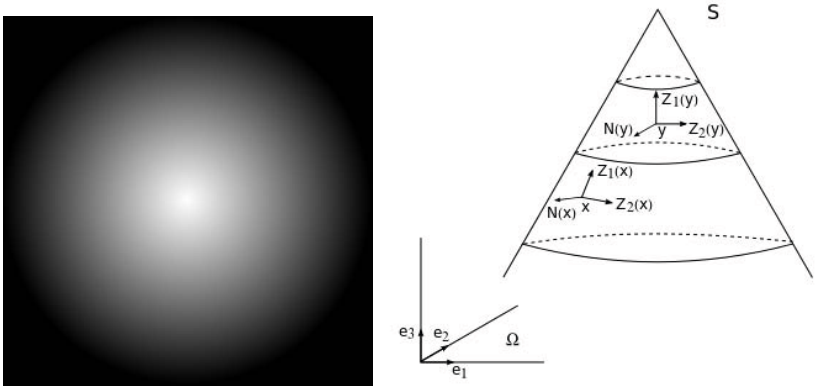

Fig. 1. Orthonormal frame field (Z_1, Z_2, N) of $(\mathbb{R}^3, \| \|_2)$ over Ω. Left: original grey-level image. Right: the frame at two points of the graph S of the image.

I, i.e. at point locations (x, y) where $I_x(x, y) = I_y(x, y) = 0$. In such cases we take $P(x, y)$ as the identity matrix.

Given a grey-level image I and a moving frame associated to I (e.g. the one defined by Eq. (2)), we denote by $J = (J^1, J^2, J^3)$ the components of I in the new frame, i.e. for each image point (x, y) we have the following:

$$\begin{pmatrix} J^1(x,y) \\ J^2(x,y) \\ J^3(x,y) \end{pmatrix} = P^{-1}(x,y) \begin{pmatrix} 0 \\ 0 \\ I(x,y) \end{pmatrix} \tag{3}$$

As $P(x, y) \in SO(3), \forall (x, y) \in \Omega$, the inverse of P is simply its transpose.

It can easily be shown that the component J^2 is always zero. Fig. 2 shows the grey-level image "Lena" and its components J^1 and J^3 when $\mu = 0.05$. We observe that the component J^1 encodes the gradient information of the image, which was expected since the vector field Z_1 is pointing in the direction of the gradient. The component J^3 is similar to the original image, but with highlighted details (contours, textures).

The approach that we take in our proposed framework is that, given a denoising method, it's better to apply it to the components of the noisy image than directly to the image itself. Specifically, let I_0 be a (noisy) grey-level image and (J_0^1, J_0^2, J_0^3) its components in a moving frame associated to I_0 (see formula (3)). The idea developed in [1], [2], [3] is first to apply a regularization method on the components (J_0^1, J_0^2, J_0^3) instead of the original image I_0, obtaining regularized components (J^1, J^2, J^3), and then find the regularized image I whose components are (J^1, J^2, J^3), through the inverse transform:

Fig. 2. From left to right: grey-level image "Lena", component J^1, component J^3

$$\begin{pmatrix} \epsilon(x,y) \\ \delta(x,y) \\ I(x,y) \end{pmatrix} = P(x,y) \begin{pmatrix} J^1(x,y) \\ J^2(x,y) \\ J^3(x,y) \end{pmatrix} \tag{4}$$

The final output of the regularization method is the grey-level image I, while ϵ and δ should be close to zero everywhere.

In [3], a Euclidean heat diffusion is performed on the function J_0 yielding an edge-preserving scale-space of the initial image I_0. In [2], this geometric context was applied to image denoising, dealing with a vectorial extension of the (regularized) Rudin-Osher-Fatemi denoising model. More recently, in [1], the regularized Total Variation in [2] has been replaced by the Vectorial Total Variation (VTV), and this new denoising model is shown to outperform state-of-the-art (local) denoising methods.

3 The Noise Level Is Higher on an Image than on Its Components in a Moving Frame

Let I be a clean grey-level image, with components J^k, $k = 1, 2, 3,$. We add to I Gaussian noise of standard deviation σ, obtaining a noisy image which we call I_0. The aim of this section is to show that the components J_0^k, $k = 1, 2, 3$, of I_0 are less noisy than I_0. For this, we compute the PSNR between J^k and J_0^k and see that it is consistently higher than the PSNR between I and I_0, for different noise levels and for all images in the standard Kodak database (`http://r0k.us/graphics/kodak/`). The computation of the PSNR requires the specification of the peak value, which is 255 for I and it can be shown that it is also 255 for J^3, and

$$peak(J^1) = 255 \times \frac{\sqrt{2} \times 127.5\,\mu}{\sqrt{1 + 2(127.5\,\mu)^2}}$$

for J^1 under the assumption that central differences are used in order to compute the derivatives I_x, I_y.

Table 1 shows the results for $\sigma = 5, 10, 15, 20, 25$ and μ =1.0, 0.1, 0.01, 0.005, 0.001, 0.0001. We observe that the PSNR of the components is higher than the PSNR of the image for $\mu \in]0, 0.005]$ and for each noise level aforementioned. The parameter μ, introduced in formula (1), acts on the range of the image I. As a consequence, it determines the shape of the surface S from which we determine the expression of the moving frame (Z_1, Z_2, N), as it can be seen in formula (2). In particular, the lower μ is, the smoother the surface S is, meaning that the parameter μ can be viewed as a smoothing parameter for the moving frame (Z_1, Z_2, N).

Table 1. Average values of the PSNR for the components J^1, J^3 and the image I over the Kodak database for different noise levels and values of the parameter μ

Noise level	Function	$\mu = 1$	$\mu = 0.1$	$\mu = 0.01$	$\mu = 0.005$	$\mu = 0.001$	$\mu = 0.0001$
$\sigma = 5$	Component J^1	20.51	20.09	34.37	37.84	40.17	40.31
	Component J^3	18.56	26.02	34.24	34.22	34.19	34.19
	Image I	34.19	34.19	34.19	34.19	34.19	34.19
$\sigma = 10$	Component J^1	19.34	15.96	28.21	31.51	33.84	33.97
	Component J^3	16.94	19.84	28.27	28.24	28.21	28.21
	Image I	28.21	28.21	28.21	28.21	28.21	28.21
$\sigma = 15$	Component J^1	18.32	14.16	24.44	27.79	30.09	30.22
	Component J^3	16.32	16.93	24.80	24.77	24.73	24.73
	Image I	24.73	24.73	24.73	24.73	24.73	24.73
$\sigma = 20$	Component J^1	17.37	13.10	21.86	25.12	27.38	27.51
	Component J^3	15.98	15.22	22.38	22.33	22.28	22.27
	Image I	22.27	22.27	22.27	22.27	22.27	22.27
$\sigma = 25$	Component J^1	16.47	12.36	19.89	23.03	25.25	25.38
	Component J^3	15.77	14.10	20.50	20.44	20.37	20.37
	Image I	20.37	20.37	20.37	20.37	20.37	20.37

4 Experiments and Comparisons

As we mentioned above, our general framework is the following: given a denoising method, it is better to apply it to the components of the noisy image than directly to the image itself. This was proved to work for local denoising methods in [1, 2], and here we want to show it is also the case for non-local denoising methods. Therefore, in this section we take a clean image I and add Gaussian noise to it to create the noisy image I_0, and then we use a state of the art non-local denoising method like Non-local Means (NLM) [6] to perform the following experiments:

1. Apply NLM to I_0, obtaining a denoised image which we call I_{NLM}.
2. Compute the components (J_0^1, J_0^2, J_0^3) of I_0 and apply NLM to them, obtaining the denoised components (J_d^1, J_d^2, J_d^3), from which we reconstruct a regularized image I_d using Eq. (4).

Knowing the ground truth I, we can compute and compare the PSNR values of I_{NLM} and I_d.

We have done this for several noise levels and for all images in the Kodak database, and the numerical results are shown in Table 2.

Table 2. Average, over the Kodak database, of the PSNR values for I_{NLM} (obtained by applying NLM to the noisy image) and I_d (obtained by applying NLM to the components of the noisy image), for different noise levels

PSNR	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$
I_d	37.48	33.59	31.57	30.12	29.00
I_{NLM}	37.41	33.38	31.05	30.04	28.91

We can see that our approach is consistently better, for all noise levels. The increase in PSNR that we obtain, while modest, is in agreement with the optimality bounds estimated in [7, 11, 12].

As the two compared methods produce close PSNR averages, we have performed, for each noise level, a t-test to check the significance of our improvement. T-test is testing the null hypothesis that there are no differences between the PSNR means of the two related groups. The five one-tailed paired t-tests applied for the Kodak database of 24 images have produced the results in table 3. Given a particular sample result, the p-value answers the question of what is the probability of obtaining a t-value at least as big as the one obtained, if the null hypothesis is true. For $\sigma = 5$, for a significance level of 0.02 we reject the null hypothesis that there are no significant differences between the PSNR means of the two methods. For $\sigma = 10, 15, 20, 25$ the data provide even stronger evidence that the null hypothesis is false, as the smaller the p-value, the larger the statistical significance. Therefore, we accept the alternative hypothesis, meaning that for each noise level the difference between the two methods is statistically significant.

Table 3. Student's t-test on the difference in PSNR average between I_{NLM} and I_d, over the Kodak database, for different noise levels

t-test	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$
t-value	2.37	8.13	6.17	5.06	5.18
p-value	13×10^{-3}	2×10^{-8}	1×10^{-6}	2×10^{-5}	3×10^{-5}

Table 4 lists the parameter values that we have used for each noise level: apart from μ, defined in Eq. (1), we have $\sigma_k, k = 1, 2, 3$, which is the parameter value employed by NLM when denoising J_0^k. These parameters are fixed for the whole image database. We must point out that although theoretically $J^2 = 0$, in practice and due to numerical errors in the estimation of the derivatives I_x and I_y, J^2 is not exactly 0 and that's why we denoise it anyway.

Fig. 3. Comparing the output of our method with that of applying NLM directly to the noisy image. Left: original noisy image. Middle: NLM result. Right: our result. First row: NLM PSNR=33.41, our result PSNR=34.88. Second row: NLM PSNR=33.10, our result PSNR=34.18. Third row: NLM PSNR=32.55, our result PSNR=33.23. Fourth row: NLM PSNR=32.24, our result PSNR=33.43.

Fig. 4. Comparing the output of our method with that of applying NLM directly to the noisy image. Left: original noisy image. Middle: NLM result. Right: our result.

Figure 3 shows several example results where we can compare the output of our method with that of applying NLM directly to the noisy image. We can see that, with our approach, homogeneous regions are better restored, while details are preserved as well.

Finally, let us briefly mention that the approach that we have proposed for denoising greyscale images can be extended in a natural way to deal with color images. The moving frame will now be five-dimensional, but otherwise the procedure remains the same: computing the components, denoising them, reconstructing an image from these regularized components. Alternatively, we could just apply the proposed approach to each color channel separately. Figure 4 shows an example result, which again compares favorably with NLM applied directly to the image.

Table 4. Parameter values for each noise level

Parameters	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$
μ	0.001	0.001	0.001	0.001	0.001
$\sigma_1, \sigma_2, \sigma_3$	5, 5, 6	12, 10, 11	15, 15, 16	20, 20, 21	27, 25, 26

5 Conclusions and Future Work

In this paper we have shown that the components of a noisy image in a moving frame are less noisy than the image itself. Therefore, it's more effective, given a denoising method such as Non-local Means, to use it to denoise the components and then reconstruct from them a denoised image result, rather than applying Non-local Means directly to the image. We are currently working on determining the optimum parameter values for the color case, testing whether or not we can improve the results by using a denoised frame for the reconstruction, and also trying our framework on BM3D[8], which is a more recent non-local denoising method.

Acknowledgments. This work was supported by European Research Council, Starting Grant ref. 306337, and by Spanish grants AACC, ref. TIN2011-15954-E, and Plan Nacional, ref. TIN2012-38112. S. Levine acknowledges partial support by NSF-DMS #1320829.

References

1. Batard, T., Bertalmío, M.: On covariant derivatives and their applications to image regularization. Preprint available at `http://hal.archives-ouvertes.fr/hal-00941712`
2. Batard, T., Bertalmío, M.: Generalized gradient on vector bundle–application to image denoising. In: Pack, T. (ed.) SSVM 2013. LNCS, vol. 7893, pp. 12–23. Springer, Heidelberg (2013)

3. Batard, T., Berthier, M.: Spinor Fourier transform for image processing. IEEE Journal of Selected Topics in Signal Processing 7(4), 605–613 (2013); Special Issue on Differential Geometry in Signal Processing
4. Bertalmío, M., Levine, S.: Denoising an Image by Denoising Its Curvature Image. SIAM J. Imaging Sci. 7(1), 187–211 (2014)
5. Buades, A., Coll, B., Morel, J.-M.: A non-local algorithm for image denoising. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 2, pp. 60–65. IEEE (2005)
6. Buades, A., Coll, B., Morel, J.-M.: Non-local Means Denoising. Image Processing On Line (2011)
7. Chatterjee, P., Milanfar, P.: Is denoising dead? IEEE Transactions on Image Processing 19(4), 895–911 (2010)
8. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Transactions on Image Processing 16(8), 2080–2095 (2007)
9. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. on Image Processing 15(12), 3736–3745 (2006)
10. Lebrun, M., Colom, M., Buades, A., Morel, J.M.: Secrets of image denoising cuisine. Acta Numer. 21, 475–576 (2012)
11. Levin, A., Nadler, B.: Natural image denoising: Optimality and inherent bounds. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2833–2840. IEEE (2011)
12. Levin, A., Nadler, B., Durand, F., Freeman, W.T.: Patch complexity, finite pixel correlations and optimal denoising. Technical report, MIT - Computer Science and Artificial Intelligence Laboratory (2012)
13. Lysaker, M., Osher, S., Tai, X.C.: Noise removal using smoothed normals and surface fitting. IEEE Transactions on Image Processing 13(10), 1345–1357 (2004)
14. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 12(7), 629–639 (1990)
15. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1-4), 259–268 (1992)