



Design and Implementation of a Decomposition Framework for Image Denoising Algorithms

Rohit Shrikhande

Electronics Department

Walchand College of Engineering Sangli

Sangli Maharashtra, rshrikhande5@gmail.com

Dr S. D. Ruikar

Electronics Department

Walchand College of Engineering Sangli

Sangli Maharashtra

ruikarsachin@gmail.com

July 6, 2018

Abstract

This paper represents a model for image denoising in spatial domain. It preserves the image parameter such as edges and textures of an image while denoising. The image subtracted to provide gradient norm of denoised image. The model is processed in a moving frame to encode its local geometry. It uses the strategy to denoise the components of the image rather than denoising the image directly. It preserves local geometries at the denoising level. The output perform on the large image database with various algorithms such as VTV, NLM, BM3D and ROR. It is observed that the image denoising method NLM performs better than other algorithms using this model in terms of PSNR and SSIM.

Key Words: moving framework, image decomposition, block matching, differential geometry

1 INTRODUCTION

Denoising is a fundamental task for correcting a noisy image. The patch based methods developed by Rudin et.al, Awate and Whisker had a great impact on denoising methods. It specifically preserves the main features of an image such as edges, textures, colours, contrasts etc. But the problem of these patch based methods is that they do not recover very fine details of an image. In this paper a strategy is developed to denoise the components of an image rather than the denoising image directly.

The image denoising techniques are improved by taking into account of its local geometries (direction of gradient and local lines). The PSNR of curvature lines and unit normal are generally high than the image itself. So it becomes easier to denoise these components. In the moving frame the components of an image are separated and then the denoising algorithms are applied on that components to remove the noise. Further a denoised image is obtained by invertible transform. The results obtained in terms of PSNR and SSIM shows that the moving framework performs better than applying image denoising algorithms directly on an image.

2 Image Decomposition in a Moving Frame

Let $I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ be a grey-level image, and (x, y) be the standard coordinate system of $\mathbb{R}^2$. We denote by I_x resp. I_y the derivative of I with respect to x resp. y , and by ∇I the gradient of I .

A. Methodology

Image decomposition model for image I is two stage approach.

1. Constructing an orthonormal moving frame (Z_1, Z_2, N) over which encodes local geometries of I .
2. Decomposition Framework for algorithms.
3. Computing the components (J_1, J_2, J_3) of $\mathbb{R}^3$ valued function

$(0 \ 0 \ 1)$ in that moving frame.

B. Moving Frame

Considering an orthonormal moving frame as $(Z1, Z2, N)$ where $Z1$ is vector field tangent to the surface S which indicates the direction of steepest slope at each point of S ; $Z2$ is vector field tangent to the surface S which gives direction of the lowest slope at each point of S . The field to be normal we require N to be normal component to the surface S so that frame becomes orthonormal.

Computing the Components

Considering a scaled version μI of I for $\mu \in [0,1]$. The moving frame can be constructed as follows: let $z1 = (\mu I_x, \mu I_y)^T$ be the gradient of μI and $z2 = (-\mu I_x, \mu I_y)^T$ gives the direction of level lines of μI . We define $z1 = (1, 0)^T$ and $z2 = (0, 1)^T$ at pixel locations (x, y) where $I_x(x, y) = I_y(x, y) = 0$ i.e. homogeneous regions of I . Then $Z1$ and $Z2$ can be calculated as $Z_i = \frac{d\psi(z_i)}{\|d\psi(z_i)\|}$ where $i = 1, 2$.

The $d\psi$ is differential of ψ which maps vector fields on Ω to tangent vector field of S . The expression of the unit normal N is then obtained as the vectorial product between $Z1$ and $Z2$. The expressions for the vector fields $Z1, Z2, N$ are given by the matrix field

$$P = \begin{bmatrix} \frac{I_x}{\sqrt{|\nabla I|^2(1+\mu^2|\nabla I|^2)}} & \frac{-I_y}{|\nabla I|} & \frac{-\mu I_x}{\sqrt{1+\mu^2|\nabla I|^2}} \\ \frac{I_y}{\sqrt{|\nabla I|^2(1+\mu^2|\nabla I|^2)}} & \frac{I_x}{|\nabla I|} & \frac{-\mu I_y}{\sqrt{1+\mu^2|\nabla I|^2}} \\ \frac{\mu^2|\nabla I|^2}{\sqrt{|\nabla I|^2(1+\mu^2|\nabla I|^2)}} & 0 & \frac{1}{\sqrt{1+\mu^2|\nabla I|^2}} \end{bmatrix}$$

The first column gives the coordinates of vector field $Z1$, second column $Z2$ and coordinates of N in the third column.

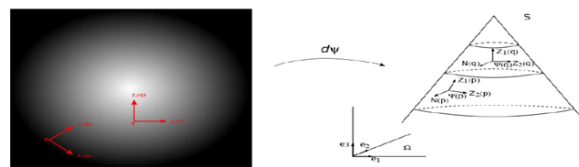


Fig.1. Moving Frame. Left: Original Gray image and moving frame which indicates geometry of an image (direction gradient and level lines) at p q. Right: The orthonormal moving frame (Z1, Z2, N).

The orthonormal frame of $(R, \|\cdot\|_2)$ denoted by (e_1, e_2, e_3) where $e_1 = (1, 0, 0)$, $e_2 = (0, 1, 0)$ and $e_3 = (0, 0, 1)$. The above P matrix we got is nothing but the frame change field from (e_1, e_2, e_3) to (Z_1, Z_2, N) . It means that the components of R_3 valued function $(0, 0, 1)$ in the new frame. These components are denoted by (J_1, J_2, J_3) given by

$$\begin{bmatrix} J_1 \\ J_2 \\ J_3 \end{bmatrix} = P^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

J^1 contains the edges and textures of the image. The component J^3 is similar to the original image but in that gradients norm has been subtracted. The component J^2 identically becomes zero. In this way we can compute the components of a moving frame.

D. Application to Image Denoising

The framework that is proposed is applying denoising techniques to the components of an image which are calculated above taking into considerations of its local geometries instead of applying directly on an image. The procedure is as follows

1. Process the image I with any of the denoising technique and call its output image as I_{den} .
2. For some scalar calculate the components (J^1, J^2, J^3) in the moving frame. Apply the same above denoising technique to the components to get the processed elements such as $(J_d^1 en, J_d^2 en, J_d^3 en)$. After this apply inverse frame change matrix field to the processed components.
3. Compare I_{den} and I_{denMF} with the metrics PSNR and SSIM. The strategy above described can be applied to any moving frame. The

best results are obtained when the components Z_1 and Z_2 are tangent spaces.

3 Framework applied to VTV denoising method

The moving frame approach is applied to the vectorial total variation based denoising method.

It is defined as follows on n channel image

$$I_0 : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^N$$

$$I_{vtv} := \operatorname{argmin} \int \frac{1}{2\lambda} \|I - I_0\|^2 + VTV(I) d\Omega$$

Where the Vectorial Total Variation $VTV(I)$ of I is defined by

$$\sup \left(\int \langle I, \nabla * \eta \rangle^2 d\Omega; \eta \in_c^\infty(\Omega; \mathbb{R}^0), \|\eta(x)\|_2 \leq 1 \forall x \in \Omega \right)$$

For ∇^* being the adjoint Jacobian operator The VTV denoising model can be solved through Chambolles [10] projection algorithm. The moving frame approach to VTV denoising model is as follows: 1. Constructing an orthonormal moving frame (Z_1, Z_2, N) over which encodes local geometries of I . 2. The Gaussian noise of a variance is added to a gray level image to create a noisy image. 3. A denoised image IROF can be obtained by applying Chambolles projection algorithm [10]. 4. For some parameter consider a moving frame for an image I and then calculate its components $(J_1, J_2, J_3)^T$ in that moving frame. Then apply VTV based denoising algorithm to these components and obtain (J^1VTV, J^2VTV, J^3VTV) from which image can be reconstructed as $I_{ROMF} := I_{ROF}^3$. 5. And finally computing PSNR and SSIM index values of I_{ROF} and I_{ROFMF} . Since the ROF model we used in step 2 does not have any parameter to tune whereas in our moving frame we have smoothing parameter μ .

4 Framework applied to NLM denoising method

Applying framework approach to non-local means algorithm defined for multichannel image $I(I_1, \dots, I_n)$, by

$$I^{K NLM}(p) := \frac{-1}{c(p)} \sum_{q \in B(p,r)} w(p,q) I^k(q)$$

Where

$$c(p) = \sum_{q \in B(p,r)} w(p,q)$$

And $B(p,r)$ denotes a neighborhood (patch) of the pixel p of size $(2r+1)(2r+1)$; the weight of the function $w(p,q)$ is of the form

$$w(p,q) = e^{\frac{-\max(d^2 - 2\sigma^2, 0)}{h^2}}$$

where h is a function of the noise level decreasing with the size of the patches and d is the distance between patches. The approach for applying moving frame in the context of NLM algorithm for gray level image is as follows

1. First create noisy image by adding Gaussian noise of variance to clean image a
2. Apply NLM to I and obtain a denoised image $INLM$.
3. Take parameter $\mu=0.001$ and compute the components (J^1, J^2, J^3) using moving frame associated with. Then apply NLM algorithm channel-wise to these components obtaining a denoised components $(J_{NLM}^1, J_{NLM}^2, J_{NLM}^3)$ from which a denoised grey level image is constructed as $I_{NLMMF} = I_{NLMMF}^3$.
4. Finally compute PSNR and SSIM index values of I_{NLM} and I_{NLMMF} .

5 Framework applied to BM3D denoising method

Now the framework is tested on the BM3D algorithm. The approach is exactly followed as it is followed for NLM algorithm; only replacing NLM in steps 2 and 3 by the BM3D algorithm. The BM3D algorithms tested with several parameters determined by noise level σ . Further in step 3 the BM3D applied on the component J^1 with the default parameters determined by the noise level $\sigma_1 = \sigma$. Testing several values of the noise levels σ_3 when applying BM3D on the components J^3 in order to find the ones that provide the best average results on the whole database in terms of PSNR.

6 Framework applied to ROR denoising method

The similarity between pixel and impulse noise is measured using ROR model. Detection and filtering are the two process carried out in an operation of ROR. The ROR value is calculated as

$$ROR(y_{i,j}) = |(y_{i,j} - Med(y))/MADN(y)|$$

ROR of each pixel in an image is calculated as follows. Let $y_{i,j} = (i,j)$ be the location of the pixel under consideration, and let

$$\Omega y(N) = \{y_{i+l,j+1} : -N \leq l, k \leq N\}$$

be the set of points in a $(2N+1) \times (2N+1)$ neighborhood centred at $y_{i,j}$, for some positive integer N .

$Med(\Omega y(N))$ is the median value of all pixels in $\Omega y(N)$, i.e.,
 $Med(\Omega y(N)) = Median(\Omega y(N))$

Then, calculate median absolute deviations $MAD(\Omega y(N))$ and $MADN(\Omega y(N))$ are : $MAD(\Omega y(N)) = Med(\Omega y(N)) - Med(\Omega y(N))$
 $MADN(y(N)) = MAD(y(N))/0.6457$ ROR is calculated and then all the pixels are divided into four clusters according to ROR values.

ROR measures the outlyingness of the pixels i.e. how impulse like pixels is. The four clusters are such as $ROR > 3$: most like

cluster; $2 < ROR \leq 3$: Second like cluster; $1 < ROR \leq 2$: third like cluster;
 $0 \leq ROR \leq 1$: fourth like cluster. The lower the ROR, the lower the impulse the pixel is

Table 1: Comparison between PSNR standard and PSNR moving frame for four algorithms for various noise variance.

Noise Variance	VTV		NLM		BM3D		ROR	
	PSNR standard	PSNR Moving frame	PSNR standard	PSNR Moving frame	PSNR standard	PSNR Moving frame	PSNR standard	PSNR Moving frame
5	38.71	38.86	39.61	39.71	38.70	38.74	37.23	37.35
10	37.62	37.77	37.95	38.16	35.39	35.41	33.94	34.07
15	36.05	36.20	37.03	37.14	33.47	33.53	31.96	32.13
20	35.42	35.72	36.26	36.57	32.07	32.11	30.31	30.49
25	34.21	34.31	35.58	35.68	30.96	30.99	28.90	29.37

Table 2: Comparison between PSNR standard and PSNR moving frame for four algorithms for various noise variance.

Noise Variance	VTV		NLM		BM3D		ROR	
	SSIM standard	SSIM Moving frame	SSIM standard	SSIM Moving frame	SSIM standard	SSIM Moving frame	SSIM standard	SSIM Moving frame
5	0.9852	0.9866	0.9832	0.9482	0.9867	0.9874	0.9821	0.9827
10	0.9687	0.9695	0.9618	0.9622	0.9665	0.9754	0.9732	0.9744
15	0.9482	0.9488	0.9318	0.9322	0.9478	0.9534	0.9543	0.9559
20	0.9226	0.9235	0.8964	0.8971	0.9312	0.9477	0.9215	0.9233
25	0.8953	0.8965	0.8609	0.8615	0.9115	0.9227	0.9086	0.9112

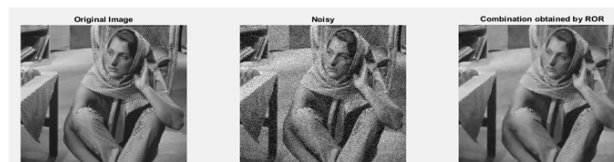


Fig.2. The above fig shows original image the Gaussian noise of variance 25 is added and finally denoised VTV image is recovered using above methodology

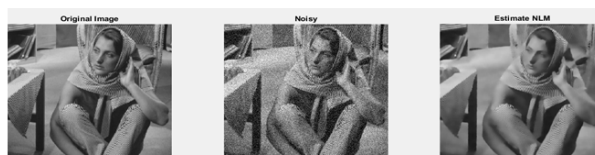


Fig.3. The above fig shows original image the Gaussian noise of variance 25 is added is added and finally denoised NLM image is recovered using above methodology

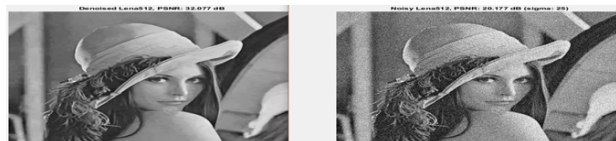


Fig.4. The above fig shows original image the Gaussian noise of variance 25 is added is added and finally denoised BM3D image is recovered using above methodology

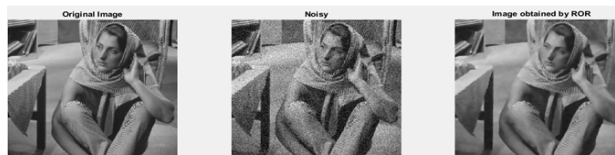


Fig.5. The above fig shows original image the Gaussian noise of variance 25 is added is added and finally denoised ROR image is recovered using above methodology

7 CONCLUSION

In this paper a decomposition framework developed for image denoising which preserves local geometries of an image. The denoising algorithms are applied on the components of an image rather than applying directly on an image. The VTV has a PSNR of 34.31 for noise variance of 25 but cannot distinguish between huge gradients due to noise and huge gradients due to similarities while the NLM has a PSNR of 35.68 for noise variance of 25. The NLM algorithm is better than other algorithms for preserving local geometries while the BM3D algorithm having PSNR of 30.99 for noise variance of 25 removes most of the noise but tends to smooth away the edges. This framework performs better than the algorithms applied directly on an image in terms of PSNR and SSIM metrics.

References

- [1] S. P. Awate and R. T. Whitaker, Higher-order image statistics for unsupervised, information-theoretic, adaptive, image filtering, in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2. Jun. 2005, pp. 4451.
- [2] T. Batard and M. Berthier, Spinor Fourier transform for image processing, IEEE J. Sel. Topics Signal Process., vol. 7, no. 4, pp. 605613, Aug. 2013.
- [3] T. Batard and M. Bertalmo, Generalized gradient on vector bundle Application to image denoising, in Scale Space and Variational Methods in Computer Vision (Lecture Notes in Computer Science), vol. 7893. Berlin, Germany: Springer-Verlag, 2013, pp. 1223.
- [4] T. Batard and M. Bertalmo, On covariant derivatives and their applications to image regularization, SIAM J. Imag. Sci., vol. 7, no. 4, pp. 23932422, 2014.
- [5] M. Bertalmo, Image Processing for Cinema. Boca Raton, FL, USA: CRC Press, 2014.
- [6] M. Bertalmo and S. Levine, Denoising an image by denoising its curvature image, SIAM J. Imag. Sci., vol. 7, no. 2, pp. 187201, 2014.
- [7] P. Blomgren and T. F. Chan, Color TV: Total variation methods for restoration of vector-valued images, IEEE Trans. Image Process, vol. 7, no. 3, pp. 304309, Mar. 1998.
- [8] X. Bresson and T. F. Chan, Fast dual minimization of the vectorial total variation norm and applications to color image processing, Inverse Problems Imag., vol. 2, no. 4, pp. 455484, 2008.
- [9] A. Buades, B. Coll, and J.-M. Morel, Non-local means denoising, Image Process. On Line, vol. 1, Sep. 2011.
- [10] A. Chambolle, An algorithm for total variation minimization and applications, J. Math. Imag. Vis., vol. 20, no. 1, pp. 8997, 2004.

- [11] P. Chatterjee and P. Milanfar, Is denoising dead? *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 895911, Apr. 2010.
- [12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, Image denoising by sparse 3D transform-domain collaborative filtering, *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 20802095, Aug. 2007.
- [13] G. Ghimpeteanu, T. Batard, M. Bertalmo, and S. Levine, Denoising an image by denoising its components in a moving frame, in *Image and Signal Processing (Lecture Notes in Computer Science)*, vol. 8509. Berlin, Germany: Springer-Verlag, 2014, pp. 375383.
- [14] J. Hahn, X.-C. Tai, S. Borok, and A. M. Bruckstein, Orientation matching minimization for image denoising and inpainting, *Int. J. Comput. Vis.*, vol. 92, no. 3, pp. 308324, 2011. Kodak Lossless True Color Image Suite. [Online]. Available: <http://r0k.us/graphics/kodak/>, accessed Oct2013
- [15] M. Lebrun, M. Colom, A. Buades, and J. M. Morel, Secrets of image denoising cuisine, *Acta Numerica*, vol. 21, no. 1, pp. 475576, 2012.
- [16] M. Lebrun, An analysis and implementation of the BM3D image denoising method, *Image Process. On Line*, vol. 2, pp. 175213, Aug. 2012.
- [17] M. Lebrun, M. Colom, and J. M. Morel, The noise clinic: A universal blind denoising algorithm, in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 26742678.
- [18] A. Levin and B. Nadler, Natural image denoising: Optimality and inherent bounds, in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun. 2011, pp. 28332840.
- [19] M. Lysaker, S. Osher, and X.-C. Tai, Noise removal using smoothed normals and surface fitting, *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 13451357, Oct. 2004.
- [20] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, An iterative regularization method for total variation-based image

restoration, Multiscale Model. Simul., vol. 4, no. 2, pp. 460489, 2005.

