

LA-UR- 09-01037

Approved for public release;
distribution is unlimited.

Title: A GENERALIZED VECTOR - VALUED TOTAL VARIATION
ALGORITHM

Author(s): PAUL RODRIGUEZ
BRENDT WOHLBERG

Intended for: PROCEEDINGS:
IEEE INTERNATIONAL CONFERENCE ON IMAGE
PROCESSING



Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

A GENERALIZED VECTOR-VALUED TOTAL VARIATION ALGORITHM

Paul Rodríguez

Digital Signal Processing Group
Pontificia Universidad Católica del Perú
Lima, Peru

Brendt Wohlberg*

T-5 Applied Mathematics and Plasma Physics
Los Alamos National Laboratory
Los Alamos, NM 87545, USA

ABSTRACT

We propose a simple but flexible method for solving the generalized vector-valued TV (VTV) functional, which includes both the ℓ^2 -VTV and ℓ^1 -VTV regularizations as special cases, to address the problems of deconvolution and denoising of vector-valued (e.g. color) images with Gaussian or salt-and-pepper noise. This algorithm is the vectorial extension of the Iteratively Reweighted Norm (IRN) algorithm [1] originally developed for scalar (grayscale) images. This method offers competitive computational performance for denoising and deconvolving vector-valued images corrupted with Gaussian (ℓ^2 -VTV case) and salt-and-pepper noise (ℓ^1 -VTV case).

Index Terms— Vector-valued Total Variation, Color image processing

1. INTRODUCTION

The development of numerical algorithms for vector-valued regularization has recently attracted considerable interest [2, 3, 4, 5, 6, 7, 8]. In particular the Total Variation (TV) minimization scheme for deblurring color images, first introduced in [9], is of special interest. Most of the publications have focused on the Gaussian noise model, and to the best of our knowledge, [5] is the only published paper to explicitly consider the salt-and-pepper noise model for color images in a variational framework.

The ℓ^2 vector-valued TV (VTV) regularized solution (with coupled-channel regularization [10]) of the inverse problem involving color image data \mathbf{b} and forward linear operator A is the minimum of the functional

$$T(\mathbf{u}) = \frac{1}{p} \left\| A\mathbf{u} - \mathbf{b} \right\|_p^p + \frac{\lambda}{q} \left\| \sqrt{\sum_{n \in C} (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2} \right\|_q^q, \quad (1)$$

for $p = 2, q = 1, n \in C = \{r, g, b\}$ (note that C could represent an arbitrary number of channels) and notation as follows:

*This research was supported by the NNSA's Laboratory Directed Research and Development Program.

- \mathbf{u}_n ($n \in C$) is a 1-dimensional (column) or 1D vector that represents a 2D grayscale image obtained via any ordering (although the most reasonable choices are row-major or column-major) of the image pixels.
- $\mathbf{u} = [(\mathbf{u}_r)^T (\mathbf{u}_g)^T (\mathbf{u}_b)^T]^T$ is a 1D (column) vector that represents a 2D color image.
- $\frac{1}{p} \|A\mathbf{u} - \mathbf{b}\|_p^p$ is the data fidelity term. For the scope of this paper, the linear operator A is assumed to be decoupled, i.e. A is a diagonal block matrix with elements A_n and $n \in C = \{r, g, b\}$,
- $\frac{1}{q} \left\| \sqrt{\sum_{n \in C} (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2} \right\|_q^q$ is the generalization of TV regularization to color images with coupled channels (see [10, Section 9], also used in [5, 7, 8]),
- the p -norm of vector \mathbf{u} is denoted by $\|\mathbf{u}\|_p$,
- scalar operations applied to a vector are considered to be applied element-wise, so that, for example, $\mathbf{u} = \mathbf{v}^2 \Rightarrow u[k] = (v[k])^2$ and $\mathbf{u} = \sqrt{\mathbf{v}} \Rightarrow u[k] = \sqrt{v[k]}$,
- $\sqrt{\sum_{n \in C} (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2}$ is the discretization of $|\nabla \mathbf{u}|$ for coupled channels (see [7, eq. (3)]), and
- horizontal and vertical discrete derivative operators are denoted by D_x and D_y respectively.

Choosing $p = 1$ (ℓ^1 norm for the fidelity term), $q = 1$ in (1) leads to ℓ^1 -VTV, which can be used to remove salt-and-pepper noise in color images. We also note that if set C has only one element (grayscale image) then (1) represents the scalar TV functional.

In this paper we present an efficient algorithm to minimize the generalized vector-valued TV functional (1) for the cases of denoising ($A = I$ in (1)) and decoupled linear operator A . This algorithm, which is a computationally efficient and flexible alternative to the extension of [11] described in [5], can handle any norm with $0 < p, q \leq 2$ (including the ℓ^2 -VTV and ℓ^1 -VTV as special cases) by representing the ℓ^p and ℓ^q norms by the equivalent weighted ℓ^2 norms.

2. THE VECTOR-VALUED ITERATIVELY REWEIGHTED NORM APPROACH

2.1. Previous related work

The vector-valued IRN approach is an extension of the IRN algorithm [1], and is closely related to Iteratively Reweighted Least Squares (IRLS) method for scalar [12] and vector [13] valued problems.

2.2. Derivation

In order to replace the ℓ^p norm of the fidelity term in (1) by a weighted ℓ^2 norm we define the quadratic functional

$$Q_F^{(k)}(\mathbf{u}) = \frac{1}{2} \left\| W_F^{(k)1/2} (A\mathbf{u} - \mathbf{b}) \right\|_2^2 + \left(1 - \frac{p}{2}\right) F(\mathbf{u}^{(k)}), \quad (2)$$

where $\mathbf{u}^{(k)}$ is a constant representing the solution of the previous iteration, $F(\mathbf{u}) = \frac{1}{p} \|A\mathbf{u} - \mathbf{b}\|_p^p$, and

$$W_F^{(k)} = \text{diag} \left(\tau_{F, \epsilon_F} (A\mathbf{u}^{(k)} - \mathbf{b}) \right). \quad (3)$$

The function

$$\tau_{F, \epsilon_F}(x) = \begin{cases} |x|^{p-2} & \text{if } |x| > \epsilon_F \\ \epsilon_F^{p-2} & \text{if } |x| \leq \epsilon_F, \end{cases}$$

is defined to avoid numerical problems when $p < 2$ and $A\mathbf{u}^{(k)} - \mathbf{b}$ has zero-valued components.

The replacement of the ℓ^q norm of the regularization term

$$R(\mathbf{u}) = \|\mathbf{r}\|_q^q, \quad \mathbf{r} = \sqrt{\sum_{n \in \{r, g, b\}} (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2}$$

in (1) by a weighted ℓ^2 norm is not as straightforward. We use the notation $(\mathbf{v}_n)^2 = (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2$, define $\Phi_n = \text{diag} \left(\frac{1}{\mathbf{v}_n} \right) = \text{diag} \left(\frac{1}{\sqrt{(D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2}} \right)$, and note that $\|\mathbf{v}_n\|_1 = \left\| \begin{bmatrix} \Phi_n^{0.5} & 0 \\ 0 & \Phi_n^{0.5} \end{bmatrix} \begin{bmatrix} D_x \mathbf{u}_n \\ D_y \mathbf{u}_n \end{bmatrix} \right\|_2$, neglecting possible divisions by zero. Furthermore, we may write $\|\mathbf{v}_n\|_1 = \left\| \begin{bmatrix} \Phi_n^{0.5} D_x \mathbf{u}_n \\ \Phi_n^{0.5} D_y \mathbf{u}_n \end{bmatrix} \right\|_2 = \|\Phi_n^{0.5} D_x \mathbf{u}_n\|_2 + \|\Phi_n^{0.5} D_y \mathbf{u}_n\|_2$. This is the key idea in [1] for solving the generalized TV for grayscale images.

For the present case of vector-valued images we first define $\Phi = \text{diag} \left(\frac{1}{\sqrt{(\mathbf{v}_r)^2 + (\mathbf{v}_g)^2 + (\mathbf{v}_b)^2}} \right)$, then the regularization term with $q = 1$ can be expressed as

$$R(\mathbf{u}) = \|\mathbf{r}\|_1 = \left\| [I_3 \otimes \Phi^{0.5}] [\mathbf{v}_r^T \mathbf{v}_g^T \mathbf{v}_b^T]^T \right\|_2, \quad (4)$$

where I_N is a $N \times N$ identity matrix and \otimes is the Kronecker product. Note that only Φ would change for $q \neq 1$ and therefore (4) is valid for the general case. Defining $\Phi[k, k] = \phi[k]$,

it is easy to check that $\|\Phi^{0.5} \mathbf{v}_n\|_2^2 = \sum_k \phi[k] (\mathbf{v}_n[k])^2 = \sum_k \phi[k] (D_x \mathbf{u}_n[k])^2 + \sum_k \phi[k] (D_y \mathbf{u}_n[k])^2$.

Using the previous equations we may write $\|\Phi^{0.5} \mathbf{v}_n\|_2^2 = \|\Phi^{0.5} D_x \mathbf{u}_n\|_2^2 + \|\Phi^{0.5} D_y \mathbf{u}_n\|_2^2$ and thus

$$R(\mathbf{u}) = \|W_R^{0.5} D\mathbf{u}\|_2^2, \quad (5)$$

where $W_R = I_6 \otimes \Phi$ and $D = I_3 \otimes [D_x^T D_y^T]^T$. It is important to emphasize that (5) is not the anisotropic separable approximation.

Now, we may replace the ℓ^q norm of the regularization term in (1) by a weighted ℓ^2 norm by means of

$$Q_R^{(k)}(\mathbf{u}) = \frac{1}{2} \left\| W_R^{(k)1/2} D\mathbf{u} \right\|_2^2 + \left(1 - \frac{q}{2}\right) R(\mathbf{u}^{(k)}), \quad (6)$$

where $\mathbf{u}^{(k)}$ is a constant representing the solution of the previous iteration. As in the case of the data fidelity term, care needs to be taken when $q < 2$ and $\mathbf{v}_n^2 = (D_x \mathbf{u}_n)^2 + (D_y \mathbf{u}_n)^2$ has zero-valued components. We therefore define

$$\tau_{R, \epsilon_R}(x) = \begin{cases} |x|^{(q-2)/2} & \text{if } |x| > \epsilon_R \\ \epsilon_R^{(q-2)/2} & \text{if } |x| \leq \epsilon_R, \end{cases} \quad (7)$$

and set $\Phi^{(k)} = \text{diag} \left(\tau_{R, \epsilon_R} \left((\mathbf{v}_r^{(k)})^2 + (\mathbf{v}_g^{(k)})^2 + (\mathbf{v}_b^{(k)})^2 \right) \right)$. The threshold values ϵ_F and ϵ_R may be automatically adapted to the input image; for details see Section IV.G in [1].

2.3. Algorithm

The vector-valued IRN algorithm for a general case is summarized in Algorithm 1. As for scalar IRN [1] there is a significant variant for the denoising-only case, which delivers both improved time and SNR performance than the general case, furthermore it has a better SNR / accuracy ratio (details [1, Sec. IV.E] are omitted here due to space constraints). While not done so here, this algorithm can easily be extended to handle a coupled forward operator A in (1).

3. RESULTS

We compare the performance of the vector-valued IRN algorithm for ℓ^1 and ℓ^2 VTV denoising and deconvolution with that of three alternative variational approaches, which we refer to as ℓ^1/ℓ^2 -MS¹, ℓ^1/ℓ^2 -VLD², and FDM³. The test color images are the ‘‘Lena’’ (256 \times 256 pixel), ‘‘Peppers’’ and ‘‘Mandrill’’ (both 512 \times 512 pixel) images (see Fig. 1). All simulations have been carried out using Matlab-only code on a 1.83GHz Intel Dual core CPU. Results corresponding

¹An approximation of the Mumford-Shah functional, see (9) in [5].

²Vectorial lagged diffusivity, an extension of [11] used in [5] for VTV.

³[14], an implementation of the fast dual minimization of VTV [7].

Initialize

$$\mathbf{u}^{(0)} = (A^T A + \lambda D^T D)^{-1} A^T \mathbf{b}$$

for $k = 0, 1, \dots$

$$W_F^{(k)} = \text{diag} \left(\tau_{F, \epsilon_F} (A \mathbf{u}^{(k)} - \mathbf{b}) \right)$$

$$\Phi^{(k)} = \text{diag} \left(\tau_{R, \epsilon_R} \left((\mathbf{v}_r^{(k)})^2 + (\mathbf{v}_b^{(k)})^2 + (\mathbf{v}_g^{(k)})^2 \right) \right)$$

$$W_R^{(k)} = I_6 \otimes \Phi^{(k)}$$

$$\mathbf{u}^{(k+1)} = \left(A^T W_F^{(k)} A + \lambda D^T W_R^{(k)} D \right)^{-1} A^T W_F^{(k)} \mathbf{b}$$

end

Algorithm 1: Vector-valued IRN algorithm.

to the vector-valued IRN algorithm presented here may be reproduced using the the NUMIPAD (v. 0.22) distribution [15], an implementation of IRN and related algorithms.

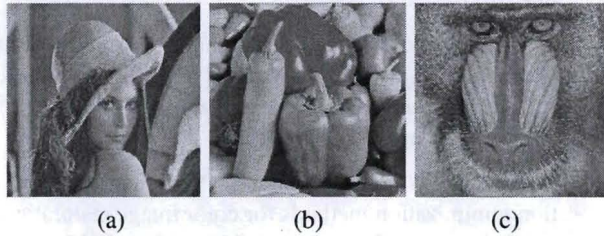


Fig. 1. Test color images: (a) "Lena" (256 × 256 pixel), (b) "Peppers" and (c) "Mandrill" (both 512 × 512 pixel).

Image	Noise	SNR (dB)			Time (s)		
		ℓ^1 -MS	ℓ^1 -VLD	IRN	ℓ^1 -MS	ℓ^1 -VLD	IRN
Lena	10%	11.1	11.2	21.2	376	92	86
	30%	11.1	11.0	19.0	375	91	70

Table 1. Deconvolution performance comparison between ℓ^1 -MS and ℓ^1 -VLD [5] methods and vector-valued IRN algorithm (ℓ^1 -VTV case), on the "Lena" test color image.

The "Lena" image was used for the deconvolution case (to match one of the experiments described in [5]) and was blurred by 7×7 out-of-focus kernel (generated by the Matlab command `fspecial('disk', 3, 2)`) and then corrupted with Gaussian additive noise or salt-and-pepper noise. Reconstruction SNR values and computation times are compared in Table 1 (deconvolution with the salt-and-pepper noise model) and Table 2 (deconvolution with the Gaussian noise model), and noisy and reconstructed images are displayed in Figs. 2(a) to 2(f). The vector-valued IRN has a better computational performance and gives significantly better results, both in terms of SNR and visual quality than the ℓ^1/ℓ^2 -MS or ℓ^1/ℓ^2 -VLD methods described in [5]. (Despite using the parameters values in [5, Table IV], we obtained different per-

Image	Noise (σ^2)	SNR (dB)			Time (s)		
		ℓ^2 -MS	ℓ^2 -VLD	IRN	ℓ^2 -MS	ℓ^2 -VLD	IRN
Lena	1.0e-5	10.0	10.0	18.4	270	89	35
	1.0e-4	6.7	7.9	17.1	363	88	35
	2.5e-3	1.3	7.1	14.5	367	88	27

Table 2. Deconvolution performance comparison between ℓ^2 -MS and ℓ^2 -VLD [5] methods and vector-valued IRN algorithm (ℓ^2 -VTV case), on the "Lena" test color image.

mance results, which we assume is due to differences in other parameters in the code provided to us.)

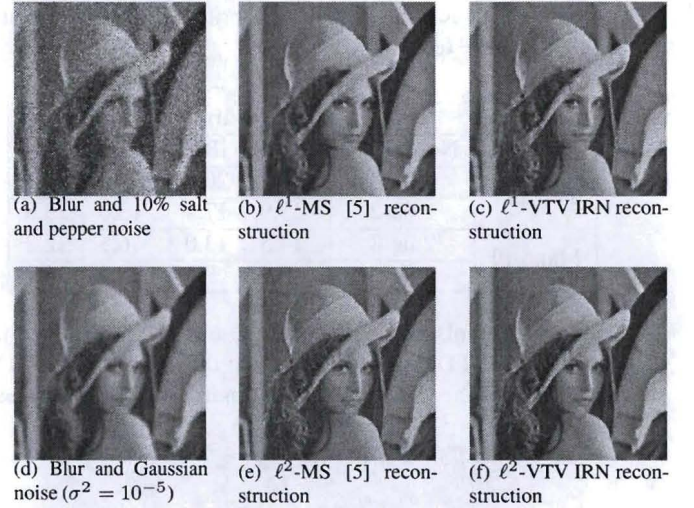


Fig. 2. Deconvolution with 10% salt and pepper noise and with Gaussian noise ($\sigma^2 = 10^{-5}$) via [5] and VTV-IRN.

The "Peppers" and "Mandrill" images were used for the denoising only case and were corrupted with Gaussian additive noise or salt-and-pepper noise. Reconstruction SNR values and computation times are shown in Table 3 and noisy and reconstructed images are displayed in Figs. 3(a) to 3(d) for ℓ^1 -VTV denoising, computed via the vector-valued IRN algorithm. Table 4 presents the results of comparing the vector-valued IRN and FDM for the ℓ^2 -VTV denoising case, where the latter has better computational performance, but SNR values and visual quality are about the same for both methods.

4. CONCLUSIONS

The vector-valued IRN algorithm gives very good reconstruction quality for the ℓ^2 and ℓ^1 -VTV deconvolution/denoising problems, with a superior computational performance than the only other published ℓ^1 -VTV algorithm of which we are aware (the vectorial extension of [11] used in [5]). The FDM method [7, 14] for ℓ^2 -VTV denoising has a better time performance, but it can not handle the ℓ^1 -VTV case or a non-trivial forward operator A .

5. ACKNOWLEDGMENT

The authors thank Leah Bar for providing us with the Matlab code implementing the methods described in [5].

Image	Noise	SNR (dB)		Time (s)	
		IRN		IRN	
Peppers	10%	19.7		73.7	
	30%	15.9		83.8	
Mandrill	10%	10.4		73.3	
	30%	8.8		81.6	

Table 3. Denoising performance results for ℓ^1 -VTV, computed via the vector-valued IRN algorithm, on the “Peppers” and “Mandrill” test color images.

Image	Noise (σ^2)	SNR (dB)		Time (s)	
		FDM	IRN	FDM	IRN
Peppers	2.5e-3	20.7	20.2	7.4	22.3
	1.0e-2	17.9	17.4	13.0	23.3
Mandrill	2.5e-3	14.5	13.0	6.5	22.2
	1.0e-2	9.9	9.4	8.4	26.7

Table 4. Denoising performance results for ℓ^2 -VTV, computed via the FDM [14] method and the vector-valued IRN algorithm, on the “Peppers” and “Mandrill” test color images.

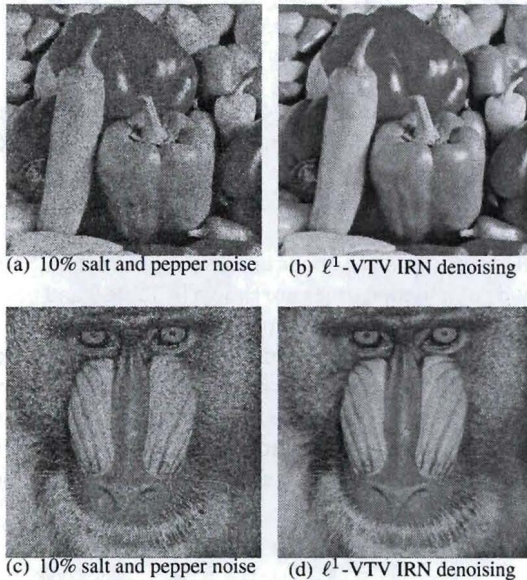


Fig. 3. Denoising with 10% salt and pepper noise.

6. REFERENCES

- [1] P. Rodríguez and B. Wohlberg, “Efficient minimization method for a generalized total variation functional,” *IEEE TIP*, vol. 18, no. 2, pp. 322–332, 2009.
- [2] T. Chan, S. Kang, and J. Shen, “Total variation denoising and enhancement of color images based on the CB and HSV color models,” *J. Visual Comm. Image Rep.*, vol. 12, pp. 422–435, 2001.
- [3] J. Weickert and T. Brox, “Diffusion and regularization of vector and matrix-valued images,” Tech. Rep., Universität des Saarlandes, 2002.
- [4] D. Tschumperle and R. Deriche, “Vector-valued image regularization with pdes: A common framework for different applications,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 506–517, 2005.
- [5] L. Bar, A. Brook, N. Sochen, and N. Kiryati, “Deblurring of color images corrupted by impulsive noise,” *IEEE TIP*, vol. 16, pp. 1101–1111, 2007.
- [6] O. Christiansen, T. Lee, J. Lie, U. Sinha, and T. Chan, “Total variation regularization of matrix-valued images,” in *Int. J. Biomed Imaging*, 2007, doi:10.1155/2007/27432.
- [7] X. Bresson and T. Chan, “Fast dual minimization of the vectorial total variation norm and applications to color image processing,” *J. of Inverse Problems and Imaging*, vol. 2, no. 4, pp. 455–484, 2008.
- [8] Y. Wen, M. Ng, and Y. Huang, “Efficient total variation minimization methods for color image restoration,” *IEEE TIP*, vol. 17, no. 11, pp. 2081–2088, 2008.
- [9] P. Blomgren and T. Chan, “Color TV: Total variation methods for restoration of vector valued images,” *IEEE TIP*, vol. 7, no. 3, pp. 304–309, 1998.
- [10] A. Bonnet, “On the regularity of edges in image segmentation,” *Annales de l’institut Henri Poincaré (C) Analyse non linéaire*, vol. 13, no. 4, pp. 485–528, 1996.
- [11] C. Vogel and M. Oman, “Iterative methods for total variation denoising,” *SIAM J. on Sci. Comp.*, vol. 17, no. 1, pp. 227–238, 1996.
- [12] A. Beaton and J. Tukey, “The fitting of power series, meaning polynomials illustrated on band-spectroscopic data,” *Technometrics*, no. 16, pp. 147–185, 1974.
- [13] N. Galatsanos, A. Katsaggelos, T. Chin, and A. Hillery, “Least squares restoration of multichannel images,” *IEEE TSP*, vol. 39, no. 10, pp. 2222–2236, 1991.
- [14] P. Getreuer, “Total variation grayscale and color image denoising,” Matlab central file <http://www.mathworks.com/matlabcentral/fileexchange/16236>.
- [15] P. Rodríguez and B. Wohlberg, “Numerical methods for inverse problems and adaptive decomposition (NUMIPAD),” <http://numipad.sourceforge.net/>.