

# A New Algorithm for Compressive Sensing Based on Total-Variation Norm

Jeevan K. Pant, Wu-Sheng Lu, and Andreas Antoniou  
 Department of Electrical and Computer Engineering, University of Victoria  
 P. O. Box 3055 STN CSC, Victoria, B.C. V8W 3P6, CANADA  
 Email: {jkpant, wslu}@ece.uvic.ca, aantoniou@ieee.org

**Abstract**—A new algorithm for the reconstruction of images with sparse gradient is proposed. The algorithm is based on the minimization of the so called *total-variation (TV)* regularized squared error and is especially suited for image reconstruction from a small number of measurements. The algorithm is developed based on a generalized *TV* norm and uses a sequential conjugate-gradient method. Simulation results are presented which demonstrate that the proposed algorithm yields significantly improved reconstruction performance for images with sparse gradient and requires significantly reduced computational effort relative to the log-barrier based *TV*-regularized least-squares algorithm.

## I. INTRODUCTION

Compressive sensing (CS) is an effective signal acquisition framework for sparse signals [1]- [3]. Sparse signal recovery is a problem associated with CS. State-of-the-art signal-recovery algorithms include  $\ell_1$ -minimization [4],  $\ell_p$ -minimization [5] [6], greedy pursuit based [7], and total-variation (TV) based algorithms. The concept of total variation (*TV*) was introduced by Rudin, Osher, and Fatemi [8] for image denoising. Many algorithms have been developed for solving *TV*-regularized least-squares (*TV*-RLS) problems [9] [10].

In this paper, a new algorithm, namely, the  $TV_p$ -regularized least-squares ( $TV_p$ -RLS) algorithm, is proposed. This is based on the minimization of a nonconvex *TV*-norm regularized squared error. A sequential conjugate-gradient (CG) method is applied for the optimization. The  $TV_p$ -RLS algorithm offers improved image reconstruction performance relative to the *TV*-minimization based algorithm for images which have a sparse gradient.

## II. BACKGROUND AND PREVIOUS WORK

A real-valued discrete-time signal  $\mathbf{x}$  of length  $N$  is said to be  $K$  sparse if it has  $K$  nonzero components typically with  $K \ll N$ . Signal acquisition in compressive sensing can be characterized by

$$\mathbf{y} = \Phi \mathbf{x}$$

where  $\mathbf{y}$  is a measurement vector of size  $M$  and  $\Phi$  is a measurement matrix of size  $M \times N$ , typically with  $M \ll N$ .

An  $\ell_1$ -minimization based algorithm which solves the optimization problem

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{x}\|_1 \\ & \text{subject to:} && \mathbf{y} = \Phi \mathbf{x} \end{aligned} \quad (1)$$

can be used to accurately reconstruct  $\mathbf{x}$  from  $\mathbf{y}$  provided that  $\mathbf{x}$  is  $K$ -sparse, the components of  $\Phi$  are independently identically distributed Gaussian random variables with zero mean and variance  $1/N$  and the number of measurements  $M$  satisfies the inequality [1]- [3]

$$M \geq c \cdot K \cdot \log(N/K).$$

Recently,  $\ell_p$ -minimization based algorithms with  $p < 1$ , which solve the problem

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{x}\|_p^p \\ & \text{subject to:} && \mathbf{y} = \Phi \mathbf{x} \end{aligned} \quad (2)$$

where  $\|\mathbf{x}\|_p^p = \sum_{i=1}^N |x_i|^p$ , have been proposed by several authors [5] [6]. These algorithms have been shown to offer improved signal reconstruction performance relative to techniques based on solving the problem in (1) [5] [6].

For image recovery, *TV*-norm minimization is preferred over  $\ell_1$ -norm minimization. Let  $\mathbf{X} = \{x_{ij}\}$  be a matrix of size  $n_1 \times n_2$  whose components represent the pixels of a grey-scale digital image. The *TV* norm of  $\mathbf{X}$  is defined as

$$\begin{aligned} TV(\mathbf{X}) = & \sum_{i=1}^{n_1-1} \sum_{j=1}^{n_2-1} \sqrt{x_{i,j}^{i2} + x_{i,j}^{j2}} + \sum_{i=1}^{n_1-1} |x_{i,n_2}^i| \\ & + \sum_{j=1}^{n_2-1} |x_{n_1,j}^j| \end{aligned} \quad (3)$$

where  $x_{i,j}^{i2} = x_{i,j} - x_{i+1,j}$  and  $x_{i,j}^{j2} = x_{i,j} - x_{i,j+1}$ .  
 Matrix

$$\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_{n_2}]$$

can be used to construct an image vector  $\mathbf{x}$  as

$$\mathbf{x} = [\mathbf{x}_1^T \ \mathbf{x}_2^T \ \cdots \ \mathbf{x}_{n_2}^T]^T.$$

A noisy measurement  $\mathbf{y}$  of an image represented by matrix  $\mathbf{X}$  can be represented by

$$\mathbf{y} = \Phi \mathbf{x} + \mathbf{w}$$

where  $\mathbf{w}$  is a noise vector and matrix  $\Phi$  is often constructed by selecting its rows uniformly at random from a matrix representing a standard transform such as the Fourier, discrete

cosine, or a wavelet transform. Matrix  $\mathbf{X}$  can be reconstructed from noisy measurement  $\mathbf{y}$  by solving the  $TV$ -RLS problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad \frac{1}{2} \|\Phi \mathbf{x} - \mathbf{y}\|_2^2 + \lambda TV(\mathbf{X}) \quad (4)$$

where  $\lambda$  is a regularization parameter [14].

### III. $TV_p$ -REGULARIZED LEAST-SQUARES ALGORITHM

#### A. Generalized $TV$ norm and Problem formulation

Inspired by the effectiveness of  $\ell_p$ -norm minimization over  $\ell_1$ -norm minimization for the reconstruction of sparse signals, we consider the approximate  $TV$  norm given by

$$\begin{aligned} TV_{p,\epsilon}^p(\mathbf{X}) &= \sum_{i=1}^{n_1-1} \sum_{j=1}^{n_2-1} \left[ x_{i,j}^{i2} + x_{i,j}^{j2} + \epsilon^2 \right]^{p/2} \\ &+ \sum_{i=1}^{n_1-1} \left[ x_{i,n_2}^{i2} + \epsilon^2 \right]^{p/2} \\ &+ \sum_{j=1}^{n_2-1} \left[ x_{n_1,j}^{j2} + \epsilon^2 \right]^{p/2} \end{aligned} \quad (5)$$

where  $0 < p \leq 1$  and  $\epsilon$  is a nonzero parameter used to render  $TV_{p,\epsilon}^p(\mathbf{X})$  differentiable. Using (5), we consider the  $TV_p$  regularized least-squares ( $TV_p$ -RLS) problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad F_{\lambda,p,\epsilon}(\mathbf{x}) = \frac{1}{2} \|\Phi \mathbf{x} - \mathbf{y}\|_2^2 + \lambda TV_{p,\epsilon}^p(\mathbf{X}). \quad (6)$$

Two remarks about the problem in (6) are as follows. First, function  $TV_{p,\epsilon}^p(\mathbf{X})$ , and as a consequence function  $F_{\lambda,p,\epsilon}(\mathbf{x})$ , is nonconvex for  $p < 1$  and an ordinary gradient descent based optimization strategy tends to converge to a suboptimal local minimum. Second, for  $\epsilon = 0$ , function  $F_{\lambda,p,\epsilon}(\mathbf{x})$  is nonsmooth and an  $\epsilon > 0$  smoothens the function  $F_{\lambda,p,\epsilon}(\mathbf{x})$ . In effect, the larger the  $\epsilon$ , the smoother the  $F_{\lambda,p,\epsilon}(\mathbf{x})$  and the easier is the optimization of  $F_{\lambda,p,\epsilon}(\mathbf{x})$ . However, a solution of the problem in (6) is accurate only for a sufficiently small value of  $\epsilon$ .

#### B. Algorithm based on $TV_p$ norm

The problem in (6) can be solved by using a sequential strategy whereby the problem is solved for a sequence of decreasing values of  $\epsilon$  with  $\lambda$  fixed as was done in [5]. Based on extensive simulations, a better solution can be obtained by solving the problem in (6) by decreasing the values of  $\lambda$  and  $\epsilon$  simultaneously as the sequential optimization proceeds.

On the basis of the above underlying principles, we propose a CG based algorithm tailored for the problem in (6). Given a pair of small target values  $(\epsilon_T, \lambda_T)$  and a pair of large initial values  $(\epsilon_1, \lambda_1)$  of parameters  $\epsilon$  and  $\lambda$ , we generate monotonically decreasing sequences  $\{\epsilon_t, t = 1, 2, \dots, T\}$  and  $\{\lambda_t : t = 1, 2, \dots, T\}$  as

$$\epsilon_t = \epsilon_1 e^{-\beta(t-1)} \quad \text{for } t = 1, 2, \dots, T \quad (7)$$

$$\lambda_t = \lambda_1 e^{-\gamma(t-1)} \quad \text{for } t = 1, 2, \dots, T \quad (8)$$

where

$$\beta = \log(\epsilon_1/\epsilon_T)/(T-1)$$

$$\gamma = \log(\lambda_1/\lambda_T)/(T-1).$$

The problem in (6) is then solved sequentially for  $(\epsilon_t, \lambda_t)$  with  $t = 1, 2, \dots, T$ . For each value of  $t$ , the problem is solved using a finite number of iterations, say,  $L_t$ , of the CG algorithm. Starting with sufficiently large values of  $\epsilon_1$  and  $\lambda_1$  and applying the CG algorithm with a sufficiently large number of iterations,  $L_1$ , a point is obtained, which would be a good initializer for the optimization of function  $F_{\lambda_2,p,\epsilon_2}(\mathbf{x})$ . By using this point as the initial point for the next round of  $L_2$  CG iterations to minimize  $F_{\lambda_2,p,\epsilon_2}(\mathbf{x})$ , an improved solution is generated. This procedure is continued until function  $f_{\lambda_T,p,\epsilon_T}(\mathbf{x})$  is minimized using  $L_T$  CG iterations.

A distinct feature of the problem in (6) is that the size of  $\mathbf{x}$ , namely,  $n_1 \times n_2$ , is very large even for images of moderate sizes. Therefore, CG based algorithms that involve the evaluation of the Hessian matrix of  $F_{\lambda,p,\epsilon}(\mathbf{x})$  are impractical. The Fletcher-Reeves' CG method [12] does not use the Hessian of  $F_{\lambda,p,\epsilon}(\mathbf{x})$  and it is, therefore, ideal for the application under consideration.

In the proposed  $TV_p$ -RLS algorithm, iterate  $\mathbf{x}_k$  is updated to  $\mathbf{x}_{k+1}$  using the CG update formula [12]

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad (9a)$$

where

$$\mathbf{d}_k = -\mathbf{g}_k + \beta_{k-1} \mathbf{d}_{k-1} \quad (9b)$$

$$\beta_{k-1} = \frac{\|\mathbf{g}_k\|_2^2}{\|\mathbf{g}_{k-1}\|_2^2}. \quad (9c)$$

In (9b) and (9c),  $\mathbf{g}_k$  is the gradient of  $F_{\lambda,p,\epsilon}(\mathbf{x})$  at  $\mathbf{x} = \mathbf{x}_k$  which assumes the form

$$\mathbf{g} = \Phi^T (\Phi \mathbf{x} - \mathbf{y}) + \lambda p \mathbf{u} \quad (10)$$

where  $\mathbf{u}$  is a vector of size  $n_1 \times n_2$ . It can be computed as shown in Appendix A.

The step size,  $\alpha_k$ , in (9a) is determined by using a line search that solves the one-dimensional minimization problem

$$\underset{\alpha}{\text{minimize}} \quad F_{\lambda,p,\epsilon}(\mathbf{x}_k + \alpha \mathbf{d}_k). \quad (11)$$

The solution of the above problem is obtained by using a line-search constructed on the basis of Banach's fixed-point theorem [13] which solves the equation

$$\alpha = G(\alpha) \quad (12)$$

for scalar variable  $\alpha$  by using the recursion  $\alpha_{l+1} = G(\alpha_l)$  for  $l = 1, 2, \dots$ . An explicit expression for  $G(\alpha)$  is given in Appendix B, and a description of such a line-search can be found in [5].

Based on the preceding principles, the  $TV_p$ -RLS algorithm summarized in Table I can be constructed.

### IV. SIMULATION RESULTS

In Experiment 1, an image  $\mathbf{X}$  was vectorized into a vector  $\mathbf{x}$  of size  $N = n_1 \times n_2$ . Then  $M_1 \approx 0.016N$  discrete-cosine transform (DCT) measurements and  $M_2 \approx 0.31N$  real-valued dragon noiselet measurements of  $\mathbf{x}$  were taken as was done by Romberg for the CS of the image in [14]. The dragon

TABLE I  
TV<sub>p</sub>-RLS ALGORITHM

---

**Step 1**  
Input:  $T, p, \epsilon_1, \epsilon_T, \lambda_1, \lambda_T, \Phi, \mathbf{y}, E_{th}, L_t$ .  
Set  $\mathbf{x}_s = \mathbf{0}$ .

**Step 2**  
Compute  $\epsilon_t$  for  $t = 2, 3, \dots, T - 1$  using (7) and  $\lambda_t$  for  $t = 2, 3, \dots, T - 1$  using (8).

**Step 3**  
Repeat the following for  $t = 1, \dots, T$   
i) Set  $\epsilon = \epsilon_t, \lambda = \lambda_t$ .  
ii) Set  $k = 0, \mathbf{x}_0 = \mathbf{x}_s, E_r = 10^{10}$ .  
iii) Repeat the following while  $E_r > E_{th}$ ,  
a) Compute  $\mathbf{x}_{k+1}$  using (9a), (9b), (10), (9c), (12).  
b) Set  $k = k + 1$ .  
c) Exit loop if  $k > L_t$ .  
d) Compute  $E_r = \|\alpha_k \mathbf{d}_k\|_2$ .  
iv) Set  $\mathbf{x}_s = \mathbf{x}_k$ .

**Step 4**  
Output  $\mathbf{x}^* = \mathbf{x}_s$  and stop.

---

noiselet refers to a family of waveforms that can be used to construct a basis which is perfectly incoherent with respect to the Haar wavelet basis. A noise vector  $\mathbf{w}$  of length  $M_1 + M_2$  was constructed by drawing its components from the Gaussian distribution  $\mathcal{N}(0, 0.05^2)$  and the first  $M_1$  components of  $\mathbf{w}$  were added to the DCT measurements and the rest were added to the noiselet measurements. The TV<sub>p</sub>-RLS algorithm was run with  $T = 40, p = 0.5, \epsilon_1 = 256, \epsilon_T = 10^{-2}, \lambda_1 = 10, \lambda_T = 10^{-2}, E_{th} = 0.1$ , and  $L_t = 3 + \text{round}(t/4)$ . The proposed TV<sub>p</sub>-RLS algorithm and the TV-regularized least-squares (TV-RLS) algorithm [14] were used to recover nine different images from their noisy DCT and noiselet measurements. Both the algorithms were run on a PC desktop with Intel Core 2 CPU 6400 2.13 GHz processor running MATLAB release R2010a(7.10).

The performance of each competing algorithm was measured in terms of the peak signal-to-noise ratio (PSNR) which is defined as

$$PSNR = 20 \log \left( \frac{I_{MAX}}{\sqrt{MSE}} \right) \text{dB} \quad (13)$$

where  $I_{MAX} = 2^b - 1$ ,  $b$  is the number of bits used to encode the components of  $\mathbf{X}$ , and  $MSE$  is the mean-square error defined as

$$MSE = \frac{1}{n_1 n_2} \left\| \mathbf{X} - \hat{\mathbf{X}} \right\|_F^2.$$

Matrices  $\mathbf{X}$  and  $\hat{\mathbf{X}}$ , each of size  $n_1 \times n_2$ , represent the original and recovered images, respectively, and  $\|\cdot\|_F$  denotes the Frobenius norm. For all the images used in this experiment, we have  $I_{MAX} = 255, n_1 = 256$ , and  $n_2 = 256$ . The PSNRs and CPU time for the two competing algorithms are given in Table II. As can be seen, the TV<sub>p</sub>-RLS algorithm requires a significantly reduced computational effort relative to the TV-RLS algorithm. Also, for images Circles, Resolution Chart, and Shepp-Logan Phantom which has strictly sparse gradient, the TV<sub>p</sub>-RLS algorithm yields significantly improved PSNR;

for images Cameraman, Aeroplane, and Clock, the TV<sub>p</sub>-RLS algorithm yields slightly improved PSNR relative to the TV-RLS algorithm.

TABLE II  
PSNR AND CPU TIME FOR TV<sub>p</sub>-RLS AND TV-RLS ALGORITHMS  
(EXPERIMENT 1)

Images	TV <sub>p</sub> -RLS		TV-RLS	
	PSNR (dB)	CPU time (seconds)	PSNR (dB)	CPU time (seconds)
Cameraman	32.8	47.1	32.2	952.8
Aeroplane	41.7	49.1	41.5	767.0
Circles	90.1	43.6	58.4	483.0
Clock	38.4	48.1	37.3	911.4
Resolution chart	74.6	45.0	49.7	1201.7
Shepp-Logan	86.5	44.1	76.2	121.2

In Experiment 2, a 3D cerebral angiogram from [15], was converted to gray scale, vectorized, and approximately 0.02N DCT and 0.3N noiselet measurements were taken. The TV<sub>p</sub>-RLS algorithm was applied with the same parameters as before except for  $T$  and  $\epsilon_T$  which were chosen as  $T = 80$  and  $\epsilon_T = 10^{-5}$ . In order to better visualize the reconstruction errors, a pixel with the value greater than 6.4 was set to 6.4 for all angiograms. All the pixel values were then multiplied by 40 so that the resulting values lie in the range 0 to 256. Fig. 1 shows the original angiogram and the angiograms reconstructed using the TV<sub>p</sub>-RLS and TV-RLS algorithms. As can be seen, by enlarging Fig. 1 by 400%, the image reconstructed using the TV<sub>p</sub>-RLS algorithm is visibly better than that obtained using the TV-RLS algorithm. The corresponding PSNR for the TV<sub>p</sub>-RLS and the TV-RLS algorithms were 45.8 dB and 40.2 dB, respectively.

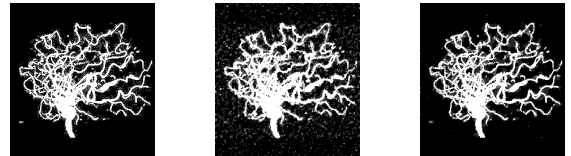


Fig. 1. Original angiogram (left panel) and the angiograms reconstructed by using the TV-RLS (middle panel) and the TV<sub>p</sub>-RLS with  $p = 0.5$  (right panel) algorithms (Experiment 2).

## V. CONCLUSION

A new and improved algorithm for the reconstruction of images, namely, the TV<sub>p</sub>-RLS algorithm, has been proposed. The TV<sub>p</sub>-RLS algorithm is based on minimizing a TV<sub>p</sub>-regularized squared-error. Using a smooth TV<sub>p</sub> norm along with Fletcher-Reeves' CG optimization method, the TV<sub>p</sub>-RLS algorithm offers significantly improved PSNR for images with strictly sparse gradient relative to the TV-RLS algorithm. Also, the TV<sub>p</sub>-RLS algorithm requires a significantly reduced computational effort relative to the log-barrier based TV-RLS algorithm.

## ACKNOWLEDGEMENT

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada for supporting this

research.

#### APPENDIX 1: GRADIENT OF $TV_{p,\epsilon}^p(\mathbf{X})$

Vector  $\mathbf{u}$  in (14) is obtained by stacking the columns of matrix  $\mathbf{U} = \{u_{ij}, 1 \leq i \leq n_1, 1 \leq j \leq n_2\}$  where

$$u_{ij} = \frac{\partial TV_{p,\epsilon}^p(\mathbf{X})}{\partial x_{ij}}. \quad (14)$$

Let  $M(:, j_1 : j_2)$  and  $M(i_1 : i_2, :)$  be the matrices composed of the  $j_1$ th to  $j_2$ th columns and the  $i_1$ th to the  $i_2$ th rows of  $M$ , respectively. Now define

$$\begin{aligned} \mathbf{Q}^h &= \mathbf{X}(:, 1 : n_2 - 1) - \mathbf{X}(:, 2 : n_2) = \{q_{ij}^h\} \\ \mathbf{Q}^v &= \mathbf{X}(1 : n_1 - 1, :) - \mathbf{X}(2 : n_1, :) = \{q_{ij}^v\} \end{aligned}$$

and

$$q_{i,j} = \begin{cases} [(q_{i,j}^v)^2 + (q_{i,j}^h)^2 + \epsilon^2]^{p/2-1} & \text{for } \begin{cases} 1 \leq i < n_1, \\ 1 \leq j < n_2 \end{cases} \\ [(q_{i,j}^h)^2 + \epsilon^2]^{p/2-1} & \text{for } \begin{cases} i = n_1, \\ 1 \leq j < n_2 \end{cases} \\ [(q_{i,j}^v)^2 + \epsilon^2]^{p/2-1} & \text{for } \begin{cases} 1 \leq i < n_1, \\ j = n_2 \end{cases} \\ 0 & \text{for } i = n_1, j = n_2 \end{cases}$$

Then  $u_{ij}$  can be obtained as

$$u_{i,j} = \begin{cases} \left. \begin{aligned} &q_{i,j} (q_{i,j}^v + q_{i,j}^h) \\ &q_{i,j} (q_{i,j}^v + q_{i,j}^h) \\ &\quad - q_{i,j-1} q_{i,j-1}^h \\ &q_{i,j} (q_{i,j}^v + q_{i,j}^h) \\ &\quad - q_{i-1,j} q_{i-1,j}^v \\ &q_{i,j} (q_{i,j}^v + q_{i,j}^h) \\ &\quad - q_{i,j-1} q_{i,j-1}^h \\ &\quad - q_{i-1,j} q_{i-1,j}^v \end{aligned} \right\} & \text{for } \begin{cases} i = 1, j = 1 \\ i = 1 \text{ and} \\ j = 2, 3, \dots, n_2 \\ i = 1, 2, \dots, n_1 \\ \text{and } j = 1 \\ i = 2, 3, \dots, n_1 \\ \text{and} \\ j = 2, 3, \dots, n_2 \end{cases} \\ \left. \begin{aligned} &q_{i,j} q_{i,j}^h \\ &\quad - q_{i,j-1} q_{i,j-1}^h \\ &\quad - q_{i-1,j} q_{i-1,j}^v \end{aligned} \right\} & \text{for } \begin{cases} i = n_1 \text{ and} \\ j = 2, 3, \dots, n_2 - 1 \end{cases} \\ \left. \begin{aligned} &q_{i,j} q_{i,j}^v \\ &\quad - q_{i,j-1} q_{i,j-1}^h \\ &\quad - q_{i-1,j} q_{i-1,j}^v \end{aligned} \right\} & \text{for } \begin{cases} i = 2, 3, \dots, n_1 - 1 \\ \text{and } j = n_2 \end{cases} \\ \left. \begin{aligned} &-q_{i,j-1} q_{i,j-1}^h \\ &\quad - q_{i-1,j} q_{i-1,j}^v \end{aligned} \right\} & \text{for } i = n_1, j = n_2 \end{cases}$$

#### APPENDIX 2: EXPRESSION FOR $G(\alpha)$

By equating the derivative  $dF_{\lambda,p,\epsilon}(\mathbf{x}_k + \alpha \mathbf{d}_k) / d\alpha$  to zero, we obtain

$$\alpha = G(\alpha)$$

where

$$\begin{aligned} G(\alpha) &= \frac{a + \lambda \cdot p \cdot b}{c + \lambda \cdot p \cdot d} \\ a &= \mathbf{d}_k^T \Phi^T (\Phi \mathbf{x}_k - \mathbf{y}) \\ c &= \mathbf{d}_k^T \Phi^T \Phi \mathbf{d}_k \\ b &= \sum_{i=1}^{n_1-1} \sum_{j=1}^{n_2-1} \hat{b}_{i,j} + \sum_{i=1}^{n_1-1} \hat{b}_{i,n_2} + \sum_{j=1}^{n_2-1} \hat{b}_{n_1,j} \\ d &= \sum_{i=1}^{n_1-1} \sum_{j=1}^{n_2-1} \hat{d}_{i,j} + \sum_{i=1}^{n_1-1} \hat{d}_{i,n_2} + \sum_{j=1}^{n_2-1} \hat{d}_{n_1,j} \end{aligned}$$

and

$$\begin{aligned} \hat{b}_{i,j} &= \gamma_{i,j}(\alpha) \left( q_{i,j}^v (d_{i,j}^i) + q_{i,j}^h (d_{i,j}^j) \right) \\ \hat{b}_{i,n_2} &= \gamma_{i,n_2}(\alpha) \cdot q_{i,n_2}^v (d_{i,n_2}^i) \\ \hat{b}_{n_1,j} &= \gamma_{n_1,j}(\alpha) \cdot q_{n_1,j}^h (d_{n_1,j}^j) \\ \hat{d}_{i,j} &= \gamma_{i,j}(\alpha) \cdot \left( d_{i,j}^{i,2} + d_{i,j}^{j,2} \right) \\ \hat{d}_{i,n_2} &= \gamma_{i,n_2}(\alpha) \cdot d_{i,n_2}^{i,2} \\ \hat{d}_{n_1,j} &= \gamma_{n_1,j}(\alpha) \cdot d_{n_1,j}^{j,2} \end{aligned}$$

where  $d_{i,j}^i = d_{i,j} - d_{i+1,j}$  and  $d_{i,j}^j = d_{i,j} - d_{i,j+1}$ .

Parameter  $\gamma_{i,j}(\alpha)$  can be determined as follows:

For  $1 \leq i < n_1$  and  $1 \leq j < n_2$ ,

$$\gamma_{i,j}(\alpha) = \left[ (q_{i,j}^v + \alpha d_{i,j}^i)^2 + (q_{i,j}^h + \alpha d_{i,j}^j)^2 + \epsilon^2 \right]^{p/2-1}.$$

For  $1 \leq i < n_1$  and  $j = n_2$ ,

$$\gamma_{i,j}(\alpha) = \left[ (q_{i,j}^v + \alpha d_{i,j}^i)^2 + \epsilon^2 \right]^{p/2-1}.$$

For  $i = n_1$  and  $1 \leq j < n_2$ ,

$$\gamma_{i,j}(\alpha) = \left[ (q_{i,j}^h + \alpha d_{i,j}^j)^2 + \epsilon^2 \right]^{p/2-1}.$$

#### REFERENCES

- [1] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489-509, Feb. 2006.
- [2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289-1306, Apr. 2006.
- [3] E. Candès and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406-5425, Dec. 2006.
- [4] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Scientif. Comput.*, vol. 20, no. 1, pp. 33-61, 1999.
- [5] J. K. Pant, W.-S. Lu, and A. Antoniou, "Unconstrained regularized  $\ell_p$ -norm based algorithm for the reconstruction of sparse signals," in *Proc. IEEE Inter. Symp. Circuits and Syst.*, pp. 1740-1743, May 2011.
- [6] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proc. IEEE Inter. Conf. Acoustics, Speech, Signal Process.*, pp. 3869-3872, 2008.
- [7] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, Dec. 2007.
- [8] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259-268, 1992.
- [9] M. Zibuleusky and M. Elad, "L1-L2 optimization in signal and image processing," *IEEE Signal Process. Mag.*, pp. 76-88, May. 2010.
- [10] UCLA, "Computational and Applied Mathematics Reports, a web site maintained by the Math Dept. UCLA," Dec. 2011 (url = "http://www.math.ucla.edu/applied/cam/").
- [11] J. K. Pant, W.-S. Lu, and A. Antoniou, "Recovery of sparse signals from noisy measurements using an  $\ell_p$  regularized least-squares algorithm," in *Proc. IEEE Pac. Rim Conf. in Comm., Computers, and Signal Process.*, pp. 48-53, Aug. 2011.
- [12] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, Springer, 2006.
- [13] R. P. Agrawal, M. Meehan, and D. O'Regan, *Fixed Point Theory and Applications*, Cambridge University Press, 2011.
- [14] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Process. Mag.*, pp. 14-20, Mar. 2008.
- [15] Website of neurosurgeon Dr. Jonathan Curtis, <http://www.drcurtis.com.au/blood-vessel-conditions-aneurysm-sydney.html> [Online: accessed April 2012].