Total Variation Filtering

Ivan W. Selesnick and İlker Bayram

February 4, 2010

1 Introduction

These notes describe the derivation of a simple algorithm for signal denoising (filtering) based on total variation (TV). Total variation based filtering was introduced by Rudin, Osher, and Fatemi [8]. TV denoising is an effective filtering method for recovering *piecewise-constant* signals. Many algorithms have been proposed to implement total variation filtering. The one described in these notes is by Chambolle [3]. (Note: Chambolle described another algorithm in [2]). Although the algorithm can be derived in several different ways, the derivation presented here is based on descriptions given in [1, 10]. The derivation is based on the min-max property and the majorization-minimization procedure.

Total variation is often used for image filtering and restoration, however, to simplify the presentation of the TV filtering algorithm these notes concentrate on one-dimensional signal filtering only. In addition, the algorithm described here may converge slowly for some problems. Faster algorithms for TV filtering have recently been developed, for example [1, 10]. The development of fast, robust algorithms for TV and related non-linear filtering is an active topic of research.

2 Total Variation

The total variation (TV) of a signal measures how much the signal changes between signal values. Specifically, the total variation of an N-point signal $x(n)$, $1 \le n \le N$ is defined as

$$\mathrm{TV}(\mathbf{x}) = \sum_{n=2}^{N} |x(n) - x(n-1)|.$$

The total variation of $\mathbf{x}$ can also be written as

$$\mathrm{TV}(\mathbf{x}) = \|\mathbf{D}\mathbf{x}\|_1$$

where $\|\cdot\|_1$ is the ℓ_1 norm and

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & & & & \\ & -1 & 1 & & & \\ & & & \ddots & & \\ & & & & -1 & 1 \end{bmatrix} \tag{1}$$

is a matrix of size $(N-1) \times N$.

3 TV Denoising

We assume we observe the signal $\mathbf{x}$ corrupted by additive white Gaussian noise,

$$\mathbf{y} = \mathbf{x} + \mathbf{n}, \qquad \mathbf{y}, \mathbf{x}, \mathbf{n} \in \mathbb{R}^N.$$

One approach to estimate $\mathbf{x}$ is to find the signal $\mathbf{x}$ minimizing the objective function

$$J(\mathbf{x}) = \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \, \|\mathbf{D}\mathbf{x}\|_1.$$

This approach is called *TV denoising*. The regularization parameter, λ, controls how much smoothing is performed. Larger noise levels call for larger λ.

4 Algorithm for TV Denoising

We will assume a more general form of the objective function:

$$J(\mathbf{x}) = \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \, \|\mathbf{A}\mathbf{x}\|_1 \tag{2}$$

where $\mathbf{A}$ is a matrix of size $M \times N$. The optimal value of the objective function is denoted

$$J_* = \min_{\mathbf{x}} \; \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \, \|\mathbf{A}\mathbf{x}\|_1. \tag{3}$$

The minimization of this objective function is complicated by the fact that the ℓ_1 norm is not differentiable. Therefore, an approach to minimize $J(\mathbf{x})$ is to use the dual formulation.

To derive the dual formulation, note that the absolute value of a scalar x can be written in the following circuitous form:

$$|x| = \max_{|z| \le 1} \; z\, x.$$

The advantage of this form is that the non-differentiability of the function is transferred to the feasible set. Likewise, note that the ℓ_1 norm of a vector $\mathbf{x}$ can be written as:

$$\|\mathbf{x}\|_1 = \max_{|\mathbf{z}| \le 1} \; \mathbf{z}^t \mathbf{x}$$

where the condition $|\mathbf{z}| \le 1$ is taken element-wise. Similarly,

$$\|\mathbf{A}\mathbf{x}\|_1 = \max_{|\mathbf{z}| \le 1} \; \mathbf{z}^t \mathbf{A}\mathbf{x}.$$

Therefore, we can write the objective function $J(\mathbf{x})$ in (2) as

$$J(\mathbf{x}) = \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \max_{|\mathbf{z}| \le 1} \; \mathbf{z}^t \mathbf{A}\mathbf{x} \tag{4}$$

or

$$J(\mathbf{x}) = \max_{|\mathbf{z}| \le 1} \; \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \, \mathbf{z}^t \mathbf{A}\mathbf{x}. \tag{5}$$

The optimal value of the objective function (3) is

$$J_* = \min_{\mathbf{x}} \max_{|\mathbf{z}| \le 1} \; \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \, \mathbf{z}^t \mathbf{A}\mathbf{x}.$$

We want to find the minimizing vector $\mathbf{x}$, but it will be convenient to find both $\mathbf{x}$ and the auxiliary vector $\mathbf{z}$.

Defining

$$F(\mathbf{x}, \mathbf{z}) := \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda\, \mathbf{z}^t \mathbf{A}\mathbf{x}, \tag{6}$$

we can write:

$$J_* = \min_{\mathbf{x}} \max_{|\mathbf{z}| \leq 1} \; F(\mathbf{x}, \mathbf{z}).$$

Because $F(\mathbf{x}, \mathbf{z})$ is convex in $\mathbf{x}$ and concave in $\mathbf{z}$, the optimal value J_* is a saddle point of $F(\mathbf{x}, \mathbf{z})$. By the *min-max* property, we can exchange the order of the maximization and minimization:

$$J_* = \max_{|\mathbf{z}| \leq 1} \min_{\mathbf{x}} \; F(\mathbf{x}, \mathbf{z})$$

or

$$\boxed{J_* = \max_{|\mathbf{z}| \leq 1} \min_{\mathbf{x}} \; \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda\, \mathbf{z}^t \mathbf{A}\mathbf{x}.} \tag{7}$$

which is the dual formulation of the TV denoising problem. The min-max property is described in [6, Chapter VII, Theorem 4.3.1] (cited by [10]) and in [7, Corollary 37.3.2] (cited by [1]).

The inner minimization problem in (7) can be solved as follows:

$$\frac{\partial}{\partial \mathbf{x}} F(\mathbf{x}, \mathbf{z}) = -2(\mathbf{y} - \mathbf{x}) + \lambda \mathbf{A}^t \mathbf{z}$$

so

$$\frac{\partial}{\partial \mathbf{x}} F(\mathbf{x}, \mathbf{z}) = \mathbf{0} \quad \implies \quad \boxed{\mathbf{x} = \mathbf{y} - \frac{\lambda}{2} \mathbf{A}^t \mathbf{z}} \tag{8}$$

Substituting (8) back into (7) gives

$$J_* = \max_{|\mathbf{z}| \leq 1} \quad \|\frac{\lambda}{2} \mathbf{A}^t \mathbf{z}\|_2^2 + \lambda\, \mathbf{z}^t \mathbf{A} \Big(\mathbf{y} - \frac{\lambda}{2} \mathbf{A}^t \mathbf{z} \Big).$$

After simplifying we have,

$$J_* = \max_{|\mathbf{z}| \leq 1} \quad -\frac{\lambda^2}{4} \mathbf{z}^t \mathbf{A}\mathbf{A}^t \mathbf{z} + \lambda \mathbf{z}^t \mathbf{A}\mathbf{y}$$

or equivalently, the minimization problem:

$$\mathbf{z}_* = \operatorname*{argmin}_{|\mathbf{z}| \leq 1} \quad \mathbf{z}^t \mathbf{A}\mathbf{A}^t \mathbf{z} - \frac{4}{\lambda}\, \mathbf{z}^t \mathbf{A}\mathbf{y}. \tag{9}$$

Setting the derivative with respect to $\mathbf{z}$ to zero gives the equation

$$\mathbf{A}\mathbf{A}^t \mathbf{z} = \frac{2}{\lambda} \mathbf{A}\mathbf{y}$$

which requires the solution to a potentially large system of linear equations and furthermore does not yield a solution $\mathbf{z}$ satisfying the constraint $|\mathbf{z}| \le 1$. To find $\mathbf{z}$ solving the constrained minimization problem (9), the majorization-minimization (MM) method can be used [5]. Defining

$$D(\mathbf{z}) = \mathbf{z}^t \mathbf{A}\mathbf{A}^t \mathbf{z} - \frac{4}{\lambda}\, \mathbf{z}^t \mathbf{A}\mathbf{y}$$

and setting $\mathbf{z}^{(i)}$ as point of coincidence, we can find a separable majorizer of $D(\mathbf{z})$ by adding the non-negative function

$$(\mathbf{z} - \mathbf{z}^{(i)})^t (\alpha\, \mathbf{I} - \mathbf{A}\mathbf{A}^t)(\mathbf{z} - \mathbf{z}^{(i)})$$

to $D(\mathbf{z})$, where α is greater than or equal to the maximum eigenvalue of $\mathbf{A}\mathbf{A}^t$. So a majorizer of $D(\mathbf{z})$ is given by

$$D(\mathbf{z}) + (\mathbf{z} - \mathbf{z}^{(i)})^t (\alpha\, \mathbf{I} - \mathbf{A}\mathbf{A}^t)(\mathbf{z} - \mathbf{z}^{(i)})$$

and, using the MM approach, the update equation for $\mathbf{z}$ is given by

$$\begin{aligned}
\mathbf{z}^{(i+1)} &= \operatorname*{argmin}_{|\mathbf{z}| \le 1} \quad D(\mathbf{z}) + (\mathbf{z} - \mathbf{z}^{(i)})^t (\alpha\, \mathbf{I} - \mathbf{A}\mathbf{A}^t)(\mathbf{z} - \mathbf{z}^{(i)}) && (10)\\
&= \operatorname*{argmin}_{|\mathbf{z}| \le 1} \quad \alpha\, \mathbf{z}^t\mathbf{z} - 2\Big(\mathbf{A}\Big(\frac{2}{\lambda}\mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}\Big) + \alpha\, \mathbf{z}^{(i)}\Big)^t \mathbf{z} + K && (11)\\
&= \operatorname*{argmin}_{|\mathbf{z}| \le 1} \quad \mathbf{z}^t\mathbf{z} - 2\Big(\frac{1}{\alpha}\mathbf{A}\Big(\frac{2}{\lambda}\mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}\Big) + \mathbf{z}^{(i)}\Big)^t \mathbf{z} && (12)\\
&= \operatorname*{argmin}_{|\mathbf{z}| \le 1} \quad \mathbf{z}^t\mathbf{z} - 2\mathbf{b}^t\mathbf{z} && (13)
\end{aligned}$$

where

$$\mathbf{b} = \mathbf{z}^{(i)} + \frac{1}{\alpha}\mathbf{A}\Big(\frac{2}{\lambda}\mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}\Big).$$

We need to find $\mathbf{z} \in \mathbb{R}^M$ minimizing $\mathbf{z}^t\mathbf{z} - 2\mathbf{b}^t\mathbf{z}$ subject to the constraint $|\mathbf{z}| \le 1$. Consider first the scalar case:

$$\operatorname*{argmin}_{|z| \le 1} \quad z^2 - 2bz. \tag{14}$$

The minimum of $z^2 - 2bz$ is at $z = b$. If $|b| \le 1$, then the solution is $z = b$. If $|b| \ge b$, then the solution is $z = \operatorname{sign}(b)$. If we define the clipping function,

$$\operatorname{clip}(b, T) := \begin{cases} b & |b| \le T \\ T \ \operatorname{sign}(b) & |b| \ge T \end{cases} \tag{15}$$

as illustrated in Fig. 1, then we can write the solution to (14) as $z = \operatorname{clip}(b, 1)$.

Note that the vector case (13) is separable — the elements of $\mathbf{z}$ are uncoupled so the constrained minimization can be performed element-wise. Therefore, an update equation for $\mathbf{z}$ is given by:

$$\boxed{\mathbf{z}^{(i+1)} = \operatorname{clip}\Big(\mathbf{z}^{(i)} + \frac{1}{\alpha}\mathbf{A}\Big(\frac{2}{\lambda}\mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}\Big), 1\Big)} \tag{16}$$

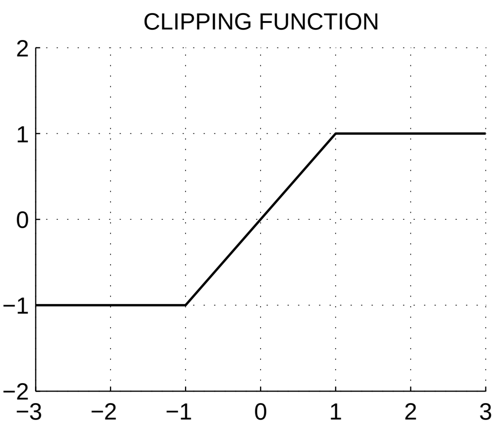

Figure 1: Clipping function (15).

where i is the iteration index. Once $\mathbf{z}^{(i)}$ has converged to one's satisfaction, the denoised signal $\mathbf{x}$ is given by (8). Because the optimization problem is convex, the iteration will converge from any initialization. We may choose, say $\mathbf{z}^{(0)} = \mathbf{0}$. We call this the *iterative clipping algorithm*.

This algorithm can also be written as

$$\mathbf{x}^{(i+1)} = \mathbf{y} - \frac{\lambda}{2}\mathbf{A}^t\mathbf{z}^{(i)} \tag{17}$$

$$\mathbf{z}^{(i+1)} = \mathrm{clip}\Big(\mathbf{z}^{(i)} + \frac{2}{\alpha\lambda}\mathbf{A}\mathbf{x}^{(i+1)}, 1\Big). \tag{18}$$

Scaling $\mathbf{z}$ by $\lambda/2$, we have the following equivalent form:

$$\mathbf{x}^{(i+1)} = \mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)} \tag{19}$$

$$\mathbf{z}^{(i+1)} = \mathrm{clip}\Big(\mathbf{z}^{(i)} + \frac{1}{\alpha}\mathbf{A}\mathbf{x}^{(i+1)}, \frac{\lambda}{2}\Big). \tag{20}$$

In summary:

> The objective function
>
> $$J(\mathbf{x}) = \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda\,\|\mathbf{A}\mathbf{x}\|_1$$
>
> can be minimized by the iterative clipping algorithm:
>
> $$\mathbf{x}^{(i+1)} = \mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}$$
> $$\mathbf{z}^{(i+1)} = \mathrm{clip}\Big(\mathbf{z}^{(i)} + \frac{1}{\alpha}\mathbf{A}\mathbf{x}^{(i+1)}, \frac{\lambda}{2}\Big)$$
>
> for $i \geq 0$ with $\mathbf{z}^{(0)} = \mathbf{0}$ and $\alpha \geq \mathrm{maxeig}(\mathbf{A}\mathbf{A}^t)$.

For the matrix $\mathbf{D}$ in (1), the maximum eigenvalue of $\mathbf{D}\mathbf{D}^t$ is less than four regardless of N, so for TV denoising we can set $\alpha = 4$. A MATLAB program to implement the TV denoising algorithm is given in

```
function [x,J] = denoiseTV(y,lambda,Nit)
% [x,J] = denoiseTV(y,lambda,a,Nit)
% Total variation filtering (denoising) using
% iterative clipping algorithm.
% INPUT
%   y - noisy signal (row vector)
%   lambda - regularization parameter
%   Nit - number of iterations
% OUTPUT
%   x - result of denoising
%   J - objective function

J = zeros(1,Nit);       % objective function
N = length(y);
z = zeros(1,N-1);       % initialize z
alpha = 4;
T = lambda/2;
for k = 1:Nit
    x = y - [-z(1) -diff(z) z(end)];      % y - D' z
    J(k) = sum(abs(x-y).^2) + lambda * sum(abs(diff(x)));
    z = z + 1/alpha * diff(x);            % z + 1/alpha D z
    z = max(min(z,T),-T);                 % clip(z,T)
end
```

Listing 1: MATLAB program for TV denoising using the iterative clipping algorithm (17) and (18).

Listing 1. In the MATLAB program, $\mathbf{D}$ is implemented with the `diff` command. Also, note from (1) that

$$\mathbf{D}^t = \begin{bmatrix} -1 & & & & \\ 1 & -1 & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & -1 \\ & & & & 1 \end{bmatrix}. \tag{21}$$

Therefore, $\mathbf{D}^t\,\mathbf{z}$ can be implemented in MATLAB as `[-z(1) -diff(z) z(end)]`.

5 Examples

Example 1: Figure 2 illustrates an example of TV denoising. We use the 'Blocks' signal in the MATLAB Wavelab toolbox [4] as a test signal. The noise-free signal, noisy signal, and signal obtained using TV filtering are shown in Fig. 2. Note that TV filtering preserves the discontinuities in the signal quite well. Conventional smoothing using a moving average filter has a greater tendency to blur the discontinuities.

Note that the objective function, illustrated in Fig. 2, is strictly decreasing through the progression of the iterative clipping algorithm. This is due to the use of the majorization-minimization procedure in the derivation of the iterative clipping algorithm. Because the majorizing function is convex, the MM procedure guarantees that the objective function decreases at each iteration.

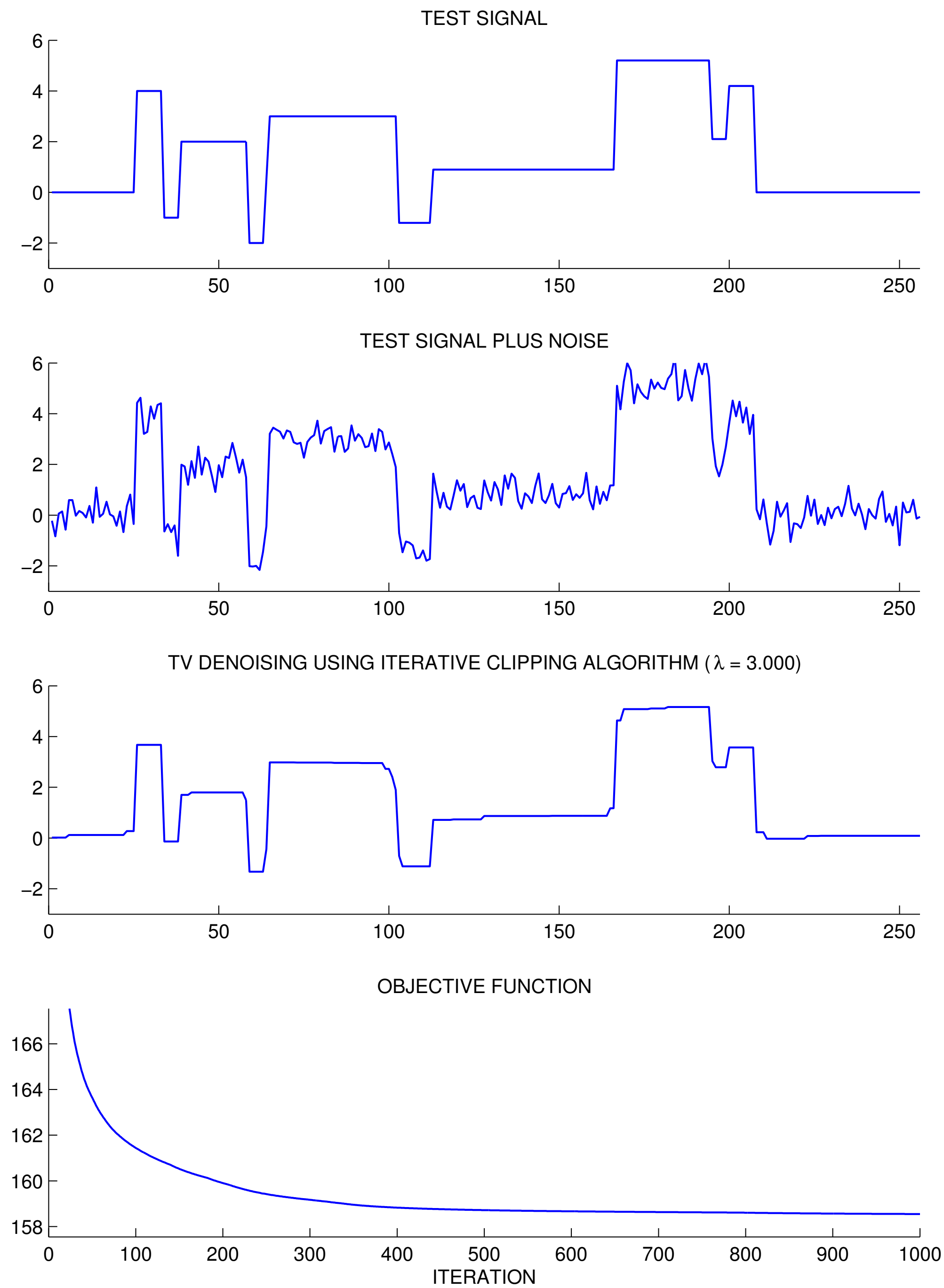

Figure 2: Example 1: An illustration of TV denoising.

Example 2: Figure 3 illustrates another example of TV denoising. This time, we use the 'Piece-Regular' signal in the Wavelab toolbox as a test signal. As illustrated in Fig. 3, for signals that are not piece-wise constant, TV denoising has the tendency to introduce a staircase effect. The staircase effect refers to the appearance of small flat regions in the denoised signal. These regions arise because the TV regularizer promotes piecewise-constant behaviour. Therefore, while TV denoising is suitable for filtering piecewise-constant signals, it is not usually the best denoising method for more general piecewise-smooth signals. For signals like the one in Fig. 3, a higher-order difference can be used instead of the first-order difference on which the TV regularizer is based. (For example, TV can be replaced by a second derivative filter or a wavelet filter [9]).

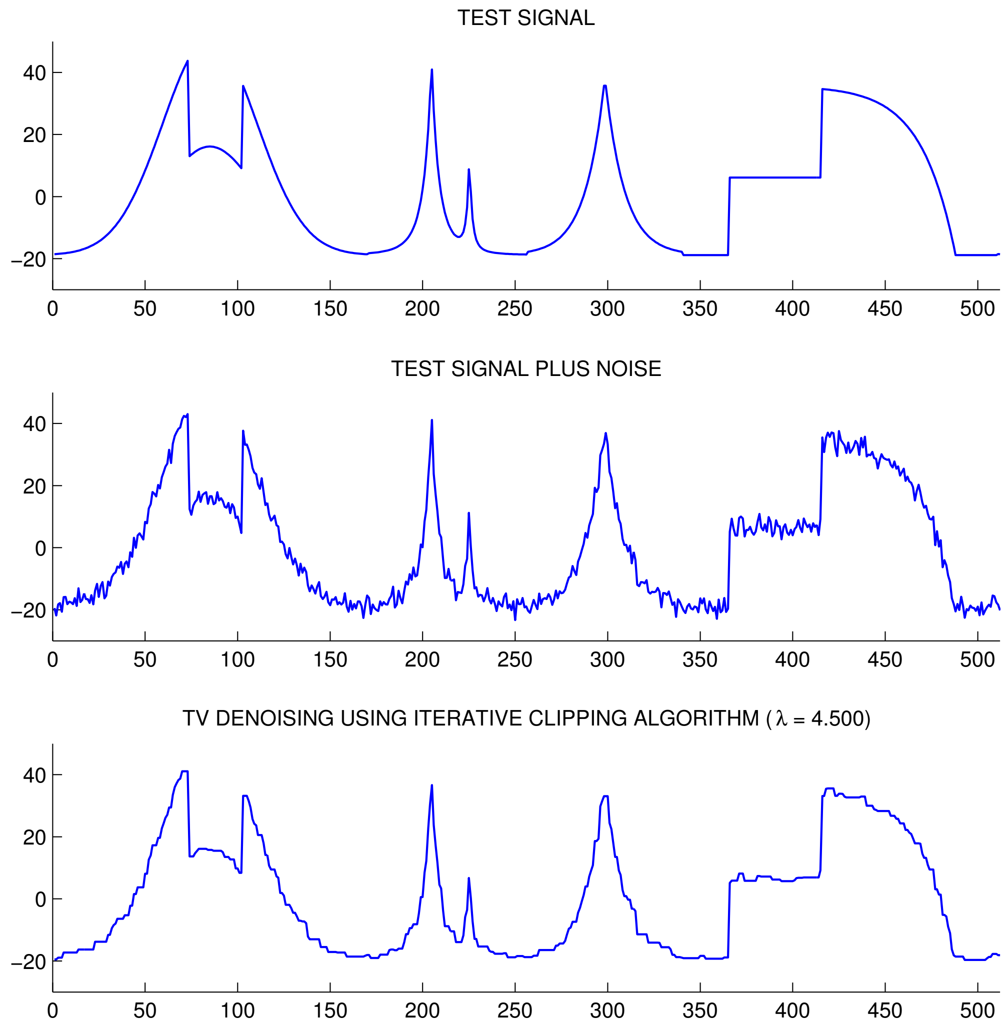

Figure 3: Example 2: TV denoising of a piecewise-smooth function. The presence of small flat regions in the denoised signal is known as the staircase effect.

6 Acceleration

It turns out that the convergence of the iterative clipping algorithm can be accelerated by using a smaller value for α. The condition $\alpha \geq \operatorname{maxeig}(\mathbf{A}\mathbf{A}^t)$ ensures that the objective function decreases from one iteration to the next. However, a smaller value of α can lead to faster convergence (although the objective function may increase on some iterations). The derivation is based on the contraction mapping principle.

The iterative clipping algorithm derived above, is given by:

$$\mathbf{z}^{(i+1)} = \operatorname{clip}\Big(\mathbf{z}^{(i)} + \frac{1}{\alpha}\mathbf{A}(\mathbf{y} - \mathbf{A}^t\mathbf{z}^{(i)}),\ \frac{\lambda}{2}\Big). \tag{22}$$

Note that $1/\alpha$ can be viewed as a step size parameter. Larger step sizes may lead to faster convergence, but if the step size is too large then the algorithm may diverge or fail to converge. We would like to find what values of α ensure convergence. To that end, we will find the values of α that make (22) a contraction. Note that the clipping function is a contraction, so let us just consider the mapping

$$T(\mathbf{z}) = \mathbf{z} + \frac{1}{\alpha}\mathbf{A}(\mathbf{y} - \mathbf{A}^t\mathbf{z}).$$

The mapping $T(\mathbf{z})$ is a contraction if

$$\mathbf{I} - \frac{1}{\alpha}\mathbf{A}\mathbf{A}^t$$

is a contraction (if all its eigenvalues are less than one in absolute value). Because this is a real symmetric matrix, all its eigenvalues are real; so it is a contraction if its eigenvalues are between -1 and 1.

If e_m (with $1 \leq m \leq M$) denote the eigenvalues of $\mathbf{A}\mathbf{A}^t$, then the eigenvalues of $\mathbf{I} - \frac{1}{\alpha}\mathbf{A}\mathbf{A}^t$ are given by $1 - e_m/\alpha$. We assume $\alpha > 0$, as it represents a step-size. We also assume that all eigenvalues e_m are positive. Then, in order that all $1 - e_m/\alpha$ lie between -1 and 1, it is required that $\alpha > e_m/2$ for $1 \leq m \leq M$. That is, α must be greater than half the maximum eigenvalue of $\mathbf{A}\mathbf{A}^t$:

$$\boxed{\alpha > 0.5\,\operatorname{maxeig}(\mathbf{A}\mathbf{A}^t)}$$

Recall that the MM procedure led to $\alpha \geq \operatorname{maxeig}(\mathbf{A}\mathbf{A}^t)$. Therefore, the contraction mapping principle allows us to basically halve the value of α while still ensuring convergence (but the objective function may increase on some iterations instead of being always decreasing).

For the TV denoising example in Fig. 2 we used $\alpha = 4$ in accordance with the MM derivation above. However, according to the contraction mapping principle, it is sufficient to use $\alpha > 2$. Fig. 4 illustrates the objective function when we use $\alpha = 2.3$. The faster convergence of objective function to its minimum is visible in the figure.

6.1 Fixed point

We showed that α can be reduced down to $0.5\,\operatorname{maxeig}(\mathbf{A}\mathbf{A}^t)$ without losing the contraction mapping property of (22). However, we should check that reducing α down to this value does not change the fixed point of the iteration; otherwise, the iteration will converge but to a signal different from the desired TV-filtered signal.

In the following we show that the fixed point of (22) is the same for the smaller value of α. Let us consider the scaler case first. Given $A, y, \alpha, T \in \mathbb{R}$ with $\alpha > 0$ and $T > 0$, suppose $z \in \mathbb{R}$ is a fixed point of

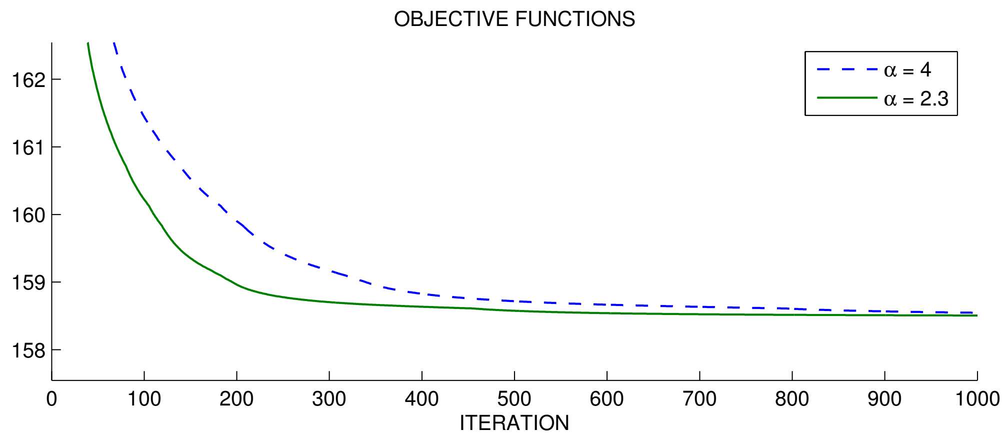

Figure 4: Improvement in convergence using a smaller α.

the iteration:

$$z^{(i+1)} = \text{clip}\Big(z^{(i)} + \frac{1}{\alpha} A(y - A^t z^{(i)}),\, T\Big). \tag{23}$$

Due to the clipping function z can not be larger than T in absolute value. Let us consider separately the three cases: $z = T$, $-T < z < T$, and $z = -T$.

Case 1: $z = T$

Suppose $z = T$ is a fixed point of (23). Then the input of the clipping function must be greater than or equal to T. Accordingly, we can write the following:

$$\begin{aligned}
z + \frac{1}{\alpha} A(y - z) &\geq T && \text{(24)}\\
T + \frac{1}{\alpha} A(y - z) &\geq T \quad \text{because } z = T && \text{(25)}\\
\frac{1}{\alpha} A(y - z) &\geq 0 && \text{(26)}\\
A(y - z) &\geq 0 \quad \text{because } \alpha > 0 && \text{(27)}\\
\frac{1}{\alpha'} A(y - z) &\geq 0 \quad \text{for all } \alpha' > 0 && \text{(28)}\\
T + \frac{1}{\alpha'} A(y - z) &\geq T \quad \text{for all } \alpha' > 0 && \text{(29)}\\
\text{clip}\Big(z + \frac{1}{\alpha'} A(y - z), T\Big) &= T \quad \text{for all } \alpha' > 0 \text{ because } z = T. && \text{(30)}
\end{aligned}$$

Therefore, if $z = T$ is a fixed point of (23) for some $\alpha > 0$, then it is a fixed point for any $\alpha > 0$.

Case 2: $-T < z < T$

Suppose $-T < z < T$ is a fixed point of (23). If the output of the clipping function is between $-T$ and T then the clipping function in not affecting its input. In this case the output of the clipping function equals its input. Accordingly, we can write the following:

$$z = z + \frac{1}{\alpha} A(y - z) \tag{31}$$

$$0 = \frac{1}{\alpha} A(y - z) \tag{32}$$

$$0 = \frac{1}{\alpha'} A(y - z) \qquad \text{for all } \alpha' > 0 \tag{33}$$

$$z = z + \frac{1}{\alpha'} A(y - z) \qquad \text{for all } \alpha' > 0 \tag{34}$$

$$z = \text{clip}\Big(z + \frac{1}{\alpha'} A(y - z), T\Big) \quad \text{for all } \alpha' > 0 \text{ because } -T < z < T. \tag{35}$$

Therefore, if $-T < z < T$ is a fixed point of (23) for some $\alpha > 0$, then it is a fixed point for any $\alpha > 0$.

Case 3: $z = -T$
The case $z = -T$ is similar to the case $z = T$.

• Therefore, if z is a fixed point of (23) for some $\alpha > 0$, then it is a fixed point for any $\alpha > 0$.

The derivation in the vector case is essentially the same as in the scalar case because the clipping function is applied element-wise.
Given: $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{y} \in \mathbb{R}^M$, $\mathbf{A} \in \mathbb{R}^{M \times N}$, and $\alpha, T \in \mathbb{R}$ with $\alpha > 0$ and $T > 0$. If $\mathbf{z} \in \mathbb{R}^M$ is a fixed point of the iteration

$$\mathbf{z}^{(i+1)} = \text{clip}\Big(\mathbf{z}^{(i)} + \frac{1}{\alpha} \mathbf{A}(\mathbf{y} - \mathbf{A}^t \mathbf{z}^{(i)}),\, T\Big) \tag{36}$$

then it is a fixed point for any $\alpha > 0$.

References

[1] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. Preprint, retrieved from http://iew3.technion.ac.il/Home/Users/becka.html, 2009.

[2] A. Chambolle. An algorithm for total variation minimization and applications. *J. of Math. Imaging and Vision*, 20:89–97, 2004.

[3] A. Chambolle. Total variation minimization and a class of binary MRF models. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 3757 of *Lecture Notes in Computer Sciences*, pages 136–152. Springer, 2005.

[4] D. Donoho, A. Maleki, and M. Shahram. Wavelab 850. `http://www-stat.stanford.edu/~wavelab/`.

[5] M. Figueiredo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. on Image Processing*, 16(12):2980–2991, December 2007.

[6] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I*. Springer-Verlag, 1993.

[7] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

[8] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[9] I. W. Selesnick and M. A. T. Figueiredo. Signal restoration with overcomplete wavelet transforms: comparison of analysis and synthesis priors. In *Proceedings of SPIE*, volume 7446 (Wavelets XIII), 2009.

[10] M. Zhu, S. J. Wright, and T. F. Chan. Duality-based algorithms for total-variation-regularized image restoration. *J. Comp. Optimization and Applications*, 2008.