Improved Total Variation-Type Regularization Using Higher-Order Edge Detectors

W. Stefan, R. Renaut, A. Gelb

Arizona State University

ICOSAHOM, July 26, 2009
Motivation

- **Deblur signals** to identify structures, e.g. tumor in brain scan
Motivation

- Deblur signals to identify structures, e.g. tumor in brain scan
- Recorded data is contaminated by noise and blur.
Motivation

- **Deblur signals** to identify structures, e.g. tumor in brain scan
- Recorded data is contaminated by **noise and blur**.
- May contain **jump discontinuities**.
Motivation

- Deblur signals to identify structures, e.g. tumor in brain scan
- Recorded data is contaminated by noise and blur.
- May contain jump discontinuities.
- Preserve edges and smooth regions.
Motivation

- **Deblur signals** to identify structures, e.g. tumor in brain scan.
- Recorded data is contaminated by **noise and blur**.
- May contain **jump discontinuities**.
- Preserve edges and smooth regions.
- Identify edges.
Motivation

- Deblur signals to identify structures, e.g. tumor in brain scan.
- Recorded data is contaminated by noise and blur.
- May contain jump discontinuities.
- Preserve edges and smooth regions.
- Identify edges.
- MRI reconstruction from truncated Fourier series (compressed sensing)
Forward Model, blurring

\[ g = f \ast h + n = Hf + n \]

- \( f \) original signal, that we try to find.
- \( h \) convolution kernel or point spread function (PSF).
- \( H \) blurring matrix (Toeplitz)
- \( n \) noise (e.g. from the camera, or quantization noise)
- \( g \) recorded data
Total Variation (TV) regularization

- **Find:** \( f \) from \( g = Hf + n \)
- **Given:** \( H \) and \( g \)
- **Unknown:** \( n \)

\[ \text{Given: } H \text{ and } g \]
\[ \text{Find: } f \text{ from } g = Hf + n \]
\[ \text{Unknown: } n \]

\[ \min \hat{f} \text{ Reg}(\hat{f}) \text{ such that } \|H \hat{f} - g\|_2^2 < \sigma \]

\[ \text{Assume } \text{Reg}(\hat{f}) \text{ is strictly convex} \]

\[ \sigma \text{ is the regularization parameter} \]

\[ \text{Popular choice: } \text{Reg}(\hat{f}) = \text{TV}(\hat{f}) = \|L \hat{f}\|_1 = \sum_i |(L \hat{f})_i| \]

\[ L \text{ being a first order derivative} \]
Total Variation (TV) regularization

- **Find:** $f$ from $g = Hf + n$
- **Given:** $H$ and $g$
- **Unknown:** $n$
- **ill posed** ⇒ use regularization

$$\min_{\hat{f}} \text{Reg}(\hat{f}) \quad \text{such that} \quad \|H\hat{f} - g\|_2^2 < \sigma$$

- assume $\text{Reg}(f)$ is strictly convex
Total Variation (TV) regularization

- **Find**: \( f \) from \( g = Hf + n \)
- **Given**: \( H \) and \( g \)
- **Unknown**: \( n \)
- **ill posed** \( \Rightarrow \) use regularization

\[
\min_{\hat{f}} \text{Reg}(\hat{f}) \quad \text{such that} \quad \| H\hat{f} - g \|_2^2 < \sigma
\]

- assume \( \text{Reg}(f) \) is strictly convex
- \( \sigma \) is the Regularization parameters
Total Variation (TV) regularization

- **Find:** $f$ from $g = Hf + n$
- **Given:** $H$ and $g$
- **Unknown:** $n$
- **Ill posed** ⇒ use regularization

\[
\min_{\hat{f}} \text{Reg}(\hat{f}) \quad \text{such that} \quad \|H\hat{f} - g\|_2^2 < \sigma
\]

- Assume $\text{Reg}(f)$ is strictly convex
- $\sigma$ is the Regularization parameters
- Popular choice: $\text{Reg}(\hat{f}) = TV(\hat{f}) = \|L\hat{f}\|_1 = \sum_i |(L\hat{f})_i|$
- $L$ being a first order derivative
Total Variation (TV) example

- piecewise constant
- loss of contrast
- preserves jump positions

\[ \min \| \cdot \|_1 \] solutions are sparse (zero almost everywhere)

- for total variation $\Rightarrow \hat{f}'(x) = 0$ for almost all $x$
- $\Rightarrow$ Total Variation solution is piecewise constant
Discrete edge detectors

**Goal:** find better operator $L$

$Idea$: Use jump function: 
$$f(x) := f(x + 1) - f(x - 1)$$

**Problem:** Only have grid data

Nonzero parts and oscillations

Fix: use 1st order at the jump and higher order away from it
Discrete edge detectors

**Goal:** find better operator $L$

- such that $Lf$ is sparse for piecewise smooth $f$

Idea: Use jump function: $f(x) := f(x+1) - f(x)$

Problem: Only have grid data

Nonzero parts and oscillations

Fix: use 1st order at the jump and higher order away from it
Discrete edge detectors

**Goal:** find better operator $L$

- such that $Lf$ is sparse for piecewise smooth $f$

**Idea:** Use jump function: $[f](x) := f(x^+) - f(x^-)$
Discrete edge detectors

**Goal:** find better operator $L$

- such that $Lf$ is sparse for piecewise smooth $f$

**Idea:** Use jump function: $[f](x) := f(x^+) - f(x^-)$

**Problem:** Only have grid data $\Rightarrow$ Need approximation of $[f](x)$

First order edge detector

Third order edge detector
Discrete edge detectors

**Goal:** find better operator $L$
- such that $Lf$ is sparse for piecewise smooth $f$

**Idea:** Use jump function: $[f](x) := f(x^+) - f(x^-)$

**Problem:** Only have grid data ⇒ Need approximation of $[f](x)$

- Nonzero parts and oscillations
- **Fix:** use 1st order at the jump and higher order away from it
Variable order example:

- Good news: Variable order method can be used as regularization
Good news: Variable order method can be used as regularization

Bad news: Need jump positions first
Structure of higher order TV solutions:

Minimizer of

$$\min_{\hat{f}} \{ \| H\hat{f} - g \|_2^2 + \lambda \| L^m f \|_1 \}$$

- $\hat{f}$ has derivative of order $m$ that is zero almost everywhere
- $\hat{f}$ is piecewise polynomial of degree $m - 1$
- $L^m f$ is non-zero at contact points between the polynomial segments.
Example 1:

\[
\min_{\hat{f}} \{ \| H\hat{f} - g \|_2^2 + \lambda \| L^1 f \|_1 \}
\]

classical TV solutions (m=1)

- piecewise constant (polynomial of degree 0) with sparse derivative
- jump discontinuities in the original signal are preserved (under some conditions)
Example 1:

\[
\min_{\hat{f}} \{ \| H\hat{f} - g \|_2^2 + \lambda \| L^1 f \|_1 \}
\]

classical TV solutions (m=1)

- piecewise constant (polynomial of degree 0) with sparse derivative
- jump discontinuities in the original signal are preserved (under some conditions)
Example 2:

\[
\min_{\hat{f}} \{\| H\hat{f} - g \|_2^2 + \lambda \| L^2 f \|_1 \}
\]

**second order TV solutions (m=2)**

- piecewise linear (polynomial of degree 1) with sparse second derivative
- no contact point of line segments at position of original jump
Example 2:

\[
\min_{\hat{f}} \left\{ \|H\hat{f} - g\|_2^2 + \lambda \|L^2 f\|_1 \right\}
\]

**second order TV solutions (m=2)**

- piecewise linear (polynomial of degree 1) with sparse second derivative
- no contact point of line segments at position of original jump
Example 3:

\[
\min_{\hat{f}} \left\{ \| H\hat{f} - g \|_2^2 + \lambda \| L^3 f \|_1 \right\}
\]

third order TV solutions (m=3)

- piecewise quadratic with sparse third derivative
- contact point of parabola segments at position of original jump
Example 3:

\[
\min_{\hat{f}} \{ \| H\hat{f} - g \|^2_2 + \lambda \| L^3f \|_1 \}
\]

**third order TV solutions (m=3)**

- piecewise quadratic with sparse third derivative
- contact point of parabola segments at position of original jump
Estimation of edges in blurred signal

Algorithm
- compute all higher odd order TV restorations
- find common contact points
Compare FOTV and VOTV

- 4th jump is missed
- VOTV restoration still has a smaller $l^2$ error
- results consistent for other blurs and resolutions
Error plot for FOTV and VOTV:
Error plot for FOTV and VOTV:

FOTV performs well in piecewise constant regions.
Error plot for FOTV and VOTV:

VOTV performs much better in piecewise smooth regions
Error plot for FOTV and VOTV:

VOTV1 misses this jump
Application 2: Compressed sensing

Research Project of Undergraduate Student Ashton Feller (ASU)
Application 2: Compressed sensing

- Research Project of Undergraduate Student Ashton Feller (ASU)
- Truncated Fourier coefficients are the given data

\[ g_j = \sum_{i=1}^{N} f_i e^{(i-1)(j-1)} + n, \quad 1 \leq j \leq N_t \ll N, \quad n \text{ is noise} \]
Application 2: Compressed sensing

- Research Project of Undergraduate Student Ashton Feller (ASU)
- Truncated Fourier coefficients are the given data

\[ g_j = \sum_{i=1}^{N} f_i e^{(i-1)(j-1)} + n, \quad 1 \leq j \leq N_t << N, \quad n \text{ is noise} \]

- Under determined system:

\[ g = F_t f \quad F_t \in R^{N_t \times N} + n \]
Application 2: Compressed sensing, no noise

- Case 1: No noise in coefficients

\[
\min_{\hat{f}} \|L^m \hat{f}\|_1 \quad \text{such that} \quad F_t \hat{f} = g
\]
Application 2: Compressed sensing, no noise

Case 1: No noise in coefficients

\[
\min_{\hat{f}} \| L^m \hat{f} \|_1 \quad \text{such that} \quad F_t \hat{f} = g
\]

Keep 50% of low frequency coefficients

First order TV

Variable order TV
Application 2: Compressed sensing, no noise

Keep 30% of low frequency coefficients

First order TV

Variable order TV
Case 2: Noise in coefficients

\[
\min_{\hat{f}} \|L^m \hat{f}\|_1 \quad \text{such that} \quad \|F_t \hat{f} - g\|_2^2 < \sigma
\]
Application 2: Compressed sensing, with noise

- Case 2: Noise in coefficients

\[
\min_{\hat{f}} \| L^m \hat{f} \|_1 \quad \text{such that} \quad \| F_t \hat{f} - g \|_2^2 < \sigma
\]

- Relative error for \( \lambda = 10^{-9} \) and \( N = 80 \)

<table>
<thead>
<tr>
<th>FOTV noise</th>
<th>discard high frequencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^{-5}</td>
<td>0.048 0.08 0.157 0.23</td>
</tr>
<tr>
<td>10^{-2}</td>
<td>0.048 0.079 0.156 0.23</td>
</tr>
<tr>
<td>10^{-1}</td>
<td>0.045 0.07 0.151 0.23</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VOTV noise</th>
<th>discard high frequencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^{-5}</td>
<td>3.8 \times 10^{-6} 2.8 \times 10^{-6} 0.119 0.17</td>
</tr>
<tr>
<td>10^{-2}</td>
<td>3.7 \times 10^{-3} 2.4 \times 10^{-3} 0.119 NaN</td>
</tr>
<tr>
<td>10^{-1}</td>
<td>2.8 \times 10^{-2} 2.4 \times 10^{-2} 0.112 NaN</td>
</tr>
</tbody>
</table>
Iterative method for edge detection

- **Problem:** Missed jumps cause problems
Iterative method for edge detection

- **Problem:** Missed jumps cause problems
- **Idea:** Use only FOTV reconstruction for jump detection
Iterative method for edge detection

- **Problem**: Missed jumps cause problems
- **Idea**: Use only FOTV reconstruction for jump detection
- **Iterate**: Obtain a “better” reconstruction using VOTV and iterate
Iterative method for edge detection

▶ Use 30% of the Fourier coefficients

1st iteration (FOTV)
Iterative method for edge detection

- Use 30% of the Fourier coefficients

2nd iteration (VOTV)
Iterative method for edge detection

- Use 30% of the Fourier coefficients

3rd iteration (VOTV)
Iterative method for edge detection

- Use 30% of the Fourier coefficients

5th iteration (VOTV)
Remove matching waveform from jump fct. approx

▶ assume $\hat{g}_j = \hat{f}_j \cdot \hat{h}_j + n$
Remove matching waveform from jump fct. approx

- assume $\hat{g}_j = \hat{f}_j \cdot \hat{h}_j + n$
- for now $n = 0$
Remove matching waveform from jump fct. approx

- assume $\hat{g}_j = \hat{f}_j \cdot \hat{h}_j + n$
- for now $n = 0$
- with $\hat{h}_j = 1$ ⇒ regular edge detection problem
Remove matching waveform from jump fct. approx

- assume $\hat{g}_j = \hat{f}_j \cdot \hat{h}_j + n$
- for now $n = 0$
- with $\hat{h}_j = 1$ ⇒ regular edge detection problem
- use edge detectors in Fourier space $S_\sigma_N[f]$
Edge detector using concentration factors

- Conjugated Fourier sum \( S_N^\sigma[f] = i \sum_{k=-N}^{N} \sigma(\frac{|k|}{N}) \text{sgn}(k) \hat{f}_i e^{ik\pi x} \)
Edge detector using concentration factors

- Conjugated Fourier sum
  \[ S_N^\sigma[f] = i \sum_{k=-N}^{N} \sigma\left(\frac{|k|}{N}\right) \text{sgn}(k) \hat{f}_i e^{ik\pi x} \]
Edge detector using concentration factors

- Conjugated Fourier sum $S_N^\sigma[f] = i \sum_{k=-N}^{N} \sigma\left(\frac{|k|}{N}\right) \text{sgn}(k) \hat{f}_i e^{i k \pi x}$

![Graph showing the edge detector](image_url)
The matching waveform

Trig. con. fac
The matching waveform

Poly. con. fac

Matching waveform

$S_N^\sigma[f]$ 2
The matching waveform

Exp. con. fac

-matching waveform
\[ S_N^\sigma[f] 3 \]
Remove matching waveform by L1 min

- Each Edge detector $S_N^\sigma[f]$ has an function independent matching waveform $W_N^\sigma$ in Fourier space
- Because testfunction is in physical space, we approximate this waveform
- by computing a $S_N^\sigma[s]$, where $s(x)$ is the saw-tooth function

$$\min \| W_N^\sigma \cdot \hat{x} - \hat{g} \cdot "edge filter" \|_2 + \lambda \|x\|_1$$
Result

Trig. con. fac
Blur 0 c. factor 1

\[ f(x) \]

\[ S_N^\sigma[f] \]

\[ \text{min L1 solution} \]
Result

Poly. con. fac
Blur 0 c. factor 2

![Graph showing the function $f(x)$, its square $\sigma^2[f]$, and the minimum L1 solution.](image)
Result

Exp. con. fac
Blur 0 c. factor 3

![Graph showing the results with curves and labels](image)

- $f(x)$
- $S_N^\sigma[f]$ 3
- min L1 solution
Edge detection in blurred signal

Easy extension to blurred signals

$$\min ||W_N^\sigma \cdot \hat{x} \cdot \hat{h} - \hat{g} \cdot "edge\ filter" ||_2 + \lambda ||x||_1$$
Result for Gauss blur

Trig. con. fac
Blur 1 c. factor 1

![Graph showing Gauss blur result with trigonometric function and L1 solution.](image)
Result for Gauss blurr

Poly. con. fac
Blur 1 c. factor 2

![Graph showing the function f(x) and the minimum L1 solution.](image)
Result for Gauss blur

Exp. con. fac
Blur 1 c. factor 3

-2 -1 0 1 2

f(x)  min L1 solution

-2 -1 0 1 2 3 4

x
Result for OOF blur

Trig. con. fac
Blur 2 c. factor 1

![](image)

- `f(x)`
- `min L1 solution`
Result for OOF blurr

Poly. con. fac
Blur 2 c. factor 2

\[
\begin{array}{c}
\text{f(x)} \\
\text{min L1 solution}
\end{array}
\]
Result for OOF blurr

Exp. con. fac
Blur 2 c. factor 3

![Graph showing f(x) and min L1 solution with x-axis from -4 to 4 and y-axis from -2 to 2.](image)
Summary and Future Work

▶ Summary

▶ Replace derivative operator in FOTV by linear **variable order**
  difference operators
▶ Use 1st order at jump, use higher order away from jumps
▶ Use **HOTV** to estimate **jump** locations from blurred data or
  Fourier coefficients
▶ Use **standard numerical minimization**
▶ Combine deconvolution and edge detection
▶ VOTV restorations have smaller $l^2$ error as compared to FOTV

▶ Future Work

▶ Extensions: 2D (and 3D), robustness under noise
▶ Application: Positron Emission Tomography (PET) and MRI
  scans
▶ Theoretical: Analyse connection between contact points of
  polynomials and jumps
Summary and Future Work

▶ Summary

► Replace derivative operator in FOTV by linear variable order difference operators
► Use 1st order at jump, use higher order away from jumps
► Use HOTV to estimate jump locations from blurred data or Fourier coefficients
► Use standard numerical minimization
► Combine deconvolution and edge detection
► VOTV restorations have smaller $l^2$ error as compared to FOTV

▶ Future Work

► Extensions: 2D (and 3D), robustness under noise
► Application: Positron Emission Tomography (PET) and MRI scans
► Theoretical: Analyse connection between contact points of polynomials and jumps
Example 2: Error plot
Example 3: Out of focus
Example 3: Out of focus restoration