Matlab Imaging Algorithms: Image Reconstruction, Restoration, and Alignment, with a Focus in Tomography. (Version 2.4)

Toby Sanders
email: toby.sanders@asu.edu
School of Mathematical and Statistical Sciences,
Arizona State University, Tempe, AZ.

Abstract

This document is a guide for several MATLAB imaging code, which can be used for image reconstruction, denoising, alignment, etc. There is a general focus on tomographic reconstruction and alignment, as well as ℓ_1 regularization algorithms.

Contents

1 Solvers for Signal, Image, and Volume Reconstruction **1**
1.1 Higher order TV based ℓ_1 Regularization 1
1.2 HOTV based ℓ_1-ℓ_1 Regularization 4
1.3 General ℓ_1 regularization algorithm 4
1.4 HOTV ℓ_2 Tikhonov Regularization 6
1.5 Tomographic reconstruction with SIRT 6
1.6 Discrete tomographic reconstruction with DART 7

2 Image Alignment **7**
2.1 Basic Cross Correlation 7
2.2 Center of mass alignment for tilt series alignment 8

3 Other utilities and codes potentially of interest **10**

1 Solvers for Signal, Image, and Volume Reconstruction

1.1 Higher order TV based ℓ_1 Regularization

- **Code name:** HOTV3D
- **Demos:** see demo_L1optimization_simple.m, demo_tomo.m, demo_MHOTV.m, demo_inpaint.m, and demo_1D_L1.m
- **References:** [2, 5, 4, 1, 3]
- **Description:** An iterative solver for higher order total variation ℓ_1 regularization minimization for inverse problems and denoising. That is given a data vector b, a map A that maps the unknown signal f to b, the algorithm seeks to find

$$\min_f \frac{\mu}{2}\|Af-b\|_2^2 + \|D^k f\|_1$$

MATLAB comments:

```
% Modifications by Toby Sanders @ASU
% School of Math & Stat Sciences
% Last update: 11/01/2018
%
%
% This code has been modified to solve l1 penalty problems with
% higher order TV operators.  Several small bugs and notation
% changes have been made as well.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%         Problem Description       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [U, out] = HOTV3D(A,b,n,opts)
%
% Motivation is to find:
%
%               min_f { mu/2*||Au - b||_2^2 + ||D^k u||_1 }
%
% where D^k is kth order finite difference.
% Multiscale finite differences D^k can also be used.
% To see how to modify these settings read the file "check_HOTV_opts.m"
%
% The problem is modified using variable splitting and this algorithm
% works with the following augmented Lagrangian function:
%
%      min_{u,w} {mu/2 ||Au - b||_2^2 + beta/2 ||D^k u - w ||_2^2
%               + ||w||_1 - (delta , Au - b ) - (sigma , D^k u - w) }
%
% delta and sigma are Lagrange multipliers
% Algorithm uses alternating direction minimization over f and w.
% by default delta = 0.
%
% This algorithm was originally authored by Chengbo Li at Rice University
% as a TV solver called TVAL3.
% original code and description can be found here:
% http://www.caam.rice.edu/~optimization/L1/TVAL3/
%
% Inputs:
%   A: matrix operator as either a matrix or function handle
%   b: data values in vector form
%   n: image/ signal dimensions in vector format
%   opts: structure containing input parameters,
%       see function check_HOTV_opts.m for these
%
% Outputs:
%   U: reconstructed signal
%   out: output numerics
```

- **Important fields in the opts structure:**

mu .... primary parameter balancing the data and regularization terms.

order .... order (k) of the finite difference operator used in the regularization term, can be set to any real number ≥ 0. Generally recommended between 1 and 3.

levels .... Set to an integer generally between 1 and 4. For values greater than 1 a multiscale regularization approach is implemented [5], which is similar to using wavelets.

scale_A .... set to true or false. This option rescales A and b by dividing by the ℓ_2 norm of A, which one may find greatly simplifies the selection of μ. Generally recommended to set to true (default).

scale_mu .... set to true or false. Similar to the "scale_A" option, μ may also be scaled based on the magnitude of b.

L1type .... set to 'isotropic' to solve the isotropic model or 'anisotropic' to solve the anisotropic model. Isotropic is generally preferred.

iter .... maximum number of total iterations

tol .... convergence tolerance to terminate the algorithm and return the solution

nonneg .... if set to true, problem is solved under the constraint $f \geq 0$

wrap_shrink ... set to to false unless signal is periodic.

** Other options are available, a full description of each is given in the "check_HOTV_opts.m" file. It is generally recommended to just use default values.*

**For automatic HOTV reconstruction of tomographic data see the code HOTV3D_tomo.m.*

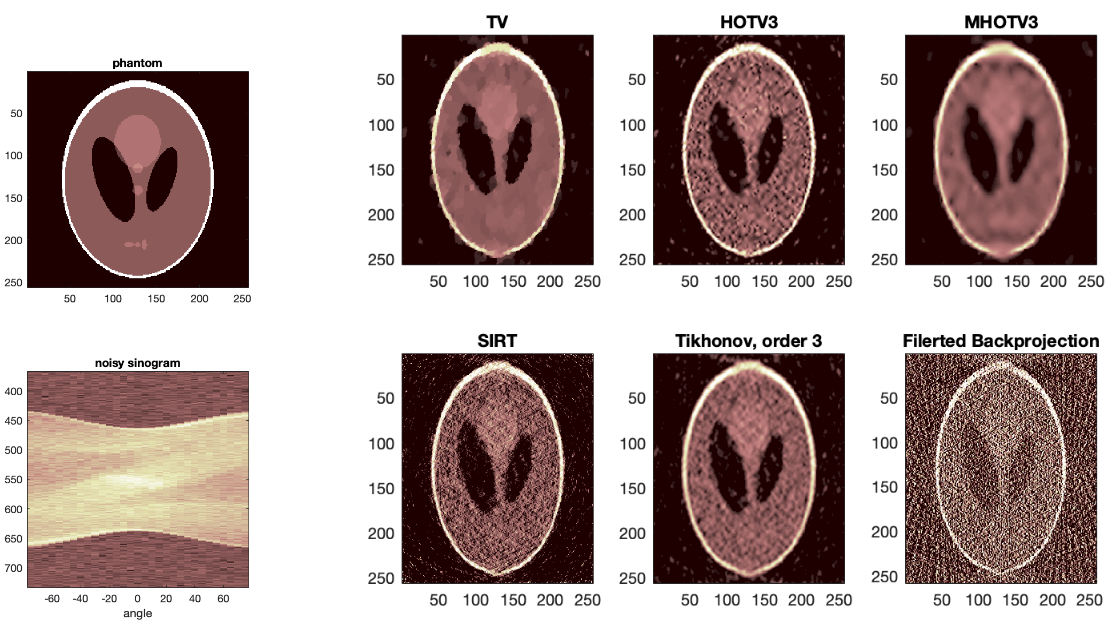

Figure 1: Tomographic reconstruction of phantom image.

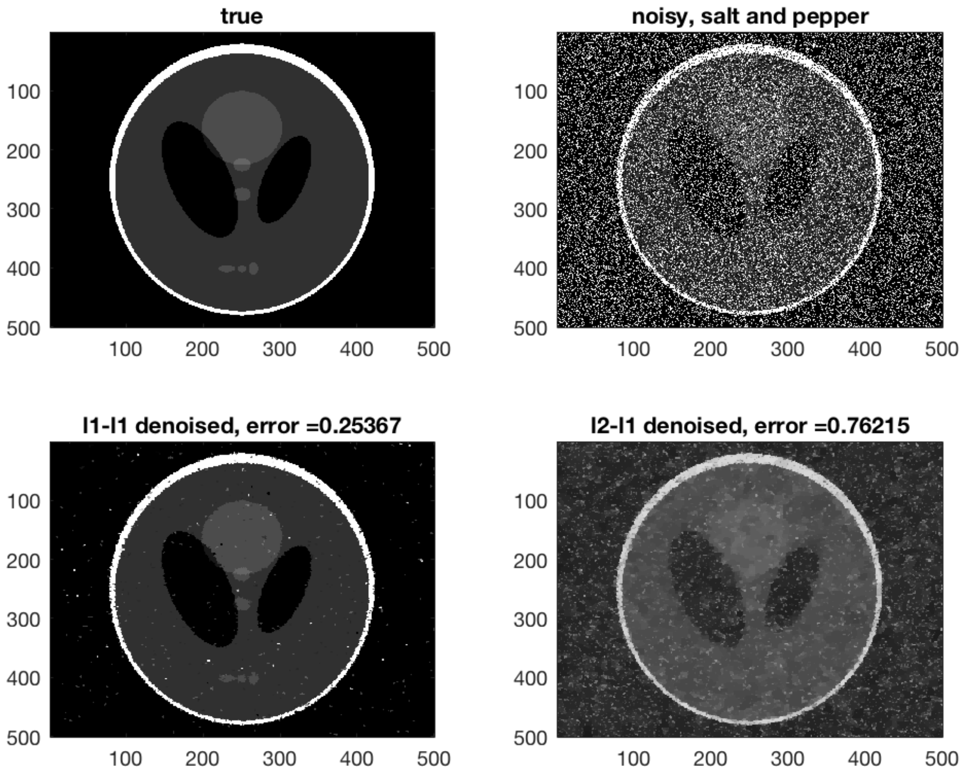

Figure 2: An example of ℓ_1 regularization image denoising from salt and pepper noise, comparing the results when the data fitting term is ℓ_1 or ℓ_2 norm.

1.2 HOTV based ℓ_1-ℓ_1 Regularization

- **Code name:** HOTV3D_L1L1
- **Demo:** see demo_L1L1.m
- **Description:** An iterative solver for HOTV ℓ_1 regularization minimization for inverse problems, where the data fitting metric is given as the ℓ_1 norm. That is given a data vector b, a map A that maps the unknown signal f to b, the algorithm tries to find

  $$\min_f \mu\|Af - b\|_1 + \|D^k f\|_1.$$

  All of the options are more or less the same as well HOTV3D. For more details, look in the description given with in the code.

1.3 General ℓ_1 regularization algorithm

- **Code name:** l1optimo
- **Demo:** see demo_MHOTV.m

- **Description:** An iterative solver for general ℓ_1 regularization minimization for inverse problems and denoising.

```
% function [U, out] = l1optimo(A,b,n,opts)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%         Problem Description       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Motivation is to find:

%               min_f { mu/2*||Af - b||_2^2 + ||D f||_1 }


% This algorithm is essentially the same as HOTV3D, however,
% the user must specify the regularization operator "D".  To do this,
% the user should input into the opts structure the options opts.D and
% opts.Dt, where opts.D is the regularization transform, and opts.Dt is
% the adjoint of opts.D.
```

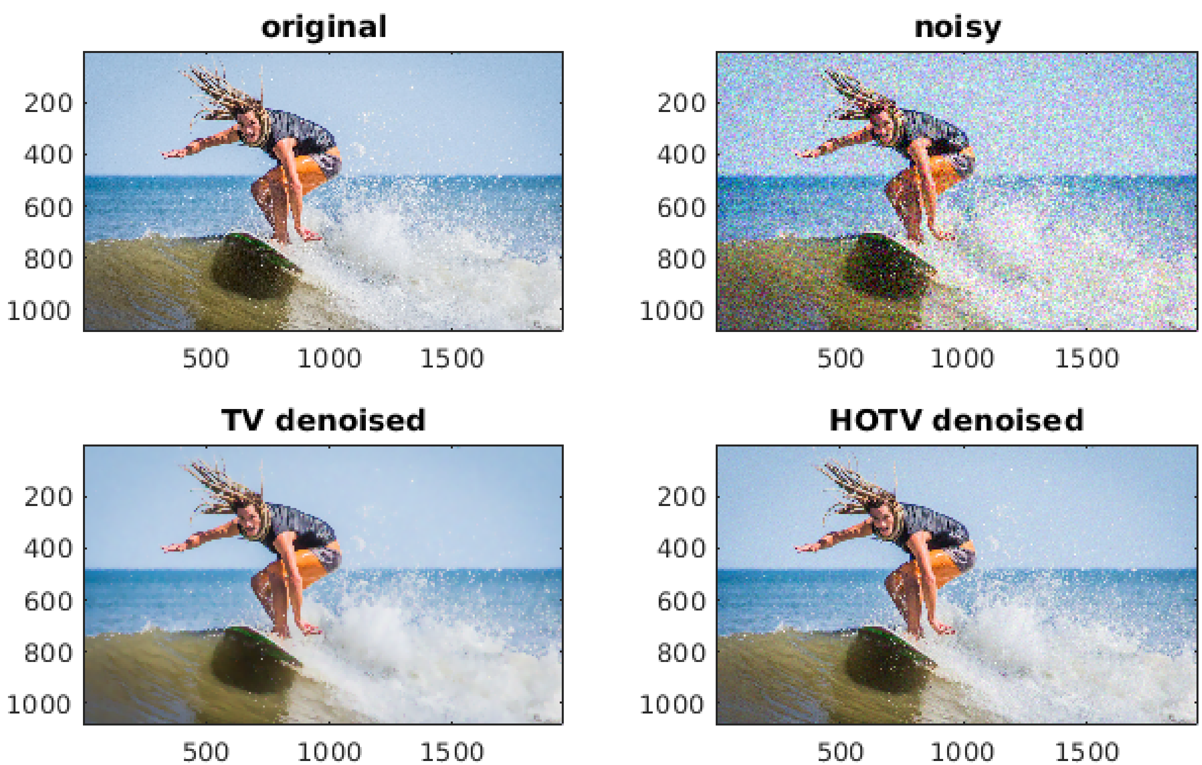

Figure 3: Denoising with ℓ_1 regularization.

1.4 HOTV ℓ_2 Tikhonov Regularization

- **Code name:** Tikhonov
- **Demo name:** see demo_tikhonov.m
- **Description:** An iterative solver for higher order total variation ℓ_2 regularization minimization for inverse problems and denoising. That is given a data vector b, a map A that maps the unknown signal f to b, the algorithm seeks to find

  $$\min_f \mu\|Af - b\|_2^2 + \|D^k f\|_2^2.$$

  All of the options are more or less the same as well HOTV3D. For more details, look in the description given with in the code.

MATLAB comments:

```
% These functions solve
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% min_x    mu*||Ax-b||^2 + ||Dx||^2
% subject to optional inequality constaints
% D is a finite difference operator
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% options are more or less the same as HOTV3D, see check_hotv_opts or the
% users guide.

% Fields in the opts structure (defaults are assigned for empty fields):
% order - order of the finite difference reg. operator, D
% iter - maximum number of iterations for CG
% mu - regularization parameter (see formulation above)
% tol - convergence tolerance for CG
% levels - default is 1, but for higher integers it uses a multiscale
% operators for D
```

1.5 Tomographic reconstruction with SIRT

- **Code name:** SIRT
- **Demo name:** see demo_tomo.m
- **References:** [7]
- **Description:** Simultaneous iterative reconstruction technique (SIRT) for tomographic reconstruction, which works by the general gradient decent method. MATLAB comments:

```
% function [x,out] = SIRT(stack,angles,recsize,iter,minc,maxc)
%
%DESCRIPTION:
    %this function performs the SIRT algorithm for tomographic reconstruction.
%
%NOTATION:
```

```
    % function [x,out] = SIRT(stack,angles,recsize,iterations,minc,maxc);
%
%INPUTS:
    % stack - the tilt series, where it is assumed the tilt axis is horizontal
    %   and located at the middle of the stack
    % angles - a vector holding the projection angles of the stack, in order,
    %   in degrees
    % recsize - the dimension of the reconstruction
    % iter - the number of SIRT iterations
    % minc - minimum density constraint, e.g. U>=0.
    % maxc - maximum density constraint, e.g. U<=1.
    %
    % default values are used for recsize, iter, and minc if they are not
    % specified, therefore one may simply input "sirtden(stack,angles)."
    % the recsize will be set to the detector count, i.e. size(stack,1)
    % 50 iterations is default, and no density constraint is used if it is not
    % specified
%
%OUTPUT:
    % x - the reconstruction from the input tilt series and other input
        %parameters
    % out - additional outputs
```

- **Additional Notes:** For automated reconstruction of a 3D volume see the algorithm *sirtauto.m*.

1.6 Discrete tomographic reconstruction with DART

I have removed the description of this code, though it is still included in the package. Please see previous versions or email me directly for information.

2 Image Alignment

2.1 Basic Cross Correlation

- **Code names:** cross_corr.m and cross_corr_pad.m
- **Demo name:** see demo_align.m
- **Description:** these algorithms perform 2D image alignment from an input 3D stack of images by cross correlation. "cross_corr.m" maintains the original image size by simply rotating or circular shifting the image and "cross_corr_pad.m" pads the stack with zeros based on the minimum and maximum shifts. MATLAB comments:

```
%DESCRIPTION
    %This function performs cross correlation of an input stack of images.
    %
%NOTATION
    %[stacknew,shifts]=cross_corr(stack,startslice);
    %
```

```
%INPUTS
    %stack - 3D matrix assumed to be a stack of images to be cross
        %correlated
    %startslice - the reference image that all slices are consecutively
    %aligned to.  This image will not move
    %
%OUTPUTS
    %stack- 3D stack of images after cross correlation
    %shifts - shifts that were applied to each image
    %
%NOTE: This function automatically uses a window to improve alignment.  To
    %remove the windowing, set pwr=0.
```

2.2 Center of mass alignment for tilt series alignment

- **Code names:** COM_align.m
- **Demo name:** see demo_align.m
- **References:** [6]
- **Description:** Center of mass alignment for electron tomography data. See `http://ascimaging.springeropen.com/articles/10.1186/s40679-015-0005-7` for details. MATLAB comments:

```
%  Center of mass alignment for electron tomography data
%
% function [stack,shifts,usables] = COM_align(stack,angles,ratio,s)
%
% Inputs: stack is 3-D matrix containing tomography data
% angles are the angles in degrees
% ratio is the ratio of slices to be used for each projection
% s is the number of projections used as a sequence to determine rigid
% alignment.  Setting s to be the number of angles is equivalent to the
% alignment method published here:
% http://ascimaging.springeropen.com/articles/10.1186/s40679-015-0005-7
%
%
%
% The ratio variable:
% should be input as a number between 0 and 1, where the user rates the
% consitency of the stacks through the slices.  From our experience, very
% small values of s are better.  If the user does not input a
% rating, the default value of .1 is used.  Setting ratio
% to perfect 1 would mean that through the slices, there is significant amount
% of mass in all of the slices, and very little mass moves in or out as the
% tilt angle changes.  A lower ratio would mean otherwise.
% This rating is then used to determine how many slices to use for the
% center of mass alignment.  For example, if you set the rating to .2, then
```

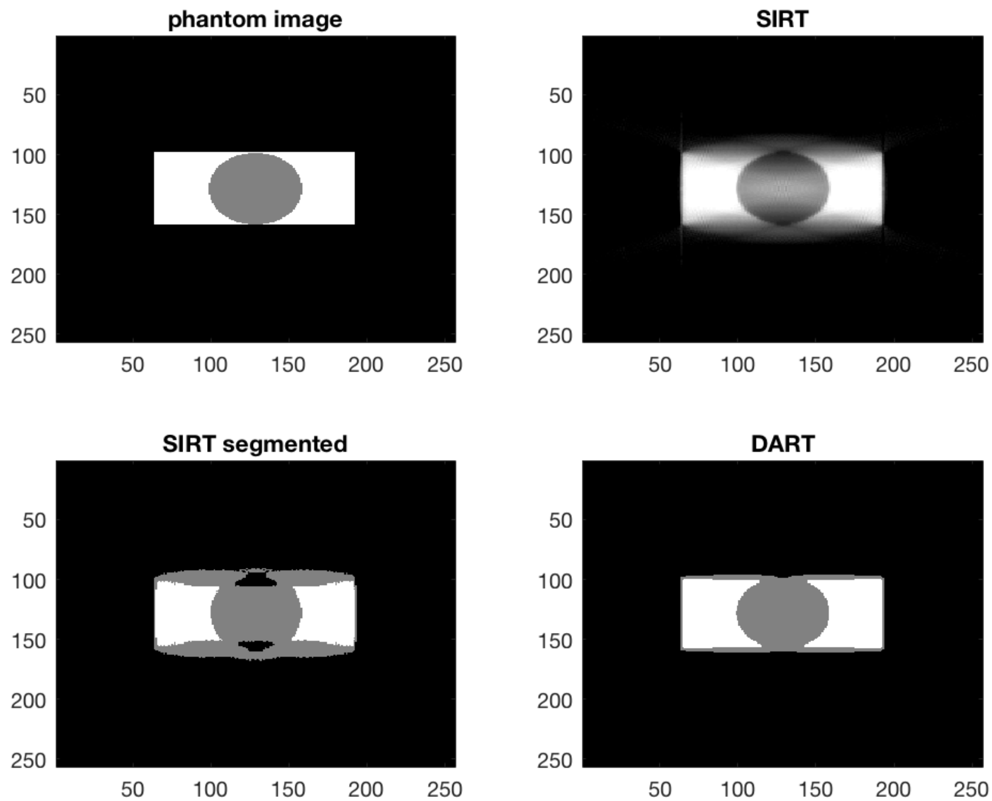

Figure 4: Tomographic reconstruction with DART.

```
% only the best 20 percent of the slices will be used in the alignment step.
% That way, if the the stack has some slices with very little mass, yet
% mass moves into these slices at high angles, these slices may be ignored
% for the aligment.
```

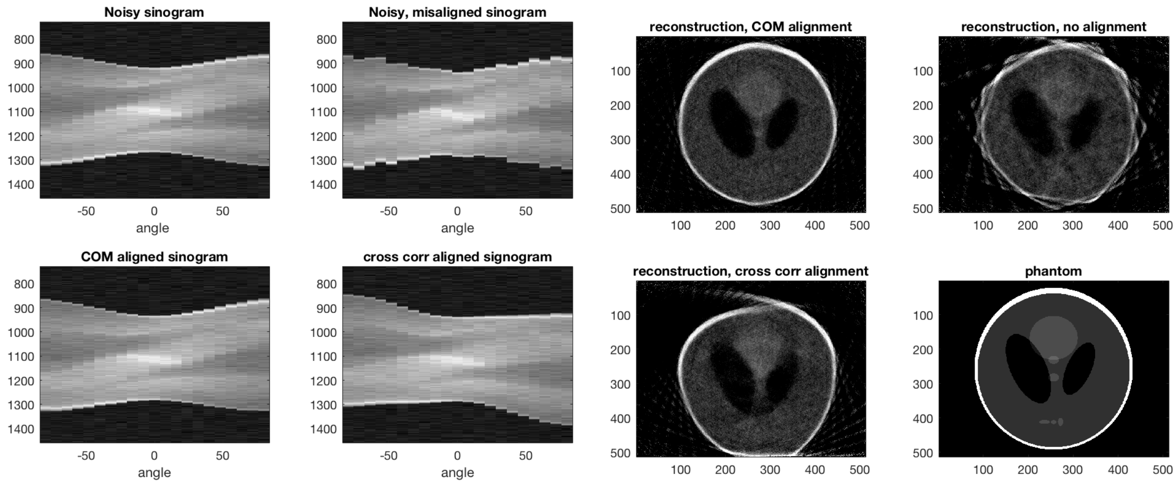

Figure 5: Tomographic alignment and reconstruction (SIRT).

3 Other utilities and codes potentially of interest

**See the comments within each code for details.*

- add_Wnoise.m - add Gaussian white noise to data with specified SNR.
- Tikhonov.m - a solver for inverse problems that uses Tikhonov regularization. It is similar to HOTV3D, except the regularization uses the ℓ_2 norm.
- moviestack.m - used to visualize a 3D volume by passing through 2D cross-sections.
- radonmatrix.m - builds a sparse projection matrix for tomography based off of input geometry.
- corrx_global.m - a horizontal image alignment procedure for a tilt series based off of the conservation of mass.
- manualalign.m - a utility that can be used to manually align images.
- forward_proj.m - builds a tilt series given input projection angles and 3D volume.

References

[1] Tony Chan, Antonio Marquina, and Pep Mulet. High-order total variation-based image restoration. *SIAM J. Sci. Comput.*, 22(2):503–516, 2000.

[2] Chengbo Li. *An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing.* PhD thesis, Rice University, 2009.

[3] Chengbo Li, Wotao Yin, Hong Jiang, and Yin Zhang. An efficient augmented lagrangian method with applications to total variation minimization. *Comput. Optim. Appl.*, 56(3):507–530, 2013.

[4] Toby Sanders, Anne Gelb, Rodrigo B Platte, Ilke Arslan, and Kai Landskron. Recovering fine details from under-resolved electron tomography data using higher order total variation l1 regularization. *Ultramicroscopy*, 174:97–105, 2017.

[5] Toby Sanders and Rodrigo B Platte. Multiscale higher-order tv operators for l1 regularization. *Advanced structural and chemical imaging*, 4(1):12, 2018.

[6] Toby Sanders, Micah Prange, Cem Akatay, and Peter Binev. Physically motivated global alignment method for electron tomography. *Advanced Structural and Chemical Imaging*, 1(1):1–11, 2015.

[7] Jeannot Trampert and Jean-Jacques Leveque. Simultaneous iterative reconstruction technique: Physical interpretation based on the generalized least squares solution. *J. Geophys. Res.*, 95(12):553–9, 1990.