USC-SIPI REPORT #414

Low-Rank Modeling of Local k-Space Neighborhoods (LORAKS): Implementation and Examples for Reproducible Research

by

Justin P. Haldar

April 2014

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Viterbi School of Engineering
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Suite 400
Los Angeles, CA 90089-2564 U.S.A.

Low-Rank Modeling of Local k-Space Neighborhoods (LORAKS): Implementation and Examples for Reproducible Research

Justin P. Haldar*

Contents

1 Overview

Our recent work [1–4] has introduced a new signal processing framework that enables the recovery of magnetic resonance (MR) images from undersampled and/or noisy Fourier data, based on the assumptions that the true image has smooth phase and/or limited spatial support. This framework, named LORAKS, relies on embedding the Fourier data from such images into high-dimensional Hankel-like matrices, which we showed [1] would have approximately low rank when the phase/support assumptions were satisfied. The fact that these matrices have low rank implies that they have relatively few degrees of freedom relative to their number of entries, and hence can potentially be recovered from subsampled data [5]. In contrast to conventional reconstruction methods that rely on assumptions of smooth image phase and/or limited spatial support [6–13], LORAKS does not place strict requirements on the sampling scheme used for data acquisition, and high-quality reconstructions can be obtained from both highly-structured and highly-unstructured Fourier sampling schemes. In contrast to other recent MR image reconstruction methods based on low-rank modeling [14–33], LORAKS can be used with relatively low-dimensional single-channel, single-contrast, single-timepoint MR imaging data. We have also previously demonstrated [1] that LORAKS has distinct features and advantages relative to sparsity-based MR image reconstruction methods [34,35] (which have been receiving substantial attention over the past several years), which suggests that phase and support constraints can be used in combination with sparsity constraints for even better performance.

In the spirit of reproducible research [36], this technical report (and corresponding supplementary material available for download at `http://mr.usc.edu/download/LORAKS/`) provides a MATLAB implementation of a LORAKS-based reconstruction algorithm proposed in our previous work [1], and presents several application examples that are not found in the existing literature [1–4] (including an example outside the context of MR imaging).

*Signal and Image Processing Institute, Departments of Electrical Engineering and Biomedical Engineering, University of Southern California, Los Angeles, CA 90089. This work has been partially supported by NSF CAREER award CCF-1350563.

This report is organized as follows: section 2 describes the LORAKS optimization problem. This description is presented at a high-level, as suitable for applications. Please refer to [1] for a more detailed description of the approach and its underlying theory. Section 3 describes the MATLAB functions we've included for performing LORAKS-based reconstruction. Section 4 describes the demos we've included for LORAKS-based reconstruction that coarsely mimic undersampled MR data. Section 5 describes a demo we've included for LORAKS in the context of 1D multiband signal reconstruction from sparsely sampled data.

2 LORAKS

The LORAKS framework is based on the linear mapping of a low-dimensional vector $\mathbf{k} \in \mathbb{C}^S$ of Cartesian-sampled spatial Fourier transform (also known as "*k*-space") coefficients into three distinct higher-dimensional matrices $\mathbf{C}$, $\mathbf{S}$, and $\mathbf{G}$. This mapping is achieved according to

$$\mathbf{C} = \mathcal{P}_{\mathbf{C}}\left(\mathbf{k}\right), \tag{1}$$

$$\mathbf{S} = \mathcal{P}_{\mathbf{S}}\left(\mathbf{k}\right), \tag{2}$$

and

$$\mathbf{G} = \mathcal{P}_{\mathbf{G}}\left(\mathbf{k}\right), \tag{3}$$

where $\mathcal{P}_{\mathbf{C}}\left(\cdot\right)$, $\mathcal{P}_{\mathbf{S}}\left(\cdot\right)$, and $\mathcal{P}_{\mathbf{G}}\left(\cdot\right)$ are linear operators that: (1) take data from local neighborhoods in *k*-space; (2) assemble the data from each neighborhood into a vector; and (3) stack the vectors obtained from multiple neighborhoods to form a matrix. Details are described in [1]. It has been shown in [1] that the matrix $\mathbf{C}$ will have approximately low-rank whenever the inverse Fourier transform of $\mathbf{k}$ is support-limited, and the matrices $\mathbf{G}$ and $\mathbf{S}$ will be rank deficient or have approximately low-rank whenever the inverse Fourier transform of $\mathbf{k}$ has slowly-varying spatial phase.

The LORAKS-based reconstruction approach used in [1] is designed for the case where $\mathbf{k}$ is noisy or only partially observed. In such cases, the LORAKS-based rank constraints can be used to regularize the reconstruction. In particular, [1] solves for an estimate of the fully-sampled *k*-space data $\hat{\mathbf{k}}$ by solving

$$\hat{\mathbf{k}} = \arg\min_{\mathbf{k}\in\mathbb{C}^S} \left\|\mathbf{F}\mathbf{k} - \mathbf{d}\right\|_{\ell_2}^2 + \lambda_{\mathbf{C}} J_{\mathbf{C}}\left(\mathcal{P}_{\mathbf{C}}\left(\mathbf{k}\right)\right) + \lambda_{\mathbf{G}} J_{\mathbf{G}}\left(\mathcal{P}_{\mathbf{G}}\left(\mathbf{k}\right)\right) + \lambda_{\mathbf{S}} J_{\mathbf{S}}\left(\mathcal{P}_{\mathbf{S}}\left(\mathbf{k}\right)\right), \tag{4}$$

where $\mathbf{d}$ is the vector of measured *k*-space data; $\mathbf{F}$ is a subsampling matrix (formed by deleting rows from the $S \times S$ identity matrix) that relates the measured *k*-space data to the subsampled *k*-space data; $\lambda_{\mathbf{C}}$, $\lambda_{\mathbf{G}}$, and $\lambda_{\mathbf{S}}$ are regularization parameters; and $J_{\mathbf{C}}(\cdot)$, $J_{\mathbf{G}}(\cdot)$, and $J_{\mathbf{S}}(\cdot)$ are regularization penalties that penalize matrices that have ranks higher than user-defined rank constraints of $r_{\mathbf{C}}$, $r_{\mathbf{G}}$, and $r_{\mathbf{S}}$, respectively. Specifically, these penalty functions are all defined in a similar way, according to

$$J_{\mathbf{X}}(\mathbf{X}) = \sum_{k>r_{\mathbf{X}}} \sigma_k^2, \tag{5}$$

where the σ_k values are the singular values of the matrix $\mathbf{X}$. This can be viewed as a penalty on the Frobenius norm of the residual we would be left with after making an optimal rank-$r_{\mathbf{X}}$ approximation of the matrix $\mathbf{X}$. The code provided along with this report implements a simple majorize-minimize algorithm to find a stationary point of (4).

3 Software

The supplementary MATLAB code contains two main LORAKS-related functions, `generate_LORAKS_operators.m` and `LORAKS.m` (which are described below), as well as two MATLAB demo scripts `MRI_demo.m` and `MB_demo.m` (described in Sections 4 and 5, respectively).

The `generate_LORAKS_operators.m` performs precomputations that will be used for solving (4). Specifically, the function takes as input the size of the Nyquist-sampled k-space grid and the radius for the LORAKS *k*-space neighborhood, and uses this information to generate the linear operators $\mathcal{P}_{\mathbf{C}}\left(\cdot\right)$, $\mathcal{P}_{\mathbf{G}}\left(\cdot\right)$, and $\mathcal{P}_{\mathbf{S}}\left(\cdot\right)$; the adjoint operators $\mathcal{P}_{\mathbf{C}}^{\star}\left(\cdot\right)$, $\mathcal{P}_{\mathbf{G}}^{\star}\left(\cdot\right)$, and $\mathcal{P}_{\mathbf{S}}^{\star}\left(\cdot\right)$; data structures for representing the diagonal-matrix representations of the self-adjoint

"normal" operators $\mathcal{P}_{\mathbf{C}}^{\star}(\mathcal{P}_{\mathbf{C}}(\cdot))$, $\mathcal{P}_{\mathbf{G}}^{\star}(\mathcal{P}_{\mathbf{G}}(\cdot))$, $\mathcal{P}_{\mathbf{S}}^{\star}(\mathcal{P}_{\mathbf{S}}(\cdot))$; and variables describing the sizes of the $\mathbf{C}$, $\mathbf{G}$, and $\mathbf{S}$ matrices. A more detailed specification of the function inputs and outputs is provided in the MATLAB help for this function, which is partially reproduced in Section 3.1.

The `LORAKS.m` function uses the precomputations obtained from `generate_LORAKS_operators.m` to actually solve (4). It requires inputs specifying the measured k-space data; the k-space sampling mask; the regularization parameters $\lambda_{\mathbf{C}}$, $\lambda_{\mathbf{G}}$, and $\lambda_{\mathbf{S}}$; the rank constraints $r_{\mathbf{C}}$, $r_{\mathbf{G}}$, and $r_{\mathbf{S}}$; the precomputed results obtained from `generate_LORAKS_operators.m`; and stopping criteria that will be used to determine when to halt the iterative algorithm. A more detailed specification of the function inputs and outputs is provided in the MATLAB help for this function, which is partially reproduced in Section 3.2.

Note that the `LORAKS.m` depends on the `lansvd.m` function from the PROPACK software [37] to compute fast partial SVDs. The original version of PROPACK can be obtained from `http://soi.stanford.edu/~rmunk/PROPACK/`. However, we recommend that you use the version of PROPACK distributed with the SVT software package (available for download at `http://svt.stanford.edu/code.html`), which includes precompiled binaries for most architectures and operating systems, is able to handle complex-valued numbers (necessary for $\mathbf{C}$-based LORAKS), and has been modified to maintain higher numerical accuracy.

3.1 generate_LORAKS_operators.m

```
function [P_C, Ph_C, P_G, Ph_G, P_S, Ph_S, cc, gg, ss, sizeC, sizeG, sizeS] = ...
    generate_LORAKS_operators(N1, N2, R)
% Inputs:
%     N1 x N2: The size of the fully-sampled k-space matrix
%     R:  The k-space radius used to construct LORAKS neighborhoods
% Outputs:
%     P_C, P_G, P_S: Operators to convert fully k-space data into the
%                    C, G, and S matrices, respectively
%     Ph_C, Ph_G, Ph_S: The adjoints of the P_C, P_G, and P_S operators,
%                       respectively
%     cc, gg, ss:  The diagonal entries of the Ph_C*P_C, Ph_G*P_G, Ph_S*P_S
%                  operators, respectively
%     sizeC, sizeG, sizeS: 2x1 vectors describing the sizes of the C, G,
%                          and S matrices, respectively
```

3.2 LORAKS.m

```
function reconstructedK = LORAKS(kdata, kMask, P_C, Ph_C, P_G, Ph_G, P_S, Ph_S, lambda_C, ...
    lambda_G, lambda_S, r_C, r_G, r_S, cc, gg,ss, tol, max_iter)
% Inputs:
%     P_C, P_G, P_S, Ph_C, Ph_G, Ph_S, cc, gg, ss:
%         These are all outputs computed using generate_LORAKS_operators.m.
%         See that function for a detailed description.
%     kdata: An N1 x N2 matrix of Nyquist-sampled k-space data, with the
%            center of k-space assumed to be at location (N1/2+1, N2/2+1)
%            when N1 and N2 are both even, and ((N1+1)/2, (N2+1)/2) when N1
%            and N2 are both odd.  If certain k-space locations from the
%            Nyquist-sampled grid have not been measured, then they should
%            be filled with 0 values.
%     kMask: An N1 x N2 binary sampling mask that takes value 1 at
%            locations where a true Fourier sample is present in the
%            corresponding entry of kdata, and takes value 0 when the
%            corresponding k-space samples are not sampled in kdata.
%     lambda_C, lambda_G, lambda_S:  The regularization parameters used for
%                                    the C, G, and S matrices,
%                                    respectively.
%     r_C, r_G, r_S: The rank constraints imposed for the C, G, and S
```

```
%                 matrices, respectively.
%     tol: A convergence tolerance.  The computation will halt if the
%          relative change (measured in the Euclidean norm) between two
%          successive iteraties is smaller than tol.  [Default: 1e-4]
%     max_iter: The computation will halt if the number of iterations
%               exceeds max_iter [Default: 100}
% Outputs:
%     reconstructedK: An N1 x N2 matrix of the k-space data reconstructed
%                     using the LORAKS algorithm
```

4 MR Image Reconstruction Demos

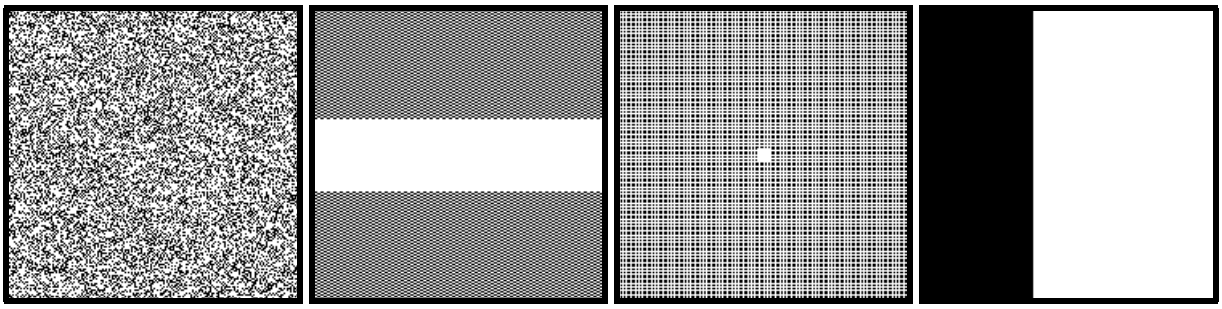
The `MRI_demo.m` MATLAB script implements four different LORAKS-based reconstructions that are intended to mimic various forms of MR image reconstruction from sparsely sampled data. All four reconstructions make use of *k*-space data on a 180×180 Cartesian lattice obtained from the analytic Fourier transform of the Shepp-Logan phantom [38], with simple modifications so that the phantom has non-negligible spatial phase variations. The fully-sampled data is illustrated in Fig. 1.

(a) *k*-Space Data (b) Magnitude Image (c) Phase Image

Figure 1: Illustration of the fully-sampled data. The (a) fully-sampled *k*-space data is shown on a logarithmic-scale. The (b) magnitude and (c) phase of the corresponding image (obtained through inverse Fourier transformation of the *k*-space data) are also shown. All images in this report are shown with a black border for enhanced visualization of information near image borders.

After executing the script, the user is presented with an option to select one of four different *k*-space sampling patterns. These sampling patterns are illustrated in Fig. 2. Sampling pattern 1 (Fig. 2(a)) is an example of "calibrationless" random sampling. Sampling pattern 2 (Fig. 2(b)) is an example of highly-structured sampling, including a substantial fully-sampled "calibration" region at the center of *k*-space, with a checkerboard-like sampling pattern at high frequencies. Sampling pattern 3 (Fig. 2(c)) is another highly-structured sampling pattern, that includes multiple sets of regularly-spaced horizontal and vertical lines through *k*-space, along with a moderately-sized "calibration" region near the center of *k*-space. Sampling pattern 4 (Fig. 2(d)) is an example of the standard 5/8ths partial Fourier sampling pattern. Please note that, with the exception of sampling pattern 1, the remaining sampling patterns have much more structure than is conventionally used for compressed sensing, since the highly-structured nature of the sampling pattern leads to highly-coherent aliasing. All four sampling patterns observe roughly 5/8ths (62.5%) of the *k*-space grid points.

For each of these sampling patterns, the MATLAB code performs simple zero-filled reconstructions, as well as computing reconstructions for $\mathbf{C}$-based LORAKS, $\mathbf{G}$-based LORAKS, and $\mathbf{S}$-based LORAKS, assuming a *k*-space neighborhood radius of $R = 4$ gridpoints. Specifically, $\mathbf{C}$-based LORAKS reconstruction is obtained by having a non-negligible value of $\lambda_{\mathbf{C}}$ and a nontrivial choice of $r_{\mathbf{C}}$ while simultaneously setting $\lambda_{\mathbf{G}} = \lambda_{\mathbf{S}} = 0$ to ensure that the $\mathbf{G}$ and $\mathbf{S}$ matrices have no influence on the reconstruction. $\mathbf{G}$-based and $\mathbf{S}$-based LORAKS reconstructions are obtained in similar ways.



(a) Sampling Pattern 1 (b) Sampling Pattern 2 (c) Sampling Pattern 3 (d) Sampling Pattern 4

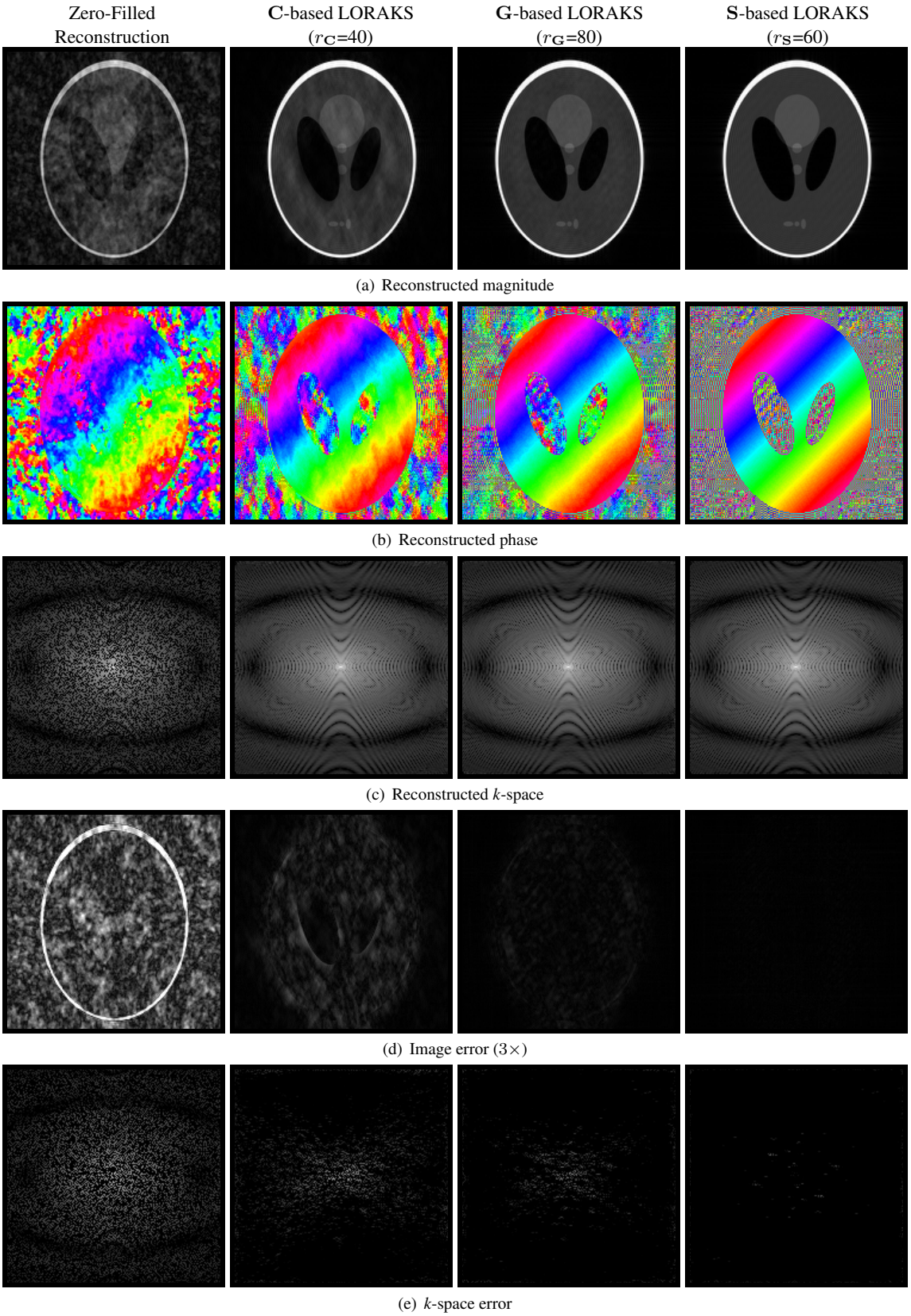
Figure 2: Illustration of the four different k-space sampling masks included with the demo.

While all three rank constraints could be used together for enhanced performance (see [1] for an example) and the MATLAB code is perfectly compatible with the use of multiple simultaneous constraints, use of only one rank-constraint at a time reduces computational complexity, in addition to reducing the number of reconstruction parameters that must be chosen by the user. For the examples described in this report, all reconstructions completed in less than 10 minutes on our workstation. The main computationally-intensive component of our algorithm is the computation of partial SVDs, and substantial speed improvements would be expected if the PROPACK partial SVD algorithm we used in this implementation were replaced with an even more efficient alternative. Similar speed improvements could be obtained by implementing the algorithms outside of MATLAB.

Reconstruction results for sampling patterns 1-4 are presented in Figs. 3–6, respectively. Each figure shows the magnitude and phase of the reconstructed image (using the same colorscales used in Figs. 1(b) and (c), respectively); the reconstructed k-space vector $\hat{\mathbf{k}}$ (using the same logarithmic colorscale used in Fig. 1(a)); the magnitude of the image error (using the same colorscale as Fig. 1(b) after multiplying the error image by a factor of 3); and the magnitude of the reconstruction error in k-space (using the same logarithmic colorscale used in Fig. 1(a)).

In all four results, we observe that the zero-filled reconstruction yields very obvious reconstruction artifacts (as would be expected). Notably, in all four cases, the zero-filled image phase is less-smooth than the true image phase, and the zero-filled image support is less-sparse than the original image support. As a result, it is unsurprising that the use of LORAKS leads to improvements over simple zero-filling in all cases. In some cases, these improvements are very substantial – for example, $\mathbf{S}$-based LORAKS yields almost perfect reconstruction results in all four cases. In other cases, LORAKS yields only moderate improvements. For example, $\mathbf{C}$-based LORAKS for sampling pattern 4 (shown in Fig. 6(b)) yields only minor improvements over the zero-filled reconstruction. This is not very surprising – since only half of k-space is sampled in this case, the reconstruction problem is similar to support-limited extrapolation, which is well-known as a difficult (ill-conditioned) inverse problem [39].

These simulation-based reconstruction results match closely with the results based on real MR data that were shown in [1]: LORAKS reconstructions can be used to improve image reconstruction for a wide variety of sampling schemes; $\mathbf{C}$-based LORAKS yields reconstructions with limited spatial support, while $\mathbf{G}$- and $\mathbf{S}$-based LORAKS yield reconstructions with smooth spatial image phase; and $\mathbf{S}$-based LORAKS is generally more powerful than $\mathbf{C}$- or $\mathbf{G}$-based LORAKS (though both $\mathbf{C}$- and $\mathbf{G}$-based LORAKS can still make substantial improvements in reconstruction quality).



(a) Reconstructed magnitude

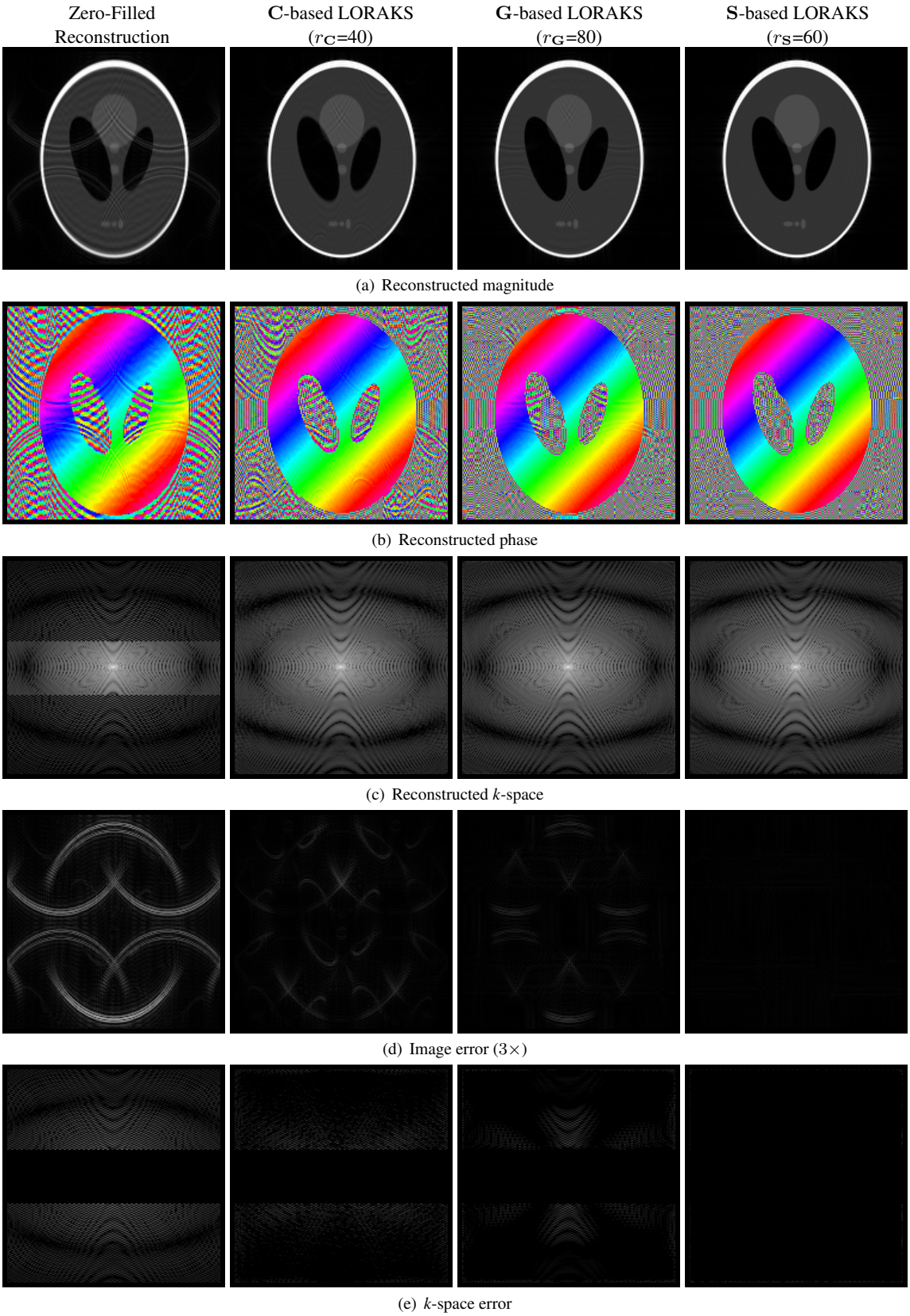
(b) Reconstructed phase

(c) Reconstructed k-space

(d) Image error ($3\times$)

(e) k-space error

Figure 3: Reconstruction results for sampling pattern 1.



(a) Reconstructed magnitude

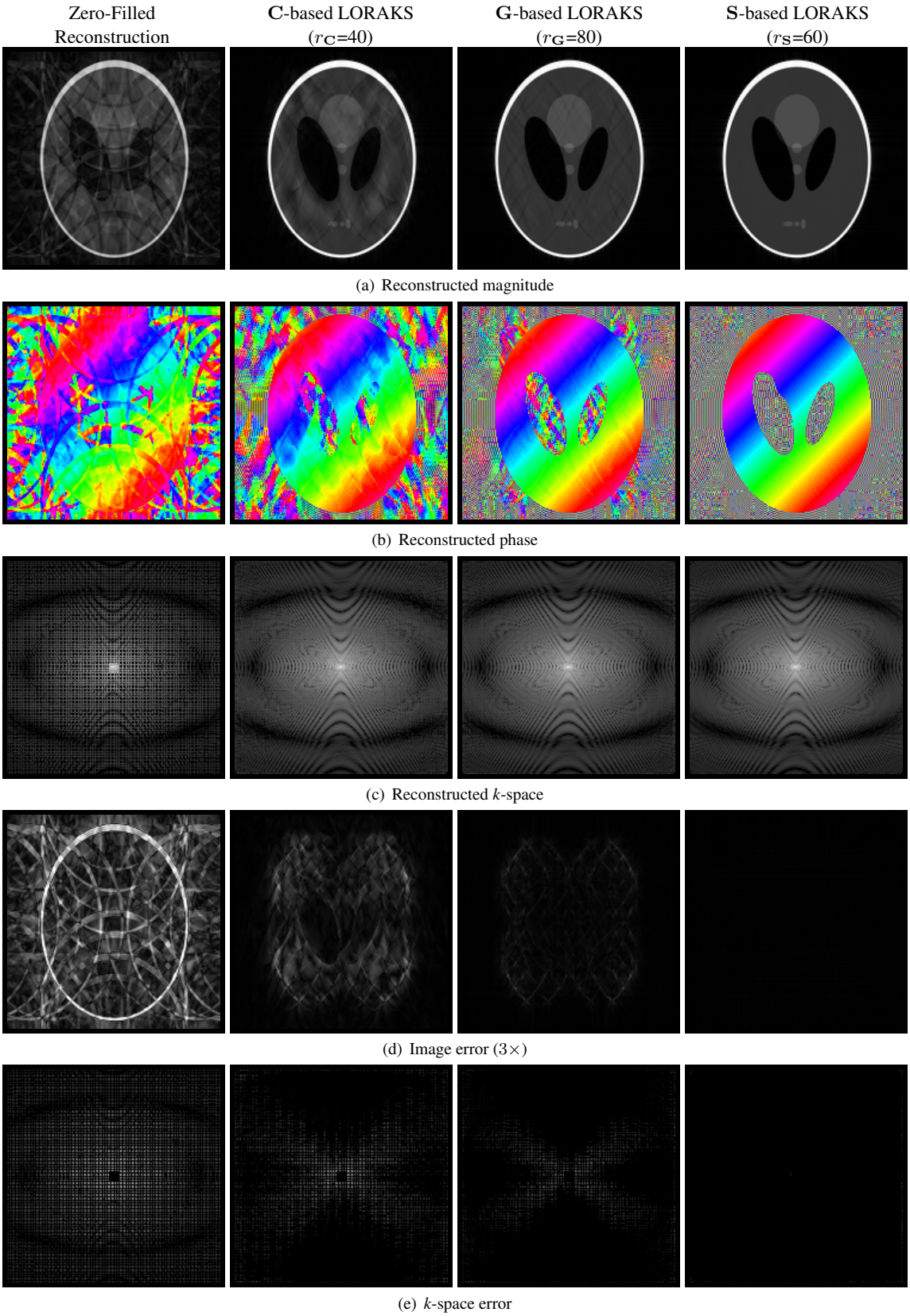
(b) Reconstructed phase

(c) Reconstructed k-space

(d) Image error ($3\times$)

(e) k-space error

Figure 4: Reconstruction results for sampling pattern 2.



(a) Reconstructed magnitude

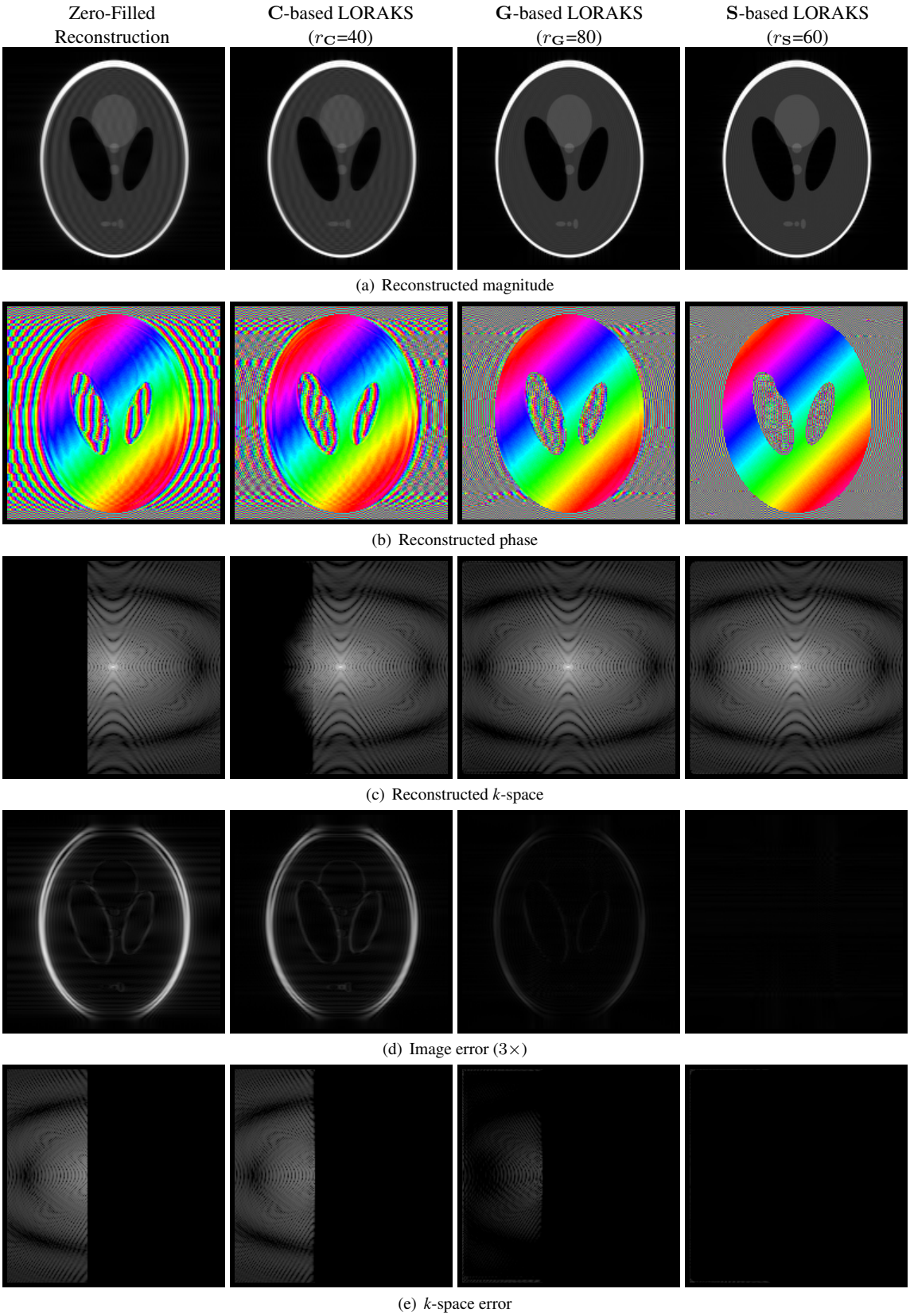
(b) Reconstructed phase

(c) Reconstructed k-space

(d) Image error ($3\times$)

(e) k-space error

Figure 5: Reconstruction results for sampling pattern 3.



(a) Reconstructed magnitude

(b) Reconstructed phase

(c) Reconstructed k-space

(d) Image error ($3\times$)

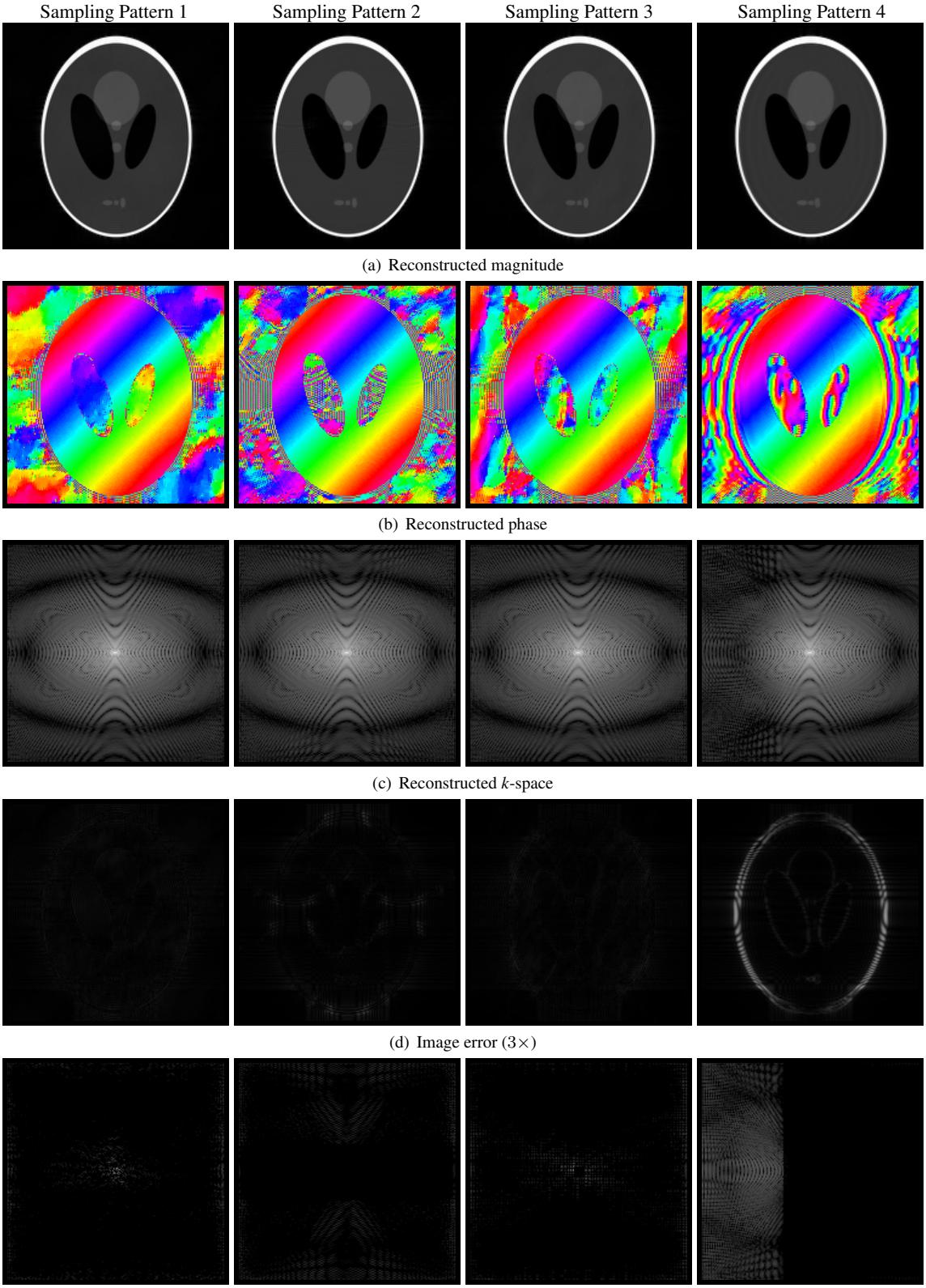
(e) k-space error

Figure 6: Reconstruction results for sampling pattern 4.

For the sake of comparison, Fig. 7 shows the results of reconstructing these same four datasets using total variation (TV) minimization, using the algorithm described in [35] (not included with the MATLAB demo). As these figures illustrate, with the exception of sampling pattern 4, TV-based reconstruction also permits highly-accurate reconstruction from this kind of data.[1] This is not surprising, given the highly-sparse nature of the Shepp-Logan phantom with respect to the kind of finite differencing operator used in TV.[2] Comparing with the LORAKS-based result, it is also notable that the error patterns for TV-based reconstruction are quite different from the error patterns observed for LORAKS-based reconstruction. Similar to our comments in [1], we believe that this is indicative of the potential complementarity of the different kinds of constraints.

[1] It is worth noting that the success of TV-based reconstruction does **NOT** depend solely on the incoherence of aliasing artifacts in a zero-filled reconstruction (a common misconception in the MR literature), since sampling patterns 2 and 3 both have highly structured sampling that yield highly-coherent aliasing. See [19, 35] for further discussion of incoherence in the context of MR imaging.

[2] Strictly speaking, the Shepp-Logan phantom used in this example is not exactly sparse with respect to TV, due to the presence of Gibbs ringing (resulting from the truncation of the analytic Fourier transform) and the presence of phase variation across the field-of-view. Despite this, this version of the Shepp-Logan phantom still has a relatively small number of non-negligible TV coefficients, and can still be considered as highly "compressible."



(a) Reconstructed magnitude

(b) Reconstructed phase

(c) Reconstructed k-space

(d) Image error ($3\times$)

(e) k-space error

Figure 7: TV-based reconstructions.

5 Multiband Signal Reconstruction Example

While LORAKS was designed with the application to MR image reconstruction in mind, it is notable that the approach can also be used in other application contexts where a signal is observed in one domain, while the corresponding Fourier transform (or inverse Fourier transform) of the signal is known to have limited support or smoothly-varying phase. One such non-MRI example that we illustrate in the `MB_demo.m` code is the spectrum-blind reconstruction of multiband signals (see [40] for context and references for this problem).

An example of a discrete-time multiband signal (the same signal used in the `MB_demo.m` code) is shown in Fig. 8. In this example, the signal has its energy concentrated on multiple frequency bands in the Fourier domain. In the spectrum-blind scenario, the locations of these bands are unknown *a priori*, and the reconstruction problem is to recover this time-domain signal based on sub-Nyquist samples.

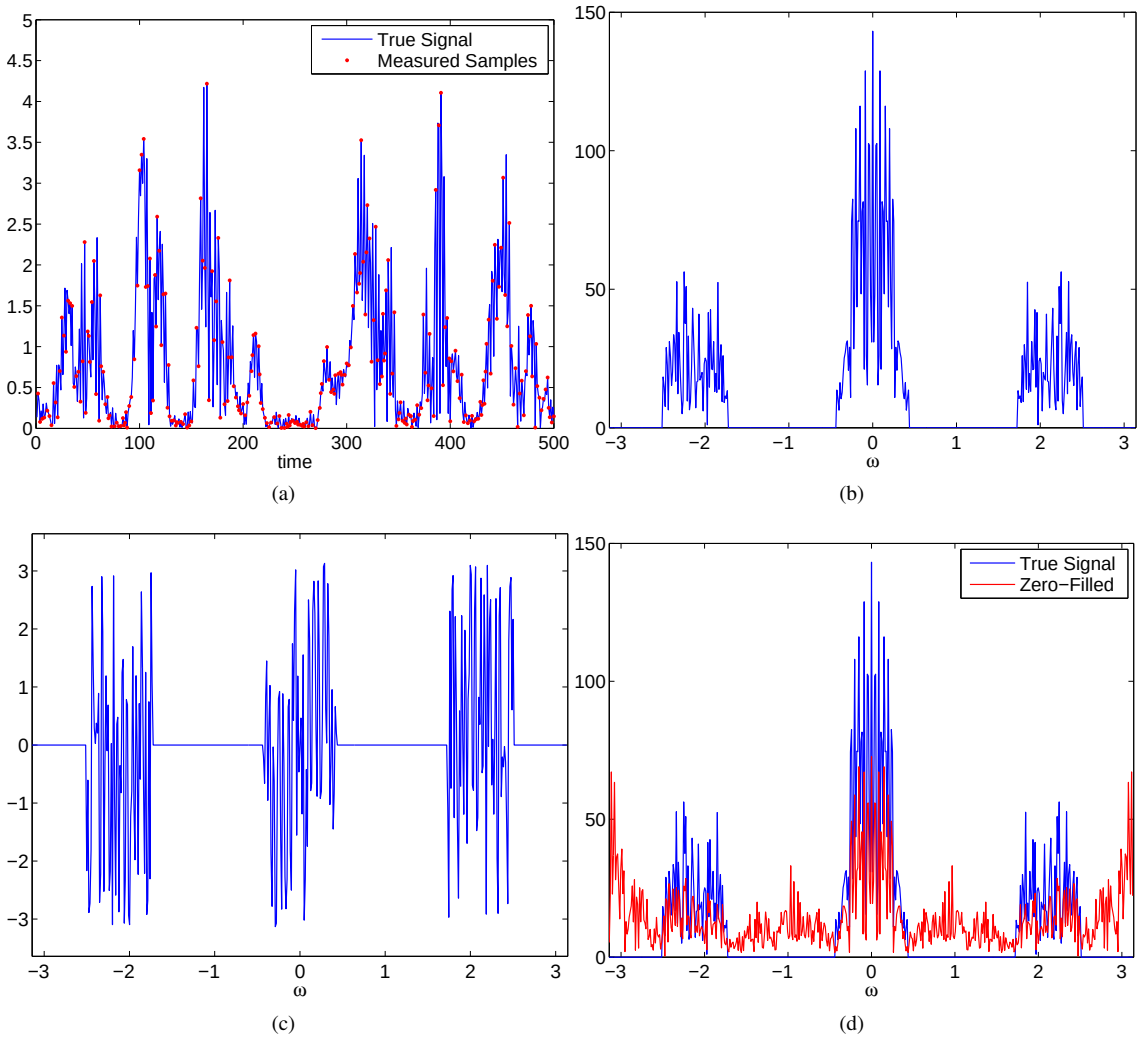


Figure 8: Illustration of the discrete-time multiband signal. (a) The magnitude of the multiband signal in the time domain. Also shown are the randomly-selected time samples that are used for reconstruction from subsampled data (55% of the samples from the original signal). (b) The magnitude of the DTFT of the signal from (a), illustrating its multiband structure. (c) The phase of the DTFT of the signal from (a). (d) Zero-padded reconstruction of the subsampled data from (a) demonstrates significant aliasing in the DTFT domain, as expected.

As can be seen from Fig. 8(b), the signal we consider can be viewed as sparse in the Fourier domain, and conventional sparsity-based compressed sensing approaches might be an obvious choice for this reconstruction problem. However, this particular example is constructed such that the sparsity level is not particularly low (i.e., 38.6% of the DFT coefficients are non-negligible), and as a result, standard ℓ_1-minimization approaches for signal recovery would generally require a large number of measurements for accurate reconstruction (e.g., following the rule of thumb to acquire a number of measurements equal to 3-4$\times$ the sparsity level [41] would result in full sampling of this signal). In addition, the multiband nature of the signal means that the Fourier-domain support of the signal is highly structured (i.e., the support is non-random, consisting of several fully-occupied frequency bands that are separated by empty regions of the spectrum) – this additional structure would not be imposed by ℓ_1-minimization.

In contrast to ℓ_1-minimization, $\mathbf{C}$-based LORAKS is very well-suited to this reconstruction problem, since the underlying assumption for $\mathbf{C}$-based LORAKS is not just that the signal has sparse support after Fourier transformation, but also that there are sizeable contiguous regions of the Fourier spectrum for which the signal possesses negligible energy. As `MB_demo.m` and the figures below illustrate, $\mathbf{C}$-based LORAKS enables accurate reconstruction of this signal from a randomly-selected set of 55% of the original time samples. In contrast, ℓ_1-minimization yields an inaccurate estimate of the signal support, as would be expected given that the number of measured time samples is not substantially bigger than the Fourier-domain support of the multiband signal.[3]

The `MB_demo.m` code reconstructs the multiband signal using $\mathbf{C}$-based LORAKS, $\mathbf{S}$-based LORAKS, and $\mathbf{G}$-based LORAKS. In each case, the neighborhood radius R and the matrix ranks $r_{\mathbf{C}}$, $r_{\mathbf{G}}$, and $r_{\mathbf{S}}$ have been adjusted for optimal reconstruction performance. The multiband signal of interest in this case is real-valued, but this fact has not been incorporated into the reconstruction procedure (i.e., LORAKS reconstruction treats the signal as complex-valued). Due to the relatively small size of this 1D signal, LORAKS reconstruction finished in less than 10 seconds on our workstation for all three cases.

Figures 9-11 show the results for the LORAKS reconstructions of this signal. Specifically, Fig. 9 shows the reconstructions in the discrete time domain, while Fig. 10 shows the reconstructions in the DTFT domain, and Fig. 11 shows the reconstruction error magnitudes in the DTFT domain. As can be seen, all three different LORAKS-based reconstruction results are quite accurate. Quantitatively, **$\mathbf{C}$-based LORAKS has 2.32% reconstruction error** (measured in the ℓ_2-norm), while **$\mathbf{G}$-based LORAKS has 2.44% reconstruction error**, and **$\mathbf{S}$-based LORAKS has 5.57% reconstruction error**.

For comparison, we also performed reconstruction of this data using Fourier-domain ℓ_1-minimization, while also imposing the additional information that the time-domain signal is real-valued. Specifically, we used SDPT3 [43] and YALMIP [44] to solve

$$\hat{\mathbf{k}} = \arg \min_{\mathbf{k} \in \mathbb{R}^S : \mathbf{F}\mathbf{k} = \mathbf{d}} \|\mathbf{D}\mathbf{k}\|_{\ell_1} , \tag{6}$$

where $\mathbf{k}$ is the vector of time samples to be reconstructed; $\mathbf{d}$ is the vector of observed time samples; $\mathbf{F}$ is the time-domain subsampling operator; and $\mathbf{D}$ is the DFT matrix. The reconstruction results for this case were also shown in Figs. 9-11, and demonstrate (as expected) that ℓ_1-minimization has very limited capabilities for this setting. Quantitatively, **ℓ_1-minimization has 32.28% reconstruction error**, which is substantially worse than the LORAKS-based reconstructions. This gap between ℓ_1-minimization and LORAKS-based reconstruction is consistent with some of the results shown in [1], and can be attributed in part to the fact that ℓ_1-minimization is blind to the fact that the support of this signal has a specific kind of structure.

In these examples, while $\mathbf{C}$-based LORAKS unsurprisingly has the best performance, it might be surprising that $\mathbf{S}$- and $\mathbf{G}$-based LORAKS still perform as well as they do. Specifically, $\mathbf{S}$- and $\mathbf{G}$-based LORAKS were derived in [1] based on smooth-phase assumptions, while Fig. 8(c) illustrates that the Fourier-domain phase of this signal varies quite rapidly. The good performance in these cases can be partially attributed to the fact that the $\mathbf{G}$ and $\mathbf{S}$ matrices share many characteristics with the $\mathbf{C}$ matrix, and can also have low-rank when the signal support is limited (see [1],

[3] It should be noted that a structured matrix-completion approach similar to $\mathbf{C}$-based LORAKS has recently been proposed for reconstruction of signals that are sparse in the Fourier domain, and that theoretical incoherence conditions were also derived under which structured matrix completion is guaranteed to be successful [42]. It is notable that these theoretical results do not seem to emphasize the importance of the geometry of the signal support on the performance of structured matrix completion approaches. However, as can be inferred from the derivation and discussion of $\mathbf{C}$-based LORAKS [1], the support geometry has a substantial influence on the low-rank characteristics of the $\mathbf{C}$-matrix, and should hence have a substantial influence on the ultimate success of the reconstruction. We conjecture that the performance guarantees from [42] are overly pessimistic in cases where the signal support is not extremely sparse but has additional useful structure.

and note that the $h(x, y)$ function used to derive the low-rank characteristics of $\mathbf{G}$ and $\mathbf{S}$ can take on arbitrary values outside the signal support).

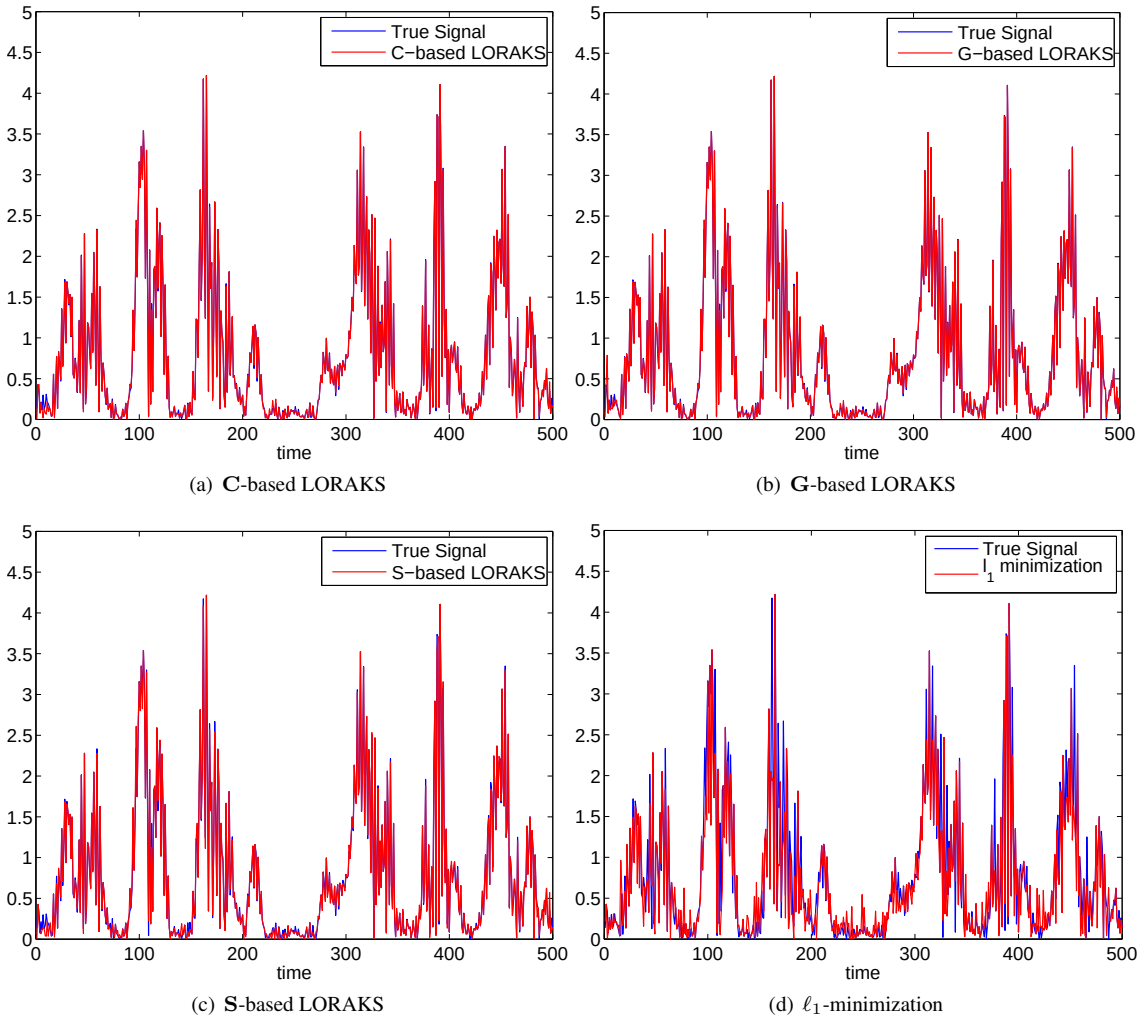
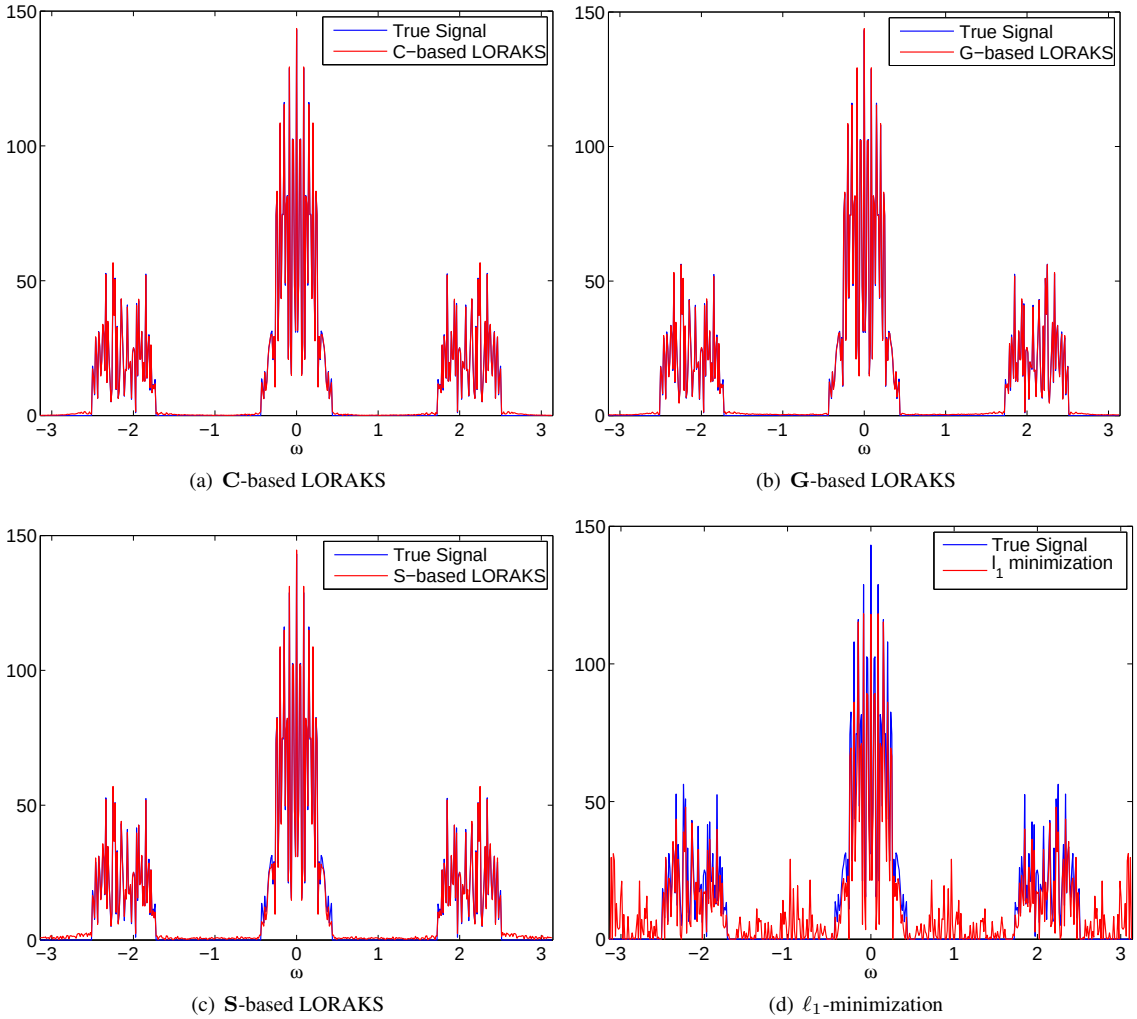


Figure 9: Reconstructions of the multiband signal from 55% of its samples using (a-c) LORAKS and (d) ℓ_1-minimization. Plots show the magnitude of the reconstructed time-domain signal in red, superimposed on the true original signal shown in blue.



(a) **C**-based LORAKS

(b) **G**-based LORAKS

(c) **S**-based LORAKS

(d) ℓ_1-minimization

Figure 10: Reconstructions of the multiband signal from 55% of its samples using (a-c) LORAKS and (d) ℓ_1-minimization. Plots show the magnitude of the reconstructed signal in the DTFT domain in red, superimposed on the true original signal shown in blue.

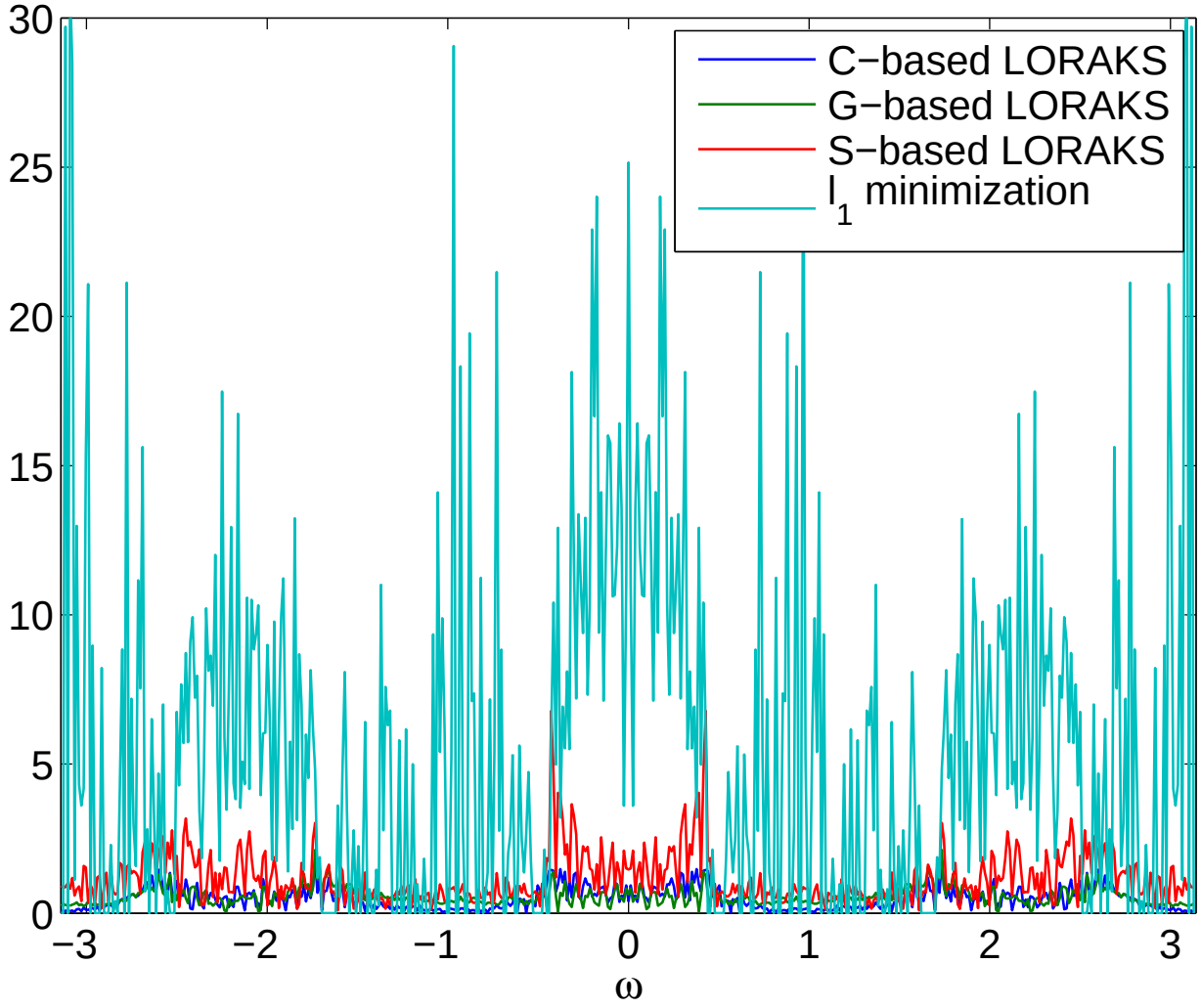


Figure 11: Reconstruction error magnitudes for LORAKS and ℓ_1-minimization plotted in the DTFT domain. Note that, relative to Fig. 10, the vertical scale of the plot has been adjusted for better visualization of small errors.

References

[1] J. P. Haldar, "Low-rank modeling of local k-space neighborhoods (LORAKS) for constrained MRI," *IEEE Trans. Med. Imag.*, vol. 33, pp. 668–681, 2014.

[2] J. P. Haldar, "Calibrationless partial Fourier reconstruction of MR images with slowly-varying phase: A rank-deficient matrix recovery approach," in *Proc. ISMRM Workshop on Data Sampling & Image Reconstruction*, 2013.

[3] J. P. Haldar, "LORAKS: Low-rank modeling of local k-space neighborhoods," in *Proc. Int. Soc. Magn. Reson. Med.*, 2014, Accepted.

[4] J. Zhuo and J. P. Haldar, "P-LORAKS: Low-rank modeling of local k-space neighborhoods with parallel imaging data," in *Proc. Int. Soc. Magn. Reson. Med.*, 2014, Accepted.

[5] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, pp. 471–501, 2010.

[6] P. Margosian, F. Schmitt, and D. Purdy, "Faster MR imaging: imaging with half the data," *Health Care Instrum.*, vol. 1, pp. 195–197, 1986.

[7] D. C. Noll, D. G. Nishimura, and A. Macovski, "Homodyne detection in magnetic resonance imaging," *IEEE Trans. Med. Imag.*, vol. 10, pp. 154–163, 1991.

[8] F. Huang, W. Lin, and Y. Li, "Partial Fourier reconstruction through data fitting and convolution in k-space," *Magn. Reson. Med.*, vol. 62, pp. 1261–1269, 2009.

[9] K. F. Cheung and R. J. Marks II, "Imaging sampling below the Nyquist density without aliasing," *J. Opt. Soc. Am. A*, vol. 7, pp. 92–105, 1990.

[10] S. K. Plevritis and A. Macovski, "Spectral extrapolation of spatially bounded images," *IEEE Trans. Med. Imag.*, vol. 14, pp. 487–497, 1995.

[11] B. Madore, G. H. Glover, and N. J. Pelc, "Unaliasing by Fourier-encoding the overlaps using the temporal dimension (UNFOLD), applied to cardiac imaging and fMRI," *Magn. Reson. Med.*, vol. 42, pp. 813–828, 1999.

[12] J. Tsao, P. Boesiger, and K. P. Pruessmann, "*k-t* BLAST and *k-t* SENSE: Dynamic MRI with high frame rate exploiting spatiotemporal correlations," *Magn. Reson. Med.*, vol. 50, pp. 1031–1042, 2003.

[13] B. Sharif, J. A. Derbyshire, A. Z. Faranesh, and Y. Bresler, "Patient-adaptive reconstruction and acquisition in dynamic imaging with sensitivity encoding (PARADISE)," *Magn. Reson. Med.*, vol. 64, pp. 501–513, 2010.

[14] J. P. Haldar and Z.-P. Liang, "Spatiotemporal imaging with partially separable functions: A matrix recovery approach," in *Proc. IEEE Int. Symp. Biomed. Imag.*, 2010, pp. 716–719.

[15] B. Zhao, J. P. Haldar, C. Brinegar, and Z.-P. Liang, "Low rank matrix recovery for real-time cardiac MRI," in *Proc. IEEE Int. Symp. Biomed. Imag.*, 2010, pp. 996–999.

[16] S. G. Lingala, Y. Hu, E. DiBella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low-rank structure: k-t SLR," *IEEE Trans. Med. Imag.*, vol. 30, pp. 1042–1054, 2011.

[17] J. Trzasko and A. Manduca, "Local versus global low-rank promotion in dynamic MRI series reconstruction," in *Proc. Int. Soc. Magn. Reson. Med.*, 2011, p. 4371.

[18] J. P. Haldar and Z.-P. Liang, "Low-rank approximations for dynamic imaging," in *Proc. IEEE Int. Symp. Biomed. Imag.*, 2011, pp. 1052–1055.

[19] J. P. Haldar, "Constrained imaging: Denoising and sparse sampling," Ph.D. dissertation, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 2011. [Online]. Available: http://hdl.handle.net/2142/24286

[20] H. Gao, S. Rapacchi, D. Wang, J. Moriarty, C. Meehan, J. Sayre, G. Laub, P. Finn, and P. Hu, "Compressed sensing using prior rank, intensity and sparsity model (PRISM): Applications in cardiac cine MRI," in *Proc. Int. Soc. Magn. Reson. Med.*, 2012, p. 2242.

[21] R. Otazo, E. Candès, and D. K. Sodickson, "Low-rank and sparse matrix decomposition for accelerated DCE-MRI with background and contrast separation," in *Proc. ISMRM Workshop on Data Sampling & Image Reconstruction*, 2013.

[22] A. Majumdar, "Improved dynamic MRI reconstruction by exploiting sparsity and rank-deficiency," *Magn. Reson. Imag.*, vol. 31, pp. 789–795, 2013.

[23] T. Zhang, M. Alley, M. Lustig, J. Pauly, and S. Vasanawala, "Fast 3D DCE-MRI with sparsity and low-rank enhanced SPIRiT (SLR-SPIRiT)," in *Proc. ISMRM Workshop on Data Sampling & Image Reconstruction*, 2013.

[24] M. Lustig, M. Elad, and J. M. Pauly, "Calibrationless parallel imaging reconstruction by structured low-rank matrix completion," in *Proc. Int. Soc. Magn. Reson. Med.*, 2010, p. 2870.

[25] M. Lustig, "Post-cartesian calibrationless parallel imaging reconstruction by structured low-rank matrix completion," in *Proc. Int. Soc. Magn. Reson. Med.*, 2011, p. 483.

[26] J. D. Trzasko and A. Manduca, "CLEAR: Calibration-free parallel imaging using locally low-rank encouraging reconstruction," in *Proc. Int. Soc. Magn. Reson. Med.*, 2012, p. 517.

[27] H. M. Nguyen, X. Peng, M. N. Do, and Z.-P. Liang, "Denoising MR spectroscopic imaging data with low-rank approximations," *IEEE Trans. Biomed. Eng.*, vol. 60, pp. 78–89, 2013.

[28] J. D. Trzasko, S. D. Sharma, and A. Manduca, "Calibrationless chemical shift encoded imaging," in *Proc. ISMRM Workshop on Data Sampling & Image Reconstruction*, 2013.

[29] F. Lam, S. D. Babacan, J. P. Haldar, M. W. Weiner, N. Schuff, and Z.-P. Liang, "Denoising diffusion-weighted magnitude MR images using rank and edge constraints." *Magn. Reson. Med.*, 2013, In Press. DOI: 10.1002/mrm.24728A.

[30] G. Adluru, L. Chen, D. Feinberg, J. Anderson, and E. V. R. DiBella, "Improving rank constrained reconstructions using prior information with reordering," in *Proc. Int. Soc. Magn. Reson. Med.*, 2012, p. 2248.

[31] H. Gao, L. Li, and X. P. Hu, "Compressive diffusion MRI – part 1: Why low rank?" in *Proc. Int. Soc. Magn. Reson. Med.*, 2013, p. 610.

[32] F. Lam, B. Zhao, Y. Liu, and Z.-P. Liang, "Accelerated fMRI using low-rank model and sparsity constraints," in *Proc. Int. Soc. Magn. Reson. Med.*, 2013, p. 2620.

[33] M. J. Muckley, S. J. Peltier, D. C. Noll, and J. A. Fessler, "Model-based reconstruction for physiological noise correction in functional MRI," in *Proc. Int. Soc. Magn. Reson. Med.*, 2013, p. 2623.

[34] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, pp. 1182–1195, 2007.

[35] J. P. Haldar, D. Hernando, and Z.-P. Liang, "Compressed-sensing MRI with random encoding," *IEEE Trans. Med. Imag.*, vol. 30, pp. 893–903, 2011.

[36] D. L. Donoho, "An invitation to reproducible computational research," *Biostat.*, vol. 11, pp. 385–388, 2010.

[37] R. M. Larsen, "Lanczos bidiagonalization with partial reorthogonalization," Department of Computer Science, Aarhus University, Tech. Rep. DAIMI PB-357, 1998. [Online]. Available: http://soi.stanford.edu/~rmunk/PROPACK/

[38] S. X. Pan and A. C. Kak, "A computational study of reconstruction algorithms for diffraction tomography: Interpolation versus filtered backpropagation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, pp. 1262–1275, 1983.

[39] A. K. Jain and S. Ranganath, "Extrapolation algorithms for discrete signals with application in spectral estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, pp. 830–845, 1981.

[40] Y. Bresler, "Spectrum-blind sampling and compressive sensing for continuous-index signals," in *Information Theory and Applications Workshop*, 2008, pp. 547–554.

[41] Y. Tsaig and D. L. Donoho, "Extensions of compressed sensing," *Signal Process.*, vol. 86, pp. 549–571, 2006.

[42] Y. Chen and Y. Chi, "Spectral compressed sensing via structured matrix completion," in *JMLR Workshop and Conference Proceedings Volume 28: Proceedings of The 30th International Conference on Machine Learning*, 2013, pp. 414–422.

[43] K. C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3 - a MATLAB software package for semidefinite-quadratic-linear programming." [Online]. Available: http://www.math.nus.edu.sg/~mattohkc/sdpt3.html

[44] J. Lofberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Symp. Comp. Aid. Control Sys. Design*, 2004, pp. 284–289.