Fast Algorithm for Multi-dimensional Low Rank Hankel Matrix Recovery

Greg Ongie*, *Student Member, IEEE*, Yasir Moshin, *Student Member, IEEE*, Mathews Jacob, *Senior Member, IEEE*

***Abstract*—We propose a new fast algorithm for the recovery of large scale low rank matricies having a multi-dimensional convolutional structure. This includes matricies having a Hankel or Toeplitz structure, and their higher dimensional analogues. We pose recovery of the matrix as the minimization of the Shatten p-norm as a surrogate for rank, which generalizes the commonly used nuclear norm penalty. We consider both convex and non-convex versions of this algorithm. Using a majorization-minimization (MM) scheme, we show we may exploit the convolutional structure to yield an SVD-free algorithm, which only requires few FFTs per iteration. The algorithm can be interpreted as alternating between the estimation of an annihilating filter and the estimation of the unknown matrix entries by linear prediction. We demonstrate the algorithm on the problem of multi-dimensional harmonic retrieval and multi-dimensional finite-rate-of-innovation MR image recovery, which is shown to be significantly faster than the current state-of-the-art algorithms based on singular value thresholding.**

I. Introduction

A diverse set of problems in signal processing can be posed as the recovery of a structured low-rank matrix, including harmonic retrial/direction-of-arrival, phase retrival (?), etc. We are motivated in particular by structured low-rank matrix formulations now being posed for compressed sensing magnetic resonance image recovery, which began with an application self-calibrating parallel MRI, and has continued to include novel single coil recovery algorithms, and simuletanous multi-slice. This mode of image recovery has been shown to be very powerful, improving significantly over the previous state-of-the-art, such as total variation minimization (TV). However, these algorithms are computationally much more expensive than a simple TV-minimization, requiring several SVD's of a very large matrix, or several large dense matrix inversions.

Early approaches to nuclear norm minimization were to solve an equivalent SDP, but this approach is only feasible for very small problem sizes. More modern approaches is to perform some form of singular value thresholding (SVT).

Various heuristics have been proposed to speed up these computations, including the well-known matrix factorization trick, which minimizes over all possible matrix factorizations subject to a Frobenius norm constraint.

However, none of the above algorithms exploit the inherent low-dimensionality of the problem. Rather, typically the structure of the matrix is ignored, and solving the problem takes on the complexity of the higher-dimensional embedding.

We show that for a large class of low-rank matricies which are convolutional in structural, i.e. matricies that represent a partial n-dimensional convolution or correlation.

In the case p=1 (nuclear norm), our algorithm can be viewed an application of the iterative reweighted least squares algorithm proposed in []. However, our algorithm is more than a straightforward application of []; our algorithm involves many refinements.

Our algorithm is closely related to the spectral MUSIC algorithm used in harmonic retrieval and direction-of-arrival (DOA) estimation.

In particular, we show that our algorithm can be thought alternating between (1) an estimation of the nullspace of the structured matrix, equivalently, a filter-bank of annihilating filters, and (2) a least squares linear prediction step, which fills in the missing entries of the structured matrix by annihilation by the filter estimated in the first step.

Things to cite: [1]–[6].

II. Problem Formulation

We are concerned with the class of problems that may be posed as

$$\min_{\mathbf{x}} \ \mathrm{rank}[\mathcal{T}(\mathbf{x})] \quad s.t. \quad \mathbf{A}\mathbf{x} = \mathbf{b}$$

where $\mathbf{x} \in \mathbb{C}^n$ is the data vector to be recovered, $\mathcal{T}(\mathbf{x})$ is a structured lifting of $\mathbf{x}$ to a matrix in $\mathbb{C}^{N\times M}$ of *convolutional type* (defined below), $\mathbf{A}$ is the measurement matrix, $\mathbf{b}$ are the ideal measurements.

III. Matricies of Convolutional Type

The algorithm we propose is applicable to a broad class of matricies which have a particular type of convolutional structure. Specifically, we call a matrix $\mathbf{X} \in \mathbb{C}^{N\times M}$ *convolutional type* if there exists a vector $\mathbf{x} \in \mathbb{C}^n$, we call the *data vector*, such that $\mathbf{X}$ can be expressed as a composition of operators

$$\mathbf{X} = \mathcal{P}\mathcal{C}(\mathbf{M}\mathbf{x}), \tag{1}$$

where $\mathbf{M} \in \mathbb{C}^{m\times n}$, $m \geq n$, is any full rank matrix, $\mathcal{C}[\mathbf{k}]$ is the linear operator representing discrete multidimensional convolution with the kernel $\mathbf{k} \in \mathbb{C}_0(\mathbb{Z}^d)$ in all of $\mathbb{C}(\mathbb{Z}^d)$ i.e. $\mathcal{C}[\mathbf{k}]\mathbf{y} = \mathbf{k} * \mathbf{y}$. Also the operator $\mathcal{P} : \mathbb{C}(\mathbb{Z}^d) \to \mathbb{C}^N$ is a projection onto a finite set of valid indices $\Delta \subseteq \mathbb{Z}^d$, $|\Delta| = N$.

G. Ongie is with the Department of Mathematics, University of Iowa, Y. Moshin and M. Jacob are with the Department of Electrical and Computer Engineering, University of Iowa, Iowa City, IA, 52245 USA (e-mail: gregory-ongie@uiowa.edu; yasir-moshin@uiowa.edu; mjacob@uiowa.edu)

This work is supported by grants NSF CCF-0844812, NSF CCF-1116067, NIH 1R21HL109710-01A1, ACS RSG-11-267-01-CCE, and ONR-N000141310202.

More generally, we will consider matrices are a vertical concatenation of matricies in the form (1), $\mathbf{X} = [\mathbf{X}_1^H, ..., \mathbf{X}_K^H]^H$, where each $\mathbf{X}_i = \mathcal{P}_i\mathcal{C}[\mathbf{M}_i\mathbf{x}]$. In this case we require the matrix $\mathbf{M} = [\mathbf{M}_1, ..., \mathbf{M}_K]$ to be full rank.

This definition generalizes Toeplitz and Hankel matrices to higher dimensions.

For every decomposition in the form (1) we also define a corresponding *lifting operator* $\mathcal{T}(\mathbf{x}) = \mathcal{PC}(\mathbf{Mx})$.

The assumption that $\mathcal{T}(\mathbf{x})$ is low-rank is equivalent the existence of a collection of large collection of linearly independent *annihilating filters* for the transformed data vector $\mathbf{y} = \mathbf{Mx}$, i.e. each vector $\mathbf{d}$ in the nullspace of $\mathcal{T}(\mathbf{x})$ satisfies

$$\mathcal{T}(\mathbf{x})\mathbf{d} = \mathcal{P}(\mathbf{y} * \mathbf{d}) = 0$$

that is, convolution by $\mathbf{d}$ annihilates $\mathbf{y}$ (when restricted to the index set Δ).

IV. Problem Formulation

Our goal is to recover the data vector $\mathbf{x}_0$ from its noisy linear measurements $\mathbf{b} = \mathbf{A}\mathbf{x}_0 + \mathbf{n}$, where $\mathbf{A}$ is the measurement matrix, and $\mathbf{n}$ is a noise vector, which for simplicity we model as additive white Gaussian noise. Under the assumption that $\mathcal{T}(\mathbf{x}_0)$ is the unique rank minimizer subject to the data constraints, we pose recovery as

$$\min_{\mathbf{x}} \|\mathcal{T}(\mathbf{x})\|_p^p + \frac{\lambda}{2}\|\mathbf{Ax} - \mathbf{b}\|_2^2, \tag{2}$$

where $\|\cdot\|_p$ is the Shatten p-norm, $0 \leq p \leq 1$.

V. Majorization-Minimization Scheme

Recall that a majorization-minimization (MM) algorithm is a iterative method for minimizing a cost function, where at each iteration one minimizes a surrogate cost function that majorizes the original. More precisely, if $f(\mathbf{x})$ is any real-valued function on a domain Ω, a function $g(\mathbf{x}; \mathbf{x}^*)$ is said to majorize $f(\mathbf{x})$ at the fixed value $\mathbf{x}_0$ if

$$g(\mathbf{x}; \mathbf{x}_0) \geq f(\mathbf{x}), \quad \text{for all} \quad \mathbf{x} \in \Omega \tag{3}$$

$$g(\mathbf{x}_0; \mathbf{x}_0) = f(\mathbf{x}_0), \quad \text{for all} \quad \mathbf{x} \in \Omega \tag{4}$$

and if this holds for all $\mathbf{x}_0 \in \Omega$ we call g a *majorizer* for f. We then solve

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x})$$

by instead solving the iterates

$$\mathbf{x}^{(n+1)} = \arg\min_{\mathbf{x}\in\Omega} g(\mathbf{x}; \mathbf{x}^{(n)}).$$

Using properties (3) and (4) it is easy to show the cost function f must monotonically decrease at each iteration. Furthermore, if f is convex, one can prove convergence of the iterates $\mathbf{x}^{(n)}$ to the unique minimizer of f.

We now construct a majorizer for the Shatten p-norm $\|\cdot\|_p^p$, including values $p \in [0, 1)$ where the penalty is non-convex. First observe that we may write

$$\|\mathbf{X}\|_p^p = Tr[(\mathbf{X}^*\mathbf{X})^{\frac{p}{2}}]$$

Here the qth power $\mathbf{Y}^q$ of any Hermitian positive semidefinite matrix Y is defined as

$$\mathbf{Y}^q = \sum_i \lambda_i^q \mathbf{P}_i$$

where $\mathbf{Y} = \sum_i \lambda_i \mathbf{P}_i$ is the spectral decomposition of $\mathbf{X}$, i.e. $\lambda_i \geq 0$ are the eigenvalues of $\mathbf{X}$ and $\mathbf{P}_i$ are the orthogonal projectors onto the corresponding eigenspaces.

Let H_+^n denote the space of Hermitian positive semidefinite $n \times n$ matrices, H_{++}^n the space of hermitian positive definite $n \times n$ matrices.

Proposition 1. *If $\mathbf{Y}_0 \in H_{++}^n$ then for all $\mathbf{Y} \in H_+^n$ we have*

$$Tr[\mathbf{Y}^q] \leq Tr[\mathbf{Y}_0^q + q\mathbf{Y}_0^{q-1}(\mathbf{Y} - \mathbf{Y}_0)]$$

for all $q \in (0, 1]$.

Proof. This is a special case of Klein's inequality [1], which states that for any differentiable concave function $f : (0, \infty) \to \mathbb{R}$ and any $\mathbf{Y} \in H_+^n$, $\mathbf{Y}_0 \in H_{++}^n$ we have

$$Tr[f(\mathbf{Y}) - f(\mathbf{Y}_0) - f'(\mathbf{Y}_0)(\mathbf{Y} - \mathbf{Y}_0)] \leq 0 \tag{5}$$

where $f(\mathbf{X}) := \sum_i f(\lambda_i)\mathbf{P}_i$ when $\mathbf{X}$ has the spectral decomposition $\mathbf{X} = \sum_i \lambda_i \mathbf{P}_i$. Choosing $f(t) = t^q$ gives the desired result. □

Provided $\mathbf{X}_0 \in \mathbb{C}^{n\times m}$ has no zero singular values, substituting $\mathbf{Y} = \mathbf{X}^H\mathbf{X}$, $\mathbf{Y}_0 = \mathbf{X}_0^H\mathbf{X}_0$, and $q = p/2$ into (5), we have that the function

$$\begin{aligned} g(\mathbf{X}; \mathbf{X}_0) &= Tr[(\mathbf{X}_0^H\mathbf{X}_0)^{\frac{p}{2}} + \frac{p}{2}(\mathbf{X}_0^H\mathbf{X}_0)^{-\frac{(2-p)}{2}}((\mathbf{X}^H\mathbf{X}) - (\mathbf{X}_0^H\mathbf{X}_0))] \\ &= \frac{p}{2} Tr[(\mathbf{X}_0^H\mathbf{X}_0)^{-\frac{(2-p)}{2}}\mathbf{X}^H\mathbf{X}] + C(\mathbf{X}_0) \\ &= \frac{p}{2}\|\mathbf{X}(\mathbf{X}_0^H\mathbf{X}_0)^{-\frac{(2-p)}{4}}\|_F^2 + C(\mathbf{X}_0) \end{aligned}$$

satisfies the majorization relations

$$g(\mathbf{X}; \mathbf{X}_0) \geq \|\mathbf{X}\|_p^p, \quad \text{for all} \quad \mathbf{X}, \mathbf{X}_0 \tag{6}$$

$$g(\mathbf{X}_0; \mathbf{X}_0) = \|\mathbf{X}_0\|_p^p, \quad \text{for all} \quad \mathbf{X}, \mathbf{X}_0 \tag{7}$$

for all $p \in (0, 1]$. Finally, setting $\mathbf{X} = \mathcal{T}(\mathbf{x})$, gives the following MM scheme for (2):

$$\mathbf{x}^{(n+1)} = \arg\min_{\mathbf{x}} \frac{p}{2}\|\mathcal{T}(\mathbf{x})\mathbf{W}\|_F^2 + \frac{\lambda}{2}\|\mathbf{Ax} - \mathbf{b}\|_2^2 \tag{8}$$

where $\mathbf{W}$ is given by

$$\mathbf{W}^{(n)} = [\mathcal{T}(\mathbf{x}^{(n)})^H\mathcal{T}(\mathbf{x}^{(n)})]^{-\frac{(2-p)}{4}}. \tag{9}$$

VI. Simplications

We now show how the iterates (8) and (9) simplify in our setting, and how the scheme may be interpreted as alternating between enforcing an annihilation constraint and estimation of an annihilating filter.

[1] See https://en.wikipedia.org/wiki/Trace_inequalities#Klein.27s_inequality

A. Step one: Least-squares annihilation

First, we focus on solving (8). Consider the more general case where in place of $\mathbf{W}^{(n)}$ we have an arbitrary filterbank $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_N]$, i.e. each one of the columns $\mathbf{d}_i$ represents a filter, and we wish to solve for the data vector $\mathbf{x}$ which is best annihilated by each filter $\mathbf{d}_i$ in a least squares sense, subject to data constraints. Then we may solve

$$\min_{\mathbf{x}} \|\mathcal{T}(\mathbf{x})\mathbf{D}\|_F^2 + \lambda\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$$

or, equivalently,

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \|\mathcal{T}(\mathbf{x})\mathbf{d}_i\|_2^2 + \lambda\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2.$$

Define the linear operator $\mathcal{Q}_i$ by

$$\mathcal{Q}_i\mathbf{x} = \mathcal{T}(\mathbf{x})\mathbf{d}_i = \mathcal{P}(\mathbf{M}\mathbf{x} * \mathbf{d}_i) = \mathcal{P}(\mathbf{d}_i * \mathbf{M}\mathbf{x})$$

for all $i = 1, \dots, N$. Therefore, we may expand each $\mathcal{Q}_i$ as

$$\mathcal{Q}_i = \mathcal{P}\mathbf{C}_i\mathbf{M}$$

where $\mathbf{C}_i$ is convolution by the filter $\mathbf{d}_i$, and $\mathbf{P}$ is projection onto the set of valid convolution indices. Because of the projection, we may assume $\mathbf{C}_i$ is a *circular* convolution, provided we zero-pad the vector $\mathbf{M}\mathbf{x}$ sufficiently, and so we may write $\mathbf{C}_i = \mathbf{F}\mathbf{S}_i\mathbf{F}^*$ where $\mathbf{F}$ is an d-dimensional DFT and $\mathbf{S}_i$ is a diagonal matrix representing pointwise multiplication by the inverse DFT of the zero-padded filter $\mathbf{d}_i$.

In this notation, the problem now is to solve

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \|\mathcal{Q}_i\mathbf{x}\|_2^2 + \frac{\lambda}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2,$$

Computing the gradient of the objective and setting it equal to zero yields the linear system:

$$\underbrace{\left(\sum_{i=1}^{N}(\mathcal{Q}_i^*\mathcal{Q}_i) + \lambda\mathbf{A}^H\mathbf{A}\right)}_{\mathbf{R}}\mathbf{x} = \lambda\mathbf{A}^H b$$

and we have

$$\mathcal{Q}_i^*\mathcal{Q}_i = \mathbf{M}^*\mathbf{C}_i^*\mathcal{P}^*\mathcal{P}\mathbf{C}_i\mathbf{M}.$$

If the projection set $\Delta \subseteq \mathbb{Z}^d$ is large relative to the filter size, the projection operator $\mathcal{P}^*\mathcal{P}$ will be close to identity. Hence, we propose making the approximation $\mathcal{P}^*\mathcal{P} \approx \mathbf{I}$, so that the above becomes

$$\mathcal{Q}_i^*\mathcal{Q}_i \approx \mathbf{M}^*\mathbf{C}_i^*\mathbf{C}_i\mathbf{M} = \mathbf{M}^*\mathbf{F}\mathbf{S}_i^*\mathbf{S}_i\mathbf{F}^*\mathbf{M}.$$

We study this approximation in detail later. And so

$$\mathbf{R} = \mathbf{M}^*\mathbf{F}\left(\sum_{i=1}^{N}\mathbf{S}_i^*\mathbf{S}_i\right)\mathbf{F}^*\mathbf{M} + \lambda\mathbf{A}^*\mathbf{A}$$

Therefore to apply $\mathbf{R}$ we only need to compute $\sum_i \mathbf{S}_i^*\mathbf{S}_i$, which can be thought of as pointwise multiplication with uniform samples of the sum-of-squares polynomial

$$\overline{\mu}(\mathbf{r}) = \sum_{i=1}^{N} |\mu_i(\mathbf{r})|^2. \tag{10}$$

where $\mu_i(\mathbf{r})$ is the inverse Fourier transform of $\mathbf{d}_i$, i.e. the trigonometric polynomial having coefficients $\mathbf{d}_i[\mathbf{k}], \mathbf{k} \in \Lambda$.

B. Step Two: Annihilating Filter Update

Now we show how to efficiently compute the weight matrix update (9). Recall that $\mathbf{W} = [\mathcal{T}(\mathbf{x})^*\mathcal{T}(\mathbf{x})]^{-r}$ for some $r > 0$ and some fixed data vector $\mathbf{x}$. Therefore, according to the previous section, the first step of the algorithm (8) becomes a least squares annihilation where the filter bank is specified by $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_N]$.

Note that the Gram matrix $\mathcal{T}(\mathbf{x})^*\mathcal{T}(\mathbf{x})$ may be computed efficiently as

$$\begin{aligned} \mathcal{T}(\mathbf{x})^H\mathcal{T}(\mathbf{x}) &= \mathcal{C}(\mathbf{M}\mathbf{x})^*\mathcal{P}^*\mathcal{P}\mathcal{C}(\mathbf{M}\mathbf{x}) && (11)\\ &= \mathcal{F}^*\mathcal{D}^*\mathcal{F}\mathcal{F}^*\mathcal{C}_{D_\Delta}\mathcal{F}\mathcal{F}^*\mathcal{D}\mathcal{F} && (12)\\ &= \mathcal{F}^*\mathcal{D}^*\mathcal{C}_{D_\Delta}\mathcal{D}\mathcal{F}(???) && (13) \end{aligned}$$

The entries of this matrix are given by $\langle\mathbf{y}[\mathbf{k}], \mathbf{y}[\boldsymbol{\ell}]\rangle$, where $\mathbf{y}[\mathbf{k}]$ is any column of the matrix $\mathcal{P}\mathcal{C}(\mathbf{M}\mathbf{x})$, indexed by $\mathbf{k} \in \Lambda$, where Λ is the filter support. But then $\mathbf{y}[\mathbf{k}]$ is any patch of $\mathbf{M}\mathbf{x}$ of size Λ supported within Δ taken from $\mathbf{M}\mathbf{x}$, so $\langle\mathbf{y}[\mathbf{k}], \mathbf{y}[\boldsymbol{\ell}]\rangle$ is the inner product of any two such patches of $\mathbf{M}\mathbf{x}$, and may be computed efficiently through an autocorrelation of $\mathbf{M}\mathbf{x}$ with itself.

Let $(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V})$ be a singular value decomposition of $\mathcal{T}(\mathbf{x})$, i.e. $\mathcal{T}(\mathbf{x}) = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$. Then $\mathbf{W} = \mathbf{V}\mathbf{\Sigma}^{-r}\mathbf{V}^*$, which has the matrix square root $\mathbf{W}^{1/2} = \mathbf{V}\mathbf{\Sigma}^{-r/2} = [\sigma_1^{-r/2}\mathbf{v}_1, \dots, \sigma_n^{-r/2}\mathbf{v}_N]$. But step one of the algorithm only needs the sum-of-squares polynomial (10) defined by the filterbank, which in this case can be expressed as

$$\overline{\mu}(\mathbf{r}) = \sum_{i=1}^{N} \sigma_i^{-r}|\gamma_i(\mathbf{r})|^2 \tag{14}$$

where γ_i is the inverse Fourier transform of the filter $\mathbf{v}_i[\mathbf{k}]$, $\mathbf{k} \in \Lambda$. Note σ_i^{-r} is small when corresponding filter is in the row space of $\mathcal{T}(\mathbf{x})$ and will be large when it is in the null space of $\mathcal{T}(\mathbf{x})$ (i.e. when it is an annihilating filter). Therefore, we may interpret $\overline{\mu}$ as a regularized annihilating filter.

Finally, note that in practice we actually use the ϵ-regularized inverse on the singular values in (14), i.e. instead of using σ^{-1}, which is undefined for $\sigma = 0$, we use

$$\sigma_\epsilon^{-1} := \frac{1}{\max(\sigma, \epsilon)}$$

or something similar. Typically, there is also a rule for shrinking $\epsilon \to 0$ as the iterates proceed, which speeds up the convergence.

C. Alternative mask updates

Rather than forming the mask according to (14), which weights the singular vectors by σ^{-1} (or σ_ϵ^{-1}), we can consider a general update of the form

$$\overline{\mu}(\mathbf{r}) = \sum_{i=1}^{N} \varphi(\sigma_i)|\gamma_i(\mathbf{r})|^2 \tag{15}$$

Where $\varphi : \mathbb{R}^+ \to \mathbb{R}^+$ is any weighting function which is large for small inputs. For example, for any $q > 0$ we could choose

$$\varphi(t) = t^{-q}$$

which generalizes the update (14). Choosing $q > 1$ could potentially work better, since this would further shrink filters corresponding to high singular values, and amplify filters in the null space, as desired. I did some experiments with $q = 2$, and it shows superior performance to $q = 1$. I believe this weighting corresponds replacing the nuclear norm with a Shatten p-norm for $p = 1/q < 1$, but I need to check the details.

Another choice would be the hard threshold

$$\varphi(t) = \begin{cases} 1 & \text{if } t < T \\ 0 & \text{else,} \end{cases}$$

for some $T > 0$, but this is not likely to be stable.

Additionally, we could consider truncate the sum in (15), as

$$\overline{\mu}(\mathbf{r}) = \sum_{i=R}^{N} \varphi(\sigma_i)|\gamma_i(\mathbf{r})|^2 \tag{16}$$

where R is some estimate on a lower-bound of the true rank. This way, we give zero weight to the first R filters corresponding to the highest singular values, which are not annihilating filters, and should not be incorporated into the mask. This showed good performance in experiments, too. I believe this corresponds to a weighting that would result from a LORAKS-type penalty in the original objective rather than the nuclear norm.

VII. Todo for Paper

1) Study $\mathbf{P}^*\mathbf{P} \approx \mathbf{I}$ approximation.
2) Study conditioning of Toeplitz system. Look at preconditioner?

VIII. Experiments

1) Empirical justication for $\mathbf{P}^*\mathbf{P} \approx \mathbf{I}$.
2) 1-D/2-D spectral estimation on synthetic data, as in [7].
    - Plot of SNR vs. CPU Time / # iterations
    - Compare against singular value thresholding, UV factorization(?), LORAKS-style penalty.
3) 2-D PWC images–Shepp-Logan/Real Data? **ISBI
4) Wavelet case **ISBI

IX. Conclusion

References

[1] J. Haldar, "Low-rank modeling of local k -space neighborhoods (loraks) for constrained MRI," *Medical Imaging, IEEE Transactions on*, vol. 33, no. 3, pp. 668–681, March 2014.

[2] P. J. Shin, P. E. Larson, M. A. Ohliger, M. Elad, J. M. Pauly, D. B. Vigneron, and M. Lustig, "Calibrationless parallel imaging reconstruction based on structured low-rank matrix completion," *Magnetic Resonance in Medicine*, 2013.

[3] W. Xu, J.-F. Cai, K. V. Mishra, M. Cho, and A. Kruger, "Precise semidefinite programming formulation of atomic norm minimization for recovering d-dimensional (d 2) off-the-grid frequencies," in *Information Theory and Applications Workshop (ITA), 2014*. IEEE, 2014, pp. 1–4.

[4] G. Ongie and M. Jacob, "Super-resolution MRI using finite rate of innovation," *IEEE International Symposium on Biomedical Imaging: ISBI 2015.*, 2015. [Online]. Available: http://arxiv.org/abs/1501.01697

[5] M. Fazel, T. K. Pong, D. Sun, and P. Tseng, "Hankel matrix rank minimization with applications to system identification and realization," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 946–977, 2013.

[6] K. Mohan and M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 3441–3473, 2012.

[7] Y. Chen and Y. Chi, "Robust spectral compressed sensing via structured matrix completion," *Information Theory, IEEE Transactions on*, vol. 60, no. 10, pp. 6576–6601, 2014.