JOINT LOW-RANK REPRESENTATION AND MATRIX COMPLETION UNDER A SINGULAR VALUE THRESHOLDING FRAMEWORK

Christos Tzagkarakis*‡ *Stephen Becker*† *Athanasios Mouchtaris**‡

*University of Crete, Department of Computer Science, Heraklion, Crete, Greece
†IBM T. J. Watson Research Center, Yorktown Heights, New York
‡Institute of Computer Science, Foundation for Research & Technology Hellas, Heraklion, Crete, Greece
{tzagarak,mouchtar}@ics.forth.gr, srbecker@us.ibm.com

ABSTRACT

Matrix completion is the process of estimating missing entries from a matrix using some prior knowledge. Typically, the prior knowledge is that the matrix is low-rank. In this paper, we present an extension of standard matrix completion that leverages prior knowledge that the matrix is low-rank *and* that the data samples can be efficiently represented by a fixed known dictionary. Specifically, we compute a low-rank representation of a data matrix with respect to a given dictionary using only a few observed entries. A novel modified version of the singular value thresholding (SVT) algorithm named joint low-rank representation and matrix completion SVT (J-SVT) is proposed. Experiments on simulated data show that the proposed J-SVT algorithm provides better reconstruction results compared to standard matrix completion.

Index Terms— low-rank representation, matrix completion, singular value thresholding, dictionary representation

1. INTRODUCTION

Many real-world problems often require the estimation of a matrix with missing entries. In general, the *matrix completion* problem involves the computation of the missing entries in a partially observed data matrix by imposing high data redundancy constraints through a *low-rank model*. The seminal papers [1, 2] prove that in many cases, the matrix can be correctly estimated with high probability from a number of observed entries greater than or equal to a certain constant value. The estimation is in the form of a rank minimization problem, where the *nuclear norm* [3], i.e., sum of the singular values, is used as the convex surrogate of the rank function.

Many algorithms have been proposed to solve the matrix completion (MC) problem. They can be summarized into two main categories with respect to the nature of the optimization problem. The first group of algorithms employs nuclear norm minimization such as in singular value thresholding (SVT) [4], templates for first-order conic solvers (TFOCS) [5], accelerated proximal gradient (APGL) [6] and augmented Lagrange multiplier (ALM) [7]. The second class of MC algorithms minimizes an approximation error objective function on a Grassmann manifold as examined in OPTSPACE [8], subspace evolution and transfer (SET) [9], Grassmanian rank-one update subspace estimation (GROUSE) [10], scaled gradients on Grassmann manifolds (ScGrassMC) [11], etc. Additionally, the low-rank matrix fitting algorithm (LMaFit) [12] optimizes an approximation error objective function based on the nuclear norm minimization framework, while in [13] MC is studied from a Bayesian point of view.

Over the last few years, MC has been tested in a wide range of practical applications including robust video denoising [14], bearing estimation of narrowband sources in sensor arrays [15], received signal-strength fingerprint based indoor localization in wireless local area networks [16] and audio bandwidth expansion [17]. It has also been utilized for other scientific problems such as position calibration in circular ultrasound tomography devices [18], high-quality reconstructions for large scale seismic interpolation problems [19], etc.

Nuclear norm minimization for subspace segmentation has been developed in parallel with MC since the germinal work introduced in [20]. The described *low-rank representation* idea looks for the lowest rank estimate of a data matrix with respect to a collection of data drawn from a union of multiple subspaces. Specifically, a learned dictionary or the data matrix itself can be exploited for seeking the low-rank representation (LRR) of the data. LRR seems to be very promising especially for classification tasks. For example, [21, 22] show that minimizing a nuclear norm based objective function coupled with sparsity constraints and a discriminative (or supervised) term enhances the power to discriminate features in image recognition. In [23] LRR is also adopted for music tagging, while in [24] is extended to the case of multiple dictionaries for music and singing voice separation.

Contributions. In this paper, we propose a joint LRR and MC approach in the light of SVT framework. Especially, we are interested in studying the effect of *estimating the lowest rank representation of a data matrix with respect to a given basis or dictionary connected with a partially observed version of it under an SVT scheme*. A dictionary based MC method has been recently proposed in [25], where a similar optimization problem is examined for reconstruction and classification of simulated sensor network data using the CVX software package [26]. This method can potentially solve problems of very small size, however, the computational time is prohibitive for practical applications even for data matrices of moderate size. The novelty of the current paper is twofold. Firstly, in the current work a more rigorous mathematical formulation of the joint LRR and MC problem is presented by restating the optimization problem and giving a detailed algorithmic process for the estimation of the data matrix. Secondly, we employ an SVT algorithmic solution especially targeted for medium scale data, where an experimental evaluation is performed on synthetic data proving the efficacy of the proposed method. To the best of our knowledge, this is the first time

This research has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thalis-DISFER

that LRR is connected with MC under an SVT algorithmic process. Our proposed approach can be regarded as an enhanced version of SVT in the case that we have knowledge of the data generation process via a dictionary or basis. Therefore, we are strongly interested in examining the performance of the proposed algorithm versus the performance of the typical SVT algorithm under these conditions.

The rest of the paper is organized as follows: Section 2 briefly overviews the matrix completion problem solved by the SVT algorithm, while Section 3 describes the proposed joint LRR and MC approach along with an SVT-based solution. An experimental evaluation of the proposed technique compared with typical SVT algorithm is described in Section 4. Finally, Section 5 summarizes the main conclusions and gives directions for future research work. For notation, we use $\|\cdot\|_2$ for the Euclidean norm on vectors and $\|\cdot\|_F$ and $\|\cdot\|$ for the Frobenius and spectral/operator norm on matrices, respectively.

2. SINGULAR VALUE THRESHOLDING FOR MATRIX COMPLETION

Matrix completion (MC) enables the recovery of a low rank or approximately low rank matrix $\boldsymbol{M} \in \mathbb{R}^{n_1 \times n_2}$ from at least $O(nr\nu \ln^2 n)$ entries selected uniformly at random (with ν corresponding to the so-called degree of incoherence) [27], where $n = \max\{n_1, n_2\}$ and $r = \mathrm{rank}(\boldsymbol{M})$. Here and for the rest of the paper we will assume that all the scalars, vectors and matrices are real-valued. The original matrix can be recovered from the partially observed matrix by solving the following convex optimization problem

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \|\boldsymbol{X}\|_* \\ \text{s.t.} \quad & X_{ij} = M_{ij}\,, \ \ (i,j) \in \mathcal{I} \subset \{1,\ldots,n_1\} \times \{1,\ldots,n_2\}, \end{aligned} \tag{1}$$

where $k = |\mathcal{I}| \geq Cnr \ln^2 n$ denotes the number of observed entries (C is a positive constant), $\boldsymbol{X} \in \mathbb{R}^{n_1 \times n_2}$ is the decision variable and the nuclear norm is defined as $\|\boldsymbol{X}\|_* = \sum_{q=1}^{\min\{n_1,n_2\}} \sigma_q$ with $\sigma_1, \ldots, \sigma_{\min\{n_1,n_2\}} \geq 0$ corresponding to the singular values of $\boldsymbol{X}$.

Let us introduce the standard matrix completion *linear map* $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \rightarrow \mathbb{R}^k$. The constraints $X_{ij} = M_{ij}\,,\ \forall (i,j) \in \mathcal{I}$ in (1) can be represented by using the linear map $\mathcal{A}_{\mathcal{I}}$ as follows

$$\min_{\boldsymbol{X}} \ \|\boldsymbol{X}\|_* \ \ \text{s.t. } \mathcal{A}_{\mathcal{I}}(\boldsymbol{X}) = \vec{\boldsymbol{b}}, \tag{2}$$

where $\vec{\boldsymbol{b}} := \mathcal{A}_{\mathcal{I}}(\boldsymbol{M})$ contains the sample values extracted from $\boldsymbol{M}$. Each row of $\mathcal{A}_{\mathcal{I}}(\boldsymbol{M})$ corresponds the sampling of a single (i,j) element of $\boldsymbol{M}$. The equality constraint in (2) can also be written in matrix form

$$(\forall \boldsymbol{X} \in \mathbb{R}^{n_1 \times n_2})\ \mathcal{A}_{\mathcal{I}}(\boldsymbol{X}) \equiv \boldsymbol{A}\vec{\boldsymbol{x}},\ \vec{\boldsymbol{x}} := \mathrm{vec}(\boldsymbol{X}) \tag{3}$$

where $\boldsymbol{A} \in \mathbb{R}^{k \times n_1 n_2}$ and $\mathrm{vec}(\cdot) : \mathbb{R}^{n_1 \times n_2} \rightarrow \mathbb{R}^{n_1 n_2 \times 1}$ denotes the vectorization mapping; any vectorization mapping (e.g., row major order or column major order) is acceptable as long as it is fixed. In matrix completion, each row of $\boldsymbol{A}$ contains exactly 1 non-zero entry.

We also make use of the adjoint of $\mathcal{A}_{\mathcal{I}}$ which takes a vector and maps it to a sparse matrix with the nonzero entries of the sparse matrix corresponding to $\mathcal{I}$. Specifically,

$$\mathcal{A}_{\mathcal{I}}^*(\cdot) : \ \mathbb{R}^{k \times 1} \rightarrow \mathbb{R}^{n_1 \times n_2} \ \text{ with } \ k = |\mathcal{I}| \leq n_1 n_2,$$

and we have the property

$$(\forall \vec{\boldsymbol{h}} \in \mathbb{R}^{k \times 1})\ \vec{\boldsymbol{h}} = \mathcal{A}_{\mathcal{I}}(\mathcal{A}_{\mathcal{I}}^*(\vec{\boldsymbol{h}})).$$

This work is based on the SVT [4] algorithm for MC problems since SVT is efficient and, as we will show, can be extended to solve the joint LRR-MC problem. Specifically, SVT minimizes the following constraint problem

$$\min_{\boldsymbol{X}} \ \tau \|\boldsymbol{X}\|_* + \frac{1}{2} \|\boldsymbol{X}\|_F^2 \ \ \text{s.t. } \mathcal{A}_{\mathcal{I}}(\boldsymbol{X}) = \mathcal{A}_{\mathcal{I}}(\boldsymbol{M}), \tag{4}$$

where the positive constant τ is a trade off between the nuclear and Frobenius norm. The solution to problem (4) converges to that of (1) as $\tau \rightarrow \infty$.

SVT comprises the two following iterative steps

$$\begin{cases} \boldsymbol{X}_t = \mathcal{D}_\tau(\mathcal{A}_{\mathcal{I}}^*(\vec{\boldsymbol{y}}_{t-1})) \\ \vec{\boldsymbol{y}}_t = \vec{\boldsymbol{y}}_{t-1} - \delta_t(\mathcal{A}_{\mathcal{I}}(\boldsymbol{X}_t) - \vec{\boldsymbol{b}}). \end{cases} \tag{5}$$

In the above equation the shrinkage operator $\mathcal{D}_\tau$, also known as *soft-thresholding operator*, is denoted as $\mathcal{D}_\tau = \boldsymbol{U}\boldsymbol{\Sigma}_\tau \boldsymbol{V}^T$ where $\boldsymbol{U}$ and $\boldsymbol{V}$ are matrices with orthonormal columns and $\boldsymbol{\Sigma}_\tau = \mathrm{diag}(\max\{\sigma_i - \tau, 0\})$ with $\{\sigma_i\}_{i=1}^{\min\{n_1,n_2\}}$ corresponding to the singular values of the decomposed matrix. The step size of the iterative algorithmic process is given by δ_t.

3. JOINT LOW RANK REPRESENTATION AND MATRIX COMPLETION USING SVT

Singular value decomposition (SVD) followed by soft-thresholding on the computed singular values constitutes the core of the SVT algorithm described in Section 2. Any procurable information of the underlying procedure that generated the data matrix $\boldsymbol{M}$ is not taken into account by MC. Sometimes this property is considered as an asset since it does not require the explicit knowledge of such a generation procedure. In other cases, however, extra information about the data matrix is available and exploiting this knowledge can lead to more accurate solutions of different tasks at hand.

As mentioned in Section 1, the low-rank representation (LRR) approach has been recently introduced as an alternative to typical subspace-based methods like the SVD. The goal is to find the lowest rank representation of a data matrix by solving the following convex optimization problem

$$\min_{\boldsymbol{L}} \ \|\boldsymbol{L}\|_* \ \ \text{s.t. } \boldsymbol{M} = \boldsymbol{M}\boldsymbol{L}, \tag{6}$$

where $\boldsymbol{M}$ is the data matrix and $\boldsymbol{L}$ is a low-rank matrix. Adopting the LRR formulation, let us assume that the additional information of the data matrix $\boldsymbol{M}$ can be modelled according to a specific matrix decomposition of the form $\boldsymbol{M} = \boldsymbol{G}\boldsymbol{L}$, where $\boldsymbol{G}$ is a known dictionary and $\boldsymbol{L}$ is a low-rank matrix containing the corresponding representation coefficients. Thus, problem (6) can be formulated as

$$\min_{\boldsymbol{L}} \ \|\boldsymbol{L}\|_* \ \ \text{s.t. } \boldsymbol{M} = \boldsymbol{G}\boldsymbol{L}. \tag{7}$$

To apply the LRR scheme on matrices with missing data, we use the linear sampling operator $\mathcal{A}_{\mathcal{I}}$. The proposed sampling scheme is a combination of MC and LRR and seeks a low-rank coefficient matrix $\boldsymbol{L}$ from a small number of measurements $\mathcal{A}_{\mathcal{I}}(\boldsymbol{M})$. Thus, the convex optimization problem takes the form below

$$\min_{\boldsymbol{L}} \ \|\boldsymbol{L}\|_* \ \ \text{s.t. } \mathcal{A}_{\mathcal{I}}(\boldsymbol{X}) = \mathcal{A}_{\mathcal{I}}(\boldsymbol{M}) \text{ and } \boldsymbol{X} = \boldsymbol{G}\boldsymbol{L}. \tag{8}$$

The goal is to efficiently solve problem (8) in the context of the SVT algorithm so that we can solve large-scale problems. Hence, combining (4) and (8) we get the joint LRR and MC version of SVT dubbed J-SVT defined as follows

$$\min_{\boldsymbol{L}} \ \tau \|\boldsymbol{L}\|_* + \frac{1}{2} \|\boldsymbol{L}\|_F^2 \ \ \text{s.t. } \mathcal{A}_{\mathcal{I}}(\boldsymbol{X}) = \vec{\boldsymbol{b}} \text{ and } \boldsymbol{X} = \boldsymbol{G}\boldsymbol{L}, \tag{9}$$

where $\boldsymbol{X} \in \mathbb{R}^{n_1 \times n_2}$, $\boldsymbol{M} \in \mathbb{R}^{n_1 \times n_2}$, $\boldsymbol{G} \in \mathbb{R}^{n_1 \times K}$, $\boldsymbol{L} \in \mathbb{R}^{K \times n_2}$ and K denotes the size of the dictionary. In the J-SVT problem (9),

Algorithm 1: J-SVT algorithm

Input: $\mathcal{A}_{\mathcal{I}}$, observed values $\vec{\boldsymbol{b}}$, dictionary $\boldsymbol{G}$, step size δ, tolerance ϵ, parameter $\tau > 0$, maximum iterations $t_{\max}$
Output: estimated matrix $\boldsymbol{X} = \boldsymbol{G}\boldsymbol{L}_T$

1 **Initialization:** $\vec{\boldsymbol{y}}_1 = \tau \vec{\boldsymbol{b}} / \|\boldsymbol{G}^T \mathcal{A}_{\mathcal{I}}^*(\vec{\boldsymbol{b}})\|$
2 **for** $t = 1$ **to** $t_{\max}$ **do**
3 $\quad [\boldsymbol{U}_t, \boldsymbol{\Sigma}_t, \boldsymbol{V}_t, s_t] = \texttt{SVDshrink}(\boldsymbol{G}^T \mathcal{A}_{\mathcal{I}}^*(\vec{\boldsymbol{y}}_t), \tau, s_{t-1})$
4 $\quad \boldsymbol{L}_t = \boldsymbol{U}_t \boldsymbol{\Sigma}_t \boldsymbol{V}_t^T$
5 $\quad$ **if** $\left\|\mathcal{A}_{\mathcal{I}}(\boldsymbol{G}\boldsymbol{L}_t) - \vec{\boldsymbol{b}}\right\|_2 \le \epsilon \left\|\vec{\boldsymbol{b}}\right\|_2$ **then**
6 $\qquad$ `break`
7 $\quad$ **end**
8 $\quad \vec{\boldsymbol{y}}_{t+1} = \vec{\boldsymbol{y}}_t - \delta(\mathcal{A}_{\mathcal{I}}(\boldsymbol{G}\boldsymbol{L}_t) - \vec{\boldsymbol{b}})$
9 **end**

we consider the additional constraint that $\boldsymbol{X}$ must be in the form $\boldsymbol{X} = \boldsymbol{G}\boldsymbol{L}$ for a fixed dictionary $\boldsymbol{G}$. This constraint only amounts to changing the linear operator, and that does not affect the convergence proofs of SVT under a correctly scaled δ. Recall that SVT converges with $\delta < 2\|\mathcal{A}_{\mathcal{I}}\|^{-2}$. We have the following similar result:

Theorem 3.1. *With step-size $\delta < 2\|\mathcal{A}_{\mathcal{I}} \circ \boldsymbol{G}\|^{-2}$, J-SVT produces a sequence $\boldsymbol{L}_t$ that converges to the unique minimizer of* (9).

Proof. The proof of convergence for the SVT algorithm only uses the fact that $\mathcal{A}_{\mathcal{I}}$ is a linear operator and can be extended to handle a generic linear operator $\mathcal{A}$. By letting $\mathcal{A} = \mathcal{A}_{\mathcal{I}} \circ \boldsymbol{G}$ and $\mathcal{A}^* = \boldsymbol{G}^T \mathcal{A}_{\mathcal{I}}^*$ we arrive at J-SVT. The step-size must satisfy $\delta < 2\|\mathcal{A}\|^{-2} = 2\|\mathcal{A}_{\mathcal{I}} \circ \boldsymbol{G}\|^{-2}$. ☐

Since $\|\mathcal{A}_{\mathcal{I}} \circ \boldsymbol{G}\| \le \|\boldsymbol{G}\|$, the step-size can best estimated using any upper bound on the spectral norm of $\boldsymbol{G}$.

Algorithm 2 implements the `SVDshrink` operation. The `partialSVD`$(\boldsymbol{Z}, s)$ algorithm returns the top s singular values and singular vectors. The most common computational approach is the Lanczos method. Here, we use the implementation in PROPACK, which re-orthogonalizes the singular vectors as needed in order to improve numerical stability. These Lanczos methods only require matrix-vector multiplies of the form $\boldsymbol{Z}\vec{\boldsymbol{u}}$ and $\boldsymbol{Z}^T\vec{\boldsymbol{v}}$, and thus we take advantage of sparsity in $\boldsymbol{Z}$. If $\boldsymbol{G}^T$ has a fast transform, we can also take advantage of this, and never even need to explicitly form the $\boldsymbol{G}$ or $\boldsymbol{G}^T$ matrix (e.g., if $\boldsymbol{G}$ is the FFT or FFT-based).

Algorithm 2: `SVDshrink` algorithm

Input: internal integer parameter ℓ
1 **function** `SVDshrink`$(\boldsymbol{Z}, \tau, s_0)$
2 $\quad s \leftarrow s_0 + 1$
3 $\quad$ **repeat**
4 $\qquad [\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}] = \texttt{partialSVD}(\boldsymbol{Z}, s)$
5 $\qquad s \leftarrow s + \ell$
6 $\quad$ **until** $\boldsymbol{\Sigma}_{s,s} \le \tau$
7 $\quad$ **return** $[\boldsymbol{U}, \mathcal{D}_\tau(\boldsymbol{\Sigma}), \boldsymbol{V}, s]$
8 **end function**

In another improvement on regular SVT, we introduce the Nesterov accelerated [28] version, which applies to both MC and LRR-MC problems.

Algorithm 3: Accelerated J-SVT algorithm: identical to J-SVT except replace line 8 in J-SVT with the following and initialize $\vec{\boldsymbol{z}}_1 = \vec{\boldsymbol{y}}_1$.

8 $\vec{\boldsymbol{z}}_{t+1} = \vec{\boldsymbol{y}}_t - \delta(\mathcal{A}_{\mathcal{I}}(\boldsymbol{G}\boldsymbol{L}_t) - \vec{\boldsymbol{b}})$
9 $\vec{\boldsymbol{y}}_{t+1} = \vec{\boldsymbol{z}}_{t+1} + \frac{t}{t+3}(\vec{\boldsymbol{z}}_{t+1} - \vec{\boldsymbol{z}}_t)$

Theorem 3.2. *Algorithm 3 produces a sequence $\boldsymbol{L}_t$ that converges to the unique minimizer of* (9) *if $\delta \le \|\mathcal{A}_{\mathcal{I}} \circ G\|^{-2}$.*

Proof. This is a special case of the framework in [5] and the strong convexity of the objective. ☐

Note that we have lost a factor of 2 in the step-size bound in the accelerated version, which is because we can no longer over-relax (see [29]). Despite the smaller step-size, it has faster convergence rate guarantees and typically works faster in practice.

4. EXPERIMENTAL RESULTS

In this section, we compare the reconstruction performance of the proposed J-SVT scheme with the performance obtained by reconstructing the missing data matrix using the SVT algorithm. For this purpose, we perform simulations on synthetic data, where the dictionary $\boldsymbol{G}$ and the low-rank representation matrix $\boldsymbol{L}$ are generated from normally distributed random samples. As an evaluation metric, we employ the relative error, which is defined as follows:

$$\text{Relative error} = \frac{\left\|\hat{\boldsymbol{X}} - \boldsymbol{M}\right\|_F}{\|\boldsymbol{M}\|_F},$$

where $\hat{\boldsymbol{X}}$ is the recovered matrix and $\boldsymbol{M}$ is the original full data matrix. In the present case study, the size of the original data matrix $\boldsymbol{M}$ is set equal to $n_1 \times n_2 = 300 \times 500$. The maximum number of iterations $t_{\max}$, the tolerance ϵ and the parameter τ are set equal to 100, 10^{-5} and $5\sqrt{n_1\, n_2}$, respectively. The step size δ is set equal to 1.9 in the case of SVT, while for the accelerated version of J-SVT we use $\delta = \|G\|^{-2}$. In the subsequent experimental evaluation, the reconstruction performance of both the J-SVT and SVT algorithms is also examined as a function of the sampling ratio, which is given by

$$\text{Sampling ratio} = \frac{\text{number of observed values (k)}}{\text{matrix size } (\mathrm{n_1} \times \mathrm{n_2})}.$$

Based on 10 Monte Carlo runs for each scenario, the total average were computed to show the overall relative errors for each algorithm.

As a first set of experiments, we examine the reconstruction performance of J-SVT for a varying matrix rank. Figure 1 reveals that our proposed J-SVT algorithm outperforms clearly the SVT counterpart in case of a dictionary size 300×1500. More specifically, Figure 1.(a) shows that the relative error achieved by J-SVT is almost zero for a sampling ratio $(\mathrm{SR}) > 0.3$, while the relative error achieved by SVT approaches zero for a significantly higher sampling ratio $\mathrm{SR} > 0.7$. The effect of a varying matrix rank is shown in Figures 1.(b)-(d), which depict the reconstruction performance for matrix ranks equal to 40, 70 and 100, respectively. As it can be seen,

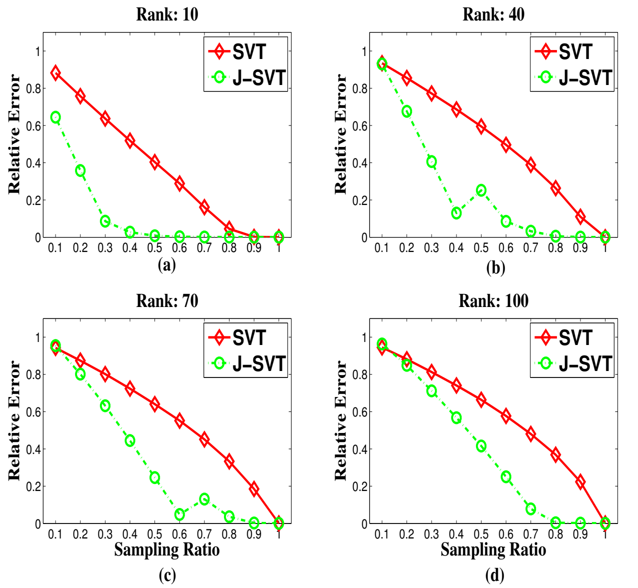

Fig. 1. Relative error as a function of sampling ratio. The size of dictionary $\boldsymbol{G}$ is 300×1500. The rank of matrix $\boldsymbol{L}$ is: (a) 10, (b) 40, (c) 70 and (d) 100.

the relative error corresponding to J-SVT is close to zero for SR ≈ 0.7, whereas the relative error of SVT approaches zero only for an almost full sampling (SR ≈ 0.9).

The second set of experiments concerns the performance evaluation of the two algorithms by varying the dictionary size. In Figure 2,

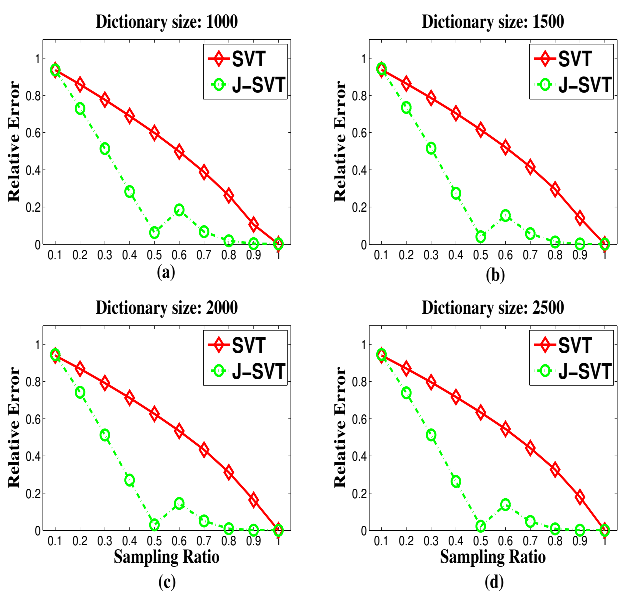

Fig. 2. Relative error as a function of sampling ratio. The rank of matrix $\boldsymbol{L}$ is 50. The size of dictionary $\boldsymbol{G}$ is: (a) 300×1000, (b) 300×1500, (c) 300×2000 and (d) 300×2500.

the reconstruction accuracy of J-SVT is compared with the performance of SVT for dictionary sizes of 300×1000, 300×1500, 300×2000 and 300×2500, by fixing $\text{rank}(\boldsymbol{L}) = 50$. Clearly, J-SVT outperforms again SVT, while we highlight the approximately constant recovery behaviour of J-SVT regardless of the dictionary size. This observation is very important, since it reveals that J-SVT is highly robust, in terms of achieving a low reconstruction error, even in case of small-sized dictionaries, which represent our data in a compact way. This comes also as a significant advantage of J-SVT towards its application in practical scenarios, where the size of the dictionary comes at the expense of an increased computational and memory complexity.

As a final experimental evaluation, we compare the robustness of J-SVT against SVT under noisy conditions. In particular, the relative error curves presented in Figure 3 correspond to observed data corrupted by additive white noise, with the signal-to-noise ratio (SNR) being equal to 10, 15, 20 and 25 dB. As it can be seen J-SVT achieves a significantly improved reconstruction quality in regard with SVT. Especially in Figure 3.(b)-(d), SVT has almost twice as high relative error on average for the same range of sampling ratio values. As expected, the performance of SVT converges to the performance of J-SVT for a full sampling ratio ($= 1$).

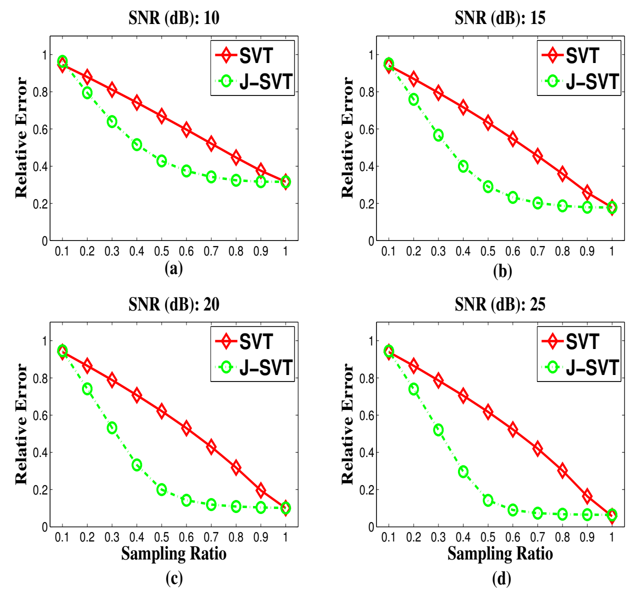

Fig. 3. Relative error as a function of sampling ratio. The size of dictionary $\boldsymbol{G}$ is 300×1500 and the rank of matrix $\boldsymbol{L}$ is 50. The SNR level is set to: (a) 10 dB, (b) 15 dB, (c) 20 dB and (d) 25 dB.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a joint low-rank representation and matrix completion scheme under a singular value thresholding framework. It was shown, through an experimental evaluation on synthetic data, that the proposed approach outperforms significantly the SVT algorithm, in terms of reconstruction error, and in case that we are aware of the data generation process via an appropriate dictionary. As a future work, we intend to perform simulations on real speech data in the context of missing data imputation. Towards this direction, robust PCA could also be used to produce low-rank speech features specially targeted for the J-SVT framework. We are also interested in reducing the computational complexity of the proposed approach by taking advantage of the dictionary structure. Besides, we intend to study if the mathematical formulation of J-SVT can be adopted from other algorithmic frameworks, such as augmented Lagrange multipliers method.

6. REFERENCES

[1] E. J. Candés and B. Recht, "Exact matrix completion via convex optimization," *Journal on Foundations of Computational Mathematics*, vol. 9(6), pp. 717–772, December 2009.

[2] E. J. Candés and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98(6), pp. 925–936, June 2010.

[3] M. Fazel, H. Hindi, and S. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. American Control Conf.*, June 2001, pp. 4734–4739.

[4] J. F. Cai, E. J. Candés, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20(4), pp. 1956–1982, March 2010.

[5] S. Becker, E. J. Candés, and M. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Mathematical Programming Computation*, vol. 3, no. 3, pp. 165–218, 2011.

[6] K. C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, pp. 615–640, 2010.

[7] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," UIUC Technical Report UILU-ENG-09-2215, October 2009.

[8] R. H. Keshavan and S. Oh, "A gradient descent algorithm on the Grassman manifold for matrix completion," arXiv:0910.5260, 2009.

[9] W. Dai, O. Milenkovic, and E. Kerman, "Subspace evolution and transfer (SET) for low-rank matrix completion," *IEEE Trans. on Signal Processing*, vol. 59, no. 7, pp. 3120–3132, July 2011.

[10] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. Annual Allerton Conf. on Communication, Control and Computing*, October 2010, pp. 704–711.

[11] T. Ngo and Y. Saad, "Scaled gradients on Grassmann manifolds for matrix completion," in *Advances in Neural Information Processing Systems (NIPS)*, Lake Tahoe, Nevada, December 2012, pp. 1421–1429.

[12] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *Mathematical Programming Computation*, vol. 4, no. 4, pp. 333–361, December 2012.

[13] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, "Sparse bayesian methods for low-rank matrix estimation," *IEEE Trans. on Signal Processing*, vol. 60, no. 8, pp. 3964–3977, August 2012.

[14] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, June 2010, pp. 1791–1798.

[15] A. Waters and V. Cevher, "Distributed bearing estimation via matrix completion," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, Texas, USA, March 2010, pp. 2590–2593.

[16] S. Nikitaki, G. Tsagkatakis, and P. Tsakalides, "Efficient training for fingerprint based positioning using matrix completion," in *Proc. European Signal Proc. Conf. (EUSIPCO)*, Bucharest, Romania, August 2012.

[17] D. L. Sun and R. Mazumder, "Non-negative matrix completion for bandwidth extension: A convex optimization approach," in *Proc. IEEE Conf. on Machine Learning for Signal Processing (MLSP)*, Southampton, UK, September 2013.

[18] R. Parhizkar, A. Karbasi, S. Oh, and M. Vetterli, "Calibration using matrix completion with application to ultrasound tomography," *IEEE Trans. on Signal Processing*, vol. 61, no. 20, pp. 4923–4933, October 2013.

[19] A. Y. Aravkin, R. Kumar, H. Mansour, B. Recht, and F. J. Herrmann, "A robust SVD-free approach to matrix completion, with applications to interpolation of large scale data," arXiv:1302.4886, 2013.

[20] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proc. Int. Conf. on Machine Learning*, Haifa, Israel, June 2010, pp. 663–670.

[21] Y. Zhang, Z. Jiang, and L. S. Davis, "Learning structured low-rank representations for image classification," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Portland, Oregon, USA, June 2013, pp. 676–683.

[22] D. Pei, F. Sun, and H. Liu, "Supervised low-rank matrix recovery for traffic sign recognition in image sequences," *IEEE Signal Proc. Letters*, vol. 20, no. 3, pp. 241–244, March 2013.

[23] Y. Panagakis and C. Kotropoulos, "Automatic music tagging by low-rank representation," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012, pp. 497–500.

[24] Y H. Yang, "Low-rank representation of both singing voice and music accompaniment via learned dictionaries," in *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, Curitiba, Brazil, November 2013.

[25] G. Tsagkatakis and P. Tsakalides, "Dictionary based reconstruction and classification of randomly sampled sensor network data," in *Proc. Sensor Array and Multichannel Signal Proc. Workshop*, Hoboken, NJ, June 2012, pp. 117–120.

[26] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming, version 1.21," 2011.

[27] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Trans. on Information Theory*, vol. 57, no. 3, pp. 1548–1566, March 2011.

[28] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, Springer Netherlands, 2004.

[29] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM Multiscale Model. Simul.*, vol. 4, no. 4, pp. 1168–1200, 2005.