# GoDec: Randomized Low-rank & Sparse Matrix Decomposition in Noisy Case

**Tianyi Zhou** TIANYI.ZHOU@STUDENT.UTS.EDU.AU
**Dacheng Tao** DACHENG.TAO@UTS.EDU.AU
Centre for Quantum Computation & Intelligent Systems, FEIT, University of Technology, Sydney, NSW 2007, Australia

## Abstract

Low-rank and sparse structures have been profoundly studied in matrix completion and compressed sensing. In this paper, we develop "Go Decomposition" (GoDec) to efficiently and robustly estimate the low-rank part $L$ and the sparse part $S$ of a matrix $X = L + S + G$ with noise $G$. GoDec alternatively assigns the low-rank approximation of $X - S$ to $L$ and the sparse approximation of $X - L$ to $S$. The algorithm can be significantly accelerated by bilateral random projections (BRP). We also propose GoDec for matrix completion as an important variant. We prove that the objective value $\|X - L - S\|_F^2$ converges to a local minimum, while $L$ and $S$ linearly converge to local optimums. Theoretically, we analyze the influence of $L$, $S$ and $G$ to the asymptotic/convergence speeds in order to discover the robustness of GoDec. Empirical studies suggest the efficiency, robustness and effectiveness of GoDec comparing with representative matrix decomposition and completion tools, e.g., Robust PCA and OptSpace.

## 1. Introduction

It has proven in compressed sensing (Donoho, 2006) that a sparse signal can be exactly recovered from a small number of its random measurements, and in matrix completion (Keshavan & Oh, 2009) that a low-rank matrix can be exactly completed from a few of its entries sampled at random. When signals are neither sparse nor low-rank, its low-rank and sparse structure can be explored by either approximation or decomposition.

Recent research about exploring low-rank and sparse structures (Zhou et al., 2011) concentrates on developing fast approximations and meaningful decompositions. Two appealing representatives are the randomized approximate matrix decomposition (Halko et al., 2009) and the robust principal component analysis (RPCA) (Candès et al., 2009). The former proves that a matrix can be well approximated by its projection onto the column space of its random projections. This rank-revealing method provides a fast approximation of SVD/PCA. The latter proves that the low-rank and the sparse components of a matrix can be exactly recovered if it has a unique and precise "low-rank+sparse" decomposition. RPCA offers a blind separation of low-rank data and sparse noises.

In this paper, we first consider the problem of fast low-rank approximation. Given $r$ bilateral random projections (BRP) of an $m \times n$ dense matrix $X$ (w.l.o.g, $m \geq n$), i.e., $Y_1 = XA_1$ and $Y_2 = X^T A_2$, wherein $A_1 \in \mathbb{R}^{n\times r}$ and $A_2 \in \mathbb{R}^{m\times r}$ are random matrices,

$$L = Y_1 \left(A_2^T Y_1\right)^{-1} Y_2^T \quad (1)$$

is a fast rank-$r$ approximation of $X$. The computation of $L$ includes an inverse of an $r \times r$ matrix and three matrix multiplications. Thus, for a dense $X$, $2mnr$ floating-point operations (flops) are required to obtain BRP, $r^2(2n+r)+mnr$ flops are required to obtain $L$. The computational cost is much less than SVD based approximation. The $L$ in (1) has been proposed in (Fazel et al., 2008) as a recovery of a rank-$r$ matrix $X$ from $Y_1$ and $Y_2$, where $A_1$ and $A_2$ are independent Gaussian/SRFT random matrices. However, we propose that $L$ is a tight rank-$r$ approximation to a full rank matrix $X$, when $A_1$ and $A_2$ are correlated random matrices updated from $Y_2$ and $Y_1$, respectively. We then apply power scheme (Roweis, 1998) to $L$ for improving the approximation precision, especially when the eigenvalues of $X$ decay slowly. The error of BRP based approximation approaches to the error of SVD approximation under mild conditions. Compared to randomized SVD (Halko et al., 2009) that extracts the column space from unilateral random projections, the BRP based method estimates both column and row spaces from bilateral random projections.

We then study the approximated "low-rank+sparse" decomposition of a matrix $X$, i.e.,

$$X = L + S + G, \mathrm{rank}(L) \leq r, \mathrm{card}(S) \leq k, \quad (2)$$

Appearing in *Proceedings of the $28^{th}$ International Conference on Machine Learning*, Bellevue, WA, USA, 2011. 

where $G$ is the noise. This problem is intrinsically different from RPCA that assumes $X = L + S$. In this paper, we develop "Go Decomposition" (GoDec) to estimate the low-rank part $L$ and the sparse part $S$ from $X$. We show that BRP can significantly accelerate GoDec.

In particular, GoDec alternatively assigns the $r$-rank approximation of $X - S$ to $L$ and assigns the sparse approximation with cardinality $k$ of $X - L$ to $S$. The updating of $L$ is obtained via singular value hard thresholding of $X - S$, while the updating of $S$ is obtained via entry-wise hard thresholding (Bredies & Lorenz, 2008) of $X - L$. The term "Go" is owing to the similarities between $L/S$ in the GoDec iteration rounds and the two players in the game of go. BRP based low-rank approximation is applied to accelerating the $r$-rank approximation of $X - S$ in GoDec. We show GoDec can be extended to solve matrix completion problem with competitive robustness and efficiency.

We theoretically analyze the convergence of GoDec. The objective value (decomposition error) $\|X-L-S\|_F^2$ monotonically decreases and converges to a local minimum. Since the updating of $L$ and $S$ in GoDec is equivalent to alternatively projecting $L$ or $S$ onto two smooth manifolds, we use the framework proposed in (Lewis & Malick, 2008) to prove the asymptotical property and linear convergence of $L$ and $S$. The asymptotic and convergence speeds are mainly determined by the angle between the two manifolds. We discuss how $L$, $S$ and $G$ influence the speeds via influencing the cosine of the angle. The analyses show the convergence of GoDec is robust to the noise $G$.

Both GoDec and RPCA can explore the low-rank and sparse structures in $X$, but they are intrinsically different. RPCA assumes $X = L + S$ ($S$ is sparse noise) and exactly decomposes $X$ into $L$ and $S$ without predefined $\mathrm{rank}(L)$ and $\mathrm{card}(S)$. However, GoDec produces approximated decomposition of a general matrix $X$ whose exact RPCA decomposition does not exist due to the additive noise $G$ and pre-defined $\mathrm{rank}(L)$ and $\mathrm{card}(S)$. In practice, $\mathrm{rank}(L)$ and $\mathrm{card}(S)$ are preferred to be restricted in order to control the model complexity. Another major difference is that GoDec directly constrains the rank range of $L$ and the cardinality range of $S$, while RPCA minimizes their corresponding convex polytopes, i.e., the nuclear norm of $L$ and $\ell_1$ norm of $S$. Chandrasekaran et al. (Chandrasekaran et al., 2009) proposed an exact decomposition based on a different assumption but the same optimization procedure used in RPCA. Stable principal component pursuit (Zhou et al., 2010) is an extension of RPCA to handle noise by minimizing the nuclear norm and $\ell_1$ norm. Therefore, they are different from GoDec. In addition, GoDec can be extended to solve matrix completion problems because it is able to control the support set of $S$, while RPCA cannot because the support set of $S$ is automatically determined.

GoDec has low computational cost in "low-rank+sparse" decomposition and matrix completion tasks. It is powerful in background modeling of videos and shadow/light removal of images. For example, it processes a 200 frame video with $256 \times 320$ resolution within 200 seconds, while RPCA requires $1,800+$ seconds.

In this paper, a standard Gaussian matrix is a random matrix whose entries are independent standard normal variables; the SVD of a matrix $X$ is $U\Lambda V^T$ and $\lambda_i$ or $\lambda_i(X)$ stands for the $i^{th}$ largest singular value of $X$; $\mathcal{P}_\Omega(\cdot)$ is the projection of a matrix to an entry set $\Omega$; and the QR decomposition of a matrix results in $Q$ and $R$.

## 2. Bilateral random projections (BRP) based low-rank approximation

We first introduce the bilateral random projections (BRP) based low-rank approximation and its power scheme modification.

### 2.1. Low-rank approximation with closed form

In order to improve the approximation precision of $L$ in (1), we use the obtained right random projection $Y_1$ to build a better left projection matrix $A_2$, and use $Y_2$ to build a better $A_1$. In particular, after $Y_1 = XA_1$, we update $A_2 = Y_1$ and calculate the left random projection $Y_2 = X^T A_2$, and then we update $A_1 = Y_2$ and calculate the right random projection $Y_1 = XA_1$. A better low-rank approximation $L$ will be obtained when the new $Y_1$ and $Y_2$ are applied to (1). This improvement requires additional flops of $mnr$.

### 2.2. Power scheme modification

When singular values of $X$ decay slowly, (1) may perform poorly. We design a modification for this situation based on the power scheme (Roweis, 1998). In the power scheme modification, we instead calculate BRP of a matrix $\tilde{X} = (XX^T)^q X$, whose singular values decay faster than $X$. In particular, $\lambda_i(\tilde{X}) = \lambda_i(\tilde{X})^{2q+1}$. Both $X$ and $\tilde{X}$ share the same singular vectors. The BRP of $\tilde{X}$ is:

$$Y_1 = \tilde{X}A_1, Y_2 = \tilde{X}^T A_2. \tag{3}$$

According to (1), the BRP based $r$ rank approximation of $\tilde{X}$ is:

$$\tilde{L} = Y_1 \left(A_2^T Y_1\right)^{-1} Y_2^T. \tag{4}$$

In order to obtain the approximation of $X$ with rank $r$, we calculate the QR decomposition of $Y_1$ and $Y_2$, i.e.,

$$Y_1 = Q_1 R_1, Y_2 = Q_2 R_2. \tag{5}$$

The low-rank approximation of $X$ is then given by:

$$L = \left(\tilde{L}\right)^{\frac{1}{2q+1}} = Q_1 \left[R_1 \left(A_2^T Y_1\right)^{-1} R_2^T\right]^{\frac{1}{2q+1}} Q_2^T. \quad (6)$$

The power scheme modification (6) requires an inverse of an $r \times r$ matrix, an SVD of an $r \times r$ matrix and five matrix multiplications. Therefore, for a dense $X$, $2(2q+1)mnr$ flops are required to obtain BRP, $r^2(m+n)$ flops are required to obtain the QR decompositions, $2r^2(n+2r)+mnr$ flops are required to obtain $L$. The power scheme modification reduces the error of (1) by increasing $q$. When the random matrices $A_1$ and $A_2$ are built from $Y_1$ and $Y_2$, $mnr$ additional flops are required in BRP.

In (Zhou & Tao, 2010), we show that the deterministic bound, average bound and deviation bound for the approximation error of BRP and its power scheme modification approach to those of SVD under mild conditions.

## 3. Go Decomposition (GoDec)

The approximated "low-rank+sparse" decomposition problem stated in (2) can be solved by minimizing the decomposition error:

$$\begin{aligned} \min_{L,S} \quad & \|X - L - S\|_F^2 \\ s.t. \quad & \operatorname{rank}(L) \leq r, \\ & \operatorname{card}(S) \leq k. \end{aligned} \quad (7)$$

### 3.1. Naïve GoDec

We propose the naïve GoDec algorithm in this section. The optimization problem of GoDec (7) can be solved by alternatively solving the following two subproblems until convergence:

$$\begin{cases} L_t = \arg \min\limits_{\operatorname{rank}(L) \leq r} \|X - L - S_{t-1}\|_F^2; \\ S_t = \arg \min\limits_{\operatorname{card}(S) \leq k} \|X - L_t - S\|_F^2. \end{cases} \quad (8)$$

Although both subproblems (8) have nonconvex constraints, their global solutions $L_t$ and $S_t$ exist.

In particular, the two subproblems in (8) can be solved by updating $L_t$ via singular value hard thresholding of $X - S_{t-1}$ and updating $S_t$ via entry-wise hard thresholding of $X - L_t$, respectively, i.e.,

$$\begin{cases} L_t = \sum\limits_{i=1}^{r} \lambda_i U_i V_i^T, \operatorname{svd}(X - S_{t-1}) = U \Lambda V^T; \\ S_t = \mathcal{P}_\Omega (X - L_t), \Omega : \left|(X - L_t)_{i,j \in \Omega}\right| \neq 0 \\ \quad \text{and} \geq \left|(X - L_t)_{i,j \in \overline{\Omega}}\right|, |\Omega| \leq k. \end{cases} \quad (9)$$

The main computation in the naïve GoDec algorithm (9) is the SVD of $X - S_{t-1}$ in the updating $L_t$ sequence. SVD requires $\min\left(mn^2, m^2 n\right)$ flops, so it is impractical when $X$ is of large size.

### 3.2. Fast GoDec via BRP based approximation

Since BRP based low-rank approximation is near optimal and efficient, we replace SVD with BRP in naïve GoDec in order to significantly reduce the time cost.

We summarize GoDec using BRP based low-rank approximation (1) and power scheme modification (6) in Algorithm 1. When $q = 0$, For dense $X$, (1) is applied. Thus the QR decomposition of $Y_1$ and $Y_2$ in Algorithm 1 are not performed, and $L_t$ is updated as $L_t = Y_1 \left(A_2^T Y_1\right)^{-1} Y_2^T$. In this case, Algorithm 1 requires $r^2(2n + r) + 4mnr$ flops per iteration. When integer $q > 0$, (6) is applied and Algorithm 1 requires $r^2(m + 3n + 4r) + (4q + 4)mnr$ flops per iteration.

**Algorithm 1** GoDec

**Input:** $X$, $r$, $k$, $\epsilon$, $q$
**Output:** $L$, $S$
**Initialize:** $L_0 := X$, $S_0 := \mathbf{0}$, $t := 0$
**while** $\|X - L_t - S_t\|_F^2 / \|X\|_F^2 > \epsilon$ **do**
  $t := t + 1$;
  $\tilde{L} = \left[(X - S_{t-1})(X - S_{t-1})^T\right]^q (X - S_{t-1})$;
  $Y_1 = \tilde{L} A_1$, $A_2 = Y_1$;
  $Y_2 = \tilde{L}^T Y_1 = Q_2 R_2$, $Y_1 = \tilde{L} Y_2 = Q_1 R_1$;
  **If** $\operatorname{rank}\left(A_2^T Y_1\right) < r$ **then** $r := \operatorname{rank}\left(A_2^T Y_1\right)$, go to the first step; **end**;
  $L_t = Q_1 \left[R_1 \left(A_2^T Y_1\right)^{-1} R_2^T\right]^{1/(2q+1)} Q_2^T$;
  $S_t = \mathcal{P}_\Omega (X - L_t)$, $\Omega$ is the nonzero subset of the first $k$ largest entries of $|X - L_t|$;
**end while**

### 3.3. GoDec for matrix completion

We consider the problem of exactly completing a low-rank matrix $X$ with $\operatorname{rank}(X) \leq r$ from a subset of its entries $Y = \mathcal{P}_\Omega(X)$, wherein $\Omega$ is the sampling index set. Different from the two conventional methods, nuclear norm minimization (Candès & Tao, 2009) and low-rank subspace optimization on Grassmann manifold (Keshavan & Oh, 2009), we formulate the matrix completion problem as a rank constrained optimization:

$$\begin{aligned} \min_{X,Z} \quad & \|Y - X - Z\|_F^2 \\ s.t. \quad & \operatorname{rank}(X) \leq r, \\ & \operatorname{supp}(Z) = \overline{\Omega}, \end{aligned} \quad (10)$$

where $Z$ is an estimate of $-\mathcal{P}_{\overline{\Omega}}(X)$. Therefore, Godec algorithms can be extended to solve (10) after the following two slight modifications.

- Replacing $X$, $L$ and $S$ in Algorithm 1 with $Y$, $X$ and $Z$, respectively.
- Replacing the entry set $\Omega$ used in the last step of Algorithm 1 with $\overline{\Omega}$, wherein $\Omega$ is the sampling index set

in matrix completion.

The same as GoDec, its extension (10) for solving the matrix completion problem converges to a local optimum. Compared with the nuclear norm minimization methods, (10) is more efficient because it does not require time consuming SVD for $X$. Compared with the subspace optimization methods, GoDec avoids the unstableness and the local barriers of the optimization on Grassmann manifold. Moreover, GoDec is parameter free (both the rank range $r$ and the tolerance $\epsilon$ are predefined parameters) and thus it is easier to use compared with existing methods.

## 4. Convergence of GoDec

In this section, we analyze the convergence properties of GoDec. In particular, we first prove that the objective value $\|X-L-S\|_F^2$ (decomposition error) converges to a local minimum. Then we demonstrate the asymptotic properties of GoDec and prove that the solutions $L$ and $S$ respectively converge to local optimums with linear rate less than 1, by using the framework presented in (Lewis & Malick, 2008). The influence of $L$, $S$ and $G$ to the asymptotic/convergence speeds is analyzed. The speeds will be slowed by augmenting the magnitude of noise part $\|G\|_F^2$. However, the convergence will not be harmed unless $\|G\|_F^2 \gg \|L\|_F^2$ or $\|G\|_F^2 \gg \|S\|_F^2$.

We have the following theorem about the convergence of the objective value $\|X-L-S\|_F^2$ in (7).

**Theorem 1.** (**Convergence of objective value**). *The alternative optimization (8) produces a sequence of $\|X-L-S\|_F^2$ that converges to a local minimum.*

*Proof.* Let the objective value $\|X-L-S\|_F^2$ after solving the two subproblems in (8) be $E_t^1$ and $E_t^2$, respectively, in the $t^{th}$ iteration. On the one hand, we have

$$E_t^1 = \|X-L_t-S_{t-1}\|_F^2, E_t^2 = \|X-L_t-S_t\|_F^2. \quad (11)$$

The global optimality of $S_t$ yields $E_t^1 \geq E_t^2$. On the other hand,

$$E_t^2 = \|X-L_t-S_t\|_F^2, E_{t+1}^1 = \|X-L_{t+1}-S_t\|_F^2. \quad (12)$$

The global optimality of $L_{t+1}$ yields $E_t^2 \geq E_{t+1}^1$. Therefore, the objective values (decomposition errors) $\|X-L-S\|_F^2$ keep decreasing throughout GoDec (8):

$$E_1^1 \geq E_1^2 \geq E_2^1 \geq \cdots \geq E_t^1 \geq E_t^2 \geq E_{t+1}^1 \geq \cdots \quad (13)$$

Since the objective of (7) is monotonically decreasing and the constraints are satisfied all the time, (8) produces a sequence of objective values that converge to a local minimum. This completes the proof. □

The asymptotic property and the linear convergence of $L$ and $S$ in GoDec are demonstrated based on the framework proposed in (Lewis & Malick, 2008). We firstly consider $L$. From a different prospective, GoDec algorithm shown in (9) is equivalent to iteratively projecting $L$ onto one manifold $\mathcal{M}$ and then onto another manifold $\mathcal{N}$. This kind of optimization method is the so called “alternating projections on manifolds”. To see this, in (9), by substituting $S_t$ into the next updating of $L_{t+1}$, we have:

$$L_{t+1} = \mathcal{P}_{\mathcal{M}}\left(X-\mathcal{P}_{\Omega}\left(X-L_t\right)\right) = \mathcal{P}_{\mathcal{M}}\mathcal{P}_{\mathcal{N}}\left(L_t\right), \quad (14)$$

Both $\mathcal{M}$ and $\mathcal{N}$ are two $C^k$-manifolds around a point $\overline{L} \in \mathcal{M}\cap\mathcal{N}$:

$$\begin{cases} \mathcal{M} = \left\{H \in \mathbb{R}^{m\times n} : \operatorname{rank}(H) = r\right\}, \\ \mathcal{N} = \left\{X - \mathcal{P}_{\Omega}(X-H) : H \in \mathbb{R}^{m\times n}\right\}. \end{cases} \quad (15)$$

According to the above definitions, any point $L \in \mathcal{M}\cap\mathcal{N}$ satisfies:

$$L = \mathcal{P}_{\mathcal{M}\cap\mathcal{N}}(L) \Rightarrow \quad (16)$$

$$L = X - \mathcal{P}_{\Omega}(X-L), \operatorname{rank}(L) = r. \quad (17)$$

Thus any point $L \in \mathcal{M}\cap\mathcal{N}$ is a local solution of $L$ in (7).

We define the angle between two manifolds $\mathcal{M}$ and $\mathcal{N}$ at point $L$ as the angle between the corresponding tangent spaces $T_{\mathcal{M}}(L)$ and $T_{\mathcal{N}}(L)$. The angle is between 0 and $\pi/2$ with cosine:

$$c(\mathcal{M},\mathcal{N},L) = c\left(T_{\mathcal{M}}(L), T_{\mathcal{N}}(L)\right). \quad (18)$$

In addition, if $\mathbb{S}$ is the unit sphere in $\mathbb{R}^{m\times n}$, the angle between two subspaces $M$ and $N$ in $\mathbb{R}^{m\times n}$ is defined as the angle between 0 and $\pi/2$ with cosine:

$$c(M,N) = \max\left\{\langle x,y\rangle : x \in \mathbb{S}\cap M\cap(M\cap N)^{\perp}, \; y \in \mathbb{S}\cap N\cap(M\cap N)^{\perp}\right\}.$$

We give the following proposition about the angle between two subspaces $M$ and $N$:

**Proposition 1.** *Following the above definition of the angle between two subspaces $M$ and $N$, we have*

$$c(M,N) = \max\left\{\langle x,y\rangle : x \in \mathbb{S}\cap M\cap N^{\perp}, \; y \in \mathbb{S}\cap N\cap M^{\perp}\right\}.$$

The angle between $\mathcal{M}$ and $\mathcal{N}$ is used in the asymptotical property and the linear convergence rate of “alternating projections on manifolds” algorithms.

**Theorem 2.** (**Asymptotic property** (Lewis & Malick, 2008)). *Let $\mathcal{M}$ and $\mathcal{N}$ be two transverse $C^2$-manifolds around a point $\overline{L} \in \mathcal{M}\cap\mathcal{N}$. Then*

$$\limsup_{L\to\overline{L}, L\notin\mathcal{M}\cap\mathcal{N}} \frac{\left\|\mathcal{P}_{\mathcal{M}}\mathcal{P}_{\mathcal{N}}(L) - \mathcal{P}_{\mathcal{M}\cap\mathcal{N}}(L)\right\|}{\left\|L - \mathcal{P}_{\mathcal{M}\cap\mathcal{N}}(L)\right\|} \leq c\left(\mathcal{M},\mathcal{N},\overline{L}\right).$$

*A refinement of the above argument is*

$$\limsup_{L\to\overline{L}, L\notin\mathcal{M}\cap\mathcal{N}} \frac{\left\|\left(\mathcal{P}_{\mathcal{M}}\mathcal{P}_{\mathcal{N}}\right)^n(L)-\mathcal{P}_{\mathcal{M}\cap\mathcal{N}}(L)\right\|}{\left\|L-\mathcal{P}_{\mathcal{M}\cap\mathcal{N}}(L)\right\|}\leq c^{2n-1}$$

*for* $n=1,2,...$ *and* $c=c\left(\mathcal{M},\mathcal{N},\overline{L}\right)$.

**Theorem 3.** (**Linear convergence of variables** (Lewis & Malick, 2008)). *In* $\mathbb{R}^{m\times n}$*, let* $\mathcal{M}$ *and* $\mathcal{N}$ *be two transverse manifolds around a point* $\overline{L}\in\mathcal{M}\cap\mathcal{N}$*. If the initial point* $L_0\in\mathbb{R}^{m\times n}$ *is close to* $\overline{L}$*, then the method of alternating projections*

$$L_{t+1}=\mathcal{P}_{\mathcal{M}}\mathcal{P}_{\mathcal{N}}\left(L_t\right),(t=0,1,2,...)$$

*is well-defined, and the distance* $d_{\mathcal{M}\cap\mathcal{N}}(L_t)$ *from the iterate* $L_t$ *to the intersection* $\mathcal{M}\cap\mathcal{N}$ *decreases Q-linearly to zero. More precisely, given any constant* $c$ *strictly larger than the cosine of the angle of the intersection between the manifolds,* $c(\mathcal{M},\mathcal{N},\overline{L})$*, if* $L_0$ *is close to* $\overline{L}$*, then the iterates satisfy*

$$d_{\mathcal{M}\cap\mathcal{N}}(L_{t+1})\leq c\cdot d_{\mathcal{M}\cap\mathcal{N}}(L_t),(t=0,1,2,...)$$

*Furthermore,* $L_t$ *converges linearly to some point* $L^*\in\mathcal{M}\cap\mathcal{N}$*, i.e., for some constant* $\alpha>0$*,*

$$\|L_t-L^*\|\leq\alpha c^t,(t=0,1,2,...).$$

Since GoDec algorithm can be written as the form of alternating projections on two manifolds $\mathcal{M}$ and $\mathcal{N}$ given in (15) and they satisfy the assumptions of Theorem 2 and Theorem 3, $L$ in GoDec converges to a local optimum with linear rate. Similarly, we can prove the linear convergence of $S$.

Since cosine $c(\mathcal{M},\mathcal{N},\overline{L})$ in Theorem 2 and Theorem 3 determines the asymptotic and convergence speeds of the algorithm. We discuss how $L$, $S$ and $G$ influence the asymptotic and convergence speeds via analyzing the relationship between $L$, $S$, $G$ and $c(\mathcal{M},\mathcal{N},\overline{L})$.

**Theorem 4.** (**Asymptotic and convergence speed**). *In GoDec, the asymptotical improvement and the linear convergence of L and S stated in Theorem 2 and Theorem 3 will be slowed by augmenting*

$$\begin{aligned}&\text{For } L: \frac{\|\Delta_L\|_F}{\|L+\Delta_L\|_F}, \Delta_L=(S+G)-\mathcal{P}_{\Omega}(S+G),\\ &\text{For } S: \frac{\|\Delta_S\|_F}{\|S+\Delta_S\|_F}, \Delta_S=(L+G)-\mathcal{P}_{\mathcal{M}}(L+G).\end{aligned}$$

*However, the asymptotical improvement and the linear convergence will not be harmed and is robust to the noise* $G$ *unless when* $\|G\|_F\gg\|S\|_F$ *and* $\|G\|_F\gg\|L\|_F$*, which lead the two terms increasing to* 1.

*Proof.* GoDec approximately decomposes a matrix $X=L+S+G$ into the low-rank part $L$ and the sparse part $S$. According to the above analysis, GoDec is equivalent to alternating projections of $L$ on $\mathcal{M}$ and $\mathcal{N}$, which are given in (15). According to Theorem 2 and Theorem 3, smaller $c(\mathcal{M},\mathcal{N},\overline{L})$ produces faster asymptotic and convergence speeds, while $c(\mathcal{M},\mathcal{N},\overline{L})=1$ is possible to make $L$ and $S$ stopping converging. Below we discuss how $L$, $S$ and $G$ influence $c(\mathcal{M},\mathcal{N},\overline{L})$ and further influence the asymptotic and convergence speeds of GeDec.

According to (18), we have

$$c\left(\mathcal{M},\mathcal{N},\overline{L}\right)=c\left(T_{\mathcal{M}}(\overline{L}),T_{\mathcal{N}}(\overline{L})\right). \tag{19}$$

Substituting the equation given in Proposition 1 into the right-hand side of the above equation yields

$$\begin{aligned}c\left(\mathcal{M},\mathcal{N},\overline{L}\right)=\max\big\{\langle x,y\rangle: x\in\mathbb{S}\cap T_{\mathcal{M}}(\overline{L})\cap N_{\mathcal{N}}(\overline{L}),\\ y\in\mathbb{S}\cap T_{\mathcal{N}}(\overline{L})\cap N_{\mathcal{M}}(\overline{L})\big\}.\end{aligned} \tag{20}$$

The normal spaces of manifolds $\mathcal{M}$ and $\mathcal{N}$ on point $\overline{L}$ is respectively given by

$$\begin{aligned}&N_{\mathcal{M}}(\overline{L})=\left\{y\in\mathbb{R}^{m\times n}: u_i^T y v_j=0, \overline{L}=UDV^T\right\},\\ &N_{\mathcal{N}}(\overline{L})=\left\{X-\mathcal{P}_{\Omega}\left(X-\overline{L}\right)\right\},\end{aligned} \tag{21}$$

where $\overline{L}=UDV^T$ represents the eigenvalue decomposition of $\overline{L}$, $U=[u_1,...,u_r]$ and $V=[v_1,...,v_r]$. Assume $X=\overline{L}+\overline{S}+\overline{G}$, wherein $\overline{G}$ is the noise corresponding to $\overline{L}$, we have

$$\begin{aligned}&\overline{L}=X-\left(\overline{S}+\overline{G}\right),\\ &\hat{L}=X-\mathcal{P}_{\Omega}\left(\overline{S}+\overline{G}\right),\Rightarrow\\ &\hat{L}=\overline{L}+\left[\left(\overline{S}+\overline{G}\right)-\mathcal{P}_{\Omega}\left(\overline{S}+\overline{G}\right)\right]=\overline{L}+\Delta.\end{aligned} \tag{22}$$

Thus the normal space of manifold $\mathcal{N}$ is

$$N_{\mathcal{N}}(\overline{L})=\left\{\overline{L}+\Delta\right\}. \tag{23}$$

Since the tangent space is the complement space of the normal space, by using the normal space of $\mathcal{M}$ in (21) and the normal space of $\mathcal{N}$ given in (23), we can verify

$$N_{\mathcal{N}}(\overline{L})\subseteq T_{\mathcal{M}}(\overline{L}), N_{\mathcal{M}}(\overline{L})\subseteq T_{\mathcal{N}}(\overline{L}). \tag{24}$$

By substituting the above results into (20), we obtain

$$\begin{aligned}c\left(\mathcal{M},\mathcal{N},\overline{L}\right)=\max\big\{\langle x,y\rangle: x\in\mathbb{S}\cap N_{\mathcal{N}}(\overline{L}),\\ y\in\mathbb{S}\cap N_{\mathcal{M}}(\overline{L})\big\}.\end{aligned} \tag{25}$$

Hence we have

$$\begin{aligned}\langle x,y\rangle&=\mathrm{tr}\left(VDU^Ty+\Delta^Ty\right)\\ &=\mathrm{tr}\left(DU^TyV\right)+\mathrm{tr}\left(\Delta^Ty\right)=\mathrm{tr}\left(\Delta^Ty\right).\end{aligned} \tag{26}$$

The last equivalence is due to $u_i^T y v_j = 0$ in (21). Thus

$$c\left(\mathcal{M},\mathcal{N},\overline{L}\right) = \max\left\{\langle x, y\rangle\right\} \leq \max\left\{\langle D_\Delta, D_y\rangle\right\}, \quad (27)$$

where the diagonal entries of $D_\Delta$ and $D_y$ are composed by eigenvalues of $\Delta$ and $y$, respectively. The last inequality is obtained by considering the case when $x$ and $y$ have identical left and right singular vectors. Because $\overline{L} + \Delta, y \in \mathbb{S}$ infers $\|\overline{L} + \Delta\|_F^2 = \|y\|_F^2 = 1$, we have

$$\begin{aligned} c\left(\mathcal{M},\mathcal{N},\overline{L}\right) &\leq \max\left\{\langle D_\Delta, D_y\rangle\right\} \\ &\leq \|D_\Delta\|_F \|D_y\|_F \leq \|D_\Delta\|_F . \end{aligned} \quad (28)$$

Since $c$ in Theorem 3 can be selected as any constant that is strictly larger than $c\left(\mathcal{M},\mathcal{N},\overline{L}\right) \leq \|D_\Delta\|_F$, we can choose $c = c\left(\mathcal{M},\mathcal{N},\overline{L}\right) + \Delta_c \leq \|D_\Delta\|_F$. In Theorem 2, the cosine $c\left(\mathcal{M},\mathcal{N},\overline{L}\right)$ is directly used.

Therefore, the asymptotic and convergence speeds of $L$ will be slowed by augmenting $\|\Delta\|_F$, and vice versa. However, the asymptotical improvement and the linear convergence will not be jeopardized unless $\|\Delta\|_F = 1$. For general $L + \Delta$ that is not normalized onto the sphere $\mathbb{S}$, $\|\Delta\|_F$ should be replaced by $\|\Delta\|_F / \|L + \Delta\|_F$.

For the variable $S$, we can obtain an analogous result via an analysis in a similar style as above. For general $L + \Delta$ without normalization, the asymptotic/convergence speed of $S$ will be slowed by augmenting $\|\Delta\|_F / \|S + \Delta\|_F$, and vice versa, wherein

$$\Delta = (L + G) - \mathcal{P}_{\mathcal{M}}(L + G). \quad (29)$$

The asymptotical improvement and the linear convergence will not be jeopardized unless $\|\Delta\|_F / \|S + \Delta\|_F = 1$.

This completes the proof. □

Theorem 4 reveals the influence of the low-rank part $L$, the sparse part $S$ and the noise part $G$ to the asymptotic/convergence speeds of $L$ and $S$ in GoDec. Both $\Delta_L$ and $\Delta_S$ are the element-wise hard thresholding error of $S + G$ and the singular value hard thresholding error of $L + G$, respectively. Large errors will slow the asymptotic and convergence speeds of GoDec. Since $S - \mathcal{P}_\Omega(S) = 0$ and $L - \mathcal{P}_{\mathcal{M}}(L) = 0$, the noise part $G$ in $\Delta_L$ and $\Delta_S$ can be interpreted as the perturbations to $S$ and $L$ and deviates the two errors from 0. Thus noise $G$ with large magnitude will decelerate the asymptotical improvement and the linear convergence, but it will not ruin the convergence unless $\|G\|_F \gg \|S\|_F$ or $\|G\|_F \gg \|L\|_F$. Therefore, GoDec is robust to the additive noise in $X$ and is able to find the approximated $L + S$ decomposition when noise $G$ is not overwhelming.

# 5. Experiments

This section evaluates both the effectiveness and the efficiency of the BRP based low-rank approximation and GoDec for computer vision applications, low-rank+sparse decomposition and matrix completion. We run all the experiments in MatLab on a server with dual quad-core 3.33 GHz Intel Xeon processors and 32 GB RAM. The relative error $\|X - \hat{X}\|_F^2 / \|X\|_F^2$ is used to evaluate the effectiveness, wherein $X$ is the original matrix and $\hat{X}$ is an estimate/approximation.

## 5.1. RPCA vs. GoDec

Since RPCA and GoDec are related in their motivations, we compare their relative errors and time costs on square matrices with different sizes, different ranks of low-rank components and different cardinality of sparse components. For a matrix $X = L + S + G$, its low-rank component is built as $L = AB$, wherein both $A$ and $B$ are $n \times r$ standard Gaussian matrices. Its sparse part is built as $S = \mathcal{P}_\Omega(D)$, wherein $D$ is a standard Gaussian matrix and $\Omega$ is an entry set of size $k$ drawn uniformly at random. Its noise part is built as $G = 10^{-3} \cdot F$, wherein $F$ is a standard Gaussian matrix. In our experiments, we compare RPCA [1] (`inexact_alm_rpca`) with GoDec (Algorithm 2 with $q = 2$). Since both algorithms adopt the relative error of $X$ as the stopping criterion, we use the same tolerance $\epsilon = 10^{-7}$. Table 1 shows the results and indicates that both algorithms are successful in recovering the correct "low-rank+sparse" decompositions with relative error less than $10^{-6}$. GoDec usually produces less relative error with much less CPU seconds than RPCA. The improvement of accuracy is due to that the model of GoDec in (2) is more general than that of RPCA by considering the noise part. The improvement of speed is due to that BRP based low-rank approximation significantly saves the computation of each iteration round.

## 5.2. Matrix completion

We test the performance of GoDec in matrix completion tasks. Each test low-rank matrix is generated by $X = AB$, wherein both $A$ and $B$ are $n \times r$ standard Gaussian matrices. We randomly sample a few entries from $X$ and recover the whole matrix by using Algorithm 1 (after the two modifications presented in Section 4.1). The experimental results are shown in Table 2. Compared with the published results (Keshavan & Oh, 2009) of the popular matrix completion methods, e.g., OptSpace, SVT, FPCA and ADMIRA, GoDec requires both less computational time and less samples to recover a low-rank matrix.

[1] http://watt.csl.illinois.edu/~perceive/matrix-rank

### 5.3. Background modeling

Background modeling (Cheng et al., 2010) is a challenging task to reveal the correlation between video frames, model background variations and foreground moving objects. A video sequence satisfies the low-rank+sparse structure, because backgrounds of all the frames are related, while the variation and the moving objects are sparse and independent. We apply GoDec (Algorithm 2 with $q = 2$) to four surveillance videos [2], respectively. The matrix $X$ is composed of the first 200 frames of each video. For example, the second video is composed of 200 frames with the resolution $256 \times 320$, we convert each frame as a vector and thus the matrix $X$ is of size $81920 \times 200$. We show the decomposition result of one frame in each video sequence in Figure 1. The background and moving objects are precisely separated (the person in $L$ of the fourth sequence does not move throughout the video) without losing details. The results of the first sequence and the fourth sequence are comparable with those shown in (Candès et al., 2009). However, compared with RPCA (36 minutes for the first sequence and 43 minutes for the fourth sequence) (Candès et al., 2009), GoDec requires around 50 seconds for each of both. Therefore, GoDec makes large-scale applications available.

#### Shadow/Light removal

Shadow and light in training images always pull down the quality of learning in computer vision applications. GoDec can remove the shadow/light noises by assuming that they are sparse and the rest parts of the images are low-rank. We apply GoDec (Algorithm 2 with $q = 2$) to face images of four individuals in the Yale B database [3]. Each individual has 64 images with resolution $192 \times 168$ captured under different illuminations. Thus the matrix $X$ for each individual is of size $32760 \times 64$. We show the GoDec of eight example images (2 per individual) in Figure 2. The real face of each individual are remained in the low rank component, while the shadow/light noises are successfully removed from the real face images and stored in the sparse component. The learning time of GoDec for each individual is less than 30 seconds, which encourages for large-scale applications, while RPCA requies around 685 seconds.

## 6. Conclusion

In this paper, we first proposed a bilateral random projections (BRP) based low-rank approximation with fast speed and nearly optimal error bounds. We then develop "Go Decomposition" (GoDec) to estimate the low-rank part $L$ and the sparse part $S$ of a general matrix $X = L + S + G$, wherein $G$ is noise. GoDec is significantly accelerated by using BRP based approximation. The discussions of asymptotic and convergence speeds indicate that GoDec is robust to noise $G$.

[2] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

[3] http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html

## Acknowledgments

This work was supported in part by Australian Research Council Future Fellowship (Grant No. FT100100971), Australian Research Council (ARC) Discovery Project DP1093762 and Discovery Project DP0988016.

## References

Bredies, K. and Lorenz, D. A. Iterated hard shrinkage for minimization problems with sparsity constraints. *SIAM Journal on Scientific Computing*, 30(2):657–683, 2008.

Cai, J., Candès, E. J., and Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

Candès, E. J. and Tao, T. The power of convex relaxation: Near-optimal matrix completion. `arXiv: 0903.1476`, 2009.

Candès, E. J., Li, X., Ma, Y., and Wright, J. Robust principal component analysis? *Journal of the ACM (submitted)*, 2009.

Chandrasekaran, Venkat, Sanghavi, Sujay, Parrilo, Pablo A, and Willsky, Alan S. Rank-sparsity incoherence for matrix decomposition. `arXiv:0906.2220`, 2009.

Cheng, L., Gong, M., Schuurmans, D., and Caelli, T. Real-time discriminative background subtraction. *to appear in IEEE Trans on Image Processing*, 2010.

Donoho, D. L. Compressed sensing. *IEEE Trans on Information Theory*, 52(4):1289–1306, 2006.

Fazel, M., Candès, E. J., Recht, B., and Parrilo, P. Compressed sensing and robust recovery of low rank matrices. In *42nd Asilomar Conference on Signals, Systems and Computers*, 2008.

Halko, N., Martinsson, P. G., and Tropp, J. A. Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions. `arXiv: 0909.4061`, 2009.

Keshavan, R. and Oh, S. Optspace: A gradient descent algorithm on grassman manifold for matrix completion. *Submitted to IEEE Trans on Signal Processing*, 2009.

Lewis, A. S. and Malick, J. Alternating projections on manifolds. *Mathematics of Operations Research*, 33(1):216–234, 2008.

Roweis, S. Em algorithms for pca and spca. In *NIPS*, pp. 626–632, 1998.

Zhou, T. and Tao, D. Bilateral random projection based low-rank approximation. Technical report, 2010.

Zhou, T., Tao, D., and Wu, X. Manifold elastic net: a unified framework for sparse dimension reduction. *Data Mining and Knowledge Discovery (Springer)*, 22(3):340–371, 2011.

Zhou, Z., Li, X., Wright, J., Candès, E. J., and Ma, Y. Stable principal component pursuit. In *ISIT*, 2010.

*Table 1.* Relative error and time cost of RPCA and GoDec in low-rank+sparse decomposition tasks. The results separated by "/" are RPCA and GoDec, respectively.

| size($X$) (square) | rank($L$) (1) | card($S$) ($10^4$) | rel.error($X$) ($10^{-8}$) | rel.error($L$) ($10^{-8}$) | rel.error($S$) ($10^{-6}$) | time (seconds) |
|---|---|---|---|---|---|---|
| 500 | 25 | 1.25 | 3.70/1.80 | 1.50/1.20 | 2.00/0.95 | 6.07/2.83 |
| 1000 | 50 | 5.00 | 4.98/4.56 | 1.82/1.85 | 5.16/4.90 | 20.96/12.71 |
| 2000 | 100 | 20.0 | 8.80/1.13 | 3.10/1.10 | 1.81/1.24 | 101.74/74.16 |
| 3000 | 250 | 45.0 | 6.29/4.98 | 5.09/5.05 | 33.9/55.3 | 562.09/266.11 |
| 5000 | 400 | 125 | 63.1/24.4 | 30.2/29.3 | 54.2/18.8 | 2495.31/840.39 |
| 10000 | 500 | 600 | 6.18/3.04 | 2.27/2.88 | 58.3/36.6 | 9560.74/3030.15 |

*Table 2.* Relative error and time cost of OptSpace and GoDec in matrix completion tasks. The results separated by "/" are SVT (Cai et al., 2010) (a nuclear norm minimization method), OptSpace (Keshavan & Oh, 2009) (a subspace optimization method on Grassmann manifold) and GoDec, respectively. See (Keshavan & Oh, 2009) for the results of the other methods, e.g., FPCA and ADMIRA.

| size($X$) (square) | rank($X$) (1) | sampling rate (%) | rel.error($X$) ($10^{-5}$) | time (seconds) |
|---|---|---|---|---|
| 1000 | 10 | 0.12/0.12/0.075 | 1.68/1.18/1.77 | 40/28/15.43 |
| | 50 | 0.39/0.39/0.18 | 1.62/0.92/1.11 | 247/212/26.36 |
| | 100 | 0.57/0.57/0.3 | 1.71/1.49/1.24 | 694/723/43.47 |
| 5000 | 10 | 0.024/0.024/0.021 | 1.76/1.51/1.39 | 112/252/300.96 |
| | 50 | 0.1/0.1/0.084 | 1.62/1.16/1.48 | 1312/850/415.96 |
| | 100 | 0.16/0.16/0.12 | 1.73/0.83/1.09 | 5432/3714/551.95 |
| 10000 | 10 | 0.012/0.012/0.04 | 1.75/0.76/0.50 | 221/632/1101.83 |
| | 50 | 0.05/0.05/0.045 | 1.63/1.19/1.17 | 2872/2585/1172.68 |
| | 100 | 0.08/0.08/0.075 | 1.76/1.46/1.84 | 10962/8514/1505.93 |

*Figure 1.* Background modeling results of four 200-frame surveillance video sequences in $X = L + S$ mode. Top left: lobby in an office building (resolution $128 \times 160$, learning time 39.75 seconds). Top right: shopping center (resolution $256 \times 320$, learning time 203.72 seconds). Bottom left: Restaurant (resolution $120 \times 160$, learning time 36.84 seconds). Bottom right: Hall of a business building (resolution $144 \times 176$, learning time 47.38 seconds).

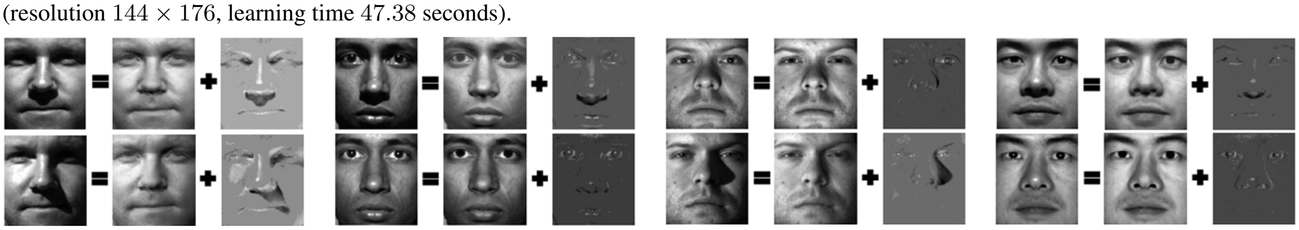

*Figure 2.* Shadow/light removal of face images from four individuals in Yale B database in $X = L + S$ mode. Each individual has 64 images with resolution $192 \times 168$ and needs 24 seconds learning time.