

RESEARCH

Open Access



Low storage space for compressive sensing: semi-tensor product approach

Jinming Wang*, Shiping Ye, Yue Ruan and Chaoxiang Chen

Abstract

Random measurement matrices play a critical role in successful recovery with the compressive sensing (CS) framework. However, due to its randomly generated elements, these matrices require massive amounts of storage space to implement a random matrix in CS applications. To effectively reduce the storage space of the random measurement matrix for CS, we propose a random sampling approach for the CS framework based on the semi-tensor product (STP). The proposed approach generates a random measurement matrix, where the dimensions of the random measurement matrix are reduced to a quarter (or 1/16, 1/64, and even 1/256) of the number of dimensions, which are used for conventional CS. We then estimate the values of the sparse vector with a modified iteratively re-weighted least-squares (IRLS) algorithm. The results of numerical simulations showed that the proposed approach can reduce the storage space of a random matrix to at least a quarter while maintaining quality of reconstruction. All results confirmed that the proposed approach significantly influences the physical implementation of the CS in images, especially on embedded system and field programmable gate array (FPGA), where storage is limited.

Keywords: Compressed sensing, Compressive sensing, Random measurement matrix, Storage space, Semi-tensor product

1 Introduction

Compressive sensing (CS) [1] theory provides a new way to sample and compress data. The basic idea of CS is that a higher-dimensional signal is projected onto a measurement matrix, by which a low-dimensional sensed sequence is obtained. Meanwhile, [2–4] prove that if the sensed sequence consists of a small number of non-zero elements, then it can recover the original signal from the sensed sequence. CS applications confirm that random measurement matrices are suitable for compressed sensing. However, these applications require considerable storage space to realize random matrices [3]. As a result, much work has been done to reduce the storage space and improve performance.

In [5, 6], an intermediate structure for the measurement matrix is proposed based on random sampling, called block compressed sensing (block CS). Using the block CS, the data sampling is conducted in a block-by-block manner through the same measurement matrix, which overcomes the difficulties encountered in traditional CS

technology, for which the random measurement ensembles are numerically unwieldy.

In [7, 8], Thong, et al. introduced a fast and efficient way to construct a measurement matrix, called the structurally random matrix (SRM), which attempted to improve the structure of an initial random measurement matrix using optimization techniques. SRM is related to large-scale, real-time CS applications for low requirements in storage space.

To reduce the storage space for CS, many deterministic measurement matrices have been designed [9–14]. They satisfy the restricted isometry property (RIP) and recovered the sparse signal successfully.

In [15, 16], low-dimensional orthogonal basis vectors or matrices were used to construct high-dimensional matrices according to the Kronecker product. The proposed algorithm effectively reduces the storage space of a measurement matrix.

Low-rank matrices and rank-one matrices are attractive because they need less storage space than general measurement matrices [17–20]. Indeed, if the measurement matrix is sparse, it takes less storage space and

* Correspondence: wjm7878@163.com
Collage of Information Science and Technology, Zhejiang Shuren University,
Hangzhou 310015, China

incurs less computational cost. Low-rank or rank-one matrices have been designed to sample and reconstruct the original signal, obtaining high-quality reconstructions.

All of the above findings are directed at how to reduce the storage space of a measurement matrix for CS. However, using the block CS, several block artifacts occur in the reconstructed images, owing to the block-by-block manner and the neglect of global sparsity. However, the SRM method is more complicated and difficult to achieve. Deterministic measurement matrices require little storage space and incur less computational cost, but the accuracy of the reconstruction is not as high as a random measurement matrix. To reconstruct the original signals, the Kronecker algorithm must generate an $M \times N$ dimensional measurement matrix, and this requires large-scale memory space.

For the same purpose, we propose a random sampling scheme for CS. The aim is to propose an algorithm that can maintain the same reconstruction performance as conventional compressive sensing, but requires less required storage for the measurement matrix and less memory for reconstructing.

The proposed algorithm is based on the semi-tensor product (STP) [21, 22], a novel matrix product that works by extending the conventional matrix product in cases of unequal dimensions. Our algorithm generates a random matrix, with dimensions that are smaller than M and N , where M is the length of the sampling vector and N is the length of signal that we want to reconstruct. Then, we use the iteratively re-weighted least squares (IRLS) algorithm to estimate the value of the sparse coefficients. Experiments were carried out using the sparse column signals and images, demonstrating that it outperforms other algorithms in terms of storage space and a suitable peak signal to noise ratio (PSNR) performance. The experimental results show that if we reduce the dimensions of the measurement matrix appropriately, there is almost no decline in the PSNR of the reconstruction, yet the storage space required by the measurement matrix can be reduced to a quarter (or even 16th) of the size.

The remainder of this paper is organized as follows: In Section 2, the preliminaries of the STP and the conventional CS algorithm are introduced. In Section 3, we describe the proposed STP approach to the CS algorithm (STP-CS). In Section 4, we present the experimental results and a discussion. Finally, Section 5 concludes the paper and contains a discussion of our plans for future research.

2 Related works

In this section, the concepts of the conventional CS algorithm and some necessary preliminaries to the STP

are briefly introduced. The STP of matrices was introduced by Cheng [21, 22].

2.1 Semi-tensor product

In [21, 22], the STP is presented as an extension of the conventional matrix product. For a conventional matrix product, if $\text{Col}(A) \neq \text{Row}(B)$, then matrices A and B are multiplicative. The STP of matrices, on the other hand, extends the conventional matrix product in cases of unequal dimensions. In [22], the STP is defined as follows:

Definition 1: Let $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$. If either n is a factor of p —i.e., if $nt = p$ (denoted as $A \prec_t B$)—or if p is a factor of n —i.e., $n = pt$ (denoted as $A \succ_t B$)—then the (left) STP of A and B can be denoted by $C = \{C^{ij}\} = A \times B$, as follows: C consists of $m \times q$ blocks, and each block is defined as

$$C^{ij} = A^i \times B_j, i = 1, \dots, m, j = 1, \dots, q,$$

where A_i is the i th row of A , and B_j is the j th column of B .

If we assume that $A \succ_t B$ (or $A \prec_t B$), then A and B are split into blockwise forms as follows:

$$A = \begin{bmatrix} A^{11} & \dots & A^{1s} \\ \vdots & & \vdots \\ A^{r1} & \dots & A^{rs} \end{bmatrix}, B = \begin{bmatrix} B^{11} & \dots & B^{1t} \\ \vdots & & \vdots \\ B^{s1} & \dots & B^{st} \end{bmatrix}.$$

If $A^{ik} \succ_t B^{kj}, \forall i, j, k$ (correspondingly, $A^{ik} \prec_t B^{kj}, \forall i, j, k$), then

$$A \times B = \begin{bmatrix} C^{11} & \dots & C^{1t} \\ \vdots & & \vdots \\ C^{r1} & \dots & C^{rt} \end{bmatrix}, \tag{1}$$

where $C^{ij} = \sum_{k=1}^s A^{ik} \times B^{kj}$, and the dimensions of $A \times B$ can be determined by deleting the largest common factor of the dimensions of the two factor matrices. If $A \succ_t B$, the dimensions of $A \times B$ are $m \times tq$. Likewise, if $A \prec_t B$, the dimensions of $A \times B$ are $mt \times q$. In order to allow the reader to clearly understand the STP, we use the following numerical example to describe it:

$$\text{Let } X = \begin{bmatrix} 1 & 2 & -1 & 2 \\ 0 & 1 & 2 & 3 \\ 3 & 3 & 1 & 1 \end{bmatrix}, Y = \begin{bmatrix} 1 & 2 \\ -1 & 3 \end{bmatrix}. \text{ Then}$$

$$\begin{aligned} X \times Y &= \begin{bmatrix} (1 \ 2) - (-1 \ 2) & 2(1 \ 2) + 3(-1 \ 2) \\ (0 \ 1) - (2 \ 3) & 2(0 \ 1) + 3(2 \ 3) \\ (3 \ 3) - (1 \ 1) & 2(3 \ 3) + 3(1 \ 1) \end{bmatrix} \\ &= \begin{bmatrix} 2 & 0 & -1 & 10 \\ -2 & -2 & 6 & 11 \\ 2 & 2 & 9 & 9 \end{bmatrix}. \end{aligned}$$

In comparing the product of the conventional matrix with the STP of the matrix, it is easy to see that there are significant differences between them.

If $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times q}$, and $n = p$, then

$$A \times B = AB.$$

Consequently, when the conventional matrix product is extended to the STP, almost all of its properties are nevertheless maintained. Two properties of the STP are introduced as follows.

Proposition 1 : Suppose $A = (\alpha_{ij}) \in \mathbb{R}^{p \times qr}$,
 $B = (b_{ij}) \in \mathbb{R}^{r \times s}$, and $C = (c_{ij}) \in \mathbb{R}^{qs \times l}$, then

$$A_{p \times qr} \times B_{r \times s} \times C_{qs \times l} = (A \times B)_{p \times qs} \times C_{qs \times l} = (A \times B \times C)_{pl \times l}. \quad (2)$$

Referring to Proposition 1, the dimensions of the STP of two matrices can be determined by removing the largest common factor of the dimensions of the two-factor matrices. As shown in (2), in the first product, r is deleted, and then the qs is deleted.

Proposition 2 : Suppose $A = (a_{ij}) \in \mathbb{R}^{m \times tp}$,
 $B = (b_{ij}) \in \mathbb{R}^{p \times p}$, and $C = (c_{ij}) \in \mathbb{R}^{p \times p}$.

The STP satisfies the associative law :

$$(A \times B) \times C = A \times (B \times C). \quad (3)$$

In recent years, the STP has been exploited by a wide range of applications: in nonlinear system control for structural analysis and control of Boolean networks [23, 24], in biological systems as a solution to Morgan's Problem [25], in a linear system for nonlinear feedback-shift registers [26, 27], etc. However, we have not yet seen publicly reported applications for the STP in the field of CS, to the best of our knowledge.

2.2 Conventional CS algorithm

The conventional CS algorithm can be described simply as follows. Assume $x \in \mathbb{R}^{N \times 1}$, which can be represented sparsely in a known transform domain (e.g., Fourier, or wavelet). Although x can be sparse in the current domain (e.g., time or pixel), we always assume that x is sparse in a known transform domain. The so-called sparsifying transform can be denoted as $\Psi \in \mathbb{R}^{N \times N}$. Considering the above notations, the signal x can be denoted as

$$x_{N \times 1} = \Psi_{N \times N} \theta_{N \times 1}, \quad (4)$$

where $\Psi \Psi^T = \Psi^T \Psi = I$, I is the unit for the matrix, and $\theta \in \mathbb{R}^{N \times 1}$ is a column vector of sparse coefficients, having merely $k \ll N$ non-zero elements.

Here, the vector θ is called exact-sparse because it has k non-zeros and the rest of the elements are equal to zero. However, there might be cases where the coefficient vector θ includes only a few large components with many

small coefficients. In this case, x is called a compressible signal, and sparse approximation methods can be applied.

Then, the data acquisition process of the conventional CS algorithm is defined as follows:

$$y_{M \times 1} = \Phi_{M \times N} \Psi_{N \times N} \theta_{N \times 1}, \quad (5)$$

where $\Phi_{M \times N} \in \mathbb{R}^{M \times N}$ ($M < N$) is defined as the measurement matrix and $y_{M \times 1} \in \mathbb{R}^{M \times 1}$ is treated as measurements.

As shown in (5), the higher dimensional signal x is projected onto the matrix $\Phi_{M \times N}$, and a low-dimensional sensed sequence $y_{M \times 1}$ is obtained. Assume a signal is sampled using the above scheme and then the measurements y are transmitted. The crucial task (for the receiver) is to reconstruct the original samples x with knowledge about the measurements $y_{M \times 1}$ and the measurement matrix $\Phi_{M \times N}$. The recovery problem is ill conditioned since $M < N$. However, several methods have been proposed to tackle this problem, such as the iteratively re-weighted least squares (IRLS) algorithm.

When the original signal is sampled and reconstructed, the role played by the measurement matrix is vital in order to faithfully reconstruct with precision and complexity [4]. However, this requires a lot of storage space in order to realize the measurement matrices in CS applications. This is especially true of random measurement matrices, because they are computationally expensive and require considerable memory [3]. Therefore, reducing the storage space of the measurement matrix is essential to practical CS applications, especially in terms of the feasibility of embedded hardware implementations.

3 Proposed algorithm

To effectively reduce the storage space of a random measurement matrix, we propose an STP approach for the CS algorithm (STP-CS), which can reduce storage space to at least a quarter of the size while maintaining the quality of reconstructed signals or images.

Then, the STP-CS algorithm is defined as follows:

$$y_{M \times 1} = \Phi_{M/t \times N/t} \Psi_{N \times N} \theta_{N \times 1}, \quad (6)$$

where the dimensions of the measurement matrix Φ are $(M/t \times N/t)$ with $t < M$ (M , N , t , M/t , and N/t are positive integers). For convenience, we denote by Φ a matrix whose dimensions had been reduced to $\Phi(t)$. Here, Ψ and θ are defined as the conventional CS.

We assume $x_{N \times 1}$ is a k -sparse signal of length N . Then, x can be represented by $\Phi(t)$ as follows:

$$y_{M \times 1} = \Phi(t) \times x_{N \times 1}, \quad (7)$$

Equation (7) can be expanded by multiplication, such that the following holds:

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{pmatrix} = \begin{pmatrix} \varphi_{1,1} & \cdots & \varphi_{1,j} & \cdots & \varphi_{1,N/t} \\ \vdots & & \vdots & & \vdots \\ \varphi_{i,1} & \cdots & \varphi_{i,j} & \cdots & \varphi_{i,N/t} \\ \vdots & & \vdots & & \vdots \\ \varphi_{M/t,1} & \cdots & \varphi_{M/t,j} & \cdots & \varphi_{M/t,N/t} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}, \quad (8)$$

where $\varphi_{i,j} \in \Phi(t)$ ($i = 1, 2, \dots, M/t, j = 1, 2, \dots, N/t$).

According to the definitions of STP shown above, Eq. (8) can be expressed as:

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{pmatrix} = \begin{pmatrix} \eta^{1,1} + \cdots + \eta^{1,j} + \cdots + \eta^{1,N/t} \\ \eta^{i,1} + \cdots + \eta^{i,j} + \cdots + \eta^{i,N/t} \\ \vdots \\ \eta^{M/t,1} + \cdots + \eta^{M/t,j} + \cdots + \eta^{M/t,N/t} \end{pmatrix}, \quad (9)$$

where $\eta^{i,j} = \varphi_{i,j}(x_{(j-1)t+1} \cdots x_{jt})^T$, $i = 1, 2, \dots, M/t$, $j = 1, 2, \dots, N/t$. $\eta^{i,j}$ is a column vector of length t , and $\sum_{j=1}^{N/t} \eta^{i,j}$ is also a column vector of length t .

If we assume $t = 2, N = 10, M = 6$, $x_{10 \times 1}$ is the sparse signal, $y_{6 \times 1}$ is the measurement, $\Phi_{3 \times 5}$ is the random measurement matrix, the acquisition process of STP-CS can be described as follows:

$$\begin{aligned} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_5 \\ y_6 \end{pmatrix} &= \begin{pmatrix} \varphi_{1,1} & \varphi_{1,2} & \varphi_{1,3} & \varphi_{1,4} & \varphi_{1,5} \\ \varphi_{2,1} & \varphi_{2,2} & \varphi_{2,3} & \varphi_{2,4} & \varphi_{2,5} \\ \varphi_{3,1} & \varphi_{3,2} & \varphi_{3,3} & \varphi_{3,4} & \varphi_{3,5} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_9 \\ x_{10} \end{pmatrix} \\ &= \begin{pmatrix} \varphi_{1,1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \varphi_{1,2} \begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \varphi_{1,3} \begin{pmatrix} x_5 \\ x_6 \end{pmatrix} + \varphi_{1,4} \begin{pmatrix} x_7 \\ x_8 \end{pmatrix} + \varphi_{1,5} \begin{pmatrix} x_9 \\ x_{10} \end{pmatrix} \\ \varphi_{2,1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \varphi_{2,2} \begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \varphi_{2,3} \begin{pmatrix} x_5 \\ x_6 \end{pmatrix} + \varphi_{2,4} \begin{pmatrix} x_7 \\ x_8 \end{pmatrix} + \varphi_{2,5} \begin{pmatrix} x_9 \\ x_{10} \end{pmatrix} \\ \varphi_{3,1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \varphi_{3,2} \begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \varphi_{3,3} \begin{pmatrix} x_5 \\ x_6 \end{pmatrix} + \varphi_{3,4} \begin{pmatrix} x_7 \\ x_8 \end{pmatrix} + \varphi_{3,5} \begin{pmatrix} x_9 \\ x_{10} \end{pmatrix} \end{pmatrix} \\ &= \begin{pmatrix} \varphi_{1,1}x_1 + \varphi_{1,2}x_3 + \varphi_{1,3}x_5 + \varphi_{1,4}x_7 + \varphi_{1,5}x_9 \\ \varphi_{1,1}x_2 + \varphi_{1,2}x_4 + \varphi_{1,3}x_6 + \varphi_{1,4}x_8 + \varphi_{1,5}x_{10} \\ \varphi_{2,1}x_1 + \varphi_{2,2}x_3 + \varphi_{2,3}x_5 + \varphi_{2,4}x_7 + \varphi_{2,5}x_9 \\ \varphi_{2,1}x_2 + \varphi_{2,2}x_4 + \varphi_{2,3}x_6 + \varphi_{2,4}x_8 + \varphi_{2,5}x_{10} \\ \varphi_{3,1}x_1 + \varphi_{3,2}x_3 + \varphi_{3,3}x_5 + \varphi_{3,4}x_7 + \varphi_{3,5}x_9 \\ \varphi_{3,1}x_2 + \varphi_{3,2}x_4 + \varphi_{3,3}x_6 + \varphi_{3,4}x_8 + \varphi_{3,5}x_{10} \end{pmatrix}. \end{aligned} \quad (10)$$

According to the example, we can see that the sparse signal $x_{10 \times 1}$ can project onto $\Phi_{3 \times 5}$, by which a low-dimensional sensed sequence $y_{6 \times 1}$ is obtained. The measurements are simply linear combinations of the elements of $x_{10 \times 1}$.

If we assume $t = 1$, measurement $y'_{6 \times 1}$ is obtained by $\Phi_{6 \times 10}$, which is the same as in the conventional CS. Thus, Eq. (7) can be expressed as $y_{M \times 1} = \Phi_{M \times N} \cdot x_{N \times 1}$.

It should be pointed out that, for the same sparse signal, the measurements obtained from different measurement matrices are different. That is $y_{6 \times 1} \neq y'_{6 \times 1}$.

Let the measurement matrix $\Phi(t)$ be of full-row rank—i.e., $\text{Rank}(\Phi(t)) = M/t$ ($t > 1$)—and be a random matrix, such as Gaussian $N(0, 1/(M/t))$ with i.i.d. entries. Given a sparsifying basis Ψ as shown in Eq. (6), we can get Θ^t as follows:

$$\Theta^t = \Phi_{M/t \times N/t} \times \Psi_{N \times N}, \quad (11)$$

where Θ^t is an $N \times N$ matrix.

The mutual coherence [3] of Θ^t can be obtained as follows:

$$\mu(\Theta^t) \triangleq \max_{\substack{i \neq j \\ 1 \leq i, j \leq N}} \frac{|\theta_i^T \theta_j|}{\|\theta_i\|_2 \|\theta_j\|_2}, \quad (12)$$

where θ_i is the column vector of Θ^t .

If $\mu(\Theta^t) \in [1, \sqrt{N}]$, $\Phi(t)$ is incoherent with basis Ψ with a high probability, such that $\Theta_{N \times N}^t$ satisfies the RIP with high probability [1–3]. This guarantees that a k -sparse or compressible signal can be fully represented by M measurements with the dimension-reduced measurement matrix $\Phi(t)$. The approach does not change the linear nature of the CS acquisition process, except for involving a smaller measurement matrix to obtain measurements. It is clear that the STP approach fits well with the conventional CS for $t > 1$.

With Propositions 1 and 2, we can verify that the STP approach in Eq. (6) is compatible with the conventional CS in Eq. (5):

$$\begin{aligned} \Phi_{M/t \times N/t} \times \Psi_{N \times N} \times \Theta_{N \times 1} &= \Phi_{M/t \times N/t} \times (\Psi \times \Theta)_{N \times 1} \\ &= (\Phi \times \Psi \times \Theta)_{M \times 1} = y_{M \times 1}, \end{aligned}$$

or

$$\begin{aligned} (\Phi_{M/t \times N/t} \times \Psi_{N \times N}) \times \Theta_{N \times 1} &= (\Phi \times \Psi)_{M \times N} \times \Theta_{N \times 1} \\ &= (\Phi \times \Psi \times \Theta)_{M \times 1} = y_{M \times 1}. \end{aligned}$$

When we assume $t = 2$, then the dimensions of $\Phi(t)$ are $(M/2) \times (N/2)$, and when we assume $t = 4$, the dimensions are $(M/4) \times (N/4)$, etc.. Thus, the storage space of $\Phi(t)$ is reduced quadratically. For example, to process a 1024×1024 image, when the sampling rate is 50% and the data is with double precision floating-point, there are 512 K measurements. A Gaussian random

matrix requires 4096 K bytes when $t = 1$. While $t = 2$, the needed storage is 1024 K bytes, and when $t = 4$, this is reduced to 256 K bytes.

Thus far, the STP approach could be an effective way to reduce the storage space of the measurement matrix. There remains a key question, however, regarding how the original signal can be reconstructed based on the STP approach acquisition process.

To reconstruct the original signal, we adopted the IRLS to reconstruct the original signal [28–35]. In [29], it was shown empirically that using ℓ^q -minimization with $0 < q < 1$ can do with fewer measurements than ℓ^1 -minimization. In case of a noisy k -sparse vector, using ℓ^q -minimization with $0 < q < 1$ is more stable than ℓ^1 -minimization [31, 32]. In [33–35], an approximate ℓ^0 -norm minimization algorithm was proposed. The approximate ℓ^0 -norm minimization shows attractive convergence properties, which is capable of very fast signal recovery, thereby reducing retrieval latency when handling high-dimensional signals.

According to the algorithm of IRLS, the solution to the original sparse signal $x_{N \times 1}$ can be obtained as follows:

$$x_{N \times 1}^{(n+1)} = D_n \times \Phi^T(t) \times (\Phi(t) \times D_n \times \Phi^T(t))^{-1} \times y_{M \times 1}, \quad (13)$$

where $x_{N \times 1}^{(n+1)}$ denotes the $(n + 1)$ th iteration, D_n is the $N \times N$ diagonal matrix, and the i th diagonal element is $1/w_i^{(n)}$ ($i = 1, 2, \dots, N$).

For ℓ^q -norm ($0 < q < 1$) minimization, weight $w_i^{(n)}$ is defined as

$$w_i^{(n)} = \left((x_i^{(n)})^2 + \varepsilon_n^{1+q} \right)^{(2-q)/q}, \quad (14)$$

and for approximate ℓ^0 -norm minimization, the weight is defined as

$$w_i^{(n)} = \exp\left(-\left(x_i^{(n)}\right)^2 / 2\varepsilon_n^2\right), \quad (15)$$

where ε_n is a positive real number. During the iterations, it decreases as $\varepsilon_{n+1} = \rho \varepsilon_n$ ($0 < \rho < 1$).

When we derive a vector of measurements $y_{M \times 1}$ by a random measurement matrix $\Phi(t)$, we initialize the algorithm by taking $w^0 = (1, \dots, 1)_{1 \times N}$, $x^0 = (1, \dots, 1)_{1 \times N}$, and $\varepsilon_0 = 1$. The k -sparse signal x is then reconstructed by iterations.

Therefore, in Section 4, we experimentally reconstruct the original sparse signal with ℓ^q -norm ($0 < q < 1$) minimization and approximate ℓ^0 -norm minimization, respectively.

4 Experiments results and discussion

In this section, we verify the performance of the proposed STP-CS. Our intent is to determine trade-offs between recovery performance and the reduction ratio of the measurement matrix. We also compared the performance of STP-CS with that of CS with ℓ^q -minimization ($0 < q < 1$) and approximate ℓ^0 -minimization. We begin the numerical experiments with some $N \times 1$ column-sparse vectors and some $N \times N$ gray-scale images. In our experiments the dimensions of the measurement matrix $\Phi(t)$ are $(M/t) \times (N/t)$, where t could be 1, 2, 4, or even larger, and the matrix $\Phi(t)$ is Gaussian $N(0, 1/(M/t))$ i.i.d. entries, which approximately satisfy the RIP with high probability [2, 32]. In addition, as we have shown in Section 3, when $t = 1$, there is no reduction to the dimensions of the matrix $\Phi(t)$. As such, it can be treated as conventional CS, whereas when $t = 2, 4$ or higher, the dimensions of the matrix $\Phi(t)$ are reduced. Therefore, we performed the comparison with different t . Our experiments and comparisons were implemented in Matlab R2010b on an Intel i7–4600 laptop with 8 GB of memory, running Windows 8.

4.1 Comparison with one-dimensional sparse signal vectors

To compare the performance with the matrices $\Phi(t)$, we measured the rate of convergence and the probability of an exact reconstruction for different sparsity values k and for different numbers of measurements.

First, we considered one-dimensional sparse vectors $x_{N \times 1}$, where N is the length of the sparse vector.

When $N = 256$, $M = 128$, and $k = 40$, a 40-sparse vector is generated with a random positioning of the non-zeros. Here, according to [1], if a sparse vector is used to ensure the uniqueness of a sparse solution, the number of non-zero elements can reach a limit of $M/2$. Therefore, we give a maximum number of k ($k = 1, 2, \dots, M/2$).

Given $N = 256$ and the measurement numbers M , we generated some Gaussian random measurement matrices $\Phi(t)$, with $t = 1, 2$, and 4. The matrices $\Phi(t)$ are provided in Table 1.

Table 1 Different dimensional measurement matrices $\Phi(t)$ for STP-CS ($N = 256$)

M	$t = 1$	$t = 2$	$t = 4$
32	$\Phi_{32 \times 256}$	$\Phi_{16 \times 128}$	$\Phi_{8 \times 64}$
64	$\Phi_{64 \times 256}$	$\Phi_{32 \times 128}$	$\Phi_{16 \times 64}$
128	$\Phi_{128 \times 256}$	$\Phi_{64 \times 128}$	$\Phi_{32 \times 64}$

As shown in Table 1, when $t = 1$, the matrix $\Phi(1)$ is $M \times N$, whereas when, $t = 2, 4$, the dimensions of the matrices $\Phi(2)$ and $\Phi(4)$ are $(M/2) \times (N/2)$ and $(M/4) \times (N/4)$, respectively. Hence, the storage space needed for the matrix Φ is reduced effectively. Meanwhile, there is also a significant reduction in the memory requirements for reconstruction.

With the sparse vector x and Gaussian random measurement matrices $\Phi(t)$, we obtained the measurement vectors of length $M = 128$ by (7). Then, we initialized the reconstruction process with $\varepsilon_0 = 1$, after iterating, if $\varepsilon_n < 10^{-8}$, the process of recovery is considered to be completed, and a solution to the sparse vector \hat{x} is returned. Then, we calculate the relative error between \hat{x} and x by following:

$$Error = \frac{\|x - \hat{x}\|_2}{\|x\|_2}. \tag{16}$$

If the relative error is less than 10^{-5} , x can be considered the correct solution, and the recovery is successful. Otherwise, the recovery is considered to have failed.

If we consider the solution to be correct, we execute the operation $T_{correct} + 1$. To measure the probability of exact reconstruction, we performed 500 trials for a single sparsity value of k . Thus, the probability of exact reconstruction is measured as follows:

$$pr = \frac{T_{correct}}{T_{total}} \tag{17}$$

where T_{total} denotes the total attempts that were made. Here, $T_{total} = 500$.

It needs to be pointed out that matrices $\Phi(t)$ were generated only once during the trails.

For a different sparsity value k , the curves of probability of exact reconstruction are shown in Fig. 1. The curve with $t = 1$ denotes that the matrix we employed is $\Phi_{128 \times 256}$. Curves with $t = 2$ and 4 denote that the matrices are $\Phi_{64 \times 128}$ and $\Phi_{32 \times 64}$ respectively.

As shown in Fig. 1, when the sparsity value k is relatively small, namely $k \leq 20$, the probability of an exact reconstruction remains almost 100%, regardless of whether the dimensions of the measurement matrix are reduced. When we increase the value k , the probability of an exact reconstruction declines. Compared to the probability curve with $t = 1$, the probability curves with $t = 2$ or 4 decline quickly. However, they nevertheless maintain a high probability of an exact reconstruction. It is clear that we can recover the original sparse vector in the matrices with reduced dimensions. Furthermore, by contrasting frames (a) and (b), we can see that when sparsity

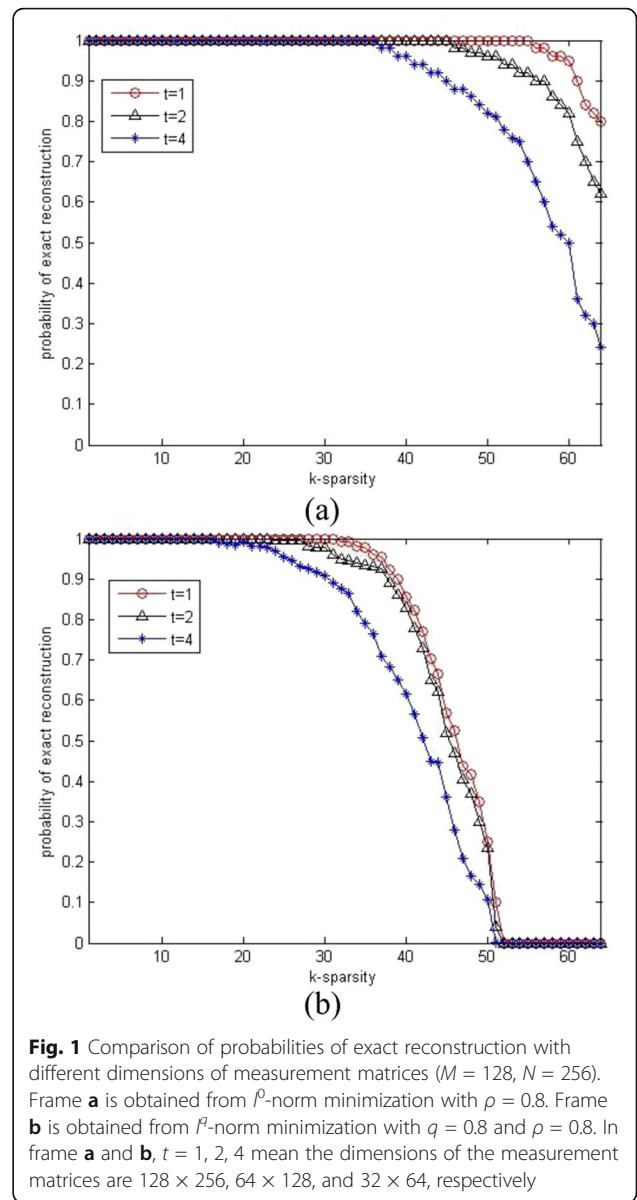


Fig. 1 Comparison of probabilities of exact reconstruction with different dimensions of measurement matrices ($M = 128, N = 256$). Frame **a** is obtained from l^p -norm minimization with $\rho = 0.8$. Frame **b** is obtained from l^q -norm minimization with $q = 0.8$ and $\rho = 0.8$. In frame **a** and **b**, $t = 1, 2, 4$ mean the dimensions of the measurement matrices are $128 \times 256, 64 \times 128,$ and 32×64 , respectively

value k approaches the limit value of $M/2$, (a) still has a higher probability of reconstruction than (b).

During the comparisons, an issue emerged that caught our attention, regarding why the probability of an exact reconstruction declines so quickly, when we reduced the dimensions of matrix $\Phi(t)$ ($t > 1$).

As shown in Eqs. (9) and (10), we can see that when we reduce the number of dimensions of matrix $\Phi(t)$, the number of coefficients is also reduced. The number of coefficients in (9) is only (M/t) . Furthermore, the measurements y with length M can be divided into (M/t) groups, where each group is defined as an adjacent measurement with length t , such that $(y_1, \dots, y_t)^T$ is the first group, and $(y_{(i-1) \times t + 1}, \dots, y_{i \times t})^T$ is the i th group. For the i th group,

$$\begin{pmatrix} \mathcal{Y}_{(i-1) \times t+1} \\ \mathcal{Y}_{(i-1) \times t+2} \\ \vdots \\ \mathcal{Y}_{i \times t} \end{pmatrix} = \begin{pmatrix} \phi_{i,1} \mathbf{x}_1 + \dots + \phi_{i,j} \mathbf{x}_{(j-1) \times t+1} + \dots + \phi_{i,N/t} \mathbf{x}_{(N/t-1) \times t+1} \\ \phi_{i,1} \mathbf{x}_2 + \dots + \phi_{i,j} \mathbf{x}_{(j-1) \times t+2} + \dots + \phi_{i,N/t} \mathbf{x}_{(N/t-1) \times t+2} \\ \vdots \\ \phi_{i,1} \mathbf{x}_t + \dots + \phi_{i,j} \mathbf{x}_{(j-1) \times t+t} + \dots + \phi_{i,N/t} \mathbf{x}_{(N/t-1) \times t+t} \end{pmatrix} \tag{18}$$

where $1 \leq i \leq (M/t)$, $1 \leq j \leq (N/t)$, and ϕ_{ij} is defined as in (8).

In (18), we see that all the coefficients for different measurements in the i th group are the same—namely, $(\phi_{i,1}, \dots, \phi_{i,N/t})$. As shown in (10), $(y_1, y_2)^T$, $(y_3, y_4)^T$, and $(y_5, y_6)^T$ are the groups.

If the original sparse vector x has a comparably large sparsity value k , it needs more iterations to derive the sparse solutions. That is, the rate of convergence is relatively slower compared to conventional CS with the IRLS algorithm. This exacerbates the decline to the probability of exact reconstruction. Whereas the sparsity value k is relatively small, less iteration are needed, and the rate of convergence is still fast, despite reducing the dimensions in the measurement matrix. To verify our analysis, we compared the rate of convergence with varying sparsity values k and $\Phi(t)$; the numerical results are shown in Fig. 2.

The sparse vector we used in this experiment had a value of k —namely, $k = 40$ or 60 . In these comparisons, 500 attempts were executed for generating the three different measurement matrices. We generated the sparse vector x only once with a given k . The numerical results on the curves represent the mean of these 500 attempts.

As shown in Fig. 2, for $k = 40$, the rate of convergence was roughly the same for different matrices. For $k = 60$, more iterations were needed to achieve a sparse solution when increasing the value of t . This showed that if the original signal is sufficiently sparse, the rate of convergence was still fast, despite a reduction in the number of dimensions of the measurement matrix.

To further compare the probability of the exact reconstruction with the matrices $\Phi(t)$ for different numbers of measurements, we conducted a third experiment, with $N = 256$ and $k = 40$. In this experiment, the conditions for completing of the reconstruction process and the success of the reconstruction were the same as they were in the first experiment. That is, there

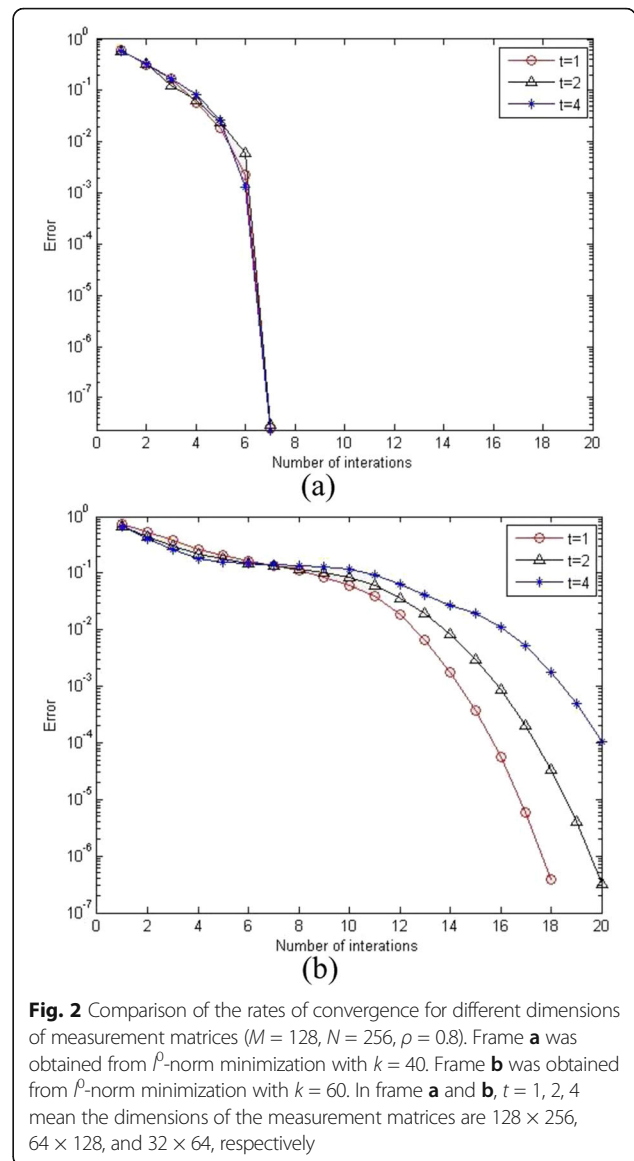


Fig. 2 Comparison of the rates of convergence for different dimensions of measurement matrices ($M = 128$, $N = 256$, $\rho = 0.8$). Frame **a** was obtained from ℓ^p -norm minimization with $k = 40$. Frame **b** was obtained from ℓ^p -norm minimization with $k = 60$. In frame **a** and **b**, $t = 1, 2, 4$ mean the dimensions of the measurement matrices are 128×256 , 64×128 , and 32×64 , respectively

were 500 attempts at generating the measurement matrices $\Phi(t)$, whereas the sparse vector x was generated only once. The numerical results on the curves represent the mean of 500 attempts, these results are shown in Fig. 3, where the number of measurements M varied from 0 to N .

As shown in Fig. 3, for the same number of measurements, the probabilities of exact reconstruction differed little with different measurement matrices. Hence, there was no need to increase the number of measurements to derive the solution when we reduced the number of dimensions of the measurement matrix.

According to the comparisons of one-dimensional sparse signals, we can see that the STP approach can reconstruct a sparse signal with a randomly

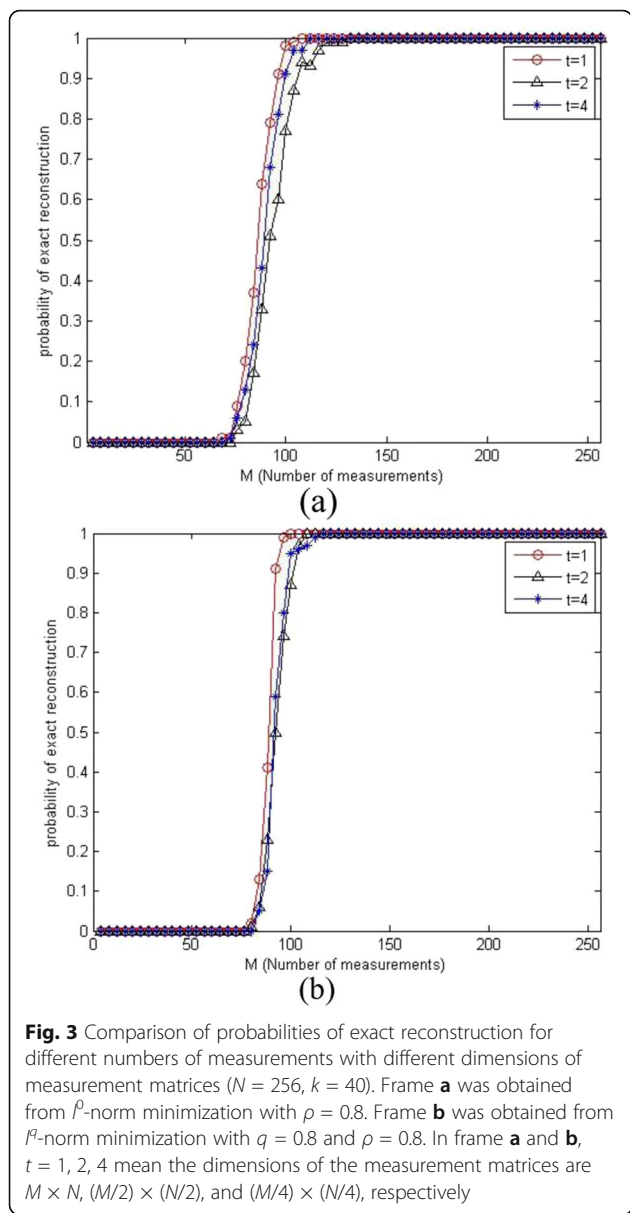


Fig. 3 Comparison of probabilities of exact reconstruction for different numbers of measurements with different dimensions of measurement matrices ($N = 256, k = 40$). Frame **a** was obtained from l^2 -norm minimization with $\rho = 0.8$. Frame **b** was obtained from l^q -norm minimization with $q = 0.8$ and $\rho = 0.8$. In frame **a** and **b**, $t = 1, 2, 4$ mean the dimensions of the measurement matrices are $M \times N, (M/2) \times (N/2),$ and $(M/4) \times (N/4),$ respectively

measurement matrix $\Phi(t)$ ($t > 1$). Moreover, performance with dimensionality reduction for a Gaussian random measurement matrix $\Phi(t)$ is generally comparable to that of the random matrix without any reduced dimensions. Furthermore, the performance with the dimensionality reduction to matrix $\Phi(t)$ depends on the sparsity of the signal x . In particular, if the original signal is sufficiently sparse, its performance with dimensionality reduction was relative good compared to that of the matrix without reduced dimensions. On the other hand, if the original signal is not sufficiently sparse, the performance declines. Therefore, there is a tradeoff between the performance of the reconstruction and the dimensionality of the measurement matrix.

4.2 Comparisons with two-dimensional signals

Here, to compare the performance with the matrices $\Phi(t)$ for two-dimensional signals, we measured the PSNR values of the reconstructed images.

In these comparisons, the signals were two-dimensional natural images. We know that signals and natural images must be sparse in a certain transform domain or dictionary, in order for them to be reconstructed exactly within the CS framework. In our experiments, we employed coefficients from the wavelet transform as two-dimensional compressible signals, and projected the coefficients onto a Gaussian random measurement matrix. When we derived the measurements y with a matrix $\Phi(t)$, the coefficients were reconstructed by IRLS with approximate l^0 -minimization. Three natural images of different sizes were used in our experiments. Lena (size: 256×256), Peppers (size: 256×256), and OT-Colon (size: 512×512), OT-Colon is a DICOM gray-scale medical image, and it can be retrieved from [36].

In this experiment, we stipulated the ratio for sampling at 0.8215, 0.75, 0.5, and 0.4375. We then generated some Gaussian random measurement matrices $\Phi(t)$, with $t = 1, 2, 4, 8,$ and 16. The number of dimensions of the matrices for $M/N = 0.5$ generated are shown in Table 2.

As shown in Table 2, with the increase in the value of t , the storage space was reduced quadratically, such that the storage space of $\Phi(16)$ was 1/256th that of $\Phi(1)$.

In this simulation, 50 attempts were made for generating these measurement matrices $\Phi(t)$, and the coefficients from wavelet transforms were generated only once. The process for reconstruction uses the IRLS algorithm with l^0 -minimization per our proposal, where $\rho = 0.8$. Visual reconstructions are shown in Figs. 4 and 5, which were reconstructed with l^0 -minimization and randomly selected from 50 attempts.

By comparing Frames (b–f) in Figs. 4 and 5, we can see that the quality of the reconstructed images remained high. That is, the subjective visual quality of the reconstructed images barely declined, despite reducing the storage spaces needed for the matrices and the memory required for reconstruction by t^2 times, where t can be 2, 4, 8, or 16. Then, we used the peak signal-to-noise ratio (PSNR) to evaluate the quality of the reconstructed images. A total of 50 attempts were carried out, and the maximum (Max), the minimum (Min), and the mean PSNR values are listed in Table 3.

Table 2 Different dimensional measurement matrices $\Phi(t)$ for images ($M/N = 0.5$)

Size of image ($N \times N$)	$t = 1$	$t = 2$	$t = 4$	$t = 8$	$t = 16$
256×256	$\Phi_{128 \times 256}$	$\Phi_{64 \times 128}$	$\Phi_{32 \times 64}$	$\Phi_{16 \times 32}$	$\Phi_{8 \times 16}$
512×512	$\Phi_{256 \times 512}$	$\Phi_{128 \times 256}$	$\Phi_{64 \times 128}$	$\Phi_{32 \times 64}$	$\Phi_{16 \times 32}$



Fig. 4 Comparison of the reconstructed images with different dimensions of measurement matrices. (Lena, $M = 128$, $N = 256$, ℓ^0 -minimization with $\rho = 0.8$). Frame **a** is the original image; Frame **b** is the reconstructed image from the original in Frame **a** using the matrix $\Phi_{128 \times 256}$; Frame **c** is the reconstructed image using the matrix $\Phi_{64 \times 128}$; Frame **d** is the reconstructed image using the matrix $\Phi_{32 \times 64}$; Frame **e** is the reconstructed image using the matrix $\Phi_{16 \times 32}$; Frame **f** is the reconstructed image using the matrix $\Phi_{8 \times 16}$

In the maximum values listed, we see that there is almost no difference among the PSNR values, regardless of whether the number of dimensions of the measurement matrix was reduced by t^2 times, even 256 times. Moreover, some values from $t > 1$ were greater than those from $t = 1$. This indicates that the proposed algorithm is effective at sampling and reconstructing sparse signals with measurement matrices where the number of

dimensions was reduced while maintaining a high level of quality. We can therefore confirm that the quality of the reconstructed image relies significantly on the random matrix generated, rather than the dimensions of the random matrix. This means that if we generate a suitable random matrix (that is, if it satisfies RIP and NSP), we can also obtain a precise reconstruction, even if the dimensions of the matrix are reduced.

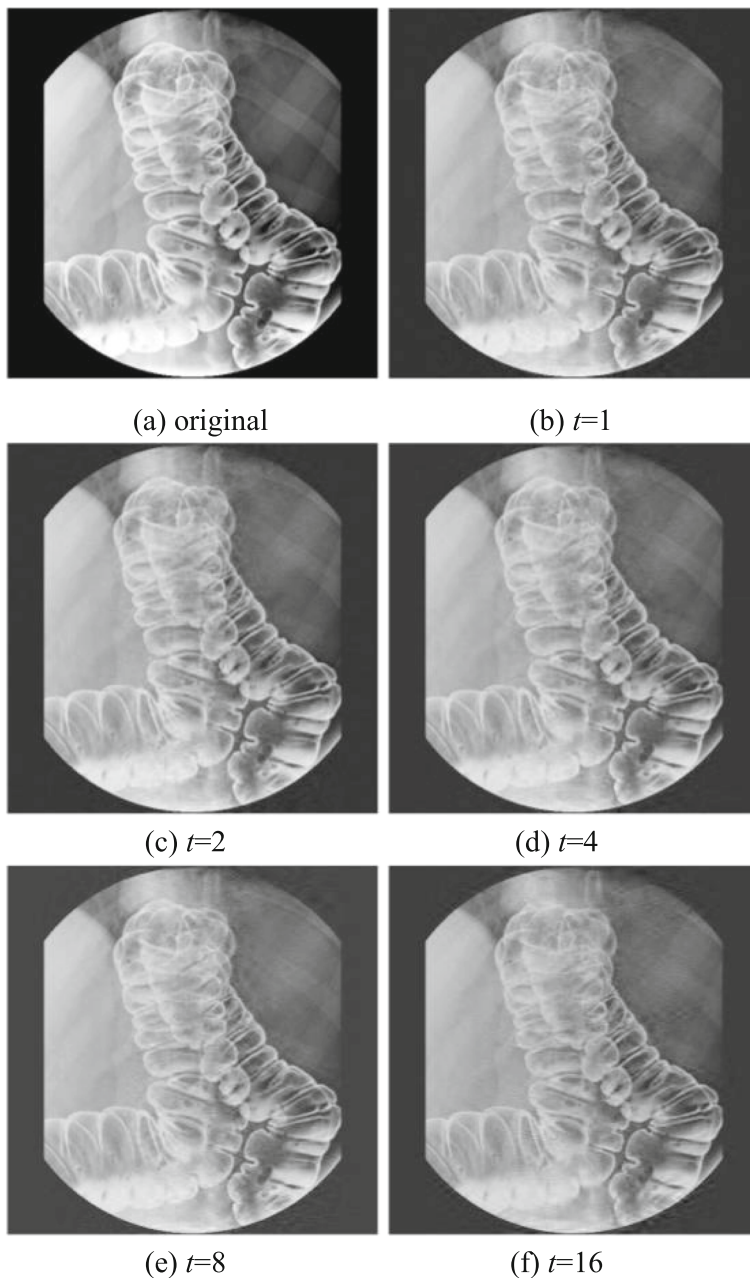


Fig. 5 Comparison of the reconstructed images with different dimensions of measurement matrices. (OT-colon, $M = 256$, $N = 512$, ℓ^0 -minimization with $\rho = 0.8$). Frame **a** is the original image; Frame **b** is the reconstructed image from the original in Frame **a** using the matrix $\Phi_{256 \times 512}$; Frame **c** is the reconstructed image using the matrix $\Phi_{128 \times 256}$; Frame **d** is the reconstructed image using the matrix $\Phi_{64 \times 128}$; Frame **e** is the reconstructed image using the matrix $\Phi_{32 \times 64}$; Frame **f** is the reconstructed image using the matrix $\Phi_{16 \times 32}$

In the minimum values listed, some values from $t > 1$ were significantly lower than those from $t = 1$. For instance, the sampling rate of Lena was 0.4375 and $t = 16$, and the PSNR was only 8.2242 dB. By calculating the corresponding mutual coherence ($\mu(\Theta^{16})$), we found that this $\mu(\Theta^{16})$ was considerably greater than others. This confirmed that the random matrix $\Phi(t)$ ($t > 1$) we generated should satisfy RIP

and NSP appropriately. Thus, we can improve the stability of reconstruction quality.

Like Gaussian random matrices, the matrices of Bernoulli, Hadamard, and Toeplitz are also the random. Therefore, we sought to verify the performance of the reconstruction with these matrices. Again, 50 attempts were made for generating random matrices with different dimensions, and the wavelet coefficients were

Table 3 Comparison of the PSNR of reconstructions with different dimensions of Gaussian measurement matrices (ℓ^0 -minimization with $\rho = 0.8$)

	Ratio of dimensionality reduction	M/N	PSNR		
			Lena	Peppers	OT-colon
Max	t = 1	0.8125	41.4152	45.1162	41.4891
	t = 2		41.2529	45.2535	41.2479
	t = 4		41.2816	45.0404	41.2863
	t = 8		41.6383	44.4560	41.3755
	t = 16		41.6666	43.9702	41.0504
	t = 1	0.7500	39.3990	42.5244	39.2847
	t = 2		39.3532	42.5405	39.1986
	t = 4		39.4859	42.8989	39.1524
	t = 8		39.3292	41.9847	39.1286
	t = 16		39.4723	41.0188	39.0994
	t = 1	0.5000	35.4444	36.4737	30.5470
	t = 2		35.3266	36.6717	30.3621
	t = 4		35.2764	37.3340	30.6017
	t = 8		35.0658	36.3877	30.5098
	t = 16		35.5209	36.6022	30.3819
	t = 1	0.4375	26.2675	27.5362	28.5509
	t = 2		26.4057	27.7582	28.4599
	t = 4		27.3657	28.1248	28.2993
	t = 8		26.8054	27.9102	27.9184
	t = 16		27.4303	29.2851	27.5377
Min	t = 1	0.8125	40.6046	44.2813	41.0969
	t = 2		40.6000	44.0375	40.5387
	t = 4		39.7232	43.0134	40.6741
	t = 8		38.3074	39.8453	38.9174
	t = 16		36.2093	37.5271	36.5640
	t = 1	0.7500	38.6814	42.0387	38.6598
	t = 2		38.4750	41.6335	38.1730
	t = 4		38.4480	39.8260	38.7530
	t = 8		37.1189	38.5990	38.1803
	t = 16		34.8398	33.6085	33.4261
	t = 1	0.5000	34.9586	36.1277	30.1636
	t = 2		34.7745	35.5442	29.9858
	t = 4		34.4683	35.7925	29.7738
	t = 8		34.0103	34.7392	29.3071
	t = 16		17.1770	8.2242	28.3865
	t = 1	0.4375	25.7846	27.1451	28.1552
	t = 2		25.5322	26.9129	27.8469
	t = 4		25.5191	26.4387	28.0847
	t = 8		24.9147	21.1803	27.5269
	t = 16		5.9977	21.0325	24.9710
Mean	t = 1	0.8125	41.0056	44.8147	41.1039

Table 3 Comparison of the PSNR of reconstructions with different dimensions of Gaussian measurement matrices (ℓ^0 -minimization with $\rho = 0.8$) (Continued)

t = 2		40.9707	44.5737	41.1145
t = 4		40.6192	44.2741	41.1011
t = 8		40.1403	43.8523	41.0180
t = 16		39.7578	42.6253	40.9543
t = 1	0.7500	39.0722	42.2699	39.1848
t = 2		38.9532	42.0751	39.1056
t = 4		38.8577	41.6732	39.0522
t = 8		38.3079	41.3472	38.6734
t = 16		36.8636	40.0724	38.3980
t = 1	0.5000	35.1674	36.3082	30.3101
t = 2		35.0681	36.1250	30.2036
t = 4		35.1892	36.2985	30.1288
t = 8		34.8385	35.6255	30.1171
t = 16		34.4991	35.2709	29.1196
t = 1	0.4375	25.9959	27.3607	28.3191
t = 2		25.9664	27.3312	28.2886
t = 4		25.9391	27.0231	28.2801
t = 8		25.6793	26.1873	27.7021
t = 16		25.7151	26.8239	27.3460

generated only once for a natural image. The IRLS reconstruction algorithm was used with approximate ℓ^0 -minimization. The results are shown in Table 4. We opted to use 256×256 images from Lena and Peppers.

As demonstrated by the results in Table 4, the proposed STP approach is suitable for other random measurement matrices. It produces high-quality images with a suitable number of dimensions in the random matrix.

Furthermore, in order to compare it with other low-memory techniques, PSNR with a reconstructed Lena under different sampling ratios are shown in Fig. 6. The curves represented are the mean of 50 attempts.

As shown in Fig. 6, we can see that the PSNR of the reconstructed Lena at different sampling ratios was better than the other two low-memory techniques.

During these experiments, we focused on the performance of the reconstructed one- and two-dimensional signals, where the dimensions of the random measurement matrices we used were reduced by t^2 times ($t \geq 1$). As mentioned above, when $t = 1$, the dimensions of the random matrix were not reduced. This can be treated as equivalent to conventional CS. When $t > 1$, such as $t = 2$, the dimensions of the random matrix were reduced by four times, and if $t = 8$, the dimensions can be reduced by 64 times. From the results, we can see that increasing the value of t is an effective way to reduce the storage space of the random measurement matrix and

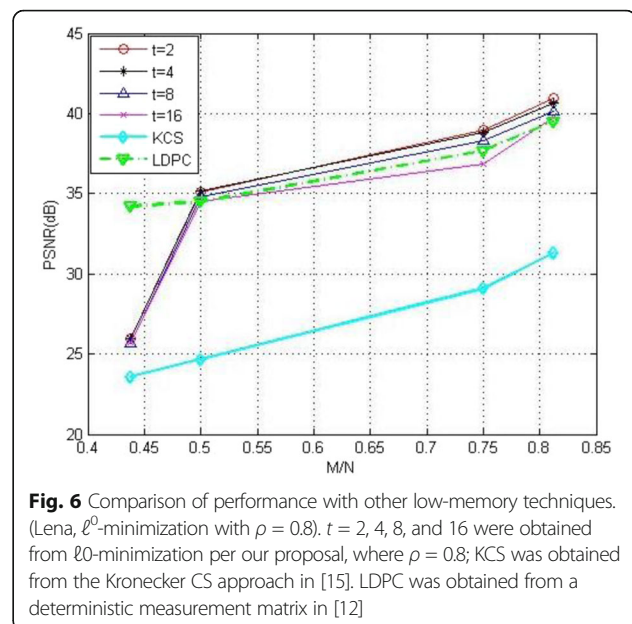
Table 4 Tests on other random measurement matrices with different dimensions. ($M/N = 0.5$, ℓ^0 -minimization with $\rho = 0.8$)

Images	Ratio of dimensionality reduction	t	PSNR		
			Bernoulli	Hadamard	Toeplitz
Lena (256 × 256)	Max	$t = 1$	35.3431	35.2931	35.6554
		$t = 2$	35.3431	35.3935	35.7550
		$t = 4$	35.6510	35.2870	35.4465
		$t = 8$	35.2639	35.3398	35.8770
		$t = 16$	36.2619	35.0118	35.9155
	Min	$t = 1$	34.9759	35.0136	34.5715
		$t = 2$	34.8563	35.0056	34.3634
		$t = 4$	34.8053	34.7791	34.1974
		$t = 8$	34.1163	34.1614	33.8705
		$t = 16$	15.8923	33.0596	13.8632
	Mean	$t = 1$	35.0819	35.1956	35.1887
		$t = 2$	35.1102	35.2093	35.1094
		$t = 4$	35.0779	35.1018	34.9980
		$t = 8$	34.6943	34.7907	34.9373
		$t = 16$	34.5464	34.4744	34.1420
	Peppers (256 × 256)	Max	$t = 1$	36.4659	36.5146
$t = 2$			36.8068	36.5124	37.1701
$t = 4$			36.6284	36.4531	37.0168
$t = 8$			37.1965	36.6096	37.0168
$t = 16$			37.6965	36.7270	37.8743
Min		$t = 1$	36.1021	35.9480	36.0183
		$t = 2$	35.9979	36.0554	35.2744
		$t = 4$	35.4798	35.7992	35.6017
		$t = 8$	34.2548	35.9329	35.6017
		$t = 16$	17.7105	34.3432	15.4983
Mean		$t = 1$	36.3395	36.2944	36.3954
		$t = 2$	36.3561	36.3110	36.2393
		$t = 4$	36.0967	36.1344	36.4202
		$t = 8$	35.9423	36.1665	36.4202
		$t = 16$	35.3987	35.7520	35.4352

the memory required for reconstruction. The tradeoff between more precise reconstructions and constrained storage requirements depends on the specific application, and this can significantly influence the physical implementation of CS in images, especially with embedded systems and FPGAs, where storage is limited.

5 Conclusions

In this paper, a novel approach to STP-CS was proposed. Our work aimed at reducing the amount of storage space needed with conventional compressive sensing. We provided a theoretical analysis of the acquisition process with STP-CS and that of the recovery algorithm



with IRLS. Furthermore, numerical experiments were conducted on one-dimensional sparse signals and two-dimensional compressible signals, where the two-dimensional signals were the coefficients from wavelet transforms. A comparison of the numerical experiments demonstrated the effectiveness of the STP approach. Moreover, they show that our proposed STP approach for compressive sensing did not improve the quality of the reconstructed signal, yet reduced the storage space of the measurement matrix and the memory requirements for sampling and reconstruction. With a suitable reduction to the dimensionality of the random measurement matrix (e.g., when $t = 2, 4, 8$, or 16), we achieved a recovery performance similar to that obtained when $t = 1$, while the storage requirements reduced by t^2 times.

Although the dimensions of the random matrix were reduced and the PSNR of the reconstructed signal declined somewhat, it was also possible to improve the accuracy, provided that the generated random matrix satisfies the RIP and NSP appropriately. Moreover, the proposed algorithm is easy to execute, and additional operations for sampling and reconstruction are unnecessary. This can significantly influence physical implementations of the CS.

However, more investigation is required to improve the recovery performance and optimize the sampling and reconstruction processes. Further work remains in constructing the so-called independent identically distributed random matrix with fewer dimensions. Moreover, we shall attempt to optimize the matrix based on QR decomposition, which could help to improve the incoherence between the measurement matrix and the

sparse basis. This will help to improve the performance of reconstruction. This has also motivated us to employ other measurement matrices, such as the structurally random matrix, low-rank matrix, rank-one matrix, and etc., in order to reduce required storage while maintaining or improving performance quality. Parallel framework [37, 38] will also be considered to reduce the time consuming during the reconstruction.

Acknowledgements

The authors wish to thank the anonymous reviewers for their valuable comments and for their help in finding errors. We appreciate their assistance in improving the quality and the clarity of the manuscript.

Funding

This work was supported by the Science and Technology Project of Zhejiang Province, China (Grant No. 2015C33074, 2015C33083).

Authors' contributions

Wang and Ye drafted the manuscript and implemented the experiments for verifying the feasibility of the proposed scheme. Ruan and Chen participated in the design of the proposed scheme and drafted the manuscript. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 22 February 2017 Accepted: 17 July 2017

Published online: 28 July 2017

References

1. D.L. Donoho, Compressed sensing. *IEEE Trans. Inf. Theory* **52**(4), 1289–1306 (2006)
2. E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **52**(2), 489–509 (2006)
3. E. Candès, J. Romberg, Sparsity and incoherence in compressive sampling. *Inverse problems* **23**(3), 969–985 (2007)
4. E.J. Candès, J.K. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.* **59**(8), 1207–1223 (2006)
5. L. Gan, Block compressed sensing of natural images, digital signal processing, 2007 15th international conference on. *IEEE*, 403–406 (2007)
6. V. Abolghasemi, S. Ferdowsi, S. Sanei, A block-wise random sampling approach: Compressed sensing problem. *Journal of AI and Data Mining* **3**(1), 93–100 (2015)
7. N. Cleju, Optimized projections for compressed sensing via rank-constrained nearest correlation matrix. *Appl. Comput. Harmon. Anal.* **36**(3), 495–507 (2014)
8. T.T. Do, L. Gan, N.H. Nguyen, et al., Fast and efficient compressive sensing using structurally random matrices. *IEEE Trans. Signal Process.* **60**(1), 139–154 (2012)
9. A. Amini, F. Marvasti, Deterministic construction of binary, bipolar, and ternary compressed sensing matrices. *IEEE Trans. Inf. Theory* **57**(4), 2360–2370 (2011)
10. R. Calderbank, S. Howard, S. Jafarpour, Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property. *IEEE Journal of Selected Topics in Signal Processing* **4**(2), 358–374 (2010)
11. L. Gan, T. T. Do, T. D. Tran, Fast compressive imaging using scrambled block Hadamard ensemble, signal processing conference, 2008 16th European. *IEEE*, 2008, 1–5
12. H. Yuan, H. Song, X. Sun, et al., Compressive sensing measurement matrix construction based on improved size compatible array LDPC code. *IET Image Process.* **9**(11), 993–1001 (2015)
13. Xu, Yangyang, W. Yin, and S. Osher. Learning circulant sensing kernels, *Inverse Problems & Imaging*, **8**(3)(2014) 901–923
14. V. Tiwari, P.P. Bansod, A. Kumar. Designing sparse sensing matrix for compressive sensing to reconstruct high resolution medical images. *Cogent Eng.* **2**(1), 1–13 (2015)
15. B. Zhang, X. Tong, W. Wang, et al., The research of Kronecker product-based measurement matrix of compressive sensing. *EURASIP J. Wirel. Commun. Netw.* **1**(2013), 1–5 (2013)
16. M.F. Duarte, R.G. Baraniuk, Kronecker compressive sensing. *IEEE Trans. Image Process.* **21**(2), 494–504 (2012)
17. R. Otazo, E. Candès, D.K. Sodickson, Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components. *Magn. Reson. Med.* **73**(3), 1125–1136 (2015)
18. T.T. Cai, A. Zhang, Sparse representation of a polytope and recovery of sparse signals and low-rank matrices. *IEEE Trans. Inf. Theory* **60**(1), 122–132 (2014)
19. E. Riegler, D. Stotz, H. Bolcskei. Information-theoretic limits of matrix completion, information theory (ISIT), 2015 IEEE international symposium on. *IEEE*. 1836–1840 (2015)
20. K. Lee, Y. Wu, Y. Bresler, Near optimal compressed sensing of sparse rank-one matrices via sparse power factorization. *Computer Science* **92**(4), 621–624 (2013)
21. D. Cheng, H. Qi, Z. Li, *Analysis and Control of Boolean Networks: A Semi-Tensor Product Approach* (Springer Science & Business Media, London, 2011), pp. 19–53
22. D.Z. Cheng, H. Qi, Y. Zhao, *An Introduction to Semi-Tensor Product of Matrices and Its Applications* (World Scientific, Singapore, 2012)
23. D.Z. Cheng, H. Qi, A linear representation of dynamics of Boolean networks. *IEEE Trans. Autom. Control* **55**(10), 2251–2258 (2010)
24. J.E. Feng, J. Yao, P. Cui, Singular Boolean networks: Semi-tensor product approach. *SCIENCE CHINA Inf. Sci.* **56**(11), 1–14 (2013)
25. E. Jurrus, S. Watanabe, R.J. Giuly, et al., Semi-automated neuron boundary detection and nonbranching process segmentation in electron microscopy images. *Neuroinformatics* **11**(1), 5–29 (2013)
26. J. Zhong, D. Lin, On maximum length nonlinear feedback shift registers using a Boolean network approach, control conference (CCC), 2014 33rd Chinese. *IEEE*. 2502–2507 (2014)
27. H. Wang, D. Lin. Stability and linearization of multi-valued nonlinear feedback shift registers, *IACR Cryptol. ePrint Arch.* 253 (2015)
28. R. Chartrand, W. Yin. Iteratively reweighted algorithms for compressive sensing, Acoustics, speech and signal processing, 2008. ICASSP 2008. *IEEE international conference on. IEEE*. 3869–3872 (2008)
29. I. Daubechies, R. DeVore, M. Fornasier, et al., Iteratively reweighted least squares minimization for sparse recovery. *Commun. Pure Appl. Math.* **63**(1), 1–38 (2010)
30. E.J. Candès, M.B. Wakin, S.P. Boyd, Enhancing sparsity by reweighted ℓ_1 -minimization. *J. Fourier Anal. Appl.* **14**(5), 877–905 (2008)
31. R. Saab, Özgür Yılmaz. Sparse recovery by non-convex optimization—instance optimality, *Applied & Computational Harmonic Analysis*, **29**(1)(2010) 30–48
32. E.J. Candès, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. Inf. Theory* **52**(12), 5406–5425 (2006)
33. X.L. Cheng, X. Zheng, W.M. Han, Algorithms on the sparse solution of under-determined linear systems. *Applied Mathematics A Journal of Chinese Universities* **28**(2), 235–248 (2013)
34. H. Bu, R. Tao, X. Bai, et al. Regularized smoothed ℓ_0 norm algorithm and its application to CS-based radar imaging, *Signal Process.*, **112**(2016) 115–122
35. C. Zhang, S. Song, X. Wen, et al., Improved sparse decomposition based on a smoothed L_0 norm using a Laplacian kernel to select features from fMRI data. *J. Neurosci. Methods* **245**, 15–24 (2015)
36. S. Barr. (2013) Medical image samples. [online]. Available: <http://www.barre.nom.fr/medical/samples/>
37. C. Yan, Y. Zhang, J. Xu, et al., A highly parallel framework for HEVC coding unit partitioning tree decision on many-core processors. *IEEE Signal Processing Letters* **21**(5), 573–576 (2014)
38. C. Yan, Y. Zhang, J. Xu, et al., Efficient parallel framework for HEVC motion estimation on many-Core processors. *IEEE Transactions on Circuits & Systems for Video Technology* **24**(12), 2077–2089 (2014)