Noname manuscript No.
(will be inserted by the editor)

Adaptive Gradient-based Block Compressive Sensing with Sparsity for Noisy Images

**Hui-Huang Zhao, Paul L. Rosin, Yu-Kun Lai,
Jin-Hua Zheng and Yao-Nan Wang**

Received: date / Accepted: date

Abstract This paper develops a novel adaptive gradient-based block compressive sensing (AGbBCS_SP) methodology for noisy image compression and reconstruction. The AGbBCS_SP approach splits an image into blocks by maximizing their sparsity, and reconstructs images by solving a convex optimization problem. In block compressive sensing, commonly used square block shapes cannot always produce the best results. The main contribution of our paper is to provide an adaptive method for block shape selection, improving noisy image reconstruction performance. The proposed algorithm can adaptively achieve better results by using the sparsity of pixels to adaptively select block shape. Experimental results with different image sets demonstrate that our AGbBCS_SP method is able to achieve better performance, in terms of peak signal to noise ratio (PSNR) and computational cost, than several classical algorithms.

Keywords Block Compressive Sensing (CS) · Adaptive · Convex Optimization · Sparsity

Hui-huang Zhao
Hunan Provincial Key Laboratory of Intelligent Information Processing and Application, Hunan, China and College of Computer Science and Technology, Hengyang Normal University, Hengyang China
E-mail: happyday.huihuang@gmail.com
Paul L. Rosin
School of Computer Science and Informatics, Cardiff University, Cardiff, UK
E-mail: Paul.Rosin@cs.cardiff.ac.uk
Yu-Kun Lai
School of Computer Science and Informatics, Cardiff University, Cardiff, UK
E-mail: laiy4@cs.cardiff.ac.uk
Jin-Hua Zheng
College of Computer Science and Technology, Hengyang Normal University, Hengyang China
E-mail: jhzheng@xtu.edu.cn
Yao-Nan Wang
College of Electrical and Information Engineering, Hunan University, Changsha, China
E-mail: yaonan@hnu.cn

1 Introduction

Compressive Sensing (CS) is a sampling paradigm that provides signal compression at a significantly lower rate than the Nyquist rate [15], [16]. It is based on signal sparse representation [24] and has been successfully applied in a wide variety of applications in recent years, including image processing [9,27,30,45], Synthetic Aperture Radar (SAR) [2], Internet of things [25,43,28], Magnetic Resonance Imaging (MRI) [38], video [51,29], and solder joint image compression [53]. The authors in [32] proposed a novel reconstruction method for X-ray images based on CS. The authors in [41] developed a new method of fast encoding for SAR raw data by using CS theory to compress and reconstruct SAR raw data. In [40] compressive sensing and matrix completion techniques are applied to recover the original spectral signals. Simulation results proved that the output was improved.

In this paper, we develop a novel CS algorithm named AGbBCS_SP for image compression and reconstruction, which is particularly beneficial for noisy images. The main contributions of this paper are summarized as follows:

- We find that the square block shape used in existing methods cannot always achieve best results. Therefore, we propose a multi-shape block splitting strategy for block Compressive Sensing. Besides splitting the image into square blocks, we also split it into rectangular blocks with different shapes (i.e. aspect ratios). By doing so, nearby pixels which are similar have a high probability to be assigned to the same block, leading to more effective compression.
- Our adaptive Compressive Sensing scheme makes a practical assumption that only a small, randomly chosen part of the image needs to be known. Our method automatically selects the appropriate block shape which maximizes the sparsity of the signal in the known region.
- The control factor for sparse regularization is also important for effective image reconstruction. We propose an adaptive approach to selecting a suitable control factor, by comparing the sparsity of the reconstruction results.

After the block shape selection, the image is split using this block shape, and then the recently proposed gradient-based method for Compressive Sensing [53] is applied. Our method involves two adaptive selection steps, optimizing the block shape and control factor, respectively. The results show that the reconstruction performance is improved significantly.

The rest of this paper is organized as follows. In section 2, we introduce some related work on CS. In section 3, we introduce the theory of Compressive Sensing and the gradient-based method for the convex optimization problem. In section 4, we describe the AGbBCS_SP method for image compression. Experimental results and comparison are shown in section 5. Finally, we conclude our paper in section 6.

2 Related work

2.1 Compressive Sensing Algorithms

The major challenge in CS is to approximate a signal given a vector of samples. In recent years, many methods have been proposed which can be roughly divided into six categories:

1. **Convex Optimization Algorithms**. These techniques solve a convex problem which is used to approximate the target signal, including Basis Pursuit [8], Greedy Basis Pursuit (GBP) [21], Basis Pursuit De-Noising (BPDN) [31].
2. **Greedy Iterative Algorithms**. These methods build up an approximation by making locally optimal choices step by step. Examples include Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP) [44], regularized OMP (ROMP) [36], Compressive Sampling MP (CoSaMP) [35] and Subspace Pursuit (SP) [10].
3. **Iterative Thresholding Algorithms**. Iterative approaches for the CS recovery problem are faster than the convex optimization method. For this type of algorithm, which assumes that the signal is sparse, correct measurements are recovered by soft or hard thresholding [4], [14] starting from an initial random noise measurement matrix.
4. **Combinatorial / Sublinear Algorithms**. This type of algorithm recovers a sparse signal through group testing [20], such as Heavy Hitters on Steroids (HHS) [39].
5. **Non Convex Minimization Algorithms**. Non-convex local minimization techniques recover compressive sensing signals from far less measurements by replacing the l_1-norm by the l_p-norm where $p \leq 1$ [7]. An example of an algorithm proposed in the literature that uses this technique is Iterative Re-weighted Least Squares [11].
6. **Bregman Iterative Algorithms**. When applied to CS problems, the iterative approach using Bregman distance regularization achieves reconstruction in four to six iterations [37].

2.2 Block Based Compressive Sensing (BCS)

In the methods above, a column or row of an image is normally viewed as a vector. But in many applications the nonzero elements of sparse vectors tend to cluster in blocks [17]. In this case the sampling problems over unions of subspaces can be converted into block-sparse recovery problems. In order to improve the performance, [19] proposed and studied block compressive sensing for natural images and this method involves Wiener filtering and projection onto the convex set and hard thresholding in the transform domain. For 512×512 size images, the author suggested block dimension 32. [33] proposed a BCS_SPL method with a variant of projected Landweber (PL) iteration and smoothing. It needed more than 200 iterations and they used different block dimensions (16, 32 or 64) according to different image sizes. [34] studied a DDWT_BCS method which was based on a 5-level dual-tree discrete wavelet transform (DDWT) which was used as the sparsity basis. For a 512×512 image, they set the block size as 16×16. [50] proposed a BCS method based on a Bayesian learning framework for Fetal ECG (FECG) telemonitoring, and it could greatly reduce code execution time in the data compression stage. They used certain block partitions and the block size ranged from 4 to 90. Both [42] and [48] proposed an adaptive block-based compressive sensing approach which collected a different number of samples of the measurement matrix for each block. [9] studied block compressive sensing in wireless sensor networks and [3] analyzed the block sampling strategies in compressive sensing. They showed the optimal number of blocks depended on the properties of block coherence.

[49] and [26] studied block compressed sensing with projected Landweber (PL). [18] developed BCS_SPL method based on a smoothed projected Landweber reconstruction algorithm. BCS_SPL has obvious defects since the Wiener filter and iterative projected Landweber discard partial information in the image. [46] proposed a block compressed sensing method based on iterative re-weighted l_1 norm minimization. [52] developed a block compressed sensing method for solder joint images based on CoSaMP. In those methods the row and column sizes of the measurement matrix are the square of the block size. So with increased block size, the algorithm requires substantially more memory.

Despite many CS algorithms appearing in the literature, there are still many challenges in compressive sampling to approximate a signal, especially for noisy signals. On one hand, in most methods, a column or row of an image is normally viewed as a vector, and so the local 2D spatial image information is ignored. All the block compressive sensing methods mentioned above used *fixed* block size and are not adaptive. On the other hand, the computational cost for many methods, such as CoSaMP, is unsatisfactory. Some classical methods, such as OMP, have good computational efficiency, but their reconstruction performance needs to be improved. Third, some of the algorithms require tuning several parameters, and are not adaptive.

3 Compressive Sensing Methodology

3.1 A Brief Review of Compressive Sensing

Given an image, the first step of CS is the construction of a k-sparse representation, where k is the number of the non-zero entries of the sparse signal. Most natural signals can be made sparse by applying orthogonal transforms, such as Wavelet Transform, Fast Fourier Transform and Discrete Cosine Transform (DCT) [6]. This step is represented as

$$x = \Psi s, \tag{1}$$

where s is an N-dimensional noise free image, x is a weighted N-dimensional vector (sparse signal with k nonzero elements), and Ψ is an $N \times N$ orthogonal basis matrix. The second step is compression. In this step, a random measurement matrix is applied to the sparse signal according to the following equation:

$$y = \Phi x = \Phi \Psi s, \tag{2}$$

where Φ is an $M \times N$ random measurement matrix $(M < N)$. In most images or videos, there is some noise [5,54]. For a noisy image, the equation is generalized as:

$$y = \Phi\Psi s + w, \tag{3}$$

where w is an N-dimensional noise signal (or measurement error). Let M be the number of measurements (the row dimension of y) sufficient for high probability of successful reconstruction. As expected, signal x in Eq.(2) and Eq.(3) may be estimated from measurement y by solving the convex minimization problem [44,35] as follows.

$$\begin{cases} \text{minimize} & \|x\|_1 \\ \text{subject to :} & \|\Phi x - y\|_2 \leq \varepsilon, \end{cases} \tag{4}$$

where ε is an upper bound on the noise in the data.

The robustness of CS heavily relies on a notion called the *restricted isometry property* (RIP) [47]. RIP is defined as follows,

$$(1-\delta_k)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta_k)\|x\|_2^2, \tag{5}$$

where $\|\cdot\|_2^2$ defines the l_2 norm, and δ_k is the k-restricted isometry constant of a matrix. RIP is used to ensure that all subsets of k columns taken from Φ are nearly orthogonal.

3.2 Gradient-Based Method for Convex Optimization Problems

Generally Eq.(4) is a constrained minimization problem of a convex function. One of the simplest methods for solving a convex minimization problem is the gradient-based algorithm which generates a sequence x_k via

$$x_0 \in \mathbb{R}^N, x_k = x_{k-1} - t_k \nabla g(x_{k-1}), \tag{6}$$

where $g(x)$ is a convex function, and $t_k > 0$ is a suitable step size. For a signal in Eq.(3), let us think about an objective function $F(x) = g(x) + f(x)$, where $g(x)$ is convex, and $f(x) = \lambda\|x\|_1$. In our method, it is more natural to study the closely related problem

$$\arg\min_x \|\Phi x - y\|_2^2 + \lambda\|x\|_1. \tag{7}$$

At point x_{k-1}, the function $F(x)$ can be approximated by the following quadratic function

$$Q_L(x, x_{k-1}) = \left\{ g(y) + < x - x_{k-1}, \nabla g(x_{k-1}) > + \frac{1}{2t_k} \| x - x_{k-1} \|_2^2 \right\}, \tag{8}$$

which admits a unique minimizer,

$$PL(x_{k-1}) = \arg\min_x \{Q_L(x, x_{k-1}), x \in \mathbb{R}^N\}. \tag{9}$$

We solve this problem using a gradient-based method, in which an iteration parameter t_k is replaced by a constant $1/L$ which is related to the Lipschitz constant [1].

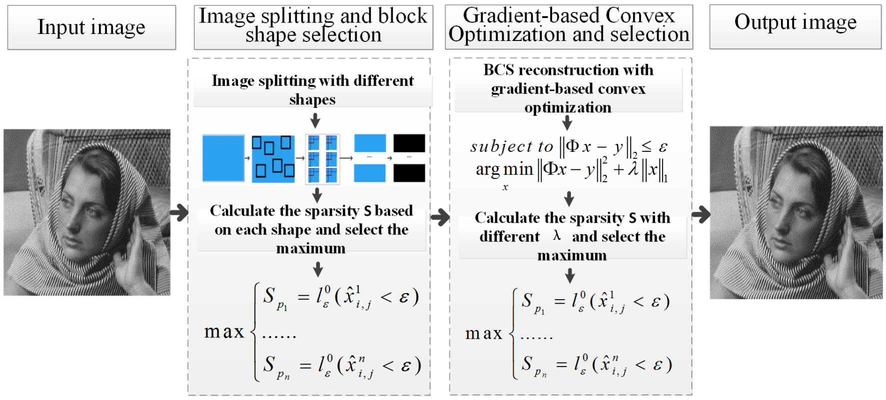


Fig. 1 The flow chart of our AGbBCS_SP approach.

4 The Adaptive Gradient-based Block Compressive Sensing with Sparsity

4.1 Framework of our method

The framework of our AGbBCS_SP approach is shown in figure 1. For an image, the main steps are:

- We propose an adaptive block CS approach in which we consider different block shapes for splitting the image into a set of non-overlapping blocks of equal shape. Assuming that the information of the entire image is unknown, our method randomly selects a small part of the image and reconstructs it, and adaptively selects one block shape which can maximize the sparsity of the signal.
- The original image is sparsified by an orthogonal transformation, treating its compression as a convex optimization problem, followed by applying a gradient-based method. Each block is then transformed into a one-dimensional data vector. Here, we assume the problem to be convex with the Lipschitz gradient. Aiming at improving the efficiency, we replace an iteration parameter by the Lipschitz constant [53].
- We apply the proposed gradient-based method for reconstruction. The proposed method also adaptively selects a control factor which controls an l_1 norm expression in the optimization problem by comparing the sparsity of the reconstruction results. After an inverse transformation, the reconstructed signal can be obtained. Finally, each one-dimensional data vector is transformed into a two-dimensional block, and those make up the image.

4.2 Block Compressive Sensing and Multi-shape Block Split Strategy

Given an $N_1 \times N_2$ image, it is split into small blocks of size $n_1 \times n_2$. Let f_i represent the vectorized signal of the i-th block through raster scanning, i=1, 2, …, K, and $K =$

$\frac{N_1 N_2}{n_1 n_2}$. One is able to get an m-dimensional sampled vector y_B through the following linear transformation,

$$y_B = \Phi_B f_i, \tag{10}$$

where Φ_B is an $m \times n_1 n_2$ measurement matrix, $m \ll n_1 n_2$. The block CS method is memory efficient as we just need to store an $m \times n_1 n_2$ Gaussian random matrix Φ_B, rather than the full $M \times N_1 N_2$ one. Small data requires less memory storage and allows faster processing, while large data produces more accurate reconstruction.

In existing methods, the blocks in the Block Compressive Sensing are fixed as squares. However, there are many different block aspect ratios with the same number of pixels. Unlike common methods, we split the image into different shapes. Given an $\tilde{N} \times \tilde{N}$ image (assuming $\tilde{N}$ is a power of 2 for simplicity), the shape of a block is $w \times h$, so

$$\begin{cases} w = 2^a, \\ h = 2^b, \\ a = 0, 1, 2, 3, \ldots \log_2 \tilde{N}. \\ b = \log_2 \tilde{N} - a, \end{cases} \tag{11}$$

For example, 9 aspect ratios are defined to split a 256×256 image with the following block-shapes: 1×256, 2×128, 4×64, 8×32, 16×16, 32×8, 64×4, 128×2 and 256×1. As we will discuss later in section 5.1, some block shapes (especially those closer to squares) are more likely to provide effective reconstruction. Also, using closer-to-square blocks also means that these blocks can be fit in smaller square regions, e.g. 8×32, 16×16, 32×8 blocks can be fit in 32×32 squares, whereas 1×256 blocks cannot. As we will discuss in section 4.3, this makes adaptive selection more effective. Detailed discussions will be presented in the experimental results.

4.3 Adaptive Block Shape Selection

In most cases, the information of the entire signal (image) is unknown. It is hard to select one block shape from several shapes if the image content is unknown. So we make a practical assumption that only a small part of the image is known and propose a new approach based on sparsity for block shape selection. We highlight the block shape selection step in our approach.

First, we randomly select a small percentage of image pixels that make up known regions. These regions are then split into smaller block shapes considering the various aspect ratios specified in Eq.(11). We reconstruct them, calculate their sparsity, and then select the block shape which maximizes sparsity.

For an image, firstly, it is split into T non-overlapping regions of size $P \times Q$, where $K_1 = T \times p$ are known regions, and p is the proportion. So K_1 regions (size $P \times Q$) are selected. There are K_2 block sizes in Eq.(11) $w_k \times h_k, (k = 1, 2, ..., K_2)$ that fit within $P \times Q$ regions. Then for K_1 regions (size $P \times Q$), they are split into K_3 blocks of size $w_k \times h_k$. Given that $\hat{x}$ is defined as the reconstructed result in Eq.(7), the summed sparsity of its blocks is defined as

$$S_p = l_\varepsilon^0 \left(\hat{x}_{i,j} \leq \varepsilon \right), \tag{12}$$

Algorithm 1: Block Shape Selection with Sparsity

Input : An input image s, a percentage p;
Output: The selected block size $w \times h$

Procedure:
Step 1:
Split s into T blocks of size of $P \times Q$
$K_1 = T \times p$
K_1 regions (each of size $P \times Q$) are selected, and those regions collectively form $\hat{s}$.
Step 2:
K_2 block shapes are considered: $w_1 \times h_1, w_2 \times h_2, ..., w_{K_2} \times h_{K_2}$.
$\hat{s}$ is split into K_3 blocks altogether with $w_k \times h_k$ through Eq.(11)
For the k-th block size $\hat{s}$={$\hat{s}^{(k)}(1), \hat{s}^{(k)}(2), \dots, \hat{s}^{(k)}(K_3)$} .
for $k = 1$ *to* K_2 **do**
  $\hat{s}^{(k)} = \emptyset$.
  for $j = 1$ *to* K_3 **do**
    Add a new signal $\hat{s}^{(k)}(j)$ to $\hat{s}^{(k)}$.
  end
end
Step 3:
for $k = 1$ *to* K_2 **do**
  Get $\hat{x}^k$ through Eq.(7) with $\hat{s}^{(k)}$
  $S_{p_k} = l^0_\varepsilon\left(\hat{x}^k \le \varepsilon\right)$ through Eq.(12),
end
$S_{p_d} = \max\{S_{p_1}, S_{p_2}, ..., S_{p_k}\}$
The d-th block shape is chosen, and the block size is $w_d \times h_d$.
Output w_d and h_d.

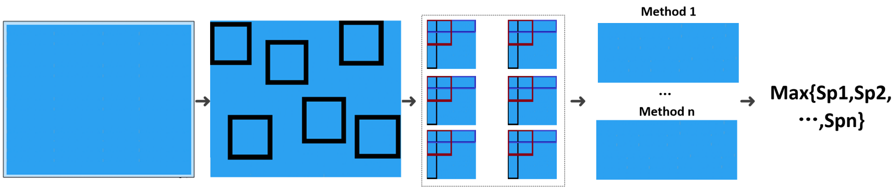


Fig. 2 Adaptive block selection based on sparsity.

where $\hat{x}_{i,j}$ is the element at location (i, j) in the reconstructed result $\hat{x}$, and $l^0_\varepsilon(\cdot)$ is a function defined in [22]. Thus, we propose the adaptive block shape selection with sparsity algorithm whose details are shown in Algorithm 1. For example, given a 256×256 image, we set $p = 0.25$. We consider splitting the image into $T = 64$ regions of size $P \times Q = 32 \times 32$, and $K_1 = 64 \times 0.25 = 16$ blocks are randomly selected, so that $K_3 = 16 \times 4 = 64$. With 32×32 regions, we consider $K_2 = 3$ block sizes 8×32, 16×16 and 32×8 which fit within the region. The process of block shape selection is shown in figure 2.

In this paper, we consider splitting an image into blocks in different ways, and the configuration with the largest sparsity is chosen for CS.

Algorithm 2: Adaptive Gradient-based Block Compressive Sensing

Input : An image I of size $\tilde{N} \times \tilde{N}$; a sparse signal transform matrix $\Psi \in \mathbb{R}^{WH \times WH}$; a measurement matrix $\Phi \in \mathbb{R}^{M \times WH}$; where W and H are the chosen block width and height, and M is the sampling rate; Lipschitz constant $L = 0.5$; the number of iterations $J = M/4$.

Output: The reconstructed image s.

Procedure:

Step 1:

begin: I is split into T regions, and p is a percentage, $K_1 = T \times p$ regions are selected.

One block shape $W \times H$ is chosen by Algorithm 1,

I is split into K_4 blocks with $W \times H$ block size.

end

Step 2:

begin: Set the block counter k=1, $\lambda = 1$, and the iteration counter j=1, $S_{p_{max}} = 0$.

while $\lambda \leq 100$ **do**

  while $k \leq K_2$ **do**

    Transform each block into a data vector; $y_0 = x_0 = 0 \in \mathbb{R}^{WH}, t_1 = 1$;

    while $j \leq J$ **do**

      $z_j^k = PL(y_j^k)$, solved through [53].

      $t_{j+1}^k = \frac{1+\sqrt{1+4t_j^{k2}}}{2}$

      x_j^k=argmin$\{F(x^k) : x^k = z_j^k, x_{j-1}^k\}$

      $y_{j+1}^k = x_j^k + \frac{t_j^k}{t_{j+1}^k}(z_j^k - x_j^k) + \frac{t_j^k - 1}{t_{j+1}^k}(x_j^k - x_{j-1}^k)$

    end

    Collect all the $\hat{x}_J^k$ to form $\hat{x}$.

  end

  $S_p = l_\varepsilon^0\,(\hat{x} \leq \varepsilon)$ through Eq.(12),

  If $S_p > S_{p_{max}}$

    $S_{p_{max}} = S_p$

    $\hat{\hat{x}} = \hat{x}$

  Endif

end

$s' = \Psi^{-1}\hat{\hat{x}}$.

For each one-dimensional data vector in s', transform it into an $W \times H$ block.

Collect all the blocks to form the reconstructed image s.

end

4.4 Adaptive Block Compressive Sensing with Sparsity Algorithm

During the minimization of Eq.(7), λ can be used to improve the result with different sampling rates. Usually $\lambda = M/4$, but in our proposed method, we set $\lambda \in [1, 100]$, and we adaptively choose λ such that the largest sparsity is achieved. Thus, we propose our AGbBCS_SP algorithm whose details are shown in Algorithm 2, where the basic sparse optimization is based on [53].

There are two steps in our method. The first step compares the sparsity for block shape selection. The second step uses the selected block shape to split the image and reconstruct the image. In comparison with other reconstruction algorithms, our algorithm has the following characteristics:

- Unlike the traditional block compressive sensing approaches, in which the block is a fixed square shape, our method considers splitting an image into multiple

Table 1 Correct ratio of AGbBCS_SP ($M = 128$)

dataset	image number	best shape selected	ratio
Holidays	157	136	86.62%
Copydays	812	691	85.10%

block shapes. Similar pixels have a high probability to be assigned to the same block.

- Furthermore the block shape selection is adaptive and it is determined by maximizing sparsity.
- Suppose the information of the entire image is unknown. Our algorithm randomly selects a small part of the image to perform block shape selection.
- The algorithm also adaptively selects the control factor λ according to the sparsity of the results.

5 Experiments and Discussion

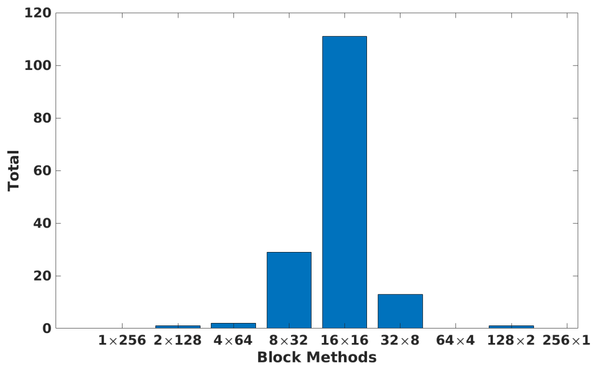
In order to evaluate the quality of the reconstructed results, many researchers used the Peak Signal to Noise Rate (PSNR) to measure the result quality in image processing [13]. In our study, PSNR is also used to compare the experimental results. The experiments were implemented on an Intel Core i5 with 2.70 GHz CPU. The test images include some standard ones (such as *woman*), INRIA Copydays dataset (157 images), and INRIA Holidays dataset (812 images) [23] to which *salt* & *pepper* noise is added with $\delta = 0.05$ by default. Since some methods require the image size to be a power of 2, we have cropped all the images to 256×256.

5.1 Experiments with different block aspect ratios

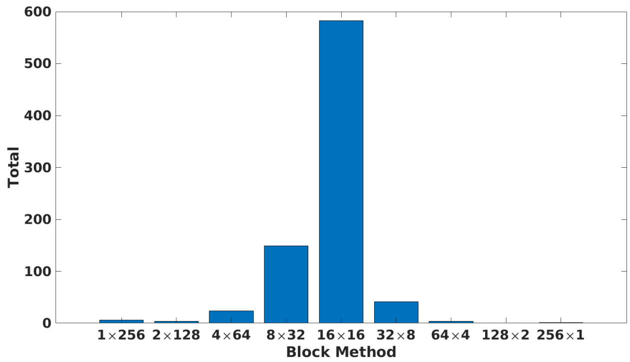
Given a 256×256 image, the block-shapes 1×256, 2×128, 4×64, 8×32, 16×16, 32×8, 64×4, 128×2 and 256×1 are considered. We used the INRIA Copydays and the INRIA Holidays datasets and the noise level is set $\delta = 0.05$. With the sampling rate $M = 128$ and $\lambda = M/4$, we test different block shapes. Then we select the best shape, and the number of times that each block shape is best is shown in figures 3(a) and (b) for the two datasets, respectively.

We find that a square block (16×16) cannot always get the best results, and 8×32, 16×16, and 32×8 can achieve the top three results. So in our AGbBCS_SP method, three block shapes are chosen. As described above, we consider splitting a 256×256 image into 64 regions, each of size 32×32, and $64 \times 0.25 = 16$ blocks are randomly selected to calculate sparsity for three block shapes (8×32, 16×16, and 32×8). Then we choose the block shape which can get maximum sparsity for the given image.

Based on the introduction above, we do a test with the INRIA Copydays dataset and the INRIA Holidays dataset with added noise $\delta = 0.05$, and we count the number of images our method selects the best block shape for based on $M = 128$, and the results are shown in table 1.



(a) Copydays dataset



(b) Holidays dataset

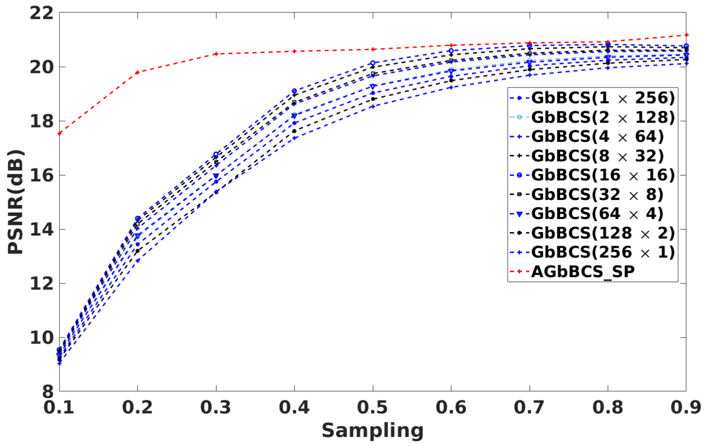
Fig. 3 The number of images that each block shape is best in the (a) INRIA and (b) Holidays datasets.

From table 1, one can see that the proposed AGbBCS_SP approach can achieve a good result for block shape selection, where the ratios of correctly selecting the best block shapes are 86.62% and 85.10%. The average reconstruction results with different block shapes and our method are shown in figures 4 (a) and (b).

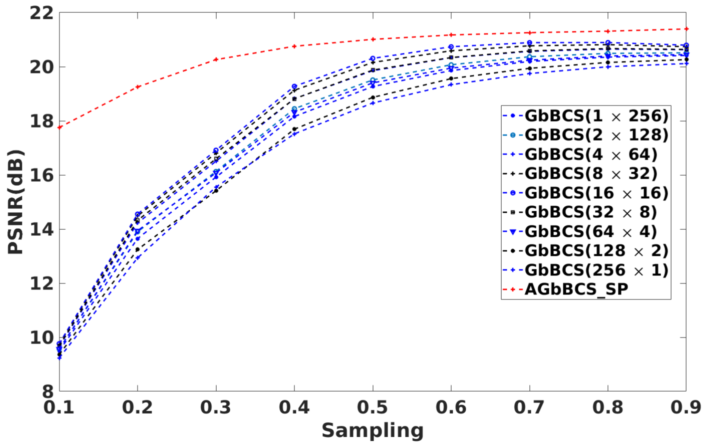
One can see from figures 4 (a) and (b) that AGbBCS_SP, which can adaptively select block shape and λ which can adjust the l_1 norm in the minimization problem, achieves the best results. Especially when the sampling rate $u \leq 0.6$, the PSNR is improved greatly.

5.2 The comparison of reconstruction results

Now let us compare the proposed AGbBCS_SP with the popular methods SP [10], OMP [44], BOMP [17], CoSaMP [12], BCoSaMP [52] BCS_SPL [18] and Deep Image Prior [45]. In BOMP, BCoSaMP and BCS_SPL, the block is set to a square shape (size 16×16). Deep Image Prior is not based on blocks, and we simply recover



(a) Copydays dataset

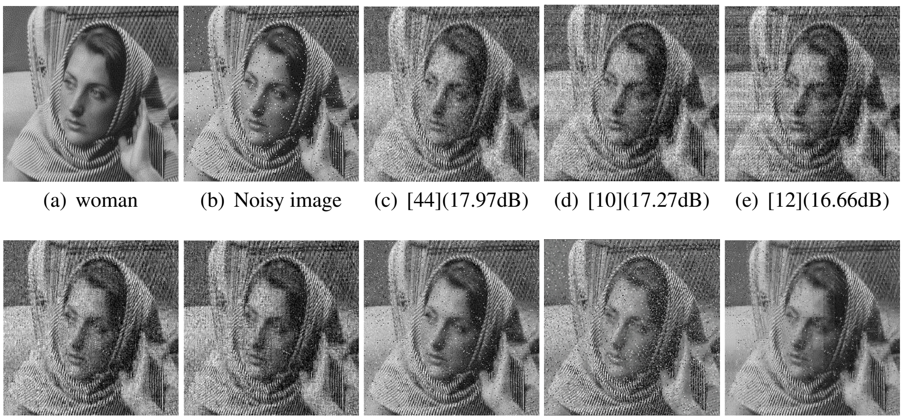


(b) Holidays dataset

Fig. 4 Quantitative comparison based on different block shapes for INRIA datasets

images based on a sparse sampling using the authors' code. The test image *woman* is used (size 256×256) with added noise $\delta = 0.03$, as shown in figure 5(a). The reconstruction results based on popular methods with sampling rate $M = 200$ are shown in figures 5(b-g) and the reconstruction result based on our AGbBCS_SP with the same sampling rate, is shown in figure 5(h).

We can see that our method can achieve a better result than SP, OMP, BOMP, CoSaMP, BCoSaMP, BCS_SPL and Deep Image Prior.With more noise added and $M = 128$ in test image *woman*, the PSNR comparisons are shown in figure 6. One can see from figure 6 that, compared to SP, OMP, BOMP, CoSaMP, BCoSaMP and Deep Image Prior, our method achieves the best result.With increasing noise in the



(a) woman (b) Noisy image (c) [44](17.97dB) (d) [10](17.27dB) (e) [12](16.66dB)

(f) [17](18.08dB) (g) [52](17.01dB) (h) [18](20.27dB) (i) [45](19.55dB) (j) Our(21.19dB)

Fig. 5 Reconstruction results based on different methods

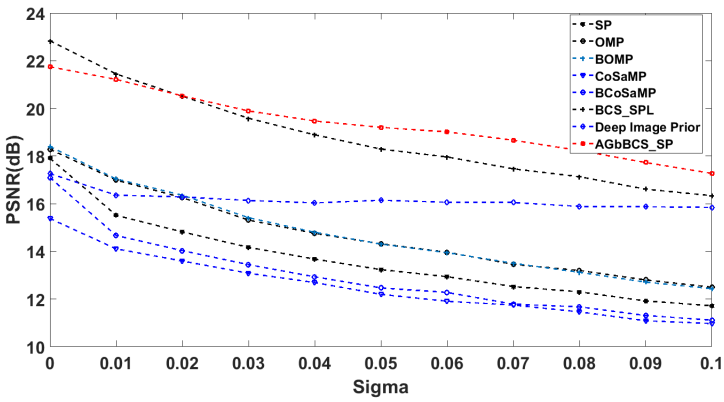


Fig. 6 PSNR comparison of different levels of added noise

image, Deep Image Prior can achieve a constant PSNR value. When $\delta > 0.025$, our method achieves a better PSNR result than BCS_SPL.

With increasing numbers of samples in the noisy image (see figure 5(a)) the PSNR comparisons are shown in figure 7. One can see from figure 7 that, compared to SP, OMP, BOMP, CoSaMP, BCoSaMP and Deep Image Prior, our method can achieve best result. When $M \geq 70$, our method achieves a better result than BCS_SPL. We also compare their number of iterations for reconstructing figure 5(a) with sampling rate $M = 200$, the iteration and the PSNR comparison are shown in figure 8 and figure 9.

One can see that our method achieves the best result after 20 iterations. It achieves a better result than SP, OMP, BOMP, CoSaMP, BCoSaMP after 7 iterations. From figure 9, one can see that Deep Image Prior can achieve a better PSNR around 350 iterations. So in the following experiments, the number of iterations is set to 350 for Deep Image Prior.

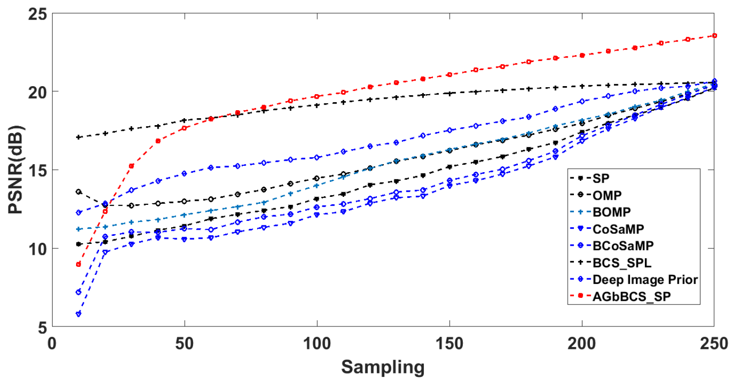


Fig. 7 PSNR comparison based on different sampling rates for *woman*

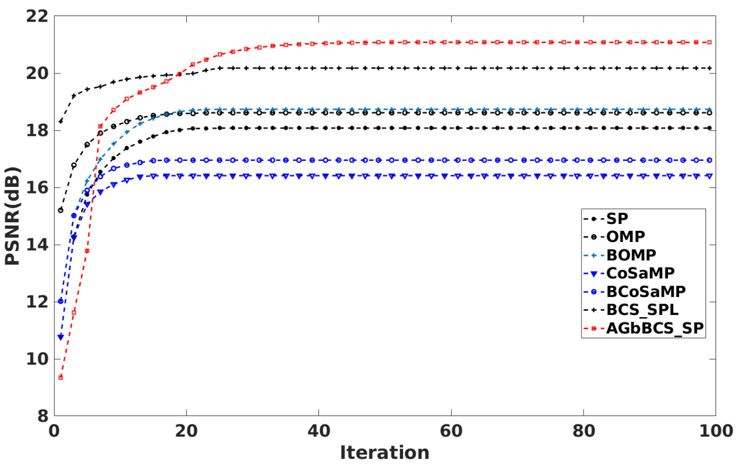


Fig. 8 PSNR comparison for different numbers of iterations for *woman*

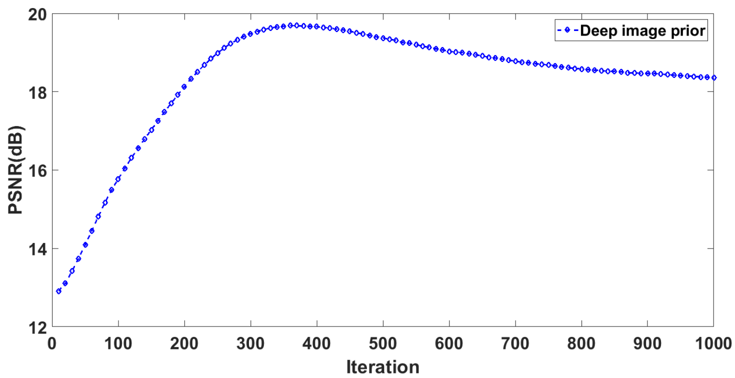
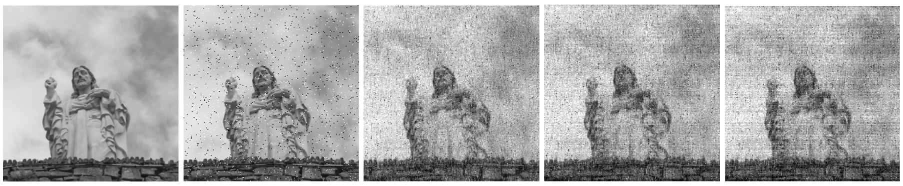


Fig. 9 PSNR result of Deep Image Prior with different numbers of iterations for *woman*

(a) (b)

Fig. 10 Source example images.



(a) Image (b) Noisy image (c) [44](19.57dB) (d) [10](18.85dB) (e) [12](18.41dB)

(f) [17](19.04dB) (g) [52](17.53) (h) [18](21.14dB) (i) [45](20.99dB) (j) Our(23.73dB)

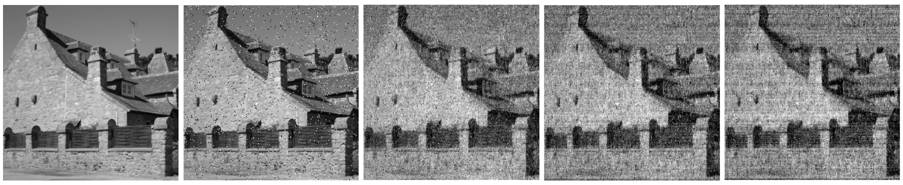
Fig. 11 More reconstruction experiments based on different methods applied to figure 10(a)

More image reconstruction results with sampling rate $M = 200$ and $\delta = 0.03$ based on different methods are shown in figures 11 and 12 which contain special content or background (shown in figure 10).

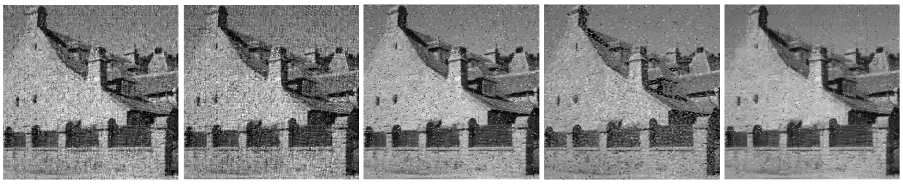
We can see that figure 10(a) contains sky as its background, while in figure 10(b), the buildings have lots of blocky areas. The content for these images is well suited for using block compressive sensing. Compared with other methods, our method can achieve the best PNSR results, but there are some blocky artifacts in the reconstructed images. This is because some blocks are sparser than other blocks, so it can generate better results than the blocks around them.

In the next experiment, we used the INRIA Copydays and Holidays datasets with added noise $\delta = 0.05$. The comparison results are shown through the experiments with different numbers of samples (from sample rate 0.1 to 0.9). The results are shown in figures 13, 14(a) and (b).

From figures 13 and 14, one can see that the proposed AGbBCS_SP approach always obtains better PSNR results compared to SP, OMP, BOMP, CoSaMP, BCoSaMP and Deep Image Prior. Increasing the number of samples can improve the reconstruc-

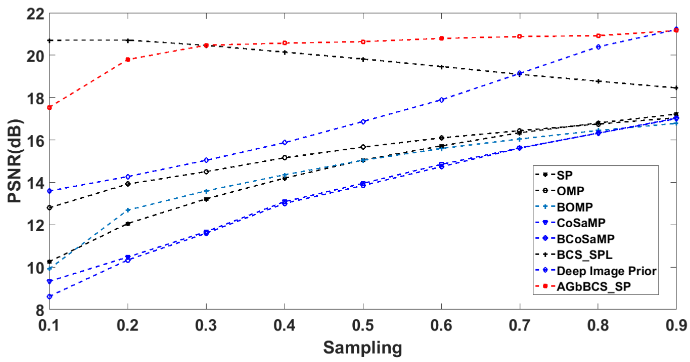


(a) Image (b) Noisy image (c) [44](18.97dB) (d) [10](18.14dB) (e) [12](17.45dB)

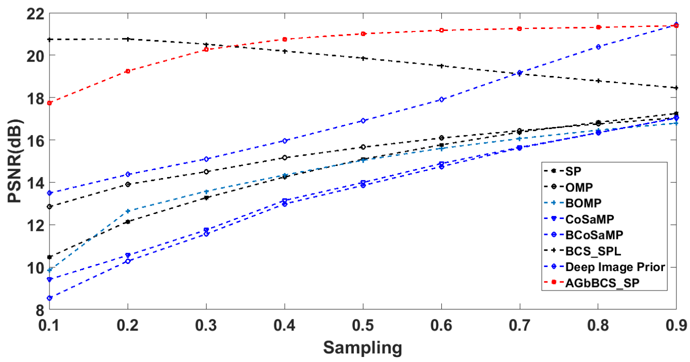


(f) [17](18.99dB) (g) [52](17.70dB) (h) [18](21.13dB) (i) [45](21.04dB) (j) Our(22.78dB)

Fig. 12 More reconstruction experiments based on different methods applied to figure 10(b)

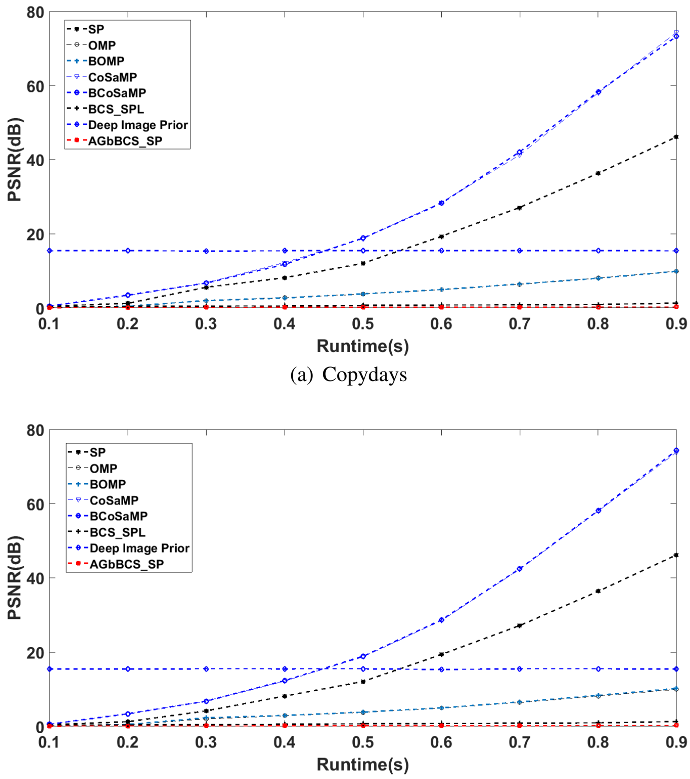


(a) Copydays



(b) Holidays

Fig. 13 Quantitative comparison based on different methods for INRIA datasets.



(a) Copydays

(b) Holidays

Fig. 14 Running time comparison based on different methods for the INRIA dataset

tion results. When the sampling rate $u = M/N > 0.3$, the proposed algorithm can achieve better results than BCS_SPL too. When the sampling rate $u = M/N > 0.9$, Deep Image Prior can achieve a similar results with AGbBCS_SP . We also find that BCS_SPL has a poor performance on de-noising. With an increasing number of samples, BCS_SPL gets worse reconstruction results. Deep Image Prior can keep a constant runtime around 15s with an increasing number of samples. At the same time, our AGbBCS_SP method has lower computational cost than BCS_SPL, CoSaMP, BCoSaMP and SP. Increasing the sample rate can improve the reconstruction result. Unlike BCS_SPL, CoSaMP, BCoSaMP and SP, our method can keep the low computational cost with the increasing sample rate.

5.3 Reconstruction for Images with *Gaussian* Noise

In the next experiment, we used the INRIA Copydays dataset with added *Gaussian* noise $\sigma \in [0.01, 0.1]$. The comparison results are shown through the experiments with sampling rate $M = 128 (u = 0.5)$. The results are shown in figure 15.

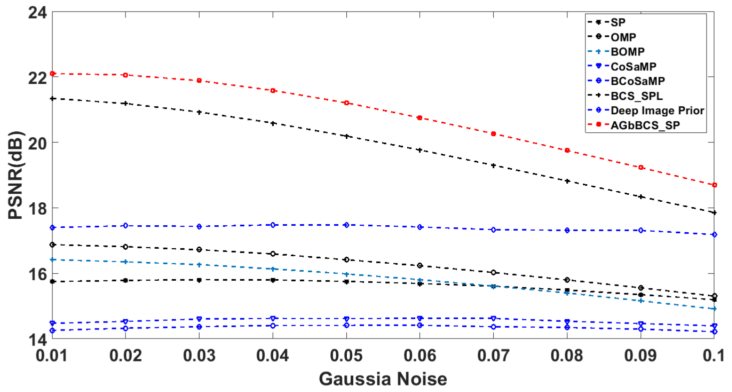


Fig. 15 Quantitative comparison based on different methods for the INRIA Copydays dataset.

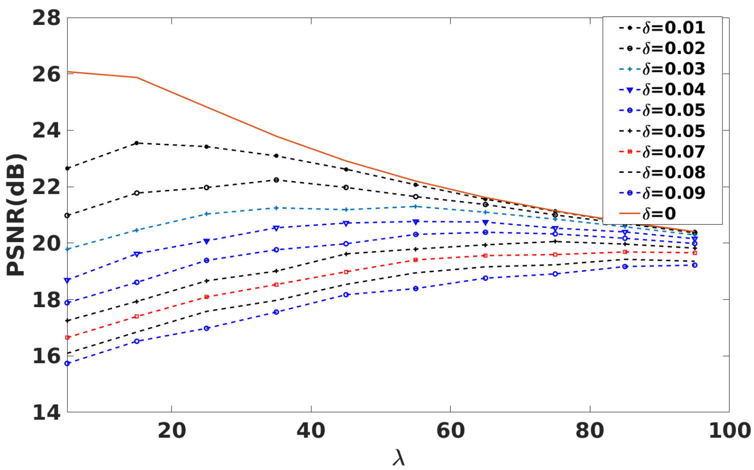


Fig. 16 Parameter analysis with different δ and λ for *woman*

From figure 15, one can see that the proposed AGbBCS_SP approach always obtains the best result in terms of PSNR as compared to SP, OMP, BOMP, CoSaMP, BCoSaMP, Deep Image Prior, and BCS_SPL.

5.4 Parameters Analysis

λ can be used to improve the result for AGbBCS_SP to cope with different noise levels δ. We do an experiment with the image *woman* with different values of δ and λ, and set $M = 200$. We find that for different values of δ, $\lambda \in [15,60]$ typically achieves the best results, and larger λ tends to produce better results with higher noise level (larger δ). Note that in our approach λ is automatically selected, which reduces the user's burden for parameter tuning.

6 Conclusions

This paper proposes an adaptive gradient-based block compressive sensing (AGb-BCS_SP) approach on the basis of the sparsity of the image.

- Besides splitting the image into square blocks, a new image block splitting method for compressive sensing is proposed. We split the image into rectangular blocks with different shapes (aspect ratios). Our adaptive Compressive Sensing scheme makes a practical assumption that only a small, randomly chosen image part requires to be known. The proposed method automatically selects the control factor and the appropriate block shape that maximizes the sparsity of the signal in the known region.
- After block shape selection, the image is split by using the selected block size, the proposed gradient-based method is applied for reconstruction. The proposed method also adaptively selects a control factor which controls an l_1 norm expression in the optimization problem by comparing the sparsity of the reconstruction results. Finally through an inverse transformation, the reconstructed signal can be obtained.
- Experiments reveal that in block compressive sensing the square block shape does not always produce the best results. Our algorithm can adaptively achieve better results by using the sparsity of pixels to adaptively select block shape. The proposed algorithm can achieve better results according to PSNR than classical algorithms with different block shapes (1×256, 2×128, 4×64, 8×32, 16×16, 32×8, 64×4, 128×2 and 256×1). The performance is improved greatly. When *Gaussian* noise [0.01, 0.1] is added, the proposed algorithm maintains better performance than SP, OMP, BOMP, CoSaMP, BCoSaMP and BCS_SPL according to their average PSNRs. For different levels of noise δ, the proposed method for adaptive selection of λ produces better results than existing methods. The proposed algorithm can achieve the best results in average PSNR than the classical algorithms SP, OMP, BOMP, CoSaMP and BCoSaMP on several datasets. With added *salt* & *pepper* noise $\delta = 0.05$. BCS_SPL can achieve better results than the proposed algorithm when the sampling rate $u \leq 0.3$. However, BCS_SPL has a poor performance on de-noising. With an increasing number of samples, BCS_SPL gets worse reconstruction results. When the sampling rate $u > 0.3$, the proposed algorithm can achieve better average PSNR than BCS_SPL.
- We also find that if a block can achieve greater sparsity than its neighboring blocks, it can generate a better reconstruction result than its neighbors. Such visual differences lead to some blocky artifacts in the reconstruction image. Future work will investigate how to avoid blocky artifacts, and more relationships between sparsity of pixels and block shape will also be researched.

Acknowledgements

This work was supported by National Natural Science Foundation of China (61503128), Science and Technology Plan Project of Hunan Province (2016TP102), Scientific Research Fund of Hunan Provincial Education Department (16C0226,17C0223,18A333), Hengyang guided science and technology projects and Application-oriented Special Disciplines (Hengkefa [2018]60-31), Double First-Class University Project of Hunan Province (Xiangjiaotong [2018]469), Hunan Province Special Funds of Central Government for Guiding Local Science and Technology Development (2018CT5001) and Subject Group Construction Project of Hengyang Normal University (18XKQ02). We would like to thank NVIDIA for the GPU donation.

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. Siam Journal on Imaging Sciences **2**(1), 183–202 (2009)
2. Bi, D., Xie, Y., Li, X., Zheng, Y.R.: Efficient 2-D synthetic aperture radar image reconstruction from compressed sampling using a parallel operator splitting structure. Digital Signal Processing **50**, 171–179 (2016)
3. Bigot, J., Boyer, C., Weiss, P.: An analysis of block sampling strategies in compressed sensing. IEEE Transactions on Information Theory **62**(4), 2125–2139 (2016)
4. Blumensath, T., Davies, M.E.: Iterative hard thresholding for compressed sensing. Applied and Computational Harmonic Analysis **27**(3), 265–274 (2009)
5. Bo, C., Wang, D.: A registration-based tracking algorithm based on noise separation. Optik-International Journal for Light and Electron Optics **126**(24), 5806–5811 (2015)
6. Cai, T.T., Wang, L.: Orthogonal matching pursuit for sparse signal recovery with noise. Information Theory, IEEE Transactions on **57**(7), 4680–4688 (2011)
7. Chartrand, R.: Exact reconstruction of sparse signals via nonconvex minimization. Signal Processing Letters, IEEE **14**(10), 707–710 (2007)
8. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM journal on scientific computing **20**(1), 33–61 (1998)
9. Cui, H., Zhang, S., Gan, X., Shen, M., Wang, X., Tian, X.: Information recovery via block compressed sensing in wireless sensor networks. In: 2016 IEEE International Conference on Communications (ICC), pp. 1–6 (2016)
10. Dai, W., Milenkovic, O.: Subspace pursuit for compressive sensing signal reconstruction. Information Theory, IEEE Transactions on **55**(5), 2230–2249 (2009)
11. Daubechies, I., DeVore, R., Fornasier, M., Güntürk, C.S.: Iteratively reweighted least squares minimization for sparse recovery. Communications on Pure and Applied Mathematics **63**(1), 1–38 (2010)
12. Davenport, M.A., Needell, D., Wakin, M.B.: Signal space CoSaMP for sparse recovery with redundant dictionaries. IEEE Transactions on Information Theory **59**(10), 6820–6829 (2013)
13. Deswal, S., Singhania, S., Gupta, S., Garg, P.: An optimised fuzzy approach to remove mixed noise from images. International Journal of Signal Processing, Image Processing and Pattern Recognition **9**(4), 293–322 (2016)
14. Donoho, D.L.: De-noising by soft-thresholding. Information Theory, IEEE Transactions on **41**(3), 613–627 (1995)
15. Donoho, D.L.: Compressed sensing. IEEE Transactions on information theory **52**(4), 1289–1306 (2006)
16. Donoho, D.L., Elad, M., Temlyakov, V.N.: Stable recovery of sparse overcomplete representations in the presence of noise. IEEE Transactions on information theory **52**(1), 6–18 (2006)
17. Eldar, Y.C., Kuppinger, P., Blcskei, H.: Block-sparse signals: Uncertainty relations and efficient recovery. IEEE Transactions on Signal Processing **58**(6), 3042–3054 (2010)
18. Fowler, J.E., Mun, S., Tramel, E.W.: Multiscale block compressed sensing with smoothed projected Landweber reconstruction. In: Signal Processing Conference, 2011 European, pp. 564–568 (2015)
19. Gan, L.: Block compressed sensing of natural images. In: Digital Signal Processing, 2007 15th International Conference on, pp. 403–406. IEEE (2007)
20. Gilbert, A.C., Hemenway, B., Rudra, A., Strauss, M.J., Wootters, M.: Recovering simple signals. In: Information Theory and Applications Workshop (ITA), 2012, pp. 382–391 (2012)
21. Huggins, P.S., Zucker, S.W.: Greedy basis pursuit. Signal Processing, IEEE Transactions on **55**(7), 3760–3772 (2007)

22. Hurley, N., Rickard, S.: Comparing measures of sparsity. IEEE Transactions on Information Theory **55**(10), 4723–4741 (2009)
23. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. Computer Vision–ECCV 2008 pp. 304–317 (2008)
24. Li, F., Lu, H., Wang, D., Wu, Y., Zhang, K.: Dual group structured tracking. IEEE Transactions on Circuits and Systems for Video Technology **26**(9), 1697–1708 (2016)
25. Li, S., Da Xu, L., Wang, X.: Compressed sensing signal and data acquisition in wireless sensor networks and internet of things. IEEE Transactions on Industrial Informatics **9**(4), 2177–2186 (2013)
26. Liu, H., Wang, W.: Block compressed sensing reconstruction with adaptive-thresholding projected landweber for aerial imagery. Journal of Applied Remote Sensing **9**(1), 095,037–095,037 (2015)
27. Lu, H., Li, B., Zhu, J., Li, Y., Li, Y., Xu, X., He, L., Li, X., Li, J., Serikawa, S.: Wound intensity correction and segmentation with convolutional neural networks. Concurrency and computation: practice and experience **29**(6), e3927 (2017)
28. Lu, H., Li, Y., Chen, M., Kim, H., Serikawa, S.: Brain intelligence: go beyond artificial intelligence. Mobile Networks and Applications **23**(2), 368–375 (2018)
29. Lu, H., Li, Y., Mu, S., Wang, D., Kim, H., Serikawa, S.: Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. IEEE Internet of Things Journal (2017)
30. Lu, H., Li, Y., Uemura, T., Kim, H., Serikawa, S.: Low illumination underwater light field images reconstruction using deep convolutional neural networks. Future Generation Computer Systems (2018)
31. Lu, W., Vaswani, N.: Modified basis pursuit denoising (modified-BPDN) for noisy compressive sensing with partially known support. In: Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, pp. 3926–3929 (2010)
32. Melli, S.A., Wahid, K.A., Babyn, P., Montgomery, J., Snead, E., El-Gayed, A., Pettitt, M., Wolkowski, B., Wesolowski, M.: A compressed sensing based reconstruction algorithm for synchrotron source propagation-based x-ray phase contrast computed tomography. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **806**, 307–317 (2016)
33. Mun, S., Fowler, J.E.: Block compressed sensing of images using directional transforms. In: Image Processing (ICIP), 2009 16th IEEE International Conference on, pp. 3021–3024. IEEE (2009)
34. Mun, S., Fowler, J.E.: DPCM for quantized block-based compressed sensing of images. In: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European, pp. 1424–1428. IEEE (2012)
35. Needell, D., Tropp, J.A.: CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Applied and Computational Harmonic Analysis **26**(3), 301–321 (2009)
36. Needell, D., Vershynin, R.: Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. Foundations of computational mathematics **9**(3), 317–334 (2009)
37. Osher, S., Mao, Y., Dong, B., Yin, W.: Fast linearized Bregman iteration for compressive sensing and sparse denoising. arXiv preprint arXiv:1104.0262 (2011)
38. Parikh, N., Ream, J.M., Zhang, H.C., Block, K.T., Chandarana, H., Rosenkrantz, A.B.: Performance of simultaneous high temporal resolution quantitative perfusion imaging of bladder tumors and conventional multi-phase urography using a novel free-breathing continuously acquired radial compressed-sensing mri sequence. Magnetic resonance imaging **34**(5), 694–698 (2016)
39. Qaisar, S., Bilal, R.M., Iqbal, W., Naureen, M., Lee, S.: Compressive sensing: from theory to applications, a survey. Communications and Networks, Journal of **15**(5), 443–456 (2013)
40. Qin, Z., Liu, Y., Gao, Y., Elkashlan, M., Nallanathan, A.: Wireless powered cognitive radio networks with compressive sensing and matrix completion. IEEE Transactions on Communications **65**(4), 1464–1476 (2017)
41. Ren, X., Qiao, L.: A novel strategy for inverse synthetic aperture radar imaging based on improved compressive sensing. IEEJ Transactions on Electrical and Electronic Engineering **11**(2), 140–145 (2016)
42. Safavi, S.H., Torkamani-Azar, F.: Sparsity-aware adaptive block-based compressive sensing. IET Signal Processing **11**(1), 36–42 (2016)
43. Serikawa, S., Lu, H.: Underwater image dehazing using joint trilateral filter. Computers & Electrical Engineering **40**(1), 41–50 (2014)
44. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. Information Theory, IEEE Transactions on **53**(12), 4655–4666 (2007)
45. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 9446–9454 (2018)

46. Unde, A.S., Deepthi, P.: Block compressive sensing: Individual and joint reconstruction of correlated images. Journal of Visual Communication and Image Representation **44**, 187–197 (2017)
47. Wang, J., Kwon, S., Li, P., Shim, B.: Recovery of sparse signals via generalized orthogonal matching pursuit: A new analysis. IEEE Transactions on Signal Processing **64**(4), 1076–1089 (2016)
48. Xu, J., Qiao, Y., Fu, Z.: Adaptive perceptual block compressive sensing for image compression. IEICE TRANSACTIONS on Information and Systems **99**(6), 1702–1706 (2016)
49. You, H., Zhu, J.: Image reconstruction based on block-based compressive sensing. In: Proceedings of the 38th Australasian Computer Science Conference (ACSC 2015), vol. 27, p. 30 (2015)
50. Zhang, Z., Jung, T.P., Makeig, S., Rao, B.D.: Compressed sensing for energy-efficient wireless telemonitoring of noninvasive fetal ecg via block sparse bayesian learning. IEEE Transactions on Biomedical Engineering **60**(2), 300–309 (2013)
51. Zhao, C., Ma, S., Zhang, J., Xiong, R., Gao, W.: Video compressive sensing reconstruction via reweighted residual sparsity. IEEE Transactions on Circuits and Systems for Video Technology **27**(6), 1182–1195 (2017)
52. Zhao, H., Wang, Y., Qiao, Z., Fu, B.: Solder joint imagery compressing and recovery based on compressive sensing. Soldering & Surface Mount Technology **26**(3), 129–138 (2014)
53. Zhao, H., Zhao, H., Chen, J., Chen, J., Xu, S., Xu, S., Wang, Y., Wang, Y., Qiao, Z., Qiao, Z.: Compressive sensing for noisy solder joint imagery based on convex optimization. Soldering & Surface Mount Technology **28**(2), 114–122 (2016)
54. Zhao, W., Lu, H., Wang, D.: Multisensor image fusion and enhancement in spectral total variation domain. IEEE Transactions on Multimedia **20**(4), 866–879 (2018)