Original scientific paper

Adaptive Thresholding for Sparse Image Reconstruction

Ivan Volaric and Victor Sucic

***Abstract* — The performance of the class of sparse reconstruction algorithms which is based on the iterative thresholding is highly dependent on a selection of the appropriate threshold value, controlling a trade-off between the algorithm execution time and the solution accuracy. This is why most of the state-of-the-art reconstruction algorithms employ some method of decreasing the threshold value as the solution converges toward the optimal one. To address this problem we propose a data-driven adaptive threshold selection method based on the fast intersection of confidence intervals (FICI) method, with which we have augmented the two-step iterative shrinkage thresholding (TwIST) algorithm. The performance of the proposed algorithm, denoted as the FICI-TwIST algorithm, has been evaluated on a problem of image reconstruction with the missing pixels, exploiting image sparsity in the discrete cosine transformation domain. The obtained results have shown competitive performance in comparison with a number of state-of-the-art sparse reconstruction algorithms, even outperforming them in some scenarios.**

***Keywords* — Compressive sensing, Fast intersection of confidence intervals (FICI) method, Image reconstruction, Iterative soft thresholding, Signal sparsity, Sparse reconstruction algorithm.**

I. Introduction

THE signal sparsity, a signal property known for a long time, has been mainly used for lossy multimedia file compression; e.g. in the widely used JPG, MP3, and MP4 standards [1]. The idea behind such compression algorithms is not to store the original data in the time or spatial domain, but rather in some transformation domain in which the original data has a sparse representation. This means that the signal energy is concentrated in a small number of samples, while most of the samples are close or equal to zero. In such a scenario, the lossy compression is achieved by storing only the most significant samples, discarding the rest.

In the last decade, the compressive sensing (CS) has gained interest of many researchers, especially after ground-breaking papers [2], [3]. The idea behind CS is similar to the previously described lossy compression with one key difference. Instead of recording a full data-set and then discarding low-energy samples in the transformation domain, we record only a portion of the data-set and reconstruct the missing samples. The reconstruction process involves solving the undetermined linear system of equations by formulating the unconstrained linear optimization problem emphasizing the signal sparsity in the selected transformation domain. In other words, the sparse reconstruction calculates the missing samples in a way which will produce the sparsest representation in the transformation domain.

In the real-life situations, the partial unavailability of samples can happen due to the various physical constraints or corrupted data, resulting in the wide-spread application of the CS based methods not only limited to multimedia applications [4]-[8], e.g. medicine [9], [10]; astronomy [11], [12]; wireless communication [13], [14]; radar [15], [16]; geoscience [17], [18]; etc.

In this paper we propose a sparse reconstruction algorithm, denoted as the FICI-TwIST. The proposed algorithm combines the two-step iterative shrinkage thresholding (TwIST) algorithm [7] with the fast intersection of confidence interval (FICI) method [19]. The FICI method optimizes the TwIST iterative step by providing a data-driven threshold value. This paper is the continuation of work presented in [5], with more detailed theoretical explanations and additional examples. Furthermore, the algorithm has been tweaked with a goal of reducing its execution time, allowing us to compare it with the other algorithms in a more realistic way. The proposed algorithm has been compared with a number of state-of-the-art reconstruction algorithms on a problem of image reconstruction with the missing pixels, by emphasizing the sparsity in the 2D discrete cosine transformation (DCT) domain.

The paper is organized as follows. Section II gives a theoretical background behind sparse image reconstruction and introduces the proposed reconstruction algorithm. Section III presents the simulation results, while Section IV gives the concluding remarks.

II. Sparse Image Reconstruction

A. Problem Formulation

Let $\mathbf{x}$ denote a grey-scale image with $N_x \times N_y$ pixels, and let $\mathbf{\Phi}$ denote an invertible linear transformation, such that:

$$\mathbf{X} = \mathbf{\Phi x \Phi}^{-1}. \tag{1}$$

As already mentioned, for a meaningful sparse

Paper received May 11, 2023; accepted June 28, 2023. Date of publication August 08, 2023. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ana Gavrovska.

This paper is revised and expanded version of the paper presented at the 30th Telecommunications Forum TELFOR 2022 [5].

This work has been fully supported by the University of Rijeka under the project number uniri-tehnic-18-67.

Corresponding author Ivan Volaric is with the Faculty of Engineering, University of Rijeka, Croatia (e-mail: ivolaric@riteh.hr).

Victor Sucic is with the Faculty of Engineering, University of Rijeka, Croatia (e-mail: vsucic@riteh.hr)

reconstruction, this transformation has to be sparsity inducing, that is, $\mathbf{X}$ has to be K-sparse where $K << N_xN_y$. It is well known that the DCT is such a transformation for images; the widely used JPG standard exploits DCT sparsity in order to achieve image compression [1]. Inspired by this, in this paper we will also use DCT, thus $\mathbf{\Phi}$ and $\mathbf{\Phi}^{-1}$ implement DCT and inverse DCT (IDCT) coefficients, respectively:

$$\mathbf{\Phi}_{k,n} = \begin{cases} \sqrt{\frac{1}{N}}, & k = 0, \\ \sqrt{\frac{2}{N}} \cos\left(\frac{\pi (2n+1) k}{2N} \right), & \text{otherwise}, \end{cases} \quad (2)$$

for k, $n \in [0, \ldots, N\text{-}1]$, where N is either N_x or N_y depending on if the $\mathbf{x}$ is multiplied from the left or the right side. Note that $\mathbf{\Phi}$ is real and orthonormal, and thus the inverse calculation is not necessary since $\mathbf{\Phi}^{-1} = \mathbf{\Phi}^{\mathrm{T}}$.

Furthermore, for more convenient future notation, we can rewrite (1), where both $\mathbf{x}$ and $\mathbf{X}$ are column vectors. Let $\mathbf{x}_V = \mathrm{vect}(\mathbf{x})$ and $\mathbf{X}_V = \mathrm{vect}(\mathbf{X})$, then:

$$\mathbf{X}_V = \mathbf{\Phi}_K \mathbf{x}_V, \quad (3)$$

where the $N_xN_y \times N_xN_y$ transformation matrix $\mathbf{\Phi}_K$ is defined through the Kronecker product as:

$$\mathbf{\Phi}_K = \mathbf{\Phi} \otimes \mathbf{\Phi} = \begin{bmatrix} \Phi_{1,1}\mathbf{\Phi} & \Phi_{1,2}\mathbf{\Phi} & \cdots & \Phi_{1,N_x}\mathbf{\Phi} \\ \Phi_{2,1}\mathbf{\Phi} & \Phi_{2,2}\mathbf{\Phi} & \cdots & \Phi_{1,N_x}\mathbf{\Phi} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{N_x,1}\mathbf{\Phi} & \Phi_{N_x,1}\mathbf{\Phi} & \cdots & \Phi_{N_x,N_x}\mathbf{\Phi} \end{bmatrix}. \quad (4)$$

Next, let $\mathbf{y}_V$ denote a column vector with a randomly selected set of $M << N_xN_y$ measurements taken from $\mathbf{x}_V$, that is, the available image pixels:

$$\mathbf{y}_V = \mathbf{\Psi} \mathbf{x}_V, \quad (5)$$

where $\mathbf{\Psi}$ is a $M \times N_xN_y$ measurement matrix with only one element per row equal to one and rest zeros, connecting a $\mathbf{y}_V$ m-th measurement with a $\mathbf{x}_V$ n-th pixel.

In order to formulize the sparse image reconstruction problem, we combine (3) and (5):

$$\mathbf{y}_V = \mathbf{\Psi} \mathbf{x}_V = \mathbf{\Psi} \mathbf{\Phi}_K^{-1} \mathbf{X}_V = \mathbf{A}^T \mathbf{X}_V, \quad (6)$$

where $\mathbf{A}^{\mathrm{T}} = \mathbf{\Psi}\mathbf{\Phi}_K^{-1} = \mathbf{\Psi}\mathbf{\Phi}_K^{\mathrm{T}}$ is truncated $\mathbf{\Phi}_K^{-1}$ matrix with deleted rows corresponding to the missing samples, representing the truncated backward transformation, that is, in our case the truncated IDCT. In future expressions, we will also use matrix $\mathbf{A} = \mathbf{\Phi}_K\mathbf{\Psi}^{\mathrm{T}}$ representing the expanded forward transformation (in our case DCT) where $\mathbf{y}_V$ is expanded in such way that the missing samples are set to zero, followed by the DCT.

The problem of sparse image reconstruction is to obtain full image $\mathbf{x}_V$ from a set of available image pixels $\mathbf{y}_V$. This problem is ill-posed since matrix $\mathbf{A}$ is not invertible, and as such could have an infinite number of solutions. In order to obtain the best solution, we formulize the following unconstrained optimization problem [6], [7]:

$$\widehat{\mathbf{X}}_V = \arg\min_{\mathbf{x}_V} \frac{1}{2} \left\| \mathbf{X}_V - \mathbf{A}\mathbf{y}_V \right\|_2^2 + \lambda c(\mathbf{X}_V), \quad (7)$$

where $c(\mathbf{X}_V) : \mathbb{R}^2 \rightarrow \mathbb{R}$ is the regularization function, while λ is the regularization parameter. The regularization function plays a most significant role in such optimization problems with a key goal of emphasizing the *a priori* known solution property. As stated before, we have selected $\mathbf{\Phi}$ such that $\mathbf{X}$ is sparse, thus for a meaningful reconstruction, the regularization function value should decrease as the solution sparsity increases. The best function for this problem is the ℓ_0-norm, as it counts the number of non-zero elements. However, the ℓ_0-norm minimization is an NP-hard problem, and it is usually solved with greedy algorithms, by searching for a good local minimum, instead of the global one [21]. Because of this, the ℓ_0-norm is often replaced with an easy to solve convex problem of ℓ_1-norm minimization [6], [7], [21], [22] introducing a new problem: the objective function does not measure the property which we want to attain. A more detailed survey of the sparse reconstruction algorithms is given in [23] and [24].

B. Problem solution with ℓ_1-norm minimization

By using the ℓ_1-norm as the regularization function, we can rewrite (7) as:

$$\mathbf{X}_V^{\ell_1} = \arg\min_{\mathbf{X}_V} \left\| \mathbf{X}_V \right\|_1, \text{ s. t.: } \left\| \mathbf{X}_V - \mathbf{A}\mathbf{y}_V \right\|_2^2 \leq \varepsilon. \quad (8)$$

Furthermore, by introducing the Moreau proximity operator we can further simplify this expression into closed form which can be iteratively solved (more detailed discussion about the proximity operator can be found in [25]):

$$\mathbf{X}_V^{\ell_1} = \mathrm{soft}_\lambda \left\{ \mathbf{X}_V \right\}, \quad (9)$$

where $\mathrm{soft}_\lambda\{\mathbf{X}_V\}$ is the component-wise soft-thresholding operator, defined as:

$$\mathrm{soft}_\lambda \left\{ \mathbf{X}_V \right\} = \mathrm{sgn}(\mathbf{X}_V) \max\left(\left| \mathbf{X}_V \right| - \lambda, 0 \right). \quad (10)$$

One simple, yet effective algorithm designed around this idea is the TwIST algorithm [7]. The solution of the $(n+1)$-th algorithm iteration is defined as:

$$\begin{aligned} \left[\mathbf{X}_V^{\ell_1} \right]^{[n+1]} = (1-\alpha) \left[\mathbf{X}_V^{\ell_1} \right]^{[n-1]} + (\alpha - \beta) \left[\mathbf{X}_V^{\ell_1} \right]^{[n]} + \\ \beta\, \mathrm{soft}_\lambda \left\{ \left[\mathbf{X}_V^{\ell_1} \right]^{[n]} + \mathbf{A}\left(\mathbf{y}_V - \mathbf{A}^T \left[\mathbf{X}_V^{\ell_1} \right]^{[n]} \right) \right\}, \end{aligned} \quad (11)$$

where α and β are the user pre-defined TwIST relaxation parameters controlling the averaging weights between the current and the previous two solutions. The final solution is obtained by iterating (11) until the stopping criterion is satisfied, i.e. the relative change, ε, between the solution of two consecutive algorithm iteration is small enough, or the maximum number of iterations, N_{it}, has been reached.

C. FICI based adaptive thresholding

The class of sparse reconstruction algorithms which is based on the previously described iterative shrinkage (thresholding) suffers from the problem that their performance is highly dependent on the proper selection of the regularization parameter in (7), that is, the threshold value λ in (9). With a high λ value, the input data is going to be tresholded more strictly, resulting in the faster algorithm convergence rate, however, at the cost of reduced solution accuracy. On the other hand, a low λ value will result in the slower algorithm convergence rate with a higher accuracy. Because of this, most state-of-the-art reconstruction algorithms employ some method of decreasing the λ value. If the reconstruction starts with a high λ value, and decreases as the reconstruction is converging towards the optimal solution, we can achieve both benefits, that is, the faster convergence rate with higher accuracy.

With such motivation, we have introduced the FICI

method [19] to the reconstruction process, by calculating the λ value in every TwIST iteration. The original FICI method has been used in combination with the local polynomial approximation (LPA) for signal denoising in order to find a data-driven window size. Basically, the FICI method searches the vicinity of a specific signal sample for a region where all samples have a statistically similar amplitude values. In the sparse reconstruction process, we have used the FICI method to find a region of the lowest amplitude samples which are going to be thresholded in the sparse reconstruction process.

Let $\ddot{\mathbf{X}}_V$ denote the ascendingly sorted argument of the soft-thresholding operator in (11), that is:

$$\ddot{\mathbf{X}}_V = \text{sort}\left\{\left[\mathbf{X}_V^{\ell_1}\right]^{[n]} + \mathbf{A}\left(\mathbf{y}_V - \mathbf{A}^T\left[\mathbf{X}_V^{\ell_1}\right]^{[n]}\right)\right\}. \quad (12)$$

In the first step of the proposed algorithm, we soft-threshold the input data:

$$\dot{\mathbf{X}}_V = \text{soft}_{\lambda_F}\left\{\ddot{\mathbf{X}}_V\right\}, \quad (13)$$

with a relatively small threshold value $\lambda_F = \lambda_P \max(\ddot{\mathbf{X}}_V)$. In our simulations it has been shown that setting λ_P in range $[10^{-4}, 2\cdot10^{-4}]$ ensures both robustness to noise and faster execution time, without a significant impact on the final solution. Starting from the first non-zero element, $\dot{\mathbf{X}}_V(i_0)$, the upper and the lower bound of the confidence interval are calculated as [19]:

$$\begin{aligned} D_{u,l}(i_0+\Delta i) = &\text{mean}\left\{\dot{\mathbf{X}}_V(i_0),\ldots,\dot{\mathbf{X}}_V(i_0+\Delta i)\right\} \\ &\pm\Gamma\,\text{std}\left\{\dot{\mathbf{X}}_V(i_0),\ldots,\dot{\mathbf{X}}_V(i_0+\Delta i)\right\}, \end{aligned} \quad (14)$$

where Δi is the current window size, starting from 1 and incrementing in each FICI iteration, Γ is the ICI threshold parameter, while mean$\{\cdot\}$ and std$\{\cdot\}$ are respectively, the mean value and the standard deviation of samples within the current window.

In the next step, the relative amount of intersection between the confidence intervals is calculated [19]:

$$R(i_0+\Delta i) = \frac{D_{u_{\min}}(i_0+\Delta i) - D_{l_{\max}}(i_0+\Delta i)}{2\Gamma\,\text{std}\left\{\dot{\mathbf{X}}_V(i_0),\ldots,\dot{\mathbf{X}}_V(i_0+\Delta i)\right\}}, \quad (15)$$

where $D_{umin}(i_0 + \Delta i)$ and $D_{lmax}(i_0 + \Delta i)$ are defined as:

$$D_{u_{\min}}(i_0+\Delta i) = \min(D_u(i_0),\ldots,D_u(i_0+\Delta i)), \quad (16)$$

$$D_{l_{\max}}(i_0+\Delta i) = \max(D_l(i_0),\ldots,D_l(i_0+\Delta i)). \quad (17)$$

These steps, that is (14)-(17), are iterated by incrementing Δi until the stopping criterion is satisfied:

$$R(i_0+\Delta i) < R_C, \quad (18)$$

where $0 \le R_C \le 1$ is the FICI threshold parameter [19].

Let Δi^+ denote the first Δi for which (18) is satisfied. The samples $\{\dot{\mathbf{X}}_V(i_0),\ldots, \dot{\mathbf{X}}_V(i_0 + \Delta i^+)\}$ represent the first non-zero region of input data with the lowest amplitude. By setting a new starting point, that is $i_0 = \Delta i^+$, we repeat (14)-(18) N_{reg} times. The final Δi^+, obtained in such a way, represents the index of a $\dot{\mathbf{X}}_V$ sample with the highest amplitude among the first N_{reg} regions with the lowest amplitude. Thus, the amplitude of this sample is used as the TwIST threshold value, that is: $\lambda = \dot{\mathbf{X}}_V(i_0 + \Delta i^+)$.

III. Experimental results

The performance of the proposed algorithm has been evaluated on three standard gray-scale test images: lenna, barbara, and cameraman; all pre-scaled to 256 × 256 pixels. Inspired by the JPG standard, images have been divided into 8 × 8 blocks, where each block has been processed separately. The comparison has been performed in terms of MSE and algorithm execution time with the following state-of-the-art sparse reconstruction algorithms: TwIST [7], split augmented Lagrangian shrinkage algorithm (SALSA) [6], Nesterov algorithm (NESTA) [22], and your augmented Lagrangian algorithm for ℓ1 (YALL1) [21]. Note that the execution times should only be taken in relative comparison, and not as absolute values.

The simulations have been performed over a range of CS ratios: $M/N_xN_y = \{0.1, 0.15, 0.2, \ldots, 0.9\}$, while the stopping criterion has been set to: $\varepsilon = 10^{-5}$ and $N_{it} = 1000$. For each CS ratio, the results have been averaged over $N_{CS} = 50$ different realizations of the randomly generated measuring matrix. The algorithm and image specific parameters are listed in Table 1. The algorithm parameters have been selected for the best performance with the CS ratio of 0.4. It is important to point out that the MSE function has a large number of local minima and our attempts of utilizing the heuristic optimization algorithms (such as the particle swarm optimization) have not resulted in the MSE value which is better than the manually selected parameters using the gradient descent idea with the intuitively selected starting point.

The obtained reconstructed MSE values are shown in Fig. 1, while the algorithm execution times are presented in Fig. 2. In terms of the MSE values, it can be seen that all considered algorithms run very competitively, with SALSA being the worst, especially for lower M/N_xN_y ratios. The proposed algorithm in some cases slightly outperforms other considered algorithms, e.g., in range $M/N_xN_y \in [0.4, 0.5]$ for lenna and barbara. The execution times are relatively consistent over the entire CS range, with TwIST being the fastest, and the proposed FICI-TwIST being the slowest algorithm. This was to be expected, since the TwIST algorithm is a relatively simple reconstruction algorithm, while the proposed algorithm requires calculation of the mean value and the standard deviation over the increasing window size (Δi) in each iteration.

Table 1: Algorithm and Image Specific Parameters

	lenna $\lambda = 1.5$	***barbara*** $\lambda = 1.5$	***cameraman*** $\lambda = 1.5$
TwIST	$\alpha = 1.97$ $\beta = 3.94$		
SALSA	$\mu_S = 1$		
NESTA	$\mu_N = 8$	$\mu_N = 8$	$\mu_N = 12$
YALL1	*No algorithm specific parameters*		
FICI-TwIST	$\Gamma = 1.1$ $R_c = 0$ $\alpha = 1$ $\beta = 1.7$ $N_{reg} = 3$ $\lambda_P = 1.2\cdot10^{-4}$	$\Gamma = 1.3$ $R_c = 0$ $\alpha = 1$ $\beta = 1.75$ $N_{reg} = 3$ $\lambda_P = 1.1\cdot10^{-4}$	$\Gamma = 1.1$ $R_c = 0$ $\alpha = 1$ $\beta = 1.75$ $N_{reg} = 3$ $\lambda_P = 2.0\cdot10^{-4}$

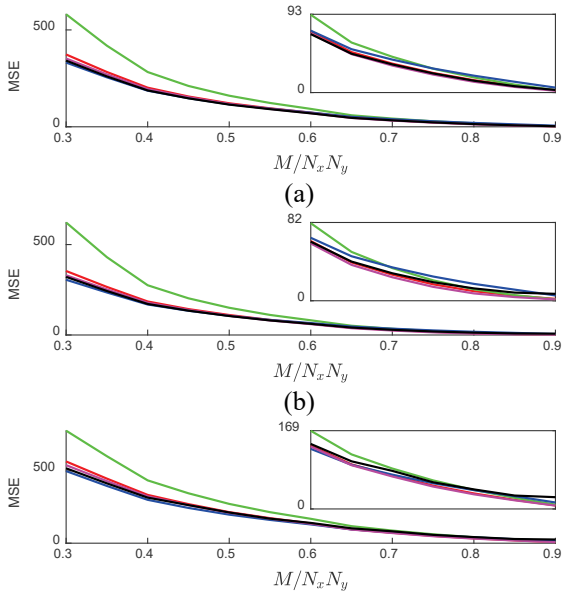

Fig. 1. Average reconstructed MSE values for TwiST (red), SALSA (green), NESTA (blue), YALL1 (magenta), and FICI-TwIST (black) over M/N_xN_y range and for: (a) lenna; (b) barbara; (c) cameraman. Range of the zoom inset is $M \in [0.6, 0.9]$.

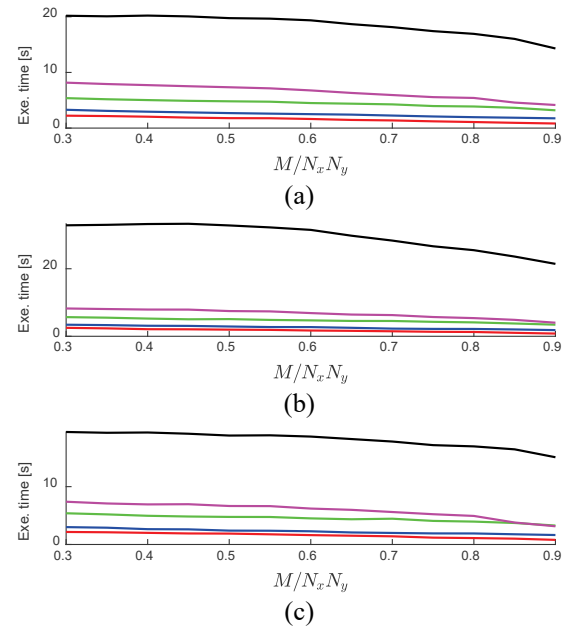

Fig. 2. Average algorithm execution times for TwiST (red), SALSA (green), NESTA (blue), YALL1 (magenta), and FICI-TwIST (black) over M/N_xN_y range and for: (a) lenna; (b) barbara; (c) cameraman.

The M/N_xN_y ratio of 0.4 in our simulations have shown to be borderline, with lower ratio resulting in the visually significantly worse reconstruction performance, while on the other hand a higher CS ratio results in a visually undistinguishable images. This is why we have highlighted the reconstruction performance for this specific CS ratio in Fig. 3 and 4, showing algorithm convergence rate (in terms of MSE) and cumulative execution time over iterations of the specific algorithm. As in previous figures, each data point has been averaged over $N_{CS} = 50$ different realizations of the randomly generated measuring matrix.

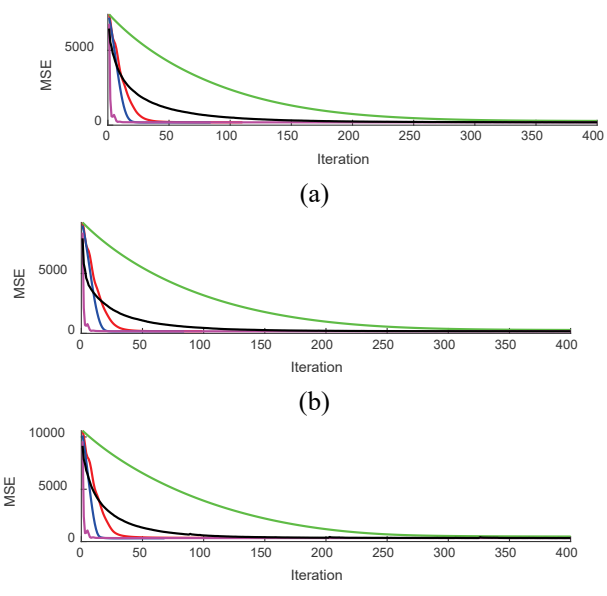

Fig. 3. Average reconstructed MSE values ($M/N_xN_y = 0.4$) for TwiST (red), SALSA (green), NESTA (blue), YALL1 (magenta), and FICI-TwIST (black) over algorithm iterations and for: (a) lenna; (b) barbara; (c) cameraman.

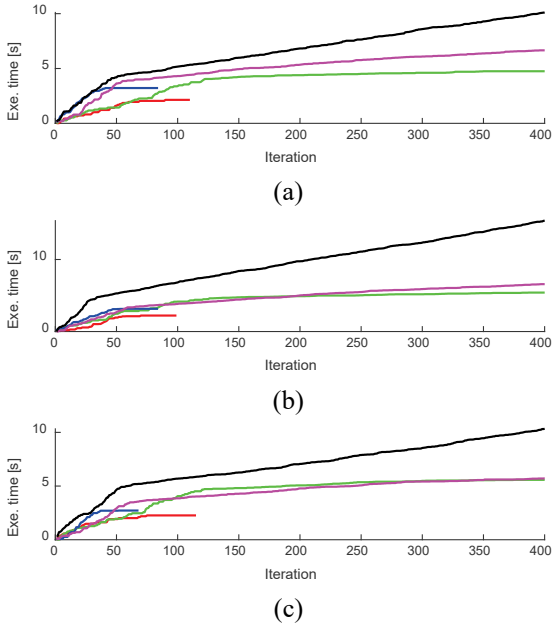

Fig. 4. Average cumulative algorithm execution times ($M/N_xN_y = 0.4$) for TwiST (red), SALSA (green), NESTA (blue), YALL1 (magenta), and FICI-TwIST (black) over algorithm iterations and for: (a) lenna; (b) barbara; (c) cameraman.

As in the previous case, we can see that all considered algorithms run competitively; with SALSA having the slowest, and YALL1 having the fastest convergence rate. In the starting iterations, the proposed algorithm runs similar to the YALL1, having one of the best MSE improvements per iteration. However, in the later algorithm iterations, convergence rate slows down, and finally MSE value settles between the 100th and the 150th algorithm iteration. This behavior reveals a shortage in the proposed algorithm (alongside SALSA and YALL1) exit criterion. All of the

mentioned algorithms, that is, all except TwIST and NESTA, exit because they have reached the maximum allowed number of iterations, while MSE stays relatively unchanged in the latter iterations. Defining a better exit criterion would greatly help in the improvement of the mentioned algorithm execution times. However, not having access to the original image in the real-life implementation significantly limits the possibilities for such criterion.

A single random realization of the measuring matrix with $M/N_xN_y = 0.4$, and the resulting reconstructed images have been presented in Figs. 5- 7. As previously stated, the performance of the considered reconstruction algorithms is very similar, which can be confirmed by visual inspection of the shown images.

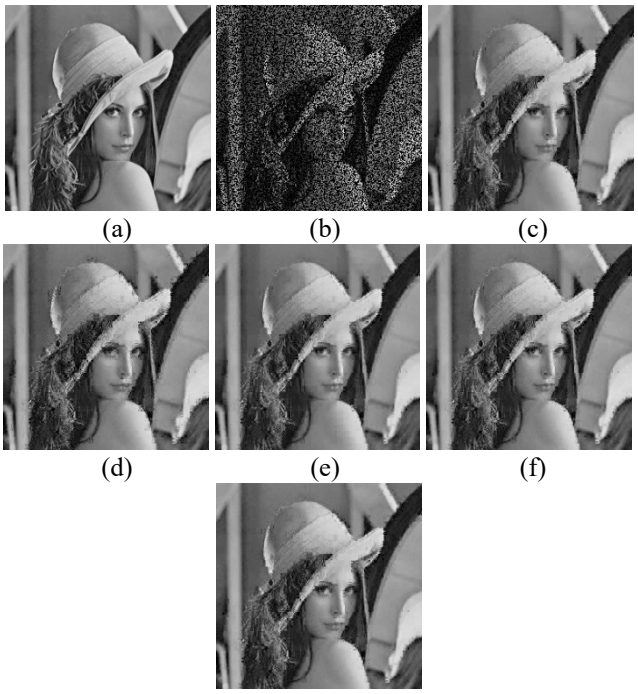

Fig. 5. Reconstruction performance for lenna:
(a) original image; (b) CS-ed image ($M/N_xN_y = 0.4$);
(c) reconstructed by the TwIST (MSE = 191.46);
(d) reconstructed by the SALSA (MSE = 273.30);
(e) reconstructed by the NESTA (MSE = 181.81);
(f) reconstructed by the YALL1 (MSE = 189.69);
(g) reconstructed by the FICI-TwIST (MSE = 176.99).

IV. Conclusion

Compressive sensing is a powerful framework, being capable of signal recovery when it is heavily under-sampled below the Nyquist-Shannon frequency. This is only possible with an *a priori* knowledge about some transformation domain in which the original signal has a sparse representation. In this paper, we have proposed a sparse reconstruction algorithm, denoted as the FICI-TwIST algorithm, in which we have addressed the problem of proper selection of the threshold value, by utilizing the FICI method for adaptive data-driven threshold calculation.

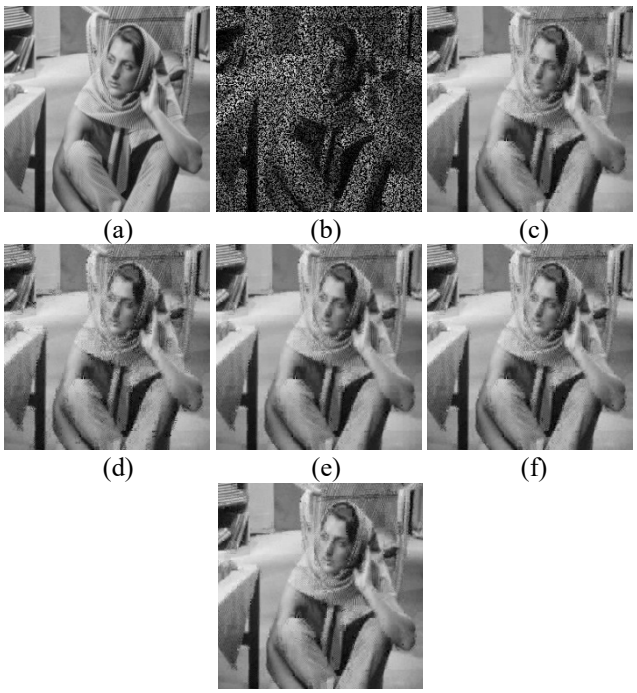

Fig. 6. Reconstruction performance for barbara:
(a) original image; (b) CS-ed image ($M/N_xN_y = 0.4$);
(c) reconstructed by the TwIST (MSE = 179.56);
(d) reconstructed by the SALSA (MSE = 272.84);
(e) reconstructed by the NESTA (MSE = 166.00);
(f) reconstructed by the YALL1 (MSE = 172.70);
(g) reconstructed by the FICI-TwIST (MSE = 165.14)

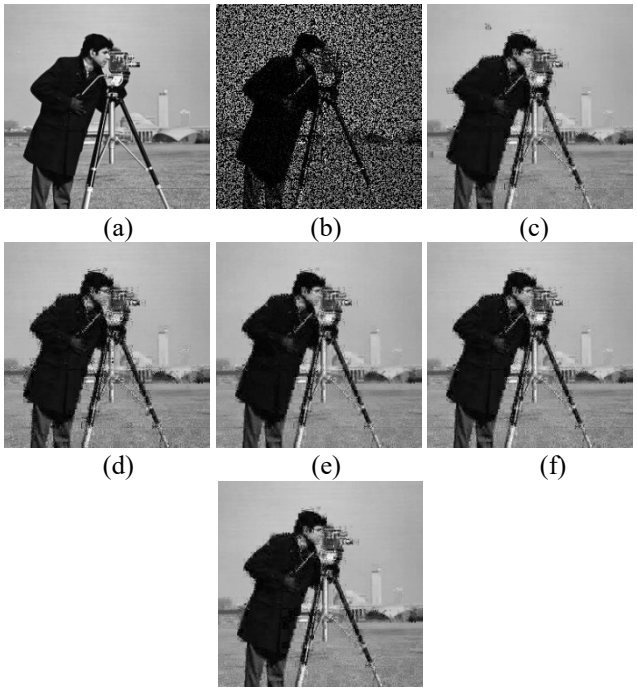

Fig. 7. Reconstruction performance for cameraman:
(a) original image; (b) CS-ed image ($M/N_xN_y = 0.4$);
(c) reconstructed by the TwIST (MSE = 358.64);
(d) reconstructed by the SALSA (MSE = 466.67);
(e) reconstructed by the NESTA (MSE = 312.38);
(f) reconstructed by the YALL1 (MSE = 334.70);
(g) reconstructed by the FICI-TwIST (MSE = 313.44)

We have tested the proposed algorithm on a problem of image reconstruction with the missing pixels and compared its reconstruction performance with those of state-of-the-art sparse reconstruction algorithms in terms of MSE and execution time. The experimental results have shown that all considered algorithms can relatively accurately reconstruct an image where up to 60% of pixels are missing. The proposed algorithm runs very competitively, even outperforming considered state-of-the-art algorithms in some scenarios in terms of MSE value. On the other hand, the shortcoming of the proposed algorithm is its execution time, as it requires the repetitive calculation of the mean value and the standard deviation. However, as shown by the simulations, the MSE improvement in latter algorithm iterations is not significant, so by formulating the alternative exit criterion, this problem can be alleviated. This will be a topic of our future research, alongside with the exploration of different sparsity inducing transformation domains, namely the wavelet transformation using the Cohen-Daubechies-Feauveau 9/7 wavelet, as used by the JPEG2000 standard.

REFERENCES

[1] S. Stanković, I. Orović, and E. Sejdić, *Multimedia Signals and Systems Basic and Advanced Algorithms for Signal Processing 2nd Ed.*. New York: Springer, 2016.

[2] D. L. Donoho, "Compressed Sensing", *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289-1306, April 2006.

[3] E. J. Candes, J. Romberg, and T Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information", *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489-509, Feb 2006.

[4] I. Stanković, I. Orović, M. Daković, and S. Stanković, "Denoising of Sparse Images in Impulsive Disturbance Environment", *Multimedia Tools and Applications*, vol. 77, no. 5, pp. 5885-5905, March 2018.

[5] I. Volaric, and V. Sucic, "Sparse Image Reconstruction via Fast ICI Based Adaptive Thresholding", in *2022 30th Telecommunications Forum (TELFOR)*, pp. 1-4, Nov 2022.

[6] M. Afonso, J. M. Bioucas-Dias, and M. A. Figueiredo, "Fast Image Recovery Using Variable Splitting and Constrained Optimization", *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2345-2356, Sept 2010.

[7] J. M. Bioucas-Dias and M. A. Figueiredo, "A New TwIST: Two-Step Iterative Shrinkage/Thresholding Algorithms for Image Restoration", *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2992-3004, Nov 2007.

[8] X. Wang, and Y. Su, "Image Encryption Based on Compressed Sensing and DNA Encoding", *Signal Processing: Image Communication*, vol. 95, pp. 116246, July 2021.

[9] G. Da Poian, R. Bernardini, and R. Rinaldo, "Separation and Analysis of Fetal-ECG Signals From Compressed Sensed Abdominal ECG Recordings", *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1269-1279, June 2016.

[10] S. G. Lingala, and M. Jacob, "Blind Compressive Sensing Dynamic MRI", *IEEE Transactions on Medical Imaging*, vol. 32, no. 6, pp. 1132-1145, June 2013.

[11] M. C. M. Cheung, *et. al*, "Multi-component Decomposition of Astronomical Spectra by Compressed Sensing", *The Astrophysical Journal*, vol. 882, no. 1, pp. 1-13, Aug 2019.

[12] P. Addesso, M. Longo, S. Marano, V. Matta, I. M. Pinto, and M. Principe, "Sparsifying Time-Frequency Distribution for Gravitational Wave Data Analysis", in *2015 3rd International Workshop on Compressed Sensing Theory and its Application to Radar, Sonar and Remote Sensing (CoSeRa)*, pp. 154-158, Nov 2015.

[13] J. E. Barcelo-Llado, A. Morell, and G. Seco-Granados, "Amplify-and-Forward Compressed Sensing as an Energy-Efficient Solution in Wireless Sensor networks", *IEEE Sensors Journal*, vol. 14, no. 5, pp. 1710-1719, May 2019.

[14] A. Draganić, I. Orović, and S. Stanković, "Blind Signals Separation in Wireless Communications based on Compressive sensing", in *2014 22nd Telecommunications Forum (TELFOR)*, pp. 561-564, Nov 2014.

[15] L. Stanković, "ISAR Image Analysis and Recovery with Unavailable or Heavily Corrupted Data", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2093-2106, July 2015.

[16] J. Akhtar, and K. E. Olsen, "Formation of Range-Doppler Maps Based on Sparse Reconstruction", *IEEE Sensors Journal*, vol. 16, no. 15, pp. 5921-5926, Aug 2016.

[17] Y. Zhou, J. Gao, W. Chen, and P. Frossard, "Seismic Simultaneous Source Separation via Patchwise Sparse Representation", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 9, pp. 5271-5284, Sept 2016.

[18] D. Kumlu, and I. Erer, "Improved Clutter Removal in GPR by Robust Nonnegative Matrix Factorization", *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 6, pp. 958-962, June 2020.

[19] I. Volaric, J. Lerga, and V. Sucic, "A Fast Signal Denoising Algorithm Based on the LPA-ICI Method for Real-time Applications", *Circuits, Systems, and Signal Processing*, vol. 36, no. 11, pp. 4653-4669, Nov 2017.

[20] T. Blumensath, and M. E. Davies, "Normalized Iterative Hard Thresholding: Guaranteed Stability and Performance", *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 298-309, April 2010.

[21] Y. Zhang, "Users Guide for YALL1: Your Algorithms for l1 Optimization", *Technique report*, pp. 9-17, 2009.

[22] S. Becker, J. Bobin, and E. J. Candes, "NESTA; A Fast and Accurate First-Order Method for Sparse Recovery", *SIAM Journal of Imaging Sciences*, vol. 4, no. 1, pp. 1-39, Jan 2011.

[23] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A Survey of Sparse Representation: Algorithms and Applications", *IEEE Access*, vol. 3, pp- 490-530, May 2015.

[24] E. C. Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, "A Review of Sparse Recovery Algorithms", *IEEE Access*, vol. 7, pp. 1300-1322, Dec 2018.

[25] P. L. Combettes, and V. R. Ways, "Signal Recovery by Proximal Forward-Backward Splitting", *Multiscale Modeling & Simulations*, vol. 4, no. 4, pp. 1168 – 1200, 2005.