A Novel Deterministic Sensing Matrix Based On Kasami Codes For Cluster Structured Sparse Signals

Hamid Nouasria, Mohamed Et-tolba
Department of Communication Systems
INPT, Rabat, Morocco
nouasria@inpt.ac.ma, ettolba@inpt.ac.ma

***Abstract*—Cluster structured compressive sensing is a new direction of compressive sensing, dealing with cluster structured sparse signals. In this paper, we propose a sensing matrix based on Kasami codes for CSS signals. The Kasami codes have been the subject of several constructions. Our idea is to make these constructions suitable to CSS signals. The proposed matrix, gives more intention to the clusters. Simulation results show the superior performance of our matrix. In that, it gives the highest rate of exact recovery. Moreover, the deterministic aspect of our matrix makes it more suitable to be implemented on hardware.**

***Index Terms*—Compressive sensing, Cluster structured compressive sensing, Sensing matrix, Orthogonality property.**

I. INTRODUCTION

Compressive sensing (CS) [1] is a new sampling framework dealing with sparse signals. As a sensing step, CS captures a low-dimensional measurement vector by correlating the sparse signal with a sensing matrix. In the literature, the sensing matrices used in CS can be divided into two categories, random and deterministic. The Gaussian [2] and Bernoulli [3] are the most familiar random matrices; the random aspect of such a matrix makes it very difficult in terms of hardware implementation. As an alternative, deterministic matrices are very easy to be implemented on hardware. At the recovery step, given that the dimension of the measurement vector is far less than the dimension of the original signal, so the problem of recovery (i.e. recover the original signal from its corresponding measurement vector) is an underdetermined problem which is impossible to solve. Fortunately, using the sparsity, CS provides several algorithms to recover the original sparse signal. For example, orthogonal matching pursuit (OMP) [4] and l_1-magic [5] are the most popular algorithms.

Cluster structured compressive sensing (CSCS) is a new direction of CS, dealing with cluster structured sparse (CSS) signals. The existing algorithms dealing with CSS signals can be categorized into three categories: Block greedy algorithms [6,7]; this category requires prior knowledge of the location and the size of every cluster, which is not always practical. Fortunately, dynamic programming algorithms [8] require only the number of clusters. Finally, blind recovery algorithms [9-12] do not require any prior knowledge. One can realize that blind recovery algorithms are the most interesting algorithms. But their drawback is that they are computationally demanding. Differently from the existing works, we try to open a new research direction by proposing efficient sensing matrices for CSS signals. Our first work has been published in [13], where a matrix has been proposed for CSS signals. In this paper, similar to the previous work, we propose a new sensing matrix based on Kasami codes for CSS signals. Our idea is to choose some good Kasami codes as columns of the proposed matrix. Moreover, we give more importance to the columns corresponding to the clusters. In terms of the required parameters, our method requires only the approximate location of each cluster. Note that, at the recovery step, we use only a basic CS algorithm in order to keep the same basic CS complexity. So we improve the quality without increasing the complexity. Extensive experiments and comparisons with the best state-of-the-art sensing matrices show that our matrix is very powerful in the case of CSS signals.

The remainder of this paper is organized as follows: In section II, we give a summary of CS and the derivation CSCS. In section III, we give a detailed description of the proposed matrix. Simulation results are given in section IV. We conclude the paper in section V.

II. CS THEORY

CS is a new framework representing a good solution for sensing and recovery of sparse signals. A signal, is said sparse, if it has a concise representation either in its original domain or in a transform domain $\boldsymbol{\Psi}$ (i.e. $\boldsymbol{x}$=$\boldsymbol{\Psi\alpha}$). Then we can define the set $\triangle_S$ of S-sparse signals as: $\triangle_S$=$\{\boldsymbol{x}{\in}R^N,\|\boldsymbol{x}\|_{l_0}{\leq}S\ \ or\ \ \|\boldsymbol{\alpha}\|_{l_0}{\leq}S\}$, where $\|\boldsymbol{b}\|_{l_0}$ is the pseudo-norm l_0, which gives the number of non-zero coefficients in $\boldsymbol{b}$. For more clarity, let's consider an N-dimensional S-sparse signal $\boldsymbol{x}$. CS proposes to acquire this signal by correlating it with an $M{\times}N$ sensing matrix $\boldsymbol{\Phi}$, with $S{<}M{\ll}N$:

$$\boldsymbol{y} = \boldsymbol{\Phi x}. \tag{1}$$

It has been proven that, by exploiting the sparsity of $\boldsymbol{x}$, the problem of inverting (1) (i.e. given $\boldsymbol{y}$ and $\boldsymbol{\Phi}$, recover the signal $\boldsymbol{x}$) is possible provided that the sensing matrix $\boldsymbol{\Phi}$ verifies some properties. One of the most popular properties is the restricted isometry property (RIP) defined in the literature as:

Definition 1 (RIP): for each sparsity level S=1,2,3,..., we define the constant δ_S of a given matrix $\mathbf{\Phi}$ as the smallest number that verifies:

$$(1-\delta_S)\|\boldsymbol{x}\|_{l_2}^2 \leq \|\mathbf{\Phi}\boldsymbol{x}\|_{l_2}^2 \leq (1+\delta_S)\|\boldsymbol{x}\|_{l_2}^2, \quad (2)$$

for all S-sparse signals $\boldsymbol{x}$. We say that $\mathbf{\Phi}$ verifies the RIP of order S if: $0<\delta_S\ll 1$. Note that there are other properties (e.g. mutual coherence property, ...) that can be used instead of RIP.

Considering that the signal of interest $\boldsymbol{x}$ is sparse in its original domain, the CS formulation for the problem of recovering a S-sparse N-dimensional signal from M linear measurements $\boldsymbol{y}$ is the following:

$$\hat{\boldsymbol{x}} = \text{arg min } F(\boldsymbol{x}) \quad s.t \quad \boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x}, \quad (3)$$

where $F(\boldsymbol{x})$ is a sparsity inducing function which allows to select, among the infinitely many solutions the desired one. Provided that the desired solution is known a priori to be sparse, so the most ultimate choice for $F(\boldsymbol{x})$ is $F(\boldsymbol{x})$=$\|\boldsymbol{x}\|_{l_0}$. Then this choice leads to the P_0 optimization problem:

$$(P_0)\ \hat{\boldsymbol{x}} = \text{arg min } \| \boldsymbol{x} \|_{l_0} \quad s.t \quad \boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x}, \quad (4)$$

the problem P_0 is known to be numerically prohibitive. Another choice for $F(\boldsymbol{x})$ is $F(\boldsymbol{x})$=$\|\boldsymbol{x}\|_{l_1}$, where $\|\boldsymbol{x}\|_{l_1}$ is the norm l_1 of $\boldsymbol{x}$, that gives the sum of the absolute values of coefficients in $\boldsymbol{x}$. This problem is called P_1 problem:

$$(P_1)\ \hat{\boldsymbol{x}} = \text{arg min } \| \boldsymbol{x} \|_{l_1} \quad s.t \quad \boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x}. \quad (5)$$

In the case of noisy measurements, $\boldsymbol{y}$=$\mathbf{\Phi}\boldsymbol{x}$+$\boldsymbol{e}$, where $\boldsymbol{e}$ is a noise term with $\|\boldsymbol{e}\|_{l_2}\leq\epsilon$, P_1 is reformulated as:

$$(P_2)\ \hat{\boldsymbol{x}} = \text{arg min } \| \boldsymbol{x} \|_{l_1} \quad s.t \quad \| \boldsymbol{y} - \mathbf{\Phi}\boldsymbol{x} \|_{l_2} \leq \epsilon, \quad (6)$$

also an unconstrained optimization problem P_3 is considered:

$$(P_3)\ \hat{\boldsymbol{x}} = \text{arg min}(\frac{1}{2} \| \boldsymbol{y} - \mathbf{\Phi}\boldsymbol{x} \|_{l_2} + \lambda \| \boldsymbol{x} \|_{l_1}), \quad (7)$$

with $\|.\|_{l_2}$ is the euclidean norm and λ is a predefined constant.

It is worth noting that if the signal is S-sparse in an appropriate transform domain $\mathbf{\Psi}$ (i.e. $\boldsymbol{x}$=$\mathbf{\Psi}\boldsymbol{\alpha}$). In this case, we acquire the signal as in (1). But in the recovery step, we search for $\boldsymbol{\alpha}$. For that, the problem P_0 is reformulated as:

$$(P_0)\ \hat{\boldsymbol{\alpha}} = \text{arg min } \| \boldsymbol{\alpha} \|_{l_0} \quad s.t \quad \boldsymbol{y} = \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{\alpha}, \quad (8)$$

the same thing for problems P_1, P_2 and P_3.

While CS provides a significant improvement in the case of sparse signals, CSCS is a new direction dealing with CSS signals. CSCS makes use of suitable sparsity inducing functions $F(\boldsymbol{x})$ that induce the sparsity of the solution with favoring certain configurations of the support of non-zero coefficients and discourages others. Several works have been devoted to integrating the clustered structure into CS. All the CSCS algorithms have proposed a good solution for CSS signals, but have a one drawback is that they are computationally demanding. This is why we are interested to change the way of acting and focus on the sensing step rather than focusing on the recovery step; in other words, our idea is to propose new appropriate sensing matrices for efficient CSS signals sensing. In [13], we have proposed a new matrix suitable to acquire CSS signals using Hadamard codes; based on same idea, we propose in the following section a new sensing matrix based on Kasami codes.

III. Proposed sensing matrix

A. Large set of Kasami codes

The large set of Kasami codes [14] contains N_c=$2^{\frac{n}{2}}(2^n$+$1)$ codes each of length M_c=2^n-1 as illustrated in TABLE I, where $\boldsymbol{u}$ is an m-sequence of length M_c generated by a primitive polynomial of degree n (i.e. n must be even and $mod(n,4)$=2), $\boldsymbol{v}$ is a sequence formed by decimating the sequence $\boldsymbol{u}$ by $2^{\frac{n}{2}+1}$+1, and $\boldsymbol{w}$ is the sequence obtained by decimating $\boldsymbol{u}$ by $2^{\frac{n}{2}}$+1. T denotes the cyclical left shift operator and $\oplus$ denotes the modulo-2 operator. In the sequel, we represent the binary codes using +1's and -1's; the appropriate mapping is that the zeros are mapped to +1's and the ones are mapped to -1's. Before presenting our sensing matrix, called block near orthogonal (BNO) Kasami matrix, we describe the construction of an intermediate matrix called near orthogonal (NO) Kasami matrix.

TABLE I: Large set of Kasami codes

$\boldsymbol{u}$	
$\boldsymbol{v}$	
$\boldsymbol{u}\oplus T^k\boldsymbol{v}$	$k=0,...,2^n-2$
$\boldsymbol{u}\oplus T^m\boldsymbol{w}$	$m=0,...,2^{\frac{n}{2}}-2$
$\boldsymbol{v}\oplus T^m\boldsymbol{w}$	$m=0,...,2^{\frac{n}{2}}-2$
$\boldsymbol{u}\oplus T^k\boldsymbol{v}\oplus T^m\boldsymbol{w}$	$k=0,...,2^n-2$ and $m=0,...,2^{\frac{n}{2}}-2$

B. Near orthogonal Kasami matrix

The sensing matrix with columns chosen at random from the large set of Kasami codes is called hereafter, Random Kasami matrix. NO Kasami matrix is based on the idea of selecting the codes that have good orthogonality rather than choosing them randomly. This selective technique is called hereafter, pruning technique. Moreover, we further improve the orthogonality of these codes by using an extra technique called padding technique.

1) pruning technique: The pruning technique consists of choosing, from the large set of Kasami codes, some specific codes, which have a good orthogonality. To simplify the problem, let's view the large set of Kasami codes as an $M_c\times N_c$ matrix, where each column represents a Kasami code from the large set. The purpose is to construct an $M_c\times N$ matrix; consequently, the conventional method is randomly eliminating N_e=N_c-N codes. NO Kasami matrix propose to eliminate N_e codes in a way to improve the orthogonality. The inner-product between the different codes takes values in $\{-t(n)$,$-s(n)$,-1,$s(n)$-2,$t(n)$-2$\}$, where $-t(n)$=-2$\times s(n)$+1=-(1+$2^{\frac{n+2}{2}}$). The idea is to eliminate the codes that give the maximum value $-t(n)$. The idea is illustrated in Fig. 1.

2) padding technique: One can notice that many inner-product values of Kasami codes are -1. So by padding $+1$ to the Kasami codes, it is possible to make these inner-product values to 0. This technique will more enhance the columns orthogonality.

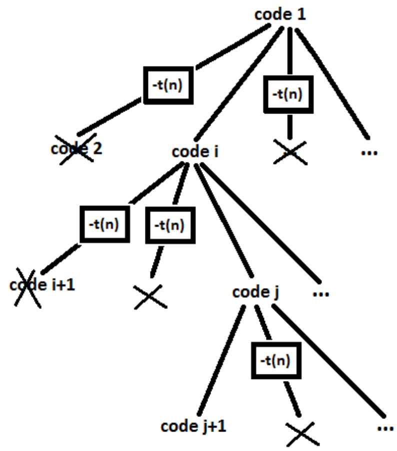

Fig. 1: First: we extend (**code 1**) by calculating its inner-product with codes of indices from 2 to N_c, and then eliminate the codes that give the maximum value $-t(n)$. Second: we cross the tree from left to right, by taking the first no eliminated code (**code i**) and calculate its inner-product with codes of indices from i+1 to N_c without considering the codes that have been already eliminated. And then eliminate the codes that give maximum inner-product. Third: similarly the (**code j**) is extended and the process is repeated until we reach the desired number of eliminated codes N_e.

3) combining the pruning and padding techniques: Here we combine the two previous mentioned techniques to construct an $(M_c$+1)$\times N$ sensing matrix. First, we apply the pruning technique to an $M_c \times N_c$ matrix to construct an $M_c \times N$ matrix. Second, we add $+1$ to each column to form an $(M_c$+1)$\times N$ sensing matrix.

C. Block near orthogonal Kasami matrix

NO Kasami matrix has been the subject of another paper [15], where we have demonstrated its power in the case of sparse signals. In this paper, we extend this matrix to make it suitable to CSS signals. The construction of the proposed matrix, called BNO Kasami matrix, gives more importance to the columns corresponding to the clusters during the sensing step. All along the construction of the NO Kasami matrix (i.e. Fig. 1), we consider the set containing the codes from **code 1** to the last extended code. The inner-product between the elements of this set take values in $\{$-$s(n)$,-1,$s(n)$-2,$t(n)$-2$\}$. So we conclude that these codes have a good orthogonality compared to the rest that take values in $\{$-$t(n)$,-$s(n)$,-1,$s(n)$-2,$t(n)$-2$\}$ which includes the maximum value $-t(n)$. These codes are always located at the beginning of the NO Kasami matrix, and their number N_{ex} is related to N_e. Using a NO Kasami matrix $\boldsymbol{\Phi}_{M\times N}$, and a G-length set $\boldsymbol{L}$ of clusters locations (G is the number of clusters in the signal), we construct our new matrix by following the next steps: 1) form a set that contains NO Kasami matrix columns of indices ranging from 1 to N_{ex}. 2) divide this set into G blocks; all the G blocks must have the same number of columns as possible. 3) form a new $M\times N$ sensing matrix by putting the G blocks in such a way that the location of i^{th} block into the matrix is given by the location p_i of i^{th} cluster into the signal, and fulfill the matrix by NO Kasami matrix columns ranging from N_{ex}+1 to N.

1) general sensing case: Finally, we note that in order to take benefit from the columns structure, the proposed sensing matrix must be correlated directly with the sparse representation $\boldsymbol{\alpha}$ instead of $\boldsymbol{x}$ (i.e. in the case of sparsity in a transform domain). For that, we have to multiply our sensing matrix by $\boldsymbol{\Psi}^{-1}$. To be clear, let's consider a BNO Kasami matrix $\boldsymbol{\Phi}$ and a signal $\boldsymbol{x}$ that has a sparse representation $\boldsymbol{\alpha}$ in a transform domain $\boldsymbol{\Psi}$ (i.e. $\boldsymbol{x}$=$\boldsymbol{\Psi\alpha}$). For that we acquire the signal $\boldsymbol{x}$ by the the matrix $\boldsymbol{\Phi}'$=$\boldsymbol{\Phi\Psi}^{-1}$ as follows:

$$\boldsymbol{y} = \boldsymbol{\Phi}'\boldsymbol{x} = \boldsymbol{\Phi\Psi}^{-1}\boldsymbol{\Psi\alpha} = \boldsymbol{\Phi\alpha}. \tag{9}$$

An important thing to note, is that $\boldsymbol{\Psi}$ is often an orthonormal matrix; the fact that reduces the complexity of its inversion since $\boldsymbol{\Psi}^{-1}$=$\boldsymbol{\Psi}^t$, where t in the transpose operator.

IV. Simulation results

In this section, we compare the proposed matrix with Random Kasami matrix, NO Kasami matrix, bBH matrix, Bernoulli matrix (its entries take values $+\frac{1}{\sqrt{M}}$ or $-\frac{1}{\sqrt{M}}$ with equal probability), and Gaussian matrix (its entries have zero mean and variance equal to $\frac{1}{M}$). Also the Kasami matrices are normalized after construction (i.e. multiplied by $\frac{1}{\sqrt{M}}$). Comparison is given in terms of the rate of the exact recovery (RER), which is defined with considering that the recovery is exact if the signal to noise ratio (SNR) is greater than 50 dB. The SNR is defined as: SNR=$20\log_{10}\frac{\|\boldsymbol{x}\|_{l_2}}{\|\boldsymbol{x}-\hat{\boldsymbol{x}}\|_{l_2}}$, where $\boldsymbol{x}$ is the original signal and $\hat{\boldsymbol{x}}$ is the approximate signal. Over all simulations, we fix the number of clusters as G=8. Each cluster has a random size. The clusters locations are fixed as $\boldsymbol{L}$=$\{\boldsymbol{L}_1$=15,$\boldsymbol{L}_2$=30,$\boldsymbol{L}_3$=45, $\boldsymbol{L}_4$=60,$\boldsymbol{L}_5$=75,$\boldsymbol{L}_6$=90,$\boldsymbol{L}_7$=105,$\boldsymbol{L}_8$=116$\}$. Three type of signals are considered, Gaussian signals (whose non-zero coefficients are drawn from a Gaussian distribution, zero mean and variance equal to 1), uniform signals (whose non-zero coefficients take values $+1$ or -1 with equal probability).

A. Sparsity in the original domain

In this case, $\boldsymbol{x}$ is itself CSS, $\boldsymbol{y}$ is formed as in (1). At the recovery step, OMP utilizes y and Φ to recover $\hat{x}$. As results, the proposed matrix gives good results with Gaussian CSS signals and uniform CSS signals (Fig. 2).

B. Sparsity in a transform domain

Here, we consider that the signal $\boldsymbol{x}$ is sparse in the discrete cosine transform (DCT) domain. For that, we form $\boldsymbol{\alpha}$ as a CSS vector with S non-zero entries drawn from a Gaussian distribution, and then form the signal $\boldsymbol{x}$ (i.e. $\boldsymbol{x}$=$\boldsymbol{\Psi\alpha}$, where $\boldsymbol{\Psi}$ is the IDCT matrix). We form $\boldsymbol{y}$ as in (9) for the BNO Kasami, NO Kasami and bBH matrices, and as in (1) for the other matrices. At the recovery step, OMP utilizes $\boldsymbol{y}$ and $\boldsymbol{A}$=$\boldsymbol{\Phi}'\boldsymbol{\Psi}$ to recover $\hat{\boldsymbol{\alpha}}$, and then $\hat{\boldsymbol{x}}$=$\boldsymbol{\Psi}\hat{\boldsymbol{\alpha}}$. From Fig. 3 one can see the

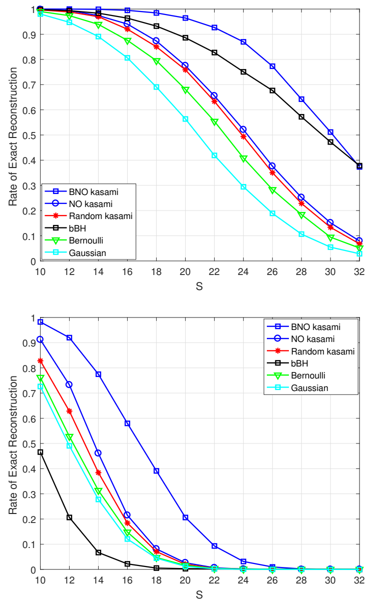

Fig. 2: N=128, M=64, G=8: Gaussian signals (top), Uniform signals (bottom).

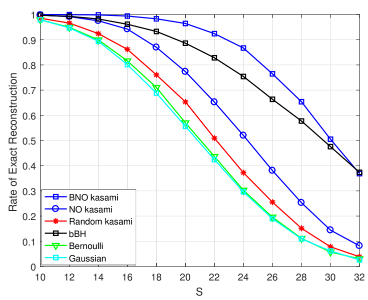

Fig. 3: N=128, M=64, G=8: Gaussian signals.

superior performance of our matrix compared with the other matrices.

C. Practical application

We consider 252×252 natural images each is divided to blocks of size 9×9 and consider the signal $\boldsymbol{x}$ as a vector 81×1 obtained by stacking the columns of a 9×9 image pixels block; $\boldsymbol{x}$ is CSS in DCT domain (i.e. $\boldsymbol{x}$=$\boldsymbol{\Psi\alpha}$ where $\boldsymbol{\Psi}$ is the IDCT transform matrix and $\boldsymbol{\alpha}$ is the DCT coefficients vector of $\boldsymbol{x}$). This cluster structured sparsity is illustrated in Fig. 4. Moreover, the clusters locations are similar for the different blocks, the fact that makes their prior knowledge possible. So, the clusters locations are chosen as $\boldsymbol{L}$={$\boldsymbol{L}_1$=1, $\boldsymbol{L}_2$=19, $\boldsymbol{L}_3$=37, $\boldsymbol{L}_4$=54, $\boldsymbol{L}_5$=73}. We construct a 64×81 NO Kasami matrix (N_{ex}=18) and then reorder its columns to form the BNO Kasami one. This columns reorder is based on the clusters locations; for that the columns of indices ranging from 1 to 18 are moved to the indices [**1**,2,3,4]∪[17,18,**19**,20]∪[35,36,**37**,38]∪[53,**54**,55]∪[72,**73**,74] and we fulfill the rest of BNO matrix by the NO matrix columns of indices ranging from 19 to 81. We form a 64×1 vector $\boldsymbol{y}$ as in (9) for the BNO Kasami, NO Kasami and bBH matrices, and as in (1) for the other matrices. At the recovery step, OMP algorithm is used to calculate $\hat{\boldsymbol{\alpha}}$. Then we form the approximate signal as $\hat{\boldsymbol{x}}$=$\boldsymbol{\Psi}\hat{\boldsymbol{\alpha}}$ and reshape it to form again the image block. Finally we calculate the SNR over the whole image. TABLE II, shows that the proposed matrix outperforms the other matrices. Also Fig. 5 gives a comparative illustration to show the power of our matrix.

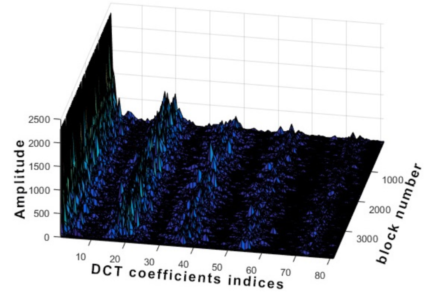

Fig. 4: Cluster structured sparsity of the DCT coefficients of natural images.

TABLE II: SNR in dB with N=81, M=64.

	Cameraman	Barbara	Rice	Lena
BNO Kasami	**23.0703**	**26.2793**	**23.0350**	**25.8999**
NO Kasami	22.1762	25.8427	22.1387	24.9411
Random Kasami	19.5143	22.4309	19.6928	22.6385
bBH	21.8584	25.8215	22.2394	25.2518
Bernoulli	20.3862	23.4640	20.4783	23.4547
Gaussian	20.3569	23.4504	20.4586	23.4056

Fig. 5: N=81, M=64: BNO Kasami (left), Random Kasami (right).

V. Conclusion

Unlike the state-of-art works dealing with CSS signals that focus on the recovery step, in this paper, we focus on the sensing step by proposing a new Kasami sensing matrix for CSS signals. The proposed matrix is based on the idea of enhancing the columns orthogonality. Simulation results show that this new matrix outperforms all the popular matrices in the case of CSS signals. Moreover, its deterministic aspect makes it more suitable to be implemented on hardware.

REFERENCES

[1] Candes, E. J., Romberg, J., Tao, T.: ”Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information”, IEEE Transactions on Information Theory, (2006), 52(2), 489-509.

[2] Candes, E. J.: ”Compressive sampling”. In Proceedings of the international congress of mathematicians (Vol. 3, pp. 1433-1452). Madrid, (2006, August).

[3] Zhang, G., Jiao, S., Xu, X., Wang, L.: ”Compressed sensing and reconstruction with bernoulli matrices”, IEEE International Conference on In Information and Automation (ICIA) (pp. 455-460). Harbin, China, 2010.

[4] Tropp, J. A., Gilbert, A. C.: ”Signal recovery from random measurements via orthogonal matching pursuit”, IEEE Transactions on Information Theory, (2007), 53(12), 4655-4666.

[5] Candes, E., Romberg, J.: ”l1-magic: Recovery of sparse signals via convex programming”, URL: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf, 4, 14, (2005).

[6] Baraniuk, R. G., Cevher, V., Duarte, M. F., Hegde, C.: ”Model-based compressive sensing”. IEEE Transactions on Information Theory, (2010), 56(4), 1982-2001.

[7] Eldar, Y. C., Kuppinger, P., Bolcskei, H.: ”Block-sparse signals: Uncertainty relations and efficient recovery”. IEEE Transactions on Signal Processing, (2010), 58(6), 3042-3054.

[8] V. Cevher, P. Indyk, C. Hegde, R. G. Baraniuk: ”Recovery of clustered sparse signals from compressive measurements”, in:International Conference on Sampling Theory and Applications (SAMPTA). Marseille, France, 2009.

[9] Merhej, D., Diab, C., Khalil, M., Prost, R.: ”Embedding prior knowledge within compressed sensing by neural networks”. IEEE transactions on neural networks, (2011), 22(10), 1638-1649.

[10] Yu, L., Sun, H., Barbot, J. P., Zheng, G.: ”Bayesian compressive sensing for cluster structured sparse signals”. Signal Processing, (2012), 92(1), 259-269.

[11] Yu, L., Wei, C., Jia, J., Sun, H.: ”Compressive sensing for cluster structured sparse signals: Variational Bayes approach”. IET Signal Processing, (2016), 10(7), 770-779.

[12] Lazzaro, D., Montefusco, L. B., Papi, S.: ”Blind cluster structured sparse signal recovery: A nonconvex approach”. Signal Processing, (2015), 109, 212-225.

[13] Nouasria, H., Et-tolba, M.: ”A novel sensing matrix for cluster structured sparse signals”. In 13th International Wireless Communications and Mobile Computing Conference (IWCMC) (pp. 998-1003). IEEE. Valence, Spain, (2017, June).

[14] Eldar, Y. C.: ”Generalized SURE for exponential families: Applications to regularization”. IEEE Transactions on Signal Processing, (2009), 57(2), 471-481.

[15] Nouasria, H., Et-tolba, M.: ”Sensing matrix based on Kasami codes for compressive sensing”. IET Signal Processing, (2018).