

# Projection Matrix Design for Co-Sparse Analysis Model Based Compressive Sensing

Endra Oey<sup>1,\*</sup>, Dadang Gunawan<sup>1</sup>, and Dodi Sudiana<sup>1</sup>

<sup>1</sup>Wireless and Signal Processing (WASP), Department of Electrical Engineering, University of Indonesia, Depok City, West Java 16424, Indonesia

**Abstract.** Co-sparse analysis model based-compressive sensing (CAMB-CS) has gained attention in recent years as alternative to conventional sparse synthesis model based (SSMB)-CS. The equivalent operator as counterpart of the equivalent dictionary in the SSMB-CS is introduced in the CAMB-CS as the product of projection matrix and transpose of the analysis dictionary. This paper proposes an algorithm for designing suitable projection matrix for CAMB-CS by minimizing the mutual coherence of the equivalent operator based on equiangular tight frames design. The simulation results show that the CAMB-CS with the proposed projection matrix outperforms the SSMB-CS in terms of the signal quality reconstruction.

## 1 Introduction

Compressive sensing (CS) as a new paradigm in signal acquisition has gained popularity over the last decade after it was introduced in [1-2]. CS acquires the signal directly in already compressed form by projecting it into a well-designed projection matrix. CS framework has been applied in many applications such as imaging applications, internet of thing, data security, and more [3-4]. A conventional CS systems works based on the sparse synthesis model of signal where a signal can be synthesized from a few atoms of a synthesis dictionary [5]. The alternative model is co-sparse model where sparse analysis coefficients can be obtained by multiplying the signal and an analysis dictionary (operator) [6]. Co-sparse analysis model based (CAMB)-CS has attracted attention in recent years because it outperforms the synthesis model as shown in [7-8].

Three main problems of CS are how to build a dictionary, design a proper projection matrix and reconstruct the signal from CS. The famous KSVd algorithm and its extensions have been commonly used to build a synthesis dictionary [9-10] also the improvements by exploiting additional structure of sparse coefficients can be found in [11-12]. The analysis version of KSVd [13] and sparsifying transforms learning algorithms have been used to build an operator [14-15]. The Convex and Relaxation, Greedy, and Bayesian algorithms are used for signal reconstruction in synthesis based CS [16] as well as the counterpart algorithms for analysis based CS [6, 17].

---

\* Corresponding author: [oe.y.endra51@ui.ac.id](mailto:oe.y.endra51@ui.ac.id)

While how to design optimal projection matrix for sparse synthesis model based (SSMB)-CS has been widely proposed such as in [18-19] but for CAMB-CS has not received attention. This paper addresses how to design a projection matrix for CAMB-CS, use it to perform CS on a natural image and compare the image reconstruction performance to SSMB-CS.

## 2 SSMB-CS And CAMB-CS

In the SSMB-CS, the signal  $x \in \mathfrak{R}^{N \times 1}$  is synthesized from a sparse linear combinations of the dictionary columns  $\Psi \triangleq [\psi_1 \psi_2 \dots \psi_L] \in \mathfrak{R}^{N \times L}$ :

$$x = \sum_{i=1}^L \theta_i \psi_i \triangleq \Psi \theta \quad (1)$$

if  $L > N$   $\Psi$  is called *overcomplete synthesis dictionary*,  $\theta \in \mathfrak{R}^{N \times 1}$  is sparse coefficients with  $\|\theta\|_0 = K$ ,  $K < N$  and  $\|\theta\|_0$  is number of non-zero elements in  $\theta$ . The CS is performed by multiplying the signal  $x$  and the projection matrix  $\Phi \in \mathfrak{R}^{M \times N}$  then  $y = \Phi x$ , where  $y \in \mathfrak{R}^{M \times 1}$  is a compressive measurement vector and  $M < N$ . The reconstructed signal  $\hat{x}$  can be obtained from  $y$  by solving the following constrained problem :

$$\min \|\hat{\theta}\|_0 \text{ s.t. } y = \Phi \Psi \hat{\theta} \text{ and } \hat{x} = \Psi \hat{\theta} \quad (2)$$

The problem in (2) is NP-hard and has combinatorial complexity but can be approximately solved using the Convex and Relaxation, Greedy, or Bayesian algorithms [16].

In CAMB-CS, the analysis coefficients  $\alpha \in \mathfrak{R}^{R \times 1}$  are obtained by multiplying the operator  $\Omega \in \mathfrak{R}^{R \times N}$  and the signal  $x \in \mathfrak{R}^{N \times 1}$  then  $\alpha = \Omega x$ . If  $R > N$ ,  $\Omega$  is called *overcomplete operator*. The *co-support*  $\Lambda \subseteq \{1 \dots R\}$  of  $x$  is the index rows of  $\Omega$  that are orthogonal to it, so  $\Omega_\Lambda x = 0$  where  $|\Lambda| = c$  is *co-sparsity* of the signal  $x$  and  $\|\alpha\|_0 = R - c$  is number of non-zero elements in  $\alpha$ . The reconstructed signal  $\hat{x}$  can be obtained from CAMB-CS by solving the following constrained problem :

$$\min \|\Omega \hat{x}\|_0 \text{ s.t. } y = \Phi \hat{x} \quad (3)$$

The problem in (3) can be solved by using the SSMB-CS counterpart algorithms for CAMB-CS [17].

## 3 Projection Matrix Design

In the SSMB-CS, the equivalent dictionary  $D \in \mathfrak{R}^{M \times L}$  is defined as  $D = \Phi \Psi$  where  $G_s = D^T D$  is the Gram matrix of  $D$ ,  $g_{s-ij}$  is the  $(i, j)$ th elements of  $G_s$  and  $Z_s \triangleq \text{diag}(g_{s-11}^{-1/2} \dots g_{s-ll}^{-1/2} \dots g_{s-LL}^{-1/2})$ . The normalized equivalent dictionary is  $\bar{D} \triangleq D Z_s$  and the normalized Gram matrix is  $\bar{G}_s = \bar{D}^T \bar{D}$  such that  $\bar{g}_{s-ll} = 1, \forall l$ . The mutual coherence

of  $D$  is defined as  $\mu(D) = \max_{i \neq j} |\bar{g}_{s-ij}|$  where  $\mu(D)$  is bounded with  $\mu_B \leq \mu(D) \leq 1$  and  $\mu_B \triangleq \sqrt{\frac{L-M}{M(L-1)}}$  is Welch bound [20]. The  $t$ -averaged mutual coherence  $\mu_t(D)$  was

introduced in [18] as  $\mu_t(D) := \frac{\sum_{i \neq j, 1 \leq i, j \leq L} \left( |\bar{g}_{s-ij}| \geq t \right) |\bar{g}_{s-ij}|}{\sum_{i \neq j, 1 \leq i, j \leq L} \left( |\bar{g}_{s-ij}| \geq t \right)}$  where  $0 \leq t < 1$  and indicator

function  $\left( |\bar{g}_{s-ij}| \geq t \right) = 1$  if the condition is true and otherwise is zero. The common projection matrix design for SSMB-CS is based on how to make  $\mu_t(D)$  as small as possible. It is done by making  $G$  as close as possible to a target Gram matrix  $G_t$  that has desired properties such as ETF Gram matrix [20]. The projection matrix design is performed by solving:

$$\min_{\Phi, G_t} \left\| G_t - \Psi^T \Phi^T \Phi \Psi \right\|_F^2 \tag{4}$$

where  $\| \cdot \|_F$  denotes the *Frobenius* norm and it can be solved based on shrinkage method [18] or alternating projection [19]. The equivalent operator  $O \in \mathfrak{R}^{M \times R}$  is introduced for CAMB-CS as  $O = \Phi \Omega^T$  where  $G_c = O^T O$  is the Gram matrix of  $O$ ,  $g_{c-ij}$  is the  $(i, j)$ th elements of  $G_c$  and  $Z_c \triangleq \text{diag}(g_{c-11}^{-1/2} \dots g_{c-ll}^{-1/2} \dots g_{c-LL}^{-1/2})$ . The normalized equivalent operator is  $\bar{O} \triangleq O Z_c$  and the normalized Gram matrix is  $\bar{G}_c = \bar{O}^T \bar{O}$  such that  $\bar{g}_{c-ll} = 1, \forall l$ . The projection matrix design for CAMB-CS is performed by solving:

$$\min_{\Phi, G_t} \left\| G_t - \Omega \Phi^T \Phi \Omega^T \right\|_F^2 \tag{5}$$

This paper adapted algorithm in [19] to solve (5) by using the following algorithm which is denoted as OGS algorithm.

### OGS Algorithm

**Initialization:**

- $\Omega$  - Operator;
- $\Phi^{(0)}$  - Initial Projection Matrix;
- $K$  - Number of Iterations;
- Set  $k = 1$ .

**Begin:** For  $1 \leq k \leq K$ , calculate (6) until (12):

$$O^{(k-1)} = \Phi^{(k-1)} \Omega^T \tag{6}$$

$$\bar{O}^{(k-1)} = O^{(k-1)} Z_c \tag{7}$$

$$\bar{G}_c^{(k-1)} = \left( \bar{O}^{(k-1)} \right)^T \left( \bar{O}^{(k-1)} \right) \tag{8}$$

$$\overline{G}_{c-e}^{(k)}(i, j) = \begin{cases} 1, & i = j \\ \overline{G}_c^{(k-1)}(i, j), & |\overline{G}_c(i, j)| \leq \xi \\ \text{sign}(\overline{G}_c^{(k-1)}(i, j))\xi & \text{otherwise} \end{cases} \quad (9)$$

eigenvalue decomposition of  $\overline{G}_{c-e}^{(k)}$  :

$$\overline{G}_{c-e}^{(k)} = V_e \begin{bmatrix} \Lambda_e & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V_e^T \quad (10)$$

where  $V_e$  is orthonormal matrix and  $\Lambda_e = \text{diag}[\lambda_1, \dots, \lambda_r, \dots, \lambda_R]$  with  $\lambda_r \geq \lambda_{r+1}, \forall r$

$$O_{eff}^{(k)} = \frac{U}{\sqrt{M}} \begin{bmatrix} \sqrt{\left(\sum_{r=1}^M \lambda_r\right)} I_M & \mathbf{0} \end{bmatrix} V_e^T \quad (11)$$

$$\Phi^{(k)} = O_{eff}^{(k)} (\Omega^T)^+ \quad (12)$$

where  $(\Omega^T)^+$  is pseudo-inverse of  $(\Omega^T)$ .

Go to **Begin** with  $k \rightarrow k + 1$  to continue the iterative procedure.

**End:** End the algorithm, output  $\Phi = \Phi^{(k-1)}$ .

## 4 Results and Discussion

This paper used 1000 training-images in LabelMe training data set [21-22] where 20 non-overlapping  $8 \times 8$  patches are taken randomly from each image and each patch is re-arranged as a vector of  $64 \times 1$ . This training patches  $P \in \mathfrak{R}^{64 \times 20000}$  were used to build synthesis dictionary  $\Psi \in \mathfrak{R}^{64 \times 96}$  by using KSVD algorithm [9] and operator  $\Omega \in \mathfrak{R}^{96 \times 64}$  by using the algorithm in [15]. Algorithm in [19] which is denoted with BLH and OGS algorithm were used for SSMB-CS and CAMB-CS projection matrix design respectively. Both algorithms used the same Gaussian random matrix  $\Phi^{(0)} \in \mathfrak{R}^{M \times 64}$  as initial projection matrix, number of iterations  $K = 1000$  and  $\xi = \mu_B$ . The test image  $I^{test}$  was divided into  $8 \times 8$  non-overlapping patches where each patch convert to vector  $x_j \in \mathfrak{R}^{64 \times 1}$ ,  $j = 1, 2, \dots, J$  where  $J$  is number of patches in the tes image. CS was performed on those patches to obtain  $y_j \in \mathfrak{R}^{M \times 64}$ . The OMP [23] and its counterpart Greedy algorithm GAP [6] were used for SSMB-CS and CAMB-CS respectively to obtain each reconstructed patch  $\hat{x}_j \in \mathfrak{R}^{64 \times 1}$  and the whole reconstructed patches are arranged to get the reconstructed image  $\hat{I}$ . The Peak Signal-to-Noise Ratio (PSNR) was used to measure image reconstruction

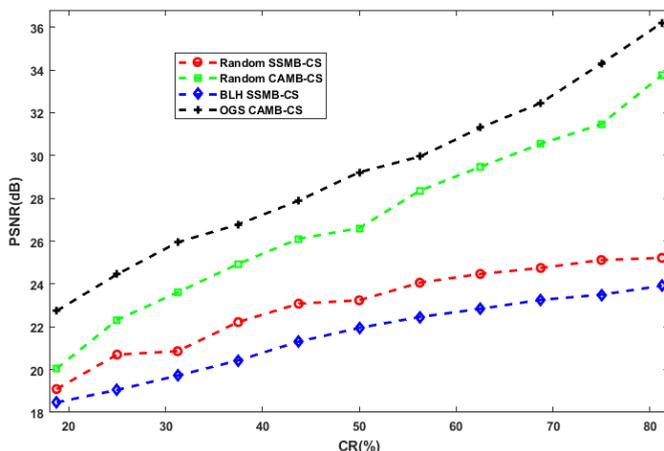
accuracy. It is defined as  $\text{PSNR (dB)} = 10 \log_{10} \left( \frac{W \times H \times 1^2}{\sum_{w=1}^W \sum_{h=1}^H (\hat{I}_{w,h} - I_{w,h}^{test})^2} \right)$  where  $W$  and  $H$

are width and height of image.

Table 1 shows the reconstruction (in PSNR (dB)) comparison of SSMB-CS and CAMB-CS for several standard test images with Compression Ratio (CR) = Ratio Measurement Number (RMN) =  $\frac{M}{N} = 31.25\%$ . Figure 1 shows the PSNR versus CR from 18.75 % until 81.25 % for Brick test image.

**Table 1.** Reconstruction comparison of SSMB-CS and CAMB-CS for CR = 31.25 %.

Projection Matrix	Test Images (Size)			
	Moon (1986 x 1986)	Brick (768 x 512)	Barbara (720 x 576)	Peppers (512 x 512)
	PSNR (dB)			
Random SSMB-CS	27.48	20.86	22.52	25.14
Random CAMB-CS	31.46	23.61	23.93	28.45
BLH SSMB-CS	27.22	19.72	21.71	24.03
<b>OGS CAMB-CS</b>	<b>33.98</b>	<b>25.95</b>	<b>25.89</b>	<b>31.25</b>



**Fig. 1.** PSNR (dB) versus CR (%) of Brick test image for SSMB-CS and CAMB-CS.

From Table 1 and Figure 1, it is clear that the proposed algorithm in this paper (OGS CAMB-CS) outperforms the random projection matrix and the previous algorithm (BLH SSMB-CS). It is noted that the reconstruction time for CAMB-CS is comparable to SSMB-CS. Reconstruction time of Barbara test image, as an example, for CR = 31.25 % are 4.22 s, 6.19 s, 4.45 s and 6.67 s for Random SSMB-CS, Random CAMB-CS, BLH SSMB-CS and OGS CAMB-CS respectively.

## 5 Conclusion

In this paper, the projection matrix design algorithm for CAMB-CS was proposed to improve image reconstruction accuracy. The results show that CAMB-CS outperforms the SSMB-CS in terms of PSNR of the image reconstruction. Further improvement can be attempted in future work by designing projection matrix and operator learning simultaneously.

This work was supported by the PITTA-Grant 2017 University of Indonesia.

## References

1. D.L. Donoho, Compressed Sensing, *IEEE Trans. Inf. Theory* **52(4)**, 1289-1306 (2006).
2. E.J. Candès, J.R. T. Tao, Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information, *IEEE Trans. Inf. Theory* **52(2)**, 489-509 (2006).
3. J.A.M Lorenzo, R. Obermeier, Sensing Matrix Design via Mutual Coherence Minimization for Electromagnetic Compressive Imaging Applications, *IEEE Trans. Comp. Img* **3(2)**, 217-229 (2017).
4. S.Qaisar, R.M. Bilal, W. Iqbal, M. Naureen, S. Lee, Compressive Sensing: From Theory to Applications, a Survey, *ComNet J* **15(5)**, 443 – 456 (2013).
5. A. M. Bruckstein, D. L. Donoho, M. Elad, From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images, *SIAM Rev* **51(1)**, 34–81 (2009).
6. S. Nam, M. Davies, M. Elad, R.Gribonval, The Cospase Analysis Model and Algorithms, *Appl. Comput. Harmon. Anal* **34(1)**, 30–56 (2013).
7. O. Endra, D. Gunawan, Comparison of Synthesis-Based and Analysis-Based Compressive Sensing, In Proceedings of IEEE International Conference QiR, 167-170 (2015).
8. R. Saiprasad, Y. Bresler, Data-Driven Learning of a Union of Sparsifying Transforms Model for Blind Compressed Sensing, *IEEE Trans. Comp. Img* **2(3)**, 294-309 (2016).
9. M. Aharon, M. Elad, A. Bruckstein, The KSVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation, *IEEE Trans. Sig. Proc*, **54(11)**, 4311-4322 (2006).
10. I. Kviatkovsky, M. Gabel, E. Rivlin, I. Shimshoni, On the Equivalence of the LC-KSVD and the D-KSVD Algorithms, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **39(2)**, 411-416 (2017).
11. M.F. Duarte, V. Cevher, R.G. Baraniuk, Model-based compressive sensing for signal ensembles, In Proceedings of 47<sup>th</sup> Annual IEEE Allerton Conference on Communication, Control, and Computing, 244-250 (2009).
12. C. Hegde, P. Indyk, L. Schmidt, Approximation-tolerant model-based compressive sensing, In Proceedings of 25<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms, 1544-1561 (2014).
13. R. Rubinstein, T. Peleg, M. Elad, Analysis K-SVD: A Dictionary-Learning Algorithm for the Analysis Sparse Model. *IEEE Trans. Sig. Proc*, **61(3)**, 661-677 (2013).
14. S. Ravishankar, Y. Bresler, Learning Sparsifying Transforms, *IEEE Trans. Sig. Proc* **61(5)**, 1072-1086 (2013).
15. B. Hou, Z. Zhu, G. Li, A. Yu, An Efficient Algorithm for Overcomplete Sparsifying Transform Learning with Signal Denoising, *Math. Problems. in Eng.* (2016).
16. Y. Arjoun, N. Kaabouch, H.E. Ghazi, A. Tamtaoui, Compressive Sensing: Performance Comparison of Sparse Recovery Algorithms, In IEEE 7<sup>th</sup> Annual Computing and Communication Workshop and Conference (CCWC), 1-7 (2017).
17. R.Giryas, A Greedy Algorithm for the Analysis Transform Domain, *Neurocomputing* **173**, 278-289 (2016).
18. M. Elad, Optimized Projections for Compressed Sensing, *IEEE Trans. Sig. Proc*, **55(12)**, 5695–5702 (2007).
19. H. Bai, S. Li, X. He, Sensing Matrix Optimization Based on Equiangular Tight Frames with Consideration of Sparse Representation Error, *IEEE Trans. Multimedia*, **18(10)**, 2040-2053 (2016).
20. T. Strohmer, R.W. Heath, Grassmannian Frames with Applications to Coding and Communication, *Appl. Comp. Harmon. Analysis*, **14(3)** (2003).
21. B. C. Russell, A. Torralba, K. P. Murphy, W. T. Freeman, LabelMe: A Database and Web-Based Tool for Image Annotation, *Comput. Vis. J.* **77(1)**, 157–173 (2008).
22. R. Uetz, S. Behnke, Large-Scale Object Recognition with CUDA-Accelerated Hierarchical Neural Networks, In Proceedings of IEEE ICIS, 536-541 (2009).
23. J. A. Tropp, A.C. Gilbert, Signal Recovery from Random Measurements via Orthogonal Matching Pursuit, *IEEE Trans. Inf. Theory* **53(12)**, 4655–4666 (2007).