Multi-Task Compressive Sensing of Vibration Signal using GMM Clustering for Wireless Transmission

Yun-Fei Ma, Hua-Jun Bai, Xi-Sheng Jia*, Guang-Long Wang, Chi-Ming Guo
Shijiazhuang Campus
Army Engineering University
Shijiazhuang 050003, P.R. China

***Abstract*—Wireless sensor network has been widely used in the prognostic and health management system. Signal compression is often required during wireless transmission because of the high sampling frequency. In recent years, compressive sensing has great potential to increase the efficiency of wireless transmission. In this paper, we use the multi-task compressive sensing within a multi-task learning setting. Under this modeling, data from all tasks contribute toward inferring a posterior on the parameters. Moreover, a GMM clustering method is used for finding the best classification among signal blocks. Simulation results show that more signal blocks may not necessarily lead to higher reconstruction accuracy. Conversely, reconstruction error may be reduced by concentrating signal blocks with similar distribution together. The performance of proposed method is validated by gearbox data.**

Keywords—Compressive sensing; gearbox; Bayesian compressive sensing; Gaussian mixture model; clustering.

I. INTRODUCTION

Mechanical machinery has been widely used in modern industry, agriculture and military field. Considering the manufacturing mistakes, working environment, the damages and faults may occur inevitably during operation. Proper monitoring for equipment conditions is then essential to guarantee the safety of machinery. It is of great potential to use distributed wireless sensor technology to realize signal transmission by monitoring equipment remotely. The compressive sensing [1-3] has emerged which can be regarded as a breakthrough of Nyquist sampling theorem. The theory describes a new data acquisition mode which combines data acquisition and data compression. Compressive sensing has the ability of reconstructing original signals from the less sampling points compared to the Nyquist sampling. In recent years, compressive sensing (CS) has been successfully applied in signal compression [4], SAR imaging [5], face recognition [6], DOA estimation [7], and long-term wireless electrocardiogram (ECG) [8]. Bayesian compressed sensing (BCS) [9-10] realizes signal reconstruction by learning Related Vector Machines (RVM) in Sparse Bayes [11]. Compared to traditional algorithms [12-13], BCS has smaller mean square error and achieves fast convergence at a small compression ratio.

Cluster analysis [14] is a kind of approaches for exploiting clusters by the most similarity within the same cluster. Previously, researchers have proposed a variety of clustering methods. These clustering methods may be divided into two parts: nonparametric method and probability-based method. Nonparametric methods define a distance function for similarity evaluation, including k-means [15], fuzzy c-means [16]. However, probability model-based methods are performed under the assumption that all data points follow the same mixture model of distribution. In fact, Dean [17] et al used both methods and found that the probability model-based approaches always give better clustering performance. Here we use a Gaussian mixture model (GMM) [18] belong to the second category of methods. In GMM model, an expectation and maximization (EM) algorithm has always been used for similarity estimation which will be explained in detail below.

The contribution of this paper is twofold: (1) we introduce a multi-task compressive sensing (MCS) model. Traditional compressed sensing reconstructs the original signal by observation values, and they perform the recovery of each signal independently. When facing multiple correlated signals, MCS can achieve unified observation and recovery all signals together. (2) we use a GMM method for sparse blocks clustering. Under this framework, all tasks may be clustered into several sets of tasks and data sharing may only be appropriate within each cluster. Each cluster often shares the similar prior which may greatly reduce the measurements for reconstruction. There are about 20 features of vibration signal selected for classification which may lead to a huge computation. Therefore, a PCA method is used here for feature-dimension reduction. Based on the improvement above, a new BCS algorithm, termed as BCS-GMM method, is proposed here for better achievement. The new method provides technical supports for wireless monitoring of mechanical equipment to some extent.

II. RELATED WORK: MULTI-TASK COMPRESSIVE SENSING

In this paper, we try to reconstruct L blocks of signal together, using multi-task compressive sensing. The model for each task can be represented as

$$y_i = \Theta_i \theta_i + n_i \qquad (1)$$

Where n_i represents zero mean Gaussian noise，y_i represents the received signal, Θ_i is the measurement matrix. And then the priors of y_i can be expressed as

$$p(y_i \mid \theta_i, \sigma_i^2) = \left(\frac{1}{\sqrt{2\pi\sigma_i^2}} \right)^K \exp\left(-\frac{\|y_i - \Theta_i \theta_i\|_2^2}{2\sigma_i^2}\right) \qquad (2)$$

978-1-7281-0860-5/19/$31.00 ©2019 IEEE

Specifically, letting $\theta_{i,j}$ represent the *jth* wavelet coefficient for task *i,* we have the priors of θ_i

$$p(\theta_i|\alpha)=\prod_{j=1}^{N}N(\theta_{i,j}|0,\alpha_j^{-1}) \tag{3}$$

Where $\alpha=(\alpha_1, \alpha_2,\ldots, \alpha_n)$ represents the sharing hyper-parameters among all tasks. To promote the sparsity, we assume that Gamma priors are placed on the hyperparameters α, as well as the noise precision α_0.

$$p(\alpha,\alpha_0|\{y_1,...,y_L\},a,b,c,d)$$
$$=\frac{p(\alpha_0|a,b)p(\alpha|c,d)\prod_{i=1}^{L}\int p(y_i|\theta_i,\alpha_0)p(\theta_i|\alpha)d\theta_i}{\iint p(\alpha_0|a,b)p(\alpha|c,d)d\alpha_0 d\alpha\prod_{i=1}^{L}\int p(y_i|\theta_i,\alpha_0)p(\theta_i|\alpha)d\theta_i} \tag{4}$$

In equation (4), a, b, c, d are the parameters used in Gamma priors of α and α_0. Given equation (4) and measurements $(y_1,y_2,\ldots,y_L)$, one may infer a posterior of α and α_0. We try to utilize $p(\theta_i \mid y_i, \alpha, \alpha_0)$ to estimate the Bayesian hyperparameters $\{\alpha, \alpha_0\}$.

$$p(\theta_i|y_i,\alpha,\alpha_0)=\frac{p(y_i|\theta_i,\alpha_0)p(\theta_i|\alpha)}{\int p(y_i|\theta_i,\alpha_0)p(\theta_i|\alpha)d\theta_i}=N(\theta_i|\mu_i,\sum_i) \tag{5}$$

The mean μ_i and covariance $\sum_i$ can be estimated using equation (5) under RVM model. Differentiating equation (5) with respect to α and α_0, we get representations of α and α_0. Then we perform iterations among the equations of μ_i, $\sum_i$, α, α_0 until a convergence has been achieved. It can be found that the process of solving α and α_0 makes full use of all the parameters among L block signals which is termed as joint learning. Thus, the efficient of this algorithm is better than solving each block of signal separately.

III. THE MCS-GMM METHOD

A. Gaussian mixture model and EM iteration solving

Gaussian mixture model(GMM) describes mixing density distribution by combining multivariate normal density components. Fig. 1 shows the process of GMM clustering. Assuming a batch of observed data, the GMM is established by mixing these data points from different Gaussian distribution together based on calculating probabilities of each data point. Therefore, the distribution of the *kth* class is a mixture Gaussian given below

$$p(x)=\sum_{i=1}^{k}\pi_i N(x|\eta_i,\Gamma_i) \tag{6}$$

where $N(x|\eta_i, \Gamma_i)$ is the density of a multivariate Gaussian random variable X with mean η_i and covariance matrix Γ_i. π_i is the weight of the kth Gaussian distribution. Since ρ is parameters to be estimated in the GMM, the density of all observed data can be represented as

$$p(x|\rho)=\sum_{i=1}^{k}\pi_i p(x|\rho)$$
$$=\sum_{i=1}^{k}\pi_i(2\pi)^{-\frac{d}{2}}|\Gamma_i|^{-\frac{1}{2}}\exp\{-\frac{1}{2}(x-\eta_i)^{\mathrm{T}}\Gamma_i^{-1}(x-\eta_i)\} \tag{7}$$

Like k-means clustering, Gaussian mixture modeling uses an iterative algorithm that converges to a local optimum. Let X be the observed data, Y be the missing data and Z={X, Y} be the complete data. We adopts an EM-iterative algorithms here.

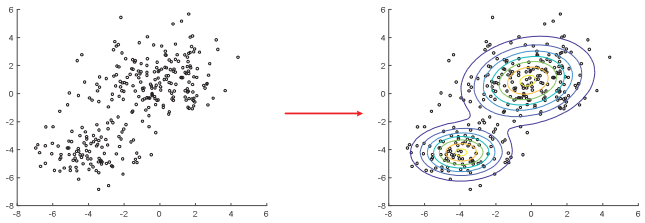


Figure 1. The schematic diagram of GMM model

E-step: Since the likelihood function of p(X, Y| ρ) can be calculated by initial value, the condition expected value of $p(Y \mid X, \rho^{\mathrm{old}})$ is:

$$Q(\rho,\rho^{old})=E_Y\left[\lg p(X,Y|\rho)\middle|X,\rho^{old}\right]$$
$$=\sum\nolimits_{y\in Y}\lg p(X,Y|\rho)p(Y|X,\rho^{old})$$
$$=\sum\nolimits_{y\in Y}\sum_{i=1}^{K}\lg(\pi_{yi}p(x_i;\eta_{yi},\Gamma_{yi}))p(Y|X,\rho^{old}) \tag{8}$$
$$=\sum_{k=1}^{N}\sum_{i=1}^{k}\lg(\pi_k p_k(x_i;\eta_k,\Gamma_k))p(k|x_i,\rho^{old})$$

M-step: To maximize $Q(\rho, \rho^{\mathrm{old}})$ and under the constraint of equation (8), we can obtain the updated equation for mixing proportions with

$$\pi_k^{new}=\frac{1}{K}\sum_{i=1}^{K}p(k|x_i,\rho^{old}) \tag{9}$$

The parameter ρ_k consists of a mean vector η_k and a covariance matrix Γ_k as follow:

$$\eta_k^{new}=\frac{\sum_{i=1}^{k}p(k|x_i,\rho^{old})x_i}{\sum_{i=1}^{k}p(k|x_i,\rho^{old})} \tag{10}$$

$$\Gamma_k^{new}=\frac{\sum_{i=1}^{k}p(k|x_i,\rho^{old})(x_i-\eta_k^{new})(x_i-\eta_k^{new})^{\mathrm{T}}}{\sum_{i=1}^{k}p(k|x_i,\rho^{old})} \tag{11}$$

With the alternate use of E-step and M-step, EM algorithms will iteratively improve the parameters of GMM until $\|\rho - \rho^{\mathrm{old}}\|$ is sufficiently small. In this way, the estimated GMM parameters can be more in line with the real parameters.

B. feature extraction

(1) Time-domain feature

The time-domain features can be divided into two parts. The dimensional part includes: maximum, minimum, peak to peak, mean, root mean square, variance, standard deviation, energy. While the non-dimensional part includes: skewness, kurtosis, waveform index, peak value index, pulse index, margin index and margin coefficient. Table 1 contains 12 calculation formulas of classical time-domain features.

TABLE I. TIME-DOMAIN FEATURE

feature	Peak to peak(T_1)	Mean(T_2)	root mean square(T_3)	Variance(T_4)	standard deviation(T_5)	Energy(T_6)
Calculation formula	$\max(\boldsymbol{x})$- $\min(\boldsymbol{x})$	$\frac{1}{N}\sum_{i=1}^{N} x_i$	$\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$	$T_3^2 - T_2^2$	$\sqrt{T_4}$	$\sum_{i=1}^{N} x_i^2$
feature	Skewness(T_7)	Kurtosis(T_8)	Peak value index(T_9)	Pulse index(T_{10})	Margin index(T_{11})	Margin coefficient(T_{12})
Calculation formula	$\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - T_2)^3}{T_3^3}$	$\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - T_2)^4}{T_3^4}$	$\frac{\max(\lvert x_i\rvert)}{T_3}$	$\frac{\max(\lvert x_i\rvert)}{\frac{1}{N}\sum_{i=1}^{N}\lvert x_i\rvert}$	$\frac{\max(\lvert x_i\rvert)}{(\frac{1}{N}\sum_{i=1}^{N}\sqrt{\lvert x_i\rvert})^2}$	$\frac{\max(\boldsymbol{x})}{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$

(2) Energy features of wavelet packet

The wavelet packet transform takes advantage of analysis tree to represent the wavelet packets. The purpose of this transform is to overcome the problem of less precise in high-frequency transform. The technology is used to decompose the signal into three layers using Matlab toolbox and the energy of eight decomposed coefficients in the third layer is taken as classification features. Thus, there are 20 features for each signal in total.

C. PCA for dimension reduction

Principal Component Analysis (PCA) has been called one of the most valuable methods for dimension reduction. PCA provides a roadmap for how to reduce a complex data set to a lower dimension. PCA can reflect information from the original data matrix by use of a few new data, which are called Principal Component (PC). PCA is used as data filter by using an orthogonal transformation to convert a set of values of uncorrelated variables. The PCA for features fusion is performed using the time-domain parameters and wavelet coefficient introduced in section 3.2.

D. MCS-GMM method

The procedures of this study are shown in Fig. 2. First, we split the original raw signal from bearing and gearbox equipment according to the block length set before. Second, a feature matrix is acquired by feature extraction and PCA-dimension-reduction. The matrix is used for signal blocks clustering under GMM model. Third, after all signal blocks are divided into k groups, a MCS algorithm has been adopted for signal reconstruction. Finally, we can obtain the original signal by putting all signal blocks together.

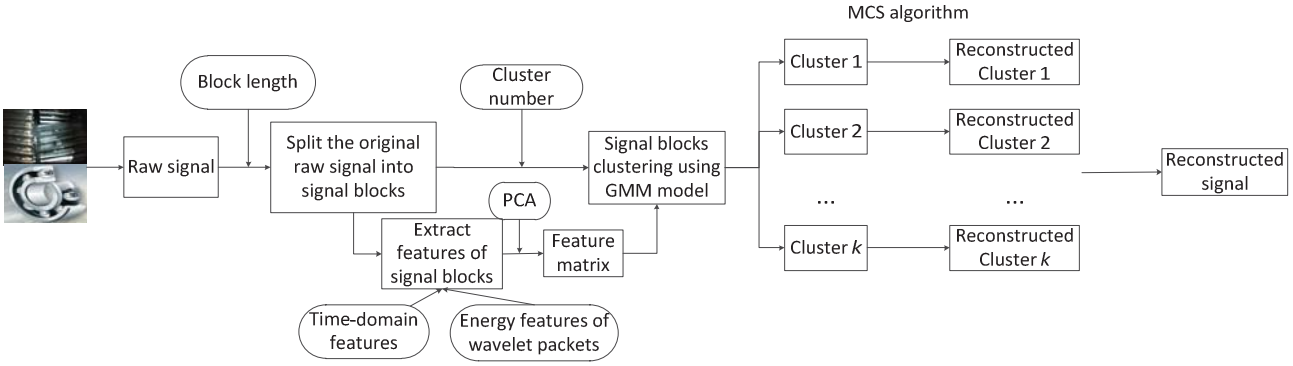


Figure 2. Architecture of the study

IV. SIMULATION

In this simulation, we use L=20 signals of length N=512, each containing 20 spikes created by choosing 20 locations at random and then putting ±1 at these points. These 20 block signals have been divided into 4 groups, in each group there are at maximum 5 spikes not at the same positions. Thus, the 5 signals in the same group perform good correlation between each other. Here we conduct two simulations of MCS for comparison and the number of measurements is set to 100. In one simulation, we perform MCS for all 20 signals to reconstruct each of them. In another simulation, we perform MCS for each group of correlated signals. Fixing the number of measurements with 100, simulations show the reconstruction results taking signal No.1,7,14,20 for example in Fig. 3. Consequently, we can see that MCS for each group reconstruction is much better than MCS for all signals. The mean square error ($MSE=\|\theta'-\theta\|_2/\|\theta\|_2$, where θ' and θ are the estimated and the true coefficient vectors) is adopted here for representing reconstruction error. The mean reconstruction error of the first simulation is 0.2033 and the second simulation is 0.0167. This case indicates that more signals do not mean

smaller reconstruction error using MCS. If the blocks of signal are clustered effectively, MCS for each cluster may get better results of reconstruction. Further, we increase the number of spikes and change the number of measurements to 300. Above two simulations are repeated and 20 blocks of reconstructed signal are obtained respectively. It is found that a similar conclusion has been obtained. Fig. 4 shows the reconstruction results of signal block No 1, 7, 14, 20. The mean reconstruction error of MCS for all signals is 0.2177 and that of MCS for each cluster is 0.0089.

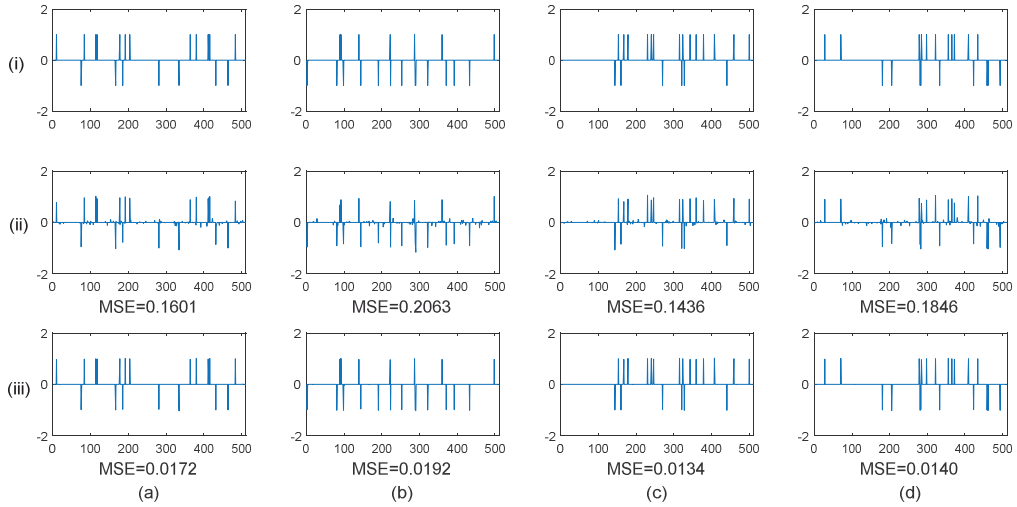


Figure 3. (i) Original signal and (ii) MCS for all signals and (iii) MCS for each cluster reconstruct signals of length 512 (measurement=100, spikes=20) for four signal blocks: (a) signal block 1; (b) signal block 7; (c) signal block 14; (d) signal block 20

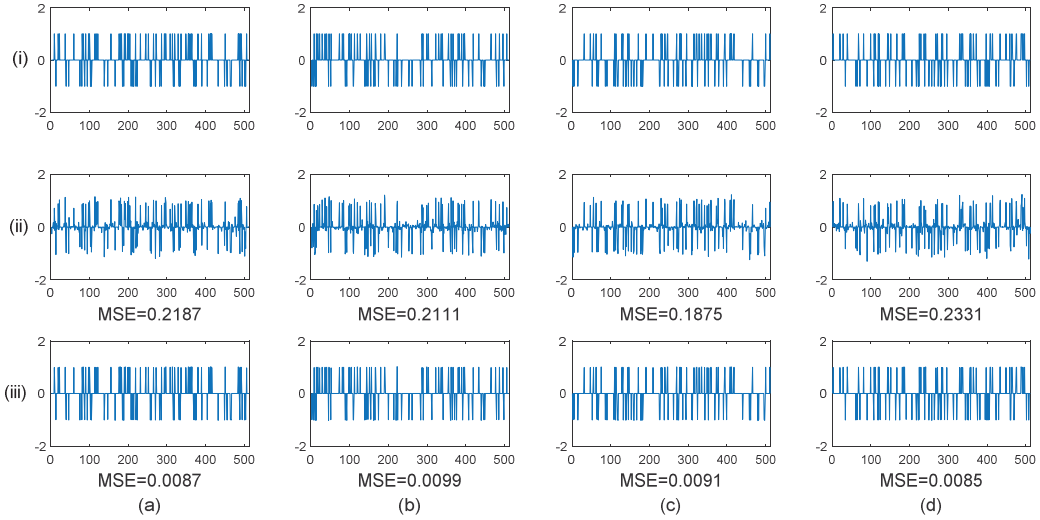


Figure 4. (i) Original signal and (ii) MCS for all signals and (iii) MCS for each cluster reconstruct signals of length 512 (measurement=300, spikes=100) for four signal blocks: (a) signal block 1; (b) signal block 7; (c) signal block 14; (d) signal block 20

V. EXPERIMENT

The proposed MCS-GMM method is validated using the vibration signals form gearbox. Data source are from our gearbox test-bed shown in Fig. 5,6. The number of teeth for gearbox is: 35 teeth on high-speed gear, 64 teeth on intermediate large gear, 19 teeth on intermediate pinion, and 81 teeth on low-speed gear. In this paper, there are four kinds of gear states and Fig. 7 shows the original signal of four states.

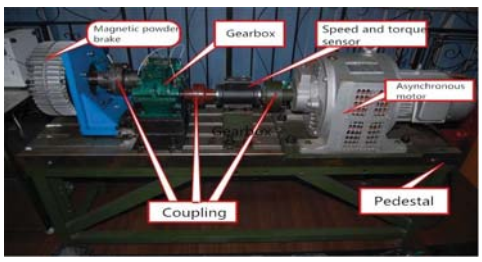


Figure 5. Test-rig of gearbox

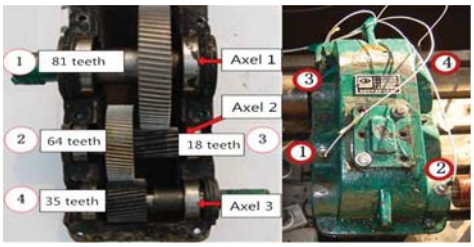
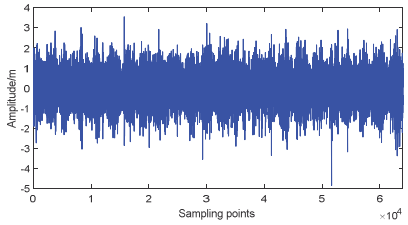
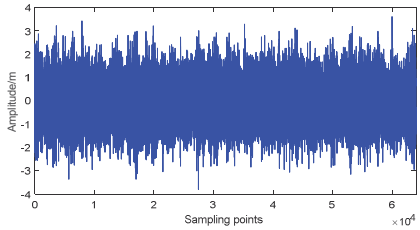


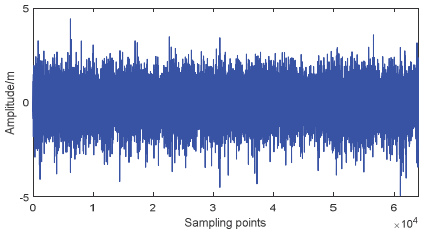
Figure 6. Main parameters of the gearbox and the transducers location



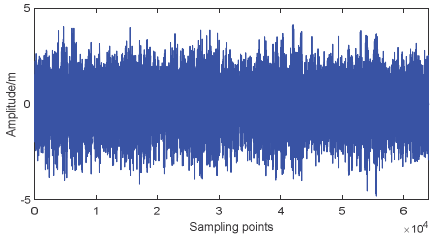
(a) 5mm crack data



(b) 5mm broken tooth data



(c) 8mm crack data



(d) 10mm broken tooth data

Figure 7. Experimental data

In this paper, 64000 signal points of each state were selected for study. The entire experimental data can be seen in Fig. 7. Assuming that signal blocks do not overlap during segmentation, the number of signal blocks is 64000/L when block length L is determined. The block length affects the performance of the algorithm to some extent. When L is small, the number of signal block is large which lead to big workload for clustering algorithm. On the other hand, when L increases, the number of random elements in the measurement matrix will increase continuously. This increases the storage pressure and reduces the matrix construction efficiency. Fig. 8 represents the variation of time consumption with the length of signal block. From Fig. 8, we can see that, when L is less than 80, the time consumption decreases sharply with the increase of block length. The number of blocks for clustering is large which leads to more time consumption. When the signal block length is greater than 80, the time consumption tends to be stable. But when L is increasing further, the complexity of measurement matrix building will lead to another increase of time consumption for entire construction process. Therefore, we have L=80 for minimal time consumption.

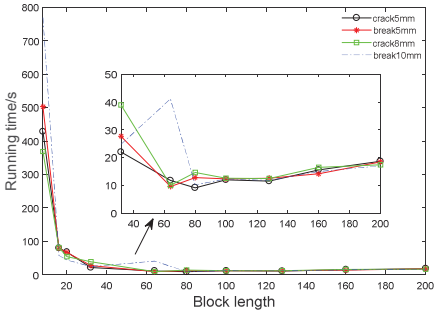
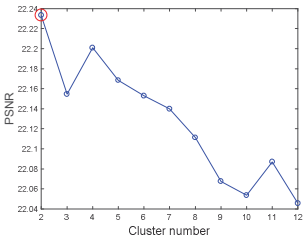


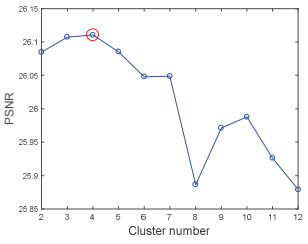
Figure 8. Running time with block length increasing

The cluster number is another important factor affecting the performance of MCS algorithm. When cluster number is large, the amount of signal blocks belonging to the same cluster will decrease. This greatly reduces the efficiency of reconstruction algorithm. Here we choose the peak signal to noise ratio (*PSNR*) as evaluation index. Larger *PSNR* usually indicates better reconstruction performance and the calculation formula of *PSNR* is as follow:

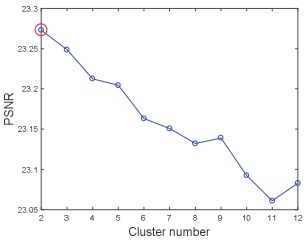
$$PSNR = 10\lg(u_{\max}^2 / (\frac{1}{N}\sum_{i=1}^{N}(u_i - u_i^{'})^2)) \qquad (12)$$



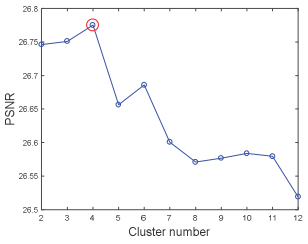
(a) 5mm crack data



(b) 5mm broken tooth data



(c) 8mm crack data



(d) 10mm broken tooth data

Figure 9. The best block number for clustering

where u represents original signal, u' represents the reconstructed signal, u_{max} represents the largest component of the vector u and N is the length of signal u. Taking 4 gearbox signals for example, we can see the variation of *PSNR* with cluster number increasing from Fig. 9. The number of cluster is set among 2~12 according to empirical knowledge and we can easily find the best cluster number for 4 signals as follow: 2, 4, 2, 4.

Fig. 10 demonstrates the reconstruction results with BP, OMP, BCS, MCS, and MCS-GMM. The 10mm broken gearbox data is chosen as example to show the reconstruction performance. We can see that Fig. 10(a) is raw signal and red lines with spikes are reconstruction signals. The MCS reconstructs multi sparse blocks jointly, while BCS directly reconstructs each signal separately. The relative error of reconstruction is calculated by $MSE=\|\theta'-\theta\|_2/\|\theta\|_2$, where θ' and θ are the estimated and the true coefficient vectors, respectively. It is shown that MCS-GMM can well recover the original signal of gearbox compared to other 4 algorithms because red line of MCS-GMM covers blue line better. In addition, the calculation results of Mean Square Error (MSE) also illustrate this inference.

Based on above theory, we investigate the effect of different choices of compression ratio on the reconstruction performance for different algorithms. In this paper, compression ratio denotes $CR=N/M$, where N is the length of original signal and M is the length of reconstructed signal. The block length is fixed with $L=80$ and block number can be decided from 2~12 according to the aforementioned method. The objective of Fig. 11 is to compare the recovery abilities of six algorithms for different CR. The reconstruction accuracy F, denoting $F=1-\|\theta'-\theta\|_2/\|\theta\|_2$, is used as evaluating indicator. When F is big, it is indicated that the more accurate of the reconstruction is. For all the results with different CR and 4 data sets, it is seen that the reconstruction accuracy of MCS-GMM decreases the slowest, which proves the effectiveness of the proposed algorithm. From Fig. 11, we can see that the reconstruction effect of MCS algorithm is only inferior to that of MCS-GMM.

Comparing the Fig. 11(a-f), it is found that the reconstruction of the gear broken data set is optimal, which indicates that the gear broken data sets are relatively simple and beneficial for reconstruction. In addition, we found that there are many factors affecting the performance of MCS-GMM algorithm, such as the block length, block number, compression ratio and initial parameters. Therefore, it is necessary to fully consider all these factors in actual compression and reconstruction process.

VI. CONCLUSION

Compressive sensing methods have the potential to increase the efficiency of wireless data which is especially beneficial to complex vibration signal. Except for the MCS model, we adapt a GMM clustering method to find the best classification among signal blocks. Simulation results show that proper clustering may concentrate signal blocks with similar distribution together to reduce the reconstruction errors. The reconstruction experiments on gearbox data show that our proposed method achieves a substantial performance improvement over previous compressive sensing methods.

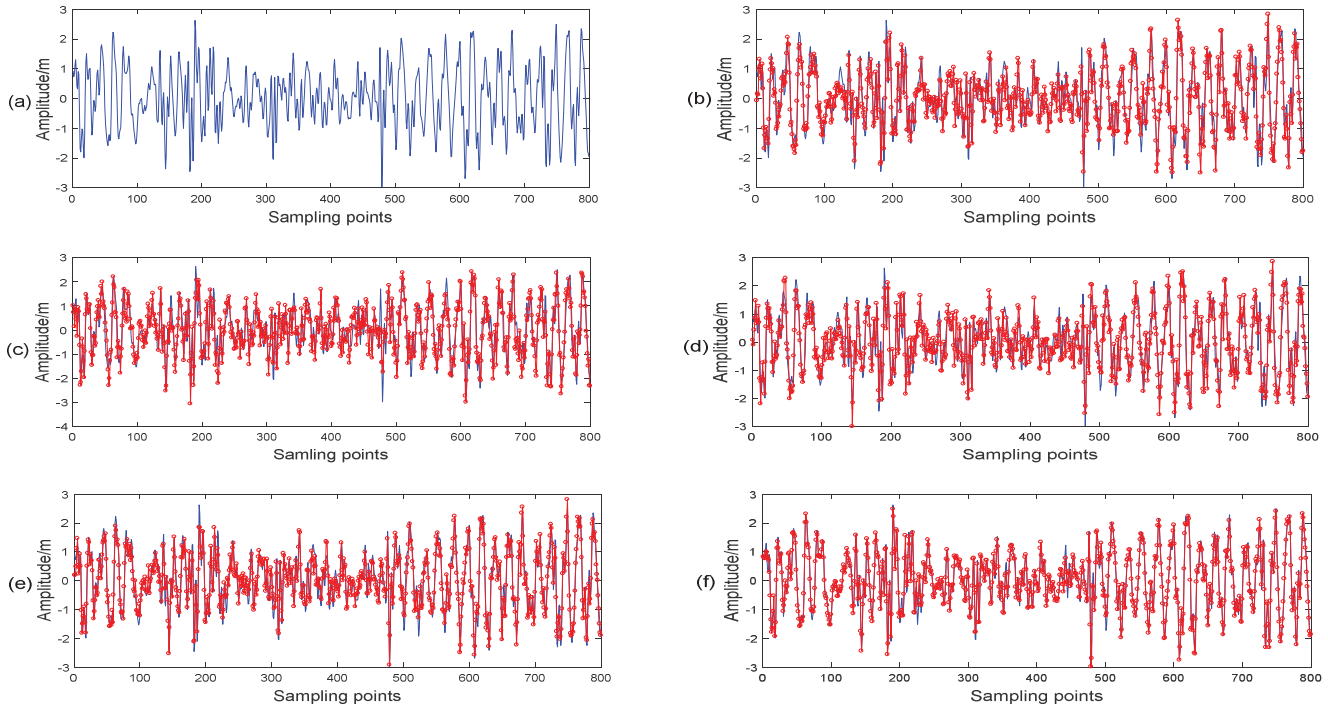
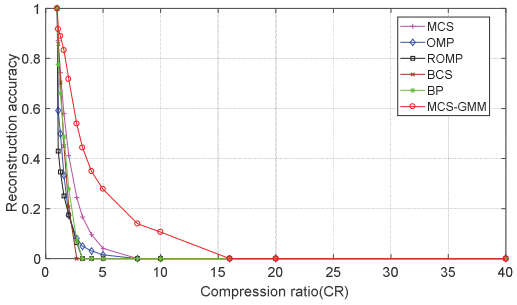
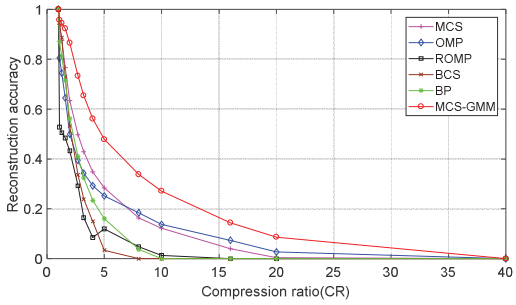


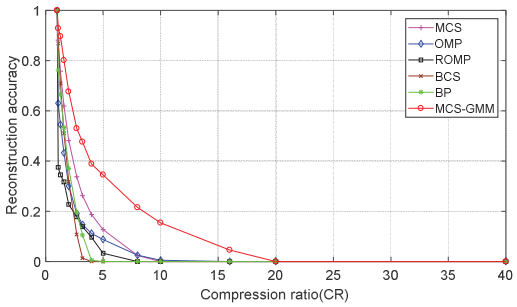
Figure 10. Reconstruction of 10mm broken tooth data under 800rpm speed and 15nm workload (a) original (b) BP mse=0.3075(c) OMP mse=0.3768 (d) BCS mse=0.2910(e) MCS mse=0.2462(f) MCS-GMM mse=0.1206



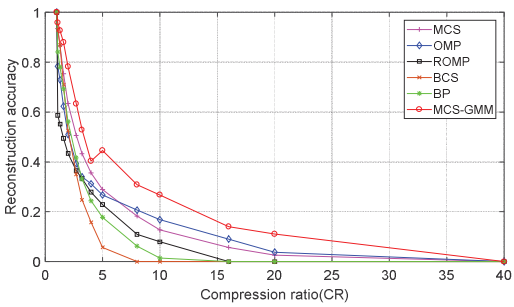
(a) 5mm crack data under 800rpm speed and 10nm workload



(b) 5mm broken tooth data under 800rpm speed and 20nm workload



(c) 8mm crack data under 800rpm speed and_20nm workload



(d) 10mm broken tooth data under 800rpm speed and 15nm workload

Figure 11. Relation of reconstruction accuracy and compression ratio for gearbox data

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under the grants 71871220. I would like to express my gratitude to all those who helped me during the writing of this paper. A special acknowledgment should be shown to my supervisor Professor Xisheng Jia from whose useful instructions I benefited greatly.

REFERENCES

[1] E. J. Candès. “Compressive sampling,” Proceedings of International Congress of Mathematics, Madrid, Spain, 2006, vol. 3, pp. 1433－1452.

[2] E. J. Candès, T. Tao. “Decoding by linear programming,” IEEE Transactions on Information Theory, 2005, vol. 51, pp. 4203－4215.

[3] D. L Donoho. “Compressed sensing,” IEEE Transactions on Information Theory, 2006, vol. 52, pp. 1289－1306.

[4] J. A. Tropp. “Computational methods for sparse solution of linear inverse problems,” Proceedings of the IEEE special issue on applications of sparse representation and compressive sensing, 2009.

[5] X. J. Zhang, P. K. Huang. “SAR ATR based on Bayesian compressive sensing,” Systems Engineering and Electronics, 2013, vol. 35, pp. 40-44.

[6] K. Zhou, C. A. Yuan, X. Qin, et al. “Face recognition based on kernel Bayesian compressive sensing,” JOURNAL OF SHANDONG UNIVERSITY (ENGINEERING SCIENCE), 2016, vol. 46, pp. 74-78.

[7] W. J. Ma. “DOA estimation through Bayesian compressive sensing algorithm,” Harbin Institute of Technology, 2014.

[8] K Luo, J. Q. Li, Z. G. Wang, et al. “Prior-block sparse Bayesian learning algorithm for compressed sensing based ECG recovery,” Chinese Journal of Scientific Instrument, 2014, vol. 35, pp. 1883-1889.

[9] S. Ji, Y. Xue, L. Carin, “Bayesian compressive sensing,” IEEE Trans. Signal Process, 2008, vol. 56, pp. 2346–2356.

[10] R. Torkamani, R. A. Sadeghzadeh. “Bayesian compressive sensing using wavelet based Markov random fields,” Signal Processing: Image Communication, vol. 58 (2017), pp. 65-72.

[11] M. E. Tipping. “Sparse Bayesian learning and the relevance vector machine,” Journal of Machine Learning Research, 2001, vol. 1, pp. 211-244.

[12] J. A. Tropp. “Computational methods for sparse solution of linear inverse problems,” Proceedings of the IEEE special issue on applications of sparse representation and compressive sensing, 2009.

[13] J. A. Tropp, A. C. Gilbert. “Signal recovery from random measurements via orthogonal matching pursuit,” IEEE Transactions on Information Theory, 2007, vol. 53, pp. 4655－4666.

[14] M. S. Yang, C. Y. Lai, C. Y. Lin. “A robust EM clustering algorithm for Gaussian mixture models,” Pattern Recongnition, 2012, vol. 45, pp. 3950-3961.

[15] D. Pollard, “Quantization and the method of k-means,” IEEE Transactions on Information Theory, 1982, vol. 28, pp. 199-205.

[16] K.L. Wu, M.S. Yang “Alternative c-means clustering algorithms,” Pattern Recognition, 2002, vol. 35, pp. 2267-2278.

[17] E. Nowakowska, J. Koronacki, S. Lipovetsky “Clusterability assessment for Gaussian mixture models,” Applied Mathematics and Computation, 2015, vol. 256, pp. 591-601.

[18] N. Dean, T. B. Murphy, G. Downey. “Using unlabelled data to update classification rules with applications in food authenticity studies,” Journal of the Royal Statistical Society, Series C, vol. 55, pp. 1-14.