IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 29, 2020, accepted March 10, 2020, date of publication March 16, 2020, date of current version March 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2981167

Overlap Aware Compressed Signal Classification

MEENU RANI[1], (Student Member, IEEE), SANJAY B. DHOK[1], RAGHAVENDRA B. DESHMUKH[1], AND PUNIT KUMAR[2], (Student Member, IEEE)

[1]Centre for VLSI and Nanotechnology, Visvesvaraya National Institute of Technology, Nagpur 440010, India
[2]Department of Computer Science Engineering, IIITDM, Jabalpur 482005, India

Corresponding author: Meenu Rani (meenubanait@gmail.com)

ABSTRACT Compressed signal processing (CSP) is a branch of compressive sensing (CS), which gives a direction to solve a class of signal processing problems directly from the compressive measurements of a signal. CSP utilizes the information preserved in the compressive measurements of a signal to solve certain inference problems like: classification, detection, and estimation, without reconstructing the original signal. It further simplifies the signal processing compared to conventional CS by omitting their complex reconstruction stage. This, in turn, reduces the implementation complexity of signal processing systems. This paper investigates the performance of CSP for classification application. After extracting the features from compressive measurements, these features or the data instances are used for classification purpose. Through experimental analysis, it has been found that as the CS undersampling factor is increased, the overlapping among the data instances predominates. This results in a complex decision boundary, which in turn degrades the classification accuracy at higher undersampling factors. To overcome the above issue, this paper proposes the use of a machine learning method known as overlap aware learning along with CSP. This generates a smoother decision boundary and hence improves the classification accuracy at higher undersampling factors. The simulation results show the trend of improved classification accuracy using the proposed method. An analysis of the proposed method has been done on different datasets and based on run-time complexity and complexity vs gain analysis to verify the effectiveness of proposed method.

INDEX TERMS Compressive sensing, compressed signal processing, decision tree classifier, feature extraction, overlap aware learning, random demodulator.

I. INTRODUCTION

Compressive sensing (CS) is a signal acquisition technique, which works on the random sensing principle. The output random samples, also called the compressive measurements, are extracted from the signal at a rate proportional to its sparsity. These measurements are independent of the highest frequency component present in the signal, as opposed to the Nyquist sensing mechanism. The various acquisition strategies supporting CS mechanism are: random demodulator, modulated wideband converter, random modulator pre-integrator, compressive multiplexer, etc. [1]. Also, the original signal can be faithfully recovered from the fewer random measurements by using CS reconstruction algorithms. The various CS reconstruction algorithms that are available, come under: convex optimization, greedy algorithms, thresholding algorithms, combinatorial, non-convex and Bayesian approaches. Incubating all these features, the applications of CS covers a wide range of signal processing tasks like: compressive imaging, video processing, communication networks, biomedical signal processing, pattern recognition, speech processing, etc. There is a rich literature available regarding all these aspects of CS, specially, on the reconstruction aspect, because, CS reconstruction algorithms are very computationally extensive, consume a lot of power and their hardware implementation is an extremely laborious task. So, performance optimization of these algorithms is an active area of research [1]–[8].

Considering this limitation of CS, researchers have also put efforts to solve signal processing tasks without reconstruction. This seemed to be possible due to the fact that the compressive measurements preserve the signal structure, which also allows reconstruction. In this regard, the way-out has been proposed in [9]–[11], to solve the signal processing problems like: classification, detection, and estimation, directly from compressive measurements. This branch of

The associate editor coordinating the review of this manuscript and approving it for publication was Quan Zou.

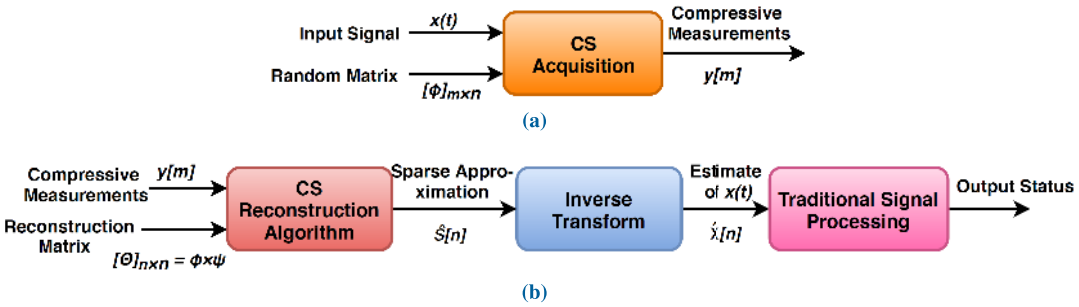


FIGURE 1. Conventional CS scenario for solving an inference task: (a) signal acquisition and (b) solving the requisite signal processing task using traditional methods from the reconstructed signal.

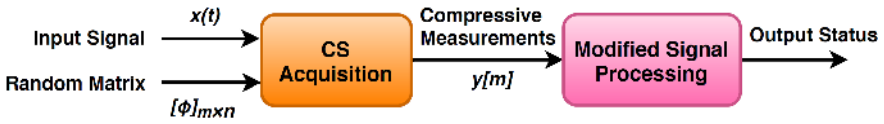


FIGURE 2. A CSP scenario to solve an inference task.

CS, in which, the signal processing problems are solved directly in compressed domain, is known by the name of compressed signal processing (CSP). It offers advantages like reduced complexity, eases hardware implementation and saves power as well as time [12]. The CSP is really helpful in the power-constrained environments like wireless sensor nodes (WSN), where if, only some inference problem needs to be solved. Fig. 1 shows the steps involved in processing signals using the conventional CS method for solving an inference problem, and the corresponding CSP scenario is shown in Fig. 2. A stage-wise comparison of these two scenarios clearly indicates that the significant improvement can be achieved by using CSP for solving the inference tasks compared to the conventional CS mechanism. CSP can also support real-time monitoring in WSN systems by omitting the complex reconstruction stage, as well as the need to transmit compressive measurements. Only the final status of the system under supervision can be transmitted by solving the inference problem from compressive measurements directly on node. In this scenario, the major challenge that needs to be addressed for the power-constrained environment is that the inference problem for the system must be solved from minimum number of measurements in order to minimize the power consumption. This can be feasible if the system performs satisfactorily even with highly undersampled input signal [12]–[17].

This paper investigates the performance of CSP for the classification application. It has been found that the classification performance degrades with the increasing undersampling and one of the reasons behind this is the increasing overlapping among the data instances used for classification. This paper proposes a method to address the above-mentioned challenge by combining CSP with a technique from machine learning known as overlap aware learning. The proposed method performs superior to the conventional CSP, without putting any extra burden on the signal acquisition part. The performance of the proposed method has been tested using simulations done in MATLAB 2017a, on the vibration dataset taken from laboratory for dynamics of machines and structures (Ladisk) [18]. The further organization of this paper is as follows: section II discusses the conventional CSP technique along with the related background, the working scenario in detail for bearing condition monitoring, and its analysis. Section III describes the proposed solution along with the motivation and problem description. Section IV presents the supporting results and related discussion. At last, section V analyzes the robustness of proposed scheme by verifying its applicability on different datasets and also by comparing its performance based on run-time complexity with the conventional techniques. To highlight the achieved gain an analysis has been presented between increased accuracy and the differential run-time complexity in this section.

II. CONVENTIONAL CSP TECHNIQUE

In this section, the working scenario of conventional CSP scheme has been presented in detail along with its background and the class separability analysis of the feature vectors obtained from compressive measurements.

A. BACKGROUND

This subsection presents the background related to CS acquisition and CSP, which are the main focus of this paper. Let, $\boldsymbol{x} \in \mathbb{R}^n$ or $\mathbb{C}^n$ be an input signal, $\boldsymbol{\varphi} \in \mathbb{R}^{m\times n}$ or $\mathbb{C}^{m\times n}$ be a random measurement matrix and $\boldsymbol{y} \in \mathbb{R}^m$ or $\mathbb{C}^m$ be the output measurement vector, where, $m \ll n$ and u be the undersampling factor, which is given by $u = n/m$. The mathematical model for generating CS measurements corresponding to an input signal is given by (1).

$$\boldsymbol{y} = \boldsymbol{\varphi x}, \tag{1}$$

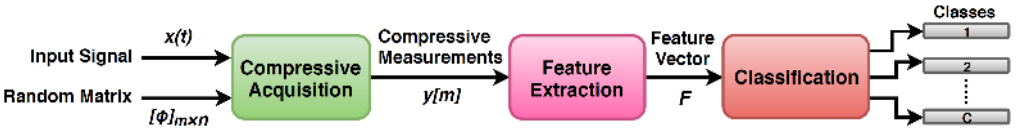


FIGURE 3. The conventional CSP scenario for bearing condition monitoring.

For gathering enough information about the original signal, the minimum number of CS measurements to be taken depends upon its sparsity. Their number can be further reduced by incorporating incoherence among the measurement and sparsifying bases of signal, where sparsifying basis is the basis, using which the signal can be represented by fewer significant components. CS basically uses random matrices as the measurement basis, which are incoherent with any other basis. A relation between number of measurements (m), sparsity (k) and coherence (μ) for perfect recovery is given by (2), where c is a positive constant [5], [6], [19].

$$m \geq c\mu^2 k \log n. \tag{2}$$

Although the original signal can always be reconstructed back from compressive measurements, but there are some signal processing problems which can be directly solved from compressive measurements, without reconstructing the original signal back. This directly reduces the reconstruction cost for solving the problems like classification, estimation and detection. Though, there are different ways by which the inference task can be solved from compressive measurements. But, for simplicity, the following discussion has been presented to understood the basic idea behind CSP mechanism:

Considering the problem of detection from the compressive measurements. For this, the exact reconstructed coefficients are not required. But the aim is to collect sufficient statistics about the components having highest contribution. The collected statistics can then be compared against certain threshold to detect the presence or absence of desired components in the signal. Consider the example of distinguishing between two hypotheses:

$$\mathcal{H}_0 : \boldsymbol{y} = \boldsymbol{\varphi N}$$
$$\mathcal{H}_1 : \boldsymbol{y} = \boldsymbol{\varphi}(\boldsymbol{x} + \boldsymbol{N})$$

where, $\boldsymbol{N} \sim \boldsymbol{N}(0, \sigma^2 I_n)$, represents the i.i.d. Gaussian noise. Here, hypothesis $\mathcal{H}_1$ represents the presence of signal x in the noise that has to be detected. Let F_r and D_r denotes the *false alarm rate* and *detection rate* respectively, then

$$F_r = P(\mathcal{H}_1 \text{ choosen when } \mathcal{H}_0 \text{ is true})$$
$$D_r = P(\mathcal{H}_1 \text{ choosen when } \mathcal{H}_1 \text{ is true})$$

Then, the Neyman-Pearson detector is derived to obtain a decision rule, which maximizes D_r. The derived decision rule d is given by:

$$d := \boldsymbol{y}^T(\boldsymbol{\varphi\varphi}^T)^{-1}\boldsymbol{\varphi x}. \tag{3}$$

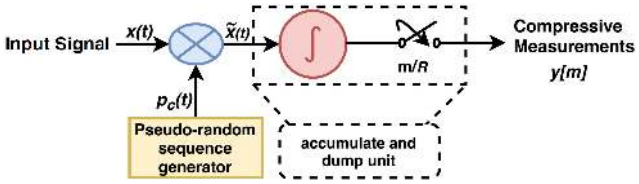


FIGURE 4. Acquisition of input signal using RD technique of CS.

This can be used as a sufficient statistics for distinguishing between the two hypotheses. For this, a threshold needs be derived and then the detection rule can be compared against this threshold to detect the presence of desired signal [9]–[16].

B. WORKING SCENARIO OF CONVENTIONAL CSP TECHNIQUE

This potion describes in detail the steps of conventional CSP mechanism generally used for classification application, shown in Fig. 3. Here, this setup has been used for classifying the bearing condition. The simulations for this purpose are done on the bearing vibration data set taken from Ladisk [18].

1) SIGNAL ACQUISITION USING CS

The first stage is the compressive acquisition stage. The CS acquisition of input signal is done using RD technique, which was proposed by Laska *et al.* in 2007 [20]. This technique is chosen because of its relatively simpler and easier to implement architecture. The first step of RD introduces randomness in the input signal $\boldsymbol{x}(t)$ by multiplying it with a pseudorandom sequence of $+/-1$s, as shown in Fig. 4. The randomized signal $\tilde{\boldsymbol{x}}(t)$ is then passed through an integrator, which accumulates the signal and also serves as a low pass filter. The low frequency signal so obtained can now be sampled at much lower rate. The final output are the compressive measurements $\boldsymbol{y}[m]$, which represents a unique signature of input signal in frequency domain.

The operation of RD in matrix form is represented by (4) and (5). Here, the matrix $\boldsymbol{P}$ is an $n \times n$ diagonal matrix of pseudorandom sequence. The multiplication of input signal $\boldsymbol{x}[n]$ with matrix $\boldsymbol{P}$ gives the randomized signal $\tilde{\boldsymbol{x}}[n]$. The operation of accumulate and dump unit is represented by an $m \times n$ matrix $\boldsymbol{H}$. This matrix is responsible for accumulating the samples of randomized signal and also for undersampling the signal. The number of samples to be accumulated are decided by the number of ones in a row of $\boldsymbol{H}$, which is given by the $R = \lfloor n/m \rfloor$. After multiplying $\tilde{\boldsymbol{x}}[n]$ with this matrix, the final output obtained is $\boldsymbol{y}[m]$, as given in (5). Also,

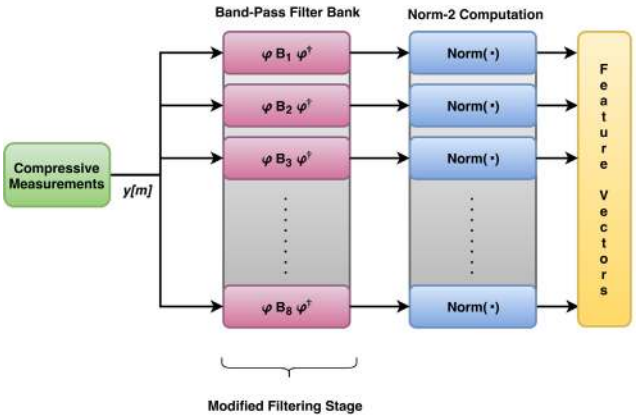


FIGURE 5. Process of feature extraction from compressive measurements.

the product $\boldsymbol{P} \times \boldsymbol{H}$ is equivalent to the CS measurement matrix $\boldsymbol{\varphi}$ [20], [21].

$$\boldsymbol{P} = \begin{bmatrix} p_1 & & \\ & \ddots & \\ & & p_n \end{bmatrix}; \quad \boldsymbol{H} = \begin{bmatrix} 11\cdots 00\cdots 00\cdots \\ 00\cdots 11\cdots 00\cdots \\ 00\cdots 00\cdots 11\cdots \end{bmatrix} \tag{4}$$

$$\left.\begin{aligned} \tilde{\boldsymbol{x}} &= \boldsymbol{P}\boldsymbol{x} \\ \boldsymbol{y} &= \boldsymbol{H}\tilde{\boldsymbol{x}} = \boldsymbol{\varphi}\boldsymbol{x} \\ \boldsymbol{\varphi} &= \boldsymbol{H}\boldsymbol{P} \end{aligned}\right\}. \tag{5}$$

In the next stage, the features are extracted from these measurements.

2) FEATURE EXTRACTION FROM COMPRESSIVE MEASUREMENTS

Although, classification can directly be done from compressive measurements, but this method has certain drawbacks like: (i). It requires comparatively large amount of compressive measurements to train and test a classifier. The problem becomes more complicated in case of multiclass classification scenario. In such cases, it takes much longer to train and test a classifier using k-fold cross validation. (ii). The classification accuracy obtained is much lower if the classification is directly done on compressive measurements. In order to overcome the above shortcomings, the features are extracted from compressive measurements. This step reduces the training and testing time and also improves the classification accuracy significantly. The compressive measurements basically represent a unique signature of the signal contents. This property can also be utilized to extract the features from compressive measurements. In this paper, energy retained by the compressive measurements is used as a feature to distinguish among the different bearing conditions, because each condition or class is characterized by different energies. The procedure of extracting features from compressive measurements is shown in Fig. 5. To extract this type of features, the segment of compressive measurements is first passed through the set of eight bandpass filters (BPFs), which divides the signal bandwidth into eight equal bands. To extract information from compressive measurements, the conventional filters are modified with the help of same random/pseudorandom matrix $\boldsymbol{\varphi}$, as per (6), where $\boldsymbol{B}_i$ denotes the i^{th} conventional BPF and $\hat{\boldsymbol{B}}_i$ is the i^{th} modified BPF.

$$\hat{\boldsymbol{B}}_i = \boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{\varphi}^{\dagger}. \tag{6}$$

Let z_i be the output of i_{th} conventional filter and $\hat{z}_i$ be the output of i_{th} modified filter, then the above formulation can be derived as follows:

$$\begin{aligned} \hat{z}_i &= \boldsymbol{\varphi} z_i \\ \hat{\boldsymbol{B}}_i\boldsymbol{y} &= \boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{x} \\ \hat{\boldsymbol{B}}_i\boldsymbol{\varphi}\boldsymbol{x} &= \boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{x} \\ \hat{\boldsymbol{B}}_i\boldsymbol{\varphi} &= \boldsymbol{\varphi}\boldsymbol{B}_i \end{aligned}$$

post-multiplication with $\boldsymbol{\varphi}^T(\boldsymbol{\varphi}\boldsymbol{\varphi}^T)^{-1}$

$$\begin{aligned} \hat{\boldsymbol{B}}_i\boldsymbol{\varphi}\boldsymbol{\varphi}^T(\boldsymbol{\varphi}\boldsymbol{\varphi}^T)^{-1} &= \boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{\varphi}^T(\boldsymbol{\varphi}\boldsymbol{\varphi}^T)^{-1} \\ \hat{\boldsymbol{B}}_i &= \boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{\varphi}^{\dagger} \end{aligned}$$

where, $\boldsymbol{y}$ is the set of compressive measurements which are also the input of modified filters here and $\boldsymbol{x}$ is the corresponding set of compressive measurements, also the input of conventional filters. The filtered signal from the eight BPFs are then passed through the set of eight norm-2 computing functions. This generates a single feature vector of dimensionality eight corresponding to a segment of compressive measurements. The process is repeated for all the segments to generate a set of feature vectors. For obtaining sets of feature vectors corresponding to different undersampling factors, the above process is repeated for each undersampling factor.

3) CLASSIFICATION

The feature vectors obtained in previous step are then used for classifying the bearing condition. The classification is done with the help of decision tree classifier, which is used with its default parameters and a fixed random state. The classifier is trained and tested using 10-fold cross validation.

C. SEPARABILITY ANALYSIS OF OBTAINED DATA

This part of paper presents the separability analysis of the features obtained in previous subsection. This analysis has been done to identify the complexity of classification problem in hand. Different measures have been calculated to analyze the class separability of these data instances, as described below:

- *Fisher's discriminant ratio*: In case of two class classification problem and for one feature dimension, the Fisher's discriminant ratio $F1$ is given by (7), where, M_1, M_2 are the means and σ_1^2, σ_2^2 are the variances of given classes.

$$F1 = \frac{(M_1 - M_2)^2}{\sigma_1^2 + \sigma_2^2}. \tag{7}$$

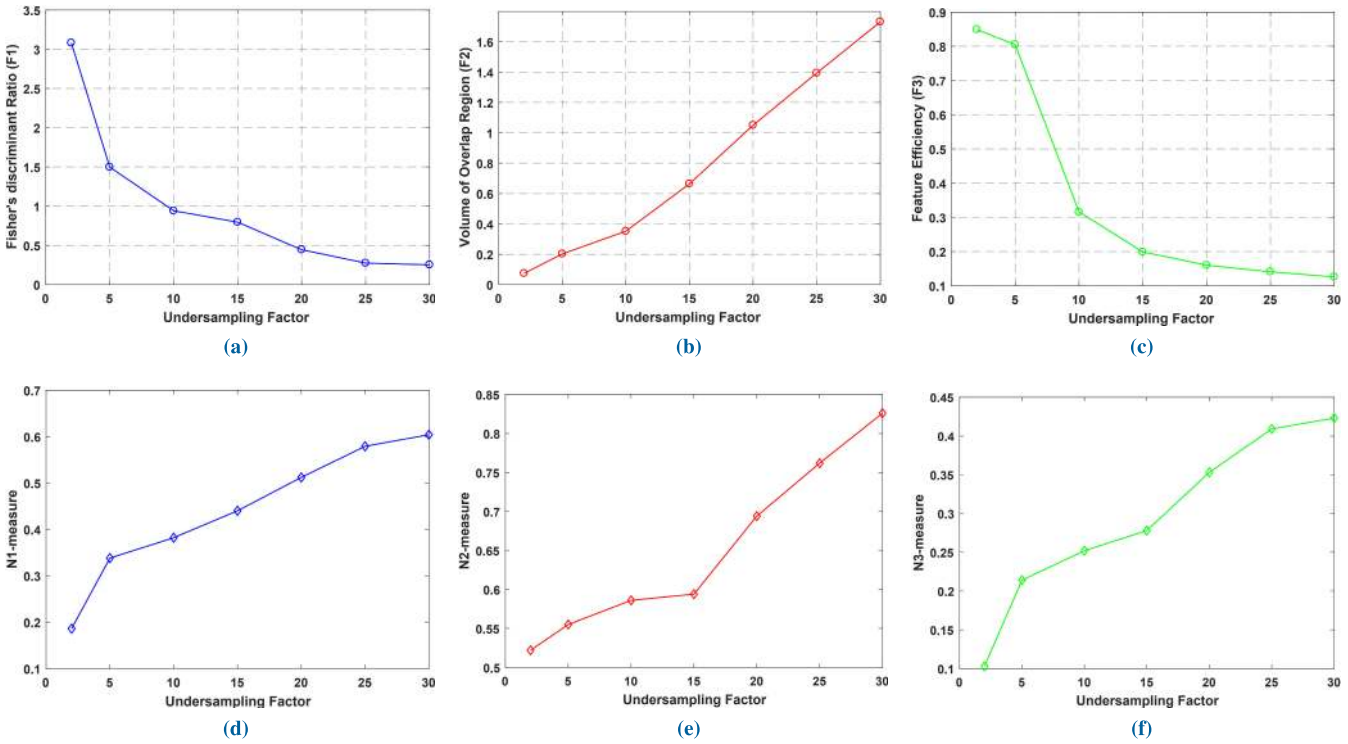


FIGURE 6. Measures for separability analysis of data: (a) Fisher's discriminant ratio (F1), (b) Volume of overlap region (F2), (c) Feature efficiency (F3), (d) N1-measure, (e) N2-measure, and (f) N3-measure.

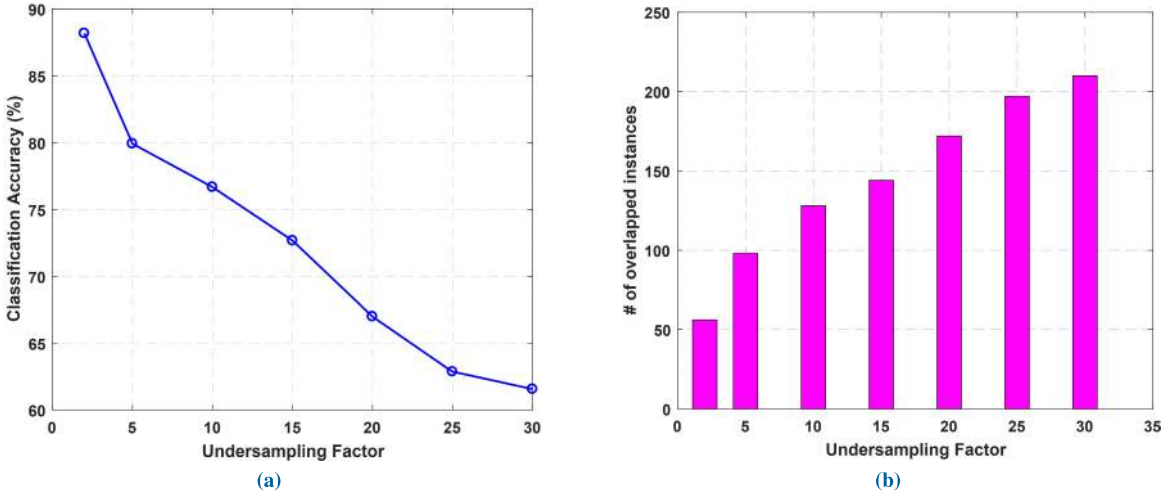


FIGURE 7. Effect of increasing undersampling on the: (a) classifier accuracy, (b) overlapping among the features obtained from compressive measurements.

In case of multidimentional feature vectors, the feature dimension having maximum $F1$ value is used determine the problem complexity. This formulation can be extended to the case of multiclass classification. A higher value of $F1$ indicates the higher separability. In case of the classification problem considered here, the variation of $F1$ with undersampling factor u is shown in Fig. 6(a). This shows that as the undersampling factor is increased, the Fisher's discriminant ratio decreases. This indicates the trend of increased overlapping among the data instances with undersampling factor.

- *Volume of overlap region:* This measure determines the overlap between the tails of conditional distributions of two classes. For the i^{th} feature dimension, this measure can be obtained by first computing the $\min(f_i, c_j)$ and $\max(f_i, c_j)$ for feature f_i and class c_j, and then multiplying their ratio for all feature dimensions as per (8), as shown at the bottom of the next page.

A lower value of $F2$ indicates the higher separability. For the feature vectors obtained in the previous subsection, the $F2$ measure is plotted for different undersampling factors, as shown in Fig. 6(b). As the undersampling factor is increased, the value of $F2$ increases, which shows that class separability decreases as the CS undersampling is increased.

- *Feature efficiency:* The feature efficiency is defined as the fraction of all other points discriminable by a feature. The maximum value of the feature efficiency calculated over all the feature dimensions is used to represent the measure $F3$. Therefore, a higher value of $F3$ indicates higher separability. A plot of this measure for the classification problem in hand is shown in Fig. 6(c). This shows that with the increasing undersampling factor, the maximum feature efficiency decreases, which indicates that the overlapping among the instances increases with the increasing undersampling factor.
- $N1-$*measure:* This method is based on forming a minimum spanning tree (MST) connecting all data instances to their nearest neighbors. The ratio of the number of edges connected to opposite classes to the total number of edges in MST is known as $N1-$measure. A lower value of $N1-$measure indicates a higher separability of classes. The $N1-$measure for the above case is shown in Fig. 6(d). This measure has an increasing trend, which indicates that with the increasing undersampling factor, the data separability decreases.
- $N2-$*measure:* Another similar measure is $N2-$measure, which makes use of euclidean distances between the nearest neighbors over all data instances instead of MST. Then, this measure is computed by taking the ratio of average of all intra-class nearest neighbor distances to the average of all inter-class nearest neighbor distances. This measure should also have a lower value for higher separability. For the features generated from compressive measurements, the values of $N2-$measure increases with increasing undersampling factor, which indicates separability decreases as the undersampling factor is increased, as shown in Fig. 6(e).
- $N3-$*measure:* This measure accounts for the proximity of data instances in opposite classes. Smaller the $N3-$measure, higher will be the separability. For the problem addressed here, the values of this measure also increases with the undersampling factor, as shown in Fig. 6(f), which shows lower separability at higher undersampling factors.

Based on all these measures, the separability analysis of the data under consideration concludes that with the increasing undersampling factor, all of these measures showed the trend of reduced separability.

III. PROPOSED CSP BASED SCHEME

This section presents the motivation for the work along with the problem description. Then the proposed solution for the problem in hand is presented in the later part.

A. MOTIVATION AND PROBLEM DESCRIPTION

The analysis presented in previous section showed that with increasing undersampling, the data separability decreases or in other words, the class overlapping increases. Here, some observations have been done on the classification results of this data. This investigation revealed that as the CS undersampling factor is increased, the classification accuracy decreases, as shown in Fig. 7(a). One of the reasons behind this is that the information preserved by compressive measurements degrades at higher undersampling factors [1]. A deeper study of this has uncovered the fact that with the increasing undersampling factor, the feature sets or the instances belonging to different classes start overlapping. The resulted graph showing this trend is plotted in Fig. 7(b).

At lower undersampling factors, the overlapped instances are also less. But, as the undersampling factor is increased, there is a rapid increase in the number of instances getting overlapped. This drastically hampers the classification accuracy at higher undersampling factors, as shown in Fig. 7(a). This happens because the classifier is trained on the class overlapped data, which is obtained after higher undersampling and in order to achieve high training accuracy on this data, the classifier tries to accommodate all the overlapped instances for learning the decision boundary. Due to this, a complex decision boundary is learned on the data, which leads to its over-fitting. This, in turn, decreases the testing accuracy, as smoother boundaries are more preferable than complex ones [22], [23]. So, there is a need to search the way out to tackle the above problem.

B. WORKING SCENARIO OF PROPOSED CSP BASED SCHEME

In this part, the proposed solution to tackle the above problem is described in detail. The authors propose the use of overlap aware learning along with CSP to handle this overlapping issue and also to improve classification accuracy at higher undersampling factors. The proposed scheme for bearing condition monitoring is shown in Fig. 8. The first two steps of this scheme i.e., compressive acquisition and feature extraction are already described in section II. After feature extraction, the procedure and application of overlap

$$F2 = \prod_i \frac{MIN(\max(f_i, c_1), \max(f_i, c_2)) - MAX(\min(f_i, c_1), \min(f_i, c_2))}{MAX(\max(f_i, c_1), \max(f_i, c_2)) - MIN(\min(f_i, c_1), \min(f_i, c_2))}. \quad (8)$$

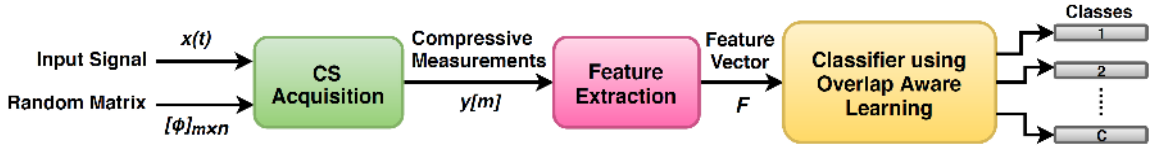


FIGURE 8. The proposed scheme for improving the performance of CSP for classification.

FIGURE 9. Overlap aware learning scheme.

aware learning on these features are described in detail in this section.

1) CLASSIFICATION USING OVERLAP AWARE LEARNING

The classification problem that needs to be handled here is the multiclass classification problem. Along with this, the other problem that has been tackled in this paper is the problem of overlapped instances that arose in the compressive measurements with the increasing undersampling factor. The severity of this problem is defined on the basis of number of instances that actually belong to different classes and have minor differences in their attributes. Identification of these attributes is necessary to distinguish their classes. Such instances seem to belong to more than one class because it is very difficult to capture the minor differences present in their attributes. This problem degrades the classifier performance [24].

This paper has proposed the use of overlap aware learning along with CSP for overcoming the problem of overlapped instances and for improving the classifier performance. The overlap aware learning scheme is shown in Fig. 9. Here, the refined dataset is obtained from the training dataset by finding and handling the overlapped instances. This has been achieved here by making use of a decision tree classifier. The main parameters used for this classifier along with a fixed random state in the experiments are as:

- Criterion = "gini": function used to measure the splitting quality.
- Splitter = "best": strategy used to decide about the split at each node.
- MaxDepth = None: setting to expand the nodes until all the obtained leaves become pure.
- MinSamplesSplit = 2: sets that a minimum of 2 samples are required to split an internal node.
- MinSamplesLeaf = 1: sets that a minimum of 1 sample is required to become a leaf node.

To correctly classify an instance using this classifier, the measure of subconcept ratio (SCR) is used to partition the feature space. Here, the nodes of decision tree represent the concepts and the leaves represents the subconcepts. Then, SCR is given by (9), where numerator represents the number of instances belonging to different classes in a

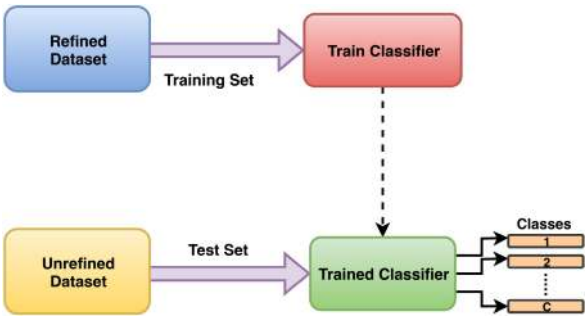


FIGURE 10. Classification using extracted features.

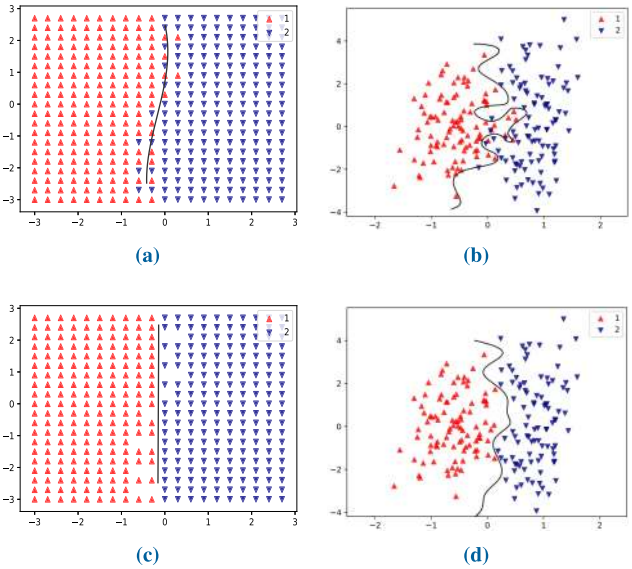


FIGURE 11. A visualization of effect of overlap aware learning on artificial data having overlapped instances: (a), (b) complex boundaries obtained using conventional learning, and (c), (d) simpler decision boundaries obtained using overlap aware learning.

subconcept and k is the total number of instances in that subconcept.

$$SCR(x) = \frac{|\, x'' \in \text{subconcept}(x) \wedge \text{class}(x'') \neq \text{class}(x) \,|}{k}. \tag{9}$$

For finding the overlapped instances using this classifier, the classifier is first trained up to an intermediate level. The output of classifier at this time contains overlapped instances. To identify those instances, the classifier is again trained using the same data up to the end level. At this time output of

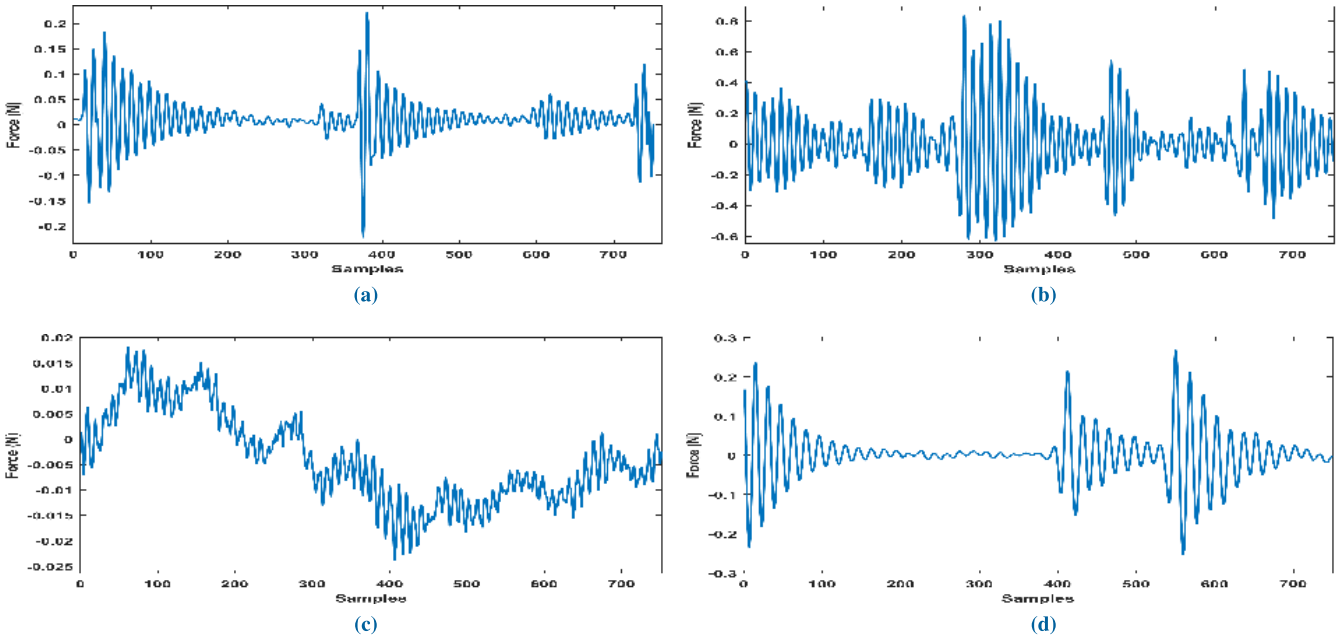


FIGURE 12. Four conditions of bearing vibration signal taken from Ladisk: (a) signal with axial fault, (b) signal with contamination in lubricant fault, (c) new bearing vibration signal and (d) signal with radial fault.

classifier gives the exact class of each instance. A comparison of outputs of both the experiments gives the overlapped instances. Keeping all other conditions same, the above experiments are repeated for different undersampling factors. This gives the number of overlapped instances for different undersampling factors. The handling of these overlapped instances is done by discarding these instances. This gives the refined dataset, which is then used to train the classifier using 10-fold cross-validation, as shown in Fig. 10. Now, the decision boundary learned on this refined dataset is a smoother boundary, which enhances the testing performance of classifier [23]–[25]. To visualize the above discussion, a demonstration has been presented on two artificial datasets with different extent of overlapping using the decision tree classifier, as shown in Fig. 11. Both of these are two-class classification problems. Depending upon the extent of overlapping between the classes, the boundary obtained is more complicated. On these datasets, a non-linear decision tree classifier is trained using the conventional method and the resulted decision boundaries are shown in Fig. 11(a) and (b). Then, the overlapped instances are identified with decision tree classifier using the same settings like that in the main experiment. All the identified instances are then removed to obtain the simpler decision boundaries, as shown in Fig. 11(c) and (d). The classifier trained using this method is used for classification purpose. On the main dataset, for comparing the performance of proposed method, the classifier has been trained using both the methods, i.e., conventional and proposed. The trained classifier is then tested on the unrefined main dataset. A comparison of the classification accuracies obtained is presented in the next section.

IV. RESULTS AND DISCUSSION

The applicability of proposed scheme has been verified using the simulations done on bearing vibration dataset taken from Ladisk [18]. The simulations have been performed using MATLAB 2017a. In the initial part of simulations, the compressive measurements have been obtained from the raw vibration dataset and then the features have been extracted from these measurements. The second part of simulations is about processing these features for finding and handling the overlapped instances. After this, a refined dataset has been obtained from the feature vectors generated in previous step. On this refined dataset, the decision tree classifier is trained using 10-fold cross-validation.

For simplicity, only four types of bearing conditions, as shown in Fig. 12, have been studied for evaluating the performance of proposed scheme. The dataset is processed by dividing it into segments of length 500 each, which is equivalent to the 0.1 second epoch of the signal. For obtaining the compressive measurements, the undersampling factors used are 2, 5, 10, 15, 20, 25, and 30. The corresponding compressive measurements obtained are of length 250, 100, 50, 25, 20, and 16, respectively. All the required specifications for experimentation are listed in Table 1. For the feature extraction stage, the coefficients of eight BPFs are used to obtain the eight filter matrices, $\boldsymbol{B}_i$, of size 500×500. For this, the filter coefficients are arranged in a row and zeros are appended after the coefficients to make the row size as 500. The subsequent rows of the filter matrix are then generated from this row by using circular right shift operation. Now, in order to use these filter matrices and to match the filter dimensions with the segment of compressive

TABLE 1. The parameters or the specifications used or obtained in experiments.

Specification or Parameter	Value
Number of bearing conditions under study	4
Number of samples in each condition	200000
Total signal length	800000
Length of each segment	500
Undersampling factors used	2, 5, 10, 15, 20, 25, 30
Corresponding measurement matrix dimensions	250×500, 100×500, 50×500, 25×500, 20×500, 16×500
Corresponding compressive measurements size	250, 100, 50, 25, 20, 16
Number of filter matrices	8
Each filter matrix ($\boldsymbol{B}_i$) dimension	500×500
Modified filter matrix ($\hat{\boldsymbol{B}}_i$)	$\boldsymbol{\varphi}\boldsymbol{B}_i\boldsymbol{\varphi}^{-1}$
Type of norm used for extracting energy from features	$\|x\|_2 = \sqrt[2]{\sum_i \mid x_i \mid^2}$
Feature set dimension	1600×8
Classifier used	Decision tree classifier
Classifier training method	10-fold cross-validation

measurements, the modified filter matrices $\hat{\boldsymbol{B}}_i$ are obtained from $\boldsymbol{B}_i$s, as per (6). After multiplying $\hat{\boldsymbol{B}}_i$ with a segment of compressive measurements, their norm-2 is computed to obtain a single feature vector of dimensionality eight as per Fig. 5. For the total signal length taken here, the overall dimensionality of the feature vector obtained is 1600×8.

From these feature vectors, a refined dataset is obtained by finding and handling the overlapped instances. Then, the decision tree classifier is trained on this dataset. To partition the feature space using this classifier, the number of instances in the subconcept taken are proportional to the undersampling factor used. This gives the advantage of using same random state for all the experiments while achieving the desired performance. The classifier is first trained up to an intermediate level which is selected as level four and then till end. The difference between the instances obtained at two levels helps in identifying the overlapped instances. The identified overlapped instances are then discarded to obtain a smoother decision boundary. The trained classifier is then tested on the unrefined dataset. For comparison purpose, the classifier is trained and tested using conventional learning also. The accuracies obtained for these two cases for different undersampling factors are presented in Table 2. At lower undersampling factors like 2 and 5, the number of overlapped instances are less, but at higher undersampling factors like 10, 15, 20, 25, and 30 the number of overlapped instances are more. Therefore, better improvement in accuracy can be seen at higher undersampling factors. The increase in accuracy obtained using the proposed scheme compared to conventional method is shown in Fig. 13. This is the improvement that can be obtained by proposed scheme on this vibration dataset and this scheme is helpful in case of other datasets also. The other reason for reduced accuracy at higher undersampling factors is that the information preserved by compressive measurements decreases with increasing undersampling.

Another performance measures used for comparing the classifier performance for proposed and conventional schemes are: precision, recall, and F-measure, which are calculated using following equations:

$$\text{Precision } (P) = \frac{TP}{TP+FP}\times 100, \tag{10}$$

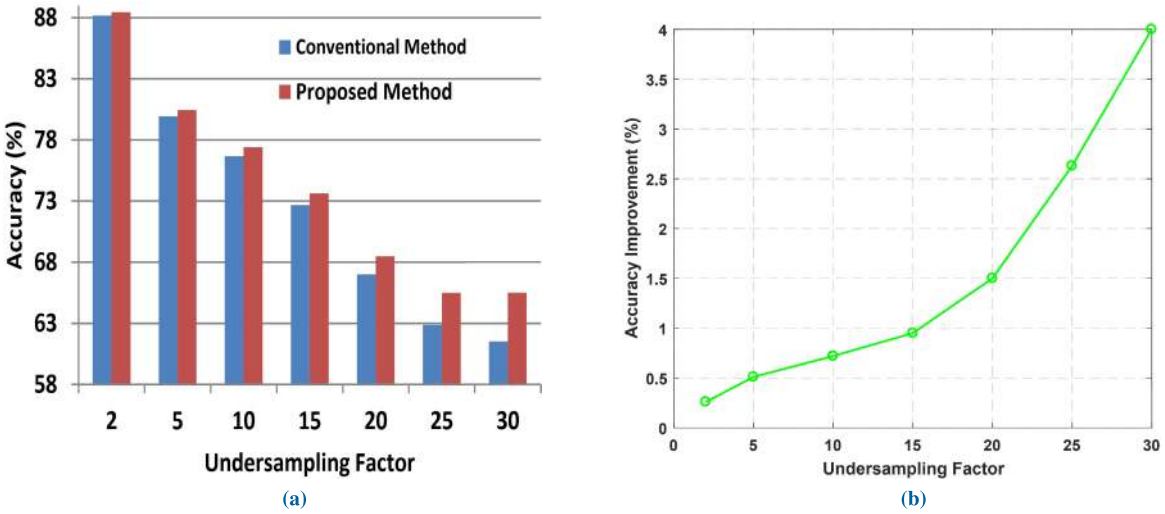
$$\text{Recall } (R) = \frac{TP}{TP+FN}\times 100, \tag{11}$$

$$\text{F-measure } (F) = \frac{2TP}{2TP+FP+FN}\times 100, \tag{12}$$

where, TP stands for true positive, FP for false positive and FN for false negative. The parameter wise comparison of these three measures for the conventional and proposed schemes for different undersampling factors are shown in Fig. 14. The results show that the use of overlap aware learning for training a classifier improves its performance in classifying the features extracted from compressive

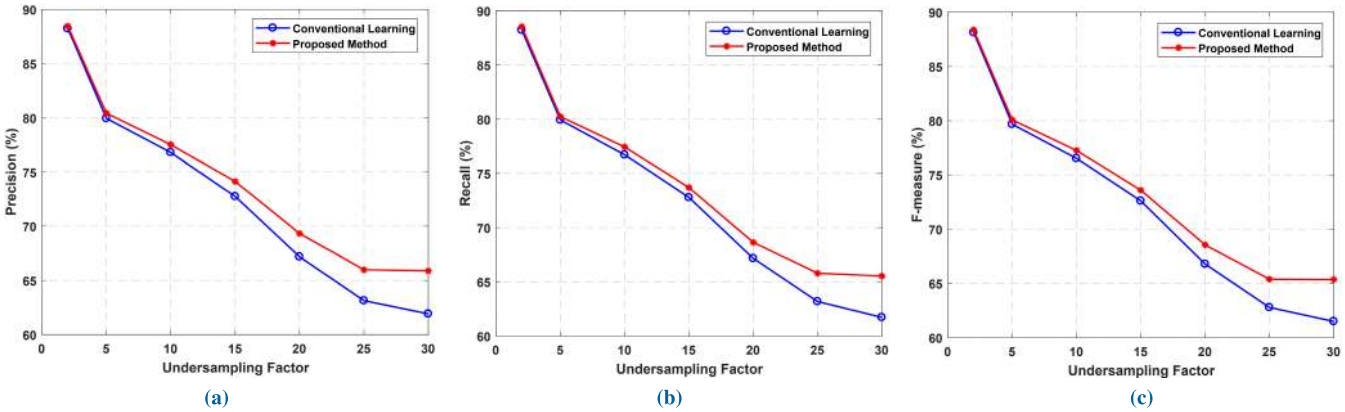
TABLE 2. Performance comparison of conventional and proposed method in classifying the features extracted from compressive measurements for different undersampling factors.

CS undersampling factor	Input signal dimension	Feature set dimension	k-fold	# of instances in subconcept	# of overlapped instances	Classification accuracy using	
						conventional method	proposed method
2	800000	1600×8	10	2	56	88.18%	88.44%
5	800000	1600×8	10	5	98	79.93%	80.44%
10	800000	1600×8	10	10	128	76.68%	77.4%
15	800000	1600×8	10	15	144	72.68%	73.63%
20	800000	1600×8	10	20	172	67.0%	68.5%
25	800000	1600×8	10	25	197	62.87%	65.5%
30	800000	1600×8	10	30	210	61.5%	65.5%



(a) (b)

FIGURE 13. Performance of proposed scheme: (a) comparison of accuracies of two schemes i.e., conventional and proposed, for different undersampling factors, (b) increase in accuracy obtained for different undersampling factors using the proposed method.



(a) (b) (c)

FIGURE 14. Comparison of classifier performance using: (a) precision, (b) recall and (c) F-measure, in classifying the features obtained from compressive measurements for conventional and proposed method for different undersampling factors.

measurements for higher undersampling factors. Plots of these three metrices look similar because the values of these metrices obtained using conventional method were changing in similar fashion as the undersampling factor is increased and then accordingly the performance gain is obtained by proposed method.

TABLE 3. Analysis on EEG epileptic seizure dataset: performance comparison of conventional and proposed method in classifying the features extracted from compressive measurements of EEG epileptic seizure dataset for different undersampling factors.

CS undersampling factor	Input signal dimension	Feature set dimension	k-fold	# of instances in subconcept	# of overlapped instances	Classification accuracy using	
						conventional method	proposed method
2	942080	1840×8	10	2	40	83.21%	83.37%
4	942080	1840×8	10	4	90	82.66%	83.32%
8	942080	1840×8	10	8	157	81.85%	82.99%
16	942080	1840×8	10	16	218	79.02%	81.90%
32	942080	1840×8	10	32	265	77.88%	80.93%

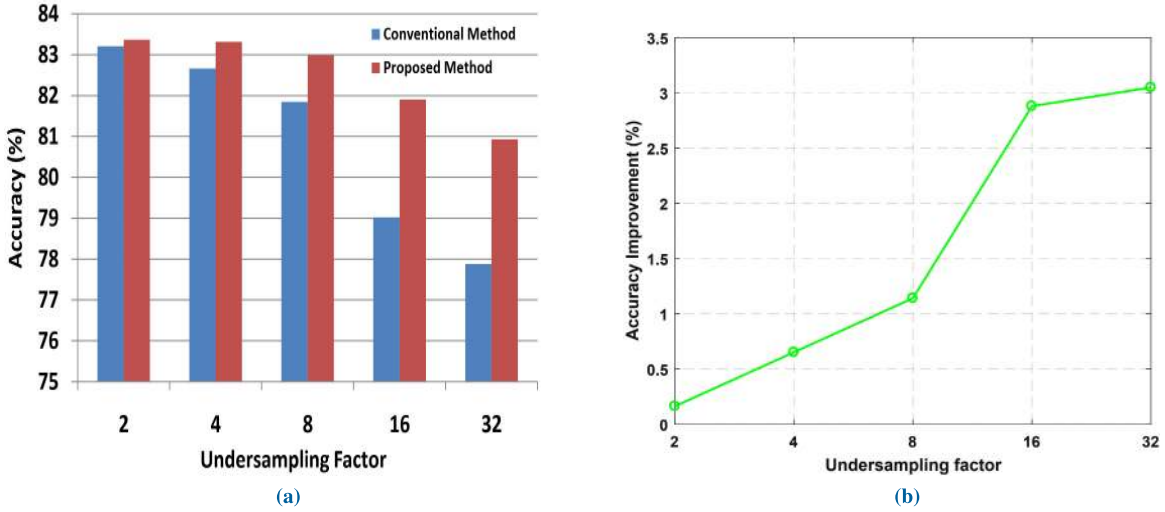


FIGURE 15. Analysis on EEG epileptic seizure dataset: (a) comparison of accuracies of two schemes i.e., conventional and proposed, for different undersampling factors, (b) increase in accuracy obtained for different undersampling factors using the proposed method.

V. ANALYSIS OF PROPOSED SCHEME

The analysis of proposed scheme has been done to verify its effectiveness. To ensure the applicability of proposed method on other datasets, its analysis has been done on the datasets having entirely different characteristics than vibration data. Another analysis has been done on the run-time computational complexity of proposed and conventional schemes to highlight the advantage of proposed scheme.

A. ANALYSIS ON DIFFERENT DATASETS

The datasets that have been analyzed are EEG epileptic seizure dataset and EEG mental arithmetic tasks dataset. These datasets have been taken from physionet [26]. Apart from some signal specific changes, all the other experimental settings are same as that in the main experiment.

1) EEG EPILEPTIC SEIZURE DATASET

The performance of proposed method is evaluated on the EEG signal with epileptic seizures. This signal is taken from physionet's CHB-MIT database [27]. This signal is processed under the same conditions as that in the main experiment. Some signal specific changes are done like: the total signal length of 942080 is divided into 1840 segments containing 512 samples each, which is equivalent to two-second epoch of the signal. The undersampling factors used are 2, 4, 8, 16, and 32, a multiple of segment length. The filter matrices are redesigned as per the length of signal segment. After filtering and norm-2, the size of feature vectors obtained is 1840×8. The decision tree classifier with the same settings as in the main experiments is used for finding and handling overlapped instances from the classifier data. This is done to obtain a refined dataset for the classifier, which can further be used for detecting the presence of seizures in the EEG signal. The simulation settings and results of this analysis are summarized in Table 3. Here also, the number of instances in the subconcept are kept proportional to undersampling factor, to achieve the desired performance. As shown, the number of overlapped instances identified increases with the undersampling factor and accordingly the performance improvement is obtained at higher undersampling factors. A comparison of classification accuracies obtained on these features using conventional and proposed methods is shown

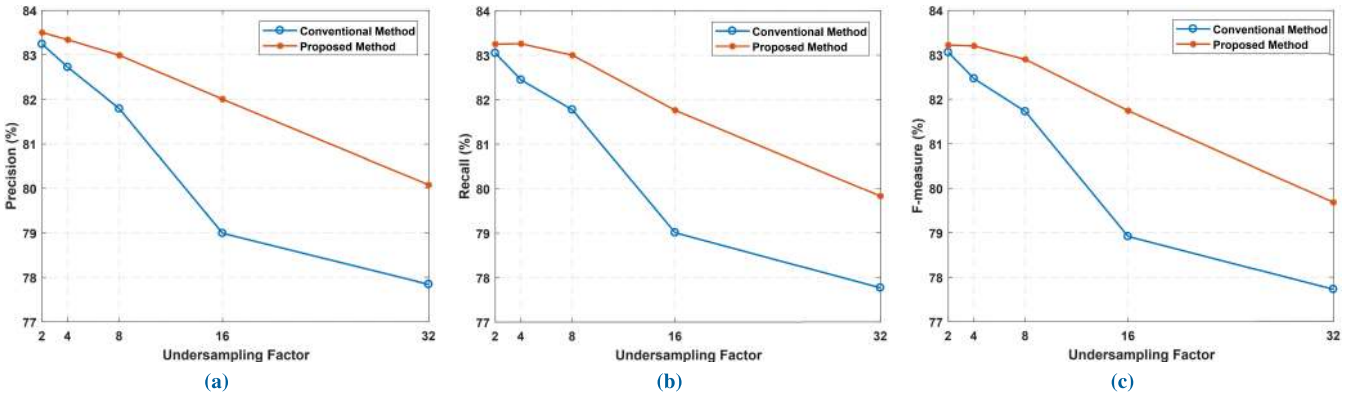


FIGURE 16. Analysis of classifier performance using: (a) precision, (b) recall and (c) F-measure, in classifying the features obtained from compressive measurements of EEG epileptic seizure signal for conventional and proposed method for different undersampling factors.

TABLE 4. Analysis on EEG mental arithmetic task dataset: performance comparison of the conventional and proposed methods in classifying the features extracted from compressive measurements of EEG epileptic seizure dataset for different undersampling factors.

CS undersampling factor	Input signal dimension	Feature set dimension	k-fold	# of instances in subconcept	# of overlapped instances	Classification accuracy using	
						conventional method	proposed method
2	930000	1860×8	10	2	56	77.53%	78.12%
5	930000	1860×8	10	5	140	75.97%	77.20%
10	930000	1860×8	10	10	213	76.34%	77.58%
15	930000	1860×8	10	15	272	70.32%	74.68%
20	930000	1860×8	10	20	319	68.87%	72.96%
25	930000	1860×8	10	25	372	65.11%	70.65%
30	930000	1860×8	10	30	421	61.08%	69.25%

in Fig. 15(a) and the corresponding accuracy improvement obtained using proposed method is shown in Fig. 15(b). It has been observed that an improvement of around 3% has been achieved by the proposed method in detecting the epileptic seizures from the EEG signal as compared to the conventional method. A performance comparison based on the parameters derived from confusion matrix as per (10), (11), and (12), is also presented in Fig. 16. This shows that the proposed scheme performs better than the conventional CSP scheme.

2) EEG MENTAL ARITHMETIC TASKS DATASET

Another dataset on which the proposed scheme has been evaluated is the EEG mental arithmetic tasks dataset taken from physionet [28]. This signal is also processed under the same conditions as that in the main experiment. The total signal length taken is 930000 samples, which is divided into 1860 segments containing 500 samples each. This choice of segment size is equivalent to one second epoch of the signal. Here, the undersampling factors used are 2, 5, 10, 15, 20, 25, and 30, which are a multiple of segment length. The filter matrices are used as per the length of signal segment. After filtering and computing norm-2 of these signal segments, the size of feature vectors obtained is 1860×8. Here also, the decision tree classifier with the same settings as in the main experiments is used for finding and handling overlapped instances from the classifier data. This is done to obtain a refined dataset for classifier, which can further be used to distinguish the EEG of mental arithmetic task from normal EEG. The simulation settings and results of this analysis are summarized in Table 4. Here also, the number of instances in the subconcept are kept proportional to undersampling factor, to achieve the desired performance. As shown, the number of overlapped instances identified increases with the undersampling factor and accordingly the performance improvement is obtained at higher undersampling factors. A comparison of classification accuracies obtained on these features using conventional and proposed methods is shown in Fig. 17(a) and the corresponding accuracy improvement obtained using proposed method is shown in Fig. 17(b). It has been observed that an improvement of around 8% has been achieved by proposed method in detecting the EEG activity during mental arithmetic task as compared to the conventional method. A performance comparison based on the parameters derived from confusion matrix as per (10), (11), and (12), is also

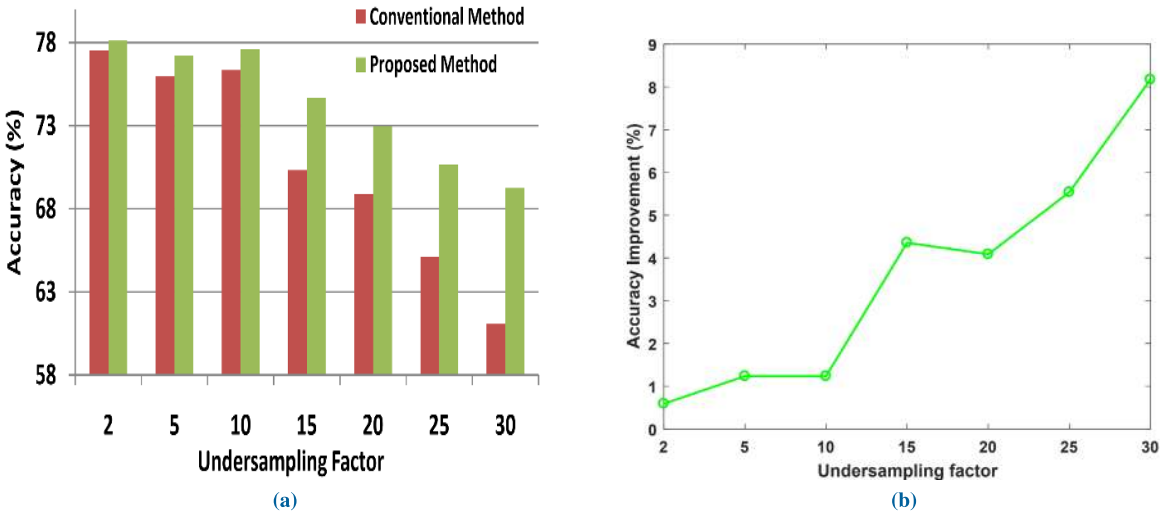


FIGURE 17. Analysis on EEG mental arithmetic task dataset: (a) comparison of accuracies of two schemes i.e., conventional and proposed, for different undersampling factors, (b) increase in accuracy obtained for different undersampling factors using the proposed method.

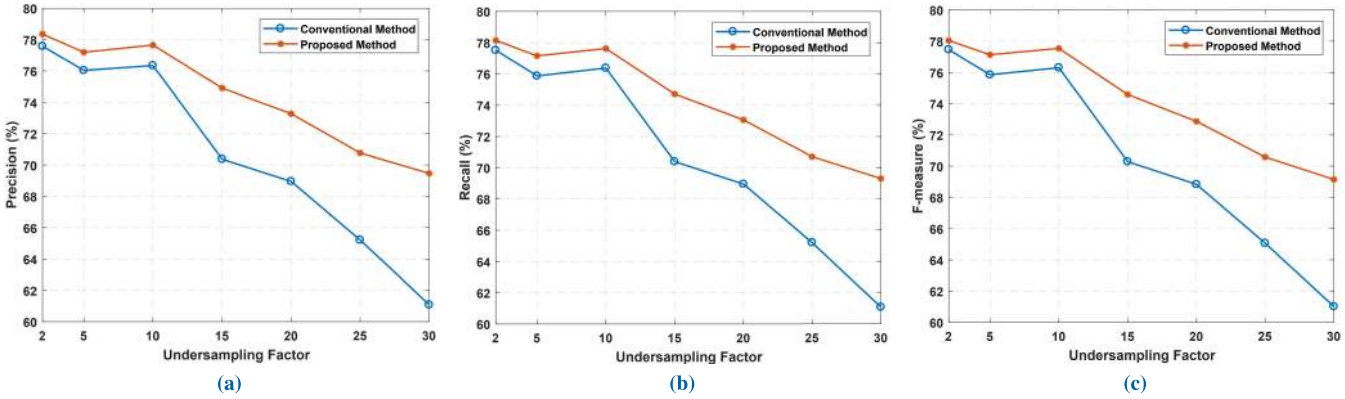


FIGURE 18. Analysis of classifier performance using: (a) precision, (b) recall and (c) F-measure, in classifying the features obtained from compressive measurements of EEG mental arithmetic task signal for conventional and proposed method for different undersampling factors.

presented in Fig. 18. This shows that the proposed scheme performs better than the conventional CSP scheme.

B. ANALYSIS OF RUN-TIME COMPLEXITY

The run-time complexity analysis of proposed and conventional schemes has been done to highlight the advantages of proposed scheme over the conventional ones. All the three datasets used above have been analyzed for run-time complexity to prove the generality of proposed approach. The run-time is measured in seconds (s). The stage-wise comparative analysis of run-time complexity for the three datasets viz. vibration [18], EEG seizure [27] and EEG mental arithmetic task [28] has been presented in Tables 5, 6 and 7, respectively. For each dataset, the proposed method has been compared against the conventional CS scheme which requires reconstruction and the conventional CSP scheme which uses conventional classification. Here, u is the undersampling factor, CS-Acq is the CS acquisition, CS-Rec is the CS reconstruction, and FE is the feature extraction. The columns named CS-Acq gives the run-time of CS acquisition step, which is same for all the three schemes. The column named CS-Rec gives the run-time of CS reconstruction stage, which is present only in case of conventional CS scheme. Similarly, the columns named FE reports the time required for extracting the features. In case of conventional CS, this time is more because the features are extracted from the CS reconstructed signal of length n, compared to the CSP based schemes where features are extracted directly from compressive measurements of length m. Therefore, the time required for feature extraction stage for both the CSP based schemes i.e., conventional and proposed, is same. corresponding to each segment a single feature vector is generated after the feature extraction stage, hence the equal number of feature vectors are obtained after the feature extraction stage. Therefore, the training time, which reflects the learning complexity of the classifier, is nearly same for all the three schemes.

It has been found that for most of the stages, the run-time complexity decreases with undersampling factor for all the

TABLE 5. Performance comparison of conventional and proposed methods based on run-time complexity analysis for vibration data.

u[1]	Run-Time Complexity (s)												
	Conventional CS					Conventional CSP				Proposed method			
	CS-Acq	CS-Rec	FE	Classifier	Total	CS-Acq	FE	Classifier	Total	CS-Acq	FE	Classifier	Total
2	0.4297	708.6501	2.5232	0.0089	711.6118	0.4297	1.0185	0.0089	1.4571	0.4297	1.0185	0.0169	1.4651
5	0.0719	128.0612	2.5623	0.0105	130.7059	0.0719	0.4024	0.0105	0.4848	0.0719	0.4024	0.0174	0.4917
10	0.0605	58.9153	2.4947	0.0108	61.4812	0.0605	0.3201	0.0108	0.3914	0.0605	0.3201	0.0153	0.3958
15	0.0522	48.744	2.5323	0.0114	51.3402	0.0522	0.1964	0.0114	0.2599	0.0522	0.1964	0.0144	0.2630
20	0.0551	43.5799	2.5550	0.0106	46.2007	0.0551	0.2172	0.0106	0.2829	0.0551	0.2172	0.0120	0.2844
25	0.0454	41.5873	2.5441	0.0114	44.1882	0.0454	0.2010	0.0114	0.2578	0.0454	0.2010	0.0117	0.2582
30	0.0427	38.0931	2.5769	0.0116	40.7244	0.0427	0.1156	0.0116	0.1699	0.0427	0.1156	0.0112	0.1695

[1]u = undersampling factor, CS Acq = CS acquisition, CS Rec = CS reconstruction, FE = feature extraction.

TABLE 6. Performance comparison of conventional and proposed methods based on run-time complexity analysis for EEG epileptic seizure dataset.

u	Run-Time Complexity (s)												
	Conventional CS					Conventional CSP				Proposed method			
	CS-Acq	CS-Rec	FE	Classifier	Total	CS-Acq	FE	Classifier	Total	CS-Acq	FE	Classifier	Total
2	0.7158	1181.5220	3.2434	0.0115	1185.4926	0.7158	1.0797	0.0115	1.8069	0.7158	1.0797	0.0212	1.8166
5	0.5931	170.8482	3.2215	0.0116	174.6744	0.5931	0.7369	0.0116	1.3416	0.5931	0.7369	0.0202	1.3502
10	0.4114	45.5046	3.2354	0.0125	49.1639	0.4114	0.5235	0.0125	0.9473	0.4114	0.5235	0.0181	0.9530
15	0.3801	18.0994	3.2503	0.0134	21.7433	0.3801	0.3844	0.0134	0.7779	0.3801	0.3844	0.0118	0.7763
20	0.3830	8.4612	3.3981	0.0122	12.2544	0.3830	0.3307	0.0122	0.7258	0.3830	0.3307	0.0118	0.7254

TABLE 7. Performance comparison of conventional and proposed methods based on run-time complexity analysis for EEG mental arithmetic task dataset.

u	Run-Time Complexity (s)												
	Conventional CS					Conventional CSP				Proposed method			
	CS-Acq	CS-Rec	FE	Classifier	Total	CS-Acq	FE	Classifier	Total	CS-Acq	FE	Classifier	Total
2	0.2551	886.7721	2.9593	0.0128	889.9992	0.2551	0.9003	0.0128	1.1682	0.2551	0.9003	0.0229	1.1783
5	0.0967	189.4037	2.9924	0.0123	192.5053	0.0967	0.3708	0.0123	0.4800	0.0967	0.3708	0.0207	0.4882
10	0.0626	111.3321	3.0507	0.0121	114.4575	0.0626	0.2877	0.0121	0.3624	0.0626	0.2877	0.0167	0.3670
15	0.0541	99.8078	2.9600	0.0131	102.835	0.0541	0.2343	0.0131	0.3015	0.0541	0.2343	0.0159	0.3043
20	0.0571	93.6550	2.9593	0.0131	96.6605	0.0571	0.2571	0.0131	0.3273	0.0571	0.2571	0.0159	0.3301
25	0.0501	90.4391	2.9591	0.0139	93.4381	0.0501	0.2062	0.0139	0.2701	0.0501	0.2062	0.0135	0.2697
30	0.0528	86.0259	2.9576	0.0143	89.0505	0.0528	0.1912	0.0143	0.2583	0.0528	0.1912	0.0136	0.2576

three datasets. In case of conventional CS, the complexity of reconstruction stage dominates the complexity of all the other stages. Compared to the conventional CSP, there is a minor increase in the run-time complexity of proposed method due to addition of overhead of overlap aware step before classifier. This minor increment in complexity can only be seen at lower undersampling factors. At higher undersampling factors like 25 and 30, classification using conventional method becomes complicated due to the presence of more number of overlapped instances. As opposed to this, the proposed method results in a smoother decision boundary by removing the overlapped instances. This makes the classification process easier, which in turn compensates the overhead of overlap aware step and makes the proposed method more efficient than the conventional schemes for higher undersampling factors like 25 and 30, as shown in Tables 5, 6 and 7. So, it can be

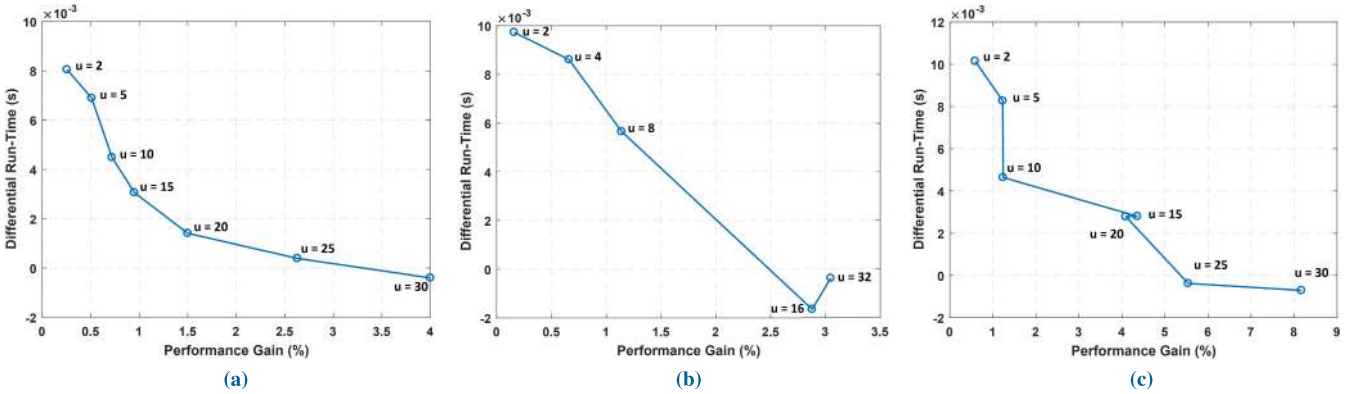


FIGURE 19. Analysis of proposed method based on obtained performance gain and the differential run-time for the three datasets: (a) vibration dataset, (b) EEG epileptic seizure dataset, and (c) EEG mental arithmetic task dataset, for different undersampling factors.

TABLE 8. Performance comparison of conventional and proposed method in classifying the features extracted from compressive measurements for different undersampling factors using neural network classifier.

CS undersampling factor	Input signal dimension	Feature set dimension	k-fold	# of overlapped instances	Classification accuracy using	
					conventional method	proposed method
2	800000	1600×8	10	26	91.75%	91.69%
5	800000	1600×8	10	99	84.50%	84.88%
10	800000	1600×8	10	122	80.99%	80.94%
15	800000	1600×8	10	144	78.06%	79.13%
20	800000	1600×8	10	170	74.75%	76.63%
25	800000	1600×8	10	200	70.87%	72.19%

inferred that the CSP based methods reduce the implementation complexity of signal processing systems compared to the conventional CS based method. Also, the proposed method is able to achieve higher accuracy (refer Tables 2, 3, and 4) than the conventional CSP scheme for comparable run-time complexity.

C. COMPLEXITY VS GAIN ANALYSIS

In this section, an analysis has been done to weight the obtained classification performance gain over the required extra execution time. The analysis has been done for all the three datasets used in this paper viz. vibration dataset [18], EEG seizure dataset [27], and EEG mental arithmetic task dataset [28]. The graphs have been plotted between the performance gain, i.e., increase in classification accuracy obtained, and the differential run-time, i.e., difference between run-times of proposed and conventional CSP methods. The resulted graphs are shown in Fig. 19(a), (b), and (c), respectively for the three datasets. All these graphs show that as the undersampling factor is increased, the performance gain increases and the differential run-time decreases. For $u = 2$, the run-time of proposed method is 8-10 ms more than the conventional CSP based method and the performance gain is also less. As the value of u increases the difference in run-times decreases and the performance gain increases. This happens because the at higher undersampling factors, the overlapping is high and the effect of proposed method is more visible. For more higher undersampling factors the differential run-time goes negative, indicating proposed method takes less time than the conventional CSP based method, while the performance gain achieved is also high. The reason behind this is the smoother decision boundary resulted by the use of overlap aware technique.

In case of EEG epileptic seizure dataset, $u = 32$ seems to be an odd point, which is actually not an odd point. At $u = 32$, the run-time complexity of conventional CSP based method is 0.7258 s and that of proposed method is 0.7254 s, as per Table 6. The proposed method takes 0.4 ms less time than the conventional CSP based method. Table 6 also shows that at $u = 32$, for the same acquisition and feature extraction time, the classification using overlap aware technique is taking less time than the conventional learning. For a 3.05% improvement in the accuracy, the overlap aware technique is taking 0.4 ms less time at $u = 32$ for this dataset. For $u = 16$, the improvement obtained in accuracy is 2.88%, for which the overlap aware technique is taking 1.6 ms less time than

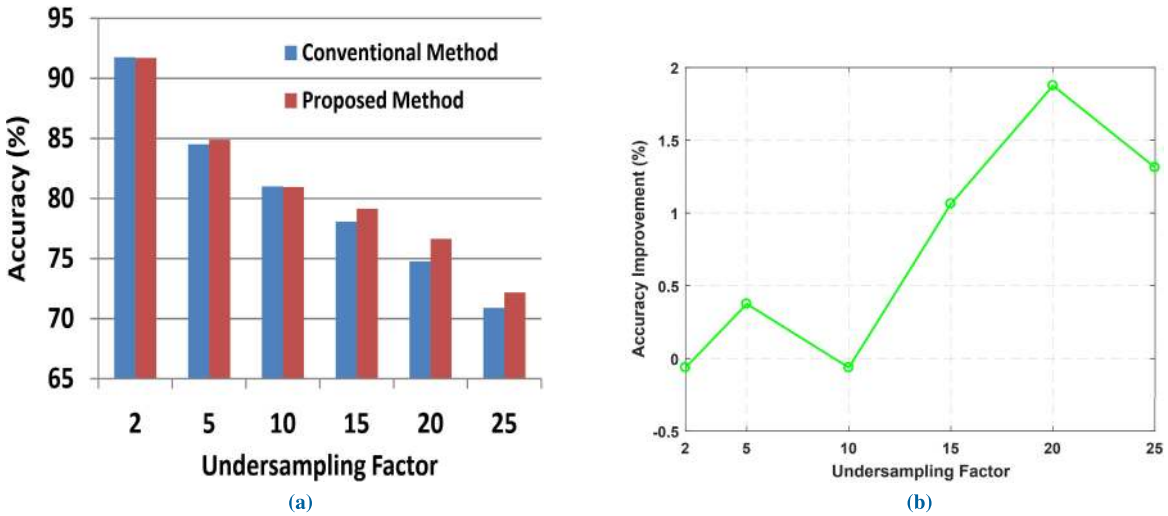


FIGURE 20. Performance of proposed scheme using neural network: (a) comparison of accuracies of two schemes i.e., conventional and proposed, for different undersampling factors, (b) increase in accuracy obtained for different undersampling factors using the proposed method.

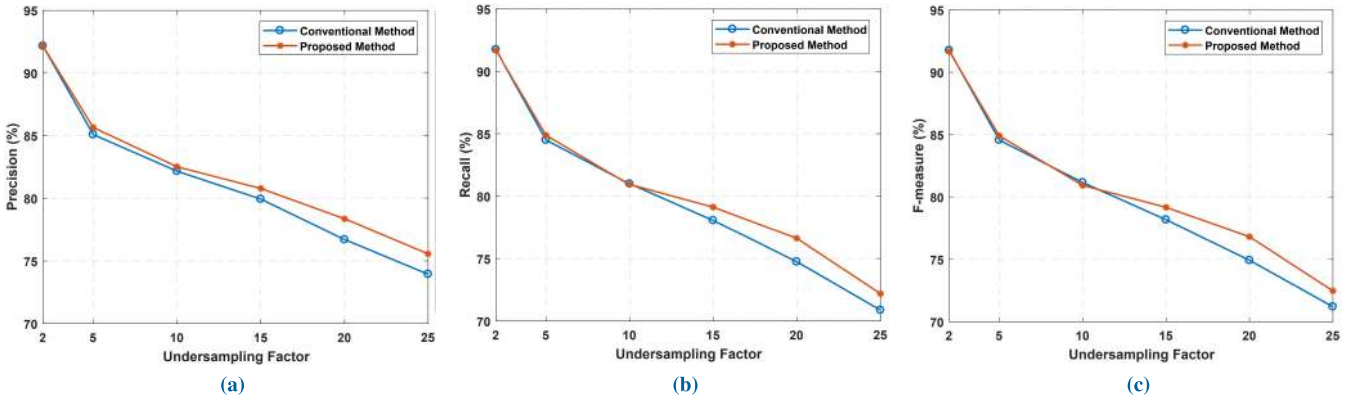


FIGURE 21. Analysis of neural network classifier performance using: (a) precision, (b) recall and (c) F-measure, in classifying the features obtained from compressive measurements of vibration signal for conventional and proposed method for different undersampling factors.

the conventional learning. The main focus of the work is to achieve an improvement in accuracy at higher undersampling factors for reasonable run-time, this is what Fig. 19 shows.

D. ANALYSIS USING NEURAL NETWORK CLASSIFIER

In this section, another analysis has been done to see the effect of using neural network classifier in-place of the decision tree classifier. The main parameters used for this classifier are as:

- HiddenLayerSizes = (8,4): The 1^{st} element represents the number of neurons in each hidden layer and the 2^{nd} element represents the number of hidden layers used in the network.
- Solver = "adam": stochastic gradient-based weight optimizer.
- Alpha = 0.00001: ℓ_2-regularization parameter.
- BatchSize = "auto" = min(200, num-of-samples): sets the size of mini-batches to either 200 or to the number of samples whichever is minimum.
- LearningRate = 0.001: defines the initial learning rate, which controls the step-size for updating the weights.
- MaxIter = 200: maximum number of iterations until convergence.
- Tol = 1e-4: tolerance for the optimization to stop the training.

With this settings, the results obtained for the vibration dataset [18] are shown in Fig. 20. Here, the input signal size used and the corresponding feature set dimensions are shown in Table 8. The results show an improvement with the increasing undersampling factor using the proposed method. An improvement near to 2% has been obtained using the neural network classifier. The classifier performance has also been compared using the parameters like precision, recall, and F-measure, which are derived from confusion matrices as per (10), (11), and (12). The results obtained are shown in Fig 21, which are in accordance with the obtained improvement. Another performance analysis has been done based on run-time complexity as shown in Table 9. This shows that

TABLE 9. Performance comparison of conventional and proposed methods based on run-time complexity analysis for vibration data.

u	Run-Time Complexity (s)							
	Conventional CSP				Proposed method			
	CS-Acq	FE	Classifier	Total	CS-Acq	FE	Classifier	Total
2	0.4297	1.0185	0.2060	1.6542	0.4297	1.0185	0.2108	1.6591
5	0.0719	0.4024	0.2493	0.7235	0.0719	0.4024	0.2466	0.7209
10	0.0605	0.3201	0.2517	0.6323	0.0605	0.3201	0.2459	0.6265
15	0.0522	0.1964	0.2416	0.4902	0.0522	0.1964	0.2371	0.4857
20	0.0551	0.2172	0.2438	0.5162	0.0551	0.2172	0.2312	0.5035
25	0.0454	0.2010	0.2225	0.4689	0.0454	0.2010	0.2116	0.4581

with the increasing undersampling factor, the run-time complexity of proposed approach decreases, which is an added advantage.

VI. CONCLUSION AND FUTURE SCOPE

CSP is an emerging aspect of CS, in which work has been done on solving some signal processing problems directly from compressive measurements. The biggest advantage in solving the signal processing tasks directly from compressive measurements is that it had made it possible to omit the complex CS reconstruction step, when solving the problems like classification, detection, estimation, etc. This drastically reduced the overall complexity, speed and power consumption in processing the signal for these applications. It has also opened the opportunities for efficient hardware designs for tackling these problems in real-time scenarios. This paper has studied the performance of CSP in solving classification problems. It has been found that classification accuracy is hampered at higher undersampling factors due to overlapping among the features extracted from compressive measurements. Maintaining good classification accuracy at higher undersampling factors is a crucial need of today's on node condition monitoring systems in order to minimize the power consumption. A solution to the above problem has been proposed using overlap aware learning for training a classifier. The simulation results showed that the CSP based methods reduce the implementation complexity of signal processing systems compared to the conventional CS based method. Also, the proposed scheme enhances the classification accuracy than conventional CSP scheme for comparable run-time complexity. Further, the work has been analyzed from different aspects like applicability on different datasets, run-time complexity, complexity vs gain analysis and using different classification scheme. All the results obtained agree that the proposed method performs better than the conventional CSP based method at higher undersampling factors.

This work opens a research direction for the signal processing applications, in which overlapping among data instances is highly predominant. For such applications, the proposed scheme can give significant improvement. This work can be further extended to implement it in hardware for real-time condition monitoring, so that, the cost and overall performance of the system can be optimized. Also, other ways can be sought to come up with the best method to handle the overlapped instances, which can further improve the performance of CSP based classification applications.

REFERENCES

[1] M. Rani, S. B. Dhok, and R. B. Deshmukh, "A systematic review of compressive sensing: Concepts, implementations and applications," *IEEE Access*, vol. 6, pp. 4875–4894, 2018.
[2] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
[3] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[4] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
[5] R. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, Jul. 2007.
[6] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
[7] E. J. Candès, "Compressive sampling," in *Proc. Int. Cong. Math.*, Madrid, Spain, vol. 3, Aug. 2006, pp. 1433–1452.
[8] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, Jun. 2007.
[9] J. Haupt, R. Castro, R. Nowak, G. Fudge, and A. Yeh, "Compressive sampling for signal classification," in *Proc. 14th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, 2006, pp. 1430–1434.
[10] M. F. Duarte, M. A. Davenport, M. B. Wakin, and R. G. Baraniuk, "Sparse signal detection from incoherent projections," in *Proc. IEEE Int. Conf. Acoust. Speed Signal Process.*, Toulouse, France, 2006, p. 3.
[11] J. Haupt and R. Nowak, "Compressive sampling for signal detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Honolulu, HI, USA, Apr. 2007, pp. III-1500–III-1509.
[12] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, "Signal processing with compressive measurements," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 445–460, Apr. 2010.
[13] J. Y. Park, M. B. Wakin, and A. C. Gilbert, "Modal analysis with compressive measurements," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1655–1670, Apr. 2014.
[14] A. Hariri and M. Babaie-Zadeh, "Compressive detection of sparse signals in additive white Gaussian noise without signal reconstruction," *Signal Process.*, vol. 131, pp. 376–385, Feb. 2017.
[15] J. Ma, J. Xie, and L. Gan, "Compressive detection of unknown-parameters signals without signal reconstruction," *Signal Process.*, vol. 142, pp. 114–118, Jan. 2018.

[16] D. Mo and M. F. Duarte, "Compressive parameter estimation via K-median clustering," *Signal Process.*, vol. 142, pp. 36–52, Jan. 2018.
[17] R. Calderbank and S. Jafarpour, "Finding Needles in Compressed Haystacks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Kyoto, Japan, 2012, pp. 3441–3444.
[18] J. Slavič, A. Brković, and M. Boltežar, "Typical bearing-fault rating using force measurements: Application to real data," *J. Vib. Control*, vol. 17, no. 14, pp. 2164–2174, Dec. 2011.
[19] R. Baraniuk. (Apr. 2, 2011). An introduction to compressive sensing. OpenStax-CNX. [Online]. Available: http://legacy.cnx.org/content/col11133/1.5/
[20] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *Proc. IEEE Int. Symp. Circuits Syst.*, New Orleans, LA, USA, May 2007, pp. 1959–1962.
[21] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.
[22] J. Sánchez, "Analysis of new techniques to obtain quality training sets," *Pattern Recognit. Lett.*, vol. 24, no. 7, pp. 1015–1022, Apr. 2003.
[23] T. M. Mitchell, *Machine Learning*, 1st ed. New York, NY, USA: McGraw-Hill, 1997.
[24] S. Gupta and A. Gupta, "A set of measures designed to identify overlapped instances in software defect prediction," *Computing*, vol. 99, no. 9, pp. 889–914, Sep. 2017.
[25] R. Belohlavek, B. De Baets, J. Outrata, and V. Vychodil, "Inducing decision trees via concept lattices," *Int. J. Gen. Syst.*, vol. 38, no. 4, pp. 455–467, May 2009.
[26] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley, "PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000. [Online]. Available: http://circ.ahajournals.org/cgi/content/full/101/23/e215
[27] A. H. Shoeb, "Application of machine learning to epileptic seizure onset detection and treatment," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, Sep. 2009.
[28] I. Zyma, S. Tukaev, I. Seleznov, K. Kiyono, A. Popov, M. Chernykh, and O. Shpenkov, "Electroencephalograms during mental arithmetic task performance," *Data*, vol. 4, no. 1, p. 14, 2019, doi: 10.3390/data4010014.

MEENU RANI (Student Member, IEEE) received the B.Tech. degree from Kurukshetra University, India, in 2008, and the M.Tech. degree in VLSI design from NIT Kurukshetra, Kurukshetra, in 2012. She is currently pursuing the Ph.D. degree with the Visvesvaraya National Institute of Technology, Nagpur, India. She was a Polytechnic Lecturer, from 2008 to 2010, and an Assistant Professor with Sharda University, Greater Noida, India, from 2012 to 2014. Her research interests include the signal processing using compressive sensing, pattern recognition and CMOS VLSI Design. She has received the Teaching Assistantship from the Ministry of Human Resource Development.

SANJAY B. DHOK received the Ph.D. degree in electronics engineering from the Visvesvaraya National Institute of Technology, Nagpur, India. He is currently an Associate Professor with the Centre for VLSI and Nanotechnology, Visvesvaraya National Institute of Technology. He has published many research articles in national and international journals and conferences. His research interests include signal processing, image processing, data compression, wireless sensor networks, and VLSI design. He is a member of the IEEE Society

RAGHAVENDRA B. DESHMUKH received the Ph.D. degree in electronics engineering from the Visvesvaraya National Institute of Technology, Nagpur, India. He is currently a Professor with the Centre for VLSI and Nanotechnology, Visvesvaraya National Institute of Technology. He has published many research articles in national and international journals and conferences. His research interests include digital design, CMOS design, and reconfigurable FPGA design. He is a member of the IEEE Society.

PUNIT KUMAR (Student Member, IEEE) received the B.Tech. degree in computer science and engineering from Kurukshetra University, Kurukshetra, Haryana, India, in 2011, and the M.Tech. degree in computer science and engineering from Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Haryana, in 2014. He is currently pursuing the Ph.D. degree with the Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, India.