Audio Signal Recognition in Complex Environments Using Sparse Representation

Xi Ren

Chongqing College of Humanities, Science & Technology, Chongqing 401524, China.
Email: renxxr@outlook.com

(Received 10 January 2024 accepted 25 July 2024)

Recognizing audio signals in complex environments is crucial for acquiring adequate information. This paper integrates the sparse expression algorithm with Mel-frequency cepstral coefficient (MFCC) features. The combined approach was applied in convolutional neural network classifiers to recognize acoustic scenes in audio signals within complex environments. The algorithm was then simulated and tested using the TUT Sound Events 2016 and TUT Acoustic Scenes 2016 datasets. In the experiments, the efficacy of the developed sparse feature extraction method was validated. Then, the appropriate sparse dictionary size was determined. The algorithm was subsequently compared with two recognition algorithms based on sparse and MFCC features, respectively. It was found that the extraction approach proposed in this paper had a higher signal-to-noise ratio. The results revealed variations in the required sparse dictionary size for different datasets: 75 for TUT Sound Events 2016 and 150 for TUT Acoustic Scenes 2016. The MFCC-combined recognition algorithm demonstrated the fastest convergence during training among the three audio scene recognition algorithms. For both the TUT Sound Events 2016 and TUT Acoustic Scenes 2016 datasets, the MFCC-combined recognition algorithm achieved the highest classification accuracy, and the recognition accuracy for the former dataset was higher.

NOMENCLATURE

$y(k)$: the signal in the frequency domain acquired through fast Fourier transform
$y(n)$: the original time domain signal
k: the serial number of the sampling point
n: the time sampling point of the time domain signal
$p(\omega)$: the instantaneous energy of $y(k)$
$h_m(k)$: the frequency response of the triangular filter
m: the serial number of a group of triangular filters
$c(l)$: the L-order MFCC characteristic parameter
$s(m)$: the energy spectral function of the filtered frequency domain signal
$\mathbf{Y}$: the set of audio samples
y_i: the i-th audio sample
$\mathbf{A}$: the set of the sparse coefficient that corresponds to the audio sample
a_i: the sparse coefficient corresponding to the i-th audio sample
$\mathbf{D}$: the sparse dictionary
d_j: the j-th atom in the sparse dictionary
$\mathbf{e}$: the set of reconstruction errors
$a_{c,j}$: the j-th sparse coefficient obtained from the sparse dictionary for the sample in the class c audio scene
$a'_{c,j}$: the score value of the sparse coefficient obtained after sigmoid function mapping of $a_{c,j}$
m_c: that there are m atoms in the class c sparse dictionary
a': the probability distribution feature of the sparse coefficient score value of the audio sample in different audio scenes

LIST OF ABBREVIATIONS

MFCC: Mel-frequency cepstral coefficient
CNN: convolutional neural network
FFT: fast Fourier transform

1. INTRODUCTION

In everyday life, audio signals are omnipresent, ranging from simple music playback to intricate recordings of natural environments, containing a wealth of information.[1] However, audio signals in complex environments often exhibit various types of high-intensity noise, posing challenges for effective signal recognition.[2] Recognizing audio signals in such conditions has thus become a challenging problem. The conventional approach often reduces noise from audio signals in complex environments. However, this method can only completely separate signals if the noise is entirely known. In practice, noise signals in complex environments are highly random,[3] making complete elimination difficult and risking the loss of original audio information during the noise reduction process. Sparse representation, as a signal processing method, operates on the fundamental idea of using a small number of key elements to reconstruct the signal. In audio signal processing, sparse representation is achieved through sparse dictionary learning - a matrix composed of a limited number of atoms that can be linearly combined to form audio signals. The extraction of useful information is accomplished through the atoms in the sparse dictionary. Liang et al.[4] utilized sparse representation for source identification of collision signals, aiming for more efficient and accurate classification in low signal-to-noise ratio (SNR) environments. They found that sparse features outperformed conventional features, such as the Mel-frequency cepstral coefficient (MFCC). Rao et al.[5] demonstrated the high resilience of sparse representation-based classification in music signal analysis by successfully applying it to automatically identify chords, even under the influence of Gaussian white noise. Additionally, Zou et al.[6] proposed a sparse representation-based verification scheme for cell phone source recordings and proved the method's effectiveness in capturing intrinsic features of cell phone recording

https://doi.org/10.20855/ijav.2024.29.32055

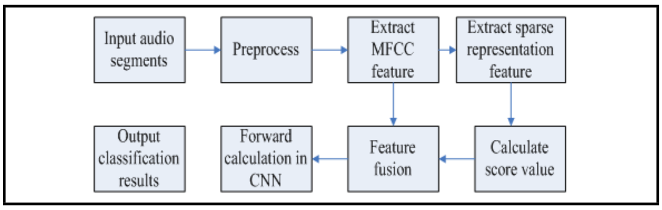
Input audio segments
Preprocess
Extract MFCC feature
Extract sparse representation feature
Output classification results
Forward calculation in CNN
Feature fusion
Calculate score value

Figure 1. Scene recognition process for audio signals in complex environments using sparse representation technique.

devices. Mishra et al.[7] proposed diagnosing multiple faults using balanced and unbalanced acoustic, vibration, and current signal sets. The experimental results validated the effectiveness and universality of this method. Mishra et al.[8] introduced a multi-fault diagnosis system that can operate under any rotational speed. This system utilizes a segmented time-frequency network to extract fault information and establish an intelligent multi-fault classification model, which greatly aids in diagnosing multiple faults at various uncertain rotational speed conditions. The above-mentioned studies have explored various approaches for audio recognition. Some studies have employed sparse features, while others have utilized different features to identify audio signals. Similarly, this paper also employs sparse features for audio signal recognition, but it fuses them with traditional MFCC features to enhance recognition accuracy. In this paper, the sparse representation algorithm was integrated with MFCC features and applied in convolutional neural network (CNN) classifiers to recognize acoustic scenes in complex environments. The algorithm was then subjected to simulation experiments using two datasets, TUT Sound Events 2016 and TUT Acoustic Scenes 2016. The novelty of this article lies in combining MFCC features with the sparse characteristics of audio and then using a CNN to recognize and classify audio data, providing an effective reference for accurately identifying audio signals in complex environments.

2. AUDIO SIGNAL RECOGNITION BASED ON SPARSE REPRESENTATION

The process of utilizing sparse representation for scene recognition of audio signals in complex environments is shown in Fig. 1. Its specific steps are shown below.

1. Audio clips undergo preliminary noise reduction, windowing, and framing as part of the preprocessing stage.[9]

2. The MFCC features are extracted from each frame of the audio signal with the following equations:

$$\begin{cases} y(k) = \sum_{n=0}^{N-1} y(n) \cdot e^{\frac{-2j\pi kn}{N}} \\ p(\omega) = |y(k)|^2 \\ s(m) = \ln\left(\sum_{k=0}^{N-1} p(\omega) \cdot h_m(k)\right) \\ \sum_{m=0}^{M-1} h_m(k) = 1 \\ c(l) = \sum_{m=1}^{M-1} s(m) \cos\left(\frac{\pi l(2m+1)}{2M}\right) \quad l = 1, 2, 3, \cdots, L \end{cases}; \tag{1}$$

where $y(k)$ denotes the signal in the frequency domain acquired through fast Fourier transform (FFT),[13] $y(n)$ denotes the original time domain signal, (k) is the serial number of the sampling point, n is the time sampling point of the time domain signal, $p(\omega)$ is the instantaneous energy of $y(k)$, $h_m(k)$ denotes the frequency response of the triangular filter,[10] m is the serial number of a group of triangular filters (there are a total of M filters), $c(l)$ is the L-order MFCC characteristic parameter, and $s(m)$ denotes the energy spectral function of the filtered frequency domain signal.[11]

3. Sparse representation feature extraction is conducted on the basis of MFCC features. The main purpose of sparse representation feature extraction for audio signals is to extract the sparse coefficients using a sparse dictionary, and the relationship between audio signal, sparse dictionary, and sparse coefficient[12] is:

$$\begin{cases} \mathbf{Y} = \mathbf{DA} + \mathbf{e} \\ \mathbf{Y} = \{y_1, y_2, \cdots, y_n\} \\ \mathbf{A} = \{a_1, a_2, \cdots, a_n\} \\ \mathbf{D} = \{d_1, d_2, \cdots, d_n\} \end{cases}; \tag{2}$$

where $\mathbf{Y}$ is the set of audio samples, y_i is the i-th audio sample (there are n samples), $\mathbf{A}$ is the set of the sparse coefficient that corresponds to the audio sample, a_i is the sparse coefficient corresponding to the i-th audio sample (there are n coefficients), D is the sparse dictionary, d_j is the j-th atom in the sparse dictionary (there are m in total, which depends on the feature dimensions of the audio samples), and e is the set of reconstruction errors.[13] In the case of known audio samples and sparse dictionaries, the process of solving the sparse coefficients of audio is the process of minimizing e, and the ideal situation is that e is 0. However, in practice, it is almost impossible, so a small tolerance value will be set. When the reconstruction error is smaller than the tolerance value, it is considered that the sparse coefficients have been obtained. The formula for solving the iterative model is:

$$\begin{cases} \mathbf{D} = \arg\min_{\mathbf{D}} \sum_{i=1}^{n} \min_{a_i} \left(||\mathbf{D}a_i - y_i||_2^2 + \lambda ||a_i||_1 \right) \\ a_i = \arg\min_{a_i} \left(||Da_i - y_i||_2^2 + \lambda ||a_i||_1 \right) \end{cases}; \tag{3}$$

In the upper part of Eq. (3), $\mathbf{D}$ is solved on the premise that the value of a_i has been determined; in the lower part, a_i is solved on the premise that the value of $\mathbf{D}$ has been determined. The reconstruction error can be gradually reduced by iterating the upper and lower components of the equation set.

4. The sparse coefficient of the audio samples is extracted using sparse dictionaries for different audio scenes in the same way as in step 3.[14] Then, the score value of the extracted sparse coefficient is calculated for feature fusion. The following equations calculate the score value:

$$\begin{cases} a'_{c,j} = \frac{1}{1 + \exp(-a_{c,j})} \\ a'_c = \sum_{j=1}^{m_c} a'_{c,j} \\ \sigma(a'_c) = \frac{\exp(a'_c)}{\sum_{k=1}^{C} \exp(a'_k)} \\ a' = \left[\sigma(a'_1), \sigma(a'_2), \cdots, \sigma(a'_C)\right] \end{cases}; \tag{4}$$

where $a_{c,j}$ is the j-th sparse coefficient obtained from the sparse dictionary for the sample in the class c audio scene, $a'_{c,j}$ is the score value of the sparse coefficient obtained after sigmoid function mapping of $a_{c,j}$,[15] m_c indicates that there are m atoms in the class c sparse dictionary, which can be used to obtain m sparse coefficients for the samples, and a' is the probability distribution feature of the sparse coefficient score value of the audio sample in different audio scenes,[16] which is used for the subsequent fusion of the features.

5. The MFCC feature and the sparse coefficient-based audio scene probability distribution features are fused. a' in each frame of the audio is combined with the MFCC feature.
6. The combined feature is input into the CNN for forward computation, yielding the classification results for the audio scene.

3. SIMULATION EXPERIMENTS

3.1. Experimental Data

The TUT Sound Events 2016 and TUT Acoustic Scenes 2016 datasets were utilized for conducting simulation experiments. The TUT Sound Events 2016 dataset comprises 22 recordings from two acoustic scenes, while the TUT Acoustic Scenes 2016 dataset includes recordings from 15 acoustic scenes. Each acoustic scene consists of 78 segments. The duration of each segment was 30 seconds. The audio parameters for both datasets were 44.1 kHz, dual-channel, and 24-bit depth. Eleven audio clips were randomly selected from each acoustic scene in the dataset to form the training set; then, another set of 11 non-repetitive audio clips was chosen from the dataset to create the testing set.

3.2. Experimental Setup

3.2.1. Testing the performance of the sparse representation feature extraction method

Firstly, the effectiveness of the sparse representation feature extraction method was verified. Eleven randomly selected audio segments from each acoustic scene in the dataset were used as training data for learning the audio dictionary. The dictionary sizes were set to 50, 100, 150, 200, and 250, respectively. The number of iterations was set at 200. Subsequently, another set of eleven non-repetitive audio segments was randomly chosen from the dataset as the testing set. Using the trained audio dictionary, a sparse representation feature extraction was performed on the audio in the test set. Subsequently, the sparse representation features were reconstructed, and the SNR ratio between the original and reconstructed audio signals was calculated to assess the performance of the sparse representation feature extraction approach proposed in this paper. Additionally, a comparison was made with another matching pursuit-based sparse feature extraction method[17] to validate the effectiveness of our proposed approach. The method of reconstruction using sparse representation features is:

$$Y = DA; \tag{5}$$

where Y is the reconstructed signal, D is the audio dictionary, and A is the extracted sparse representation feature. This article used SNR to evaluate the performance of the sparse representation feature extraction method:

$$SNR = 10\lg\frac{||x||_2^2}{||x-y||_2^2}; \tag{6}$$

where x is the original signal and y is the reconstructed signal. The SNR reflects the difference between the original signal and the reconstructed signal, with a larger SNR indicating a smaller difference.

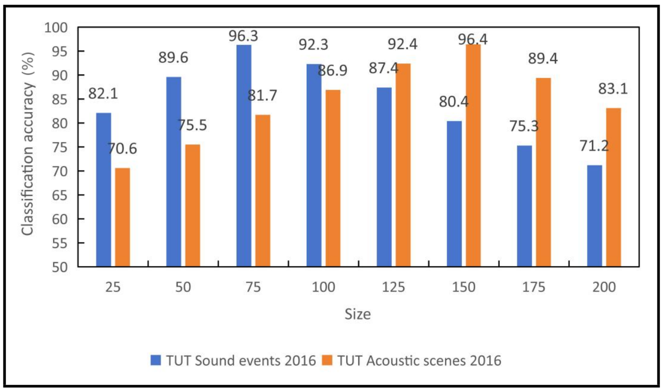

Figure 2. Recognition results of the sparse feature and MFCC feature-based algorithm for two datasets under different dictionary sizes.

3.2.2. Testing for the optimal sparse dictionary size

For comparison, the dictionary size of each audio scene was set to be the same. The dictionary size was set to 25, 50, 75, 100, 125, 150, 175, and 200. The dimension of the MFCC feature was set to 60 through orthogonal experiments. The relevant parameters of the CNN obtained through orthogonal experiments are shown in Table 1. C is the number of kinds of audio scenes.

3.2.3. Comparison of the recognition performance of various audio scene recognition algorithms for audio signals in complex environments

To validate the audio scene recognition algorithm, which combined sparse and MFCC features, it was compared with two other algorithms: one based solely on sparse features and the other based solely on MFCC. The parameters of the fusion feature-based recognition algorithm remained consistent with those mentioned earlier, and the optimal dictionary size was selected. The recognition algorithm based on sparse features did not incorporate MFCC features, while the recognition algorithm based on MFCC did not use sparse features.

3.3. Experimental Results

This article measured the performance of sparse feature extraction methods by evaluating the SNR between the original signal and the reconstructed signal using the extracted sparse features. A higher SNR indicated a greater similarity between the original and reconstructed signals, as shown in Table 2. It can be observed that as the dictionary size increased, both extraction algorithms initially showed an increase in SNR followed by a decrease. Furthermore, in the same dictionary specification, the sparse feature extraction method proposed in this paper exhibited a higher SNR compared to the matching pursuit-based method.

The recognition results of the algorithm, based on the fusion of sparse features and MFCC features, for two audio datasets under different dictionary sizes are depicted in Fig. 2. It can be observed from Fig. 2 that with an increase in dictionary size, the recognition accuracy of this algorithm initially rose and then declined for both datasets. However, the optimal recognition accuracy was achieved with different dictionary sizes for the two datasets. Specifically, for TUT Sound Events 2016, the highest recognition accuracy was attained with a dictionary size of 75, while for TUT Acoustic Scenes 2016, the peak recognition accuracy was observed with a dictionary size of 150.

The convergence curves of the algorithms based on sparse features, MFCC features, and sparse features + MFCC features during the training process are illustrated in Fig. 3. As observed from Fig. 3, the

Table 1. Relevant parameter settings for CNN.

Structure	Parameter	Structure	Parameter
Input layer	Audio feature dimension number × number of audio frames	Convolutional layer 1	32 convolution kernels (5 × 5), a step size of 2, and Relu activation function are used.
Convolutional layer 2	32 convolution kernels (3 × 3); a step size of 1; Relu activation function	Pooling layer 1	A 2 × 2 maximum pooling frame
Convolutional layer 3	64 convolution kernels (3 × 3); a step size of 1; Relu activation function	Convolutional layer 4	64 convolution kernels (3 × 3); a step size of 1; Relu activation function
Pooling layer 2	2 × 2 maximum pooling frame	Convolutional layer 5	128 convolution kernels (3 × 3); a step size of 1; Relu activation function
Convolutional layer 6	128 convolution kernels (3 × 3); a step size of 1; Relu activation function	Convolutional layer 7	128 convolutional kernels (3 × 3); a step size of 1; Relu activation function[18]
Pooling layer	A 2 × 2 maximum pooling frame	Convolutional layer 8	C convolutional kernels (1 × 1); a step size of 1; Relu activation function
Pooling layer	Global mean pooling	Output layer	Using the softmax function; C output channels

Table 2. The reconstruction SNR of the two sparse feature extraction methods under different dictionary sizes.

Dictionary size	50	100	150	200	250
The sparse feature extraction method based on matching pursuit	18.4 dB	22.3 dB	25.5 dB	21.9 dB	19.7 dB
The sparse representation feature extraction method	21.4 dB	25.9 dB	30.8 dB	26.3 dB	23.2 dB

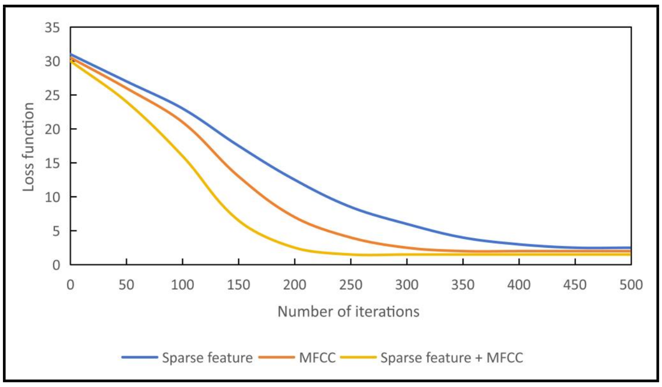

Figure 3. Convergence curves of three audio scene recognition algorithms during training.

loss function of the three recognition algorithms, employing CNN as the classifier, decreased with an increasing number of iterations. Notably, the recognition algorithm based on sparse features and MFCC features exhibited the fastest convergence, stabilizing after approximately 250 iterations. The recognition algorithm based on MFCC features was closely followed, and it stabilized after approximately 320 iterations. In contrast, the recognition algorithm based solely on sparse features converged the slowest, stabilizing after approximately 440 iterations.

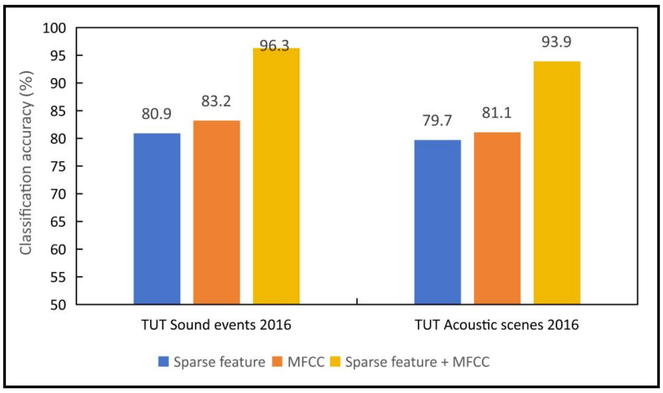

Figure 4. Recognition results of three algorithms for different datasets.

Table 3. Recognition results of three algorithms in different acoustic scenes.

Dataset	No. of acoustic scene	Based on sparse feature	Based on MFCC	Based on sparse feature + MFCC
TUT Sound events 2016	Scene 1	80.1	82.9	95.8
	Scene 2	79.8	83.3	96.7
TUT Acoustic scenes 2016	Scene 1	81.4	82.9	95.5
	Scene 2	80.7	82.3	95.7
	Scene 3	81.0	81.5	95.4
	Scene 4	79.9	81.7	94.8
	Scene 5	80.4	83.0	94.3
	Scene 6	80.3	83.0	94.3
	Scene 7	80.2	81.4	94.7
	Scene 8	80.5	81.7	95.2
	Scene 9	81.0	82.9	94.7
	Scene 10	81.4	81.3	94.3
	Scene 11	79.8	81.9	95.6
	Scene 12	81.0	82.8	95.2
	Scene 13	81.7	82.5	94.8
	Scene 14	80.2	82.1	95.6
	Scene 15	80.8	81.4	95.2

The recognition results of sparse feature-based, MFCC feature-based, and sparse feature + MFCC feature-based audio scene recognition algorithms for the TUT Sound Events 2016 and TUT Acoustic Scenes 2016 datasets are depicted in Fig. 4. The recognition results in various acoustic scenarios are shown in Table 3. As observed from Fig. 4, irrespective of the dataset used, the classification accuracy of the algorithm was highest when the sparse features + MFCC features were used. The two recognition algorithms, based on sparse features and MFCC features, respectively, exhibited comparable performance. When comparing the recognition performance of the same algorithm for different datasets, it is evident that the algorithm achieved higher accuracy for the TUT Sound Events 2016 dataset. The data in Table 3 also shows the same trend. This discrepancy is because the TUT Sound Events 2016 dataset comprised only two acoustic scenes, whereas the other dataset included 15 acoustic scenes. The higher number of scenes in the latter dataset increased the susceptibility of the recognition algorithm to misjudgments.

4. CONCLUSION

This paper combined the sparse representation algorithm with MFCC features for a CNN classifier to recognize acoustic scenes in audio signals within complex environments. The proposed algorithm was then subjected to simulation experiments. In the experiments, the efficacy of the sparse feature extraction method proposed in this paper was validated. Then, the optimal sparse dictionary size was determined, and the performance of this algorithm was compared with two other algorithms - one based on sparse features and the other on MFCC features. The key findings are as follows. (1) As the size of the dictionary increases, both extraction algorithms show a trend of initially increasing and then decreasing SNR. However, the method combining sparse features with MFCC features had a higher SNR for the same dictionary size. (2) The highest recognition accuracy was achieved with a dictionary size of 75 for the TUT Sound Events 2016 dataset and 150 for the TUT Acoustic Scenes 2016 dataset. (3) During the training, the loss function of the three audio scene recognition algorithms decreased as the number of iterations increased. Specifically, the feature fusion-based algorithm converged and stabilized after approximately 250 iterations, the one based on MFCC features stabilized after about 320 iterations, and the one based on sparse features stabilized after around 440 iterations. (4) For both datasets and the 17 acoustic scenes, the recognition algorithms based on feature fusion achieved the highest classification accuracy. Additionally, the recognition algorithm exhibited slightly higher accuracy for the TUT Sound Events 2016 dataset compared to the other one.

The future research direction involves further enhancing the performance of the sparse feature extraction algorithm to make the signal after the sparse feature reconstruction is closer to the original signal. The goal is to improve and enhance the recognition and classification performance of audio signals.

REFERENCES

[1] Xu, L., Huang, D., Guo, X., Rao, W., Ji, Y., Li, R., and Lu, X. A novel robust zero-watermarking algorithm for audio based on sparse representation, *China Commun.*, **18** (8), 237–248, (2021). https://doi.org/10.23919/JCC.2021.08.017

[2] Xu, J., and Xia, J. Digital audio resampling detection based on sparse representation classifier and periodicity of second derivative, *J. Digit. Inform. Manag.*, **14** (2), 101–109, (2015).

[3] Priya, B., and Dandapat, S. Sparse representation of LPC for analysis of stressed speech in lower dimensional subspace, *IEEE Region 10 Conference*, Singapore, (2016). https://doi.org/10.1109/TENCON.2016.7848085

[4] Liang, Y., and Chen, K. Sparse feature extraction and sound source classification for impact sounds, *Acta Acust.*, **43** (4), 708–718, (2018).

[5] Rao, Z., and Feng, C. Sparse representation classification-based automatic chord recognition for noisy music, *J. Inf. Hiding Multimed. Signal Process.*, **9** (2), 400–409, (2018).

[6] Zou, L., He, Q., and Wu, J. Source cell phone verification from speech recordings using sparse representation, *Digit. Signal Process.*, **62**, 125–136, (2017). https://doi.org/10.1016/j.dsp.2016.10.017

[7] Mishra, R. K., Choudhary, A., Fatima, S., Mohanty, A.R., and Panigrahi, B.K. A generalized method for diagnosing multi-faults in rotating machines using imbalance datasets of different sensor modalities. *Engineering Applications of Artificial Intelligence*, **132**, 107973, (2024). https://doi.org/10.1016/j.engappai.2024.107973

[8] Mishra, R.K., Choudhary, A., Fatima, S. Mohanty, A. R., and Panigrahi, B. K. Multi-fault diagnosis of rotating machine under uncertain speed conditions. *J. Vib. Eng. Technol.*, (2023). https://doi.org/10.1007/s42417-023-01141-x

[9] Kwek, L. C., Tan, W. C., Lim, H. S., Tan, C. H., and Alaghbari, K. A. Sparse representation and reproduction of speech signals in complex Fourier basis, *Int. J. Speech Technol.*, **25** (1), 211–217, (2022). https://doi.org/10.1007/s10772-021-09941-w

[10] Dighe, P., Asaei, A., and Bourlard, H. Sparse modeling of neural network posterior probabilities for exemplar-based speech recognition, *Speech Commun.*, **76**, 230–244, (2016). https://doi.org/10.1016/j.specom.2015.06.002

[11] Isnard, V., Suied, C., and Lemaitre, G. Auditory bubbles reveal sparse time-frequency cues subserving identification of musical voices and instruments, *Acoust. Soc. Am. J.*, **140** (4), 3267–3267, (2016). https://doi.org/10.1121/1.4970361

[12] He, Y., Sun, G., and Han, J. Spectrum enhancement with sparse coding for robust speech recognition, Digit. *Signal Process.*, **43**, 59–70, (2015). https://doi.org/10.1016/j.dsp.2015.04.014

[13] Wu, H., and Sangaiah, A. K. Oral English speech recognition based on enhanced temporal convolutional network, *Intell. Autom. Soft Co.*, **28** (1), 121–132, (2021). https://doi.org/10.32604/iasc.2021.016457

[14] Evers, K., and Chen, S. Effects of automatic speech recognition software on pronunciation for adults with different learning styles, *J. Educ. Comput. Res.*, **59** (4), 669–685, (2021). https://doi.org/10.1177/0735633120972011

[15] Cao, Q., and Hao, H. Optimization of intelligent English pronunciation training system based on android platform, *Complexity*, **2021** (4), 1–11, (2021). https://doi.org/10.1155/2021/5537101

[16] Zhan, W., and Chen, Y. Application of machine learning and image target recognition in English learning task, *J. Intell. Fuzzy Syst.*, **39** (4), 5499–5510, (2020). https://doi.org/10.3233/JIFS-189032

[17] Chu, S., Narayanan, S., and Kuo, C.C. J. Environmental sound recognition with time-frequency audio features, *IEEE T. Audio Speech*, **17** (6), 1142–1158, (2009). https://doi.org/10.1109/TASL.2009.2017438

[18] Sui, P. Research on interactive English speech recognition algorithm in multimedia cooperative teaching, *Int. English Educ. Res.*, (1), 79–82, (2018).