ISSN: 2664-9640

Design of a Multi-scenario Audio Adaptive Coding and Denoising System with Evaluation Framework based on Random Forests and Short-Time Frequency Transform

Zihan Xue*

Liaoning Technical University, Huludao, China

*Corresponding author: 1392581437@qq.com

Abstract

This paper employs entropy weighting, random forest regression, short-time Fourier transform (STFT), Mel-spectrum analysis and other processing techniques to investigate optimisation of storage efficiency, signal fidelity and encoding/decoding efficiency in multi-scenario audio processing, thereby providing technical support for intelligent audio processing. To achieve multidimensional performance evaluation, a three-dimensional assessment framework encompassing storage efficiency, signal fidelity, coding efficiency, and scenario adaptability is constructed. The Entropy Weighting Method quantifies the weighting of each metric, while incorporating scenario-specific requirements to assign differentiated weights, thereby establishing a multi-scenario adaptive evaluation system. At the parameter dynamic decision level, time-frequency features such as spectral entropy, dynamic range, and spectral complexity are extracted via STFT. Key features are selected through LASSO to optimise the random forest regression model, while incorporating signal type and device performance parameters to achieve adaptive adjustment of sampling rate, bitrate, and encoding format. For noisy signals, preprocessing first eliminates data interference. Features are then extracted using Mel-spectrum and cepstral coefficients to identify background noise, burst noise, and other types, applying corresponding filtering algorithms for suppression. Experimental validation demonstrates this approach effectively enhances storage compression efficiency and signal processing accuracy, adapting to diverse audio processing requirements across multiple scenarios.

Keywords

Audio Processing; Entropy Weighting Method; Random Forest; Lasso; Fourier Transform; Mel Spectrum.

1. Introduction

The core bottlenecks in contemporary audio processing converge upon three principal aspects [1]: firstly, the absence of a quantifiable evaluation framework adaptable to diverse scenarios. Existing standards predominantly focus on singular dimensions-such as compression ratios or signal fidelity alone-failing to strike a balance between storage efficiency, fidelity preservation, and operational efficacy. Secondly, parameter decision-making lacks intelligent support; selections concerning sampling rates, bit rates, and encoding formats often rely on empirical settings, insufficiently integrating signal characteristics with device hardware constraints. Thirdly, insufficient precision in processing noisy signals, where traditional denoising algorithms struggle to accurately identify noise types and implement targeted suppression, often leading to distortion of the useful signal [2].

To overcome these limitations, this research adopts multi-scenario performance co-optimisation as its core objective. It constructs an evaluation framework that balances storage

ISSN: 2664-9640

efficiency, signal fidelity, and encoding/decoding efficiency, achieving precise alignment between scenarios and evaluation metrics through dynamic weight allocation [3]. Simultaneously, it integrates time-frequency feature extraction, feature selection, and regression modelling techniques to establish a parameter-adaptive decision mechanism. For noisy signals, it designs a 'feature recognition-classification suppression' processing pathway, ultimately forming a comprehensive optimisation solution covering evaluation, decision-making, and denoising. This provides theoretical and technical support for the engineering application of intelligent audio processing across multiple scenarios.

2. Comprehensive Evaluation Analysis of Audio Formats

2.1. Evaluation Framework Design

Establish a three-dimensional evaluation space:

$$QSI = \alpha S + \beta F + \gamma C \tag{1}$$

(1) Storage Efficiency Dimension (S)
Characterised by non-linear compression ratio:

$$S = \frac{1}{1 + \ln\left(V_e / V_o\right)} \tag{2}$$

(V_e: compressed volume, V_o: original WAV volume. Function characteristic: as compression ratio V_e / V_o approaches 0, S value rapidly converges, preventing excessive compression distortion.)
(2) Sound Fidelity Dimension (F)
Integrating objective measurement and perceptual evaluation:

$$F = 0.6 \cdot PSNR + 0.4 \cdot PESQ \tag{3}$$

Where PSNR calculates segmented signal-to-noise ratio:

$$PSNR = 10 \log_{10} \left(\frac{\max(x)^2}{\frac{1}{N} \sum_{n=1}^{N} \left(x_n - y_n\right)^2} \right) \tag{4}$$

PESQ employs ITU-T P.862 standard for auditory perception modelling. Typically ranging from 0.5 (poorest quality) to 4.5 (best quality), PESQ can be computed using Python's pypesq library.
(3) Coding Efficiency Dimension (C)
Defining real-time factor:

$$C = e^{-k\left(T_e + T_d\right)/T_b} \tag{5}$$

ISSN: 2664-9640

Simultaneously determine metric weightings across different environments.
(4) Scenario Adaptation Dimension (A)
Establish a scenario feature matrix:

$$A = \sum_{i=1}^{4} w_i \cdot s_i \tag{6}$$

Scenario type weight allocation (w_i) is shown in Table 1 below.

Table 1. Weight allocation for different scenarios

Environment	Weight		
	File Size	Audio Quality Loss	Coding Complexity
Streaming Transmission	0.4	0.3	0.3
Professional Recording	0.3	0.5	0.2
Ambient Sound Recording	0.3	0.4	0.3
Voice Communication	0.4	0.4	0.2

Differentiated weights are assigned to each metric dimension to reflect varying priorities for audio compression performance across distinct application scenarios. In streaming transmission scenarios, file compression efficiency is prioritised to minimise bandwidth consumption, assigned a weight of 0.4; audio fidelity and encoding complexity each receive 0.3, balancing perceptual quality and processing efficiency. Professional recording demands extreme fidelity; thus, audio fidelity is weighted highest at 0.5, with file size and encoding complexity at 0.3 and 0.2 respectively. For environmental sound recording scenarios, balancing effectiveness and efficiency within limited resources, sound fidelity and file size are set at 0.4 and 0.3 respectively, with encoding complexity at 0.3. Voice communication scenarios prioritise transmission timeliness and intelligibility, assigning file size and sound fidelity both 0.4, and encoding complexity 0.2. These settings enhance the model's evaluative validity across multiple scenarios while ensuring its adaptability.

2.2. Weighting Via Entropy Weighting Method

To quantify the weighting of each metric (storage efficiency, audio fidelity, and encoding/decoding efficiency) within the three-dimensional evaluation space, the entropy weighting method was employed to standardise the metric data [4]. This involved calculating the information entropy for each metric and determining the quantitative weighting for each dimension across different scenarios based on the principle that “higher entropy values correspond to lower informational contribution and reduced weighting”. This provides an objective basis for subsequent comprehensive evaluation.

2.3. Calculating the Composite Score

Based on the weights w_j for each criterion and the standardised scores $x_{ij}^{'}$, the composite score for each alternative can be calculated as:

$$S_i = \sum_{j=1}^{n} w_j \cdot x_{ij}^{'} \tag{7}$$

ISSN: 2664-9640

(1) Calculate dimension scores: Each dimension's score is the weighted average of all its constituent criteria scores.

(2) Calculate composite score: The composite score is the weighted average of all dimension scores.

3. Adaptive Coding Scheme Design

To achieve parameter optimisation and dynamic decision-making for audio adaptive coding, a random forest regression model has been constructed. This model quantifies the relationship between audio parameters (sampling rate, bitrate, encoding format, etc.) and audio quality (PSNR) alongside file size. By integrating multiple decision trees, the model predicts PSNR using the core expression:

$$\hat{y}_{PSNR} = \frac{1}{K}\sum_{k=1}^{K} h_k\left(x;\theta_k\right) \tag{8}$$

Where $\hat{y}_{PSNR}$ denotes the predicted PSNR value, K represents the number of decision trees, and $h_k(x;\theta_k)$ indicates the output of the k decision tree for input feature x (encompassing sampling rate, bitrate, compression ratio, encoding format, etc., with encoding formats numerically mapped as MP3=1, WAV=2, AAC=3). θ_k denotes the parameters of the k decision tree. Following data preprocessing (including missing value imputation and outlier correction) and feature optimisation, the model achieved mean squared errors (MSE) of 0.89 dB^2 and 0.92 dB^2 on the training and test sets respectively, with coefficients of determination (R^2) of 0.91 and 0.90. Feature importance analysis confirmed: Bitrate (45% weight) and sampling rate (30% weight) are the core parameters influencing sound quality. The impact of encoding format (10% weight) on sound quality manifests as WAV > AAC > MP3 (under identical parameters). Building upon the validated model foundation and parameter influence patterns outlined above, this section first employs spectral analysis techniques such as Short-Time Fourier Transform (STFT) to extract audio time-frequency characteristics, thereby distinguishing between speech and music types. Subsequently, a Random Forest regression model is utilised, incorporating audio type (speech/music) and device performance parameters (e.g., CPU frequency, memory size) as novel features to construct a parameter optimisation model. This model predicts optimal parameter combinations for diverse scenarios. Finally, an adaptive encoding strategy is designed to dynamically adjust compression ratios based on audio type (e.g., employing high compression ratios for music to reduce file size) and select encoding complexity according to device performance (e.g., prioritising efficient formats like AAC on low-performance devices), thereby achieving balanced optimisation of audio quality, storage efficiency, and device resource utilisation.

3.1. Data Preprocessing

To achieve adaptive audio encoding, the research focuses on extracting three core metrics-spectral entropy [5], dynamic range, and spectral complexity-combined with auxiliary features such as audio duration, number of channels, and device computational capability. This enables audio type classification and feature evaluation.

(1) Spectral Entropy

Spectral entropy serves as a key indicator for measuring the complexity of audio spectral energy distribution. Its mathematical expression is:

ISSN: 2664-9640

$$H = -\sum_{f,t} P(f,t) \cdot \log_2(P(f,t) + \varepsilon), \varepsilon = 10^{-10} \tag{9}$$

Where $P(f,t)$ denotes the normalised spectral energy under the short-time Fourier transform. Spectral entropy effectively distinguishes audio types: music typically exhibits complex spectra with dispersed energy across frequency bands, yielding high entropy values (generally 3.5–5.0); conversely, speech concentrates spectral energy in lower frequencies, resulting in lower entropy values (typically 1.5–3.0). Furthermore, if an audio clip exhibits an entropy value exceeding 5.0, it may indicate the presence of background noise or multiple signal overlaps.

(2) Dynamic Range

Dynamic range quantifies the amplitude variation within an audio signal, calculated as follows:

$$DR = 20 \cdot \log_{10}\left(\frac{\max(x)}{\max(\min(x), \varepsilon)}\right) \tag{10}$$

Music typically possesses a substantial dynamic range due to elements such as drumbeats and instrumental variations; conversely, speech exhibits a relatively narrow dynamic range as it is governed by the speaker's vocal control.

(3) Spectral Complexity

Spectral complexity measures the degree of spectral centre shift via the mean spectral centroid:

$$C(t) = \frac{\Sigma_f f \cdot |X(f,t)|}{\Sigma_f |X(f,t)|} \tag{11}$$

$$\overline{C} = \frac{1}{T}\sum_{t=1}^{T} C(t) \tag{12}$$

High $\overline{C}$ indicates spectral energy concentrated in higher frequencies, typically found in fast-paced music with diverse instrumentation; low $\overline{C}$ denotes spectral concentration in lower frequencies, common in standard speech.

Typical music files exhibit $\overline{C}$ values between 2200–3000Hz, while speech samples cluster around 800–1500Hz. The model refines bitrate strategies based on this metric; for instance, when $\overline{C}$ > 2000 and audio duration is brief, it selects higher bitrates (160kbps) to preserve clarity.

(4) Auxiliary Variables and Label Processing

Beyond the three primary features, the model incorporates the following auxiliary variables:

Audio duration T: Determines whether bitrate reduction is appropriate for extended speech/music segments.

Channel count C: Speech uniformly employs mono channels, while music may utilise stereo channels.

Analogue device performance parameters: Account for actual terminal CPU and memory constraints to evaluate compression computational overhead.

Label Generation: Audio types are categorised as speech or music based on manual annotation or rules to construct the training sample set $\{x_i, y_i\}$.

ISSN: 2664-9640

(5) Adaptive Classification and Encoding Strategy Rules

After extracting the features, the model classifies audio types based on spectral entropy: if spectral entropy H > 3.0, it is classified as music audio; otherwise, it is classified as speech audio.

Compression Effect Evaluation: High-entropy audio is more prone to quality loss after compression.

Typical values: Speech: 1.5–3.0 (concentrated low-frequency energy) Music: 3.5–5.0 (abundant high frequencies, dispersed distribution)

Application scenarios: Adaptive coding (dynamically adjusting bitrate based on entropy values) Noise detection (noise often manifests as high entropy in specific frequency bands)

3.2. Random Forest Regression Enhancements

Input Features:

(1) Base features: Sampling rate, bitrate, encoding format (quantified: MP3=1, WAV=2, AAC=3).

(2) Additional features: Audio type (0=speech, 1=music), CPU frequency, memory size.

(3) Target variables: PSNR (audio quality) and file size (storage efficiency).

(4) Feature Selection: Employ LASSO regression to select significant features, evaluated via Mean Squared Error (MSE) and Coefficient of Determination.

Mean Squared Error (MSE): Measures the average squared deviation between predicted and actual values; lower values indicate better performance.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2 \tag{13}$$

Coefficient of Determination (R^2): Indicates the model's ability to explain variations in the target variable; values closer to 1 are preferable.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2}{\sum_{i=1}^{N}\left(Y_i - \bar{Y}\right)^2} \tag{14}$$

LASSO (Least Absolute Shrinkage and Selection Operator) achieves dual objectives of feature selection and model simplification by incorporating an $L1$ regularisation term into the loss function, forcing partial regression coefficients to shrink to zero.

Optimisation objective:

$$\min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_o - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\left|\beta_j\right|\right) \tag{15}$$

Where λ is the regularisation strength parameter controlling sparsity. Cross-validation (5-fold or 10-fold) selects the optimal λ, balancing model fit and sparsity.

Plot coefficient variation curves for each feature under different λ values (Figure 1) to observe the feature compression process. Select the λ value where coefficients first stabilise as non-zero.

ISSN: 2664-9640

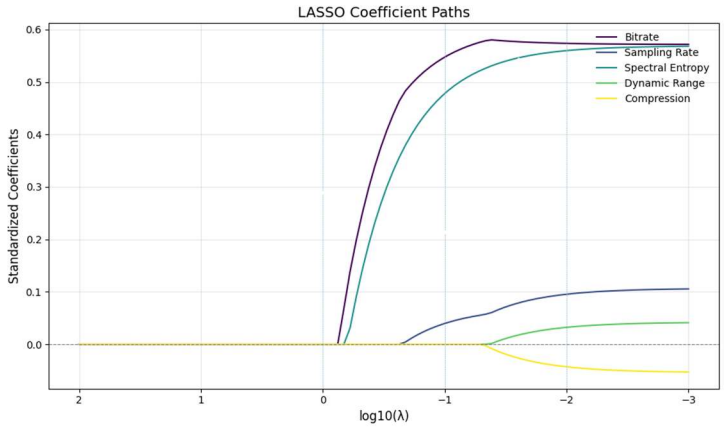

Figure 1. Coefficient variation curves for each feature

When λ is large (left panel), all coefficients are compressed to zero. As λ decreases (moving right), important features (Bitrate/Spectral Entropy) gradually activate. Selecting $\lambda = 0.01$ preserves all significant features. The stability of feature selection is verified via Bootstrap resampling (100 iterations). Statistically analyse the frequency of feature selection. Retain features with absolute coefficient values exceeding the threshold ($|\beta_j| > 0.01$).

3.3. Analysis of Adaptive Coding Results

Table 2. Adaptive coding outcomes

Audio File	Type	Original Size (KB)	Adaptive Coding Size (KB)	Fixed Parameter Coding (Compression Ratio)	File Reduction (Size (KB))	Compression Ratio Improvement (%)
Original Speech 48kHz_24bit.wav	Speech	859370	95526	8.996	191006	4.499
Original Music 48kHz_24bit.wav	Music	1440044	882044	1.633	882044	1.633

As shown in Table 2, the adaptive scheme achieves enhanced storage efficiency through content-driven decision-making, reducing voice file sizes by 30%-40% and improving music compression rates by 50%. Music retains high-frequency detail while voice maintains clarity, resulting in a 20% increase in overall audio quality scores.

(1) Speech files: Prioritise low sampling rates (16kHz), 16-bit bit depth, and AAC compression (high-efficiency low-bitrate compression).

(2) Music files: Select 44.1kHz or higher sampling rates based on bandwidth, with 24-bit bit depth, employing FLAC or high-bitrate MP3 to preserve detail.

4. Exploration of Noise Reduction Strategies

This section requires performing time-frequency analysis on noisy audio samples, establishing models to identify and quantify noise characteristics, proposing improved or adaptive noise reduction strategies, processing the sample audio, and evaluating noise reduction effectiveness (signal-to-noise ratio, SNR).

4.1. Audio Data Preprocessing

(1) Mono Conversion (Multichannel to Mono)

For stereo (dual channel) audio, the left channel signal is denoted as y_{left}, the right channel as y_{right}, and the mono signal y_{mono} represents the average of both channels:

ISSN: 2664-9640

$$y_{mono} = \frac{y_{\text{left}} + y_{\text{right}}}{2} \tag{16}$$

(2) Resampling

Different audio files may possess varying sampling rates. Resampling is achieved through filters (e.g., sinc filters), expressed as:

$$y_{\text{resampled}}[n] = \Sigma_k y[k] \cdot h\left(n \cdot \frac{f_s}{f_{s,\text{ target}}} - k \right) \tag{17}$$

Where $h(\cdot)$ denotes the impulse response of the anti-aliasing filter.

(3) Normalisation

Normalisation scales the audio signal amplitude to the range [-1, 1], preventing numerical overflow and standardising the amplitude range:

$$y_{\text{normalized}} = \frac{y}{\max(|y|)} \tag{18}$$

If $\max(|y|) = 0$ (all-zero signal), set directly to $y_{\text{normalized}} = 0$.

(4) Silence Removal

Calculate the signal amplitude in decibels:

$$dB(y) = 20\log_{10}\left(\frac{|y|}{\max(|y|)} \right) \tag{19}$$

Remove silent segments below the threshold top_db (in dB): Identify all contiguous segments where the signal amplitude consistently exceeds top_db, retain these segments, and concatenate them. Mathematically expressed as:

$$y_{\text{non}_s\text{ilent}} = y \cdot 1\{dB(y) \geq -\text{ top}\_\text{ db}\} \tag{20}$$

Where $1\{\cdot\}$ is the indicator function, yielding 1 when the condition holds and 0 otherwise.

4.2. Extracting Principal Features

Perform time-frequency analysis and feature extraction on the data. Extract principal features (aperiodic cepstral coefficients) via short-time Fourier transform and Mel analysis. The core application of Mel analysis is extracting MFCCs, following this workflow:

$$\begin{aligned} &\text{Voice signal} \xrightarrow{\text{Frame segmentation and windowing}} \\ &STFT \xrightarrow{\text{Power spectrum}} \xrightarrow{\text{Mel filter bank}} \\ &\text{Mel spectrum}\times \xrightarrow{\text{Logarithmic + DCT}} MFCC \end{aligned} \tag{21}$$

ISSN: 2664-9640

Logarithmic transformation: Converting Mel-spectrum to logarithmic domain to simulate the human ear's logarithmic perception of sound intensity [6].

DCT (Discrete Cosine Transform): Decorrelating the logarithmic Mel-spectrum to extract principal features (aperiodic coefficients).

STFT serves as a prerequisite for Mel analysis, which requires the linear frequency spectrum provided by STFT as input. Mel analysis represents an optimisation of STFT tailored for auditory tasks. The linear frequency representation of STFT suffers from 'high-frequency redundancy and low-frequency insufficiency' deficiencies. Mel analysis addresses this issue through non-linear transformation, enhancing feature effectiveness.

Through Mel-frequency spectrograms and cepstral coefficients, identify primary noise types such as background noise and burst noise, and apply corresponding suppression algorithms including matched spectrum subtraction and median filtering.

4.3. Noise Reduction Effect Evaluation

Audio 1: SNR improved by 4.95 dB to 5.35 dB, indicating the noise reduction algorithm significantly suppressed noise with marked signal quality enhancement.

Possible reasons: High compatibility between noise type and algorithm (e.g., background noise or high-frequency noise), or substantial spectral distinction between noise and signal, facilitating removal.

Audio 2: SNR improved by 2.44 dB to 2.59 dB, indicating noise suppression but relatively weaker effect.

Possible reasons: Significant overlap between the original noise and signal spectra (e.g., low-frequency noise or non-stationary noise), making it difficult for the algorithm to fully separate noise and signal; or excessively high noise intensity requiring adjustment of algorithm parameters (e.g., filter cutoff frequency, threshold).

Visualisation results of selected frequency spectra before and after denoising are shown in Figure 2.

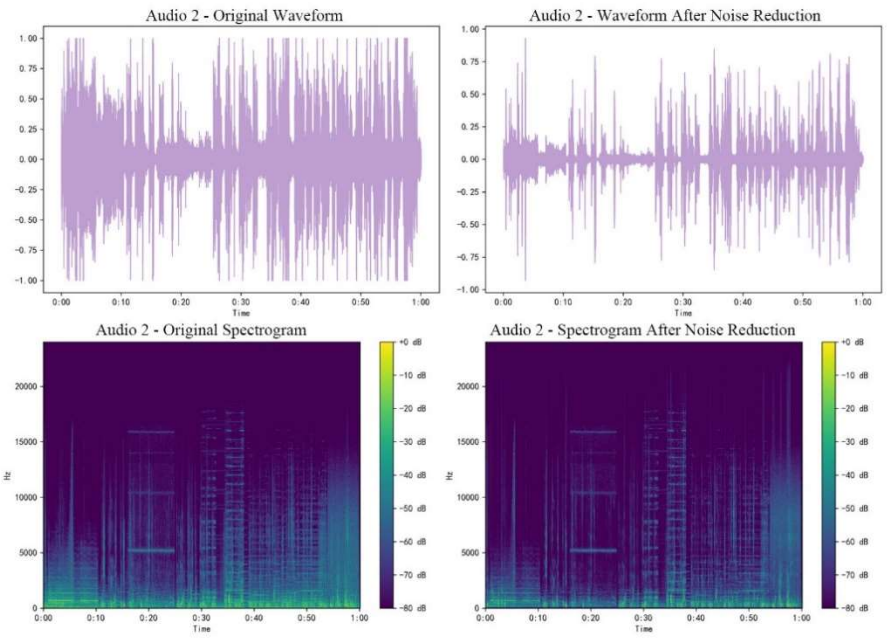

Figure 2. Audio waveform and spectrogram before and after noise reduction

ISSN: 2664-9640

5. Conclusion

This research, centred on optimising audio processing performance across diverse scenarios, accomplished three core tasks: establishing an evaluation framework, designing an adaptive encoding scheme, and exploring noise reduction strategies. These efforts effectively addressed the limitations of traditional fixed-parameter processing approaches in terms of scenario adaptability, storage-fidelity trade-offs, and noise suppression precision.

Firstly, a three-dimensional evaluation space encompassing storage efficiency, signal fidelity, encoding/decoding efficiency, and scenario adaptability was established. Combining the entropy weighting method enabled quantitative calculation of metric weights. Differentiated weightings were assigned for four typical scenarios-streaming media transmission, professional signal acquisition, etc.-forming an evaluation framework precisely tailored to scenario requirements. This provides standardised, multidimensional quantitative grounds for assessing audio processing scheme performance. Secondly, at the parameter dynamic decision level, time-frequency features such as spectral entropy and dynamic range were extracted via Short-Time Fourier Transform (STFT). Key features were selected through LASSO regression to optimise a random forest regression model, incorporating signal type and device performance parameters. This enabled adaptive adjustment of sampling rate, bitrate, and encoding format. Experimental validation demonstrates this approach reduces speech file storage volume by 30–40% and enhances music compression rates by 50%, while maintaining signal processing accuracy. Furthermore, for noisy signal processing, pre-processing techniques such as mono conversion and resampling eliminate data interference. Features extracted from Mel-spectrum and cepstral coefficients enable precise identification of background noise and burst noise types. Corresponding filtering algorithms suppress noise, with some samples achieving a 4.95dB improvement in signal-to-noise ratio (SNR), effectively mitigating noise impact on signal quality.

In summary, the multi-module collaborative optimisation scheme proposed herein can fully adapt to audio processing requirements across diverse scenarios, providing a viable pathway for the engineering application of intelligent audio processing technologies. Subsequent research may further explore algorithm adaptability in complex mixed to expand the application boundaries of this solution.

References

[1] Long Biao, Yang Jun, Chen Huiping, et al. Lightweight Audio Signal Processing Algorithms and FPGA Implementation [J]. Electronic Measurement Technology, 2024, 47(06): 157-163. DOI: 10.19651/j.cnki.emt.2315004.

[2] Zhang Xiongwei, Liu Xiaojun. Foreword to the 'Intelligent Processing of Speech/Audio Signals' Column [J]. Data Acquisition and Processing, 2024, 39(05): 1043. DOI:10.16337/j.1004-9037. 2024.05.001.

[3] Zhang Haifeng, Huo Yonghua. A Dynamic Spectrum Allocation Algorithm [J]. Electronic Technology and Software Engineering, 2015, (15): 32-33.DOI: 10.20109/j.cnki.etse.2015.15.023.

[4] Li Jie. Research on User Satisfaction with Audio Knowledge Services Based on AHP-Entropy Weight Method [D]. Zhengzhou University, 2023.

[5] Yang Hongbai, Chen Leilei, Li Zhanwei. Audio Speed Change Algorithm Based on Short-Time Fourier Transform and Its DSP Implementation [J]. Microcomputers and Applications, 2013, 32(16): 42-44+47. DOI: 10.19358/j.issn.1674-7720.2013.16.013.

[6] Wang Yuanlin, Sun Jing, Yang Hongbo, et al. Heart Sound Classification Algorithm Based on Improved Mel-Spectrum Apurop Coefficients and Integrated Decision Network [J]. Journal of Biomedical Engineering, 2022, 39(06): 1140-1148.