International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 2626-2642

https://doi.org/10.70517/ijhsa464220

Real-time audio source separation in live music performances using edge AI technology

Jingjing Yang[1,*]
[1] School of Norms and Education, Jingchu Institute of Science and Technology, Jingmen, Hubei, 448000, China
Corresponding authors: (e-mail: yjj_2025@163.com).

Abstract With the improvement of computer performance, audio processing technology has also made tremendous progress. In recent years, edge AI technology has been used in audio signal separation research, becoming an increasingly popular topic in the field of audio signal processing and driving the development of source separation based on deep learning technology. After clarifying the basic theories of music source separation and the preprocessing workflow of audio signals, the study incorporates an attention mechanism and employs a dual-gate mechanism to better control the flow of feature information across different convolutional layers, filtering out unnecessary feature information to achieve effective audio source separation in live music performances. The research results indicate that the proposed algorithm achieves a performance improvement of approximately 4 dB to 10 dB compared to HPSS in terms of SIR values, and at least a 1 dB improvement compared to the REPET algorithm, thereby demonstrating that the proposed method is a more effective separation approach.

Index Terms audio source separation, edge AI technology, attention mechanism, dual-gate mechanism, music performance

I. Introduction

In everyday life, almost all the sounds people hear are combinations of many different voices. For example, these can be human conversations, songs, or sounds from nature such as wind, or noise such as car horns. These sound sources are composed of relatively simple tones, and multiple sounds may coexist within the same medium [1], [2]. In such environments, listeners may be interested in identifying the individual sources within the existing sounds. Therefore, listeners face the task of separating or extracting the source of interest from the mixture. The cognitive capabilities of the human auditory system enable people to follow the movement of a speaker in a noisy environment without affecting sound quality, a capability that is highly similar to sound source separation technology [3], [4]. The formal definition of the sound source separation task is to identify and separate different sound sources using a reasonable model [5]. Although this task is natural and easy for humans, developing algorithms to automatically perform the same task is challenging [6], [7].

Music is an important form of artistic expression and plays a significant role in the entertainment industry. Digitalization and the internet have brought about a major transformation in music dissemination methods [8]. Audio source separation models are widely applied in tasks such as music lyric alignment, lyric transcription, music transcription, and vocal melody extraction [9], [10]. Additionally, numerous music-related multimedia applications have embedded music source separation functionality within their software [11], [12]. In live music performance recording, various applications have already achieved the ability to interact with individual audio objects, such as music mixing, remixing, and object equalization [13]-[15]. Most publicly available music performances are conducted in monaural or stereo combinations, where multiple sound objects share a single audio track [16], [17]. This has made the development of a high-performance, reliable music source separation system an urgent issue, attracting significant research attention in recent years.

Currently, audio source separation has been extensively studied and applied in downstream tasks, and the various source separation methods proposed have achieved high separation accuracy. Takahashi, N, and Mitsufuji, Y extended the DenseNet model by introducing a sampling layer, block-jumping connections, and dedicated dense blocks, enabling it to handle complex and ill-posed audio source separation tasks [18]. Févotte, C., et al. investigated the application of spectral decomposition techniques based on non-negative matrix factorization (NMF) in multi-audio signal processing tasks, formulating it as an optimization problem to enable the model to achieve good audio decomposition capabilities in both unsupervised and supervised environments [19]. Sawada, H et al. developed an independent low-rank matrix analysis technique for audio blind source separation, combining independent vector analysis (IVA) and multi-channel non-negative matrix factorization (MNMF) to achieve high

performance in mixed source separation [20]. Michelsanti, D et al. explored deep learning techniques applied to speech enhancement and speech separation tasks, integrating multi-modal audio-visual information to reconstruct audiovisual speech, significantly enhancing audio separation performance in mixed signals [21]. Kavalerov, I et al. compared the performance of short-time Fourier transform (STFT) and time-domain enhancement networks (ConvTasNet) in general sound separation tasks, finding that STFTs can achieve mixed sound separation with scale-invariant and low information distortion characteristics [22].

Since music itself is a combination of different frequency components at different times, how to utilize computers to identify these features and apply them to the analysis and recognition of music signals is a highly meaningful research direction. Chandna, P., et al. proposed a low-latency single-source separation framework based on convolutional neural networks (CNNs), which relies on CCNN to estimate time-frequency soft masks, demonstrating excellent separation performance and processing efficiency in mixed music audio separation tasks [23]. Luo, Y, and Yu, J designed a frequency-domain analysis model based on a band-splitting recurrent neural network (BSRNN), which achieves better instrument track separation results by segmenting and modeling mixed music signals [24]. Yar, G et al. introduced the application of speech conversion and audio source separation fusion technology in the music industry, using Demucs as the audio source separation model and a random neural network as the speech conversion model. The proposed fusion model enables arbitrary conversion between singers and songs [25]. Pardo, B et al. introduced two methods for audio source separation in music audio. The first method uses repetitive elements in the music scene as separation conditions, while the second method tracks the pitch of the audio stream to achieve melody-based audio separation [26]. Slizovskaia, O et al. demonstrated that sound sources in musical works share tonal characteristics, and proposed adjustment techniques at different levels in the main source separation network. By adding instrument information and video stream data, they improved the quality of audio source separation [27].

The study first introduces the relevant theoretical knowledge of music source separation, including basic music theory knowledge such as the frequency ranges of vocal, string, and percussion instruments, as well as music signal processing (the necessity of short-time Fourier transform in music signal processing, homomorphic processing of spectrograms, and some basic features of MFCC), and briefly analyzes the time-frequency domain characteristics of audio. Next, a cross-channel attention mechanism is employed to enhance the encoder's ability to extract audio features in both the time and frequency domains, thereby improving the model's separation performance and enabling audio source separation in live music performances. Finally, relevant experiments are designed to validate the method's superior performance in separating accompaniment and vocals in music.

II. Audio feature processing and speech separation models in music performance

II. A.Basic Music Theory Knowledge

Music, like the history of human civilization, has a long history and has become an integral part of people's lives. Different types of music, due to their distinct developmental histories and cultural backgrounds, exhibit unique characteristics. A complete piece of music is typically composed of various sounds and instruments. Based on the principle of sound production, musical instruments can be categorized into two types: string instruments and percussion instruments. String instruments are those that produce sound through the vibration of strings. Known for their warm and melodious tones, they are frequently used in modern and classical music to create captivating melodies. Common string instruments include the erhu, guitar, and piano, which form the traditional instrumental ensemble. Percussion instruments are those that produce sound through playing, shaking, rubbing, or scraping, such as drums, xylophones, and triangles. In the study of musical signals, overtones are often used to distinguish differences between various instruments. The high-frequency range audible to the human ear is 12,000 Hz to 20,000 Hz, which is also the high-frequency range of some instruments, while the fundamental frequency range of the human voice is 500 Hz to 1,000 Hz.

II. B.Music Signal Processing

II. B. 1) Short-time Fourier transform

Currently, there is limited research on source separation of speech signals, and source separation of audio signals is rarely performed in the time domain; therefore, mixed signals are typically processed in the frequency domain. This paper focuses on two signal representations: the first is provided by the short-time Fourier transform (STFT) [28] of the analyzed sound, which decomposes the entire time-domain process into an infinite number of equal-length subprocesses, each of which is approximately stationary. By performing a Fourier transform on these subprocesses, one can determine when specific frequencies occur, i.e., the sliding window Fourier transform or the output from a set of equally spaced bandpass filters. The STFT analysis spectrum contains frequency and phase information. Due to the non-negative constraint, NMF can only analyze the STFT amplitude spectrum and cannot

analyze the STFT phase information.

If the frequency of a signal does not change over time, it is called a stationary signal. However, most signals studied in real life are non-stationary signals, so the short-time Fourier transform is particularly important. In fact, most speech signals in audio signal processing are discrete. To meet the requirements of computer discreteness and stationarity, it is necessary to apply windowing and frame division to the signal. Windowing involves truncating the time-domain signal, and the processed data then exhibits short-time stationarity. The length of each speech frame after truncation typically ranges from 11 to 30 milliseconds. When performing STFT on speech signals, adding a window function can reduce spectral leakage.

In practical applications, the principle for setting the window function is typically as follows: achieve high frequency resolution in spectral analysis, narrow the main lobe of the window function's spectrum as much as possible, concentrate energy within the main lobe, minimize side lobe gain, and ensure rapid attenuation of side lobe gain with frequency to reduce spectral leakage distortion during analysis. When it is difficult to find a window function with a very narrow main lobe width and rapid side lobe attenuation, the rectangular window is the opposite: it has the narrowest main lobe width but very wide side lobes. Therefore, comprehensive consideration is required when analyzing and processing corresponding data. Commonly used window functions are as follows:

(1) Rectangular window:

$$w(n)=\begin{cases}1, & 0\le n\le N-1\\ 0, & Other\end{cases} \tag{1}$$

The frequency response is:

$$W_R(e^{j\omega})=\frac{\sin(\omega N/2)}{\sin(\omega/2)}e^{-j\omega\left(\frac{N-1}{2}\right)} \tag{2}$$

(2) Han Ning Window:

$$w_{Hn}(n)=\begin{cases}0.54-0.46\cos\left(\dfrac{2\pi n}{N-1}\right), & 0\le n\le N-1\\ 0, & Other\end{cases} \tag{3}$$

The frequency response is:

$$\begin{aligned}W_{Hn}(\omega)&=0.54W_R(\omega)\\&+0.23\left[W_R\left(\omega-\frac{2\pi}{N-1}\right)+W_R\left(\omega+\frac{2\pi}{N-1}\right)\right]\end{aligned} \tag{4}$$

(3) Hanming Window:

$$w_{Hm}(n)=\begin{cases}0.5\left[1-\cos\left(\dfrac{2\pi n}{N-1}\right)\right], & 0\le n\le N-1\\ 0, & Other\end{cases} \tag{5}$$

The frequency response is:

$$\begin{aligned}&W_{Hn}(\omega)=\\&\left\{0.5W_R(\omega)+0.25\left[W_R\left(\omega-\frac{2\pi}{N-1}\right)+W_R\left(\omega+\frac{2\pi}{N-1}\right)\right]\right\}e^{-j\omega\left(\frac{N-1}{2}\right)}\end{aligned} \tag{6}$$

The definition of the short-time Fourier transform is shown in Equation (7), where $x(t)$ is the time signal, $w(t)$ is the window function, $X(w,t)$ is the spectrum at time t, and * denotes linear convolution. That is:

$$X(\omega,t)=\int_{-\infty}^{\infty}x(\tau)*w(\tau-t)*e^{-j\omega t}d\tau \tag{7}$$

Taking the absolute value of equation (7) gives the amplitude spectrum of the desired signal.

II. B. 2) Homomorphic Signal Processing of Inverted Spectra

Another method for time-frequency analysis is the cepstrum transform. Cepstrum involves taking the logarithm of the values of a signal in its frequency spectrum. The cepstrum transform represents the inverse Fourier transform of the logarithmic amplitude spectrum of a speech signal. This analysis method facilitates the extraction and analysis of periodic signals that are difficult to identify in the original frequency spectrum.

Common methods for separating composite signals include linear filtering and nonlinear filtering. Linear filtering techniques are used to separate additive combined signals that influence each other; nonlinear filtering techniques can separate two signals synthesized by multiplication or convolution. Separating the signals involved in convolution from the convolution result is called deconvolution, and deconvolution algorithms are divided into two major categories: parametric deconvolution and nonparametric deconvolution. Homomorphic signal processing is the most important type of nonparametric deconvolution algorithm. Homomorphic signal processing transforms the product or convolution relationship between signals into a summation relationship, thereby enabling the extraction of musical signals such as percussion, string instruments, and vocals from mixed signals. Applying homomorphic signal processing to speech signals yields their spectrogram parameters, which contain more information and yield better results than other parameters. Since speech signal analysis is performed on a frame-by-frame basis, the resulting parameters are short-term spectrogram parameters.

II. C.Audio Feature Analysis

II. C. 1) Time domain characteristics of audio signals

1. Time-domain characteristics of speech signals

To illustrate the differences in time-domain characteristics between pure speech signals and mixed speech signals (with background music), the audio waveform diagram of the recitation of the “Tengwang Pavilion Sequence” is shown in Figure 1, and the audio waveform diagram of the mixed speech recitation of the “Tengwang Pavilion Sequence” (with background music) is shown in Figure 2.

As can be seen from Figures 1 and 2, although the time-domain waveform envelopes of the pure recitation of the “Tengwang Pavilion Sequence” and the mixed recitation of the “Tengwang Pavilion Sequence” with background music are roughly the same, there are still differences in amplitude. The amplitude values of the audio recording of the “Tengwang Pavilion Sequence” recitation with background music are greater than those of the pure audio recording of the “Tengwang Pavilion Sequence” recitation. This can also be understood as the energy of the final output audio being the result of the superposition of the two or more audio signals. Furthermore, it can be observed that at the point where the waveform amplitude is approximately 50 in Figure 1, there is no voice output at a lower level. However, in the audio recording of the “Tengwang Pavilion Sequence” with background music, due to the presence of the background music's audio amplitude, there is still a waveform present at the point corresponding to where the waveform amplitude is approximately 0 in Figure 1 in Figure 2. This also demonstrates that the energy of the final output audio is the result of superposition. Furthermore, there are multiple points in Figure 1 where the waveform amplitude is approximately 50, and these points are very sparse, indicating that the speech signal has sparsity.

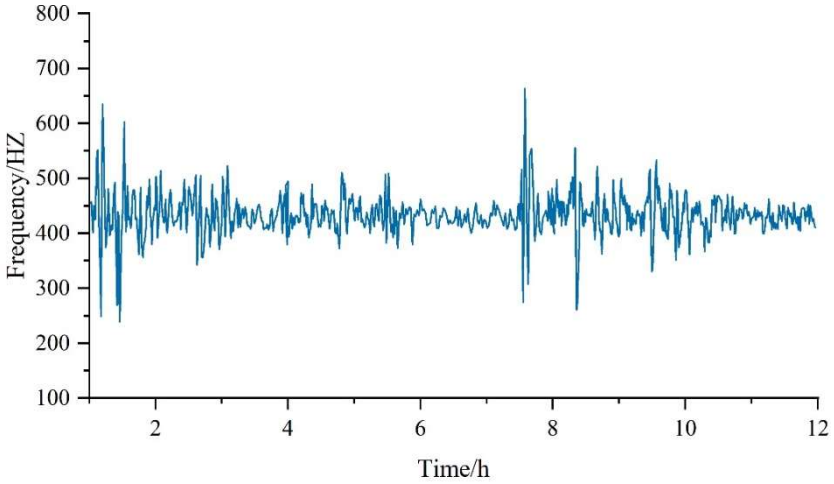


Figure 1: Waveform of the recitation audio of Preface to the Tower of King Teng

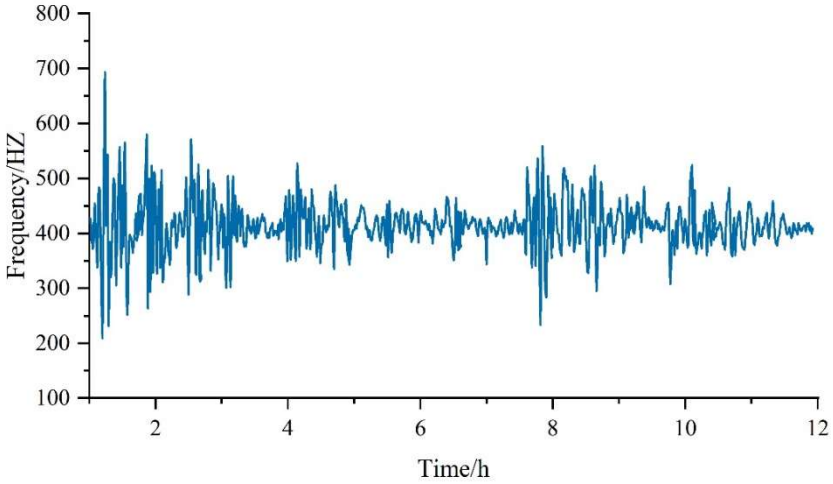


Figure 2: Teng wang Pavilion Preface recitation (with background music) audio waveform diagram

2. Time domain characteristics of musical signals

To illustrate the different time domain characteristics of audio signals produced by different instruments, Figure 3 shows the audio waveform of the world-famous piece “Canon” played on a violin. Figure 4 shows the audio waveform of “Canon” played on a piano and violin together. The onset phase of the violin is quite distinct, characterized by a significant increase in energy at the beginning of a note. As shown in Figure 4, the energy of the audio waveform from the piano and violin playing together is the result of the energy from both instruments being superimposed. Additionally, when comparing audio waveform diagrams of the same piece of music at multiple different time intervals, it can be observed that there are repeated waveform signals. This indicates that the same audio appears at different times, which can also be observed in the sheet music of the piece, as musical pieces are divided into measures, and there are often repeated measures. This demonstrates that musical signals have the characteristic of multiple repetitions.

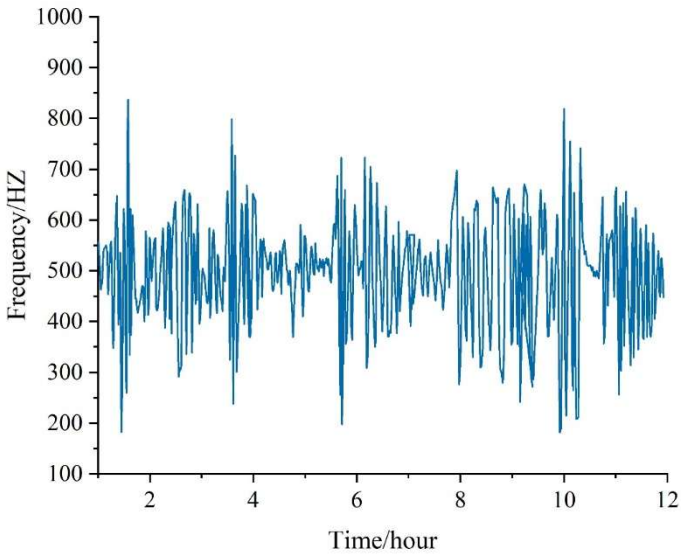


Figure 3: Audio waveform of the violin playing the Canon

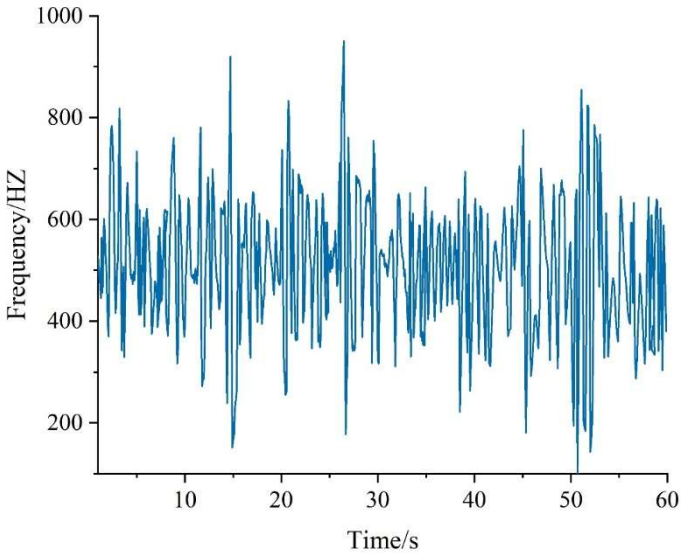


Figure 4: Audio waveform of the piano and violin mixed performance of Canon

II. C. 2) Frequency domain characteristics of audio signals

In addition to time-domain characteristics, speech signals also exhibit certain frequency-domain characteristics. From a subjective perspective, the human ear perceives frequencies within the range of 20–20 kHz. Sounds within this frequency band evoke a particular sensory experience. For human hearing, the frequency range between 150–250 Hz constitutes the low-frequency sensitive region for human voice; The 4-6 kHz range is the mid-high frequency range and the most sensitive part of the human ear; the 10 kHz range is the high-frequency sensitive part of human voice. In general, the higher the signal frequency, the clearer the sound; the lower the frequency, the deeper the sound. To illustrate how humans perceive sounds of different frequencies, Table 1 shows the impact of frequency bands on sound. As shown in the table, different instruments produce music with varying frequency ranges and differing proportions of overtone components, resulting in each instrument having its own unique audio characteristics. Different frequency ranges exert distinct effects on an instrument's timbre. These frequency differences enable the identification of certain frequency-domain feature parameters by analyzing the frequency composition of each instrument in a mixed audio signal. This allows for the separation of individual instrument audio sources within the mixed audio.

Table 1: The effect of frequency bands on sound

Frequency band(HZ)	Too low	Plentiful	Overtop
16-20kHZ	The loss of flavor and color lacks the expressiveness of timbre	The human skull conducts sound to feel the flavor of sound, and the color is rich in timbre expression	Floating and unstable feelings
12-16kHZ	Lose your luster, lose your personality	The gold glitters	It produces burrs and is grating
10-12kHZ	Boring and losing its shine	The metallic sound was intense	High noise, increase the background noise
8-10kHZ	Flat	The s sound is obvious and transparent	Sharp-pointed
4-8kHZ	Dim	Transparency affects the timbre	A dental sound is produced
2-4kHZ	Vague	Strong penetration, bright and clear	Cough volume
1-2kHZ	Loose, causing the timbre to be disjointed	Strong transparency	The jump makes the sound hard
800kHZ	Relaxation	Strong and powerful	The voice is heavy
300HZ-500HZ	Hollow, thin, cloudy	The voice is powerful, bright and clear	Voice tone of the phone
150HZ-300HZ	Soft	The sound is strong, thick and solid	Brusgue
100HZ-150HZ	Thin	Fullness increased	The muddy oil shows a "hmm" sound
60HZ-100HZ	The tone is weak	The timbre is thick and mixed	Low frequency resonance sound, with a rumbling feeling

20HZ-60HZ	Emptiness	Good sense of space, the fundamental frequency of the musical sound	Low-frequency resonance produces a buzzing sound

To illustrate the differences in the frequency domain characteristics between pure speech signals and mixed speech signals, the frequency spectra of the recitation of the “Tengwang Pavilion Sequence” and the mixed speech recitation of the “Tengwang Pavilion Sequence” (with background music) were plotted separately. The spectrum exhibits a segmented band-like distribution, indicating that the energy of each frequency component rapidly decreases from a high level to a low level within a short time, and then increases again. This is because the reciter of the “Tengwang Pavilion Sequence” employs a rhythmic and expressive delivery style, resulting in a segmented spectrum for the speech signal. This also demonstrates that audio characteristics can be represented in the frequency domain.

II. D.Speech separation

II. D. 1) Classification of speech separation tasks

Speech separation refers to the technology of extracting and restoring the speech signals of individual speakers from overlapping speech signals of multiple speakers, also known as multi-speaker separation. Speech separation without prior information about the speakers is classified as blind source separation [29].

Based on the degree of dependence on prior information about the sound source signals, speech separation tasks can be divided into three types: speaker-dependent, target-dependent, and speaker-independent. In speaker-dependent speech separation tasks, all speakers present in the mixed speech signals in the test set must also be present in the training set. This means the speech separation model can only separate mixed speech from specific speakers, and the output order of the separated signals is fixed, with each output stream corresponding to a single speaker. To address speech separation for other speaker combinations, a new separation model must be trained using corresponding data, so it lacks generalization capability beyond the training set. In target-related speech separation tasks, the target speaker to be separated remains consistent across the training and test sets, while there are no constraints on other interfering speakers. Target-related speech separation is similar to speech denoising, as both models have only one output stream—the target speaker's speech signal—while the speech of other interfering speakers is treated as noise signals.

II. D. 2) Speech Separation Signal Model

Speech signals are time-varying, non-stationary, and discrete. However, the characteristics of speech signals remain essentially unchanged over small time scales, a phenomenon known as the short-term stationarity of speech. Short-term analysis techniques are integral to the entire process of speech signal processing. Typically, time-domain waveform signals captured by microphones require framing and windowing processing, with frame lengths generally ranging from 10 to 30 ms. All speech separation methods discussed in this paper are based on this short-term stationarity assumption.

Single-channel speech separation typically uses a linear instantaneous mixing model, which is expressed mathematically as shown in formula (8). The mixed signal $y=\{y_0,\ldots,y_{L-1}\}$ collected by the microphone is a linear mixture of the speech signals $s_i=\{s_{i,0},\ldots,s_{i,L-1}\}\left(i\in\{0,\ldots,C-1\}\right)$ of C speakers and background noise $n=\{n_0,\ldots,n_{L-1}\}$ in the time domain, where L represents the number of signal sampling points:

$$y=\sum_i s_i+n \quad i\in 0,\ldots,C-1 \tag{8}$$

Among these, y, s_i, and n all belong to time-domain signals. The problem can be defined as the task of extracting and reconstructing the waveform signals of all sound sources given the time-series waveform signal of a mixed speech signal. The time-domain waveform of a speech signal contains all information, including the content of the speech and the characteristics of the speaker. Directly processing the waveform sampling points is quite challenging. Time-domain feature analysis methods include pre-emphasis, short-time energy analysis, short-time zero-crossing analysis, and short-time correlation analysis, which are typically used for initial parameter feature extraction and speech preprocessing. Traditional time-domain speech separation methods are generally designed for speech enhancement tasks, where the interfering signal is non-speech noise, and primarily include parameter-based and filtering methods, as well as signal subspace methods.

The most important perceptual characteristics of speech are embedded in the spectrum and power spectrum, so speech time-domain waveforms are typically transformed into the frequency domain for analysis. The Fourier transform, which is suitable for periodic signals or stationary random signals, cannot directly represent non-stationary speech signals. Combining short-time analysis of speech signals with the Fourier transform yields the

commonly used short-time Fourier transform (STFT). After STFT, the mixed signal can also be represented as the linear superposition of corresponding signal components in the time-frequency domain, yielding the time-frequency domain form of the instantaneous mixture model:

$$Y(t,f)=\sum_i S_i(t,f)+N(t,f) \quad i\in 0,\ldots,C-1 \tag{9}$$

In this context, Y, S_i, and $N\in\square^{T\times F}$, where T and F represent the number of frames in the time dimension and the number of frequency points in the frequency dimension, respectively. The problem can be defined as extracting the STFT time-frequency spectrum of a given mixed speech signal and reconstructing the STFT time-frequency spectrum of all sound sources. $|Y|$ is called the amplitude spectrum, commonly referred to as the spectrogram. $|Y|^2$ is called the power spectrum, and $\angle Y$ is called the phase spectrum. The spectrogram combines the advantages of the frequency spectrum and the time-domain waveform, reflecting the dynamic changes in the spectral characteristics of speech over time. Therefore, the STFT is a joint time-domain and frequency-domain analysis method, and once the window function is determined, the time-frequency resolution is also determined.

III. Audio separation model based on edge AI

III. A. Attention Mechanism

III. A. 1) Channel Attention

The process of adding channel attention to a network typically involves three steps. First, each feature map is compressed using global pooling, which compresses each feature map on each channel into a real number, as shown in the following formula:

$$Z_c=\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}u_c(i,j) \tag{10}$$

where H and W are the height and width of the feature map, respectively, and Z is the feature vector containing the global attention information corresponding to the feature map.

Then, the squeezed vector containing the global attention information of different feature layers is passed through a fully connected layer and an activation layer, as shown in the following formula:

$$s_c=\sigma\left(W_2\delta\left(W_1Z_c\right)\right) \quad (4-2) \tag{11}$$

σ denotes the ReLU activation function, δ denotes the sigmoid activation function, and w_1 and w_2 denote two different fully connected layers. The vector s_c obtained from equation (11) can represent the importance of different channels. Finally, s_c is applied to the features u extracted by convolution in a weighted manner, as shown in the following formula:

$$x_c=s_c*u_c \tag{12}$$

x_c represents the weighted features on the channel. Compared with the features u_c obtained after direct convolution, it can enhance the importance of useful channel features while reducing the importance of useless channel features. The importance of different feature layers is achieved through an adaptive method.

III. A. 2) Spatial Attention

The spatial attention mechanism is implemented as shown in Figure 5. First, the feature map is passed through a max pooling operation and a mean pooling operation. The results of the pooling operations are then concatenated along the channel dimension to obtain a feature map with 2 channels. A convolution operation is then applied to reduce the number of channels to 1. The result is passed through a sigmoid activation function to obtain the spatial attention weight feature. Finally, the feature map is multiplied by the obtained spatial attention weights to obtain the spatial attention feature map. The implementation formula is as follows:

$$\begin{cases} u_{avg}=AvgPool(u) \\ u_{\max}=MaxPool(u) \\ u'=concat([u_{avg},u_{\max}]) \end{cases} \tag{13}$$

where u represents the input feature map, $AvgPool$ and $MaxPool$ represent the average pooling and max

pooling operations, respectively, and u' represents the concatenated features:

$$u_c = \sigma\left(f\left(u'\right)\right) \tag{14}$$

f denotes the convolution operation, which primarily serves to reduce the dimensionality, while σ represents the activation function.

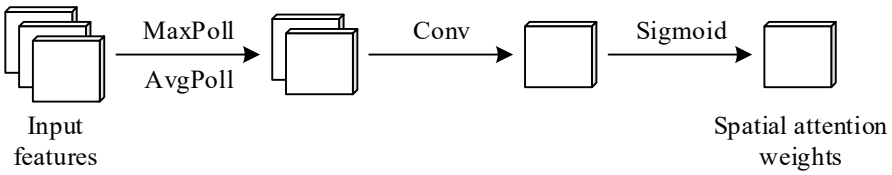


Figure 5: Spatial attention mechanism

III. A. 3) Time Attention

First, time attention was introduced into the RNN model, known as LSTM, which improved the RNN cell unit by adding a forget gate, a memory gate, and an output gate. These gates control the machine's actions, selectively remembering and forgetting long sequences, enabling it to process long sequence data while avoiding the problem of gradient disappearance during model training.

Another approach involves applying weighting to the time series. The neural network used in translation tasks shares a similar overall structure with audio signal separation networks, typically adopting an Encoder-Decoder architecture. The key challenge in translation tasks is how to enable the encoder to effectively aggregate information from long sequence inputs. By employing temporal attention mechanisms, the model can focus on relevant information in long sequences while ignoring less important details, thereby enhancing its performance. Specifically, this is achieved by applying a weight to the encoder's output state feature h, as shown in the following formula:

$$c_i = \sum_{j=1}^{l} \alpha_{ij} h_j \tag{15}$$

α_{ij} denotes the weight matrix corresponding to the encoder output state feature h, c_i denotes the feature that aggregates all the information of the encoder output state through attention, and l denotes the length of the input sequence.

α_{ij} attention weights are determined based on the previous state of the decoder output, as shown in the following formula:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{l} exp(e_{ik})} \tag{16}$$

$$e_{ij} = f\left(s_{i-1}, h_j\right) \tag{17}$$

where s_{i-1} represents the previous state output by the decoder, and f represents a function that calculates the correlation between s_{i-1} and h_j.

The final hidden state output by the decoder is given by the following formula:

$$s_i = g\left(s_{i-1}, y_{i-1}, c_i\right) \tag{18}$$

It can be seen that the output of the next state of the final decoder is determined jointly by its previous output state and c_i, which contains all the information of the encoder.

III. A. 4) Self-attention

The fully connected self-attention calculation process can be implemented in the following steps. First, the input sequence is passed through an embedding layer to obtain the corresponding feature vector sequence. Then, three matrices w_Q, w_K, and w_V are multiplied by the feature vectors. This yields three vectors: the query vector q, the key vector k, and the value vector v. The formulas are as follows:

$$\begin{cases} q_i = W_Q x_i \\ k_i = W_K x_i \\ v_i = W_V x_i \end{cases} \tag{19}$$

Then, calculate the score corresponding to each feature vector by performing an inner product operation between the query vector q at the corresponding position and the key-value feature vector v at other positions. This score reflects the degree of relevance between the input at that position and other positions, as well as the degree of attention paid to other positions. After that, all scores are normalized and subjected to a softmax operation to ensure that the sum of the weights is one. When using multi-head attention, the normalization operation also ensures that the scores obtained from different attention heads are on the same scale. The formula for this process is as follows:

$$score = soft\max\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) \tag{20}$$

Here, Q and K represent the query matrix and key matrix, respectively, and d_k represents the dimension of the key vector. Finally, the obtained scores are weighted and summed with the corresponding value matrix V. The sum of the results is the self-attention result corresponding to the current input.

$$z_i = \sum_{j=1}^{l} score_j v_j \tag{21}$$

III. B. Model Design Based on Attention Mechanisms

III. B. 1) Channel Attention Module

Additionally, channel attention can be added to the separation module, as shown in Figure 6. The SK (Selective Kernel Networks) [30] channel attention module is added to the output layer of each submodule in the separation module, primarily considering that features obtained from different convolution kernel sizes should have different weights. The entire module consists of three parts. First is feature separation, which primarily obtains two features U_1 and U_2 from the input features through two convolution kernels of different sizes.

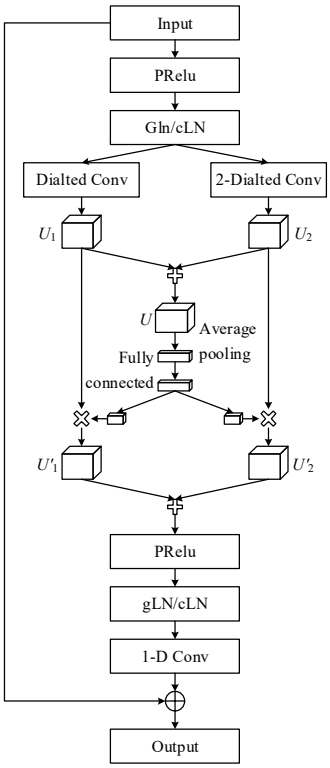


Figure 6: Channel attention module in the separation module

III. B. 2) Spatial Attention Module

CBAM is a simple and effective lightweight attention module that incorporates spatial attention, as shown in Figure 7. This module mainly consists of a channel attention module and a spatial attention module, each of which maps the attention mechanism from the channel and spatial dimensions, respectively. The advantage of this approach is that the output features are more concentrated on the channels that contribute more to the model through channel attention in the dilated convolution layer.

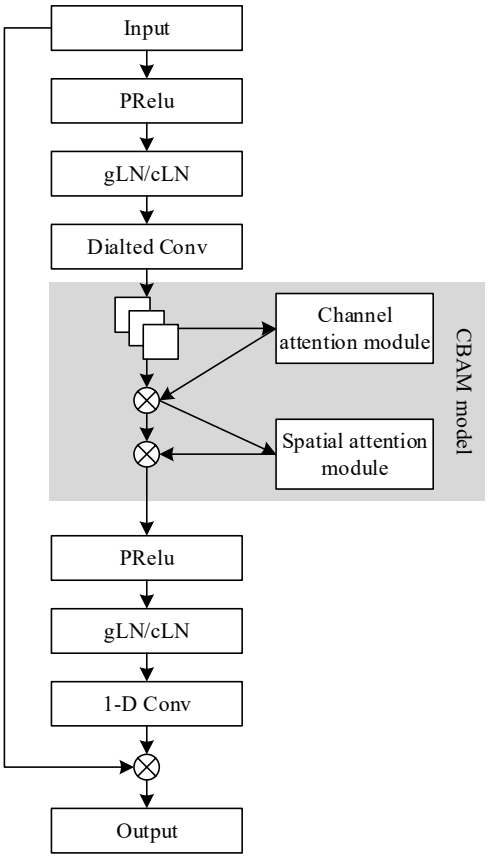


Figure 7: The CBAM in the separation module

III. B. 3) Time Attention Module

For the DPRNN model with a recurrent network structure, a windowing and framing approach similar to that used in the STFT transform is employed. The one-dimensional audio signal is sliced and then reassembled to obtain two new dimensions: one representing the temporal signal in each frame, and the other representing the relationship between frames. This approach decomposes an audio signal of length L into two sequences of approximately $\sqrt{L}$ each in the two dimensions. This avoids the issues of gradient vanishing and information loss during training caused by overly long sequences while ensuring that the RNN model retains the ability to process the input sequence with an approximate global receptive field.

III. B. 4) Self-attention module

In the field of natural language processing (NLP), models based on self-attention have garnered widespread attention due to their outstanding performance in various NLP tasks, such as translation, question answering, entity recognition, and text classification. Their core idea is to perform pre-training through self-supervised learning, enabling larger and deeper network models to learn more profound feature representations of individual characters or words. These pre-trained models are then applied as feature extraction components in downstream tasks, often requiring only minor adjustments to the overall model to achieve good results. The attention module in these models is illustrated in Figure 8.

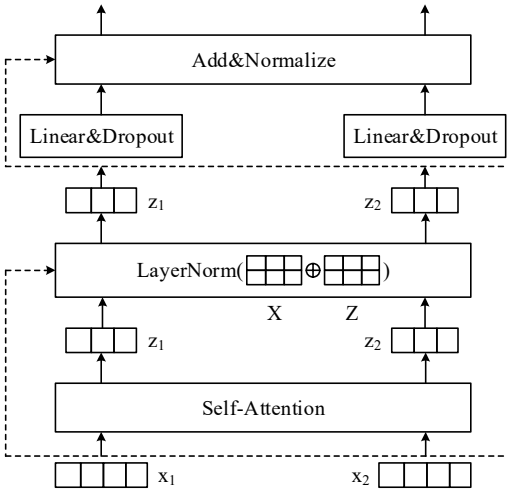


Figure 8: To the attention module

III. C. Dual-door control mechanism module design

The use of different attention mechanisms is equivalent to weighting audio features in a separated network in different ways. When the attention mechanism acts on each feature point, it functions as a gating mechanism to determine which feature information should be retained and which should be discarded. Non-linear gating mechanisms have been proven in previous sequence models to control information flow and may assist the network in modeling more complex interactive information. Therefore, in the original TCN network structure, two gates were added to each one-dimensional convolution module. One corresponds to the first 1×1 convolution layer in the one-dimensional convolution module, generating a control gate for the input by adding a convolution layer. The other is the output gate for the TCN convolution module, generating the gate weights through self-attention. The network structure is shown in Figure 9. Through the dual gate mechanism, the flow of feature information across different convolutional layers is better controlled, filtering out unnecessary feature information to achieve better separation effects.

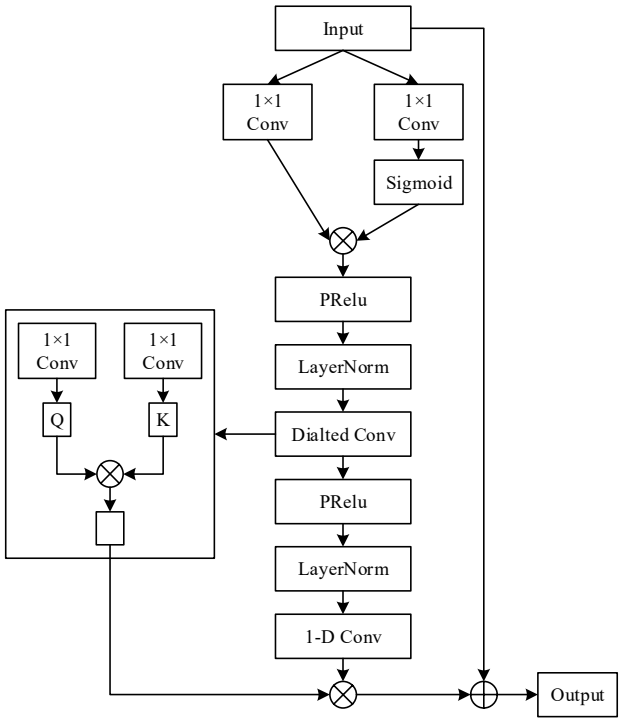


Figure 9: Use the TCN convolution layer with a double door mechanism

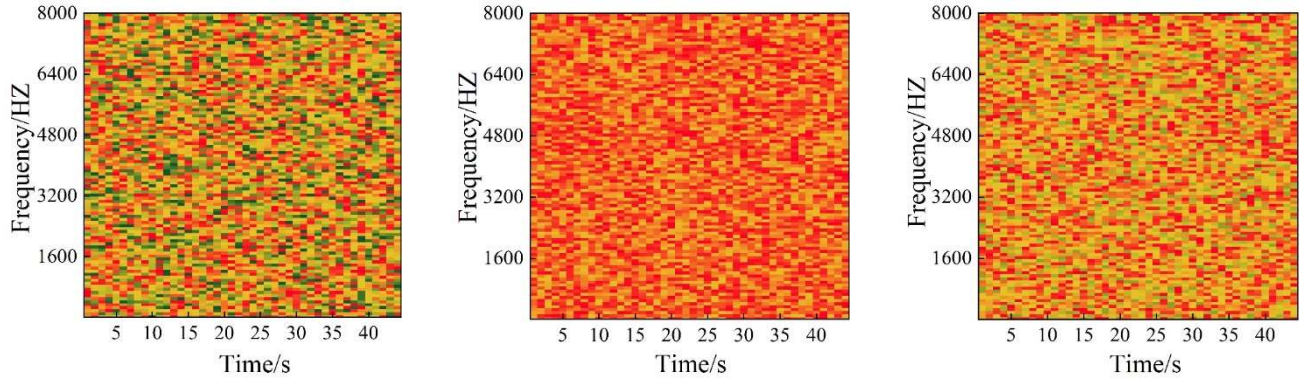
IV. Experimental Results and Analysis

IV. A. Experimental Data and Evaluation Indicators

The separation method uses the MIR-1K dataset, which contains 1,000 music clips sung by amateur singers. The music clips have a sampling frequency of 16 kHz and durations ranging from 4 to 13 seconds. The vocals and background music are recorded separately in the left and right channels. The evaluation metrics use the BSS_EVAL toolbox to calculate the SIR, SDR, and SAR values after separation.

IV. B. Experimental Results

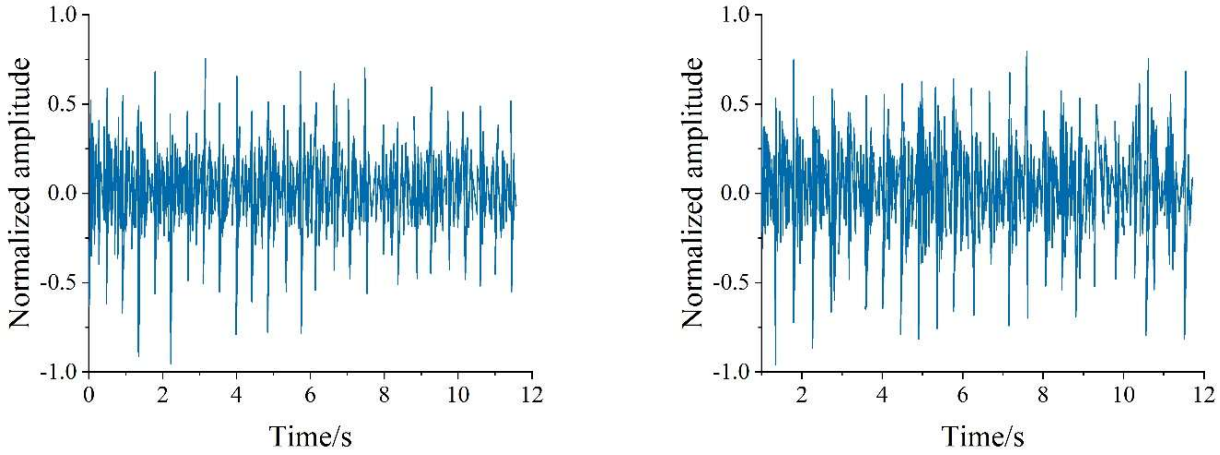
A random music segment was selected from the MIR-1K database and separated using the method described in this paper. Taking amy_11_01.wav as an example, the spectrograms before and after separation are shown in Figure 10. As can be seen from Figure 10(a), the original mixed music spectrogram contains horizontal ridges representing background music and vertical ridges representing vocals. After separation, as shown in Figure 10(b), only the horizontal ridge representing background music is clearly visible, while the vertical ridge is not prominent, indicating that only the background music has been separated. In Figure 10(c), the vertical ridge is more prominent, indicating that the vocal information has been retained after separation.



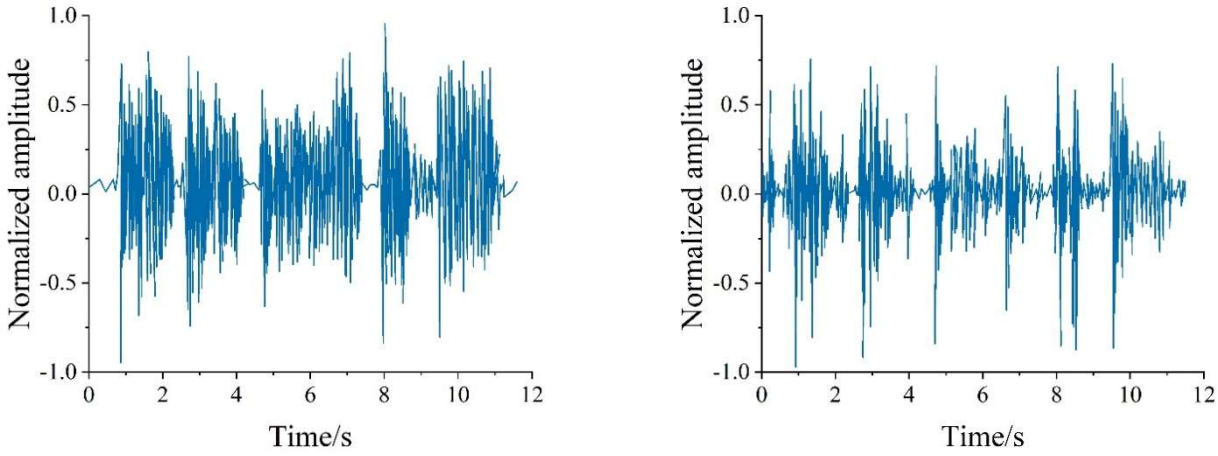
(a) Original mixed song spectrum (b) Separate the background music score diagram (c) Separate song spectrum diagram

Figure 10: Amy 1101.wav Comparison of spectrograms before and after separation

Figure 11 shows a comparison of the waveforms of each component before and after separation. As can be seen from the figure, the waveforms of both the background music and vocals are basically consistent with the original waveforms after separation. The spectrogram and waveform comparison diagram show that the separation algorithm described in this paper can effectively separate the background music and vocals in a song.



(a)Original background music waveform (b)The waveform of the background music

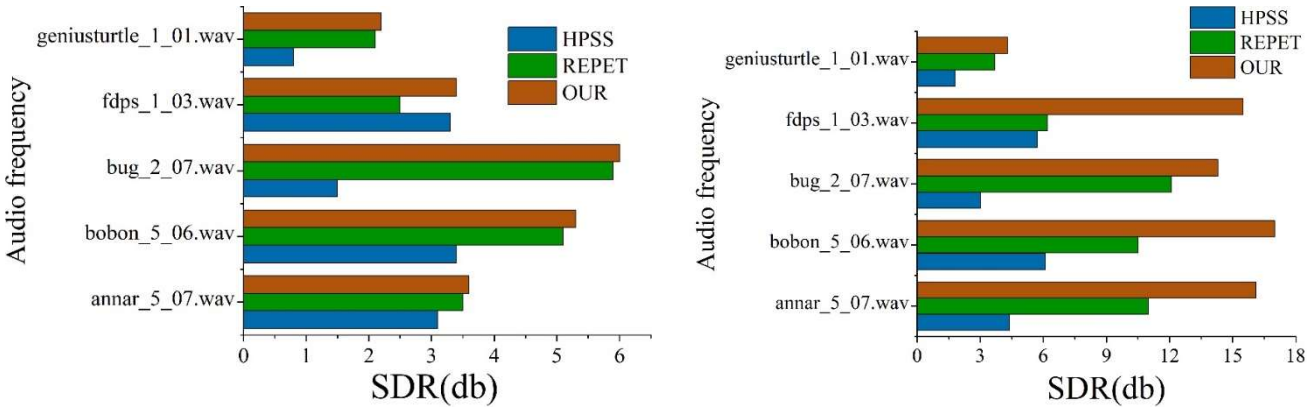


(c) Original voice waveform (d) The waveform of the separated song

Figure 11: Amy 11 01.wav Comparison chart of waveform before and after separation

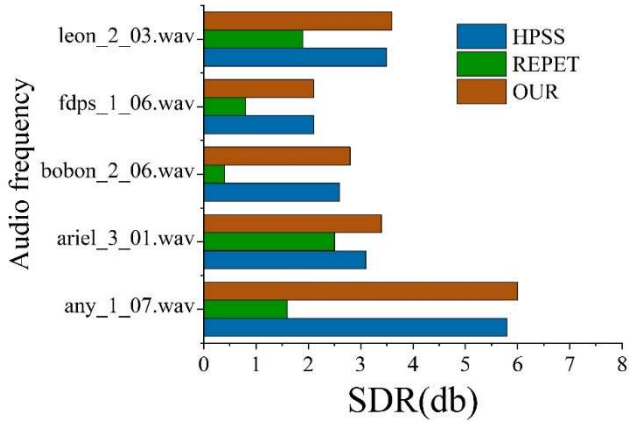
After verifying that the separation method in this paper can effectively separate songs, the advantages of the proposed method over the REPET algorithm and HPSS separation algorithm will be explained. For the 1000 music clips of MIR-1K, 10 pieces were randomly selected for separation and their SDR and SIR values were calculated (the extracted music clips were: "annar_5_07.wav", "bobon_5_06.wav", "bug_2_07.wav", "fdps_1_03.wav", "geniusturtle_1_01.wav", "any_1_07.wav", "ariel_3_ 01.wav", "bobon_2_06.wav", "fdps_1_06.wav", "leon_2_03.wav", the results are shown in Figure 12, and the subjective auditory perception test is performed on the results after separation.

As shown in the figure, regardless of whether it is background music or vocals, the algorithm proposed in this paper outperforms the HPSS and REPET algorithms in terms of separation metrics such as SDR and SIR. As shown in Figure 12(b), when separating background music, the algorithm proposed in this paper achieves an improvement of approximately 4 dB to 10 dB in SIR values compared to the HPSS algorithm, and at least a 1 dB improvement in performance compared to the REPET algorithm. Figure 12(d) shows that when separating vocals, the proposed algorithm achieves a certain improvement in separation performance compared to both the REPET and HPSS algorithms.

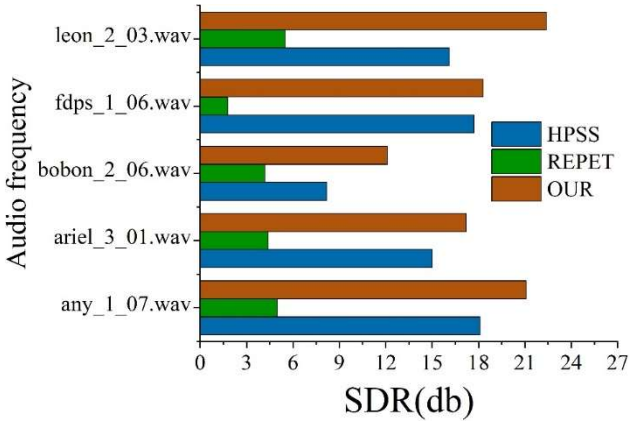


(a)Background music separation index SDR comparison chart

(b)Background music separation index SIR comparison chart



(c)Comparison chart of voice separation index SDR



(d)Singing separation index SIR comparison chart

Figure 12: Random 10 music fragment separation performance index comparison chart

A subjective auditory perception test was conducted with 10 participants, divided into two groups of 5 males and 5 females, to evaluate the separation results using the MOS metric. The results are shown in Table 2. From a subjective auditory perspective, the background music and vocals separated by the algorithm in this paper can be clearly distinguished, with good quality. However, there is still minor noise interference in the separated vocals. The background music separated by the HPSS algorithm has poor quality, while the separated vocals have slightly better quality but still exhibit noticeable system noise interference. The REPET algorithm achieves good separation of background music, but the separated vocals still contain noticeable background music.

Table 2: Random 10 music MOS test results

	Background music			Singing		
Algorithm	Male	Female	Mean value	Male	Female	Mean value
HPSS	2.1	2.3	2.2	3.4	2.9	3.2
REPET	3.5	3.8	3.7	2.5	2.2	2.4
OUR	4.2	3.7	4.0	3.9	3.8	3.9

Processing 1,000 music clips from the MIR-1K database, we calculated the average values of SDR, SIR, and SAR. We compared the proposed method with the HPSS algorithm, REPET, and its improved algorithms, with the results shown in Tables 3 and 4. As shown in Table 3, when separating background music, the proposed algorithm achieves an improvement of approximately 3–6 dB in SIR compared to the HPSS and REPET algorithms and their improved versions, indicating that the proposed algorithm outperforms the HPSS and REPET algorithms and their improved versions in terms of separation quality when separating background music. As shown in Table 4, when separating vocals from music, the proposed algorithm achieves an improvement of approximately 2–8 dB in SIR compared to the HPSS and REPET algorithms and their improved versions. Combining the two tables, it can be seen that while the SAR metric remains unchanged, the SDR improves by approximately 2 dB.

Table 3: Comparison of background music separation performance (average results)

Algorithm	Background music		
	SDR(dB)	SIR(dB)	SAR(dB)
HPSS	1.927	4.157	8.211
REPET	2.151	8.307	4.622
REPET-SIM	1.542	4.619	6.563
REPET-MFCC	2.293	7.381	5.255
OUR	4.211	11.762	8.359

Table 4: Comparison of background music separation performance (average results)

Algorithm	Background music		
	SDR(dB)	SIR(dB)	SAR(dB)
HPSS	-0.683	11.853	0.502
REPET	2.5582	6.481	6.943
REPET-SIM	1.247	6.073	5.002
REPET-MFCC	2.874	6.382	7.574
OUR	4.236	14.228	6.511

V. Conclusion

After preprocessing the audio signals in musical performances, this study proposes a source separation model based on deep neural networks to achieve audio source separation in live musical performances. Experimental simulations demonstrate that, for the 1,000 music segments in the MIR-1K database, the proposed attention-based music separation method effectively improves the performance of vocal separation and background music separation compared to the existing REPET algorithm and its improved versions, as well as the HPSS algorithm, particularly in segments with distinct rhythmic patterns. Specifically, compared to existing single-source separation algorithms, the proposed method can significantly improve SIR and SDR while maintaining SAR at a similar level, indicating that the method can effectively separate music while maintaining robustness.

In future research, it may be appropriate to incorporate some musical attributes, such as fundamental frequency, rhythm, and meter. Additionally, further research could be conducted in data-driven approaches to explore deep learning-based vocal separation algorithms. However, to apply it to practical applications, further research is needed, and the following issues deserve attention:

1. Using deep neural networks to optimize separation

In recent years, deep learning methods based on deep learning theory have received increasing attention. In the field of music, deep learning methods are used to separate vocals from accompaniment. With sufficient data and an appropriate network structure, the corresponding neural network can be trained. Through discriminative training, the model can achieve strong fault tolerance, thereby improving separation performance. However, this method requires a large number of samples and significant hardware support. Therefore, how to effectively simplify experiments, reduce experimental time, and minimize hardware resource requirements has become a key issue to address in the next phase.

2. Incorporating musical characteristics to assist analysis

Music contains various features, such as fundamental frequency and pitch. These properties can be used for matrix decomposition, particularly by incorporating them into the cost function of matrix decomposition or as an auxiliary factor within the cost function. Additionally, by combining spectrogram analysis with matrix decomposition algorithms, they can be performed in parallel or sequentially to maximize the effectiveness of vocal separation. However, how to integrate these methods remains a technical challenge requiring further research.

References

[1] Swarbrick, D., Bosnyak, D., Livingstone, S. R., Bansal, J., Marsh-Rollo, S., Woolhouse, M. H., & Trainor, L. J. (2019). How live music moves us: head movement differences in audiences to live versus recorded music. Frontiers in psychology, 9, 2682.

[2] Holland, S., Mudd, T., Wilkie-McKenna, K., McPherson, A., & Wanderley, M. M. (2019). Understanding music interaction, and why it matters. New directions in music and human-computer interaction, 1-20.

[3] Adavanne, S., Politis, A., Nikunen, J., & Virtanen, T. (2018). Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. IEEE Journal of Selected Topics in Signal Processing, 13(1), 34-48.

[4] Wu, Y., Zhu, L., Yan, Y., & Yang, Y. (2019). Dual attention matching for audio-visual event localization. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 6292-6300).

[5] Cano, E., FitzGerald, D., Liutkus, A., Plumbley, M. D., & Stöter, F. R. (2018). Musical source separation: An introduction. IEEE Signal Processing Magazine, 36(1), 31-40.

[6] Mesaros, A., Heittola, T., Virtanen, T., & Plumbley, M. D. (2021). Sound event detection: A tutorial. IEEE Signal Processing Magazine, 38(5), 67-83.

[7] Cakır, E., Parascandolo, G., Heittola, T., Huttunen, H., & Virtanen, T. (2017). Convolutional recurrent neural networks for polyphonic sound event detection. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(6), 1291-1303.

[8] Van der Hoeven, A., & Hitters, E. (2019). The social and cultural values of live music: Sustaining urban live music ecologies. Cities, 90, 263-271.

[9] Gannot, S., Vincent, E., Markovich-Golan, S., & Ozerov, A. (2017). A consolidated perspective on multimicrophone speech enhancement and source separation. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(4), 692-730.

[10] Stöter, F. R., Uhlich, S., Liutkus, A., & Mitsufuji, Y. (2019). Open-unmix-a reference implementation for music source separation. Journal of Open Source Software, 4(41), 1667.

[11] Hennequin, R., Khlif, A., Voituret, F., & Moussallam, M. (2020). Spleeter: a fast and efficient music source separation tool with pre-trained models. Journal of Open Source Software, 5(50), 2154.
[12] Rafii, Z., Liutkus, A., Stöter, F. R., Mimilakis, S. I., FitzGerald, D., & Pardo, B. (2018). An overview of lead and accompaniment separation in music. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 26(8), 1307-1335.
[13] Rosner, A., & Kostek, B. (2018). Automatic music genre classification based on musical instrument track separation. Journal of Intelligent Information Systems, 50, 363-384.
[14] Gómez, J. S., Abeßer, J., & Cano, E. (2018, September). Jazz Solo Instrument Classification with Convolutional Neural Networks, Source Separation, and Transfer Learning. In ISMIR (pp. 577-584).
[15] Ward, D., Mason, R. D., Kim, C., Stöter, F. R., Liutkus, A., & Plumbley, M. D. (2018, September). SiSEC 2018: State of the art in musical audio source separation-subjective selection of the best algorithm. In WIMP: Workshop on Intelligent Music Production.
[16] Chatterjee, M., Ahuja, N., & Cherian, A. (2022). Learning audio-visual dynamics using scene graphs for audio source separation. Advances in Neural Information Processing Systems, 35, 16975-16988.
[17] Magron, P., Badeau, R., & David, B. (2018). Model-based STFT phase recovery for audio source separation. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 26(6), 1095-1105.
[18] Takahashi, N., & Mitsufuji, Y. (2017, October). Multi-scale multi-band densenets for audio source separation. In 2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (pp. 21-25). IEEE.
[19] Févotte, C., Vincent, E., & Ozerov, A. (2018). Single-channel audio source separation with NMF: divergences, constraints and algorithms. Audio Source Separation, 1-24.
[20] Sawada, H., Ono, N., Kameoka, H., Kitamura, D., & Saruwatari, H. (2019). A review of blind source separation methods: two converging routes to ILRMA originating from ICA and NMF. APSIPA Transactions on Signal and Information Processing, 8, e12.
[21] Michelsanti, D., Tan, Z. H., Zhang, S. X., Xu, Y., Yu, M., Yu, D., & Jensen, J. (2021). An overview of deep-learning-based audio-visual speech enhancement and separation. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 1368-1396.
[22] Kavalerov, I., Wisdom, S., Erdogan, H., Patton, B., Wilson, K., Le Roux, J., & Hershey, J. R. (2019, October). Universal sound separation. In 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (pp. 175-179). IEEE.
[23] Chandna, P., Miron, M., Janer, J., & Gómez, E. (2017). Monoaural audio source separation using deep convolutional neural networks. In Latent Variable Analysis and Signal Separation: 13th International Conference, LVA/ICA 2017, Grenoble, France, February 21-23, 2017, Proceedings 13 (pp. 258-266). Springer International Publishing.
[24] Luo, Y., & Yu, J. (2023). Music source separation with band-split RNN. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 31, 1893-1901.
[25] Yar, G. N. A. H., Maqbool, A., Noor-Ul-Hassan, A. B., & Afzal, Z. (2023, January). Audio Source Separation and Voice Conversion, an Application in Music Industry. In 2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICES&T) (pp. 1-6). IEEE.
[26] Pardo, B., Rafii, Z., & Duan, Z. (2018). Audio source separation in a musical context. Springer Handbook of Systematic Musicology, 285-298.
[27] Slizovskaia, O., Haro, G., & Gómez, E. (2021). Conditioned source separation for musical instrument performances. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 2083-2095.
[28] Xiaohan Guan & Rui Liu. (2025). Research on an efficient acquisition method for beidou signals based on improved short-time fourier transform. Journal of Physics: Conference Series,2996(1),012013-012013.
[29] Swarnadeep Bagchi & Ruairí de Fréin. (2025). Big-Delay Estimation for Speech Separation in Assisted Living Environments. Future Internet,17(4),184-184.
[30] Jia Cheng-kun,Long Min & Liu Yong-chao. (2024). Enhanced face morphing attack detection using error-level analysis and efficient selective kernel network. Computers & Security,137,103640-.