Hindawi
International Journal of Intelligent Systems
Volume 2024, Article ID 6668582, 11 pages
https://doi.org/10.1155/2024/6668582

Research Article

Anime Audio Retrieval Based on Audio Separation and Feature Recognition

De Li, Wenying Xu, and Xun Jin

Department of Computer Science and Technology, Yanbian University, Yanji, China

Correspondence should be addressed to Xun Jin; xunjin@ybu.edu.cn

De Li and Wenying Xu contributed equally to this work.

Received 10 August 2023; Revised 22 March 2024; Accepted 2 May 2024; Published 21 May 2024

Academic Editor: Alexander Hošovský

This paper proposes an anime audio retrieval method based on audio separation and feature recognition techniques, aiming to help users conveniently locate their desired audio segments and enhance the overall user experience. Additionally, by establishing an audio fingerprint database and a corresponding copyright information management system, it becomes possible to track and manage the audio content within anime, effectively preventing piracy and unauthorized use, thereby improving the management and protection of audio resources. Traditional methods for anime audio feature recognition suffer from issues like low efficiency and subjective factors. In contrast, the proposed approach overcomes these limitations by automatically separating and extracting audio fingerprints from different audio sources within anime and creating an anime audio fingerprint database for fast retrieval. The paper utilizes an improved audio separation model based on the efficient channel attention mechanism to separate the anime audio. Subsequently, feature recognition is performed on the separated anime audio, employing a contrastive learning-based audio fingerprint retrieval method for anime audio fingerprinting. Experimental results demonstrate that the proposed algorithm effectively alleviates the issue of poor audio separation performance in anime audio, while also improving retrieval efficiency and accuracy, meeting the demands for anime audio content retrieval.

1. Introduction

With the rapid development of the anime industry, there is an increasing demand for processing and retrieving anime audio content. Anime viewers often develop an interest in the audio content, such as searching for a particular iconic line, a background music track, or a specific sound effect. Despite the existence of many mature methods, there are still limitations when it comes to anime audio. Firstly, anime audio typically consists of multiple sound sources, such as background music, dialogue, and sound effects, which pose challenges in audio processing and retrieval. Current methods often struggle to accurately separate different sound sources. Additionally, anime audio exhibits complex audio characteristics, including highly varied pitch, speech rate, and timbre, which leads to decreased accuracy in feature extraction. Moreover, traditional audio retrieval methods often rely on manually designed features and similarity measurement approaches, lacking the deep learning modeling and representation capabilities for anime audio features.

To address the aforementioned issues, we propose an anime audio retrieval method based on audio separation and audio feature recognition. By separating the sound sources, we can obtain more accurate audio feature representations, which helps improve the accuracy and robustness of audio retrieval, enabling users to search and locate specific audio segments or content more quickly and accurately. The audio separation utilizes a U-Net architecture to effectively separate different sound sources in mixed audio, enhancing the accuracy and efficiency of anime audio processing. To enhance the performance of audio separation, we introduce the Efficient Channel Attention (ECA) mechanism between the downsampling blocks of the encoder, which enhances the focus on important features while effectively handling the complex audio characteristics of anime audio. Additionally, we employ deep learning-based audio feature recognition methods that can extract and represent key features of anime

audio. Through deep learning modeling, we can capture the semantic information of anime audio more accurately, thereby improving the accuracy and efficiency of audio retrieval. The improvements made in this study effectively alleviate the problem of poor performance in audio separation for anime audio. Furthermore, the anime audio retrieval method proposed in this paper is capable of performing fast matching while occupying minimal memory.

The contributions in this research can be summarized as follows:

(1) We propose an anime audio retrieval framework based on the feature of separated anime audio and relieve the difficulty of recognizing audio contents in animations.

(2) The experimental results of the proposed audio separation achieved an average improvement of 0.23 dB in Signal-to-Distortion Ratio (SDR) and 1.08 dB in Signal-to-Interference Ratio (SIR) showing high performance in anime audio separation.

(3) With the contrastive learning on the Mel spectrograms after Short-Time Fourier Transform (STFT), the proposed method can extract representative anime audio features and outperform the other existing methods in recognition.

The rest of the paper is organized as follows. In Section 2, we discuss the related works about audio separation and feature recognition methods. In Section 3, we introduce the framework of the proposed anime audio retrieval and the details of the proposed methods. The experimental results and analysis are given in Section 4. In Section 5, we give a brief conclusion of this work.

2. Related Work

2.1. Audio Separation Methods. During the research on model approaches for audio separation, Grais and Erdogan [1] proposed a single-channel speech-music separation method based on Nonnegative Matrix Factorization (NMF) and spectral masking. This method involves decomposing the mixed audio signal into spectrograms of speech and music. Subsequently, spectral masking techniques are applied to mask the speech and music components at each frequency, facilitating the separation of the signals. To enhance the ability of NMF to extract meaningful audio and achieve higher accuracy, Hayashi et al. [2] introduced a method based on periodic NMF. In frequency-domain methods for audio separation, neural networks are commonly used to estimate Time-Frequency Masks to enhance the precision of signal processing. Luo et al. [3] proposed a method that combines deep clustering with traditional neural networks, demonstrating its effectiveness for music separation. Chandna et al. [4] introduced a low-latency single-channel audio separation framework based on Convolutional Neural Networks (CNNs). This framework utilizes CNN to estimate time-frequency soft masks for source separation, significantly improving the processing speed. Time-domain audio separation network was the first proposed network for audio separation in the time domain [5, 6]. It utilizes an encoder-separation-decoder framework to directly model the audio signal in the time domain. However, this approach may cause distortions, resulting in an imperfect reconstruction of the input signal. To address this issue, Tzinis et al. [7] proposed a two-step separation method for audio separation. This method divides the separation process into two stages: in the first stage, only the encoder and decoder are trained, and in the second stage, the encoder and decoder are fixed while only the separation part is trained. This approach reduces distortions and improves the separation upper limit of the model. Ditter et al. [8] proposed the Multiphase Gammatone Filterbank, which utilizes a deterministic Gammatone Filterbank to enhance the performance of TasNet in handling high-frequency speech signals. Compared to randomly initialized frequency responses, it achieves a better distribution of frequency responses. Nugraha et al. [9] introduced a DNN-based multichannel music separation method, and experimental results demonstrated its performance in separating vocals and accompaniments. Furthermore, improvements were made to the training objectives and overall architecture design. Zeghidour et al. [10] proposed an end-to-end audio separation method called Wavesplit. It calculates a global speaker vector from speaker vectors within short time windows and feeds it into the source separation network to obtain the final source separation results. This approach addresses the fundamental permutation problem in source separation and provides a longer and more robust method for audio separation. Jansson et al. [11, 12] applied the U-Net neural network structure widely used in medical image segmentation to the field of audio separation. They transformed the audio signals into frequency and phase information using the STFT and used the frequency information of the mixed audio as input to the U-Net network. After training the network, they obtained the separated target audio. Stoller et al. [13] proposed a Wave-U-Net, which allows information exchange and fusion from multiple scales and different levels of abstraction. By utilizing one-dimensional (1D) convolution operations, Wave-U-Net can directly map waveform-to-waveform, breaking away from the traditional Encoder-Separation-Decoder structure. Slizovskaia et al. [14] made improvements to the Wave-U-Net network to support a dynamic number of input sources. Cohen-Hadria et al. [15] compared U-Net and Wave-U-Net models and found that pitch shifting is the most effective data augmentation technique for U-Net, while techniques like channel swapping and time stretching show little difference in performance on Wave-U-Net. Meseguer-Brocal et al. [16] proposed the Conditioned-U-Net model, which incorporates a control mechanism that allows training a unique and versatile U-Net network for separating various musical instruments.

2.2. Audio Feature Recognition Methods. The feature recognition in this paper adopts audio fingerprinting technology, which can convert audio files into compact and unique feature vectors. By comparing the features of audio files using audio fingerprinting methods, the desired audio

can be quickly and accurately identified and located. Shazam and Philips proposed classic approaches in the field of audio fingerprinting. Shazam proposed a method based on spectral analysis, extracting spectral peak points with high energy from the audio signal to generate a sparse set of points known as a "constellation map" [13]. Using the information of an anchor point and its surrounding peak points in the constellation map, an audio fingerprint containing frequency, time difference, and the time position of the previous point is generated. Due to the strong robustness and linearity of spectral energy peaks, the extracted fingerprint exhibits high robustness against audio signal compression, foreground speech, and various types of noise. Philips proposed a method that transforms the audio signal into frequency-domain information, divides it into overlapping frames, maps them into 33 frequency bands, and computes the energy between adjacent audio frames to generate fingerprints for matching and retrieval purposes [17]. Jia et al. [18] proposed a modified fingerprint and matching approach to enhance robustness against noise interference. Zhang et al. [19] introduced a turning point alignment method that improves the robustness of sampling and counting methods against time scaling, enabling Philips and Philips-like fingerprints to be resistant to time scaling while improving retrieval performance under different noise distortions. Building upon this, Yao et al. [20] further improved the robustness of the Philips fingerprint by utilizing a band energy calculation method for peak points. This method not only enables the audio fingerprint to be resistant to time stretching and pitch shifting but also maintains robustness against various types of noise distortion. Chu et al. [21] proposed a robust audio fingerprint recognition method against various attacks. They conducted experiments in six different environments, including rhythm, pitch, speed changes, and noise addition, and employed a novel hashing method for audio content comparison in the similarity calculation process, leading to a significant improvement in accuracy. With the development of deep learning, there have also been deep learning-based audio fingerprinting methods [22, 23].

3. The Framework of the Proposed Anime Audio Retrieval

The overall framework of the proposed method is illustrated in Figure 1. The anime audio retrieval method consists of two main components: the audio separation model and the audio fingerprint retrieval model. Firstly, the anime audio is processed through the audio separation model to separate different audio sources. Subsequently, the separated audio sources undergo audio fingerprint extraction, and the extracted audio fingerprints are used to construct an audio fingerprint database. When a specific anime audio segment needs to be retrieved, the same process is applied to obtain its audio fingerprint, which is then matched against the index entries in the audio fingerprint database. Based on the matching results, the corresponding audio segment and its relevant information are retrieved. Table 1 shows the details of the parameters used in the proposed models.

3.1. Anime Audio Separation. Anime audio typically consists of multiple tracks, such as vocals, music, and sound effects, which need to be mixed to create a more immersive and vibrant audio experience. Additionally, anime often portrays fictional or supernatural scenarios and scenes, meaning that the sounds in anime audio are often generated by fictional characters or situations. This implies that anime audio features may involve more digital processing and manipulation. These features often require a broader context to capture, and the ECA mechanism can assist the model in better learning long-term dependencies and global contextual information. Therefore, incorporating the ECA mechanism into the audio separation model can enhance the model's performance and enable better learning of the unique characteristics of anime audio.

The model in this paper adopts an encoder-decoder structure, where the encoder and decoder consist of five downsampling blocks and five upsampling blocks, with a convolutional layer in between as illustrated in Figure 2. Skip connections can be used to connect the encoder and decoder. To better capture the features of anime audio, this paper employs the ECA mechanism module in the encoder part of the audio separation model. This module is inserted between each downsampling block and utilizes an adaptive selection of 1D convolutional kernel sizes to determine the coverage of local interchannel information interaction. This improves the accuracy of extracting audio signal features.

The ECA mechanism is a lightweight attention mechanism that is an improvement over the Squeeze-and-Excitation attention mechanism [24]. It aims to strike a balance between model performance and complexity. Without reducing dimensions, the ECA mechanism calculates the interdependencies between channels by performing 1D convolution along the channel dimension. It weights each channel to achieve interchannel interaction [25]. The ECA mechanism first applies Global Average Pooling (GAP) to the input feature map. The GAP is more native to the convolution structure by enforcing correspondences between feature maps and categories [26]. It can also avoid overfitting. After the GAP, a 1D convolution with kernel size k is performed, followed by a Sigmoid activation function to obtain the weights ω for each channel. Finally, the weights are multiplied elementwise with the original input feature map to obtain the final output feature map.

Moreover, the ECA mechanism enhances the correlation between different channels, thereby improving the expressive power of the network. It enables better capturing of signal features at different frequencies while reducing the number of model parameters, leading to improved training and inference efficiency. In this paper, for the task of audio separation in anime audio, the ECA module precisely captures the unique features of anime character voices, thereby improving the accuracy and efficiency of audio separation.

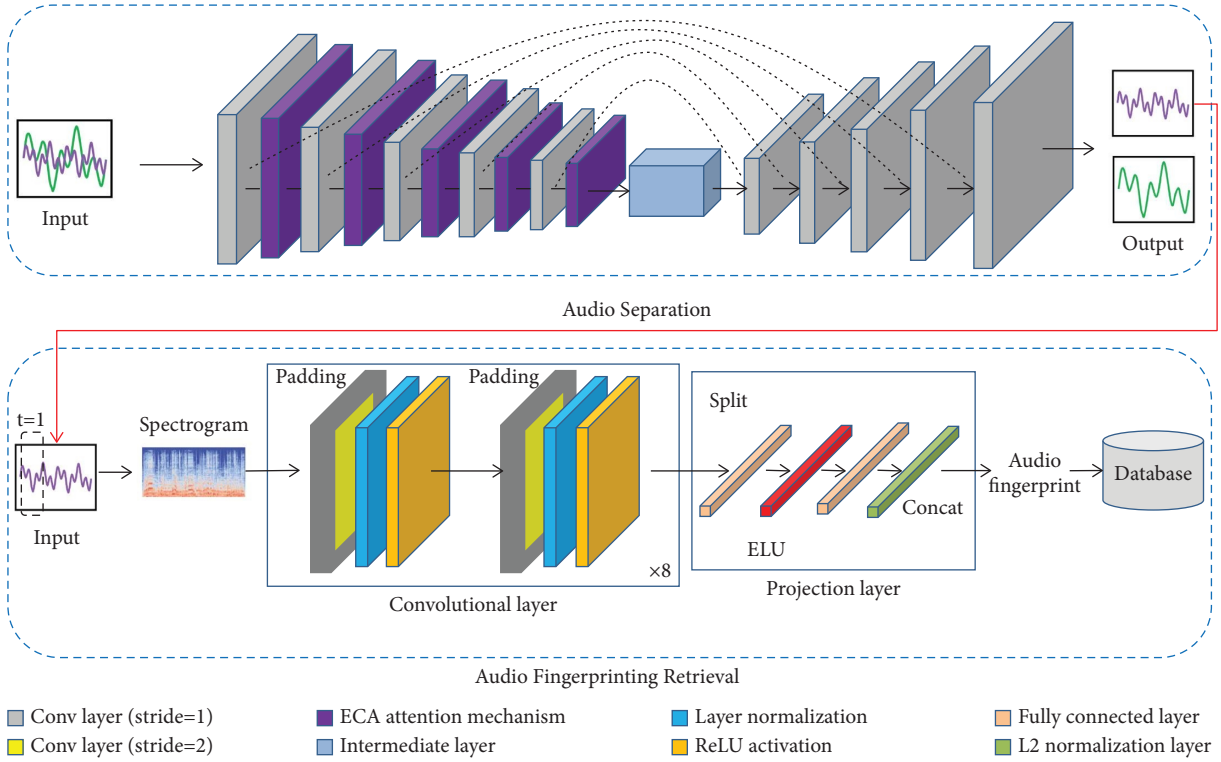

Figure 1: The proposed anime audio retrieval framework.

Table 1: The detail information of the model.

Parameters	Values
Number of downsampling blocks in encoder	5
Number of upsampling blocks in decoder	5
Kernel size	3
Stride of audio separation model	1
Stride of audio fingerprinting model	2
Number of convolutional layers in audio fingerprinting model	8

3.2. Anime Audio Feature Recognition. In this paper, a contrastive learning-based audio fingerprint retrieval method is adopted for anime audio retrieval. After separating the anime audio through audio separation, the separated audio signal is further transformed into a log-Mel spectrogram, which is then fed into the audio fingerprint retrieval model to extract the audio fingerprint of the segmented audio. Finally, the audio fingerprint is searched in the anime audio database to quickly obtain relevant information. Figure 3 illustrates the structure of the proposed audio fingerprint retrieval model.

For each input audio segment, the model converts it into a 128-dimensional embedding vector, which is considered as the audio fingerprint of that segment. These embedding vectors are used for similarity measurement to determine whether they belong to the same audio. To achieve contrastive learning, the model is trained by using the embedding vectors of positive and negative samples to maximize the similarity measurement of positive samples and minimize the similarity measurement of negative samples.

From the figure, it can be observed that the model consists of three main parts. In the preprocessing stage, the audio signal is divided into fixed-length audio segments. Each audio segment includes a portion of the previous segment to facilitate the encoder in learning the similarity between segments during contrastive learning. The audio signal is then subjected to the STFT.

The STFT is an analysis method that converts signals from the time domain to the frequency domain, widely used in audio signal processing and analysis [27]. The STFT is an improvement over the Fourier Transform (FT) [28], which cannot capture the temporal changes of a signal or handle nonstationary signals since it operates on the entire signal. The STFT divides the signal into multiple short-time windows and applies FT to each window to obtain the frequency-domain information at that particular moment. The STFT can capture the temporal changes of a signal, making it more suitable for processing nonstationary signals. The mathematical expression of the STFT is shown as follows:

$$X_f(\omega, u) = \int_{-\infty}^{+\infty} f(t) g(t-u) e^{-j\omega t} dt. \quad (1)$$

$f(t)$ represents the original signal, $g(t-u)$ represents the window function, u represents the center of the window function, ω represents the frequency, and $X_f(\omega, u)$ represents the STFT result. The window function $g(t-u)$ can take various forms such as rectangular window, Hann window, and Hamming window. It applies weighting to the signal within the window to better reflect the frequency-domain information of the signal at that particular moment.

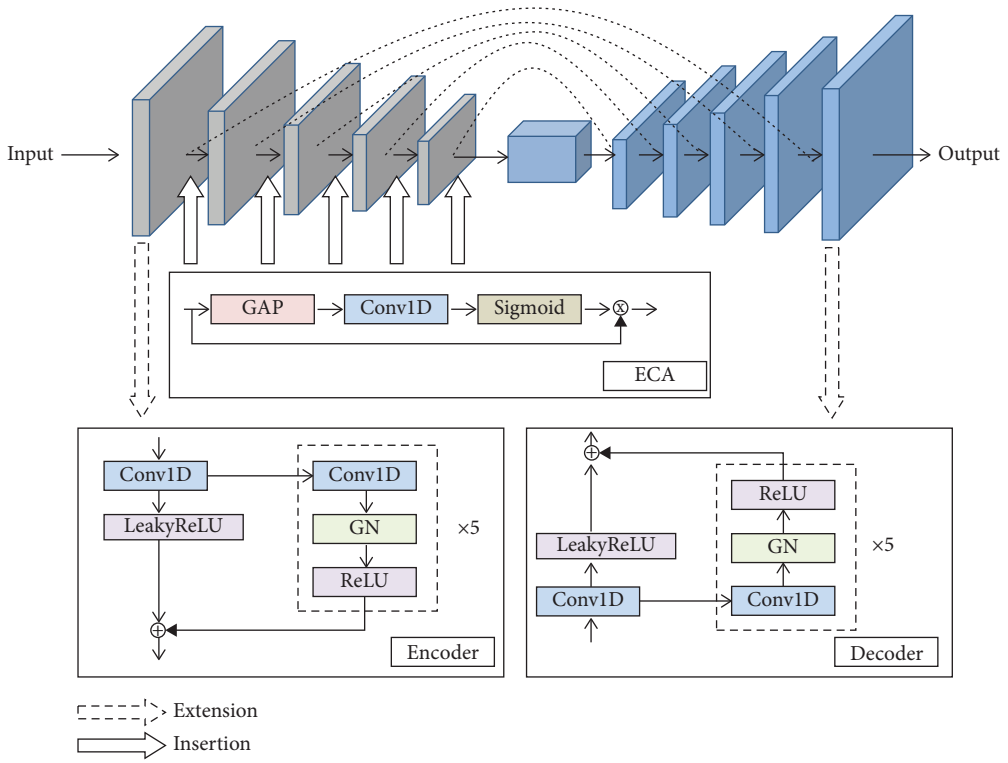

Figure 2: The proposed audio separation model.

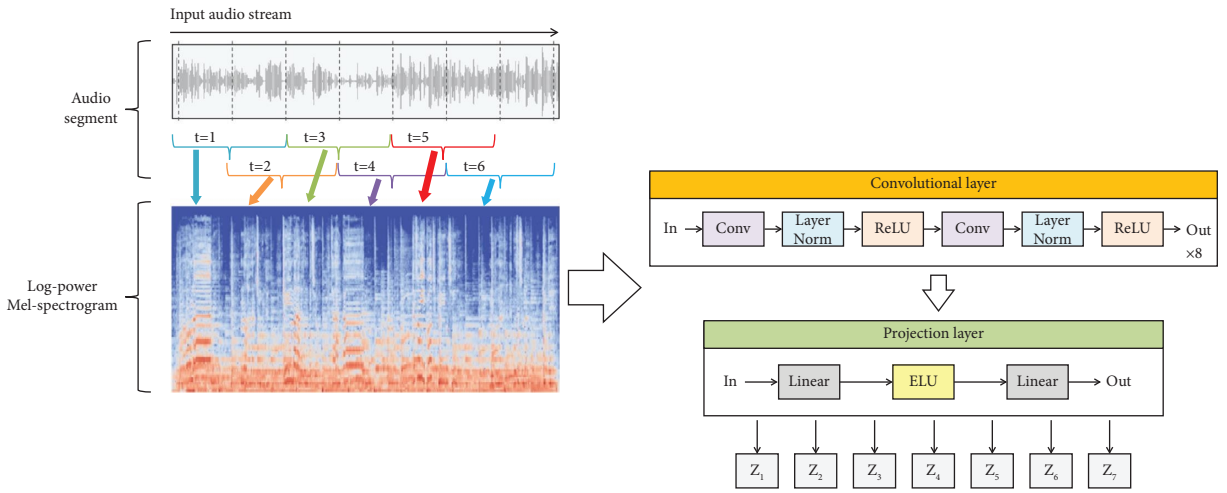

Figure 3: Audio fingerprint retrieval model.

Different window functions have different effects on the STFT result. The window length and overlap ratio are two crucial parameters in STFT. A longer window length provides a finer frequency resolution and a shorter window length provides a finer time resolution in the frequency domain. The overlap ratio affects the smoothness and stability of the analysis result.

After the STFT, we use Mel filter banks to obtain Mel spectrograms. The Mel spectrograms are further processed by taking the logarithm and squaring, resulting in log-power Mel spectrograms. Finally, these log-power Mel spectrogram segments are fed into the model for training.

The convolutional encoder takes the log-power Mel spectrogram as input and performs encoding using depthwise separable convolutions. It consists of two sets of convolutional layers, layer normalization, and ReLU activation functions. The hyperparameter padding (P) for convolutional layer in audio fingerprint retrieval model is set as "same" mode, to keep the output of each convolutional layer have the same shape as the input. Eight of these convolutional layers can transform the original audio signal into a high-dimensional feature vector, enabling the model to learn key features from the Mel spectrogram.

Next, the feature vectors extracted from the convolutional encoder are projected into the contrastive learning space through a projection layer. The projection layer adopts a Linear-Exponential Linear Unit (ELU)-Linear structure. Firstly, a linear transformation is applied to map the feature vectors into a fixed-dimensional space, compressing and reducing the dimensionality of the vectors. Then, the output of the linear layer is passed through the ELU function to introduce nonlinearity, enhancing the model's expressive power and learning efficiency while capturing features at different resolutions and abstraction levels. The transformed features are concatenated and subsequently subjected to L2 normalization. The model parameters are updated through backpropagation by minimizing the contrastive loss function on the training and validation sets, enabling audio retrieval in the test set through similarity calculations.

Due to various interfering factors such as noise and variable speed, anime audio fingerprints may be affected, resulting in the inability to retrieve audio fingerprints accurately or a decrease in the model's recognition ability for certain rare fingerprints due to the unique characteristics of anime audio. To address this issue, the Normalization Temperature-Scaled Cross-Entropy (NT-Xent) loss function [29] is employed to optimize the model in this study. This loss function enhances the model's robustness by learning the semantic relationships among data samples.

The NT-Xent is a contrastive learning loss function based on cross-entropy, used to train neural networks to learn the similarity between samples. It maximizes the similarity score between positive samples and minimizes the similarity score between negative samples to learn the embedding vectors. In contrastive learning, the dataset is typically divided into two parts: one serving as an "anchor" and the other as a "positive," while a random "negative" sample is selected from the dataset. The NT-Xent loss function uses the softmax function to compute similarity scores, aiming to maximize the similarity score between the "anchor" and "positive" while minimizing the similarity score between the "anchor" and "negative." The NT-Xent loss is defined as follows:

$$\ell_{i,j} = -\log\left(\frac{\exp\left(a_{i,j}/\tau\right)}{\sum_{k=1}^{N} 1\,(k \neq i)\exp\left(a_{i,k}/\tau\right)}\right), \tag{2}$$

where N represents the number of samples in a batch, $a_{i,j}$ denotes the similarity score between sample i and its corresponding positive sample j, with higher scores desired, and $a_{i,k}$ represents the similarity score between sample i and other negative samples k, with lower scores desired. The term $1\,(k \neq i)$ outputs 1 when $k = i$ and 0 otherwise. The denominator represents the sum of similarity scores between sample i and all negative samples. τ is the temperature parameter used to control the smoothness of the probability distribution. Additionally, in order to maximize the results of the softmax, the negative logarithm of the loss function is taken. Finally, the total loss L, including the NT-Xent loss, can be calculated as follows:

$$L = \frac{1}{N}\sum_{k=1}^{N}\left[\ell\,(2k-1, 2k), \ell\,(2k, 2k-1)\right]. \tag{3}$$

4. Experiments

4.1. Dataset. Due to the scarcity of research on anime audio, there is a limited availability of datasets specifically designed for anime audio. Most existing audio datasets are focused on natural language processing or music, and their features differ from those of anime audio, making it difficult to directly apply them to anime audio research. Therefore, this paper has constructed an audio dataset specifically tailored for anime. The construction process is illustrated in Figure 4.

A total of 150 anime audio clips were created for this study. The audio files were in WAV and MP3 formats, with a sampling rate of 44100 Hz. Among these, 100 clips were used for training and 50 clips for testing. The anime audio data used in this study were obtained by downloading anime videos from various websites and video platforms. To ensure the performance of the model and the reliability and comprehensiveness of the results, a large and diverse anime audio dataset was necessary for training. In addition to the dataset's scale, the diversity of the dataset also affects the model's results. Therefore, this dataset consists of various anime genres such as science fiction, romance, comedy, and action, as well as anime works in Chinese, English, Korean, and Japanese languages. The FFmpeg library was used to extract all audio tracks and convert them to WAV and MP3 formats, while also performing resampling to ensure consistent sampling rates. To reduce training time, memory consumption, and the risk of overfitting, long anime audio segments were cropped into multiple shorter segments of approximately 3 minutes each for easier processing. Subsequently, incomplete, noisy, or nonrepresentative parts of the audio segments were removed to ensure the quality and usability of the dataset. Finally, the processed audio segments were fed into a model trained on the MUSDB18 dataset to obtain audio tracks such as character voices and background sounds. The MUSDB18 dataset is divided into training set (70%) and test set (30%). The number of epochs is 200 and the batch size is 16. Table 2 shows the detail information of the experimental platform.

4.2. Evaluation Criterion. We employ two evaluation criteria for audio separation including SDR and SIR where higher values indicate better performance. The formulas for the SDR and SIR are shown as follows:

$$SDR = 10\log_{10}\left(\frac{\|S_{\text{target}}\|^2}{\|e_{\text{interf}} + e_{\text{noise}} + e_{\text{artif}}\|}\right), \tag{4}$$

$$SIR = 10\log_{10}\frac{\|S_{\text{target}}\|^2}{\|e_{\text{interf}}\|^2}. \tag{5}$$

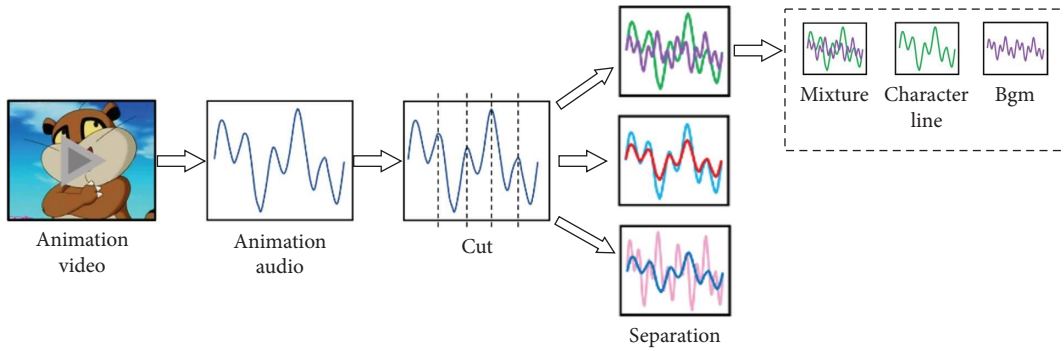

Figure 4: Dataset construction process.

Table 2: The detail information of the experimental environment.

Platform	Attribute
OS	Ubuntu
CPU	Intel (R) Xeon (R) Silver 4210 CPU@2.20 GHz
RAM	64 GB
GPU	Nvidia A40 40 GB
Deep learning framework	PyTorch

S_{target} represents the true source signal, e_{interf} denotes the interference error, e_{noise} represents the noise error, and e_{artif} corresponds to the artificially added distortion.

The audio fingerprint retrieval model utilizes the Top-1 Hit Rate (HR) as an evaluation criterion to measure its performance in both fragment-level and long audio segment-level retrieval. The Top-1 HR represents the ratio of correctly retrieved results among the sum of correctly retrieved results and erroneously nonretrieved results from the top-ranked results. The specific calculation is given by

$$\text{Top} - 1 = \frac{(n\,\text{of hits @Top} - 1)}{(n\,i\,\text{hits@Top} - 1) + (n\,\text{of miss @Top} - 1)}. \tag{6}$$

n of hits@Top − 1 represents the number of matches in the nearest neighbors of the retrieval vector during the Top-1 retrieval process, while n of miss @ Top − 1 represents the number of nonmatches in the nearest neighbors of the retrieval vector during the Top-1 retrieval process.

4.3. Experimental Results and Analysis for Audio Separation. The original anime audio segment is separated into two audio segments: anime character voices and background audio, using the audio separation model. To analyze the results of audio separation, it is necessary to compare the waveforms and spectrograms of the original mixed audio, anime character voices, and background audio. Figure 5 shows the comparison of waveforms and spectrograms for the three audio segments.

Firstly, waveform graphs can be used to observe the temporal characteristics of an audio. In the waveform graph of the original mixed audio, it can be seen that the waveform is complex due to the superposition of multiple audio signals. In the waveform graph of the background audio, the amplitude is significantly reduced. However, in the waveform graph of the anime character voices, the waveform appears relatively simple, indicating that the audio separation model is able to separate them from the mixed audio. Additionally, in the waveform graph of the anime character voices, periodic oscillations of the human voice can be observed, indicating that the separation method is able to preserve the original signal's characteristics reasonably well.

Secondly, spectrogram graphs can be used to observe the frequency-domain characteristics of an audio. From the spectrogram graph of the anime character voices, it can be observed that the frequency components are clear and exhibit distinct resonance peaks, indicating that the separation method is able to preserve the original signal's frequency-domain characteristics reasonably well.

4.3.1. Evaluation of the Audio Separation Performance under Different Loss Functions and Normalizations. To investigate the practicality of different loss functions in audio separation and demonstrate the advantages of the Mean Absolute Error (MAE) loss function in anime audio separation, as well as validate the necessity of the Group Normalization (GN) layer for anime audio separation tasks, ablation experiments were conducted to compare the performance of audio separation with and without the GN layer under different loss functions. The experimental results are shown in Table 3. MSE represents the Mean Squared Error loss function and BN denotes Batch Normalization. The evaluation criteria are SDR and SIR. The "✓" symbol indicates the utilization of the corresponding method.

MSE is one of the commonly used loss functions in audio separation tasks, as it effectively balances the energy differences between different sources, thereby improving the quality and effectiveness of audio separation. However, in the context of anime audio separation addressed in this paper, there are unique characteristics, such as the presence of many silent regions during character speech. It is crucial to accurately separate these silent regions in the given task. Since MSE loss function focuses on the sum of squared errors, it is sensitive to outliers and can adversely affect the model's performance. Based on the aforementioned results, it can be observed that the overall performance with the MSE loss function is not ideal.

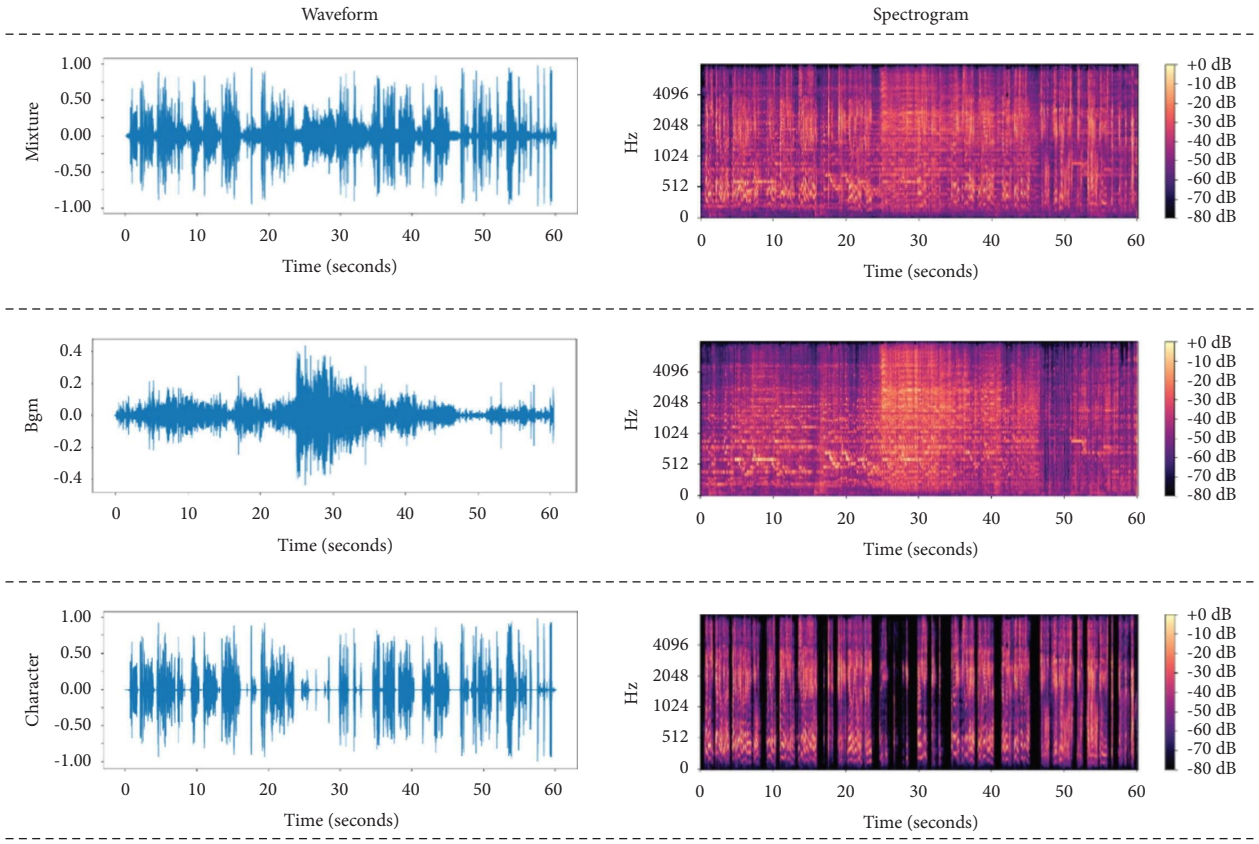

Figure 5: Anime audio separation results.

Table 3: The results of the ablation experiments.

MSE	MAE	BN	GN	SDR (dB)			SIR (dB)		
				Avg	C-line	Bgm	Avg	C-line	Bgm
	✓	✓		5.25	6.56	3.95	15.07	18.05	12.08
✓		✓		0.30	−0.56	1.16	11.51	13.52	9.50
	✓		✓	**6.73**	**8.20**	**5.26**	**15.58**	**18.21**	**12.94**
✓			✓	3.91	5.08	2.73	12.06	14.56	9.56

Bold values represent the result of the best combination.

The MAE loss function, which focuses on the absolute value of errors, is well suited to address the sparsity of anime audio and its impact on the performance of audio separation models. In this paper, the MAE loss function is employed as a replacement for the MSE loss function to better emphasize details, handle the sparsity of anime audio, and improve the effectiveness of audio separation. The results indicate that using MAE yields higher performance compared to using MSE.

The BN is a commonly used normalization technique applied after each convolutional layer in both encoder and decoder of the model. Its main purpose is to normalize the output of convolutional layers, thereby accelerating model training and improving generalization performance. Additionally, BN can mitigate the issue of vanishing gradients, further enhancing the effectiveness of model training. However, in this paper, the anime audio dataset is independently constructed, and the sample size may be relatively small. In situations where the sample size is insufficient, BN may lead to larger batchwise sample variances, which can impact performance.

Due to the presence of multiple audio sources alternating at different time points in anime audio, each batch of audio samples contains different audio sources. GN treats each audio source as a group and performs normalization specifically for each group. This enables more accurate estimation of the mean and variance for each group, resulting in better handling of the situation where multiple audio sources alternate in anime audio. Additionally, GN's computations are not dependent on the batch size, allowing it to achieve desirable performance even with small batch sizes. When combined with the MAE loss function, the combination of MAE and GN achieves a 1.48 dB improvement in the SDR evaluation and also demonstrates improvement in the SIR evaluation, compared to the combination of MAE and BN.

4.3.2. The Performance of Using the ECA. To validate the contribution of introducing the ECA mechanism in enhancing performance, a comparison was made between the performances of audio separation with and without the incorporation of ECA. The results are presented in Table 4, where the best performance values corresponding to the same column are highlighted in bold.

The experimental results demonstrate that the ECA enhances the performance of anime audio separation compared to before its inclusion. Anime audio typically

Table 4: Comparison of audio separation performance with and without ECA.

	SDR (dB)			SIR (dB)		
	Avg	C-line	Bgm	Avg	C-line	Bgm
Without ECA	6.73	5.26	8.20	15.58	12.94	18.21
With ECA	**6.96**	**5.33**	**8.58**	**16.65**	**13.98**	**19.32**

contains a significant amount of environmental noise and background music, which can have a substantial impact on the effectiveness of audio separation. By adopting the ECA mechanism, the influence of noise is reduced, thereby improving the robustness of audio separation.

In the anime audio separation task of this study, the ECA mechanism proves to be effective in reducing the number of model parameters, thereby enhancing the training and inference speed of the model. With the inclusion of the ECA mechanism, both SDR and SIR criteria exhibit improvement over their respective values before incorporation. The average SDR improves by 0.23 dB, and the average SIR improves by 1.08 dB.

To evaluate the audio separation performance of the proposed method, we compare it with other methods: extended Open-Unmix (X-UMX) [30], extended Densely Connected Dilated DenseNet (X-D3Net) [30], Hybrid Spectrogram Time-domain Audio Separation Network (HS-TasNet) [31], and Hybrid Transformer Demucs (HT Demucs) [32]. The comparison results are shown in Table 5. The average SDR of the proposed method with ECA is 6.96, higher than those of the HS-TasNet and HT Demucs.

4.4. Experimental Results and Analysis for Audio Feature Recognition. In this paper, L2 index is selected as the indexing method. L2 index is used for efficient nearest neighbor search in a vector collection. It constructs a data structure in the vector space, partitioning the vector collection into multiple subspaces to accelerate the search process. In L2 index, the objective of the search is to find the nearest neighbor vector to the query vector. L2 index uses the Euclidean distance to measure the distance between vectors and assigns each vector to the corresponding subspace. In this study, the L2 distance is used to calculate the similarity between two vectors. By comparing the loss function, the L2 distance between two vectors of the same audio is minimized, while the L2 distance between different audio vectors is maximized. Table 6 presents the HR of audio fingerprint retrieval using the L2 index.

Audio retrieval experiments were conducted on anime audio segments with lengths of 1 s, 3 s, and 5 s. Top 1 exact indicates whether there exists an audio in the returned results that is an exact match to the query audio for each retrieval. Top 1 near indicates whether there exists an audio in the returned results that is very close to the query audio for each retrieval. Top 3 exact and Top 10 exact indicate whether there exist exact matches in the top three and top ten most similar audios, respectively, for each retrieval.

The results of comparing different indexing methods are shown in Table 7, including Inverted-file index (IVF), IVF-Product Quantization (PQ), and Hierarchical Navigable Small World graph (HNSW) index. From the experimental results, it can be observed that the performance of the model improves as the length of the query audio sequence increases. When the query length is only 1s, except for the HNSW indexing method, the top-1 HRs of the other indexing methods reach around 80%. When the query length increases to 3 s, the segment-level HR can be improved from around 80% to around 90%. The L2 indexing method achieves the best performance, with segment-level HRs of 81.10% $\longrightarrow$ 93.30% $\longrightarrow$ 97.50%. The performance difference between approximate matching results and exact matching results is 3.15% for 1s queries. This difference decreases as the length of the query audio increases.

To further demonstrate the superiority of the proposed method, we compare the retrieval performance of the proposed method with those of other methods: Neural Audio Fingerprint (NAF) [33], Robust and Lightweight Audio Fingerprint (RLAF) [34], Attention-based Audio Embeddings (AAE) [35], and Contrastive Learning-based Audio Fingerprinting (CLAF) [36]. Table 8 shows the comparison results of Top-1 HRs with 3s audio segments using different methods by L2 indexing. The Top-1 HR of the proposed method is higher than that of other methods about 2% to 8%.

Table 5: Comparison results of audio separation with different methods.

Methods	SDR (dB)-Avg
X-UMX	5.72
X-D3Net	5.16
HS-TasNet	6.37
HT Demucs	6.24
Proposed method	6.96

Table 6: HRs of L2 index.

Seconds	1 s	3 s	5 s
Top 1 exact	81.1	93.3	97.5
Top 1 near	84.3	94.2	97.7
Top 3 exact	87.8	95.7	98.1
Top 10 exact	90.4	96.5	98.7

Table 7: Comparison of Top-1 HRs in different indexing methods.

Indexing mode	1 s	3 s	5 s
IVF	78.2	92.4	97.3
IVF-PQ	78.9	93.1	97.4
HNSW	55.1	80.4	90.4
L2	81.1	93.3	97.5

Table 8: Comparison results of Top-1 HRs with 3 s audio segments using different methods.

Methods	L2
NAF	89.7
RLAF	91.4
AAE	88.2
CLAF	84.6
Proposed method	93.3

However, as an anime audio contains multiple sounds of several animation characters, performance of separating each character voice needs further improvement, especially when the characters speak at the same time because the more clearly separated audio prompts the more accurate recognition.

5. Conclusion

To improve the efficiency and accuracy of anime audio retrieval, enhance user experience, and support copyright protection and content management, this paper proposes a novel approach that combines audio separation with audio feature recognition for anime audio retrieval. The proposed method utilizes an ECA-based audio separation technique to separate different audio sources within anime audio. Furthermore, an efficient indexing database is constructed to extract and match fingerprints of the separated anime audio sources. Experimental evaluations conducted on multiple anime segments demonstrate that the proposed method achieves fast and accurate anime audio retrieval, improving retrieval efficiency. Additionally, the proposed framework provides effective methods for copyright protection and content management by enabling the tracking and management of audio resources within anime productions. With its wide range of potential applications, this approach holds significant importance in the anime industry and the field of audio processing. In the future work, we will further improve the anime audio separating performance, especially for the anime audio that contains multiple sounds of several characters, to further increase the anime audio retrieval accuracy [37, 38].

Data Availability

The datasets used during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

De Li and Wenying Xu are cofirst authors. De Li was responsible for conceptualization, supervision, project administration, and funding acquisition. Wenying Xu was responsible for methodology, original draft preparation, investigation, data curation, and software. Xun Jin was responsible for methodology, original draft preparation, review and editing, and validation.

Acknowledgments

This research project was supported by the National Natural Science Foundation of China (grant no. 62062064).

References

[1] E. M. Grais and H. Erdogan, "Single Channel speech music separation using nonnegative Matrix factorization and spectral masks," in *Proceedings of the IEEE International Conference on Digital Signal Processing*, Corfu, Greece, July 2011.

[2] A. Hayashi, H. Kameoka, T. Matsubayashi, and H. Sawada, "Non-negative periodic component analysis for music source separation," in *Proceedings of the IEEE Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Jeju, Korea, December 2016.

[3] Y. Luo, Z. Chen, J. R. Hershey, J. L. Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: stronger together," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Processing*, New Orleans, LA, USA, November 2017.

[4] P. Chandna, M. Miron, J. Janer, and E. Gomez, "Monoaural audio source separation using deep convolutional neural networks," in *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation*, Grenoble, France, September 2017.

[5] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Calgary, Canada, October 2018.

[6] Y. Luo and N. Mesgarani, "Conv-tasnet: surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[7] E. Tzinis, S. Venkataramani, Z. Wang, C. Subakan, and P. Smaragdis, "Two-step sound source separation: training on learned latent targets," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Barcelona, Spain, July 2020.

[8] D. Ditter and T. Gerkmann, "A multi-phase Gammatone Filterbank for speech separation via TasNet," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Barcelona, Spain, November 2020.

[9] A. A. Nugraha, A. Liutkus, and E. Vincent, "Multichannel music separation with deep neural networks," in *Proceedings of the European Signal Processing Conference*, Budapest, Hungary, July 2016.

[10] N. Zeghidour and D. Grangier, "Wavesplit: end-to-end speech separation by speaker clustering," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2840–2849, 2021.

[11] A. Jansson, E. Humphrey, and N. Montecchio, "Singing voice separation with deep U-net convolutional networks," in *Proceedings of the International Society for Music Information Retrieval Conference*, Suzhou, China, November 2017.

[12] A. Jansson, R. M. Bittner, S. Ewert, and T. Weyde, "Joint singing voice separation and F0 estimation with deep U-net architectures," in *Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5, New Paltz, NY, USA, October 2019.

[13] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: a multi-scale neural network for end-to-end audio source separation," in *Proceedings of the International Society for Music Information Retrieval Conference*, Paris, France, November 2018.

[14] O. Slizovskaia, L. Kim, G. Haro, and E. Gomez, "End-to-end sound source separation conditioned on instrument labels," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Brighton, UK, May 2019.

[15] A. Cohen-Hadria, A. Roebel, and G. Peeters, "Improving singing voice separation using deep U-net and Wave-U-net

with data augmentation," in *Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5, Coruna, Spain, September 2019.

[16] G. Meseguer-Brocal and G. Peeters, "Conditioned-U-Net: Introducing a Control Mechanism in the U-Net for Multiple Source Separations," 2019, https://arxiv.org/abs/1907.01277.

[17] J. Six, "Panako 2.0: updates for an acoustic fingerprinting system," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference*, Online, November 2021.

[18] M. Jia, T. Li, and J. Wang, "Audio fingerprint extraction based on locally linear embedding for audio retrieval system," *Electronics*, vol. 9, no. 9, p. 1483, 2020.

[19] X. Zhang, G. Zhan, W. Wang, P. Zhang, and Y. Yan, "Robust audio retrieval method based on anti-noise fingerprinting and segmental matching," *Electronics Letters*, vol. 56, no. 5, pp. 245–247, 2020.

[20] S. Yao, B. Niu, and J. Liu, "Enhancing sampling and counting method for audio retrieval with time-stretch resistance," in *Proceedings of the IEEE International Conference on Multimedia Big Data*, Xian, China, September 2018.

[21] R. Chu, B. Niu, S. Yao, and J. Liu, "Peak-based Philips fingerprint robust to pitch-shift for massive audio retrieval," in *Proceedings of the IEEE International Conference on Multimedia Big Data*, Singapore, September 2019.

[22] H. S. Son, S. W. Byun, and S. P. Lee, "A robust audio fingerprinting using a new hashing method," *IEEE Access*, vol. 8, pp. 172343–172351, 2020.

[23] K. Akesbi, *Audio Denoising for Robust Audio Fingerprinting*, 2022, https://arxiv.org/abs/2212.11277.

[24] H. Purwins, B. Li, T. Virtanen, J. Schluter, S. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

[25] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June 2018.

[26] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-net: efficient Channel Attention for deep convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, June 2020.

[27] M. Lin, Q. Chen, and S. Yan, "Network in network," *CoRR*, 2013, https://arxiv.org/pdf/1312.4400.

[28] A. B. Labao, R. C. Camaclang, and J. D. Caro, "Staggered parallel short-time fourier transform," *Digital Signal Processing*, vol. 93, pp. 70–86, 2019.

[29] Z. Q. J. Xu, Y. Zhang, T. Luo, Y. Xiao, and Z. Ma, *Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks*, 2019, https://arxiv.org/abs/1901.06523.

[30] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the International Conference on Machine Learning*, Vienna, Austria, July 2020.

[31] R. Sawata, N. Takahashi, S. Uhlich, S. Takahashi, and Y. Mitsufuji, "The whole is greater than the sum of its parts: improving dnn-based music source separation," 2023, https://arxiv.org/pdf/2305.07855.

[32] S. Venkatesh, A. Benilov, P. Coleman, and F. Roskam, "Real-time low-latency music source separation using Hybrid spectrogram-TasNet," https://arxiv.org/pdf/2402.17701.pdf.

[33] S. Rouard, F. Massa, and A. Defossez, "Hybrid transformers for music source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, Rhodes Island, Greece, June 2023.

[34] S. Chang, D. Lee, J. Park et al., "Neural audio fingerprint for high-specific audio retrieval based on contrastive learning," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, Canada, June 2020.

[35] A. Agarwaal, P. Kanaujia, S. S. Roy, and S. Ghose, "Robust and lightweight audio fingerprint for automatic content recognition," 2023, https://arxiv.org/abs/2305.09559.

[36] A. Singh, K. Demuynck, and V. Arora, "Attention-based audio embeddings for query-by-example," in *Proceedings of the International Society for Music Information Retrieval Conference*, Milan, Italy, November 2022.

[37] X. Mei, X. Liu, H. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Language-Based audio retrieval with pre-trained models. Detection and classification of acoustic scenes and events (DCASE) challenge," *Technical Reports Series*, 2022.

[38] Z. Yu, X. Du, B. Zhu, and Z. Ma, "Contrastive unsupervised learning for audio fingerprinting," 2020, https://arxiv.org/abs/2010.13540.