Hindawi
Mobile Information Systems
Volume 2022, Article ID 8030569, 10 pages
https://doi.org/10.1155/2022/8030569

Research Article

Music Data Feature Analysis and Extraction Algorithm Based on Music Melody Contour

Jingwen Zhang

Music and Dance, Xi'an Peihua University, Xi'an 710199, China

Correspondence should be addressed to Jingwen Zhang; zhangjingwenpeihua@163.com

Received 21 April 2022; Revised 3 June 2022; Accepted 6 June 2022; Published 18 July 2022

Academic Editor: Liping Zhang

Music is a way to reflect people's real-life emotions, and listening to music has become an inseparable habit of the daily life. Text-based music information retrieval is still the main way for people to find music, but this method has obvious shortcomings and deficiencies, and it is a relatively cumbersome and inefficient method. In order to solve this problem, this paper proposes a feature extraction LAM algorithm based on the contour of music melody. Melody is the most important extraction feature in content-based music retrieval. Users can hum a song according to their own memory and then extract the rhythm, melody, and other information of the hummed audio information to match and identify the rhythm and melody features of the original song stored in the database. The retrieval method is based on the melody, rhythm, and other musical features of music and involves many issues such as the expression of musical melody, feature extraction of musical melody, user query construction, music melody matching, and music database construction. With the help of the customized query interface, media information can be retrieved. Finally, the experiment proves that the top-ten hit rate of the LAM algorithm after clustering is 91.3%, the top-three hit rate is 78.8%, and the first hit rate is 71.2%. The approximate symbol matching DP algorithm has a top-ten hit rate of 83.6%, a top-three hit rate of 66.4%, and a first hit rate of 63.6%. The method proposed in this paper has a high retrieval hit rate.

1. Introduction

Human auditory perception is closely linked to music. It expresses a feeling, a type of emotion that is hard to quantify. The song title, singer, and other aspects of music are determined by this characteristic in audio classification and retrieval technology. Extrinsic information is irrelevant when it comes to music analysis. Traditional audio retrieval uses text-based retrieval technology, which means that the required audio information is retrieved by inputting keywords such as the audio file name, author, and lyrics [1]. Despite their strength, they all have insurmountable limitations because they rely solely on text to describe audio data. It is difficult to put into words how humans perceive audio, such as melody, pitch, and sound quality in music. The amount of audio data available is enormous and explosive. The previous text annotation is not only time-consuming, but also costly, and annotating such large amounts of audio data is impossible. The audio data itself is a binary stream, lacking semantic description, and the text-based annotation is subjective and incomplete. It is impossible for users to keep fresh memories of audio keywords that they have always cared about. Maybe they only remembered a rough melody. At this time, text-based retrieval technology can no longer meet the needs of users [2].

To solve the above problems, melody-based audio retrieval technology came into being. The so-called melody-based audio retrieval refers to the retrieval based on the melody characteristics of the audio, that is, the use of the physical characteristics such as the amplitude and frequency spectrum of the audio signal, the auditory characteristics such as loudness, pitch, and timbre, and the semantic characteristics such as rhythm and melody retrieval. It extracts the semantics and features of objects directly from audio data and then uses this information to search a large amount of audio data stored in the database for audio data with similar features. Melody-based music retrieval has a branch called humming-based music retrieval [3]. It uses the

user's humming or singing to search the music database. The user only must hum a portion of a song, and the retrieval system will search the song database for similar songs based on the melody hummed by the user. Humming-based music retrieval is more convenient, natural, and user-friendly than traditional text-based retrieval methods and provides a better user experience. As a result, humming-based music retrieval is gaining popularity.

Sound is a sound wave produced by the continuous vibration of an object, and the sound wave propagates in the presence of a medium. Nature is full of sounds of all kinds. Related scholars have found that the sound that humans can perceive is actually related to the range of vibration frequencies. Humans can only perceive them in the frequency range of 20 Hz to 20000 Hz. Sound waves beyond human perception are divided into ultrasonic waves and infrasound waves; that is, sounds with a vibration frequency exceeding 20,000 Hz are called ultrasonic waves; sounds below 20 Hz are called infrasound waves. The frequency range of human speech is generally 300 Hz to 4000 Hz, but in music, in addition to the singer's singing, there are also various musical instrument accompaniment sounds [4, 5]. Some of the sounds emitted by these instruments have certain regularity, and some do not, but the frequencies of the sounds cover the entire frequency range that humans can hear. According to the law of vibration, sound can be divided into music and noise. The sound produced by regular vibrations is called tones; otherwise, it is noise. The humming and vocals in the song are both human voices, and their analysis principles and angles are not much different from those of speech analysis [6]. As a kind of natural sound, music also has some basic properties of sound, including time domain features and frequency domain features, which are also applicable to music research. At the same time, as the soul of a piece of music, the extraction and representation of melody are very closely related to the relevant music theory. This article will introduce and analyze the music theory related to melody and then carry out the melody feature matching engine on this basis. Design and related algorithm research will also be introduced.

The innovation of this paper: the paper proposes a music retrieval model based on the feature extraction LAM algorithm based on the contour of the music melody because text-based music information retrieval is inefficient and cumbersome to use. The user only needs to hum a portion of a song for the system to recognize it. According to the melody hummed by the user, similar songs can be found in the song database. Humming-based music retrieval is more convenient, natural, and user-friendly than traditional text-based retrieval methods and provides a better user experience. The article's chapter structure is as follows: the first chapter introduces related scholars' research on music retrieval; the second chapter examines the extraction algorithm of music melody features from short-term energy, endpoint detection, and frequency domain features of music; the third chapter uses class and time-frequency domain mixing to extract music data and conduct comparative experiments; and the fourth chapter is a summary of the full text.

2. Related Work

As an important method and means of information collection, computer information retrieval technology has been developing for decades. With the development of computer software and hardware technology and the Internet, the object of information retrieval has developed from a single text information to two-dimensional images, audio, video, and other multimedia information [7].

Zhang et al. perform feature extraction on the input audio, cut the notes by analyzing the energy, calculate the zero-crossing rate and the autocorrelation function to extract the pitch, and use triples as the unit to represent the melody. In terms of matching, first use the DP algorithm to roughly compare the pitch contour, and then use a more accurate algorithm to compare the pitch interval and duration for the melody whose error is less than a certain threshold. Their system has no restrictions on the user's humming pronunciation; just use the usual "DaDa" pronunciation. Searching in a music library with a scale of 1000 music, only 74% of the top three hits were obtained. However, their idea of graded matching is cited by most of the subsequent studies [8]. Liu et al. used pitch change and length change to encode the melody, which can retrieve 10,000 songs in one second, and achieved a top-five hit rate of 75%. The user must hum to the accompaniment of a metronome. Although the system greatly improves the precision and speed, it is extremely inconvenient for the user to use [9]. Juan and Zhou improved the geometric similarity matching method and proposed a new method of approximate melody matching—Linear Alignment Matching Method (LAM). Their humming system contains 3864 pieces of music and retrieves 62 vocal humming segments. The matching algorithm achieves a top-three hit rate of 90.3%, which is more than 11% higher than the traditional approximate symbol matching algorithm [10]. Chen et al. tried two different similarity calculation methods for humming search. One is to use the distance to estimate the difference between the target and the data in the database; the other is to regard the melody sequence in the database as an HMM form, and the input data as observations sequence, which only matches if some HMM structure appears to be able to generate query sequences [11]. Bradley proposed to use both pitch variation and pitch distribution to improve the performance of the system, and they developed a system called Sound Compass that could retrieve 10,086 songs in 1 second and achieve a top-five hit rate of 75%. However, it needs to be hummed with a metronome when actually using it, which is quite inconvenient and not suitable for most nonprofessional users [12]. Ventura extracts the features of the input audio, calculates the autocorrelation function to extract the pitch, and finally converts to a sequence of triples (pitch curve, pitch distance, and duration). The matching stage adopts two-stage matching. First, the dynamic programming algorithm is used to roughly compare the pitch curves, and then the exact matching is performed after matching. Their system retrieved 1000 songs and achieved a top-three hit rate of 74% [13]. Wee proposed to use both pitch and length to search and use Euclidean distance to

search in the system, and the user input and database content in the system are divided into fixed window lengths. The system first filters out dissimilar data and then compares the remaining data, but this system consumes a lot of memory when running, so it has not been widely used [14]. Hashiguchi proposed a tree-based database retrieval method to reduce the number of matching computations, thereby improving the retrieval accuracy and speed of the system [15]. Thornburg et al. used a method based on note segmentation, which uses the time domain method to segment the notes of the humming song, extracts the pitch contour of each note after segmentation, and then uses three letters to represent the change of pitch. The three letters are *S*, *U*, *D*, where *S* represents the same pitch as the previous note, *U* represents a higher pitch than the previous note, and *D* represents a lower pitch than the previous note. The melody information is represented by a string sequence, and finally, the string fuzzy matching algorithm is used to match the songs in the database, so as to retrieve the required song information [16]. George et al. proposed an enhanced humming retrieval system, which combines the humming melody and the lyrics information of the song, which improves the retrieval accuracy and produces a good retrieval effect [17]. Larrouy-Maestri et al. proposed a music retrieval system that can recognize both cover songs and humming songs. This system uses the combination of HPCP, melody feature, and bassline feature and then uses the Qmax algorithm for melody matching. It also showed good music retrieval performance in subsequent system tests [18].

The matching algorithm chosen in this paper not only is related to the specific system resources, but also has a close relationship with the melody representation method mentioned above. How to better combine them organically is still a hot research issue. This paper aims to design an improved algorithm for matching and retrieval of music melody after clustering analysis [19] of system resources, because of the two characteristics of music melody. The effect on retrieval is the most obvious and direct.

3. Representation and Extraction of Musical Melody Features

3.1. Short-Term Energy Analysis. This paper is extremely important in the delivery of speech and music. Melody is one of the two most commonly used and important basic attributes in related academic research. Similarly, pitch and duration are the most important factors that influence music's perceptual qualities. On the one hand, a melody can be performed by different instruments or people with different pitch and timbre, which means that each instrument or person will have their own unique timbre and pitch, but this has no effect on people's perception of Cognition and evaluation of the melody; on the other hand, if different pitches or rhythms are used to play or sing a specific melody, the melody will be damaged, and in the worst case scenario, the melody may even be lost in cognition. As a result, both data collection and feature extraction, as well as music retrieval, are based on the combined use of pitch, pitch, or length for research and design in melody-based music retrieval. An audio signal's energy varies dramatically over time, and its short-term energy analysis provides a useful description for capturing these amplitude shifts. The formula for short-term energy is shown as follows:

$$E_n = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2. \tag{1}$$

Among them, the first point of the signal represents the short-term energy to start the windowing function. It can be seen that the short-term energy can be regarded as the output of the square of the audio signal passing through a linear filter, and the unit impulse response of the linear filter is $h(n)$. Short-term energy can effectively judge the magnitude of the signal amplitude and can be used to determine whether there is sound or no sound. The analysis found that anomalies may also occur when different audio signals are used. For example, explosions generally only last for a few short-time frames, and the energy carried by the short audio frames before and after the explosion sound is extremely low. If only short-term energy is used for the audible silence detection algorithm, there will be a problem of judging the explosion audio example as silence. The formula for the short-term average zero-crossing rate is shown as follows:

$$Z_n = \frac{1}{2} \sum_{-\infty}^{\infty} |\mathrm{sgn}[x(m)] - \mathrm{sgn}[x(m-1)]| w(n-w), \tag{2}$$

where $\mathrm{sgn}[\bullet]$ is the conforming function, as shown in the following formula:

$$\mathrm{sgn}[\bullet] = \begin{cases} 1, & x(n) \geq 0, \\ -1, & x(n) < 0. \end{cases} \tag{3}$$

The process of human perception of audio signals is closely related to the spectrum analysis function of the human auditory system. Fourier spectrum analysis is a widely used method in frequency domain analysis of audio signals. The basis of Fourier spectrum analysis is Fourier transform. Fourier transform and its inverse transform can be used to obtain Fourier spectrum, autocorrelation function, power spectrum, cepstrum, etc. Therefore, spectrum analysis of audio signals is an important method to recognize and process audio signals. In this paper, based on the original extraction method, the endpoint detection is applied to the pitch extraction algorithm to distinguish the silent segment and the noise segment mixed in the humming sound, so that the result of the note segmentation is more accurate [20]. The process of melody extraction in this paper is shown in Figure 1.

The preprocessing of the sound signal is mainly to window the audio music file to get a sequence of audio clips. In this paper, when the audio signal is windowed and divided into frames, each frame is processed with a Hamming window. The function formula of the Hamming window is shown as follows:

$$w_m(n) \begin{cases} 0.54 - 0.46 \cos \dfrac{6n\pi}{N}, & 1 \leq n \leq N, \\ 0, & \text{other}, \end{cases} \tag{4}$$

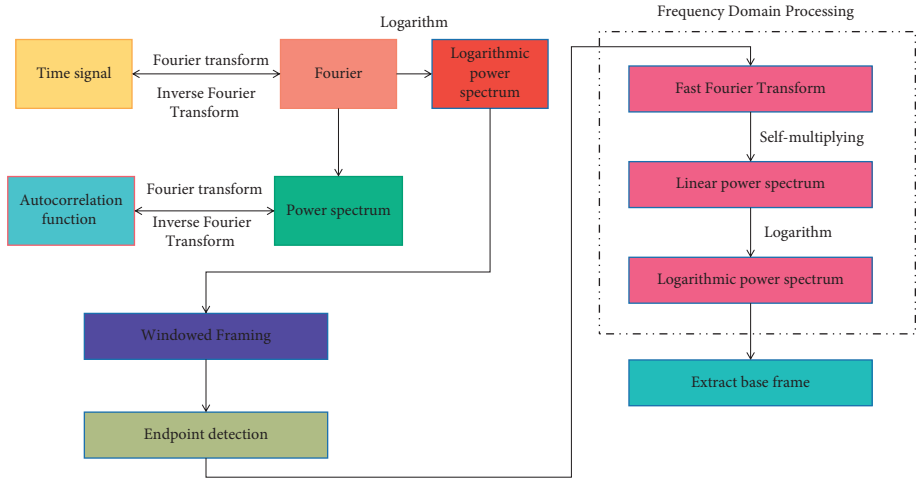

Figure 1: Melody extraction process.

where N represents the frame length and n represents the sampling points in the frame. In order to maintain the continuity of the smooth transition between frames, the overlapping method is adopted for the windowing and framing processing of the audio signal. The overlapping part of the previous frame and the next frame is called frame shift, and the ratio of frame shift to frame length is generally taken as 0.5.

3.2. Frequency Domain Features. Melody was previously expressed as a relative pitch sequence, which was expressed in a string-based manner in some previous music retrieval systems, only those retrieved using a correlation algorithm for string similarity while humming the melody and matching the melody in the music library. At the same time, it significantly obscures the humming melodic features, increasing the likelihood of false retrievals. As a result, the melody is depicted in this paper by the fundamental frequency contour.

Audio signals such as speech and music are all nonstationary random signals. For a nonstationary random process, the traditional standard Fourier transform suitable for periodic, transient, or stationary random signals cannot be used directly. Short-term audio random: the signal satisfies the conditions of the traditional Fourier transform, resulting in the short-term spectrum of the short-term audio signal [21]. Human ear perception is like passing through a filter bank, and the distribution of these filters on the frequency axis is not uniform. There are many filters in the low frequency region, and the distribution is relatively dense, but in the high frequency region, the number of filters becomes smaller [22]. In the Mel frequency domain, the perception of the human ear is linear, and the general frequency is converted into the Mel frequency formula as shown in the following formula:

$$\mathrm{Mel}(f) = 2595 * \log_{10}\left(1 + \frac{f}{763}\right). \tag{5}$$

The basic idea of linear prediction analysis of music signal is as follows: the sampling of audio signal can be approximated by the linear combination of several music signal samples in the past, by making the linearly predicted sampling approximate the actual audio signal sampling in the sense of minimum mean square error. A unique set of prediction coefficients can be obtained. Then, the prediction signal can be expressed as follows:

$$x(n) = \sum_{i=1}^{p} a_i\, x(n-i). \tag{6}$$

Among them, a_i represents the weighting coefficient, which is called the prediction coefficient, and the prediction error is shown in the following formula:

$$\varepsilon(n) = x(n) - \sum_{i=1}^{p} a_i x(n-i). \tag{7}$$

A set of linear prediction coefficients is uniquely determined by making the prediction error to a minimum value under a certain criterion. The short-term average energy of the music signal can well reflect the change of the energy of the signal with time. The number of sampling points in the frame is shown in Figure 2.

The frequency domain energy is based on the Fourier transform coefficients as shown in the following formula:

$$\mathrm{FE} = \log\left(\int_0^w |F(w)|^2 \mathrm{d}w\right). \tag{8}$$

If the frequency domain energy of a certain frame is less than the threshold, the frame is marked as a silent frame; otherwise, it is a nonsilent frame; that is, the frequency domain energy can be used to judge whether it is a silent frame.

3.3. Endpoint Detection. The problem of endpoint detection is essentially a problem of distinguishing speech from noise. Short-time energy detection and short-time zero-crossing rate statistics are commonly used endpoint detection methods. Specifically, short-term energy detection is used to distinguish silent segments, and short-term zero-crossing

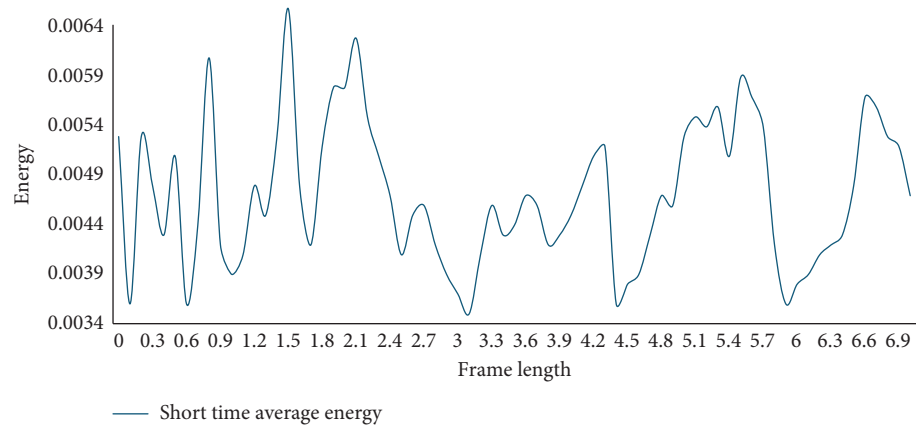

Figure 2: Short-term average energy.

rate detection is used to distinguish noisy segments [23]. The program flow of endpoint detection is shown in Figure 3.

The entire sound signal can be divided into four segments for endpoint detection: silence, transition, speech, and end. If the energy or zero-crossing rate exceeds the low threshold in the silent section, you should start marking the start point and move into the transition section. Because the value of the parameter is relatively small in the transition section, it is unknown whether it is in the real speech section, so as long as the values of the two parameters fall below the low threshold, the current state will be restored to mute. You will be sure to enter the speech section if either of the two parameters in the transition section exceeds the high threshold. If the values of the two parameters fall below the low threshold, and the total timing length falls below the shortest time threshold, while the current state is in the speech segment, it is considered a piece of noise, and the scanning of future speech data continues. Otherwise, return after marking the end endpoint. This paper uses a clustering algorithm to classify candidate song sets before retrieval, then marks the center of each cluster, stores it in the feature database, and accurately matches the music to be matched with each music in the cluster.

In this paper, the clustering algorithm is used to classify the candidate song sets before the matching retrieval; that is, the audio features of the music database are clustered before the retrieval, and the center of each cluster is marked and stored in the feature database. First, match the pieces of music with the center of each cluster, select the cluster class where the cluster center with higher similarity is located, and then accurately match the music to be matched with each music in the cluster. When the fundamental frequency is relatively low, the number of harmonics will be relatively large. When the fundamental frequency is relatively high, the number of harmonics may be relatively small, but the frequency difference between the higher-frequency semitones is also relatively large. The formula for frequency error is shown in the following formula:

$$f_e = \frac{1}{2} * \frac{1}{n} * \frac{T/2}{\text{frame\_len}/2}. \tag{9}$$

Among them, frame_len/2 is the frame length of frame processing, and n is the number of harmonics. Arbitrarily increasing the frame length is also not allowed, and the sound is only stable for a short period of time. If the frame length is too long, multiple sounds of different frequencies at different times may be superimposed together. Since FFT changes will lose time information, it is impossible to distinguish the sequence of these sounds of different frequencies.

4. Improve Music Data Extraction for Musical Melody Contours

4.1. Time-Frequency Domain Mixing. A time series is a way of capturing the process of random events changing and developing over time. Time series analysis is the process of observing and studying a time series, looking for the law of its change and development, and predicting its future trend. In most cases, two approaches are used to calculate the distance between time series. One method is to map the time series to a point in dimensional space and calculate the distance between the sequences using a multidimensional space distance calculation formula, such as the Euclidean distance formula. The Achilles heel of this distance calculation method is that it is overly sensitive to noise, and because each sequence contains more data points in general, calculating the distance takes a long time. The melody feature is used to implement the humming retrieval system in this paper. The fundamental frequency contour is linked to the melody feature. The ability to express and extract the melody correctly is critical to the humming retrieval system because it directly affects melody matching accuracy. The system is implemented based on the fundamental frequency feature sequence in this paper, and the fundamental frequency extraction algorithms of various monophonic humming and the fundamental frequency of composite musicians are studied in depth, the benefits and drawbacks of different melody extraction methods are compared, and a new method is proposed. The cepstrum fundamental frequency extraction algorithm has been improved.

Wavelet analysis is another effective harmonic analysis tool developed based on Fourier transform analysis. Compared with Fourier transform analysis, it is a local transform

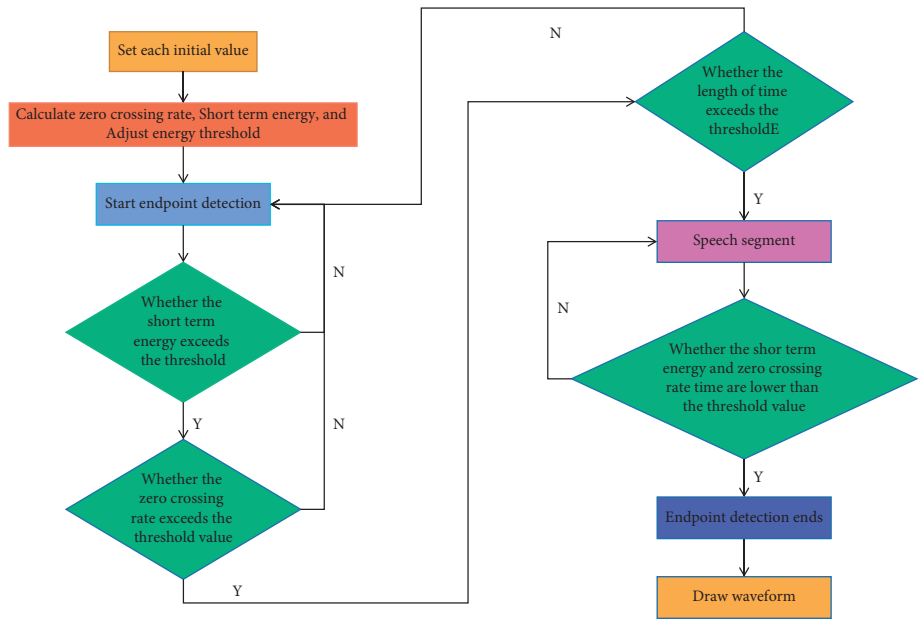

Figure 3: Endpoint detection process.

of time and frequency, so it can effectively extract local information from the signal. The instantaneous spectrum is used to improve the accuracy of fundamental frequency extraction. At the same time, this paper also compares the evaluation index of fundamental frequency extraction with other excellent fundamental frequency extraction algorithms to verify the performance of the improved algorithm proposed in this paper. An audio signal and a waveform of calculated energy values are shown in Figures 4 and 5.

Compare the energy in each frame of audio time, retain the signal value when the energy is the largest, where the frame length is 10 ms, and then perform Fourier transform on the retained waveform signal, and then the root cepstral sequence can be obtained. As shown in Figures 6 and 7, these are the retained audio signal and the frequency domain envelope and cepstrum after Fourier transform.

For cepstrum sequence peak detection, the inverse of the time corresponding to the first peak is the magnitude of the fundamental frequency. Generally, the fundamental frequency of human voice is between 50 Hz and 400 Hz. If the first peak is between 2.5 ms and 20 ms, it is considered that the audio pitch period of the frame is detected. After a series of fundamental frequency sequences are obtained, the fundamental frequency contour needs to be smoothed. Its function is to remove some fundamental frequency points that deviate greatly from the contour and improve the accuracy of fundamental frequency extraction. The time domain feature only uses the audio signal in time. The information on the domain does not need special conversion when extracting, the processing time is short, and it has the advantages of simplicity, small computational complexity, and clear physical meaning. Common time-domain features include short-time zero-crossing rate, average energy, autocorrelation function, and short-time average amplitude difference function. The frequency domain feature needs to convert the time domain waveform signal to the spectral or cepstral domain and then perform the calculation.

4.2. Feature Aggregation. To generate richer audio feature representations, the feature aggregation module effectively fuses the audio-level features obtained by the audio feature module with the music label vectors learned by the label vector extraction module. The module first summarizes the audio-level features using the max pooling and average pooling operations, resulting in multiple one-dimensional audio feature vectors. The max pooling operation extracts representative features from the convolution results, while the average pooling operation summarizes segment features to capture local information. Finally, the feature aggregation module batch normalizes the multiple one-dimensional audio feature vectors and label feature vectors obtained by each pooling layer and then linearly concatenates the vectors to produce the final fusion feature vector.

In this paper, the correctness of the clustering is verified by the hit rate and retrieval effect of the subsequent retrieval. Under the above conditions, the method in this paper and the classical approximate symbol matching algorithm are used to retrieve the humming recordings, and the hit rate of the retrieval is verified. The effectiveness and superiority of the retrieval method in this paper are explained. The approximate symbol matching algorithm is that two note clusters are the linear superposition of their pitch difference and pitch difference and use the appropriate transfer cost to express the melody difference caused by the increase or decrease of the note. Determining the main melody track is directly related to the extraction accuracy of the music feature library. If all the tracks are added to the music library, unnecessary data will be introduced, and the retrieval complexity will be increased. Therefore, the music library should contain the data of the main melody track as much as possible.

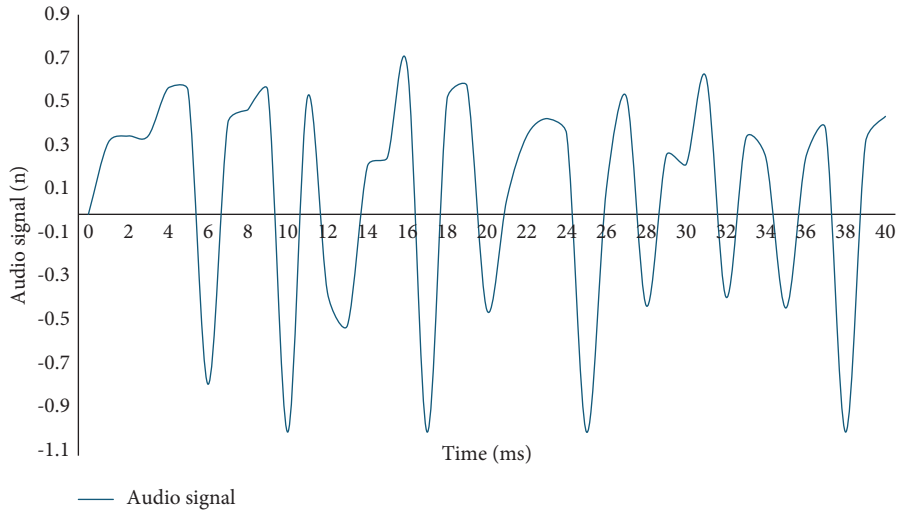

Figure 4: Audio signal waveform.

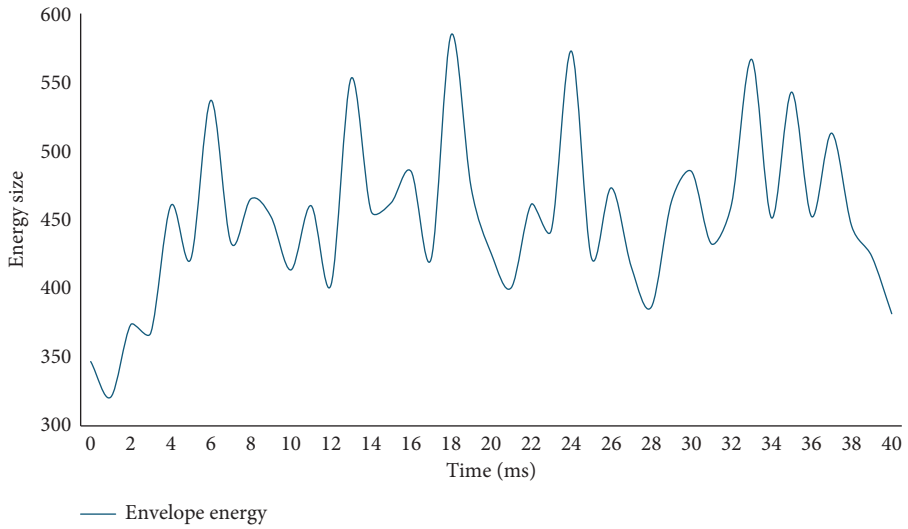

Figure 5: Envelope energy size.

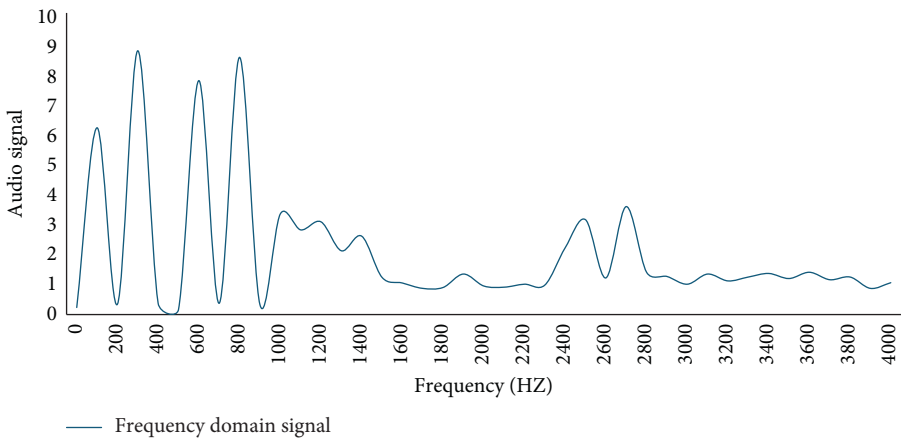

Figure 6: Preserved frequency domain envelope.

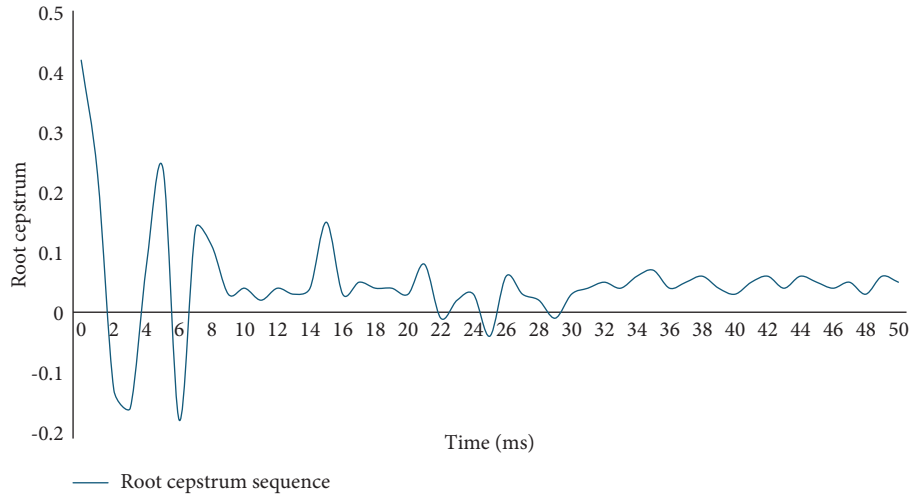

Figure 7: Cepstrum after Fourier transform.

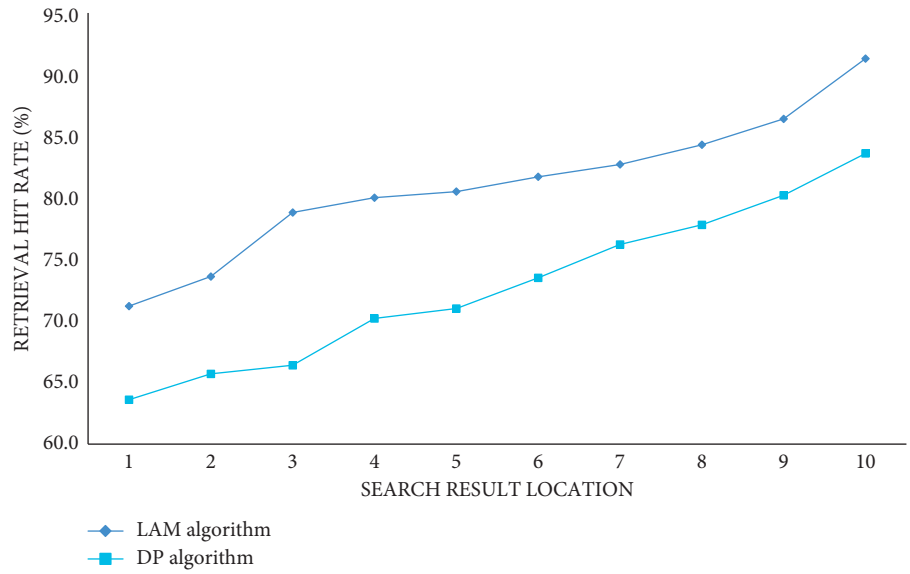

Figure 8: Retrieval hits of two algorithms after clustering.

The LAM algorithm works by first linearly extending two melody, that is, two note sequences, to the same length on the time axis, aligning the notes with close sounding moments within a certain error range, and then examining the melody's rhythmic similarity. Then, at each time point, compare the pitch frequency distance of two equal-length melodies. The melody is expressed in terms of pitch difference, allowing the user to hum at any pitch. Finally, a matching score is assigned based on the rhythm and pitch similarities. On note units, heuristic alignment matching is used. There will be pitch and rhythm errors in the user's humming. Many previous studies have suggested that rhythmic factors be considered in order to match the melody, but none have considered how to optimize and tolerate rhythm errors. The LAM algorithm attempts to break through this barrier by aligning notes that are close to the sounding moment within a specified error range before matching, allowing for the error caused by the user humming the note too long or too short. Figure 8 shows the retrieval hit rates of the LAM algorithm and the approximate symbol matching DP algorithm after clustering.

After clustering, the top-ten hit rate of the LAM algorithm is 91.3%, the top-three hit rate is 78.8%, and the first hit rate is 71.2%. The approximate symbol matching DP algorithm has a top-ten hit rate of 83.6%, a top-three hit rate of 66.4%, and a first hit rate of 63.6%. It can be seen from Figure 9 that applying the LAM algorithm after clustering has obvious advantages in the average retrieval hit rate.

It can be seen from Figure 9 that, for the same scale of music library, the algorithm after clustering is only the time required to use the algorithm. At the same time, if the size of the music library is about 10,000 songs, 2/7, the one-sided continuous matching-based algorithm is applied. After the music clustering algorithm, the running time of music retrieval is only about 2 to 6 seconds. The clustered music library has been well optimized, and the retrieval speed has

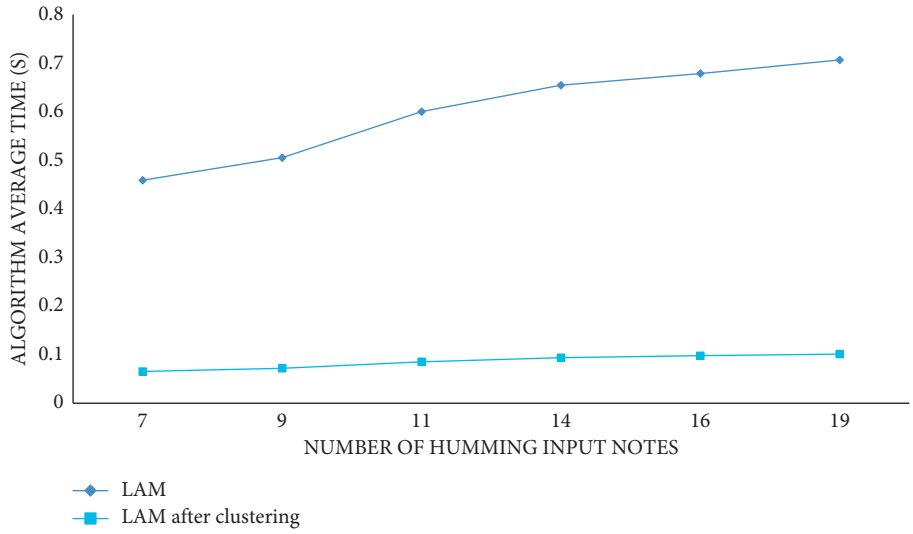

Figure 9: Average time-consuming of the algorithm after clustering.

been greatly improved. The clustering algorithm is a promising music library optimization algorithm. The time required for retrieval is affected by the classification results, but the number of classes will not increase significantly with the increase of the music library, so the increase of the music library will not greatly affect the retrieval time.

The performance test data results in this paper have certain limitations; that is, they cannot be rigorously compared directly with test data in other similar studies. This is because there is no standard library of test samples and test templates in the current humming retrieval field. The tests performed in different studies were performed on different test samples and test template libraries, and it is not meaningful to directly compare the obtained results. It is for this reason that we test two different matching algorithms on the same data set. Only in this way can we perform the comparison scientifically and explain the pros and cons of the two matching algorithms.

5. Conclusion

A crucial step in the music retrieval system is extracting the contour features of the music score from music. This paper uses the LAM algorithm to extract the melody based on an analysis of existing melody extraction algorithms. The melody track of music is segmented, and the humming contour is obtained; the humming contour is converted into a score contour, and a searchable score contour sequence string is formed for the search matching algorithm, using the standard sound to construct a standard pitch difference map and table use. We examine the attributes and features of format files and humming waveform files in terms of feature extraction based on music melody. The main audio track extraction method of multiple audio tracks is analyzed in the file, and a single audio track is used as the main source file for feature extraction to meet the experimental requirements. Experiments on feature extraction show that the endpoint detection method improves feature extraction accuracy significantly. Experiments show that when this algorithm's music features are used for music retrieval, it improves search accuracy and has a certain ability to adapt to noisy environments. At the same time, because the proposed algorithm uses music score information as the search target directly, it will have an advantage in terms of building large-scale music databases and search speed. We store data using two representation methods, string, and pitch contour, which are based on music melody features. The former is used for audio feature clustering analysis, while the latter is used for humming data matching and retrieval. This paper conducts a thorough investigation of the entire humming retrieval system and suggests ways to improve the algorithm's precision rate and search efficiency. It is also difficult to be put into practice, and there's currently no standard set of test samples or data for humming retrieval based on music melody or audio feature clustering. Audio clustering is essentially clustering for different attributes of music in most literature. The accuracy of the classification can only be verified because this paper can only perform similarity clustering on the string information of the entire music. It is indirectly attested by the retrieval's accuracy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] W. Zhang, Z. Chen, and F. Yin, "Main melody extraction from polyphonic music based on modified Euclidean algorithm," *Applied Acoustics*, vol. 112, pp. 70–78, 2016.

[2] K. Zhang, "Music style classification algorithm based on music feature extraction and deep neural network," *Wireless Communications and Mobile Computing*, vol. 2021, no. 4, Article ID 9298654, 7 pages, 2021.

[3] Y. Jiao, "Digital music waveform analysis and retrieval based on feature extraction algorithm," *Advances in Multimedia*, vol. 2021, Article ID 7131992, 10 pages, 2021.

[4] W. U. Yin-Feng and M. Zhang, "Research on a new technology of music melody extraction," *Modern Computer*, vol. 10, no. 21, pp. 133–144, 2018.

[5] S. L. Lai, "A polyphonic music humming retrieval system based on repetitive patterns," *Dissertation of the Department of Information Engineering, Datong University*, vol. 01, no. 14, pp. 26–35, 2012.

[6] Y. Zhao, X. Liu, and T. Su, "Piano accompaniment features and performance processing based on music feature matching algorithm," in *Proceedings of the 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, pp. 525–529, IEEE, Dalian, China, 2021 August.

[7] C. Xizheng, M. Wentao, Q. Kun, X. Cheng, H. Cai, and H. Wen, "Automated Composition Algorithm for Gentle Music Based on Pitch Melody Unit," *Zidonghua Xuebao*, vol. 38, pp. 1627–1638, 2015.

[8] W. W. Zhang, Z. Chen, and F. L. Yin, "Melody extraction from polyphonic music combining modified euclidean algorithm and dynamic programming," *Journal of Signal Processing*, vol. 33, no. 13, pp. 20–35, 2018.

[9] Y. Liu, J. D. Lin, and M. U. Wei-Li, "Melody extraction method from MIDI based on H-K algorithm," *Computer Technology and Development*, vol. 27, no. 16, pp. 41–55, 2011.

[10] L. I. Juan and M. Zhou, "Music database construction based on MIDI melody feature extraction[J]," *Computer Engineering and Applications*, vol. 47, no. 26, pp. 124–128, 2011.

[11] Z. Chen, W. U. Ya-Lian, and H. E. Jie, "Recognition of musical notation based on improved feature extraction algorithm," *Software Guide*, vol. 20, no. 5, pp. 48–62, 2019.

[12] E. D. Bradley, "Phonetic dimensions of tone language effects on musical melody perception," *Psychomusicology: Music, Mind & Brain*, vol. 19, no. 2, pp. 37–55, 2016.

[13] M. D. Ventura, "Relations between melody and rhythm on music analysis: representations and algorithms for symbolic musical data," *International Journal of Applied Physics and Mathematics*, vol. 3, no. 2, pp. 87–91, 2013.

[14] L. H. Wee, "Unraveling the relation between Mandarin tones and musical melody[J]," *Journal of Chinese Linguistics*, vol. 35, no. 1, pp. 128–144, 2007.

[15] H. Hashiguchi, "Visualizing similarity among estimated melody sequences from musical audio," *Springer Japan*, vol. 23, no. 22, pp. 68–71, 2008.

[16] H. Thornburg, R. J. Leistikow, and J. Berger, "Melody extraction and musical onset detection via probabilistic models of framewise STFT peak data," *IEEE Transactions on Audio Speech and Language Processing*, vol. 15, no. 4, pp. 1257–1272, 2007.

[17] A. George, X. A. Mary, and S. T. George, "Development of an intelligent model for musical key estimation using machine learning techniques," *Multimedia Tools and Applications*, vol. 41, no. 2, pp. 79–88, 2022.

[18] P. Larrouy-Maestri, D. Magis, and D. Morsomme, "Effects of melody and technique on acoustical and musical features of western operatic singing voices," *Journal of Voice*, vol. 28, no. 3, pp. 332–340, 2014.

[19] J. Chen, Y. Zhang, L. Wu, Ning, and X. Ning, "An adaptive clustering-based algorithm for automatic path planning of heterogeneous UAVs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, pp. 1–12, 2021.

[20] Y. Goto, "The relation between musical experience and ability to extract of pitch information from melody," *Hokusei Review the School of Humanities*, vol. 46, pp. 55–66, 2009.

[21] E. D. Bradley, "Phonetic dimensions of tone language effects on musical melody perception," *Psychomusicology: Music, Mind, and Brain*, vol. 26, no. 4, pp. 337–345, 2016.

[22] Y. Zhang and L. Mengru, "A method for two dimension visualization of music melody based on MATLAB," *Journal of Shanxi Normal University (Philosophy and Social Sciences edition)*, vol. 5, no. 27, pp. 80–86, 2018.

[23] T. Arai, "Learning about acoustics and signal processing by synthesizing a melody using musical tones and singing," *Japanese Acoustic Society Research and Performance Lecture Proceedings Edited by Japan Acoustic Society*, vol. 27, no. 06, pp. 41–46, 2014.