Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Main melody extraction from polyphonic music based on frequency amplitude and multi-octave relation[☆]

Chen Li, Yajun Liang, Hongmei Li, Lihua Tian [*]

Xi'an Jiaotong University, Xi'an, China

ARTICLE INFO

Keywords:
Main melody extraction
Music information retrieval
Polyphonic music
Pitch contours

ABSTRACT

Most melody extraction methods focus only on the temporal continuity of the frequency, which would result in the poor effect of melody extraction for many algorithms. This paper proposes a new main melody extraction framework based on a new salience function. Firstly In the multi-pitch extraction stage, a new salience function combining the frequency and frequency amplitude continuity between adjacent frames is proposed to select fundamental frequency points. By this way, we can not only retain the latent fundamental frequencies at a maximum, but also reduce the number of pseudo fundamental frequency points caused by noise and other factors. Secondly in the melody selection stage, the contours formed by the obtained fundamental frequencies are grouped based on the multi-octave relation and the contour length. For the contours with a multi-octave relation, to reduce the errors caused by the octave problem, the only candidate contour is selected according to the salience and amplitude of the pitch points in each frame of each group. These contours are then merged with the contours without multi-octave relations. Subsequently, the main melody line of each frame is again determined by the salience of pitch points, and is corrected according to the pitch interval progression relation. Some evaluation experiments are carried out to analyze the proposed method, and the experimental results show that the performance of the proposed method is better.

1. Introduction

With the continuous development of science and technology, there have been significant changes in people's lives [1–7], and human society has entered a new era. However, music remains an important part of our lives. The main melody is the soul of music, which is the basis of many applications, such as music score recognition, pitch analysis, and music theme analysis. From the perspective of music signal processing, music can be divided into monophonic music and polyphonic music according to the number of simultaneous sound sources. Compared with monophonic music, polyphonic music has more complex pitch components, and it is more difficult to extract the main melody. Therefore, in the field of music information retrieval, the main melody extraction mainly focuses on polyphony music. There are many definitions of the main melody of polyphonic music, and the definition given by Poliner [8] is generally accepted in the field of music information retrieval. That definition is that a melody is a single (monophonic) pitch sequence that a listener might reproduce if asked to whistle or hum a piece of polyphonic music, and that a listener would recognize as being the essence of that music when heard in comparison. The sound is generally composed of the fundamental frequency (F0) and harmonics, but its pitch is only determined by the fundamental frequency, even if the energy of the harmonic

[☆] This paper is for CAEE special section VSI-aicv4. Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. Yujie Li.
[*] Corresponding author.
E-mail address: lhtian@xjtu.edu.cn (L. Tian).

https://doi.org/10.1016/j.compeleceng.2021.106985
Received 23 December 2019; Received in revised form 29 September 2020; Accepted 18 January 2021
Available online 6 February 2021

is greater than that of the fundamental frequency. Based on this, the goal of melody extraction is to extract the predominant fundamental frequency of each frame from polyphonic music.

At present, mainstream approaches to achieving melody extraction from polyphonic signals can be divided into three categories [9]: source separation-based methods [10,11], salience-based methods [12–15], and deep learning-based methods [16–18]. Most of the separation models of the source separation-based methods are very complex, and the accuracy of the separation results directly limits the performance of melody extraction methods. In recent years, deep learning-based methods have been developed with the increased use of machine learning. However, with the diversity of music, the lack of labeled data, and the fact that some music features still cannot be expressed correctly, the performance of deep learning-based methods is limited. Therefore, most studies on melody extraction have focused on the salience-based method. In salience-based methods, it is considered that the peaks of the amplitude points are the fundamental frequency points. However, many fundamental frequency points have larger amplitude, and they are not the peak points besides only near the peak points. If the non-peak points in each frame are directly filtered out, many real fundamental frequency points would be lost, resulting in inaccurate melody extraction. Furthermore, most salience-based methods often ignore the influence of harmonic richness on the pitch selection, which also affects the melody estimation result.

In summary, the contributions of this paper are as follow: first, after pre-processing and spectral transformation, a new salience function combining the frequency and frequency amplitude continuity between adjacent frames is utilized to select the existing frequency points. In this way, the latent fundamental frequencies can be not only retained as a maximum, but also the number of pseudo fundamental frequency points caused by noise and other factors can be reduced. Second, some perceived pitches are supplemented through reverse-reasoning by a pair of high-frequency points, which can solve the problem where the fundamental frequency component is occasionally absent in some music (such as opera). Finally, in the melody selection stage, the contours formed by the obtained fundamental frequencies are classified based on the multi-octave relation and the contour length. For the contours with a multi-octave relation, the only candidate contour is selected according to the salience and amplitude of the pitch points in each frame of each group in order to reduce the errors caused by the octave problem.

The remainder of this paper is organized as follows. In Section 2, related works are introduced briefly. Section 3 provides the related theorem and proof. Then, Section 4 clarifies each step of the proposed melody extraction method. In Section 5, the evaluation datasets and evaluation metrics are described. Some evaluation experiments are discussed, and the results are listed in Section 6.

2. Related work

In the field of music analysis, music melody is an important feature for many applications, such as melody-based music retrieval systems and music recommendation systems. In order to meet the needs of these applications, various music melody extraction algorithms have been developed over the past decade. Up to the present, the main approaches of melody extraction from polyphonic music can be divided into three categories: source separation-based methods, salience-based methods, and deep learning-based methods.

In the source separation-based methods, Durrieu et al. adopt a smooth instantaneous mixture model (SIMM) and a smooth Gaussian-scaled mixture model (SGSMM) as the source-separation mode to extract main melody parameters [10]. Huang et al. used robust principal component analysis ((RPCA)) to separate singing voices from music accompaniment [19]. Ikemiya et al. have improved singing voice separation by combining a time–frequency mask based on RPCA with a mask based on harmonic structures [11]. However, this type of algorithm relies heavily on the separated model, and has high computational complexity.

In recent years, owing to the development of deep learning technology, deep learning-based methods have become increasingly active. Basaran et al. [20] have proposed a convolve-cyclical neural network architecture based on the non-negative matrix factorization (NMF) of the source filter. Lu et al. [16] have proposed a melody extraction method based on a semantic segmentation model. In [17] and [18], Sangeun Kum et al. have presented a classification-based approach for melody extraction on vocal segments. It is difficult to obtain sufficient and accurate music prior information as well as the lack of an accurate expression of human auditory perception for music, and all of these factors limit the performance of deep learning-based methods.

Therefore, currently the primary method is still based on the saliency method. In [12], Goto first solves the problem of detecting melodies and bases from real-world audio signals using a salience function. In [21], Degani et al. have proposed an approach that depended not only on the harmonic energy but also on the position of the harmonic frequency. However, this salience function neglects the temporal correlation of the amplitude. In [13], Chen uses the spectral peaks in each frame after a short-time Fourier transform (STFT) as the potential fundamental frequencies for the multi-pitch estimation. If the non-peak points are directly filtered out in each frame, some true fundamental frequencies are lost, and the result of the melody extraction is not sufficiently accurate. In [14], Justin Salamon et al. created melody contours based on time continuity and frequency continuity, and they selected the main melody from the candidate contours according to the characteristics of the contour. If the two adjacent frames satisfying the frequency continuity are located in the melodic and non-melodic segments, the method would incorrectly classify many non-melodic frames as melodic frames. Zhang et al. use the improved Euclidean algorithm to estimate the perceived pitch [15]. However, in this method, many perceived pitches are the same because many harmonics originate from the same fundamental frequency. In addition, the computation requirements of this method are too large because the Euclidean algorithm is repeated multiple times for a pair of frequency points without multi-octave relation. In summary, most melody extraction methods ignore the effect of harmonic saturation on pitch selection, and they do not consider the correlation between frequency and amplitude on the same melody line, which may lead to incorrect results in melody extraction.

Based on music theory, harmonic saturation is a key feature of the main melody, and should be incorporated in our extraction algorithm. The salience function should also be modified to synthesize the frequency and frequency amplitude continuity in order to obtain more accurate pitch candidates. For the situation in which the fundamental frequency component is absent occasionally, we should take more fine processing. In the last melody selection stage, if several contours are obtained, the main melody is identified based on the whole salience. Our proposed approach revolves around the improvement of these aspects.

3. Related theorem and proof

Because the following theorems are used in the methods below, the theorem and its proof are given here. If two numbers a and b (where $a > b > 0$) are integer multiples of $a - b$, then $a - b$ is the greatest common divisor of a and b. The proof is as follows: Suppose that k_1 and k_2 are two positive integers, and two numbers a and b (where $a > b > 0$) are integer multiples of $a - b$, we can obtain

$$a = k_1 \times (a - b) \tag{1}$$

$$a = k_2 \times (a - b) \tag{2}$$

Subtract Eq. (1) from Eq. (2), we obtain

$$k_1 - k_2 = 1 \tag{3}$$

Then, k_1 and k_2 are homogeneous, and the common divisor of k_1 and k_2 is only 1. Furthermore, $a - b$ is the common divisor of a and b, so $a - b$ is the greatest common divisor of a and b.

Similarly, if two numbers a and b (where $a > b > 0$) are the odd integer multiples of $(a - b)/2$, then $(a - b)/2$ is the greatest common divisor of a and b.

4. Salience-based melody extraction framework

The proposed method is designed based on a framework of the salience-based method, as shown in Fig. 1. It consists of three parts: pre-processing and spectrum transformation, multi-pitch extraction, melody creation and selection. In each part, many modified methods are presented. In the following sections, we describe the specific procedures for each section in detail.

4.1. Pre-processing and spectral transform

The purpose of this process is to enhance the signal of the frequency region in which the melody exists and to transform the signal from the time domain to the frequency domain. It includes three steps: pre-processing, spectral transformation, and IF correction. In pre-processing, we apply an equal loudness filter to enhance the music signal. The equal loudness filter is implemented as a 10th-order infinite impulse response (IIR) filter cascaded with a 2nd order Butterworth high-pass filter. The filter can enhance the signals of the frequency range that are more sensitive to humans, which helps in melody extraction [22]. In the process of the spectral transform, the enhanced signal should be framed and processed by the Hamming window. Then, the fast Fourier transform (FFT) with a high-frequency resolution is used for time–frequency transformation. However, music is a non-stationary signal, and the FFT is appropriate for periodic stationary signals, so the frequency-domain signal obtained by STFT cannot reflect the instantaneous characteristics of the original signal.

In this paper, the method based on the phase vocoder can more precisely estimate the true frequency and amplitude [23], and is used to correct the frequency and amplitude of the spectrum after STFT. Suppose that $X_t(i)$ is the STFT of the music signal $x(t)$, then $X_t(i) = a_t(i) + jb_t(i)$, where $a_t(i)$ is the real part of $X_t(i)$ and $b_t(i)$ is the imaginary part of $X_t(i)$. The phase of $X_t(i)$ is denoted as $\phi_t(i) = arctan(bt(i)/at(i))$. The instantaneous frequency $F_i(t)$ of bin i at frame t is computed by the phase spectrum using the phase vocoder method, as shown in Eqs. (5) and (6):

$$F_t(i) = (i + \Delta(i))\frac{fs}{N} \tag{4}$$

$$\Delta(i) = \frac{N}{fs \cdot H}\psi(\varphi_t(i) - \varphi_{t-1}(i) - \frac{H \cdot 2\pi}{N}i) \tag{5}$$

where fs is the frequency sampling rate, and ψ is the function that maps the phase to the $\pm\pi$ range. Δ is the offset of bin. The instantaneous amplitude $A_t(i)$ is calculated by the spectrum magnitude $X_t(i)$ and the bin offset Δ as follows:

$$A_t(i) = \frac{1}{2}\frac{|X_t(i)|}{W_{Hann}(\frac{M}{N}\Delta(i))} \tag{6}$$

$$W_{Hann}(k) = \frac{1}{2}\frac{sinc(k)}{1 - k^2} \tag{7}$$

4.2. Multi-pitch extraction

After the spectral transform, the next task is to extract the pitch of the melody according to the instantaneous frequency and amplitude in each frame, which is multi-pitch extraction. The goal of this part is to keep the pitch point as many as possible and filter out the no-pitch point. In this paper, multi-pitch extraction includes the following three steps.

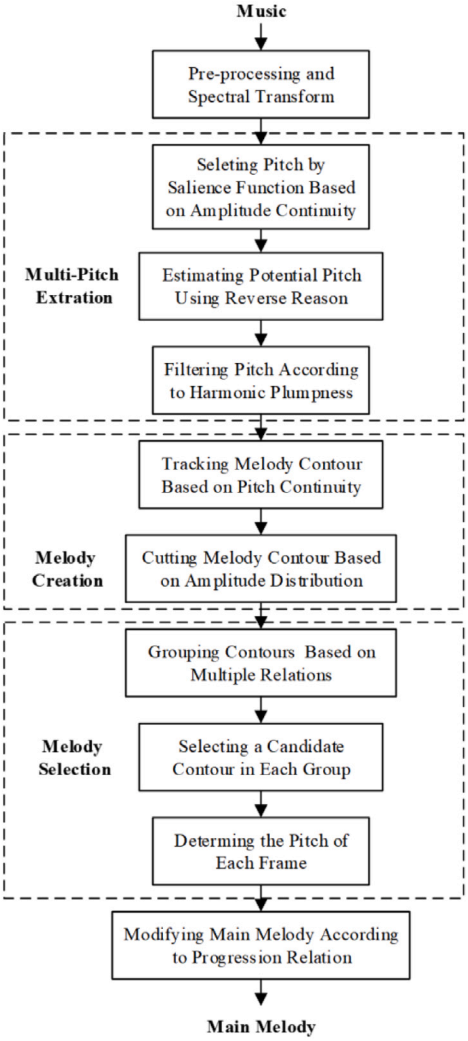

Music
Pre-processing and Spectral Transform
Seleting Pitch by Salience Function Based on Amplitude Continuity
Multi-Pitch Extration
Estimating Potential Pitch Using Reverse Reason
Filtering Pitch According to Harmonic Plumpness
Tracking Melody Contour Based on Pitch Continuity
Melody Creation
Cutting Melody Contour Based on Amplitude Distribution
Grouping Contours Based on Multiple Relations
Melody Selection
Selecting a Candidate Contour in Each Group
Determing the Pitch of Each Frame
Modifying Main Melody According to Progression Relation
Main Melody

Fig. 1. Overview of the proposed main melody extraction method.

4.2.1. Pitch selection using salience function based on amplitude continuity

According to music theory knowledge, the amplitude of the melody is always large, but the fundamental point is not always the peak. In order to reduce the loss of fundamental frequencies, several frequency points near the peak are considered as pitch candidates, which are determined by the proposed salience in this study. In order to obtain as much harmonic information as possible from the frequency band of the human voice, the frequency range from 55 Hz to 2200 Hz is divided into three sub-bands. According to the different influences of the frequency amplitude on timbre in different frequency bands, different threshold coefficients $Thr1$ are used to filter out the frequency points whose amplitudes are less than the product of the given threshold coefficient and the local maximum amplitude in a certain sub-band.

After that, the remaining frequency points are near the local spectral peak. Then, the salience of these points is calculated by using the new salience function, and the frequency points with the local greatest salience are selected as the pitch candidates in each frame.

Now, most existing methods that are based on harmonic products focus only on the relationship between several harmonics in the same frame, and do not consider the influence of the harmonic distribution of adjacent frames on the current frame harmonics. Actually, the frequency amplitudes in the same melody are continuous. To utilize this continuity, the salience function proposed in this paper uses the average frequency amplitude of adjacent frames as the amplitude of the current frame. The salience function is defined as follows:

$$S_t(i) = \prod_{k=1}^{K} \frac{1}{2T+1} (\sum_{j=-T}^{T} A_{t+j}(i \cdot k)) \tag{8}$$

where i is the order number of the sampling point in a frame, t is the order number of the frame, K is the number of harmonics, T is the number of adjacent frames, and $X_t(i)$ is the instantaneous amplitude of the sampling point i in frame t. Higher salience pitches are selected.

4.2.2. Estimation of potential pitch using the reverse-reason

In general, each natural sound consists of a fundamental frequency and harmonics. The fundamental frequency determines the pitch, and the harmonics are frequencies that are integer multiples of the fundamental frequency. However, the energy of the fundamental frequency is at times weaker in some kinds of music (such as opera), but its harmonic components are stronger. In order to reduce the effect of pitch absence, a pair of high-frequency points is drawn to calculate their greatest common divisor and to further estimate the perceptive pitch. Because there are several harmonics for the same pitch, and their greatest common divisors are often the same, we can obtain the perceptive pitch using a pair of high-frequency points, and do not need to calculate all pairs from the same pitch. Therefore, the proposed algorithm only calculates the frequency pairs in which multiple differences are one or two, and it estimates the perceived pitch according to the difference. Assuming that a and b $(a > b)$ are two frequency points, and that $g(a, b)$ is the perceived pitch of a and b, the calculation of $g(a, b)$ is as follows:

$$g(a,b)=\begin{cases} a-b & , \; if \quad r(a,a-b)<\Delta \quad \cap \quad r(b,a-b)<\Delta, \\ (a-b)/2 & , \; if \quad |\, r(a,a-b)-0.5 \,|<\Delta \quad \cap \quad |\, r(b,a-b)-0.5 \,|<\Delta, \\ 0 & , \; otherwise. \end{cases} \tag{9}$$

$$r(x,y)=\left|\, \frac{x}{y}-\left[\frac{x}{y}\right] \,\right| \tag{10}$$

where Δ is an error threshold, $[\bullet]$ represents the round operator, and $|\bullet|$ represents the absolute operator. The formula is proven in Section 2.

4.2.3. Filtering pitches according to harmonic plumpness

The greater the plumpness of the harmonics, the more pleasant does the music sound. Normally, the melody of music has abundant harmonics to express a singer's feelings. That is, there should be as many fundamental harmonics as possible. In all of the harmonics of a certain fundamental frequency, compared with the harmonics with multiple, the harmonics without multiple are irreplaceable and have greater significance for harmonic distribution. And the lower the harmonic multiple, the more important the harmonic is. The frequency points whose harmonic distribution is non-uniform and not abundant are deleted as false fundamental frequency points. Based on this, we make corresponding filtering rules. Because the frequency of most musical instruments and voices is generally less than 2200 Hz, we only consider the harmonics whose fundamental frequency does not exceed 2200 Hz. The detailed rules are as follows:

(1) If the number of possible harmonics is less than or equal to 3, then all harmonics should exist in order to meet the requirements of harmonic richness.

(2) If the possible number of harmonics is greater than 3 and less than or equal to 10, then 1–2 harmonics of the fundamental frequency may be missing, but the even harmonics should account for 50% of the total harmonics. And there should be at least 2 odd harmonics, which should include the 3rd harmonic or the 6th harmonic, and should include the 5th harmonic or the 7th harmonic.

(3) If the possible number of harmonics is greater than 10 and less than or equal to 15, then the 2nd–3rd harmonic can be missing in the harmonics of the fundamental frequency, but the even harmonics should account for at least 60% of the total harmonics, and the odd harmonics should have account for 50% of the total harmonics. There should be the 3rd harmonic or 6th harmonic or 9th harmonic, and there should be at least two of the 5th harmonic, the 7th harmonic, and the 11th harmonic.

(4) If the possible number of harmonics is greater than 15, approximately 20% of the total harmonics of the fundamental frequency can be missing.

The detailed rules can be properly adjusted using different datasets and data types.

4.3. Melody creation

At this stage, we have two subtasks. First, the candidates need to be grouped into the pitch contours, and then the no-melody part of the pitch contour should be removed.

4.3.1. Tracking melody contour based on pitch continuity

From the above definition in Section 1, we know that the melody is the time and frequency continuous sequence, and we track the melody directly from the set of candidate pitch points in a common way based on continuity. During the process of producing the sound, the change in sound from the same source is smooth. Even if there are multiple sound sources at the same time, the sound characteristics of each source are independent of each other, and are not affected by other sources. According to the sound characteristic, we set up the rules of contour construction as follows:

(1) A candidate fundamental frequency point can belong to only one contour line.

(2) The frequencies of the same pitch contour are continuous. This means that the pitch difference between two adjacent fundamental points in one contour is less than 50 cent.

(3) Within a half-interval (50 cent), only one contour can exist.

(4) The minimum duration of the contour is 50 ms.

To implement the above rules, we developed the corresponding process. Note that the entire music piece contains a total of N frames, and the set of candidate fundamental frequency points for each frame is denoted as M_i $(i = 1, \ldots, N)$. The total number of melodic contours is N_C, and the initial value of N_C is 0. The set of fundamental frequency points that belong to each contour created is marked as the set C_j $(j = 1, \ldots, N_C)$. Then, the set of candidate fundamental frequency points is searched and matched frame by frame. The process of contour tracking based on the above principles is as following.

(1) If M_i is empty, continue to search in the next frame. Else, perform step 2.

(2) If there is no fundamental frequency point in the current frame set M_i meets the above principles of contour construction (i.e., the corresponding relationship) for all existing melodic contours, then a new melodic contours should be created. It starts from the fundamental frequency points in the current frame that have no corresponding relationship with all existing contours, and these points should be deleted from the set M_i. Note that the number of search failures for one existing contour is set as num_f, and the initial value of num_f is set to 0. The value of num_f is increased by 1 when this contour does not find a fundamental frequency point with the corresponding relationship in the current frame, and then jump to step 4. If there exist such frequency points for a contour j with the corresponding relationship, continue step 3.

(3) If there are many fundamental frequency points in M_i that have a corresponding relationship with an existing contour j, select only the point with the highest saliency according to Eq. (9) to add the set C_j of contour j, and delete all the points related to contour j from the set M_i. Then, perform step 4.

(4) Check the number of search failures num_f for all contours. If it exceeds the threshold $p1$, end this contour and record it. This means that this contour would not participate in the following search and match. Continue step 5.

(5) If the candidate sets of all frames are searched, then check and output all the resulting contour sets whose duration exceeds 50 ms. Otherwise, continue to search backward for the next frame, that is, $i = i + 1$, and then jump to step 1.

4.3.2. Cutting pitch contour based on amplitude distribution

The pitch contour that was created in the previous steps is not entirely correct. There is this phenomenon where two pitches satisfying the time and frequency continuity may come from the melody frame and the non-melody frame, respectively. By performing many experiments, we can find that overall, the pitch amplitudes of the melodic segment are continuous, and are larger than those of the adjacent non-melodic segment in the same contour. According to the above contour characteristic, every contour is cut as follows. First, the pitch amplitude of a contour is normalized to avoid the influence of the individual abnormal amplitude on the accuracy. Then, the range from 0 to 1 is evenly quantified into N bins, and the number of pitch points in each bin is counted. Finally, the normalized amplitude is filtered by the threshold, which is the first valley of the N numbers.

At the beginning and end of the contour, the pitch can remain stable, but the corresponding amplitude continuously increases or decreases. Therefore, some pitch points of the real contour would be filtered out by a threshold. To obtain the complete contour, the new start and end points of the contour are modified to their nearly valley. At the same time, the filtered part with lengths less than the threshold $p2$ is saved to avoid splitting a real contour into some short ones. We can remove the segments of the contour, which are the weaker or non-melody frames, and effectively determine the start and end points of the contour using the above steps.

4.4. Melody selection

The purpose of the melody selection is to select the main melody from many obtained melody contours. In order to reduce the occurrence of octave errors, contours are grouped according to the multi-octave relation. For the obtained contours with multi-octave relations, only one candidate contour in a group is selected by scoring based on the salience and the amplitude. And then these contours are merged with the contours without multi-octave relation. Finally, the main melody line of each frame is again determined by the salience of pitch points.

4.4.1. Grouping contours based on multi-octave relations

From musical theory, the time interval of the fundamental frequency contour and that of the harmonic contour derived from the same melody are generally approximately equal. Thus, the contours with the multi-octave relation, but for which the overlap time is short cannot originate from the same melody.

Among all the contours, we find the longest contour and compare the remaining contours with it. The contours whose temporal overlap with the longest contour exceeds 80% are selected from the set of contours, and the multi-octave relation between every two contours is calculated. The two contours whose overlap times are very short are considered irrelevant. If the frequency ratio of the two contours at the corresponding time points cannot be stable, they are not in an octave relation. If two short contours are very close and have multi-octave relations with the same contour, it is considered that a real long contour is divided into two short contours, so the two short contours should be merged and the interval is filled. If there are short contours whose length is less than half of the longest contour, they are deleted. Finally, the contours that have multi-octave relations are seen as a group, where the group ID and its multiples are recorded. If a contour does not have a multi-octave relation with other contours, it is separately divided into a group.

4.4.2. Selecting a candidate contour in each group

For many contours in the same group, the candidate contour is determined by comparing every two contours. For the compared contours, scoring is performed by comparing the product of the salience and the amplitude of the pitch point in each frame. We consider not only the salience of each fundamental frequency in each contour, but we also consider the magnitude and length of the contour in order to reduce the octave error caused by relying only on salience. For comparison, assuming that a and b represent the two contours, and their scores are a_score and b_score, respectively, the calculation of the scoring strategy is as follows:

$$\begin{cases} a\_score = a\_score + 0.5 & , \quad if \quad A_a(t) = 0, \\ b\_score = b\_score + 0.5 & , \quad if \quad A_b(t) = 0, \\ a\_score = a\_score + 1 & , \quad if \quad A_a(t)S_a(t) > A_b(t)S_b(t), \\ b\_score = b\_score + 1 & , \quad if \quad A_a(t)S_a(t) < A_b(t)S_b(t). \end{cases} \tag{11}$$

where $A_a(t)$ and $A_b(t)$ are respectively the instantaneous amplitudes of contours a and b at time t, and $S_a(t)$ and $S_b(t)$ are respectively the salience of contours a and b at time t. When $A_a(t)S_a(t) == A_b(t)S_b(t)$, neither score is added.

4.4.3. Determining the pitch of each frame

After the previous step, there is only one contour left in each group, and the remaining contours do not have multi-octave relations with each other. The most significant fundamental frequency point is selected as the candidate for the main melody in each frame. These candidate points may be scattered, but we can roughly identify the main melody from them. The next step is to cluster them into a continuous line using the smoothing method. First, the remaining pitch points of the contour that have lost more pitch points are moved to the adjoining contour. Then, the small gap (less than 50 ms) in the remaining contour is filled, and the fragment with the most salient points is saved in the overlapping time interval. Finally, two contours satisfying the following conditions are merged into one contour:

(1) $t_{i,start} - t_{j,end} < 50$ ms

(2) $| f_{i,start} - -f_{j,end} |< 50$ cent

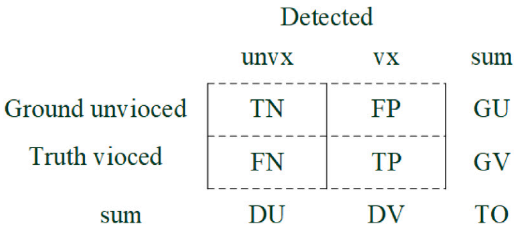
(3) $Min(X_i, X_j)/Max(X_i, X_j) > m$ where $t_{i,start}$ is the start time of contour i, $t_{j,end}$ is the end time of contour j, $f_{i,start}$ is the frequency of contour i at $t_{i,start}$, $f_{j,end}$ is the frequency of contour j at $t_{j,end}$, X_i is the mean amplitude of contour i, X_j is the mean amplitude of contour j, and m is the threshold.

4.5. Modifying the main melody

In music, there is normally a progression of the pitch interval. Different progressions express different feelings of the composer. However, the progression is generally maintained within the three intervals (called step-progression) to maintain the stability of the interval, and there is rarely progression over the three intervals (called leap-progression). After progression, it can keep the new pitch or return to the original pitch according to the degree of leap-progression. There is no continuous leap-progression of single notes.

This paper proposes a main melody modifying method according to the above-mentioned music theory knowledge for the progression. Before modifying the melody, we first traverse all melodies in time order to obtain the progression information between the melody contours. Let the total number of existing melodic contours be M, where the detection is carried out with a sliding window and each sliding window contains three consecutive melodic contours. Note that each melody contour is set to be C_i ($i = 1, \dots, M$). We searched each continuous melody contour set noted as $C_{continue_j}$ ($j = 1, \dots, S$), and S is the total number of continuous melody contour sets, and the initial value of S is 1. Let C_{num_j} represent the number of melody contours in each continuous melody contour set. Initially, the first continuous melody line set $C_{continue_1}$ is established with the first melody contour as its first element. The initial number of melody contours in each continuous melody contour set C_{num_j} is 1. The process is as follows:

(1) Start the comparison using the first melody contour C_i of the current sliding window. In this sliding window, if the second melody contour C_{i+1} has a progression relationship with the first melody contour C_i, then check if C_{i+1} has been assigned to an existing continuous melody contour set. If not, establish a new continuous melody contour set with the melody contour C_{i+1} as its first element. Let $S = S + 1$, and $C_{continue_S} = \{C - i + 1\}$, set $C_n umS = 1$, and then continue to step 2. If so, continue to step 2. Otherwise, add the contour C_{i+1} to the existing set $C_{continue_S}$ to which the contour C_i belongs, and the number of contours in this



		Detected		
		unvx	vx	sum
Ground	unvioced	TN	FP	GU
Truth	vioced	FN	TP	GV
	sum	DU	DV	TO

Fig. 2. Relationship between parameters.

set is increased by 1, that is, $C_{num_S} = C_{n}um_{S} + 1$. Then, move the sliding window backward one melody contour and jump to step 4.

(2) Check whether there is a progression relationship between the third contour C_{i+2} and the second contour C_{i+1} in the current sliding window. If so, continue to step 3. Otherwise, move the sliding window backward by one melody contour, and jump to step 4.

(3) Check whether there is a progression relationship between the third melody contour C_{i+2} and the first melody contour C_i in the current sliding window. If so, move the sliding window backward one melody contour to continue step 4. Otherwise, add the contour C_{i+2} to the existing set $C_{continue_S}$, to which contour Ci belongs, and the number of contours in this set is increased by 1, that is, $C_{num_S} = C_{num_S} + 1$. Then, move the sliding window backward by one melody contour and continue step 4.

(4) Check whether $i + 1$ is equal to M. If so, output the final set of continuous melody contours. Otherwise, return to step 1.

If the number of melody contours in a certain continuous melody contour set is 1, the melody contour is marked as a suspected progression out-of-group melody contour, which would be assessed and modified again. Next, we calculate the weighted sum of the mean frequency of all pitch points and the mean frequency of the previous melody as the frequency standard line, and we predict the frequency range of the current melody according to the standard line. If the mean frequency of the current melody is not within the predicted frequency range and its progression information is empty, the current melody needs to be modified. At this time, we select a melody from the melodies that are in the predicted frequency range and which have the same time range as the current melody. The melody that has multi-octave relation with the current melody is preferentially chosen to be modified. In the end, we filter out the non-melody contour using a threshold based on the maximum average amplitude.

5. Evaluation and analysis

In this section, evaluation datasets and evaluation metrics that are mentioned in the Music Information Retrieval Evaluation eXchange (MIREX) are used to evaluate the proposed method by performing some evaluation experiments, and we compare our results with the results of MIREX2018. The results are shown to prove the performance of our methods.

5.1. Evaluation datasets

The ADC2004 dataset consists of 20 clips of music, each of which lasts 10 to 1 s. It includes jazz, daisy, opera, pop, and MIDI. There are 12 clips with voice and eight clips with non-voice. The sampling rate of the clip is 44.1 kHz. The MIREX05 training dataset consists of 13 clips of music, each of which lasts 24 to 39 s. It includes rock, jazz, pop, solo, and piano. Some clips are real recorded songs, while some are songs produced by MIDI files. The sampling rate of the clip is 44.1 kHz. The MIR-1K dataset consists of 1000 clips cutting from 110 karaoke double-channel Chinese songs with a 16 kHz sampling rate, each of which lasts 4 s–13 s. The songs are recorded by 19 amateur singers, and the total time length is 133 min.

5.2. Evaluation metrics

Mirex organizations usually evaluate the main melody extraction method by obtaining the following five evaluation metrics [24]: VD (voicing detection), VFA (voicing false alarm), RPA (aw pitch accuracy), RCA (raw chroma accuracy), and OA (Overall Accuracy). These metrics are calculated using special parameters. The details of the parameters are as follows:

TPs (true positives): frames where the voicing is correctly detected, and these are further broken down into pitch correct and pitch incorrect, say $TP = TPC + TPI$, that is ignoring octave errors, say $TP = TPCch + TPIch$.

TNs (true negatives): frames where unvoicing is correctly detected.

FNs (false negatives): frames that are actually pitched but detected as unpitched, and these are further broken down into pitch correct and pitch incorrect, say $FN = FNC + FNI$, that is ignoring octave errors, say $FN = FNCch + FNIch$.

FPs (false positives): frames that are actually unpitched but which are detected as pitched.

Their relationships are shown in Fig. 2.

Based on these parameters, the evaluation metrics are defined as follow.

Table 1
The loss ratio of the pitch after correcting pitch.

Dataset	ADC2004	MIREX05 train	MIR-1K
DPeak	16.07%	18.86%	8.41%
Ours	10.47%	5.47%	3.47%

The voicing detection (VD) is the probability that a frame that is truly voiced is labeled as voiced.

$$VD = \frac{TP}{GV} \tag{12}$$

The voicing false alarm (VFA) is the probability that a frame that is not actually voiced is labeled as voiced.

$$VFA = \frac{FP}{GU} \tag{13}$$

The raw pitch accuracy (RPA) is the probability of a correct pitch value (to within ±1/4 tone) given that the frame is indeed pitched.

$$RPA = \frac{TPC + FNC}{GV} \tag{14}$$

The raw chroma accuracy (RCA) is the probability that the chroma (i.e., the note name) is correct over the voiced frames.

$$RCA = \frac{TPCch + FNCch}{GV} \tag{15}$$

The overall accuracy (OA) combines both the voicing detection and pitch estimation to give the proportion of frames that are correctly labeled with both pitch and voicing.

$$OA = \frac{TPC + TN}{TO} \tag{16}$$

5.3. Overall performance of proposed method

In order to show the overall performance of the proposed method, we provide an example of a clip of about 6 s of polyphonic music mixed with a voice and instrument in the MIR-1K set. Fig. 3 shows the processing results of each step of the proposed algorithm. Here, the asterisk indicates the frequency points obtained at a certain step of our method, the blue line represents the annotation by hand, and its harmonics are also given by the light dotted line.

From Fig. 3(a), we can see that based only on the selection of the pitch by the salience function, there is often the loss of many fundamental frequencies during the end period of music pieces. After the following estimation of the potential pitch using reverse-reason, we can regain most of the lost pitches, especially in the interval from 4.5 s to 6.5 s. Then, from Figs. 3(c) and 3(d), it can be seen clearly that although selecting in grouping contours, the main melody extracted is basically accurate, except for a few octave errors. After making the necessary modifications according to the interval progression relation, the final melody is fairly accurate. The overall performance of the proposed method is improved.

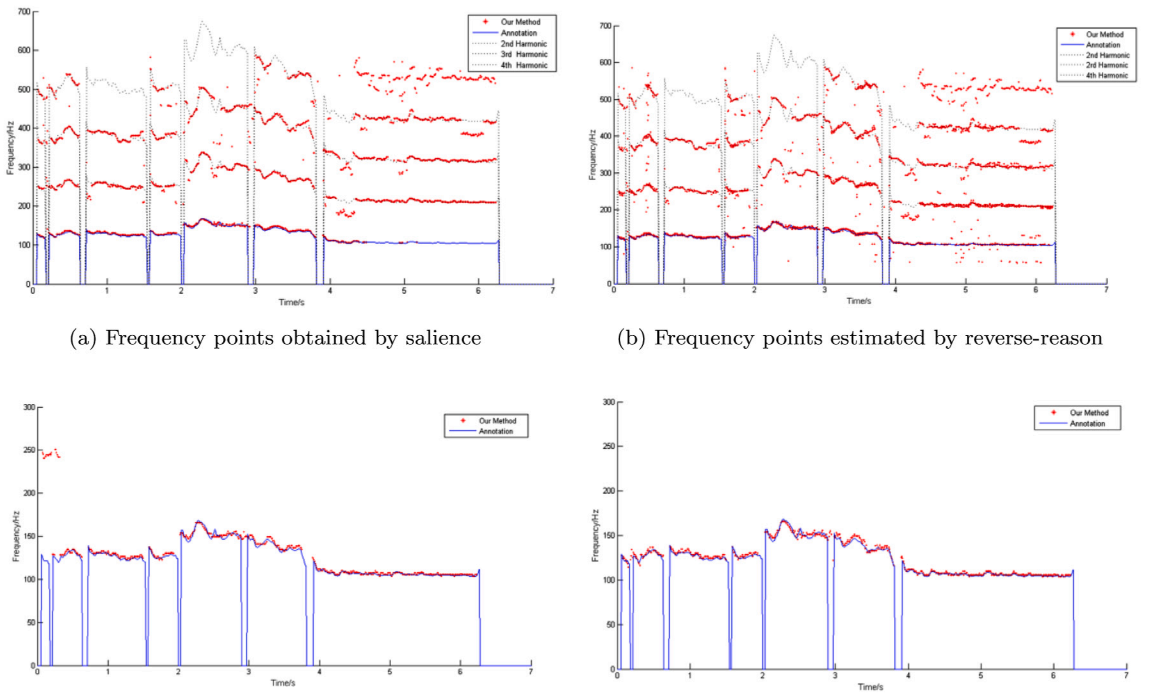
5.4. Performance of coarse selected pitch method

We compare the traditional method of selecting the spectrum peak directly (DPeak) with the coarse pitch selection method that is proposed in this paper, based on three datasets according to the loss rate of fundamental frequency points (i.e., whether there are frequency points within 50 cents above and below the true value). The results are shown in Table 1. Based on Table 1, the method of selecting the spectrum peak directly can make the loss ratio of the pitch as high as 18%, which would seriously affect the extraction of the main melody. By comparison, the proposed method can effectively reduce the loss of fundamental frequency points.

5.5. Performance of perceived pitch method

In this experiment, the obtained spectral peaks are used as the input to estimate the perceived pitch by employing Zhang's method and our proposed method. Then, the accuracy of the two methods is compared by comparing them with the labeled pitch, as shown in Fig. 4 and Table 2.

As shown in Fig. 4, both methods can effectively reduce the loss of fundamental frequency. The loss rate of the proposed method is lower than that of Zhang's GCD method in the first two datasets, and it is more than that in the third datasets. However, from Table 2, it is obvious that the pitch candidates of the proposed method are much less than those of Zhang's method, which reduces the influence of non-pitch points on the selection of the main melody.



(a) Frequency points obtained by salience (b) Frequency points estimated by reverse-reason

(c) Main melody selected by grouping contours (d) Main melody modified by interval progression relation

Fig. 3. The results of each step of the proposed algorithm.

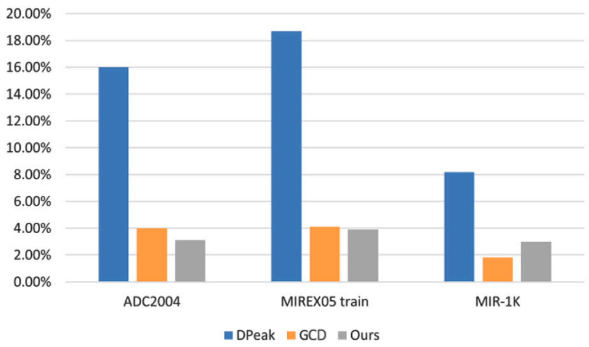


Fig. 4. Loss ratio of the pitch after correcting pitch.

Table 2
Number of candidates after correcting the pitch.

Dataset	ADC2004	MIREX05 train	MIR-1K
DPeak	215691	348053	25599
GCD	1830644	2930333	198871
Ours	439521	708634	47822

5.6. Performance comparison of the proposed melody extraction method

The problem solved in this paper is consistent with the audio melody extraction task proposed by MIREX. Therefore, the proposed algorithm is compared with the results provided by the task using the ADC2004 dataset in the MIREX competition in 2018, as shown in Table 3. Among them, IIY1[11] combines a time–frequency mask based on RPCA with a mask based on harmonic structures, in which they first extract the singing voice using RPCA and estimate the FO contour from the separated singing voice by finding the optimal path over an FO saliency spectrogram based on sub harmonic summation (SHS). KN1_Dense [17] and KN3[18] are all proposed by the Sangeun Kum team. They are a classification-based approach for melody extraction on vocal segments

Table 3
Number of candidates after correcting the pitch.

Method	OA	RPA	RCA	VD	VFA
IIY1[11]	0.7187	0.8139	0.8457	0.8488	0.4348
KN1_Dense [17]	0.7034	0.7178	0.7517	0.8286	0.3411
KN3[18]	0.6772	0.6661	0.6954	*0.7686*	0.2246
LS1[16]	0.6586	0.6571	0.6673	0.8086	0.3238
Ours	0.7188	0.7466	0.7719	0.8858	0.3319

using MCDNNs, in which each neural network predicts the pitch label of a singing voice from a spectrogram with different pitch resolutions. The final melody contour is inferred by combining the outputs of the networks and post-processing them with a hidden Markov model. The main difference between them is that the parameters of the network are set differently. LS1[16] is a melody extraction method that is based on a semantic segmentation model. In this method, the problem of pitch extraction is equivalent to the semantic segmentation of a spectrogram. DCNN and APNN are utilized to map the spectrogram into the symbol domain and to further obtain the optimal pitches, which would produce the final melody contour.

From Table 3, it can be seen that the VD of our proposed method is the highest, reaching 88.58%. The higher the VD is, the better the recognition of the method on the melody frame is. The more accurate the recognition of the melody frame is, the more likely the subsequent accurate recognition of the melody pitch is. However, the VFA of our algorithm was also higher (33.19%). The higher the VFA is, the worse the recognition ability of non-melody frames is. This shows that our method misclassifies many non-melody frames into melody frames. The main reason is that we designed the extraction algorithm based on the following two objectives. First, it improves the accuracy of melody frame extraction as much as possible to obtain a higher VD. In order to achieve this goal, we utilized a new salience function to extract the correct melody frame, adopted the reverse-reason approach to correct the wrong melody frame, and modified the main melody using progression characteristics. Second, it filters out non-melody frames and reduces the VFA as much as possible. To achieve this goal, we filter out non-melody frames as much as possible by clipping the contour based on the amplitude distribution. However, after performing many experiments, we found that these two goals cannot be optimized simultaneously by setting parameters. Because the task of melody extraction is primarily to determine the melody frame, the priority of the first objectives should be focused more on achieving a higher VD when setting parameters.

For the original pitch accuracy (RPA), our proposed method is also better (0.7466). This is because we estimate the potential fundamental frequency using multiple frequency points, thus effectively improving the pitch accuracy. The RCA (original chroma accuracy) is a metric that shows the pitch accuracy after ignoring the octave problem, so the RCA is generally greater than or equal to the RPA. The difference between RCA and RPA can reflect the ability to solve the octave problem. The smaller the difference between RCA and RPA is, the better the ability to solve the octave problem is. The difference between the RCA and the RPA of this proposed method is 2.53%, which is very small. This shows the effectiveness of our method for correcting the melody frame using the reverse-reason method.

Generally, the experimental results show that although the individual metric of the proposed method is not optimal, its overall performance is better.

In addition to comparing the performance of the above evaluation metrics, the performance of the proposed method in music types is discussed. The average overall accuracy of each method is calculated for five music types labeled by the ADC2004 dataset, and the comparison method is the KN1_Dense method with the highest overall accuracy in the results of 2018 and the IIY1 method with the highest overall accuracy in the results of 2016. The average overall accuracy of the different methods for the different music types is as shown in Fig. 5.

From Fig. 5, we can see that all three compared methods have high overall accuracy values for the first two types of music, and the average accuracy of our proposed method is the highest. However, for the latter three types of music, the average accuracy of the three extraction methods is significantly different, and is affected by the characteristics of these types of music itself. MIDID-type music is composed of electronic instruments according to the instructions. Its pitch is generally clearer and more stable, but there are short melodies and large pitch progression. However, the pitch of opera music is continuously variable and extremely unstable. The amplitude of some fundamental frequency is often smaller than their multiples. IIY1 is based on the source separation-based method, which makes it easy to identify the pitch of MIDI, but it cannot solve the octave problem in opera very well. The KN1_Dense method based on MCDNNs can extract the melody frame very well, but its training model is not suitable for MIDI melody and pitch progression characteristics, so the extraction performance on MIDI music is very poor. Therefore, the KN1_Dense and IIY1 methods have some disadvantages when dealing with a certain type of music, and our algorithm has a small difference in terms of the extraction performance for different types of music, so it has better stability.

6. Conclusion

We present a new framework for extracting the main melody based on the melody amplitude. First, Besides considering the frequency amplitude continuity, we introduce the constraint of frequency time continuity in the new saliency function to make the selected fundamental frequency more accurate. Next, we use the improved greatest common divisor algorithm to estimate the potential fundamental frequency and filter out the fundamental frequency lacking harmonics. Then, the melody contour is created according to the time and frequency continuity, and the non-melodic segment is cut off by the distribution of the amplitude in

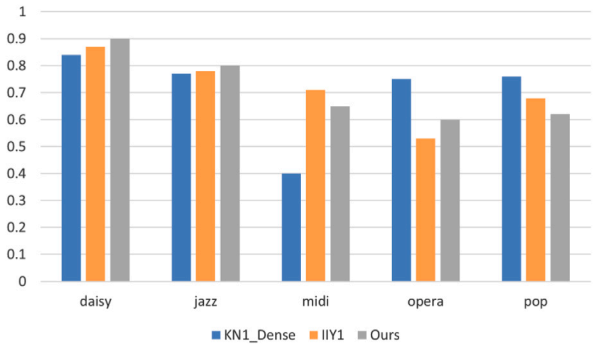


Fig. 5. Average overall accuracy of the proposed method for different types.

each contour. According to the multi-octave relation, we divide the contours into different groups. In each frame of each group, we use the salience and amplitude of the pitch points simultaneously to select a unique candidate again. Finally, the main melody is corrected by the progression relation of the pitch interval. The experimental results show that the proposed method has stable and good performance for all test datasets.

CRediT authorship contribution statement

Chen Li: Conceptualization, Methodology, Visualization, Writing - original draft, Formal analysis. **Yajun Liang:** Methodology, Software, Data curation, Writing - original draft. **Hongmei Li:** Writing - review & editing, Software. **Lihua Tian:** Resources, Supervision, Formal analysis.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.compeleceng.2021.106985.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 61901356 and the HPC Platform of Xi'an Jiaotong University, China. The source code in matlab is open, and any reader who needs them, can visit my homepage http://gr.xjtu.edu.cn/web/cclidd/1.

References

[1] Xu X, Lu H, Song J, Yang Y, Shen HT, Li X. Ternary adversarial networks with self-supervision for zero-shot cross-modal retrieval. IEEE Trans Cybern 2019;50(6):2400–13.

[2] Zhang Y, Wang R, Hossain MS, Alhamid MF, Guizani M. Heterogeneous information network-based content caching in the internet of vehicles. IEEE Trans Veh Technol 2019;68(10):10216–26.

[3] Zhang Y, Ma X, Zhang J, Hossain MS, Muhammad G, Amin SU. Edge intelligence in the cognitive Internet of Things: Improving sensitivity and interactivity. IEEE Netw 2019;33(3):58–64.

[4] Lu W, Zhang X, Lu H, Li F. Deep hierarchical encoding model for sentence semantic matching. J Vis Commun Image Represent 2020;102794.

[5] Zhang Y, Lu W, Ou W, Zhang G, Zhang X, Cheng J, Zhang W. Chinese medical question answer selection via hybrid models based on CNN and GRU. Multimedia Tools Appl 2019;1–26.

[6] Zhou Q, Wang Y, Liu J, Jin X, Latecki LJ. An open-source project for real-time image semantic segmentation. Sci China Inf Sci 2019;62(12):227101.

[7] Zhou Q, Yang W, Gao G, Ou W, Lu H, Chen J, Latecki LJ. Multi-scale deep context convolutional neural networks for semantic segmentation. World Wide Web 2019;22(2):555–70.

[8] Poliner GE, Ellis DP, Ehmann AF, Gómez E, Streich S, Ong B. Melody transcription from music audio: Approaches and evaluation. IEEE Trans Audio Speech Lang Process 2007;15(4):1247–56.

[9] Salamon J, Gómez E, Ellis DP, Richard G. Melody extraction from polyphonic music signals: Approaches, applications, and challenges. IEEE Signal Process Mag 2014;31(2):118–34.

[10] Durrieu J-L, Richard G, David B, Févotte C. Source/filter model for unsupervised main melody extraction from polyphonic audio signals. IEEE Trans Audio Speech Lang Process 2010;18(3):564–75.

[11] Ikemiya Y, Yoshii K, Itoyama K. Singing voice analysis and editing based on mutually dependent F0 estimation and source separation. In: 2015 IEEE International conference on acoustics, speech and signal processing (ICASSP). IEEE; 2015, p. 574–8.

[12] Goto M. A real-time music-scene-description system: Predominant-f0 estimation for detecting melody and bass lines in real-world audio signals. Speech Commun 2004;43(4):311–29.

[13] Chen X, Liu R. Multiple pitch estimation based on modified harmonic product spectrum. In: Proceedings of the 2012 international conference on information technology and software engineering. Springer; 2013, p. 271–9.

[14] Salamon J, Gómez E. Melody extraction from polyphonic music signals using pitch contour characteristics. IEEE Trans Audio Speech Lang Process 2012;20(6):1759–70.

[15] Zhang W, Chen Z, Yin F. Main melody extraction from polyphonic music based on modified euclidean algorithm. Appl Acoust 2016;112:70–8.

[16] Lu WT, Su L, et al. Vocal melody extraction with semantic segmentation and audio-symbolic domain transfer learning. In: Proceedings of the 19th international society for music information retrieval conference; 2018. p. 521–8.

[17] online.https://nema.lis.illinois.edu/nema_out/mirex2018/results/ame/adc04/summary.html.

[18] Kum S, Oh C, Nam J. Melody extraction on vocal segments using multi-column deep neural networks. In: Proceedings of the 17th international society for music information retrieval conference; 2016. p. 819–5.

[19] Huang P-S, Chen SD, Smaragdis P, Hasegawa-Johnson M. Singing-voice separation from monaural recordings using robust principal component analysis. In: 2012 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE; 2012, p. 57–60.

[20] Basaran D, Essid S, Peeters G. Main melody extraction with source-filter nmf and crnn. In: 19th international society for music information retrieval conference, Paris, France; 2018.

[21] Degani A, Leonardi R, Migliorati P, Peeters G. A pitch salience function derived from harmonic frequency deviations for polyphonic music analysis. In: Proceedings of the 17th international conference on digital audio effects, 2014; p. 195–201.

[22] Salamon J, Gómez E, Bonada J. Sinusoid extraction and salience function design for predominant melody estimation. In: Proc. 14th Int. conf. on digital audio effects (DAFx-11), Paris, France; 2011. p. 73–80.

[23] Flanagan JL, Golden R. Phase vocoder. Bell Syst Tech J 1966;45(9):1493–509.

[24] online.https://www.music-ir.org/mirex/wiki/2018:Audio_Melody_Extraction.

Chen Li received Ph.D. degree, in 2008, from School of Electronic and Information Engineering, Xi'an JiaoTong University, China. She is currently an Assistant Professor of Xi'an JiaoTong University. Her research interests include multi-media technology.

Yajun Liang received M.S. degree, in 2019, from School of Electronic and Information Engineering, Xi'an JiaoTong University, China. His current research interests include audio analysis, audio compression.

Hongmei Li received the bachelor's degree in computer science and technology from Tianjin University, Tianjin, in 2019 and she is pursuing the M.S. degree in school of software engineering in Xi'an Jiaotong University, Xi'an. Her current research interests include multi-media technology.

Lihua Tian received the M.S. and Ph.D. degrees in pattern recognition and intelligent system from Xi'an Jiaotong University, Xi'an, in 2005 and 2012, respectively. Her research interests are image/video processing, computer vision, and multimedia application.