

Research on Music Adaptive Method Based on Dance Robots

Suli He^{1,a,*}, Min Liu¹ and Liguang Dong¹

¹ South China Institute of Software Engineering GU, Department of Electronic Studies, 510990, Guangzhou, China

Abstract. Music adaptive method, one of the important directions of dance robot research, will improve the robot's intelligence and adaptive ability, thereby greatly enriching people's growing spiritual and cultural life. From the current point of view, an overly complex adaptive method used, a large amount of computer resources will be encroached on, and the dance robot system will freeze or even crash. The dance robot is a kind of embedded device. Under the premise of saving system resources, how to make the robot do different actions according to different music and how to make it produce movements in step with the music will become a hot topic in the research of robot adaptive methods. This paper proposes a dance robot adaptive method based on music analysis. After the music signal is preprocessed, a reasonable threshold range is set to obtain music features. Through the feature and coordinate transformation, the action duration is obtained, alternative actions are selected, and action frequency is set to achieve the coordination of amplitude change and action switching. Experiments have shown that with this method used on a dance robot, the system runs smoothly featuring coordinated movement, hence a better adaptive effect.

1 Introduction

Amusement robots, an important branch of robots, is an important field of robot research. Dance robots, one of the amusement robots, will greatly enrich people's spiritual and cultural life. One one hand, with the general standard of living improving, people have great demand for amusement robots. Therefore, research on dance robots will help to enrich people's life. On the other hand, with a great variety of musics, the dance movements of robots can hardly be coordinated with all kinds of musics. More often than not, it takes a great amount of manpower to do manual programming. In this sense, research on music adaptive method based on dance robots is of vital practical and lasting significance.

Music adaptive method of dance robots involves two parts: music analysis and motion matching of robots. Currently, as China's research on dance robot adaptive method based on music analysis is still in the stage of initial development, few finished products in this regard have been successfully developed. On one hand, among the existing methods of music analysis, with such processing techniques as short-time autocorrelation function, short-time frequency-domain analysis or WT (Wavelet Transform) adopted, the computational process is so complex that it not only encroaches on a large amount of computer resources, but also causes the dance robot system to freeze or even crash. On the other hand, in the motion matching process of the robot, uncoordinated movement switching is a frequent occurrence. In a word, if the traditional approach is adopted, the robot's requirement for self-adaptive can hardly be met.

To sum up, this paper puts forward preprocessing technique, i.e., acquiring feature points through delimiting threshold values, and obtaining the duration of dance movements based on coordinate transformation so as to select alternative movements and movement speed, hence the dance robot's adaptive to music.

2 Music Signal Processing

The four basic features of music, i.e., pitch, sound length, volume and tone color, determines its advanced features including melody, meter and rhythm of music. In terms of musical performance, pitch and sound length are more important. For example, if the pitch and sound length of a piece of music remain unchanged, using a different musical instrument, we can still recognize this piece of music despite different tone colors. However, once its pitch or sound length is changed, the meter, rhythm and melody of the music will change. Pitch may find expression in range while sound length in time. Therefore, music range is acquired with preprocessing technique, and sound length through feature points.

2.1 Music Signal Preprocessing

Prior to the analysis and processing of music signals, they need to be preprocessed. Preprocessing is an important step prior to the early-stage processing of music signals, which will directly affect the precision and accuracy of the parameter extraction of music signal features. Music signal preprocessing involves determining sampling rate, windowing framing and pre-filtering.

^a 735661205@qq.com

2.1.1 Sampling Rate Determining

To take samples of music signals, appropriate sampling frequency should be chosen. The higher the sampling frequency is, i.e, the shorter the interval time of sampling is, the more sample data of music signals one will acquire, and the more accurate the waveform of music signals will be. However, more sampling data means greater sampling data size. For this reason, the data size to be calculated will greatly increase.

According to Nyquist's law, only when the sampling frequency is higher than or two times the highest frequency of music signals can data collecting of music signals be effectively done^[1]. It is defined as follows:

If $f(t)$ is a limited continuous signal of bandwidth, and its spectral maximum frequency is ω_m , then with sampling frequency

$$\omega_s \geq 2 \omega_m \quad (1)$$

$f_s(t)$ acquired through even sampling of $f(t)$ will include all the messages of the original signal $f(t)$.

The frequency of music signals lies between 20Hz and 20000Hz. Therefore, sampling frequencies are divided into three grades, i.e, 11k Hz, 22k Hz and 44k Hz. The fundamental frequencies of music signals are mainly distributed between 20Hz and 4200Hz^[2].

2.1.2 Windowing Framing

Short-time music signals may be intercepted by windowing framing the signals to make them stable in a short period of time so as to extract features with various digital signal processing algorithms. Generally, the window length range is 30ms-50ms^[3]. However, to guarantee the smooth transition between frames and the continuity of music signals, overlapping between two frames is required. The overlapping parts are what is called frame shift. The length of the window is the frame size. Generally, the frame shift is 1/3 to 1/2 the frame length. To intercept music signals, window function is multiplied by the music signal (n), so the music signals after windowing framing are

$$S_w = S(n) * \omega(n) \quad (2)$$

Rectangular window belongs to window function with power of zero. It is most widely used. Advantages

of rectangular window lie in its concentrated main lobe and fine smooth texture of spectrum. It has also some disadvantages in that it has side lobe with the greatest height and that the main lobe is featured by small width. These disadvantages lead to such phenomena as high-frequency interference and leakage in the course of conversion.

$$\omega(n) = 1, 0 \leq n \leq n-1 \quad (3)$$

N is the frame size of music signals. Different window functions will have different effects on characteristic parameter analysis of the music signals. Therefore, choice of window function should be based on the characteristics of the parameter of music signals, hence better substantive characteristics of music extraction.

2.1.3 Chebyshev Filter

Music signals are often affected by noise and disturbances. If filtering methods, such as arithmetic mean value, sliding average value, and anti-pulse jamming average value, may effectively filter out the disturbance caused by accidental pulse. However, these methods are less effective in filtering out the white noise of the environment or 50Hz power line interference of power supply. So far, Chebyshev filters have found wide application in such fields as picture processing, voice signal analysis and medical diagnosis. Chebyshev filters can recognize useful elements from the intricate sound signals and suppress disturbing signals in the environment. The frequency response of Chebyshev filters shows the characteristic equi-ripple and has fewer orders. Therefore, Chebyshev filters have good effect on filtering the environmental noise and disturbance in music signals. Type I Chebyshev filters are often used. Its definitional equation is^[4]:

$$T_n(w) = \begin{cases} \cos(n \arccos w) & |w| \leq 1 \\ \cosh(n \operatorname{arccosh} w) & |w| > 1 \end{cases} \quad (4)$$

In Formula 4, n stands for the order of Chebyshev polynomials while $T_n(w)$ is Chebyshev polynomials of n th order.

We use developed Android music mapping software and randomly select a piece of music, and acquire the extraction result of the preprocessed mus

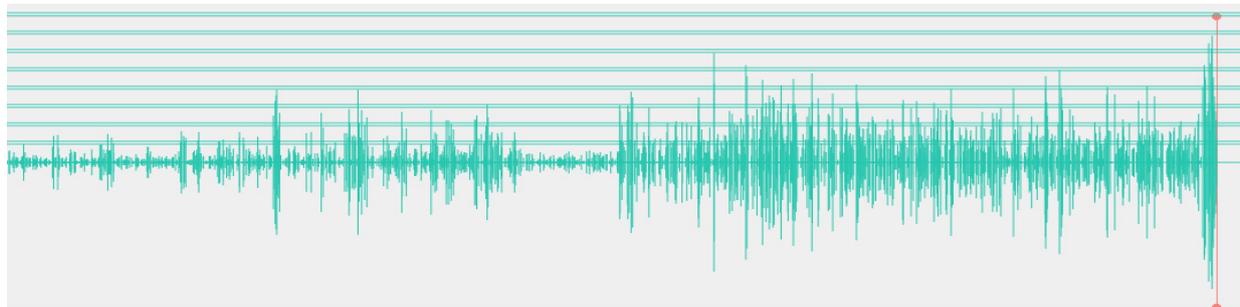


Fig. 1. Distribution of threshold value boundary areas

2.2 Acquisition of Feature Points [5-9]

To effectively acquire music feature points, the threshold value should cover high, medium and low amplitude ranges to the greatest extent. After many experiments on the music mapping software, the width of the threshold value boundary areas is 2, and the distance of the boundary areas is 20. They are equally spaced and add up to 8 as shown in Fig. 1.

2.3 Acquisition of Feature Points [5]-[9]

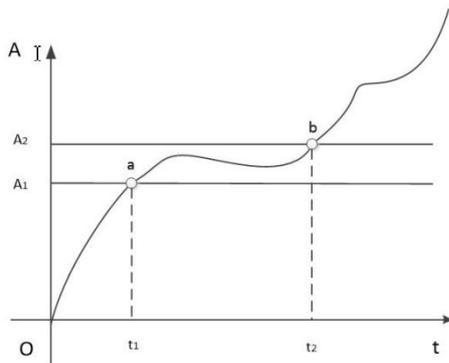


Fig. 2 Sketch Map of Feature Point Acquisition

By delimiting the threshold value, it aims to acquire music feature points. Threshold value appropriately selected, the feature points may be effectively acquired. As shown in Fig 2, the periphery of the threshold value boundary area intersects the music amplitude curve at a (A_1, t_1) and b (A_2, t_2). These two points are feature points, which satisfy this formula:

$$A_n = f(t_n), n=1,2,3... \quad (5)$$

In the formula, A is amplitude, t_n is time, and $f(t_n)$ is the amplitude function within the time domain.

Time duration may be acquired through Formula:

$$\Delta t_n = t_{n+1} - t_n \quad n=1,2,3... \quad (6)$$

In the formula, Δt_n is time duration, t_n is the time for entering threshold value boundary area, and t_{n+1} is the time for going out of the threshold value boundary area. Time duration in Fig 3 is:

$$\Delta t_1 = t_2 - t_1 \quad (7)$$

3 Robot Motion Matching

For music dance robots, it is required that the robot's movements can vary with the changing music. The music features should be matched to each other so as to coordinate various movements, hence the robot dancing trippingly to the music. This requires realizing the matching between music feature parameter and movements of the robot by extracting the characteristic parameters of the music signals.

3.1 CMovement Arrangement

To find out appropriate performing movements for dance robots according to music features, a basic performing movement database is needed, which facilitates calling different performing movement databases in different amplitude ranges. In the dance movement database of the robot, different movement names and their corresponding movement ranges and times give expression to different emotional connotations. For example, when the amplitude of feature points is high, rapid and vigorous movements are required to express passion. On the contrary, slow and continuous movements give expression to tenderness. Dance movement databases of different emotions are set up based on different amplitudes of feature points.

In accordance with the actual conditions, motor speed is set to be three units. 10 movement databases are set up as show in Table 1:

Table 1. Movement Database of Dance Robots

Range of Feature Points	Alternative Movement Database	speed	Time (ms)
Threshold Value Boundary Areas 1,2,3	M1	3	1600
	M2	3	3200
	M3	3	4800
Threshold Value Boundary Areas 4,5,6	M4	3	1600
	M5	3	2800
	M6	3	3900
	M7	3	4700
Threshold Value Boundary Areas 7,8	M8	3	5600
	M9	3	1600
	M10	3	2900

3.2 Acquisition of Movement Time

PAfter range of feature points is determined, movement time follows the following principles:

$$\Delta t_n = \Delta t_n, \Delta t_n > 1600ms \quad (8)$$

$$\Delta t_n = \Delta t_n + \Delta t_{n+1}, \Delta t_n < 1600ms \quad (9)$$

3.3 Movement Acquisition

After range of features points and movement time are determined, corresponding movement database should be selected in accordance with the principle of proximity. Times of repetition (N) for calling movement database conforms to:

$$N = \Delta t_n / T, (N \text{ is rounded}) \quad (10)$$

In the formula, T is the periodic time of the movement database.

Movement speed will become:

$$V = 3 * N * T / \Delta t_n \quad (11)$$

In the formula, 3 is the original speed, N is the times of repetition for calling the movement database, and T is the periodic time of movement database.

4 Experiments

To this end, an Arduino dance robot is set up ^{[10]-[14]}, and a great number of experiments have been done through this device. These experiments lead to extraordinary results. The dance robot is shown as in Fig 3.



Figure 3. Arduino Dance Robot

The above-mentioned method adopted, an executable file is generated via our mobile phone APP, and then the file is transplanted to the embedded device (model: RoboSoul H3P). With different threads and musics, a comparison is made between the traditional adaptive algorithm and ours, as shown in Table 2 and Table 3. It can be seen from the experiment that in the single thread, with the increase of music duration, the dancing robot system will freeze or even crash if traditional method is adopted, and a large amount of computer resources will be encroached on—an apparent disadvantage. Comparatively, if our method is adopted, the system will operate well despite a longer operational time. With the increase of thread, our method has its advantages in that it occupies fewer resources. In this way, the operational stability of the embedded device increases.

Table 2. A comparison of system operation between the traditional method and our method in the case of single thread and different music durations

Method	1 m	3m	5 m	10m
Tradition	smooth	smooth	freeze	freeze
New Detection	smooth	smooth	smooth	freeze

Table 3. A comparison of system operation between the traditional method and our method in the case of different threads and the same music

Method	Single thread	3 threads	Multi-thread
Tradition	smooth	Slight freeze	freeze
New Detection	smooth	smooth	smooth

Randomly select five out of the 100 musics to have a test as shown in Table 4. It is found that in the matching process of robot's movements, our method meets the requirement for movement coordination.

Table 4. A Test of movement coordination with our method adopted in the case of multi-thread and different musics

Song	Low amplitude	Medium amplitude	High amplitude
1	coordinated	coordinated	coordinated
2	coordinated	coordinated	coordinated
3	coordinated	coordinated	coordinated
4	coordinated	coordinated	coordinated
5	coordinated	coordinated	coordinated

5 Experiments

We put forward music adaptive method for dance robots. It is an efficient and reliable method. After music signal reprocessing, with this method adopted, reasonable range of threshold is set to acquire music features. The time of movement duration is acquired through transformation of features and coordinates. Then, alternative movements are selected, and movement speed is set up to realize the coordination of amplitude of variation and movement switch. So far, a great number of experiments on embedded device have been done. Compared with the traditional method, the method this paper put forwards helps to reach equally effective results. Moreover, the music features acquired are more succinct, hence greatly reducing the memory usage of the embedded device and enhancing its system performance.

Fund project

Foundation for Young Innovative Talents in Higher Education of Guangdong, China (No.2020KQNCX139, 2019KQNCX221); Foundation for Construction of Intelligent Scientific Research and Application Team (No. ST201903).

References

1. Zhao Li. Speech Signal Processing [M]. Beijing: China Machine Press. 2003.
2. Tian Hongwei. Application of Independent Component Analysis in Music Signal Processing [D]. [Master Thesis]. Tianjin: Tianjin University. 2009.
3. Chen Xiaolei. Extraction of Pitch Period, [J]. Popular science & technology. 2009(9): 47-49.
4. Rolf Schaumann, Haiqiao Xiao, Mac E.van Valkenburg, Analog Filter Design3nd Indian Edition, Oxford: Oxford University Press, 2013: 295-298.

5. Shahnaz, C. Zhu, W. P.Ahmad, M. O. A pitch extraction algorithm in noise based on temporal and spectral representations. 2008 Ieee International Conference on Acoustics, Speech and Signal Processing, Vols 1-12.
6. Kadambe, S.Boudreauxbartels, G. F. Application of the Wavelet Transform for Pitch Detection of Speech Signals.1992 Ieee Transactions on Information Theory .
7. Popescu, A.Gavat, I.Datcu, M. Wavelet Analysis for Audio Signals with Music Classification Applications. From Speech Processing to Spoken Language Technology.2009: 189-197.
8. George Tzanetakis, Georg Essl, Perry Cook. Audio Analysis using the Discrete Wavelet Transform. Computer Science Department. 2001.
9. Jianling Hu, Wireless Communications and Signal Processing (WCSP), 2010 International Conference on Real-time perceptual tempo estimation for music signal based on envelope autocorrelation.
10. Han Ben. Embedded IOT Systematic Design [J]. Scientific and Technological Innovation,2020(27).
11. Zhou Tong, Application of Embedded System in IOT Field [J].Electronic Technology & Software Engineering,2019(10).
12. Wang Guobin, Embedded Linux Communication Server Software Design [J].Electronic Technology & Software Engineering,2020(10).
13. Peng Jian, Liu Yansong & Tang Wen, Realization of MySQL Master and Slave Servers Database Synchronization [J]. Fujian Electronics Journals,2020(7).
14. Sun Lanlan, Strategy for the Realization of APP Development Based on Development Tools and API Interface of Android [J].Hei He University Daily,2018,9(5).