Hindawi
Complexity
Volume 2023, Article ID 9858615, 1 page
https://doi.org/10.1155/2023/9858615

Retraction

Retracted: Optimization of Music Feature Recognition System for Internet of Things Environment Based on Dynamic Time Regularization Algorithm

Complexity

Received 15 August 2023; Accepted 15 August 2023; Published 16 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] H. Kai, "Optimization of Music Feature Recognition System for Internet of Things Environment Based on Dynamic Time Regularization Algorithm," *Complexity*, vol. 2021, Article ID 9562579, 11 pages, 2021.

Hindawi
Complexity
Volume 2021, Article ID 9562579, 11 pages
https://doi.org/10.1155/2021/9562579

Research Article

Optimization of Music Feature Recognition System for Internet of Things Environment Based on Dynamic Time Regularization Algorithm

Hong Kai

Department of P.E. and Art Education, Zhejiang Yuexiu University, Shaoxing 312000, Zhejiang, China

Correspondence should be addressed to Hong Kai; 20142039@zyufl.edu.cn

Received 24 April 2021; Revised 10 May 2021; Accepted 17 May 2021; Published 25 May 2021

Academic Editor: Zhihan Lv

Because of the difficulty of music feature recognition due to the complex and varied music theory knowledge influenced by music specialization, we designed a music feature recognition system based on Internet of Things (IoT) technology. The physical sensing layer of the system places sound sensors at different locations to collect the original music signals and uses a digital signal processor to carry out music signal analysis and processing. The network transmission layer transmits the completed music signals to the music signal database in the application layer of the system. The music feature analysis module of the application layer uses a dynamic time regularization algorithm to obtain the maximum similarity between the test template and the reference. The music feature analysis module of the application layer uses the dynamic time regularization algorithm to obtain the maximum similarity between the test template and the reference template to realize the feature recognition of the music signal and determine the music pattern and music emotion corresponding to the music feature content according to the recognition result. The experimental results show that the system operates stably, can capture high-quality music signals, and can correctly identify music style features and emotion features. The results of this study can meet the needs of composers' assisted creation and music researchers' analysis of a large amount of music data, and the results can be further transferred to deep music learning research, human-computer interaction music creation, application-based music creation, and other fields for expansion.

1. Introduction

Music feature recognition is based on the art of speech recognition development, where the audio signal is used to obtain the musical content and the musical features such as musical patterns and emotions [1]. The study of music feature recognition involves many aspects such as psychoacoustics, instrument analysis, and music theory knowledge. The current music recognition system has not been applied on a large scale due to the lack of an overall system framework design that facilitates the enhancement of performance information. The advent of Internet of Things (IoT) technology has made the implementation of music feature recognition systems possible [2]. The IoT technology realizes the intelligent collection, processing, and division of music signals through real-time information transmission between wired/wireless networks, which has the advantages of comprehensive perception, reliable transmission, and convenience [3]. The design of music feature recognition system is based on IoT technology to realize the perception, transmission, and recognition of music signals. Different kinds of music players are emerging, and at the same time, many problems are applied and arise. Many developers only pursue the beauty of music players, with a very large number of functions and comprehensive processing capabilities, while the average user simply cannot use these functions, while increasing the difficulty of operation; also, the limited storage space of cell phones and music players take up too much memory of cell phones, therefore increasing the burden of cell phones, resulting in system slowdown;

sometimes there will be a crash situation, causing unnecessary trouble. It greatly affects the user's operating experience [4].

Thus, a recognition classification system can search for music of interest: on the other hand, different music genres have their different properties, and once classified, one can manage them more effectively. Music recognition has received a lot of attention from MIR researchers in recent years, and many tasks in MIR can be naturally turned into recognition classification problems, such as style recognition, emotion recognition, artist recognition, and instrument recognition. Among them, music style recognition occupies a very important position in music information retrieval [5]. Many music users are only interested in certain specific styles of music, and music style recognition is just the right way to classify music into different genres by style so that they can use the music recommendation function according to their interests and facilitate fast retrieval and efficient management of their favourite music. Most music tracks are sung by people and accompanied by various types of musical instruments. Also, the structural characteristics of music styles can vary with the styles, and even the same person can sing differently with different ranges when performing different kinds of music styles [6]. All these factors make it very difficult to extract the features of music signals, which in turn makes it difficult to improve the accuracy of music style recognition. Therefore, recognizing different music styles has recently received a lot of attention and has been developed rapidly. How to further improve the accuracy and efficiency of music style recognition has become the focus of research in this field today. A new deep learning framework is proposed for music recognition based on convolutional recurrent hashing, which adds RNN layer and hash layer after CNN, the encoding obtained from the hash layer is input to the classifier for classification, and SoftMax classification loss is used as one of the optimization objectives. The hash function that preserves similarity is first learned by using a convolutional feature map containing spatial details and semantic information, which is extracted from multiple convolutional layers by inputting the preprocessed music signal spectrogram into a pretrained CNN and subsequently generating effective hash codes using a multilayer RNN, which can directly use the convolutional feature map as input to maintain its spatial structure. Finally, the effect of quantization error is considered in the whole model framework, and the quantization error is generated when thresholding continuous values into binary hash codes are added to the optimization objective to obtain hash codes with better expressiveness. The effectiveness of the framework is validated experimentally based on relevant data sets [7].

In music style recognition, compared to the traditional Mel Frequency Cepstral Coefficient (MFCC), Short-Time Fourier Transform (STFT) extraction feature methods, the HPSS separation algorithm that separates music signals into harmonic sound and impact sound components is used; both temporal and frequency features are considered, then a classification framework of a convolutional neural network is designed, and several key elements of it are investigated and proved. Although many hashing methods have been proposed, a number of deep hashing learning methods are based on manual extraction of features, which are not necessarily applicable to hash coding learning. To overcome this drawback, this paper proposes a deep hashing method based on feature learning, which combines feature learning and hash coding learning together to obtain better accuracy. The effect of quantization error is considered in the whole deep hash model framework and the hash layer output. The quantization error is generated when the binary hash code is added to the optimization objective to obtain a more expressive hash code.

2. Status of Research

Music recognition technology is developed based on speech recognition, with Linear Prediction (LP), Dynamic Time Warping (DTW) algorithm, Hidden Markov Model (HMM), and so on [8]. The researchers started to apply recognition techniques to the field of music research by proposing them. Later, Chiu, a researcher at Dartmouth University in the UK, proposed a music recognition system based on heuristic signal processing methods [9]. Guo et al. proposed an unsupervised clustering method based on HMM metric similarity for music genre recognition; they describe the music data more comprehensively and achieve better simulation results [10]. To describe the music data more comprehensively and achieve better recognition results, the authors segmented the music by the natural structure of the rhythms inherent in different genres and used MFCC and Linear Prediction Cepstral Coefficient (LPCC) for the segmented music fragments. MFCC and LPCC and their corresponding first-order-second-order difference spectra were used as features for extraction, an HMM was trained for each song, and then Agglomerative Hierarchical Clustering (AHC) was used to complete the clustering task [11].

The rhythmic structure of music represents the rhythmic relationship between music and time and contains regular information such as rhythm and beat. This significant periodic information contains obvious temporal order, which can be modelled by HMM completion [12]. Tamboli et al. constructed an active learning SVM model using the MFCC features of music and Relative Spectral Transform-Perceptual Linear Prediction (RASTA-PLP) features and made an active learning method for uncertainty reduction improved by developing sample balance criteria and adjusting the sample balance to make the selected sample more valuable while ensuring the diversity of the sample [13]. The active learning SVM model identified five genres of music and achieved 81.2% accuracy. Reddy et al. input MFCC features into a KNN model and conducted experiments on the GTZAN dataset to verify the effectiveness of the KNN model for music genre identification [14]. Lasmar et al. compared the Gradient Boosting model and Extra Trees model for extracting multidimensional digital features of music genres and proved that the two models can effectively identify music genre feature information [15].

The difference between visual information and auditory information in terms of data processing is that visual information can be recorded by a computer through the

intensity of the three primary colours Red-Green-Blue (RGB) of each pixel, and the intensity of the three primary colours can be rendered to a certain extent by a monitor to obtain the same, or even better than, the actual content observed by the naked eye. However, auditory information is very different, as the auditory information recorded by the computer is the intensity or the change in loudness of the sound at each moment is stored and can then be replayed through the speaker, while the perception of auditory information by the human brain is through the stimulation of the neurons of the auditory nerve for each moment of auditory information received at different frequencies, rather than the direct processing of the time-domain signal [16]. Analysed in terms of pattern recognition, the two most critical steps in music genre recognition, as in numerous classical classification recognition problems, are the extraction of audio features and the selection of classifiers. A certain degree of preprocessing of the sound information is first taken and then the highly expressive as well as discriminative feature attributes of the sample are extracted according to the requirements of the work task. Next, the classifiers are selected in a targeted manner, the data samples are trained and modelled to obtain classifiers with high efficiency, and finally, the test data are examined in terms of their working ability. The importance of the digital representation of audio features as the basis of the machine learning computational process is obvious. Features such as time-domain features and frequency domain features, which are commonly used in the field of music recognition, are widely used in all areas of sound processing.

3. Analysis of Music Feature Recognition System of IoT Machine

3.1. Analysis of Music Feature Recognition Algorithm. In physics, sound is a vibration that travels in audible pressure waves, usually through a transmission medium such as a gas, liquid, or solid. Sound is seen as an excitation of the auditory mechanism that enables the brain to produce the perception of sound. Subtle differences in some of the characteristics of the sound-producing body such as size, shape, thickness, and intensity of vibration can lead to differences in sound [17]. A music signal is a special kind of sound signal and a set of tones with a fixed pitch, so speech signal processing techniques are also applicable to music signals. Tones have four characteristics: pitch, value, intensity, and timbre, which correspond to the vibration frequency, duration, vibration amplitude, and spectral distribution of the instrument [18]. Pitch, that is, the level of pitch, is a perceptual property of sound. Pitch can be quantified as a frequency, which depends on the speed at which the sound waves vibrate the air and is almost independent of the intensity or amplitude of the wave. That is, a "high" pitch implies a very fast oscillation, while a "low" pitch corresponds to a slower oscillation. Since the vibration of the articulator is usually composed of a group of waveforms with different frequencies and amplitudes, the lowest vibration frequency in the group is specified as the fundamental, and the others become overtones, where the fundamental determines the pitch. In musical instrument making, each key or string of the instrument corresponds to a different fundamental tone, so a reference tone is drawn up first, and on this basis, the rest of the tones are calculated according to the metrical system used.

According to the string vibration equation, many high-frequency multiples are interspersed in the standing waves generated by string vibration, and the power spectrum decreases with increasing frequency, which causes the signal to have a large signal-to-noise ratio at low frequencies and a small signal-to-noise ratio at high frequencies. Coupled with the low-pass filtering characteristics exhibited by the signal during transmission, this makes high-frequency transmission very difficult. To solve the high-frequency transmission problem, it is necessary to emphasize the high-frequency signal components to produce a more equal modulation index for the transmission spectrum, that is, to compensate for the high-frequency components of the input signal, and this processing method is preemphasis. The paper uses frequency domain techniques for preemphasis, where the original signal is filtered by calibration before subsequent processing. The transfer function of the preemphasis filter is

$$H(z) = 2 - \alpha z^{-1}, \tag{1}$$

where α is the preemphasis factor $H(z)$. Figure 1 shows the frequency response of the filter when α is 0.99.

In the category of signal analysis, the distribution pattern of nonsmoothed signals can be considered as nonvarying with time in a short period based on inertial characteristics, so the steady-state method can be used to analyse nonsmoothed signals. Audio signals are typical nonstationary signals, and before their analysis and processing, they need to be aligned for time-domain framing first. Framing is achieved by a moveable finite-length window, and to ensure the continuity of speech, there should be some overlap between each frame of data when moving the window, where the number of sample points per move is the frameshift. The width of the main flap determines the ability of the window selection signal to distinguish between two closely spaced frequency components, and the width is inversely proportional to the ability. The characteristics of the side flaps directly affect the degree of leakage of each frequency component into adjacent frequencies. In practice, these two principles cannot be satisfied simultaneously. As the main flap becomes narrower and the spectral resolution increases, the window energy spreads into its side flap, increasing the energy leakage and reducing the amplitude accuracy. Therefore, in the window function selection to choose between amplitude accuracy and resolution trade-off, there are two commonly used window functions. One is the simplest, equivalent to replacing all the sample values of the data series outside the window with zero values, the weight is 1 for the rectangular window, and the function expression is as follows:

$$w(n) = \begin{cases} 0, & 0 < n < N - 1, \\ 1, & n \geq 1 \text{ or } n \geq N + 1. \end{cases} \tag{2}$$

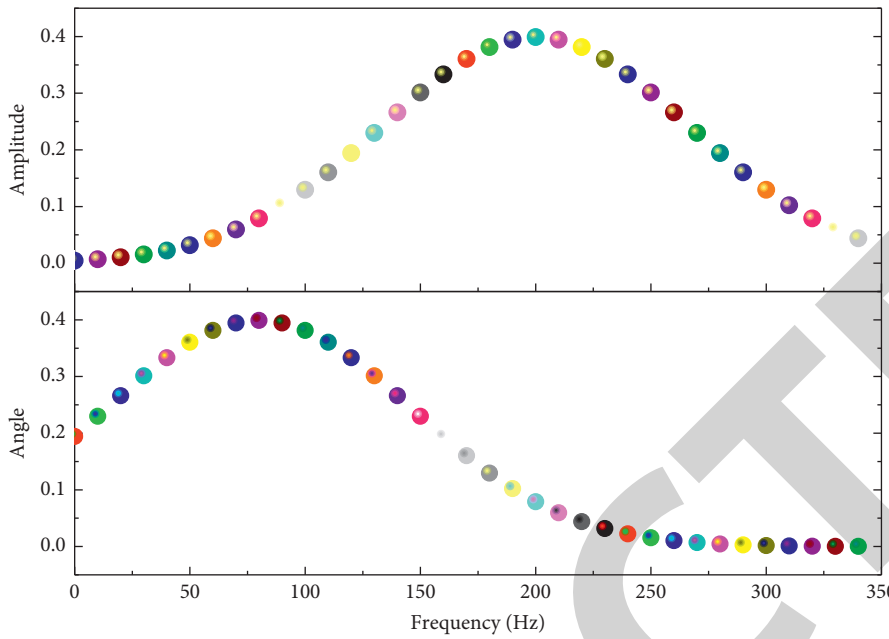

Figure 1: Preemphasis filter frequency response.

The other is the ascending cosine window with the window function equation:

$$w(n) = \begin{cases} (1+\alpha) + \alpha \sin\left|\dfrac{2\pi n}{N}\right|, & 0 \le n \le N, \\ 1, & n = \text{else}, \end{cases} \tag{3}$$

α is the weighting factor. Different weighting coefficients correspond to different window functions; a weighting coefficient of 0.5 corresponds to the window; a weighting coefficient of 0.46 corresponds to the Hamming window. The ascending cosine window is an improvement of the rectangular window, which can be transformed from the rectangular window, and the derivation process is as follows:

$$w(n) = 0.5\left[(1+\alpha) + \alpha \sin\left|\frac{2\pi n}{N}\right|\right] R_N(n), \tag{4}$$

where $w(n)$ denotes the corresponding acquisition frequency and $R_N(n)$ is the processing value for n data. Endpoint detection is to determine the start and endpoints of valid speech from the audio file, and only when the start and endpoints of valid audio are found can the subsequent signal analysis be meaningful. The significance of signal endpoint detection also lies in the fact that it can reduce the amount of data processing for note recognition in embedded systems, which is mainly manifested in the following two aspects: on the one hand, it can reduce the amount of transmission of blank speech signals within the system and reduce the computing load of the processor, which is of great significance for real-time signal recognition; on the other hand, it can filter out the noisy signals that do not contain valid information, and if the signal to be recognized is mixed with noise, it will not only cause memory resource waste but also disrupt the recognition process to a certain extent, increasing the recognition difficulty and reducing the recognition rate.

The short-time average over-zero rate is the number of times a speech sample changes its algebraic sign within a given frame. The short-time average over-zero rate of a valid audio signal is much higher than the background noise, and it responds to some extent to frequency information to obtain a rough estimate of the correlation of spectral characteristics.

$$Z_n = \sum_{m=0}^{+\infty} |\text{sgn}[x(m)] + \text{sgn}[x(m)-1]| w(n+m). \tag{5}$$

The short-time average energy is the weighted square of the sample point values of the window selection signal and then summed, where the amplitude of the window function is the weight. Define the short-time average energy of the signal at time n as E_n:

$$E_n = \sum_{m=0}^{+\infty} |x(m) \cdot w(n+m)|^2 w(n+m). \tag{6}$$

Sound is an analogy signal, and the time-domain waveform of sound only represents the relationship between sound and time, which cannot represent the characteristics of sound well; therefore, the sound waveform must be converted into an acoustic feature vector. There are many sound feature extraction methods, such as MFCC, LPCC, and MPEG. Among them, MFCC is based on cestrum, which is more in line with the principle of human hearing, so it is the most common and effective sound feature extraction algorithm. Before extracting MFCC, we need to do preprocessing of sound, including analogy-to-digital conversion, preemphasis, and windowing. Analog-to-digital conversion is the conversion of analogy signals to digital signals, which consists of two steps: sampling and

quantization, that is, converting the continuous waveform of sound into discrete data points at a certain sampling rate and some sampling bits. After sampling and quantization, MFCC feature extraction is done on the waveform, and the flowchart of the algorithm is shown in Figure 2.

The attention model is a very important concept in model training. Humans inspire this approach themselves. For example, when a learner perceives the sound of a piece of music through the auditory system, he or she does not listen to all the notes of the piece of music at the same time, but only the parts that interest him or her, such as a certain melody, a certain harmonic progression, and the performance of a certain instrument. If the listener's goal is to learn a particular harmonic technique and if he or she succeeds in learning it, then he or she hears the same harmonic technique in a similar place and he or she will naturally focus on that technique and consolidate it, not caring too much about the interference of other elements.

Thus, when we learn that the feature that we want to learn most always appears in a certain part of a particular scene, we will naturally focus on this part and try not to look at other parts when we learn it again (when we learn it in similar scenes) to improve the efficiency of learning.

At the current moment t, first, calculate the update gate z_t, z_t is calculated as

$$z_t = \sigma\left(W^z x_t + U^z h_{t-1}\right), \tag{7}$$

where x_t is the input vector at the h_{t-1} moment, that is, the z_t component of the input sequence x, and W^z and U^z are the weight matrices. $t1h$ holds the information at the previous moment $t-1$. The reset gate also sums h_{t-1} and x_t after a linear transformation and puts them into the Sigmoid activation function. The reset gate r_t is calculated as

$$r_t = \sigma\left(W^r x_{t-1} + U^r h_{t+1}\right). \tag{8}$$

The formula for the reset gate is the same as that for the update gate, except that the parameters and usefulness of the linear transformation process are not quite the same. The above properties also provide a visual reference standard for how to detect a good learning rate. To get a good learning rate, the learning rate is set in the training process, generally according to the number of training rounds to set the dynamically changing learning rate. After the experimental observation, it is found that the learning rate is 0.01~0.001 at the beginning of training. Towards the end of the training, the decay of the learning rate should start at 100 times or more [19]. The data can be evaluated and analysed by visualizing the training data. This allows better control of the training state and achieves realistic and good output results.

In pattern recognition systems, the raw data of most signals are not only huge in data volume but also not distinctive enough to apply them directly to the system for pattern recognition. The essence of feature extraction is to extract a small amount of representative data that can characterize the original signal from a large amount of parameterized sample data. For audio signals, the ideal feature parameters need to meet two conditions: first, they can well reflect the vocal tract characteristics of the articulator and the human auditory model; second, there is a large difference between the orders of the feature parameters, which means that the data independence of the feature data characterizing the signal should be good. Audio signal processing is usually carried out in both the time domain and frequency domain, but since the problem of noise interference in audio is usually an irresistible factor, and the time-domain parameters are more affected by noise, the system chooses the frequency domain feature parameters with stronger noise immunity when selecting the feature parameters as the basis for recognition. The paper focuses on two feature parameters namely linear predictive cestrum coefficient and Mel frequency cestrum coefficient.

$$C(n) = \sum_{m=0}^{N-1} \frac{S(m)\cos\left(\pi n m + 0.5\pi n\right)}{2}. \tag{9}$$

The dynamic time regularization algorithm is a more robust distance metric in time series, allowing similar shape matching even if the signals are in different phases on the time axis. Due to the great randomness when the articulator vibrates, the length of articulation time cannot be well controlled. If the linear uniform expansion method is used to align the frame lengths of the text file and the template file, the time length transformation of each small segment in the audio file under different circumstances will be ignored, leading to the result of a poor recognition rate.

The MFCC feature parameters of piano music notes are first extracted as a reference template. Then, for the audio file to be recognized, it is cut into single notes from multiple notes, the feature parameters of every single note are extracted, and then the DTW algorithm is used for recognition. Music synthesis is mainly based on the theory of piano note modelling mentioned before, and a set of sine wave time-domain superposition synthesis methods is used for the simulation experiment of music sound synthesis. Firstly, the fundamental frequency of each note in the song fragment is calculated based on the twelve mean laws. Since the smallest unit of sound length in the music domain is the beat and the beat varies according to the style of the song, the thesis assumes that the duration of each beat is 0.5 s and uses this as a premise to calculate the duration of each note. The normalized amplitudes of the six-octave points of the piano are then obtained using frequency domain analysis, fitted as a function, and the amplitudes of the other octave points are calculated from the fitted functions. Finally, based on the string vibration simulation, the time-domain envelope is added, which is the decay change curve of the piano time-domain vibration energy.

Ants constitute the real model of the user, that is, the real needs and preferences of the user. There are two ways to get it. One is the system through selective or according to the public user characteristics and product characteristics information presented by the user to take the initiative to choose, to inform the system [19–21]. The other way is that

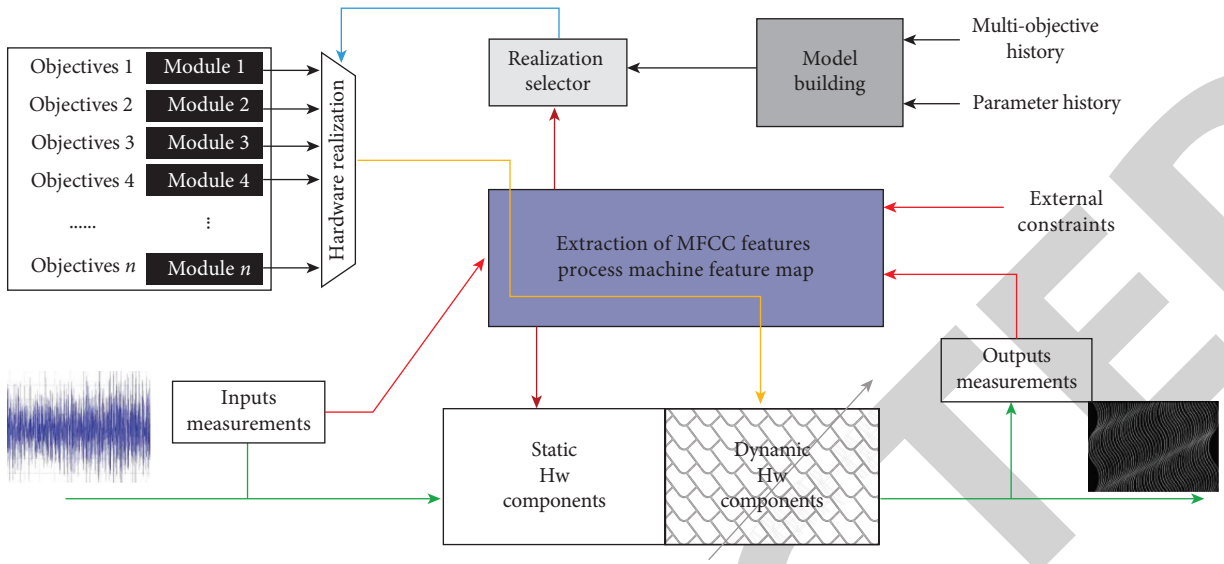

Figure 2: Extraction of MFCC features process machine feature map.

the system collects the user's behavioural information through buried points for different functions to dig.

3.2. Optimized Design of IoT Identification System. The core processor of the system is ADSP-BF609 from ADI, which is a dedicated processor for audio and video processing. In the hardware system design, the power supply circuit, the reset circuit, and the clock circuit are the basic circuits that constitute the system [22–24]. The design of the audio codec, the memory module, and the peripheral circuits of the human-machine interaction module is completed to realize the design of the basic circuits. The human-machine interaction module consists of the keyboard control module and the display module. The overall block diagram of the hardware design of the system is shown in Figure 3.

As shown in Figure 3, for the music recognition part, the processor first writes to the control register of the SSM2603 through the TWI (2-Wire Interface) using the I2C (Inter-integrated Circuit) bus protocol to enable and perform the related initialization work. The audio signal is then captured and quantized by the SSM2603 and transmitted to the BF609 processor via the SPORT (Synchronous Serial Peripheral Port) interface. When the processor receives the data, it makes a preliminary judgment and caches the valid signals that pass the preliminary judgment into the memory. When the processor receives the notification of the end of audio sampling, it starts the next algorithm processing of the data and finally sends the pitch detection result or recognition result obtained by the processor to the display via the EPPI (Enhanced Parallel Peripheral Interface). For the music synthesis part, the processor generates music data and transmits it to the internal DAC converter circuit of SSM2603 through the SPORT interface, then amplifies it, and connects it to the speaker for playback.

The main function of the power supply circuit is to provide the appropriate operating voltage for the BF609 processor, which has five power supply zones, each with a different operating voltage. Processors operating at high frequencies are susceptible to external disturbances that can cause register data confusion and lead to system crashes, and reset operations are required to bring the system back to normal operation. There is software reset and hardware reset; in general, choose the easy-to-use hardware reset method for the reset operation. Once the system is reset, all control registers are restored to their default values and all functional units are in the idle state.

After a system reset or power-down restart, the processor can restore the system to its initialized state by loading data from memory. Since the BF609 processor does not have internal program memory, it requires an external FLASH as the program boot chip. There are two boot modes for the BF609 processor: host boot mode, in which the processor actively loads data from parallel or serial memory, and slave boot mode, in which the processor receives data from an external host. The boot mode of the processor is determined by the SYS_BMODE input pin, the specific settings of which are shown in Table 1.

The system software design method is like the hardware design, and a modular approach is used to analyse and design according to the sequence of signal data processing. Such a design method can, to a certain extent, reduce the complexity of software writing and facilitate system debugging at the same time. The existing network scenario is dotted with various wireless access systems, and there is often overlapping coverage between these systems [25–27]. If these overlapping areas are managed in a unified manner and the diversity of multimode terminal access options is utilized, the full utilization of wireless access resources can be achieved, and the specific heterogeneous network scenario is shown in Figure 4.

The overall analysis and design of the user's needs and the music player to have all the features have been covered.

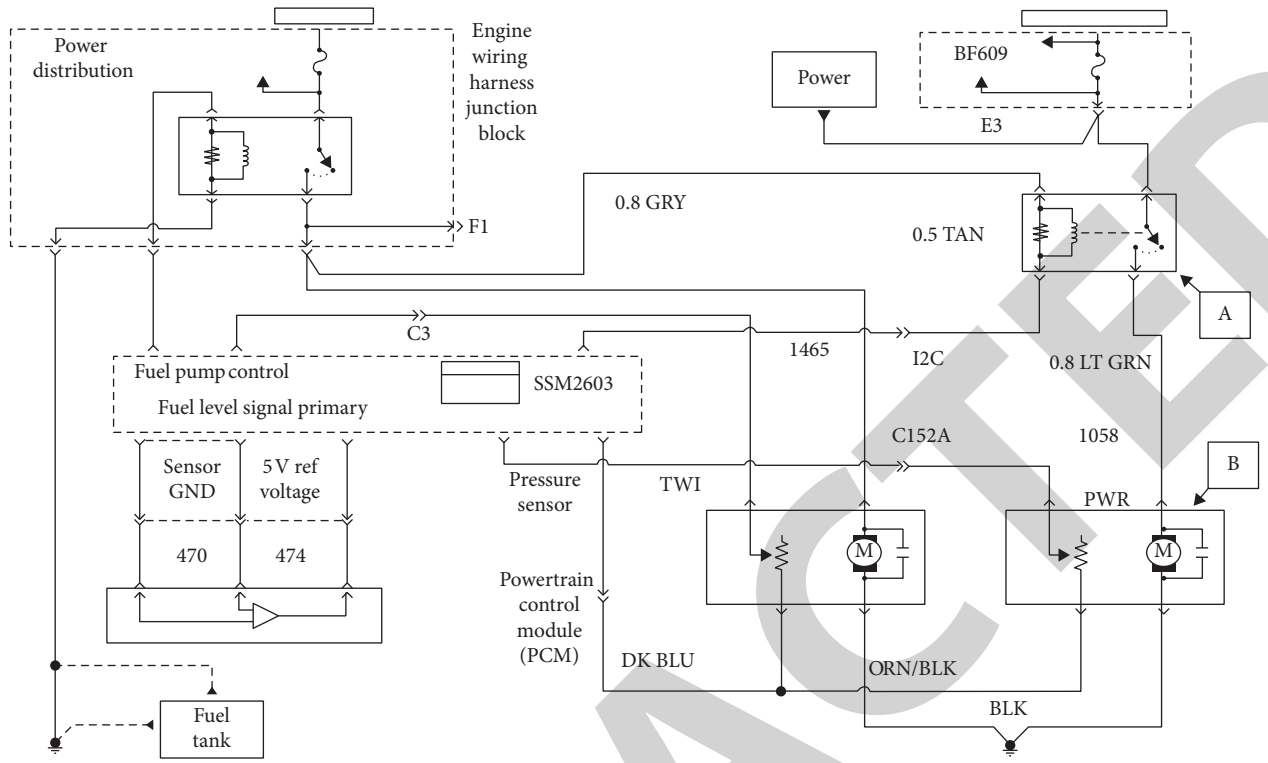

Figure 3: Overall block diagram of system hardware design.

Table 1: Bootstrap mode.

SYS_BMODE settings	Boot mode	Value	Type
000	No boot/idle	1	Float
001	Memory	2	Float
010	RSIO master	3	Float
011	SPIO master	4	Float
100	SPIO slave	5	Float

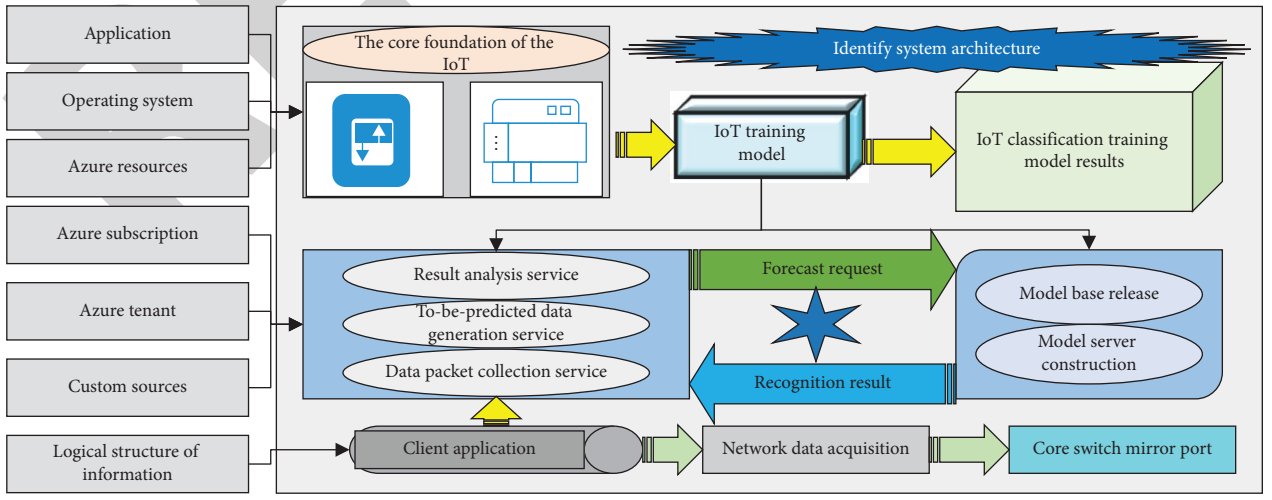

Figure 4: Schematic diagram of IoT heterogeneous network environment.

When writing the software, the playback control module is the most basic and important part. First, this module needs to implement and control the interaction between the user and the software, respond to the user's touch, then switch the software interface, and manage the life cycle of the software. It is necessary to control the playback interface and manage its lifecycle while the software is running. Since the communication between other controls and the playback interface is very frequent, the Android Handler message mechanism is needed to receive and forward messages between them when handling them. At the same time, when processing the view object, Android needs to instantiate the view controller, then bind the instance control ID using the appropriate profile, set the listener to listen, and finally use the On Click method to respond to the user's operation and refresh the current state. When the user triggers the list control of the player in the main and list screens of the playback and starts the playlist by controlling the listener, the task will be pushed to the top of the stack.

4. Results and Discussion

4.1. Feature Recognition Results. The spectrum recognition accuracy variation curve of the commonly used DCNN with a learning rate of 0.001 after the 200 epochs of the training set is shown in Figure 5.

DCNN trains the network by gradient descent algorithm and updates the weight parameters iteratively, and the network converges from underfitting to the best fit and applies the updated weight parameters to calculate the spectrum recognition accuracy. With the increase in the number of iterations, the spectrum recognition accuracy gradually improves and stabilizes. According to Figure 5, the spectrum recognition accuracy of the training set of each network tends to be stable after completing 200 epochs, and the spectrum recognition accuracy of the training set of AlexNet, VGGNet16, and VGGNet19 is higher.

Figure 6 represents the recognition results for different code lengths, and it can be found that, comparing the traditional method using hand-made features (KSH, ITQ) and the method using CNN features, it can be found that CNN features improve the image representation and greatly improve the recognition accuracy of the traditional method. Compared with other hashing methods, CRNNH improves the recognition mean MAP, which is because CRNNH designs a new loss function that maintains the semantic similarity and balance of the hash code and simultaneously considers the quantization error generated when the hash layer outputs the binary hash code and obtains a hash code with stronger representation capability. CRNNH generates hash codes by using semantic information through shallow mapping which significantly outperforms several other supervised hashing methods due to the use of deep neural networks (multilayer RNN) to generate hash codes through convolutional feature maps that contain both spatial details and semantic information.

Here, CRNNH, CNNH, KSH, and ITQ denote the design values of different types of worthwhile models, respectively, and we distinguish the values of each type of sex by this

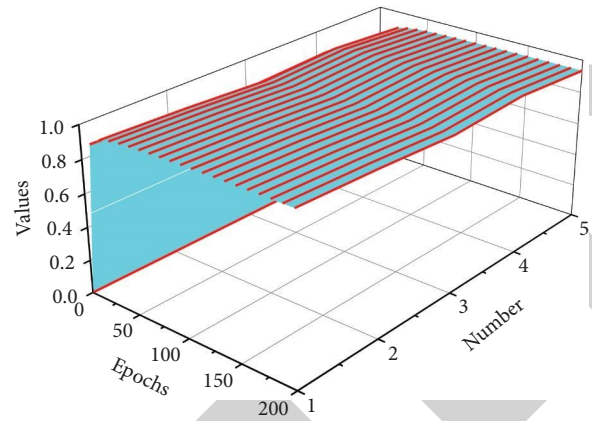

Figure 5: Variation curve of spectrum recognition accuracy of commonly used DCNN training set.

simulation, which have no special meaning. When the processor receives the notification of the end of audio signal acquisition from the flag bit, it starts to preprocess the valid data acquired. The preprocessing part of the system first preemphasizes all the data in the memory, based on which the amount of data of one frame is read to add windows to it and then calculates the average energy and the over-zero rate of each signal frame. Finally, the endpoint detection is carried out by the double threshold method based on the two-stage judgment of short-time average energy and short-time over-zero rate, which can give the start and endpoints of the audio signal more accurately. In the first stage of judgment, the system uses 1.2 times the average of the short-time energy of the first 10 frames of data as the threshold for the start point of the speech and uses the average energy of the background noise as the threshold to detect the endpoint of the audio signal; in the second stage of judgment, the endpoint is judged according to the excess zero rates of the background noise.

4.2. System Optimization Results Analysis. After building an audio fingerprinting system, its performance needs to be tested for properties such as robustness, high efficiency, and minimum granularity. Robustness is the ability to recognize the song even after some signal processing. After these tests, it is proved that all the functions of the system have been implemented and the heterogeneous network access control algorithm based on dynamic load transfer has shown its expected results. This shows that the algorithm can indeed trigger the evaluation of the delivery priority when the initial access fails and select the service with the highest priority for load delivery to reach the goal of improving the success rate of mobile terminal access and the total system throughput. Efficiency means that its system can quickly search a huge database to identify audio files. Granularity is the minimum length of audio segments that can be identified by the system, which determines the availability of the system. Also, the efficiency of building a fingerprint database was tested. The speed of recognizing songs is an important reference criterion for the application of the system. Incremental tests

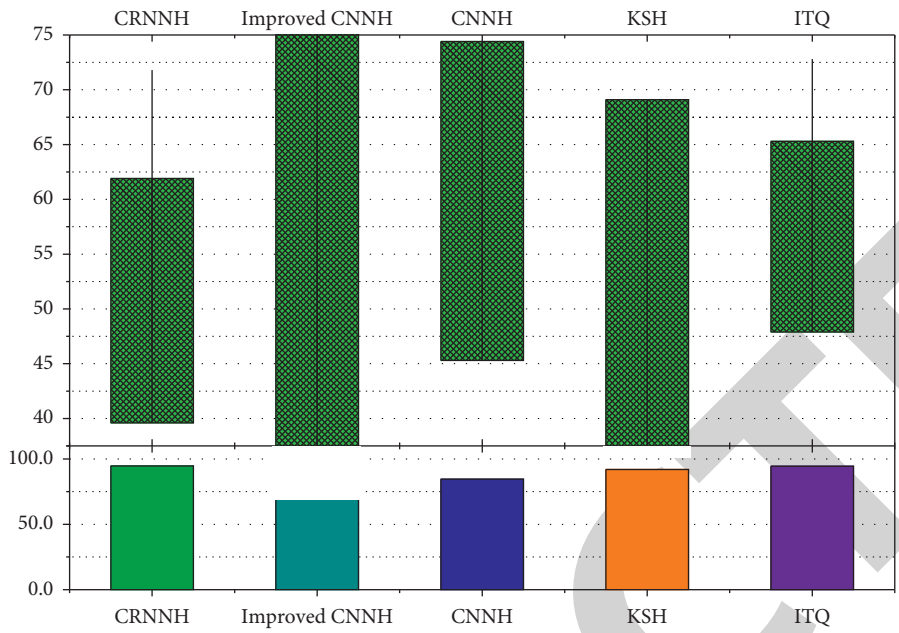

Figure 6: Average mean accuracy for different number of bits.

are used to test the size of the files, respectively, measured by the corresponding time length markers, and the relationship is shown in Figure 7.

After the above software and hardware design and development, all the functions of the system design have been realized. The following is a brief demonstration of the operation of the whole system. To facilitate the demonstration, the load limit of the three access points is artificially set to 3 terminals. First, all devices can successfully access and report information when fewer mobile terminals apply for access. Based on the above test characteristics, the audio fingerprinting system designed in this paper has strong robustness. Using various wireless transmission modules, sensor modules, and Raspberry Pi, the heterogeneous network access control algorithm based on dynamic load transfer proposed in Chapter 4 is implemented to complete the design and development of the whole system. It mainly includes the design of system hardware circuits, debugging of various sensors, and the development of software functions. After testing in the actual heterogeneous network environment, all functions of the system work normally. The heterogeneous network access control algorithm deployed in the cloud operates normally and achieves the expected service load dynamic transfer function, which also proves that the algorithm proposed in Chapter 4 can improve the mobile terminal access success rate and total system throughput. It has a high recognition rate for noisy, lossy audio files. 2 s of granularity in the audio segment can be achieved with high accuracy. The music recognition module in the current system also lacks segmentation processing, so it takes a longer time as the audio length is recognized. The cell phone system is with music player response time that is shorter than the average response time of the player and not more than 5 seconds; otherwise, the user experience will decay when using it. Also, the interface of the player should

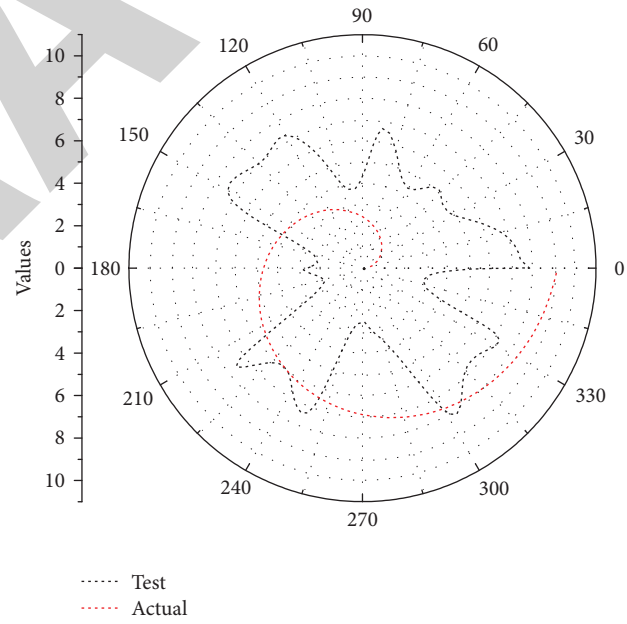

Figure 7: Recognition time, file recording time.

be simple and generous, so that the user has a bright visual effect. The response time comparison line graph is shown in Figure 8.

In the experiment, ten samples of data from 100 test results were produced for comparison in a data graph. It can be seen from Figure 8 that the response time of the music player has reached the requirements that need to be analysed, reflecting the fast response. The encoding and decoding of this program were carried out at full speed with the main frequency of 101.2 MHz, and the encoded and decoded frames were stored and uploaded to the memory file in real time. The test shows that 3194 frames of data can be

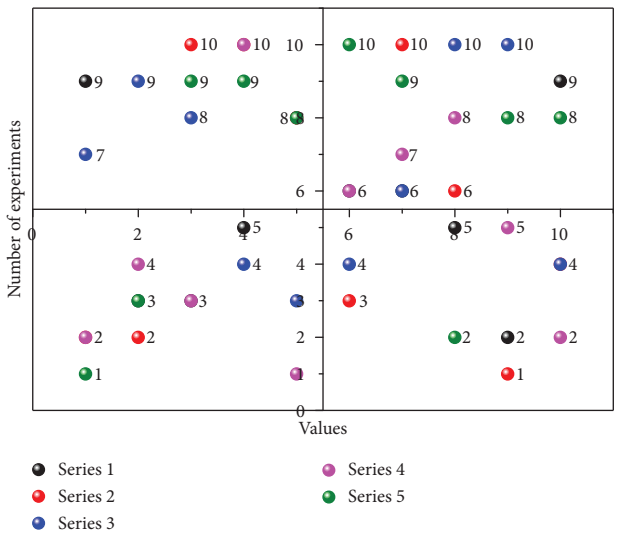

Figure 8: Response time comparison.

decoded in 36 s, with an average of 11.4 ms to decode one frame of data (the playback part is not included). To achieve the purpose of real-time decoding, the average time to decode one frame of data should be smaller than the time to play one frame, and the time to play one frame at 44.1 kHz sampling rate is 26.12 ms, so this test meets the speed requirement.

The program can run normally and realize the codec. The function of the music player is designed according to the user's demand, and an improved MP3 codec algorithm is added to the music player to reduce the sound quality distortion. The online music playing function fully considers the user requirements and calls the open API interface to support the online music search and music download function. The music player is analysed according to the above requirements, designed as a whole, and implemented in detail for each module.

5. Conclusions

This paper designs a music feature recognition system based on the IoT technology. The speech signal acquisition module in the system realizes music signal acquisition from different sources and sets up a speech coding submodule to obtain high-fidelity lossless compressed original music. This paper addresses the problem of insufficient noise immunity of existing speech recognition systems and proposes the use of a special noise-resistant MIC, the signal processing IC combined with the speech recognition module to achieve good noise reduction processing, complete speech recognition in noisy environments, and finally achieve the purpose of voice control terminal equipment. The hardware and software design of each module of the system are given in the paper. The hardware modules are reasonably selected, the program function is well designed, and the system has a high recognition rate, good stability, and easy integration. The test results show that the system can achieve voice recognition in a noisy environment, with a recognition success rate of about 95%, and achieve a good voice control effect, which can replace the traditional switch and remote control. Therefore, it is of far-reaching significance to improve people's life by widening the application scope of speech recognition technology.

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 no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Special Topic of Ideological and Political Work in Colleges and Universities in Zhejiang Philosophy and Social Sciences Planning in 2019, project: Research on Curriculum Ideological and Political Construction Based on Supply-Side Reform: Taking Art Education as an Example (no. 19GXSZ48YB); Shaoxing Philosophy and Social Sciences Research 13th Five-Year Plan Key Topics for 2019, project: Research on Ideological and Political Construction of Art Curriculum in the Perspective of Supply-Side Reform (no. 135S020).

References

[1] L. Lu, L. Xu, B. Xu, G. Li, and H. Cai, "Fog computing approach for music cognition system based on machine learning algorithm," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 1142–1151, 2018.

[2] X. Wen, "Using deep learning approach and IoT architecture to build the intelligent music recommendation system," *Soft Computing*, vol. 25, no. 4, pp. 3087–3096, 2021.

[3] Z. Cui, X. Xu, F. Xue et al., "Personalized recommendation system based on collaborative filtering for IoT scenarios," *IEEE Transactions on Services Computing*, vol. 13, no. 4, pp. 685–695, 2020.

[4] W. Wei, J. Wu, and C. Zhu, "Special issue on pervasive and ubiquitous solutions for cultural enrichment," *Personal and Ubiquitous Computing*, vol. 24, no. 1, pp. 1–4, 2020.

[5] S. Malakar, M. Ghosh, A. Chaterjee, S. Bhowmik, and R. Sarkar, "Offline music symbol recognition using daisy feature and quantum grey wolf optimization based feature selection," *Multimedia Tools and Applications*, vol. 79, no. 43-44, pp. 32011–32036, 2020.

[6] S. Panwar, P. Rad, K.-K. R. Choo, and M. Roopaei, "Are you emotional or depressed? learning about your emotional state from your music using machine learning," *The Journal of Supercomputing*, vol. 75, no. 6, pp. 2986–3009, 2019.

[7] K. Lopez-de-Ipina, N. Barroso, P. M. Calvo et al., "Multilingual audio information management system based on semantic knowledge in complex environments," *Neural Computing and Applications*, vol. 32, no. 24, pp. 17869–17886, 2020.

[8] J. Yang, X. Meng, B. Jiang, J. Man, Q. Meng, and B. Li, "FADN: fully connected attitude detection network based on

industrial video," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 2011–2020, 2020.

[9] M.-C. Chiu and L.-W. Ko, "Develop a personalized intelligent music selection system based on heart rate variability and machine learning," *Multimedia Tools and Applications*, vol. 76, no. 14, pp. 15607–15639, 2017.

[10] S. Guo, X. Zhang, Y. Du, Y. Zheng, and Z. Cao, "Path planning of coastal ships based on optimized DQN reward function," *Journal of Marine Science and Engineering*, vol. 9, no. 2, p. 210, 2021.

[11] C. Shuo and C. Xiao, "The construction of internet + piano intelligent network teaching system model," *Journal of Intelligent & Fuzzy Systems*, vol. 37, no. 5, pp. 5819–5827, 2019.

[12] P. G. Shambharkar and M. N. Doja, "Movie trailer classification using deer hunting optimization based deep convolutional neural network in video sequences," *Multimedia Tools and Applications*, vol. 79, no. 29-30, pp. 21197–21222, 2020.

[13] A. I. Tamboli and R. D. Kokate, "An effective optimization-based neural network for musical note recognition," *Journal of Intelligent Systems*, vol. 28, no. 1, pp. 173–183, 2019.

[14] G. R. Reddy, H. Singh, A. Domeika, N. M. Kumar, and M. Quanjin, "Modelling of heuristic distribution algorithm to optimize flexible production scheduling in Indian industry," *Procedia Computer Science*, vol. 167, pp. 1120–1127, 2020.

[15] E. L. Lasmar, F. O. de Paula, R. L. Rosa, J. I. Abrahão, and D. Z. Rodríguez, "RsRS: ridesharing recommendation system based on social networks to improve the user's QoE," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4728–4740, 2019.

[16] N. Gupta, S. Gupta, M. Khosravy et al., "Economic IoT strategy: the future technology for health monitoring and diagnostic of agriculture vehicles," *Journal of Intelligent Manufacturing*, vol. 32, no. 4, pp. 1117–1128, 2021.

[17] A. Firdaus, N. B. Anuar, M. F. A. Razak, I. A. T. Hashem, S. Bachok, and A. Kumar Sangaiah, "Root exploit detection and features optimization: mobile device and blockchain based medical data management," *Journal of Medical Systems*, vol. 42, no. 6, pp. 1–23, 2018.

[18] M. Muhairat, S. ALZu'bi, B. Hawashin, M. Elbes, and M. Al-Ayyoub, "An intelligent recommender system based on association rule analysis for requirement engineering," *Journal of Universal Computer Science*, vol. 26, no. 1, pp. 33–49, 2020.

[19] V. Subramaniyaswamy, R. Logesh, D. Malathi, V. Vijayakumar, H. R. Karimi, and M. Karuppiah, "Effective user preference mining-based personalised movie recommendation system," *International Journal of Computer Aided Engineering and Technology*, vol. 13, no. 3, pp. 371–387, 2020.

[20] S. Roy, M. Biswas, and D. De, "iMusic: a session-sensitive clustered classical music recommender system using contextual representation learning," *Multimedia Tools and Applications*, vol. 79, no. 33-34, pp. 24119–24155, 2020.

[21] X. Gao and Y. Wang, "Optimized integration of traditional folk culture based on DSOM-FCM," *Personal and Ubiquitous Computing*, vol. 24, no. 2, pp. 273–286, 2020.

[22] J. Zhang and G. Qu, "Physical unclonable function-based key sharing via machine learning for IoT security," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 7025–7033, 2019.

[23] W. Wang, N. Kumar, J. Chen et al., "Realizing the potential of the internet of Things for smart tourism with 5G and AI," *IEEE Network*, vol. 34, no. 6, pp. 295–301, 2020.

[24] S. Yang, T. Gao, J. Wang, B. Deng, B. Lansdell, and B. Linares-Barranco, "Efficient spike-driven learning with dendritic event-based processing," *Frontiers in Neuroscience*, vol. 15, Article ID 601109, 2021.

[25] J. Yang, C. Wang, H. Wang, and Q. Li, "A RGB-D based real-time multiple object detection and ranging system for autonomous driving," *IEEE Sensors Journal*, vol. 20, no. 20, pp. 11959–11966, 2020.

[26] A. Zielonka, A. Sikora, M. Wozniak, W. Wei, Q. Ke, and Z. Bai, "Intelligent internet of things system for smart home optimal convection," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4308–4317, 2021.

[27] N. Koroniotis, N. Moustafa, and E. Sitnikova, "A new network forensic framework based on deep learning for Internet of things networks: a particle deep framework," *Future Generation Computer Systems*, vol. 110, pp. 91–106, 2020.