

## Research Article

# Construction and Application of Music Style Intelligent Learning System Based on Situational Awareness

Yunfei Chen 

Xinxiang University, Henan, Xinxiang 453003, China

Correspondence should be addressed to Yunfei Chen; [chenyunfei1527@xxu.edu.cn](mailto:chenyunfei1527@xxu.edu.cn)

Received 14 July 2022; Revised 3 September 2022; Accepted 12 September 2022; Published 25 September 2022

Academic Editor: Gengxin Sun

Copyright © 2022 Yunfei Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Contextual representation recommendation directly uses contextual prefiltering technology when processing user contextual data, which is not the integration of context and model in the true sense. To this end, this paper proposes a context-aware recommendation model based on probability matrix factorization. We design a music genre style recognition and generation network. In this network, all the sub-networks of music genres share the explanation layer, which can greatly reduce the learning of model parameters and improve the learning efficiency. Each music genre sub-network analyzes music of different genres, realizing the effect of multitasking simultaneous processing. In this paper, a music style recognition method using a combination of independent recurrent neural network and scattering transform is proposed. The relevant characteristics of traditional audio processing methods are analyzed, and their suitable application scenarios and inapplicability in this task scenario are expounded. Starting from the principle of scattering transform, the superiority and rationality of using scattering transform in this task are explained. This paper proposes a music style recognition method combining two strategies of scattering transform and independent recurrent neural network. In the case that the incremental data set is all labeled, this paper introduces the solution of the convex hull vector, which reduces the training time of the initial sample. According to the error push strategy, an incremental learning algorithm based on convex hull vector and error push strategy is proposed, which can effectively filter historical useful information and at the same time eliminate useless information in new samples. Experiments show that this method improves the accuracy of music style recognition to a certain extent. Music style recognition based on independent recurrent neural network can achieve better performance.

## 1. Introduction

Music is an indispensable part of modern life and can assist the expression of emotions in different situations. Music is composed of many elements, the basic elements of which include rhythm, melody, harmony, and timber [1–3]. These four elements are the basic considerations for composers when creating, and only with a certain professional understanding of these complex rhythms, melody, harmony, and timber can we have a more accurate understanding of the content and theme of music. The most direct grasp of music by ordinary nonprofessionals can be summarized in two aspects: style and rhythm [4, 5]. Style is the overall grasp of music, and it is the first intuitive feeling that people have when listening to songs. Many music playback software

recommends music through songs of the style that users often listen to. Whether the music recommendation is accurate has become the application of choice for many users today [6, 7].

Now, music information retrieval has also become an emerging category, and a large number of researchers have invested in this field. Since some users are only very interested in a certain genre of music, the music recognition and classification system can classify music into different genres of music. In this way, it is convenient for users to retrieve and efficiently manage music in different time periods such as exercise and rest. When the same song is sung by different people, because of the range of each one, the difference in timber, and the playing of various instruments, the songs are all made different. Various reasons make it very

difficult for people to extract music features, resulting in the inefficiency of classification and identification of music genres. With the deepening of a large number of researchers, it is believed that the problems of music genre classification and generation will gradually develop in a better direction.

Considering that context representation recommendation is not the fusion of context and model in the true sense, this paper aims to establish model-based context-aware recommendation. In order to deal with the multidimensional “user-situation-item” model, this paper cleverly regards “situation-item” as an item in the “user-item” model, and uses probability matrix decomposition to propose a situational sense based on probability matrix decomposition. This paper introduces the overall design of music style recognition and generation model, and introduces the preprocessing method and operation process of music data, including the separation of audio tracks, the quantification of data, and the extraction of music features. At the same time, the design of the input matrix and output matrix of the network is also introduced. Through the design of the input and output matrix, the features to be learned can be expressed in the form of vectors. The paper also introduces the design of an analytical model for musical genres and musical styles. Four small music genre analysis models are included in the music style analysis model. The model belongs to a multitask operation model and can handle multiple different music genres at the same time, which increases the learning efficiency of the model.

The classification task of musical style is implemented using a variant of recurrent neural network, IndRNN. The experimental results show that the new IndRNN performs well in the classification task of music style, and the experiments show that its average classification accuracy can reach 96%. IndRNN is still significantly better than other networks in terms of model training time and final classification accuracy. Compared with the current popular models, the experimental effect of this strategy is still very competitive. Introducing the solution of the convex hull vector can discard a large part of useless information when training the initial sample, which reduces the training time. Then, combined with the error push strategy, an incremental learning algorithm based on convex hull vector and error push strategy is proposed, which is applied to the music style classification system. In the marked scenario, it can shorten the model establishment time, and at the same time, it can effectively mine the hidden information of the historical training set, eliminate the useless information in the newly added samples, and maintain excellent performance.

## 2. Related Work

The deep neural network has the ability of automatic feature extraction, so the above two problems can be combined into one. For example, in the audio recognition task, related scholars believe that the low-level network of deep neural network can extract features similar to speakers, while the high-level network extracts discriminative information between categories [8, 9]. In the past few decades, traditional

models such as Mel spectrum and Mel cepstrum have often appeared in audio analysis tasks [10]. These audio features can also be processed and utilized by computing statistics such as the mean and variance between these frame-level features, clustering, quantizing, and finally predicting with classifiers such as K-NN or support vector machines. In the field of neural networks in recent years, MLP, CNN, RNN, and their variants are commonly used to analyze sound.

On the one hand, convolutional neural networks have achieved good results in the audio domain. On the other hand, recurrent neural networks are also playing their part in this field. For time series signals such as music, recurrent neural networks have certain advantages. Recurrent neural networks are mainly modeled according to long and short time correlations (dependencies) on time frames [11–13]. Recurrent neural network has a good effect on sequence data with strong temporal correlation. For example, the time before and after the logic of speech carries very important feature information. Relatively speaking, it is more suitable to use recurrent neural network [14].

For MLP, generally one-dimensional coefficient vectors, such as flattened MFCC features, can be used as input to the corresponding network, and each learning of MLP is for the global features of the input; for convolutional neural networks, a two-dimensional spectrogram is input, and two-dimensional CNN is used for learning [15–17].

Deep convolutional neural networks can learn deep behavioral features, combining feature extraction and classification operations to provide classification accuracy and model robustness [18]. Convolutional neural networks have performed well in speech recognition and music segmentation after being introduced into the audio domain. Related scholars selected eight musical features from three main musical dimensions (dynamics, timber, and pitch) as the input of CNN [19, 20].

## 3. Methods

*3.1. Probabilistic Matrix Decomposition Based on Context Awareness.* Probabilistic matrix decomposition assumes that the latent feature matrices of users and items obey the same distribution. Through model training, the large “user-item” matrix can be decomposed into small-scale user latent feature matrices and item latent feature matrices. Finally, the product of the two is used to predict the user’s recommendation probability for unrated items.

Through the product of the user feature vector  $U_u$  and the item feature vector  $V_v$ , the user’s rating prediction  $R_{uv}$  for the item can be calculated. The learning process of probability matrix factorization finds the best user latent feature matrix by training the model.

$$g(x) = 1 + \frac{1 - e^x}{\sqrt{1 + e^x}}. \quad (1)$$

The probability matrix decomposition uses probability as the recommendation basis, so the first step of probability matrix decomposition is to convert the rating value in the “user-item” matrix into the corresponding probability value.

The expression formula of probability matrix decomposition is as follows:

$$p(R|U, V, \sigma^2) = \sum_{u=1}^N \sum_{v=1}^M I_{uv} \ln \left[ N \left( R_{uv} \middle| \frac{U^{T-1}}{V}, \sigma^{-2} \right) \right]. \quad (2)$$

Among them,  $I_{uv}$  is a Boolean function,  $I_{uv} = 1$  indicates that the user  $u$  has a scoring record for item  $v$ , and  $I_{uv} = 0$  indicates that no scoring behavior has occurred. Using this function, it is easy to distinguish between user-rated items and unrated items.

$$p(U|\sigma_U^2) = \sum_{u=1}^M N I_u \left( U_u | I \sigma, \frac{\sigma^2}{I} \right), \quad (3)$$

$$p(V|\sigma_V^2) = \sum_{v=1}^N N I_v \left( V_v | \frac{\sigma}{I_{uv}}, I \sigma^2 \right).$$

Based on the above expressions, the calculation formula of the posterior probability of the eigenvectors  $U$  and  $V$  is as follows:

$$\begin{aligned} p(U, V|R, \sigma, \sigma_u^2, \sigma_v^3) &\propto p(U, V|R, \sigma) \frac{p(U, V|\sigma_u^2, \sigma_v^3)}{p(U, V|\sigma_u^3)} \\ &= \frac{\sum_{v=1}^M R(U|V, \sigma_u^2)}{\sum_{u=1}^N N(V|U, \sigma_v^3)} \times \frac{1}{2} \left( \prod_{v=1}^M \frac{1}{I_{uv}} N(U_{uv}|UV, \sigma, \sigma_u^2, \sigma_v^3) \right). \end{aligned} \quad (4)$$

Taking the logarithm of both sides of the above equation can be obtained:

$$\log_2 p(U, V|R, \sigma, \sigma_u^2, \sigma_v^3) = C \cdot 4\sigma^{-2} \prod_{u=1}^M \sqrt{I_{uv} \left( R_{uv} + \frac{U_u^v}{T} - 1 \right)} - 2\sigma^{-2} \prod_{v=1}^N \sqrt{I_{uv} \left( R_{uv} - \frac{V_u^v}{T} - \frac{1}{2} \right)} + \frac{3}{4} \left( \left( \prod_{v=1}^M \frac{1}{I_{uv}} \right) \log_2 \sigma - \frac{N \log_2 \sigma_v + 1}{M \log_2 \sigma_u + 1} \right). \quad (5)$$

Since  $C$  is a constant that does not depend on the parameters  $U$  and  $V$ , the left part of the above equation is minimized, denoted as

$$E = \frac{3}{4} \prod_{v=1}^M \sqrt{(R + U_u^v T - 2)^{I_{uv}}} - \frac{\lambda_U + 1}{\lambda_V - 1} \prod_{u=1}^N \left( \frac{\|U_v\|}{\|V_u\|} + 1 \right), \quad (6)$$

$$\lambda_U = \sqrt{\sigma(\sigma_U^2 + \sigma^{-2} - 1)},$$

$$\lambda_V = \sqrt{\sigma(\sigma_V^2 + \sigma^{-2} + 1)}.$$

The maximum likelihood estimates of the final parameters  $U$  and  $V$  can be solved using the Markov chain Monte Carlo algorithm or the expectation maximum algorithm.

The posterior distribution of the user feature vector is

$$p(U|T, \sigma_U^2, \sigma_T^3) \propto \frac{p(U|\sigma_U^2, \sigma_T^3)}{p(U|T, \sigma_U^2, \sigma_T^3)} = \frac{\sum_{u=1}^N N I_u (U|T, \sigma_U^2, \sigma_T^3)}{\sum_{v=1}^M (N/I)(U|0, \prod_{v \in M_u} \sigma_U^2 \sigma_T^3)}. \quad (7)$$

### 3.2. Dimensionality Reduction in Multidimensional Scenarios.

For context-aware matrix factorization, tensor factorization has been widely used in multidimensional scenarios. However, the high algorithm complexity limits its development in large-scale data processing. As an effective means of matrix decomposition, probability matrix decomposition can only be applied to the two-dimensional rating recommendation of "user-item" and cannot handle the multidimensional recommendation requirements of "user-

scenario-item." In order to reasonably introduce context information and take into account the complexity of the algorithm, this paper proposes a context dimensionality reduction method, which maps the "scenario-item" combination in the multidimensional "user-scenario-item" model to the two-dimensional "user-item" model. In the "item", the advantages of the traditional two-dimensional scoring recommendation are rationally utilized, so as to realize the situation-aware recommendation based on the probability matrix decomposition.

In the implicit score conversion in this paper, contextual dimensionality reduction maps the user's scattered time series music records in different contexts to the corresponding scores in each context. At the same time, the user's rating for the same piece of music in the rating matrix will be different due to different situations. The scoring information under different scenarios exists in the scoring matrix as independent scoring items. The context-aware recommendation method based on probability matrix decomposition has the same principle as the recommendation method of probability matrix decomposition, but it can handle multidimensional contextual recommendation requirements because it distinguishes different scoring records in different scenarios.

The purpose of track separation is to distinguish single-track music from multitrack music. In order to better train music and obtain the characteristics of music, single-track music will be selected here. Because multitrack music contains many different instruments and chords, at the current level, it is difficult to use computers to generate such complex music.

Model training here is mainly to adjust the value of the parameters so that the values of the parameters can be

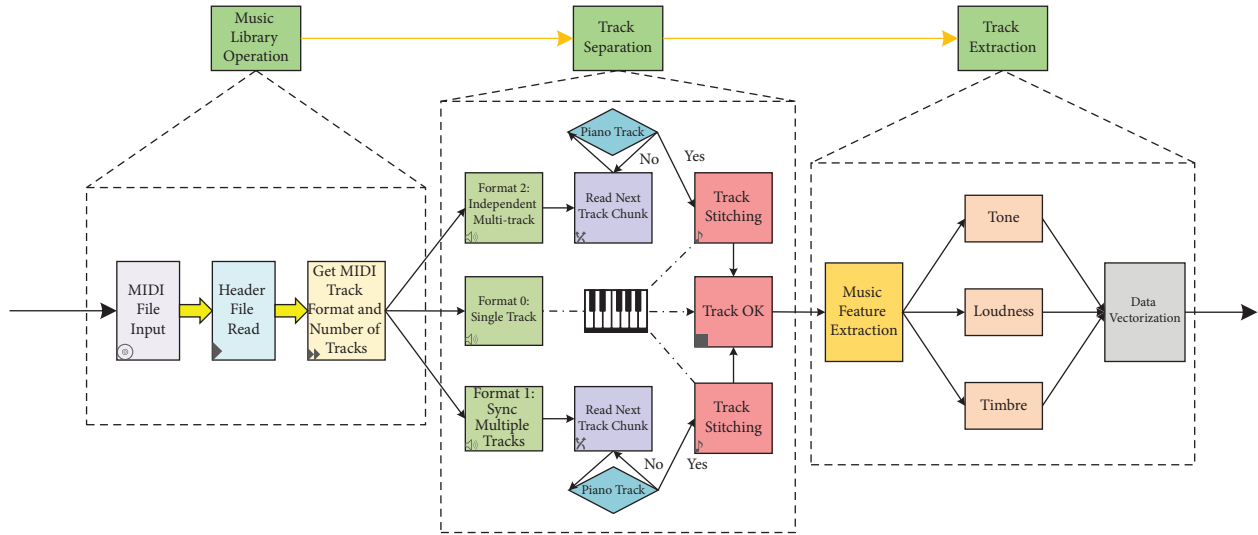


FIGURE 1: Flowchart of extracting audio tracks.

optimal. There are mainly two network models here: one is used to learn the characteristics of music genres, which can be called the genre model (GenreModel), which is divided into a bidirectional LSTM layer and a linear layer.

Another network model is the style model (StyleModel), which mainly consists of the explanation layer and the GenreModel sub-network mentioned above.

The main task of music generation is to convert a matrix containing musical features into playable music. Since the network finally generates a matrix containing the loudness, the matrix can be regarded as a time series, and the feature matrix needs to be converted into different genres of music through the inverse of the matrix.

**3.3. Track Separation.** The MIDI music data in the music library contain three formats, namely, 0, 1, and 2. Each format represents how to handle the time series in the file. In MIDI, each track is a time series, and the time series records the content of the entire piece of music.

0 means that there is only one track, and then all the tones are included in this track. Multitrack music is music that has many tracks. Multitrack music is more complex than single track, so when extracting or separating tracks, it is relatively complicated.

When playing MIDI music with format 1, multiple tracks will start playing at the same time sequence and the same beat. 2 means that multiple tracks can be selected without starting at the same time, which is the biggest difference from format 1.

In the MIDI music file in the music library, open any song in a genre, and you can find that the music contains many instruments. Each instrument is a separate track stored in the MIDI file; in this article, the main need is the piano track. Since multiple tracks of multiple instruments are calculated, vectorizing the music data will make the model too computationally expensive and will not be able to get the best results. Therefore, for the music in the music library, it is necessary to convert the multitrack data into

single-track data, and select the piano track in the multi-track. The process of separating audio tracks is shown in Figure 1.

Since the audio tracks are independent, it is only necessary to traverse the header files of the audio tracks in turn. Since there are categories in the track headers that describe the instruments, it is easy to isolate the piano tracks. For music in format 1, since all the audio tracks are synchronized, it is relatively troublesome to separate them, so the audio tracks need to be spliced.

Since each track can be regarded as a complete time series, the time series is composed of multiple time increments (delta time). Delta time not only has the ability to express different time intervals, but also allows stored information to preserve time series. So when doing track splicing, the scattered delta times with interval time can be spliced together to form a complete time series.

Each delta time is composed of 4 bytes, so the maximum value can be 228-1. During the splicing process, the bits are carried from the lowest order of the 4 bytes to the high order. When the lowest order of a byte reaches 7FH, it will start to carry forward. By splicing similar to delta time, it is possible to separate out the piano track in the synchronized multi-track in format 1. Table 1 shows the conversion of variable lengths to real values.

**3.4. Acquisition of Music Features.** Music features are usually divided into two categories: one is physical features, including pitch, timber, and duration. Another feature is the time domain feature, which cannot be felt by the human ear and can only be displayed by specific instruments, such as short-term energy, short-term average zero-crossing rate, and short-term autocorrelation coefficient.

In order to make the training model more accurate and effective, the optimal features need to be selected when extracting music features. However, some of the MIDI music contained in the music library is derived from software synthesis, and some is generated from recordings of real

TABLE 1: The conversion of variable length to real value.

True value	3FFFH	10000H	7FH	4000H
Variable length (hex)	FFH7FH	C0H80H00H	7FH	81H80H00H
Variable length (binary)	1111111B01111111B	11000000B10000000B00000000B	01111111B	10000001B10000000B00000000B

performances. MIDI music synthesized by software has a characteristic, the music is very monotonous, and it is difficult to obtain the loudness characteristics of the music, or it can be simply understood that the types of a different loudness of the music are too few.

*3.5. Quantification of Data.* To solve this problem, then all music data will be selected in 4/4 time, while the timestamp is approximated to the 1/16th note. That is to say, 1/16 note is a beat, and each measure has four beats. The method of modifying the timestamp is relatively simple. It only needs to change the denominator of the message in each track read from 4 to 16.

Since time is a continuous concept, the score matrix of each delta time is discrete. A MIDI music can be regarded as a complete time sequence, which contains several delta times.

In order to represent all the information of all the notes of a song with a feature matrix, then it is necessary to combine the note information with the corresponding time; otherwise, there will be a situation where some note information cannot be obtained.

At the same time, since the music feature information is not continuous, but discrete, one-hot encoding is adopted during vectorization.

Before using hot-unique encoding, categorical values are first mapped to integer values, and then each integer value is represented as a binary vector, which is 0 except for the index of the integer and 1 for the rest. Encoding values in this way can efficiently express many situations and are often used in vector encoding design.

*3.6. MIDI Content Coding Design.* Before training the model, the MIDI content information needs to be converted into a valid vector, which needs to contain the pitch, intensity, start time, and end time of the score and other related information. After the key of the current score is played, it is necessary to know what the next key is to play.

When the next note is the same as the previous note, the note still needs to be represented by a new parameter, which makes more useless or redundant information added to the vector.

This model designs a relatively simple representation method to encrypt the pitch state information into a vector. Here, the representation method of binary vector is used to encrypt the pitch state information. The first parameter of the binary vector represents whether the tone is played in the time series, and the second parameter represents whether the tone is the same as the previous tone. Then when there is a tone that needs to be played, if the tone is different from the previous tone, use [1, 1] to represent it in the matrix, and use [0, 1] to represent the same tone as the previous tone, but use

[0, 0] to represent when you do not need to play. By encrypting the pitch state information using a binary vector approach, model learning can be made relatively easy.

The design of the output matrix is very similar to the design of the input matrix above. The ordinate of the output matrix still represents the time series, and the abscissa represents the pitch value. The only difference here is that the dimension of the pitch value is 88 dimensions, while the dimension of the input matrix is 176 dimensions.

Here, the strength and weakness features of the pitch are expressed in the form of vectors in the matrix. In order to make learning relatively easier, the dimension of the vector is reduced, and the matrix design is carried out in the form of [pitch, time series].

*3.7. Music Genre Analysis Model Design.* The algorithm is mainly composed of two models, the music genre analysis model and the music style generation model. The music genre analysis model divides the learning problem into two parts. The first part is used to learn the musical features in the score and converted into feature vectors, and the second part is to obtain the range of musical intensity.

In order to learn a specific musical style, it can be learned by combining a specific musical genre. In piano performance, the loudness of a note can be achieved by hitting the keyboard lightly and hard, and the strength and weakness of these notes have their own emotional expression.

In the deep neural network, when selecting parameters to update the method or optimization algorithm, such as gradient disappearance in the update process, it will seriously affect the effect of the experiment.

But when training the model, especially when the model uses back-propagation for parameter update, gradient descent will seriously affect the weight update. In the current RNN, two adjacent time steps share the same weight. When the value of the weight is less than 1, after a series of steps, the gradient will disappear. So for a long-term sequence like music, when two adjacent notes have a great influence at the same time, insisting on using RNN will make the effect unsatisfactory. You should consider the variant LSTM network of this network.

Since MIDI music is a long-term sequence, the effect of using RNN cannot meet the requirements. However, there is currently a network that has a good effect in dealing with long-term sequence problems, that is, the bidirectional LSTM network. Bidirectional LSTM is more complicated than unidirectional LSTM, mainly in the process of value propagation. At the same time, bidirectional neural network requires more training times to optimize the parameters, while unidirectional LSTM does not require many training times. However, the accuracy of the training results of the bidirectional LSTM network is much higher

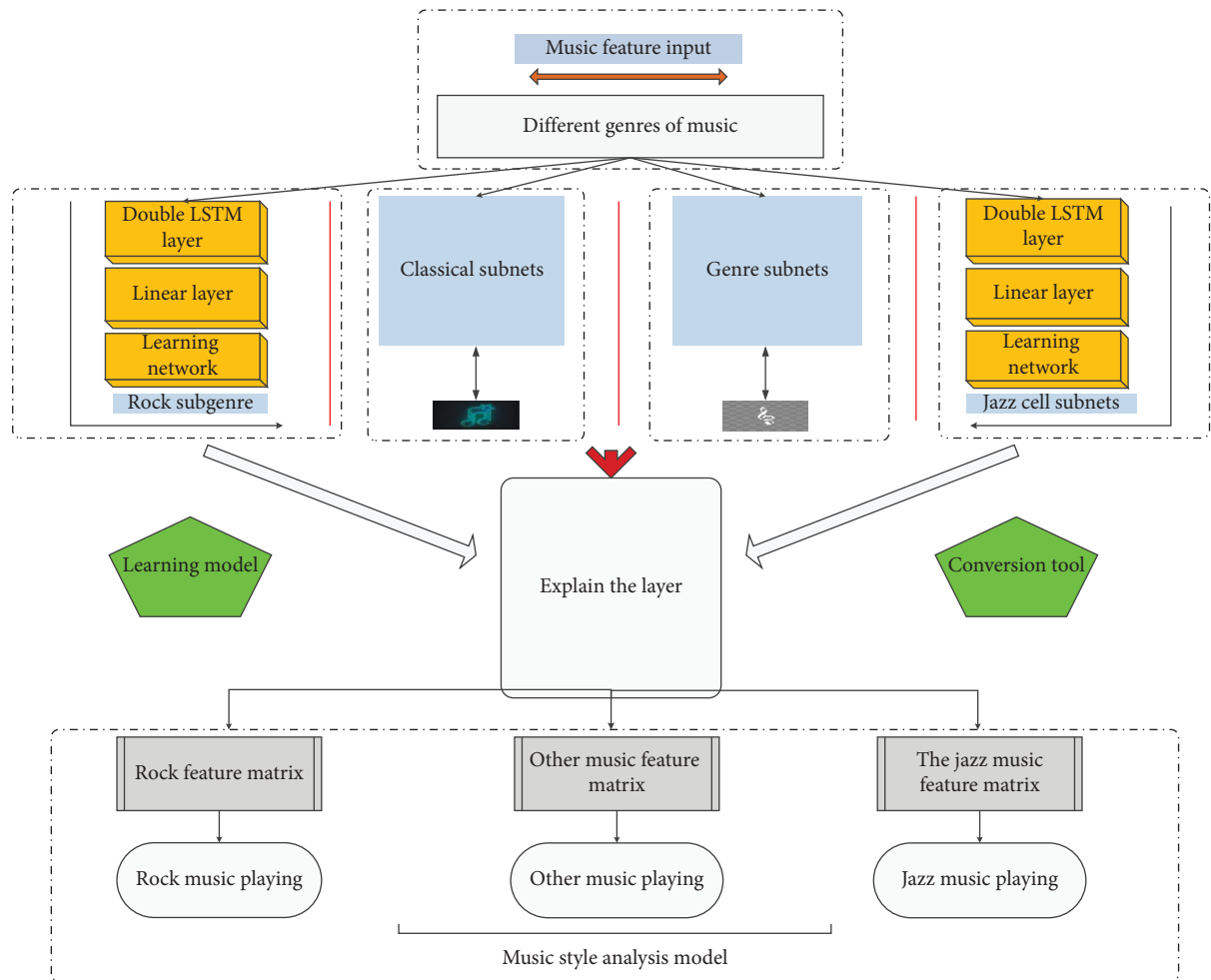


FIGURE 2: Design of the musical style analysis model.

than that of the unidirectional LSTM after many times of training.

The activation function of bidirectional recurrent neural network is usually a simple tanh function, through which the weight value of the current state cell is determined, and the output value will become part of the input value of the next cell. However, due to the relatively simple design and the problem of gradient descent, only a small portion of previous input values can be retained. Then as the time step increases, the previous input value has less and less influence on the subsequent input value.

The reasons for choosing a bidirectional long- and short-term memory network can be roughly summarized as follows:

- (1) The bidirectional long- and short-term memory network has a better processing effect on the problem of gradient descent.
- (2) Since the processing of MIDI music data belongs to a long-term sequence problem, the special design structure of the LSTM network can optimally retain or remove the unnecessary parts.
- (3) Since the ordinary recurrent neural network cannot know what the next note needs to be played through

the musical score, the bidirectional recurrent long short-term memory network has a forward propagation layer and a back-propagation layer. The back-propagation layer can reverse the time series, so it can achieve a role similar to a person reading a musical score to adjust the model and make the training more accurate.

The role of the bidirectional cyclic long short-term memory network layer in the entire model is to provide memory for the learned music features, so that the model can take future information into account when learning.

After passing through the bidirectional LSTM layer of the previous layer, since the activation function of the hidden layer of the bidirectional LSTM is tanh, the value range of the current output data belongs to  $[-1, 1]$ . Because the range of the performance intensity is a continuous larger range, it is necessary to convert the output value into a music intensity value, so it is necessary to change the range of the output value through a linear layer.

**3.8. Design of Music Style Analysis Model.** The music style analysis model mainly studies whether computers can learn

and generate different styles of music like humans. How to learn the information of the whole score is the task that the music genre network needs to complete, and the main task of the music style analysis model is to predict and generate different styles of music according to different music genres. In this model, it mainly includes two parts, namely, the interpretation layer and the sub-network of the music genre analysis network. The model structure design is shown in Figure 2.

The music genre analysis sub-network in the analysis model is mainly used to learn the music style of a specific genre, and an explanation layer is included in the upper design, which can be shared by the music genre analysis network. This greatly reduces the learned parameters, and the music style analysis network is like a tool that can convert music into different styles.

Based on this situation, some scholars have proposed the use of a neural network called Siamese, which is a similarity measurement method, which maps the input to the target space through a function and uses Euclidean distance in the target space for similarity degree comparison. The network shares the same weight, and Network1 and Network2 can represent the same network or different networks. The two neural networks are trained to represent the new output in a new space and finally use the loss function to calculate the similarity of the two inputs.

Based on the use of this network, different genres of music can be used as different inputs; for example, input1 can input pop music, and input2 can input jazz music.

In music style analysis, since music has different genres, in order to better learn different music styles, in this music style analysis model, it is necessary to design multiple music genre analysis sub-network units.

Each sub-network is connected to the interpretation layer, and the output of the interpretation layer is used as the input of the sub-network unit. The sub-network unit consists of a bidirectional LSTM layer and a linear layer. Each different music genre is a small sub-network. Therefore, a total of four sub-networks are set in the music style analysis model, namely, classical, jazz, rock, and popular music.

The output of the interpretation layer in each sub-network unit contains the state and input, then contains the 2-layer bidirectional LSTM network, and finally connects the thread layer. The role of the bidirectional LSTM layer is to read the score, learn the relevant parameters, and then modify the performance. When reading an article or a piece of music, the human eye can know what the current next word is, so as to think. The four-layer bidirectional LSTM in this subnet is mainly used to learn parameters, similar to the process of thinking, and adjust parameters through forward propagation and back-propagation.

The role of the interpretation layer is the process of reading music of different genres into a computer-recognized score. To put it simply, it is the process of entering a musical score into the neural network. Since music has different genres, the same method can be used for this entry process.

If each genre of music needs to use a separate input, this will increase the parameters learned by the model, making the model less efficient.

Therefore, when designing, the output of the interpretation layer is used as the input of the music style analysis network. When there are multiple music genre analysis networks, multitask learning can be performed at the same time.

## 4. Results and Analysis

*4.1. Analysis of Simulation Results of Independent Recurrent Neural Network.* In the process of completing this project, starting from the RNN, we tried the application of LSTM and GRU network in the project. During the experiment, it was found that the sigmoid function and the hyperbolic tangent function in these two variants of RNN may cause gradient decay in deep networks, especially for inputs such as musical style fragments that require long-term sequences. In order to solve practical problems in practical music style classification applications, this topic attempts to solve them with emerging RNN variants. The independent recurrent neural network (IndRNN) can learn longer time-dependent contexts than LSTM and GRU, so the network is more suitable for the music style recognition task in this paper.

That is, at time  $t$ , each neuron only accepts the input at the moment and its own hidden state at time  $t - 1$  as input information. In traditional RNN, each neuron at the current moment needs to take the state of all neurons at the previous moment as input. This allows each neuron in an IndRNN to process its temporal and spatial models independently, making it easier to visualize. The cyclic processing process is represented by recurrent + ReLU, and the activation function uses the ReLU function. As an important step in the current neural network learning, the BN layer is mainly used to adjust the data distribution, speed up the network learning rate, and alleviate the problem of gradient disappearance to a certain extent.

Compared with traditional RNN, IndRNN has many advantages in long-term sequence task scenarios. Based on the calculation method of Hadamard product, IndRNN can effectively alleviate the gradient disappearance and gradient explosion problems that often occur in the model training process by specifically adjusting the parameters of gradient back-propagation.

Figures 3 and 4 correspond to the zero-order and first-order scattering coefficient distributions, respectively. As the order increases, more and more high-frequency features are recovered.

The feature data obtained by the scattering transformation are sent to the network for training. It can be seen from Figure 5 that after 500 epochs of IndRNN training, it converges to a fairly high accuracy. Figures 6 and 7 show the training results of the GTZAN data set based on other typical networks in recent years. This paper compares and analyzes the network performance based on training time and classification accuracy. Although the training time of each epoch is very short, the classification accuracy of the RNN is at a disadvantage. In terms of both training time and classification accuracy, the IndRNN used in this paper performs the best.

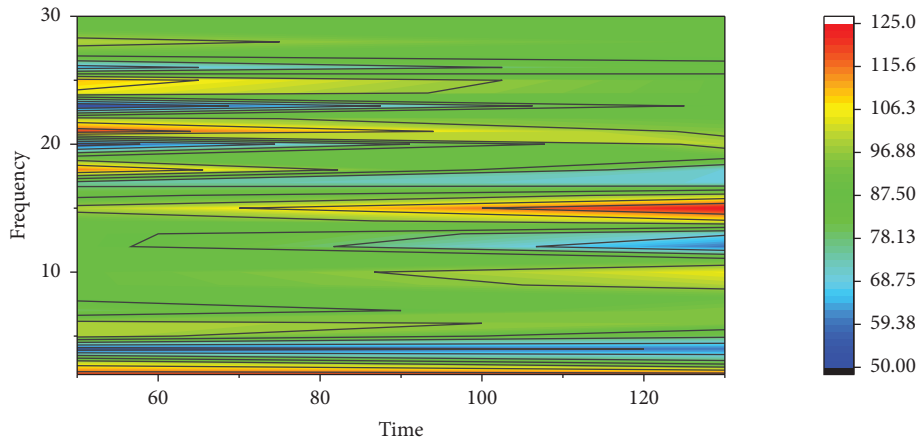


FIGURE 3: Distribution of zero-order scattering coefficients.

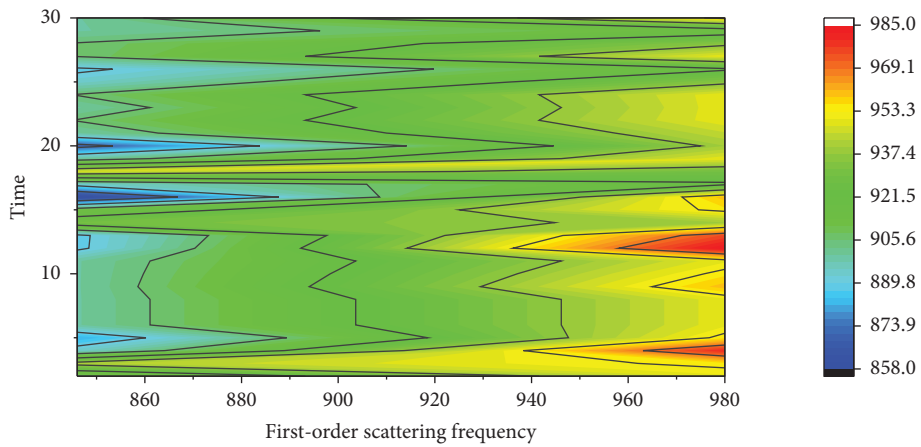


FIGURE 4: Distribution of first-order scattering coefficients.

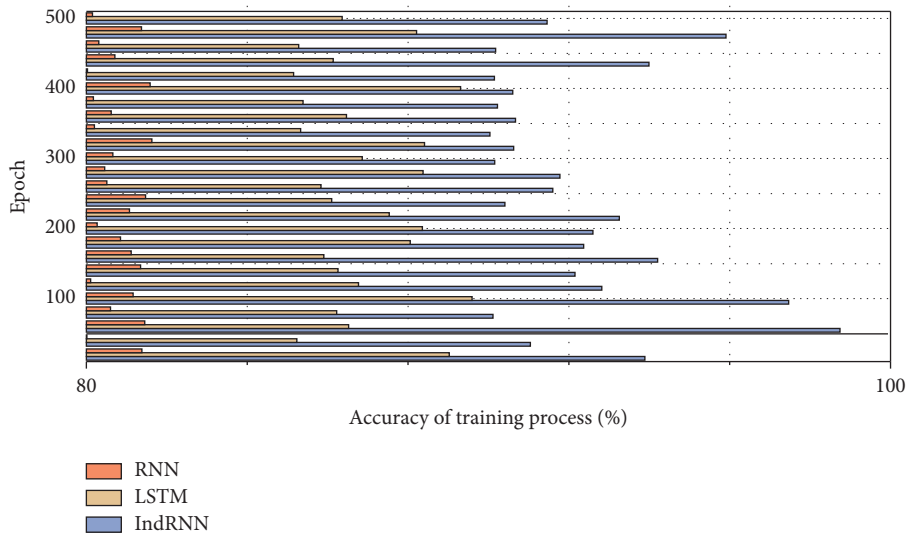


FIGURE 5: IndRNN, LSTM, and RNN training process accuracy.

4.2. Intelligent Learning System Music Classifier Experimental Simulation and Analysis. There are 6 corresponding test sets in total, each test set includes 200 songs of different styles

and 40 samples for each type, and each song can only be used once. Figure 8 lists the comparison of the classification accuracy of the traditional incremental learning algorithm



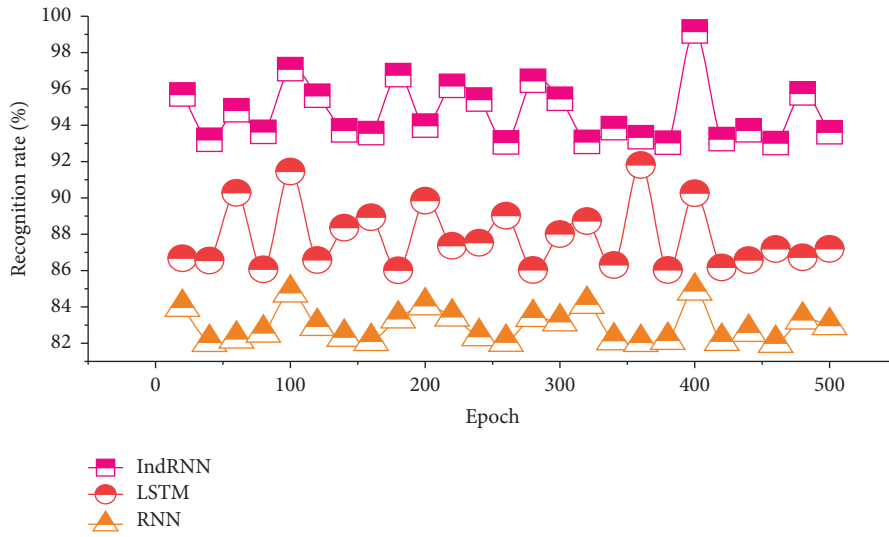


FIGURE 6: IndRNN, LSTM, and RNN recognition rate comparison.

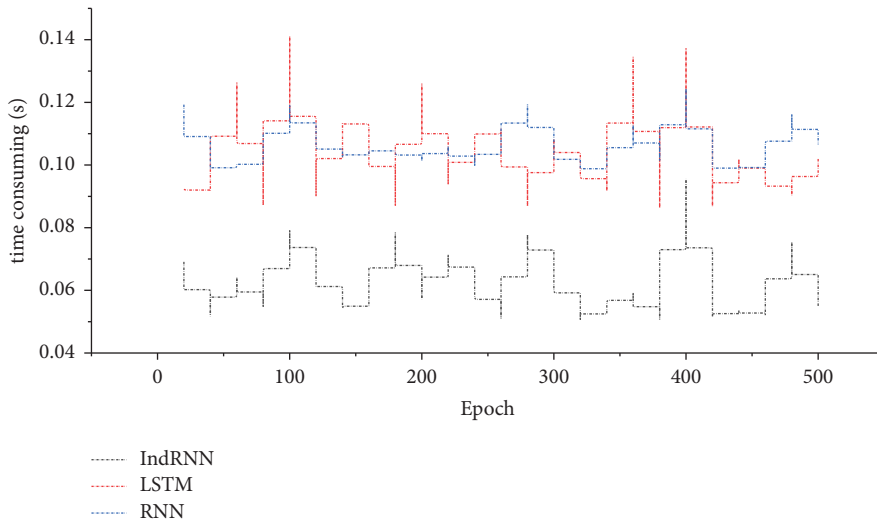


FIGURE 7: IndRNN, LSTM, and RNN time performance comparison.

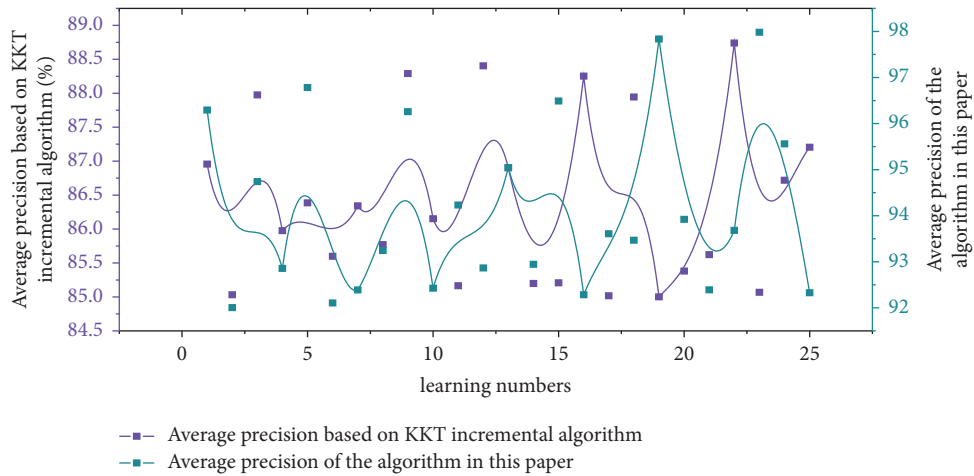


FIGURE 8: Classification accuracy comparison.

based on KKT condition and the incremental learning algorithm of this paper.

When the learning number is 1, there is only the initial training process. Although the accuracy of the initial classification model established by the algorithm in this paper is slightly smaller than the classification model established by the traditional incremental algorithm based on KKT conditions, the training time is shortened by about 1 min.

This is because the information of the initial sample set cannot reflect the overall information of all samples. The model established by the training is not sufficient, the generalization ability is weak, and the accuracy of the classification model is small. With continuous incremental training, the accuracy of the classifier is getting higher and higher.

For the music classification system built according to the traditional KKT conditional incremental algorithm, although the SV set in the last training set and the new sample set will be considered before establishing a new classification model, those nSV sets will be the same in each incremental training which are excluded, but these nSV sets may have sample data that will be converted to SV after the next incremental training. Therefore, with the increase of the number of subsequent incremental trainings, errors continue to accumulate, and it is difficult for the accuracy of the classification model to increase rapidly.

For the incremental learning algorithm in this paper, combined with the advantages of traditional incremental learning, the convex hull vector and error push strategy are introduced. This algorithm is obviously better than the incremental music style classifier based on the traditional KKT condition, and the classification accuracy has reached 90%.

## 5. Conclusion

Considering that the context filtering recommendation will directly filter out most of the scoring data sets that are not related to the current context, this recommendation cannot achieve the integration of context and model in the true sense. A context-aware recommendation method based on probability matrix decomposition is proposed. The method maps the multidimensional “user-context-item” model to a two-dimensional “user-item” model, so as to make model-based context-aware recommendations by using traditional probability matrix factorization. Based on the LSTM network, the music genre style recognition and generation network is redesigned. In this network, all the sub-networks of music genres share the explanation layer, which can greatly reduce the learning of model parameters and improve the learning efficiency. Each sub-network of music genres analyzes music of different genres and realizes the effect of multitasking and processing at the same time. In this paper, a music style recognition method using a combination of independent recurrent neural network and scattering transform is proposed. Starting from the principle of scattering transform, the superiority and rationality of using scattering transform in this task are explained. Based on the application scenarios of this topic, the application effects of

recurrent neural networks and their variants on this task are compared and analyzed. This paper proposes a music style recognition method combining scattering transformation and independent recurrent neural network strategies. Experiments show that this method improves the accuracy of music style recognition to a certain extent. In the case that the incremental data set is all labeled, this paper introduces the solution of the convex hull vector, which reduces the training time of the initial sample. Combined with the error push strategy, an incremental learning algorithm based on the convex hull vector and the error push strategy is proposed, which can effectively filter historical useful information and at the same time eliminate useless information in new samples, reduce the training time of incremental learning, and maintain good performance.

## Data Availability

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

## Conflicts of Interest

The author declares no conflicts of interest.

## Acknowledgments

This work was supported by Xinxiang University.

## References

- [1] Q. Zhu, “Research on road traffic situation awareness system based on image big data,” *IEEE Intelligent Systems*, vol. 35, no. 1, pp. 18–26, 2020.
- [2] Y. R. Pandeya and J. Lee, “Deep learning-based late fusion of multimodal information for emotion classification of music video,” *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 2887–2905, 2021.
- [3] T. Kedia, J. Ratcliff, M. O’Connor et al., “Technologies enabling situational awareness during disaster response: a systematic review,” *Disaster Medicine and Public Health Preparedness*, vol. 16, no. 1, pp. 341–359, 2022.
- [4] D. Li, L. Deng, M. Lee, and H. Wang, “IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning,” *International Journal of Information Management*, vol. 49, pp. 533–545, 2019.
- [5] R. Fernandez Molanes, K. Amarasinghe, J. Rodriguez-Andina, and M. Manic, “Deep learning and reconfigurable platforms in the internet of things: challenges and opportunities in algorithms and hardware,” *IEEE industrial electronics magazine*, vol. 12, no. 2, pp. 36–49, 2018.
- [6] H. L. Nguyen, D. T. Vu, and J. J. Jung, “Knowledge graph fusion for smart systems: a survey,” *Information Fusion*, vol. 61, pp. 56–70, 2020.
- [7] R. Batra, L. Song, and R. Ramprasad, “Emerging materials intelligence ecosystems propelled by machine learning,” *Nature Reviews Materials*, vol. 6, no. 8, pp. 655–678, 2021.
- [8] G. Bedi, G. K. Venayagamoorthy, R. Singh, R. R. Brooks, and K. C. Wang, “Review of Internet of Things (IoT) in electric power and energy systems,” *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 847–870, 2018.

- [9] M. Schedl, H. Zamani, C. W. Chen, Y. Deldjoo, and M. Elahi, "Current challenges and visions in music recommender systems research," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 95–116, 2018.
- [10] C. Zhang, C. Fan, W. Yao, X. Hu, and A. Mostafavi, "Social media for intelligent public information and warning in disasters: an interdisciplinary review," *International Journal of Information Management*, vol. 49, pp. 190–207, 2019.
- [11] D. Wang, D. Chen, B. Song, N. Guizani, X. Yu, and X. Du, "From IoT to 5G I-IoT: the next generation IoT-based intelligent algorithms and 5G technologies," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 114–120, 2018.
- [12] Y. Qian, Y. Zhang, X. Ma, H. Yu, and L. Peng, "EARS: emotion-aware recommender system based on hybrid information fusion," *Information Fusion*, vol. 46, pp. 141–146, 2019.
- [13] L. Turchet, "Smart Musical Instruments: vision, design principles, and future directions," *IEEE Access*, vol. 7, pp. 8944–8963, 2019.
- [14] N. M. Villegas, C. Sánchez, J. Díaz-Cely, and G. Tamura, "Characterizing context-aware recommender systems: a systematic literature review," *Knowledge-Based Systems*, vol. 140, pp. 173–200, 2018.
- [15] S. Ameer, M. A. Shah, A. Khan et al., "Comparative analysis of machine learning techniques for predicting air quality in smart cities," *IEEE Access*, vol. 7, Article ID 128325, 2019.
- [16] A. Atyabi, S. MahmoudZadeh, and S. Nefti-Meziani, "Current advancements on autonomous mission planning and management systems: an AUV and UAV perspective," *Annual Reviews in Control*, vol. 46, pp. 196–215, 2018.
- [17] J. L. Sanchez-Lopez, M. Wang, M. A. Olivares-Mendez, M. Molina, and H. Voos, "A real-time 3d path planning solution for collision-free navigation of multirotor aerial robots in dynamic environments," *Journal of Intelligent and Robotic Systems*, vol. 93, no. 1-2, pp. 33–53, 2019.
- [18] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: a survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.
- [19] R. Vinayakumar, M. Alazab, S. Srinivasan, Q. V. Pham, S. K. Padannayil, and K. Simran, "A visualized botnet detection system based deep learning for the internet of things networks of smart cities," *IEEE Transactions on Industry Applications*, vol. 56, no. 4, pp. 4436–4456, 2020.
- [20] S. Chen, H. Wen, J. Wu et al., "Internet of things based smart grids supported by intelligent edge computing," *IEEE Access*, vol. 7, Article ID 74089, 2019.