ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Construction and practice of AI-driven mutual accompaniment system for piano teaching in musicology majors

Li Lin[1,*] *and Ping'an Zheng*[2]

[1] School of Music and Dance, Hunan University of Arts and Science, Changde, Hunan, 415000, China

[2] Changde Branch, China United Network Communications Group Co., Ltd., Changde, Hunan, 415000, China

SUMMARY: *The incorporation of artificial intelligence into the piano instruction may result in a more individualized approach to learning. With the use of AI technology to assist intelligently, students will be able to learn piano more efficiently, increase the quality of teaching, inspire their interest in music, and finally enhance their musical abilities and performance. The paper provides the guidelines to integrating AI technology into the piano instruction and suggests a model of piano performance that relies on human-machine collaboration with the help of deep learning. Based on the theoretical framework of deep learning sequence processing and combined with the Onsets and Frames model, a model of piano music transcription is built. This study is using a binary cross-entropy objective in order to guide the parameter tuning through step by step. A framework of piano evaluation based on MIDI was subsequently created to confirm its usefulness in practice. The results of the experiment gave an average F-Measure of 96.43% when tested on several piano pieces, indicating highly effective performance and consistent testing results among piano students. When students completed a platform-supported learning experience, they were assessed on applied learning, knowledge extension, and innovative thinking using p-values of 0.052, 0.054, and 0.05 compared to traditional teaching methods. These results indicate that the online teaching strategy presented in the current paper would provide higher educational results as compared to the ones reached under conventional teaching conditions.*

KEYWORDS: *AI technology; deep learning; Onsets and Frames model; binary cross-entropy loss function; piano teaching*

1 Introduction

The digital economy has been the first economic model after the agricultural and industrial economies but has now entered a new stage of smart economy which is fueled by artificial intelligence [1, 2]. The traditional approach to learning the piano, namely, the piano mutual practice, is also changing due to the development of digitalization, which results in the creation of piano AI practice systems [3]. As compared to conventional in-person piano practice classes, AI practice has several benefits including no limits to space and time, the existence of an extensive curriculum system, sound incentive systems in addition to a lower cost and thus it is very popular in the market [4-6]. There is a growing number of consumers picking up piano AI practice goods to help them study, and these goods are taking over a

*13508466213@163.com
https://doi.org/10.65102/is2026464
Published: 30 April, 2026

share of the traditional piano practice market demand. The pandemic came abruptly to further hasten this change [7-9]. Moreover, the availability of piano AI practice products at low prices and easy access has allowed certain adults who could not learn piano previously because of these aspects to take piano classes, not necessarily to those of younger age or people involved in exams. An increasing number of people are now pursuing music education [10-13]. In professional piano teaching within music education, accompaniment plays a role in enhancing performance skills, optimizing thinking patterns, stimulating creativity, and shaping a well-rounded personality [14, 15]. As an emerging teaching model, whether the AI piano collaborative accompaniment system can continue to and better achieve these educational objectives beyond providing convenient practice opportunities requires feedback from piano teaching practice.

This paper utilizes built-in high-precision sensors and advanced algorithms to capture and analyze students' piano performance data, establishing a human-machine collaborative piano performance model. Based on deep learning sequence processing theory, an AI-based piano teaching training model is established. The Onsets and Frames model is employed to achieve signal transcription training for piano music, and a binary cross-entropy loss function is introduced to guide the gradual optimization of model parameters. A MIDI piano evaluation model framework is designed, with music features vectorized. Attention mechanism layers and softmax classification algorithms are introduced to obtain MIDI music classification after normalization. An audio preprocessing module and performance evaluation module are designed to jointly establish a piano teaching mutual assistance and accompaniment system, with practical analysis conducted on the system.

2 Building an AI-driven piano teaching system

2.1 Methods for integrating AI technology into piano teaching

2.1.1 Intelligent Training System

One of the major innovative applications of AI technology in piano teaching is the intelligent practice companion system. These systems use built-in high-precision sensors and advanced algorithms to capture and analyze students' performance data in real time, providing immediate feedback and significantly improving teaching effectiveness [16, 17].

This paper developed the “AI Piano Mutual Teaching and Companion Incentive System,” which supports real-time audio and video communication between teachers and students or between students and AI practice companions, simulating a real teaching environment. The system can identify the pieces students are playing and provide corresponding accompaniment or demonstration audio to assist students in practicing along. Based on students' playing levels and feedback, the system intelligently adjusts the difficulty of the practice sessions to ensure both effectiveness and challenge. Additionally, the system offers high-quality demonstration performance videos or audio for students to reference and learn from.

2.1.2 Personalized teaching plans

Personalized teaching plans not only improve teaching effectiveness but also enhance students' interest and motivation in learning. Many students have reported that using personalized teaching plans has made learning the piano more enjoyable. This paper utilizes the “AI Piano Mutual Teaching and Companion Motivation System” to design personalized learning paths for students. The system first conducts a series of tests and analyses to identify

students' weaknesses in areas such as note recognition, rhythm mastery, and harmonic understanding. Then, based on the analysis results, the system intelligently recommends suitable pieces, practice methods, and teaching resources. For example, for students who are weaker in rhythm mastery, the system recommends pieces with strong rhythmic elements that are easy to learn, along with corresponding rhythm practice methods.

AI technology also plays a supportive role in personalized teaching. Some smart pianos can capture students' playing movements through sensors, analyze them in real time, and provide feedback on playing quality, including evaluations of pitch, rhythm, and dynamics. For example, when playing a complex classical piece, the smart piano analyzes the student's performance in real time and points out which sections require further practice and which sections have been mastered well. This personalized guidance significantly improves students' learning efficiency and playing skills.

2.1.3 Virtual Concerts and Interactive Teaching

The AI technology has also been used to exploit virtual reality (VR) and augmented reality (AR) technologies to produce an immersive virtual concert and interactive teaching platform [18]. Virtual concerts and interactive teaching are not just a way of adding value to the teaching methods and formats, but they can make teaching more effective and learning experience exciting.

This paper utilized the “AI Piano Mutual Teaching and Companion Incentive System” to conduct remote teaching and successfully held a virtual concert. During remote teaching, students can participate in online courses from home by connecting to the internet via a smart piano. Teachers can view students' performances in real time through the remote teaching platform and provide guidance and feedback. This teaching method not only breaks the constraints of time and space but also makes communication and feedback between teachers and students more timely and frequent. At the same time, students can also interact and communicate with each other through the platform, sharing their learning experiences and performance insights.

In the virtual concert, the AI system can generate realistic virtual music scenes and audience reactions based on students' performance data and style characteristics. Students can interact with virtual instruments in this virtual environment and experience the performance sensations of different music scenes. For example, when performing an energetic symphony, the AI system generates a grand concert hall scene, allowing students to feel as though they are in a real concert hall, performing the piece alongside an orchestra. Additionally, AI teachers can provide real-time guidance in virtual classrooms, offering one-on-one teaching services. This teaching method not only addresses the issue of insufficient teaching resources in traditional education but also ensures that each student receives personalized guidance.

2.2 Human-machine collaboration piano performance mode

An audio analysis system that runs on deep learning can simultaneously determine the player touch and pedal depth, dynamically modifying the harmonic structure of the synthesised electronic tone in order to achieve acoustic resonance between conventional grand pianos and digital sound resources. The University of Tokyo intelligent accompaniment system analyses the rhythm patterns of a performer with a convolutional neural network and creates harmonious textures to match these rhythms in less than 0.3 seconds. The bidirectional audio stream processing technology has been implemented in the automatic playing piano series of Yamaha Disklavier. The emotional recognition module combines facial micro-expression detection and electromyographic signal measurement. Upon the detection of an increased limb tension of the performer, the system automatically decreases the complexity of the

accompaniment part. This physical-mental coupled modeling approach has been commercialized in the Steinway SPIRIO | r series.

The Carnegie Mellon University developed dynamic score system is based on reinforcement learning algorithms to provide real-time visual fingering cues depending on the variation in the rate of play. Its haptic feedback device communicates the distribution of key strike forces through the use of piezoelectric ceramic plates. Knowledge graph technology was used by the intelligent teaching system that has been presented by the Berlin University of the Arts to integrate both historical performance records and pedagogical theory. To overcome such challenges as lack of support of the wrist joint in students, it produces individualized training programs with the help of 3D motion capture. It is worth noting that the French company PianoTech revealed an AI accompaniment robot whose multimodal interaction system played an improvised version of Debussy's Clair de Lune with a pianist during the 2023 International Music Instrument Exhibition. The robot employed voiceprint recognition technology to record the rubato phrasing of the pianist and produce real time string quartet parts and thus cross instrumental collaboration.

3 Building an AI-based piano teaching and training model

3.1 Theoretical Foundations of Deep Learning Sequence Processing

3.1.1 Recurrent Neural Networks

Figure 1 shows the architecture of an RNN. Recurrent neural networks (RNNs) are a certain kind of deep learning model, which are particularly well-suited to working with sequential data. By virtue of the recurrent interactions within the network framework, RNNs have the power to maintain and use the past data, and therefore they are able to represent the temporal dynamics in time series data accurately. This special modeling functionality allows RNNs to show great application prospects in solving complicated problems like diverse kinds of time series data, natural language text, and audio signals. The fundamental aspect of the RNN model is based on its special recurrent structure design, which provides a mechanism to pass information across different time steps in a sequence. There are cases where every recurrent unit in an RNN is capable of processing two distinct types of input information, namely, the input data at the given time step and the hidden state of the last time step. This architecture allows RNNs to take in the current input data and combine it with the appropriate previous steps, hence being able to capture and analyze long-term dependencies and dynamics of time-series data. In response to receiving new input data, RNNs react to this modification by updating their internal hidden states, which involves the use of non-linear activation functions in most implementations. This update mechanism not only enhances the network's sensitivity to changes in sequence data but also improves the model's ability to model complex dynamic characteristics in time series data.

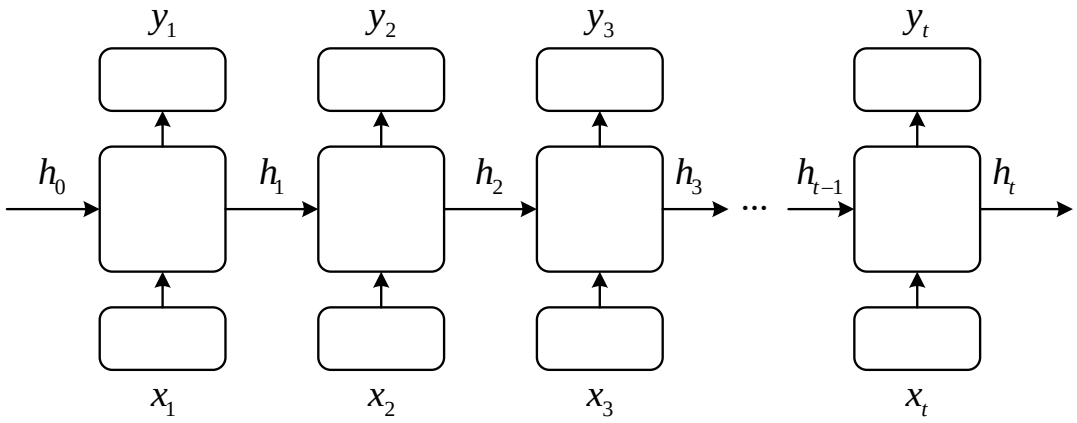


Figure 1: Structure diagram of the recurrent neural network

RNNs utilize historical information to model the current task. To achieve this, the network structure maintains a hidden state within the model, which captures the information from the sequence processed so far. The calculation of the hidden state can be described using a set of simple formulas. For a given time step t, the RNN unit receives the input vector x_i at the current time step and the hidden state h_{i-1} from the previous time step. The update of the hidden state h_i can be calculated using the following formula:

$$h_t = f(W_h h_{t-1} + W_x x_t + b) \tag{1}$$

In the formula, f represents the activation function, which is typically a nonlinear function such as tanh or ReLU. W_h and W_k are the weight matrices for the hidden state and input vector, respectively, while b is the bias term. In addition, RNNs can produce an output y at each time step, which is typically calculated by applying another set of weights to the updated hidden state h_t:

$$y_i = W_y h_i + b_y \tag{2}$$

Conclusively, recurrent neural networks (RNNs) have become one of the most important technologies in the area of deep learning to solve time-related issues because of their efficiency and flexibility in handling sequential information. Whether in applications such as time series prediction, natural language processing, or audio signal analysis, RNNs have demonstrated their powerful capabilities and broad applicability, as historical information serves as a crucial reference for predicting current states. They have made significant contributions to advancing research and development in related fields.

3.1.2 Long Short-Term Memory Network

The calculation process is shown in the following formula. First, based on the output h_{t-1} of the previous time step and the input x of this time step, calculate the forgetfulness gate f_t:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

Then, calculate the candidate vector values for input gates and updating cell states to determine which new information will be updated to the cell states:

$$i_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

$$\hat{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$

Next, combine the information from the forget gate and the input gate to update the cell state:

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{6}$$

Finally, based on the current cell state, the final output value is determined:

$$o_i = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{7}$$

$$h_t = o_t * \tanh(C_t) \tag{8}$$

In this context, σ denotes the sigmoid activation function, whose primary function is to compress input values from any range into the interval between 0 and 1, thereby effectively determining the activation level of the gating unit. This property enables the model to flexibly control the on/off switching of information flow, deciding which information should be retained or discarded. Meanwhile, the tanh function provides a mechanism to normalize input values within the range of -1 to 1, which helps maintain the stability of the network state. The * operator represents element-wise multiplication, a crucial computational step in the gating logic, enabling selective updates to the cell state. Through these carefully designed gating control mechanisms, LSTM not only demonstrates strong capabilities in learning long-term dependencies, overcoming the inefficiency of standard RNNs in handling long sequences, but also enhances the model's flexibility and accuracy in information processing. By precisely controlling the retention, updating, and forgetting of information, LSTM can maintain critical information across different time steps while discarding data that is no longer relevant.

3.1.3 Gate-controlled cycle unit

Specifically, the operation of the GRU unit can be described in the following steps:

First, calculate the update gate and reset gate based on the input and the previous hidden state, which determines how much information needs to be updated and forgotten in the current state:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{9}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{10}$$

Then, combine the output of the reset gate to calculate the candidate hidden state, which contains new information that may be added to the current state:

$$h_t^{'} = \tanh(W x_t + U(r_t * h_{t-1}) + b_h) \tag{11}$$

The final hidden state is the weighted sum of the previous hidden state controlled by the update gate and the current candidate hidden state:

$$h_i = z_i * h_{i-1} + (1 + z_t) * h_t^{'} \tag{12}$$

In summary, compared to LSTM, GRU has fewer gate controls and parameters, making it more efficient than LSTM in certain tasks. The fewer parameters and simplified structure also make GRU easier to train than LSTM while maintaining similar performance. These features make GRU a powerful and efficient tool for many sequence processing tasks.

3.1.4 Transformer

The core idea behind Transformer is the self-attention mechanism, which enables the model to focus on information at different positions in the input sequence and dynamically adjust its focus based on this information. The attention weights can be calculated using the following formula:

$$Q = W^Q X, K = W^K X, V = W^v X \tag{13}$$

$$on(Q, K, V) = softmax(\frac{QK^\tau}{\sqrt{d_k}})V \tag{14}$$

where Q, K, V are the results of multiplying the input X by the weight matrices W^Q, W^K, W^V, respectively. The multiplication operation of QK^T in (14) is used to calculate the similarity between each position of the input and all other positions. $\sqrt{d_k}$ is a scaling factor used to stabilize the gradient. where d_k is the dimension of the key vector.

In contrast to the sequential processing method of RNNs, Transformer model has the ability to compute all the elements in a sequence at once, which allows efficient parallel computing abilities. The self-attention mechanism is able to effectively represent the interactions between any two positions in a sequence with high accuracy, thus overcoming long-distance dependency problems. Moreover, the Transformer structure is highly flexible, and therefore may be used on different types of tasks (such as sequence-to-sequence translation models and single-sequence classification tasks). Also, through adding more layers to both the encoder and decoder, the Transformer model can be simply scaled to support tasks of different levels of complexity, which highlights its excellent scalability and flexibility.

3.2 Training Strategy for Existing Piano Music Transcription Models

3.2.1 Onsets and Frames Model

The Onsets and Frames model has also become one of the foundations of numerous existing research algorithms in the area of automatic piano music transcription because of its excellent performance in transcription. This model is designed according to an important music theory observation that does not all audio frames play an equal role in the music transcription but that some audio frames are given more emphasis as the onset frames of notes. Given the physical characteristic that piano notes rapidly decay in energy after their initial sound is produced, onset frames are not only easy to identify but also play a crucial role in the entire music perception process. Based on this understanding, the Onsets and Frames model has designed a detection mechanism for note onset frames and uses this information to enhance

the accuracy of pitch detection during the note's sustained state. The network structure is shown in Figure 2.

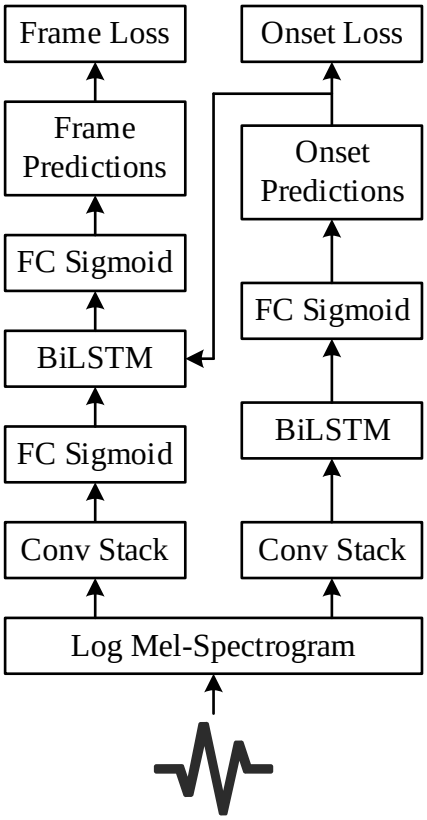


Figure 2: Network structure diagram of the starting and frame detection model

3.2.2 Binary cross-entropy loss function

The binary cross-entropy loss function (BCE) plays a central role in supervised learning, especially in binary classification problems, to measure the inconsistency between the model's predicted probabilities and the actual labels. This mechanism guides the gradual optimization of model parameters by minimizing the gap between the predicted probabilities and the target actual values. Its calculation formula is shown in Equation (15), where y_i and p_i represent the true labels and predicted probabilities, respectively:

$$BCE = -y_i \log(p_i) - (1 - y_i) \log(1 - p_i) \tag{15}$$

The BCE loss function is based on the notion of cross-entropy in information theory. The optimization of the model learning path is done by quantifying the difference in information between predicted output and actual output. In this process, when the model's predicted probability approaches the actual label, the loss value decreases accordingly; otherwise, it increases. This method is particularly suitable for scenarios where the output results are probability values with a range between 0 and 1.

BCE uses logarithmic functions for calculation and can be understood from the perspective of maximum likelihood estimation (MLE). MLE is a method used in statistics to estimate model parameters, aiming to find the parameter values that maximize the probability of the observed data occurring. For binary classification problems, the model typically sets the predicted probability of each sample belonging to the positive class (label 1) as p, while

the predicted probability of the negative class (label 0) is $1-p$. Under this setting, the likelihood function for the entire dataset is defined as the product of the observed probabilities for all samples, i.e., the product of the predicted probabilities for all samples, as shown in Equation (16):

$$L(\theta)=\prod_{i=1}^{n} p_i^{\,y_i}(1-p_i)^{1-y_i} \tag{16}$$

Here, θ refers to the model parameters. To simplify the mathematical processing and solution complexity, the log of the likelihood function is typically taken to obtain the log-likelihood function. The transformation is based on the fact that the logarithmic function is a monotonically increasing function, which transforms multiplication into addition without changing the extremes of the likelihood function. This results in a more efficient and stable mathematical optimization process, as shown in Equation (17):

$$\log(L(\theta))=\sum_{i=1}^{n}(y_i\log(p_i)+(1-y_i)\log(1-p_i)) \tag{17}$$

Within this framework, maximizing the log-likelihood function is actually transformed into minimizing the negative value of the log-likelihood function, which corresponds exactly to the form of the BCE loss function. Therefore, in practice, minimizing the BCE loss function is actually equivalent to performing a maximum likelihood estimation process, i.e., finding the set of model parameters that maximizes the probability of the observed data.

3.3 Design of MIDI Piano Performance Evaluation Model

3.3.1 MIDI Piano Evaluation Model Framework Design

Figure 3 shows the MIDI piano evaluation model framework. In the data collection module, the Sqoop tool is used to migrate data to the distributed data storage system HDFS. In the data preprocessing module, it is necessary to filter out raw data that is unsuitable for training and convert the raw data into an input matrix format suitable for training neural network models, and then divide it into training sets, validation sets, and test sets. In the music evaluation classification module, a Spark-Yarn cluster is set up, and a neural network model is constructed on the distributed framework. The preprocessed data is fed into the model for training, and the model parameters are adjusted in real time through the UI interface provided by Deeplearning4J to obtain model parameters with good evaluation results. The following sections will provide detailed introductions to the data preprocessing design and the MIDI piano evaluation neural network model design [19, 20].

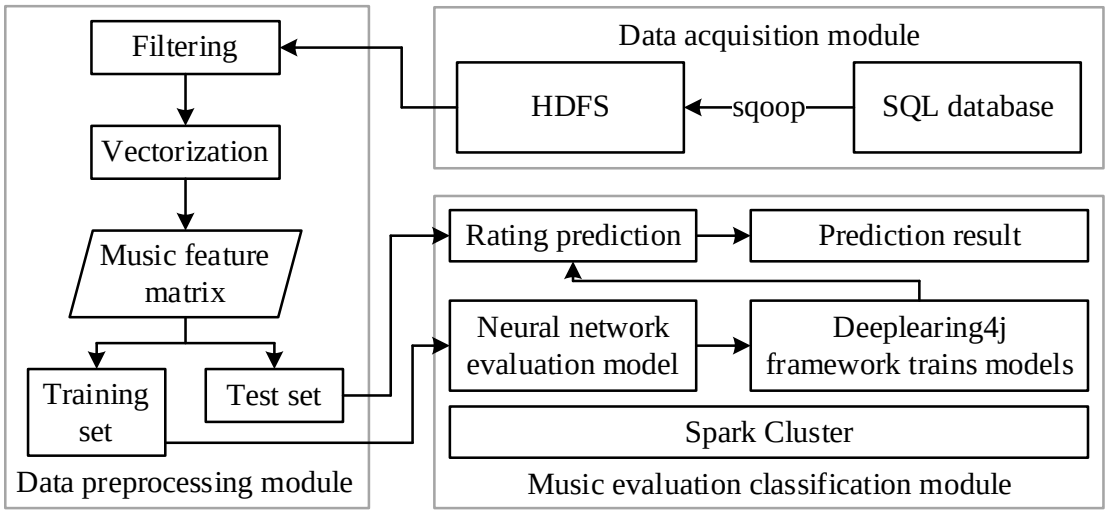


Figure 3: Framework diagram of the MIDI piano evaluation model

3.3.2 MIDI Music Feature Vectorization

This is why the horizontal axis is defined in 88 dimensions, and the vertical axis is defined in terms of the musical timeline, because the piano keyboard has 88 keys. This representation will record the state of the keys at various points in time and provide an input feature matrix of the keys being pressed. The matrix retains note tone, pitch, and duration in a fairly complete manner.

3.4 Design of a neural network model for evaluating MIDI pianos

3.4.1 Attention Mechanism Layer

The attention mechanism was originally applied in image processing. Its fundamental idea is to develop a recognition model that highlights critical information, similar to how humans recognize image classes by concentrating on salient areas when viewing an image. Both CNNs and RNNs do the overall recognition by extracting and aggregating features, but when it comes to modelling longer-range dependencies, both might suffer from diminished or vanishing dependency relationships. When the interval gets larger, these relationships become harder and harder to describe. Attention mechanism alleviates the issue of information dependence across large distances by mathematically connecting two data points regardless of their distance. This mechanism has been supported by Deeplearning4J since version 1.0.0-beta4. The respective algorithms are given by Equations (18) and (19).

$$\alpha = \frac{\exp(e_{t,i})}{\sum_{j=1}^{N} \exp(e_{t,j})} \tag{18}$$

$$c = \sum_{j=1}^{N} \alpha_{t,i} h_i \tag{19}$$

where $e_{t,j}$ represents the network output of the bidirectional LSTM.

3.4.2 Softmax Classification Algorithm

After passing through the attention mechanism layer, the Softmax function is used for classification. After normalization, the probability of which class the MIDI music belongs to is obtained. The Softmax function is a nonlinear function that maps the outputs of multiple neurons to a real vector in the (0, 1) interval, where the sum of all elements in the real vector is 1. The expression for the Softmax function is:

$$S(x_j) = \frac{e^{x_j}}{\sum_{k=1}^{k} e^{x_k}} \text{ j=1, 2, …, K} \tag{20}$$

where $S(x_j)$ represents the value of the i th dimension of the feature vector, and k represents the number of categories.

The classification label calculation formula for MIDI music evaluation prediction is:

$$\hat{m} = \arg\max S(x_j) \tag{21}$$

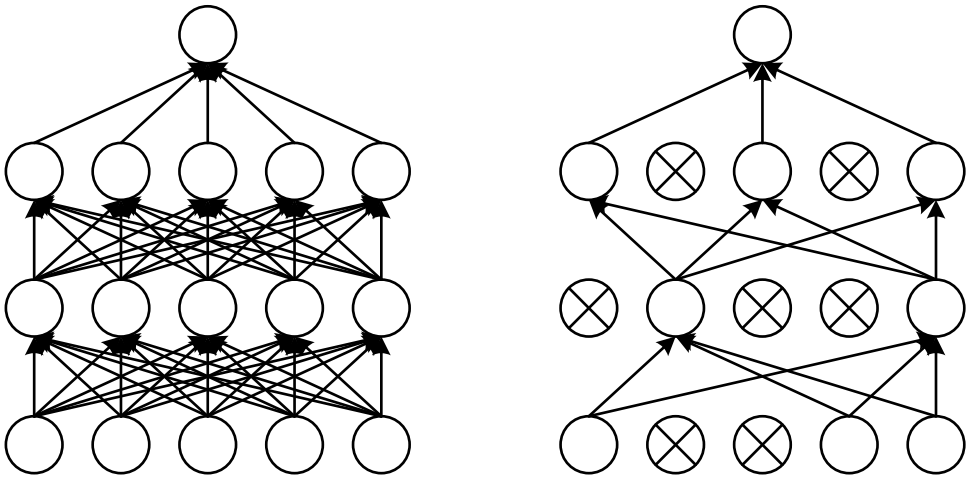
There is a very simple functional relationship between the gradient of the cross-entropy function and the output value of Softmax, which saves a lot of time in gradient calculation, making the calculation faster and more stable. Therefore, the loss function selected for training the MIDI music evaluation prediction model is the classification cross-entropy function, whose functional expression is:

$$L = -\sum_{j=1}^{T} y_j \log s_j \tag{22}$$

where S_j denotes the estimated probability of each category in the classification, T denotes the number of categories in the classification, and y_j denotes a vector of size $1*T$, where the j th element is 1 and all others are 0.

3.4.3 Dropout Optimization

Dropout is a technique used to prevent overfitting in neural networks. The Dropout optimization process is illustrated in Figure 4. The network structure before applying the Dropout optimization algorithm is shown in Figure (a), while the network structure after applying the Dropout optimization algorithm is shown in Figure (b). The principle behind this technique is that during neural network training, some hidden layer nodes are rendered inactive, and their parameters are not updated. However, this inactivity is only relative to the current training session. In practice, any hidden layer node can be trained during the entire training process. To determine which hidden layer nodes are inactive and which are active, each hidden layer node is assigned a probability P that follows a Bernoulli probability distribution, with P ranging from 0 to 1. The closer P is to 0, the lower the probability that the hidden layer node is active. When P equals 0, the hidden layer node is temporarily deactivated. Through extensive experimentation, it has been proven that Dropout effectively addresses overfitting issues.



(a) Neural network before Dropout

(b) Neural network after Dropout

Figure 4: Neural network before Dropout and Neural network after Dropout

4 Building a piano training system combining Kinect and MIDI

4.1 Chord Extraction and Chord Progression

4.1.1 MIDI signal acquisition

The system uses a MIDI keyboard as the chord input device and JFugue as the interface for obtaining MIDI signals. JFugue is a Java API for music composition. It can specify notes, instruments, chords, and other musical data using data strings without requiring low-level MIDI manipulation. It uses the MusicString format to define notes, converting them into digital form. In MusicString, note names are represented by Arabic numerals, sharps and flats are indicated by # and b, and chords are represented by digitized notes.

4.1.2 Chord Extraction and Chord Progression

A chord, by definition, refers to a group of notes that have specific interval relationships. On a piano staff, their positions and pitches vary. A chord consists of a root note and other notes, with three notes sounding simultaneously. A chord with intervals of thirds is called a triad, and there are also seventh chords, triads, and others.

Taking a triad as an example, let's explain chord inversions. A triad has two additional notes besides the root note (the third and fifth notes), so it can have two inversions. When the third is the lowest note, it is called the “first inversion” or “sixth chord.” This is because the third, as the lowest note, is a sixth interval from the root (highest note). When the fifth of the triad is the lowest note, it is called the “second inversion” or “fourth-sixth chord.” This is because the fifth, as the lowest note, is a fourth interval from the root, while the third is a sixth interval from the root.

To facilitate computer recognition of chord inversions, this paper encodes chords according to type and establishes a chord state transition table based on the encoding table. By comparing the pre-established chord state transition table, the correctness of students' chord inversion playing is evaluated. For all chords stored in the list, if the number of MIDI signals exceeds three notes within a unit of time (this time can be defined within the system), the

chord is subjected to chord judgment (excluding omitted notes). Each note is selected as the fundamental note, and its adjacent notes are looked up in the table. If the interval relationship meets the chord interval relationship, it is judged to be a chord.

4.2 Gesture Recognition

4.2.1 Gesture region segmentation based on Kinect

In the process of piano gesture recognition, the user's hands are placed on the piano, and the skin color differs from the black and white keys of the piano. Therefore, this paper chooses to extract gestures based on color image background difference and skin color modeling and hand threshold segmentation based on depth images.

Equation (23) is used to model the background, and equation (24) is used to complete the background difference:

$$B(x,y)=\frac{\sum_{i=1}^{n} f(x,y,t_i)}{n} i=1,2,\cdots,20 \tag{23}$$

$$D(x,y)=f(x,y,t_i)-B(x,y) \tag{24}$$

In the formula, $B(x,y)$ represents the grayscale value of each pixel on the color background, and $f(x,y,t_i)$ represents the grayscale value of the pixel at position (x,y) at time t_i, ranging from 0 to 255.

4.2.2 Gesture Feature Extraction

The moment is a rough feature obtained by summing all the points on the contour. Hu moment theory was first proposed by M. K. Hu, who proved the properties of moment translation invariance, rotation invariance, and proportional invariance, and gave expressions for seven invariant moments. These seven invariant moments are composed of linear combinations of second- and third-order central moments. These transformation-invariant region-based moments can be used to describe the shape features of an image.

For discrete digital images, the $p+q$ -order standard moment is defined as:

$$m_{p,q}=\sum_{1}^{n}\sum_{i}^{n} I(x,y)x^p y^q \tag{25}$$

The center distance of the $p+q$ stage is defined as:

$$\mu_{p,q}=\sum_{1}^{n}\sum_{i}^{m} I(x,y)(x-\bar{x})^p(y-\bar{y})^q \tag{26}$$

In the formula, p corresponds to the moment in the x dimension, q corresponds to the moment in the y dimension, $\bar{x}$ and $\bar{y}$ represent the center of gravity of the image, n and m represent the width and height of the image, and the order represents the exponent of the corresponding part. The normalized center distance is defined as:

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^{p}} \quad \rho = \frac{(p+q)}{2} + 1 \tag{27}$$

Constructing seven eigenvectors of the Hu moment using second- and third-order normalized centroid distances $h_k, k = 1, 2, \cdots, 7$.

4.3 System Function Module Design

4.3.1 Audio Preprocessing Module

The audio preprocessing module performs preprocessing on the audio before the piano performance evaluation, which includes file format conversion and performance piece identification. This module is linked to the WeChat interaction module and the performance evaluation module. Through the WeChat interaction module, users send requests for performance evaluation results, which are then parsed and processed by the server and transferred to the preprocessing module. Subsequently, the communication submodule receives the relevant requests and completes the preprocessing, after which the performance evaluation module detects and evaluates the audio transmitted by the user.

4.3.2 Performance Evaluation Module

(1) Performance Positioning

This module aims to match the time of the recorded performance with the time of the reference audio. With time-alignment, it will determine the precise points of correspondence between the timeline of the user playing and the timeline of the reference recording so that the notes to be played at any particular moment can be obtained out of reference performance information database. Then the output of the alignment is written into a location information database, which is subsequently used in assessing the notes. In this section, it primarily uses an alignment algorithm based on music fingerprints.

(2) Multi-Fundamental Frequency Detection

The module conducts frame-by-frame analysis of fundamental-frequency data in the audio signal to assess the content played more accurately. The first step is to determine the notes that sound in each frame of the recorded performance. Multi-pitch detection algorithm is used to detect multiple fundamental frequencies at the same time and it outputs indicate what notes are sounding in each frame. This detection result is saved on a frame-wise basis in the detection database to be assessed later on a note level.

(3) Note-level judgment

The module calculates note-level precision of the audio produced by the user, in particular, whether the notes played by the performer are similar to the target notes indicated in the reference recording. The system will use the alignment information created when performing localization to identify the precise locations that correspond to the reference audio to achieve this. Based on that, the anticipated note sequence is restored in the reference musical performance database and compared with the pitch-related data that was obtained after multi-pitch analysis of every frame. By this means, it becomes possible to decide whether the note performed in the current frame is correct.

(4) Result Generation

The final results of assessment are produced by this module based on the note-level analysis and it will create the associated feedback graph. It is presented in two large formats. There is one overall evaluation, which indicates the quality of the musical performance as a whole. Following the total accuracy score of the piece, the value is multiplied by 100,

rounded to an integer between 0 and 100 and placed on the score. The second is bar-level evaluation, which considers the quality of performance in each bar. The system, given the accuracy value at the measure level, accesses the appropriate information in the note-position database and makes a conclusion about whether the performance was correct in that measure.

5 Practical Analysis of a Mutual Assistance and Support System for Piano Teaching

5.1 System Performance Testing

5.1.1 Test Indicators

Currently, the mainstream evaluation standard for multi-tone detection in the field of music retrieval uses the F-Measure at the note level. The F-Measure is the weighted harmonic mean of precision and recall, and it is a commonly used evaluation standard in the field of information retrieval. The calculation process of the F-Measure is shown in Formula (30), and the calculation processes of precision and recall required for the F-Measure are shown in Formulas (28) and Formula (29) (where total represents the total number of notes in the sheet music, detected represents the number of notes detected by the system, and correct represents the number of correctly detected notes).

$$\text{Precision} = \frac{\text{correct}}{\text{detected}} \tag{28}$$

$$\text{Recall} = \frac{\text{correct}}{\text{total}} \tag{29}$$

$$\text{F-Measure} = 2^{*} \frac{\text{Recall* Precision}}{\text{Recall+ Precision}} \tag{30}$$

5.1.2 Test Results

This system is designed for piano learners, so the test set selected for the system's performance testing consists of classic piano textbooks: Thompson's Easy Piano Tutorial, from which representative pieces were selected as test samples. Figure 5 shows the system performance test results. For beginner piano learners practicing the sample pieces from Thompson's Easy Piano Tutorial, the overall F-Measure values range from 90% to 100%, with all values above 90%. On an average, the F-Measure value is 96.43%, which means that the system operates well and gives correct assessment results to piano learners.

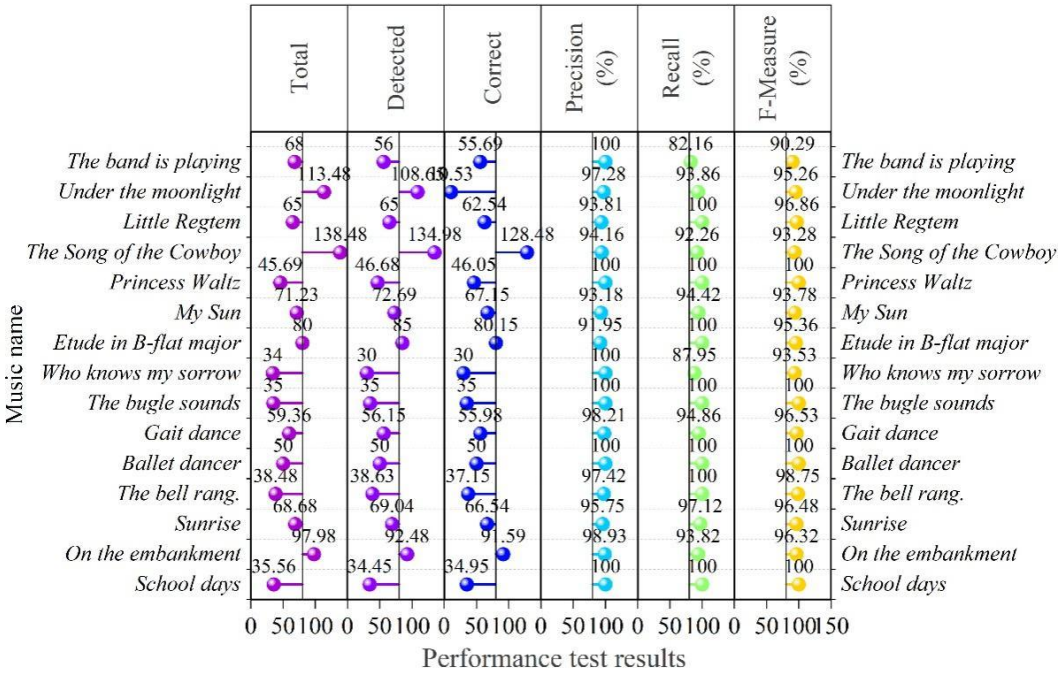
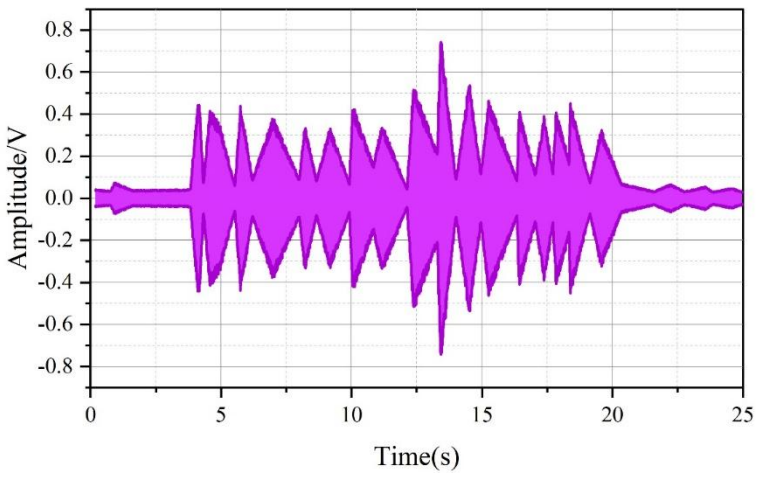


Figure 5: Systematic test results

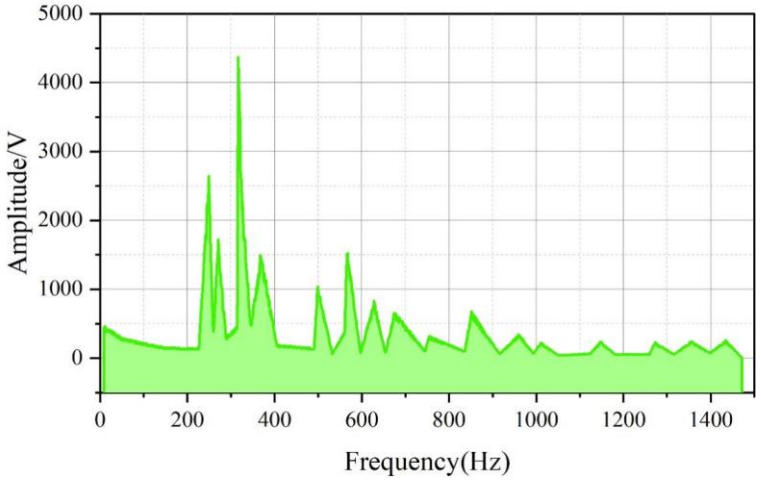
5.2 Correct and incorrect piano playing identification

5.2.1 Time-frequency, frequency

To facilitate the use of the "Piano Performance Error Recognition and Evaluation System," I developed a user interface using MATLAB as part of my final project. This interface is simple to operate, with all preprocessing, feature extraction, and matching comparison tasks performed in the background, outputting only the results for display. Figure 6 shows the time domain and frequency domain of the piano performance error recognition system, with Figure (a) representing the time domain and Figure (b) representing the frequency domain. The time domain range of the piano performance being played is between [-0.8, 0.8] V, and the frequency fluctuates between 0 and 1500 Hz, reaching a peak of 4458 V at approximately 350 Hz.



(a)Time domain



(b)Frequency

Figure 6: The time domain and frequency of the positive fault identification system of the piano

5.2.2 Performance accuracy rate

The system performs calculations and image output simultaneously while the user selects files. There are three buttons in the lower-left corner of the main interface. After clicking “Play MIDI Format,” users can listen to MIDI music through speakers or headphones. The music content corresponds to the piece displayed in the sheet music, and this .mid file is pre-produced and stored on the computer. Clicking the “Play WAV Format” button will play the actual recording file, i.e., the WAV format file to be analyzed. Before clicking the ‘Analyze’ button, users must select “Music Type” and “First Note Duration” from the dropdown menu in the top-left corner. These two options serve as input selections for feature extraction and matching analysis comparisons. After selecting the above two parameters, you can click the “Analyze” button to analyze the piano performance stored in the .wav format (the system can only analyze monophonic music). The progress bar displayed by the system shows the computer's processing progress. The system's analysis and processing work—preprocessing, various music feature extraction and storage, matching and comparison with standard templates, and result generation—is completed in the background. After the “Analysis” process is completed, the “Processing Results Screen” appears. Figure 7 shows the processing results of the piano performance accuracy recognition system. It can be seen that the model analysis results align with the fluctuation trends of the standard tempo's piano keyboard positions, with an error of approximately 0.2, achieving a performance accuracy rate of 97.56%.

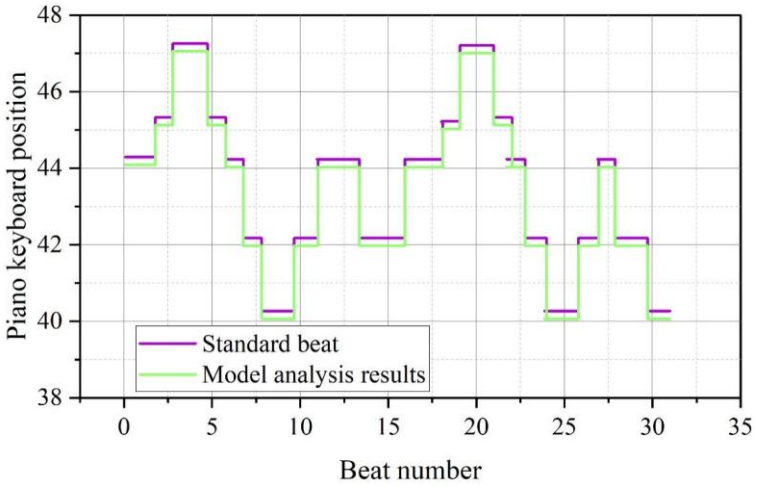


Figure 7: The piano plays the fault identification system processing result

5.2.3 Piano playing, chord fingering, and finger results

The main indicators used in this experiment are as follows.

The consistency rate indicator calculates the ratio of algorithm-marked fingerings to manually marked fingerings at each time point. The result is similar to the accuracy rate, with a higher index indicating a better result. The formula is as follows:

$$\alpha = 1 - \frac{|z \Lambda q|}{n} \tag{31}$$

In the formula: z, q represent the manually annotated fingering sequences and the algorithm-annotated fingering sequences, respectively, n represents the .sequence length, and Λ denotes the XOR operation.

The P-score metric primarily calculates the total number of finite cross-correlations between the standard chord sequence and the sequence of the chord nodes to be annotated, with a tolerance of approximately 20% of the median interval. Within this metric, it is considered correct. The algorithm involves pre-constructing 40 sequences of 25 seconds each, as expressed by the following formula:

$$T_i(n) = \begin{cases} 1 & n = a_i \\ 0 & n \neq a_i \end{cases} \tag{32}$$

In the formula, a_i represents the cross-correlation function.

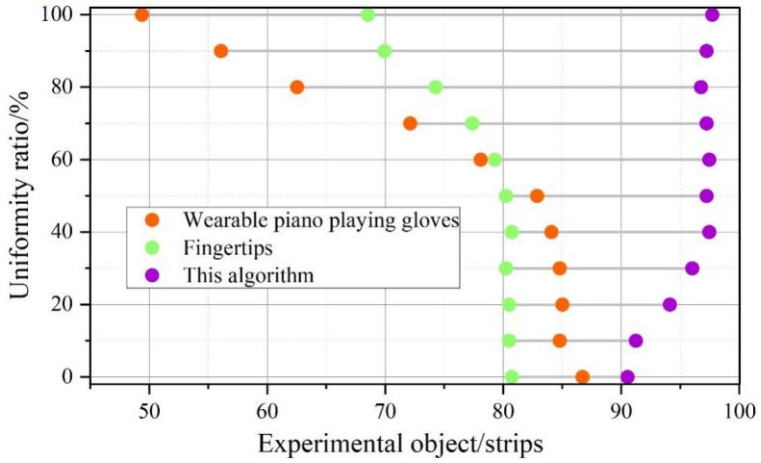
The proportion of non-pluckable fingerings measures the proportion of non-pluckable fingerings. In the fingering annotation process, non-pluckable fingerings must first be avoided in order to find the optimal fingering among numerous fingering sequences. The formula is expressed as follows:

$$\rho = \frac{\psi_1 - \psi_2 + 2N^{'} - m_d}{N^{'}} \tag{33}$$

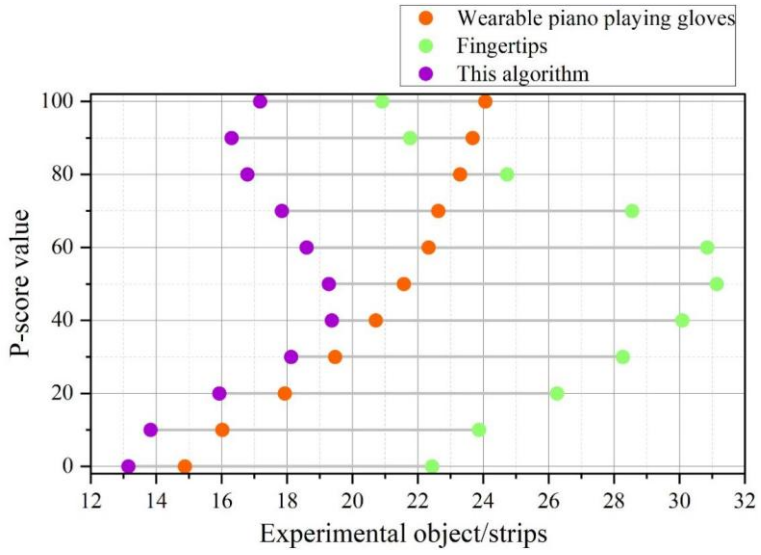
In the formula: m_d represents the number of segments, and $\psi_1 \cdot \psi_2$ represents the number of non-elastic fingerings.

To validate the effectiveness of the chord fingering automatic annotation algorithm in this study, a piano fingering automatic recognition method based on machine vision technology and a smart sensing and gesture recognition method using wearable piano playing gloves were used as control groups. The experimental dataset was collected using media statistics from the Amazon website, where users can label different styles on different tags according to their preferences, such as “lonely” or “fast-paced.” A total of 920 20-second videos were selected for this study, and 100 segments were extracted for fingering annotation.

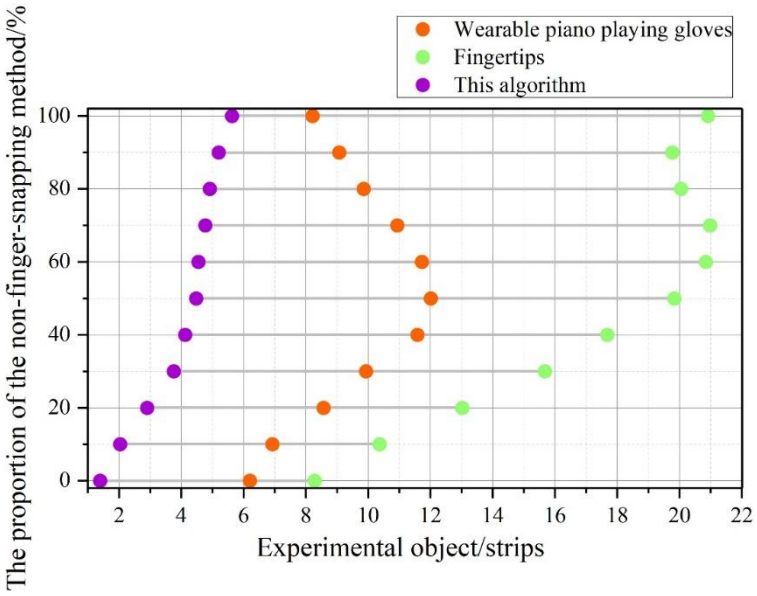
Figure 8 shows the experimental results analysis. Figure (a) compares the consistency rate, Figure (b) compares the P-score metric, and Figure (c) compares the proportion of unplayable fingerings. The algorithm proposed in this paper achieves a significant improvement in consistency rate, a reduction in the proportion of unplayable fingerings, and a lower P-score metric. The average values of the three metrics are 95.72%, 16.94%, and 3.98%, respectively. This demonstrates that the proposed algorithm can improve the consistency rate of chord fingering annotation. This is because the proposed algorithm pre-processes the chord sequence, increasing the number of available search paths, resulting in better performance compared to the other two methods.



(a)Consistency rate comparison



(b)Comparison of P-score indicators



(c)Comparison of the proportion of the finger-snapping method

Figure 8: Experimental results analysis

5.3 Learning Outcomes

5.3.1 Differences in Learning Outcomes

In the same piano room, the first group of students selected four pieces from Thompson Jr.1's "Bell", "Tug-of-War", "Symphony from the New World" and "Catching Game", the second group of students chose "The Song of Angels", and the third group of students chose the first movement of Mozart's "Piano Sonata K283 in G major". The four items of "able to achieve", "basically achieved" and "failed to achieve" were evaluated by ABCD. In order to understand the learning outcomes and skill levels of students using the system and traditional methods of teaching in this paper. The statistical results of the scores and the visual histogram are shown below.

Figure 9 shows the differences in learning outcomes. Students who learned piano using online teaching (the system described in this paper) scored 8, 9, 7, 7, 6, 5, and 7 points respectively on seven indicators, including playing technique, with an average score of 7 points, which is 1.29 points higher than the 5.71 points achieved by traditional teaching methods. This shows that using the system described in this paper for piano teaching and mutual assistance can help students improve their piano playing skills.

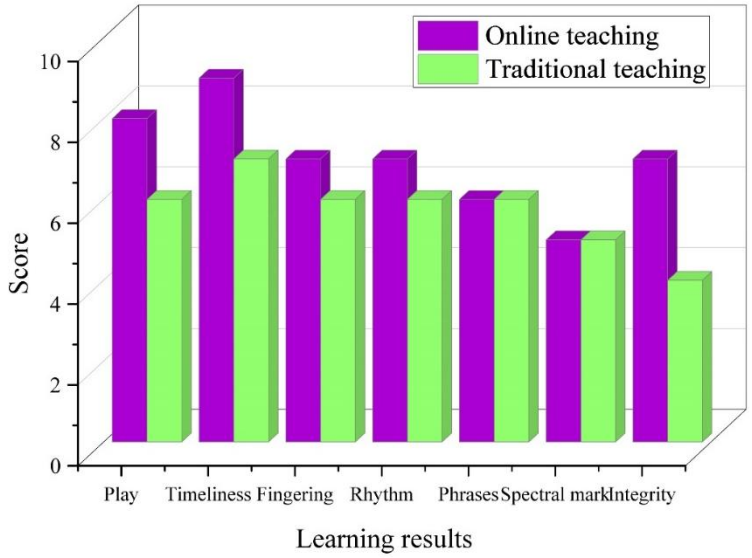


Figure 9: Learning difference

5.3.2 Learning Outcomes

(1) Classroom Organization

The secondary indicators in the classroom organization domain include three dimensions: behavior management, productivity, and teaching activity formats. Table 1 shows the score differences in the classroom organization domain. The p-values for behavior management and teaching activity formats are p=0.005 and 0.039, respectively, both of which are less than 0.05. Specifically, the t-value for behavior management is -3.19, indicating that traditional piano teaching classrooms significantly outperform online piano teaching in terms of behavior management. However, the t-value for teaching activity formats is 2.26, indicating that online piano teaching significantly outperforms traditional piano teaching classrooms in this dimension. The p-value for the output dimension was >0.05, indicating no significant difference between the two.

Table 1: The classroom organization field scores differences

Dimension	Teaching method	Mean	T value	P value
Behavior management	Online teaching	5.38	-3.19	0.005
	Traditional education	6.09		
Productive mind	Online teaching	5.98	0.76	0.458
	Traditional education	5.84		
Forms of teaching activities	Online teaching	5.98	2.26	0.039
	Traditional education	5.42		

(2) Teaching Support

The secondary indicators in the teaching support domain include cognitive development, feedback quality, and language modeling. After quantifying and analyzing the data from the observation form, Table 2 shows the differences in the teaching support domain. The p-values for online piano instruction and traditional piano instruction in the three dimensions of cognitive development, feedback quality, and language modeling are 0.869, 0.798, and 0.836, respectively, all of which are greater than 0.05, indicating no significant differences.

Table 2: Teaching support field differences

Dimension	Teaching method	Mean	T value	P value
Cognitive development	Online teaching	6.28	-0.29	0.869
	Traditional education	6.24		
Feedback quality	Online teaching	6.08	-0.34	0.798
	Traditional education	6.05		
Voice demonstration	Online teaching	6.28	0.25	0.836
	Traditional education	6.15		

(3) Teaching Effectiveness

Table 3 shows the differences in teaching effectiveness dimensions, measuring student feedback on teaching effectiveness, including three indicators: “students can apply knowledge flexibly,” “students' knowledge base is expanded,” and “students' innovative thinking is stimulated.” The p-values for the three indicators are 0.052, 0.054, and 0.05, respectively, indicating that there are significant differences between online piano instruction and traditional piano instruction in terms of students' ability to apply knowledge flexibly, expand their knowledge base, and stimulate thinking. The t-value for “students can apply knowledge flexibly” is -2.95, while the t-values for “students' knowledge base is expanded” and “stimulating students' innovative thinking” are both greater than 4. This indicates that in the “students can apply knowledge flexibly” indicator, traditional teaching is more effective than online teaching, but in the “students' knowledge base is expanded” and “stimulating students' innovative thinking” indicators, online teaching is more effective than traditional teaching.

Table 3: Difference in teaching effect dimension

/	Teaching method	Mean	Standard deviation	T value	P value
Apply what you have learned flexibly	Online teaching	8.08	1.03	-2.95	0.052
	Traditional education	9.63	0.59		
Knowledge development	Online teaching	10	0	4.03	0.054
	Traditional education	8.48	0.55		
Inspire creative thinking	Online teaching	10	0	4.18	0.05
	Traditional education	8.98	0.55		

6 Conclusion

This paper constructs a piano teaching and training system based on AI technology, combining deep learning models to process piano signal sequences. Additionally, a piano mutual practice model is trained using a transcription model, and a piano performance evaluation model based on MIDI is designed. Emphasis is placed on two aspects: audio preprocessing and performance evaluation, with a focus on designing the system's functional modules. An analysis of the practical effectiveness of the piano teaching mutual accompaniment system reveals that the model aligns with the standard tempo in terms of piano keyboard fluctuation trends, with errors consistently maintained around 0.2 throughout, achieving a performance accuracy rate of 97.56%. Compared to the other two algorithms, the proposed algorithm achieved a significantly higher standard result consistency rate, a lower proportion of unplayable fingerings, and a lower P-score. The average values of the three metrics were 95.72%, 16.94%, and 3.98%, respectively. Compared to the other two methods,

the proposed algorithm demonstrated superior performance. When comparing learning outcomes under different teaching modes, students who learned using the mutual assistance and accompaniment system in this paper achieved P-values of 0.052, 0.054, and 0.05 for the three indicators of “students can apply knowledge flexibly,” “students' knowledge base is expanded,” and “students' innovative thinking is stimulated,” respectively. This indicates that online piano teaching has significant differences in these three aspects, with online teaching outcomes being superior to traditional teaching.

Funding

This article is supported by the Scientific Research Project of Hunan Provincial Department of Education (Grant No. 24C0340).

About the Author

Li Lin (b. 1992, Changde, Hunan, China) holds a Ph.D. in Music Performance and serves as a Lecturer and Dance Teacher at the School of Music and Dance, Hunan University of Arts and Science. Her primary research focuses on music education and music performance.

Ping’an Zheng (b. 1988, Changde, Hunan, China) holds a Master’s degree in Computer Science and is the Director of the Digital Technology Office at China United Network Communications Group Co., Ltd. (Changde Branch). His research interests include ICT solutions and applications of artificial intelligence algorithms.

References

[1] Li, S., & Zhao, R. (2024). Establishment and application of diversified piano teaching system in colleges and universities from the perspective of artificial intelligence. J Appl Sci Eng, 27(7), 2813-2823.

[2] Tang, J. (2025). The role of artificial intelligence in improving the efficiency of piano online learning. Journal of Computational Methods in Sciences and Engineering, 25(2), 1504-1518.

[3] Zhai, Y., & Xu, C. (2022). The application of artificial intelligence-assisted computer on piano education. Comput. Aided. Des. Appl, 157-167.

[4] Hou, Y. (2022). Research on piano informatization teaching strategy based on deep learning. Mathematical Problems in Engineering, 2022(1), 5817752.

[5] Song, X., & Phokha, P. (2024). Applications Of Artificial Intelligence-Assisted Computing In “Piano Education”. Journal of Ecohumanism, 3(7), 1648-1659.

[6] Chen, H. (2021, September). Application of Piano Automatic Accompaniment System Based on Artificial Intelligence in Piano Enlightenment Education. In 2021 4th International Conference on Information Systems and Computer Aided Education (pp. 1351-1355).

[7] Li, C. (2022). Innovative application of the teaching mode of piano impromptu

accompaniment course under the perspective of" Internet+". Advances in Engineering Technology Research, 1(3), 161-161.

[8] Chen, Y., Chen, J. Y., & Zhang, A. P. (2021, October). Design of Mobile Teaching Platform for Vocal Piano Accompaniment Course Based on Feature Comparison. In International Conference on Advanced Hybrid Information Processing (pp. 430-442). Cham: Springer International Publishing.

[9] Li, Y. (2020). Application of computer-based auto accompaniment in music education. International Journal of Emerging Technologies in Learning (iJET), 15(6), 140-151.

[10] Mo, Y. (2022). Designing an automatic piano accompaniment system using artificial intelligence and sound pattern database. Mobile Information Systems, 2022(1), 7875818.

[11] Yuning, L. (2018). The Application of Introducing Piano Accompaniment into Ruan Xian Music Teaching. In Proceedings of 4th International Conference on Education, Management and Information Technology (ICEMIT 2018) (pp. 1478-1481).

[12] Zhou, D., & Nataliia, S. (2025). Generative Adversarial Network for Adaptive Piano Accompaniment. Systems and Soft Computing, 200289.

[13] Gao, M. (2020, April). Research and analysis of piano teaching based on intelligent interactive background. In Journal of Physics: Conference Series (Vol. 1533, No. 3). IOP Publishing.

[14] Wang, H., Zhang, X., & Iida, F. (2024). Human-robot cooperative piano playing with learning-based real-time music accompaniment. IEEE Transactions on Robotics.

[15] Jamal, I. M., & Kilic, E. (2021, December). EasyARPiano: piano teaching mobile app with augmented reality. In 2021 International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT) (pp. 66-71). IEEE.

[16] Yang Yaokun. (2021). Piano Performance and Music Automatic Notation Algorithm Teaching System Based on Artificial Intelligence. Mobile Information Systems,2021,

[17] Wei Cui. (2021). A Study of Piano Timbre Teaching in the Context of Artificial Intelligence Interaction. Computational Intelligence and Neuroscience, 2021, 4920250-4920250.

[18] Peng Liu. (2023). Application of virtual reality and multimedia integration in piano teaching of sound education major in colleges and universities. International Journal of Networking and Virtual Organisations, 28(2-4), 430-444.

[19] Lili Zhang. (2024). Research on Digital Application of MIDI Technology to Enable Piano Courses in Higher Vocational Colleges. Research and Commentary on Humanities and Arts, 2(5),

[20] Kong Qiuqiang, Li Bochen, Chen Jitong & Wang Yuxuan. (2022). GiantMIDI-Piano: A Large-Scale MIDI Dataset for Classical Piano Music. Transactions of the International Society for Music Information Retrieval, 5(1),87-98.