Improving the Interactive Accuracy of Intelligent Piano Teaching Using a PSO-SVM Optimized Fingering Recognition Model

Jian Zhang

Department of Music, Xi'an Shiyou University, Xi'an 710065, Shaanxi, China

Corresponding author: Jian Zhang (e-mail: 13051039320@163.com)

ABSTRACT Current intelligent piano teaching systems are limited by insufficient feature extraction and poorly optimized classification parameters, which reduces fingering-recognition accuracy and weakens human–machine interaction feedback. This paper proposes a high-precision fingering recognition method based on Particle Swarm Optimization and Support Vector Machine. Keystroke timing, force, and duration are first collected and normalized as discriminative feature vectors. A standard Particle Swarm Optimization algorithm is then designed with cross-validation accuracy as the fitness function to iteratively optimize the SVM kernel parameter γ and penalty factor C. The optimized classifier is trained on fingering samples and embedded into an intelligent piano teaching system for real-time recognition. Experiments on the PIG dataset, including 13,800 annotated fingering samples from ten trained pianists performing six graded piano pieces on a Yamaha U1 acoustic piano, show 91.2% recognition accuracy for two-finger fingering and an 87.0% F1-score for inter-finger repeated fingering. With an average response time below 67 ms, the system demonstrates high recognition accuracy and real-time interaction capability, providing a feasible technical path for intelligent piano instruction and short-duration motion-signal classification.

INDEX TERMS piano fingering recognition, particle swarm optimization, support vector machine, real-time classification, human-machine interaction

I. INTRODUCTION

AMID the ongoing development of intelligent music education, piano teaching systems are evolving from traditional models to more intelligent, human-computer interactive ones.

Fingering recognition, as a key component of instructional feedback, directly determines how the system evaluates a performer's technical movements, thereby influencing the effectiveness of personalized error correction and skill assessment [1,2]. Therefore, improving fingering recognition accuracy is of great significance for building an efficient interactive piano teaching system [3,4].

To improve the accuracy of human-computer interaction in local piano performance scenarios, this paper proposes applying the particle swarm optimization algorithm to the Support Vector Machine (SVM) parameter optimization process, constructing a global search strategy with cross-validation accuracy as the fitness function, and establishing a normalization and correlation control mechanism for three types of features: keystroke timing, keystroke force, and duration. This approach ensures that the discriminant tension of fingering differences is reflected at the feature input stage, providing separability support for subsequent classification boundaries. The method first collects and normalizes three types of low- dimensional dynamic features during the keystroke process. A particle swarm optimization strategy, with cross-validation accuracy as the fitness function, is then constructed. The SVM kernel parameter γ and penalty factor C are globally optimized in a continuous search space to obtain the optimal parameter combination with generalization capabilities. The optimized parameters are then used to train a classification model to recognize input fingering patterns. Finally, this recognition module is embedded in a piano teaching system, enabling online processing and classification of performance data. By applying a global modeling mechanism for performance differences within the parameter optimization layer, this method improves recognition accuracy and response time, providing a feasible path for building a highly interactive local piano teaching platform based on a standardized dataset recorded under controlled acoustic and hardware settings to ensure the stability and comparability of experimental results. In this study, a SVM is selected as the core classification model to ensure stable performance under a limited sample size and high inter-subject variability. Deep sequential architectures require large-scale temporal datasets and exhibit parameter sensitivity that may lead to unstable convergence with small-sample piano data. In contrast, SVM maintains robustness in nonlinear, low-dimensional feature spaces, and the use of particle swarm optimization ensures adaptive adjustment of kernel parameters to capture temporal variance within short-term keystroke patterns. This structure achieves both generalization stability and computational efficiency for real-time inference.

II. RELATED WORK

In intelligent piano teaching systems, the construction of the fingering recognition model is the core technical link for achieving efficient human-computer interaction. Its modeling method directly determines the accuracy and practicality of teaching feedback. Guan et al. [5] proposed a model based on pitch difference and fingering sequence matching. By integrating fingering transfer knowledge and prior rules, it improved the rationality of fingering prediction and solved the limitations of the traditional hidden Markov model in compact-scale processing. This idea emphasizes the complementarity of combining data-driven and rule-driven approaches in terms of modeling accuracy [6,7]. In intelligent music teaching systems, optimizing SVM to improve recognition accuracy and interaction efficiency has become a research focus. Ye [8] constructed an SVM model that integrates music emotion and instrument recognition based on the particle swarm optimization algorithm for vocal teaching scenarios. Its recognition accuracy and teaching interactivity were significantly enhanced, providing an optimization strategy for building highly responsive intelligent systems.

This strategy improves the adaptability of kernel function parameter adjustment and further leads to attempts to model complex physiological signals. Prabhakar et al. [9] proposed a hybrid kernel least squares SVM algorithm for gesture recognition tasks, applying local mean decomposition and fuzzy C-means clustering to electromyographic signal processing, achieving a classification accuracy of up to 98.5% and enhancing the generalization ability of SVM under multidimensional nonlinear data.

III. FINGERING RECOGNITION MODEL CONSTRUCTION

To achieve high-precision and high-generalization fingering recognition, the model design fully integrates feature processing, classifier construction, and optimization strategies, forming a system architecture with clear layers and tightly connected processes. Figure 1 shows the overall construction framework of the fingering recognition model. The modules maintain an upstream and downstream structural connection, covering the entire process from three-dimensional keystroke feature acquisition and normalization processing to SVM-based classification model design and particle swarm optimization algorithm parameter optimization, ultimately completing model training and system deployment.

A. FEATURE COLLECTION AND NORMALIZATION PROCESSING

During the feature acquisition and normalization phase, the system uses a high-speed sensor module to track the performance in fine-grained detail. It extracts three continuous features from the raw data: keystroke time t_k, keystroke force f_k, and duration d_k. These features form a three-dimensional feature vector $\mathbf{x}_k = [t_k, f_k, d_k]^T$ for each keystroke, thereby constructing a preliminary data representation for fingering recognition. The selection of keystroke timing, force, and duration is based on their direct mapping to biomechanical control variables that define finger motion precision in piano performance. Timing reflects temporal synchronization, force encodes dynamic control intensity, and duration expresses the stability of key depression. These three parameters together span the fundamental physical space of fingering action, allowing the recognition model to capture discriminative variance among gesture types without redundancy.

Sequence-level dependencies are implicitly preserved in the continuous temporal alignment of the collected samples, ensuring contextual consistency during classification. The keystroke time is calculated from the difference in timestamps between key trigger and release. Keystroke force is obtained by mapping the output voltage of the force sensor, and duration is directly measured by the length of time the key is held. To ensure uniform scaling and convergence stability of features across all dimensions during model training, a linear normalization strategy is used to map the original feature vectors to the interval [0, 1]. The transformation formula is:

$$x_k^{(i)\prime} = \frac{x_k^{(i)} - \min\left(x^{(i)}\right)}{\max\left(x^{(i)}\right) - \min\left(x^{(i)}\right)} \tag{1}$$

In formula (1), $x_k^{(i)}$ represents the i-th original feature component of the k-th data point in the sample, and $\min(x^{(i)})$ and $\max(x^{(i)})$ are the minimum and maximum values of the i-th feature dimension in the training set, respectively. This normalization process is completed uniformly during the data preprocessing stage to eliminate the offset interference caused by different dimensions on the discrimination boundary of the classification model.

After structuring, the dataset forms a standard three-dimensional feature matrix $X \in \mathbb{R}^{n\times 3}$, where n represents the total number of keystroke samples. To capture the dynamic variability caused by differences in keystroke habits and styles between performers, the system maintains data collection continuity throughout the entire repertoire, ensuring temporal consistency and behavioral continuity of each sample in the actual context. To further verify the discriminative power of features in the fingering recognition task, Principal Component Analysis (PCA) is used to visualize the separability of features, projecting the processed high-dimensional samples onto a two-dimensional plane. Figure 2 shows the distribution structure of the

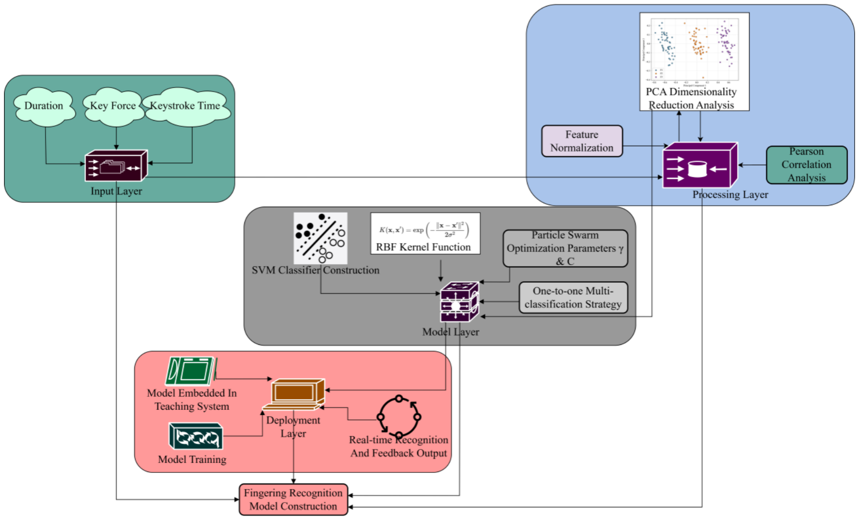

FIGURE 1. Framework diagram of the fingering recognition model construction

normalized three- dimensional keystroke features in the principal component space, which is used to analyze the separability of different fingerings in the low-dimensional embedding space. F1 represents the thumb-in-span fingering, F2 represents the cross-finger key-switching fingering, and F3 represents the two-finger continuous fingering. They show a relatively clear cluster distribution in the projected space, indicating that the three features of keystroke time, keystroke force, and duration have strong discriminative ability in the classification task. The principal component axis carries the linear combination with the largest variance in the original features, and its spatial structure reflects the overall differences in fingering in terms of timing rhythm, force control, and keystroke stability. The boundary intervals between the fingering categories are formed in this space, indicating that the normalized keystroke features provide good discriminative support for the classification model.

The standard feature input obtained after normalization is in the form of $X' = \{x'_1, x'_2, \ldots, x'_n\}$, which serves as direct input to the SVM training module. To enhance the robustness of the original features for subsequent classification model adaptation, the feature extraction process incorporates soft trigger delay control and a resampling mechanism to mitigate the impact of heterodyne noise on the accuracy of force and duration measurements. To mitigate the impact of dimensional correlations in the input space on the model boundary construction, the Pearson correlation coefficient between the three-dimensional features is calculated and the sample order is optimized accordingly to ensure a stable distribution of training data during the input phase.

B. SVM CLASSIFIER DESIGN

The adoption of a classical SVM framework rather than a deep sequential network reflects the data scale and temporal granularity characteristics of piano fingering tasks. Each keystroke action is treated as a quasi- independent event with short-duration dynamics, where contextual dependencies are represented implicitly in the continuity of feature sampling. This modeling assumption aligns with the design objective of real-time response, as SVM inference latency remains consistently low compared to multi-layer recurrent or convolutional models. In the fingering recognition model, the classifier structure is composed of a SVM and adopts a binary classification form with a radial basis kernel function to address the nonlinear separability of fingering samples in the feature space. The normalized three-dimensional feature vector is input into the SVM model, and a decision function is constructed to find the optimal hyperplane boundary in the high- dimensional mapping space to achieve the mapping relationship between keystroke behavior and fingering labels. The kernel function form is selected as a Gaussian kernel, which is expressed as:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right) \tag{2}$$

In formula (2), $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^3$ represent the normalized input sample pair, and $\gamma > 0$ is the bandwidth control parameter of the kernel function, which is used to adjust the mapping scale of the distance between samples in the feature space, thereby affecting the nonlinear complexity of the classification boundary.

When building the recognition model, to address the multi-class fingering recognition task, the basic binary classification architecture is expanded to a multi-classification system using a one-to-one strategy. Assuming the training set contains M fingering labels, a total of $\frac{M(M-1)}{2}$ basic classifiers are constructed. A voting decision mechanism is used to fuse the outputs of each sub-classifier to generate the final predicted label. This mechanism maintains the flexibility of the classification boundaries in the local space, enhancing the model's adaptability to local differences in keystroke style.

Model training constructs the support vector boundary by minimizing the objective function with a penalty term. Its optimization goal is:

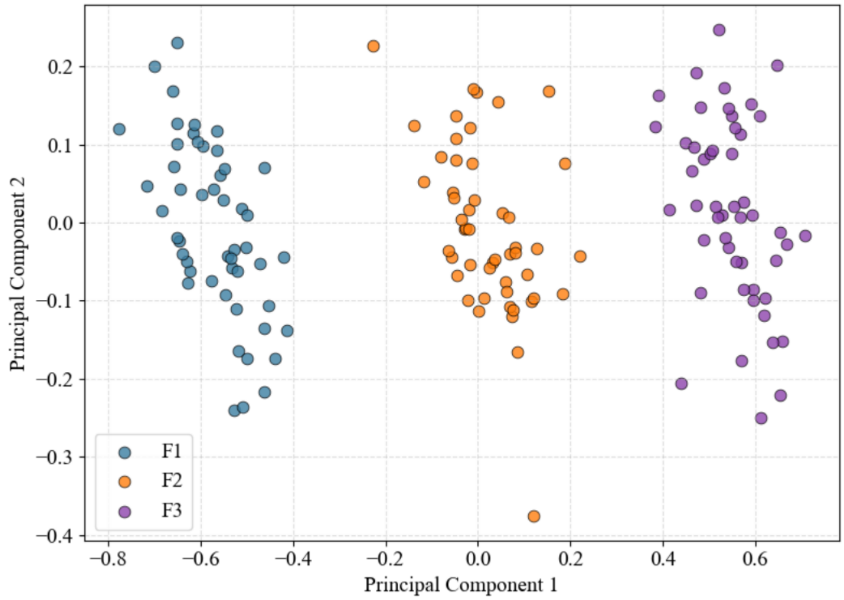

FIGURE 2. Visualization of the inter-class distribution of normalized keystroke features in PCA space

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \tag{3}$$

$$\text{subject to } y_i\left(\mathbf{w}^T\phi(\mathbf{x}_i)+b\right) \geq 1-\xi_i, \quad \xi_i \geq 0 \tag{4}$$

In formulas (3) and (4), $\phi(\cdot)$ is the nonlinear mapping function, $\mathbf{w}$ is the hyperplane normal vector, b is the bias term, ξ_i represents the slack variable, which is used to accommodate inseparable samples, and C is the penalty factor, which controls the trade-off between error tolerance and margin maximization.

C. PARTICLE SWARM PARAMETER OPTIMIZATION STRATEGY

When selecting the kernel parameter γ and penalty factor C for a SVM, traditional fixed strategies and grid search methods cannot meet the model's requirements for optimal parameter matching in the high- dimensional, nonlinear fingering feature space, easily leading to underfitting or overfitting of the classification boundary. To obtain the parameter combination with the highest recognition performance in cross-validation of the training set, a standard particle swarm optimization algorithm is employed to perform a global optimization search within a continuous search space. A particle swarm structure is constructed, encompassing position, velocity, individual extrema, and global extrema. The parameter solution encoded for each particle group is iteratively updated, using the SVM's classification accuracy under cross-validation as the fitness function. The fingering dataset used for optimization maintains a nearly balanced distribution across six classes, with sample proportions varying within a 1.0% margin, ensuring that the optimization objective based on accuracy remains representative of global classification performance. During each k-fold iteration, both accuracy and F1-score are monitored to verify convergence consistency and prevent bias toward majority classes. The accuracy metric is retained as the fitness criterion due to its stability under balanced conditions and its direct interpretability for performance evaluation in cross-validation.

Individual particles are encoded as two-dimensional real vectors, with each particle corresponding to a parameter combination $\mathbf{x} = [\gamma, C]$. The velocity and position update formulas are:

$$\mathbf{v}_i^{(t+1)} = \omega\mathbf{v}_i^{(t)} + c_1r_1\left(\mathbf{p}_i - \mathbf{x}_i^{(t)}\right) + c_2r_2\left(\mathbf{g} - \mathbf{x}_i^{(t)}\right) \tag{5}$$

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \mathbf{v}_i^{(t+1)} \tag{6}$$

ω represents the inertia factor, $\mathbf{p}_i$ is the optimal solution for each particle, $\mathbf{g}$ is the global optimal solution for the current population, c_1 and c_2 are the individual and group learning factors, respectively, and r_1 and r_2 are random numbers in the range [0, 1] used to apply search perturbations.

In the fitness function design, the objective is to maximize the average recognition accuracy based on k-fold cross-validation, denoted as:

$$f(\gamma, C) = \frac{1}{k}\sum_{i=1}^{k} Acc_i(\gamma, C) \tag{7}$$

In formula (7), Acc_i represents the accuracy of the SVM model trained with the parameter combination (γ, C) in the i-th fold cross-validation on the validation set. The global maximum of this function corresponds to the optimal position in the parameter search space, guiding the particles to converge toward the direction with the best accuracy.

Considering the high-dimensional non-convex nature of the parameter space, a logarithmic scale transformation is used when setting the search boundary, mapping the original search interval into a continuous logarithmic space suitable for particle motion. To intuitively present the population search behavior and convergence trend during the optimization process, a visualization of the particle path in the parameter search space is constructed, as shown in Figure 3:

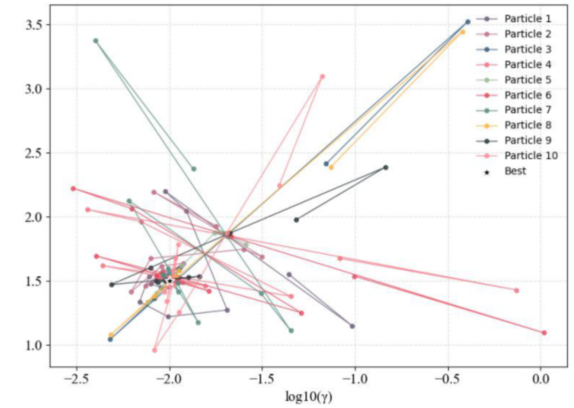

FIGURE 3. Schematic diagram of particle convergence path in parameter search space

Figure 3 illustrates the convergence process of the particle swarm optimization algorithm in a SVM parameter search task. Each curve represents the search path of a particle in parameter space. Starting from an initial random position, the particle gradually adjusts its position to improve its fitness. During the iteration process, guided by both individual experience and the swarm's optimality, it continuously converges toward a high- performance region. The asterisk at the end of the trajectory represents the final converged global optimal parameter combination. The particle paths show a clear clustering trend, demonstrating the algorithm's strong global optimization capability and convergence stability.

Table 1 lists the statistical results of the fitness mean, standard deviation, and optimal accuracy at key rounds, reflecting the convergence dynamics and optimization effect of the particle swarm during the model construction phase.

TABLE 1. Fitness statistics during particle swarm optimization

Iteration Round	Average Fitness (Accuracy)	Standard Deviation	Optimal Fitness (Accuracy)
Round 10	84.63%	3.21%	88.12%
Round 20	86.75%	2.15%	89.63%
Round 40	88.40%	1.47%	90.75%
Round 60	89.25%	0.96%	91.01%
Round 100	89.83%	0.58%	91.20%

As the iteration rounds progress, the particle swarm as a whole gathers toward the optimal solution area and finally achieves the optimal accuracy of 91.20% in the 100th round, fully verifying the stability and optimization performance of the training process.

D. OPTIMIZING RECOGNITION MODEL TRAINING

After completing the parameter optimization, the normalized three-dimensional feature vector and the optimal kernel function parameter γ and penalty factor C obtained by the particle swarm algorithm are input into the SVM model to perform full sample training and build a fingering recognition classifier with the ability to adapt to cross-player differences. The feature vector consists of keystroke time, keystroke force, and duration, and is mapped to the normalized space as the input tensor $X \in \mathbb{R}^{n\times 3}$, where n is the number of training samples. The label vector is set to $y \in \{-1, 1\}^n$, indicating different categories of fingering samples. The Radial Basis Function (RBF) kernel function is used to define the nonlinear mapping relationship between input samples, and its kernel function form is:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2\right) \tag{8}$$

In formula (8), $\mathbf{x}_i$ and $\mathbf{x}_j$ are any two sample feature vectors, and γ is the kernel parameter after the particle swarm algorithm is optimized. The SVM training process is transformed into a convex quadratic optimization problem, and the objective function is defined as:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{n}\alpha_i \tag{9}$$

The constraints are $0 \le \alpha_i \le C$ and $\sum_{i=1}^{n}\alpha_i y_i = 0$, where α_i is the Lagrange multiplier and C is the penalty factor. This optimization process is performed by an in-library optimizer, given the parameters γ and C. The output is a set of support vectors and decision function coefficients, achieving nonlinear boundary demarcation for each category of fingering samples in the high-dimensional feature space.

To verify the convergence trend of model performance and the stability of boundary construction during training, the loss function value, the number of support vectors, and their changes in proportion are recorded for each round of training. Table 2 shows the training metric statistics for key rounds, reflecting the model's optimization dynamics and structural compression capabilities during continuous iteration.

TABLE 2. Statistics of key indicators during SVM training

Training Rounds	Loss Function Value (L)	Number of Support Vectors	Percentage of Support Vectors
Round 1	2.732	276	40.3%
Round 5	1.985	254	37.1%
Round 10	1.322	238	34.7%
Round 20	0.886	221	32.3%
Round 30	0.511	215	31.5%

During the training process, the loss function shows a monotonically decreasing trend, and the number of support vectors gradually decreases, indicating that the model boundary gradually converges during the optimization process and has compression characteristics, which helps to improve the robustness and generalization performance of classification decisions.

E. MODULE EMBEDDING AND ONLINE RECOGNITION PROCESS

The optimized trained recognition model is deployed in a frozen form to the main control program of the teaching system. A lightweight model loading mechanism is used in combination with an embedded cache structure so that the model parameters and support vectors do not need to be frequently loaded into memory. During the system initialization phase, the kernel function type, parameter γ, and penalty factor C are loaded, and the support vector set and decision function bias term b are generated after training. The performance data acquisition module continuously receives three indicators: keystroke timing, keystroke force, and duration, and constructs a three-dimensional feature vector $\mathbf{x}_t \in \mathbb{R}^3$ after normalization, which is input into the model inference engine in real time. Figure 4 [10] shows the interactive interface of the fingering recognition module in the intelligent piano teaching system, which intuitively presents the workflow of real-time inference and feedback of the model.

In the recognition process, the input feature vector is transformed by the RBF kernel function, and the kernel mapping value between it and the i-th sample $\mathbf{x}_i$ in the support vector set is calculated. The model decision function output is calculated by the following function:

$$f(\mathbf{x}_t) = \sum_{i=1}^{n} \alpha_i y_i \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_t\|^2\right) + b \qquad (10)$$

In formula (10), α_i is the Lagrange multiplier corresponding to the i-th support vector, y_i is its category label, γ is the model kernel function parameter, and b is the bias term. The system determines the fingering category of the current input based on the sign of the decision function. The classification result is bound to the signal timestamp and sent to the feedback management thread.

To maintain inference stability and reduce runtime latency, the system uses a sliding window cache mechanism to batch process continuous keystroke data. The window size is set to w, and at each time t, the feature set $\{\mathbf{x}_{t-w+1}, \ldots, \mathbf{x}_t\}$ is extracted. W groups of output labels are obtained through batch inference, and the final recognition result is determined by majority voting within the time window:

$$\hat{y}_t = \text{mode}\left(\{f(\mathbf{x}_{t-w+1}), \ldots, f(\mathbf{x}_t)\}\right) \qquad (11)$$

After the model is embedded, no further training or parameter adjustment is required. All parameters are fixed during the training phase and are coupled to the teaching system's scoring and feedback mechanism via a standard interface. When a new keystroke event is detected, the system immediately completes feature acquisition, normalization, inference output, and result feedback, forming a complete real-time interactive closed loop. The fingering prompts, practice feedback, and performance correction modules use the recognition output. The interactive module of the teaching system integrates a dual-channel feedback interface. Visual guidance is presented through on-screen key illumination and finger position diagrams that highlight correct fingering sequences in synchronization with the performance timeline. Auditory cues are generated through a real-time correction tone that distinguishes accurate keystrokes from erroneous ones, while a scoring panel dynamically updates precision and timing deviations. This interaction process establishes a continuous perception–action cycle, linking model recognition outcomes to perceptible teaching feedback for the performer.

IV. MODEL DEPLOYMENT AND EVALUATION

A. EXPERIMENTAL PLATFORM AND OPERATING ENVIRONMENT

The experiments are conducted on a local computing platform with real-time data processing capabilities.

The host computer is equipped with an Intel i7-11700F processor, 16 GB of DDR4 memory, and an NVIDIA RTX 3060 Graphics Processing Unit. The operating system, Windows 10 x64, ensures that keystroke data and recognition results are synchronously transmitted and inferred within milliseconds.

To further quantify the operational efficiency of the deployed recognition module, the system evaluated the response latency performance of online inference under multi-threaded configurations and different sliding window sizes. During the recognition process, the input feature vector is transformed by the RBF kernel function, and its kernel mapping value with the i-th sample in the support vector set is calculated. The fingering classification label of the current input is output through the decision function, and the classification result is bound to the timestamp and sent to the feedback management thread. To verify the responsiveness of the recognition module during system operation, it calculates inference latency under multi-threaded configurations and varying sliding window sizes, observing the system's interactive performance under high- frequency input conditions. Table 3 lists the average and maximum response times for typical parameter combinations, reflecting the model's real-time performance in a multi-task environment.

TABLE 3. Statistics of the average response delay of the online inference of the finger recognition module

Number of Threads	Window Size (w)	Average Response Time (ms)	Maximum Response Time (ms)
1	5	71.4	83.2
2	5	65.1	74.9
2	10	61.7	71.3
4	10	58.4	67.0
4	20	55.9	66.7

The results show that multi-threading and increasing the

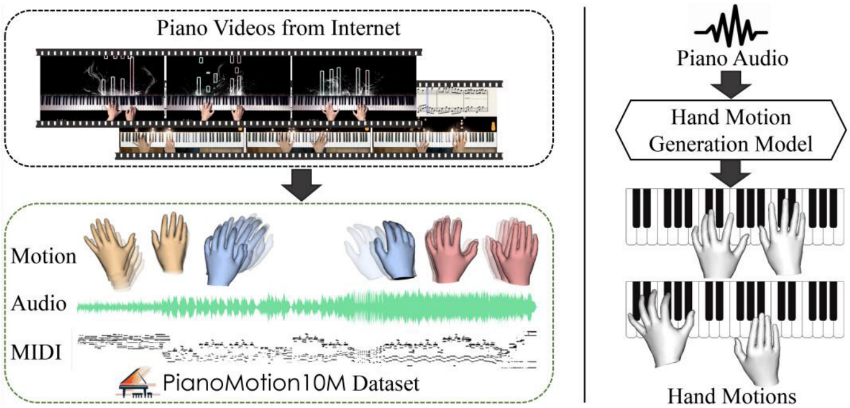

FIGURE 4. Schematic diagram of piano interaction effects in the intelligent piano teaching system

window size configuration have an inhibitory effect on response delay. The system achieves an average response time of 55.9 ms under the conditions of four threads and a window size of twenty, and the overall maximum delay does not exceed 67 ms, meeting the real-time requirements of human-computer interaction in teaching scenarios.

B. DATASET CONSTRUCTION AND ANNOTATION PROCESS

The PIG dataset used in the experiment was constructed by the Music and Intelligent Engineering Experimental Center of an art college, integrating ten pianists' performances on a Yamaha U1 piano and expanded from an existing benchmark dataset. This study was approved by the Ethics Committee of Xi'an Shiyou University. All eligible participants signed an informed consent form. The data collection was organized and implemented by the Music and Intelligent Engineering Experimental Center of an art college.

It encompasses multiple standardized recording scenarios and high-fidelity performance equipment to ensure the integrity and temporal consistency of the original data. The dataset covers typical local piano repertoire and diverse fingering movements. To support the model's generalization analysis during the classification phase, all sampled data are manually annotated and corrected. Labels are assigned to six standard fingering types and aligned with keystrokes along the timeline. After data processing, the total number of samples reaches 13,800, each consisting of three-dimensional keystroke features and fingering labels. The sample label distribution is kept approximately balanced across the six fingering categories to reduce error accumulation caused by class imbalance during subsequent model training. Table 4 summarizes the sample count for each fingering label. The average keystroke duration for each category is also calculated to assist in analyzing rhythmic differences across different movement patterns.

TABLE 4. Distribution of sample numbers of different fingering categories in the PIG dataset

Fingering Type	Number of Samples (Groups)	Percentage of Total Samples (%)	Average Keystroke Duration (ms)
Thumb inward span	2,280	16.5	174.2
Cross-finger key change	2,325	16.8	165.7
Two-finger combo	2,360	17.1	152.4
Inter-finger repetition	2,305	16.7	158.6
Five fingers naturally relaxed	2,220	16.1	181.9
Staccato jump	2,310	16.7	139.3

The balanced distribution of sample numbers in each category helps alleviate the category shift problem during training. The numerical difference in duration also reflects the inherent distinction of fingering in time control, providing a stable input feature basis for the classification model. The proportion of samples across the six fingering categories remains within a narrow deviation range, ensuring that no category exerts dominant influence during the optimization phase. This balanced composition supports the validity of using accuracy-based metrics in the PSO fitness evaluation process.

C. MODEL TRAINING PARAMETER SETTING

The SVM recognition model training phase uses three-dimensional normalized feature vectors as input.

Parameter setting is driven by a particle swarm global optimization algorithm, which searches for optimality in the continuous space of kernel parameter γ and penalty factor C. The optimization objective is the average recognition accuracy under ten-fold cross-validation. The search space is logarithmically scaled, with kernel parameters set between 10^{-4} and 10^{1}, and penalty factors set between 10^{-1} and 10^{3}, matching the distribution density of the initial particle solution and boundary convergence requirements. The particle swarm size is fixed at 30, with a maximum number of iterations of 100. The inertia factor w is set to 0.5, and the individual and group learning factors are each set to 1.5. Accuracy evaluation and fitness feedback are performed simultaneously on the same platform during training.

V. RESULTS ANALYSIS

A. COMPARATIVE ANALYSIS OF RECOGNITION ACCURACY

To evaluate the accuracy of the fingering recognition model and verify the substantial performance improvements achieved through optimization strategies, this experiment focuses on the performance of different recognition models on typical fingering movements. Model 1 is a standard Base-SVM without parameter adjustment, which has a fixed structure and limited generalization capabilities. Model 2 is a Grid- SVM using traditional grid search for parameter optimization, which shows some improvement in local optimal search compared to the previous model. Model 3 is a PSO-SVM using a particle swarm optimization algorithm for global parameter optimization, which offers the advantages of both dynamic optimization capabilities and stability. Considering the significant differences in fingering movements in piano playing, six representative fingering types are selected: thumb-in (F1), finger-crossing (F2), two-finger combo (F3), inter- finger repetition (F4), five-finger natural relaxation (F5), and staccato jump (F6). A comparative analysis is conducted, covering different characteristics such as control difficulty, finger span, and rhythm changes. By presenting three performance indicators—accuracy, F1-score, and recall—in multiple dimensions, the performance differences of each model in complex fingering recognition scenarios are systematically analyzed, as shown in Figure 5:

PSO-SVM outperforms the other models across all fingering recognition tasks, achieving accuracies of 90.7% and 91.2% for the thumb-in and two-finger tap, respectively—approximately 8 percentage points higher than the Base-SVM. This performance improvement stems from the global optimality of the parameter combination achieved by particle swarm optimization, which enables the model to maintain greater recognition robustness across different motion forms. In terms of F1-score, PSO-SVM achieves 87.0% for the high-confusion fingering inter-finger repetition, significantly exceeding the 82.1% achieved by the Grid-SVM.

This demonstrates that the optimization strategy establishes a more stable classification boundary across imbalanced samples, reducing the precision-recall bias. In the recall dimension, PSO-SVM achieves 84.9% for the weak-signal staccato jump, significantly exceeding the 72.6% achieved by the Base-SVM, reflecting its greater responsiveness in low-intensity input scenarios. Overall, the results demonstrate that the particle swarm optimization strategy significantly improves fingering recognition performance. The stability of high accuracy across six fingering categories further confirms that the three-dimensional feature space constructed from timing, force, and duration provides sufficient discriminatory representation for the classification task. This indicates that extending to higher-dimensional feature sets would not yield proportional gains under the current modeling structure.

B. VERIFICATION OF PARAMETER OPTIMIZATION CONVERGENCE

To comprehensively evaluate the convergence performance and search efficiency of the PSO algorithm in the high-dimensional kernel parameter space, this section conducts a visual analysis from two aspects: fitness evolution trend and parameter value distribution. Figure 5(a) focuses on the dynamic changes of the global optimum, swarm mean, and worst fitness during the particle swarm iteration process, and uses standard deviation shading to indicate convergence stability, showing the key stages of model performance improvement. Figure 5(b) shows the distribution trajectory of particles in the logarithmic scale C and γ search space. The color depth corresponds to its cross-validation accuracy, reflecting the density of the optimal parameter region and the search characteristics of the convergence path. Figure 6 shows the performance evolution and search characteristics of the fingering recognition model parameter optimization process.

In Figure 6(a), the global optimal curve rises rapidly in the early stages, then stabilizes after approximately 40 iterations, ultimately settling in a relatively high fitness range, indicating that the model has achieved primary convergence during this phase. The mean curve approaches the optimal curve, indicating that the particle population as a whole is moving toward the optimal solution, and the gradual convergence of the standard deviation reflects enhanced search stability. The lag in the worst fitness curve reveals that local particles are still in the non-optimal range, indicating a certain degree of complexity in the search space. Figure 6(b) shows that high-accuracy particles are concentrated in the region where $\log_{10}(C)$ lies between 1 and 2 and $\log_{10}(\gamma)$ lies between -3 and -1, exhibiting a significant shrinking trend. This suggests that the PSO has excellent ability to identify the optimal range of kernel parameters and exhibits convergence bias. Overall, the designed PSO optimization strategy demonstrates excellent global optimization capabilities and parameter convergence stability in high-dimensional continuous search spaces.

C. MODEL IDENTIFICATION STABILITY AND ROBUSTNESS ANALYSIS

To thoroughly evaluate the stability and generalization capabilities of fingering recognition models under complex task conditions, this section conducts a comparative analysis of three models: Model 1 is a vanilla SVM with no parameter tuning, whose parameters are set to default and generalization is limited; Model 2 is a Grid-SVM that uses a grid search approach to optimize hyperparameters, achieving local fine-tuning within the parameter space; and Model 3 is a PSO-SVM with a particle swarm optimization algorithm. This algorithm uses cross-validation accuracy as its fitness function to optimize kernel parameters and penalty factors within a global continuous space, thus enabling adaptive performance variability. Six levels of fingering complexity are set, ranging from basic single-finger

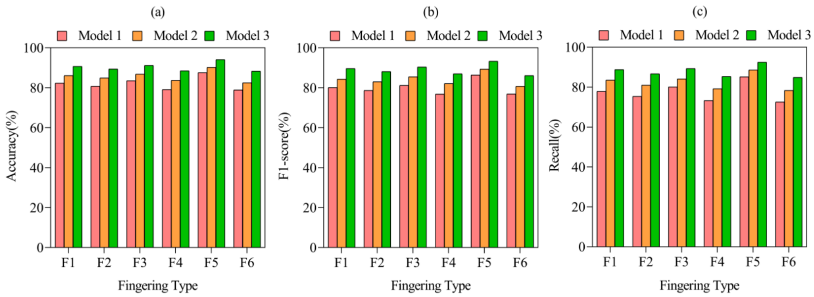

FIGURE 5. Comparative analysis of the multi-metric performance of three recognition models under different fingering techniques. (a) Accuracy, (b) F1-score, (c) Recall

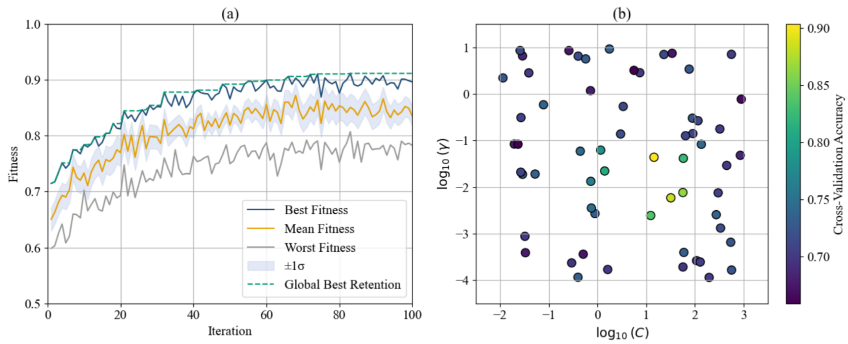

FIGURE 6. Fitness evolution and parameter search trajectory during parameter optimization of the fingering recognition model. (a) Fitness value change curve during PSO optimization, (b) Particle distribution and accuracy mapping in parameter search space

keystrokes to advanced, complex hand-overs, encompassing typical performance scenarios, from local linearity to inter-key fingering jumps. Six levels of perturbation are set, from no perturbation to a $\pm 10\%$ error range, to simulate velocity offsets, keystroke timing drift, and duration fluctuations found in real-world input. Generalization tests are conducted using data from ten different performers to examine the model's adaptability across a diverse range of subjects.

Figure 7 comprehensively illustrates the model's stability under increasing complexity, input variation, and data subject changes.

PSO-SVM consistently outperforms the other models in six complex fingering tasks. Its recognition accuracy for complex fingering changes is 83.8%, a 9.6% improvement over the untuned SVM, demonstrating its ability to adapt parameters to challenging performances. Perturbation analysis shows that keystroke force error has the greatest impact on the model among the three types of perturbations. Under a $\pm 10\%$ perturbation, keystroke force perturbation shows the largest decrease, with recognition accuracy dropping from 91.5% to 82.4%, indicating that this type of perturbation has a more significant impact on the system's overall recognition boundaries. In a cross-generalization evaluation of ten performers, the PSO-SVM's accuracy ranges from 86.3% to 90.1%, reflecting its stable performance across performers of different styles.

D. FEEDBACK RESPONSE TIME STATISTICS

To gain a deeper understanding of the dynamic response time characteristics of different models under multidimensional input conditions, this paper constructs three-dimensional response time surfaces for three representative algorithms in the dual-variable space of simulated keystroke frequency and velocity. Model 1 is an SVM, which exhibits excellent classification boundary discrimination performance in small-sample, high- dimensional spaces; Model 2 is a grid-search optimized Grid-SVM, which improves response generalization through systematic hyperparameter search; and Model 3 is a PSO-SVM, which further enhances the model's stable adaptability to input perturbations through a global heuristic search. In terms of input dimension design, frequency sampling covers six common rhythm density points, ranging from 1 Hz to 10 Hz; velocity covers five typical MIDI velocity ranges, encompassing performance levels from pp (pianissimo, 20) to ff (fortissimo, 110). Figure 8 illustrates the coupling relationship between frequency, velocity, and response time.

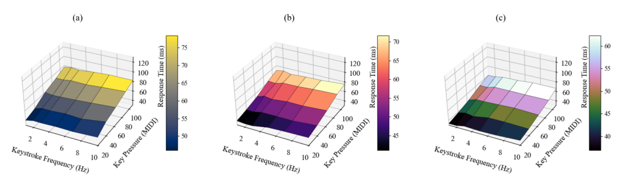

FIGURE 8. 3D response surface plots of response time for different models under different keystroke frequencies and forces. (a) SVM, (b) Grid-SVM, (c) PSO-SVM

As can be observed from Figure 8, the response time of the three models generally increases with the increase of the frequency and force dual variables, but there are

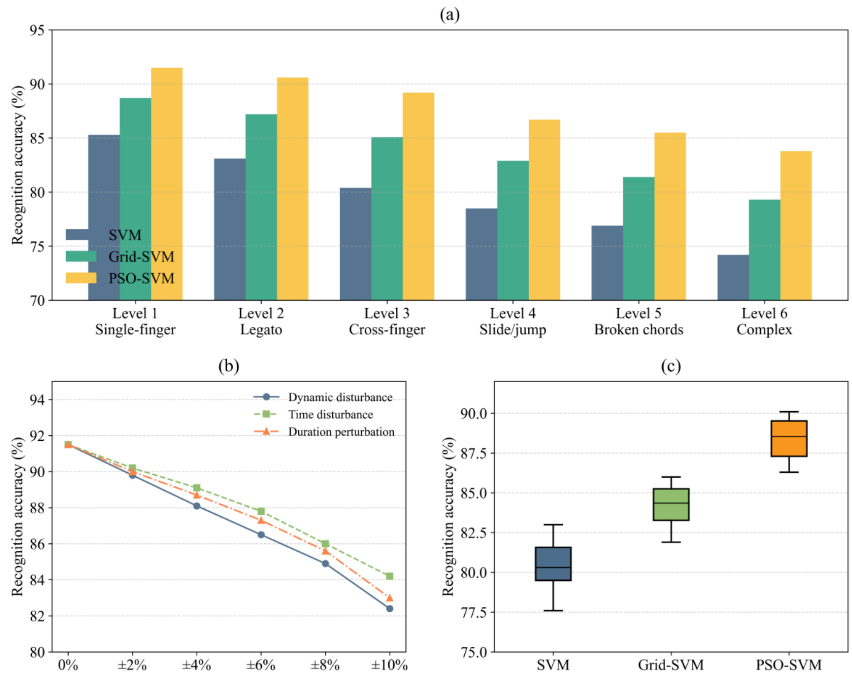

FIGURE 7. Comparison of robustness and generalization performance of fingering recognition models. (a) Model accuracy performance at different fingering complexities, (b) Model robustness trends at different perturbation levels, (c) Distribution of recognition accuracy of the three models across ten performers

significant differences in the rate and amplitude of increase. The response time of SVM in the high-frequency and strong pressure interval increases significantly, with the maximum value close to 83 ms, indicating that it is more sensitive to extreme input conditions. The overall surface of Grid-SVM is smooth, the response time interval is relatively convergent, and the maximum response is about 75 ms, indicating that its parameter optimization suppresses the discreteness of the response in the complex input space. The PSO-SVM response surface has the lowest inclination, the global fluctuation is controlled, and the response time is always less than 67 ms, showing good delay stability. Considering the performance of the system under real-time requirements, PSO-SVM shows stronger response constraint capabilities in multivariable interaction scenarios. The response time of this method is controlled within 67 ms, meeting the timeliness requirements of high-density interactive applications.

E. ANALYSIS OF MODEL ADAPTABILITY UNDER DIFFERENT TRACKS

To explore the performance of the PSO optimization strategy in the generalization of fingering recognition, this experiment selected six piano pieces with obvious differences in style and technical level and constructed a heat map of the SVM recognition model with a parameter dimension of 7×7 to examine the matching effect of the kernel parameter γ and the penalty factor C under different fingering structures. Minuet has a regular structure and clear rhythm, making it suitable for testing the model's training adequacy under stable input. Colorful Clouds Chasing the Moon features rapid legato and ornamentation variations, reflecting the need to identify detailed fingerings. Fisherman's Song at Dusk has a free rhythm and a wide melody span, demonstrating strong dynamic fingering variation. Defending the Yellow River emphasizes rhythmic intensity and accent control, relying heavily on dynamics and duration. Jasmine Flower has strong melodic continuity and frequent fingering across keys, requiring the model to be able to stably respond to strategy switching.

Dream Wedding has dense chords and complex rhythms, making it suitable for testing the model's discrimination ability under complex performance movements. By constructing a search grid with $\gamma \in [0.001, 10]$ and $C \in [0.1, 500]$ and combining it with the PSO global optimization strategy, the classification accuracy of the model for each piece is calculated, as shown in Figure 9.

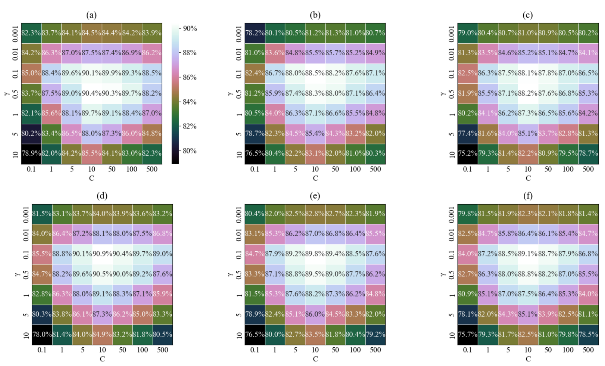

FIGURE 9. Heatmap of fingering recognition accuracy for each piece under different PSO parameter combinations. (a) Minuet, (b) Colorful Clouds Chasing the Moon, (c) Fisherman's Song at Dusk, (d) Defending the Yellow River, (e) Jasmine Flower, (f) Wedding in a Dream

For highly complex pieces, accuracy generally peaked in the range of $\gamma = 0.1$ to 0.5 and $C = 10$ to 100. Dream Wedding achieves the highest recognition accuracy of 89.1% for $\gamma = 0.1$ and $C = 10$. However, accuracy dropped significantly when γ increased to 10 or C increased to 500, indicating a tendency for the model to overfit. Jasmine Flower achieves an accuracy of 89.8% within the same parameter range, with a smaller fluctuation range, indicating that its fingering structure is stable and adaptable to model training. Defending the Yellow River and Minuet maintain accuracy around 90% at medium-to-high C values, demonstrating good convergence for input with accented and regular rhythms. In comparison, Colorful Clouds Chasing the Moon and Fisherman's Song at Dusk perform slightly worse in the optimal range, with accuracy concentrated between 86% and 88.5%, primarily due to the training difficulty associated with frequent fingering changes and high rhythmic tension. Overall, PSO guides the model to achieve strong cross-piece generalization within this parameter range, and the optimal parameter combination shows consistent trends across different pieces.

VI. CONCLUSION

This paper addresses the critical issue of insufficient fingering recognition accuracy in intelligent piano instruction and proposes a high-precision fingering recognition method based on PSO-SVM.

By constructing a three-dimensional feature vector of keystroke time, keystroke force, and duration, and using it as input after normalization, this method effectively models and classifies performance characteristics. The system uses cross-validation accuracy as a fitness function, guiding a particle swarm to perform a global optimization search over the continuous spaces of the SVM kernel parameter γ and the penalty factor C, thereby improving the model's generalization and stability. Experimental results show that the PSO-SVM model achieves 91.2% accuracy in recognizing the two-finger tapping fingering, an 87.0% F1- score in the inter-finger repetition fingering with high confusion, and 84.9% recall under low-intensity signal conditions such as staccato jumps. The overall performance is better than that of the Grid-SVM and Base- SVM models. This method has been successfully embedded in the main control module of the piano teaching system. While maintaining the classification accuracy, the average response time is controlled within 67 ms to meet the needs of real-time interaction. Based on the support vector mechanism optimized by PSO, a recognition path with adaptive capabilities and cross-player generalization capabilities is constructed, which improves the model's stable adaptability to complex playing styles and provides reliable technical support and application value for the realization of intelligent piano teaching with efficient human-computer interaction. The teaching system's interactive mechanism directly couples the recognition results with multimodal user feedback, ensuring that learners perceive fingering errors in real time through both visual and auditory channels. This design enhances the interpretability of recognition accuracy improvements in the practical teaching environment.

AUTHOR CONTRIBUTIONS

Jian Zhang designed, collected and analyzed the data, and drafted the manuscript. Jian Zhang conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Jian Zhang participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

This study was approved by the Ethics Committee of Xi'an Shiyou University. All eligible participants signed an informed consent form, and responses were collected anonymously. No personally identifiable information was stored.

REFERENCES

[1] H. Liu, "Quantitative Analysis and AI Assessment of Piano Performance Techniques," Applied Mathematics and Nonlinear Sciences, vol. 10, no. 1, pp. 1-16, 2025, doi: 10.2478/amns-2025-0231.

[2] H. Pan and W. Wu, "Online learning to play the piano: perspectives and achievements/Aprender a tocar el piano en línea: perspectivas y logros," Culture and Education, vol. 36, no. 2, pp. 370-393, 2024, doi: 10.1177/11356405241259599.

[3] Y. Wang, "Research on Artificial Intelligence and Piano Playing Technology in Colleges and Universities Based on 5g Big Data Networks," Applied Mathematics and Nonlinear Sciences, vol. 9, no. 1, pp. 1-16, 2024, doi: 10.2478/amns-2024-1881.

[4] V. Phanichraksaphong and W. H. Tsai, "Automatic evaluation of piano performances for STEAM education," Applied Sciences, vol. 11, no. 24, pp. 11783-11805, 2021, doi: 10.3390/app112411783.

[5] X. Guan, H. Zhao, and Q. Li, "Estimation of playable piano fingering by pitch-difference fingering match model," EURASIP Journal on Audio, Speech, and Music Processing, vol. 2022, no. 1, pp. 7-20, 2022, doi: 10.1186/s13636-022-00237-8.

[6] T. Oku and S. Furuya, "Noncontact and high-precision sensing system for piano keys identified fingerprints of virtuosity," Sensors, vol. 22, no. 13, pp. 4891-4902, 2022, doi: 10.3390/s22134891.

[7] N. Srivatsan and T. Berg-Kirkpatrick, "Checklist models for improved output fluency in piano fingering prediction," arXiv preprint arXiv, vol. 22, no. 09, Art. no. 05622, 2022, doi: 10.48550/arXiv.2209.05622.

[8] J. Ye, "Research on the application of vocal music teaching methods and techniques in colleges and universities based on proportional integral differentiation algorithm," Applied Mathematics and Nonlinear Sciences, vol. 9, no. 1, pp. 1–21, 2024, doi: 10.2478/amns-2024-1327.

[9] S. K. Prabhakar and D. O. Won, “Efficient strategies for finger movement classification using surface electromyogram signals,” Frontiers in neuroscience, vol. 17, Art. no. 1168112, 2023, doi: 10.3389/fnins.2023.1168112.
[10] Q. Gan, S. Wang, S. Wu, and J. Zhu, “Pianomotion10m: Dataset and benchmark for hand motion generation in piano performance,” arXiv preprint arXiv, vol. 2406, Art. no. 09326, 2024, doi: 10.48550/arXiv.2406.09326.