J. COMBIN. MATH. COMBIN. COMPUT. 127a (2025) 8797--8811

Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Optimal Design of Pitch Adjustment in AI Models for Opera Vocal Interpretation Based on Support Vector Regression

Yu Wang[1, ✉]

[1] *School of Music, Shanghai Normal University, Shanghai, 200233, China*

ABSTRACT

Since the 21st century, the rapid development of artificial intelligence technology, artificial intelligence in many fields have achieved remarkable research results and applications, the integration of AI technology and music has also gradually become an emerging research field. In this paper, first of all, the generation principle of vocal interpretation AI model is studied, in order to realize the digital conversion of vocal interpretation this paper constructs a converter model so as to facilitate the application of artificial intelligence algorithm model. In this paper, in order to match the generated opera vocal music with the given opera performance background, the rhythmic relationship between opera and vocal interpretation is established, and the relationship between motion salience and note intensity is constructed. On this basis, the generator model is changed to a model with a loop structure, and the music theory is mathematically modeled to propose an adversarial network model based on improved multi-track sequence generation. Finally, for the prediction problem in the vocal interpretation AI model, this paper is optimized based on support vector regression. Through empirical analysis, the improved model in this paper has a smaller gap with the real dataset on the metrics of pitch use, pitch shift, note interval and polyphony rate within the track. Meanwhile, the TD distances of this paper's improved model on the three datasets are 0.655, 0.784, and 0.685, respectively, which is the smallest in the experimental data, and the quality of the improved model's vocal music generation is excellent. The pitch distribution of this paper's improved model and the original vocal data basically match, indicating that this paper's model has better effect on pitch adjustment. In addition, the improved model of this paper generates vocal music with better musicality effect, which has higher musicality while avoiding the generation of more invalid notes. The research work of the paper proves the feasibility of the AI model for opera vocal interpretation and provides a new solution for the current field of vocal music generation.

Keywords: deep learning, generative adversarial network, support vector regression, AI generation model, vocal interpretation

✉ Corresponding author.
E-mail address: 13816749976@163.com (Y. Wang).
Received 15 January 2024; Revised 25 March 2024; Accepted 30 November 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-488

1. Introduction

Vocal performance is an art form in which music, dance and performance are created in the context of art. In this artistic creation, the task is to discover the beauty in life, express it, create it and let the audience feel it. The aesthetics of vocal performance of musical theater has such qualities as beauty of sound, rhythm, emotion, word sound and singing voice [1-4]. The vocal performance of musical theater is an important component of its artistic expression, which affects the success or failure of the musical theater stage. Performers should pay attention to the improvement of their own aesthetic ability, the enhancement of artistic expression and the cultivation of infectious power. The rich ideological connotation of the work, touching and delicate emotions, vivid image of the characters rely on the performer to present, the performer should continue to strengthen the practice, in-depth works, integration of their own aesthetics, to enhance the aesthetic value of the art of vocal music [5-8].

In recent years, with the rapid development of artificial intelligence (AI) technology, its application in the field of audio processing has become more and more extensive. The application of AI technology in audio processing has a broad prospect and potential. By selecting appropriate AI models, preparing high-quality training data, performing data preprocessing and model training optimization, we can achieve more efficient, accurate and innovative audio processing [9-12]. However, when applying AI techniques for audio processing, we also need to pay attention to some techniques and considerations to ensure the accuracy and reliability of the processing results [13-15]. Only on the basis of fully understanding and grasping these techniques and precautions, we can better apply AI technology for audio adjustment, in order to assist the perfect performance of opera vocal interpretation [16-17].

This paper is oriented to the opera vocal interpretation in the deep learning basis first compared with the advantages of unified machine learning, for the needs of vocal interpretation AI model to realize the digital conversion of vocal music, this paper builds converter model to deal with the sequence data of these vocal music. Then the time, motion speed, and motion salience of vocal interpretation in opera are correlated with the beat, note density, and note intensity in music, and mathematical models are established. Based on this, this paper proposes an adversarial network model based on improved multi-track sequence generation, and uses support vector regression to optimize the vocal prediction of the AI model, and constructs an AI generation model for vocal interpretation in opera. Finally, the performance of the improved model in this paper is verified through experimental analysis to explore the actual effect of the opera vocal interpretation AI model.

2. Support vector regression-based AI model for vocal interpretation generation

2.1. Deep Learning-based Vocal Interpretation Generation

2.1.1. Principles of AI generation for vocal interpretation. At present, AI vocal interpretation does not yet have an optimal solution in the main technology, and most of the hybrid algorithms are used. In practice, the state-of-the-art deep learning algorithm is one of the most prominent techniques in the research field of AI vocal interpretation, with self-learning ability, associative storage function, and high-speed ability to find optimal solutions compared with other algorithms. The comparison between deep learning and traditional machine learning is shown in Figure 1, where the feature extraction of traditional machine learning mainly relies on manual labor, however, this method is not very versatile and susceptible to subjective factors. In contrast, deep learning has significant advantages when dealing with data that is difficult to represent symbolically. In the process of feature extraction, deep learning mainly relies on the automation of the machine to complete, from the actual results, its effect is significant, the performance is often better than traditional machine learning.

This algorithm mimics the principle of neural network transmission in the human brain, so in this deep learning, the programmer needs to build a multi-layer “neural network” and program in the

multi-layer structure to process the information between input and output points. After inputting the data, the artificial neural network will find patterns between the many input vocal pieces and form an understanding of the music, which the AI will take with it to make predictions about where the music will go, and the validation data will tell it whether the predictions are correct or not, and both correct and incorrect feedbacks will be memorized by the AI. After a great deal of learning, the AI becomes more and more predictive, and eventually grasps the summarized information and then composes.

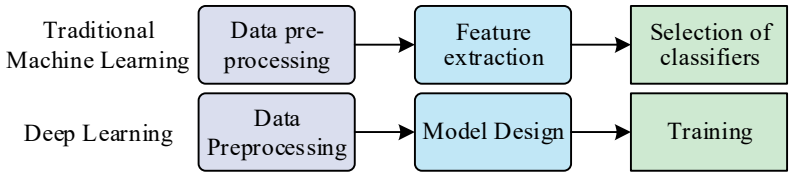


Fig. 1. The contrast between deep learning and traditional machine learning

2.1.2. Converter model for vocal interpretation. The way vocal audio is stored in a computer is usually done digitally. Specifically, audio data is represented as a series of binary digits, each representing a sample. The frequency and sampling rate of the samples determine the quality and size of the audio, which needs to be converted from analog to digital. The conversion process can be divided into three steps, namely sampling, quantization and encoding. Digital Interface for Vocal Music (MIDI) is a communication protocol between electronic musical instruments and between electronic musical instruments and computers. MIDI has become an integral part of music production and performance, and is widely used in music production software, electronic musical instruments, and stage performances. It allows different electronic instruments and computers to communicate with each other via cable or radio waves, enabling them to work together to perform complex musical pieces. The protocol specifies a series of digital messages that are used to describe the various actions of musical performance, such as key presses, strumming, volume, pitch, and so on. These messages are encoded as MIDI data and then transmitted, which can be decoded and converted into an actual musical performance by an electronic musical instrument or computer at the receiving end. The advantage of this protocol is that it provides standardized communication between different electronic musical instruments and computers, allowing different devices to work together to perform complex musical compositions. In addition, the MIDI protocol supports multi-channel communication, allowing multiple devices to play different tracks at the same time, thus creating richer musical effects.

The structure of the converter model is shown in Figure 2. In the vocal interpretation AI raw model, elements such as melody, harmony and rhythm are in the form of sequences, so the converter model can be used to process these sequence data. The converter model has many advantages over previous AI algorithmic models. For example, it can handle sequence data up to thousands of elements long, which makes it better able to handle complex sequence data in music composition. It has a powerful modeling capability that captures complex patterns and regularities in music composition, thus generating more beautiful and expressive musical compositions. It employs techniques such as self-attention mechanism and residual networks, which enable it to accomplish the task of vocal interpretation generation with high computational efficiency.

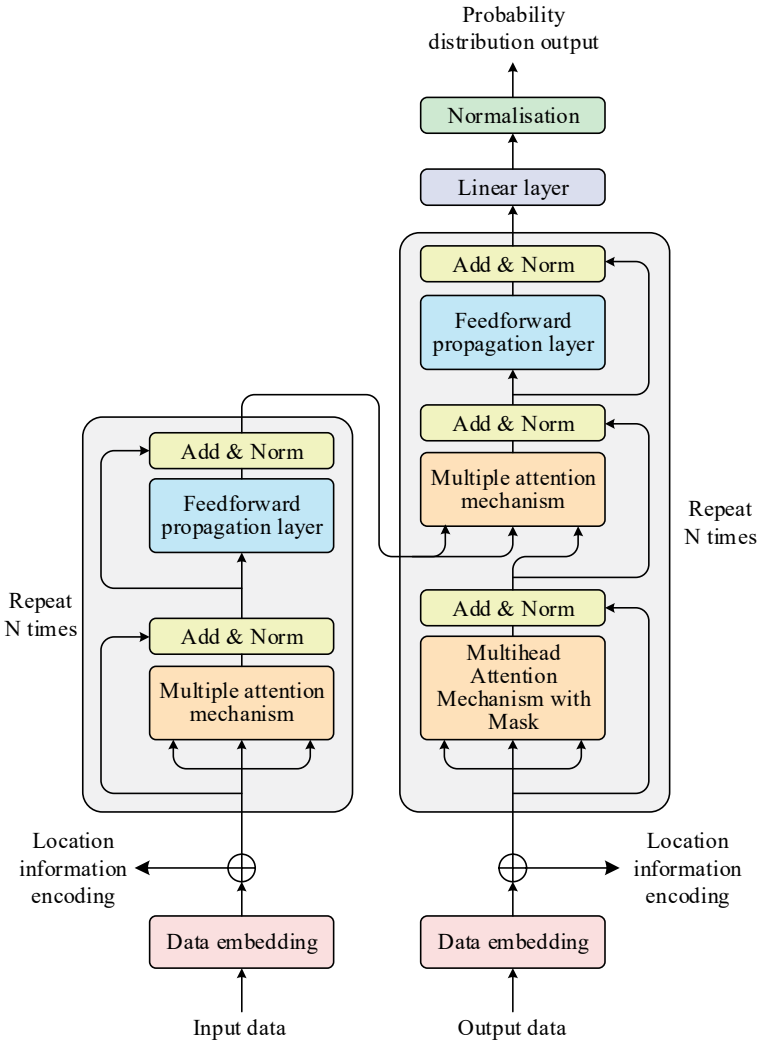


Fig. 2. Converter model architecture

2.2. Vocal Interpretation Generation Model Based on Multitrack Sequence Generation Adversarial Networks

2.2.1. Establishing the relationship between opera and vocal interpretation. Rhythm exists not only in music but also in vision, and rhythm can reflect visual motion in opera vocal performances and the distribution of notes in time in the vocal score. In order to match the generated background vocal music with the given opera performance context, this paper analyzes and establishes the rhythmic relationship between opera and vocal interpretation.

First the relationship between time and vocal beats in the opera performance, based on this connection then the relationship between the speed of movement and the density of notes in the opera performance, and finally the relationship between movement salience and note intensity.

Assuming that an opera contains T frame, use the following formula to convert the $t\left(0<t\le T\right)$nd frame to the corresponding number of beats:

$$f_{beat}(t)=\frac{Tempo\cdot t}{FPS.60} \tag{1}$$

where Tempo is the tempo of the background soundtrack and FPS is the abbreviation for frames per second, which is an intrinsic property of video, with one beat being the shortest unit. It is also possible to convert the ist beat to a frame of the video based on its inverse function:

$$f_{trame}(i) = f_{beat}^{-1}(i) = \frac{i \cdot FPS \cdot 60}{Tempo} \tag{2}$$

Taking a video of 24 frames per second as an example, a clip in 1 second is intercepted, which corresponds to a one in the piece of music these two equations are the basic building blocks for constructing the rhythmic relationship between the video and the vocal music.

First the whole video is divided into M clip, where M is defined as:

$$M = \left\lceil \frac{f_{bat}(T)}{S} \right\rceil \tag{3}$$

Customize T to be the total number of frames in the video and set $S = 4$, which means that the length of each clip corresponds to four beats in vocal music, i.e., one bar, and then this paper calculates the speed of motion based on the optical flow in each clip. Optical flow is an effective tool for analyzing video motion, using an optical flow field to measure the displacement of a single pixel point between two consecutive frames. Analogous to distance and velocity, F_1 is defined as the average of the absolute optical flow to measure the magnitude of motion in frame t, and H and W represent the height and width of the video, respectively:

$$F_t = \frac{\sum_{x,y} \left| f_i(x,y) \right|}{HW} \tag{4}$$

Define the motion speed of the m st video clip segment as $speed_m$, where f_{frame} is the number of frames corresponding to the number of beats:

$$speed_m = \frac{\sum_{i=f_{i=1}(S(m-1))+1}^{f_{frame}(S_m)} F_t}{f_{frame}(S)} \tag{5}$$

The density of notes is then used to connect the speed of movement, where the concept of notes focuses more on rhythmic characteristics, since all notes arranged vertically within a unit note have the same starting position, whether it is a seventh chord or a ninth chord:

$$note_{i,j,k} = \{n_1, n_2, \ldots, n_N\} \tag{6}$$

where note denotes a combination of notes arranged vertically within a unit note, i denotes the ind measure, j denotes the j th beat in the measure, k denotes an instrument, and n denotes a single note. In addition, a measure can be represented as a set of non-empty notes:

$$bar_{i,k} = \left\{ note_{i,j,k} \mid note_{i,j,k} \neq \varnothing, j = 1,2,\ldots,16 \right\} \tag{7}$$

where $j = 1,2\ldots.16$ denotes the division of a measure into 16 scales, with the note density of each measure defined as:

$$density_j = \left| \left\{ j \mid \exists k \in K,\ note_{i,jk} \in bar_{i,k} \right\} \right| \tag{8}$$

Motion saliency on frame t was calculated as the average positive change in optical flow in each direction between two consecutive frames. A series of visual beats were then obtained by selecting frames with both a local maximum of motion saliency and an approximately constant tempo, where saliency would have a large value when there was a sudden visible change. The intensity of a note was defined as the number of vertically aligned notes per unit note, as described above:

$$strength_{i,j,k} = \left| note_{i,j,k} \right| \tag{9}$$

The intensity of the notes indicates the richness of the extended chords or harmonies, giving the audience a sense of rhythm that accompanies their progression; the higher the intensity of the notes, the stronger the auditory impact the audience will feel.

2.2.2. Improved Multi-Track Sequence Generation Adversarial Network Modeling. This paper proposes some improvements based on the multi-track sequential generative adversarial network model, by improving the structure of the network model to make it a recurrent generative adversarial network with a feature extractor, this model can improve the coherence between the generated vocal phrases, and the overall auditory effect of the vocal music. At the same time, the connection between each data sample is increased to make the articulation between notes more natural, and the pitch adjustment is enhanced in the model to avoid the fragmentation problem. This paper focuses on the generation of vocal music for a specific opera performance context, so the idea of conditional generative adversarial networks is borrowed here, and vocal music attributes, including beat, note density, and note intensity, are fed as conditional variables into the generator and the discriminator.

In this paper, the cross-entropy loss function will be used as the loss function of the discriminator, which can well improve the quality of the generated music. Meanwhile, in order to make the generated vocal music follow the rules of music theory, the basic music theory is modeled by means of mathematical modeling, and the importance of different contents in music theory varies in composing, which will be reflected in the reward and penalty values of the generating network, i.e., the more important contents bring higher reward and penalty values. The purpose of this is to enable the generated vocal music to be constrained by the rules of music theory, leading to a better listening experience.

In order to make the generated vocal music more in line with the real vocal music, the pitch ranges of the notes generated from different tracks should be set, $y_{\min}$ and $y_{\max}$ representing the lowest and highest notes of the range respectively, y_t representing the pitch value of the note at the moment of t, and R representing the value of the prize or penalty:

$$R^{m_1}\left(S_{tz}, y_t\right)=\begin{cases}0.1 & y_t \in \left[y_{\min}, y_{\max}\right] \\ -0.6 & y_t \notin \left[y_{\min}, y_{\max}\right]\end{cases} \tag{10}$$

Limits the number of notes that can be sounded at the same time in different tracks, a_t is the number of notes that can be sounded at the same time in t moments, n is the number of notes that can be sounded at the same time in each track, and R is the bonus or penalty value:

$$R^{m_2}\left(a_t\right)=\begin{cases}0.2 & a_t \le n \\ -0.5 & a_t > n\end{cases} \tag{11}$$

In a piece of music, rests can help the piece to breathe and change the rhythm. The use of rests can make the whole piece of vocal music more rhythmic and is an important way to emphasize the style of the piece. In order to make the generated music closer to real vocal works, the number of rests used in each measure should be appropriately limited, where the number of rests should not exceed 40% of the total number of notes in a measure. Where S is the number of notes in the measure, y is the number of rests, and R is the penalty value:

$$R^{m_3}(y)=\begin{cases}0.1 & y \le 0.4 S_{1t} \\ -0.3 & y > 0.4 S_{1t}\end{cases} \tag{12}$$

Typically, the notes in the strong beat of a measure tend to represent the harmonic tendency of the measure, and these tones are generally the intra-chord tones of the chords of the measure. The vocal data sets used in this paper are all in 4/4 time, and the chords are all triads (three tones superimposed on each other in thirds), and thus the laws of the strengths and weaknesses within the measure are,

strong, weak, sub-strong, and weak.

Based on the above four music theory rules that have a strong influence on vocal music, different weights are assigned to them according to their importance. a_i indicates the weight assigned to the ind music theory rule:

$$R_G\left(S_{1y}, y_t\right) = \alpha_1 R^{m_1}\left(S_{12}, y_2\right) + \alpha_2 R^{m_2}\left(a_2\right) + \alpha_3 R^{w_1}(y) + \alpha_4 R^{c}\left(S_{12}, y_2\right) \tag{13}$$

By assigning different weights to the reward function and the cross-entropy loss function by the discriminator, the following objective function can be derived, with β_1 and β_2 being the weight values:

$$J_{G_g} = \beta_1 R_G + \beta_2 D_g \tag{14}$$

The vocal music interpretation generation model based on the improved multi-track sequence generation adversarial network is shown in Fig. 3, in order to make the generated vocal music closer to the real vocal music in the sense of hearing, so that the transition is more harmonious, this paper has adjusted the existing timing model, the specific implementation of this paper is to generate the vocal music data samples through the self-generated timing model first, due to the generation of samples from this timing model for the monotrack melody, so it will be used as the a priori Since the sample generated by the timing model is a single-track melody, it is used as an a priori condition to generate the remaining four tracks in the conditional timing model. The results are then fed into the conditional timing model again as a priori conditions for the next round of training. The improved timing model only utilizes the vocal data samples generated by the autonomous timing model as the a priori condition at the beginning of the training, and then obtains the important features of the vocal clips generated in the previous round through the feature extractor during the subsequent training, splices them with the random noise vector as the a priori condition, and feeds them into the conditional generation timing model for the next round of training. The purpose of this is to improve the smoothness of the generated vocal music and to make it more natural in the listening experience.

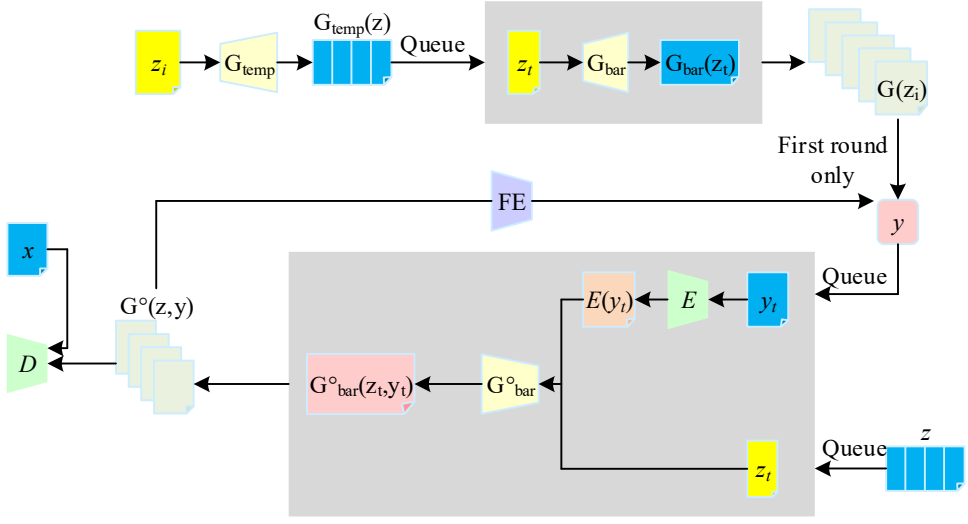


Fig. 3. Vocal performance generation model

2.2.3. Optimization of vocal prediction based on support vector machine regression. In model training, the accurate prediction of vocal music is the key to improve the performance of AI-generated models for vocal rendition, so in this paper, we optimize vocal music prediction based on support vector machine regression. Support vector machines are very special algorithms, this algorithm is

based on the VC dimension theory of statistical learning theory and the principle of structural risk minimization, which seeks the best compromise between the complexity of the model and the learning ability according to the finite samples, with a view to obtaining the best generalization ability.

Consider the training data $\{(x_1,y_1),(x_2,y_2),\cdots,(x_l,y_l)\}\subset X\times$ □, where the space of independent variables X is often □^{p}. In ε support vector machine regression, the aim is to find a function $f(x)$ that must be as flat as possible (i.e., as simple as possible) and that deviates from ε at most for all values y_i of the dependent variable in the training set, i.e., we don't care about errors smaller than ε but cannot accept ones larger than it. The loss is computed when, and only when, the absolute value of the difference between $f(x)$ and y_i is greater than ε, which is equivalent to constructing an interval band of width 2ε centered on $f(x)$, into which the training samples are considered to be correctly predicted if they fall.

Consider first the linear function:

$$f(x)=\langle w,x\rangle+bx, w\in X \tag{15}$$

The symbol $\langle\cdot,\cdot\rangle$ in Eq. denotes the inner product in X. Flatness in this context implies finding very small w, one way to minimize the Euclidean paradigm squared $\|w\|^2=\langle w,w\rangle$. Based on the structural risk minimization principle, the optimization problem to be solved can be described as:

$$\begin{aligned}&\min_w \frac{1}{2}\|w\|^2\\ &\text{Bound for:}\begin{cases}y_i-\langle w,x_i\rangle-b\le\varepsilon,\forall i\\ \langle w,x_i\rangle+b-y_i\le\varepsilon,\forall i\end{cases}\end{aligned} \tag{16}$$

The assumption of the equation is that for almost all (x_i,y_i) such a solution to the problem of ε accuracy exists, but the conditions of the equation seem to be too harsh, so a penalty factor C and loose independent variables ξ and ξ^* are introduced to relax the conditions to allow for some error, and the optimization problem with relaxed conditions can be written as:

$$\begin{aligned}&\min_w \frac{1}{2}\|w\|^2+C\sum_{i=1}^{l}(\xi_i+\xi_i^*)\\ &\text{Bound for:}\begin{cases}y_i-\langle w,x_i\rangle-b\le\varepsilon+\xi_i,\forall i\\ \langle w,x_i\rangle+b-y_i\le\varepsilon+\xi_i^*,\forall i\\ \xi_i\ge 0,\xi_i^*\ge 0,\forall i\end{cases}\end{aligned} \tag{17}$$

The penalty factor $C(>0)$ represents all the points that violate the constraints, and the purpose of the support vector machine algorithm is to minimize C. After adding the penalty factor C, the value of C can be modified to adjust the strength of the penalty for violating the constraints, and the Lagrangian is used to obtain the formula:

$$\begin{aligned}&\min\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle\\ &-\varepsilon\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)+\sum_{i=1}{}'y_i(\alpha_i-\alpha_i^*)\end{aligned} \tag{18}$$

The constraints are $\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0$, α_i, $\alpha_i^*\in[0,C]$. where α_i, α_i^* are Lagrange multipliers.

The key to the support vector machine is the inner product kernel function $K(x,x_i)$ between the input vector $x(i)$ and the vector x drawn from the input space, which is introduced to obtain Eq:

$$\min \frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i-\alpha_i^*\right)\left(\alpha_j-\alpha_j^*\right)K\left(x_i,x_j\right) \\ -\varepsilon\sum_{i=1}^{l}\left(\alpha_i+\alpha_i^*\right)+\sum_{i=1}^{l}y_i\left(\alpha_i-\alpha_i^*\right) \tag{19}$$

The constraints are $\sum_{i=1}^{l}\left(\alpha_i-\alpha_i^*\right)=0$, α_i, and $\alpha_i^*\in[0,C]$.

Solving further yields w and b:

$$w=\sum_{i=1}^{l}\left(\alpha_i-\alpha_i^*\right)x_i$$

$$b=\begin{cases} y_j+\varepsilon-\sum_{i,j=1}^{l}\left(\alpha_i^*-\alpha_i\right)K\left(x_i,x_j\right) \text{ Pawn } \alpha_i\in(0,C) \\ y_j-\varepsilon-\sum_{i,j=1}^{l}\left(\alpha_i^*-\alpha_i\right)K\left(x_i,x_j\right) \text{ Pawn } \alpha_i^*\in(0,C) \end{cases} \tag{20}$$

The decision function formula is obtained:

$$f(x)=\sum_{i=1}^{l}\left(\alpha_i^*-\alpha_i\right)K\left(x_i,x_j\right)+b \tag{21}$$

3. Empirical analysis of AI generation model for opera vocal interpretation

3.1. Performance analysis of the AI generation model for vocal interpretation

In order to verify the performance of the model vocal generation in this paper, it was compared with the multi-track vocal models MuseGan and MMM in a comparative experiment. In this paper, the number of pitches (PC), the average pitch spacing (PS), the average intervocalization interval (IOI), and the polyphony rate (PR) are used as the metrics, and the above metrics only measure the quality of note generation within a single track. In order to measure the inter-track harmony, this paper combines the tracks two by two as a pair and uses the inter-track distance (TD) metric as a measure, which is a measure of the harmonic relationship between a pair of tracks, the smaller the better. In order to measure the model effect, in LPD, Freemidi and GPMD three datasets randomly selected 1000 clips as a test case, respectively, using MuseGAN, MMM and this paper's model according to the main theme of the vocal music, generating the music with the same input format generates the same 4/4 beat. The model itself is set up for comparison experiments during the experimental process, with MFLow denoting the training effect of the unimproved deep learning benchmark model, and MFOur denoting the improved model of this paper.

3.1.1. Qualitative Analysis of Vocal Generation Models. The quality comparison of the generated vocal music is shown in Table 1, and the experimental results show the performance on the three datasets respectively, and Ground-Truth indicates the indicator results of the original vocal music data for reference and comparison. According to the results of the generation quality index measurements of in-track vocal music, comparing the data of the three datasets, this paper's model achieves the optimal results for each index. The gap between this paper's improved model and the real dataset is even smaller on the metrics of pitch use, pitch shift, note spacing, and polyphony rate intra-track, ranging from 0.105 to 1.005 on the LPD data, and from 0.115 to 1.132 and 0.042 to 1.600 on the Freemidi and GPMD, respectively.In terms of the harmonization metrics of the inter-tracks, the TD, the TD distance on the three datasets are 0.655, 0.784 and 0.685, respectively, and the experimental data is the smallest, i.e., the experimental results are the best. Comprehensively analyzed, the improved

model of this paper has the best performance in terms of vocal generation quality and track harmony, indicating that the improved model of this paper can further improve the quality of vocal generation.

Table 1. The quality of the music is compared

Data set	Model	PC	PS	IOI	PR	Mean TD
LPD	Ground-Truth	3.561	3.972	3.330	0.456	1.005
	MuseGan	+4.755	+12.150	-0.851	+0.562	1.035
	MMM	+0.760	+1.303	+0.286	-0.189	1.715
	MFLow	-0.482	-1.102	+0.205	-0.172	0.987
	MFOur	-0.471	-1.005	+0.105	-0.165	0.655
Freemidi	Ground-Truth	3.632	4.885	2.489	0.465	0.571
	MuseGan	+4.531	+7.156	-1.026	+0.546	1.042
	MMM	+1.005	+1.182	-0.398	+0.201	1.562
	MFLow	-0.432	-1.168	-0.179	-0.154	0.958
	MFOur	-0.412	-1.132	-0.115	-0.151	0.784
GPMD	Ground-Truth	3.632	4.885	2.489	0.465	0.985
	MuseGan	+4.455	+6.822	-0.954	+0.615	1.052
	MMM	+0.815	+3.564	-0.165	-0.195	1.567
	MFLow	-0.675	-1.811	-0.055	-0.161	0.987
	MFOur	-0.461	-1.600	-0.042	-0.085	0.685

3.1.2. Comparison of pitch adjustment effects of vocal AI models. Pitch reflects the melodic direction of a piece of music, in order to further quantify the difference between the vocal interpretation AI model-generated works and the real dataset, this paper calculates the pitch distribution of all the vocal music generated by the three models, converts it into a probability distribution function using kernel density estimation and plots it into a graph, and the results of the comparison of the pitch adjustment effect are shown in Figure 4. From the figure, it is easy to see that the model generation effect of MMM and MuseGan has a large pitch variance and the mean value deviates from the real data. While the improved model of this paper basically matches the pitch distribution of the original vocal data, indicating that the model of this paper has better effect on pitch adjustment.

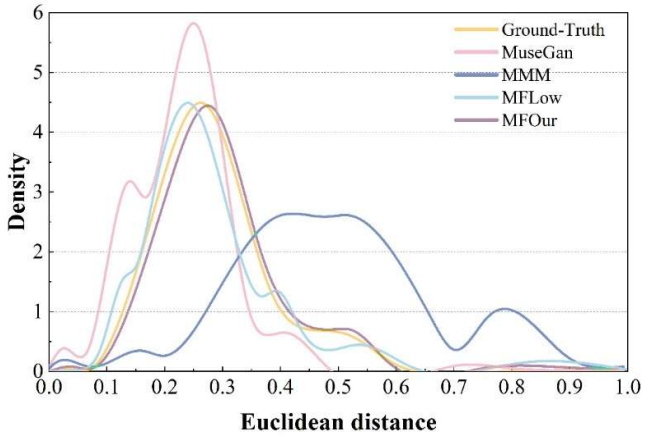


Fig. 4. Pitch adjustment

In order to observe the gap more intuitively, this paper calculates the overlap area between the data generated by each model and the distribution function of the real dataset. The larger the value the higher the coverage and the closer it is to the real acoustic music, and the OA distances on NLH and PCH for different models are shown in Fig. 5. Observation shows that the model in this paper has the highest NLH and PCH values of 0.915 and 0.936, respectively. Compared with the original MFLow model, the improvement is 0.026 and 0.011, respectively, and the improved model in this paper has the best effect in pitch adjustment, which further validates the performance of the improved model in

this paper.

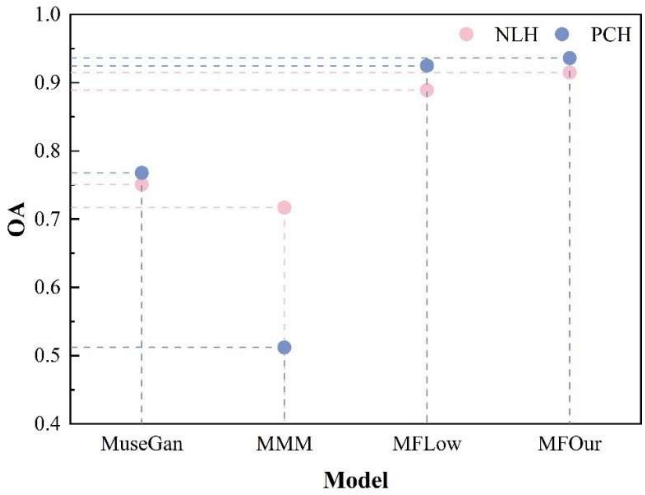


Fig. 5. The OA distance from the different models NLL and PCH

3.2. Analysis of the Generative Effects of Opera Vocal Interpretation

3.2.1. Analysis of vocal production effects. The dataset selected for the experiments in this paper is the JS Fake set based on the JS Bach dataset expanded with similar vocals and manually labeled chord codes containing 500 vocals. There is no consistent evaluation standard for the measurement of vocal generation samples. From a generative point of view, a model's ability to model vocal music can be initially measured by evaluating the loss function and the accuracy of the generated notes when the loss function converges during training. In terms of musicality, statistical metrics on chord tones and chord extents provide a musicality measure of harmonic harmony, and in this paper, we choose the chord tone to non-chord tone ratio (CTnCTR), the note and chord consistency score (PCS), and the melodic chord tonal distance (MCTD) as the musicality metrics for vocal music generation.

In this paper, three metrics of chord and melody introduced are used to measure musicality, and the quantitative results of the vocal generation effect are shown in Fig. 6. Among them, the RS metric is the quantitative evaluation of the generated vocal music by professional musicians and normalized to between 0.2 and 1, and h is the coefficient of gamma sampling. In this paper, the musicality of the generated vocal music is measured. In this paper, the Bach music in the test set is selected to compare the metrics of the generated vocal music, and the input is the notation of the Bach samples. The generated samples are already very close to Bach's originals, comparing the metrics of Bach's originals (Bach), the experimental results prove that the coefficient of gamma sampling is 0.5, the effect is more compatible with the originals, and the respective metrics of RS, MCTD, PCS, and CTnCTR are 0.619, 0.948, 0.668, and 0.866, respectively.

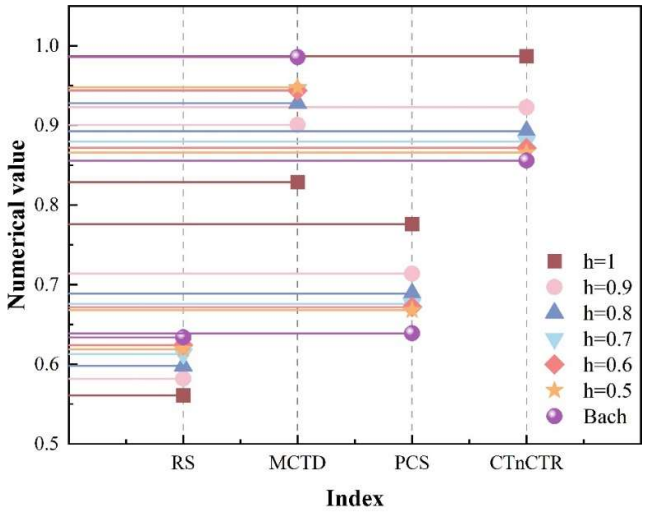
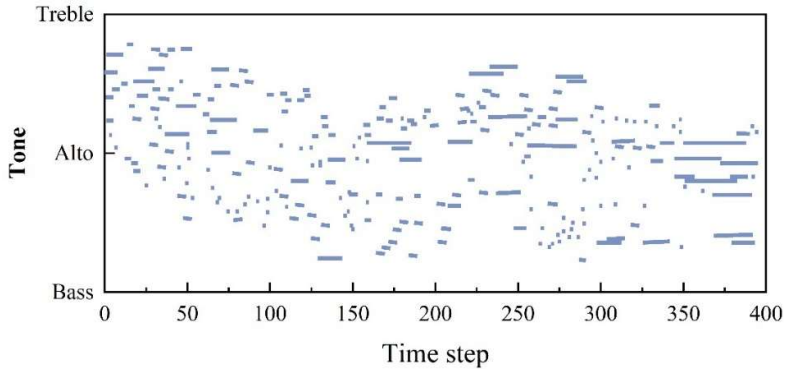
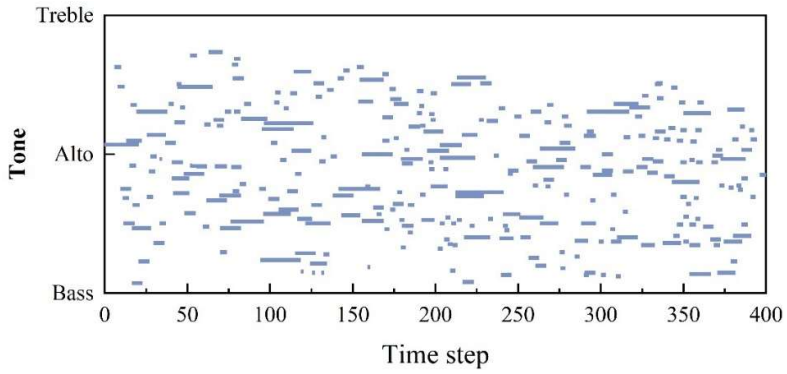


Fig. 6. The results of the results of vocal music production

3.2.2. Comparative analysis of vocal production. The visualization of piano rolls for the generated vocal music can visualize the melody line of the generated vocal music. The visualization results of the generated vocal music are shown in Fig. 7, (a) and (b) are the traditional AI model and the improved model in this paper, respectively. The piano roll can be seen as a two-dimensional table, with the x-axis representing the duration time step and the y-axis representing the pitch level, and the part marked in blue (x, y) represents the y pitch activated at time x. Pianoroll is one of the widespread musical representations that are clear to visualize, but there is no Note-off signal, and thus it is not possible to distinguish between long tones and repetitive short note structures. The comparison of the visualization of vocal music generation by the AI model shows that after generating 350 notes, the traditional original model generates long monophthongs, while the improved model in this paper can generate longer effective lengths with rich variations. This proves that the model in this paper has been further enhanced in generating effective musical lengths with higher musicality while avoiding generating more invalid notes.



(a)Traditional model



(b)Improved model of this article

Fig. 7. Visual results of vocal music generation

4. Conclusion

In this paper, we constructed an AI model for opera interpretation based on an improved multi-track sequential generative adversarial network model, and optimized the AI model for vocal prediction using support vector regression. The main research work of this paper is as follows:

1) The improved AI model in this paper generates vocal music with smaller gaps from the real dataset on the metrics of pitch use, pitch shift, note spacing, and polyphony rate within the tracks, which are 0.105 to 1.005, 0.115 to 1.132, and 0.042 to 1.600 for LPD, Freemidi, and GPMD data, respectively.In terms of the harmony metrics TD between tracks, the paper's improved model TD is 0.655, 0.784 and 0.685, respectively, with the best experimental results. It is verified that the improved model of this paper has the best vocal generation quality and track harmony performance, which can effectively improve the quality of vocal generation. In addition, this paper calculates the overlap area between the data generated by each model and the distribution function of the real dataset. The NLH and PCH values of the improved model in this paper are the highest, which are 0.915 and 0.936. Compared with the original MFLow model, they are improved by 0.026 and 0.011, respectively, and the improved model of this paper performs better in pitch adjustment, which further validates the performance of the improved model of this paper.

2) In the comparison of the generation effect of vocal interpretation AI, it can be seen that the samples generated by this paper's improved model are more compatible with the original work when the coefficient of gamma sampling is 0.5, and the indicators of RS, MCTD, PCS, and CTnCTR are 0.619, 0.948, 0.668, and 0.866, respectively. At the same time, from the perspective of generating scales, the traditional original model generates long monophonic tones. The improved model of this paper has been effectively improved, and the generated scale changes are richer, indicating that the effect of vocal generation of the improved model of this paper has been further enhanced, while avoiding the problem of invalid notes, with higher musicality and appreciation value.

References

[1] Ning, L. (2017, May). On the importance of stage performance in vocal music performance. In 2017 4th International Conference on Education, Management and Computing Technology (ICEMCT 2017) (pp. 12-15). Atlantis Press.

[2] Avila, C., Furnham, A., & McClelland, A. (2012). The influence of distracting familiar vocal music on cognitive performance of introverts and extraverts. Psychology of Music, 40(1), 84-93.

[3] Utz, C., & Lau, F. (2013). Vocal Music and Contemporary Identities. Unlimited Voices in East Asia and the West, New York.

[4] Dalla Bella, S., Tremblay-Champoux, A., Berkowska, M., & Peretz, I. (2012). Memory disorders and vocal performance. Annals of the New York Academy of Sciences, 1252(1), 338-344.

[5] Zuim, A. F., Stewart, C. F., & Titze, I. R. (2021). Vocal dose and vocal demands in contemporary musical theatre. Journal of Voice.

[6] D'haeseleer, E., Claeys, S., Meerschman, I., Bettens, K., Degeest, S., Dijckmans, C., ... & Van Lierde, K. (2017). Vocal characteristics and laryngoscopic findings in future musical theater performers. Journal of Voice, 31(4), 462-469.

[7] Scherer, K. R., Sundberg, J., Fantini, B., Trznadel, S., & Eyben, F. (2017). The expression of emotion in the singing voice: Acoustic patterns in vocal performance. The Journal of the Acoustical Society of America, 142(4), 1805-1815.

[8] Phyland, D. J., Thibeault, S. L., Benninger, M. S., Vallance, N., Greenwood, K. M., & Smith, J. A. (2013). Perspectives on the impact on vocal function of heavy vocal load among working professional music theater performers. Journal of Voice, 27(3), 390-e31.

[9] Birtchnell, T. (2018). Listening without ears: Artificial intelligence in audio mastering. Big Data & Society, 5(2), 2053951718808553.

[10] Deshpande, G., Batliner, A., & Schuller, B. W. (2022). AI-Based human audio processing for COVID-19: A comprehensive overview. Pattern recognition, 122, 108289.

[11] Costantini, G., Casali, D., & Cesarini, V. (2024). New Advances in Audio Signal Processing. Applied Sciences, 14(6), 2321.

[12] Akman, A., & Schuller, B. W. (2024). Audio Explainable Artificial Intelligence: A Review. Intelligent Computing, 2, 0074.

[13] Radaković, M. (2021). Audio signal preparation process for deep learning application using Python. In Sinteza 2021-International Scientific Conference on Information Technology and Data Related Research (pp. 146-152). Singidunum University.

[14] Shahbazova, S. N. (2024). Obtaining Visual and Audio Information Based on Artificial Intelligence. In Recent Advances in Intelligent Engineering: Volume Dedicated to Imre J. Rudas' Seventy-Fifth Birthday (pp. 271-282). Cham: Springer Nature Switzerland.

[15] Gupta, M., & Vaikole, S. (2022). Audio signal based stress recognition system using AI and machine learning. Journal of Algebraic Statistics, 13(2), 1731-1740.

[16] Kumar, R., Gupta, M., Ahmed, S., Alhumam, A., & Aggarwal, T. (2022). Intelligent Audio Signal Processing for Detecting Rainforest Species Using Deep Learning. Intelligent Automation & Soft Computing, 31(2).

[17] Cobos, M., Ahrens, J., Kowalczyk, K., & Politis, A. (2022). An overview of machine learning and other data-based methods for spatial audio capture, processing, and reproduction. EURASIP Journal on Audio, Speech, and Music Processing, 2022(1), 10.