Piano Transcription Using Temporal Harmonic Diagram and Transfer Window Attention in Self-Attention Networks

Qiong Wu*, Tao Yu
School of Music, Tonghua Normal University, Tonghua Jilin 134001, China
E-mail: wuqiong@thnu.edu.cn, yutao12611@sohu.com
*Corresponding author

Keywords: self attention, piano music transcription, constant q conversion, time-series harmonic diagram, mel frequency cepstral coefficients

Received: August 17, 2024

Music transcription is an important means to record and transmit music culture. However, the existing music transcription algorithms still have certain errors in practical applications. To address this problem, the study adopts constant Q conversion to process music signals, introduces note starting point and frame-level pitch recognition module and transfer window attention, constructs temporal harmonic map for music melody extraction, and adopts significance function for music melody smoothing. The study uses the MAESTRO dataset containing about 200 hours of paired audio and MIDI recordings covering different performance styles, and the MedleyDB dataset protecting 122 pieces of music. The experimental results show that the overall accuracy of the transcription algorithm is 2.58% and 2.35% higher than the other algorithms, and the raw pitch accuracy is 2.23% and 1.06% higher than the other algorithms, respectively, for a frequency point count of 600 and a search range of 0.5. The accuracy, recall, and F1 value of the transcription algorithm are 2.11%, 2.27%, and 2.21% higher than the second-best algorithm, and the removal of the window attention and recognition modules decreases the accuracy of the algorithm by 8.07% and 16.76%, respectively. The average processing time of the transcription algorithm is 7.2ms lower than that of the traditional method, and the computational complexity grows more slowly as the amount of data grows. It can be concluded that the piano music transcription algorithm can effectively improve the accuracy of music recognition and transcription, and quickly and accurately convert the relevant audio into the corresponding notes.

Povzetek: Objava predstavi uvajanje algoritem SADLN za transkripcijo klavirske glasbe. Z izboljšano točnostjo in učinkovitostjo prepoznava melodije, optimizira signale in spodbuja ohranjanje glasbene kulture.

1 Introduction

Music transcription (MT) is a process of transforming music performance into written form, which captures the basic elements of melody, rhythm, harmony, and provides a written foundation for subsequent research, teaching, and replication [1]. Although MT may not fully capture the subtle differences in actual performance, it can serve as a rough record, providing people with a framework close to primitive music performance, and has important practical significance for recording and inheriting related music culture [2-3]. The early forms of MT are oral transmission and manual recording, which can result in missing details and be easily influenced by the recorder's subjectivity. As music gradually become more diverse and complex, manual transcription clearly can not meet the needs [4]. Therefore, MT based on computer technology has replaced traditional transcription methods, but music is composed of various elements such as melody, tone, and emotion. Most existing transcription algorithms only start from signal processing, and there is a certain degree of error between the transcribed music and the actual situation [5-6]. Regarding the emotional processing of music, He N and Ferguson found that few people have applied unsupervised learning to music emotion recognition. Therefore, they proposed a two-stage model that combines supervised and unsupervised learning. This model divided music into multiple continuous segments, generated feature representations using autoencoders, and used a bidirectional long short-term memory deep learning model for music emotion classification. Experiments denoted that this model could effectively reduce the risk of overfitting in the deep learning process, and its performance was significantly better than other methods [7]. Qureshi et al. proposed a new aspect level sentiment automatic annotation technique to solve the problem of time-consuming manual annotation in sentiment analysis. This technique extracted data from five aspects: sound, video, music, lyrics, and songs, and

used the N-Gram language model to standardize the data. Experiments showed that the annotation time of this technology was much shorter than that of manual standards, and the consistency with manual standards was high, with a Kappa value of 0.9571 [8]. Orjesek et al. extracted significant emotional features from audio signals and proposed a deep learning-based convolutional neural network (CNN). This network used one-dimensional convolution as the overlay layer, and employed iterative reconstruction layers based on autoencoders and bidirectional gated loop units to mine significant features related to music emotions from the original audio waveform. Experiments showed that this network could significantly improve regression accuracy and enhance the ability to judge emotional features [9]. Yin et al. found that the reliability of emotion recognition was low during large-scale testing, and therefore proposed an end-to-end multimodal framework. This framework was grounded on a one-dimensional residual spatiotemporal attention network, which utilized EDA's emotion recognition and channel spatiotemporal attention mechanism to mine the dynamic and stable features of the channel. Experiments showed that this framework could effectively mine EDA features and achieve large-scale emotion recognition [10].

In terms of accuracy in transcription, Venkata Lakshmi et al. proposed a model combining discriminant fuzzy function and deep neural network to improve the anti-interference ability of speech recognition. This model used Mel frequency cepstral coefficients (MFCC) to extract speech features. Hybrid deep neural networks and discriminative fuzzy logic were used to improve speech clarity, and the bat algorithm with enhanced modular functions was used to optimize model parameters. Experiments showed that this model could effectively improve speech recognition performance and anti-interference ability, with an accuracy 10.25% higher than that of deep autoencoders [11]. Ríos-Vila proposed a speech recognition algorithm grounded on Hidden Markov Model to improve the accuracy and recognition efficiency of speech recognition. This algorithm used Hamming windows to add windows to digital signals and employs backpropagation neural networks to reduce speech recognition errors. Experiments showed that this algorithm had the lowest word recognition error rate and the shortest recognition time [12]. Shashidhar et al. proposed a model combining MFCC and long short-term memory networks to improve speech recognition accuracy. This model used MFCC for audio processing, employed long short-term memory networks for visual speech recognition, and finally integrated them in a deep neural network. The experiment showed that the accuracy of the model for speech recognition was 91%, which was significantly better than existing algorithms [13].

In summary, existing methods have studied the accuracy and anti-interference of extracting and recognizing melody emotions in MT from multiple aspects, and have achieved certain research results. However, there are still some errors between the transcribed music and the actual situation. Therefore, a piano MT algorithm grounded on self-attention deep learning network (SADLN) is proposed, which innovatively uses constant Q conversion to process music signals, introduces note start and frame level pitch recognition modules and transfer window attention modules, constructs a temporal harmonic map to extract music melodies, and uses saliency function to smooth and optimize melodies. The piano MT algorithm aims to improve the accuracy and recognition efficiency of MT, and promote the protection and dissemination of music culture. This study is divided into three parts: the first part is to construct a piano MT system, the second part is to evaluate the performance of optimization algorithms, and the third part is a summary of this study and prospects for future research directions.

Based on the above related studies, Table 1 is summarized, which summarizes the characteristics, main index methods, research results and shortcomings of the related studies.

Table 1: Summary of relevant information of relevant studies

Method	Characteristics	Index	Results	Shortcomings
[7]	Unsupervised learning	Identification accuracy and efficiency	Effectively reduce the risk of overfitting	In large data environments, performance can be limited
[8]	Emotional automatic labeling technology	Kappa price	High standard consistency, fast speed	Performance degrades faster in complex environments
[9]	Iterative reconstruction layer and bi-directional gating cycle cells	Regression accuracy	Explore the significant characteristics of music emotion	High model complexity
[10]	End-to-end multimodal framework	Emotion recognition accuracy	Large-scale emotion recognition	Large-scale emotion recognition
[11]	Discriminating the fuzzy function	Identification accuracy	The accuracy was 10.25% higher than the depth autoencoder	The algorithm has a long running time

[12]	Hidden Markov model	Identify error rates and recognition time	The recognition error rate was reduced by 6.4%	The recognition accuracy is low
[13]	Mel frequency cepstral spectrum coefficient	Mel frequency cepstral spectrum coefficient	The recognition accuracy was 91%	It takes a lot of data to train
Research method	Self-attention mechanism and timing harmonic characteristics	Accuracy, recall and F1 values	Improve the accuracy and speed of music recognition and transliteration	/

2 Methods and materials

2.1 Construction of piano music transcription system based on SADLN

There are some similarities between MT and speech recognition, including recognizing the frequency and rhythm of sound signals, etc. Most of the modules of existing MT methods are converted from speech recognition methods. However, there are still some differences between the two. Speech recognition focuses on recognizing and understanding the semantics, grammatical structure, tone, and punctuation of a language [14]. MT, on the other hand, focuses on the melody, rhythm, and harmony of music, converting the notes and melodies in music into textual form or sheet music for recording and analysis. Because the study needs to obtain the time and frequency domain signals of audio, the signals of music need to be converted, and the existing conversion methods include constant Q-conversion and short time-distance Fourier transformation [15]. Considering the characteristics of music signals, the study uses constant Q conversion for music signal processing, and constant Q-conversion can better reflect the frequency characteristics of music signals. The center frequency of constant Q-conversion is distributed according to the exponential law, and the filtering bandwidth is different, but the ratio of the center frequency to the bandwidth is a constant Q. This makes the frequency of the spectral transverse axis of the constant Q-conversion is based on log2 as the bottom, and it can change the length of the filtering window according to the different frequencies of the spectral lines, so as to obtain a better performance. Since the constant Q-conversion is the same as the distribution of the scale frequencies, the amplitude values of the music signal at each note frequency can be obtained directly by calculating the constant Q-conversion spectrum of the music signal, whereas the short-time Fourier transform is suitable for analyzing non-stationary signals. The constant Q-transform produces a spectrum that is logarithmically scaled on the frequency axis rather than linearly scaled, and the window length of the spectrum changes with frequency. The variation of the window length with frequency ensures that the best frequency resolution is obtained when analyzing different frequency components. This property makes the constant Q-conversion very useful in music signal processing to better capture the time-frequency characteristics of music. The expression for constant Q-conversion is shown in equation (1).

$$Q = \frac{f}{\delta_f} = \frac{1}{2^{1/b} - 1} \tag{1}$$

In equation (1), f represents frequency, δ_f represents the frequency bandwidth at frequency f, and b represents the amount of spectral lines contained within an octave. As shown in equation (1), the constant Q is only related to the number of spectral lines b. The calculation of window length is shown in equation (2).

$$N_k = \left\lceil Q \frac{f}{f_k} \right\rceil, k = 0,1,...,K-1 \tag{2}$$

In equation (2), N_k represents the window length that varies with frequency, f means the sampling frequency, and f means the frequency of the k th component. The k th semitone frequency component of the n th frame in constant Q-conversion is calculated as shown in equation (3) [16].

$$X^Q(k) = \frac{1}{N_k} \sum_{n=0}^{N_k - 1} x(n) w_{N_k}(n) \exp\left(-j \frac{2\pi Q}{N_k} n \right) \tag{3}$$

In equation (3), $X^Q(k)$ represents the k th semitone frequency component, w represents the window function, j represents the complex frequency domain,

and n represents the current frame rate. Using constant Q-conversion can also lower the final spectral dimension obtained, reduce the number of parameters required for the model, and improve computational speed. Frequency domain quantization is performed on the music signal as shown in equation (4).

$$B(f)=\left\lfloor bins\cdot\log\left(\frac{f}{f_0}\right)+0.5\right\rfloor \quad (4)$$

In equation (4), $bins$ represents quantifying an octave into different intervals, f_0 represents the lowest frequency of each piano note, and $\lfloor\ \rfloor$ represents rounding down. The signal harmonics are calculated in the corresponding frequency domain as shown in equation (5).

$$B\left(f_v^m\right)=\left\lfloor bins\cdot\log(m+1)+bins\cdot\frac{v}{12}+0.5\right\rfloor \quad (5)$$

In equation (5), f_v^m represents the frequency of the mth harmonic of note v, m is equal to 0, it represents the fundamental frequency of the note, v is an integer between [0, 87], and m is a positive integer. After the signal conversion is completed, the signal is downsampled to reduce the audio sampling rate to 16kHz, and the audio segments are divided into small segments every 10ms with a frame length of 512bt. The allocated fragments are normalized to a mean of 0 and a variance of 1. After the data processing is completed, the specific process of the piano MT system is denoted in Figure 1.

In Figure 1, the audio data is first down sampled and divided into segments every 10 seconds to reduce data dimensionality and improve computational speed. Using MFCC for feature extraction, the audio signal is converted into a set of feature vectors with good discrimination. The SADLN has 4 layers, and in the first three layers, a random deactivation of 0.1 is used to improve the system's generalization ability and reduce the occurrence of overfitting. The attention of the first three layers of the network is also set to 0.1. In the early stage of training, the number of attention heads in the multi-head attention module of the network is set to 8, the output dimension of the attention module is set to 512, and the output dimension of the fully connected layer is set to 2048. In the later stage of system training, the number of attention heads in the attention module is set to 12, the output dimension is set to 768, and the output dimension of the fully connected layer is set to 3072. The training results are mapped onto the output result with dimension 88. The system parameters are adjusted through target fitting, and after decoding, the output results are evaluated to obtain corresponding scores.

Because the starting point density of notes is low, it is easy to produce class imbalance during training, so the loss function is used to optimize. However, although the traditional focal loss function can be adjusted to a certain extent in the process of note translation, it is easy to pay too much attention to the samples that are difficult to classify. At the same time, the precision of parameters is high, which is easy to reduce the calculation efficiency. Therefore, the objective fitting of the music transfer system is calculated using the Logit-adjust loss function, as shown in equation (6) [17].

$$BER(f)=\frac{1}{L}\sum\nolimits_{y\in L}P_{x|y}\left(y\in\arg\max_{y\in\hat{y}}\ f_y(x)\right) \quad (6)$$

In equation (6), $BER(f)$ represents the error rate, L represents the number of categories, y represents the actual category, $P_{x|y}$ represents the probability of sample x appearing in actual category y, and $f_y(x)$ represents the model's prediction score for sample on category y. The Logit adjust loss function can record prior information based on different frequencies and adjust the bias state according to the positive and negative categories of the information. To raise the accuracy of note recognition, a note starting point and frame level pitch recognition module is added to the system, as shown in Figure 2.

In Figure 2, there are 5 shared layers, 1 independent layer, and 1 fully connected layer. After the note start point and frame level pitch features are extracted in the shared layer, the frame level pitch features enter the independent layer for prediction, which merged with other prediction terms that have added note start information in the fully connected layer. In response to the long-term dependence of MT, this study introduces transfer window attention into deep learning networks, as shown in Figure 3.

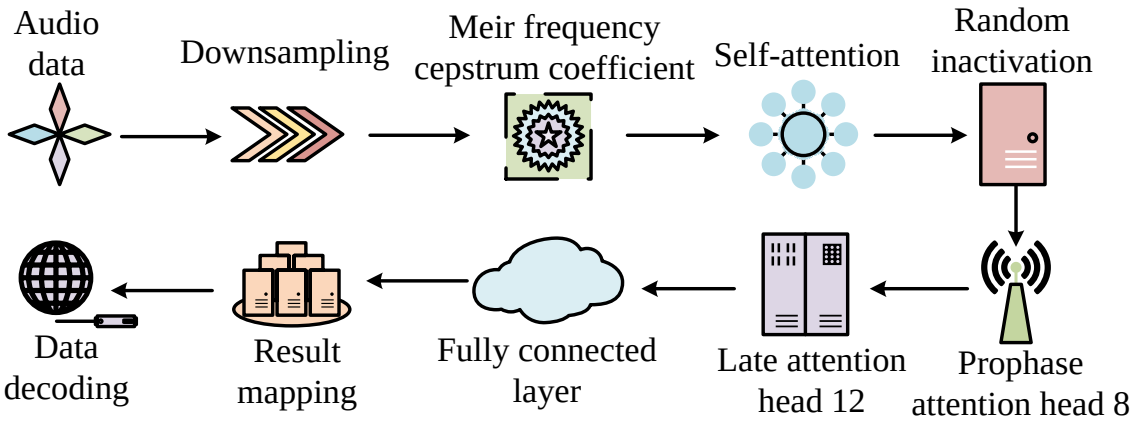

Figure 1: Specific flow of piano music transfer system

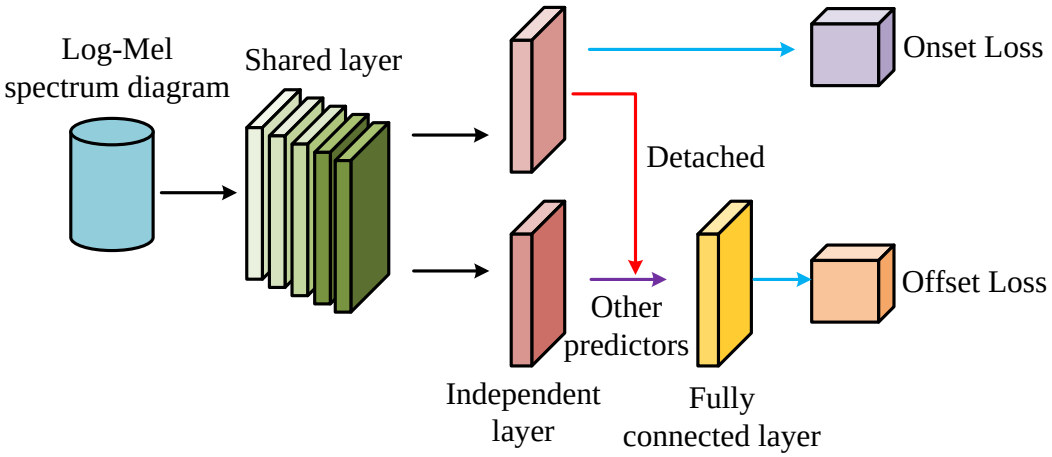

Figure 2: Note starting point and frame level pitch recognition module

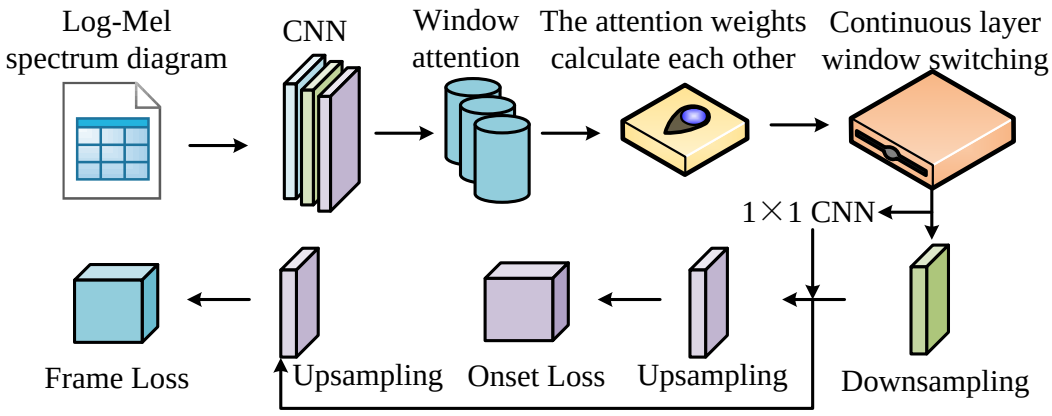

Figure 3: The introduction of transfer window attention into deep learning networks

In Figure 3, the input features are up-sampled by a CNN, and after the sampling is completed, they enter the window attention layer, which divides the information into windows of multiple sizes, and the elements within the windows can only compute the attention weights with each other. In the Shift-Window layer, the computational efficiency is improved by restricting the self-attention computation to non-overlapping localized windows while allowing cross-window connections. Window switching between consecutive layers is performed to improve the computational efficiency of the model and memory usage. After the switching is completed, one more downsampling and multiple up-sampling are performed to obtain the loss of the note onset information and the

loss of the frame-level pitch. Multiple upsampling solves the imbalance problem in the data by sampling and calculating multiple times at the note level and aggregating the final calculation results into samples, and solving the imbalance problem in the data by multiple upsampling. Up-sampling is used to increase the sampling rate of a signal by increasing the number of sampling points, while down-sampling is used to decrease the sampling rate of a signal by decreasing the number of sampling points.

2.2 Music melody extraction based on temporal harmonic characteristics

When processing audio signals, it is necessary to consider their variability and continuity on the time axis to ensure the smoothness of the audio signal. In addition to timing, audio signals also have harmonic characteristics, that is, additional harmonic components caused by nonlinear elements in the output signal that are more than the input signal. The timing and harmonic characteristics of the signal are crucial for extracting the melody of music [18]. Therefore, the study improves the transcription system based on two characteristics and proposed the Time-series Harmonic Graph Deep Learning Network (THGDLN) as shown in Figure 4.

In Figure 4, after obtaining the initial audio information, noise reduction processing is performed to enter the temporal harmonic graph network to construct the temporal and harmonic information models of the audio. Softmax is used as the activation function to output the pitch curve of the audio signal. The pitch sequence of the signal is adjusted using fine-grained functions to obtain the final musical melody. The extraction of harmonic information from audio signals adopts an undirected graph, where the nodes in the graph represent the frequency points of the spectrum, and the edges of the graph represent the connection relationships between signals with the same pitch but different frequencies. The timing information of the signal is extracted using a gate-controlled loop unit with a simple structure, short running time, and less susceptibility to gradient vanishing. The specific extraction process is shown in Figure 5.

In Figure 5, the spectral information from the previous moment is inputted, and the input value is mapped between [0,1] using the Sigmond activation function. Then, the state information from the previous moment is ignored through a reset gate. The Tanh function is used to obtain a larger output range while preserving the characteristics of the input data. Sigmond activation function can reduce the problem of gradient vanishing by normalizing or standardizing the input features so that they are distributed in a smaller range. To ensure Tanh function to prevent the gradient explosion problem, the gradient trimming technique is applied to limit the size of the gradient during the training process. The update gate controls the degree of retention of old information, determines how the information from the previous time and the current time are integrated, and weights the reset gate, update gate, and stored information from the current time to obtain the output features of the current time [19]. The music melody curve obtained by THGDLN occasionally has some abrupt changes, resulting in errors between the predicted and the true values. In this study, median filtering is used to eliminate the abrupt changes in the melody, and the calculation is shown in equation (7).

$$Y_i = me\left(s_{i-r}, s_{i-r+1}, ..., s_i, ..., s_{i+r}\right) \tag{7}$$

In equation (7), Y_i represents the filtering result, me represents sorting the data points and taking the middle value, r represents the radius of the filter, and i represents the current sitting position. According to relevant research literature, a filter radius that is too large will eliminate normal values, while a radius that is too small will result in incomplete elimination of abrupt points. Therefore, a radius size of 7 is chosen. The pitch sequence output by THGDLN is relatively discrete, and the conversion between the pitch sequence and the signal frequency is shown in equation (8).

$$f_z = 2^{(p-69)/12} \cdot 440 \tag{8}$$

In equation (8), f_z represents the converted frequency, and p represents the output pitch sequence value. However, in practical applications, each pitch value corresponds to a wide frequency range, and there is an error between the frequency converted by equation (8) and the actual frequency value, resulting in a deviation in the prediction of music melody. Therefore, it is necessary to optimize the converted frequency. The study uses a significance function for optimization, and the optimization steps are as follows. Firstly, the short-time Fourier transform is used for spectral transformation to obtain the time-frequency data of the signal, and the calculation is shown in equation (9).

$$X_l(k) = \sum_{n=0}^{M-1} w(n)x(n+lH)\exp\left(-j\frac{2\pi}{N}k_f n\right) \tag{9}$$

In equation (9), $X_l(k)$ represents the frequency spectrum composed of time-frequency data, M represents the window length of the short-time Fourier transform spectrum, $w(n)$ represents the window function of the short-time Fourier transform, l represents the frame number of the current time sequence, H represents the frame shift, k_f denotes the frame number of the time sequence in the spectrum, and N represents the number of short-time Fourier transform

points. After completing the spectrum transformation, instantaneous amplitude frequency correction is performed as shown in equation (10).

$$\begin{cases} \hat{f}_i = \left(k_i + \kappa\left(k_i\right)\right)\dfrac{f_s}{N} \\ \kappa\left(k_i\right) = \dfrac{N}{2\pi H}\varepsilon\left(\phi_l\left(k_i\right) - \phi_{l-1}\left(k_i\right) - \dfrac{2\pi H}{N}k_i\right) \end{cases} \tag{10}$$

In equation (10), $\hat{f}$ represents the corrected instantaneous frequency, k_i represents the point with the highest local absolute value in the spectrum, f_s represents the signal sampling rate, $\kappa\left(k_i\right)$ represents the frequency offset distance, ε represents the main value function, and $\phi_l\left(k_i\right)$ represents the phase spectrum at frame number l. After completing the amplitude frequency correction, it constructs the saliency function and divides the spectrum into 600 different frequency points. The calculation for each frequency point is shown in equation (11) [20].

$$B\left(\hat{f}\right) = \left\lfloor 120 \cdot \log_2\left(\frac{\hat{f}}{55}\right) + 1 \right\rfloor \tag{11}$$

In equation (11), $B\left(\hat{f}\right)$ represents the position of frequency points, and the significance value of each frequency point is calculated as shown in equation (12).

$$Sign\left(B\right) = \sum_{h=1}^{10}\sum_{i=1}^{3} T\left(\hat{a}_i\right) \cdot w\left(B, h, \hat{f}_i\right) \cdot \left(\hat{a}_i\right) \tag{12}$$

In equation (12), $Sign\left(B\right)$ denotes the significance value of the frequency point, $\left(\hat{a}_i\right)$ denotes the linear amplitude, $T\left(\hat{a}_i\right)$ denotes the amplitude threshold function, h denotes the harmonic number, and $w\left(B, h, \hat{f}_i\right)$ denotes the weight function. The value of the amplitude threshold function is shown in equation (13).

$$\begin{cases} T\left(\hat{a}_i\right) = 1, if\ 20\log_{10}\left(\dfrac{\hat{a}_m}{\hat{a}_i}\right) < 40 \\ T\left(\hat{a}_i\right) = 0, otherwise \end{cases} \tag{13}$$

In equation (13), $\hat{a}_m$ represents the maximum amplitude in the frame where the frequency point is located, and the value of the weight function is shown in equation (14).

$$\begin{cases} w\left(B, h, \hat{f}_i\right) = \cos^2\left(\delta \cdot \dfrac{\pi}{2}\right) \cdot 0.8^{h-1}, if\ \left|\delta\right| < 1 \\ w\left(B, h, \hat{f}_i\right) = 0, if\ \left|\delta\right| > 1 \end{cases} \tag{14}$$

In equation (14), δ represents the half tone distance between the harmonic and the frequency center spectrum, and the value of the weight function decreases continuously with the increase of harmonic order. After optimizing using the saliency function, the distinction between harmonics and fundamental frequencies in the spectrum becomes more distinct, and the curve is also clearer. The smooth process of the transcribed music melody is shown in Figure 6.

In Figure 6, the initial pitch sequence of the input music is transformed into a frequency sequence, and median filtering is used to eliminate abrupt changes. The position of the frequency points is transformed using a fine-grained saliency function to determine the magnitude of the transformed frequency point saliency value compared to the saliency values of adjacent frequency points. If the saliency value of adjacent frequency points is larger, the adjacent frequency point is selected as the new frequency point. Otherwise, no correction is made. Finally, the new frequency point is transformed into a frequency output, and the frequency point transformation calculation is shown in equation (15).

$$f_i^{new} = 2^{\left(B_i^{\max} - 1\right)/120} \cdot 55 \tag{15}$$

In equation (15), f_i^{new} represents the frequency converted from the new frequency point, and $B_i^{\max}$ represents the position of the frequency point with the highest significance. The processor used in the study is i7-12400F with 2.5GHz CPU frequency and 6 cores and 12 threads. The study uses the loss on the validation set no longer decreasing as an early stopping criterion. When the loss on the validation set no longer decreases significantly after multiple consecutive epochs, the model can be considered to have started overfitting the training data.

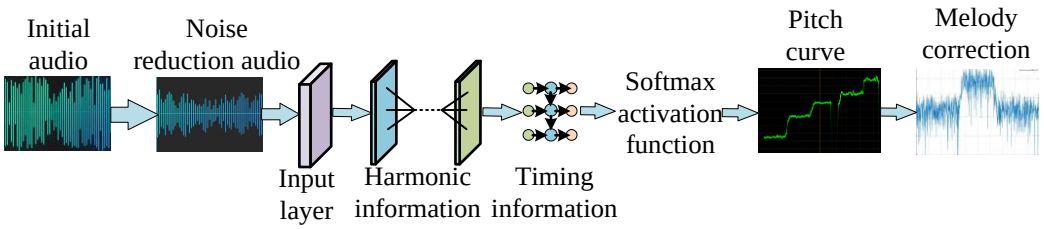

Figure 4: Specific structure of THGDLN

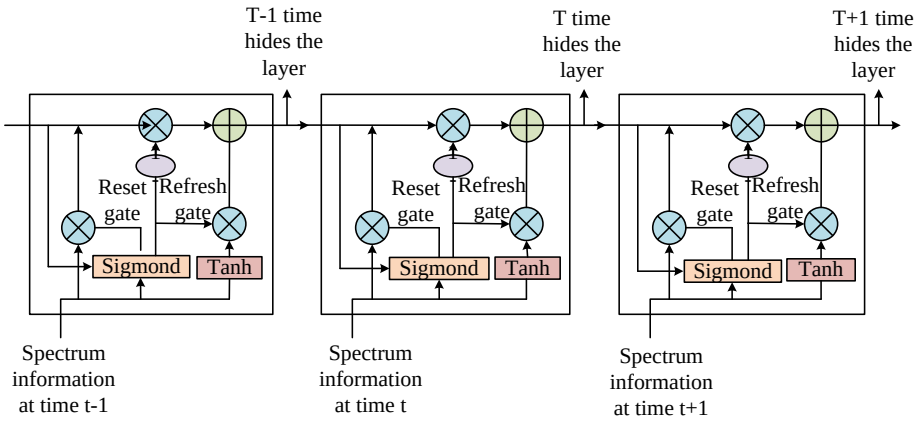

Figure 5: Sequence information extraction process

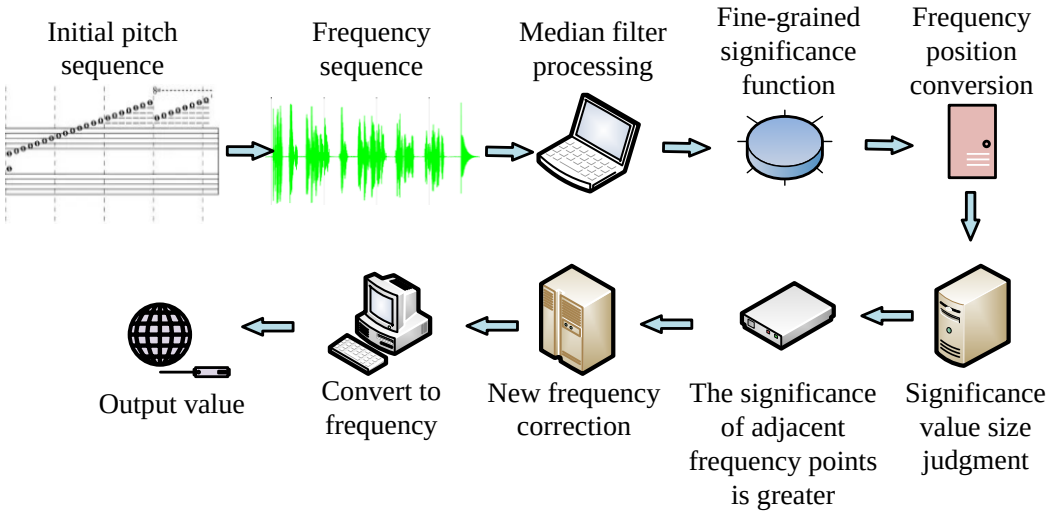

Figure 6: Smooth flow of music melody transfer

3 Results

The study conducted experiments using the MAESTRO dataset, which is a collaborative dataset created by international piano competition organizers. It contains approximately 200 hours of paired audio and MIDI records, covering different performance styles. The data format is universal, easy to process and convert, and MIDI data includes keystroke speed and positions of sustain/slow/bass pedals. The comparative dataset was the MedleyDB dataset. The comparative algorithms used in the study included Multiple Column Deep Neural Networks (MCDNN), Deep Neural Networks (DNNs), and Bayesian Harmonic Model (BHM). In addition to common accuracy, recall, and F1 score, the experimental evaluation indicators also selected Raw Chroma Accuracy (RCA), Raw Pitch Accuracy (RPA), and Overall Accuracy (OA) as performance evaluation

indicators. The impact of frequency point count on the performance of THGDLN algorithm is shown in Figure 7.

In Figure 7 (a), the overall recognition accuracy of the algorithm gradually improved with the increase of frequency points. When the frequency point was 600, the OA value of the THGDLN algorithm tended to converge, reaching a maximum of 89.76%, which was 4.72% and 3.65% higher than the BHM and MCDNN algorithms, respectively. In Figure 7 (b), the RPA of the algorithm gradually improved with the increase of frequency points. The convergence speed was faster before the frequency point reached 600, and slowed down after 600. The maximum value of THGDLN algorithm was 87.14%, which was 6.86% and 4.25% higher than BHM and MCDNN algorithms, respectively. Therefore, choosing a frequency point of 600 only increased accuracy by 1% when the frequency point was between 600 and 800, but it would significantly reduce computation speed and increase operating costs. The impact of frequency search range on the performance of THGDLN algorithm is shown in Figure 8.

In Figure 8 (a), the overall recognition accuracy of the algorithm first increased and then decreased with the increase of frequency search range. When the search range was 0.5, the THGDLN algorithm achieved the maximum OA value of 89.52%, which was 2.58% and 2.35% higher than the BHM and MCDNN algorithms, respectively. BHM achieved the maximum value when the search range was 1.0. In Figure 8 (b), the RPA of the algorithm first increased and then decreased with the increase of the frequency search range. When the search range was 0.5, the THGDLN algorithm achieved the maximum RPA value of 85.94%, which was 2.23% and 1.06% higher than the BHM and MCDNN algorithms, respectively. Therefore, the frequency search range of the algorithm was set to 0.5 semitones. When the number of selected frequency points was 600 and the frequency point search range was 0.5 semitones, the performance of the algorithm before and after melody smoothing was compared as shown in Figure 9.

In Figure 9 (a), the THGDLN algorithm showed significant improvements in RCA, RPA, and OA after melody smoothing. The OA value increased from 79.37% to 82.77% at a runtime of 12 seconds, an increase of 3.4%, the RCA value increased from 80.52% to 87.72%, an increase of 7.2%, and the RPA value increased from 78.05% to 82.14%, an increase of 4.09%. The impact of different network structure layers on model performance in the note starting point and frame level pitch recognition module is denoted in Table 2.

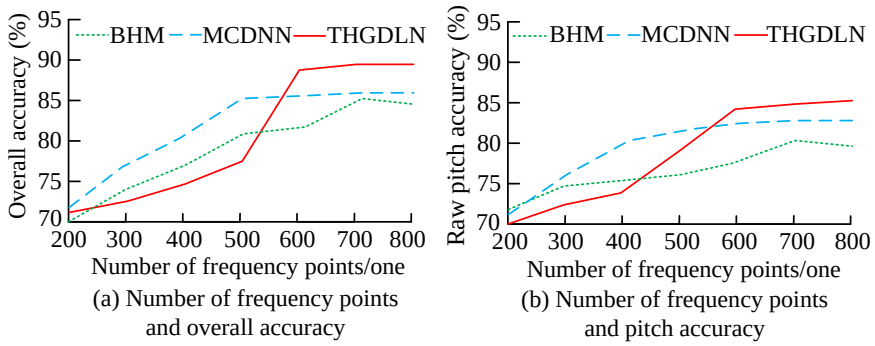

Figure 7: Influence of different frequency points on the performance of THGDLN algorithm

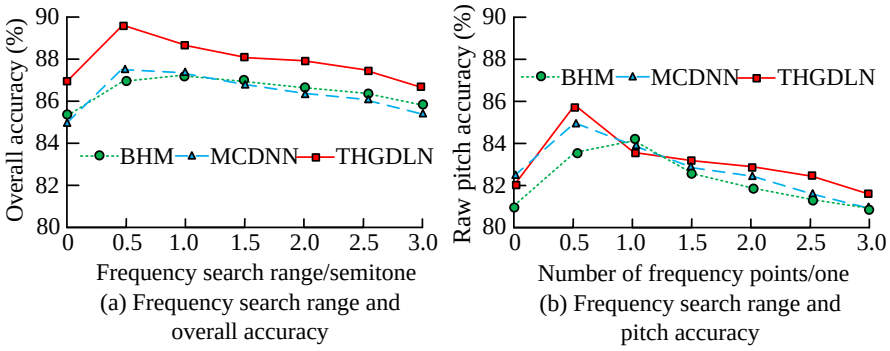

Figure 8: Influence of frequency search range on the performance of THGDLN algorithm

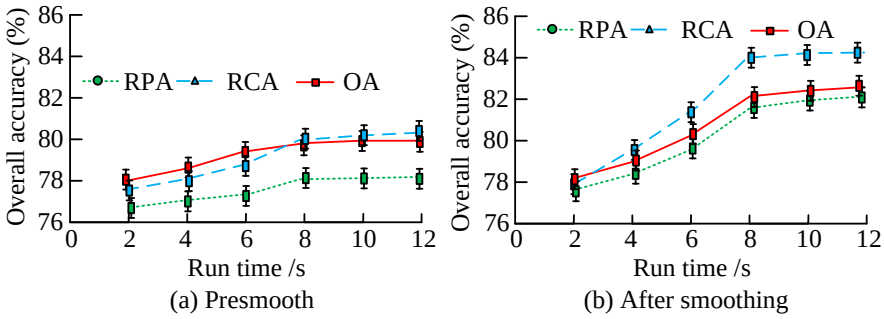

Figure 9: Performance comparison of algorithms before and after smoothing

Table 2: Influence of different network structure layers on model performance

Number of structural layers	Note origin Accuracy rate(%)	Recall rate(%)	F1(%)	End of note Accuracy rate(%)	Recall rate(%)	F1(%)
511	99.15	90.48	94.36	78.92	78.09	79.21
411	98.13	87.62	91.58	78.16	78.45	79.18
311	98.08	86.35	90.58	78.01	77.94	78.24
211	97.87	85.64	89.35	77.84	77.32	78.02
521	99.09	90.52	92.15	78.18	78.19	77.25

Table 3: Performance comparison of different algorithms

Algorithm	MAESTRO Accuracy rate(%)	Recall rate(%)	F1(%)	MedleyDB Accuracy rate(%)	Recall rate(%)	F1(%)
MCDNN	97.04	88.21	92.15	85.42	83.04	84.65
DNNs	85.48	82.19	84.18	78.16	77.36	74.25
BHM	95.14	87.29	89.17	81.09	79.15	79.08
THGDLN	99.15	90.48	94.36	88.91	88.29	89.21

In Table 2, network structure 511 represents 5 shared layers, 1 independent layer, and 1 fully connected layer. When the network structure was 511, the performance of the starting and ending points of the notes was optimal. As the number of shared layers decreased, the performance gradually decreased. When the shared layers were reduced to 2, the accuracy, recall, and F1 score decreased by 1.28%, 4.84%, and 5.11%, respectively. Adding an independent layer resulted in a decrease of 0.06%, -0.04%, and 2.21% in various performance indicators. When the shared layer at the end of the note dropped to 2 layers, the predictive performance decreased by 1.08%, 0.77%, and 1.19%, respectively. Increasing the independent layer by one layer resulted in a decrease of 0.74%, -0.1%, and 1.96%, respectively. The performance comparison of algorithms in different datasets is shown in Table 3.

In Table 3, the THGDLN algorithm had the best performance in all aspects. In the MAESTRO dataset, its accuracy, recall, and F1 score were 2.11%, 2.27%, and 2.21% higher than the second-best algorithm, respectively. In the MedleyDB dataset with higher melody complexity, the accuracy, recall, and F1 score of the THGDLN algorithm were 3.49%, 5.25%, and 4.56% higher than the second-best algorithm, respectively, indicating that the THGDLN algorithm had stronger processing ability for complex audio and higher accuracy after conversion. Ablation experiments were performed on the algorithm, and the performance comparison is shown in Figure 10.

In Fig. 10(a), after removing the window attention, the maximum transcription accuracy of the THGDLN algorithm decreased from 99.15% to 91.08%, which was a decrease of 8.07%, and after removing the note starting point and frame-level pitch recognition module, the

maximum transcription accuracy of the THGDLN algorithm decreased to 82.39%, which was a decrease of 16.76%, and the algorithm's convergence was further back. Results indicated that the note starting point and frame-level pitch recognition module not only improved the transcription performance of the algorithm, but also enhanced the computational efficiency. In Figure 10(b), the F1 maximum of the THGDLN algorithm decreased by 9.53% after removing the window attention, and the F1 maximum of the algorithm decreased by 12.65% after removing the note starting point and frame-level pitch recognition module. The self-attention mechanism takes into account the relationship of each note to all other notes. This mechanism not only focuses on the current note, but also takes into account the information before and after, which helps the model to better understand the contextual information in the music, thus improving the accuracy and smoothness of the transcription. The self-attention mechanism can effectively deal with the harmonic overlap problem by capturing the global dependencies in the music signal and adjusting the degree of attention to different parts.

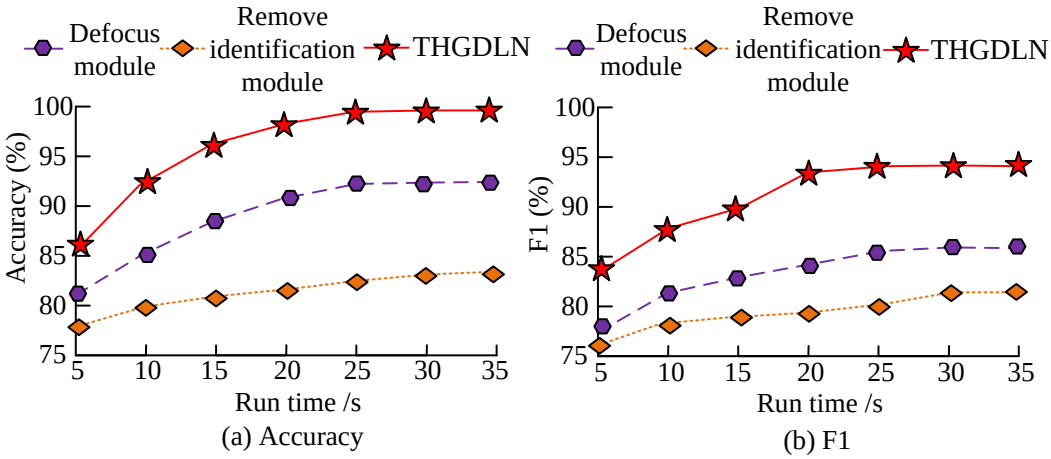

Figure 10: Comparison of experimental performance of ablation algorithms

Table 4: Comparison of the time complexity of the three algorithms

Sample size	Model type	Average processing time (ms)	Standard deviation (ms)	Time complexity O(n2) evaluation
1000	THGDLN	21.4	2.2	Lower
	MCDNN	28.6	3.0	Normal
	DNN	18.2	2.3	Lower
2000	THGDLN	64.5	5.2	Lower
	MCDNN	120.3	7.7	Normal
	DNN	53.6	5.4	Lower
3000	THGDLN	176.2	8.3	Normal
	MCDNN	268.4	11.9	Higher
	DNN	151.8	8.5	Lower

To analyze the computational complexity of THGDLN algorithm, the processing time of THGDLN algorithm under different data amounts was measured through experiments. The computational complexity pairs of THGDLN algorithm are shown in Table 4.

In Table 4, when the sample data volume was 1000, 2000, and 3000, the average processing time of THGDLN algorithm was 21.4ms, 64.5ms, and 176.2ms respectively, which were all lower than MCDNN algorithm and only 3.2ms, 10.9ms, and 24.4ms higher than common DNN algorithm. The results show that THGDLN algorithm has greater computational complexity and higher efficiency than other improved methods.

4 Discussion

The study proposed a piano MT algorithm based on SADLN for existing piano MT algorithms, which has certain errors and other problems. The experimental results showed that the THGDLN algorithm proposed by the study outperformed other algorithms in terms of

accuracy, recall, and F1 score, and the three metrics were 2.11%, 2.27%, and 2.21% higher than the second-best algorithm, respectively. The self-attention mechanism in the THGDLN algorithm took into account the relationship between each note and all other notes, and considered the information of the notes in the preceding and following contexts. This can help the model better understand the contextual information in the music and improve the accuracy and smoothness of the transcription. The window attention mechanism, on the other hand, by applying a local window on the time series data, enabled the model to focus on the information within a specific time period, thus improving the understanding and processing of the music signal. The self-attention mechanism could effectively deal with the harmonic overlap problem by capturing the global dependencies in the music signal and adjusting the degree of attention to different parts. The average processing time of the THGDLN algorithm was only 3.2ms higher than that of the base algorithm DNN when the amount of data was low, and was 7.2ms lower than the average of the existing algorithm MCDNN. As the amount of data increased, the computational complexity of the THGDLN algorithm increased in a significantly lower trend than that of the MCDNN algorithm. The algorithm is slightly less sensitive to the order of recognition. The recognition results between frames are prone to breakpoints. The transcription system that adopts the general method also needs to carry out corresponding processing such as median filtering, and the corresponding details need to be improved.

5 Conclusion

A piano MT algorithm based on SADLN was proposed to address the problems of insufficient accuracy and low transcription accuracy of existing piano MT algorithms. The experiment findings indicated that as the number of frequency points increased, the OA of the algorithm gradually improved. At a frequency point of 600, it tended to converge, with OA values 4.72% and 3.65% higher than BHM and MCDNN algorithms, respectively. The RPA values of THGDLN algorithm were 6.86% and 4.25% higher than BHM and MCDNN algorithms, respectively. Therefore, although there was a slight improvement in algorithm performance after setting the frequency points to 600600, it would reduce computational depth and increase operating costs. The OA and RPA values of the THGDLN algorithm first increased and then decreased with the increase of the frequency search range. When the search range was 0.5, the max value was obtained. The OA values were 2.58% and 2.35% higher than those of the BHM and MCDNN algorithms, respectively, and the RPA values were 2.23% and 1.06% higher than those of the BHM and MCDNN algorithms, respectively. The THGDLN algorithm improved OA, RCA, and RPA by 3.4%, 7.2%, and 4.09%, respectively, after smoothing. As the number of shared layers in the network structure decreased, the performance of the algorithm gradually declined. When the shared layer was reduced to 2 layers, the accuracy, recall, and F1 score decreased by 1.28%, 4.84%, and 5.11%, respectively. In the MAESTRO dataset, the accuracy, recall, and F1 score of the THGDLN algorithm were 2.11%, 2.27%, and 2.21% higher than the second best algorithm, respectively. After removing window attention, the transcription accuracy of the THGDLN algorithm decreased by 8.07%. After removing the note start and frame level pitch recognition modules, the transcription accuracy decreased by 16.76%. There are still some issues with this study, such as the lack of discussion on the algorithm's ability to transcribe synthesized tracks in music creation. In the future, a dataset composed of synthesized music can be added for testing to improve the algorithm's generalization performance.

References

[1] X. Li, Y. Zhang, P. Tiwari, D. Song, B. Hu, M. Yang, Z. Zhao, N. Kumar, and P. Marttinen, "EEG based emotion recognition: A tutorial and review," ACM Computing Surveys, vol. 55, no. 4, pp. 1-57, 2022. https://doi.org/10.1145/3524499

[2] A. Battcock, and M. Schutz, "Emotion and expertise: How listeners with formal music training use cues to perceive emotion," Psychological Research, vol. 86, no. 1, pp. 66-86, 2022. https://doi.org/10.1007/s00426-020-01467-1

[3] J. W. Li, S. Barma, P. U. Mak, F. Chen, C. Li, M. T. Li, M. I. Vai, and S. H. Pun, "Single-channel selection for EEG-based emotion recognition using brain rhythm sequencing," IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 6, pp. 2493-2503, 2022. https://doi.org/10.1109/JBHI.2022.3148109

[4] M. J. Lucia-Mulas, P. Revuelta-Sanz, B. Ruiz-Mezcua, and I. Gonzalez-Carrasco, "Automatic music emotion classification model for movie soundtrack subtitling based on neuroscientific premises," Applied Intelligence, vol. 53, no 22, pp. 27096-27109, 2023. https://doi.org/10.1007/s10489-023-04967-w

[5] V. Chaturvedi, A. B. Kaur, V. Varshney, A. Garg, G. S. Chhabra, and M. Kumar, "Music mood and human emotion recognition based on physiological signals: A systematic review," Multimedia Systems, vol. 28, no. 1, pp. 21-44, 2022. https://doi.org/10.1007/s00530-021-00786-6

[6] J. S. Gómez-Cañón, N. Gutiérrez-Páez, L. Porcaro, A. Porter, E. Cano, P. Herrera-Boyer, A. Gkiokas, P. Santos, D. Hernández-Leo, C. Karreman, and E. Gómez, "TROMPA-MER: An open dataset for personalized Music Emotion Recognition," Journal

of Intelligent Information Systems, vol. 60, no. 2, pp. 549-570, 2023. https://doi.org/10.1007/s10844-022-00746-0

[7] N. He, and S. Ferguson, "Music emotion recognition based on segment-level two-stage learning," International Journal of Multimedia Information Retrieval, vol. 11, no. 3, pp. 383-394, 2022. https://doi.org/10.1007/s13735-022-00230-z

[8] M. A. Qureshi, M. Asif, M. F. Hassan, G. Mustafa, M. K. Ehsan, A. Ali, and U. Sajid, "A novel auto-annotation technique for aspect level sentiment analysis," Computers, Materials and Continua, vol. 70, no. 3, pp. 4987-5004, 2022. https://doi.org/10.32604/cmc.2022.020544

[9] R. Orjesek, R. Jarina, and M. Chmulik, "End-to-end music emotion variation detection using iteratively reconstructed deep features," Multimedia Tools and Applications, vol. 81, no. 4, 5017-5031, 2022. https://doi.org/10.1007/s11042-021-11584-7

[10] G. Yin, S. Sun, D. Yu, D. Li, and K. Zhang, "A multimodal framework for large-scale emotion recognition by fusing music and electrodermal activity signal," ACM Transactions on Multimedia Computing, Communications, and Applications, vol. 18, no. 3, pp. 1-23, 2022. https://doi.org/10.1145/3490686

[11] S. Venkata Lakshmi, K. Sujatha, and J. Janet, "A hybrid discriminant fuzzy DNN with enhanced modularity bat algorithm for speech recognition," Journal of Intelligent & Fuzzy Systems, vol. 44, no. 3, pp. 4079-4091, 2023. https://doi.org/10.3233/JIFS-212945

[12] A. Ríos-Vila, D. Rizo, J. M. Iñesta, and J. Calvo-Zaragoza, "End-to-end optical music recognition for pianoform sheet music," International Journal of Document Analysis and Recognition, vol. 26, no. 4, pp. 347-362, 2023. https://doi.org/10.1007/s10032-023-00432-z

[13] R. Shashidhar, S. Patilkulkarni, and S. B. Puneeth, "Combining audio and visual speech recognition using LSTM and deep convolutional neural network," International Journal of Information Technology, vol. 14, no. 7, pp. 3425-3436, 2022. https://doi.org/10.1007/s41870-022-00907-y

[14] S. Choudhuri, S. Adeniye, and A. Sen, "Distribution alignment using complement entropy objective and adaptive consensus-based label refinement for partial domain adaptation," Artificial Intelligence and Applications, vol. 1, no. 1, pp. 43-51, 2023. https://doi.org/10.47852/bonviewAIA2202524

[15] A. Aristidou, A. Yiannakidis, K. Aberman, D. Cohen-Or, A. Shamir, and Y. Chrysanthou, "Rhythm is a dancer: Music-driven motion synthesis with global structure," IEEE Transactions on Visualization and Computer Graphics, vol. 29, no. 8, pp. 3519-34, 2022. https://doi.org/10.1109/TVCG.2022.3163676

[16] X. Chen, J. Zhou, D. Li, J. Liu, and Y. Dai, T. Zhou, " Enjoyment of Chinese and mathematics and school performance in Chinese children and adolescents," Child Development, vol. 94, no. 1, pp. 126-141, 2023. https://doi.org/10.1111/cdev.13843

[17] F. De Arriba-Pérez, S. García-Méndez, F. Leal, B. Malheiro, and J. C. Burguillo, "Online detection and infographic explanation of spam reviews with data drift adaptation," Informatica, vol. 35, no. 3, pp. 483-507, 2024. https://doi.org/10.15388/24-INFOR562

[18] E. Deruty, M. Grachten, S. Lattner, J. Nistal, and C. Aouameur, "On the development and practice of ai technology for contemporary popular music production," Transactions of the International Society for Music Information Retrieval, vol. 5, no. 1, pp. 35-49, 2022. https://doi.org/10.5334/tismir.100

[19] G. Keerti, A. N. Vaishnavi, P. Mukherjee, A. S. Vidya, G. S. Sreenithya, and D. Nayab, "Attentional networks for music generation," Multimedia Tools and Applications, vol. 81, no. 4, pp. 5179-5189, 2022. https://doi.org/10.1007/s11042-021-11881-1

[20] A. K. Herget, and J. Albrecht, "Soundtrack for reality? How to use music effectively in non-fictional media formats," Psychology of Music, vol. 50, no. 2, pp. 508-29, 2022. https://doi.org/10.1177/03057356219990