Processing piano audio: research on an automatic transcription model for sound signals

Peng Wang[1], Ning Dai[2]
Cangzhou Normal University, Cangzhou, 061001, China
[1]Corresponding author
E-mail: [1]*wpengpw@hotmail.com*, [2]*dn_ning@hotmail.com*

Received 10 July 2024; accepted 14 November 2024; published online 15 December 2024
DOI https://doi.org/10.21595/jme.2024.24345

Abstract. Automatic transcription of sound signals can convert audio to musical notes, which has significant research value. This paper extracted dual-channel constant Q transform (CQT) spectra from piano audio as features. In the design of the automatic transcription model, a CNN was employed to extract local features and then combined with a Transformer model to obtain global features. A CNN-Transformer automatic transcription model was established using a two-layer CNN and three-layer Transformers. Experiments were conducted on the MAPS and MAESTRO datasets. The results showed that dual-channel CQT outperformed short-time Fourier transform (STFT) and mono CQT in auto-transcription. Dual-channel CQT achieved the best results on frame-level transcription for the MAPS dataset, with a P value of 0.9115, an R value of 0.8055, and an F1 value of 0.8551. A sliding window with seven frames yielded the best transcription results. Compared with the deep neural network and CNN models, the CNN-Transformer model demonstrated superior performance, achieving an F1 value of 0.8551 and 0.9042 at the frame level for MAPS and MAESTRO datasets, respectively. These findings confirm the designed model's reliability for automatic piano audio transcription and highlight its practical applicability.

Keywords: piano audio, sound signal, constant Q transform, automatic transcription.

1. Introduction

The inheritance and development of music are very important [1]. With the application of digitalization, informatization, and other methods in the field of music, there has been increasing research on the recording and preservation of music. Automatic transcription of sound signals involves converting raw audio into music notation or musical instrument digital interface (MIDI) data, which plays an important role in various applications [2]. It supports music information retrieval (MIR) [3], enables error checking of audio via scores, and improves the efficiency of score annotation, reducing human resources and the cost of implementation. In music composition, real-time recording of music scores can be achieved through automatic transcription models, facilitating composers' modifications. Automatic sound signal transcription is based on the perspective of signal processing, and many relevant methods have been applied [4]. Simonetta et al. [5] proposed a hidden Markov model-based method for note-level pairing and verified its reliability through experiments on multiple datasets. Meng and Chen [6] employed the Mel-frequency cepstral coefficient (MFCC) for timbre judgment and constant Q transform (CQT) for pitch judgment. They utilized a convolutional neural network (CNN) to achieve automatic transcription and obtained a recognition success rate of 95 %. Alfaro-Contreras et al. [7] studied the effect of combining different modalities (e.g., image, audio) in automatic transcription and found that two modalities could effectively improve the single-mode recognition framework. Lee [8] designed a method using Stein's unbiased risk estimator and eigenvalue decomposition for automatic transcription and found a 2 %-3 % improvement in F1 value compared to traditional approaches. Marolt [9] proposed an automatic transcription method for polyphonic piano music based on an auditory model and adaptive oscillator network. They discovered that the oscillator network can improve the accuracy of neural network transcription. Ryynanen and Klapuri [10]

extracted three acoustic features through a multi-fundamental frequency estimator, implemented music transcription by searching multiple paths in the note model, and achieved 41 % precision and 39 % recall rate. Benetos and Dixon [11] simulated the temporal evolution of music tones and proposed a polyphonic music transcription method based on a hidden Markov model. The method achieved better results than the non-temporal-constrained model in multi-instrument recordings. Ju et al. [12] proposed a method for recognizing the duration of musical notes considering time-varying tempo. They conducted experiments on transcribing 16 monophonic children's songs and achieved matching rates of 89.4 % and 84.8 % for note duration and pitch recognition, respectively. O'Hanlon et al. [13] presented a greedy sparse pursuit approach based on nearest subspace classification, applied it in automatic music transcription, and verified its effectiveness. Cazau et al. [14] developed a Markov model-based general transcription system that can host various prior knowledge parameters. They also created a sound corpus for experiments and performed comparative evaluations on the transcription results. Wang et al. [15] calculated the timbre vector based on the short-time Fourier transform and peak detection algorithm and classified timbres using a support vector machine to achieve automatic transcription. The method achieved a hit rate of 73 % on a small database containing two timbres. The piano is a keyboard instrument [16]. Given the wide range of registers and common usage in solo and accompaniment, the study of automatic transcription modeling for piano audio holds paramount importance. However, the performance of many transcription systems is still significantly lower than that of human experts [17], and some automatic transcription models for piano audio have poor transcription effects, which cannot yet meet the needs of practical applications. In order to obtain a more efficient automatic transcription mode, this paper processed piano audio using the CQT and extracted features. Subsequently, an automatic transcription model based on CNN and Transformer was designed. The effectiveness of the proposed approach was verified through experiments on a dataset. The approach offers a novel and reliable technique for practical piano audio processing and provides theoretical support for the further study of CNN and Transformer architectures in automatic sound signal transcription, which is beneficial to the further development of automatic transcription models. Moreover, this article also provides a theoretical reference for the transcription of other music, which is advantageous to the more profound development of the field of music information retrieval.

2. Piano audio and constant-Q transform

The piano produces sound by striking the keys and driving the strings to vibrate. Its sound signal is a superposition of a series of sinusoidal signals. The core sounding part can be called the fundamental, which determines the pitch. The corresponding frequency is the fundamental frequency. By identifying the fundamental frequency of piano audio, it is possible to identify the pitch. Scientific pitch notation is generally used in music based on the letters C, D, E, F, G, A, and B. In the digital domain, MIDI [18] is used to indicate. The relationship between the MIDI and fundamental frequency f is written as Eq. (1):

$$p = 69 + 12 * \log_2\left(\frac{f}{440}\right). \tag{1}$$

In 88-key pianos, the pitch range is A0-C8, and the MIDI numbering range is 21-108, corresponding to a fundamental frequency range of 27.5-4186 Hz.

Before transcription, extracting features from the original piano audio data is necessary to achieve the signal transformation from the time domain to the frequency domain and provide reliable feature input for the automatic transcription model. The short-time Fourier transform (STFT) has an extensive range of applications in sound signal processing [19] and has good performance in various speech recognition tasks [20]. The computational formula can be written as Eq. (2):

$$x[k] = \sum_{n=0}^{N-1} w[n]x[n]e^{-j2\pi kn/N}, \tag{2}$$

where $w[n]$ is the window function, $x[n]$ is the original sound signal, and N is the window length, which directly affects the time and frequency resolution of the spectrogram. The main drawback of STFT is that it obtains the same frequency resolution at high and low frequencies, which is not conducive to recognizing piano tones. Therefore, this paper chooses another frequency domain feature, CQT [21], as the piano audio feature.

CQT is characterized by its window length changing with frequency, thus ensuring high resolution at low frequencies. In CQT, there exists a quality factor Q, as shown in Eq. (3):

$$Q = \frac{f_k}{\delta_{f_k}}, \tag{3}$$

where δ_{f_k} is the bandwidth of the k-th filter. Combined with Q, the window length is written as $N = N[k] = Q \cdot (f_s/f_k)$, where f_s is the sampling frequency. The final CQT formula can be written as Eq. (4):

$$x[k] = \frac{1}{N[k]} \sum_{n=0}^{N[k]-1} w[k,n]x[n]e^{-j2\pi Qn/N[k]}. \tag{4}$$

In automatic transcription, mono sampling data are mainly used for analysis. To obtain comprehensive characteristics, this paper uses a dual-channel approach to read the data and obtains a dual-channel CQT spectrogram as input to the transcription model.

3. Automatic transcription model for sound signals

3.1. Convolutional neural network

CNN is a deep learning model [22], which performs well in processing high-dimensional data. In addition to a low network complexity, CNNs possess excellent extraction capabilities for deep features. The composition of a CNN mainly includes convolutional layers and pooling layers. Firstly, the convolutional layer is employed to extract the implicit features of the data. The output is expressed as Eq. (5):

$$X_j = f\left(\sum_j W_i^j * x_j + b_j\right), \tag{5}$$

where $f(\cdot)$ is the activation function, W_i^j is the convolution kernel parameter, b_j is the bias, and $*$ denotes the convolution operation.

The pooling layer is used for feature downsampling to reduce the amount of computation, and there are two common pooling operations:

1) Maximum pooling: $MP_c(x) = max_{x_i \in C} x_i$, $x_i \in C$.

2) Mean pooling: $AP_c(x) = \sum_{i=1}^{N} x_i/N$, $x_i \in C$.

In the above equations, x is the input tensor, C is the pooling region, and N is the quantity of elements in C.

The sound signal is convolved and pooled to get the features, which are then transformed into one-dimensional vectors by a fully connected (FC) layer. After the FC layer, the probability of

each classification is calculated using a softmax layer. The output of the k-th neuron can be written as Eq. (6). c_i is the input signal, and n is the number of neurons:

$$P_k = \frac{\exp(c_k)}{\sum_{i=1}^{n} \exp(c_i)}. \tag{6}$$

3.2. CNN-Transformer transcription model

One of the main advantages of a CNN is its ability to capture local features in a sequence. However, it has limitations when it comes to processing long sequences. To address this issue, CNN is often integrated with a recurrent neural network (RNN) to handle time series features. However, this combined architecture tends to have low training speeds. The Transformer model, which utilizes an encoder-decoder structure, offers high training speeds and has shown excellent performance in various tasks such as machine translation and speech recognition [23]. The present study proposes combining the Transformer model with a CNN to construct an automatic transcription model.

The Transformer model relies on a multi-head self-attention mechanism [24] to effectively capture sequence information. Firstly, a linear transformation is performed on input sequence x to get three sequences, Q (Query), K (Key), and V (Value), and then the attention is obtained, as shown in Eq. (7):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{7}$$

where Q, K, and V are the Query, key, and Value indicators, d_k refers to the dimension of Key. The multi-head attention splices the output of each attention and passes it into the linear layer for linear transformation. The formula of the multi-head attention is shown in Eq. (8):

$$MultiHeadAttention = concat(head_1, head_2, \cdots, head_h)W^O, \tag{8}$$
$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right), \tag{9}$$

where W is the weight matrix and h is the number of heads in the attention.

A residual and normalization layer follows the multi-head self-attention layer to avoid network degradation. Then, a two-layer FC layer is employed to obtain the new representation vector, as shown in Eq. (10):

$$FCN(x) = max(0, xW_1 + b_1)W_2 + b_2, \tag{10}$$

where x is the input vector, W and b are the weight and bias of every layer.

In the Transformer model, residual connections are used between the input and output of each layer to alleviate the problem of gradient vanishing. Layer normalization is then applied after the residual connection to accelerate training and improve robustness, making the model more stable.

The CNN can effectively capture local features in a sequence, and the Transformer model can effectively treat the logical dependency relationship. Therefore, combining a CNN with a Transformer can better capture both local and global information in sound signals, thereby improving the effectiveness of automatic transcription for piano audio. The structure of the designed CNN-Transformer transcription model is illustrated in Fig. 1.

The transcription model comprises two CNN layers and three Transformer layers. The dual-channel CQT spectrogram extracted in the last section is used as the model input. There are two CNN layers on the left. Each CNN layer has a 3×3 convolutional layer followed by a 2×2 maximum pooling layer. The input size of the first CNN is 128×2, the size of the convolution kernel is 3×3, and the size of the output channel is 32. The input size of the pooling layer is 128×2,

the size of the pooling kernel is 2×2, the step size is 2, and the output size is 64×1. The input size of the second CNN is 64×32, the size of the convolution kernel is 3×3, and the number of output channels is 64. The input size of the second pooling layer is 64×32, the size of the pooling size is 2×2, the step size is 2, and the output size is 32×1. After the processing by the two CNN layers, the feature vector obtained is 32×64. The resulting feature vectors are fed into the Transformer layer on the right for further processing. Each Transformer layer comprises four multi-head attention layers and two FC layers. After feature extraction by three layers of Transformer, the outputs are concatenated in the FC layers. Finally, the features are input into the softmax layer to recognize the piano audio.

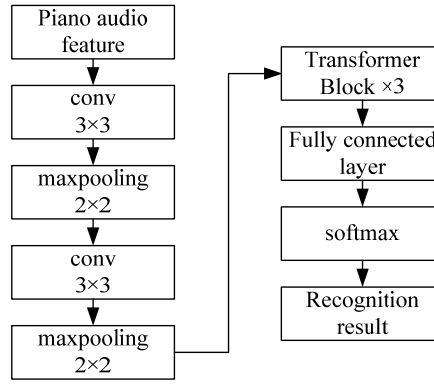

Fig. 1. The CNN-Transformer transcription model

4. Results and analysis

4.1. Experimental setup

The experiments were performed on a computer with a Windows 10 system, an Intel(R) Core(TM)i7-10750H central processing unit (CPU), and 16 GB memory. The software development environment was PyTorch and Python 3.8. The CNN-Transformer model was trained using the Adam algorithm. The initial learning rate was 0.005. Through continuous optimization, the learning rate was determined to be 0.0001. The maximum number of iterations was 1500, and the batch size was 64. To extract the dual-channel CQT spectrum, the Python library Librosa [25] was utilized. Librosa is dedicated to audio and music signal analysis, providing functions such as audio feature extraction and visualization. It has a wide range of applications in music information retrieval. Each channel's CQT spectrum had 356 dimensions, and a sliding window with a length of 7 was applied to capture spectrogram snapshots. Consequently, the final input feature was a 356×9×2 matrix, while the output was an 88-dimensional note.

4.2. Experimental data were obtained from two databases:

MAPS dataset [26]: it includes nine directories, seven of which are synthesized piano music, and two directories (cl and am) are actual piano recordings. The first seven directories were used as the training set, and cl and am were used as the test set.

MAESTRO dataset [27]: it includes piano performance data from the International Piano-e-Competition, with a size of about 90 GB, from which 283 tracks from 2017 were taken for experiments. Among them, 226 tracks were taken for training, and 57 tracks were used for testing.

The transcription effect was evaluated using the mir_eval library [28], and the generalized indicators included: precision: $P = TP/(TP + FP)$, recall rate: $R = TP/(TP + FN)$, F1 value: $F1 = (2 \times P \times R)/(P + R)$, where TP is the quantity of positive samples with correct

predictions, TP is the quantity of positive samples with wrong predictions, and FN is the quantity of negative samples with wrong predictions.

5. Analysis of results

Using the MAPS dataset, the transcription effects of STFT, mono CQT, and dual-channel CQT were compared, and the outcomes are presented in Table 1.

Table 1. Transcription results for different feature inputs

		STFT	Mono CQT	Dual-channel CQT
Frame level	P	0.8312	0.8864	0.9112
	R	0.7521	0.7895	0.8055
	F1	0.7897	0.8351	0.8551
Note (no cutoff)	P	0.8254	0.8741	0.9055
	R	0.7358	0.7719	0.8046
	F1	0.7780	0.8198	0.8521
Complete note	P	0.5694	0.6225	0.6552
	R	0.5127	0.5576	0.5867
	F1	0.5396	0.5883	0.6191

It was seen that the transcription result using STFT was poor, with an F1 value of 0.7894 at the frame level, an F1 value of 0.7780 at the note level (no cutoff), and an F1 value of 0.5396 at the complete note level. These results indicated that STFT failed to effectively differentiate piano notes when used as an input to the transcription model. In contrast, the comparison between mono CQT and dual-channel CQT revealed that the dual-channel CQT achieved better results. At the frame level, the dual-channel CQT achieved an F1 value of 0.8551, showing a 0.02 improvement compared to mono. At the note level (no cutoff), the F1 value was 0.8521, showing a 0.0323 improvement compared to mono. Additionally, the F1 value was 0.6191 at the complete note level, demonstrating a 0.0308 improvement compared to mono. These findings concluded that employing the dual-channel CQT technique led to better outcomes in the automatic transcription of piano audio. In comparing frame level, note (no cutoff), and complete note transcription, it can be seen that the difficulty of transcribing complete notes was the highest, resulting in the worst transcription results. However, the dual-channel CQT used in this article still demonstrated the best performance, with an F1 value of 0.6191, further proving the advantage of using dual-channel CQT as a feature.

Considering the sensitivity of the model to the input window length, various input window lengths were chosen to analyze the impact of the sliding window on transcription results. The frame-level recognition results are presented in Fig. 2.

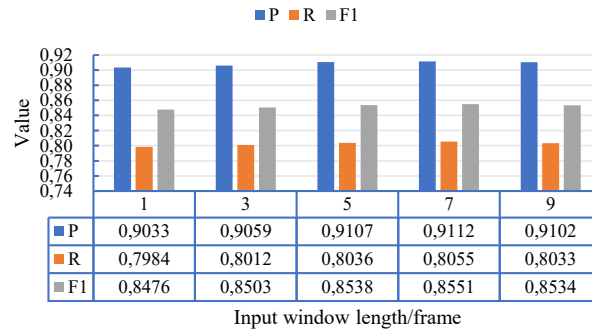

Fig. 2. The sensitivity analysis results of the input window length

It was found that the optimal transcription results (a P value of 0.9112, a R value of 0.8055, an F1 value of 0.8551) were achieved with an input window length of seven frames. When the

number of frames was small, the spectrum contained less information, resulting in poor recognition of piano notes. As the number of frames increased, the amount of information provided in the spectrum gradually increased, leading to a gradual improvement in the model's P value, R value, and F1 value. However, when the growth reached nine frames, the results were inferior to those obtained with seven frames. For example, the F1 value decreased to 0.8524, which was 0.0017 smaller than that when it was seven frames. It is because the large sliding window caused the merging of similar notes, which negatively impacted the recognition results. Therefore, when this model is used for transcription, better results can be obtained when the input window length is 7.

A sensitivity analysis was performed on the CNN layer number in the model. The transcription results of the frame level under different convolutional layers are presented in Table 2.

Table 2. The sensitivity analysis results of the CNN layer number

	1	2	3
P	0.8554	0.9112	0.8745
R	0.7492	0.8055	0.7545
F1	0.7988	0.8551	0.8101

It can be seen that when only one CNN layer was used, the CNN-Transformer model obtained a P value of 0.8554, a R value of 0.7492, and an F1 value of 0.7988 for the MAPS dataset. When two CNN layers were used, this model exhibited the best transcription performance, with an F1 value of 0.8551. When the number of layers continued to increase, the transcription effect declined. These results demonstrated the correctness of using two CNN layers.

Next, the CNN-Transformer model was compared to the deep neural network (DNN) [29] and CNN models, and the frame-level results for MAPS and MAESTRO datasets are presented in Table 3.

Table 3. Frame level results for different transcription models

		DNN	CNN	CNN-Transformer
MAPS	P	0.6534	0.8721	0.9112
	R	0.7469	0.7653	0.8055
	F1	0.6970	0.8152	0.8551
MAESTRO	P	0.6874	0.8456	0.8886
	R	0.7112	0.8895	0.9204
	F1	0.6991	0.8670	0.9042

It was found that the DNN model yielded inferior transcription results for both the MAPS and MAESTRO datasets, with an F1 value of only 0.6970 for MAPS and 0.6991 for MAESTRO. The CNN model showed relatively better results. It obtained an F1 value of 0.8152 for the MAPS dataset and an F1 value of 0.8670 for the MAESTRO dataset, which was increased by 0.1182 and 0.1679 respectively compared to the DNN model. These results demonstrated its more robust feature extraction capability for piano audio processing. Furthermore, the CNN-Transformer model achieved an F1 value of 0.8551 for MAPS, showing a significant improvement of 0.0399 compared to the CNN model. Similarly, for the MAESTRO dataset, the CNN-Transformer model achieved an F1 value of 0.9042, demonstrating an improvement of 0.0372 compared to the CNN model. These results validated the advantage of combining the Transformer with the CNN architecture for piano audio transcription.

6. Conclusions

This paper designed a CNN-Transformer automatic transcription model for recognizing piano notes by processing piano audio and extracting the dual-channel CQT spectrum. Experiments conducted on the MAPS and MAESTRO datasets obtained the following results.

1) The STFT and mono-channel CQT performed poorly; the dual-channel CQT obtained the optimal results in the transcription of frame levels, notes (no cutoff), and complete notes, with F1 values of 0.8551, 0.8521, and 0.6191, respectively.

2) The input window length could affect transcription results. When the input window length was 7, the transcription result was the best: the P, R, and F1 values for the frame-level transcription were 0.9112, 0.8055, and 0.8551, respectively.

3) The CNN layer number in the CNN-Transformer model could affect transcription results. When there were two CNN layers, the transcription performance was the best.

4) The CNN-Transformer model performed superior results in frame-level transcription compared to the CNN and CNN models.

The results demonstrated the advantages of the CNN-Transformer model, which was designed using the dual-channel CQT, in transcribing piano audio and its application usability in practical transcription. However, this study also has some limitations. For example, there is a lack of a more comprehensive piano audio database, and only one type of feature with superior performance was selected for feature selection without considering the comparison of more features. In future work, further exploration will be conducted on feature selection in piano audio transcription, collecting actual piano audio data will be considered to establish a database, and the CNN-Transformer model will be further optimized to achieve better transcription performance.

Acknowledgements

The authors have not disclosed any funding.

Data availability

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Author contributions

Peng Wang and Ning Dai designed research, performed research, analyzed data, and wrote the paper.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] X. Wang, X. Li, and P. Wang, "The integration and inheritance of Hebei section's Grand Canal music culture in higher normal music teaching," *Journal of Cangzhou Normal University*, Vol. 40, No. 2, pp. 1–5, 2024.

[2] A. Holzapfel, E. Benetos, A. Killick, and R. Widdess, "Humanities and engineering perspectives on music transcription," *Digital Scholarship in the Humanities*, Vol. 37, No. 3, pp. 747–764, Aug. 2022, https://doi.org/10.1093/llc/fqab074

[3] J. Liu, W. Xu, X. Wang, and W. Cheng, "An EB-enhanced CNN Model for piano music transcription," in *ICMLC 2021: 2021 13th International Conference on Machine Learning and Computing*, pp. 186–190, Feb. 2021, https://doi.org/10.1145/3457682.3457710

[4] X. Fu, H. Deng, and J. Hu, "Automatic label calibration for singing annotation using fully convolutional neural network," *IEEJ Transactions on Electrical and Electronic Engineering*, Vol. 18, No. 6, pp. 945–952, Apr. 2023, https://doi.org/10.1002/tee.23804

[5] F. Simonetta, S. Ntalampiras, and F. Avanzini, "Audio-to-score alignment using deep automatic music transcription," in *IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6, Oct. 2021, https://doi.org/10.1109/mmsp53017.2021.9733531

[6] Z. Meng and W. Chen, "Automatic music transcription based on convolutional neural network, constant Q transform and MFCC," in *Journal of Physics: Conference Series*, Vol. 1651, No. 1, p. 012192, Nov. 2020, https://doi.org/10.1088/1742-6596/1651/1/012192

[7] M. Alfaro-Contreras, J. J. Valero-Mas, J. M. Iñesta, and J. Calvo-Zaragoza, "Late multimodal fusion for image and audio music transcription," *Expert Systems with Applications*, Vol. 216, p. 119491, Apr. 2023, https://doi.org/10.1016/j.eswa.2022.119491

[8] S. Lee, "Estimating the rank of a nonnegative matrix factorization model for automatic music transcription based on stein's unbiased risk estimator," *Applied Sciences*, Vol. 10, No. 8, p. 2911, Apr. 2020, https://doi.org/10.3390/app10082911

[9] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, Vol. 6, No. 3, pp. 439–449, Jun. 2004, https://doi.org/10.1109/tmm.2004.827507

[10] M. P. Ryynanen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005.*, pp. 319–322, Oct. 2024, https://doi.org/10.1109/aspaa.2005.1540233

[11] E. Benetos and S. Dixon, "Multiple-instrument polyphonic music transcription using a temporally constrained shift-invariant model," *The Journal of the Acoustical Society of America*, Vol. 133, No. 3, pp. 1727–1741, Mar. 2013, https://doi.org/10.1121/1.4790351

[12] Y. Ju, B. Babukaji, and J. Lee, "Automatic music transcription considering time-varying tempo," *The Journal of the Korea Contents Association*, Vol. 12, No. 11, pp. 9–19, Nov. 2012, https://doi.org/10.5392/jkca.2012.12.11.009

[13] K. O. 'Hanlon, H. Nagano, and M. D. Plumbley, "Structured sparsity for automatic music transcription," in *ICASSP 2012 – 2012 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 441–444, Mar. 2012, https://doi.org/10.1109/icassp.2012.6287911

[14] D. Cazau, G. Revillon, J. Krywyk, and O. Adam, "An investigation of prior knowledge in Automatic Music Transcription systems," *The Journal of the Acoustical Society of America*, Vol. 138, No. 4, pp. 2561–2573, Oct. 2015, https://doi.org/10.1121/1.4932584

[15] Y.-S. Wang, T.-Y. Hu, and S.-K. Jeng, "Automatic transcription for music with two timbres from monaural sound source," in *IEEE International Symposium on Multimedia (ISM)*, pp. 314–317, Dec. 2010, https://doi.org/10.1109/ism.2010.54

[16] A. Kilian, J. Karolus, T. Kosch, A. Schmidt, and P. W. Woźniak, "EMPiano: electromyographic pitch control on the piano keyboard," in *CHI '21: CHI Conference on Human Factors in Computing Systems*, pp. 1–4, May 2021, https://doi.org/10.1145/3411763.3451556

[17] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions," *Journal of Intelligent Information Systems*, Vol. 41, No. 3, pp. 407–434, Jul. 2013, https://doi.org/10.1007/s10844-013-0258-3

[18] E. Idrobo-Ávila, H. Loaiza-Correa, F. Muñoz-Bolaños, L. van Noorden, and R. Vargas-Cañas, "Development of a biofeedback system using harmonic musical intervals to control heart rate variability with a generative adversarial network," *Biomedical Signal Processing and Control*, Vol. 71, No. Part A, p. 103095, Jan. 2022, https://doi.org/10.1016/j.bspc.2021.103095

[19] W.-B. Gao and B.-Z. Li, "Octonion short-time Fourier transform for time-frequency representation and its applications," *IEEE Transactions on Signal Processing*, Vol. 69, pp. 6386–6398, Jan. 2021, https://doi.org/10.1109/tsp.2021.3127678

[20] T. Kaneko, K. Tanaka, H. Kameoka, and S. Seki, "ISTFTNET: fast and lightweight Mel-spectrogram vocoder incorporating inverse short-time Fourier transform," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 2022, May 2022, https://doi.org/10.1109/icassp43922.2022.9746713

[21] Y. Huang, H. Hou, Y. Wang, Y. Zhang, and M. Fan, "A long sequence speech perceptual hashing authentication algorithm based on constant q transform and tensor decomposition," *IEEE Access*, Vol. 8, pp. 34140–34152, Jan. 2020, https://doi.org/10.1109/access.2020.2974029

[22] K. E. Tokarev, V. M. Zotov, V. N. Khavronina, and O. V. Rodionova, "Convolutional neural network of deep learning in computer vision and image classification problems," in *IOP Conference Series: Earth and Environmental Science*, Vol. 786, No. 1, p. 012040, Jun. 2021, https://doi.org/10.1088/1755-1315/786/1/012040

[23] Y. Kawara, C. Chu, and Y. Arase, "Preordering encoding on transformer for translation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 29, pp. 644–655, Jan. 2021, https://doi.org/10.1109/taslp.2020.3042001

[24] S. Sridhar and S. Sanagavarapu, "Multi-head self-attention transformer for dogecoin price prediction," in *2021 14th International Conference on Human System Interaction (HSI)*, pp. 1–6, Jul. 2021, https://doi.org/10.1109/hsi52170.2021.9538640

[25] P. A. Babu, V. Siva Nagaraju, and R. R. Vallabhuni, "Speech emotion recognition system with Librosa," in *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 421–424, Jun. 2021, https://doi.org/10.1109/csnt51715.2021.9509714

[26] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 18, No. 6, pp. 1643–1654, Aug. 2010, https://doi.org/10.1109/tasl.2009.2038819

[27] S. Mukherjee and M. Mulimani, "ComposeInStyle: Music composition with and without Style Transfer," *Expert Systems with Applications*, Vol. 191, p. 116195, Apr. 2022, https://doi.org/10.1016/j.eswa.2021.116195

[28] C. Raffel et al., "mir_eval: a transparent implementation of common MIR Metrics," in *15th International Society for Music Information Retrieval Conference (ISMIR 2014)*, 2014.

[29] A. J. Rogers et al., "Abstract 17420: enhanced identification of cardiac wall motion abnormalities: an externally validated deep neural network approach outperforms expert and quantitative analysis of electrocardiograms," *Circulation*, Vol. 148, No. Suppl_1, Nov. 2023, https://doi.org/10.1161/circ.148.suppl_1.17420

Peng Wang graduated from Ukrainian National Conservatory of Music with a doctor's degree in music art. He is a lecturer working in Cangzhou Normal University. His research interests are vocal teaching and disaster prevention and music stage art director. He has published 4 papers.

Ning Dai received the B.S. degree in music education from Central Conservatory of Music, Beijing, China, in 2006, and the M.S. degree in music education from Central Conservatory of Music, Beijing, China, in 2009. Her research interests include music education and musicology.