Jin *et al.*
EURASIP Journal on Audio, Speech, and Music Processing (2022) 2022:31
https://doi.org/10.1186/s13636-022-00261-8

EURASIP Journal on Audio, Speech, and Music Processing

EMPIRICAL RESEARCH

Open Access

MetaMGC: a music generation framework for concerts in metaverse

Cong Jin[1], Fengjuan Wu[1], Jing Wang[2*], Yang Liu[1], Zixuan Guan[1] and Zhe Han[1]

Abstract

In recent years, there has been a national craze for metaverse concerts. However, existing meta-universe concert efforts often focus on immersive visual experiences and lack consideration of the musical and aural experience. But for concerts, it is the beautiful music and the immersive listening experience that deserve the most attention. Therefore, enhancing intelligent and immersive musical experiences is essential for the further development of the metaverse. With this in mind, we propose a metaverse concert generation framework — from intelligent music generation to stereo conversion and sound field design for virtual concert stages. First, combining the ideas of reinforcement learning and value functions, the Transformer-XL music generation network is improved and used in training all the music in the POP909 dataset. Experiments show that both improved algorithms have advantages over the original method in terms of objective evaluation and subjective evaluation metrics. In addition, this paper validates a neural rendering method that can be used to generate spatial audio based on a binaural-integrated neural network with a fully convolutional technique. And the purely data-driven end-to-end model performs to be more reliable compared with traditional spatial audio generation methods such as HRTF. Finally, we propose a metadata-based audio rendering algorithm to simulate real-world acoustic environments.

Keywords: Metaverse concert, Transformer-XL, Audio digital twin, Neural network, Audio rendering

1 Introduction

With the global explosion of metaverse discussions and the huge impact of the New Crown epidemic on the offline music performance industry, the virtual world has emerged as an ideal stage for music concerts and festivals. But the current metaverse concerts focused heavily on embodying the concept of virtual reality and the digital twin [1, 2]. Secondly, it still takes a long production line from the beginning of planning to hold a metaverse concert, which is contrary to the real-time nature of the metaverse.

The metaverse concert also contains technologies such as virtual stages, motion capture, and digital human. But these virtual performance-related fields are already very mature. And, by the nature of the concerts, it is the beautiful music and immersive online listening that are more worthy of our attention.

In 2019, Danowski et al. proposed Connexion [3] which surrounds the audience with an eight-channel sound system that immerses the audience from all directions. More recently, PatchXR allows artists to turn a place into an instant music studio by providing spatial equivalents of a visual programming engine to create and perform music on a spatial level with sound building blocks [4]. The virtual reality concert of the Philharmonic Orchestra conducted by Esa-Pekka allowed the audience to move between each instrument group and even freely through the church [5], enabling an interactive experience.

However, most of the existing research works on metaverse-based sound field design for music composition have either considered only real-time music performance online or immersive experience online, without addressing both real-time and intelligence in metaverse

*Correspondence: wangjing@bit.edu.cn

[2] School of Information and Electronicsy, Beijing Institute of Technology, Zhongguancun Nanda Street, 100081 Beijing, China
Full list of author information is available at the end of the article

concerts. Therefore, we propose a framework to efficiently and intelligently implement music generation and immersive sound field twinning for concerts in the metaverse, namely MetaMGC (Music Generation Framework for Concerts in Metaverse). It consists of three main parts: (1) a music generation part that enables improvised accompaniment of a virtual orchestra for metaverse concerts; (2) a digital audio twin part that enables virtual sound field reconstruction for metaverse concerts [6]; and (3) an audio rendering part that realizes the virtual soundstage production of the metaverse concert.

For the three elements above, we investigate the Transformer-XL music generation model based on two methods: Monte Carlo search as well as deep reinforcement learning. Meanwhile, we propose a reward function to control the music generation-related rewards, which strengthens the constraints on the music theory knowledge in the music generation network. In addition, this paper investigates an end-to-end neural network synthesis method that simulates the differences caused by subtle effects on the final output signal through a temporal convolutional neural network module. We conducted extensive experiments on the POP909 dataset [7] and HRFT data [8] to evaluate the effectiveness and generality of the proposed method.

To summarize, the main contributions are as follows:

- Optimization of Transformer-XL by Monte Carlo and DQN methods based on reinforcement learning.
- A value function-based music generation system to intelligently generate accompaniment.
- An end-to-end neural synthesis method capable of synthesizing natural and accurate binaural audio.
- A metadata-based audio rendering system and sound field reconstruction in UE4 [9].The audio rendering system is tested to be effective.

2 Related work

2.1 Automatic music generation

Eck et al. first used LSTM for music production, improvising well-paced and structured blues music based on short recordings. In 2012, Boulanger et al. [10] proposed an RNN-RBM model that outperformed traditional models in generating polyphonic music from different datasets. In 2016, Google Brain's Magenta team further improved the RNN's ability to learn long-term structure by proposing the MelodyRNN model [11]. Hadjeres et al. proposed Anticipation-RNN that allowed user-defined positional constraints to be enforced [12]. Johnson et al. proposed TP-LSTM-NADE and BALSTM incorporating a parallel set of weighted recurrent networks for polyphonic music prediction and composition [13].

With the development of deep learning techniques, powerful deep generative models such as VAE, GAN, and Transformer have gradually emerged. In 2015, Samuel et al. first proposed the Variational Autoencoder (VAE) [14]. Roberts et al. proposed MusicVAE [15], a hierarchical VAE model that captures the long-term structure of polyphonic music with good interpolation and reconstruction performance. Jia et al. [16] proposed a coupled latent variable model with a binary regularizer to implement improvised accompaniment generation. Yang et al. proposed a MidiNet [17] network based on GAN networks that can generate music bar after bar and proposed a new conditional mechanism to generate chord-based music. Yu et al. proposed a sequence generation framework, SeqGAN [18], which successfully applied RNN-based GAN networks to the music generation process for the first time by combining reinforcement learning techniques. In 2018, Dong et al. proposed the MuseGAN model [19], which is considered to be the first model to generate multi-track polyphonic music.

More recently, Anna Huang et al. successfully applied the Transformer technique for the first time [20]. Donahuel et al. generated multiple instrument music using Transformer and proposed a pre-training technique based on migration learning [21]. Moreover, Huang et al. proposed a new music representation called REMI [22] and used the Transformer-XL sequence model [23] to generate popular piano music. The emergence of Transformer-XL further optimized the original Transformer model.

2.2 Binaural audio generation

The majority of existing binaural audio generation techniques are based primarily on conventional digital signal processing (DSP). Head-related transfer functions are measured in a radio wave darkroom [24], while high-quality 2-spatialization requires binaural recordings at different spatial locations over nearly 10k [25]. To generate binaural audio, DSP-based renderers typically perform a series of convolutions on the fractional impulse responses [26].

More recently, neural network techniques have gained attention in audio generation as a result of the success of neural networks in speech synthesis [27]. Current approaches to neural networks focus primarily on frequency domain models [28, 29], but the greater difficulty in modeling the long-term dependence of high-frequency audio signals has led to the long-term neglect of the original waveform model. With the success of WaveNet proposed by Van Den Oord et al. [30], direct wave-to-wave modeling has received tremendous attention and led to significant improvements in

speech enhancement [31], denoising [32], speech synthesis [33] and music style translation [34].

The process of spatializing neural networks is now underway. A study by Gebru et al. [35] showed that HRTF enables implicit learning of neural networks by training raw waveforms. Morgado et al. [36] worked on predicting spatial sounds conditioned on visual information, but their work was limited to first-order binaural channels and did not exhaustively model binaural effects. In closer comparison, a series of papers by Gao and Grauman [37] targeting 2.5D visual sound systems, in which binaural audio is generated conditionally using video frame embedding. Thus, the location of the sound source can be effectively determined.

3 Approach

Our MetaMGC consists of three main systems(as is shown in Fig. 1): a value function-based music generation system, an end-to-end neural synthesis-based system, and an audio rendering system. First, we trained the Transformer-XL music generation network by inputting the music MIDI dataset. The value function in reinforcement learning (Monte Carlo or DQN) is combined to improve the music generation network and generate the mono MIDI events of the generated music. To create a spatial immersion experience, we converted the input mono audio into spatial stereo audio using a neural network-based spatial audio twin system. Finally, we input the generated stereo audio into an audio rendering system and then presented it in a virtual reality stage by building a digital twin sound field. The result turned out to be the creation of a meta-universe concert from intelligent music generation rendered with immersive binaural audio.

3.1 Music generation based on value functions

In this subsection, our music generation network based on the Transformer-XL network is presented, which transforms music theory rules into multiple reward functions to control the music generation process. This approach is to solve the optimal value function in reinforcement learning and makes the generated music more musical.

3.1.1 Music theory reward mechanism

To enable the network model to learn music theory, we quantify music theory knowledge in the form of textual descriptions and present it in the form of a reward function to control music generation during the reinforcement learning process. The music theory reward mechanism is set up in two parts: a basic music theory reward (see Table 1) ,where $R^{m1}(s_{t:1}, a_t)$ is to determine whether the 4 generated pitches are the same;$R^{m2}(s_{t:1}, a_t)$ is to determine that the intervals of two adjacent notes in a piece of music should be no greater than an octave; and $R^{m3}(s_{t:1}, a_t)$ is a predetermined range for the sound range. and a melody writing reward (see Table 2). In this mechanism, R denotes the total reward for the current time step t, and a_{max} and a_{min} are the highest and lowest notes set according to demand.

3.1.2 Transformer-XL with the Monte Carlo method

The Monte Carlo method uses time-step limited, complete empirical trajectories and the resulting empirical information to derive the average reward for each state. In the case of an unknown environment, the intelligence samples are according to the strategy π. From the starting state, it executes this strategy T steps before reaching the termination state, thus obtaining an empirical trajectory and then calculating the cumulative future discounted rewards [38, 39].

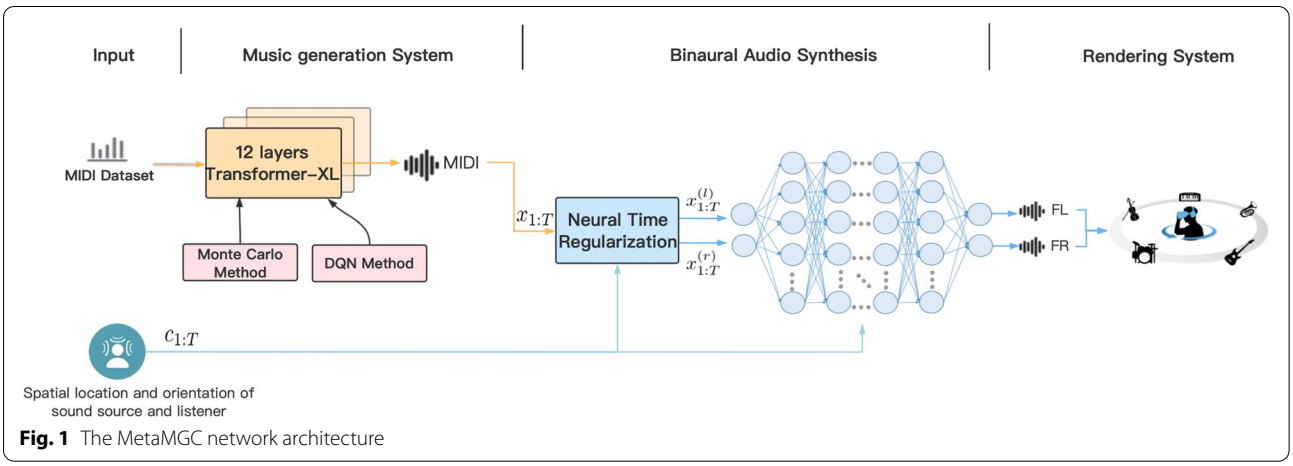

Fig. 1 The MetaMGC network architecture

Table 1 Basic music theory reward

	Condition	Meaning	Reward
$R^{m1}(s_{t:1}, a_t)$	$a_{t-1} = a_t, (if\ a_{t-3} = a_{t-2} = a_{t-1})$	Pitch recurrence 4 times	0.9
	$a_{t-1} \neq a_t, (if\ a_t \neq a_t)$	Pitch recurrence 3 times	0
$R^{m2}(s_{t:1}, a_t)$	$\mid a_t - a_{t-1} \mid \leq 12$	Intervals greater than an octave	0.1
	$\mid a_t - a_{t-1} \mid > 12$	Intervals less than an octave	− 0.8
$R^{m3}(s_{t:1}, a_t)$	$a_t \in [a_{min}, a_{max}]$	Within the set vocal range	0.1
	$a_t \notin [a_{min}, a_{max}]$	Not in the set vocal range	− 0.8

Table 2 Melodic writing incentives: (a) melodic interval reward; and (b) melodic tone towards reward

(a)		(b)		
$\mid a_t - a_{t-1} \mid$	$R^{w1}(s_{t:1}, a_t)$	**Condition**	**Meaning**	$R^{w2}(s_{t:1}, a_t)$
0	0.3	$0 < a_t - a_{t-1} < 4$	Step-in ascending	0.5
1	0	$-4 < a_t - a_{t-1} < 0$	Step-in descending	0.5
2	0.2	$a_t - a_{t-2} > 4, -4 < a_t - a_{t-1} < 0$	Ascend in plunge, than descend in step	0.4
3	0.2	$a_t - a_{t-2} < 4, 0 < a_t - a_{t-1} < 4$	Descend in plunge, than ascend in step	0.4
4	0.3	$a_t - a_{t-2} > 4, 0 < a_t - a_{t-1} < 4$	Ascend in plunge, than ascend in step	0.1
5	0.4	$a_t - a_{t-2} < 4, 04 < a_t - a_{t-1} < 0$	Descend in plunge, than descend in step	0.1
6	− 0.8	$a_t - a_{t-2} > 4, a_t - a_{t-1} > 4$	Ascend in plunge, than ascend in plunge	− 0.5
7	0.4	$a_t - a_{t-2} < 4, a_t - a_{t-1} < -4$	Descend in plunge, than descend in plunge	− 0.5
8	0.2	-	-	-
9	0.2	-	-	-
10	− 0.5	-	-	-
11	− 0.7	-	-	-
12	0.1	-	-	-

The Monte Carlo method utilizes the average future discounted cumulative reward G of the experience trajectory as the expectation of the state value:

$$G = average(G_1 + G_2 + ... + G_T). \quad (1)$$

If a large enough sample of empirical trajectories is processed, it is possible to accurately estimate the expectation of following the policy in state s, known as the state value function $v_\pi(s)$:

$$v_\pi(s) = E_\pi[G \mid s] \in S. \quad (2)$$

Since the P_{pitch}(probability of each note) at the time of generation is different, a greater probability of the selected note itself being in the probability distribution indicates that the note has a higher value. Thus, in combination with the music theory reward function, the pitch reward value G for each time step can be defined as:

$$G_t = \sum_{i=1}^{3} R^{mi}(s_{t:1}, a_t) + \sum_{i-1}^{2} R^{wi}(s_{t:1}, a_t) + P_{pitch}. \quad (3)$$

This way the optimal pitch trajectory is found by obtaining the reward value G_t returned by the state at a certain moment.

3.1.3 Transformer-XL with DQN method

The DQN model consists of four parts (Fig. 2): (1) data preprocessing, which sends the processed data to the generative network for training; (2) outputting, which outputs the probability distribution of all event indexes through the Softmax layer after training the Transformer-XL network (12 layers, each containing 8 Multi-head Attention layers and 12 Feedforward layers) and selecting the event index with the highest probability as the currently generated value; (3) the reward network, which includes the probability of the current generated number and the sum of multiple lemma rewards; (4) the DQN, which extracts all the indexes representing the pitch events from the generated sequence, combines the reward network to calculate the reward to train the Q network, controls the generated note sequence, and finally decodes the sequence to generate music.

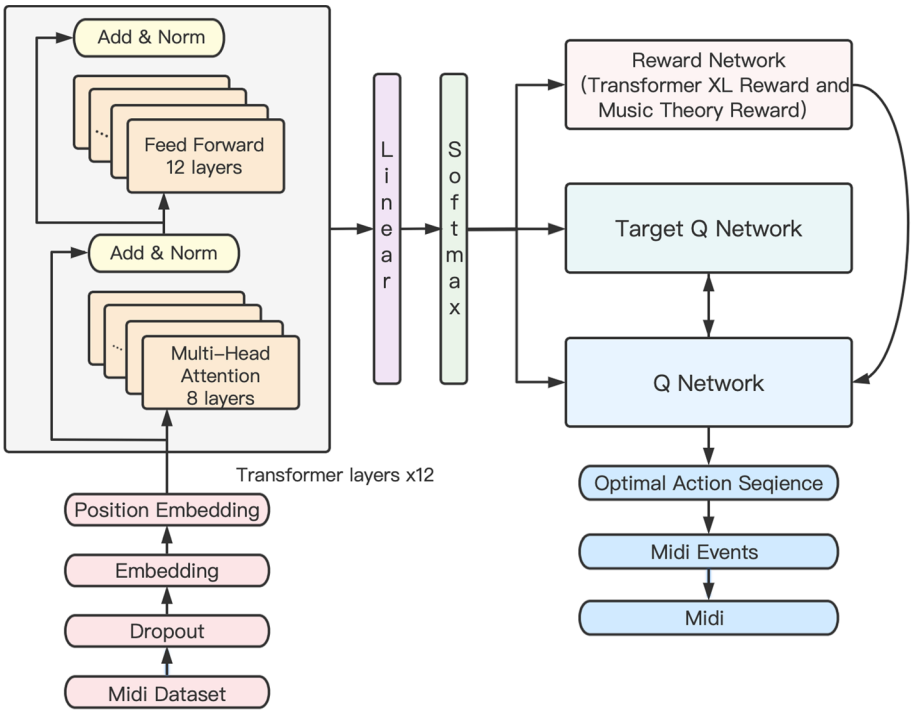

Fig. 2 Transformer-XL+DQN model

The DQN method is a basic algorithm in deep reinforcement learning that makes the action value function $Q(s,a,\theta)$ converge to the optimal action value function $Q^*(s,a)$ by training the parameters [40] .

Combine the probability of note generation to obtain the current total bonus value:

$$R^{G_{\theta_A}}(s_{1:t-1},a_t)=R^m(s_{1:t-1},a_t)+R^w(s_{1:t-1},a_t)+P_{pitch}, \tag{4}$$

where

$$\begin{aligned} R^m(s_{1:t-1},a_t)&=\sum_{i=1}^{3}R^{mi}(s_{1:t-1},a_t)\\ \text{and } R^w(s_{1:t-1},a_t)&=\sum_{i=1}^{3}R^{wi}(s_{1:t-1},a_t) \end{aligned} \tag{5}$$

are the basic music theory rules bonus and the writing rules bonus, respectively.

Calculate the gradient derivation of the objective function with respect to the parameters of the Q-network and conversion to an unbiased estimate using:

$$\nabla L(\theta_A)\cong \sum_{i=1}^{T}E_{a_t\sim G_{\theta_A}(a_t|s_{1:t-1})}\left[\sum_{a_i\in A}\nabla_{\theta_A}\log G_{\theta_A}(a_t\mid s_{1:t-1})*R^{G_{\theta_A}}(s_{1:t-1},a_t)\right]. \tag{6}$$

Then update the Q network parameters:

$$\theta_A+a\nabla L(\theta_A)\rightarrow\theta_A. \tag{7}$$

Finally, the model is updated by training the Q-network so that the Transformer-XL music generation network learns the rules of basic music theory and melody writing rules for note constraints to obtain the optimal strategy.

3.2 End-to-end binaural audio synthesis

Given the source and listener positions and orientations $c_{1:T}$ for each time step, the single-channel input signal $x_{1:T}$ is converted into a binaural signal. The final system is shown in Fig. 3.

We convert a single channel signal of length T: $x_{1:T}=(x_1,...,x_T)$ into binaural (stereo) signals $y^{(l)}_{1:T}=(y^{(l)}_1,...,y^{(l)}_T)$ and $y^{(r)}_{1:T}=(y^{(r)}_1,...,y^{(r)}_T)$, with the former representing the left-ear signal and the latter representing the right-ear signal. x_t, $y^{(l)}_t$, and $y^{(r)}_t$ all represent the sample scalar of the audio at moment t. The conditional time signal $c_{1:T}$ represents the position and direction of the source and the listener. Our goal is to obtain the following function:

$$\left(y^{(l)}_t,y^{(r)}_t\right)=f(x_{t-\Delta:t}\mid c_{t-\Delta:t}), \tag{8}$$

where Δ is the receptive domain in the time domain $c_t\in R^14$ containing the three-dimensional positions of

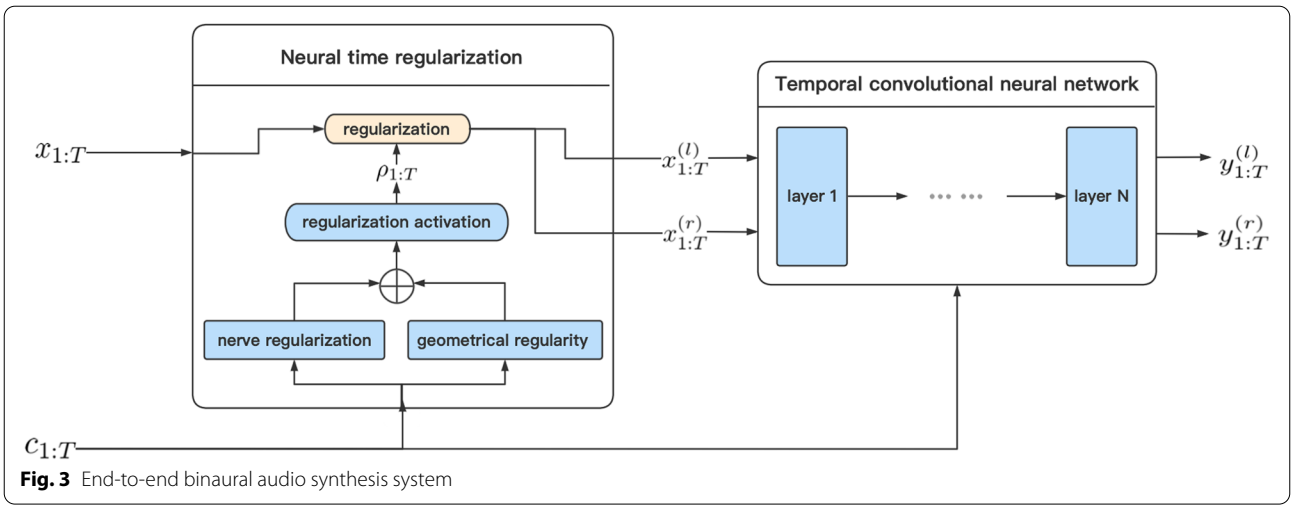

Fig. 3 End-to-end binaural audio synthesis system

the source and the listener (three values each) and the quaternions representing the directions (four values each).

3.2.1 Neural time regularization

We estimate a neural distortion field $\rho^{(neural)}_{1:T} = WarpNet(c_{1:T})$ and add it to the geometric regularization above (see Fig. 3):

$$\rho_t = \sigma^{(warp)}(\rho_{t-1}, \hat{\rho}_t),\ \hat{\rho}_t = \rho^{(goem)}_t + \rho^{(neural)}_t, \quad (9)$$

where $\sigma^{(warp)}(\rho_{t-1}, \hat{\rho}_t) = max(\rho_{t-1}, min(t, \hat{\rho}_t))$ is the recursive activation function that ensures monotonicity and causality.

WarpNet is a shallow temporal convolutional network with four layers and 64 channels per layer. We define the distorted signal $\hat{x_{1:T}}$ as a linear interpolation of the source signal $x_{1:T}$ at $\lfloor \rho_t \rfloor$ and $\lceil \rho_t \rceil$:

$$\hat{x_t} = (\lceil \rho_t \rceil - \rho_t) x_{\lfloor \rho_t \rfloor} + (\rho_t - \lfloor \rho_t \rfloor) x_{\lceil \rho_t \rceil} \quad (10)$$

In practice two warp fields are generated, one for each of the two ears. We enforce the physical constraint using $\sigma^{(warp)}$. $min(t, \hat{\rho}_t)$ forces the tth element of the twisted field to be no larger than t itself to ensure causality while $max(\rho_{t-1}, \cdot)$ implies that an element was twisted from ρ_{t-1} to $(t-1)$, then the next element at position t must be twisted from ρ_{t-1} or the subsequent position, thus ensuring monotonicity. Therefore, compared with related methods such as deformable convolution and spatial transformer networks, the neural time regularization in this paper performs constrained regularization for input signals of arbitrary length as well as directly simulates the physical phenomena of sound.

3.2.2 Conditional superconvolution

Inspired by the DSP formulation, we predict the convolutional weights and biases of the input $x_{1:T}$ of a given layer as a function of the conditional input $c_{1:T}$. Weights are generated from the conditional input $c_{1:T}$ containing physical information about the relationship between the sound source and the listener:

$$z_t = \sum_{k=1}^{K} [H^{(W)}(c_{1:t})]_{:,:,k} x_{t-k+1} + H^{(b)}(c_{1:t}), \quad (11)$$

where $H^{(W)}$ and $H^{(b)}$ are small convolutional hypernetworks that receive $c_{1:t}$ as input and predict convolutional weights and biases as output, respectively. Thus, the input of the convolutional layer is not just a time series, and its weights and biases change over time.

3.2.3 Phase reconstruction using L2-loss

Using the L2-loss of the original waveform to train a generative audio model can lead to poor sound quality and signal distortion. Therefore, a fundamental problem of phase estimation of L2-loss on waveforms is analytically explained. A simple additional loss term can mitigate this problem.Define:

$$L_2(y_{1:T}, \hat{y}_{1:T}) = \sum_t (y_t, \hat{y}_t)^2 \quad (12)$$

as the time domain L2-loss between the predicted audio signal $y_{1:T}$ and the target signal $\hat{y}_1$: T. $Y_k, \hat{Y}_k \in C$ denotes the kth frequency component of $y_{1:T}$, and $\hat{y}_1$: T is in the Fourier frequency domain. The amplitude error and angular phase error of the kth frequency component are denoted as

$$L^{(amp)}(Y_k, \hat{Y}_k) = || Y_k | - | \hat{Y}_k ||$$
$$L^{(phase)}(Y_k, \hat{Y}_k) = \angle(Y_k, \hat{Y}_k). \tag{13}$$

where $|\cdot|$ is the modulo operation of the complex numbers.

According to the Parseval theorem, we write the L2-loss in the time domain as the L2-loss in the complex frequency domain as follows:

$$L_2(y_{1:T}, \hat{y}_{1:T}) = \sum_t | Y_k - \hat{Y}_k |^2 . \tag{14}$$

Now, the distance $| Y_k - \hat{Y}_k |$ is denoted as ε.

Theorem 1 *Define $\hat{Y} \in C$ as a specified complex number and $Y \in B_{\varepsilon,\hat{Y}} = \{Y \in C : | Y - \hat{Y} | = \varepsilon\}$ as any complex number with distance ε from $\hat{Y}$. The expected amplitude error and the expected angular phase error with respect to $\hat{Y}$ are*:

$$E_Y(L^{(amp)}(Y, \hat{Y})) = \frac{1}{2\pi} | \hat{Y} | \int_{-\pi}^{\pi} || \frac{\varepsilon}{| \hat{Y} |} + e^{j\varphi} | - 1 | d\varphi$$
and
$$E_Y(L^{(phase)}(Y, \hat{Y})) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \arccos \frac{Re(\frac{\varepsilon}{|\hat{Y}|} e^{j\varphi} + 1)}{| \frac{\varepsilon}{|\hat{Y}|} + e^{j\varphi} |} d\varphi. \tag{15}$$

According to Theorem 1, we can analyze the expected amplitude error and phase error along the kth frequency component. First, in the early stage of training, the expected amplitude error is low for higher energy signals, even with large L2-values. On the contrary, the phase is hardly optimized when the L2-loss is large. Second, the expected amplitude error in all target energies decreases during the training process when the L2-loss decreases with time. In contrast, the improvement of the expected phase error is primarily for the high-energy components, while the phase accuracy is poor for the medium and low-energy components. Therefore, optimizing the original waveform using the L2-loss in the time domain is not sufficient to achieve accurate phase reconstruction.

Due to the limited capacity of the model, the training data can usually only be fitted to L2-loss εmin. If this ε min is too large, the amplitude of the signal is modeled well but has a large phase error. To overcome the shortcomings of time-domain L2-loss in phase optimization, we add an explicit phase term to the loss function:

$$L(y_{1:T}, \hat{y}_{1:T}) = L_2(y_{1:T}, \hat{y}_{1:T}) + \lambda L^{(phase)}(STFT(y_{1:T}), STFT(\hat{y}_{1:T})), \tag{16}$$

where $STFT(y_{1:T})$ is the short-time Fourier transform of the audio signal $y_{1:T}$.

3.3 Audio rendering system

The sound quality produced at a concert can determine the success or failure of a real concert, whether it is produced directly by the instrument or by the amplifier. To improve this aspect in a virtual concert, we divide the metadata-based audio rendering into two separate parts that are independent of each other: environment sound rendering and location-dependent sound (object) rendering. This is only an abstract logical distinction. In the actual rendering algorithm, ambient sound rendering is usually convolved with a post-reverberation tail, and location-dependent sound rendering handles direct sound and early reflections. The difference between these two components is whether their operation depends on the location of the sound source and the listener. For ambient sound rendering (i.e., the part that does not depend on the source location), our method sums the source signal and performs a single rendering. An overview of the algorithm is shown in the following Fig. 4.

Classify the elements of the audio to correspond to the metadata. Each audioFormatExtended in the metadata is regarded as a scene, the scene includes sub-scenes with an audio programme to correspond, and the audioContent in the scene is used as the scene audio library SAL. SAL is classified as ancient and modern scenes, industrial scenes,nature-based scenes, and urban scenes.

The distinction described above is not used in some rendering algorithms. One method follows the rendering method based on the image source approach, where the reverberation tail is modeled with higher-order reflections with no ambient sound rendering and only position-dependent rendering (i.e., each part of the rendering algorithm depends on the source and listener positions). Another method employs an air volume simulation-based approach that requires only the source location and the listener location to inject the signal into another system where the source location and the listener location are unknown, retrieving information from them. With our distinction, the air volume modeling-based approach is similar to the algorithm that performs only the ambient sound rendering, which suggests that adopting a co-processing mechanism is most suitable between different sound sources.

4 Experiments

To evaluate our approach, we performed subjective and objective experiments for each of the two major models of metaverse concerts [41, 42]. In Section 4.1, we present our experiments and results for the effectiveness of our Transformer-XL music generation network controlled with Monte Carlo and DQN methods using the POP909

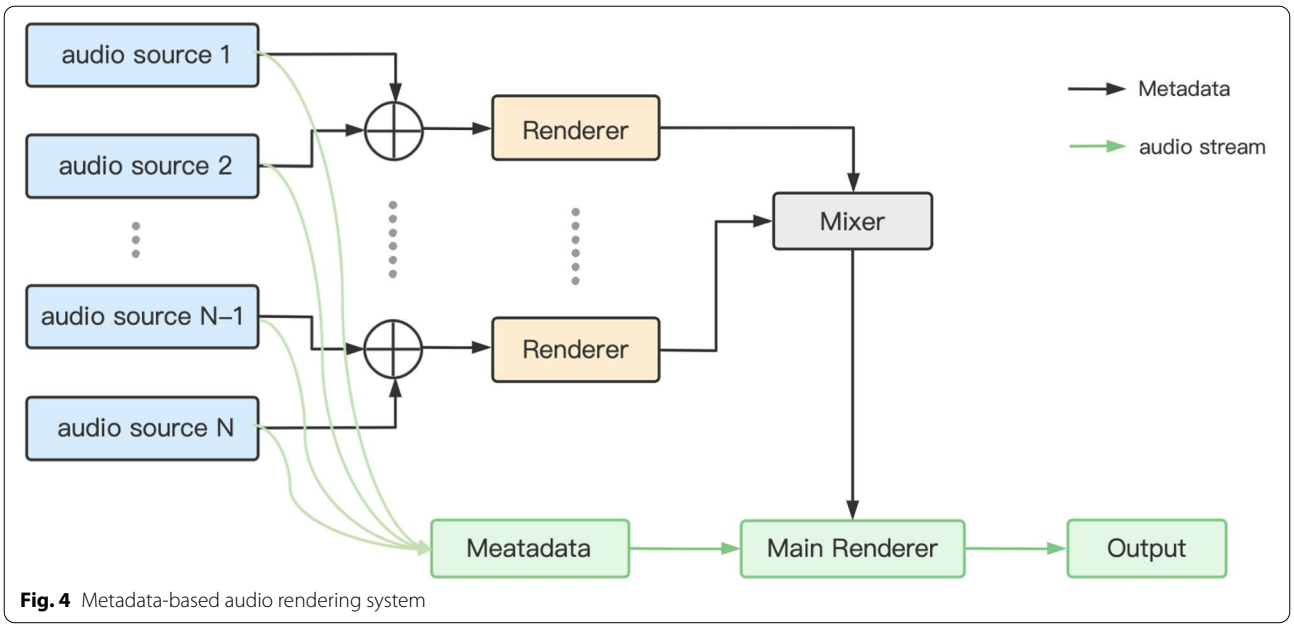

Fig. 4 Metadata-based audio rendering system

dataset [7]. The dataset was assembled by the Music X lab team at New York University in Shanghai in 2020 and contains a total of 909 popular music tracks from 462 artists with a total duration of about 60 h. The experimentally generated music was compared with the music generated by the original Transformer-XL network and two basic algorithms (Melody_LSTM and RL-tuner) to obtain objective and subjective scores.

In Section 4.2, we present experiments using the HRTF dataset as a control with a neural network-based synthesis dataset collected to verify the reliability of the pure data-driven end-to-end model, using the HRTF data of KEMAR No. 21 (Subject_003) from the CIPIC HRTF database. This CIPIC HRTF database includes high spatial resolution HRTF measurements for 45 different subjects, including KEMAR human models with small and large plumes. The HRTF data for each of these subjects included 2500 head-related impulse response measurements. These "standard" measurements were recorded at 25 different intermembrane polarity azimuths and 50 different intermembrane polarity elevations. Section 4.3 presents experiments using a sound field designed in UE4 combining digital twin and virtual reality technologies to achieve a virtual concert sound field simulation.

4.1 Music generation experiments

4.1.1 Training environment

The Transformer-XL generative network has been trained over 120 rounds.The pitches generated by the generative network were controlled by Monte Carlo and DQN methods. The Q-network was a 3-layer 256-cell LSTM network with dropout = 0.5, using an Adam optimizer and a cross-entropy loss function due to the softmax output, which was trained over 120 rounds.

In the experiment, the Transformer-XL+MC method sampled 50 pitch event trajectories and generated 100 music tracks over 100 cycles. The average reward value was 0.142. The Transformer-XL+DQN method sampled 100 pitch event trajectories, with a final loss value of 0.157 and an average reward value of 0.185.

4.1.2 Objective evaluation

To demonstrate the effectiveness of the Transformer-XL+MC method and the Transformer-XL+DQN method on Transformer-XL, the generated results of the three methods were compared with the original dataset and subsequently with the results generated by Melody_LSTM and RL-tuner methods. A total of 3 sets of 100 16-bar music clips were generated using each of the three methods to calculate seven metrics for objective evaluation against the music in the POP909 dataset [43]. The results are given in Table 3. In addition, we randomly select a part of the generated result as an example and visualization, as shown in Fig. 5, our generated music consists of three instrument tracks.

The first four items reflect the complexity of the generated music. The range of the music generated by all three methods was controlled at about 3 octaves, a relatively stable range. The mean values of the number of different pitches were similar for the three methods but slightly higher for the Transformer-XL+DQN

Table 3 Objective evaluation comparison of three methods

Metrics	Transformer XL	Transformer-XL+MC	Transformer-XL+DQN	POP909 Dataset	Melody_LSTM	RL-tuner
Sound range	39.25	37.12	39.09	53.04	37.46	53.04
Average of the number of different pitches	19.6	20.62	24.1	36.74	18.6	36.74
Average number of pitches played simultaneously	2.81	2.85	2.79	3.77	2.15	3.77
Retuning time step ratio	71.2%	72.7%	74.2%	74.9%	64.8%	74.9%
Ratio of tones in the tonality	49.3%	58.5%	61.05%	61.3%	40.5%	61.3%
Information entropy of pitch	2.73	2.68	2.63	2.82	2.91	2.82
Rhythmic consistency with the first two bars	94.3%	97.1%	97.3%	98.8%	81.3%	98.8%

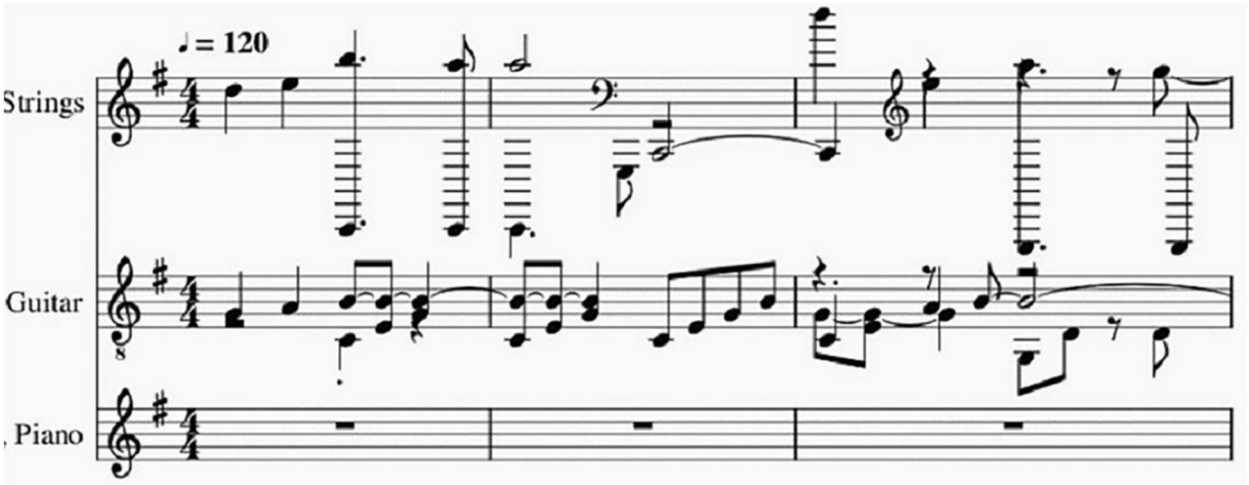

Fig. 5 Part of the music score of the generated result

method. The mean number of pitches played and the polyphony time step ratio hardly differed.

The latter three indicators reflect the goodness of the generated results. For the ratio of tones in tonality, both improved methods were significantly higher than the original method, Melody_LSTM and RL-tuner method. For pitch information entropy, both improved algorithms had lower values than the original method. For the first two bars of the rhythmic consistency, the two improved methods were basically equal and were improved for the original method as well as being close to the original data set.

The Transformer-XL+MC and Transformer-XL+DQN methods generated music with better tonality, richer melodies, and more stable rhythms, which is a good improvement of the Transformer-XL generation network as well as Melody_LSTM method and RL-tuner method in objective terms.

4.1.3 Subjective evaluation

Beautiful music possesses diversity, innovation, and flexibility while taking into account theoretical support and artistic aesthetics [44]. In this experiment, a popular subjective assessment experiment was conducted on music generated using the three methods examined so far. The subjects were divided into two groups: a non-professional group of 25 people consisting of music lovers who were not music students, and a professional group of 5 people consisting of testers with advanced musical education. Five pieces of music generated by the three methods were scored, with the professional group scoring the music-related knowledge and the non-professional group scoring the human ear. The test results are shown in Table 4.

The results in the table suggest that the Transformer-XL+MC model and the Transformer-XL+DQN model outperformed the original model and two basic algorithms in the subjective evaluation. In particular, the

Table 4 Subjective evaluation results of human ear assessment in the (a) non-professional groups and (b) professional groups

Metrics	Transformer XL	Transformer-XL+MC	Transformer-XL+DQN	Melody_LSTM	RL tuner
(a)					
Harmony level	3.848	3.856	4.152	3.643	3.689
Melodic and smooth	3.832	4.016	4.208	3.256	3.314
Pleasing to the ear	3.72	3.784	4.04	3.105	3.287
Structural integrity	3.696	3.952	4.088	2.94	3.051
Define the style	3.816	3.926	4.2	3.023	3.125
Overall Score	4.016	3.912	4.136	3.193	3.293
(b)					
Overall Evaluation	3.84	4.04	4.32	3.13	3.26
Harmonic direction	3.72	3.8	4.24	2.88	2.94
Melodic direction	3.64	3.88	4.12	3.25	3.34
Rhythm Type	3.44	3.64	4.16	3.17	3.21
Tone solidity	4.12	4.12	4.16	2.94	3.34
Overall Score	3.752	3.896	4.2	3.074	3.218

music generated by the Transformer-XL+DQN model achieved the highest scores in all five metrics for both the non-professional and professional groups.

4.2 Spatial audio generation experiments

4.2.1 Data setup

A total of 2 h of paired mono and bi-mono data at 48 kHz were recorded from eight different speakers (4 male and 4 female). The listener was a mannequin with binaural microphones in the ears. Participants were asked to walk around the mannequin within a radius of 1.5 meters and to engage in an unscripted conversation with it. The location and orientation of the sound source and listener were tracked throughout the recording using an OptiTrack system. Using a validation sequence and the last 2 min of each participant as test data, and the remainder as training data, our model was trained 100 times on the Adam Optimizer.

4.2.2 Objective evaluation

According to their audio waveforms, audio spectrum, and audio sound spectrogram,the mono audio, HRTF-based and neural network-generated audio were analyzed separately .

For monaural audio waveforms, the intensities of the audio in the left and right ears were the same at each time point so that the brain did not have a stereo sensation as when one human ear received audio with the same energy received by both ears. In contrast, the audio waveforms generated by HRTF and the audio waveforms generated by neural networks had different intensities for the left and right channels at the same point in time. Therefore, when the audio was received by both ears, the energy received by the left and right ears was different, and the brain produced a stereo sensation for the signal with different energy (Figs. 6 and 7).

The sound spectrum and the corresponding frequency spectrum (Fig. 8) show that the energy of the mono audio is concentrated between 0 and 2 kHz, with lower energy in the middle and high-frequency parts. The energy of the audio generated using HRTF was concentrated between 0 and 2 kHz, with lower energy in the high-frequency part and enhanced energy in the middle frequency part (4 kHz to 5 kHz) as compared with mono audio. The energy in the low-frequency part decreased compared to the mono audio energy. The energy of the neural network-based generated audio was mainly concentrated between 0 and 2 kHz, lower in the high-frequency part, and enhanced in the middle frequency part (4 kHz to 5 kHz) as compared to the mono audio. The overall energy of each frequency band was slightly increased; the energy of the low-frequency bands was decreased compared with the mono audio, and the energy of the low-frequency bands was decreased more than that of the HRTF-based audio.

4.2.3 Subjective evaluation

We used 27 testers with stereo music experience in this experiment to obtain the statistical results shown in Table 5 and visualized in Fig. 9.

The generated music showed overall higher scoring results and overall better music generation. The binaural_neural generated by the neural network was significantly more comfortable than the binaural_hrtf_003 generated by the traditional method of the head-related transfer function HRTF, with significant improvement in the four measures of fullness, intimacy, roundness, and

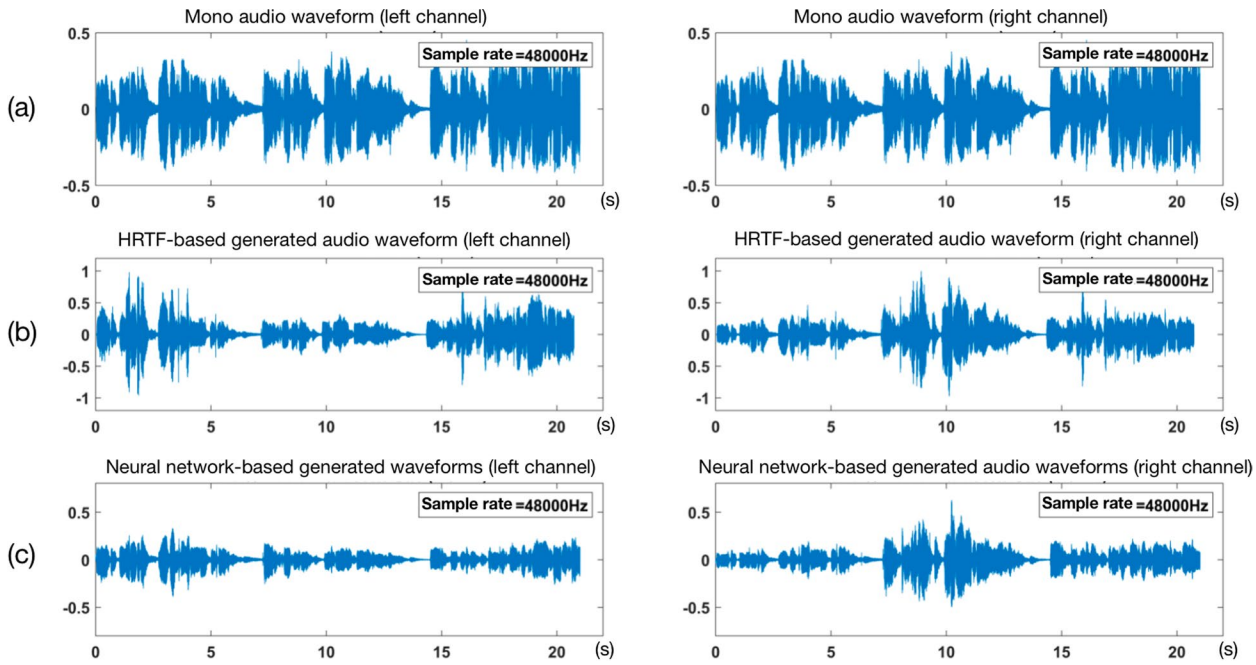

Fig. 6 Sound waveform of **a** mono audio, **b** HRTF-based audio, and **c** neural network-generated audio

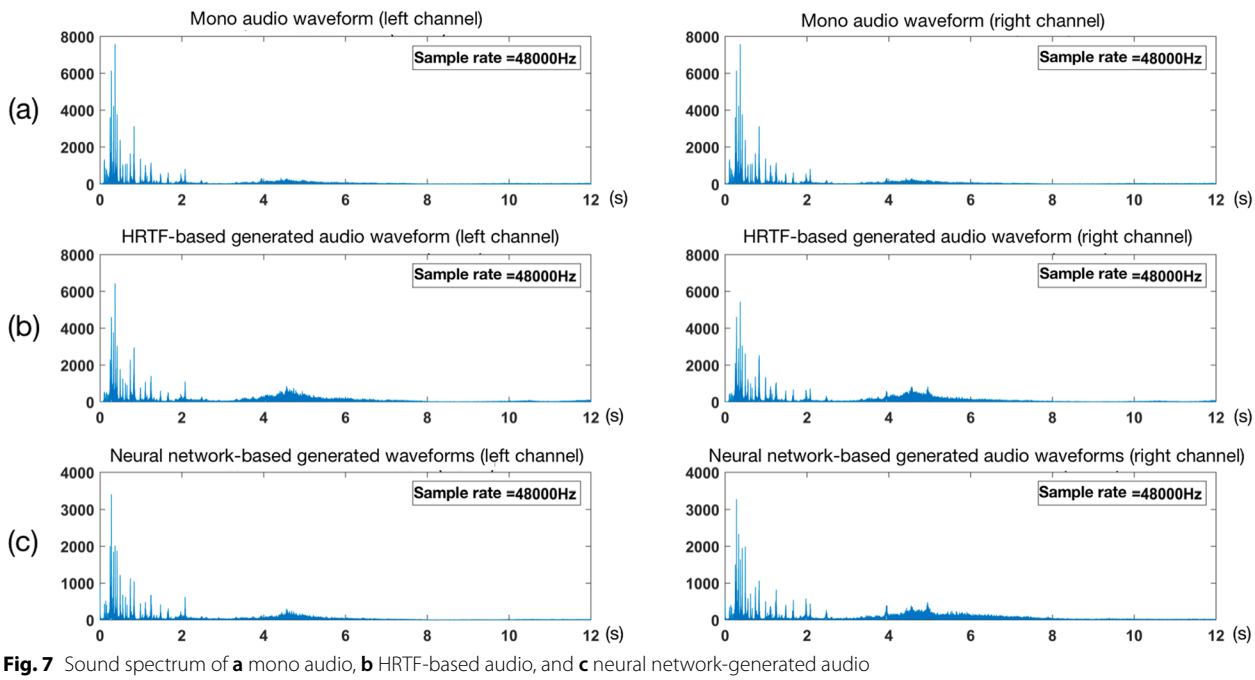

Fig. 7 Sound spectrum of **a** mono audio, **b** HRTF-based audio, and **c** neural network-generated audio

brightness. However, there was no significant improvement in the performance of clarity using the neural network method, which failed to retain more information about the original song; there were still some phase and energy errors that needed to be improved. Combining the other four dimensions, the audio clips generated by the neural network were considered to have high clarity, good naturalness, wide range, low distortion in the pathway, low noise, good transient response, and sufficient reverberation.

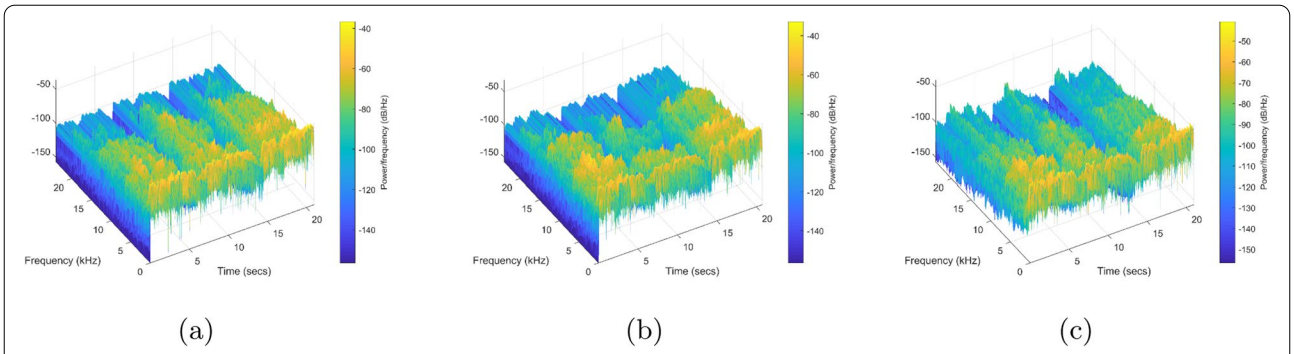

Fig. 8 Sound spectrogram of **a** mono audio, **b** HRTF-based audio, and **c** neural network-generated audio

Table 5 Subjective evaluation score

	Brightness	Comfort	Plumpness	Intimacy	Roundness	Clarity	Balance	Stereo effect	Total score
binaural_neural	7.6	7.8	7.2	7.4	7.2	7.6	7.1	7.3	**58.2**
binaural_hrtf_003	7.2	7.0	6.7	7.0	6.9	7.6	6.8	7.1	56.1

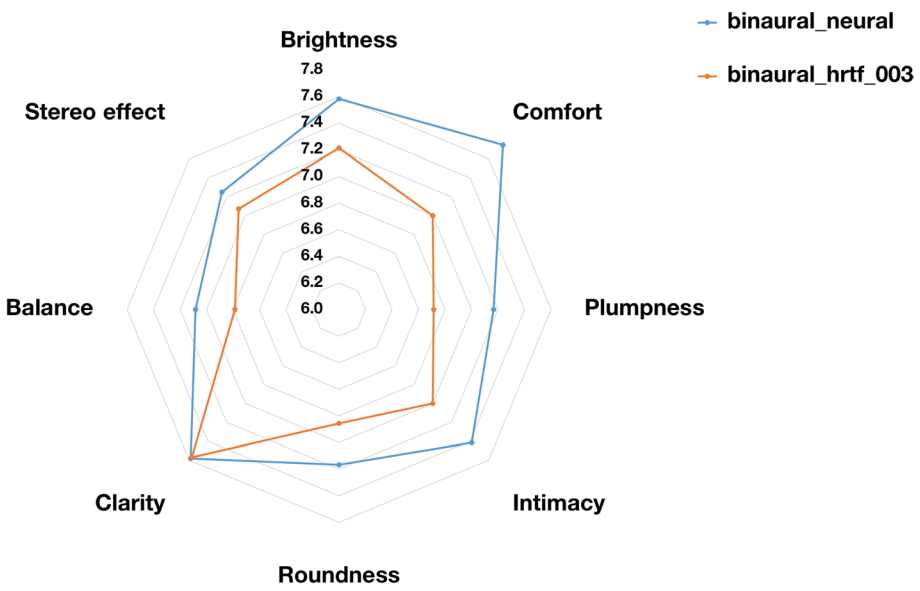

Fig. 9 Results of subjective evaluation

4.3 Virtual concert soundstage design

The designed virtual soundstage included 12 speakers: four top left, right front and rear speakers, two right rear and left rear speakers, two left and right surround sound field speakers, three left, center, and right speakers, and one subwoofer, as shown in Fig. 10. The top left and right front and rear speakers of the virtual soundstage used the same full-range design and were placed according to the main listening seat. The right rear speaker and left rear speaker increased the intensity of the listening experience by further positioning the sound, placing them behind the seating area at an angle of 135° to 150° to the center. The left surrounds sound field speakers and right surround sound field speakers played a role in creating a realistic sense of space and providing ambient sound. The two are arranged in the seat position slightly behind the

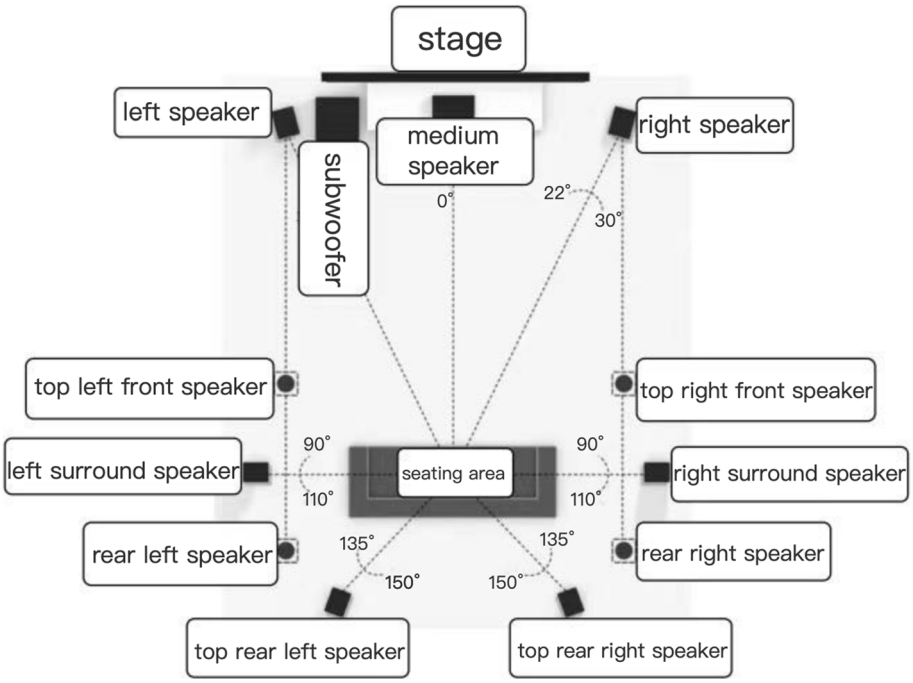

Fig. 10 Design of virtual concert soundstage

area and form a certain angle, preferably just above the ear height. The left, center, and right speakers assisted the music with the change of stage lighting. The subwoofer emitted the strongest bass, thus adding power to the music.

The 12 speaker placements designed in the UE4 virtual stage allowed each speaker to emit a different sound, each with its independent source, forming a new front, surround, and ceiling sound channel. Thus the external surround sound brought an immersive sound experience.

5 Conclusion and future work

In this paper, we have proposed a framework for meta-universe music generation from intelligent music generation and spatial audio twinning. Through subjective evaluation and objective experiments on The results of subjective evaluation and objective experiments on POP909 and HTRF datasets show that MetaMGC achieves superior results in both music generation and digital audio twinning.

However, although the model makes a good contribution to generating musical compositions, it is still not perfect. An important characteristic of live concerts is that listeners can feel and immerse themselves in the emotion and atmosphere conveyed by the music at close range [45], while our model only improves on the musicality of the music. Therefore, a music generation model that generates emotionally rich music is a better choice [46]. In subsequent experiments, we will also consider adding emotional expression factors to the digital audio twin system to make the meta-universe concert intelligent music generation framework closer to realistic emotion-rich live concert scenarios.

Abbreviations

Transformer-XL: Transfomer extra long; MC: Monte Carlo method; DQN: Deep Q network method; UE4: Unreal Engine 4; RL: Reinforcement learning; RNN: Recurrent neural network; LSTM: Long short-term memory; VAE: Variational Auto-Encoder; GAN: Generative adversarial network; MIDI: Musical instrument digital interface; REMI: REvamped MIDI-derived events; DSP: Digital singnal processor; L2-loss: Mean Squared Loss; HRTF: Head related transfer functions; SAL: Scene audio library.

Acknowledgements

We thank LetPub (www.letpub.com) for its linguistic assistance during the preparation of this manuscript.

Authors' contributions

Cong Jin conceived the algorithm.Fengjuan Wu performed the experimental part and wrote the article. Jing Wang sorted out the whole framework.Yang Liu, Zixuan Guan and Zhe Han participated in literature research and data compilation. All authors read and approved the final manuscript.

Funding

This study is the result of the research project funded by the National Key R&D Program of China (Grant No. 2021YFF0900700), the National Natural Science Foundation of China (Grant No. 62207029 and 62271454), Beijing Natural Science Foundation (Grant No. L223033) and supported by the Fundamental Research Funds for the Central Universities (Grant No. CUC220B018).

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

[1]School of Information and Communication Engineering, Communication University of China, Dingfuzhuang East Street, 100024 Beijing, China. [2]School of Information and Electronicsy, Beijing Institute of Technology, Zhongguancun Nanda Street, 100081 Beijing, China.

Received: 25 July 2022 Accepted: 3 November 2022
Published online: 13 December 2022

References

1. L.H. Lee, Z. Lin, R. Hu, Z. Gong, A. Kumar, T. Li, S. Li, P. Hui, When creators meet the metaverse: A survey on computational arts. arXiv preprint arXiv:2111.13486 (2021)
2. Sony: Making Madison Beer's Immersive Reality Concert Experience. https://www.youtube.com/watch?v=qm1PWZoWI (2021)
3. P. Danowski. Connexion. https://www.youtube.com/watch?v=OUE8V8x28wQ (2019)
4. P. Danowski. Sound of the metaverse: A breif history of virtual reality music instruments and virtual music venues. https://panopticon.am/a-brief-history-of-virtual-reality-music-instruments-and-virtual-music-venues (2021)
5. T. Scott. Travis scott and fortnite present: Astronomical. (2020)
6. C. Jin, T. Wang, X. Li, C.J.J. Tie, Y. Tie, S. Liu, M. Yan, Y. Li, J. Wang, S. Huang, A transformer generative adversarial network for multi-track music generation. CAAI Transactions on Intelligence Technology. 7(3):369–380(2022)
7. Z. Wang, K. Chen, J. Jiang, Y. Zhang, M. Xu, S. Dai, X. Gu, G. Xia, Pop909: A pop-song dataset for music arrangement generation. arXiv preprint arXiv:2008.07142 (2020)
8. I. D. Gebru et al., "Implicit HRTF Modeling Using Temporal Convolutional Networks," ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 3385–3389. https://doi.org/10.1109/ICASSP39728.2021.9414750
9. A. Kumar, in *Immersive 3D Design Visualization: With Autodesk Maya and Unreal Engine 4*. Interactive Visualization with UE4 (Apress, Berkeley, 2021), pp. 103–115
10. N. Boulanger-Lewandowski, Y. Bengio, P. Vincent, Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. arXiv preprint arXiv:1206.6392 (2012)
11. E. Waite, et al., Generating long-term structure in songs and stories. Web Blog Post Magenta.**4**,231-234(2016)
12. G. Hadjeres, F. Nielsen, Interactive music generation with positional constraints using anticipation-rnns. arXiv preprint arXiv:1709.06404 (2017)
13. Johnson D D. Generating polyphonic music using tied parallel networks[C]//International conference on evolutionary and biologically inspired music and art. Springer, Cham, 2017: 128-143.
14. S.R. Bowman, L. Vilnis, O. Vinyals, A.M. Dai, R. Jozefowicz, S. Bengio, Generating sentences from a continuous space. arXiv preprint arXiv:1511.06349 (2015)
15. A. Roberts, J. Engel, C. Raffel, C. Hawthorne, D. Eck, in *International conference on machine learning*. A hierarchical latent vector model for learning long-term structure in music (PMLR, 2018), pp. 4364–4373. http://proceedings.mlr.press/v80/roberts18a/roberts18a.pdf
16. B. Jia, J. Lv, Y. Pu, X. Yang, in *2019 International Joint Conference on Neural Networks (IJCNN)*. Impromptu accompaniment of pop music using coupled latent variable model with binary regularizer (IEEE, 2019), pp. 1–6
17. L.C. Yang, S.Y. Chou, Y.H. Yang, Midinet: A convolutional generative adversarial network for symbolic-domain music generation. arXiv preprint arXiv:1703.10847 (2017)
18. L. Yu, W. Zhang, J. Wang, Y. Yu, in *Proceedings of the AAAI conference on artificial intelligence*. Seqgan: Sequence generative adversarial nets with policy gradient, AAAI, vol. 31 (2017). https://doi.org/10.1609/aaai.v31i1.10804
19. H.W. Dong, W.Y. Hsiao, L.C. Yang, Y.H. Yang, in *Proceedings of the AAAI Conference on Artificial Intelligence*. Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment, AAAI, vol. 32 (2018). https://doi.org/10.1609/aaai.v32i1.11312
20. C.Z.A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, I. Simon, C. Hawthorne, A.M. Dai, M.D. Hoffman, M. Dinculescu, D. Eck, Music transformer. arXiv preprint arXiv:1809.04281 (2018)
21. C. Donahue, H.H. Mao, Y.E. Li, G.W. Cottrell, J. McAuley, Lakhnes: Improving multi-instrumental music generation with cross-domain pre-training. arXiv preprint arXiv:1907.04868 (2019)
22. Y.S. Huang, Y.H. Yang, in *Proceedings of the 28th ACM International Conference on Multimedia*. Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions (2020),ACM Multimedia, pp. 1180–1188. https://doi.org/10.1145/3394171.3413671
23. Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q.V. Le, R. Salakhutdinov, Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860 (2019)
24. S. Li, J. Peissig, Measurement of head-related transfer functions: A review. Appl Sci **10**(14), 5014 (2020)
25. C. Armstrong, T. McKenzie, D. Murphy, and G. Kearney, "A Perceptual Spectral Difference Model for Binaural Signals," Engineering Brief 457, (2018 October.). http://www.aes.org/e-lib/browse.cfm?elib=19722
26. W. Zhang, P.N. Samarasinghe, H. Chen, T.D. Abhayapala, Surround by sound: A review of spatial audio recording and reproduction. Appl. Sci. **7**(5), 532 (2017)
27. Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R.J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, et al., Tacotron: Towards end-to-end speech synthesis. arXiv preprint arXiv:1703.10135 (2017)
28. J.Y. Lee, S.J. Cheon, B.J. Choi, N.S. Kim, E. Song, in *INTERSPEECH*. Acoustic Modeling Using Adversarially Trained Variational Recurrent Neural Network for Speech Synthesis (2018), INTERSPEECH ,pp. 917–921
29. S. Vasquez, M. Lewis, Melnet: A generative model for audio in the frequency domain. arXiv preprint arXiv:1906.01083 (2019)
30. A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A.W. Senior, K. Kavukcuoglu, Wavenet: A generative model for raw audio. SSW **125**, 2 (2016)
31. A. Defossez, G. Synnaeve, Y. Adi, Real time speech enhancement in the waveform domain. arXiv preprint arXiv:2006.12847 (2020)
32. D. Rethage, J. Pons, X. Serra, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. A wavenet for speech denoising (IEEE, 2018), pp. 5069–5073.Calgary, AB, Canada.
33. N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, K. Kavukcuoglu, in *International Conference on Machine Learning*. Efficient neural audio synthesis. Efficient neural audio synthesis (PMLR, 2018), pp. 2410–2419.Available from https://proceedings.mlr.press/v80/kalchbrenner18a.html
34. N. Mor, L. Wolf, A. Polyak, Y. Taigman, A universal music translation network. arXiv preprint arXiv:1805.07848 (2018)
35. A. Richard, D. Markovic, I.D. Gebru, S. Krenn, G.A. Butler, F. Torre, Y. Sheikh, in *International Conference on Learning Representations*. Neural synthesis of binaural speech from mono audio ,In International Conference on Learning Representations. (2020)
36. P. Morgado, N. Nvasconcelos, T. Langlois, O. Wang, Self-supervised generation of spatial audio for 360 video. Adv. Neural Inf. Process. Syst.**31**,1-15 (2018)
37. R. Gao, K. Grauman, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2.5 d visual sound (2019), IEEE Piscataway, pp. 324–333.Honolulu, HI, USA.
38. I.M. Sobol. A Primer for the Monte Carlo Method (1st ed.). CRC Press, Boca Raton. (1994) https://doi.org/10.1201/9781315136448
39. N. Metropolis, S. Ulam, The monte carlo method. J. Am. Stat. Assoc. **44**(247), 335–341 (1949)
40. R.S. Sutton, A.G. Barto, Reinforcement learning: An introduction. Robotica **17**(2), 229–235 (1999)
41. L.C. Yang, A. Lerch, On the evaluation of generative models in music. Neural Comput. Applic. **32**(9), 4773–4784 (2020)
42. I.P. Yamshchikov, A. Tikhonov, Music generation with variational recurrent autoencoder supported by history. SN Appl. Sci. **2**(12), 1–7 (2020)
43. H.W. Dong, K. Chen, J. McAuley, T. Berg-Kirkpatrick, Muspy: A toolkit for symbolic music generation. arXiv preprint arXiv:2008.01951 (2020)
44. W. Li, *The intersection of audio music and computers-audio music technology [m]* (Fudan University Press, Shanghai, 2019), pp.297–316
45. L. Mou, J. Li, J. Li, F. Gao, R. Jain, B. Yin, in *2021 IEEE 4th International Conference on Multimedia Information Processing and Retrieval (MIPR)*. MemoMusic: A Personalized Music Recommendation Framework Based on Emotion and

Memory (IEEE, 2021), pp. 341–347. Tokyo, Japan. https://doi.org/10.1109/MIPR51284.2021.00064
46. L. Mou, Y. Zhao, Q. Hao, Y. Tian, J. Li, J. Li, Y. Sun, F. Gao, B. Yin, in *2022 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*. Memomusic Version 2.0: Extending Personalized Music Recommendation with Automatic Music Generation (IEEE, 2022), Taipei City, Taiwan. pp. 1–6. https://doi.org/10.1109/ICMEW56448.2022.9859356.

Publisher's Note