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

Combinatorial Press

Artificial Intelligence-driven Speech Signal Extraction and Generation Based on Wave RNN Models

Yanze Wang[1,✉], Xuming Han[1]

[1] *School of Information Science and Technology, Jinan University, Guangzhou, Guangdong, 510632, China*

ABSTRACT
In the era of artificial intelligence, the technology of speech conversion has developed rapidly and has gradually become a hot topic of research in the field of speech processing. This paper explores the problem of speech signal extraction and generation based on Wave RNN model, and constructs a speech conversion generation model driven by artificial intelligence. First, the short-time Fourier transform is utilized to convert and preprocess the speech signal in the time-frequency domain. Second, a stepwise speech enhancement model is proposed to enhance the perceived strength of the speech signal. Then, a speech generation model based on improved self-attention mechanism and RNN is designed to realize the generation of speech signals. Finally, the model effect is evaluated for application. The time-frequency domain feature that mixes time-domain features and frequency-domain features is able to capture the characteristics of speech signals more comprehensively than a single time-domain feature and frequency-domain feature, which corresponds to a higher recognition accuracy and a lower training loss value. Meanwhile, after speech enhancement, the average accuracy of model A∼D speech recognition is improved by 19%∼25%, which indicates that the stepped speech enhancement model used in this paper can substantially enhance the perceptual strength of speech signals. In addition, the language conversion model in this paper outperforms other speech conversion models in both MCD and RMSE, and its advantage in rhyme mapping is obvious, and the pitch of the output speech is more accurate and natural. The model in this paper has high practical value in speech signal generation and conversion.
Keywords: wave RNN; short-time Fourier transform; stepwise speech enhancement; speech conversion

1. Introduction

Voice is one of the main ways for human beings to carry out daily communication and exchange, and is an extremely important tool in the development of human society [1]. With the maturity of the Internet and the breakthrough of communication speed, it becomes simple and feasible to transmit large amounts of data on the mobile terminal, and people's interaction needs on the mobile terminal have increased dramatically, and software functions such as video voice and live chat have gradually

✉ Corresponding author.
E-mail address: hanxibird@163.com (Y. Wang).
Received 25 March 2024; Revised 05 June 2024; Accepted 15 December 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-300

come into daily life and become an indispensable part of daily communication [2-5]. Compared with the previous way of relying on keyboard input, voice interaction can greatly liberate the user's hands, through speech recognition, semantic understanding, speech synthesis, so as to realize the whole process of interaction [6-8]. However, with the development of modern technology, people need to communicate more and more information in their lives, so the amount of information in the communication process is also exploding in the development of technology, but at the same time, the bandwidth resources used for communication are not unlimited [9-11]. In the process of voice communication, before transmission at the sending end and storage at the receiving end, the voice signal needs to be compressed and encoded, the purpose of which is mainly to reduce the amount of transmission code and storage in the process of communication processing [12-14]. And low coding rate and high quality are often a pair of contradictory indicators, in the process of voice transmission at a lower bit rate, the voice quality will also have a certain degree of degradation [15-16]. The generation of low-rate speech compression coding technology can extract the feature parameters representing semantic information in the speech signal, followed by quantization coding and decoding inverse quantization, and the output side for the synthesis of the extracted feature parameters, and the final output reconstruction of the speech waveform [17-20]. Therefore, how to effectively extract and generate speech signals is a prerequisite to facilitate the digital transmission of speech signals [21].

In the current social environment, the bottleneck of traditional feature extraction methods is mainly in the efficiency and accuracy when dealing with large amounts of speech data. Literature [22] combines machine learning algorithms with audio signal processing techniques to achieve key feature extraction of audio signals through large-scale training and testing. Literature [23] also applies machine learning algorithms to the process of extracting feature parameters of speech signals, in addition to the use of fast algorithms to further improve the acceleration of feature parameter extraction, and enhance the performance of hardware and software processing of speech signals to complete the speech recognition process. Literature [24] shows that speech signal feature extraction is an important part of compression of raw speech data and enhancement of features in speech recognition, and proposes the Mel Frequency Cepstrum Coefficient Extraction Method (MSP-MFCC) under mixed signal processing architecture, whose extracted speech features can show good performance in low-power speech recognition applications. Literature [25] examined the speech signal processing method in speech emotion recognition, using algorithms such as Mel frequency cepstrum coefficients to extract features from speech signals, combining the feature selection stage with the feature classification stage to effectively recognize generic emotions in speech. Literature [26] describes the neural network-based target speech feature extraction method, which can separate the target speech under the acoustic conditions of noise, reverberation, and interfering loudspeakers by processing similar features in the speech signal.

In this paper, based on the Wave RNN model, the natural and accurate conversion of speech signal is realized by preprocessing the speech signal, speech enhancement and speech generation. Then the speech enhancement is divided into two steps, the first step generates the enhanced speech that is favorable to the machine hearing sense, and the second step combines the vocoder WaveRNN to get the speech that is suitable for the human ear hearing sense. Meanwhile, the acoustic model of speech synthesis is constructed by combining the multi-head self-attention mechanism, the relative position-aware self-attention mechanism and the RNN model. In order to verify the effect of the model, in terms of the speech enhancement model, the features in different time-frequency domains are analyzed and the recognition accuracy of the speech signal before and after enhancement is compared. In terms of the speech signal generation model, and the model is applied with the speech conversion task to compare the similarity and naturalness of the generated speech from both objective and subjective dimensions.

2. Artificial intelligence-driven speech signal extraction and generation models

In order to realize intelligent extraction and generation of speech signals, this paper first preprocesses the speech signals, then enhances the perceived intensity of the target speech signals by using a stepwise speech enhancement model, and constructs a sequence-to-sequence hybrid speech generation model by combining the improved self-attention mechanism and RNN to realize the synthesis of speech signals and their output.

2.1. Speech signal processing

Speech signals are analog signals that carry specific information. The extracted speech signal needs to be processed before proceeding to speech generation. Speech signal processing is the use of digital signal processing techniques to process the speech signal so that the information carried by the speech signal can be maximized.

2.1.1. Basic concepts of speech signals:

1) Sampling frequency: the total number of times per second the voice signal is sampled, the unit is Hertz (Hz), the common sampling frequency is 8kHz, 16kHz, 22.05kHz, 44.1kHz. The higher the sampling frequency, the more realistic and natural the sound reproduction.

2) Sampling precision: also known as bit depth, indicates the number of bits of information contained in each sample point, the unit is bit (Bit), such as 8-bit, 16-bit. 16-bit is the most common sampling precision.

3) Sample Size: This refers to the value of the amplitude of the sound waveform at that sampling point, in bits/s (i.e., bps).

2.1.2. Pre-exacerbation. The transmission of signals in a medium generates certain losses, and the higher the frequency, the higher the losses. In order to obtain the highest quality signal possible, the high frequency loss part needs to be compensated. In speech signals, high-frequency signals are mainly affected by the lip-oral radiation, in order to compensate for this part of the loss and enhance the high-frequency resolution of speech signals, pre-emphasis processing is required. The core idea of the pre-emphasis technique is to amplify the high-frequency signal through a first-order high-pass filter, so that the spectrum is balanced and the signal has uniform energy in different frequency bands, which is given by the following formula:

$$H(z) = 1 - \alpha z^{-1} \tag{1}$$

Where α is the pre-emphasis coefficient, generally take the value of $0.9 < \alpha < 1.0$. The speech sampling value at the moment of setting n is $x(n)$, and the result after the pre-emphasis processing is $y(n)$. The pre-emphasis processing is expressed by the differential equation as follows:

$$y(n) = x(n) - \alpha x(n-1) \tag{2}$$

2.1.3. Split-frame windowing. Fourier transform is used in speech signal processing to convert the speech signal from time domain signal to frequency domain signal. The input requirement for the Fourier transform is a smooth signal, but the speech signal is not smooth from the macroscopic point of view because the lip changes will lead to a change in the speech signal, but from the microscopic point of view the speech signal is smooth for a shorter period of time because the lip changes will not be very fast. Therefore, the speech signal can be intercepted for Fourier transform to fulfill the input requirement of Fourier transform. The process of splitting the speech signal into small segments is called subframing, and a small segment of the speech signal is called a “frame”. Macroscopically, there is no significant change in the speech signal within a phoneme, so the size of a frame has to be less than the length of a phoneme. The duration of a phoneme is 50ms-200ms, so the length of a frame of speech signal should be less than 50ms. From a microscopic point of view, a frame of speech signal

needs to contain enough vibration cycles for Fourier transform. The fundamental frequency of male voice is around 100 Hz, and the fundamental frequency of female voice is around 200 Hz, which translates into periods of 10 ms and 5 ms. Because of the need to include multiple vibrational cycles, the length of a frame of speech signal should be longer than 20 ms.

After the frame splitting operation, there will be discontinuities at the beginning and end of each frame. When the speech signal is split into more frames, the error between the original speech signal and the original speech signal will be bigger. The windowing operation solves this problem and makes the speech signal more continuous after being divided into frames. After adding window, the speech signal without periodicity shows some characteristics of the periodic signal. Currently, the mainstream window functions include rectangular window, Hanning window and Hemming window.

2.1.4. Fourier Transforms and Short-Time Fourier Transforms. A signal is in the time domain where the independent variable is time and the dependent variable is the change in the signal, so a time domain plot describes the relationship between amplitude and time. A signal is in the frequency domain where the independent variable is frequency and the dependent variable is the amplitude of the signal at that frequency, so the frequency domain plot describes the relationship between frequency and amplitude. For speech signals, features are harder to extract in the time domain, but are instead easier to extract in the frequency domain. The Fourier transform is the tool that converts the signal from the time domain to the frequency domain. The physical meaning of the Fourier transform is that a non-periodic signal is considered as the sum of infinitely many harmonic components with different and continuously varying frequencies [27]. The Fourier transformed signal is the spectrum of the original signal and is a complex function containing both amplitude and phase information. The Fourier transform equation is as follows:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \tag{3}$$

Where ω denotes frequency, t denotes time, $e^{-i\omega t}$ is a complex function, $F(\omega)$ is the image function of $f(t)$, and $f(t)$ is the image primitive function of $F(\omega)$. In speech signals, $f(t)$ is the time domain representation of the signal and $F(\omega)$ is the frequency domain representation of the signal.

Fourier transform is a powerful means to analyze linear systems and smooth signals, but speech signals are not smooth, so in practice, the short-time Fourier transform (STFT) is often used [28]. The short-time Fourier transform is a process of splitting a longer signal into multiple segments of equal length, performing the Fourier transform on these segments, and stitching the results together. In the case of speech signals, after the signal has been processed by splitting the frames and adding windows, it is only necessary to perform the Fourier transform on each frame and splice the results with the following formula:

$$STFT(t, f) = \int_{-\infty}^{\infty} x(\tau) h(\tau - t) e^{-i2\pi f \tau} d\tau \tag{4}$$

where $h(\tau - t)$ is the analysis window function and $x(\tau)$ is the source signal.

2.1.5. Mel spectrum. Mel spectrum and linear spectrum are two acoustic features of speech signals. The main steps of linear spectrum and Mel spectrum extraction are shown in Fig. 1. Linear spectrum is a speech signal feature obtained by pre-emphasizing the original speech signal language, adding windows in frames and short-time Fourier transform after several steps of operation. The linear spectrum reflects the changes of the speech signal in the frequency domain, but the relationship between the human ear's perception of the sound and the frequency of the sound is more complicated. Generally speaking, the human ear is more sensitive to low-frequency sounds compared to high-

frequency sounds. A set of Mel filters is used to process the linear spectrum to obtain the Mel spectrum, which makes the low-frequency characteristics of the linear spectrum more prominent in the Mel spectrum.

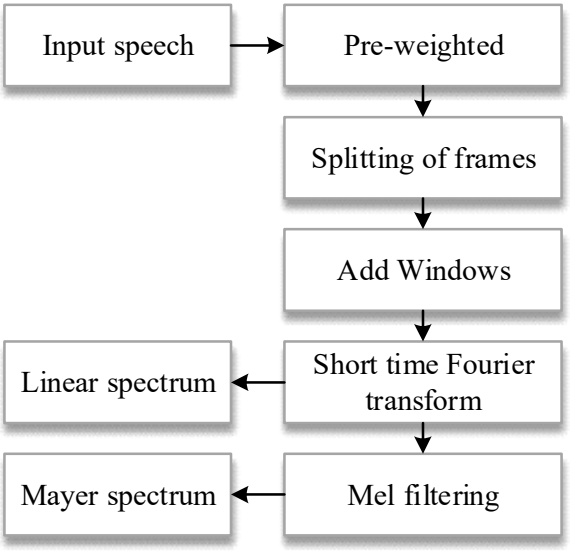


Fig. 1. Feature extraction process of linear spectrum and Meir spectrum

2.2. Stepped Speech Enhancement Model

The key technology to improve speech quality is to enhance the perceived strength of the target signal, i.e., speech enhancement, the essence of which is to suppress the noise signal expression so as to enhance the target speech content. In this paper, a stepped speech enhancement framework is proposed, and the main idea of the method is to design a stepped cascade network, where different enhanced speech is obtained at each stage respectively.

2.2.1. General Architecture of Speech Enhancement Modeling. The general structure of the stepped speech enhancement model is shown in Figure 2. The model divides speech enhancement into two stages, the first stage is suitable for generating enhanced speech suitable for machine perception, and the second stage continues to enhance on the basis of the first stage, transforms the linear spectrum into Mel spectrum, and then synthesizes speech signals suitable for human ear perception by using the Wave RNN vocoder, which better avoids the influence of the noise phase information on the final result after enhancement by the traditional method and the resulting speech sound quality This can better avoid the influence of noise phase information on the final result and the resulting speech sound quality after enhancement by traditional methods.

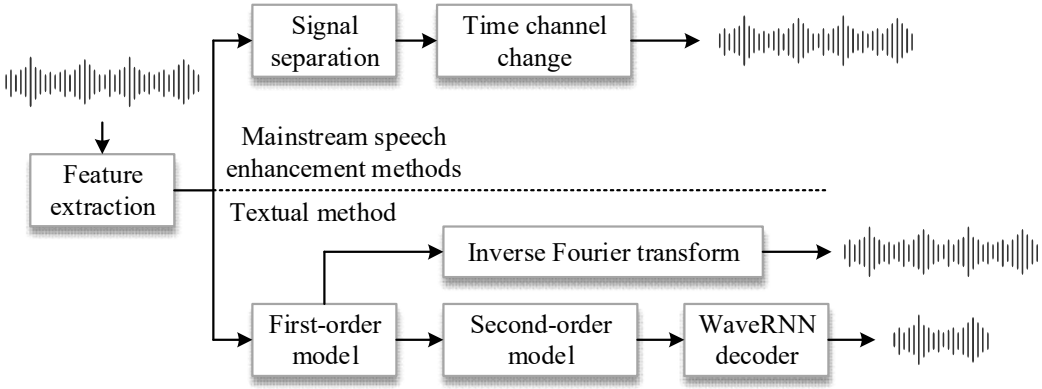


Fig. 2. The overall structure of the stepped speech enhancement model

1) Model Input. In most speech enhancement studies, the time-domain signal is converted to a frequency-domain signal for study and then inverted to a time-domain signal. The specific expression is shown in Equation (5):

$$X_n(e^j\omega) = \sum_{m=-\infty}^{+\infty} x(m)\omega(n-m)e^{-j\omega m} \tag{5}$$

Where: n is the key to distinguish from the standard Fourier transform, and $\omega(n-m)$ is the sequence of window functions. For the Fourier transform, different window functions will get different transformation results, generally when $\omega = 2\pi k / N$, get the short-time Fourier transform expression as shown in equation (6):

$$X_n(e^j\omega) = X_n(k) = \sum_{m=-\infty}^{+\infty} x(m)\omega(n-m)e^{-j\frac{2\pi km}{N}}, 0 \le k \le N-1 \tag{6}$$

In order to better utilize the signal energy in the frequency domain segment characteristics, this paper utilizes the Gaussian filter principle to make the energy in different intra-frame frequency bands more concentrated and filter part of the noise energy signal. At the same time, the original energy model is retained by splicing, combined with the Gaussian filter principle to enhance the expression in the low-pass frequency domain region, which is to some extent conducive to the overall noise suppression performance. The specific representation is shown in Eq. (7):

$$Y = (G(x_i),\ldots,G(x_n),Y_i,\ldots,Y_n),(0 \le i \le n) \tag{7}$$

Where Y denotes the model input, Y_i denotes the speech energy spectral information at frame i, $G(x)$ denotes the energy spectral information after Gaussian filter processing, the filtering formula is shown in (8), and the final input to the model is the splicing of the pre-activation and post-activation:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{8}$$

2) Speech enhancement framework based on machine hearing sense. The speech enhancement framework based on machine hearing sense is generally based on the design of DNN residual structure, and its main part consists of feed-forward neural network. In the network structure, each modular feature is directly fused with the features of the latter n+1 layers, which can eliminate the gradient disappearance in the deep structure of the network and help the fusion of local and global information. In the hidden layer structure for better integration of contextual information, this paper adds a global shared regular expression, using the network characteristic batch (number of frames of a single segment of speech) to strengthen the frame-to-frame connection, so that the context between independent frames can affect the global expression, eliminating the independence of frames from each other, the specific expression shown in equation (9):

$$\hat{x}^{(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{Var[x^{(k)}]}} \tag{9}$$

Where $E[x^{(k)}]$ denotes the global mean and $Var[x^{(k)}]$ denotes the global variance.

The idea of residual connection is borrowed from ResNet network structure on DNN, which makes better use of model expressivity. The residual injection expression is defined as shown in Eq. (10):

$$F(x) = H(x) + x \tag{10}$$

where x denotes the input vector, $H(x)$ is the deep layer output, and $F(x)$ denotes the short-connected transformed input of the different layers. The final output expression is built on an unactivated linear layer that expresses the final augmented amplitude value obtained by the model.

3) Speech enhancement framework based on human ear hearing sense. In this paper, we propose to use the enhanced amplitude spectral features for input and continue to enhance to obtain the enhanced Mel spectral features. In the enhancement framework based on human ear auditory perception, this paper discards the original phase information and reconstructs the speech timing signal through WaveRNN vocoder to get the final enhanced speech suitable for human ear auditory perception. The overall enhancement expression is shown in Eq. (11):

$$second\_order(first\_order_{output}) \rightarrow WaveRNN \rightarrow denoisy_{wav} \tag{11}$$

4) Vocoder Wave RNN. In order to better adapt to the future low-resource environment, this paper adopts the Wave RNN vocoder in the back-end speech reconstruction. The Wave RNN adopts a single-layer recurrent neural network structure in the structure, combined with dual softmax layer structure, and achieves the optimal level of synthesis on the current back-end synthesis, which can generate real-time 24KHz 16-bit high-fidelity speech. In generating speech, a long sequence is mainly represented by multiple short sequences, and each short sequence is generated at the same time, in such a way to ensure that the long sequence generation speed, so as to ensure that the final synthesized speech synthesis in real time.

In WaveRNN, in order to better ensure the sequence generation model real-time, the specific decoding speed is shown in Equation (12):

$$T(u) = |u| \sum_{i=1}^{N} (c(op_i) + d(op_i)) \tag{12}$$

Where $T(u)$ denotes the final vocalization time to generate a sentence, a sentence has $|u|$ sample points. n denotes the number of model layers, $c(op)$ denotes the time required for each layer of computation in the speech generation stage, and $d(op)$ denotes the hardware consumption time.

WaveRNN has two softmax layers, the high and low results are multiplied by GRU at the same time to get the value of the final sample point, as shown in Eqs. (13) to (20):

$$x_t = [c_{t-1}, f_{t-1}, c_t) \tag{13}$$

$$u_t = \sigma(R_u h_{t-1} + I_u^* x_t) \tag{14}$$

$$r_t = (R_r h_{t-1} + I_r^* x_t \tag{15}$$

$$e_t = t(r_t o(R_e h_{t-1}) + I_e^* x_t \tag{16}$$

$$h_t = u_t o h_{t-1} + (1 - u_t) \circ e_t \tag{17}$$

$$y_c, y_f = split(h_t) \tag{18}$$

$$P(c_t) = softmax(O_2 relu(O_1 y_c)) \tag{19}$$

$$P(f_t) = softmax(O_4 relu(O_3 y_f)) \tag{20}$$

where * denotes the mask matrix, where the last coarse input ct is connected only to the fine part of state u_r, r_t, e_t, h_t and therefore only affects the fine output y_f. The fine parts c_t and f_t are encoded as scalars of [0, 255] and scaled to the interval [1, 1]. Matrix R, consisting of matrix R_u, R_r, R_e, is computed as a single matrix-vector product to produce contributions to all three gates u_i, r_i and e_i. The activation functions are standard sigmoid and tanh nonlinear functions. One possible architectural variant is to have h_i depend only on x_{t-1} and use a fully connected layer that is then summed or connected.

2.2.2. Loss function:

1) Loss function. In previous speech enhancement models, most of them follow the Euclidean distance as the final optimization function, and the mathematical formulation of the root mean square error (MSE) loss function is obtained as shown in Equation (21):

$$E_r = \frac{1}{N}\sum_{n=1}^{N} \| \hat{X}_n(Y_{n-}^{n+}, W, b) - X_n \|_2^2 \tag{21}$$

where E_r denotes the root mean square error, $\hat{X}_n(Y_{n-l}^{n+l}, W, b)$ and X_n denote the estimated and true eigenvalues at a sampling rate of n, respectively, N is the number of features, and Y_{n-l}^{n+l} denotes the number of relevant frames in the inputs $(2\times l+1), W, b$ denotes the set of two covariates of the model, respectively, and in general $l=5$.

In this paper, the MSE and cosine similarity measures are united so that the two vectors not only have approximate modal values but also have the same direction. The modified mathematical expression of MSE loss is shown in Eq. (22), respectively:

$$E_{cr} = \frac{1}{N}\sum_{n=1}^{N} \frac{\| \hat{X}_n(Y_n^{n+l}, W, b) - X_n \|_2^2)}{\cos(\hat{X}_n(Y_n^{n+}, W, b) - X_n)} \tag{22}$$

where E_{cr} denotes the corrected error. Then for the reverse update process, the w', b' -iteration expression for layer l is shown in Eqs. (23) to (28):

$$\Delta(W_{n+1}^l, b_{n+1}^l) = -\lambda \frac{\delta E_c r}{\delta(W_n^l, b_n^l)} - \gamma\lambda(W_n^l, b_n^l) + \omega\Delta(W_n^l, b_n^l), 1 \le l \le L+1 \tag{23}$$

$$\Delta(W(n+1)^l, b(n+1)^l) = -\lambda \frac{(\phi'\psi - \phi\psi')}{\psi^2 - \Gamma'} \tag{24}$$

$$1 \le l \le L+1$$

$$\psi = cosa(\hat{X}_n(Y_n^{n+l}, W, b) - X_n)$$

$$\psi' = \delta \frac{cosa(\hat{X}_n(Y_n^{n+l}, W, b) - X_n)}{\delta(W_n^l, b_n^l)} \tag{25}$$

$$\phi = \| \hat{X}_n(Y_n^{n+l}, W, b) - X_n \|_2^2 \tag{26}$$

$$\phi' = \frac{\partial \| \hat{X}_n(Y_n^{m+l}, W, b) - X_n k_2^2}{\partial(W_n^l, b_n^l)} \tag{27}$$

$$\Gamma' = \gamma\lambda(W_n^l, b_n^l) + \omega\Delta(W_n^l, b_n^l) \tag{28}$$

where L denotes how many layers the model has in total, $L+1$ denotes the regression layer, γ denotes the weight decay coefficient, and ω denotes the update momentum.

2) Joint loss function. In order to better express the performance of the stepwise model, in this paper, an end-to-end training method is used, and a joint optimization loss is designed to combine the results of the two phases and train the model together, which strengthens the linkage of the model in the whole, and helps to improve the overall performance. The specific loss function is shown in Eq. (29):

$$L_{total} = \lambda_1 E_{cr}(\hat{X}_{Magnitude}, Target_{Magnitude} + \lambda_2 E_{cr}(\hat{X}_{Mel}, Target_{Mel}) \tag{29}$$

where λ_1, λ_2 denotes the weighting relationship between the two models, respectively. In the joint training of this paper, the two models before and after are set to be equally important proportions, and the final weight distribution is $\lambda_1 = 1, \lambda_2 = 1$. $\hat{X}_{Magnitude}, \hat{X}_{Mel}, T\arg et_{Magnitude}$ and $Target_{Mel}$ denote the amplitude spectrum spectra after first-order enhancement, the Mel spectra after second-order enhancement, the target amplitude spectra and the target Mel spectra, respectively.

2.3. Speech generation model based on self-attention mechanism and RNN

In this section, a hybrid sequence-to-sequence speech generation model based on RNN and improved self-attention mechanism is proposed, which not only models the global context-dependent information in text and speech features well, but also takes into account the temporal relationship in the speech generation process, which greatly improves the stability of the sequence-to-sequence acoustic model based on the self-attention mechanism, and at the same time obtains a better naturalness of speech generation.

2.3.1. Multi-pronged self-attention mechanisms. In this paper, we use a parallelizable multi-head self-attention mechanism to learn the hidden features in the encoder and decoder. In the traditional single-head attention mechanism, given the query vector Q of the decoder and the output features V of the encoder, the computation procedure is:

$$Attention(Q,K,V) = f(Q,K)V \tag{30}$$

Where, Q corresponds to the decoder hidden layer features $H = \{h_1, h_2, ..., h_N\}$, V corresponds to the encoder output representation L, V is the index matrix of the encoder output features, and $f(Q,K)$ is the scoring function using softmax normalization. The Transformer framework uses the dot product scoring function:

$$f(Q,K) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{31}$$

where d_k is the dimension of the index vector K.

In contrast, in the multi-head attention mechanism, we decompose Q,K,V into different subspaces by linear transformation and compute the content vectors in each subspace using the attention mechanism in Eq. (31):

$$MultiHead(Q,K,V) = Concat(head_1, \ldots, head_P)W^O \tag{32}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{33}$$

where P indicates that this multi-head attention mechanism contains P heads, $head_i$ is the content vector of the ith subspace, and W^Q, W^K, W^V, W^O is the trainable matrix of different linear transformations, respectively. By learning different content vectors from multiple subspaces, the multi-head attention mechanism is able to jointly learn better content representations from different subspaces.

The self-attention mechanism is a special case of the attention mechanism, which aims to learn the hidden layer representation of the input features themselves, rather than learning the content vectors between the encoder and decoder's [29]. In the multi-head self-attention mechanism, both Q and V come from the same input features. As an example, the hidden layer representation $h_t = AttRNN(g(y_t), s_{t-1})$ of the decoder at moment t is learned by the self-attention mechanism:

$$h_t = MultiHead(Y,K,Y) \tag{34}$$

where $Y = \{y_1, ..., y_N\}$ is the input vector for all moments of that hidden layer, and K is the index vector obtained from Y by linear transformation. Similarly, the above equation can be used to learn each hidden layer feature representation in the encoder and decoder.

From Eq. (34), it can be found that the hidden layer feature h_t at moment t is able to directly access the input information of all moments at a distance of $O(1)$, which means that the self-attention mechanism is able to model the global contextual information well. Meanwhile, since the structure does not contain recurrent neural network units, it can avoid the problem of gradient vanishing in RNN and can be trained in full parallel, which greatly improves the training speed of the model. It is worth

noting that since the decoder in the sequence-to-sequence model generates each frame output vector autoregressively, this means that all hidden layer features after the moment t cannot be used in the process of generating $\hat{y}_{t+1}$. Therefore, in the decoder based on the self-attention mechanism, each self-attention layer executing the scoring function needs to add a masking vector with negative infinity to the weights of all connections after t moments before the softmax operation to ensure that the model will not compute inputs after t moments during parallel training.

2.3.2. Self-attention mechanism based on relative position perception. Self-attention mechanisms that do not take into account the connection between similar features may lead to the problem of distraction during feature learning. In this paper, we propose to solve this problem by using a relative position-aware self-attention mechanism for modeling speech generation models.

The process of learning the hidden layer representation feature $H = \{h_1, ..., h_N\}$ by the self-attention mechanism can be written as follows:

$$h_i = \sum_{j=1}^{N} \alpha_{ij} (x_j W^V) \tag{35}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{N} \exp(e_{ik})} \tag{36}$$

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_k}} \tag{37}$$

where $X = \{x_1, ..., x_N\}$ is the input of the self-attention layer, N is the length of the input and output sequences, and e_{ij} i.e., the connection relationship between the output of frame i and the input of frame j. From this equation, it can be seen that in the process of generating the output h_i of the hidden layer, $x_j W^K$ directly determines the contribution of each frame input sequence x_j to the generation e_{ij}.

In the relative position-aware self-attention mechanism, in this paper, a trainable local connection network is added to the original global connection network, so as to take more into account the input representations of the nearest neighbors at the current moment in the model training and generation process. At this time e_{ij} the computational process is:

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K + a_{ij})^T}{\sqrt{d_k}} \tag{38}$$

where a_{ij} is the nearest-neighbor connectivity network, representing the additional contribution of x_j to the generation of e_{ij}.

In order to effectively enhance the nearest neighbor connections and at the same time be able to apply to all lengths of training and test data, assuming that we want to enhance the front and back m connections at the current moment, the relative position-aware self-attention mechanism uses the operation of truncation to ignore the connections with a distance greater than m in the nearest neighbor network, at which time each nearest neighbor connection a_{ij} can be written as:

$$a_{ij} = \omega_{clip(j-i,m)} \tag{39}$$

$$\mathrm{clip}(\Delta t, m) = \max(-m, \min(m, \Delta t)) \tag{40}$$

where $\omega = \{\omega_{-m}, ..., \omega_m\}$ is a trainable matrix to control the contribution of the nearest neighbor input features. In order to ensure the parallelizable computational capability of the model, this truncation

operation directly assigns the boundary connection vectors of the nearest-neighbor networks to the more distant connections.

2.3.3. Hybrid speech generation model. Although the self-attention mechanism uses position-encoded information to model the temporal information of sequences, neither absolute nor relative position representation can well represent the temporal relationships within the sequences, while RNN can model the temporal information well, but suffers from the problems of weak global dependency modeling ability and vanishing gradient [30]. In order to model the global context information of sequences well and retain the temporal information of sequences, this paper further proposes to construct a hybrid speech generation model using a relative position-aware self-attention mechanism and RNN networks to enhance the naturalness of synthesized speech. The framework of the hybrid speech generation model proposed in this paper is shown in Fig. 3, which mainly includes a multiplexed text encoder and an autoregressive decoder based on the self-attention mechanism. Among them, the text encoder contains an RNN-based coding sub-module and a coding sub-module based on the self-attention mechanism, and the outputs of the two sub-modules are spliced together after a layer of linear transformation as the final text feature representation.

In the coding module of RNN, this paper uses the CBHG network to model the temporal relationships of text sequences [31]. The main purpose of the coding module based entirely on the self-attention mechanism is to model the global contextual information in the text, which is closely related to the rhythms of the generated speech. The self-attention mechanism-based coding module contains a total of N operation block, each of which contains two sublayers: a self-attention mechanism layer and a feed-forward neural network layer. The operation process of each sub-layer is as follows:

$$y = LayerNorm(x + SubLayer(x)) \tag{41}$$

where x is the input of each sublayer, y is the output of each sublayer, SubLayer denotes the self-attention mechanism or feed-forward neural network sublayer, and LayerNorm denotes the output of the neuron for normalization. With the multiplexed text encoder, we are able to retain both the temporal modeling capability of RNN and the global dependency modeling capability of the self-attention mechanism, which significantly improves the stability and effectiveness of the speech generation model. Although the decoder of the framework is also based on an autoregressive structure, the decoder uses a fully auto-attentive mechanism to learn the hidden layer features, so the decoder can be trained in full parallel.

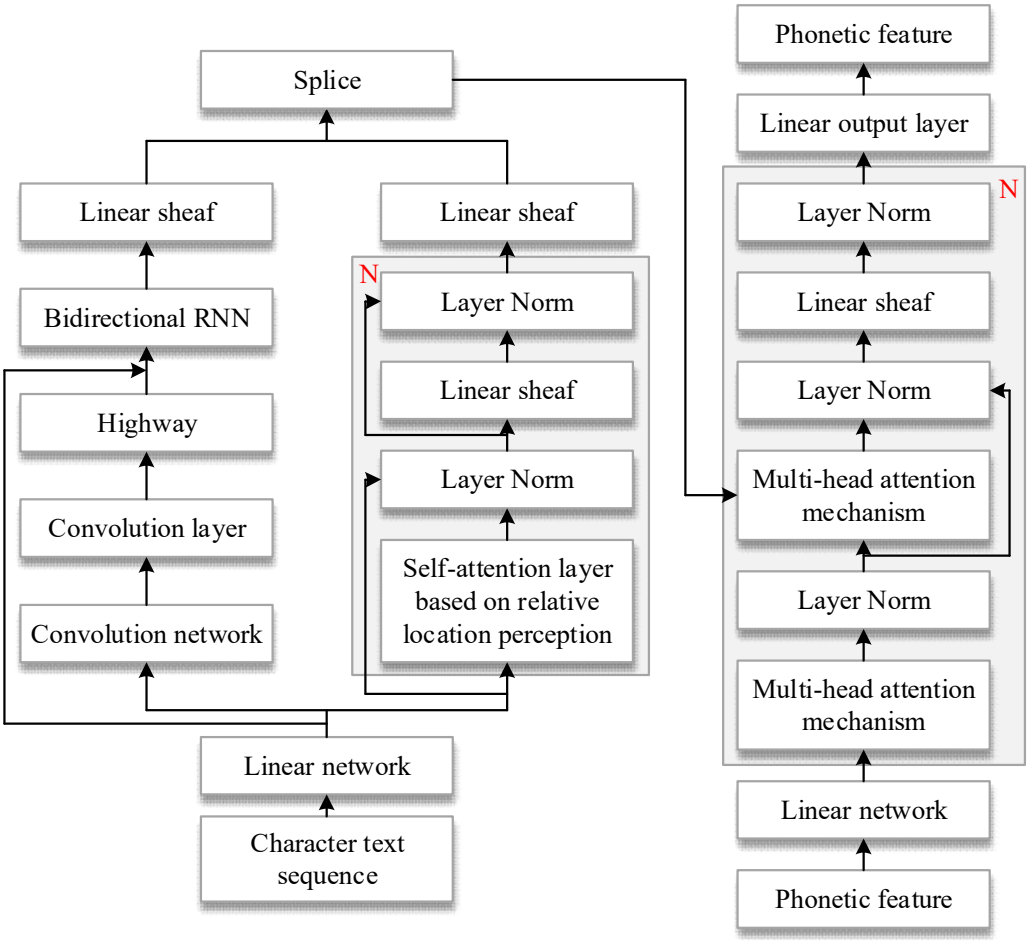


Fig. 3. Hybrid acoustic model framework based on self-attention mechanism and RNN

3. Speech enhancement modeling experiments and analysis

3.1. Experimental environment and parameters

In this paper, experiments are conducted on two public datasets, RAVDESS and SAVEE, to validate the model performance. To enhance data diversity, various data enhancement techniques are employed to facilitate the model to learn a wider range of feature representations. The datasets are divided into training, validation, and testing sets in the ratio of 8:1:1 for effective model training and evaluation. For deep learning framework selection, Keras is used for model training and hyperparameters are adjusted by grid search method to optimize model performance. The optimal hyperparameters identified include a batch size of 64 and appropriate weight selection to improve the weighted average accuracy of the model in this paper. In addition, the "Adam" optimizer and the "categorical cross-entropy" loss function were chosen to facilitate effective model learning.

To ensure that the model learns sufficiently, the model is trained on a high-performance Tesla P100 GPU for 120 training cycles, and the effectiveness of the model is verified by comparing the model in this paper with the other three models.

3.2. Analysis of data results

3.2.1. Different characterizations. In this section, firstly, different features are extracted in the SAVEE dataset, which are time-domain features (ZCR, RMS), frequency-domain features (MFCC, Log-Mel, and CQT), and hybrid features composed of time-domain plus frequency-domain (ZCR, RMS, MFCC, LogMel, and CQT), and experiments are carried out by utilizing the speech enhancement model of this paper, and except for the different features, the other conditions are all the same, and a comparison of the

results of different features is obtained as shown in Table 1. Where UAR denotes unweighted average accuracy and WAR denotes weighted average accuracy.

According to the evaluation metrics presented in Table 1, the performance differences of different feature combinations in the classification task can be observed. The hybrid time-frequency feature set exhibits excellent performance in all key performance metrics, reaching 67.86%, 66.01%, 68.31%, 69.18%, 68.74%, and 67.38% in UAR, WAR, Precision, Recall, F1-Score, and Accuracy, respectively. In contrast, the frequency domain features, although slightly lower than the hybrid time-frequency features on most of the metrics, still outperform the case of using only the time domain features. The performance of time-domain features in all evaluation metrics is unsatisfactory, with its unweighted average accuracy of 64.22%, weighted average accuracy and accuracy failing to exceed 64%. It can be shown through experiments that the method of fusing time-frequency features proves to have obvious advantages. And the time domain features, although not as directly reflecting the speech state as the frequency domain features, contain key clues such as the beginning, end and duration of the sound. In order to fully utilize the potential of these features, hybrid time-frequency features are used, and careful trade-offs and adjustments are made based on the specific needs of the task and the characteristics of the data to optimize the model performance.

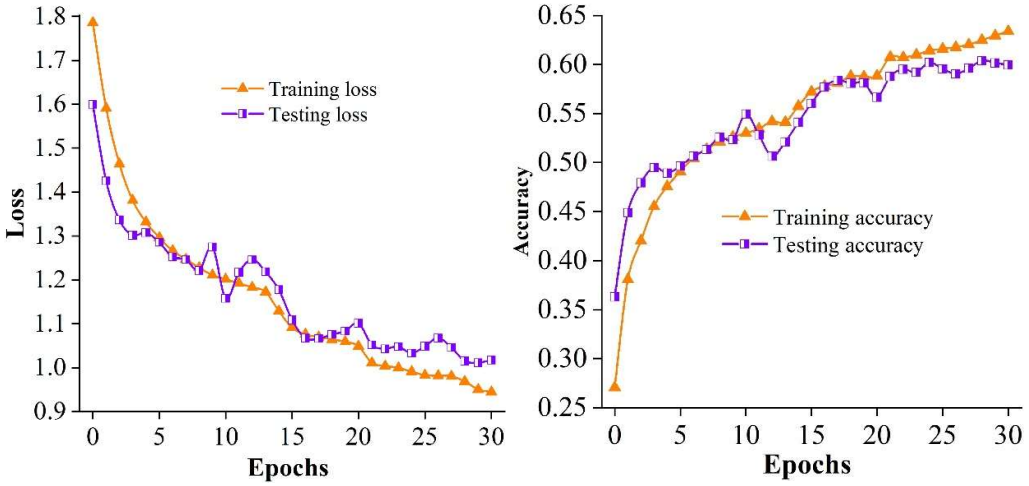
Table 1. Comparison of results with different features

Feature	Evaluation index/%					
	UAR	WAR	Precision	Recall	F1-Score	Accuracy
Time domain	64.22	61.83	63.24	63.05	63.14	63.42
Frequency domain	66.54	63.12	66.07	64.23	65.14	64.54
Time-frequency domain	67.86	66.01	68.31	69.18	68.74	67.38

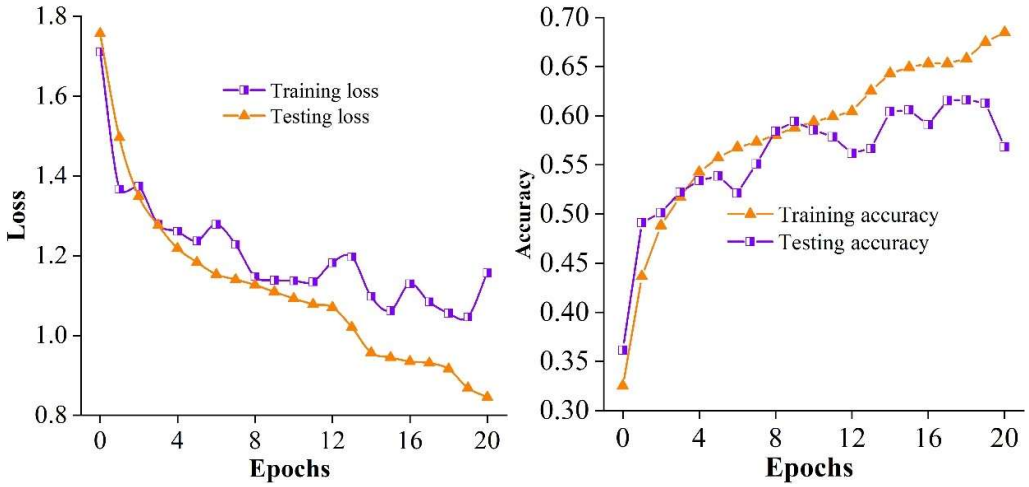
The loss function and accuracy curves under the time-domain feature, the frequency-domain feature, and the hybrid time-domain plus frequency-domain feature are shown in Figs. 4(a), 4(b), and 4(c), respectively, where the left graphs both represent the model loss function values on the training set and the test set, and the right graphs both represent the model accuracies on the training set and the test set.

As can be seen from Fig. 4(a), at the beginning of training, the loss function curves of both the training set and the test set show a significant decreasing trend, reflecting the gradual optimization of the model prediction performance. However, when the training proceeds to about the 28th cycle, the growth of the model performance stabilizes and the loss function curve becomes smooth. At the same time, the accuracy curve continues to climb, indicating that the model's accuracy in classifying samples continues to improve. Eventually, the model performance under the time-domain feature is 1.0112 loss function value and 63.42% accuracy. Then, further analyzing the experimental results of frequency domain features and time-frequency domain features, it is observed that the loss function curve also shows a decreasing trend while the accuracy curve rises, which further confirms the optimization trend of the model during the learning process of different features, and the model performance reaches a stable state at about the 19th of the training cycle, and the early stopping strategy is also effective at this stage. Under the frequency domain features, the final performance of the model is a loss function value of 1.0461 and an accuracy of 64.54%. Under the time-frequency domain features, the model performance is even more outstanding, reaching a loss function value of 0.9932 and an accuracy rate of 67.38%. By comprehensively comparing the experimental results of these different feature sets, it can be found that using a combination of time-domain and frequency-domain features is able to complement each other in terms of feature representation, thus capturing the characteristics of the speech signal in a more comprehensive way. This may be due to the fact that the time-domain features capture the temporal dynamics of the signal, while the frequency-domain features reveal information about the frequency distribution of the signal, and the combination of the two provides a richer and more comprehensive representation of the data for the model. Therefore,

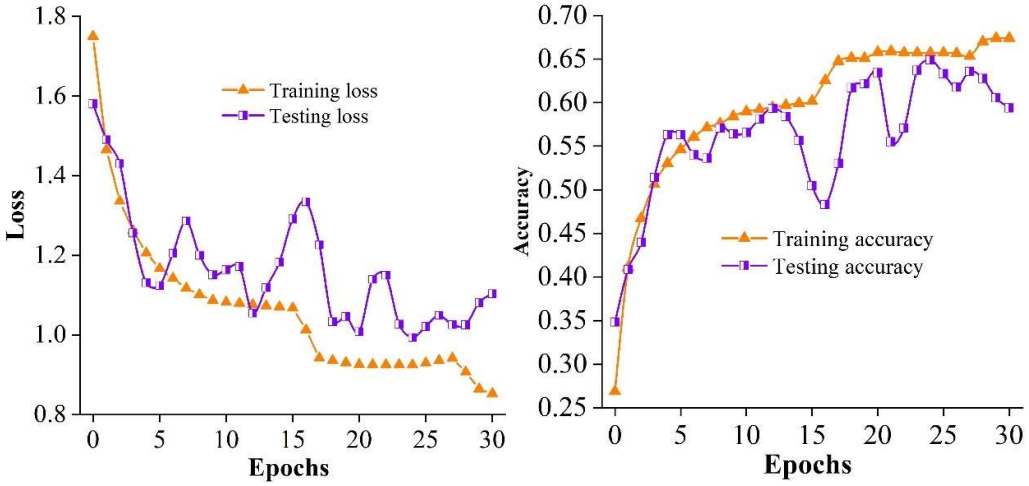
the combination of time-domain and frequency-domain features will continue to be used in subsequent studies for in-depth exploration.



(a) Time-domain features



(b) Frequency-domain features



(c) Time-frequency domain features

Fig. 4. Loss function and accuracy curves under different feature sets

3.2.2. Speech enhancement analysis. Facing the quality problem of speech, this paper adopts the stepped speech enhancement model to enhance the perceived strength of the target speech signal. In order to explore the effectiveness of the speech enhancement model, the performance of multiple models on different datasets is analyzed. The target speech recognition results before and after speech enhancement by different models are shown in Table 2. Among them, Models A~C represent the three

speech recognition models of CNN, CNN+BiLSTM, and CNN+BiLSTM+Attention, respectively, while Model D is the model after integrating Models A~C with weighted average.

Without speech enhancement processing, the performance of each model shows significant variability on different datasets. In particular, Model D demonstrates excellent classification performance on the RAVDESS and SAVEE datasets, achieving average accuracies of 67.39% and 64.85%, respectively, showing significant advantages over the other models.The high performance of Model D on the RAVDESS dataset may be attributed to the high complexity of the dataset, which are factors that increase the difficulty of the speech recognition task. On the contrary, the simplicity of the SAVEE dataset may contribute to the model's speech recognition, but its smaller size may limit the model's ability to generalize, resulting in relatively low accuracy. In contrast, after speech enhancement, the average accuracy of speech recognition of models A~D on the RAVDESS dataset is improved by 20.11%, 19.49%, 23.99%, and 20.22%, respectively, and on the SAVEE dataset by 24.18%, 21.81%, 20.57%, and 20.95%, respectively. This indicates that the stepped speech enhancement model used in this paper works significantly in enhancing the perceived strength of speech, and can substantially increase the likelihood of speech signals being accurately recognized.

Table 2. Different models recognition results before and after speech enhancement

Data set	Model	Pre-speech enhancement /%	Post-enhanced speech /%
RAVDESS	Model A	54.62	74.73
	Model B	59.35	78.84
	Model C	61.46	85.45
	Model D	67.39	87.61
SAVEE	Model A	54.67	78.85
	Model B	57.43	79.24
	Model C	60.85	81.42
	Model D	64.85	85.80

4. Experimental results and analysis of speech signal generation

In order to verify the effectiveness of the proposed speech signal extraction and generation model, this paper conducts model application experiments to evaluate the quality of converted speech obtained through speech extraction and generation from both objective and subjective perspectives.

4.1. Objective assessment

1) Evaluation metrics. In this paper, the model performance is evaluated based on the Mel's Cepstrum Distortion (MCD) and the Root Mean Square Error (RMSE) of $\log F_0$ between the modeled converted speech and the reference speech.

MCD is defined as:

$$MCD[dB] = 10/\ln 10\sqrt{2\sum_{d=1}^{D}(\hat{Y}_d - Y_d)^2} \tag{42}$$

where D is the characteristic dimension of the Mel cepstrum coefficients, and $\hat{Y}_d$ and Y_d are the d th dimensional characteristics of the transformed and original Mel cepstrum coefficients, respectively, with smaller values of MCD reflecting smaller distortions.

$\log F_0$ RMSE is defined as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\log \hat{F}_{0_i} - \log F_{0_i})^2} \tag{43}$$

where N denotes the total number of speech frames, and $\hat{F}_{0_i}$ and F_{0_i} denote the F_0 features of the converted speech and the reference speech frame i respectively.

Since all the conversion systems use Mel spectral features as acoustic features, it is not easy to obtain F_0 and Mel inverted spectral coefficient features in the converted acoustic features. In this paper, the Wave RNN vocoder is utilized to extract F_0 and 25-dimensional Mel inverted spectral coefficients from the reconstructed waveforms for evaluation. Considering the time difference, dynamic time planning is used to align the converted speech with the target reference speech to compute the MCD and $\log F_0$ RMSE, where the $\log F_0$ RMSE is computed only on the voiced frames in the reference utterance.

2) Objective assessment results and analysis. The results of the evaluation of the MCD and RMSE for the different systems are shown in Table 3, and the results are derived from the average of 288 transformed pairs (4 source speakers × 2 target speakers × 36 utterances) for each source-target gender group and 72 utterances (2 target speakers × 36 utterances) for the MSTTS speech synthesis for both genders. The "source" in the table represents the direct distortion between the source and target utterances, and since it was not transformed, it represents the worst case of distortion.

From the results, it can be seen that the model in this paper outperforms all other speech conversion models in both MCD and $\log F_0$ RMSE. Female speakers have higher $\log F_0$ -square deviations than male speakers, which is also reflected in the $\log F_0$ RMSE of the MS-TTS system that generates the speech of female target speakers (0.40 > 0.36). For both male and female target speakers, the model in this paper shows a clear advantage in rhyme mapping, which not only provides accurate linguistic content, but also a more natural pitch of the output speech.

Table 3. Average MCD and logF0 RMSE results

Source-target	Model	MCD (dB)	logF0 RMSE
Female-Female	The source	8.52	0.28
	AutoVC	4.11	0.31
	PPG-VC	4.24	0.34
	Ours	3.86	0.25
Female-Male	The source	8.59	0.51
	AutoVC	3.48	0.43
	PPG-VC	3.49	0.38
	Ours	3.32	0.32
Male-Male	The source	9.58	0.42
	AutoVC	3.62	0.31
	PPG-VC	3.64	0.39
	Ours	3.42	0.27
Male-Female	The source	10.47	0.65
	AutoVC	4.26	0.33
	PPG-VC	4.32	0.35
	Ours	3.98	0.28
Text-Female	MS-TTS	4.41	0.40
Text-Male	MS-TTS	3.93	0.36

By comparing the model in this paper with MS-TTS, it is noted that the difference lies in the encoder. The former takes speech as input while the latter takes text as input. The results show that the source speech is more informative than the text input in terms of providing speaker-independent rhyme patterns for target speech generation. This can be explained by the fact that the rhyme pattern of a sentence is a modulation between speaker-independent components (e.g., rhyming phrases, intonation) and speaker-related components (e.g., stress, pitch).

In order to visualize the speech conversion effect, male to female is taken as an example to compare the natural speech and the generated speech of each model F_0 shown in Fig. 5. Figures (a)~(f) represent the conversion effects of source, target, AutoVC, PPG-VC, MS-TTS and the model in this paper, respectively.

The following three observations can be obtained from Fig. 5:

1) PPG-VC, AutoVC and the model in this paper generate speech with the same duration as the source speech because they perform frame-level mapping.

2) The F_0-rhyme patterns of both PPG-VC and AutoVC are closer to the source utterances (their $\log F_0$ RMSEs are 0.22 and 0.26, respectively) than to the target utterances (their $\log F_0$ RMSEs are 0.24 and 0.27, respectively). This suggests that the implicit codes of both the PPG and the autoencoder are influenced by the source speaker.

3) The F_0-rhyme pattern of the model in this paper is closer to the target utterance ($\log F_0$ RMSE=0.89) than to the source utterance ($\log F_0$ RMSE=0.09). This indicates that the context vectors are speaker-independent, which allows the decoder to perform good speech restructuring of the target speaker using the target speaker embedding.

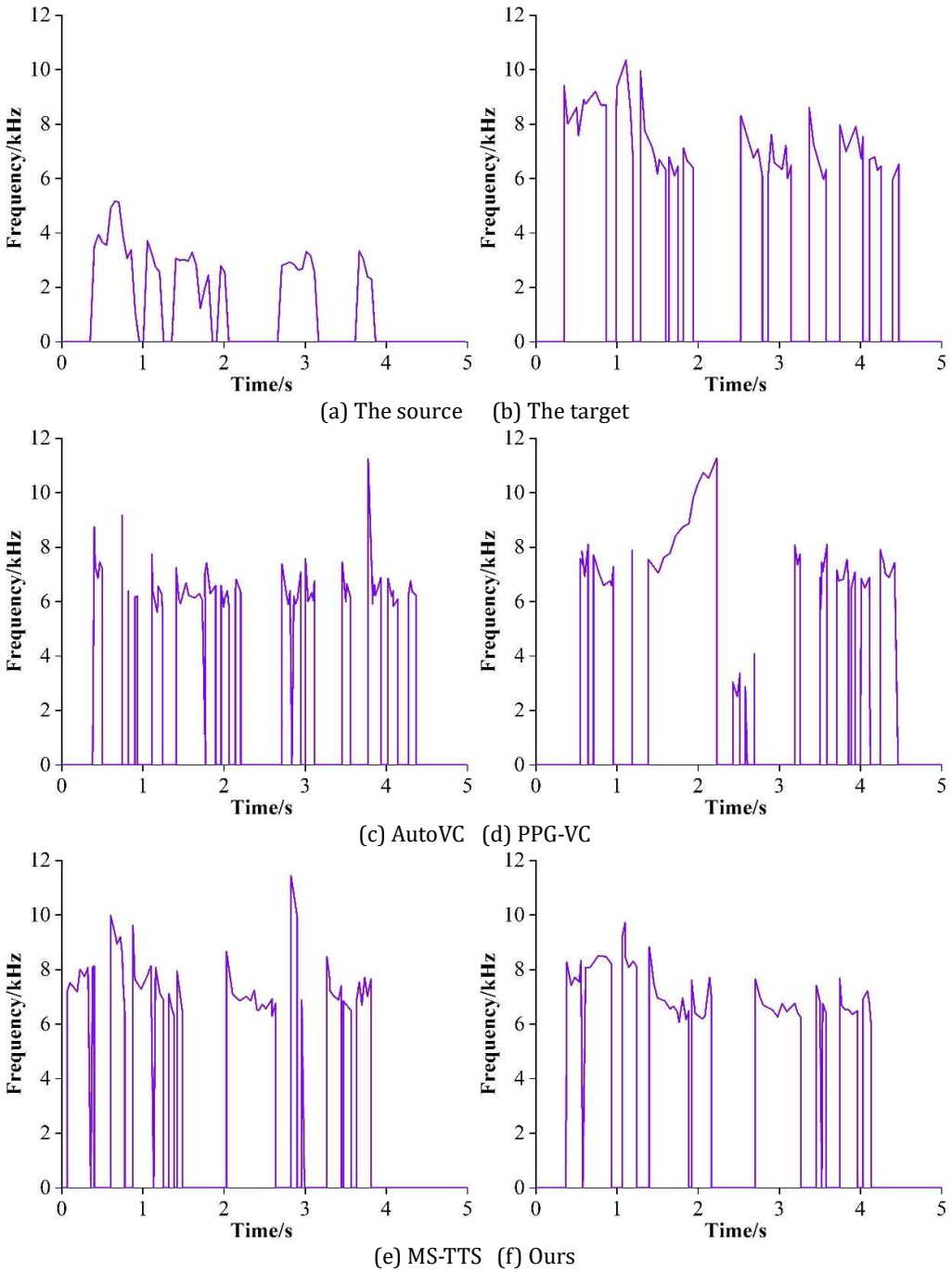


Fig. 5. Comparison example of a male-female speech conversion generating F0

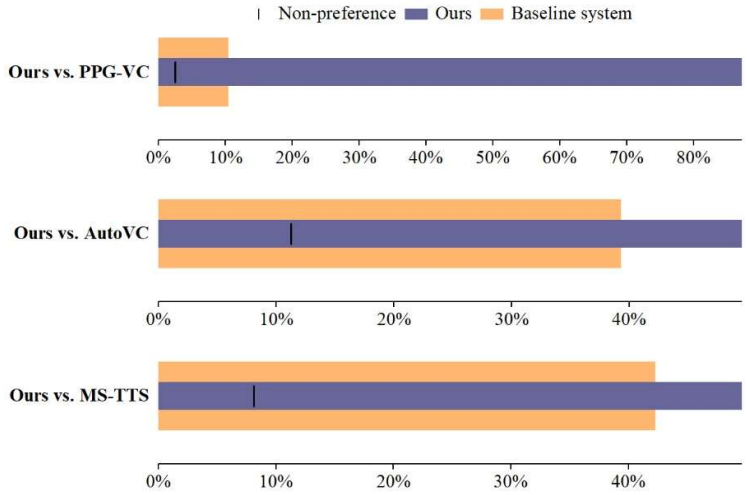
4.2. Subjective assessment

The subjective assessment was determined based on the test results of the listening test participants. In this paper, AB and XAB preference tests were conducted to assess speech quality and speaker similarity, respectively. Twenty-four samples were randomly selected from the converted samples of each experimental system and provided to each of the 18 participants for all listening tests. All listening test participants were university students and faculty members for whom English was the official language of instruction.

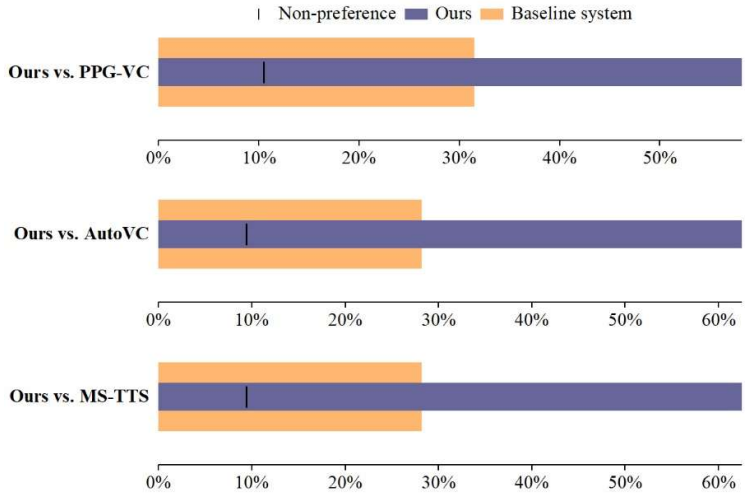
4.2.1. AB Preference Test. A and B were randomly selected speech samples from different systems, and the listening test participants were asked to select samples with higher quality and naturalness. Three sets of comparisons were made, and each participant evaluated eight samples separately. The

results of the ABC test experiments for the four source-target groups are shown in Fig. 6, (a) to (d) for female-female transitions versus text-female transitions, female-male transitions versus text-male transitions, male-male transitions versus text-male transitions, and male-female transitions versus text-female transitions, respectively.

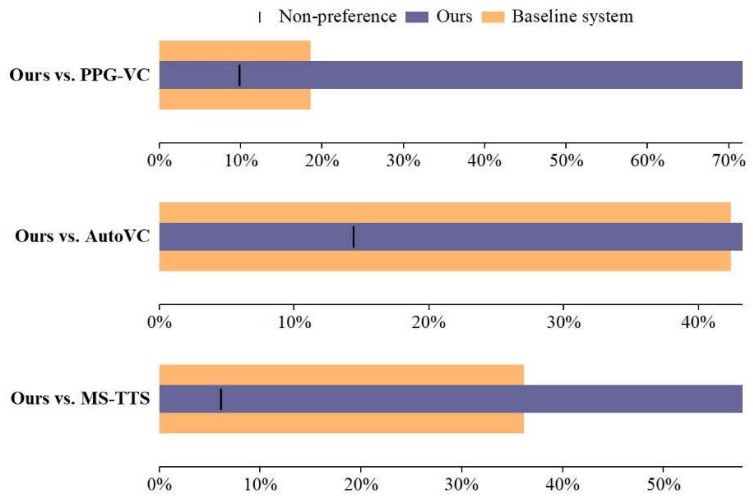
The p-value was calculated for all the experimental results and was <0.01 for all the experiments except for the "Ours vs. AutoVC" comparison experiment in the female-to-female transition, where the p-value = 0.016. Therefore, the performance improvement of this paper's model over other methods is statistically significant. By comparing this paper's model with PPG-VC and AutoVC models, it is observed that this paper's model also consistently outperforms PPG-VC and AutoVC in all source-target pairs. And the model in this paper outperforms the baseline MS-TTS in female-female, female-male, and male-female transitions, and slightly underperforms the MS-TTS in male-male transitions.Overall, the model in this paper outperforms the other baseline VC models and is on average comparable to the performance of the MS-TTS system.



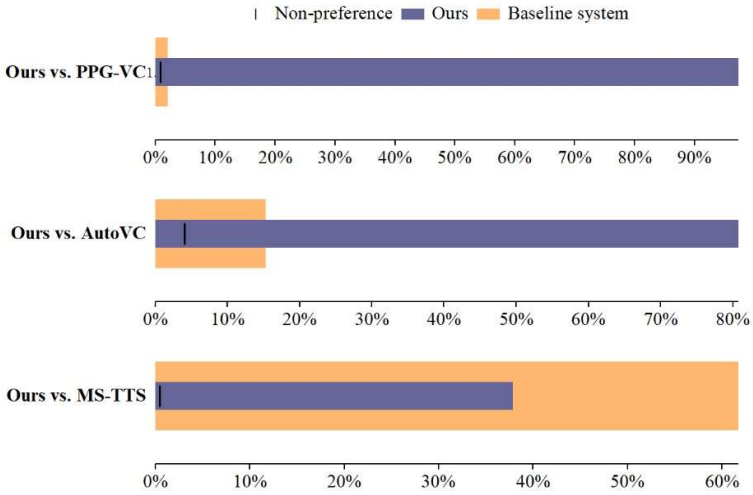
(a) Female-Female and Text-Female



(b) Female-Male and Text-Male



(c) Male-Male and Text-Male

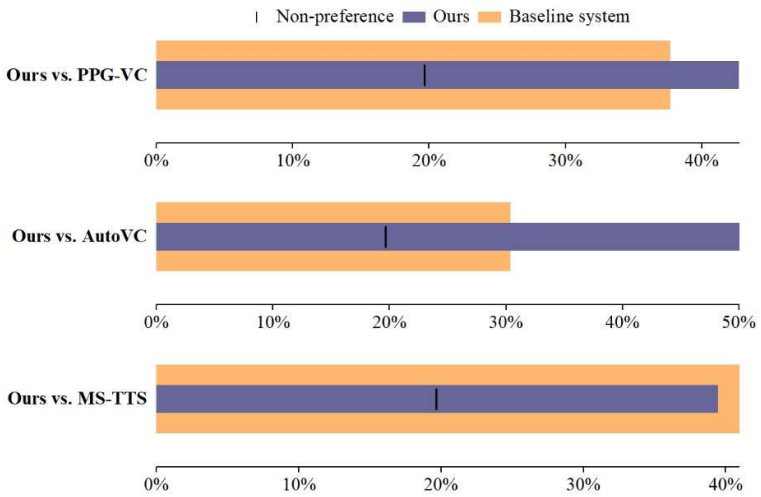


(d) Male-Female and Text-Female

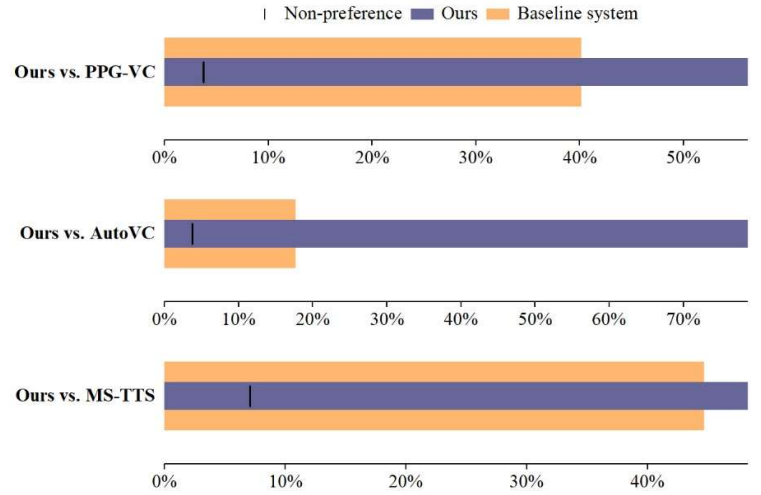
Fig. 6. Four source-target group AB test results

4.2.2. XAB preference testing. X is the reference target speech sample, and A and B are randomly selected speech samples from different systems. Listening test participants were asked to evaluate both test samples simultaneously and select the one that was more similar to the target speaker. For each comparison group, participants similarly evaluated an average of eight samples, and the ABC test results for the four source-target groups are shown in Figure 7, with (a) to (d) being female-female transitions versus text-female transitions, female-male transitions versus text-male transitions, male-male transitions versus text-male transitions, and male-female transitions versus text-female transitions, respectively.

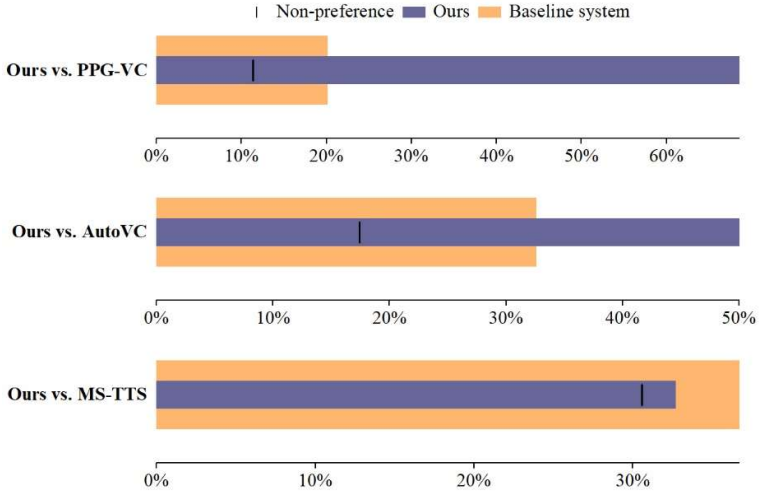
The p-value was calculated for all experimental results and was <0.01 for all experiments except for the “Ours vs. MS-TTS” comparison experiment in text-to-female conversion where the p-value = 0.029. This again can show that the performance improvement of this paper's model is statistically significant. By comparing this paper's model with other baseline models, it can be seen that this paper's model significantly outperforms PPG-VC and AutoVC for all speaker pairs in terms of speaker similarity. In addition, the performance of this paper's model is comparable to that of MS-TTS.



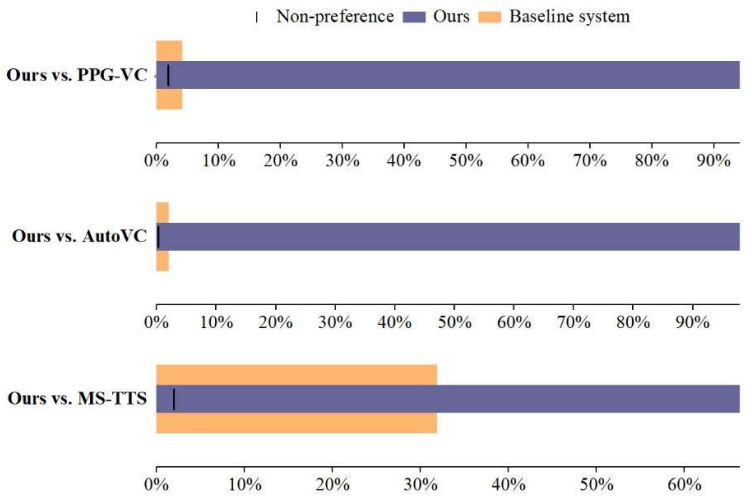
(a) Female-Female and Text-Female



(b) Female-Male and Text-Male



(c) Male-Male and Text-Male



(d) Male-Female and Text-Female

Fig. 7. Four source-target group XAB test results

It can be observed from the evaluation experiments that the speech signal extraction and generation model proposed in this paper significantly outperforms the AutoVC and PPG-VC baseline systems in terms of speech quality, naturalness, and speaker similarity, and is able to achieve the generated speech quality of TTS without the need to use text as an additional input at runtime.

5. Conclusion

In this paper, the processes of speech signal preprocessing, speech enhancement, and speech generation are converted naturally and accurately using the Wave RNN vocoder model.

The hybrid time-domain and frequency-domain feature set achieves 67.86%, 66.01%, 68.31%, 69.18%, 68.74%, and 67.38% in UAR, WAR, Precision, Recall, F1-Score, and Accuracy, respectively, which is significantly better than single time-domain features and frequency-domain features. And the model performance is more prominent under the time-frequency domain feature, reaching a loss function value of 0.9932 and an accuracy of 67.38%. It indicates that the combination of time-domain and frequency-domain features can capture the characteristics of speech signals more comprehensively and provide a richer and more comprehensive data representation for the model. Speech enhancement experiments were conducted using a mixture of time and frequency domain features. After speech enhancement, the average accuracy of speech recognition of models A~D is improved by 20.11%, 19.49%, 23.99%, and 20.22% on the RAVDESS dataset, and 24.18%, 21.81%, 20.57%, and 20.95% on the SAVEE dataset, respectively. It shows that the stepped speech enhancement model used in this paper can significantly enhance the perceived strength of speech signals and substantially increase the possibility of speech signals being accurately recognized.

In addition, the speech conversion model proposed in this paper outperforms all other speech conversion models in both MCD and $\log F_0$ RMSE. For both male and female target speakers, this paper's model shows a clear advantage in rhyme mapping, which not only provides accurate linguistic content, but also outputs speech with a more natural pitch. In the AB preference test and XAB preference test experiments, except for the p-value=0.016 for “Ours vs. AutoVC” in female-female transitions, the p-value=0.016 for “Ours vs. MS-TTS” in text-female transitions. “in female-to-female conversion with p-value=0.029, and p-value<0.01 in all other experiments, which indicates that the performance enhancement of the model in this paper is statistically significant compared to other methods, and it has a significant advantage in terms of the quality of the generated speech, its naturalness, and the similarity of the speakers.

References

[1] Prasad, K. S., Ramaiah, G. K., & Manjunatha, M. B. (2017). Speech features extraction techniques for robust emotional speech analysis/recognition. Indian J. Sci. Technol, 10(3), 1-9.

[2] Nassif, A. B., Shahin, I., Attili, I., Azzeh, M., & Shaalan, K. (2019). Speech recognition using deep neural networks: A systematic review. IEEE access, 7, 19143-19165.

[3] Abhishek, S., Sathish, H., Kumar, A., & Anjali, T. (2022, September). Aiding the visually impaired using artificial intelligence and speech recognition technology. In 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 1356-1362). IEEE.

[4] Raghavan, A. M., Lipschitz, N., Breen, J. T., Samy, R. N., & Kohlberg, G. D. (2020). Visual speech recognition: Improving speech perception in noise through artificial intelligence. Otolaryngology–Head and Neck Surgery, 163(4), 771-777.

[5] Benkerzaz, S., Elmir, Y., & Dennai, A. (2019). A study on automatic speech recognition. Journal of Information Technology Review, 10(3), 77-85.

[6] Sathya, R., Pavithra, M., & Girubaa, G. (2017). Artificial intelligence for speech recognition. International Journal of Computer Science & Engineering Technology (IJCSET), ISSN, 2229-3345.

[7] Alotto, F., Scidà, I., & Osello, A. (2020). Building modeling with artificial intelligence and speech recognition for learning purpose. In EDULEARN20 Proceedings (pp. 5866-5875). IATED.

[8] Yuan, T. (2024, December). Research on artificial intelligence speech recognition technology. In AIP Conference Proceedings (Vol. 3194, No. 1). AIP Publishing.

[9] Valentini-Botinhao, C., Wang, X., Takaki, S., & Yamagishi, J. (2016, September). Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech. In SSW (pp. 146-152).

[10] Zhang, M., Yuan, Z. M., Dai, S. S., Chen, M. L., & Incecik, A. (2024). LSTM RNN-based excitation force prediction for the real-time control of wave energy converters. Ocean Engineering, 306, 118023.

[11] Hsieh, T. A., Wang, H. M., Lu, X., & Tsao, Y. (2020). Wavecrn: An efficient convolutional recurrent neural network for end-to-end speech enhancement. IEEE Signal Processing Letters, 27, 2149-2153.

[12] El-Moneim, S. A., Nassar, M. A., Dessouky, M. I., Ismail, N. A., El-Fishawy, A. S., & Abd El-Samie, F. E. (2020). Text-independent speaker recognition using LSTM-RNN and speech enhancement. Multimedia Tools and Applications, 79, 24013-24028.

[13] Rezaul, K. M., Jewel, M., Islam, M. S., Siddiquee, K. N. E. A., Barua, N., Rahman, M. A., ... & Asha, U. F. T. (2024). Enhancing Audio Classification Through MFCC Feature Extraction and Data Augmentation with CNN and RNN Models. International Journal of Advanced Computer Science and Applications, 15(7), 37-53.

[14] Tirumala, S. S., Shahamiri, S. R., Garhwal, A. S., & Wang, R. (2017). Speaker identification features extraction methods: A systematic review. Expert Systems with Applications, 90, 250-271.

[15] Wen, G., Li, H., Huang, J., Li, D., & Xun, E. (2017). Random deep belief networks for recognizing emotions from speech signals. Computational intelligence and neuroscience, 2017(1), 1945630.

[16] Muckenhirn, H., Doss, M. M., & Marcell, S. (2018, April). Towards directly modeling raw speech signal for speaker verification using CNNs. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4884-4888). IEEE.

[17] Xu, C., Rao, W., Chng, E. S., & Li, H. (2020). Spex: Multi-scale time domain speaker extraction network. IEEE/ACM transactions on audio, speech, and language processing, 28, 1370-1384.

[18] Wang, D., & Chen, J. (2018). Supervised speech separation based on deep learning: An overview. IEEE/ACM transactions on audio, speech, and language processing, 26(10), 1702-1726.

[19] Yang, C. H. H., Qi, J., Chen, S. Y. C., Chen, P. Y., Siniscalchi, S. M., Ma, X., & Lee, C. H. (2021, June). Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6523-6527). IEEE.

[20] Aouani, H., & Ayed, Y. B. (2020). Speech emotion recognition with deep learning. Procedia Computer Science, 176, 251-260.

[21] Ghule, K. R., & Deshmukh, R. R. (2015). Feature extraction techniques for speech recognition: A review. International Journal of Scientific & Engineering Research, 6(5), 2229-5518.

[22] Sharma, G., Umapathy, K., & Krishnan, S. (2020). Trends in audio signal feature extraction methods. Applied Acoustics, 158, 107020.

[23] Abdusalomov, A. B., Safarov, F., Rakhimov, M., Turaev, B., & Whangbo, T. K. (2022). Improved feature parameter extraction from speech signals using machine learning algorithm. Sensors, 22(21), 8122.

[24] Li, Q., Yang, Y., Lan, T., Zhu, H., Wei, Q., Qiao, F., ... & Yang, H. (2020). MSP-MFCC: Energy-efficient MFCC feature extraction method with mixed-signal processing architecture for wearable speech recognition applications. IEEE Access, 8, 48720-48730.

[25] Koduru, A., Valiveti, H. B., & Budati, A. K. (2020). Feature extraction algorithms to improve the speech emotion recognition rate. International Journal of Speech Technology, 23(1), 45-55.

[26] Zmolikova, K., Delcroix, M., Ochiai, T., Kinoshita, K., Černocký, J., & Yu, D. (2023). Neural target speech extraction: An overview. IEEE Signal Processing Magazine, 40(3), 8-29.

[27] L. Srata, M. Chikri, S. Farres,I. Hamdani, Y. Tmimi & F. Fethi. (2024). A comparative study of discrimination and parameter identification of oil types using near-infrared spectroscopy, fourier transform infrared spectroscopy, and laser-induced fluorescence spectroscopy combined with chemometrics tools.Interactions(1),12-12.

[28] Martin Buck & Kasso A. Okoudjou. (2025). Tight frames generated by a graph short-time Fourier transform. Linear Algebra and Its Applications107-125.

[29] Han Li & Duan Qianqian. (2024). Solving method of traveling salesman problem based on performer graph self-attention mechanism. Signal, Image and Video Processing(1),154-154.

[30] Venkateswara RaoN & B. T.Krishna. (2024). CARNet: An Efficient Cascaded and Attention - Based RNN Architecture for Modulation Classification in Cognitive Radio Network Using Improved Kookaburra Optimization Strategy. International Journal of Communication Systems(2),e6088-e6088.

[31] Huang Tianyuan, Su Wei, Liu Lei, Zhang Jing, Cai Chuan, Yuan Yongna & Xu Cunlu. (2023). Translating Braille into Chinese based on improved CBHG model. Displays.