Fuzzy Speech Recognition Algorithm Based on Continuous Density Hidden Markov Model and Self Organizing Feature Map

Yanning Zhang
Telecommunication Engineering Institute
Beijing Polytechnic University, Beijing
100176, China
zhangyanning316@126.com

Lei Ma
Telecommunication Engineering Institute
Beijing Polytechnic University, Beijing
100176, China
malei235@126.com

Yunwei Li
Dean's Office, Beijing Youth Politics
College, China
lyw_lunwen@163.com

Abstract: *Speech recognition refers to the process of receiving and understanding human speech input through a computer, converting it into readable text or instructions. In order to improve the denoising effect and speech recognition effect of fuzzy speech, a fuzzy speech recognition algorithm based on continuous density hidden Markov model and self-organizing feature map is proposed. Firstly, the conventional Wiener filtering algorithm is improved by using the dynamic estimation algorithm of noise power spectrum, and the endpoint detection of noisy speech signal is performed by using spectral entropy, and the noise power spectrum of the silent segment is dynamically updated according to the detection results to obtain a more ideal priori signal to noise ratio; Secondly, the fuzzy speech is input into the Wiener filter to eliminate the noise in the speech signal; then, Mel-Frequency Cepstrum Coefficient (MFCC) of speech signal is extracted as speech feature; Finally, combined with the continuous hidden Markov model and the self-organizing feature neural network in the artificial intelligence algorithm, through the process of adjusting parameters, Viterbi decoding, and the time adjustment of the voice signal in the same state, the speech classification and recognition are realized according to the speech characteristics. In the experiment, comparative experiments were conducted on the LibriSpeech dataset using speech recognition algorithms based on convolutional neural networks and recurrent neural networks, speech recognition algorithms based on residual networks and gated convolutional networks, speech recognition algorithms based on multi-scale Mel domain feature map extraction. The experimental results show that the algorithm has good denoising performance. With the increase of added environmental noise intensity, the algorithm can maintain the Signal-to-Noise Ratio (SNR) of speech signals between 88dB-98dB; This algorithm can accurately detect the sound areas in the signal, and the endpoint detection accuracy is high; The accuracy and recall of the Continuous Density Hidden Markov Model-Self-Organizing Feature Neural Network (CDHMM-SOFM) designed in the algorithm increase with the number of iterations, and the highest levels of accuracy and recall can reach 0.89, respectively; The minimum recognition time of this algorithm is only 8.2 seconds, and the highest recognition rate can reach 98.7%; after applying this algorithm, the user's error rate ranges from 0.0031 to 0.0084. The above results indicate that the algorithm has good application performance.*

Keywords: *Speech recognition, wiener filter, Mel-frequency cepstrum coefficient, continuous hidden Markov model, self-organizing feature neural network.*

Received June 07, 2024; accepted November 14, 2024
https://doi.org/10.34028/iajit/22/2/11

1. Introduction

With the increasingly portable development of computers and the increasingly complex computing environment, people are eager to get rid of the shackles of keyboards and replace them with convenient, natural, and user-friendly input methods such as voice input [13]. Therefore, human-computer interaction using voice is an extremely important research topic. As a hot research topic in the field of high-tech applications, speech recognition has made considerable progress from theoretical research to product development [3]. It is directly connected with various practical application fields such as voice consultation and management in office, transportation, finance, public security, commerce, tourism and other industries, voice control in industrial production departments, automatic dialing, auxiliary control and query in telephone and telecommunications systems, and life support systems in medical and health care and welfare undertakings [7]. The research of voice signal recognition technology will be a highly marketable and challenging work.

Currently, the field of speech recognition is facing a series of challenges, which have been highlighted in the latest research. For example, in speech recognition algorithms based on convolutional neural networks and recurrent neural networks, Chinese characters are used as label samples, and training algorithms and sequence loss functions are used for iterative model training [18]. Although the algorithm has been effectively trained on Chinese character label samples, its denoising performance is limited and cannot completely eliminate noise in speech signals, which to some extent affects the accuracy of recognition. In speech recognition algorithms based on residual networks and gated

convolutional neural networks, the spectrogram is used as input to extract high-level abstract features through residual networks, and then effective long-term memory is captured through stacked gated convolutional neural networks. A feedforward neural network is added to the gated convolutional neural network to achieve speech recognition [19]. Although this algorithm combines residual networks and gated convolutional neural networks, and uses a concatenated temporal classification algorithm, there are still difficulties in detecting voiced regions in speech signals, and the detection performance of endpoints is average, which limits the application of the model in complex environments. In the speech signal recognition algorithm based on multi-scale Mel domain feature map extraction, the empirical mode decomposition method is used to decompose the speech signal, and filter bank features and their first-order differences are extracted from the three effective components respectively, thereby generating and constructing a new feature map, which captures the speech details in the frequency domain [23]. Although this algorithm has made some breakthroughs in speech signal recognition with articulation disorders, the recognition efficiency is not high due to the need for two rounds of feature extraction. Additionally, single channel neural networks suffer from effective feature loss and high computational complexity during training, which affects recognition efficiency.

In order to solve the problems in the above algorithms, this study proposes a new fuzzy speech recognition algorithm based on the Continuous Hidden Markov Model (CHMM) and Self-Organizing Feature Map (SOFM) neural network in the field of artificial intelligence. This algorithm has been improved based on the conventional Wiener filtering algorithm by dynamically estimating and updating the noise power spectrum, making the denoising process more adaptable to the noise changes in actual speech signals. Traditional Wiener filtering methods typically use fixed noise estimates and are difficult to handle non-stationary noise environments. And this algorithm utilizes spectral entropy to analyze the energy distribution of speech signals, dynamically adjusting the estimation of noise power spectrum, thus achieving good denoising effects at different noise levels. At the same time, this algorithm and method also have the characteristic of spectral entropy, which effectively detects the endpoints of speech signals in low signal-to-noise ratio environments, providing a reliable basis for subsequent noise power spectrum updates. In addition, this study combines CDHMM and SOFM in the field of artificial intelligence, utilizing the advantages of CDHMM in processing time series data to model and recognize speech signals, and utilizing the self-learning and feature extraction capabilities of SOFM to further optimize the feature representation in the recognition process. Through the collaborative work of these two algorithms, this algorithm not only improves the accuracy of recognition, but also accelerates recognition efficiency.

The rest of this article is organized as follows, in section 2, it was introduced that in order to improve the signal-to-noise ratio of speech signals, this study proposed a noise power spectrum dynamic estimation algorithm and improved the Wiener filtering algorithm based on this algorithm, which increased the signal-to-noise ratio of speech signals and laid the foundation for the accuracy of subsequent fuzzy speech recognition. Meanwhile, the sub-band spectral entropy endpoint detection method was also introduced, which is used to distinguish between speech segments and non-speech segments in the input signal, thereby reducing computational complexity and improving recognition rate.

In section 3, efficient speech feature extraction techniques and advanced artificial intelligence algorithms were combined to achieve accurate recognition of fuzzy speech signals. Among them, the extraction of Mel-Frequency Cepstrum Coefficien (MFCC) feature parameters ensure the discriminability and independence of the recognition algorithm, while the combination of CDHMM and SOFM further improves the adaptability and recognition accuracy of the algorithm, providing strong support for achieving efficient and accurate fuzzy speech recognition.

In section 4, the performance of the fuzzy speech recognition algorithm based on artificial intelligence proposed in this paper was verified through comparative experiments. The experiment used the LibriSpeech dataset to compare the performance of different algorithms in terms of signal-to-noise ratio, endpoint detection accuracy, recognition rate, and recognition efficiency. The experimental results show that the algorithm proposed in this paper outperforms other compared algorithms in dealing with background noise, improving endpoint detection accuracy, recognition rate, and recognition efficiency.

Section 5 summarizes the entire text and emphasizes the importance and application value of fuzzy speech recognition algorithms based on artificial intelligence in the field of speech recognition. At the same time, it also pointed out future research directions, including further optimization of algorithms and expansion of applications.

2. Speech Preprocessing

In order to improve the signal-to-noise ratio of the speech signal, this algorithm proposes a dynamic estimation algorithm of noise power spectrum, and improves the Wiener filter algorithm on this basis to remove the noise in the speech signal, so as to improve the accuracy of speech recognition.

However, prior to this, this article also designed two steps: anonymization of the speech signal acquisition

process and emphasis of the speech signal. Among them:

The anonymization of the voice signal collection process aims to protect the personal privacy information involved in the voice signal collection process, ensuring the security and privacy of the data. The processing method for this stage can include the following steps:

a) Sound desensitization: processing or modifying sound to make it impossible to directly identify personal identity or sensitive information from the sound.
b) Data encryption: encrypt the collected voice signals to ensure that the voice signal data is not easily stolen or tampered with during transmission and storage.
c) Anonymization processing: anonymize the collected voice data to remove personal identity information and sensitive identifiers, making it impossible to trace the source of the original data.

Speech signal emphasis is usually used in signal preprocessing to emphasize high-frequency components, improve the clarity and recognizability of speech signals. The processing method for this stage can include the following steps:

a) Pre emphasis filter: applying a first-order high pass filter to enhance the frequency characteristics of speech signals and reduce the influence of low-frequency components by enhancing the high-frequency part.
b) Filter design: set appropriate pre-emphasis filter coefficients, commonly used coefficients are 0.95.
c) Filtering processing: the speech signal is processed through a pre-emphasis filter to obtain the weighted speech signal, which is used for subsequent speech processing and feature extraction.

After completing the anonymization of speech signal acquisition and speech signal weighting processing, the Wiener filter algorithm is improved by using noise power spectrum dynamic estimation algorithm to input fuzzy speech into the Wiener filter and eliminate the noise in the speech signal. The specific process is as follows:

2.1. Sub-Band Spectral Entropy

The purpose of speech signal endpoint detection is to reduce the computation and improve the recognition rate by distinguishing the speech segment from the non-speech segment of the input signal. This process plays a key role in speech signal processing. In order to achieve robust endpoint detection, a sub-band spectral entropy endpoint detection method is proposed. Because of its good anti-noise effect, this method is widely used in speech recognition, speech coding and speech communication [8, 9].

a) Spectral entropy

The energy spectrum of a frequency component can be obtained by expanding the fast Fourier transform on the speech signal [16, 24]. The normalized spectral probability density of this frequency component can be defined as shown in Equation (1):

$$A_i = \frac{d(g_i)}{\sum_{k=1}^{M} d(g_k)} \tag{1}$$

In the above equation, i represents the voice signal of a certain frame analyzed; $d(g_i)$ represents the spectral energy of a frequency component g_i; M is the total length of Fourier transform. In order to reduce the impact of noise on endpoint detection and improve the ability to distinguish between voice segments and non-voice segments, according to the feature that voice signal energy is mainly concentrated in the range of 250Hz to 5000Hz, two constraints are added to the above equation, as shown in Equation (2):

$$d(g_i) = 0 \; if \;\; g_i < 250Hz \;\; or \;\; g_i > 5000Hz \tag{2}$$

Secondly, determine the upper and lower limits of probability density, as shown in Equation (3):

$$A_i = 0 \; if \;\; A_i < \varepsilon_2 \; or \; A_i > \varepsilon_1 \tag{3}$$

The lower limit ε_2 can be used to remove relatively constant power spectral density values at all frequency points, such as white noise; The upper limit ε_1 is used to remove noise concentrated at certain frequency points. After the above normalization and speech enhancement, the short-term spectral entropy of each analyzed speech frame can be defined as J_m, as shown in Equation (4):

$$J_m = -\sum_{K=1}^{M} A_K \log A_K \tag{4}$$

According to the above equation, spectral entropy has the following basic characteristics:

1. Speech signals and noise have different spectral entropy. This is because they are different in power spectrum and probability density distribution.
2. Compared with the energy feature, the speech spectral entropy has a small change.
3. The spectral entropy feature has certain anti-interference ability in noise environment.

b) Sub-band spectral entropy

In order to reduce the noise interference caused by the amplitude of each spectral point and improve the ability of the algorithm to distinguish speech signals from noise under low Signal-to-Noise Ratio (SNR) conditions, the concept of sub-band spectral entropy is proposed. Sub-band spectral entropy refers to dividing a frame of speech signal into multiple sub-bands, and then calculating the spectral entropy value of each sub-band.

Let $R_b(m,l)$ represent the energy of the m sub-band, and M_b represent the number of subbands divided by each frame of the voice signal. The probability of the

updated subband energy can be rewritten as $A_b(m,l)$, as shown in Equation (5):

$$A_b(m,l)=\frac{R_b(m,l)}{\sum_{k=1}^{M_b}R_b(k,l)} \tag{5}$$

Among them, $1\le m\le M_b$. The sub-band spectral entropy $J_b(l)$ can ultimately be defined in the form shown in Equation (6):

$$J_b(l)=\sum_{m=1}^{M_b}A_b(m,l)\log[\frac{1}{A_b(m,l)}] \tag{6}$$

c) Improvement of energy-weighted sub-band spectral entropy

Short-time energy is a calculation of the energy of a speech signal within a certain time window and is often used to describe the energy intensity of a speech signal. It can be obtained by calculating the sum of squares of the samples within the window. Improvement of sub-band spectral entropy using short-time energy can combine both short-time energy and sub-band spectral entropy to characterize the dynamics and harmonics of speech signals, which can be extracted to characterize the changes of speech signals in time and frequency, and help to improve the accuracy and robustness of speech processing tasks. The calculation process of energy-weighted sub-band spectral entropy is shown in Equation (7), in which and denote the short-time energy and spectral entropy values, and and denote the average of short-time energy and spectral entropy values of the previous frame, as shown in Equation (7):

$$EF(i)=\sqrt{1+\left|E(i)-E_m\right|*\left|H(i)-H_m\right|} \tag{7}$$

Translated with www.DeepL.com/Translator (free version)

2.2. Wiener Filter Based on Prior SNR Estimation

Wiener filtering has the characteristics of simple algorithm, easy implementation and good noise reduction effect [5, 12]. In order to realize the Wiener filtering algorithm in real time and find a balance between high quality noise reduction and low computational complexity, a Wiener filtering algorithm based on prior SNR estimation is proposed. The algorithm description is as follows:

Let $d(t)$ and $m(t)$ represent pure voice signal and noise respectively, then the observation signal $y(t)$ can be expressed in the form shown in Equation (8):

$$y(t)=d(t)+m(t) \tag{8}$$

Let $D_k=S_k e^{j\beta k}$, $M_k=N_k e^{j\chi k}$ and $Y_k=T_k e^{j\omega k}$ represent the k spectral components of pure speech signal $d(t)$, noise $m(t)$ and observed signal $y(t)$ respectively, then the posterior SNR $SNR_{pt}(g_k)$ and the prior SNR $SNR_{po}(g_k)$ can be defined as shown in Equation (9):

$$\begin{cases}SNR_{pt}(g_k)=|Y_k|^2/R\{|M_k|^2\}\\ SNR_{po}(g_k)=R\{|D_k|^2\}/R\{|M_k|^2\}\end{cases} \tag{9}$$

In the above equation, the amplitude is estimated under the assumption that the prior signal-to-noise ratio and the noise power spectral density function are known. Therefore, the Wiener filter H can be expressed in the form shown in Equation (10):

$$H=\frac{SNR_{po}(g_k)-1}{SNR_{pt}(g_k)} \tag{10}$$

2.3. Improved Wiener Filter

Wiener filter based on a priori signal-to-noise ratio estimation can effectively reduce "music noise" without causing high computational complexity. The accuracy of noise power spectrum estimation directly determines the denoising effect of the algorithm. Because the speech signal is random, it is unreasonable to use a fixed noise spectrum to estimate the prior SNR. In order to solve this problem, this paper proposes an improved Wiener filter algorithm based on spectral entropy and prior SNR estimation. The algorithm detects the endpoint of noisy speech signals through spectral entropy, and dynamically updates the noise power spectrum of the silent segment according to the detection results, so as to obtain a more ideal prior SNR and improve the denoising performance. The algorithm flow of improved Wiener filter is shown in Figure 1.

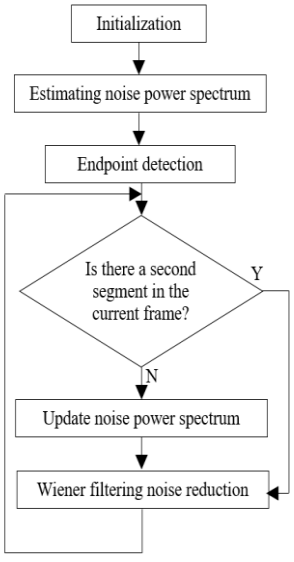


Figure 1. Improved Wiener filtering algorithm.

The specific process of improving the filtering algorithm is as follows:

- *Step* 1: pretreatment. After inputting a noisy speech signal, the signal is processed by framing and windowing, and the noise is pre-removed using spectral subtraction to improve the signal-to-noise ratio of the input speech signal. At the beginning, the

noise power spectrum of the first five frames of signal is calculated as the initial value of dynamic estimation.

- *Step* 2: Endpoint detection. The endpoint detection of noisy speech signals is performed using sub-band spectral entropy, and the starting and ending points of the vocal segments of the speech signal are recorded. This step is the combination of the sub-band spectral entropy algorithm and the Wiener filter algorithm based on a priori signal-to-noise ratio. Its main purpose is to effectively reduce the impact of noise on speech signal endpoint detection, and achieve the suppression of noise interference on speech recognition from the source.

The threshold of endpoint detection can be calculated by the following Equation (11):

$$T_s = \frac{\sum_{t=1}^{5} J_b(l)}{5} + \beta \frac{\sum J_b(l)^2}{5} \quad (11)$$

Where, β is an empirical value, usually equal to 1.25. If the detection result indicates that the current input voice frame is in the voice segment, go to step 3 to de-noise the frame signal; On the contrary, if the detection result indicates that the current input voice frame is in the silent segment, go to step 4 to dynamically update the noise power spectrum of the frame signal.

- *Step* 3: Wiener filtering noise reduction. The power spectrum obtained from the silent segment data of the latest frame is used as the average power spectrum of the noise, and the prior signal-to-noise ratio of the current frame is estimated. Then calculate the gain H_0 of Wiener filter according to the prior SNR obtained in the previous step; Multiply the power spectrum of the current frame by the filter gain H_0 to obtain the power spectrum of the speech signal after noise reduction. Finally, the de-noised speech signal is output through inverse Fourier transform.
- *Step* 4: Update the noise power spectrum. The current frame data is extracted, and the frame data is weighted with the data of the last voiceless segment, as shown in Equation (12):

$$|M_t(\xi)|^2 = (1-\eta)|M_{t-1}(\xi)|^2 + \eta |M_t(\xi)|^2 \quad (12)$$

Where, $|M_t(\xi)|^2$ is the noise power spectrum estimated by the current frame data; η is an adjustment factor used to adjust the weight of the current frame and the previous frame when the power spectrum is weighted. The weighted average of the above equation realizes the dynamic update of the noise power spectrum of the voiceless segment.

3. Fuzzy Speech Recognition

3.1. Speech Feature Extraction

A good characteristic parameter should have the following characteristics:

1. The extracted feature parameters can effectively represent speech features and have good distinguishability.
2. The parameters of each order have good independence.
3. The feature parameters should be calculated conveniently, and it is better to have an efficient calculation method to ensure the real-time realization of speech recognition.

In long-term practice, it is found that the cepstrum coefficient of Mel-frequency has the above advantages [6, 14]. The relationship between Mel scale and frequency scale is shown in Figure 2.

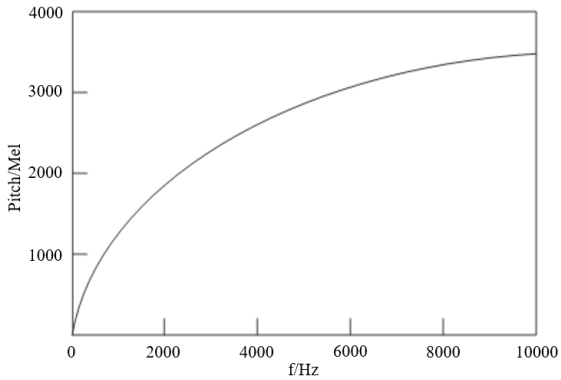


Figure 2. Relation between Mel scale and frequency scale.

When the frequency of sound is greater than 1000Hz, the human ear's perception of pitch does not follow a linear relationship, but follows an approximate linear relationship on the logarithmic frequency coordinate. Objectively, the pitch of sound is expressed by frequency, and its unit is Hz; Subjectively, measured by the unit "mel", it is based on the pure tone of 1000Hz and 40phon set as 1000 mel. For example, if the tone of a pure tone sounds twice as high as that of a 1000 mel voice, the tone will be set as 2000 Mel. The relationship between Mel scale and frequency scale is shown in Figure 2. The (Hz) horizontal axis is frequency and the vertical axis is beauty (Mel).

The calculation process of MFCC is shown in Figure 3.

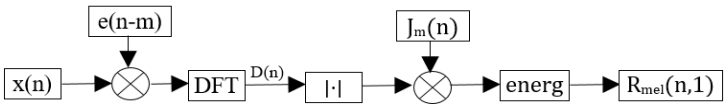


Figure 3. MFCC calculation process.

- *Step* 1: First, turn the voice signal into a short time signal after windowing and framing: assuming $x(n)$ is

a voice signal, the process of adding Hamming window as shown in Equation (13):

$$\begin{cases} \bar{x}(n) = x(n) \cdot e(n-m) \\ e(n-m) = 0.54 - 0.46\cos\dfrac{2\pi i}{m-1} \end{cases} \quad (13)$$

Where, $0 \le i \le m\text{-}1$, m is the window length.

- *Step* 2: Convert these time-domain signals into frequency-domain signals $A_n(f)$ with fast Fourier transform [11, 15], and calculate their short-time energy $A_n(e)$, as shown in Equation (14):

$$A_n(e) = |A_n(f)|^2 = |x_m(e^{je})|^2 \quad (14)$$

- *Step* 3: Add the triangular band-pass filter (24) to the Meier coordinate within the Meier frequency to obtain the filter bank conversion relationship.
- *Step* 4: Calculate the output $\vartheta(M_k)$ of energy spectrum $A(e)$ through this Mel filter bank. The frame calculation method is as follows: collect 12 of the center frequency above 1000Hz and below. The equation for $\vartheta(M_k)$ is shown in Equation (15):

$$\vartheta(M_k) = \ln \sum_{k=1}^{K} |A_k(f)|^2 H_m(k) \quad (15)$$

Where, $H_m(k)$ is a filter; k represents the k-th filter; K represents the number of filters.

- *Step* 5: Meier frequency cepstrum $V_{mel}(n)$ can be obtained by discrete cosine transform on Meier scale spectrum [1, 17], The calculation process is shown in Equation (16):

$$V_{mel}(n) = \sum_{k=1}^{K} \vartheta(M_k) \cos[n(k) - 0.5\frac{\pi}{K}] \quad (16)$$

3.2. Speech Recognition Based on AI Technology

This study is based on the CDHMM [2] and SOFM [20] in the field of artificial intelligence to achieve fuzzy speech recognition. The specific process is shown in Figure 4.

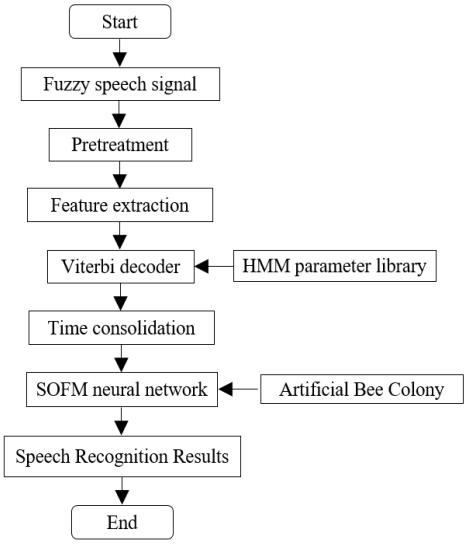


Figure 4. Flow chart of fuzzy speech recognition.

3.2.1. Analysis and Application of CDHMM

In conventional Hidden Markov Model (HMM), states are usually discrete and transitioning from one state to another is sudden. However, in the real world, many processes are continuous rather than discrete, so traditional HMM may not be able to capture the dynamic characteristics of these processes. CDHMM is an extension of Hidden Markov Model, which allows for continuous and non-discrete changes between states. It extends HMM by introducing continuous states and corresponding continuous observations. In CDHMM, the state is no longer discrete, but can change within a continuous range. These states then generate continuous observations, which can be time series data or other types of continuous data.

The structure of CDHMM is shown in Figure 5.

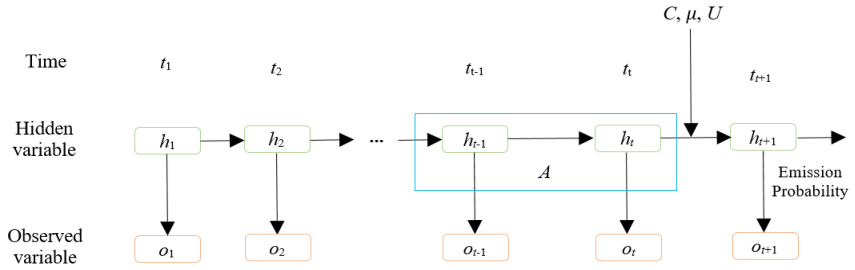


Figure 5. Structural diagram of CDHMM.

The parameters of CDHMM include the following: A is the probability of state transition; C represents the mixed gain, μ represents the mean of the mixed components, and U represents the variance of the mixed components, all of which are mixed Gaussian distribution parameters of state observation probability; t is the number of states.

The probability of state transition determines the possibility of mutual transition between different states. For speech recognition, if the probability of state transition between two phonemes is high, the likelihood of transitioning from one phoneme to another is greater. The probability matrix of state transition needs to satisfy the stationary condition, that is, the sum of probabilities starting from a certain state and eventually returning to that state over a long period of time must be equal to 1.

a) The mixed gain determines the probability of

generating observations for each state. If the mixed gain distribution is uneven, the observation probability of certain states may be amplified or reduced. Usually, the mixed gain is estimated by maximizing the logarithmic likelihood function.

b) In a continuous hidden Markov model, the mixed component describes the Gaussian observation distribution corresponding to each state, and its variance determines the distribution characteristics of the observation values. Incorrect means and variances can lead to mismatches between the observations generated by the model and the actual observation data.

c) The number of states determines the complexity of the model. Increasing the number of states can make the model better adapt to the data, but it may also lead to overfitting. If the number of states is not selected properly, the model may not be able to accurately capture the dynamic characteristics of the data. Too many states may lead to overfitting, while too few states may lead to underfitting. In this study, the optimal number of states was selected based on cross validation and set to 200.

The continuous hidden Markov model is trained, and the parameters are continuously adjusted until convergence, and then the fuzzy speech is recognized. The speech recognition system of CDHMM has a total of 10 models. The corresponding mathematical models are "0-19" in order. After the speech signal is processed by the front end, the 12-dimensional MFCC feature vector is extracted. EM algorithm is used to train its model parameters, and Viterbi algorithm is used to identify it. The training process of Markov model is shown in Figure 6.

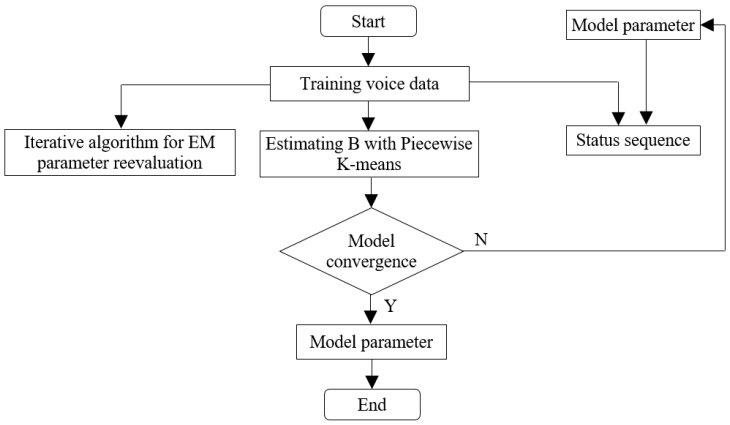


Figure 6. Markov model training process.

- *Step* 1: Initialize parameters.

The initialization of the model includes setting t and M values and setting an initial value for A, C, μ and U. There is no clear rule for selecting the number of states t. One way is to make the number of states roughly correspond to the number of phonemes in the word; Another method is to make the number of states roughly equal to a fraction of the average number of frames of a word. For Chinese single character recognition, the state is generally selected between 4 and 8. For isolated word speech recognition with multiple word lengths or different word lengths, the number of states should be compromised. When the length of each word in the word list is different, the same number of states is usually used.

For the mixed Gaussian density in continuous HMM, the number of mixtures M depends on the number of training data. As the number of parameters to be trained increases, M should also increase accordingly. The covariance matrix is a symmetric full matrix. In order to reduce the computational complexity and storage requirements, it is generally simplified to a diagonal matrix.

The setting for A is relatively simple, and can be randomly selected or uniformly selected, as long as the probability requirements are met, and the impact on recognition rate is not significant. The selection for C, μ, and U is a bit more complex. Firstly, divide all training sequences into t segments, and then use K-means algorithm to cluster each segment into M classes. The class center x of the M-th class in the t-th segment is the mean vector μ_{im} of the M-th mixture in the i -th state, and the corresponding variance vector of this class is the square difference vector U_{im} of the M-th mixture in the t-th state. The proportion of the number of observation vectors for each class in a segment is the mixing coefficient v_{im}, which can be simply expressed by an Equation:

Suppose $p_{l_{im}}$ is a certain observation vector of Class m in paragraph i, and $1 \le i \le t$, $1 \le l_{im} \le L_{im}$, $1 \le m \le M$, then there are:

$$\begin{cases} \mu_{im} = x_{im} \\ U_{im} = \dfrac{\sum_{l_{im}}^{L_{im}} (p_{l_{im}} - x_{im})(p_{l_{im}} - x_{im})^{\mathrm{T}}}{L_{im}} \\ v_{im} = L_{im} / \sum_{k=1}^{M} L_{km} \end{cases} \tag{17}$$

Where, L_{im} is a constant, and x_{im}, $p_{l_{im}}$, μ_{im} and E are U_{im}-dimensional vectors.

In addition, it should be noted that when using a continuous hidden Markov model to recognize fuzzy speech, it has the following characteristics:

1. Strictly follow the rules of jumping from left to right and single step.
2. Consider the duration of the state.
3. Gaussian mixture density function is adopted [4, 22].

3.2.2. Analysis and Application of SOFM

SOFM is a special type of self-organizing competitive neural network. The core idea is to gradually reduce the interaction neighborhood between neurons during the learning process, and enhance the activation level of central neurons according to relevant learning rules, thereby removing the lateral connections between neurons, in order to achieve the effect of simulating the real brain nervous system's "near excitation and far inhibition."

The characteristics of SOFM networks are similar to the self-organizing characteristics of the human brain, as they can map high-dimensional inputs to low dimensional spaces while maintaining the topological structure of the input space. When different samples are input from the outside, the excitation of neurons at which position in the network is random at the beginning of training. But after self-organizing training, an ordered arrangement of neurons will be formed in the competitive layer, with neurons with similar functions very close and neurons with different functions far apart. This characteristic is very similar to the organizational principle of human brain neurons. This also enables SOFM networks to be used for data dimensionality reduction and clustering analysis, maintaining the topological structure of the data when processing high-dimensional data, thereby achieving dimensionality reduction representation of the data. The self-organizing nature of this network enables it to effectively handle large-scale datasets and has good generalization performance and robustness.

The structure of SOFM is shown in Figure 7.

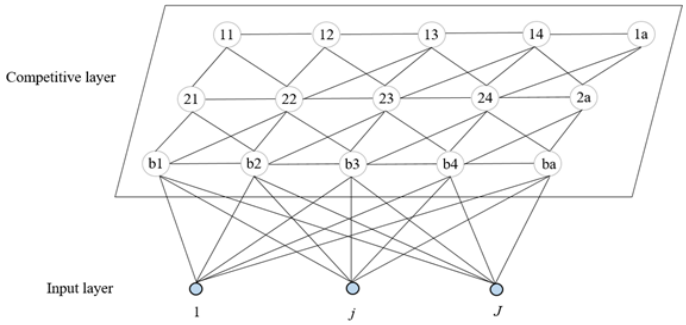


Figure 7. Structure diagram of SOFM.

In Figure 7, there are J neurons in the input layer and $b \times a$ neurons in the competition layer. The input layer is fully connected to the competition layer.

The training process of SOFM mainly includes the following steps:

1) Initialization: assign small random numbers to each weight vector in the output layer and normalize them to obtain the initial optimal neighborhood and learning rate.
2) Accept input: randomly select an input mode from the training set and normalize it.
3) Searching for winning neurons: calculate the dot product of the input pattern and the inner star weight vector, and select the winning neuron with the highest dot product from them.
4) Define the winning neighborhood: determine the weight adjustment domain at time T with the winning neuron as the center. Generally, the initial neighborhood is large, and the size of the neighborhood gradually shrinks with training time during the training process.
5) Adjust weights: adjust weights for all neurons in the winning neighborhood. In the Equation, $\eta(T, s)$ is a function of the training time T and the topological distance s between the j-th neuron and the winning neuron in the neighborhood. In the actual training process, $\eta(T, s)$ can use the monotonic descent function of T.
6) End: check if there is a concept of output error in the training of SOFM network, as it is unsupervised learning. When the training ends, the condition is whether the learning rate $\eta(T)$ decays to 0 or a predetermined positive decimal. If the end condition is not met, go back to step 2.

In practical work, in order to reduce the computational complexity of the algorithm, network pruning technology is adopted to reduce the number of layers and hidden units in SOFM. The process is as follows:

- *Step* 1: Firstly, use the original SOFM network structure for training to obtain basic speech feature representations. This initial network includes the set number of layers and the number of hidden units. Secondly, by analyzing the importance of each node in the network, identify the nodes that contribute less to the overall network performance, and remove their

connection weights to prune these nodes from the network. After pruning the nodes, retrain the pruned network using the original training data. This process can help the network adapt to new structures and parameters to improve performance.

In the process of training continuous hidden Markov models and SOFM networks, the size of the training dataset has a significant impact on the performance and generalization ability of the model. Usually, larger datasets can provide more contextual information and speech variants, which helps improve the robustness of the model. But at the same time, processing large datasets requires more computing resources and time. Therefore, during the training phase, a portion of the data can be used to train the model, and then the remaining data can be used for validation. By adjusting parameters, dataset size, and diversity, the performance changes of the model can be observed to find the optimal balance point. The selected training dataset should cover various speech patterns and environmental conditions, including different speakers (age, gender, accent, etc.,) different recording devices, different environmental noise conditions, etc. This can ensure that the model has a certain adaptability to various situations.

- *Step* 2: Less than 2: Viterbi decoding.

The third basic problem of speech recognition is time alignment. The main task of time alignment is how to align the voice observation sequence with variable length with the model according to some best principle and calculate the matching score, that is, search the best state sequence. In order to obtain the best state sequence, first train HMM to obtain HMM model parameters of each number. The training algorithm is EM algorithm. Then use this parameter and Viterbi algorithm [10] to find the best state sequence of the voice signal, that is, decoding.

The optimal state sequence is decoded by Viterbi algorithm. In order to use SOFM neural network for recognition, it is also necessary to conduct time normalization, so that the voice signals with different lengths can be normalized into feature vectors with the same dimension.

- *Step* 3: Time adjustment of voice signal in the same state.

The voice signal has a strong randomness. Even if the same person utters the same voice in the same sentence at different times, it is impossible to have the same time length. Different pronunciation habits, different environments and different moods will lead to different duration of pronunciation. Therefore, in speech recognition, it is necessary to adjust the time of speech signal first. The fuzzy speech recognition algorithm based on artificial intelligence uses front-end network for time regulation processing mainly for the following two reasons:

Because the speed of the speaker's pronunciation is inconsistent, some phonemes are pronounced faster and have a shorter duration, while others are pronounced slower and have a longer duration. In this case, the feature vectors of all frames of speech cannot be processed with equal weight. It is necessary to merge some feature vectors so that the final extracted feature vector sequence is consistent with the phoneme events in speech, but independent of the speed of pronunciation, so as to remove the distortion of the resulting speech signal.

In order to facilitate the classification and recognition of neural network classifiers, the same number of feature vectors should be extracted for different words, and the final number of segments of speech should be greater than the number of phonemes contained in each word in the vocabulary. In the practical application of isolated word recognition, the number of segments is usually 4~8 to ensure that each phoneme is described by 2-3 feature vectors [21].

The network shown in Figure 8 can better solve the problem of feature parameter time regulation in neural network speech recognition. The network extracts a set of fixed number of feature vectors from the feature vector sequence of the input speech signal, and then inputs the set of feature vectors into the neural network classifier for recognition.

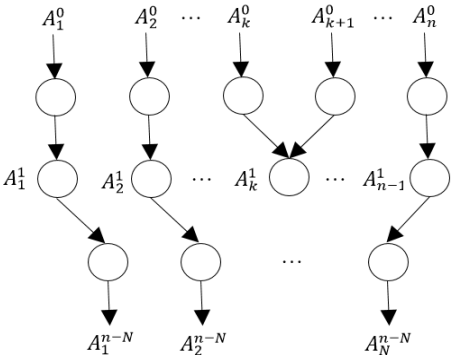


Figure 8. Time integration network structure.

The time warping algorithm is described as follows: The input layer of the network has n nodes, and n is the number of frames of the input voice signal. Each node has an input vector A_k^0(k=1, 2, …n) associated with it, and A_k^0 is the characteristic vector of the k-th frame voice signal. After entering the first layer, the two closest sequential vectors are combined with a weighted average, and the remaining vectors remain unchanged. This way, the first layer has n-1 nodes and n-1 vectors A_k^0(k=1, 2, …n-1) associated with them. After merging through n-N steps, the output layer of the final network has N nodes and N vectors A_k^0(k=1, 2, …N) associated with them. The clustering and merging process of feature vector sequences in time warping networks is

also a segmentation process of speech signals as a whole.

- *Step* 4: Speech recognition.

After the CDHMM model generates the best state sequence of the speech signal, it generates the equidimensional speech feature vector x_i after time normalization, and then classifies it with the SOFM classifier as shown in Figure 9 to obtain the recognition result y_i of fuzzy speech.

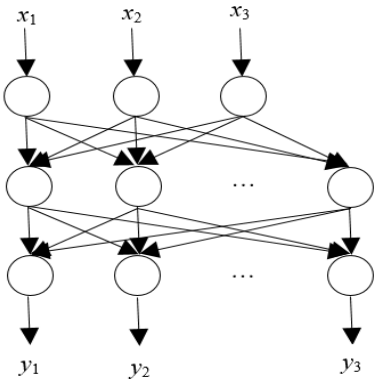


Figure 9. SOFM classifier.

SOFM realizes the mapping from high-dimensional data to low-dimensional data, which reduces the dimensionality of data features, allows for a more intuitive understanding of the structure and relationships of the data, and simplifies the model to improve computational efficiency. At the same time, SOFM recognizes and extracts patterns and laws in the data by learning the distribution and features of the data, and is able to maintain the topological structure of the relationship between the input data during the training process, which ensures the local relationship and continuity of the data. Therefore, SOFM is widely used to handle complex data analysis tasks. However, during the training process, the SOFM network structure is fixed and cannot adaptively increase or decrease neuron nodes, which limits the expressive ability and adaptability of the network. And different initial weights and learning parameters of SOFM may lead to unstable clustering results. In this regard, the study introduces the Artificial Bee Colony (ABC) [25] algorithm to improve the SOFM. On the one hand, the initial weights of the SOFM are optimized so that the SOFM can better adapt to the feature distribution of the input data. On the other hand, the learning process of SOFM is optimized to adjust the network weights to fit the input data. And the parameters such as: learning rate and neighborhood radius in SOFM network are adjusted.

ABC is an optimization-based meta-heuristic algorithm that simulates the behavior of bees foraging for food. The basic idea of ABC is to consider the search space as an environment in which bees are searching for food. The traditional ABC algorithm divides the artificial bee colony into three groups, honey gathering, following and scouting, based on the fitness values of the bees. The nectar-gathering swarm searches locally in its neighborhood to find new nectar sources based on old nectar source information and shares the information with the following bees, which then join the search process based on the shared information and select a better solution based on a certain probability. Scouting bees randomly search for valuable nectar sources in the vicinity of the hive, and if the selected nectar source is not improved within a certain period of time, then the scouting bees randomly generate a new solution vector. The mutual cooperation between the nectar-gathering, following and scouting bees can quickly find the globally optimal solution in the search space.

Set the total number of bees N_s, the honey-picking bees as N_e, the following bees as N_u, and the search space S. The calculation process of randomly generated position of honey bee population is shown in Equation (18), in which i and j denote the honey source number and its component, respectively; X_{max}^{j} and X_{min}^{j} denote the maximum and minimum values of the honey source j dimension component, respectively.

$$X_i^j = X_{\min}^j + rand(0,1)\left(X_{\max}^j - X_{\min}^j\right) \tag{18}$$

The process of calculating the adaptation value fi of the nectar source is shown in Equation (19). In Equation (19), P denotes the probability that the following bee is selected.

$$P_i = \frac{f_i}{\sum_{n=1}^{N_e} f_n} \tag{19}$$

The computation process of the position of honey picking bees is shown in Equation (20). In Equation (20), D denotes the individual vector dimension, and k and i denote random numbers. When the value of the new position adaptation is large, the honey location is updated.

$$\begin{aligned} &new\_X_i^j = \\ &X_i^j + rand[-1,1]\left(X_i^j - X_k^j\right) \\ &j \in \{1,2,...,D\} \quad k \in \{1,2,...,N_e\} \quad k \neq i \end{aligned} \tag{20}$$

When the ideal nectar source is not found when the following colony reaches the limit of the number of iterations, the honey harvesting bees near the source are converted into scout bees, and the calculation process is shown in Equation (21).

$$X_i(n) = X_{\min} + rand(0,1)\left(X_{\max} - X_{\min}\right) \tag{21}$$

4. Experiment and Result Discovery

To verify the overall effectiveness of the artificial intelligence based fuzzy speech recognition algorithm, it is necessary to conduct relevant tests on it.

The experiment adopts a comparative form to avoid the singularity of experimental results. Comparing speech recognition algorithms based on convolutional

neural networks and recurrent neural networks (algorithm of reference [18]), speech recognition algorithms based on residual networks and gated convolutional networks (algorithm of reference [19]), and speech recognition algorithms based on multi-scale Mel domain feature map extraction (algorithm of reference [23]). The denoising effect, endpoint detection, algorithm accuracy, recall rate, recognition rate, and recognition efficiency of the above algorithms were tested.

The data is sourced from the LibriSpeech dataset, which is a publicly available speech recognition dataset. To ensure the fairness of the experiment, 3000 speech segments with a duration greater than 5 seconds were randomly selected from the LibriSpeech dataset as test data. In order to simulate the noise situation in the real environment, different degrees of environmental noise were added to the speech samples before the experiment began, thus forming raw fuzzy speech signals that can be used for the experiment.

The speech recognition interface is shown in Figure 10.

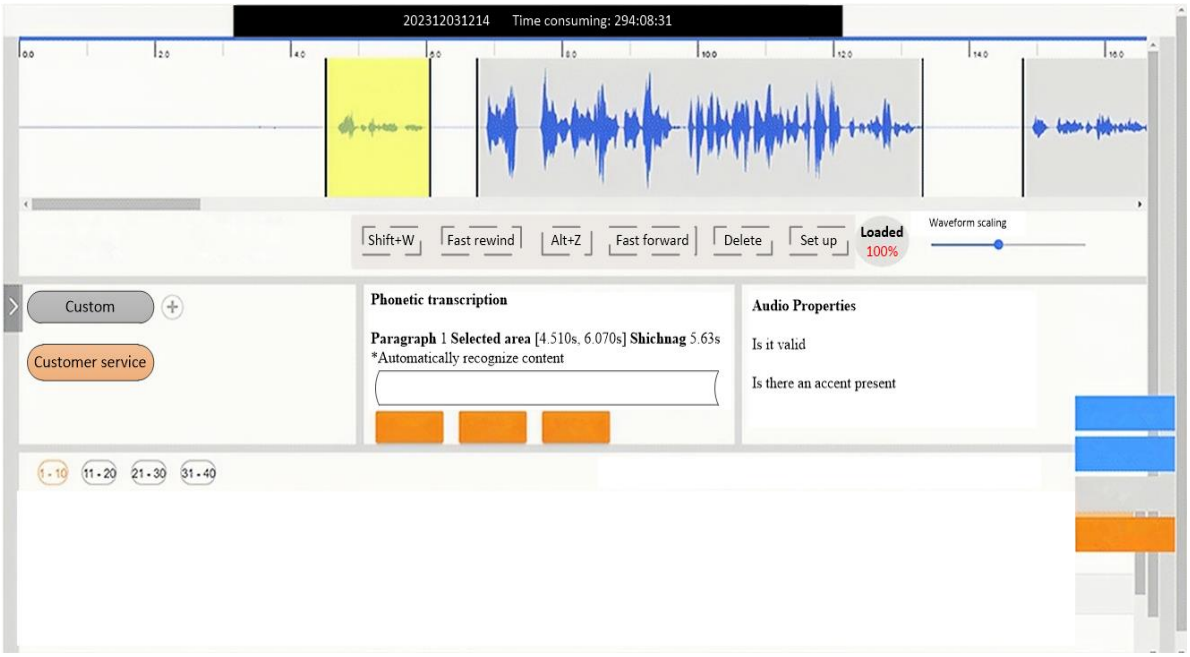


Figure 10. Speech recognition interface diagram.

4.1. Verification of Denoising Effect

Figure 11 shows the time-frequency distribution of the original fuzzy voice signal. From Figure 11, it can be seen that there is a large amount of noise in the time-frequency distribution of the original fuzzy voice signal, which is not conducive to subsequent speech recognition. The existence of noise will interfere with the speech recognition process, such as affecting the accuracy of speech feature extraction, reducing the efficiency of speech recognition, etc. In order to accurately recognize the speech signal, the original speech signal needs to be de-noised. Now, the fuzzy speech recognition algorithm based on artificial intelligence, algorithm of reference [18], reference [19] and reference [23] are used to de-noise the original fuzzy speech signal. Test the denoising effect of the algorithm by observing the distribution of noise in the time-frequency domain distribution map of the speech signal after denoising using the above algorithm. The results are shown in Figure 12.

According to Figure 12, the algorithm in this paper can effectively eliminate the noise points in the original fuzzy speech signal and avoid the interference of noise points on the speech feature extraction and speech recognition process. However, there are a large number of noise points in the time-frequency distribution of the other three algorithms, which indicates that the noise in the original fuzzy speech signal cannot be eliminated. This algorithm uses a dynamic estimation algorithm for noise power spectrum to improve the Wiener filter. The improved Wiener filter can effectively eliminate noise in speech signals and improve the quality of speech signals.

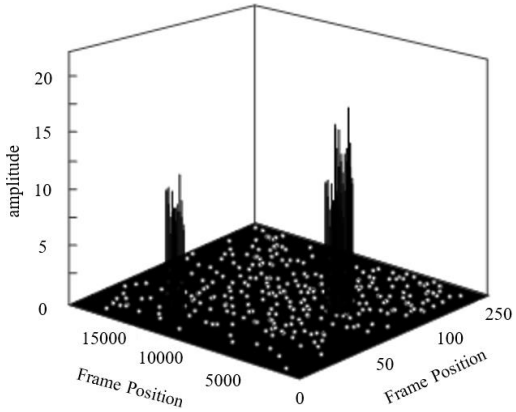


Figure 11. Time-frequency distribution of original fuzzy voice signal.

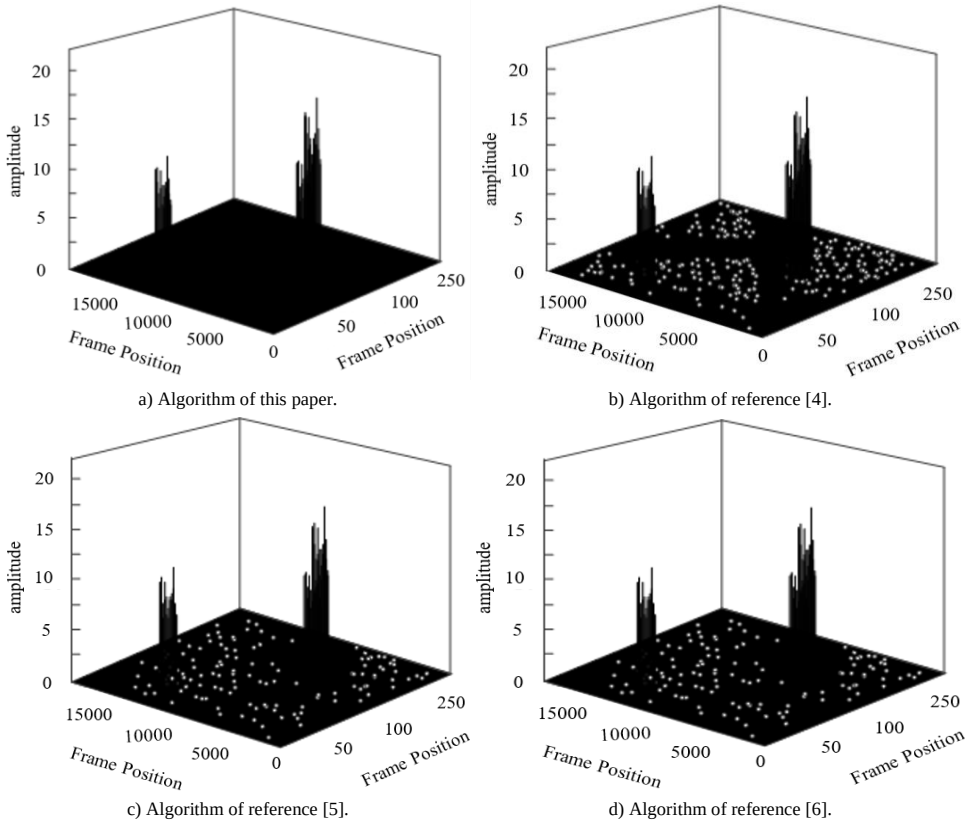


Figure 12. Noise removal results of different algorithms.

To further highlight the robustness of the algorithm in dealing with changes in background noise levels, change the intensity of environmental noise, and analyze the signal-to-noise ratio of speech signals after applying different algorithms. The signal-to-noise ratio of speech signals can be used to measure the ratio of signal power to background noise power in speech signals, which can describe the clarity and recognizability of speech signals. Speech signals with high signal-to-noise ratios sound clearer. The experimental results are shown in Table 1.

Table 1. The signal-to-noise ratio of speech signals after applying different algorithms.

The intensity of environmental noise/dB	The signal-to-noise ratio of speech signals/dB			
	Algorithm of this paper	Algorithm of reference [4]	Algorithm of reference [5]	Algorithm of reference [6]
20	98	87	72	83
25	95	86	70	80
30	90	82	68	77
35	88	79	64	72

According to Table 1, as the intensity of added environmental noise increases, the signal-to-noise ratio of speech signals decreases after applying different algorithms. After applying the algorithm of this paper, the signal-to-noise ratio of the speech signal was controlled between 88 dB-98 dB. After applying the algorithms of references [18, 19, 23], the maximum signal-to-noise values of speech signals are 87 dB, 72 dB, and 83 dB, respectively. In contrast, the algorithm proposed of this paper can maintain a high signal-to-noise ratio of speech signals, indicating that the algorithm has high robustness in dealing with changes in background noise levels.

The reason for the above results is that the algorithm of this paper improved the Wiener filter by dynamically estimating the noise power spectrum. Traditional Wiener filters typically require accurate noise estimation for signal enhancement, but in practical applications, the intensity and characteristics of noise often vary. By dynamically estimating the noise power spectrum, it is possible to more accurately estimate the noise characteristics in the current environment, thereby better eliminating the noise present in speech signals.

4.2. Endpoint Detection

Endpoint detection can effectively obtain the starting point and ending point of the voice signal. Through endpoint detection, the voice region in the signal can be obtained, and then the characteristics of the voice signal can be extracted. Figure 13 shows the denoised speech signal, which is subjected to endpoint detection using the algorithms of this paper, reference [18], reference [19] algorithm, and reference [23]. Analyze the accuracy of endpoint detection based on the results of different algorithms.

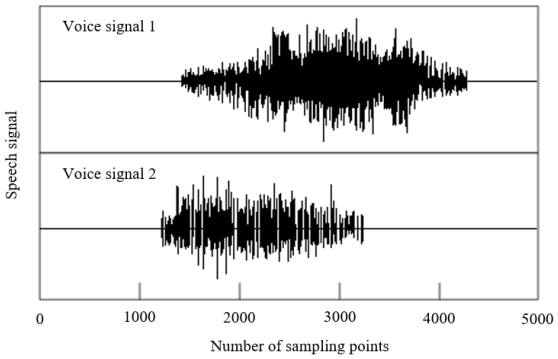


Figure 13. Voice signal.

The endpoint detection accuracy here refers to the ability of algorithms to accurately detect the start and end positions of speech signals in speech signal processing. The accuracy of endpoint detection mainly involves the following two aspects:

1. The detection accuracy of the starting position: whether the algorithm can accurately recognize the starting position of the speech signal, that is, the junction between the silent and audible segments. High quality algorithms should be able to quickly and accurately recognize this location, thus starting processing speech signals in a timely manner.
2. Detection accuracy of end position: can the algorithm accurately recognize the end position of the speech signal, that is, the junction between the speech segment and the silent segment. This is crucial for subsequent speech analysis and processing, as ending the detection of speech signals too early or too late can lead to loss of speech information or introduce additional noise.

The specific experimental results are shown in Figure 14.

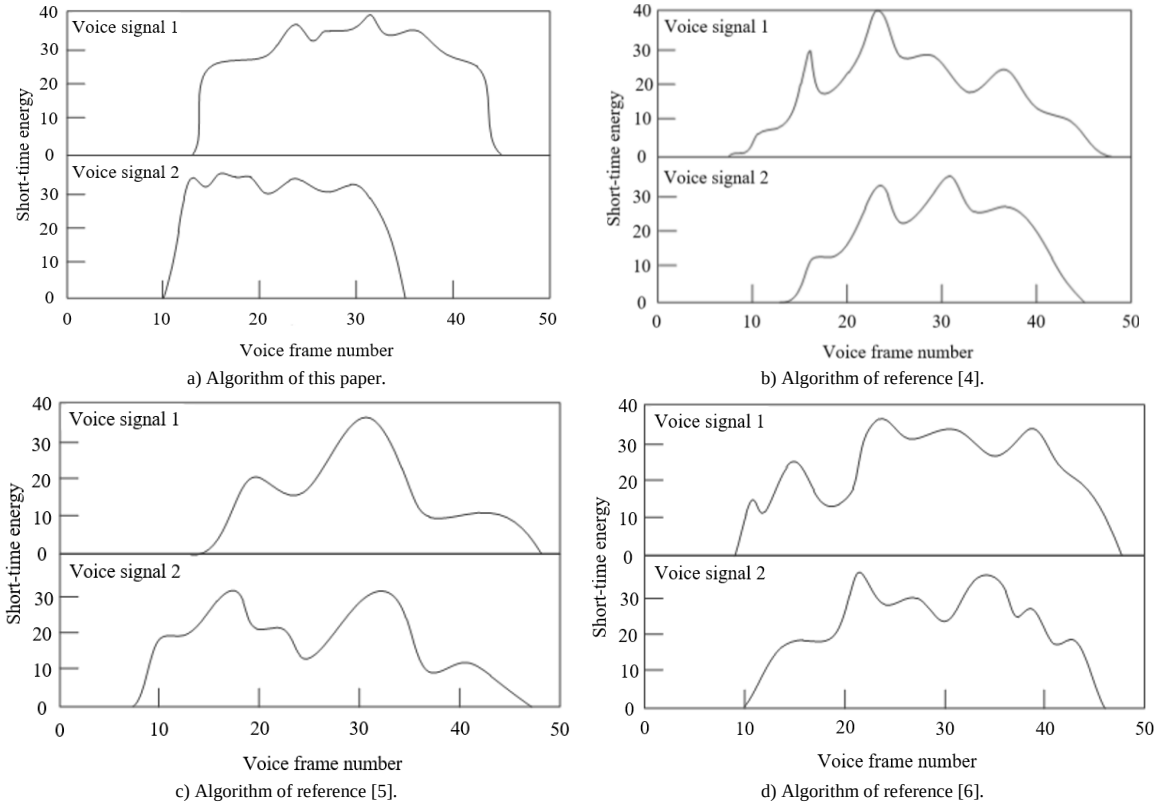


Figure 14. Endpoint detection results of different algorithms.

According to Figure 14, it can be seen that the algorithm proposed in this paper can accurately detect the audible areas in the signal, accurately distinguish the starting and ending positions of the speech signal, and achieve high accuracy in endpoint detection. However, after using other algorithms for testing, it was not possible to accurately detect the start and end positions of the audio region in the speech signal, resulting in low endpoint detection accuracy.

The reason for the above results is that the algorithm of this paper extracts Mel-frequency cepstral coefficients of speech signals as speech features, which can capture and represent the spectral characteristics of speech signals. These features can better distinguish between audible and silent parts. In speech signals, areas with sound typically contain distinct spectral features, while silent areas have fewer distinct spectral features.

4.3. Accuracy and Recall of Algorithms

The speech recognition algorithm CDHMM-SOFM algorithm designed by the research is compared with the traditional SOFM and HMM to verify the performance of the research algorithm, and the precision and recall are chosen as the evaluation indexes, and the experimental results are shown in Figure 15 Among them:

Accuracy refers to the proportion of truly correct

results identified by an algorithm, which measures the accuracy of the algorithm. The calculation process is shown in Equation (22):

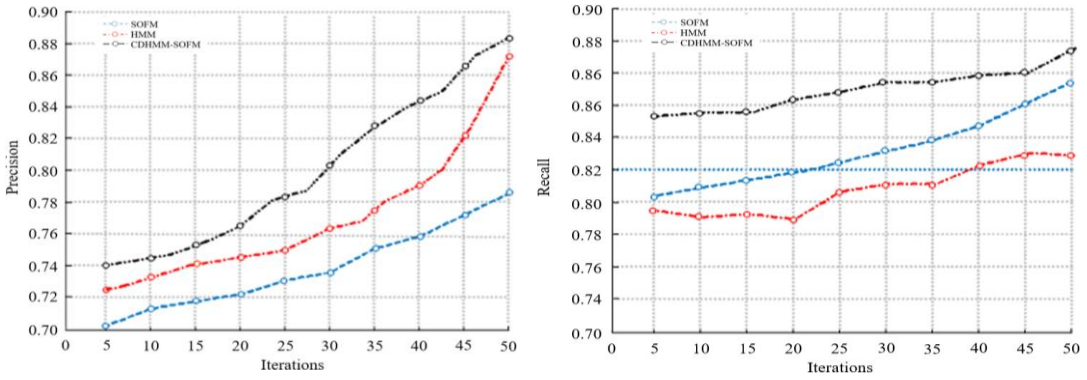
$$\text{Precision} = \frac{TP}{TP + FP} \tag{22}$$

Where, *TP* represents accuracy, which refers to the true correct results identified by the algorithm, and *FP* represents the number of samples incorrectly identified as positive by the algorithm.

The recall rate refers to the proportion of all truly correct recognition results that are correctly recognized by the algorithm, it measures the completeness of the algorithm. The calculation process is shown in Equation (23):

$$\text{Recall} = \frac{TP}{TP + FN} \tag{23}$$

Where, *FN* represents the number of samples incorrectly identified as negative by the algorithm.



a) Precision. b) Recall.

Figure 15. Precision and recall of different algorithms.

As can be seen in Figure 15, the precision rate and recall rate of the CDHMM-SOFM algorithm designed in the study increase with the number of iterations, and the difference is obvious compared with the traditional SOFM and HMM models, with a faster growth rate; and the precision rate and the recall rate take a better value, with the highest levels of 0.89 and 0.87, respectively. at the same time, the CDHMM-SOFM algorithm achieves a better balance in the pair of contradictory indexes of the precision rate and recall rate, both of which are at a high level, and this characteristic shows the superiority of the improved CDHMM-SOFM algorithm.

The reason for the above results is that the algorithm of this paper combines the advantages of CDHMM and SOFM in speech recognition. CDHMM can model temporal speech data, while SOFM can automatically learn and map the features of input data through unsupervised learning. Therefore, this combination can improve the accuracy and recall of the algorithm in classifying speech signals.

4.4. Detection of Recognition Rate and Recognition Efficiency

In order to further test the overall effectiveness of the above algorithms, under the same experimental environment, the recognition rate and recognition time are selected as the test indicators to test the recognition performance of the above algorithms. The results are shown in Figure 16. Among them, recognition rate is an indicator that measures the accuracy of the algorithm in recognizing speech signals, which represents the proportion of the speech content correctly recognized by the algorithm to the total speech content. The calculation process is shown in Equation (24):

$$\text{Recognition rate} = \frac{A_1}{A_2} \times 100\% \tag{24}$$

Where, A_1 represents the number of correctly recognized voices, and A_2 represents the total number of voices.

High recognition rate means that the algorithm can more accurately recognize the content in the speech signal, thereby improving the accuracy and availability of speech input. Recognition time is the time required for an algorithm to process input speech and return recognition results.

As shown in Figure 16, with the increase of sample size, the recognition rates of all four algorithms show a decreasing trend, and the recognition time also continues to increase. However, under the same sample size, the recognition time of algorithm of this paper is lower, with a minimum of only 8.2 seconds and a higher recognition rate of up to 98.7%, indicating that algorithm of this paper has better recognition performance.

The reason for the above results is that the Mel frequency cepstral coefficient applied in the algorithm of this paper is a commonly used speech feature extraction method, which can capture important features of speech signals. MFCC can describe the spectral characteristics of speech to a certain extent and has good robustness and discrimination. Using MFCC as a speech feature can improve the accuracy of the algorithm of this paper. In addition, the combination of CDHMM and SOFM can better classify and recognize speech, improving the accuracy of the algorithm.

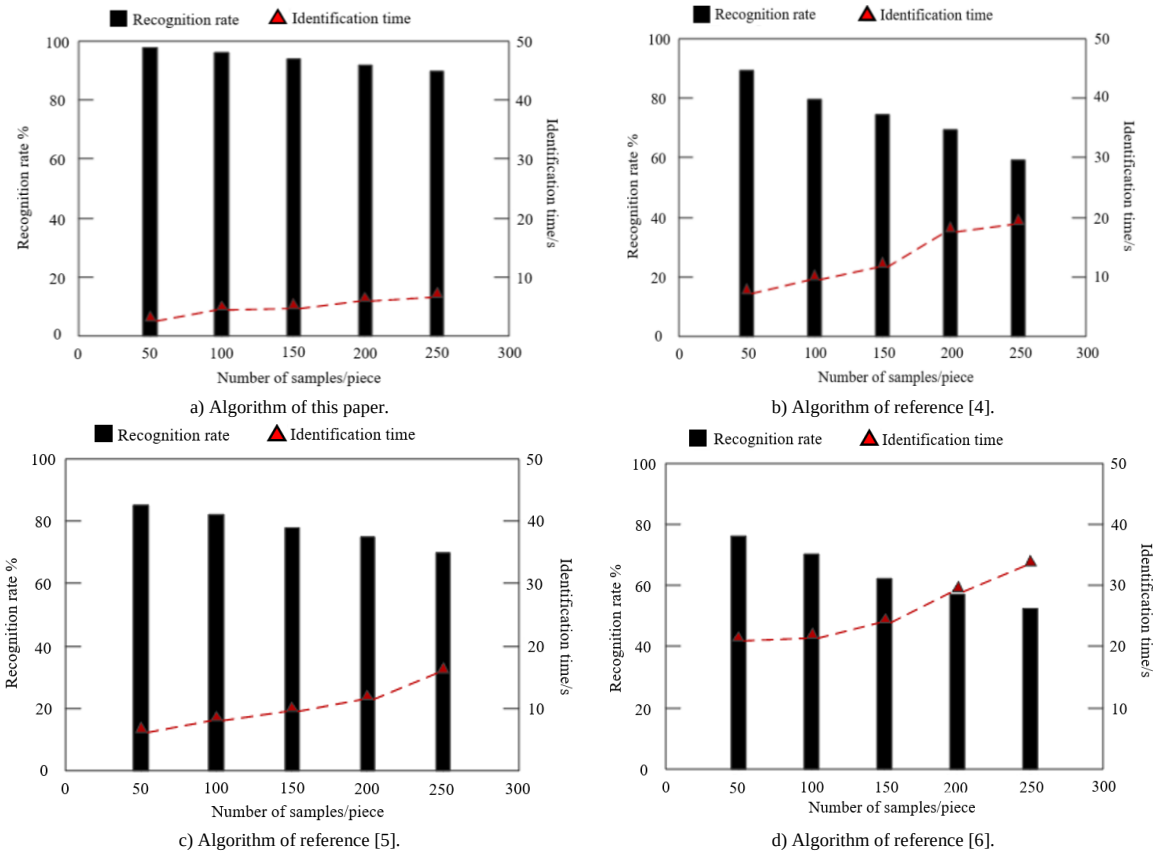


Figure 16. Recognition rate and recognition time of different algorithms.

4.5. Error Rate of User Operations

Record the number of erroneous operations performed by users during the application of different algorithms compared to the total number of operations, and obtain the Error rate of user operations by the ratio of the two. The calculation process is shown in Equation (25):

$$E = \frac{B_1}{B_2} \tag{25}$$

Where, B_1 represents the number of erroneous operations of the algorithm, and B_2 represents the total number of operations of the algorithm.

Using this as an indicator, the performance of different algorithms in terms of user interaction usability was verified, and the results are shown in Table 2.

Table 2. Error rate of user operations after applying different algorithms.

Number of tests	Error rate of user operations			
	Algorithm of this paper	Algorithm of reference [4]	Algorithm of reference [5]	Algorithm of reference [6]
20	0.0072	0.0131	0.0158	0.0097
40	0.0084	0.0144	0.0117	0.0087
60	0.0056	0.0135	0.0126	0.0014
80	0.0031	0.0095	0.0175	0.0102
100	0.0040	0.0106	0.0149	0.0198

According to Table 2, after applying the algorithm proposed in this article, the Error rate of user operations ranges from 0.0031 to 0.0084. After applying the algorithms in references [18, 19, 23], the minimum values of Error rate of user operations are 0.0095, 0.0117, and 0.0087, respectively, which are higher than the algorithms in this paper. This indicates that the algorithm in this article has better user interaction usability.

The reason for the above results is that the improved Wiener filtering algorithm, the use of MFCC as speech features, and the combination of CDHMM and SOFM algorithms can effectively reduce user operation error rates and improve system interaction usability through the comprehensive application of the algorithm of this paper.

4.6. Convergence Performance Verification

In this study, CDHMM was applied to achieve speech classification and recognition based on speech features through parameter adjustment, Viterbi decoding, and other processes. Convergence has a direct impact on the recognition efficiency of algorithms. In order to verify the convergence performance of CDHMM, it was compared with conventional HMM and conventional Markov Model, and the results are shown in Figure 17.

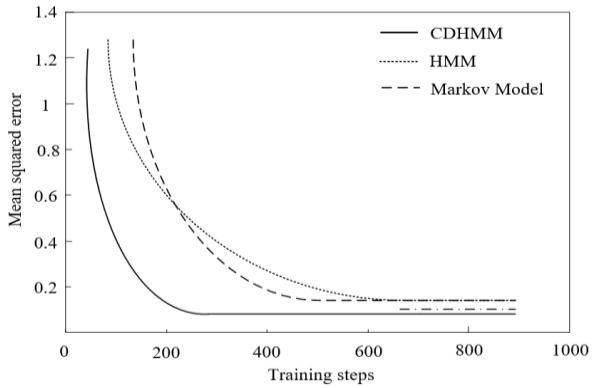


Figure 17. Analysis of convergence performance.

From Figure 17, it can be seen that CDHMM has a faster convergence speed, reaching a stable state at 200 training steps, while HMM and Markov models have slower convergence speeds. This is because CDHMM is an extension of Hidden Markov Model, which allows for continuous, non-discrete changes between states and extends HMM by introducing continuous states and corresponding continuous observations. In CDHMM, the state is no longer discrete, but can vary within a continuous range, thereby improving its convergence performance.

5. Conclusions

Speech recognition technology is a subject that uses computers to analyze speech signals, thus realizing automatic understanding of human speech. At present, the speech recognition algorithm has the problems of poor de-noising performance, low endpoint detection accuracy, low recognition rate and low recognition efficiency. A fuzzy speech recognition algorithm based on artificial intelligence is proposed. This algorithm eliminates the noise in the speech signal in the preprocessing stage, extracts the signal characteristics, and combines CDHMM and SOFM to realize speech recognition, which can effectively solve the problems in the current algorithm.

According to the experimental results, it can be concluded that:

1. This algorithm has good denoising performance. As the intensity of added environmental noise increases, the algorithm can maintain the signal-to-noise ratio of speech signals between 88dB-98dB, which can effectively improve the quality of speech signals.
2. This algorithm can accurately detect the audible areas in the signal, accurately distinguish the starting and ending positions of the speech signal, and achieve high accuracy in endpoint detection.
3. The accuracy and recall of the CDHMM-SOFM designed in the algorithm increase with the number of iterations, and the highest levels of accuracy and recall can reach 0.89, respectively, indicating the effectiveness of the application of CDHMM-SOFM.
4. The minimum recognition time of this algorithm is only 8.2 seconds, and the highest recognition rate can reach 98.7%, indicating that the algorithm has better recognition performance.
5. After applying this algorithm, the user error rate ranges from 0.0031 to 0.0084, indicating that the algorithm can effectively reduce the user error rate and improve the system's interaction availability.

In this study, Wiener filtering denoising is an important step. However, in practical work, the effectiveness of Wiener filtering may be affected by noise and signal characteristics. When there is nonlinear noise, Wiener filtering assumes that the noise is Gaussian white noise and the signal is independent of the noise. In this case, Wiener filtering may not be effective in suppressing noise. Therefore, in future research, for scenarios with nonlinear noise, nonlinear filtering methods such as wavelet transform based denoising algorithms can be considered to solve this problem.

References

[1] Abdul-Ghaffar M., Khan U., Iqbal J., Rashid N., Hamza A., Qureshi W., Tiwana M., and Izhar U., "Improving Classification Performance of Four Class FNIRS-BCI Using Mel Frequency Cepstral Coefficients (MFCC)," *Infrared Physics and Technology*, vol. 112, pp. 103589-103597, 2020. https://doi.org/10.1016/j.infrared.2020.103589

[2] Ali S. and Bouguila N., "Multimodal Action Recognition Using Variational-Based Beta-Liouville Hidden Markov Models," *IET Image Processing*, vol. 14, no. 17, pp. 4785-4794, 2020. https://doi.org/10.1049/iet-ipr.2020.0709

[3] Bhardwaj V. and Kukreja V., "Effect of Pitch Enhancement in Punjabi Children's Speech Recognition System under Disparate Acoustic Conditions," *Applied Acoustics*, vol. 177, pp. 1-7, 2021. https://doi.org/10.1016/j.apacoust.2021.107918

[4] Gao Z., Sun Z., and Liang S., "Probability Density Function for Wave Elevation Based on Gaussian Mixture Models," *Ocean Engineering*, vol. 213, no. 3, pp. 1-10, 2020. https://doi.org/10.1016/j.oceaneng.2020.107815

[5] Gurov I., Kapranova V., and Skakov P., "Dynamical Evaluation of Interference Fringe Parameters by the Wiener Adaptive Filtering Method," *Applied Optics*, vol. 60, no. 23, pp. 6799-6808, 2021. https://doi.org/10.1364/AO.428251

[6] He T., Dong C., Yuan L., and Yin H., "Motion State Classification for Micro-Drones Via Modified Mel Frequency Cepstral Coefficient and Hidden Markov Mode," *Electronics Letters*, vol. 58, no. 4, pp. 164-166, 2022. https://doi.org/10.1049/ell2.12384

[7] Li L., Watze T., Ludwig K., and Rigoll G., "Towards Constructing HMM Structure for

Speech Recognition with Deep Neural Fenonic Baseform Growing," *IEEE Access*, vol. 9, no. 8, pp. 39098-39110, 2021. DOI:10.1109/ACCESS.2021.3064197

[8] Li Z., Ma J., Wang X., and Li X., "An Optimal Parameter Selection Method for MOMEDA Based on EHNR and its Spectral Entropy," *Sensors*, vol. 21, no. 2, pp. 533-541, 2021. DOI:10.3390/s21020533

[9] Lin Y., Yu M., Chen K., Jiang G., Chen F., and Peng Z., "Blind Mesh Assessment Based on Graph Spectral Entropy and Spatial Features," *Entropy*, vol. 22, no. 2, pp. 190-197, 2020. https://doi.org/10.3390/e22020190

[10] Liu S., Wang P., Zhang H., and Tu W., "Multi-Dimensional Speech Information Recognition Method of Human-Computer Interaction System," *Computer Simulation*, vol. 38, no. 12, pp. 367-370, 2021.

[11] Oane M., Mahmood M., and Popescu A., "A State-of-the-Art Review on Integral Transform Technique in Laser-Material Interaction: Fourier and Non-Fourier Heat Equations," *Materials*, vol. 14, no. 16, pp. 4733-4745, 2021. https://doi.org/10.3390/ma14164733

[12] Shy D., Chen Z., Fessler J., and He Z., "Filtered Back Projection in Compton Imaging Using a Spherical Harmonic Wiener Filter with Pixelated CdZnTe," *IEEE Transactions on Nuclear Science*, vol. 68, no. 2, pp. 211-219, 2020. DOI:10.1109/TNS.2020.3045878

[13] Sun Z. and Tang P., "Automatic Communication Error Detection Using Speech Recognition and Linguistic Analysis for Proactive Control of Loss of Separation," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2675, no. 5, pp. 1-12, 2021. https://doi.org/10.1177/0361198120983004

[14] Taufik D. and Hanafiah N., "AutoVAT: An Automated Visual Acuity Test Using Spoken Digit Recognition with Mel Frequency Cepstral Coefficients and Convolutional Neural Network," *Procedia Computer Science*, vol. 179, pp. 458-467, 2021. https://doi.org/10.1016/j.procs.2021.01.029

[15] Teyfouri N., Rabbani H., Kafieh R., and Jabbari I., "An Exact and Fast CBCT Reconstruction Via Pseudo-Polar Fourier Transform-Based Discrete Grangeat's Formula," *IEEE Transactions on Image Processing*, vol. 29, pp. 5832-5847, 2020. DOI:10.1109/TIP.2020.2985874

[16] Tonolini M., Sorensen K., Skou P., Ray C., and Engelsen S., "Prediction of α-Lactalbumin and β-Lactoglobulin Composition of Aqueous Whey Solutions Using Fourier Transform Mid-Infrared Spectroscopy and Near-Infrared Spectroscopy," *Applied Spectroscopy*, vol. 75, no. 6, pp. 718-727, 2021. DOI:10.1177/0003702820979747

[17] Tzhir H., Iqbal N., Maqbool H., Khan M., and Tahir M., "Amputee Walking Mode Recognition Based on Mel Frequency Cepstral Coefficients Using Surface Electromyography Sensor," *International Journal of Sensor Networks*, vol. 32, no. 3, pp. 139-152, 2020. https://doi.org/10.1504/IJSNET.2020.105562

[18] Wei D. and Hong L., "An Chinese Voice Recognition Technology Based on Neural Network," *Journal of Sichuan Normal University*, vol. 45, no. 1, pp. 131-135, 2022.

[19] Xuechao Z., Zhang F., Gao L., Ren X., and Hao B., "Research on Speech Recognition Based on Residual Network and Gated Convolution Network," *Computer Engineering and Applications*, vol. 58, no. 7, pp. 185-191, 2022.

[20] Yanxia Y., Pu W., Xuejin G., Huihui G., and Zeyang Q., "Optimization Learning Algorithm Based on Hybrid Bilevel Self-Organizing Radial Basis Function Neural Network," *Journal of Beijing University of Technology*, vol. 50, no. 1, pp. 38-49, 2024. DOI:10.11936/bjutxb2022020006

[21] Zarrouk E. and Benayed Y., "Hybrid SVM/HMM Model for the Arab Phonemes," *The International Arab Journal of Information Technology*, vol. 13, no. 5, pp. 45-53, 2016.

[22] Zhang Y., Yang K., and Yang Q., "Probability Density Function of Ocean Noise Based on a Variational Bayesian Gaussian Mixture Model," *The Journal of the Acoustical Society of America*, vol. 147, no. 4, pp. 2087-2097, 2020. https://doi.org/10.1121/10.0000972

[23] Zhao J., Xue P., Bai J., Shi C., Yuan B., and Shi T., "A Multiscale Feature Extraction Algorithm for Dysarthric Speech Recognition," *Journal of Biomedical Engineering*, vol. 40, no. 1, pp. 44-50, 2023. DOI:10.7507/1001-5515.202205049

[24] Zhou G., Sun L., Lu C., and Lau A., "Multi-Symbol Digital Signal Processing Techniques for Discrete Eigenvalue Transmissions Based on Nonlinear Fourier Transform," *Journal of Lightwave Technology*, vol. 39, no. 17, pp. 5459-5467, 2021. DOI:10.1109/JLT.2021.3084825

[25] Zhou X., Liu Y., Wu Y., and Guo J., "Artificial Bee Colony Algorithm Based on Multiple Information Guidance," *Acta Electronica Sinica*, vol. 52, no. 4, pp. 1349-1363, 2024. https://doi.org/10.1016/j.eswa.2024.125283

Yanning Zhang has a Bachelor's degree in Computer Application Technology from Beijing University of Technology in 2001 and a Master's degree from North China University of Technology in 2008. Her research direction is Computer Application Technology. Work experience: From 2001 to present, she has been a Teacher at Beijing Polytechnic. Academic Achievements: she has over ten academic papers published.

Lei Ma has a Bachelor's degree in Computer Application Technology from China Agricultural University in 2002, Master's degree from Beihang University in 2008. Her research direction is Computer Application Technology. Work Experience 2002 to present, she has been a Teacher at Beijing Polytechnic. academic Achievements: She has over ten academic papers published.

Yunwei Li has a Bachelor's degree from Capital Normal University in 2004, Master's degree from China University of Mining and Technology (Beijing) in 2007, research direction: Data Mining. Work experience: From July 2007 to August 2015, Beijing Electronic Technology Vocational College from September 2015 to the present, Beijing Youth University for Political Sciences. Academic information: published 10 academic papers, published 3 academic works, presided over 7 scientific research projects.