Research on a Deep Learning Method for Speech Recognition

Jia Xiao and Sun Xiaolin

Abstract—**Deep convolutional neural network (CNN) has been widely used in speech recognition technology. The model based on deep CNN can effectively improve the quality of human-computer interaction. However, the existing CNN with fixed convolutional kernel size has a disadvantage on extracting data features. It is hard to effectively identify whether the extracted features sufficient or not. As a result, a self-tuning convolutional kernel (STCK) algorithm is proposed to solve the mentioned problem. Firstly, the computational process of STCK algorithm is derived. Then the calculation formula of the convolutional kernel size is obtained. Meanwhile, Bark-spectrum is introduced to extract the spectrogram of speech signal, which is used as the CNN input to adapt to the human hearing. In addition, the data enhancement strategies are proposed, namely frame channel shielding and Bark-band channel shielding. The presented strategies can further improve the generalization ability of the recognition model. The experimental results show that, compared with another two models (the CNN model without STCK algorithm and the CNN model without the data enhancement strategy), the training loss of the proposed method is minimum. And the recognition error rates for the test samples are reduced by 3.9% and 1%, respectively.**

Index Terms—**speech recognition, convolutional neural network, self-tuning convolutional kernel, Bark-spectrum, data enhancement**

I. Introduction

SPEECH recognition technology is an effective tool to facilitate human communication, which promotes the development of human-computer interaction and improves work efficiency. Meanwhile speech recognition technology is constantly developing with the improvement of science and technology. And the application fields of speech recognition are also expanding. In the living field, speech recognition technology can allow people to interact with intelligent devices through voice [1,2], realizing remote control and command operation. In the office field, speech recognition technology can automatically convert speech into text [3,4], increasing the speed of text entry. In the driving field, speech recognition technology can help drivers find their destinations quickly and accurately [5,6], improving driving safety. In the medical field, speech recognition can help doctors enter medical records quickly and accurately [7,8], improving the efficiency of medical consultations.

The common deep learning techniques for speech recognition include convolutional neural networks (CNN) [9-11], long short-term memory (LSTM) [12] and connectionist temporal classification (CTC) based neural network [13]. Some literatures have been conducted on speech recognition. A speech recognition model is proposed combining bidirectional LSTM and conditional random field (CRF) in literature [14]. The model is divided into two parts: text error detection and correction. To fully learn the relevance of the corpus, a LSTM-DNN based speech recognition model is proposed in literature [15]. The experiment is applied on a telephone conversation corpus. The experimental results show the performance of LSTM-DNN is superior than other methods, such as feed forward neural network (FFNN) and recurrent neural network (RNN). A one-dimensional dilated convolutional neural network (DCNN) is presented for speech emotion recognition in literature [16]. The model utilizes a multi-learning strategy to parallelly extract spatial salient emotional features and learn long term contextual dependencies from the speech signals. To solve the problem of speech recognition with spoken pronunciation or dialect pronunciation, literature [17] provides a CNN model with time-delay neural network and output-gate.

The convolution kernel in CNN is commonly used to extract features from the dataset. The quality of extracted features may directly affect the performance of the model recognition. The size of the convolution kernel in CNN usually stays fixed. The fixed convolution kernel may lead to excessive extraction or insufficient extraction of data features during the model training process. Either over-extraction or under-extraction affects the computational speed and recognition accuracy. It is necessary to investigate a way in which convolution kernel can be dynamically tuned [18]. The convolution process is improved by analyzing the error frequency during density sampling in literature [19]. And the reconstruction error of the model is reduced. The size of the convolution kernel can be adjusted in literature [20]. The kernel is tuned in different convolution layers according to the network training result. The size of convolution kernel such as 5×5 and 11×11 is used to strengthen the effect of feature extraction.

According to the above research, the improved CNN with adjustable size of convolutional kernel is proposed in this paper. The proposed method changes kernel size according to the CNN updating process. Meanwhile the dynamic balance of CNN training can be maintained. The extraction efficiency of data features is enhanced to improve the recognition performance. In addition, Bark-spectrum is extracted as the data feature for model learning, which is close to the human

Manuscript received January 8, 2024; revised July 11, 2024.

Jia Xiao is a lecturer of School of Artificial Intelligence and Software Engineering, Nanyang Normal University, Nanyang, 473061, China. (e-mail: cyny@nynu.edu.cn).

Sun Xiaolin is a lecturer of School of Artificial Intelligence and Software Engineering, Nanyang Normal University, Nanyang, 473061, China. (e-mail: 2117587831@qq.com)

hearing for speech recognition. And the frame channel shielding and Bark-band channel shielding are introduced to realize speech data enhancement. The experimental results prove the validity of the proposed method.

II. Speech feature extraction

The human hearing has different sensitivities to speech waves of different frequencies. The speech spectrum obtained by Fourier transform is large. The Fourier transform uniformly extracts the speech power of the full frequency band, without considering the sensitivity of human hearing to different frequencies. As a result, the linear spectrum should be mapped onto the nonlinear spectrum for the extracted speech spectral features. The commonly used speech features include Mel-spectrum and Bark-spectrum, etc.

A. *Mel-spectrum*

According to the mechanism research of human hearing, it is found that 200Hz to 5000Hz has great impact on speech intelligibility. The frequency perception of human hearing is a nonlinear relationship. The speech with low frequency tends to mask the one with high frequency. If the bandwidth of low frequency is reduced while the bandwidth of high frequency is increased, the frequency perception can change from a nonlinear relationship to a linear one. The nonlinear perception may improve the accuracy of speech recognition. Mel-spectrum is a nonlinear mapping according to different auditory sensitivities. The frequency axis is first converted to the Mel-frequency scale. Then the inverse spectrum is transformed to obtain the inversion coefficient. The relationship between the Mel-frequency and the conventional frequency scale is as follows:

$$mel = 2595\log_{10}\left(1+\frac{f}{700}\right)=1127\log_{e}\left(1+\frac{f}{700}\right) \quad (1)$$

B. *Bark-spectrum*

The characteristic of Bark-spectrum is similar to the Mel-spectrum. The linear spectrum is mapped into a nonlinear spectrum by Bark-spectrum. The main difference is that the speech spectrum is converted into the auditory-perceptual spectrum by Bark-spectrum. 24 critical bands are divided in the audible domain of 20Hz-16kHz. And the Bark frequency group is formed with 24 different center frequencies. The relationship between the bandwidth number *k* and the frequency *f* is as follows.

$$k=\left(\frac{26.81f}{1960+f}\right)-0.53 \quad （2）$$

Comparing with Mel-spectrum, Bark-spectrum has already divided the frequency domain into 24 critical frequency bands, which can well emulate the speech content heard by the human ear. As a result, Bark-spectrum is selected to extract the speech feature for training the deep learning model of speech recognition.

III. Data enhancement

In speech recognition systems, the overfitting problem can greatly affect the speech recognition accuracy. Data enhancement is a way to enlarge the diversity of training dataset, which can effectively solve the overfitting problem. Data enhancement is to process a feature image in different ways. And one feature image is expanded to multiple feature images with different information. As a result, the training dataset is expanded and the data feature is enhanced. The traditional ways of image enhancement include rotating, panning, and flipping. However, the continuity of the speech signal may be destroyed using the mentioned traditional methods. And the recognition accuracy may not improve significantly.

Based on the above analysis, data enhancement is performed by shielding the frame channel and the Bark-band channel of the Bark-spectrum feature map. The data enhancement is applied to help the network learn useful features. The horizontal axis represents the frame channel and the vertical axis represents the Bark-band channel in the feature map.

A. *Frame Channel Shielding*

By shielding *m* consecutive frame channels, a new feature map with frame information loss is formed. The deformed feature map improves the model robust with regards to the phenomenon of frame loss. The range of shielding frame is $[m_0, m_0+m)$, where m_0 is chosen from the range of $[0, t-m)$. *t* is the maximum value of the frame. *m* is randomly selected from $(0, p]$. *p* is the maximum value of frame that can be shielded.

B. *Bark-band Channel Shielding*

Similar to frame channel shielding, Bark-band channel shielding forms the deformed feature map with Bark-band information loss. *n* consecutive Bark-band channels are shielded to improve the model robust. The range of shielding frame is $[n_0, n_0+n)$, where n_0 is chosen from the range of $[0, r-n)$. *r* is the maximum value of the Bark-band. *n* is randomly selected from $(0, q]$. *q* is the maximum value of Bark-band that can be shielded.

IV. Self-tuning convolutional kernel (STCK) algorithm

Typically, the CNN convolutional layer has multiple convolutional kernels with different sizes. And the gradient of the parameter is calculated after each model update. If the gradient of a parameter is close to the mean value of the parameter, the convolutional kernel with the current size has extracted enough data features. It may lead to repeated extraction of the same data features if the kernel size is kept unchanged. It may result in a waste of computational resources, and increase the computing time. Thus, the size of the convolution kernel should be reduced to improve the efficiency of feature extraction and the model recognition rate. If the difference between the gradient of a parameter and the mean value of the parameter is large, the convolution kernel with the current size fails to extract enough data features. It may cause insufficient data feature extraction if the kernel size is kept unchanged, resulting in a low recognition rate and underfitting. And the size of the convolution kernel should be increased to extract more data features.

To adaptively change the size of convolutional kernel based on the updating condition, the self-tuning convolutional kernel (STCK) algorithm is derived to enhance the efficiency of data feature extraction and improve the model recognition performance.

The CNN initialization is carried out. And the initial value of each convolutional kernel size is $\alpha h_{i,j}$. It is assumed that there are three different convolutional kernels α_1, α_2, and α_3 for each CNN convolutional layer. And the coefficients are set to $\alpha h_{ij} = g_{ij} \cdot g_{ij}$, i=1,2,3, respectively. The relationship among the state y, the parameter k and l of the convolutional layer is shown in equation (3).

$$y^i = y^{i-1} \times k^i + l \tag{3}$$

The gradient of parameter k in the convolutional layer is shown in equation (4).

$$\frac{\partial L(k,l)}{\partial k^i} = \frac{\partial L(k,l)}{\partial y^i}\frac{\partial y^i}{\partial k^i} = b^{i-1} \times \gamma^i \tag{4}$$

where L denotes the loss function. the gradient ∇k denotes the difference of the corresponding parameter when the network is updated. γ^i is the output of the convolutional layer. If ∇k is large, the convolutional kernel has not extracted enough features. Then the size of the convolutional kernel should be increased to extract more features with the next updating. Conversely, the size of the convolution kernel should be reduced to prevent extracting the same features. And the size of the convolution kernel should be reduced to prevent overfitting. Minkowski distance is used to determine whether the parameter ∇k is too large, Minkowski distance is shown in equation (5).

$$t = \sqrt[r]{\sum_{i=1}^{p} \left| m_i - n_i \right|^r} \tag{5}$$

where m_i and n_i are two r-dimensional variables. t is the distance between m_i and n_i. Equation (4) represents the Manhattan distance when $r = 1$, the Euclidean distance when $r = 2$, and the Chebyshev distance when $r \longrightarrow \infty$. The variables in the Minkowski distance are replaced with the relevant parameters of the convolution kernel, as shown in Equation (6), which is used to determine whether ∇k is too large.

$$t_{ij} = \sqrt[r]{\sum_{l=1}^{p/2} \left| m_l - n_{2l+1} \right|^r} \tag{6}$$

where t_{ij} denotes the Minkowski distance of the jth convolution kernel parameter of the ith layer. m_l denotes the even parameter of the convolution kernel, while n_{2l+1} denotes the odd parameter. According to the STCK algorithm, if ∇k is larger than t_{ij}, the gradient is judged to be large. Then the size of the convolution kernel should be increased. On the other hand, the size of the convolution kernel should be reduced. The ratio of ∇k and t_{ij} is used to indicate the size change of the kernel, as shown in equation (7).

$$\nabla w_{ij} = \frac{\nabla k_{ij} - t_{ij}}{t_{ij}} \tag{7}$$

The updated convolution kernel size can be calculated in equation (8)

$$w_{ij}^{'} = w_{ij} \pm \nabla w_{ij} = w_{ij} \pm \frac{\nabla k_{ij} - t_{ij}}{t_{ij}} \tag{8}$$

V. Speech recognition model

For the extracted speech features, the proposed STCK algorithm is used with data enhancement processing. STCK dynamically change the size of the convolution kernel when the network is updated. It utilizes the gradient of the parameter and the corresponding Minkowski distance. The ability of the convolution kernel to extract the features can be adaptively changed. The modeling process of speech recognition model is as follows, shown in Figure 1.

Step 1 For the training dataset, the Bark-spectrum feature mapping is extracted.

Step 2 Frame channel and Bark-band channel shielding are performed with the extracted Bark-Spectrum feature. The training dataset for speech recognition is enhanced.

Step 3 The feature map by convolution kernel is computed.

Step 4 The Minkowski distance is calculated.

Step 5 The loss function after the fully connected layer is calculated.

Step 6 The training model is updated inversely. And each parameter ∇k_{ij} is computed.

Step 7 The size between t_{ij} and ∇k_{ij} is compared.

Step 8 The convolutional kernel size $w_{ij}^{'}$ is updated.

Step 9 The size of convolution kernel is changed.

Step 10 After the model training is finished, it is used to recognize speech.

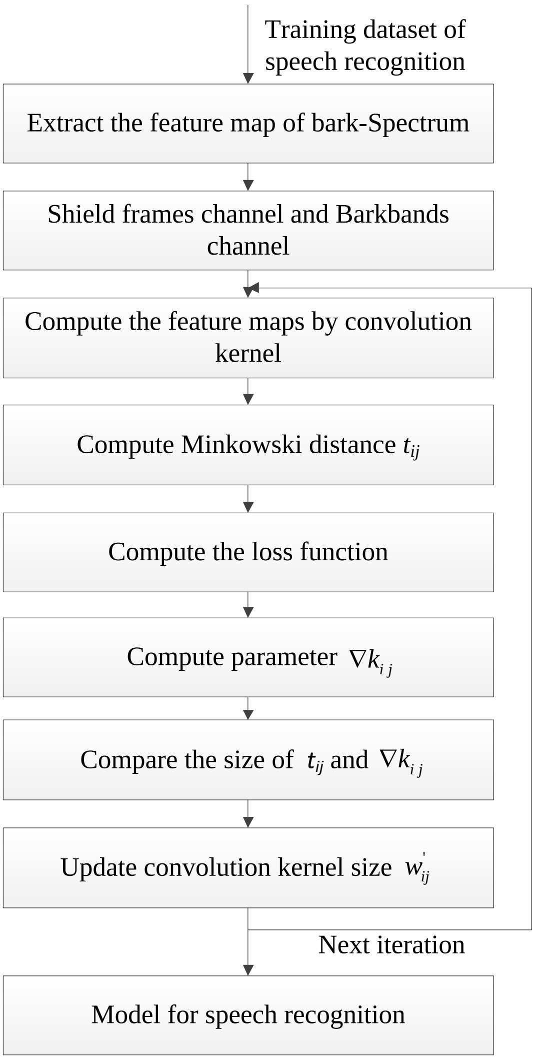

Training dataset of speech recognition
Extract the feature map of bark-Spectrum
Shield frames channel and Barkbands channel
Compute the feature maps by convolution kernel
Compute Minkowski distance t_{ij}
Compute the loss function
Compute parameter ∇k_{ij}
Compare the size of t_{ij} and ∇k_{ij}
Update convolution kernel size $w_{ij}^{'}$
Next iteration
Model for speech recognition

Fig. 1. Speech Recognition Process

TABLE I
Number of Training and Testing Samples

Sample	yes	no	up	down	left	right	on	off	stop	go	unknown	background noise
training sample	1800	1800	1800	1800	1800	1800	1800	1800	1800	1800	6300	3400
testing sample	321	323	303	306	286	308	321	295	331	321	871	600

VI. Experimental results and analysis

A. Dataset

The proposed model is trained and tested on Google Speech Commands Dataset. The dataset consists of 64,721 recordings. These recordings contain pronunciations of 30 words recorded by thousands of different people, along with background noise samples such as pink noise and white noise. Ten words were selected from the dataset, i.e., ‘yes’, ‘no’, ‘up’, ‘down’, ‘left’, ‘right’, ‘on’, ‘off’, ‘stop’, ‘go’. And two other types of data such as background noise and unknown commands are selected. The number of training and testing samples for each type of data are shown in Table 1.

B. Baseline model

To validate the effectiveness of the proposed method, the following benchmark models are selected, and the dataset is utilized for model training and testing, respectively.

Model 1: A traditional deep convolutional neural network is selected to construct the speech recognition model.

Model 2: Deep convolutional neural network with STCK algorithm is selected to construct speech recognition model. But the model is without data enhancement.

Model 3: Deep convolutional neural network with STCK algorithm and data enhancement is selected to construct the speech recognition model.

C. Experimental setup

The deep convolutional neural network parameters are set as follows: miniBatchSize = 128, InitialLearnRate = 0.0001, dropoutProb = 0.2. The model contains six convolutional layers, one dropout layer, one fully connected layer, and one softmax layer. And each layer is followed by a batch normalization layer, ReLU layer, and max-pooling layer.

D. Experiment result for Training samples

To evaluate the training result of the proposed method (Model 3) with two other methods (Model 1 and Model 2), the training accuracy of different models are compared. The curves of the prediction accuracy and loss function corresponding to the three models are shown in Figure 2 and Figure 3. The experimental results of Model 1, Model 2, and Model 3 are represented by solid lines with circle markers, dashed lines with plus markers, and dotted lines, respectively. From Figure 2, the three models gradually converge with the increase of the number of iterations. And the loss values eventually converge to a fixed range. Comparing with Model 1 and Model 2, Model 3 converges fast in the initial stage of training. The loss value of Model 1 reduces to about 0.3, while Model 2 and Model 3 can be reduced to about 0.02 and less than 0.02 in the final stage of training, respectively. It proves that Model 3 converges faster and better than Model 1 and Model 2. From Figure 3, the prediction accuracy of the three models gradually increases with the increase number of iterations. And the prediction accuracy eventually improves to between 85% and 99%. In the initial stage of training, the prediction accuracy curves of the three models show an increasing trend. The accuracies of Models 1 and Model 2 are lower than that of Model 3. In the mid-term stage of training, the prediction accuracy of Model 1 is 85% to 90%. And the prediction accuracy of Model 2 is 99% or more. The prediction accuracy of Model 1 is 85% to 90%, while the prediction accuracy of Model 2 is 99% or more. In the final stage of training, the prediction accuracy of Model 1 is between 85% to 90%. The prediction accuracy of Model 2 is more than 99%, while the prediction accuracy of Model 3 is about 99%. The experiment result verifies that Model 3 can significantly reduce the error rate of speech recognition.

E. Experiment result for testing samples

(1) Comparison of different models

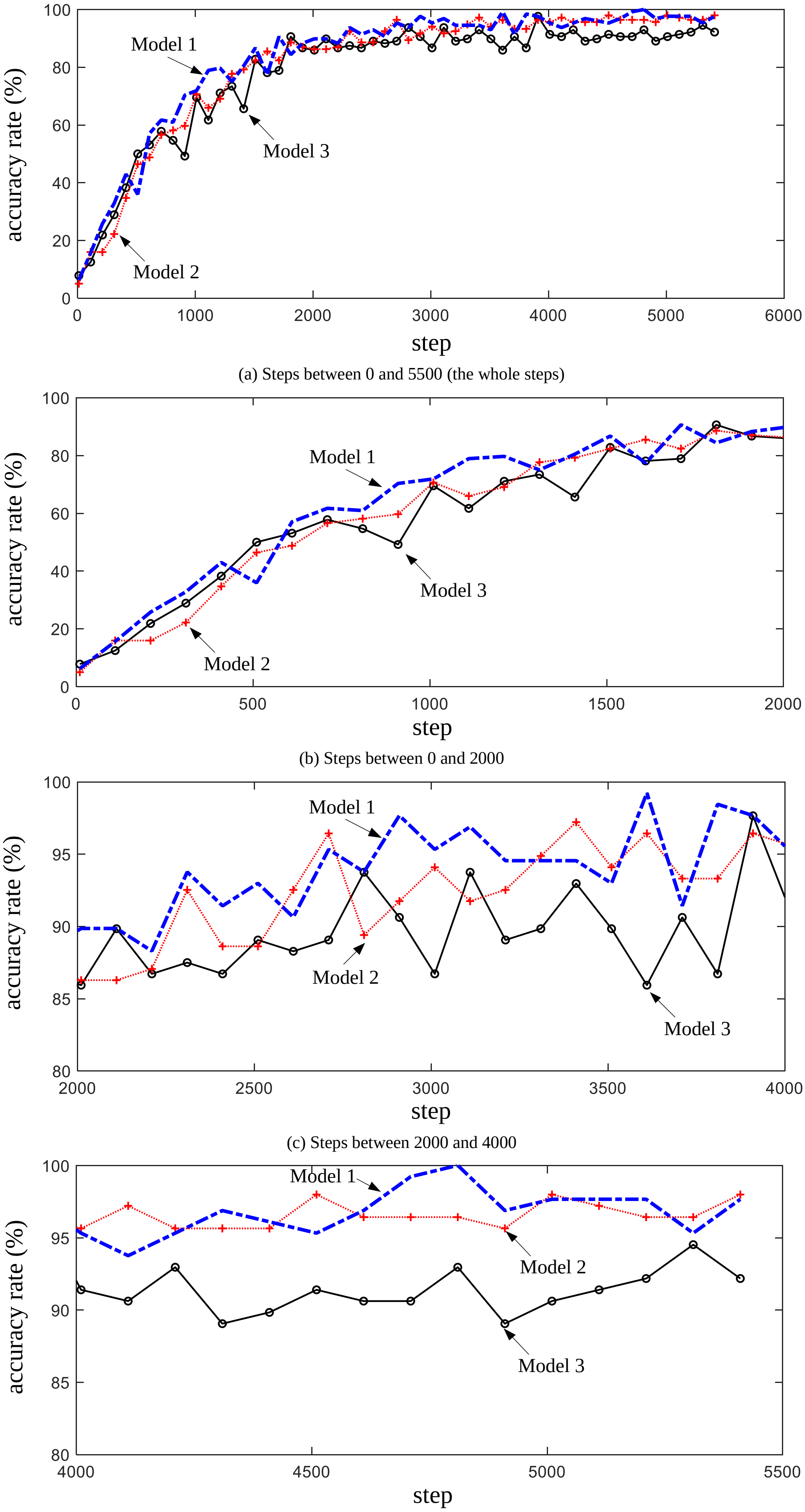
Figure 4 shows the experimental results of the testing data. It indicates that Model 3 is the optimal speech recognition model. Firstly, the training data without enhancement is used to analyze the effect of the STCK algorithm. Seen from Figure 4, the recognition accuracy of Model 2 (introducing the STCK method) is higher than that of Model 1 (the traditional CNN method). Compared with Model 1, the recognition accuracy of Model 2 is improved by 3.9%. Then the experimental results are compared between Model 2 and Model 3 (introducing the STCK method and data enhancement). Seen from Figure 4, the introduction of data enhancement can further improve the recognition accuracy. And the recognition accuracy of Model 3 is improved by 1% compared with Model 2. The final recognition accuracy of the testing sample is up to 98.1%. From Figure 4, it can be concluded that the STCK is effective in improving the recognition accuracy, which reduces the error rate by 3.9%. It verifies superiority for speech recognition. Model 3 reduces the error rate of speech recognition to 1.9%, which proves the improved algorithm is suitable for speech recognition

(2) Impact of data enhancement on experiment result

To illustrate the effectiveness of the data enhancement strategy, the following four strategies are compared based on CNN with STCK algorithm, as shown in Table 2. Strategy A indicates the original feature without using the enhancement strategy. Strategy B indicates that only the frame channel is shielded. Strategy C indicates that only the Bark-band channel is shielded. And strategy D indicates that the mixed shielding of frame and Bark-band channel. The maximum frame value *p* that can be shielded is set as 15 and the maximum Bark-band value *q* that can be shielded is set as 5.

TABLE II
Parameter Settings of Different Data Enhancement Strategy

strategy	p	num_p	q	num_q
strategy A	0	0	0	0
strategy B	0	0	5	1
strategy C	15	1	0	0
strategy D	15	1	5	1



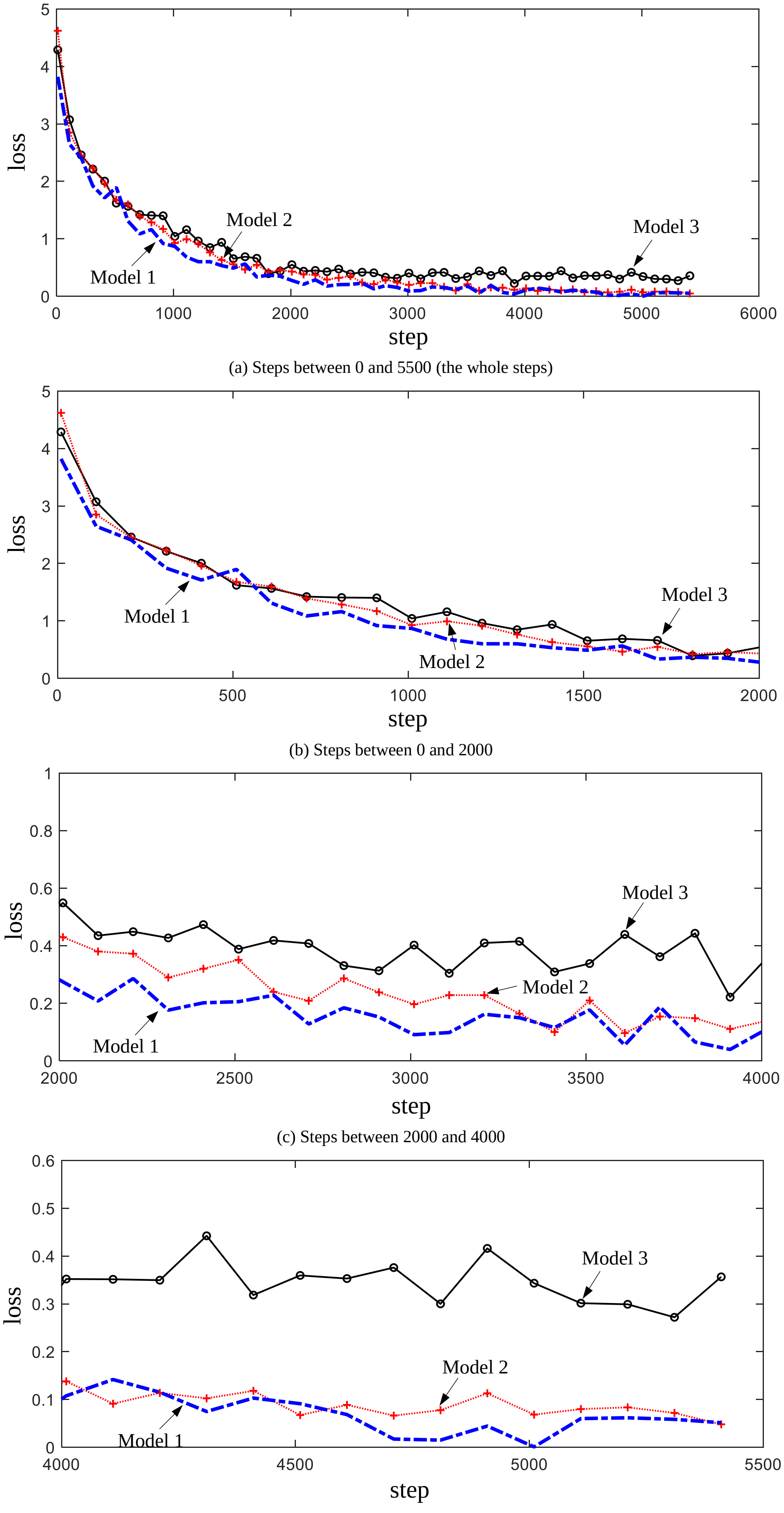
(a) Steps between 0 and 5500 (the whole steps)

(b) Steps between 0 and 2000

(c) Steps between 2000 and 4000

(d) Steps between 4000 and 5500

Fig. 2. The accuracy of the training sample



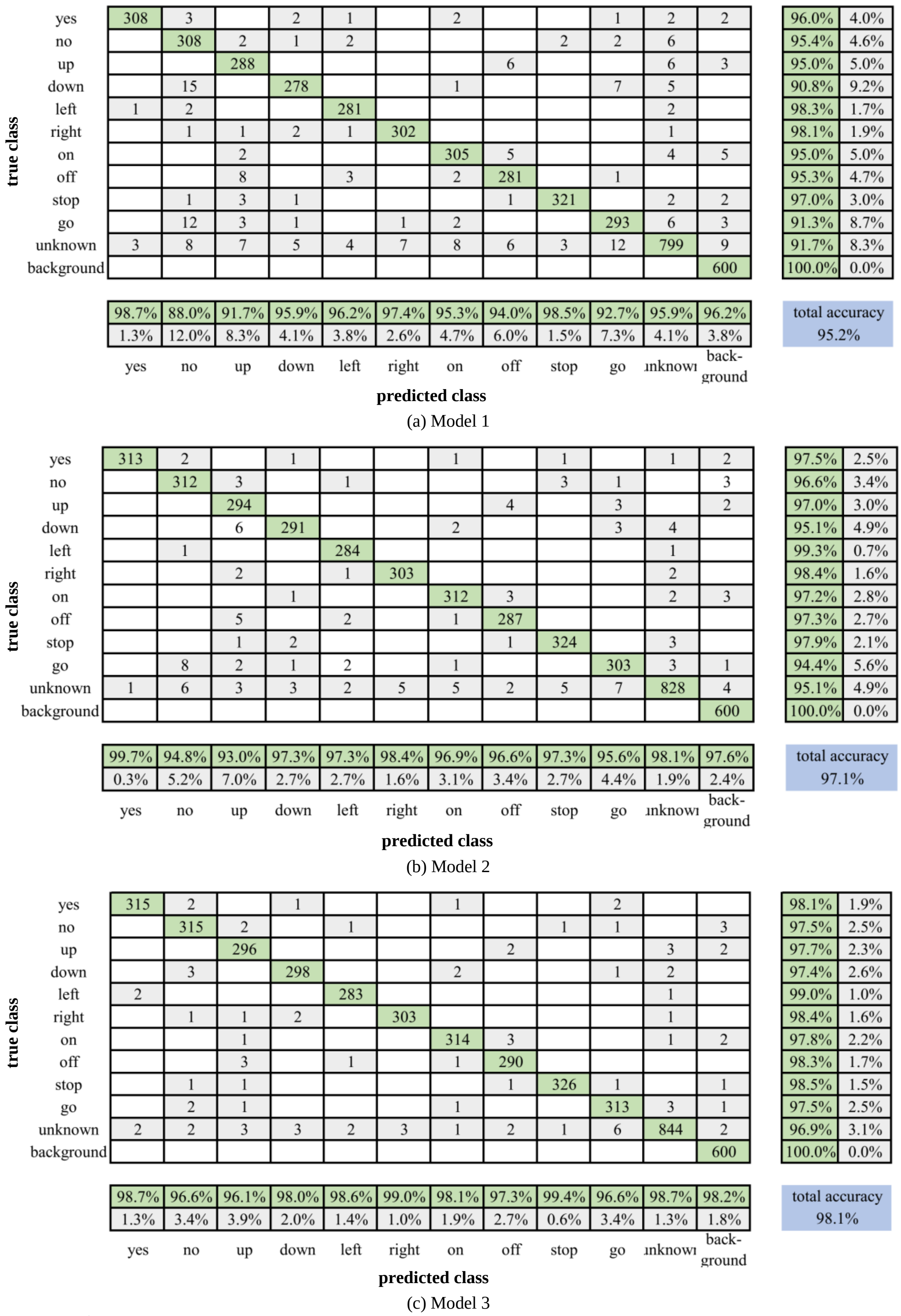
(a) Steps between 0 and 5500 (the whole steps)

(b) Steps between 0 and 2000

(c) Steps between 2000 and 4000

(d) Steps between 4000 and 5500

Fig. 3 The loss values of the training sample



true class

	yes	no	up	down	left	right	on	off	stop	go	unknown	back-ground		
yes	308	3		2	1		2			1	2	2	96.0%	4.0%
no		308	2	1	2				2	2	6		95.4%	4.6%
up			288					6			6	3	95.0%	5.0%
down		15		278			1			7	5		90.8%	9.2%
left	1	2			281						2		98.3%	1.7%
right		1	1	2	1	302					1		98.1%	1.9%
on			2				305	5			4	5	95.0%	5.0%
off			8		3		2	281		1			95.3%	4.7%
stop		1	3	1				1	321		2	2	97.0%	3.0%
go		12	3	1		1	2			293	6	3	91.3%	8.7%
unknown	3	8	7	5	4	7	8	6	3	12	799	9	91.7%	8.3%
background												600	100.0%	0.0%
	98.7%	88.0%	91.7%	95.9%	96.2%	97.4%	95.3%	94.0%	98.5%	92.7%	95.9%	96.2%	total accuracy 95.2%	
	1.3%	12.0%	8.3%	4.1%	3.8%	2.6%	4.7%	6.0%	1.5%	7.3%	4.1%	3.8%		

predicted class

(a) Model 1

true class

	yes	no	up	down	left	right	on	off	stop	go	unknown	back-ground		
yes	313	2		1			1		1		1	2	97.5%	2.5%
no		312	3		1				3	1		3	96.6%	3.4%
up			294					4		3		2	97.0%	3.0%
down			6	291			2			3	4		95.1%	4.9%
left		1			284						1		99.3%	0.7%
right			2		1	303					2		98.4%	1.6%
on				1			312	3			2	3	97.2%	2.8%
off			5		2		1	287					97.3%	2.7%
stop			1	2				1	324		3		97.9%	2.1%
go		8	2	1	2		1			303	3	1	94.4%	5.6%
unknown	1	6	3	3	2	5	5	2	5	7	828	4	95.1%	4.9%
background												600	100.0%	0.0%
	99.7%	94.8%	93.0%	97.3%	97.3%	98.4%	96.9%	96.6%	97.3%	95.6%	98.1%	97.6%	total accuracy 97.1%	
	0.3%	5.2%	7.0%	2.7%	2.7%	1.6%	3.1%	3.4%	2.7%	4.4%	1.9%	2.4%		

predicted class

(b) Model 2

true class

	yes	no	up	down	left	right	on	off	stop	go	unknown	back-ground		
yes	315	2		1			1			2			98.1%	1.9%
no		315	2		1				1	1		3	97.5%	2.5%
up			296					2			3	2	97.7%	2.3%
down		3		298			2			1	2		97.4%	2.6%
left	2				283						1		99.0%	1.0%
right		1	1	2		303					1		98.4%	1.6%
on			1				314	3			1	2	97.8%	2.2%
off			3		1		1	290					98.3%	1.7%
stop		1	1					1	326	1		1	98.5%	1.5%
go		2	1				1			313	3	1	97.5%	2.5%
unknown	2	2	3	3	2	3	1	2	1	6	844	2	96.9%	3.1%
background												600	100.0%	0.0%
	98.7%	96.6%	96.1%	98.0%	98.6%	99.0%	98.1%	97.3%	99.4%	96.6%	98.7%	98.2%	total accuracy 98.1%	
	1.3%	3.4%	3.9%	2.0%	1.4%	1.0%	1.9%	2.7%	0.6%	3.4%	1.3%	1.8%		

predicted class

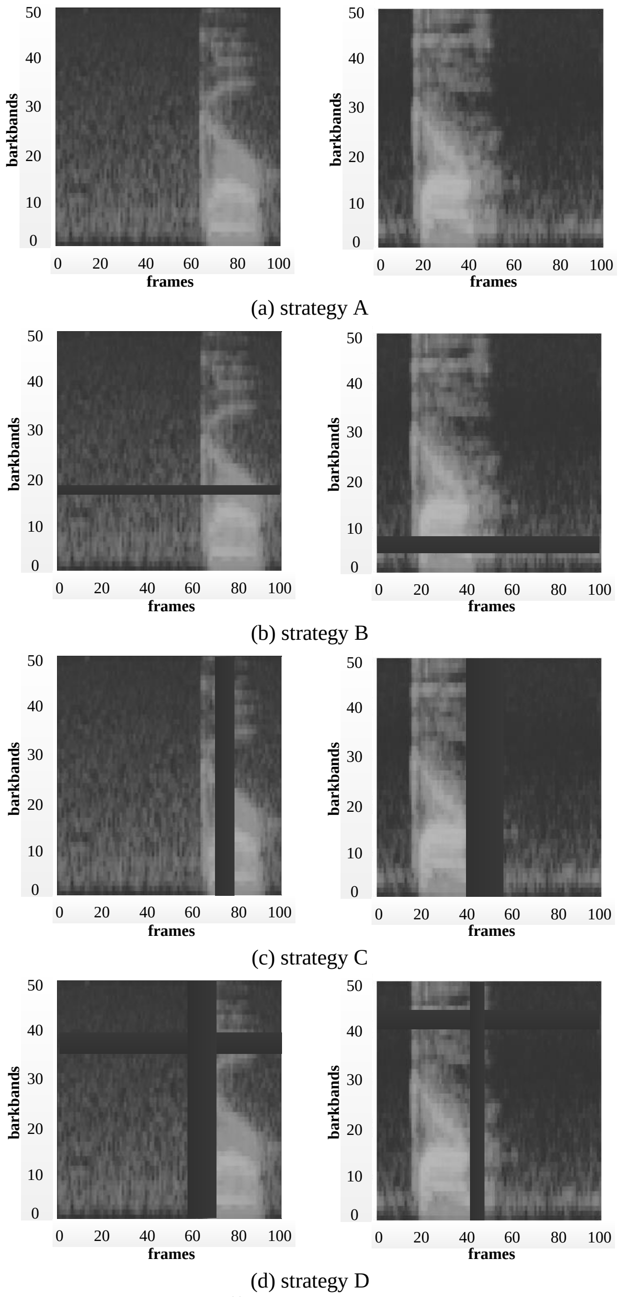
(c) Model 3

Fig. 4. Confusion matrix of testing samples

Figure 5 shows the Bark-Spectrum feature maps extracted from two different ‘go’ commands. Figure 5(a) represents the original feature map. Figure 5(b) represents the Bark-band channel shielding, in which the shielding width of the left image is 2 and that of the right image is 5. Figure 5(c) represents the frame channel shielding, in which the shielding width of the left image is 7 and that of the right image is 15. Figure 5(d) represents the shielding of both Bark-band channel and frame channel, where the left figure has a shielding width of 5 for Bark-band and 10 for frame. And the right figure has a shielding width of 4 for Bark-band and 2 for frame.

The accuracy of the training samples and testing samples corresponding to the four strategies are shown in Table 3. The accuracy of the training samples is slightly decreased after using data enhancement. But the accuracy of the testing samples is significantly improved. It indicates that the data enhancement strategy can effectively alleviate overfitting.

Among the three enhancement strategies (strategy B, C, and D), the difference between the training accuracy and the testing accuracy is the smallest for strategy D. The hybrid shielding strategy is effective in solving the overfitting problem. Data enhancement can further reduce the error rate. And the error rate of the optimal acoustic model is reduced to 1.9%.



(a) strategy A

(b) strategy B

(c) strategy C

(d) strategy D

Fig. 5. Feature map under different data enhancement strategies

TABLE III
Experimental Results of Data Enhancement Strategies

strategy	Training Sample Accuracy	Testing Sample Accuracy
strategy A	99.1 %	97.1%
strategy B	98.5%	97.8%
strategy C	98.8%	97.5%
strategy D	98.3%	98.1%

VII. Conclusion

To meet the demand of dynamic adjusting the size of the CNN convolution kernel, the STCK algorithm is proposed by combining the Minkowski distance. And the Bark-spectrum feature map of the speech signal is extracted. The extracted feature can emulate the human hearing, which is used to train the speech recognition model. To enhance the model robustness, a data enhancement method is used to shield part of the frame channel and Bark-band channel. The performance of the proposed method is compared with two other methods, i.e., the traditional CNN and CNN with STCK algorithm. Experiments on the speech dataset show that the proposed method can keep the model in a steady state. And the proposed method has the highest recognition accuracy. The accuracy of the proposed method is improved by 4.9% and 1%, respectively. It demonstrates the proposed method is a feasible model to perform speech recognition.

References

[1] Venkatesh G, Valliappan A, and Mahadeokar J, "Memory-efficient Speech Recognition on Smart Devices," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 8368-8372, 2021.

[2] Isyanto H, Arifin A S, and Suryanegara M, "Performance of Smart Personal Assistant Applications based on Speech Recognition Technology Using IoT-based Voice Commands," *International Conference on Information and Communication Technology Convergence*, pp. 640-645, 2020.

[3] Ren Y, Tan X, and Qin T, "Almost Unsupervised Text to Speech and Automatic Speech Recognition," *International Conference on Machine Learning*, pp. 5410-5419, 2019.

[4] Yang Y, Xu H, and Huang H, "Speech-text based Multi-modal Training with Bidirectional Attention for Improved Speech Recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1-5, 2023.

[5] Li K, Wang X, and Xu Y, "Lane Changing Intention Recognition based on Speech Recognition Models," *Transportation Research Part C: Emerging Technologies*, vol. 69, pp. 497-514, 2016.

[6] Ivanko D, Ryumin D, and Kashevnik A, "Visual Speech Recognition in a Driver Assistance System," *30th European Signal Processing Conference*, pp. 1131-1135, 2022.

[7] Dong F, Qian Y, and Wang T, "A Transformer-Based End-to-End Automatic Speech Recognition Algorithm," *IEEE Signal Processing Letters*, vol. 30, pp. 1592-1596, 2023.

[8] Bansal V, Raj T T, and Ravi N, "Parturition Hindi Speech Dataset for Automatic Speech Recognition," *National Conference on Communications*, pp. 1-6, 2023.

[9] Choudhary T, Bansal A, and Goyal V, "Investigation of CNN-based Acoustic Modeling for Continuous Hindi Speech Recognition," *IoT and Analytics for Sensor Networks*, pp. 425-431, 2022.

[10] Kadyrbek N, Mansurova M, and Shomanov A, "The Development of a Kazakh Speech Recognition Model Using a Convolutional Neural Network with Fixed Character Level Filters," *Big Data and Cognitive Computing*, vol. 7, no. 3, pp. 132-133, 2023.

[11] Shasha Wang, "A Face Recognition Method based on Lightweight Neural Network and Multi Hash Recognition Degree Weighting," *IAENG International Journal of Applied Mathematics*, vol. 54, no. 3, pp. 581-586, 2024.

[12] Bhaskar S, and Thasleema T M, "LSTM Model for Visual Speech Recognition Through Facial Expressions," *Multimedia Tools and Applications*, vol. 82, no. 4, pp. 5455-5472, 2023.

[13] Dingliwal S, Sunkara M, and Ronanki S, "Personalization of Ctc Speech Recognition Models," *IEEE Spoken Language Technology Workshop* pp. 302-309, 2023.

[14] Yang L, Li Y, and Wang J, "Post Text Processing of Chinese Speech Recognition Based on Bidirectional LSTM Networks and CRF," *Electronics*, vol. 8, no. 11, pp. 1248-1251, 2019.

[15] Lingyun Z, Qingqing Z, and Ta L I, "Revaluation based on LSTM-DNN Language Model in Telephone Conversation Speech Recognition," *Journal of Chongqing University of Posts and Telecommunications*, 2016.

[16] Mustaqeem, and Kwon S, "MLT-DNet: Speech Emotion Recognition using 1D Dilated CNN based on Multi-learning Trick Approach," *Expert Systems with Applications*, 2020.

[17] Qiu S, "Construction of English Speech Recognition Model by Fusing CNN and Random Deep Factorization TDNN," *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2023.

[18] Yi-Peng Shang-Guan, Cheng Xing, Jie-Sheng Wang, Yong-Cheng Sun, and Qing-Da Yang, "Multi-layer Perception Neural Network Soft-sensor Modeling of Grinding Process Based on Swarm Intelligent Optimization Algorithms," *Engineering Letters*, vol. 32, no. 3, pp. 463-476, 2024.

[19] Johnson K, and James G P, "Convolution Kernel Design and Efficient Algorithm for Sampling Density Correction," *Magnetic Resonance in Medicine*, vol. 61, no. 2, pp. 439-447, 2009.

[20] Oviedo J, Velastin S, and Branch J W, "Vehicle Detection using Alex Net and Faster R-CNN Deep Learning Models: A Comparative Study," *International Visual Informatics Conference*. pp. 3-15, 2017.

Jia Xiao (1984-) is a lecturer of Nanyang Normal University. His research direction is computer technology, computer vision, and virtual reality.

Sun Xiaolin (1986-) is a lecturer of Nanyang Normal University. Her research direction is wireless network and computer application.