ISBN: 978-981-18-1791-5
2021 the 11th International Workshop on Computer Science and Engineering (WCSE 2021)
doi: 10.18178/wcse.2021.02.002

Spoken Digit Classification: A Method Using Convolutional Neural Network and Mixed Feature

He Ba+

College of Artificial Intelligence, Nankai University, Tianjin, 300350, China

Abstract. Spoken digit recognition is one of the hot research fields of artificial intelligence. Many previous works have been done in this field, but few of them focus on recognizing a single digit and few of them would use Convolutional Neural Network (CNN). In addition, as a widely spoken language, relatively few works have been done in recognizing Chinese digits. Among the existing Chinese spoken digit recognition method, no previous works have been done using convolutional neural network and Mel Frequency Cepstral Coefficents (MFCC) feature. This paper proposes a new method that uses the mixture of short-time Fourier transform (STFT) and MFCC feature as the neural network's input and uses a convolutional neural network as a classifier. Besides, this paper, our method is applied to Chinese dataset. The new method acquires an accuracy higher than 90% in both English and Chinese dataset.

Keywords: Chinese spoken digit recognition; Deep neural network; Signal processing; MFCC; Spectrogram

1. Introduction

Converting speech signals into characters that can be understood by the computer, spoken digit recognition has been widely used in daily life and is significant in human-computer interaction. Among the tasks of recognition of spoken words, the recognition of digits is one of the most important tasks since the numbers contain a lot of valuable information compared to other words people usually speak.

Converting speech signal to text has been extensively studied. A variety of methods has been applied to them. The earliest approach was to use template matching [1]. In Automatic Speech Recognition (ASR), Recurrent Neural Network (RNN) and Long Short Term Memory Networks (LSTM) are frequently used [2, 3]. They can bridge long time lags and adapt to time-warped data. However, training RNN or LSTM can be strenuous since its architecture is complex since the input data need to be divided into subsections and then fed into the network separately. Although when it comes to recognizing a sentence, RNN and LSTM can perform significantly better, when recognizing a single number, other neural networks can also achieve good results. Artificial Neural Network (ANN) is also widely used in spoken digit recognition but its accuracy is lower than using Convolutional Neural Networks (CNN). Some state-of-the-art methods use CNN to extract features from the short-time Fourier transform (STFT) of a spoken sentence [4,5]. Besides, most of the works described before are not applied to Chinese dataset. Even among the previous works that are applied to Chinese dataset, most of them focused on recognizing a spoken sentence using ANN or Hidden Markov Model (HMM). No previous works have been done using CNN and Mel Frequency Cepstrum Coefficient (MFCC) feature to classify Chinese spoken digits.

In this paper, we use a dataset contains 2300 spoken digits from 23 different speakers. This paper proposed a new method that uses MFCC feature and STFT feature together and uses CNN to extract features and classify the digits. Besides, this paper applied this method to Chinese dataset and received a good result. The result shows that using MFCC feature is better than using STFT feature and using the mixed feature is better than using MFCC feature. Besides, using CNN will lead to a better result compared to using ANN.

+ Corresponding author.
E-mail address: Heba private@163.com

2. Proposed Method

2.1. Dataset

The dataset in this paper contains 2300 English spoken digits and 2300 Chinese spoken digits from 23 native Chinese speakers who are educated and can speak fluent English. Each one spoke each number from 0-9 ten times in English and ten times in Chinese. Among these people, 5 of them are female and 18 are male. Their ages range from 16 to 25 years. The sampling frequency is 8000HZ and all of them have the same duration. Because of the difference in each number's pronunciation, although the duration of every signal is all the same, the primary energy of the signal is distributed over different periods.

2.2. Input Feature

Our method described in this paper proposed a novel method to generate the input data of the neural network. Firstly, our method will divide the raw spoken digit signal into 25 frames and calculate the MFCC [6] feature of every frame using the tools in Matlab. In this way, each frame will get a 13-length MFCC feature. Putting these features line by line, the raw signal will be converted into an (25, 13) image.

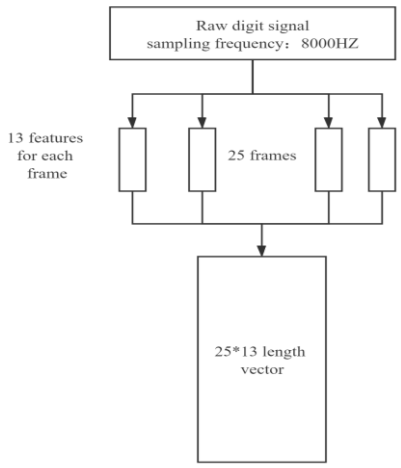

Fig. 1: Input images

Like the CNN used in MNIST Dataset, the network can extract an image's feature. The image used here is just like the image of a handwritten number used in handwritten digit recognition.

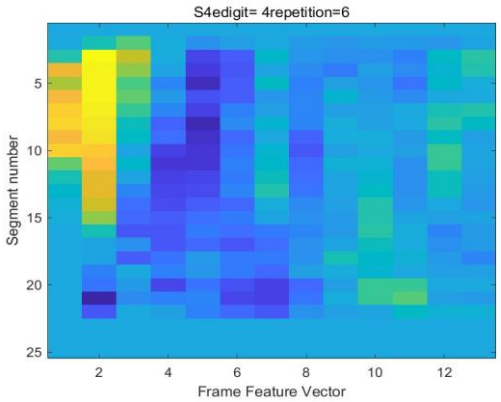

Fig. 2: MFCC digit 6 image

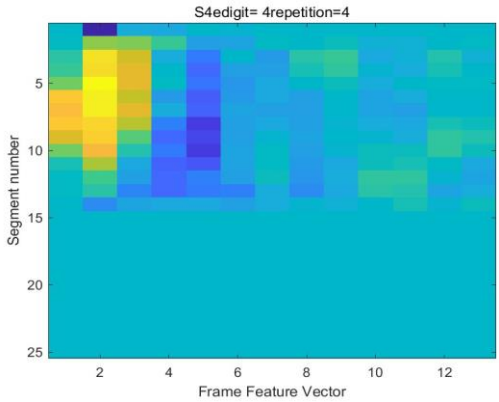

Fig. 3: MFCC digit 4 image

Figure 2 and Figure 3 are two examples of spoken digits' images. Obviously, different digit's image has a different feature. Naked eye may not tell accurately the difference among them but the CNN can extract them.

In addition to using only MFCC features, our method also divided the raw digit signal into 25 frames just as we did in calculating MFCC features. However, this time our method will calculate the STFT, also known as Spectrogram of these 25 frames. It is noteworthy that our method still uses the 13 MFCC-Scale filters to calculate the features, hence the format of the resultant image is also (25,13). Figure 4 shows the image of STFT feature.

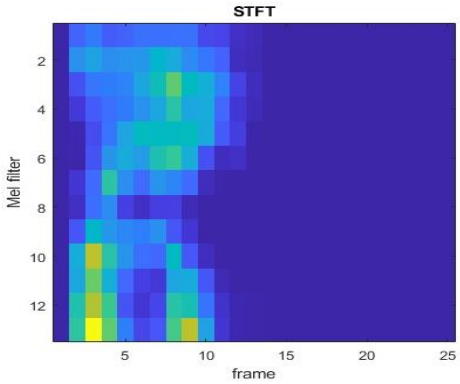

Fig. 4: STFT images

After getting these features, our method will concatenate a digit signal's MFCC feature image and its STFT feature image. In this paper, the concatenated feature is named as 'mixed feature'. The experiment shows that using the mixed feature can lead to a slightly better result. Since generating both features will not require additional information and the computational resources required in generating them are limited, the benefits it brings can outweigh its cost. Our method decided to put the two images side by side which means the resultant image is an (25,26) image. We choose not to stack the two images into an (25,13,2) image because the difference between the two images is of little use for classification.

2.3. Neural Network Architecture

This paper uses a Convolutional Neural Network as our classifier. The architecture of the CNN is shown in Figure 5. The input of the neural network is an image with 25 lines and 26 columns just like we described in the previous section. The image will pass through a convolutional layer with 32 filters whose size is (4,4) and go through a Relu activation function. Then the image will go through a max-pooling layer whose filter size is (2,2). That means the four adjacent pixels will become one pixel. Then the image will go through another convolutional layer with 64 filters whose size is (4,4) and a Relu activation function. Next, the images will go through a max-pooling layer whose architecture is identical to the previous max-pooling layer. Then the output images will go through a flatten layer whose function is to transform these images into a unidimensional vector.

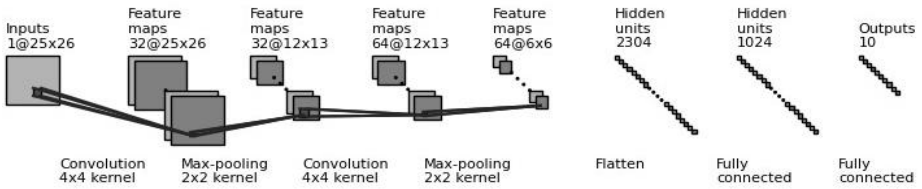

Fig. 5: Architecture of The Neural Network

Then the input data will go through a 1024-units fully connected Dense layer with a Relu activation function. Between the fully connected Dense layer and the output layer, there is a dropout layer whose function is to prevent the network from overfitting by dropping some links between the two layers. The dropout rate is 0.25 in our experiment. In the output layer, the activation function used in this paper is Softmax and the loss function is cross-entropy. The optimizer in the experiment is ADAM [7].

3. Results

Several experiments have been conducted to prove the effectiveness of our method. All the experiments used the dataset mentioned above. The data used in the experiments is the 2300 English spoken digits. The program would split the data into the training set and test set randomly. The ratio of them is 4:1, which means there are 1840 spoken digit signals for training and 460 spoken digit signals for test.

The first experiment is conducted to prove that using CNN will receive a better result than using ANN. This paper uses English dataset in this experiment. The CNN group used the architecture presented above. The ANN group used a fully connected dense neural network with two 8-units hidden layer followed with a

Relu activation function and there are dropout layers between hidden layers. Both would use softmax as loss function and batch gradient descent to optimize the loss function. Besides, both would be trained for 600 epochs. The input of the ANN group is the 325-length vector which is concatenated by all 25 MFCC vectors of length 13 and the input of the CNN group is the (25,13) MFCC image. The accuracy of the ANN group is 0.802. Its confusion matrix is as followed.

[35 0 3 1 4 0 1 0 0 0]
[0 39 1 0 4 5 0 2 1 0]
[0 2 38 6 1 0 0 0 0 2]
[0 0 0 40 0 1 0 2 2 0]
[1 2 0 0 27 0 0 2 0 0]
[0 1 0 0 8 32 1 0 0 3]
[0 0 0 0 1 0 40 2 3 0]
[0 0 3 3 3 3 4 32 1 0]
[0 0 0 1 0 0 3 0 54 0]
[1 2 0 0 1 3 0 0 1 32]

while the CNN group using MFCC feature instead of mixed feature, its accuracy is 0.922. Its confusion matrix is as followed.

[[40 0 2 0 0 0 1 1 0 0]
[0 49 1 0 2 0 0 0 0 0]
[1 0 45 1 1 0 1 0 0 0]
[0 0 0 42 0 0 1 0 1 1]
[0 1 0 1 29 0 1 0 0 0]
[0 1 0 0 0 40 2 1 0 1]
[0 0 0 0 0 0 44 1 1 0]
[0 0 1 0 0 0 3 45 0 0]
[0 0 0 1 0 0 2 0 55 0]
[0 3 0 0 0 2 0 0 0 35]]

The experiment result shows that using CNN to extract features and classify the digits is more effective than using ANN. The accuracy in CNN group is much higher.

The second experiment is conducted to prove the effectiveness of mixed feature and uses English dataset. There are three groups. Each of them uses MFCC feature, STFT feature, and mixed feature in the way described in the previous section. All these three groups have the same neural network architecture, loss function, training data, and optimization method (Batch gradient descent with the learning rate of 0.025). The results of the experiments are shown in the Figure 6 and Figure 7. Table 1 summarizes the results.

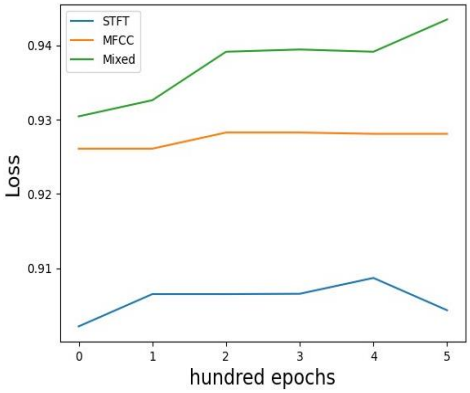

Fig. 6: Accuracy

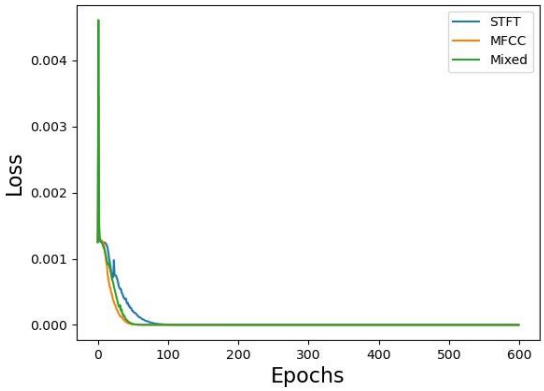

Fig. 7: Epochs

Table 1: Input Features and Results in English dataset

Input Features	Results
STFT	90.4%
MFCC	92.6%
Mixed	94.3%

From the second results, it is obvious that using mixed feature will improve the accuracy compared to using only MFCC feature or STFT feature. we assume that the additional data can provide extra features that will help the neural network to classify the digits. These two experiments showed the effectiveness of this method by proving the improvement resulted from using CNN and using the mixed feature.

The third experiment is conducted to prove our method's effectiveness in Chinese. Our method used Chinese dataset this time. There are four groups and they would use MFCC with ANN, MFCC with CNN, STFT with CNN and mixed feature with CNN. In the following experiment, they are named as ANN. MFCC, STFT and Mixed. The result is shown in Table 2.

Table 2: Input Features and Results in Chinese dataset

Input Features	Results
ANN	76.8%
STFT	79.5%
MFCC	88.6%
Mixed	90.3%

From the result, we can conclude that our method using MFCC and mixed features in Chinese dataset can have a greater promotion than the promotion we got in the second experiment using English datasets. It can also draw the conclusion that using CNN and MFCC feature in Chinese dataset can improve the accuracy a lot compared to using CNN and STFT feature or using ANN and MFCC feature. In addition, using mixing feature in Chinese dataset will also lead to an improvement compared to using just one kind of feature.

4. Discussion

The same experiments are conducted on our Chinese dataset. Although our method shows a great improvement: the best accuracy it reached is 90.3%. It is lower than the accuracy in English dataset which is 94.3%. We suspect that since these speakers are native Chinese, their pronunciation can be quite different because of the dialects. Given that the speakers are well educated in English, the English dataset does not suffer a lot from the dialect problem. More works should be done to solve the problem caused by dialects. For example, until the paper is completed, there is no Chinese spoken digit dataset that contains a variety of dialects.

5. Conclusion

Spoken digit recognition is of vital importance. It can be applied in password recognition or extracting important digits from human voice. Unlike other methods using template matching, ANN, HMM or CNN with STFT feature, our method uses CNN with MFCC and mixed feature in a unique way. The results show that our method is effective in improving accuracy. In addition, the experiments have demonstrated our method's viability in Chinese dataset, which means this work is significant in Chinese spoken digit recognition. In further study, considering the improvement that this paper has achieved is through simply concatenating the two features' images, maybe other ways of combining these different features could lead to a greater improvement. How to mix these features is of great research value.

6. Acknowledgements

The author would like to acknowledge and thank Professor Roman Kuc from Yale University for his help in providing knowledge about signal processing and providing the database used in this paper.

7. References

[1] Denes and P. (1960). Spoken digit recognition using time-frequency pattern matching. Journal of the Acoustical Society of America, 32(11):1450.

[2] Graves, A., Beringer, N., and Schmidhuber, J. (2004). A comparison between spiking and differentiable recurrent neural networks on spoken digit recognition. In Iasted International Conference on Neural Networks Computational Intelligence.

[3] Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural Networks, 18(5–6):602–610.

[4] Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Bengio, C. L. Y., and Courville, A. (2016). Towards end-to-end speech recognition with deep convolutional neural networks.

[5] Abdelhamid, O., Deng, L., and Yu, D. (2013). Exploring convolutional neural network structures and optimization techniques for speech recognition. In Interspeech.

[6] B, C, Moore, B, R, and Glasberg (1983). Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. Journal of the Acoustical Society of America.

[7] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. Computer ence.