Sparse Wavelet Decomposition and Filter Banks with CNN Deep Learning for Speech Recognition

Jingzhao Dai[1], Yaan Zhang[1], Jintao Hou[1], Xiewen Wang[1], Lizhe Tan[1], and Jean Jiang[2]
[1]Department of Electrical and Computer Engineering, Purdue University Northwest, Indiana, USA
[2]College of Technology, Purdue University Northwest, Indiana, USA
Email: dai114@pnw.edu, zhan2722@pnw.edu, hou100@pnw.edu, wang3309@pnw.edu, lizhetan@pnw.edu, jjiang@pnw.edu

***Abstract*—In this paper, the speech recognition algorithms using CNN deep learning based on the sparse discrete wavelet decomposition (SDWD) and bandpass filter banks (BPFB) are proposed. The proposed algorithms consist of three stages. First, speech signal is decomposed into sub-band signals according to the Mel filter bank frequency specification using the SDWD or BPFB. The power values from sub-bands form a feature vector for the speech frame. Cascading feature vectors for consecutive speech frames constructs a two-dimension feature image. Secondly, each obtained feature image is subject to flipping operations in order to reduce edge effect when using the standard CNN. Finally, the CNN deep learning is adopted for training and recognition. The experimental results demonstrate that our proposed SDWD-CNN and BPFB-CNN outperforms the support vector machine (SVM), K-nearest neighbors (KNN), and random forest (RF) algorithms.**

***Keywords*—Sparse discrete wavelet decomposition, Mel filter bank, filter bank, Bandpass filter banks and convolutional neural network.**

I. Introduction

Speech recognition has been widely applied in many fields such as voice driven commands, interface between human and machine as well as the text translation [1]-[8]. Speech recognition is a technique with huge prospect; and it has made Human's lives more convenient. To improve the accuracy and robust of the speech recognition, it is a consensus that employing perceptual human auditory information is efficient [1], [3]. A widely proposed method for extracting speech features uses Mel frequency cepstral coefficients (MFCC), which are derived from a warped spectrum for perceptually auditory information using the short time Fourier transform (STFT); logarithmic computation of energy values (energy vector) with the triangular filters called Mel filter banks; and then applying the discrete cosine transform (DCT) to the obtained energy vector. In the training and recognition stages, many standard algorithms are available. Among them, there are artificial neural network (ANN), support vector machine (SVM), K-nearest neighbors (KNN), and random forest (RF) [1], [2]. As the alternatives of using Mel filter banks, the discrete wavelet packet transform (DWPT) [9], sparse discrete wavelet packet transform [10], and IIR bandpass filter banks can be used to decompose the speech into sub-band signals whose power values can form the feature vector. The extracted feature vectors can naturally be used for the ANN, KNN, SVM, and RF algorithms. Recently, the convolutional neural network (CNN) has significantly been drawn an attention in the research area of artificial intelligence. The CNN network uses the two-dimensional feature images at its input layer.

It is known that the DWPT decomposition cannot match the Mel filter bank frequency specification well. Instead, the sparse discrete wavelet decomposition (SDWD) algorithm [10] can be applied for extracting sub-band signal information [11], which closely matches the Mel filter band frequency specification. However, the approximately matching frequency bands between the SDWD and the Mel filter bank can still lead to a degraded speech recognition performance. The bandpass filter bank (BPFB) with accurate magnitude frequency responses is another alternative. Each bandpass filter can be well designed with the Mel filter bank frequency specification [9]. For CNN deep learning, two-dimensional features can be formed in terms of the frequency-power-axis and frame-axis so that the constructed two-dimensional feature images can be used as the inputs to the standard CNN network [12].

This paper is organized as follows. Section II introduces the SDWD and BPFB according to the Mel filter bank frequency specification to decompose speech signal into sub-band signals. Section III presents the proposed SDWD and BPFB feature extractions for CNN. Performance evaluation for isolated word recognition (IWR) using the developed SDWD-CNN and BPFB-CNN algorithms is illustrated in Section IV. Finally, Section V presents the conclusion.

II. Sparse Discrete Wavelet Decomposition and Filter Bank with Mel Filter Bank Frequency Specification

A. Mel Filter Bank Frequency Specification

The first step of our developed algorithms is to decompose the speech into sub-band signals using the Mel filter bank frequency specification [3]. The Mel bank frequency edges can be determined by

978-1-7281-0927-5/19/$31.00 ©2019 IEEE

$$f = Mel^{-1}(m) = 700 \times \exp\left[\left(\frac{m}{1125}\right) - 1\right], \quad (1)$$

where m is the Mel scale given by

$$m = Mel(f) = 1125 \times \ln\left(1 + \frac{f}{700}\right). \quad (2)$$

Consider that the sampling rate of speech is 8000 Hz and the lowest and highest frequencies are 300 Hz and 4000 Hz, respectively, and 32 Mel filter banks are used. The Mel filter bank frequency specification can be computed as follows: using the minimum and maximum frequency values to compute the minimum and maximum Mel scales according to (2); equally dividing Mel scale range for 32 banks to obtain the sampled Mel scales; remapping the sampled Mel scales back to achieve the Mel filter bank frequency edges using (1). The obtained Mel frequency edges are listed below:

$$\begin{aligned} f = [&300\ 348.031\ 398.331\ 451.065\ 506.331\ 564.250\ 624.950... \\ &688.565\ 755.234\ 825.104\ 898.328\ 975.069\ 1055.493... \\ &1139.780\ 1228.113\ 1320.687\ 1417.706\ 1519.383... \\ &1625.942\ 1737.617\ 1854.654\ 1977.310\ 2105.855... \\ &2240.572\ 2381.757\ 2529.721\ 2684.789\ 2847.303... \\ &3017.619\ 3196.112\ 3383.176\ 3579.220\ 3784.678\ 4000]. \end{aligned} \quad (3)$$

Fig. 1 shows the Mel filter banks with their frequency edges. Note that these triangular filters are overlapped in frequency domain.

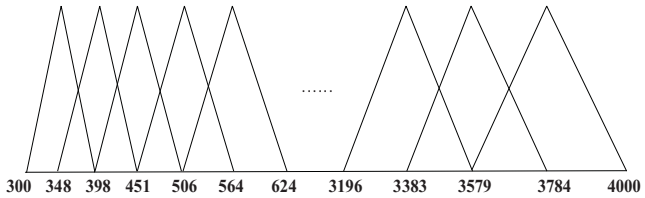


Fig. 1 32 Mel filter banks.

B. Sparse Discrete Wavelet Decomposition (SDWD)

To decompose the speech signal, a discrete wavelet packet transform (DWPT) [9], which decomposes the speech signal into high and low frequency components based on a binary dyadic tree structure, has been widely applied. As described in [9], [11], when the DWPT process begins, speech signal $s(k)$ is initially set as

$$c_{j+1}(k) = s(k). \quad (4)$$

Next, the speech signal of $c_{j+1}(k)$ at $(j+1)$ level will be decomposed to down-sampled low-pass signal $c_j(k)$ and high-pass signal $d_j(k)$ at j level:

$$c_j(k) = \sum_{m=-\infty}^{\infty} c_{j+1}(m) h_0(m-2k), \quad (5)$$

$$d_j(k) = \sum_{m=-\infty}^{\infty} c_{j+1}(m) h_1(m-2k), \quad (6)$$

where $h_0(k)$ and $h_1(k)$ are the low-pass and high-pass wavelet filters, respectively. The relation between these two filters is given by

$$h_1(k) = (-1)^k h_0(N-1-k). \quad (7)$$

For the DWPT, the decomposed high frequency signal is kept, while the low frequency signal is continuously decomposed at next level. For a 10-level decomposition for 1024 speech data samples, the obtained coefficients $\{ c_0(k) \}$, $\{ d_0(k) \}$, $\{ d_1(k) \}$,…,$\{ d_9(k) \}$ represent 11 different band signal information and the bandwidth of signal $\{ d_j(k) \}$ is doubled when j increases each time. It is obvious that they cannot match the Mel filter bands very well [11]. Thus, the performance of speech recognition using the DWPT is degraded. In addition, the computation load for decomposing is large; and a larger speech frame size is also required if we continue to decompose speech signal to obtain a fine bandwidth. Instead, the SDWD can be applied with the dedicated decomposition path, which can cover the modified Mel filter frequency bands (see Table I) as close as possible. In our investigation, the modified bank lower edge, center frequency, and upper edge are given below:

$$Mbw(i) = [(f(i-1) + f(i))/2, f(i), (f(i) + f(i+1))/2], \quad (8)$$

where $f(i)$ is the center frequency for the i-th Mel band. Table II lists all edge values. Note that each dedicated decomposition path is the sequence following the Gray code. For instance, the frequency range for band 11 (936.6985 Hz to 1015.2810 Hz in Table I) can be approximately covered by the decomposition path: LLHLLL (937.5-1000 Hz). The range for band 15 (1274.3998 Hz to 1468.5443 Hz in Table I) can be covered using two decomposition paths: LHHHLH and LHHHLLH with a combined frequency range from 1375-1468.75 Hz.

TABLE I SPARSE DISCRETE WAVELET DECOMPOSITION

Band number i	$MBW(i)$ & decomposition paths
1	[324.0064, 348.0128, 373.1719] LLLHHH
2	[373.1719, 398.3309, 424.6979] LLLHLH
3	[424.6979, 451.0649, 478.6978] LLLHLLH
4	[478.6978, 506.3308, 535.2905] LLLHLLL, LLHHLLL
5	[535.2905, 564.2501, 594.6004] LLHHLLH, LLHHLHH
6	[594.6003, 624.9504, 656.7577] LLHHLHL, LLHHHHL
7	[656.7577, 688.5650, 721.8995] LLHHHHH, LLHHHLH
8	[721.8995, 755.2339, 790.1689] LLHHHLL, LLHLHLL
9	[790.1689, 825.1038, 861.7161] LLHLHLH, LLHLHHH
10	[861.7161, 898.3284, 936.6895] LLHLHHL, LLHLLH
11	[936.6985, 975.0687, 1015.2810] LLHLLL

12	[1015.2810, 1055.4934, 1097.6365] LHHLLL, LHHLLHH
13	[1097.6365, 1139.7797, 1183.9462] LHHLLHL, LHHLHH
14	[1183.9462, 1228.1127, 1274.3998] LHHLHL, LHHHHLL
15	[1274.3998, 1320.6869, 1369.1963] LHHHHLH, LHHHHH
16	[1369.1963, 1417.7058, 1468.5443] LHHHLH, LHHHLLH
17	[1468.5443, 1519.3828, 1572.6623] LHHHLLL, LHLHLL
18	[1572.6623, 1625.9417, 1681.7792] LHLHLH, LHLHHH
19	[1681.7792, 1737.6167, 1796.1352] LHLHHL, LHLLHL
20	[1796.1352, 1854.6536, 1915.9817] LHLLHH, LHLLLHL
21	[1915.9817, 1977.3098, 2041.5824] LHLLLHH, LHLLLL, HHLLLLL
22	[2041.5824, 2105.8550, 2173.2136] HHLLLLH, HHLLLH, HHLLHH
23	[2173.2136, 2240.5721, 2311.1647] HHLLHL, HHLHHL
24	[2311.1647, 2381.7573, 2455.7393] HHLHHH, HHLHLH
25	[2455.7393, 2529.7212, 2607.2553] HHLHLL, HHHHLL, HHHHLHH
26	[2607.2553, 2684.7893, 2766.0460] HHHHLHL, HHHHH
27	[2766.0460, 2847.3026, 2932.4607] HHHLH, HHHLLH
28	[2932.4607, 3017.6187, 3106.8654] HHHLLL, HLHLL
29	[3106.8654, 3196.1121, 3289.6438] HLHLH, HLHHHLL
30	[3289.6438, 3383.1755, 3481.1979] HLHHHLH, HLHHHH, HLHHL
31	[3481.1979, 3579.2203, 3681.9491] HLLHL, HLLHHH
32	[3681.9491, 3784.6778, 3892.3389] HLLHHL, HLLLH

Note that the signal power value for each sub-band is determined by

$$P_i = \sum_{j=1}^{J} E\{x_{ij}^2(k)\}, \tag{9}$$

where $E\{\ \}$ is the expectation operator and k denotes the sample number. $x_{ij}(k)$ is the decomposed sub-band signal for band i with J paths ($0 < J \le 3$ as shown in Table I). The achieved feature vector has the following form:

$$[P_1 \quad P_2 \quad \cdots \quad P_{32}]^T. \tag{10}$$

Note that the higher the level for signal decomposition, the more possible the decomposed bands will well match the Mel filter bands. However, the high-level decomposition may require a larger speech frame. In our work, we restrict the decomposition level up to seven (7).

C. Bandpass Filter Bank Decomposition (BPFB)

Using thirty-two (32) second-order IIR bandpass filter banks (BPFB) in parallel as shown in Fig. 2, the speech signal can be decomposed into sub-band signals as

$$x(k) = x_1(k) + x_2(k) + \ldots + x_{32}(k), \tag{11}$$

where the decomposed signals of $x_1(n)$, $x_2(n)$,…, and $x_{32}(n)$ are obtained via bandpass filtering, that is,

$$x_1(n) = h_1(n) * x(n), \tag{12}$$

$$x_2(n) = h_2(n) * x(n), \tag{13}$$

$$\cdots$$

$$x_{32}(n) = h_{32}(n) * x(n), \tag{14}$$

where $h_1(n)$, $h_2(n)$, … and $h_{32}(n)$ are the impulse responses for band 1, band 2, …, and band 32, respectively, labeled as the "BPF1", "BPF2" and "BPF32" in Fig. 2.

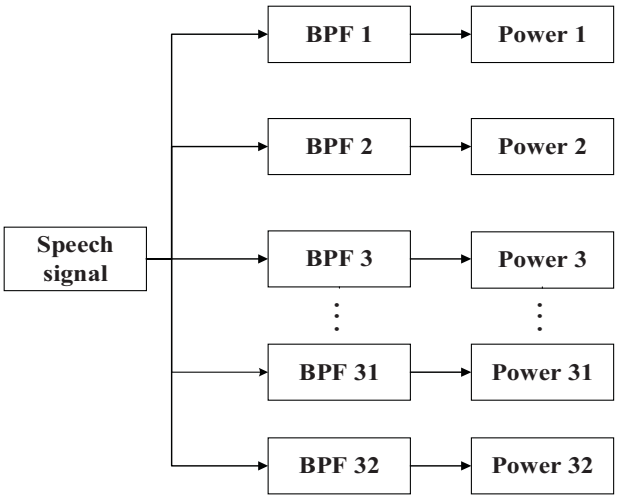


Fig. 2 Speech decomposition using the bandpass filter bank.

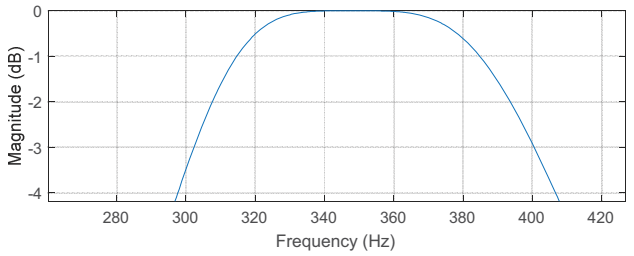
Each bandpass filter is designed according to the bilinear transformation technique [9]. The center frequency for band i is required to match the i-th Mel band center frequency, that is, $f(i)$ while its bandwidth is determined by

$$[f(i+1) - f(i-1)]/\alpha, \tag{15}$$

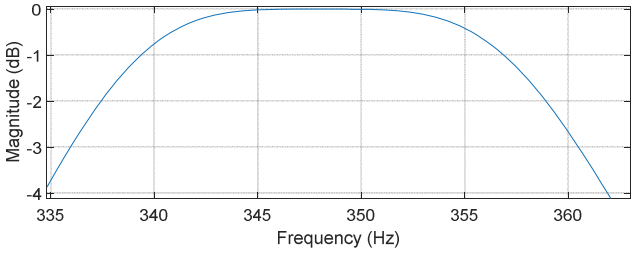
where $\alpha \ge 1$. Fig. 3 depicts the magnitude frequency responses of band 1, which has the center frequency as $f(1)$ =348.031 Hz and the bandwidth as

$$([f(2) - f(0)]/\alpha = (300 - 398.3310)/\alpha \text{ Hz}.$$

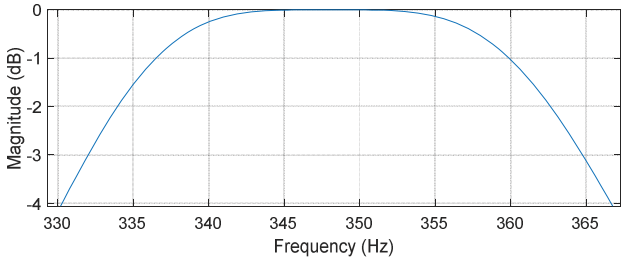
Fig. 3a shows the bandwidth when $\alpha = 1$ while Fig. 3b and Fig. 3c display the case when $\alpha = 4$ and $\alpha = 3$, respectively. Through our experiments, each bandpass filter with its bandwidth determined using $\alpha = 4$ for standard datasets and $\alpha = 3$ for recorded datasets achieve the best performance of the success recognition rate, respectively.



(a) Bandwidth (98.331 Hz) for $\alpha = 1$



(b) Bandwidth (24.5828 Hz) for $\alpha = 4$



(c) Bandwidth (32.777 Hz) for $\alpha = 3$

Fig. 3. Magnitude frequency responses for the first bandpass filter.

As shown in Fig. 2, after speech samples are processed through thirty-two (32) bandpass filters, respectively, the signal power value for band i can be calculated as

$$P_i = E\{x_i^2(n)\}, \tag{16}$$

where n denotes time index. The feature vector has the same form as shown in (10).

III. Feature Extraction Procedure

In order to apply CNN deep learning, two-dimension features (images) are generated as described in Fig. 4.

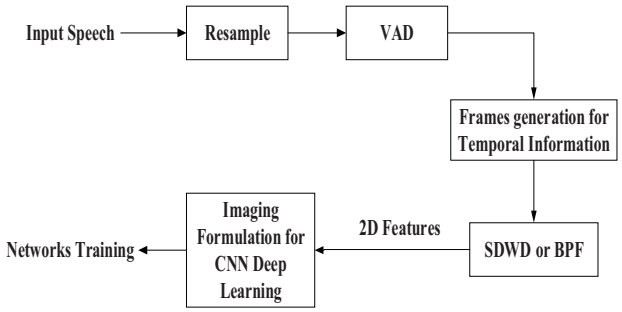


Fig. 4. Two-dimension feature extraction and image generation.

As shown in Fig. 4, the first step is speech preprocessing. The speech signal is subject to dc removal and normalization, resampling to 8 kHz; and then the voiced segment is retrieved using the voice activity detection (VAD) algorithm. Each voice segment is divided into 32 frames with 50% overlapping in time domain. Secondly, each speech frame is decomposed into 32 sub-band signals using the SDWD or BPFB. Next, 32 sub-band power values are calculated to form a power vector for each frame. Thus, by cascading power vectors for consecutive 32 speech frames, two-dimension feature (image) is generated. Finally, in the last block in Fig. 4, the two-dimension feature is further formulated with flipping operations (about horizontal, vertical, diagonal axes) in order to reduce the edge effect when applying CNN deep learning.

IV. Experiments

A. Data Sets

In our experiments, 10 digits (zero to nine) are to be recognized. The recorded datasets and standard datasets are applied, respectively. Recorded datasets are recorded by authors from two (2) females and two (2) males at a sample rate of 8kHz. Eight utterances are recorded for each digit per person. The standard recordings have a sample rate of 16 kHz from two (2) females and two (2) males [13]. Three to five utterances are chosen for each digit per person. In comparison with the recorded datasets, the standard recordings exhibit more ambient noise and various accents. The total number of utterances for both datasets is 320. The training and testing data sets (320 utterances) are randomly selected. Among them, there are 280 training data set and 40 testing data set. The recognition accuracy is measured using the percentage of the number of correct recognition over the number of the testing data (40 utterances).

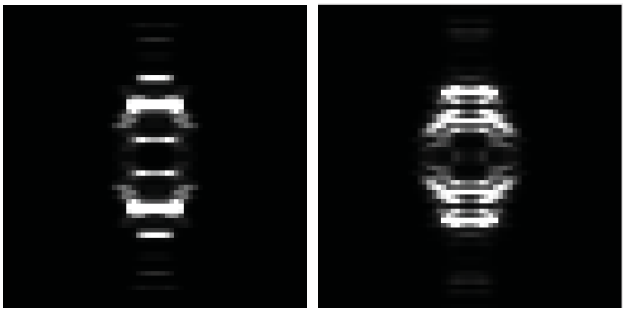
B. Speech Preprocessing

Since all standard recordings are initially sampled at 16 kHz, they are resampled to 8 kHz. After resampling, dc removal, and normalization, the voice activity detection (VAD) algorithm is applied to retrieve voiced region before speech framing [11], SDWD decomposition or BPFB decomposition, features extraction, and network training.

C. Feature Extraction

The SDWD decomposition or bandpass filter banks decompose the speech frame based on the Mel filter bank frequency specification. In our experiments and for purpose of comparisons, there are two forms for the extracted features. The first form is the one-dimension feature. Note that each SDWD path is searched via Gray code sequence (see Table I). The power values are calculated from the decomposed sub-band signals. As shown in Table I, since 32 Mel filter banks are used, the size of one-dimension features is 32×1 for each utterance. The proposed second form is a two-dimension feature for CNN. The feature image is expressed in terms of the frame-axis (horizontal axis) and frequency axis in (vertical-axis) [12]. The

sub-band power values (based on Mel filter banks) for each speech frame are calculated. The size for a two-dimension feature is 32×32. To reduce the edge effect caused by convolution and pooling stages in CNN, the two-dimension feature image (upper right corner, see Figs. 8 and 9) is filliped about the vertical, horizontal, and diagonal axes, respectively. Fig. 8 and Fig. 9 display the final feature images for "eight" from female (a) and male (b), respectively.



(a) 2D features (female,"eight") (b) 2D features(male,"eight")

Fig. 5. Two-dimension features (recorded datasets).

Note that for the SDWD decomposition, an 8-tap Daubechies wavelet filter [9], [14] is employed because of its compact support and orthogonal ability. Comparing with the other Daubechies wavelet filters with different lengths such as Daubechies 4 and 16, Daubechies 8 offers a good frequency response with the reasonable computation load.

To demonstrate the performance of the SDWD-CNN and BPFB-CNN, the support vector machine (SVM), random forest (RF), and k-nearest neighbors (KNN) methods using the one-dimension feature are included for comparisons.

V. Training and Results

A. Convolutional Neural Network

Fig. 6 describes a standard CNN structure containing the input layer, convolutional layer, pooling layer, and finally fully connected layers.

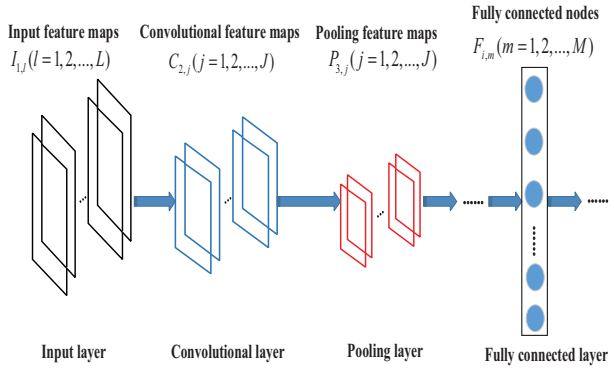


Fig. 6 CNN structure.

As shown in Fig. 6, I_{1l} represents the number of input feature maps at the input layer; F_{im} represents the number of fully connected nodes at the i-th layer.

In this paper, the layers for the SDWD-CNN or BPFB-CNN algorithms has one input layer, successive two pairs of convolutional and pooling layers, and three fully connected layers. The CNN feature maps are organized below:

Our recorded datasets training:

SDWD: $[I_{1,280}, C_{2,10}, P_{3,10}, C_{4,20}, P_{4,20}, F_{5,150}, F_{6,50}, F_{7,10}]$;

BPFB: $[I_{1,280}, C_{2,10}, P_{3,10}, C_{4,20}, P_{4,20}, F_{5,150}, F_{6,50}, F_{7,10}]$;

The standard datasets [13]:

SDWD: $[I_{1,280}, C_{2,11}, P_{3,11}, C_{4,22}, P_{4,22}, F_{5,150}, F_{6,50}, F_{7,10}]$;

BPFB: $[I_{1,280}, C_{2,10}, P_{3,10}, C_{4,20}, P_{4,20}, F_{5,150}, F_{6,50}, F_{7,10}]$.

B. Recognition Performance

Table II shows the recognition performance results for our recorded datasets. For SVM-SDWD and SVM-BPFB, the polynomial kernel functions with the orders of 2 and 3 are selected, respectively, based on their best performances. Similarly, for RF-SDWD and RF-BPFB, the numbers of 19 and 20 trees are chosen, respectively. The numbers of nearest neighbors used for KNN-SDWD and KNN-BPBF are set to 6 and 5, respectively. The VAD threshold for both SDWD and BPFB is set to 0.025 while $\alpha = 3$ is chosen to determine the bandwidth for BPFB. As shown in Table II,, the SDWD-CNN and BPFB-CNN clearly outperform the SVM, RF and KNN methods in which only one-dimensional features are used.

The additional results using the standard datasets [13] are listed in Table III. Due to large ambient noise, the VAD threshold is set to 0.03 and $\alpha = 4$ is used for BPFB. Note that only performances from the SDWD-CNN and BPFB-CNN algorithms are shown in Table III. It is clear that the BPFB-CNN algorithm achieves higher recognition accuracy than the SDWD-CNN algorithm.

TABLE II Recognition Test Accuracy Using Recorded Data Set

Method \ Accuracy		Mean accuracy	Standard deviation
CNN	SDWD	97.00%	0.0150
	BPFB	98.00%	0.0105
SVM	SDWD	92.50% (3)	0.0000
	BPFB	92.50% (2)	0.0000
RF	SDWD	98.75% (19)	1.7678
	BPFB	93.25% (20)	2.3717
KNN	SDWD	92.50% (6)	0.0000
	BPFB	92.50 (5)	0.0000

TABLE III Recognition Test Accuracy Using Standard Data Set [13]

Method \ Accuracy		Mean accuracy	Standard deviation
CNN	SDWD	86.25%	2.01560
	BPFB	92.50%	0.00069

VI. Conclusions

In this paper, we have proposed the SDWD-CNN and BPFB-CNN algorithms for isolated word recognition. Each proposed algorithm consists of three key stages. First, speech frame is decomposed into sub-bands according to the Mel filter bank frequency specification using the SDWD or BPFB. The retrieved sub-band power values formulate the feature vector for each speech frame. Cascading the feature vectors for consecutive speech frames constructs a two-dimension feature image. Secondly, each obtained the two-dimension feature image is subject to flipping operations in order to reduce the edge effect when using CNN. The CNN deep learning is finally applied for training and speech recognition. The experimental results demonstrate that our proposed SDWD-CNN and BPFB-CNN algorithms significantly outperform the SVM, KNN, and RF classifiers.

References

[1] L. Rabiner, B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Printice-Hall, 1993.

[2] X. Huang, A. Acero, H. Hon, *Spoken Language Processing: A guide to theory, algorithm, and system development*. Prentice Hall, 2001.

[3] S. Davis, P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357-366, Agust 1980.

[4] X. He, L. Deng, W. Chou, "Discriminative learning in sequential pattern recognition—A unifying review for optimization-oriented speech recognition," *IEEE Signal Process. Mag.*, vol. 25, no. 5, pp. 14–36, Sep. 2008.

[5] L. Deng, X. Li, "Machine learning paradigms for speech recognition: An overview," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 5, pp. 1060–1089, May 2013.

[6] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[7] Y. Bao, H. Jiang, L.-R. Dai, and C. Liu, "Incoherent training of deepneural networks to de-correlate bottleneck features for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 6980–6984, May 2013

[8] K. F. Lee and H. W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 37, no. 11, pp. 1641–1648, Nov. 1989.

[9] L. Tan, J. Jiang, *Digital Signal Processing: Fundamentals and Applications.* Third Edition, Elsevier/Academic Press, 2018.

[10] N. Chimitt, W. Misch, L. Tan, A. Togbe, J. Jiang, "Comparative study of simple feature extraction for single-channel EEG based classification," 2017 IEEE International Conference on Electro/Information Technology, pp. 166-170, University of Nebraska, Lincoln, Nebraska, May 2017.

[11] J. Dai, V. Vijayarajan, X. Peng, L. Tan, J. Jiang, "Speech recognition using sparse wavelet decomposition features," *2018 IEEE International Conference on Electro/Information Technology*, pp.812-816, May 2018.

[12] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, D. Yu, "Convolutional Neural Networks for Speech Recognition", *IEEE/ACM Trans. on Audio, Speech, and language Processing*, vol. 22, no. 10, pp. 1533-1545, Oct. 2014.

[13] https://www.tensorflow.org/tutorials/sequences/audio_recognition #preparation

[14] A. Akansu, P. Haddad, *Multiresolution Signal Decomposition: Transforms, Subbands, and Wavelets*, Second Edition, Academic Press, 2001.