# LD-CNN: A Lightweight Dilated Convolutional Neural Network for Environmental Sound Classification

Xiaohu Zhang，Yuexian Zou*
ADSPLAB/Shenzhen Key Laboratory for IMVR
Peking University Shenzhen Graduate School
Shenzhen, China
*{zouyx@pkusz.edu.cn}

Wenwu Wang
Centre for Vision, Speech and Signal Processing, University of Surrey, UK

***Abstract*—Environmental Sound Classification (ESC) plays a vital role in machine auditory scene perception. Deep learning based ESC methods, such as the Dilated Convolutional Neural Network (D-CNN), have achieved the state-of-art results on public datasets. However, the D-CNN ESC model size is often larger than 100MB and is only suitable for the systems with powerful GPUs, which prevent their applications in handheld devices. In this study, we take the D-CNN ESC framework and focus on reducing the model size while maintaining the ESC performance. As a result, a lightweight D-CNN (termed as LD-CNN) ESC system is developed. Our work lies on twofold. First, we propose to reduce the number of parameters in the convolution layers by factorizing a two-dimensional convolution filters ($L \times W$) to two separable one-dimensional convolution filters ($L \times 1$ and $1 \times W$). Second, we propose to replace the first fully connection layer (FCL) by a Feature Sum layer (FSL) to further reduce the number of parameters. This is motivated by our finding that the features of the environmental sounds have weak absolute locality property and a global sum operation can be applied to compress the feature map. Experiments on three public datasets (ESC50, UrbanSound8K, and CICESE) show that the proposed system offers comparable classification performance but with a much smaller model size. For example, the model size of our proposed system is about 2.05MB, which is 50 times smaller than the original D-CNN model, but at a loss of only 1%-2% classification accuracy.**

***Keywords—Environmental Sound Classification, Convolutional Neural Network, Lightweight Dilated Convolutional Neural Network, Spatial Factorization Convolution Layer, FeatureSum Layer***

## I. INTRODUCTION

Environmental Sound Classification (ESC) has become increasingly popular recently, with many potential applications, such as abnormal sound detection, human emotion estimation and robot interaction.

Deep learning based ESC systems have been proposed and achieved the outstanding classification accuracy [1, 2, 4, 5]. Typically, 1D convolution is used to extract high-level feature information from mel-spectragram automatically [1]. Results showed that 1D convolution based ESC performs much better than the traditional SVM based ESC. Dai et al. [2] presented a very deep convolutional Neural Network (CNN) ESC model (up to 34 weight layers) that directly uses raw waveforms as inputs [3], and their experimental results showed that the CNN-ESC method outperforms the traditional SVM-ESC method with human designed MFCC features by 30% accuracy. Moreover, Piczak [4] designed a shallow ESC model using 2D convolution which consists of two 2D convolutional layers with max-pooling and two fully connected layers [4] and they obtain 64.5% accuracy on the ESC50 dataset. In our previous work [5], a dilated convolution network (D-CNN) ESC model was developed in which enlarged convolution filters are applied for extracting long contextual feature information, and it surpasses the method proposed by Dai et al over 10% accuracy.

Although these existing DNN-based ESC systems achieve good classification accuracy, they normally have large model size. For illustrating purpose, the model size of several state-of-art deep neural network based ESC systems is listed in Table I. It is clear that these existing models have a size larger than 100MB, which need powerful GPUs to compute and greatly limit their applications in handheld devices.

TABLE Ⅰ. BASIC INFORMATION OF MAIN STREAM ESC SYSTEMS

| Reference | ESC method | Model size |
|---|---|---|
| 2017 ICASSP [2] | Very Deep CNNs | 128M |
| 2015 MLSP [4] | PiczakCNN | 105M |
| 2017 DSP [5] | D-CNN | 105.3M |

In this paper, we focus on the D-CNN ESC model and aim to reduce its model size while maintaining the ESC performance. As a result, we propose a Lightweight D-CNN ESC model (named as LD-CNN for short). Our contribution mainly lies on twofold. First, a spatial factorization convolution layer is used to reduce the number of parameters in the convolution layers of the D-CNN model by decomposing a two-dimensional convolution filters ($L \times W$) into two separable one-dimensional convolution filters ($L \times 1$ and $1 \times W$). Second, a FeatureSum layer is introduced to replace a fully connection layer of the D-CNN model, which is able to further reduce the number of network parameters.

1). **Spatial factorization convolution layer**: It is well-known that the size of receptive field in a CNN model significantly affects its ability in learning contextual and spatial feature

information. Generally, the size of the receptive field is dependent on the size of convolution filters. However, large convolution filters usually require large number of parameters in the CNN model. For enhancing computational efficiency, the InceptionNet v2 has been proposed in Googlenet [6], where an $N\times N$ convolution filter is decomposed into two separate convolution filters with $1\times N$ and $N\times 1$ size respectively. It is noted that two separate filters with $1\times N$ and $N\times 1$ size can be used to obtain an equal size of the receptive field with an $N\times N$ convolution filter, while reducing the number of parameters in the convolution layers. In addition, a large number of experiments in Googlenet [6] have demonstrated that this kind of factorization would not have negative effect on the distribution of the extracted features if the same size of receptive field is retained. Bearing this concept in mind, we propose to decompose a two-dimensional convolution filters ($L\times W$) into two separate one-dimensional convolution filters ($L\times 1$ and $1\times W$). As a result, the number of parameters in the convolution layer can be reduced significantly.

2). **FeatureSum layer**: It is well-known that the fully connected layers (FCL) in CNN, such as AlexNet [16] and VGG [17], are modelled with a large number of parameters. Examining the property of general environmental sounds, we found that they have weak absolute locality in the time-frequency spectrogram [7]. Hence, it may be inferred that the spatial information extracted in high-level feature maps does not contribute much to the final classification accuracy. Making use of this property, we propose a FeatureSum layer to replace the fully connected layer in the D-CNN ESC model. Specifically, a global sum operation is applied for each input feature map so that only the global statistical features of environmental sounds are preserved. The detailed design is given in Section II-B.

## II. The Proposed Method

In this section, we present the scheme of the proposed lightweight dilated convolutional neural network (LD-CNN), whose architecture is shown in Fig. 1.

This network is a two-channel system, each with 8 layers including the input and the output layer. As shown in Fig. 1, two-channels have the same model structure. For the left channel, the input is the log-mel spectrum, while for the right channel, the input is the delta spectrum. The log-mel spectrum and the delta spectrum represent the static and dynamic features of sound events, respectively. As shown in Fig. 1, these two features are separately fed into the input layer of the right and left channel in the LD-CNN network. Followed by the input layer, there is a spatial factorization convolution layer (SFCL). The SFCL contains two separable layers, the first layer has 80 convolution filters with $57\times 1$ size, while the second layer has 80 convolution filters with $1\times 6$ size. Then a max pooling layer (MPL1) is followed by the SFCL. The pooling size and stride size is $(4\times 3)$ and $(1\times 3)$ respectively. After that, a dilated convolution layer (DCL) with 80 convolution filters is used to increase the receptive field of the network. The size of these filters are $1\times 3$. Next, an 80 channels max pooling layer (MPL2) is used to generate more abstract features. The pooling size is $(1\times 3)$ and stride size is set as the same used for MPL1. Followed by the MPL2, a FeatureSum Layer (FSL) is designed to compress high dimensional feature maps. In our design, there are no trainable parameters in FSL. The output of the FSL is a $1\times 80$ dimensional vector which is input to the fully connected layer (FCL) with 5000 neurons. Finally, an output layer with softmax activation function gives the classification result. Moreover, we use uniform initialization for fully connected layers, and normal initialization for convolution layers.

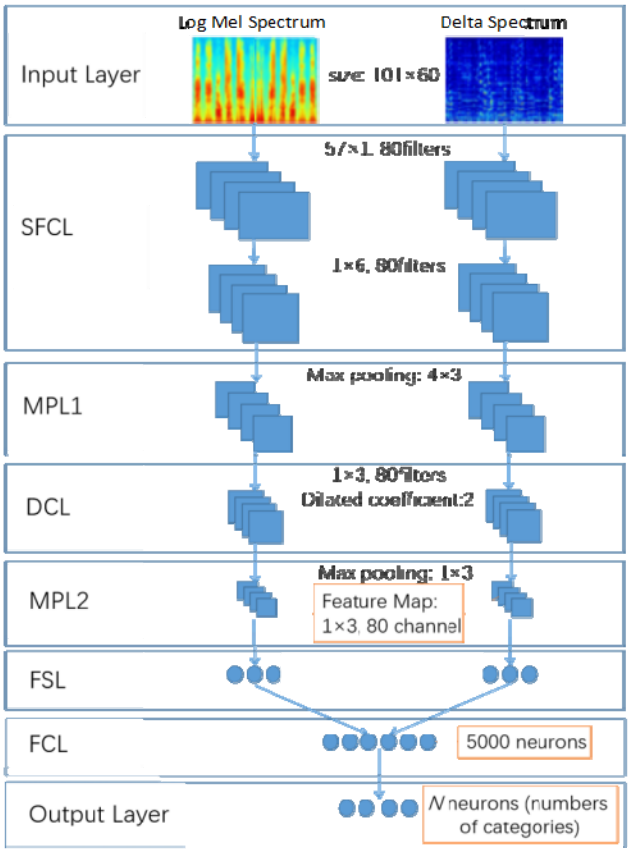


Fig 1. Architecture of the LD-CNN model.

In the following subsection, we will introduce the details of our LD-CNN model presented in Fig 1.

### *A. The Spatial Factorization Convolution Layer (SFCL)*

The structures of the traditional convolution layer and our proposed spatial factorization convolution layer (SFCL) are shown in Fig. 2.

From Fig. 2, it can be seen that our designed SFCL is a factorization of the traditional convolution layer (CL). Here, we denote $L$ and $W$ as the length and width of a convolution filter respectively, and $Nc$ as the number of convolution filters in a convolution layer.

A commonly used CL usually contains $N_c$ filters with $L\times W$ size. However, in our SFCL, the traditional convolution layer is factorized into two separable convolution layers as shown in Fig. 2 (b). The first layer contains $N_c$ filters with $L\times 1$ size and the second layer contains $N_c$ filters with $1\times W$ size.

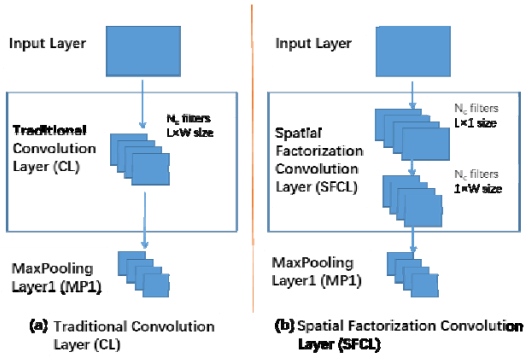


Fig 2. Illustration of (a) traditional convolution layer and (b) our proposed spatial factorization convolution layer (SFCL).

Specifically, in the CL, the size of parameters $U_1$ is calculated by:

$$U_1 = L \times W \times N_c \tag{1}$$

In the SFCL, the size of parameters $U_2$ is calculated by (2):

$$U_2 = L \times 1 \times N_c + 1 \times W \times N_c \tag{2}$$

Comparing (2) with (1), we could easily find that $U_2$ is much smaller than $U_1$.

In our proposed LD-CNN ESC model, $L$ and $W$ are set to 57 and 6 respectively, and $N_c$ is set to 80. Then, we can get the following results: $U_1$=57×6×80=27360 and $U_2$= 57×1×80 + 1×6×80=5040. It is easy to see that $U_1$ is more than 5 times larger than $U_2$.

In addition, in order to intuitively illustrate and compare the ability of using the spatial factorization convolution layer for feature extraction, the feature maps extracted by the traditional convolution layer and the spatial factorization convolution layer are illustrated respectively through the T-SNE visualization tool, which is shown in Fig. 3. From this figure, we can see that the features extracted by the spatial factorization convolution layer have similar distribution to those extracted by the traditional convolution layer, which indirectly verifies that the capability of our proposed spatial factorization convolution layer for feature extraction is maintained if the receptive field obtained by convolution layers remains the same.

*B. The FeatureSum Layer*

Usually, in traditional CNN models, the network parameters are mainly from the two fully connected layers in the higher layers. The structure of a traditional CNN model is shown in Fig. 4 (a). Let us define the following parameters: For feature maps generated from the second max pooling layer (MPL2), $T$ and $R$ are the length and width of each feature map respectively, $N_a$ is the number of feature maps. In addition, we set the number of neurons in the FCL1 and FCL2 to be the same, denoted as $N_{fc}$, and $N_o$ is the number of neurons in the output layer.

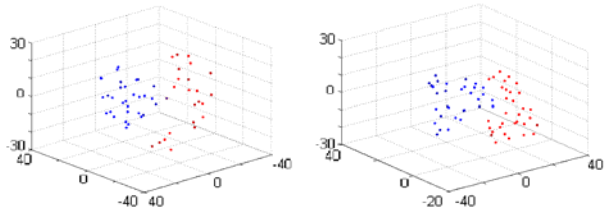

Fig 3. Illustration of the distribution of high-level features extracted by traditional convolutional layer and our proposed SFCL (input audio file consists of 101 frames, two acoustic events: mouse click and keyboard typing. Each point represents one feature vector)

Fig 4 (a) shows that in traditional CNN model, $N_a$ feature maps with $T \times R$ size output from the max pooling layer (MPL2) are directly transformed into the 1-dimensional vector with the size of $1 \times T \times R \times N_a$ through a flatten operation [8] and then fed into the following fully connected layer (FCL1) with $N_{fc}$ neurons. Followed by the FCL1, there is another fully connected layer (FCL2). Therefore, the size of parameters $U_3$ between MPL2 and FCL2 is calculated by (3):

$$U_3 = (1 \times T \times R \times N_a \times N_{fc} + N_{fc}) + (N_{fc} \times N_{fc} + N_{fc}) \tag{3}$$

To reduce the size of parameters caused by the fully connected layers, a FeatureSum layer is proposed to replace the first fully connected layer (FCL1), which is shown in Fig. 4 (b).

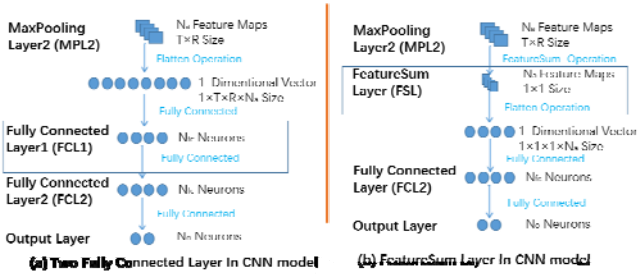


Fig 4. Details of (a) two fully connnected layers used in the traditional CNN model; (b) a featuresum layer and a fully connected layer used in our proposed CNN model.

As shown in Fig. 4 (b), for the $N_a$ feature maps with $T \times R$ size generated from the MPL2, we make a featuresum operation for every feature map. The process of featuresum operation is illustrated in Fig. 5.

For every feature map included in the $N_a$ feature maps, we denote $b_i$ as the sum value of features in the $i$-th feature map and $a_{ix}$ as features in the $i$-th feature map. The featuresum operation calculates the sum value of the features in the $i$-th feature map and output it to the following layers. As environmental sounds have weak absolute locality in the time-frequency spectrogram [7], using a more spatial abstract high-level feature map would have less effect on the final classification accuracy. Therefore, based on this observation, the Featuresum operation calculate the statistic values of features like global pooling, and thus renders more abstract spatial feature maps.


Fig 5. Illustration of the FeatureSum Operation ($a_{ij}$ represents the $j$-th value in the $i$-th feature map, $b_i$ represents the value of the $i$-th neuron in the FeatureSum Layer).

Through the Featuresum operation, the dimension of feature maps has been largely compressed.

As shown in Fig. 4 (b), through replacing the FCL1 with a FeatureSum Layer (FSL), the size of parameters $U_4$ between the MPL2 and the FCL2 is calculated by (4):

$$U_4 = (1 \times 1 \times N_a \times N_{fc}) + N_{fc} \quad (4)$$

Comparing (4) with (3), we could easily find that $U_4$ is much smaller than $U_3$, which means that our featuresum layer can effectively reduce parameters in the CNN model.

In our LD-CNN model, the length $T$ and width $R$ of feature maps generated from MPL2 are 1 and 3 respectively, and the number of feature maps $N_a$ is 80. The number of neurons $N_{fc}$ in each fully connected layer is 5000. Then, we can get the following results: $U_3 = 1 \times 1 \times 3 \times 80 \times 5000 + 5000 + 5000 \times 5000 + 5000 = 26210000$ and $U_4 = 1 \times 1 \times 80 \times 5000 + 5000 = 405000$. It is easy to see that $U_3$ is more than 60 times larger than $U_4$.

In addition, we visualize feature maps by the T-SNE visualization tool to intuitively compare the feature distribution with and without Featuresum layer. Fig. 6 shows that the feature distributions are changed slightly, which implies a similar discriminability for environmental sound classification.

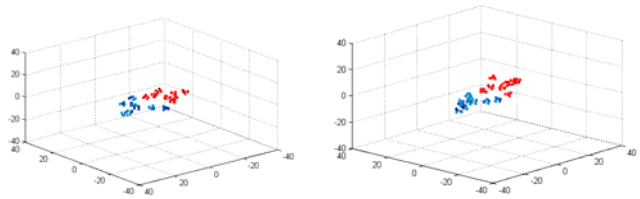

Fig 6. Illustration of features maps output by the fully connected layer with and without the global sum operation (input audio file consists of 101 frames, two acoustic events: mouse click and keyboard typing. Each point represents one feature vector , other model parameters are kept the same).

## III. EXPERIMENTS AND RESULTS

The performance of our proposed LD-CNN is evaluated and compared in this section. . The procedure of LD-CNN ESC system is shown in Fig. 7. In the testing stage, feature extraction module and audio segmentation module are the same as those in the training stage. Key steps are as follows:

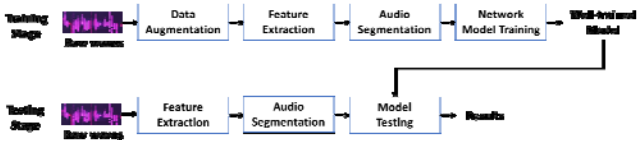


Fig. 7. Procedure of the LD-CNN ESC system

1) Data augmentation module: At the beginning, raw waves of sound event are input into the data augmentation module to increase the size of the datasets. To mitigate the overfitting issue, time-stretch transforming method [5] is used to get slightly faster or slower audio examples.

2) Feature extraction module: We use Hamming window to extract the log-mel spectrum and the delta spectrum from raw wave data, which follows the commonly used method of feature extraction as in [5].

3) Audio segmentation module: Following the method in [5], the whole feature spectrogram of an audio event is split into several segments, which essentially increases the size of training data.

4) Network model training module: All the segments generated from the audio segmentation module are used as input (i.e. mini-batch in turn) to train a suitable LD-CNN model for the ESC task. The SGD method [10] is used to train the LD-CNN network and the batch normalization operation [9] is used in the spatial factorization convolution layer. The learning rate and momentum of training stage is set to 0.01 and 0.9 respectively. In addition, the cross entropy [10] is used as the loss function in the output layer. The key experimental settings are listed in Fig. 1 and more details of the LD-CNN are described in Section Ⅱ.

5) Model testing module: The well-trained LD-CNN model is used to extract high-level feature maps and then classify these extracted features. Finally, in the output layer, the probability voting method is adopted to obtain the average of the posterior class probabilities for all the segments. Then the class with highest average posterior probability is chosen as the output class for this testing.

### *A. Datasets*

In order to evaluate the performance of the proposed LD-CNN, similar to D-CNN [5], we conducted several tests over three public datasets (ESC50, UrbanSound8K, and CICESE). Some statistical information of these three datasets including the split of training/testing datasets, duration time, and number of classes are shown in Table Ⅱ. As there is much difference between the length of an audio file in different datasets, so the size of input feature map is chosen differently (UrbanSound8K: $2 \times 60 \times 31$, ESC50: $2 \times 60 \times 101$, CICESE: $2 \times 60 \times 41$)

TABLE Ⅱ. BASIC INFORMATION OF DATASETS

| **Datasets** | **Classes** | **Train/Test** | **Duration** | **Content** |
|---|---|---|---|---|
| UrbanSound8K | 10 | 90%/10% | 9.7 hours | Surrounding sounds |
| ESC50 | 50 | 80%/20% | 2.8 hours | Life sounds |
| CICESE | 7 | 75%/25% | 14 min | Indoor sounds |

### *B.Experimental Comparison and Analysis*

#### *1) Comparison with the state-of-the-art ESC methods*

We compare the classification accuracy and model size of the proposed LD-CNN with several state-of-the-art ESC methods. The results are shown in Table Ⅲ. It is clear to see that the size of the LD-CNN model is over about 50 times smaller than other state-of-the-art methods, while it nearly retains comparable classification accuracy on three datasets,

which demonstrates that the LD-CNN offers a good tradeoff between the performance and the model size.

TABLE Ⅲ. COMPARISON WITH THE STATE-OF-THE-ART ESC METHODS

| ESC system | UrbanSound8K | ESC50 | CICESE | Network Size |
|---|---|---|---|---|
| TF-CNN[1] | - | 55% | - | - |
| Very Deep CNNs [2] | 72% | 48.4% | - | 128M |
| PiczakCNN [4] | 80.3% | 64.5% | 81% | 105M |
| D-CNN [5] | 81% | 68.5% | 87.1% | 105.3M |
| LD-CNN (ours) | 79% | 66% | 86% | 2.05M |

*2) Comparison of different lightweight networks*

Next, for ESC task, we compare the proposed LD-CNN with two general lightweight networks (Fully-CNN [11], DenseNet ESC [13]) and three other lightweight neural networks based on D-CNN through using different network compression operation (pruning-X [14], Depthwise Separable convolution [15], and LZ Coding [12]). Here, pruning-X means that 5000 neurons in each fully connected layer in D-CNN are pruned into X according to the pruning method in [14]. The experimental results are shown in Table Ⅳ.

TABLE Ⅳ. COMPARISON OF DIFFERENT LIGHTWEIGHT NETWORKS

| ESC system | UrbanSound8K | ESC50 | CICESE | Network Size |
|---|---|---|---|---|
| Fully-CNN [11] | 72% | 60.8% | 88% | 16.7M |
| LZ Coding [12] | 81% | 68.5% | 87.1% | 93M |
| DenseNet [13] | - | 65.7% | 81% | 390.3KB |
| pruning-2000[14] | 80.3% | 64% | 85.7% | 18.3M |
| pruning-1000[14] | 79% | 62% | 82.9% | 5.3M |
| DepthWise [15] | 80% | 67% | 87.6% | 103M |
| LD-CNN (ours) | 79% | 66% | 86% | 2.05M |

From Table Ⅳ, we can see that, compared with Fully-CNN, our LD-CNN has smaller model size better or comparable classification accuracy. The main reason is that Fully-CNN uses many large convolution filters in every convolution layer, usually every layer with 1024 or 2048 filters, which induce a high-computational complexity. Compared with DenseNet, our LD-CNN has larger model size but higher classification accuracy. The main reason is that DenseNet has no fully connected layers which gives smaller model size but less feature representation ability, especially for sound events with complex conditions (CICESE). Compared with DepthWise and LZ Coding methods, our LD-CNN has much smaller model size. Obviously, DepthWise method only compresses parameters in convolution layer while retaining the same parameters in the fully connected layer. And LZ Coding method only compresses the storage of weight files on disk, which is less likely to achieve a very high compression ratio. Compared with the pruning-X compression methods, the proposed LD-CNN performs better both on the classification accuracy and the compression ability. The main reason is that the pruning-X method directly drops all relatively small weights in the fully connected layer which may cause feature information loss if they are not carefully selected.

*3) Effects of filter size in the Spatial Factorization Convolution Layer*

To evaluate the effect of the filter size (*L* and *W*) in Spatial Factorization Convolution Layer on ESC task, we conduct an experiment on three datasets. Specifically, we vary the length parameter *L* from 37 to 77 at the step size of 10 and vary the width parameter *W* from 4 to 8 at the step size of 1. The experimental results are given in Fig.8. From the experimental results, we have the following observations: 1) The filter size used in Spatial Factorization Convolution Layer has different impact on the ESC classification accuracy over three datasets. The range of variation is about 5% for three datasets. 2) With the increasing of *L* and *W*, the classification accuracy firstly goes up, reaches the maximum value and then drop down. The main reason possible is that a relatively larger filter size has stronger capability of extracting contextual information as well as frequency information in features of sound event. However, even larger filter size may introduce some unhelpful information since zero padding effect in convolutional operation. According to the results obtained in Fig 8, in our experiments, we set filter size *L*=57 and *W*=6, respectively.

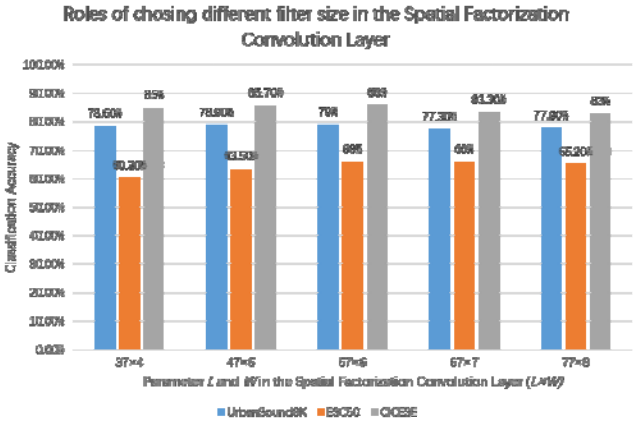


Fig 8. The classification accuracy versus filter size in the Spatial Factorization Convolution Layer.

## IV. CONCLUSIONS

In this paper, we have proposed a lightweight dilated convolutional neural network (LD-CNN) for environmental sound classification task. In LD-CNN, a spatial factorization Convolution Layer and a FeatureSum Layer have been developed to reduce the number of network parameters meanwhile maintain the performance of classification. Compared with the state-of-the-art D-CNN ESC methods and other lightweight networks for ESC task, our proposed LD-CNN ESC system demonstrates competitive classification performance but with a much smaller model size. For example, our LD-CNN model size is about 2.05M, which is 50 times smaller than the original D-CNN model [5], but at the loss of only 1%-2% classification accuracy. Our future research will focus on improving classification accuracy of the lightweight convolutional neural network based ESC system.

## ACKNOWLEDGEMENT

This project was partially supported by Shenzhen Science & Technology Fundamental Research Programs (No: JCYJ20170306165153653 and JCYJ20170817160058246).

## REFERENCES

[1] Huzaifah, Muhammad. "Comparison of Time-Frequency Representations for Environmental Sound Classification using Convolutional Neural Networks." arXiv preprint arXiv:1706.07156 (2017).

[2] Dai, Wei, et al. "Very deep convolutional neural networks for raw waveforms." Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017.

[3] Salamon, Justin, Christopher Jacoby, and Juan Pablo Bello. "A dataset and taxonomy for urban sound research." Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014.

[4] Piczak, Karol J. "Environmental sound classification with convolutional neural networks." Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on. IEEE, 2015.

[5] Zhang Xiaohu, Y. Zou, and W. Shi. "Dilated convolution neural network with LeakyReLU for environmental sound classification." *International Conference on Digital Signal Processing* 2017:1-5.

[6] Szegedy, Christian, et al. "Rethinking the Inception Architecture for Computer Vision." (2015):2818-2826.

[7] Zhang, Haomin, I. Mcloughlin, and Y. Song. "Robust sound event recognition using convolutional neural networks." IEEE International Conference on Acoustics, Speech and Signal Processing IEEE, 2015:559-563.

[8] McLoughlin, Ian, et al. "Robust sound event classification using deep neural networks." IEEE/ACM Transactions on Audio, Speech, and Language Processing 23.3 (2015): 540-552.

[9] Dennis, Jonathan, H. D. Tran, and E. S. Chng. Image Feature Representation of the Subband Power Distribution for Robust Sound Event Classification. IEEE Press, 2013.

[10] Laurent, César, et al. "Batch normalized recurrent neural networks." Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE, 2016.

[11] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[12] Salama, Aly E., and Ahmed H. Khalil. "Design and implementation of FPGA-based systolic array for LZ data compression." Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on. IEEE, 2007.

[13] Iandola, Forrest, et al. "Densenet: Implementing efficient convnet descriptor pyramids." arXiv preprint arXiv:1404.1869 (2014).

[14] He, Tianxing, et al. "Reshaping deep neural network for fast decoding by node-pruning." Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014.

[15] Chollet, François. "Xception: Deep Learning with Depthwise Separable Convolutions." arXiv preprint arXiv:1610.02357 (2016).

[16] Yuan, Zheng Wu, and J. Zhang. "Feature extraction and image retrieval based on AlexNet." Eighth International Conference on Digital Image Processing 2016:100330E.

[17] Mudhsh, Mohammed, et al. "Arabic Handwritten Alphanumeric Character Recognition Using Very Deep Neural Network." Information 8.3(2017):105.