LEARNING FILTER BANKS USING DEEP LEARNING FOR ACOUSTIC SIGNALS

Shuhui Qu, Juncheng Li*, Wei Dai, Samarjit Das*

shuhuiq@stanford.edu, billy.li@us.bosch.com, wdai@cs.cmu.edu, samarjit.das@us.bosch.com

ABSTRACT

Designing appropriate features for acoustic event recognition tasks is an active field of research. Expressive features should both improve the performance of the tasks and also be interpret-able. Currently, heuristically designed features based on the domain knowledge requires tremendous effort in hand-crafting, while features extracted through deep network are difficult for human to interpret. In this work, we explore the experience guided learning method for designing acoustic features. This is a novel hybrid approach combining both domain knowledge and purely data driven feature designing. Based on the procedure of log Mel-filter banks, we design a filter bank learning layer. We concatenate this layer with a convolutional neural network (CNN) model. After training the network, the weight of the filter bank learning layer is extracted to facilitate the design of acoustic features. We smooth the trained weight of the learning layer and re-initialize it in filter bank learning layer as audio feature extractor. For the environmental sound recognition task based on the *Urbansound8K* dataset [1], the experience guided learning leads to a 2% accuracy improvement compared with the fixed feature extractors (the log Mel-filter bank). The shape of the new filter banks are visualized and explained to prove the effectiveness of the feature design process.

Index Terms— filter bank, feature learning, experience guide learning, data driven, neural network

1. INTRODUCTION

In the past few years, the research community has put significant efforts in designing feature representations for acoustic sound recognition. Good features should improve the performance of various audio analytic tasks such as classification and detection. Traditionally, features are heuristically designed, based on the understanding of spectral characteristics of natural sounds. Meanwhile, since this process is separate from the classification process [2], heuristically designed features do not always contain enough information to obtain a high classification accuracy.

Thanks to the development of deep learning methods and rich dataset for sound, deep learning is increasingly becoming a popular candidate for acoustic recognition tasks [3, 4, 5]. Recently, CNN has shown the superior performance in feature extraction and classification in visual [6, 7] and acoustic domain [8], especially in speech recognition [7, 8, 6, 9]. It could not only reduce the dimension of data and but also could extract features as well. However, training a CNN model requires huge computational effort. Therefore, we leverage human experience (i.e. domain knowledge), to design the deep learning model, understand the features from the model, and finally use the learned features to improve the audio recognition tasks' performance.

Currently, most works in sound recognition area use log-mel filter banks as features. These features are not optimized for a particular audio recognition task at hand, and thus might not lead to high accuracy [2]. In this paper, we design our feature extractor by the studying the procedure of designing log-mel filter banks. We build a special filter bank learning layer and concatenate it with a CNN architecture. After training, the weight of the filter bank learning layer is post-processed with human experience. Then, the filter bank learning layer is re-initialized with the processed weight. The weight could be iteratively improved for feature extraction. This process is shown in Fig. 1. We call it as experience guided learning. To our knowledge, this is the first attempt to infuse domain knowledge of feature design to a deep learning pipeline for acoustic recognition tasks.

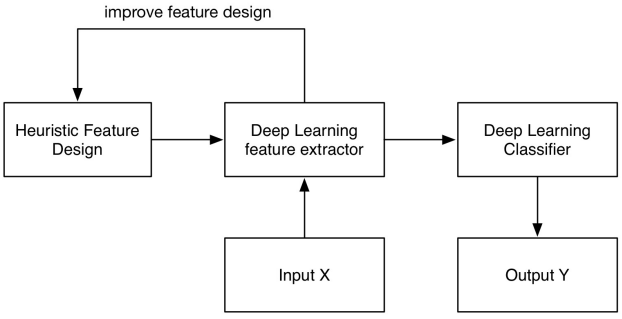

Fig. 1. Experience Guided Learning framework. The weight of the network is initialized by human heuristics. After training, the trained weight is post-processed and used to initialize the extractor. This process would iteratively improves the classification accuracy

By using this method, the accuracy of recognition for the

Urbansound8K sound [1] increases at least 1.5% accuracy based on the human designed filter bank under different settings, such as triangular window for MFCC.

The rest of the paper is structured as follows: Section 2 introduces the related work by using various methods to improve sound recognition tasks. Section 3 describes the special layer, a layer that could extract log-mel features and the CNN architecture in our work in detail. The experimental setup and result are shown in section 4. Finally, conclusion to our work can be found in Section 5.

2. RELATED WORK

There is a wide range of studies related with sound recognition, especially in speech recognition. [10] provided a detailed implementation of the Hidden Markov Model (HMM) on speech recognition by using the Linear predictive coding (LPC) features. [11] applied the HMM model on the MFCC features for speech recognition. With the advancement of deep learning, people applied different deep learning techniques, CNN in particular, for recognition. [12] applied convolutional deep belief networks to audio data and evaluated them on various audio classification tasks by using the MFCC feature. Their feature representations trained from unlabeled audio data showed very good performance. However, the MFCC feature is not generalized and not learned for improving different task objectives. [2] thus proposed a filter learning layer to adaptively learn filter banks from the spectrum, and obtained good result in speech recognition. However, this learning layer is complex (multiple non-linear operations) and requires pre-estimation of the spectrum features' mean and standard deviation. Therefore, in this study, we propose a new filter learning layer based on the procedure of designing log-mel filter banks.

3. FILTER BANK LEARNING LAYER

The mechanism of the filter bank learning layer is similar to the design of log-mel spectrogram [13], which has been widely used in automatic speech recognition. In general, there are several steps to calculate this feature:

1. Perform Fourier Transform to calculate power spectrogram
2. Apply the mel filter banks to all power spectrogram
3. Take the logarithm of all filter banks' energy

Similar to this process, we design the network layer as following: the filter bank learning layer takes power spectrogram of a waveform as input. The layer generates the mel-features by multiplying the filters and individual spectrum. The number of filters is a hyper-parameter that represents the number of features to be learned. After that, we take the logarithm of these features and input into a CNN architecture that has high performance in sound recognition. The filter bank learning layer's weight is not randomly initialized. Similar to triangular window or gamma-tone filter window, each row of the weight is activated once (non-zeros value) within a localized frequency range.

Mathematically, the filter bank learning layer is described by the following equation:

$$m_{i,t} = W_i^T f_t = \sum W_{i,j} f_{j,t} \tag{1}$$

where f_t is the individual power spectrum of the acoustic clip at time t, W_i is the weight of i^{th} filter bank. j represents each individual element. This operation's output is the energy of the filter bank.

Then, we take the logarithm of $m_{i,t}$ to get the log-mel filter coefficient for filter bank i

$$l_{i,t} = log(m_{i,t}) \tag{2}$$

Here, to prevent the log non positive number error, the equation is further developed as:

$$l_{i,t} = log(Relu(m_{i,t}) + \epsilon) \tag{3}$$

where Linear Rectified Units $Relu(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$ and ϵ is a small constant (e.g. $1e-10$). In order to optimize the objective function L, the filter bank learning layer's weight is gradually updated by taking the derivative of the objective function with respect to the weight. The update equation is:

$$W_i = W_i - \alpha \frac{\partial L}{\partial W_i} \tag{4}$$

here, α is the learning rate and L is the loss. By taking the derivative of the weight. The derivative function could be calculated through chain rule:

$$\begin{aligned} \frac{\partial L}{\partial W_i} &= \frac{\partial L}{\partial l_i} \frac{\partial l_i}{\partial m_{i,t}} \frac{m_{i,t}}{W_i} \\ &= \frac{\partial L}{\partial l_i} \mathbf{1}\{m_{i,t} > 0\} f_{i,t} \end{aligned} \tag{5}$$

and here, $\frac{\partial L}{\partial l_i}$ is the loss gradient from previous layers.

The filter bank learning layer could adaptively extract features from the power spectrogram. Combining domain knowledge, the learned filter bank's weight could be further developed into generic filters. Different from [2]'s work, our filter bank learning layer does not require estimating the mean and standard deviation of the input beforehand. Also, our method incurs less computation cost.

4. EXPERIMENT

In this study, the training of the CNN model is performed on the natural sounds dataset , the *Urbansound8k* [1]. This

dataset contains 8732 labeled sound excerpts ($<= 4s$) of urban sounds from 10 classes: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gun shot, jackhammer, siren, and street music. They are evenly divided into 10 folds.

The original sound is at 44.1kHz, we down sample it to 22.5kHz and 8kHz. For the 22.5kHz sound, due to the dimension of raw waveform, we divide it into 1 second each clip (in this case, we use majority voting method to obtain the output). After that, we take the power spectrogram of the sound by using libROSA [14](nfft equals to sampling rate, default hop length). The weight of the filter bank learning layer is initialized by triangular filter banks of MFCC. We build two CNN architectures, one is deep VGG architecture [7] while the other one is shallow as shown in Fig. 4. The parameters are as following. The optimizer is default Adam optimizer [15] with learning rate 0.001. The learning rate decays every three epochs with the decay rate 0.006. The update function is:

$$lr = lr/(1 + decayrate \times epoch) \quad (6)$$

where, lr is the learning rate.

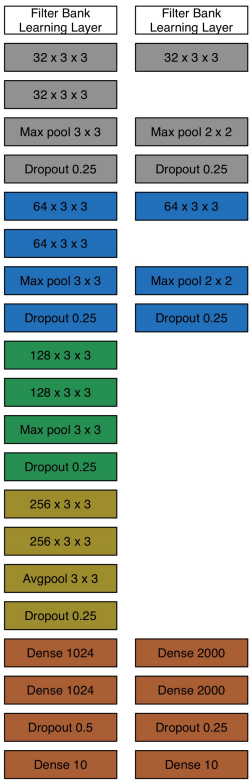

Fig. 2. We designed two architectures. The left model is a deep VGG CNN (arch 1), the right one is shallow architecture(arch 2)

After each layer, we apply leakyrelu[16] with parameter 0.33.

The baseline is around 70% [1] by using svm with rbf kernel and 73.7% [17]. We also test the [2]'s filter bank learning layer for comparison.

Table 1. Result

Win	Arch	n_filt	Weight	Dura	Freq	Acc
T	1	128	Fix	1	22.5	71.88
T	1	128	Trained	1	22.5	72.21
T	1	128	Improved	1	22.5	73.63
T	1	128	Fix	4(MV)	22.5	78.34
T	2	40	Fix	4	8	69.03
T	2	40	Trained	4	8	69.43
T	2	40	Improved	4	8	71.41

Table 2. T means initialized by triangular window; fix means fix the initial weight(in this case, the layer is the same as log-mel feature extractor), train means training with trainable weight, improve means smoothing the weight and reinitialize the filter bank learning layer with this weight. (MV) means post-processing using majority voting method.

5. RESULT AND ANALYSIS

5.1. Experiment Result

The result is shown in Table 2. The proposed method could provide a modest 0.4% of improvement in the classification accuracy. We take out the weight of the filter bank learning layer and use the Savitzky-Golay function [18] to smooth it. We then re-initialize the filter bank layer with the smoothed weight. After retraining the model, the accuracy is improved by 1.5%. We didn't concanate the 4 second clip for the 22.5kHz sound, but we expect improvement compared to the 8kHz result. We also test the filter bank learning layer proposed in [2], but the accuracy is lower than other baselines. This might be caused by the complex non-linearity of this layer and our estimation of input's mean and standard deviation might be too rough. To our knowledge, our method obtains the highest accuracy of *Urbansound8K* dataset.

We also notice that the sampling rate of the sound affect the detection accuracy. For natural sound, different events happen at different frequency levels. Therefore, a relatively high sampling rate is essential for natural sound recognition tasks.

5.2. Filter Bank Analysis

The purpose of this work is to understand the mechanism of filter bank and further facilitate the design of filter banks to generate better feature extractors. Here, we visualize the filter

banks from the triangular window, and smoothed weight from the trained filter bank learning layer that is trained by fold 1-9 in the following picture.

As we can notice, the first few learned filter banks (1st row) conform with the triangle filter banks, which means these triangular filter banks capture most information in low frequency range. However, in the second row, we notice that the learned filter banks are activated around 0.4kHz to 0.5kHz and 0.75kHz to 0.9kHz, while frequency between 0.6 to 0.7 kHz is less interested.

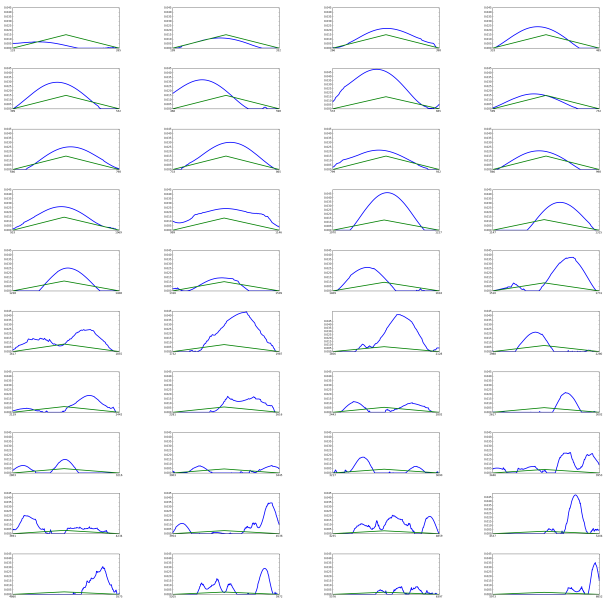

Fig. 3. Filter banks learned from Triangular Window, from top to down, from left to right is the increasing frequency band.

In triangular windows, the bandwidth of filters increases as the frequency level increases. Contrary to this, the learned filters show smaller bandwidth at relatively high frequency area. Fig.3 also shows there are several new peaks within the original single window, which means more filter banks are required. For instance, in the last row, the third picture shows that there are three different frequency ranges that are activated and their bandwidths are relatively small. This information could provide more intuition for audio experts to design new filters.

One problem with these learned filter banks is that they have a lot of serration along the the shape. This is primarily due to the bias of the model. By smoothing the learned filter banks, the model could be generalized, however, more expert experience would be beneficial to improve the recognition accuracy. Here, we apply the Savitzky-Golay function, however, different smooth function might result to different performance. Also, adding some regularization on model's parameters would smooth these filters as well.

6. CONCLUSION

In this paper, we explore the possibility of using the deep learning methods to facilitate the design of filter banks by incorporating human expert knowledge. We first design a filter bank learning layer that takes in frequency features. The output of the layer is fed to two different CNN architectures. This layer is designed according to the design procedure of the log-mel-spectrogram. By taking the weight of the filter bank learning layer, we apply a smooth function on the weight. This gives us at least 1.5% accuracy improvement on the *Urbansound8K* dataset. We further investigate the learned filter banks, and they provide us some intuitions to facilitate the feature design for the recognition task.

7. REFERENCES

[1] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 1041–1044.

[2] Tara N Sainath, Brian Kingsbury, Abdel-rahman Mohamed, and Bhuvana Ramabhadran, "Learning filter banks within a deep neural network framework," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 297–302.

[3] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[4] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[5] George E Dahl, Tara N Sainath, and Geoffrey E Hinton, "Improving deep neural networks for lvcsr using rectified linear units and dropout," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 8609–8613.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[7] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[8] Tara N Sainath, Ron J Weiss, Andrew Senior, Kevin W Wilson, and Oriol Vinyals, "Learning the speech front-end with raw waveform cldnns," in *Proc. Interspeech*, 2015.

[9] Yann LeCun and Yoshua Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, pp. 1995, 1995.

[10] Lawrence R Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[11] Hans-Günter Hirsch and David Pearce, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.

[12] Honglak Lee, Peter Pham, Yan Largman, and Andrew Y Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in neural information processing systems*, 2009, pp. 1096–1104.

[13] Satoshi Imai, "Cepstral analysis synthesis on the mel frequency scale," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'83.* IEEE, 1983, vol. 8, pp. 93–96.

[14] Brian McFee, Matt McVicar, Colin Raffel, Dawen Liang, Oriol Nieto, Eric Battenberg, Josh Moore, Dan Ellis, Ryuichi YAMAMOTO, Rachel Bittner, Douglas Repetto, Petr Viktorin, Joo Felipe Santos, and Adrian Holovaty, "librosa: 0.4.1," Oct. 2015.

[15] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[16] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li, "Empirical evaluation of rectified activations in convolutional network," *CoRR*, vol. abs/1505.00853, 2015.

[17] Karol J Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2015, pp. 1–6.

[18] Ronald W Schafer, "What is a savitzky-golay filter?[lecture notes]," *IEEE Signal Processing Magazine*, vol. 28, no. 4, pp. 111–117, 2011.