URBANSOUND CLASSIFICATION

Wei Wang, Hangquan Zhao, Tianyu Zhao

University of California San Diego, La Jolla, CA 92093-0238

ABSTRACT

In this paper, we propose to explore several deep convolutional neural networks for Urban sound classification (USC) tasks. Our network architecture extracts high-level feature representations from spectrogram-like features (MFCC and Mel spectrogram). Furthermore, we tested traditional machine learning methods and compare their performance with deep learning models. Experiments were conducted on UrbanSound8K. Our experimental results demonstrated that deep learning module (ResNet) has achieved the best performance on UrbanSound8K.

Index Terms**—** Sound Classification, Machine Learning, MFCC, Mel spectrogram, Urbansound8k

1. INTRODUCTION

Over the past five years, developments in artificial intelligence have moved into the medium of sound, whether it be in generating new forms of music (with varying degrees of success), or identifying specific instruments from a video. Sound recognition is a front and center topic in today's pattern recognition theories, which covers a rich variety of fields. Some of sound recognition topics have made remarkable research progress, such as automatic speech recognition (ASR) [1] [2] and music information retrieval (MIR) [3] [4]. Urban sound classification (USC) is an another important branch of sound recognition and is widely applied in surveillance [5], home automation [6], scene analysis [7] and machine hearing [8]. However, unlike speech and music, sound events are more diverse with a wide range of frequencies and often less well defined, which makes Urban Sound Classification tasks more difficult than ASR and MIR. Hence, USC still faces critical design issues in performance and accuracy improvement.

Traditional machine learning methods such as random forest, are usually associated with audio-based machine learning projects, but convolutional neural networks also work well on sound classification. In this paper, we will use neutral networks, together with some helpful audio analysis methods, and compare the performance of traditional machine learning method and convolutional neutral network. In this paper, we will use some neutral network made in pytorch, together with some helpful audio analysis libraries, which can distinguish between 10 different sounds with high accuracy.

2. RELATED WORK

Sound classification has always been an important research topic and traditional machine learning algorithms including KNN, SVM and random forest are the main classification methods. However, they performed poorly in large sound classification datasets including Urbansound8k dataset [9] and ESC dataset [10] proposed in recent years. With the emerge of deep learning in computer vision area, using convolutional neural networks (CNN) has proved to be a promising direction in several researches [11] [12]. It also adopted several useful feature extraction techniques including MFCC and Mel spectrogram.

CNN has become the most popular method in image classification task since its first success in large scale image recognition tasks [13]. CNN has evolved through several generation since then. Important milestones also includes VGGNet [14] and GoogLeNet [15]. ResNet [16] is the latest and most successful CNN architecture so far. Before ResNet, deep networks usually suffer from severe gradient vanishing problem. Kaiming He introduced a residual connection that establish shortcuts between each CNN block to help the gradient to propagate to deeper layers. Because of its stable performance in spatial feature extraction, variations of ResNet are also commonly used in tasks other than image classification. Sound recognition problem can also be considered as a variation of sequence understanding. LSTM [17] and GRU [18] module are most frequent temporal context extractor in sequence problem.

3. DATASET AND FEATURES

3.1. Urbansound8k

We used Urbansound8k, a standard benchmark for sound classification, to evaluate out methods. It contains 8732 sounds excerpts of various lengths. The excerpts are sampled from the 10 classes including air conditioner, car horn, children playing, dog bark, drilling, enginge idling, gun shot, jackhammer, siren, and street music. Visualization of random chosen examples are shown in figure 1. A detailed statistics of Urbansound8k is shown in table 1. We have to use 10-fold cross validation to fully evaluate our methods following the origin split of the dataset. The 10 folds are not as difficult and there are correlations between some folds. Sound clips in different fold might be sampled from a same sound source. Thus, using different data splits will produce false results.

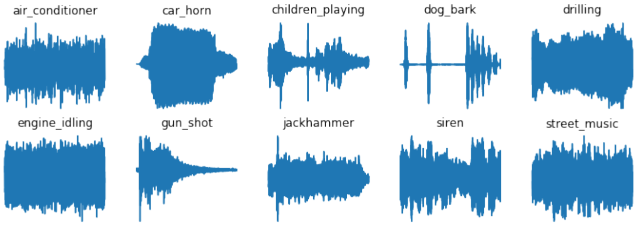


Fig. 1. Audio Samples

categories	ac	ch	cp	db	dr
samples	1000	429	1000	1000	1000
categories	ei	gs	jh	si	sm
samples	1000	374	1000	929	1000

Table 1. Data Statistics: ac: air conditioner, ch: car horn, cp: children playing, db: dog bark, dr: drilling, ei: engine idling, gs: gun shot, jh: jackhammer, si: siren, sm: street music

3.2. Feature Extraction

In audio classification, we always transform the audio data from time field into the frequency field for speech processing and then analysis. Here we apply two features which are mostly used and helpful – Mel-Spectrogram and Mel-Frequency Cepstral Coefficient (MFCC). We use the Python library “librosa” to extract them.

3.2.1. Mel-Spectrogram

The Mel-Spectrogram is a Spectrogram with the Mel Scale as its y axis. Usually, the Mel Spectrogram is the result of these four steps: 1. Separate to windows: Sample the input with windows of size n_{fft}. And to sample the next window, make hops of size hop_length each time. 2. Compute FFT (Fast Fourier Transform) for each window to transform from time domain to frequency domain. 3. Generate a Mel scale: Take the entire frequency spectrum, and separate it into n_{mels} evenly spaced frequencies. This Mel Scale is constructed such that sounds of equal distance from each other on the Mel Scale, also “sound” to humans as they are equal in distance from one another. 4. Generate Spectrogram: For each window, decompose the magnitude of the signal into its components, corresponding to the frequencies in the mel scale. Mel-Spectrograms plots of 10 urban sound classes are shown in figure 2.

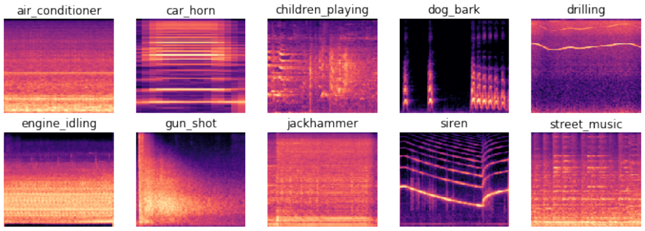


Fig. 2. Mel-Spectrograms plots of 10 urban sound classes

3.2.2. Mel-Frequency Cepstral Coefficient (MFCC)

Mel-Frequency cepstral coefficients (MFCCs) are coefficients that collectively make up an Mel-Frequency Cepstrum (MFC). And the MFC is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. MFCC mimics how human’s hearing system works. [19] Usually, MFCC is the result of these four steps: [20] 1.Take the Fourier transform of (a windowed excerpt of) a signal. 2. Map the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows. 3. Take the logs of the powers at each of the mel frequencies. 4. Take the discrete cosine transform of the list of mel log powers, as if it were a signal. 5. The MFCCs are the amplitudes of the resulting spectrum. MFCCs plots of 10 urban sound classes are shown in figure 3.

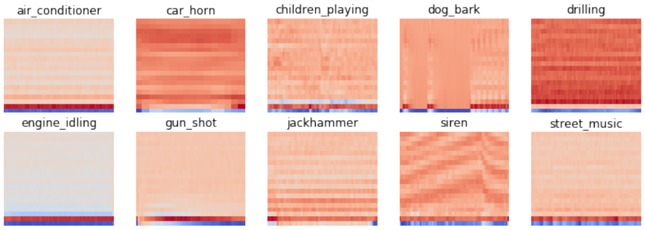


Fig. 3. MFCC plots of 10 urban sound classes

4. METHODS

4.1. CNN Architecture

VGG is a Convolutional Neural Network architecture invented by Visual Geometry Group (Oxford University). [14]There are some VGGn models in which n is the layer number. Compared the results and because of the time limit, we choose VGG11 finally. In figure 4, there shows the architecture of VGG11.

A residual neural network (ResNet) is an artificial neural network (ANN) of a kind that builds on constructs known from pyramidal cells in the cerebral cortex. Residual neural networks do this by utilizing skip connections, or shortcuts

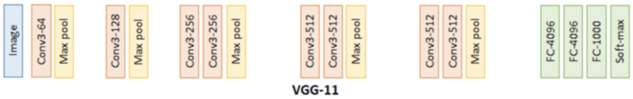


Fig. 4. Experimental Architecture of VGGNet-11

to jump over some layers. Typical ResNet models are implemented with double- or triple- layer skips that contain nonlinearities (ReLU) and batch normalization in between. Models with several parallel skips are referred to as DenseNet. For single skips, the layers may be indexed either as $l-2$ to l or as l to $l+2$. Given a weight matrix $W^{l-1,l}$, for connection weights from layer $l-1$ to l, the forward propagation through the activation function would be as follow:

$$a^l := g(W^{l-1,l}a^{l-1} + b^l + W^{l-2,l}a^{l-2}) \tag{1}$$

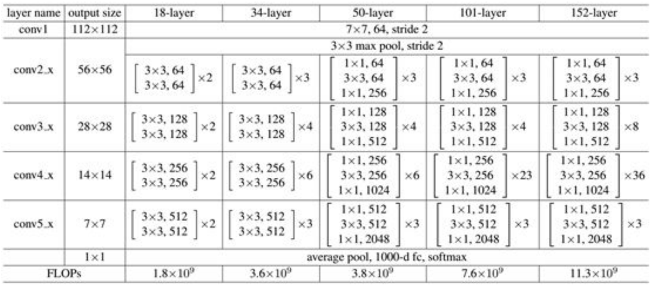
Where a^l is the activation (outputs) of neurons in layer, g is the activation function for layer l, $W^{l-1,l}$ is the weight matrix for neurons between layer $l-1$ and l, $Z^t = W^{l-2,l}a^{l-2}$. The backward propagation is shows as: For normal path:

$$\Delta w^{l-1,l} = -\eta \frac{\partial E^l}{\partial w^{l-1,l}} = -\eta \alpha^{l-1} \delta^l \tag{2}$$

For skip path:

$$\Delta w^{l-2,l} = -\eta \frac{\partial E^l}{\partial w^{l-2,l}} = -\eta \alpha^{l-2} \delta^l \tag{3}$$

Skipping effectively simplifies the network, using fewer layers in the initial training stages. This method speeds learning by reducing the impact of vanishing gradients, as there are fewer layers to propagate through. The configuration of ResNet with different depth is shown in figure 5.



layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Fig. 5. Experimental Architecture of VGGNet-11

4.2. RNN

Both mel-spectrogram and MFCC can be seen as a sequence of consecutive features. We tried to use an LSTM cell in order to capture not only spatial but also temporal patterns. A 2-layer LSTM aims at aggregating the feature overtime to the hidden state of the final LSTM cell. The state is considered as the feature and feed to a fully connected classifier.

4.3. Traditional Methods

Although there are some reasons that deep learning methods behave better. For example, traditional machine learning methods experience underfit in high dimensional data and dimension reduction may lose detailed information. But we do try some traditional method to verify its inability. We then use K-Nearest Neighbors Algorithm (KNN), Support Vector Machine (SVM), Random Forest (RF) and XGBoost. We use the Python library "sklearn" [21] to apply them. The results are shown in part V.

5. EXPERIMENTS

5.1. Preprocessing

We use a 22.05kHz sampling rate for UrbanSound8K datasets. Different preprocessing is applied depending on desired input feature for deep learning models. We used two spectrogram-like representations, Mel spectrogram and Mel-frequency cepstral coefficients(MFCC) as described in section 3.2. When using RGB image as input, we apply random crop for data augmentation and normalization for projecting them to perferable input distribution.

5.2. Training settings

All deep learning models are trained with batch size of 32. We used a learning rate decay schedule with an initial learning rate of 0.001. Then the learning rate will decrease with a factor $\gamma = 0.1$ every 20 epoch of UrbanSound8K. The models are trained for 100 epochs for UrbanSound8K. We use cross entropy as the loss function, which is typically used for multi-classification task. In the testing stage, feature extraction and normalization remains the same. Random crop will not be applied to testing data. Model outputs are projected to probability space by Softmax function. The classification performance of the methods is evaluated by the 10-fold cross validation.

5.3. Results

As presented in 4, we used the data split of fold 3 to briefly evaluate traditional machine learning methods including SVM, KNN, Random Forest and XGBoost. Based on our testing result, in general the performance is ranked as XGBoost > Random Forest > KNN > SVM. MFCC is the better feature for KNN, Random Forest and XGBoost, while Mel spectrogram is most suitable for SVM. MFCC feature ($n_{mfcc} = 80$) with XGBoost achieves the best testing accuracy, 51.03%, out of all possible combinations of features and methods. In comparison of this, the third column of table 2 presents the performance of ResNet18 using the same data.

We did a comprehensive evaluation on the performance of CNN models. Figure 6 and 7 shows the difference be-

Folds (Resnet18)	1	2	3	4	5	6	7	8	9	10	mean
Train, spec	99.68	99.66	99.63	99.50	99.74	99.66	99.72	99.68	99.52	99.47	99.63
Test, spec	73.77	79.17	70.80	76.06	84.08	78.86	71.48	66.38	78.19	83.39	76.22
Train, mfcc	99.41	99.22	90.43	99.24	98.77	99.18	99.52	99.35	99.42	99.04	98.36
Test, mfcc	62.31	53.38	53.3	58.89	62.29	62.58	53.58	54.09	62.99	55.2	57.86

Table 2. 10-fold experiments of ResNet18

feature	mel spectrogram			
model	VGG11	ResNet18	ResNet34	ResNet50
test accuracy	73.10	**76.22**	74.31	75.63

Table 3. spec model compare

	SVM	KNN	Random Forest	XGBoost
Mel spec	21.08	35.14	40.43	42.05
MFCC	10.92	37.73	52.00	51.03

Table 4. Traditional Methods

tween folds which shows the value of 10-fold cross validation. Table 2 gives an example of our results under 10-fold metric. Table 3 and Table 5 summarized our evaluation accuracy of VGG11, ResNet18, ResNet34 and ResNet50 with MFCC and Mel spectrogram feature. In our experiments, Mel spectrogram with ResNet18 achieved the best testing accuracy of 76.22%. When using CNN models, Mel spectrogram outperformed MFCC by a large margin.

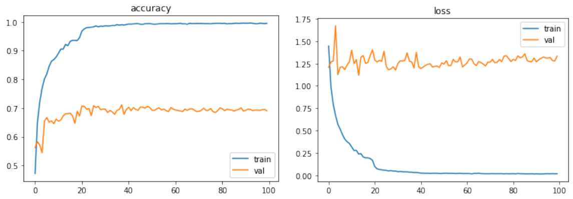


Fig. 6. Accuracy and Loss in fold 3

feature	mfcc			
model	VGG11	ResNet18	ResNet34	ResNet50
test accuracy	57.33	**57.86**	56.60	57.71

Table 5. mfcc model compare

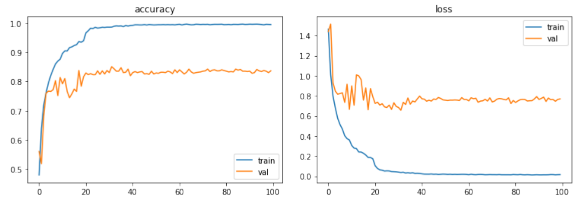


Fig. 7. Accuracy and Loss in fold 10

We also tested the performance of RNN, DNN model but they both had unsatisfying performance of around 50% and 35% respectively. To explore the underlying cause and fact of our results we investigate in to number of parameters in these 6 deep learning models that we tested. As presented in 6, ResNet18 has 5× more parameters than LSTM model while LSTM model has 10× more parameters than DNN. Thus the poor performance of DNN and LSTM model is due to the lack of representational abilities. In other words, the models experience underfit when training such a large dataset. VGG11 has 10× more parameters than ResNet18, but as mentioned before, network degrading problem and gradient vanishing problem make the parameters less efficient compared with ResNet. ResNet34 and ResNet50 both have 2× parameters than ResNet18. But these deep models have stronger representational ablities we can observe overfitting in the training. ResNet50 has more filter but similar total number of parameters, thus, it outperformed ResNet34 in this task.

model	VGG11	ResNet18	ResNet34	ResNet50	LSTM	DNN
param	129M	11M	21M	23M	265k	27k

Table 6. Parameters

6. CONCLUSION

In this project, we compared several neutral network as well as different feature extraction. Results showed that ResNet18 always has the best performance among deep learning methods. Just as our expected, the results of traditional machine learning methods are unsatisfying. Our evaluation proved that extracting Mel spectrogram of sounds and converting the sound recognition to image recognitiaon problem is the most promising direction. However, our solution is far from perfect although we can achieve an average evaluation accuracy of 76.22% in 10-fold cross validation. ResNet18 is not powerful enough to reach high accuracy but deeper model starts to overfit, given that Urbansound8k have a relatively small scale.

For future work, firstly, we would design a task-specific model since ResNet is adaptive feature extractor but may not be suitable for sound classification task. Secondly, we can apply some data augmentation approaches on Urbansound8k to prevent deeper networks from overfitting. Thirdly, we also want to explore other feature such as root-mean square (RMS) level, spectral centroid, bandwidth and so on. Furthermore, de-noising can be useful in preprocessing stage.

7. REFERENCES

[1] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

[2] 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*, 29(6):82–97, 2012.

[3] Michael A Casey, Remco Veltkamp, Masataka Goto, Marc Leman, Christophe Rhodes, and Malcolm Slaney. Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, 96(4):668–696, 2008.

[4] Rainer Typke, Frans Wiering, and Remco C Veltkamp. A survey of music information retrieval systems. In *Proc. 6th international conference on music information retrieval*, pages 153–160. Queen Mary, University of London, 2005.

[5] Regunathan Radhakrishnan, Ajay Divakaran, and A Smaragdis. Audio analysis for surveillance applications. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005.*, pages 158–161. IEEE, 2005.

[6] Michel Vacher, Jean-François Serignat, and Stephane Chaillol. Sound classification in a smart room environment: an approach using gmm and hmm methods. 2007.

[7] Daniele Barchiesi, Dimitrios Giannoulis, Dan Stowell, and Mark D Plumbley. Acoustic scene classification: Classifying environments from the sounds they produce. *IEEE Signal Processing Magazine*, 32(3):16–34, 2015.

[8] Richard F Lyon. Machine hearing: An emerging field [exploratory dsp]. *IEEE signal processing magazine*, 27(5):131–139, 2010.

[9] 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*, pages 1041–1044, 2014.

[10] Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018, 2015.

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

[12] Justin Salamon and Juan Pablo Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3):279–283, 2017.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

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

[15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[18] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[19] Hasan Muaidi, Ayat Al-Ahmad, Thaer Khdoor, Shihadeh Alqrainy, and Mahmud Alkoffash. Arabic audio news retrieval system using dependent speaker mode, mel frequency cepstral coefficient and dynamic time warping techniques. *Research Journal of Applied Sciences, Engineering and Technology*, 7(24):5082–5097, 2014.

[20] Md Sahidullah and Goutam Saha. Design, analysis and experimental evaluation of block based transformation in mfcc computation for speaker recognition. *Speech communication*, 54(4):543–565, 2012.

[21] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel,

Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

A. INDIVIDUAL CONTRIBUTION

Wei Wang: For theory foundation, I took part in every discussion like theme choosing and so on. For code part, I developed lower dimensional features extraction, all the traditional methods and vgg11 method. Also, I participated in some experiments of Resnet34 and Resnet50. As for writing, I helped with PPT presentation and final report about what I did (about 1/3 of all work).

Tianyu Zhao: In theory part, I researched into feature extraction methods especially MFCC. I contribute to the development of dataset API and CNN, RNN training framework. For experiments, I tuned the hyper-parameter, evaluate the performance of ResNet18 and etc. I also participate in presentation and final report writing. (about 1/3 of all work).

Hangquan Zhao: I participate in every project selection, discussion and design. In the coding part, I respond for the ResNet18, 34, 50 part. I construct the training and testing network and make data collection of the result. Also, I make some contribution to the VGG and Mel Spectrum feature extraction part. In the writing part, I take part in all the process of PPT presentation, code reviewing and the final report(nearly 1/3 of all work).

B. REPLY TO REVIEW

Critical review from team 12:
Question: You should cite all references you used images from.
Response: We have deleted some unnecessary images and cited references for others.
QuestionInstead of only top-1 accuracy, you can try more metrics like top-3 accuracy.
Response: Since we just have 10 classes, although top-3 accuracy might be larger, but top-1 accuracy is more accurate and meaningful for comparison.
QuestionYou should explain MFCC and MEL since I think not everyone is familiar with signal processing.
Response: They are now explained on the “feature extraction” part of the report.
QuestionYou should provide more detailed observations instead of just going through which models perform better, and list them down.
Questione.g. Why do you think resnet 50 performs worse than resnet 18 and 34?
Response: According to our further experiments, ResNet50 outperformed ResNet34 but still can’t exceed the accuracy achieved by ResNet18. Possible reasons are summarized in the last paragraph of section 5.3.
QuestionIs 10-fold cross validation necessary? Maybe 4-fold or 3-fold is enough?
Response: In the dataset website “https://urbansounddataset.weebly.com/urbansound8k.html”, it says “Use the predefined 10 folds and perform 10-fold (not 5-fold) cross validation”. The reason is that our results will NOT be comparable to previous results in the literature, meaning any claims to an improvement on previous research will be invalid.
QuestionReferences should also include the papers of xgboost, knn, and svm, random forest, related works you presented, mel-frequency spectrogram and MFCC.
Response: We have deleted some unnecessary introductions for the traditional methods and cited references for the left part.

Critical review from team 24:
QuestionSome of the slides seem to be crowded. I suggest you could either delete some images that are duplicated in context, or put some content on a new slide.
Response: When doing the presentation, we followed the requirements strictly on canvas. Because of the page limitation and lots of contents. That makes some of the slides crowded. Sorry for your inconvenience when reading.
QuestionIn the “Why deep learning” slide, I see the need to use state-of-the-art vision model like VGG. Could you please explain why such models could outperform the traditional methods even if the overfitting still exists?
Response: In this slide, we explained that it is because traditional methods can’t do well on higher dimensional data like images and reduce the dimension will cause data information loss. And from the experiments, there are always overfitting problems which we must admit. For example, the accuracy is 99%on the training dataset and 75% on the test dataset. But compared with the all the experiments with dataset “UrbanSound8K” on the Internet, our results are among the best ones which means overfitting problem is inevitable. Maybe you can find some results online reached 99% accuracy also on the test dataset. But after we read the original papers, we found either they used the wrong methods or faked the results.
QuestionI can see the project is kind of results-oriented. But I think it’s better if you could explain the difference between the results of different combinations. Why Mel-Spectrum and ResNet is much better than the others you tried?

Response: The explanations are in the results part of the report.
QuestionOn a personal note, I would have the "Results/Observations" between "Traditional Methods" and "Deep Learning Methods" to be first of this parts. Because it seems that you first try to show deep learning is much better than traditional ways and then tried to find the best one among the deep learning methods.
Response: Yes, we followed strictly the requirements on canvas so the contents are ordered also following the requirements. Sorry for the inconvenience when reading.

Critical review from team 42:
QuestionFor the background part, did not clearly explain what you have done. I expected a brief-intro like stuff rather than what the outcome could be used for.
Response: The improvement background part is shown in the report.
QuestionIt is unclear about what this project actually did, stated the superiority of deep leaning before things started while seems still used some traditional methods later. Concludes after comparison maybe comes better.
Response: Yes, it's better that concludes after comparison. But when presentation, we followed strictly the requirements on canvas. It let us to talk "Why Deep Learning Methods" first. Sorry for the inconvenience when reading.
QuestionImages quoted lack of consistency, some with label while some are not.
Response: This has been fixed in the report.
QuestionFor the relate works part, explanation is too brief, did not explain the figures shown on the slides.
Response: Related works are well explained then in the report.
QuestionFor the feature extraction part, there are some raw codes for dimensionality reduction, which seems unnecessary.
Response: Those codes have been removed in the report.
QuestionFor the details on used models, the frames of CNN models are too small to read.
Response: This has been fixed in the report.